



**UNIVERSIDAD NACIONAL AUTÓNOMA DE
MÉXICO**

ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES

CAMPUS ARAGÓN

**“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE
TRANSFERENCIA DE DEPÓSITOS A LA VISTA EN
DOLARES”**

T E S I S

**QUE PARA OBTENER EL TÍTULO DE
INGENIERO EN COMPUTACIÓN**

P R E S E N T A :

ALICIA ISLAS GARCÍA

**ASESOR DE TESIS:
M. EN I. JUAN CARLOS ROA BEIZA**

México.

2000





Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

A DIOS GRACIAS, POR PERMITIRME TERMINAR MIS ESTUDIOS, Y EL
DESARROLLO DE ESTE TRABAJO

A MIS PADRES

POR EL ESFUERZO QUE HICIERON POR BRINDARME UNA CARRERA, SIENDO
ESTA
LA MEJOR HERENCIA QUE ME DEJAN.

A MIS HERMANOS:

POR TODO SU APOYO INCONDICIONAL

ESPECIALMENTE:

A TI MARTÍN POR TU AYUDA, COMPRENSIÓN Y DEDICACIÓN, QUE
AYUDARON A TENER UN BUEN TERMINO DE ESTE TRABAJO.

CAPITULO 1. CONCEPTOS BÁSICOS DEL NEGOCIO

1.1 Conceptos Básicos sobre Cuentas de Cheques en Dólares

- 1.1.1 Que es una Chequera
- 1.1.2 Que es una Administración de Chequera
- 1.1.3 Que es una Persona Física
- 1.1.4 Que es una Persona Moral
- 1.1.5 Normatividad de Cuentas de Cheques en Dólares
 - 1.1.5.1 Banco de México
 - 1.1.5.2 SHCP
- 1.1.6 Que es el Lavado de Dinero
 - 1.1.6.1 Como Evitar el Lavado de Dinero
 - 1.1.6.2 Como detectar el Lavado de Dinero
- 1.1.7 Contabilidad y Finanzas
 - 1.1.7.1 Contabilidad
 - 1.1.7.2 Finanzas

1.2 Conceptos Básicos de Contabilidad Financiera

- 1.2.1 Origen y Definición
- 1.2.2 Importancia y Utilidad
- 1.2.3 Principios de Contabilidad Generalmente Aceptados
- 1.2.4 Registro de Operaciones
- 1.2.5 Ecuación Básica de la Contabilidad
 - 1.2.5.1 Definición y Clasificación del Activo
 - 1.2.5.2 Definición y Clasificación del Pasivo
 - 1.2.5.3 Definición de Capital Contable
- 1.2.6 Estados Financieros
 - 1.2.6.1 Objetivo y Utilidad
 - 1.2.6.2 Balance General
 - 1.2.6.3 Estado de Resultados

CAPITULO 2. MARCO TEÓRICO Y HERRAMIENTAS

2.1 Teoría de Bases de Datos Relacionales

- 2.1.1 Concepto de Base de Datos
- 2.1.2 Modelo de Datos
- 2.1.3 Modelo de Bases de Datos Relacionales
- 2.1.4 Dinámica del Modelo Relacional

2.2 Metodología de Diseño de Bases de Datos Relacionales

- 2.2.1 Transformación del Esquema Conceptual al Relacional
- 2.2.2 Normalización

2.3 Sistemas Operativos

- 2.3.1 Windows 9x / Windows NT (Server y Workstation)
 - 2.3.1.1 Windows 98
 - 2.3.1.2 Windows NT
-

2.4 Tipos de Redes

- 2.4.1 Antecedentes
- 2.4.2 Redes de Computadoras, Tipos y Configuraciones
 - 2.4.2.1 Topología
 - 2.4.2.1.1 Configuración Estrella
 - 2.4.2.1.2 Configuración Anillo
 - 2.4.2.1.3 Configuración Bus
 - 2.4.2.1.4 Configuración Malla
 - 2.4.2.2 Redes LAN,
 - 2.4.2.3 Redes WAN
 - 2.4.2.4 Redes Ethernet
- 2.4.3 Conectividad de redes
 - 2.4.3.1 Repetidores
 - 2.4.3.2 Bridges
 - 2.4.3.3 Routers
 - 2.4.3.4 Brouters
- 2.4.4 Sistemas de Distribución de Redes
 - 2.4.4.1 Gateways
 - 2.4.4.2 Concentradores
 - 2.4.4.3 Concepto de "Switching"

2.5 Protocolos de Comunicación

- 2.5.1 El Modelo OSI
 - 2.5.1.1 Las Capas OSI
 - 2.5.1.2 El Flujo de Datos en el Modelo OSI
 - 2.5.1.3 La terminología de Interfaz Vertical en el Modelo OSI
 - 2.5.1.4 El Estándar 802 de IEEE para los Protocolos de Bajo Nivel
 - 2.5.1.5 La especificación ANSI FDDI
 - 2.5.1.6 Servicios de Transferencia de Datos
 - 2.5.1.7 Modos de Transferencia de Datos
 - 2.5.1.8 Encaminamiento.
 - 2.5.1.9 Relación de los Principales Protocolos con el Modelo OSI
- 2.5.2 TCP/IP
 - 2.5.2.1 Antecedentes
 - 2.5.2.2 Ventajas de usar TCP/IP
 - 2.5.2.3 La Arquitectura TCP/IP
 - 2.5.2.4 La Suite de Protocolos TCP/IP
 - 2.5.2.5 El Núcleo de la tecnología de TCP/IP
 - 2.5.2.6 RFCs
 - 2.5.2.7 Direcciones de IP
 - 2.5.2.8 Encaminamiento de IP

2.6 Características de Progress y Herramientas de Desarrollo 4GL

- 2.6.1 Progress Enterprise RDBMS
 - 2.6.1.1 Soporta Múltiples Sistemas Operativos y Redes
 - 2.6.1.2 Características
 - 2.6.1.3 Protege los datos y proporciona alta disponibilidad
 - 2.6.1.4 Simplifica la administración
 - 2.6.1.5 Soporte de Acceso a Base de Datos Distribuidos
- 2.6.2 Progress Personal RDBMS
- 2.6.3 Especificaciones de Progress RDBMS
- 2.6.4 Componentes de Progress 4GL / RDBMS
 - 2.6.4.1 Progress 4GL
 - 2.6.4.2 Progress ProVision
 - 2.6.4.3 Progress AppServer
 - 2.6.4.4 Progress SmartObjects
 - 2.6.4.5 Replication Toolkit
 - 2.6.4.6 Toolkit
 - 2.6.4.7 Client Networking

- 2.6.4.8 Progress Debugger
- 2.6.4.9 Progress WebSpeed
- 2.6.5 Requerimientos para el uso de Progress
 - 2.6.5.1 Requerimientos Mínimos para ejecutar Progress en Windows NT, Windows 95 y Windows 98
 - 2.6.5.2 Espacio Aproximado Requerido en Disco Duro
- 2.7 Características del "Front-End"**
 - 2.7.1 Conquistando el Iceberg de Client / Server
 - 2.7.2 Herramientas de Desarrollo de Progress
 - 2.7.2.1 Application development Environment (ADE)
 - 2.7.2.2 Fourth Generation Language (4GL)
 - 2.7.2.3 Data Dictionary (Diccionario de Datos)
 - 2.7.2.4 User Interface Builder (UIB)
 - 2.7.2.5 Integración de herramientas de desarrollo de terceros
 - 2.7.2.6 Ayuda en Línea
 - 2.7.2.7 Editor de procedimientos
 - 2.7.2.8 Application Debugger (Depurador de Aplicaciones)
 - 2.7.2.9 "Application Component Environment" (Ambiente de Componentes de Aplicación) ACE
 - 2.7.2.10 Progress SmartObjects
 - 2.7.2.11 Soporte VBX
 - 2.7.2.12 Soporte OLE
 - 2.7.2.13 Roundtable Total Software Management System
 - 2.7.3 Herramientas de Implantación de Progress
 - 2.7.3.1 Progress Results
 - 2.7.3.2 "Report Builder" (Constructor de Reportes)
 - 2.7.3.3 "Developer's Toolkit" (Caja de Herramientas del Desarrollador)
 - 2.7.3.4 "Database Administration" (Administración de Bases de Datos)
 - 2.7.3.5 "Translation Manager" (Manejador de Traducciones)
 - 2.7.3.6 "Visual Translator" (Traductor Visual)
 - 2.7.3.7 La Arquitectura de Progress DataServer
 - 2.7.3.8 Progress AppServers
 - 2.7.3.9 "Data Replication" (Réplica de Datos)
 - 2.7.3.10 Progress Relational Data Base Management System (RDBMS)
 - 2.7.3.11 Progress DataServers
 - 2.7.3.12 Amplio Rango de Portabilidad y Soporte

CAPITULO 3. PLANTEAMIENTO DEL PROBLEMA Y PROPUESTA DE SOLUCIÓN

3.1 Definición del Problema

- 3.1.1 Antecedentes
- 3.1.2 Definición del Problema

3.2 Situación Actual

3.3 Requerimientos del Usuario, Recopilación y Análisis de la Información

- 3.3.1 Especificación Preliminar de los Requerimientos
- 3.3.2 Especificación Detallada de los Requerimientos de Software
 - 3.3.2.1 Reportes a Generar

3.4 Descomposición Funcional

- 3.4.1 Estructura Jerárquica de la Institución
 - 3.4.1.1 Subdirector de Operaciones y Chequeras
 - 3.4.1.2 Subdirector de Documentación y Custodia
 - 3.4.1.3 Subdirector de Contabilidad
 - 3.4.1.4 Subdirección de Sistemas
 - 3.4.1.5 Subdirector de Productos Bancarios

- 3.4.1.6 Gerente de Mesa
- 3.4.1.7 Administrador de Tesorería
- 3.4.1.8 Director de Créditos
- 3.4.1.9 Subdirector de Crédito y Cobranza
- 3.4.2 Diagrama estructural de la Problemática de una Institución Financiera
- 3.4.3 Apertura de una cuenta de cheques

3.5 Opciones de Solución

- 3.5.1 Sistemas Manejadores de Bases de Datos
 - 3.5.1.1 Manejadores de Bases de Datos Relacionales (RDBMS)
- 3.5.2 Análisis de los Posibles RDBMS a Utilizar
 - 3.5.2.1 Oracle
 - 3.5.2.2 Sybase
 - 3.5.2.3 Progress
- 3.5.3 Evaluación
- 3.5.4 Selección del RDBMS

CAPITULO 4. DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA

4.1 Diseño de la Aplicación

- 4.1.1 Diseño Preliminar
- 4.1.2 Diagrama de Contexto y de Flujo de Datos
- 4.1.3 Diagrama de Entidad-Relación
- 4.1.4 Normalización
 - 4.1.4.1 Tabla Q_ESTADISTICAS
 - 4.1.4.2 Descripción de los Campos de Q_ESTADISTICAS
 - 4.1.4.3 Primera Forma Normal
 - 4.1.4.4 Segunda Forma Normal
 - 4.1.4.5 Tercera Forma Normal
- 4.1.5 Diccionario de Datos
 - 4.1.5.1 Introducción
 - 4.1.5.2 Descripción del Reporte del Diccionario de Datos
 - 4.1.5.3 Reporte del Diccionario de Datos

4.2 Diseño y Construcción del "Front-End"

- 4.2.1 Fases para el Desarrollo de Aplicaciones Gráficas con Progress
 - 4.2.1.1 "Text" (Texto)
 - 4.2.1.2 "Windows" (Ventanas)
 - 4.2.1.3 "Buttons" (Botones)
 - 4.2.1.4 "Rectangle" (Rectángulo)
 - 4.2.1.5 "Fill-In" (Campo de Llenado)
 - 4.2.1.6 "Frame" (Marco)
 - 4.2.1.7 "SmartPanel" (Panel Inteligente)
 - 4.2.1.8 "SmartViewer" (Visor Inteligente)
- 4.2.2 Pantallas de la Aplicación
 - 4.2.2.1 Pantalla del Menú inicial
 - 4.2.2.2 Pantalla de ABC de Bancos
 - 4.2.2.3 Pantalla de Consulta de Saldo de Cuenta

4.3 Generación de Código

- 4.3.1 Creación y Ejecución de una Pantalla con SmartObjects
- 4.3.2 Código Fuente Generado para el SmartQuery "q-tpmov.w"
- 4.3.3 Código Fuente Generado para el SmartViewer "v-tpmov.w"

4.4 Diseño y Estrategia de Pruebas

- 4.4.1 Fundamentos de las Pruebas de Software
- 4.4.2 Pasos en la Prueba de los Sistemas de Software

- 4.4.3 Diferentes Tipos de Pruebas
 - 4.4.3.1 Pruebas Unitarias y de Integración
 - 4.4.3.2 Pruebas de Sistema
 - 4.4.3.3 Pruebas de Validación
- 4.4.4 Diseño de Pruebas Unitarias
 - 4.4.4.1 Código 4GL (Generado por Progress y Adicional) de un Módulo
 - 4.4.4.2 Pruebas Unitarias Generales
 - 4.4.4.3 Pruebas de Caja Blanca
- 4.4.5 Diseño de Pruebas de Integración

4.5 Resultado de las Pruebas

- 4.5.1 Pruebas Unitarias
 - 4.5.1.1 Pruebas Unitarias Generales
 - 4.5.1.2 Pruebas de Caja Blanca
 - 4.5.1.3 Pruebas de Caja Blanca para c-valpag.w
 - 4.5.1.4 Pruebas de Caja Blanca para genmovcierre.w
- 4.5.2 Pruebas de Integración
- 4.5.3 Pruebas del Sistema
 - 4.5.3.1 Pruebas de Recuperación
 - 4.5.3.2 Pruebas de Seguridad
 - 4.5.3.3 Pruebas de "Stress" (Resistencia) y de Rendimiento
- 4.5.4 Configuración de los Equipos de Prueba

4.6 Factibilidad Técnica y Operativa

- 4.6.1 Normatividad para Redes bajo Windows NT como Sistema Operativo de Red en la Institución
 - 4.6.1.1 Requerimientos del Servidor
 - 4.6.1.2 Requerimientos para las Estaciones de Trabajo
 - 4.6.1.3 Impresoras
 - 4.6.1.4 Puesta a Punto
 - 4.6.1.5 Instalación
 - 4.6.1.6 Configuración TCP/IP
- 4.6.2 Puesta a punto del servidor del Sistema de Cheques en Dólares
- 4.6.3 Instalación de Progress
- 4.6.4 Mantenimiento
 - 4.6.4.1 Mantenimiento Perfectivo
 - 4.6.4.2 Mantenimiento Preventivo
 - 4.6.4.3 Mantenimiento Adaptativo
 - 4.6.4.4 Mantenimiento correctivo
- 4.6.5 Mecanismos de Respaldo y Protección
- 4.6.6 Factibilidad Operativa
 - 4.6.6.1 Capacitación a usuarios y administrador del sistema
 - 4.6.6.2 Soporte Técnico
- 4.6.7 Análisis del costo y venta del sistema

CONCLUSIONES

BIBLIOGRAFÍA

APÉNDICE A. Banco de México

APÉNDICE B. SHCP y CNByV

APÉNDICE C. Diccionario de Datos

APÉNDICE D. Código 4GL Adicional al Generado por Progress

APÉNDICE E. Manual de Usuario

ANEXO 1. Diagrama Entidad-Relación

100

100

100

100

100

100

100

100

Faint, illegible text covering the right side of the page, possibly bleed-through from the reverse side.

CAPITULO 1

CONCEPTOS BÁSICOS DEL NEGOCIO

1.1 Conceptos Básicos sobre Cuentas de Cheques en Dólares

1.1.1 Que es una Chequera

Introducción

Titulos y Operaciones de Crédito

El sustentante deberá acreditar que conoce la naturaleza jurídica y la forma legal en que circulan los títulos de crédito y valores que manejará en la práctica.

Conceptos de Título de Crédito

Son Títulos de Crédito los documentos necesarios para ejercitar el derecho literal que en ellos se consigna. Una de las principales características de los Títulos de Crédito, es que son documentos ejecutivos, lo cual significa que son suficientes para comprobar a favor de su legítimo titular la existencia de los derechos que el título confiere. En nuestro derecho los títulos de crédito principales son los siguientes:

1. Letra de Cambio.
2. El Pagaré.
3. **El Cheque.**
4. Las Obligaciones.
5. Los Certificados de Participación.
6. Los Certificados de Vivienda.

Características de los Títulos de Crédito

- **Incorporación**

El derecho que se consigna se encuentra inmerso al Título mismo, otorgándole un rango jurídico superior a lo que sería un pedazo de papel, convirtiéndolo en un derecho patrimonial de cobro; es por ello que para ejercitar dicho derecho es necesario la exhibición del Título de Crédito.

- **Literalidad**

Es el elemento que establece los límites de exigencia a los que puede aspirar el titular o beneficiario del documento, el límite más importante del derecho incorporado en el título es la cantidad que podrá ser ejercida en virtud del mismo, ésta se podrá especificar tanto en cifras como en palabras, pero en caso de diferencia, la que prevalece es la que esté escrita en palabras. Si la cantidad estuviere inscrita varias veces, tanto en palabras como en cifras, el documento valdrá, en caso de diferencias, por la suma menor.

- **Autonomía**

Es independiente de la causa que le da origen, es decir el objeto y causa de expedición de un documento es irrelevante respecto de la deuda y obligación de pago en él consignadas.

- **Legitimación**

Quien exhiba el título para exhibir el derecho consignado, deberá acreditar ser el beneficiario legítimo. Ello puede ser porque sea el primero y único beneficiario o porque le haya sido transmitido; en tal caso deberá verificarse la continuidad de los endosos, aún cuando no la autenticidad de los mismos.

- **Circulación**

Elemento importantísimo del título de crédito es su carácter ambulatorio que, desde el punto de vista de su consecuencia comercial, llamaremos circulación, cabe señalar que la Ley General de Títulos y Operaciones de Crédito solo es aplicable a los títulos destinados a circular, y no a los que sirvan sólo para efectos de identificación.

Diversas Clasificaciones de los Títulos de Crédito

- **POR SU FORMA DE CIRCULAR.**

- Al Portador.- Sin beneficiario determinado. La propiedad se acredita con la posesión.
- Nominativo.- Designan expresamente el beneficiario en título. Se transmite por endoso y entrega del título.
- A la Orden.- Los títulos nominativos se entenderán siempre "A la orden", salvo que en el título se inserte la cláusula "No a la Orden" o "No negociable", con ello, el título de crédito ya no circulará.

- **POR SU FORMA DE EMISIÓN.**

- Singulares.- Es creado uno solo en cada acto de emisión (**cheques**).
- Seriales o de Masa.- Se crean en serie (acciones y obligaciones).

- **POR EL TIPO DE EMISOR.**

- Individuales.- Los crea una sola persona física o moral (**cheque**).
- Emitidos por personas morales.- Sólo las personas morales pueden crearlos (acciones y obligaciones).

- Gubernamentales.- Sólo los crea el Gobierno (CETES).
 - Garantizados por Bancos.- Aceptaciones Bancarias y Papel Comercial.
-
- BENEFICIARIOS QUE OTORGAN.
 - Principalmente es en cuanto hace a la seguridad en el pago.

Definición de Cheque

Es un producto que sólo puede ser expedido a cargo de una institución de crédito y en los esqueletos que para ello proporcionará la institución respectiva. Implica una provisión de fondos (depósito) previo.

Requisitos:

- La mención de ser cheque, inserta en el documento.
- Lugar y fecha en que se expide.
- Orden incondicional de pagar una suma determinada de dinero.
- Nombre del librado.
- Lugar de pago.
- Firma del librador.

Intervienen 3 personas:

- Librado: Institución de crédito a cargo de la cual se libra el cheque.
- Librador: Persona que libra el cheque, y
- Beneficiario: El titular del cheque, puede ser nominativo o portador.

Aún cuando se regula y se le da el carácter de Título de Crédito por cuanto a la naturaleza jurídica se le clasifica como INSTRUMENTO DE PAGO en virtud de que en realidad no hay crédito (que implica plazo) para cobro de cheque. Ello se fundamenta en lo siguiente.

El cheque siempre es pagado a la vista. Aún cuando el cheque no haya sido presentado en tiempo, el librado (banco) debe pagarlo mientras tenga fondos suficientes para ello.

La muerte o la incapacidad superveniente del librador, no autorizan al librado para dejar de pagar el cheque.

La alteración de la cantidad por la que el cheque fue expedido, o la falsificación de la firma del librador, no pueden ser invocadas por éste para objetar el pago hecho por el librador, si el librador ha dado lugar a ellas por su culpa o por la de sus factores, representantes o dependientes. Cuando el cheque aparezca extendiendo en esqueletos de los que el librado hubiera proporcionado al librador, éste sólo podrá objetar el pago, si la alteración o la falsificación fueren notarias, o si habiendo

CAPITULO 1. CONCEPTOS BÁSICOS DEL NEGOCIO

perdido el esqueleto o el talonario, hubiere dado aviso oportuno de la pérdida al librado. Todo convenio contrario a lo dispuesto en este artículo es nulo.

El que pague con el cheque un título de crédito mencionándolo así en el cheque, será considerado como depositario del título, mientras el cheque no sea cubierto durante el plazo legal señalado para su presentación. La falta de pago o el pago o pago parcial del título de crédito, y una vez protestado el cheque, el tenedor, tendrá derecho a la restitución del título y al pago de los gastos de cobranza y de protesto del cheque; y previo el protesto correspondiente, podrá ejercitar las acciones que por el título no pagado le competan.

El cheque debe ser presentado para su pago:

- Dentro de los 15 días siguientes a su libramiento si es pagadero en la misma plaza.
- Dentro de un mes siguiente a su libramiento si es pagadero en plaza distinta.
- Dentro de tres meses siguientes a su libramiento si es pagadero en plaza distinta y si fue expedido en el extranjero pero pagaderos en la República Mexicana.

El cheque sólo puede ser revocado, transcurrido el plazo de presentación.

El cheque debe protestarse. La presentación del cheque al banco librado o por conducto de cámara de compensación, surte efectos de protesto.

El beneficiario del cheque admite el pago parcial en su pago, el banco librado está obligado a efectuarlo. Si el cheque es presentado en tiempo y no pagado, el librador debe ganar, además, los daños y perjuicios que no podrán ser menores al 20% del importe del cheque.

Formas Especiales del Cheque

- **CHEQUE CRUZADO.**- El librador o beneficiario puede cruzar el cheque, mediante dos líneas paralelas en el anverso del cheque. En tales condiciones, el cheque cruzado sólo podrá ser cobrado por una institución de crédito. El cruzamiento puede ser:

General.- Entre las líneas no se anota el nombre de la institución de crédito que deba presentarlo al cobro.

Especial.- Cuando entre las líneas se anote el nombre de la institución de crédito que lo presentara al cobro.

El cruzamiento general puede convertirse en especial pero no a la inversa.

- **CHEQUE PARA ABONO EN CUENTA.**- El librador o beneficiario, pueden prohibir que el cheque sea pagado en efectivo mediante la inserción, en el cheque de la leyenda: Para Abono en Cuenta. En tal caso, el banco librado sólo podrá hacer el pago mediante abono a la cuenta del beneficiario que le lleve o le abra.

- **CHEQUE NO NEGOCIABLE.** - El librador o beneficiario pueden impedir que el cheque circule mediante la inserción en el cheque de la cláusula: No Negociable.
- **CHEQUE CERTIFICADO.** - Antes de la emisión de cheque, el librador puede solicitar al banco librado que certifique el importe por el cual se expedirá el cheque.
Una vez certificado el cheque, el banco se convierte en el principal obligado (surte efectos de aceptación). Tienen que ser nominativos y no negociables y la certificación no pueden ser parciales.

El librador puede revocar el cheque, devolviéndolo al banco para su cancelación.

- **CHEQUES DE VIAJERO.** - Son expedidos por el librador a su propio cargo. Son nominativos. Al momento de pagarse, deberá cotejarse la firma lo que se hará en presencia de quien lo pague. Serán pagaderos en cualquier sucursal o corresponsal del librado, cuya lista debe proporcionar. Pueden presentarse en cualquier tiempo para su pago.

1.1.2 Que es una Administración de Chequera

Introducción

Con el objeto de establecer la mecánica funcional para la apertura de nuevas cuentas, modificaciones y consultas a las ya existentes y a fin de proporcionar una base documental para la administración de chequeras, así como salvaguardar la información de los clientes así como los recursos, y los servicios ofrecidos en su apertura de cuenta, a continuación se definen los procedimientos de la Institución Financiera.

Objetivo

Establecer los procedimientos para llevar a cabo la apertura de cuentas Cheques en la Institución Financiera, mediante la infraestructura del Sistema y las políticas establecidas del área de negocio que posee para ello, a través de un soporte documental que muestre las actividades inherentes a estos procesos.

Estos procesos pretenden mantener y establecer los controles necesarios no solo para conocimiento y regulación del proceso de apertura de cuentas, sino para que el personal involucrado se apegue a las disposiciones y políticas establecidas.

Políticas

- El ámbito de aplicación de las políticas y procedimientos contenidos en este documento será exclusivo para la Institución Financiera.
- Para autorizar la asignación de número de cuenta en lo referente a Personas Físicas y Fideicomisos de los centros regionales, los promotores responsables deberán enviar vía fax toda la documentación completa, según el "Manual de

CAPITULO 1. CONCEPTOS BÁSICOS DEL NEGOCIO

Documentos y Formas, Requeridos para la Apertura de Cuentas"; en el caso de Personas Morales deberán enviar por valija interna los documentos comprobatorios que acrediten su existencia y su nacionalidad para que la Gerencia Jurídica del Área Internacional emita dictamen jurídico y posteriormente envíen vía fax la documentación de apertura. En ambos casos, de no cumplirse lo anterior no se asignará número de cuenta.

- Los promotores responsables deberán enviar el mismo día que se asignó el número de cuenta los documentos originales y copias legibles vía valija interna al departamento de Administración y Custodia de la Institución Financiera.
- Los horarios de envío de documentos vía fax de los prospectos de clientes podrán realizarse a cualquier hora del día, sin embargo para la asignación de número de cuenta el Departamento de Administración y Custodia de la Institución Financiera deberá recibirlos antes de las 12:00 hrs.
- El promotor responsable no podrá cerrar operaciones con Central de Cambios referentes al prospecto del cliente para realizar el depósito inicial si no ha recibido por parte del departamento de Administración y Custodia de la Institución Financiera la asignación del número de la cuenta, en caso de hacerlo, el quebranto que se ocasioné será exclusivamente responsabilidad del promotor que lo haya realizado.
- No se podrá dar de alta en el Sistema ningún número de cuenta si no obran en poder del departamento de Administración y Custodia de la Institución Financiera los documentos originales y copias legibles del cliente en referencia.
- No se podrán realizar débitos ni se podrán imprimir estados de cuenta a aquellos números de cuenta que no se hayan dado de alta en el Sistema.
- En caso de que se reciba documentación incompleta, el Departamento de Administración y Custodia archivará la papelería por un plazo máximo de 30 días hábiles, esperando que el Centro Financiero proporcione la documentación faltante. En caso contrario se destruirá la papelería.
- El Departamento de Administración y Custodia se reserva el derecho de retener la documentación o de no proporcionar un número de cuenta, cuando alguna información le resulte sospechosa o incongruente, hasta no hacer la revisión o investigación con los departamentos de Contraloría Normativa y/o Auditoría.

Limitaciones

- No se podrán efectuar operaciones con personas físicas y morales, residentes, ó nacionales de los siguientes países: Estados Unidos de América, Islas Caimán, Cuba, Corea del Norte, Libia, Irak, República Federal de Yugoslavia (Servia y Montenegro), Angola, Irán y de aquellos países restringidos por el FED y la OFAC (salvo las excepciones previstas en este escrito). Así mismo de corporaciones en las cuales los firmantes autorizados sean personas con las nacionalidades antes descritas.

- Para los casos de solicitudes de apertura de cuenta que deseen manejar el producto "cuenta de cheques", de la Institución financiera debe adherirse a las políticas y requerimientos normativos de las autoridades del gobierno de los Estados Unidos de América y del Barclays Bank, PLC (banco a través del cuál se realiza el "Payable Through" o cobro de cheques), las cuales establecen que no se pueden abrir ciertos tipos de cuentas, como son: bancos, compañías cuyo objeto social sea inversión, ahorro o crédito, casas de cambio, casas de bolsa, compañías financieras, compañías de juego o apuestas, y/o cualquier otra institución que su negocio se dedique a la intermediación.
- Además, el Barclays Bank, PLC no establecerá cuentas de individuos y/o corporaciones de las siguientes categorías: individuos que sean miembros activos del gobierno militar así como de ciertos cargos públicos del gobierno federal, estatal o municipal (ejemplos: presidente municipal, gobernador, secretario de hacienda, contralor mayor, etc.), corporaciones y/o entidades cuyo objeto social sea considerado ilegal en los Estados Unidos de América (ejemplos: juego y/o apuestas).
- Por disposiciones internas, no se aceptarán en ningún caso solicitudes de clientes cuya actividad u operaciones se encuentren vinculadas a casas de cambio.

Consideraciones Normativas

- La razón social o denominación de la persona moral de que se trate y el (los) nombre(s) y apellidos completos de cada una de las personas físicas, apoderados legales y/o delegados fiduciarios que firmaran en la cuenta, deberán escribirse de la misma manera en cada uno de los formatos de apertura proporcionados por la Institución Financiera, mismos que deberán concordar con la(s) identificación(es) oficial(es) vigente(s) que se presente(n).
- Las fotocopias de los documentos comprobatorios y/o identificaciones que se requieren para cubrir con los requisitos de apertura de la Cuenta de Cheques, deberán obtenerse de los originales (en su anverso y reverso) en el centro regional, o bien, dichas fotocopias también podrán ser presentadas por las personas físicas, apoderados legales y/o delegados fiduciarios autorizados para manejar la cuenta junto con sus documentos originales.
- En ambos casos, las fotocopias deben ser legibles, debiendo revisar que los documentos originales no presenten alteraciones de ningún tipo, verificando que concuerden con los originales y, en el caso de la identificación verificar que la fotografía y firma correspondan al interesado.
- En señal de haber realizado lo anterior, el ejecutivo responsable de la cuenta deberá escribir al calce de las fotocopias la siguiente leyenda: "Manifiesto bajo protesta de decir verdad que estos documentos son copia fiel y verdadera de los documentos originales los cuales tuve a la vista" y después escribir su nombre, su número de empleado y firmar.

CAPITULO 1. CONCEPTOS BÁSICOS DEL NEGOCIO

Actividades de Administración (Chequeras en Dólares)

- Proceso para la asignación de número de cuenta,

Una vez firmados por el Director General los documentos que componen el expediente del cliente, se procede a dar de alta.

- Proceso para dar de alta la cuenta en el sistema.

1 Alta de documentación recibida "Personas Físicas"

1-A Alta de documentación recibida "Personas Morales"

2 Alta de Información General

3 Alta de Instrucciones de Pago en Cuentas del Grupo Financiero

4 Alta de Instrucciones de Pago en Cuentas de Bancos Externos

- Proceso de envío al cliente de sus documentos de apertura

#1 Cuentas sin uso de Chequera

#2 Cuentas con uso de Chequera

- Proceso de alta y/o modificación de cuentas en Barclays Bank PLC

#1 Cuentas Nuevas

#2 Cuentas Sustituyes

- Proceso de impresión y recepción de chequeras.

#1 Cuentas Nuevas

#2 Cuentas Sustituyes

#3 Solicitudes de chequeras definitivas con dotación de cheques del 31 en adelante.

- Recepción de Chequeras Enviadas por el Proveedor

- Proceso de notificación de firmas a control de cuentas y centros regionales

#1 Cuentas Nuevas

#2 Cuentas Sustituyes

- Proceso de impresión "mensual" de estados de cuenta

- Proceso de distribución de estados de cuenta

- Proceso de actas de destrucción de chequeras

#1 Actas por Cancelación de "Cuenta Cheques"

#2 Actas por "Cheques Físicos Devueltos por el Cliente"

- Actas por Cancelación de "Cuenta de Cheques"
- Proceso de congelación de cuentas
 - #1 Por Robo o Extravío
 - #2 Por Fallecimiento
- Proceso de modificaciones de cuentas abiertas
 - #1 Inclusión o Alta de Firmantes
 - #2 Baja o Cancelación de Firmantes
 - #3 Cambio en el Tipo de Firma
 - #4 Alta del Producto "Cuenta de Cheques"
 - #5 Cancelación del Producto "Cuenta de Cheques"
 - #6 Cambio de Domicilios
- Transferencias de cuenta de Cheques con otros Bancos
 - #1 Solicita Cliente Traspaso al área de Chequeras
 - #2 Verifican Saldo en su Cuenta de Cheques
 - #3 Alta del Banco a transferir depósitos en transferencias
 - #4 Verificación de firma del Cliente
 - #5 Se imprime Ticket "T2" de pago del cliente
 - #6 Débito de cuenta de cheques
 - #7 Envío de transferencia con el FTFC del Chase Bank
 - #8 Débito de la cuenta concentradora del Chase Bank a otro Banco
 - #9 Se genera el movimiento contable de movimiento del cliente
- Pagos de Cheques en Centros Regionales
 - #1 Vía telefónica solicita autorización de pago de cheques
 - #2 El área de chequeras debe recibir la siguiente información para el pago
 - No. Centro Financiero
 - Nombre Funcionario
 - Número de Cuenta
 - Monto
 - Número de Cheques
 - Forma de Pago (EFED = Dólares, EFEM = M.N., CTRA = Depósito)
 - #3 El área chequeras proporciona un listado de los movimientos del día al Departamento de Operaciones y otro Departamento de Giros, Remesas y Cheques de viajero.

CAPITULO 1. CONCEPTOS BÁSICOS DEL NEGOCIO

- #4 El Departamento Giros, Remesa y Cheques de viajero solicita su pago por el cobro de cheques en los centros regionales.
- Cierre del Chase Bank
 - #1 Se imprimen en el Chase Link los saldos de las cuentas concentradoras.
 - #2 Saldos Largos se invierten en Tesorería, Saldos Cortos se solicita a Tesorería.
 - #3 Se verifica que los trasposos fueron efectuados a las cuentas externas.
 - #4 Chase Bank proporciona el número de FED al cual fue realizado el depósito.
- Deposito de Cheques en Chase Bank
 - #1 El departamento de chequeras recibe cheques para su deposito de los diferentes centros regionales.
 - #2 Se verifican cheques para su deposito con las siguientes características: Fecha no mayor a seis meses, monto protegido con letra, banco pagador en EUA y endoso de manera correcta.
 - #3 Sello con deposito de la Institución financiera.
 - #4 Captura de la remesa y liberación de la misma afectando movimientos de las cuentas de cheques con crédito.
 - #5 A partir de seis días naturales el sistema afecta el saldo disponible del cliente.
- Cierre del Barclays Bank
 - #1 Barclays recibe todos los cheques de los clientes de la Institución financiera "salvo buen cobro".
 - #2 Barclays envía vía fax los cobros realizados de los clientes y los devueltos, se afecta cuentas de los clientes.
- Cierre del Mes
 - #1 Se calculan los saldos promedios de los clientes pagando intereses o cobrando por bajo promedio.
 - #2 Se genera la estadística por cliente: Saldos Iniciales, Saldos finales, cheques girados, cheques devueltos, # débitos, # créditos, monto de débitos, monto de créditos, etc.

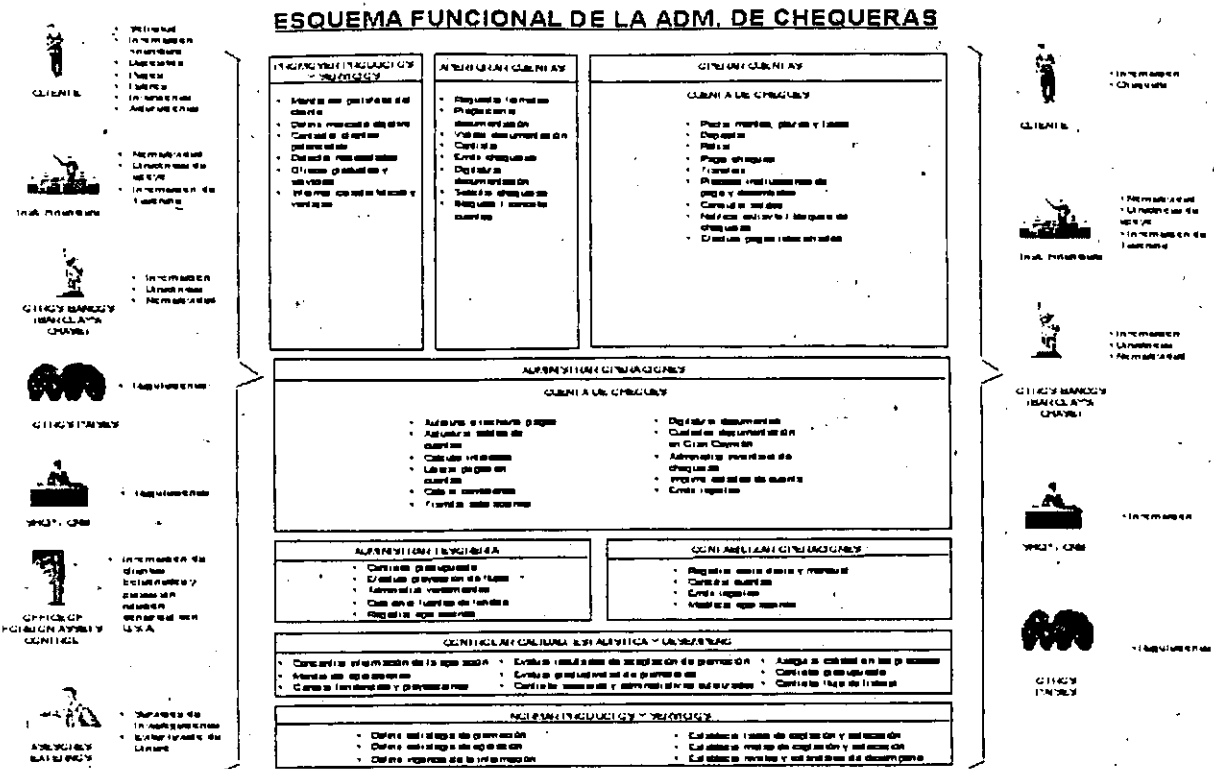


Figura 1.1_1 Esquema funcional de la administración de la Chequera

1.1.3 Que es una Persona Física

La capacidad jurídica de las personas físicas se adquiere por el nacimiento y se pierde por la muerte; pero desde el momento que un individuo es concebido, entra bajo la protección de la Ley y se le tiene por nacido para los efectos declarados en el presente Código (Art. 22 del Código Civil para el Distrito Federal y Territorios).

Las personas físicas, para poder ostentar capacidad jurídica, deben reunir los requisitos de edad, salud y los demás que marcan las leyes respectivas o, en su defecto, ejercer sus derechos y contratar obligaciones por medio de sus representantes (Art. 23 del citado ordenamiento).

1.1.4 Que es una Persona Moral

El artículo 25 del Código Civil mencionado las clasifica de la siguiente manera:

- I.- La Nación, los Estados y los Municipios.

- II.- Las demás corporaciones de carácter público, reconocidas por la Ley.
- III.- Las sociedades civiles o mercantiles.
- IV.- Los sindicatos, las asociaciones profesionales y las demás a que se refiere la Fracción XVI del Artículo 123 de la Constitución Federal.
- V.- Las sociedades cooperativas y mutualistas.
- VI.- Las asociaciones distintas de las enumeradas, que se propagan para fines políticos, científicos, artísticos. De recreo o cualquier otro fin lícito, siempre que no fueren desconocidos por la Ley.

La Fracción II mencionada se debe entender en el sentido de que dichas corporaciones son los llamados organismos descentralizados y los desconcentrados.

La Fracción VI se refiere principalmente a la asociación civil, la cual está regulada por el artículo 2670 del precitado ordenamiento.

El artículo 28 ya mencionado señala que las personas colectivas o morales se regirán por las leyes correspondientes, por su escritura constitutiva y por sus estados.

1.1.5 Normatividad de Cuentas de Cheques en Dólares

1.1.5.1 Banco de México

Para el desarrollo del sistema financiero, entre otros fines, fue creado el Banco de México, como organismo público descentralizado, con personalidad y patrimonio propio. Entre otras funciones, el Banco de México es el asesor financiero del Gobierno Federal y tiene facultades, entre otras, para dictar disposiciones referente al crédito.

Operaciones pasivas con depósitos a la vista en cuenta de cheques pagaderos en la República Mexicana.

- La apertura de cuenta podría ser para personas físicas y morales.
- Las personas físicas deberán estar domiciliadas en una franja de 20 Kilómetros de la línea divisoria con el país del norte y los estados de Baja California y Baja California Sur.
- Las personas morales en cualquier parte del país.
- Representantes oficiales de gobiernos federales.
- Los depósitos deberán ser moneda extranjera (USD) y pagaderos de igual forma.
- Los montos serán libres de acuerdo de la normatividad de la institución financiera y la constitución de la cuenta de cheques.
- Rendimientos los depósitos podrán o no devengar intereses. Las instituciones pactarán las tasas de interés.
- Las instituciones deberán documentar estos depósitos en contratos
- La prohibición para el depositante de ceder los derechos que para él se deriven del contrato respectivo.

- El esqueleto de la expedición de los cheques contendrá, en el anverso, la leyenda siguiente: "Este título se pagará precisamente en dólares de los EUA".
- Los retiros serán a la vista mediante el libramiento de cheques a cargo de oficinas que las instituciones tengan establecidas en la República Mexicana.
- La institución podrá libremente determinar las comisiones que se apliquen en el manejo de la chequera.

Para leyes y normatividad del Banco de México ver el **Apéndice A**.

1.1.5.2 Secretaría de Hacienda y Crédito Público

El sistema financiero, corresponde su regulación e implementación al Poder Ejecutivo Federal quien lo hace a través de la SHCP. De esto se desprende que tanto la legislación como las autoridades encargadas de ello, tienen el carácter de Federal, es decir, son aplicables y tienen competencia en toda la República Mexicana. De igual naturaleza federal es lo relativo a Sociedades Mercantiles y Títulos y Operaciones de Crédito. Como órganos especializados, en un campo específico son creadas las Comisiones Nacional Bancaria y Valores, Seguros y Fianzas.

Dichas comisiones, dependientes de la SHCP, tienen por objeto, en forma general, la inspección y vigilancia de Bancos de Instituciones de Seguros y Fianzas, y Mercado de Valores (CNBV). El sistema financiero, se desarrolla a través de intermediarios, es decir, de personas encargadas de poner en contrato la oferta y la demanda del crédito en general. Dichos intermediarios operan bajo el régimen de concesión o autorización, que discrecionalmente otorga la SHCP, o a veces delega ello en algún organismo especializado.

Los intermediarios pueden ser:

- Bancarios (Banco de México, Instituciones de Banca Múltiple, etc.).
- No Bancarios (Organizaciones Auxiliares de Crédito, Actividades Auxiliares del Crédito, etc.).

La SHCP tiene como objetivo principal presentar leyes para regular agrupaciones financieras, las cuales se presentarán las más importantes.

- Establecer los términos bajo los cuales habrán de operar los Grupos Financieros así como la protección de los intereses de quienes celebren operaciones con los integrantes de dichos grupos.
- Las entidades financieras no deberán utilizar denominaciones iguales a las otras entidades financieras.
- Se requerirá autorización de la SHCP para la constitución y funcionamiento de Grupos Financieros.
- La separación de alguno o algunos de los integrantes de un grupo, así como la disolución; esto deberá ser autorizado por la SHCP, oyendo opiniones de la CNByV, el Banco de México y quien corresponda.
- El sistema Bancario Mexicano estará integrado por el Banco de México, las instituciones de Banca Múltiple, las instituciones de banca de desarrollo, el Patronato del Ahorro Nacional y de fideicomisos públicos.

- El estado ejercerá la rectoría del Sistema Bancario Mexicano, a fin de que éste oriente fundamentalmente sus actividades a apoyar y promover el desarrollo de las fuerzas productivas del país.
- La SHCP podrá autorizar, el establecimiento en el territorio nacional de oficinas de representación de entidades financieras del exterior.
- Las actividades que realicen las oficinas de representación se sujetarán a las reglas expedidas por la SHCP y Banco de México.
- Se crea La Comisión Nacional Bancaria y de Valores como órgano desconcentrado de la SHCP, con autonomía técnica y facultades ejecutivas.
- La CNByV tendrá por objeto supervisar y regular, en el ámbito de su competencia, a las entidades financieras y procurar su correcto funcionamiento.
- La CNByV realiza supervisión de las entidades, así como de las personas físicas y demás personas morales cuando realizan actividades.
- La CNByV dictara normas de registro de operaciones aplicables a las entidades.
- La CNByV fijara reglas para la estimación de los activos.
- La CNByV expedirá normas respecto a la información que deberán proporcionar las entidades financieras.
- La CNByV fungirá como órgano de consulta del Gobierno Federal en materia financiera.
- La CNByV procura a través de los procedimientos establecidos en las leyes que regulan al sistema financiero, que las entidades cumplan debida y eficazmente las operaciones y servicios, en los términos y condiciones concertados, con los usuarios de servicios financieros.
- La CNByV autoriza la constitución y operación, así como determinar el capital mínimo de aquellas entidades.
- La CNByV autoriza o aprueba los nombramientos de consejeros, directivos, comisarios y apoderados de las entidades.
- La CNByV determina o recomienda que se proceda a la amonestación, suspensión, veto o remoción y, en su caso, inhabilitación de los consejeros, directivos, comisarios, delegados fiduciarios, apoderados.
- La CNByV puede intervenir administrativamente o gerencialmente a las entidades, con el objeto de suspender, normalizar o resolver las operaciones que pongan en peligro su solvencia, estabilidad o liquidez.

Para leyes y normatividad del SHCP y CNByV ver el **Apéndice B**.

1.1.6 Que es el Lavado de Dinero

Esta expresión tiene su origen en los años 20's cuando los seguidores de los Gángster de Chicago. Al Capone y "Bugsy" Morgan, usaron lavanderías para "lavar" el dinero obtenido de la extorsión. En los 80's, la Mafia Americana utilizó pizzerías y joyerías. A principio de los 90's los Gángsteres de Córcega en Marsella usaban clínicas de cirugía plástica, donde pacientes adinerados generalmente, pagaban de contado para que les realizaran mejoras estéticas.

El "Lavado de Dinero" hoy en día es sumamente sencillo, debido al uso del sistema financiero para hacer pagos y transferencias de fondos de una cuenta a otra con el fin de ocultar la fuente de los recursos y beneficiar al dueño del dinero.

La expresión es usada generalmente en relación con los ingresos derivados del narcotráfico y del terrorismo, pero puede aplicarse a ingresos provenientes de cualquier acto ilícito.

El ingreso de fondos producto del "Lavado de Dinero" generalmente se localiza en cualquiera de las siguientes tres etapas.

Colocación : El posicionamiento físico del dinero en efectivo en un banco u otra institución dentro del sistema financiero;

Distribución : El movimiento de fondos para crear una compleja transacción, la cual las autoridades tendrán dificultad en rastrear.

Integración : Reingresar el dinero a la economía legítima a través de la compra de artículos o de servicios bancarios.

1.1.6.1 Como Evitar el Lavado de Dinero

La forma más importante para evitar el riesgo de que una Institución Financiera sea involucrada o utilizado como intermediario en operaciones con recursos de procedencia ilícita, es a través del conocimiento del cliente.

A la apertura de cualquier contrato y/o cuenta, se abrirá un expediente de identificación de los clientes (titulares, cotitulares, apoderados y personas facultadas para manejar las cuentas).

Excepciones. La Secretaría de Hacienda y Crédito Público ha autorizado a Instituciones Financieras a no abrir expedientes de identificación en la apertura de algunos productos, que por sus características están enfocados principalmente a sectores de la población de menores recursos.

1.1.6.2 Como Detectar el Lavado de Dinero

Son aquellas operaciones que se realicen en cualquier instrumento monetario por un monto iguales o superiores a \$ 10,000.00 USD, en moneda nacional o en cualquier otra de curso legal. Para efectos del cálculo se considerará el tipo de cambio que publique Banco de México en el Diario Oficial de la Federación, el día hábil bancario inmediato anterior a la fecha en que se realice la operación.

El funcionario o empleado autorizado para recibir y tramitar las operaciones de los clientes, deberá obtener fotocopia de su identificación y comprobante de domicilio, al realizar cualquier operación relevante (excepto depósitos). Si existiera negativa por parte del cliente a proporcionar identificación o no se proporciona en los términos indicados, no se llevará a cabo ninguna operación.

La secretaria de Hacienda y Crédito Público ha autorizado a Instituciones Financieras a no obtener fotocopia de la identificación, cuando las transacciones sean realizadas por clientes (titulares, cotitulares, apoderados y personas facultadas), siempre y cuando se cuente con su expediente de identificación.

CAPITULO 1. CONCEPTOS BÁSICOS DEL NEGOCIO

Por actos y operaciones internas preocupantes se entenderá aquella conducta u operación que realice un empleado, funcionario, miembro del Consejo de Administración o Comisario de la Institución Financiera que pueda considerarse sospechosa.

Los funcionarios con personal a su cargo son responsables de determinar la falta de correspondencia entre el nivel de ingresos de sus subordinados y su estilo de vida, o incongruencias entre las funciones encomendadas y las actividades que lleven a cabo.

Los Reportes de Operaciones Sospechosas y Relevantes derivadas de la aplicación del Manual, se presentarán a la Comisión Nacional Bancaria y de Valores.

Los Reportes de Operaciones Relevantes, se enviarán trimestralmente a más tardar 20 días hábiles posteriores al cierre de operaciones del último mes, a través de medios magnéticos con las especificaciones técnicas establecidas. Los Reportes de Operaciones Sospechosas, se presentarán a más tardar 20 días hábiles posteriores a aquél en que se detecte, en el formato autorizado por la Secretaría de Hacienda y Crédito Público.

Las áreas responsables de la guarda y custodia de expedientes deberán conservar los mismos bajo medidas de seguridad, establecer un espacio físico exclusivo para su conservación, restringir el acceso a terceros, archivar oportunamente la documentación recibida, verificar que la integración de documentación cumpla con la normatividad vigente, controlar su tránsito y mantenerlos completos y actualizados. La figura 1.1_2 muestra como evitar el lavado de dinero, desde la apertura de cuenta.

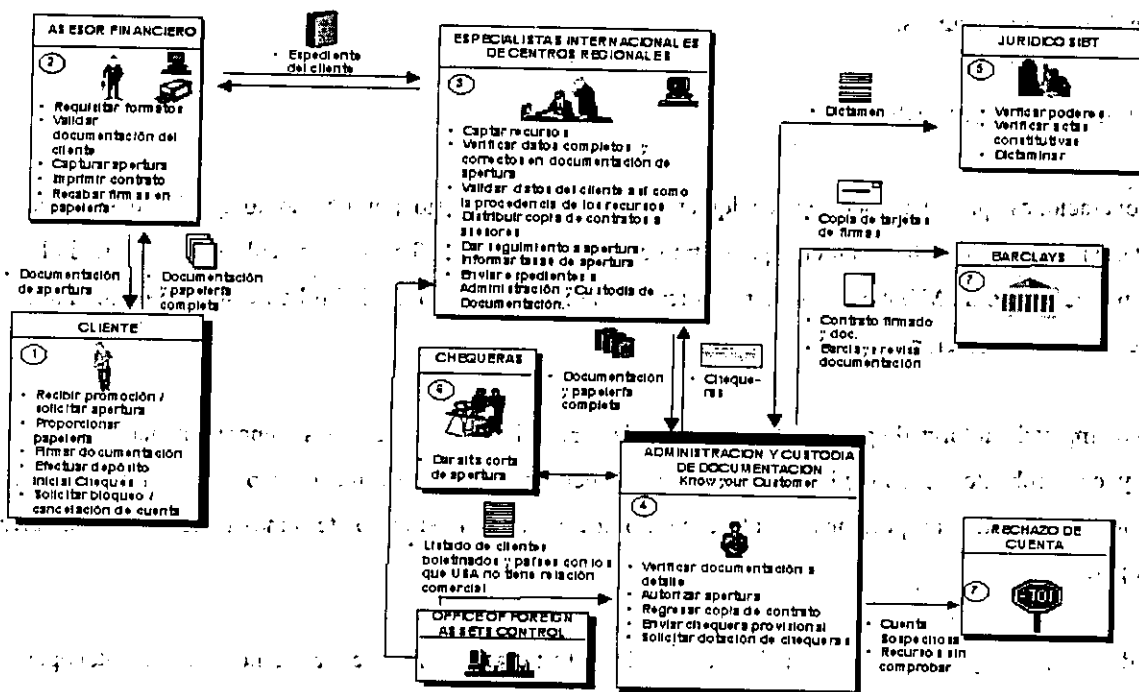


Figura 1.1_2 Cómo Evitar el Lavado de Dinero

1.1.7 Contabilidad y Finanzas

1.1.7.1 Contabilidad

Características de la Información Contable

Los métodos de cuantificación varían en complejidad, dependiendo del grado de dificultad que implique el obtener la información cuantitativa. Cuando la información se deriva de una observación inmediata del fenómeno medido, la tarea es más fácil; Tal es el caso de una simple estadística sobre temperatura, o una sobre ventas. Pero cuando se refiere a los múltiples eventos de una entidad económica, la información cuantitativa resulta compleja.

La definición de contabilidad, como técnica utilizada para producir información cuantitativa que sirva de base para tomar decisiones económicas a los usuarios de la misma, implica que la información y el proceso de cuantificación deben cumplir con una serie de requisitos para que satisfaga adecuadamente las necesidades que mantienen vigente su utilidad.

Las características fundamentales que debe tener la información contable son utilidad y confiabilidad. Además, la información contable tiene, impuesta por su función en la toma de decisiones, la característica de la provisionalidad.

La utilidad como características de la información contable es la cualidad de adecuarse al propósito del usuario. Estos propósitos son diferentes en detalle para cada usuario pero todos tienen la comunidad de interés económico en la entidad económica; entre estos interesados se encuentran: la administración, inversionistas, accionistas, trabajadores, proveedores, acreedores, autoridades gubernamentales, financieras, etc. Dada la imposibilidad de conocer al usuario específico y sus necesidades particulares, se presenta información general por medio de los estados financieros: el balance general, el estado de resultados y el estado de cambios en la situación financiera. La utilidad de la información está en función de su contenido informativo y de su oportunidad.

El contenido informativo está basado en: la significación de la información, es decir, en su capacidad de representar simbólicamente –con palabras y cantidades- la entidad y su evolución, su estado en diferentes puntos en el tiempo y los resultados de su operación; la relevancia de la información, que es la cualidad de seleccionar los elementos de la misma que mejor permitan al usuario captar el mensaje y operar sobre ella para lograr sus fines particulares; la veracidad, cualidad esencial, que abarca la inclusión de eventos realmente sucedidos y de correcta medición de acuerdo con las reglas aceptadas como válidas por el sistema y por último, la comparabilidad, es decir, la cualidad de la información de ser válidamente comparable en los diferentes puntos de tiempo para una entidad y de ser válidamente comparables dos o más entidades entre sí, permitiendo juzgar la evolución de la entidades económicas.

Conceptos que componen la estructura básica de la contabilidad.

- Principios.
- Reglas particulares.
- Criterio prudencial de aplicación de las reglas particulares.

El orden en que se presentan los conceptos que forman la estructura de la teoría de la contabilidad está basado en la generalidad decreciente, que se da entre los principios y las reglas particulares. El criterio prudencial de aplicación se refiere a las reglas particulares, ya que los principios son proposiciones de carácter más amplio que deben aplicarse en todos los casos.

Principios

Los principios de contabilidad son conceptos básicos que establecen la delimitación e identificación del ente económico, las bases de cuantificación de las operaciones y la presentación de la información financiera cuantitativa por medio de los estados financieros. Los principios de contabilidad que identifican y delimitan al ente económico y a sus aspectos financieros, son: la entidad, la realización y el periodo contable.

Reglas Particulares

Son la especificación individual y concreta de los conceptos que integran los estados financieros. Se dividen en: reglas de valuación y reglas de presentación. Las primeras se refieren a la aplicación de los principios y a la cuantificación de los conceptos específicos de los estados financieros.

Criterio Prudencial de Aplicación de las Reglas Particulares

La medición o cuantificación contable no obedece a un modelo rígido, sino que requiere de la utilización de un criterio general para elegir entre alternativas que se presentan como equivalentes, tomando en consideración los elementos de juicio disponibles. Este criterio se aplica al nivel de las reglas particulares.

1.1.7.2 Finanzas

Las finanzas se reducen a algo tan simple de como disponer y manipular los elementos necesarios, para transportar cantidades que representan dinero, a través del tiempo, sin afectar un valor; así puede apreciarse que la fórmula que se emplea en éste estudio, es una ecuación de valores equivalentes, ya que es útil para transponer cantidades de una fecha a otra, y la forma de visualizarlo será a través de diagramas de tiempo.

Ganancia por el uso temporal o cantidad del dinero que se paga por el uso del capital ajeno. El interés se calcula como un porcentaje sobre un capital dado por la tasa de interés que se establece convencionalmente por cada operación particular. La tasa de interés suele expresarse como el porcentaje del capital, si la operación estuviera vigente un año (Tasa Nominal).

Tasa de Interés = Rendimiento real + Inflación + Prima de Riesgo.

El Cálculo de Interés o Costo del Dinero se puede expresar en dos modalidades: Interés simple e Interés compuesto; en donde:

- Concepto de Interés Simple.- La base o el capital no cambia, y el rendimiento depende del capital, del tiempo y la tasa de interés.

- **Concepto de Interés Compuesto.**- La base o el capital a medida que se le van incorporando los intereses, es decir se van capitalizando; de ahí entonces que el rendimiento depende: del capital, del tiempo, de la tasa de interés y de la frecuencia de la capitalización.

Las Finanzas o Administración Financiera es el área que establece un conjunto de técnicas y métodos que buscan la optimización de los recursos monetarios de una empresa. La Función financiera en las empresas se divide en tres partes:

- **Financiamiento.**- Allegarse de recursos buscando conseguirlos en los montos necesarios al costo más bajo y con los menores riesgos.
- **Inversión.**- Aplicar los recursos en los renglones más rentables sin descuidar la liquidez.
- **Dividendos.**- En la retribución a los accionistas por su inversión en la empresa.

Objetivos Financieros:

- **Rentabilidad o Rendimientos.**- Maximización del valor de la empresa.
- **Liquidez.**- Capacidad de pago a tiempo.

Políticas de Inversión

- **Postura líquida o conservadora.**- Invertir fuertemente en partidas de activo circulante.
- **Postura productiva o liberal o agresiva.**- Invertir grandemente en activos fijos.
- El rendimiento que se obtenga de las inversiones en activos deberá ser superior al costo de financiamiento. Las inversiones permanentes (capital neto de trabajo y activos fijos) deben financiarse con fuentes de financiamiento a largo plazo.
- Las necesidades temporales para inversión en activos circulantes, deben financiarse con fuentes de financiamiento a corto plazo.

En la actualidad el desarrollo de nuevas técnicas financieras se ha dado en el diseño y aplicación de modelos matemáticos.

Políticas de Financiamiento

- **Estructura financiera.**- Forma de financiar a una empresa. Relación entre las principales fuentes de financiamiento (Pasivo a corto, largo plazo y Capital común). Financia todas las inversiones de Activo.
- **Estructura de capital.**- Relación de fuentes de financiamiento a largo plazo (Pasivo a largo plazo y capital contable). Financia inversiones de activo fijo y capital neto de trabajo.

Las empresas pueden usar una estructura financiera, conservadora o liberal.

- **Estructura financiera liberal o riesgosa.**- Usa grandemente la palanca financiera o capital ajeno (pasivo y capital preferente).
- **Estructura financiera conservadora o poco riesgosa.**- Usa grandemente el capital propio.

1.2 Conceptos Básicos de Contabilidad Financiera

1.2.1 Origen y Definición

Existen diversas definiciones de Contabilidad, definiciones que, en el transcurso del tiempo, la describen como arte, como una ciencia, como una disciplina, etc.

Es indudable que la Contabilidad ha llegado a un grado de depuración jamás alcanzado en épocas anteriores; y es en virtud de este refinamiento de sus normas y principios que se la describe fundamentalmente como un proceso de razonamiento e interpretación. La Contabilidad es un problema de juicio, no de Aritmética.

Una de las más autorizadas definiciones de Contabilidad dice que "es el arte de registrar, clasificar y resumir de manera significativa y en términos monetarios, transacciones que son, de carácter financiero, así como de los resultados obtenidos".

Se trata, pues, de registrar las transacciones de manera "significativa y en términos monetarios", para después clasificarlas, resumirlas e interpretarlas.

Ahora bien, para hacer que el registro de una operación sea significativo habrá de describir la operación; y para expresarlo en términos monetarios, habrá que calcularla, computarla, contarla.

La palabra contar, de la que se deriva contabilidad, significa tanto referir como computar.

La contabilidad financiera es una técnica que se utiliza para producir sistemática y estructuralmente información cuantitativa expresada en unidades monetarias de las transacciones que realiza una entidad económica y de ciertos eventos económicos identificables y cuantificables que la afectan, con el objeto de facilitar a los diversos interesados el tomar decisiones en relación con dicha entidad económica.

1.2.2 Importancia y Utilidad

El desarrollo de los negocios y el surgimiento frecuentemente de nuevas prácticas en el mundo mercantil, requieren que la contabilidad se adapte a las necesidades actuales, por lo que los métodos y teorías contables han evolucionado.

La contabilidad no sólo sirve a los negocios. Si no también a las asociaciones benéficas, culturales y deportivas, las instituciones educativas, y en fin, toda clase de entidades con fines no lucrativos requiere también de información financiera suministrada por la contabilidad, y así emplear en la mejor forma los recursos destinados a los fines de la entidad.

Entre los distintos usuarios de la información financiera, se pueden mencionar los siguientes:

1. Los directivos de la empresa y los funcionarios de alto nivel, son los primeros interesados, para llevar a cabo con mayor eficiencia sus funciones administrativas, por lo tanto la información les resulta indispensable.
2. Los socios o accionistas de la empresa necesitan conocer cuál es la situación financiera de esta, sus resultados periódicos y otros datos, para estar al tanto del desarrollo de la misma, y de esta forma participar en las asambleas de socios con los elementos de juicio necesarios para tratar los asuntos que hayan de resolverse.
3. Los trabajadores de la empresa, especialmente en lo que concierne a su participación en las utilidades del negocio.
4. El gobierno, tanto para efectos de la captación de cierto tipo de impuestos, como por otros requerimientos legales y para la formación de estadísticas.
5. Bancos y otros acreedores, actuales o potenciales, a quienes la información financiera les ayuda a determinar el grado en que están o estarán garantizando sus créditos, así como a decidir sobre la posibilidad de conceder créditos o ampliar los existentes.
6. Inversionistas, ya sean personas físicas o morales, que estén considerando la conveniencia de invertir en la empresa, comprarla, adquirir parte de sus acciones, fusionarse con ella o establecer algún tipo de asociación de negocio

1.2.3 Principios de Contabilidad Generalmente Aceptados

Los principios de contabilidad son conceptos básicos que establecen la delimitación de las operaciones y la presentación de la información financiera cuantitativa por medio de los estados financieros.

Los principios de contabilidad que identifican y delimitan al ente económico y sus aspectos financieros, son: la entidad, la realización y el período contable.

Los principios de contabilidad que establecen la base para cuantificar las operaciones del ente económico y su presentación, son: el valor histórico original, el negocio en marcha y la dualidad económica.

Estos principios se refieren al aspecto de transformar los datos en información y a la presentación de ésta, adecuada a sus usos o fines.

1. **ENTIDAD.** La actividad económica es realizada por entidades identificables, las que constituyen combinaciones de recursos humanos, recursos naturales y capital, coordinados por una autoridad que toma decisiones encaminadas a la consecución de los fines de la entidad.

CAPÍTULO I. CONCEPTOS BÁSICOS DEL NEGOCIO

2. **REALIZACION.** La contabilidad cuantifica en términos monetarios las operaciones que realiza una entidad con otros participantes en la actividad económica y ciertos eventos económicos que la afectan.
3. Las operaciones y eventos económicos que la contabilidad cuantifica se considerarán por ella realizados: 1) cuando ha efectuado transacciones con otros entes económicos. 2) cuando han tenido lugar transformaciones internas que modifican la estructura de recursos o de sus funciones o 3) cuando han ocurrido eventos económicos externos a la entidad o derivados de las operaciones de ésta y cuyo efecto pueden cuantificarse razonablemente en términos monetarios.
4. **PERIODO CONTABLE:** La necesidad de conocer los resultados de operaciones financieras de la entidad, que tienen una existencia continua, obliga a dividir su vida en periodos convencionales.
5. **VALOR HISTORICO ORIGINAL.** Las transacciones y eventos económicos que la contabilidad cuantifica se registran según las cantidades de efectivo que se afecten o su equivalente o la estimación razonables contablemente. Estas cifras deberán ser modificadas en el caso de que ocurran eventos posteriores que les hagan perder su significado, aplicando métodos de ajuste en forma sistemática que preserven la imparcialidad y objetividad de la información contable.
6. **NEGOCIO EN MARCHA.** La entidad se presume en existencia permanente, salvo especificación en contrario; por lo que las cifras de sus estados financieros representarán valores históricos, o modificaciones de ellos, sistemáticamente obtenidos.
7. **DUALIDAD ECONOMICA.** Esta dualidad se constituye de:
 - Los recursos de los que dispone la entidad para la realización de sus fines
 - Las fuentes de dicho recursos, que a su vez, son la especificación de los derechos que sobre los mismos existen considerados en su conjunto.
8. **REVELACION SUFICIENTE.** La información contable presentada en los estados financieros debe contener en forma clara y comprensible todo lo necesario para juzgar los resultados de operación y la situación financieras de la entidad.
9. **IMPORTANCIA "RELATIVA."** La información que aparece en los estados financieros debe mostrar los aspectos importantes de la entidad susceptibles de ser cuantificados en términos monetarios.
10. **CONSISTENCIA:** Los usos de la información contable requieren que se siga un procedimiento de calificación que permanezca en el tiempo.

1.2.4 Registro de Operaciones

Todas las transacciones que realiza una empresa se traducen en aumentos o disminuciones de activo, pasivo o capital. El recurso que utiliza la contabilidad para registrar esas transacciones se llama cuenta. Ejemplo:

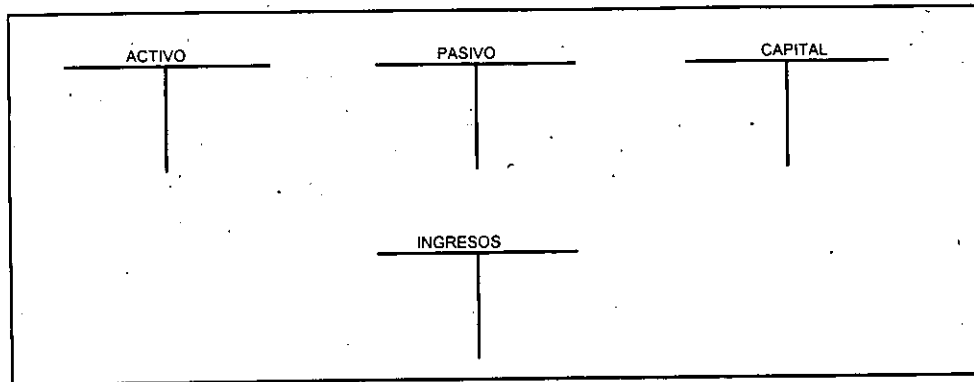


Figura 1.2_1 Cuenta

El registro de las operaciones se basa en el razonamiento de los efectos financieros de cada transacción, para traducirlos a Cargo y Abono. Lo que a todos los registros del lado izquierdo de la cuenta se llaman Cargos y los registros del lado derechos abonos. Al lado izquierdo de la cuenta se le llama Debe y al lado derecho se le llama Haber. A la suma de cargos en una cuenta se le llama movimiento deudor; a la suma de los abonos, movimiento acreedor.

Existe un saldo en el debe cuando la suma de todos los cargos en una cuenta es mayor a la suma de los abonos; y a este saldo se le llama saldo deudor.

Por lo general, las cuentas de activo y de gastos tienen saldos deudores, excepto las complementarias de activo.

Existe un saldo en el haber cuando la suma de todos los abonos en una cuenta es mayor a la suma de los cargos, y se le llama saldo acreedor.

Por lo general, las cuentas de pasivo, capital e ingresos tienen saldos acreedores.

En las cuentas de activo, los aumentos se registran en el lado izquierdo de la cuenta y las disminuciones, en el lado derecho de la cuenta.

En las cuentas de pasivo y capital los aumentos se registran en el lado derecho de la cuenta y las disminuciones, en el lado izquierdo.

CAPITULO 1. CONCEPTOS BÁSICOS DEL NEGOCIO

En las cuentas de resultados los ingresos se registran en el lado derecho de la cuenta y los gastos en el lado izquierdo.

Por lo que no siempre un cargo representa algo desfavorable y un abono algo favorable, esto depende de la naturaleza de la cuenta.

Lo anterior se expresa en la figura 1.2 2.

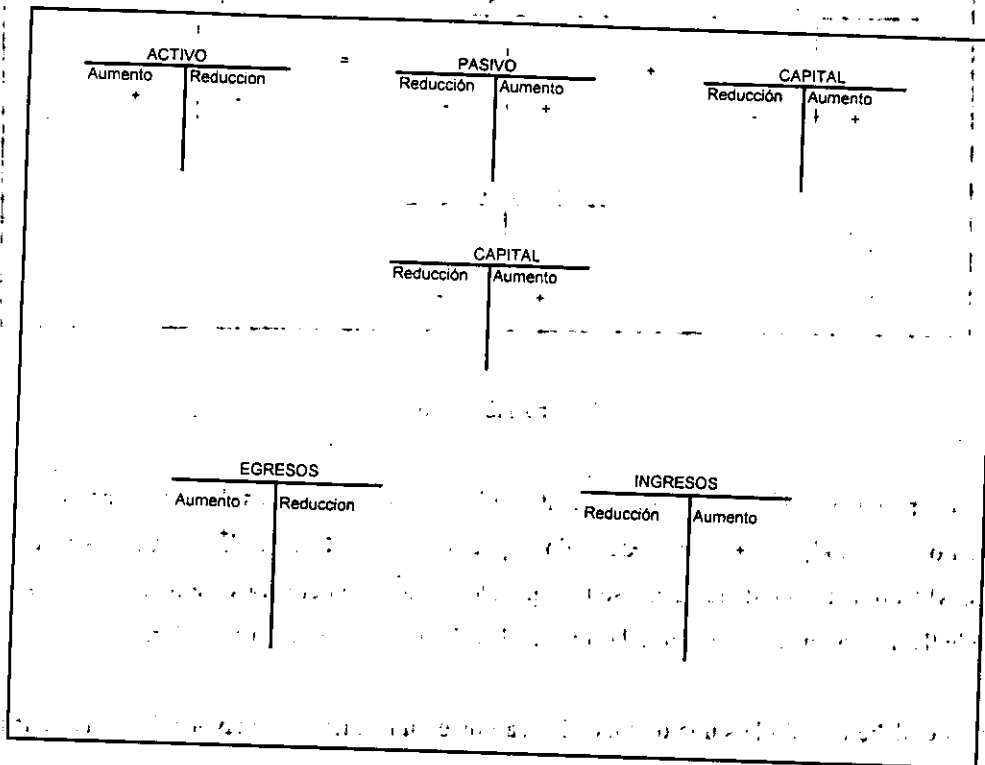


FIGURA 1.2_2 Representación de Cuentas

1.2.5 Ecuación Básica de la Contabilidad

La terminología técnica de la Contabilidad, se designa como Activo a los recursos de que dispone la empresa. A las deudas que se encuentran pendientes de pago se les denomina Pasivo. La propiedad neta que en la entidad tienen sus dueños se conoce como Capital.

Los recursos con que cuenta una empresa han sido entregados a ésta ya sea por sus dueños o por terceras personas que los otorgan en calidad de préstamo, se infiere que el total de los recursos es igual a la suma de las fuentes.

Esto significa que el Activo es igual a la suma de pasivo y el Capital, lo que se representa por la siguiente fórmula, misma que constituye la ecuación del Balance: $A = P + C$

Donde :

A = Activo

P = Pasivo

C = Capital

Las diversas transacciones que efectúe la empresa, modifican la composición de uno o más de los tres grupos principales de que compone el balance, pero la igualdad indicada nunca se altera.

1.2.5.1 Definición y Clasificación del Activo

El activo, es el conjunto de bienes y derechos que posee una negociación o empresa para su uso y aprovechamiento. Es decir, son las propiedades que tiene o cuenta la empresa para su operación, por ejemplo: todos los bienes, derechos y servicios que posee. El activo se clasifica de acuerdo con su disponibilidad, esto es, la rapidez con que se convierte en dinero.

- **Activo Circulante.** En este grupo deberán clasificarse todos aquellos valores de los cuales la empresa podrá disponer inmediatamente, así como todos aquellos valores que para la negociación o empresa, representan la misma base de sus operaciones, y la cual representa a la vez su fuente principal de utilidades.

Además el efectivo y todas aquellas partidas que se convierten en efectivo en un plazo menor a un año, con las que se realiza el ciclo normal de operaciones de una empresa y que a su vez sea la fuente principal de sus utilidades. Ejemplo: dinero en efectivo, dinero depositado en cuentas de cheques, inversiones en valores cuando estas sean a corto plazo, las mercancías, los artículos a crédito, los documentos (pagarés, letras de cambio) firmados a favor de la negociación.

Por lo general, los activos circulantes que tiene una empresa son:

CAJA. Las entidades cuentan con dinero recibido por diversos motivos, uno de ellos puede ser la operación que el dueño o socio efectúan para iniciar el negocio, y posteriormente habrá entradas y salidas de dinero por las operaciones del mismo. Es el primer renglón del Balance y se le da el nombre técnico de Caja o Efectivo en Caja.

BANCOS. Comúnmente la gran parte del efectivo o incluso el total del mismo se deposita en una o más instituciones bancarias, evitando así el guardarlo en las oficinas de la empresa.

ACCIONES Y VALORES NEGOCIABLES. Las empresas pueden tener temporalmente, excedentes de efectivo. Algunas de ellas invierten dichos excedentes en acciones y otros valores cotizados en Bolsa de Valores, con el fin de ganar un rendimiento de tal inversión. El que se cotice en Bolsa significa que son negociables, es decir, que pueden venderse fácilmente para convertirlos en efectivo casi de inmediato.

CLIENTES. Muchas empresas venden a crédito sus mercancías. Las cantidades que por este concepto tiene pendientes de cobrar el negocio.

DOCUMENTOS POR COBRAR. Aquellos adeudos que la entidad tiene a su favor, ya sean que provengan de ventas de mercancías o de otros conceptos, que estén amparados por pagarés o letras de cambio; constituyen este renglón del Activo. Donde con dichos documentos hacen constar las obligaciones de una persona de pagar una cantidad de dinero por otra, pero con ciertas peculiaridades que les son propias por tratarse de títulos de crédito.

DEUDORES DIVERSOS. Se refieren a las cantidades por cobrar que tenga la empresa por motivos diferentes a la venta de mercancías, como pueden ser prestamos a empleados y otras personas, reclamaciones a compañías de seguro sobre bienes asegurados que se hayan dañado o perdido, etcétera, siempre que en todos estos casos el adeudo no este amparado por letras de cambio o pagares.

ALMACEN. Se refiere al costo de los artículos que la empresa haya comprado para posteriormente revenderlos, como operación que constituya su principal actividad mercantil. A este tipo de artículos destinados al tráfico mercantil propio de la empresa, les conocemos como mercancías, razón por la cual a este elemento del Activo también se le puede dar la denominación de Almacén de Mercancías.

- **Activo Fijo.** Se consideran como activo fijo, todas aquellas inversiones que se hacen con un carácter permanente, como bienes muebles e inmuebles que posee la empresa para realizar sus operaciones y con el propósito de usarlos y no se compran con la intención de venderlos. Ejemplo: la adquisición de terrenos, en edificios, en máquinas y equipos de oficina, en mobiliario y equipo, en vehículos, depósitos en garantía etc. Por lo general, los activos fijos de una empresa son:

TERRENOS. Este renglón consiste de bienes propiedad de la empresa cuya función es ser usados para el desarrollo de sus operaciones. En este renglón de los terrenos, se incluye el costo de aquellos destinados a construir posteriormente un edificio en el que la empresa instale sus oficinas, bodegas u otras dependencias.

Las empresas que no tienen edificio propio para instalaciones, deberán alquilar un local, que precisamente por ser ajeno no forma parte del Activo.

EDIFICIOS. Estructuras propiedad de la empresa que se utilizan para realizar sus operaciones.

MAQUINARIA. Esta cuenta solo aparece en las empresas industriales y se utiliza para la fabricación o transformación del producto.

MOBILIARIO Y EQUIPO DE OFICINA. Dentro de este rubro se incluye todos los muebles de la oficina

EQUIPO DE TRANSPORTE. Dentro de este rubro es el que se utiliza para la entrega de mercancías o uso del personal de la empresa.

- **Activo no circulante.** Este activo se divide en dos grandes rubros.

1. **Cargos diferidos.** Son todos aquellas partidas que se espera que aporten beneficios a la empresa y son amortizables. En este grupo se clasifican todas aquellas operaciones que en ocasiones tienen el carácter de

inversiones, pero que después de un tiempo pasan a ser gastos. También en el presente grupo deberán incluirse los valores que en ocasiones representan conceptos intangibles y que cubrimos en forma anticipada. Ejemplos de estas operaciones son los gastos de organización e instalación, que realiza la empresa antes de empezar sus operaciones, las rentas, primas de seguros, interés e impuestos pagados por anticipado.

GASTOS DE ORGANIZACIÓN. Al iniciar un negocio, es frecuente que se incurra en ciertos gastos para dotarlo de una estructura formal de operación. Algunos de ellos son de índole legal, como los relacionados con la escritura constitutiva de una sociedad mercantil; otros son de carácter contable, para organizar el sistema de contabilidad y control interno, etc.

GASTOS DE INSTALACION. Cuando para instalar un negocio se alquila un local, no siempre reúne este las condiciones requeridas para el adecuado desarrollo de la actividad de la empresa. Tales gastos dan al local las condiciones apropiadas para el funcionamiento del negocio, y por lo tanto benefician a las operaciones futuras, durante años por lo regular, aun cuando no se puede conocer el tiempo preciso.

PATENTES. Son los documentos que acreditan oficialmente los derechos sobre un invento o propiedad industrial.

MARCA. Es el distintivo que el fabricante añade a sus productos para identificarlos.

2. **Gastos Diferidos.** Son aquellos gastos pagados por anticipado, que se devengan en un plazo mayor a un año, por lo que se espera un servicio aprovechable posteriormente.

SEGUROS PAGADOS POR ANTICIPADO, RENTAS PAGADAS POR ANTICIPADO, INTERESES PAGADOS POR ANTICIPADO, ETCETERA. Se trata de ciertos gastos de la operación normal de la entidad, que por determinadas causas han sido pagados con antelación a los meses o periodos contables a los que corresponden. Conforme se vaya recibiendo el servicio o beneficio pagado estas partidas dejan de ser un recurso, por lo que se deben traspasar a gastos.

Los periodos o meses en los cuales se obtendrán los servicios o beneficios pagados por anticipado, están determinados o son susceptibles de determinar con exactitud. A los diferentes conceptos de esta naturaleza, se les conoce con el término genérico de "Pagos Anticipados".

La diferencia básica entre cargos diferidos y gastos diferidos es que, en los primeros, el beneficio que aportan es por un periodo ilimitado, que depende de la duración de la empresa y, en los segundos, el beneficio será por un periodo limitado, inferior a la duración de la empresa, pero mayor al ciclo normal de operaciones, es decir, mayor a un año.

1.2.5.2 Definición y Clasificación del Pasivo

El pasivo comprende obligaciones presentes provenientes de operaciones o transacciones, tales como la adquisición de mercancías o servicios, pérdidas o gastos en que se han incurrido, o por la obtención de préstamos para el financiamiento de los bienes que constituyen el activo. Dicho de otra manera, son todas aquellas deudas u obligaciones que tiene una empresa.

El pasivo se clasifica de acuerdo a su exigibilidad, esto es, la fecha de liquidación o vencimiento en que la empresa debe pagar, en pasivo a corto plazo y pasivo a largo plazo.

1. **A Corto Plazo.** El pasivo a corto plazo es aquel cuya liquidación se producirá dentro de un año, en el curso normal de las operaciones. Se hace excepción a esta regla en el caso de que el ciclo normal de las operaciones de la empresa sea mayor a un año y se haga una excepción similar en la clasificación del activo circulante. En este caso, el pasivo a corto plazo incluirá todos los adeudos que sean pagaderos dentro del ciclo normal de las operaciones y deberá mencionarse esta situación en los estados financieros y notas complementarias.

Esta formado por todas las cuentas que se tienen que pagar en un plazo menor o igual a un año, y por las cantidades cobradas por anticipado. Por lo general, las cuentas de pasivo a corto plazo que tiene una empresa son:

PROVEEDORES. Representa el importe que se debe a otras empresas o personas, con motivo de haberles comprado a crédito mercancías, las cuales como ya se dijo, son los artículos por pagarés o letras de cambio.

ACREEDORES DIVERSOS. Se refiere a lo que debe la empresa, por diversos conceptos, que no sean por compra de mercancías ni impuestos, y tampoco estén amparados por pagarés o de letras de cambio. Puede tratarse de rentas; honorarios y otros diversos gastos pendientes de pagar; préstamos obtenidos que no se documentaron con títulos de crédito; compra a crédito de mobiliario y equipo de oficina u otros equipos de oficina u otros equipos, etc.

RENTAS COBRADAS POR ANTICIPADO. Se integra por el importe de las rentas que se hayan cobrado con anticipación a los meses a que corresponden. Las rentas pueden derivar del arrendamiento o subarrendamiento que la empresa haya otorgado de algún local.

Este concepto y otros cobros anticipados, representan para la empresa la situación opuesta a los Pagos Anticipados que aparecen en el Activo. Las Rentas Cobradas por Anticipado, y otros conceptos de la misma índole, se clasifican en el Pasivo, pues representan una obligación financiera que se ha de cubrir con un servicio futuro.

2. **A Largo Plazo.** El pasivo a largo plazo esta representado por los adeudos cuyo vencimiento sea posterior a un año, o al ciclo normal de las operaciones si este es mayor. La parte del pasivo a largo plazo que por el transcurso del tiempo llega a ser pagadero dentro de los próximos doce meses o dentro del ciclo normal de las operaciones si este ultimo es mayor, se convierte en pasivo a corto plazo y por lo tanto debe clasificarse como tal. Asimismo, aquellos adeudos cuyo

vencimiento no este sujeto al control de la empresa, se clasifican comúnmente como pasivo a corto plazo. Son las cuentas que se deben pagar en un plazo mayor a un año.

DOCUMENTOS POR PAGAR. Son las cantidades que se deben por cualquier concepto, pero que estén amparadas por pagarés o letras a cambio.

IMPUESTOS POR PAGAR. Este renglón registra las cantidades pendientes de pagar por diversos impuestos.

ACREEDORES HIPOTECARIOS. Se trata de préstamos obtenidos, en los que para garantizar el pago, se ha constituido hipoteca sobre bienes inmuebles de la empresa. Básicamente, son préstamos a largo plazo, de modo que se dispone de varios años para su pago.

PRÉSTAMOS BANCARIOS A LARGO PLAZO. Bajo este rubro se agrupan los préstamos obtenidos de instituciones bancarias, a largo plazo, y que sean diferentes al crédito hipotecario.

1.2.5.3 Definición de Capital Contable

Es el que está formado por el capital social y la utilidad o pérdida del ejercicio, más reservas creadas y superávits.

Se dice del capital, en su aspecto económico, que es uno de los factores de la producción y que está representado por el conjunto de bienes que sirven para producir una riqueza.

Se denomina así la diferencia aritmética que existe, entre los bienes, de una negociación o empresa y las deudas (pasivo) de la misma.

Existen diferentes clases de capital:

- Capital económico, es uno de los factores de la producción que nos sirve para producir riqueza.
- Capital Financiero, es toda aquella inversión, que nos produce un interés o renta.
- Capital social, se le denomina así al capital de las empresas que están constituidas bajo la forma de sociedades mercantiles.
- Capital individual, es la parte proporcional de cada uno de los socios de las empresas constituidas bajo la forma de sociedades mercantiles.
- Capital invertido, es el monto del capital contribuido o aportado a un negocio por sus propietarios.

- **Utilidad del ejercicio.** Como consecuencia de las operaciones efectuadas por el negocio durante un cierto periodo, principalmente por la venta de mercancías, y después de considerar sus diversos gastos, el negocio obtiene una ganancia. La utilidad del ejercicio pertenece a los dueños del negocio, es decir aumenta el capital contable.
- Puede suceder que en vez de ganancia, las operaciones del negocio arrojen pérdidas, en cuyo caso el concepto que aparecerá en el balance será el de Perdida del Ejercicio.
- **Utilidades de ejercicios anteriores.** Significa el importe de las utilidades obtenidas en los años anteriores al último que no hayan sido retiradas del negocio por los socios o el dueño.

1.2.6 Estados Financieros

1.2.6.1 Objetivo y Utilidad

Las transacciones que realiza una entidad económica y ciertos eventos económicos identificables y cuantificables que la afectan, son medidos, registrados, clasificados, analizados, resumidos y finalmente reportados como información por medio de los estados financieros.

Siendo así, el propósito final de la contabilidad es que la información sea útil para llegar a conclusiones acerca de los resultados de la gestión empresarial, compara las expectativas con las situaciones alcanzadas, a juzgar el grado de cumplimiento de las proyecciones, y en general para basar decisiones y cursos de acción.

Para cumplir dicha finalidad, la información financiera debe comunicarse en la forma adecuada, a efecto de facilitar su análisis e implementación. Esta es la función que compete a los estados financieros y demás reportes contables.

Los estados financieros son documentos que describen un determinado tipo de información financiera, resumida en forma significativa. Se conoce como estados financieros "principales", a los que contienen la información esencial para conocer en una perspectiva los aspectos financieros relevantes de una entidad.

Los estados financieros constituyen el conjunto de documentos con la información básica que muestra la posición financiera de la empresa y los resultados de sus operaciones en un periodo determinado.

Resulta de interés para casi todos los usuarios de la información financiera, y son los siguientes:

1. Estado de Resultados.
2. Balance General.

1.2.6.2 Balance General

El Balance general es uno de los estados financieros principales, y su comprensión facilita no sólo la de los demás estados, sino también los aspectos fundamentales del proceso de registro contable.

Se define como: Balance General es el estado financiero que muestra la situación financiera de una entidad, a una fecha determinada.

Por situación financiera se entiende la composición de los recursos que tiene una empresa y de las fuentes de esos recursos. Los recursos provienen de dos tipos de fuentes: las personas ajenas a la entidad y los dueños de ésta. En otras palabras la situación financiera se integra por las propiedades de una entidad, sus obligaciones o adeudos y el valor neto de que la misma pertenece a sus dueños.

Al Balance se le clasifica como estado financiero estático, debido a que la información que muestra se refiere a una fecha determinada, es decir, indica en qué circunstancias se encuentran las cosas en un momento dado.

Todo Balance General, debe practicarse en determinada fecha, escogiéndose, por regla general, la que sirve de base para el cierre de las operaciones anuales. Pero esto no quiere decir que el balance se deba realizar cada año, ya que ni desde el punto de vista legal ni técnico, existe inconveniente o impedimento para que se realice en periodos más cortos. Y al periodo de tiempo que comprende un Balance General, Técnicamente se le llama **ejercicio social**.

Formas de Representar un Balance General

Existen en uso dos formas para representar los balances generales, las cuales son:

- a) En forma de cuenta u horizontal.
- b) En forma de reporte o vertical.

Una y otra forma dependen de la disposición que se les dé a los datos relativos al activo, pasivo y capital no variando en lo más mínimo el contenido del balance.

Todo balance se compone esencialmente de dos partes:

- 1) El encabezado
- 2) El rayado, en donde se anotan los datos y cantidades y el cual varía según sea la forma de presentación que se escoja.

A lo que se refiere al encabezado, diremos que debe contener los datos que identifiquen tanto al negocio como la naturaleza del estado que se va a formular y, por consiguiente, la práctica establecida es anotar como encabezado lo siguiente:

CAPITULO 1. CONCEPTOS BÁSICOS DEL NEGOCIO

- a) El nombre del comerciante, empresa o sociedad mercantil de que se trate.
- b) El nombre del estado que se va a formular.
- c) La fecha en la cual se formula dicho balance.

Balance en Forma de Cuenta

La forma indicada consiste en el uso de un rayado que se compone de dos secciones: una que se coloca en el lado izquierdo y la otra en el derecho. La sección del lado de izquierdo sirve para anotar todas aquellas partidas que constituyen el activo del comerciante y la del lado derecho, para anotar todas las que forman su pasivo y además el capital. Tal como se muestra en la figura 1.2_3.

" EL CREDITO MERCANTIL "															
BALANCE GENERAL AL 31 DE DICIEMBRE DE 19															
ACTIVO							PASIVO								
Disponible:															
Caja	\$	100	000				Proveedores	\$	600	000					
Bancos		300	000	\$	400	000	Cuentas por pagar		400	000					
							Acreedores a corto plazo		200	000	\$	200	000		
Circulante:															
Mercancías		500	000				Hipotecas	21	500	000					
Documentos por cobrar		300	000				Acreedores a corto plazo	10	300	000	31	800	000		
Clientes		500	000	2	300	000									
Fijo:															
Equipo de oficina		250	000				<u>Suma el pasivo</u>				\$	33	000		
Equipo de reparto	10	400	000				CAPITAL								
Edificio	22	500	000	33	150	000	Capital	2	000	000					
Diferido:							Utilidad del ejercicio		940	000		2	940	000	
Gastos pagados por adelantado		30	000												
Gastos de instalacion		60	000		90	000									
<u>Suma del Activo</u>				\$	35	940	000	<u>Suma el Pasivo y el Capital</u>					35	940	000

FIGURA 1.2_3 Balance en Forma de Cuenta

Cuando se usa esta forma de presentación, la suma que arrojan los valores de activo debe ser igual a la suma de los valores de pasivo más el capital. Esta igualdad entre el total de los valores de activo y el total de los valores de pasivo más capital, es lo que ya se mencionó y que se conoce con el nombre de fórmula del balance general:

$$A = P + C$$

Donde :

A = Activo

P = Pasivo

C = Capital

Balance en Forma de Reporte

La diferencia que hay entre la presentación de un balance en forma de reporte con respecto a la forma de cuenta, consiste en que los datos relativos al activo, pasivo y capital se anotan en forma diferente.

Cuando se emplea esta forma de presentación, en lugar de la anterior, los datos correspondientes se disponen como sigue: en primer lugar, se anotan las partidas que comprenden el activo, obteniéndose el total correspondiente; después e inmediatamente abajo del total del activo, se anotan las partidas que constituyen el pasivo, y también se obtiene el total correspondiente, tal y como se muestra en la figura 1.2_4.

"EL CREDITO MERCANTIL"					
BALANCE GENERAL AL 31 DE DICIEMBRE DE 19					
ACTIVO					
Disponible:					
Caja		100 000			
Bancos	\$	300 000		\$	400 000
Circulante:					
Mercancías		500 000			
Documentos por cobrar		300 000			
Clientes		500 000		2	300 000
Fijo:					
Equipo de oficina		250 000			
Equipo de reparto	10	400 000			
Edificio	22	500 000		33	150 000
Diferido:					
Gastos pagados por adelantado		30 000			
Gastos de instalación		60 000			90 000
Suma del Activo					940 000
PASIVO					
Proveedores	\$	600 000			
Cuentas por pagar		400 000			
Acreedores a corto plazo		200 000		1	200 000
Hipotecas	21	500 000			
Acreedores a corto plazo	10	300 000		31	800 000
Suma el pasivo					000 000
CAPITAL					
Capital		2 000 000			
Utilidad del ejercicio		940 000		2	940 000

FIGURA 1.2_4 Balance en Forma de Reporte

Una vez determinado el total del activo y el total del pasivo, se obtiene la diferencia entre estas dos cantidades, a fin de determinar el capital, cuyo dato se anota después de la suma del pasivo. A esta forma de presentación se le conoce con el nombre de reporte o vertical.

$$A - P = C$$

A esta fórmula se le conoce con el nombre de fórmula de capital.

1.2.6.3 Estado de Resultados

El Estado de Resultados es un estado financiero que muestra la forma en que se genera la utilidad o pérdida de un periodo, especificando los elementos que la integran. Se puede denominar de algunas otras formas; como por ejemplo Estado de Perdidas y Ganancias, término que se empleó en alguna época pero actualmente a caído en desuso; o bien Estado de Ingresos y Egresos, mejor que la anterior pero escasamente empleada.

Se le clasifica como un estado dinámico, lo cual deriva de que presenta información relativa a un periodo, y por lo mismo las cifras que lo constituyen reflejan las operaciones efectuadas en el transcurso de ese periodo.

Así como el Balance General se hace el símil con una fotografía, que muestra una situación en un momento dado, el estado de resultados se le compara con la película, en la cual se puede observar el movimiento de algo o alguien durante cierto lapso.

Entre el conjunto de transacciones financieras que efectúa un negocio, un segmento de ellas se relaciona directamente con la obtención de una cierta utilidad o pérdida, el resumen de estas operaciones se plasma en el estado de resultados.

Las cuentas que se utilizan para realizar el estado de resultados son las cuentas de ingresos y egresos, ya que están definen si una empresa tuvo utilidad o pérdida en un periodo determinado. Como se puede apreciar en la figura 1.2_5.

Estado de resultados del 1o. de enero al 31 de diciembre de 19		
Ventas		\$1,000,000.00
Menos:		
Costo de ventas		500,000.00
Utilidad bruta		\$500,000.00
Menos		
Gastos de Operación		
Administración	100,000.00	
Venta	50,000.00	
Financieros	50,000.00	
Utilidad de operación		\$300,000.00
Menos:		
Impuestos y PTU		150,000.00
Resultado de ejercicio (utilidad)		\$150,000.00

FIGURA 1.2_5 Compañía La Utilidad o Pérdida, S.A.

CAPITULO 2

MARCO TEÓRICO Y HERRAMIENTAS

2.1 Teoría de Bases de Datos Relacionales

2.1.1 Concepto de Base de Datos

Son muy numerosas las definiciones de base de datos, y si se analizan detenidamente, se suele observar en casi todas ellas coincidencias en ciertos elementos; aunque también se detecta la falta de otros, fundamentales, o al menos muy importantes, que son característicos de las bases de datos y que marcan la diferencia entre este concepto y el de ficheros. Resumiendo las características esenciales de una base de datos se exponen los siguientes párrafos.

Una base de datos es un conjunto, colección o depósito de datos almacenados en un soporte informático. Los datos están interrelacionados y estructurados de acuerdo con un modelo capaz de recoger el máximo contenido semántico. Dada la relevancia que tienen en el mundo real las interrelaciones entre los datos, es imprescindible que la base de datos sea capaz de almacenar estas interrelaciones. En el mundo real existen, además, restricciones semánticas, a las que se está concediendo una importancia creciente y que, en los sistemas actuales, tienden a almacenarse junto con los datos, al igual que ocurre con las interrelaciones. La base de datos se describe y se manipula apoyándose en un modelo de datos.

La redundancia de los datos debe ser controlada, de forma que no existan duplicidades perjudiciales ni innecesarias, y que las redundancias físicas, convenientes muchas veces a fin de responder a objetivos de eficiencia, sean tratadas por el mismo sistema, de modo que no puedan reproducirse inconsistencias. Esto podría resumirse diciendo que en las bases de datos no debe existir redundancia lógica, aunque si se admite cierta redundancia física por motivos de eficiencia.

La definición o descripción del conjunto de datos contenidos en la base (lo que se denomina estructura o esquema de la base de datos) deben ser únicas y estar integradas con los mismos datos.

2.1.2 Modelo de Datos

Un modelo de datos es un conjunto de conceptos que permiten describir, a distintos niveles de abstracción, la estructura de una base de datos, a la cual denominamos esquema.

Las propiedades del modelo de datos son de dos tipos: estáticas y dinámicas.

La estática de un modelo de datos está compuesta por:

a.- Elementos permitidos.

No son los mismos para todos los modelos de datos (varían especialmente en su terminología), pero en general son:

- Objetos (entidades, relaciones, registros, etc.)
- Asociaciones entre objetos (interrelaciones, "set" o conjunto, etc.)
- Propiedades o características de los objetos o de las asociaciones (atributos, campos, elementos de datos, etc.).
- Dominios, conjuntos nominados de valores sobre los que se definen las propiedades.

b.- Elementos no permitidos o restricciones:

Dinámica

- Los valores que toman los distintos objetos de un esquema en un momento determinado t_1 reciben el nombre de ocurrencia del esquema o estado de la base de datos en el tiempo t_1 (BD_1); en otro momento t_2 la ocurrencia del esquema será (BD_2).
- Esto es si entre t_1 y t_2 se ha producido un cambio en algún valor de la base de datos (alta, baja o modificación), tanto BD_1 como BD_2 deben ser ocurrencias válidas de la base de datos, es decir, ambas deben cumplir las restricciones de integridad así como las posibles restricciones asociadas al cambio de estado.

Modelos Lógicos Basados en Registros

- Los modelos lógicos basados en registros se llaman así, porque la base de datos está estructurada en registros de formato fijo de varios tipos. Cada registro define un número fijo de campos, o atributos, y cada campo normalmente es de longitud fija. Esto contrasta con los modelos orientados a objetos en los que los objetos pueden estar compuestos por objetos a un nivel de anidamiento de profundidad arbitraria.

- Los modelos lógicos basados en registros, no incluyen un mecanismo para la representación directa de código en la base de datos. En cambio, tienen asociados lenguajes que permiten expresar consultas y actualizaciones sobre la base de datos.

Dentro de los modelos lógicos basados en registros tenemos los siguientes:

- Modelo de red.
- Modelo jerárquico.
- Modelo relacional.

Modelo de Red

En este modelo, los datos se representan como una colección de registros, la relación entre ellos se da por medio de apuntadores. Los registros se organizan como una colección de gráficas arbitrarias.

Desventajas:

- Los apuntadores o direcciones se deben almacenar junto con los datos.
- Para recuperar información se debe "navegar" a través de la gráfica.
- No se puede obtener información no planeada antes de modelar.
- Modificar la estructura de la base de datos implica redefinir todo el esquema.

Modelo Jerárquico

En este modelo, los datos se representan como una colección de registros, mientras que la relación entre ellos se da por medio de ligas o apuntadores. Se diferencia del modelo de red, en que los registros están organizados como colecciones de árboles en vez de gráficas arbitrarias.

Desventajas:

- No puede haber ciclos y sólo puede haber asociaciones 1:N y 1:1.
- Los apuntadores o direcciones se deben almacenar junto con los datos.
- Para recuperar información se debe recorrer el árbol.
- No se puede obtener información no planeada antes de modelar.
- Modificar la estructura de la base de datos implica redefinir todo el esquema.
- Se pueden representar asociaciones M:N manteniendo datos duplicados.

Modelo Relacional

El modelo relacional, es el modelo lógico que se seleccionó para la elaboración del sistema y se explicará con detalle en la siguiente sección.

2.1.3 Modelo de Bases de Datos Relacionales

Una de las mayores dificultades que existen a la hora de diseñar una base de datos es que el diseñador la concibe pensando en el modelo de datos con el que va a trabajar la máquina. Esto lleva consigo perder la visión conceptual del problema, quedando el diseño influido (contaminado) por cuestiones de implementación, de modo que el diseño es inválido cuando se implanta la base de datos en otro sistema.

El problema podría solucionarse si el diseñador fuese capaz de concebir la base de datos a un nivel superior, abstrayéndose de consideraciones relativas a la instrumentación, sea lógica o física y trabajando con elementos (entidades e interrelaciones) que son relevantes al tema en cuestión.

Modelo Relacional

En el modelo relacional, los datos y las relaciones entre los datos se representan por medio de una serie de tablas, cada una de las tablas esta compuesta por columnas con nombres únicos, las cuales tienen asociado un tipo de dato. Una columna de una tabla representa una relación entre un conjunto de valores. Existe una correspondencia entre el concepto de tabla y el concepto matemático de relación, del cual recibe su nombre el modelo relacional.

En la figura 2.1_1 se distingue de la tabla su **nombre**, un conjunto de columnas, denominadas, **atributos** que representan propiedades de la tabla y que también están caracterizadas por su nombre, y un conjunto de filas llamadas **tuplas**, que contienen los valores que toma cada uno de los atributos para cada elemento de la relación.

NOMBRE

Atributo1	Atributo2	Atributo n
XXX	XXX	XXX
XXX	XXX	XXX
XXX	XXX	XXX
XXX	XXX	XXX

Figura 2.1_1 Representación de una relación en forma de tabla

Una relación se puede representar en forma de tabla, aunque tiene una serie de elementos característicos que la distinguen de la tabla, ya que no se admiten filas duplicadas, las filas y las columnas no están ordenadas y es plan, es decir, que en el cruce de una fila y de una columna sólo puede haber un valor (no se admiten atributos multi-valorados).

En una tabla se puede distinguir una cabecera que define la estructura de la tabla, es decir, sus atributos con los dominios subyacentes, y un cuerpo que está formado por un conjunto de tuplas que varían en el tiempo.

Esta representación de la relación como una tabla ha sido el origen de que los productos relacionales y los usuarios utilicen habitualmente el nombre de tabla, para denominar las relaciones y, como consecuencia de ello, se llame filas a las tuplas y columnas a los atributos, si bien, la terminología es irrelevante y un producto no es más o menos relacional por utilizar una u otra terminología.

Ventajas

- Tiene una base matemática, conocida como álgebra y/o cálculo relacional.
- Se pueden representar fácilmente asociaciones M:N.

- No existen apuntadores u otro tipo de información que no sea la que creó la necesidad de la Base de Datos.
- Las operaciones efectuadas para obtener información se realizan en el ámbito de la tabla completa y no en el de los registros.
- No es necesario diseñar el esquema de la base de Datos de acuerdo a las operaciones o consultas que se van a llevar a cabo. Es posible obtener información no prevista.
- Se puede modificar la estructura de la Base de datos sin que esto obligue a un cambio de las aplicaciones.
- La forma de explotar la información es por medio de operaciones relacionales, a través de lenguajes de cuarta generación.

El modelo relacional tiene un fuerte apoyo matemático que permite fundamentar su estructura y las operaciones que se pueden realizar con los datos.

Un **dominio** es un conjunto de valores. Un atributo tiene asociado un conjunto de valores permitidos, es decir, un dominio.

Una **relación** es un subconjunto de un producto cartesiano de una lista de dominios, por lo tanto, las tablas son esencialmente relaciones.

Una relación es un conjunto de **tuplas** con valores para cada uno de los atributos.

Características de una Relación

Una relación en el modelo relacional tiene las siguientes características:

- Los dominios de los atributos deben ser atómicos, es decir los elementos del dominio son unidades indivisibles.
- Cada atributo tiene un nombre único en la relación.
- Los valores de una tupla corresponden a los dominios de los atributos.
- El orden de los atributos no tiene importancia.
- Cada tupla es única, no existen tuplas duplicadas.
- El orden de las tuplas no es importante.

Clases de Relación

Existen diversas clasificaciones de las relaciones. Se tienen relaciones nominadas y sin nombre. Las relaciones nominadas, a su vez pueden ser:

- Persistentes. Son aquellas relaciones cuya definición (esquema de relación) permanece en la base de datos, borrándose solamente mediante una acción explícita del usuario. Las relaciones persistentes se dividen en:
 - Relaciones base (corresponden con el nivel conceptual de la arquitectura ANSI). Existen por sí mismas, no en función de otras relaciones, y se crean especificando explícitamente su esquema de relación (nombre y conjunto de

pares: atributo / dominio). Sus extensiones (ocurrencias de relación), al igual que su definición, también se encuentran almacenadas.

- **Vistas** (se corresponden con el nivel externo de la arquitectura ANSI). Son relaciones derivadas que se definen dando un nombre a una expresión de consulta. Se podría decir que son relaciones virtuales (como ventanas sobre otras relaciones), en el sentido de que no tienen datos almacenados, sino que lo único que se almacena es su definición en términos de otras relaciones con nombre, las cuales pueden ser relaciones base, otras vistas o instantáneas.
- **Instantáneas** (se corresponden con el nivel interno de la arquitectura ANSI). Son relaciones derivadas, al igual que las vistas, es decir, se definen en términos de otras relaciones nominadas, pero tienen datos propios almacenados, los cuales son el resultado de ejecutar la consulta especificada o de guardar una relación base.
- **Temporales**. A diferencia de las relaciones persistentes, una relación temporal desaparece de la base de datos en un cierto momento sin necesidad de una acción de borrado específica del usuario.

Las relaciones sin nombre son los resultados de las consultas que no se materializan sino que se entregan al usuario que ha realizado la consulta, y pueden ser tanto resultados intermedios como finales; en consecuencia, las relaciones no nominadas son siempre temporales.

2.1.4 Dinámica del Modelo Relacional

El modelo relacional, como todo modelo de datos, lleva asociado a su parte estática (estructura y restricciones) una dinámica que permite la transformación entre estados de la base de datos. Esta transformación de un estado origen a un estado objetivo se realiza aplicando un conjunto de operadores, mediante los cuales se llevan a cabo las siguientes operaciones:

- Inserción de tuplas.
- Borrado de tuplas.
- Modificación de tuplas.
- Consulta.

Álgebra Relacional

La dinámica del modelo relacional, en lo que el álgebra se refiere, la constituye una colección de operadores de alto nivel que, aplicados a las relaciones, dan como resultado nuevas relaciones. Sean R y R' dos relaciones y O un operador cualquiera del álgebra relacional. Una operación del álgebra relacional consiste en aplicar O a la relación R , obteniéndose

R' . Al ser el resultado de la operación otra relación, se cumple la propiedad de cierre, es decir, si $O_1 \dots O_n$ representan operadores, se cumple:

$$O_n(\dots(O_1(R)))=R'$$

Los operandos del álgebra son siempre relaciones y los operandos se aplican, por tanto, a relaciones a fin de formular consultas a la base de datos.

Operadores

Para el álgebra relacional existen ocho operadores divididos en dos grupos de cuatro.

- Los operadores tradicionales de conjuntos: unión, intersección, diferencia y producto cartesiano.
- Los operadores relacionales especiales. Restricción, proyección, combinación y división.

Aparte de esta clasificación, los operadores se pueden dividir también en operadores primitivos y operadores derivados.

- Los operadores primitivos son los operadores esenciales que no pueden obtenerse de otros (sin ellos, el álgebra relacional no sería un lenguaje completo).
- Los operadores derivados se pueden obtener aplicando varios de los operadores primitivos. Se podría prescindir de ellos, ya que no aportan nada nuevo, sin embargo, son muy útiles y simplifican muchas operaciones habituales.

Otra posible clasificación, atendiendo al número de operandos de cada operador, los divide en:

- Unitarios, si el operador tiene una única relación como operando.
- Binarios, si el operador tiene dos relaciones como operandos.

Es posible definir más operandos aparte de los ocho originales, siempre que tengan una o dos relaciones como operandos y una relación como resultado.

Operaciones

Las operaciones que se pueden llevar a cabo en el modelo relacional tienen un fundamento matemático conocido como Álgebra Relacional. El Álgebra Relacional es un lenguaje de consulta abstracto y de procedimiento ("procedural"). Consta de un conjunto de operaciones que toman como entrada una o dos relaciones.

Las operaciones definidas en el Álgebra Relacional tienen la propiedad de cerradura, ya que el resultado de la operación es una relación. Existen seis operaciones fundamentales:

- Proyección Π .
- Elección σ .

- Producto Cartesiano \times .
- Unión \cup .
- Resta de conjuntos $-$.
- Intersección de conjuntos \cap .

Para ejemplificar las operaciones, se utilizaran las siguientes relaciones:

CARRERA

Cve carr	carrera
027	Industrial
032	Computación
039	Electrónica

ALUMNO

no. cta	nom alum	cve carr
105	Sandra	027
107	José	032
101	Luis	032
103	Lourdes	027
106	Juan	029
102	Santiago	029
108	Verónica	027

CARRERA OPCION

Cve carr	Carrera
015	Geólogo
020	Mecánico
035	Topógrafo

Figura 2.1_2 Relaciones de ejemplo

Proyección

Es una operación unitaria, ya que actúa sobre una sola relación. Se representa con la letra griega Π . Se utiliza para proyectar atributos, de una relación, su sintaxis es la siguiente:

$$\Pi_{\text{atributo 1, atributo 2, ..., atributo n}} (\text{Relación})$$

Ejemplo: Enumerar a los alumnos y su número de cuenta

$$\Pi_{\text{nom_alumn, no_cta}} (\text{ALUMNO})$$

nom_alum	no_cta
Sandra	105
José	107
Luis	101
Lourdes	103
Juan	106
Santiago	102
Verónica	108

Figura 2.1_3 Proyección sobre una relación

Elección

Al igual que la proyección la elección es una operación unitaria. Se representa con la letra griega σ , y su función es elegir aquellas tuplas que cumplan con la condición especificada. Su sintaxis es la siguiente:

$$\sigma_{\text{condición}} (\text{Relación})$$

Ejemplo: ¿que alumnos cursan la carrera cuya clave de carrera es 027?

$$\sigma_{\text{cve_carr}=027} (\text{ALUMNOS})$$

nom_alum	cve_carr
Sandra	027
Lourdes	027
Verónica	027

Figura 2.1_4 Elección de una relación

Producto Cruz

Es una operación binaria ya que actúa sobre dos relaciones, se representa por una cruz X. Esta operación permite combinar o relacionar información de dos o más relaciones. Su sintaxis es la siguiente:

CAPITULO 2. MARCO TEÓRICO Y HERRAMIENTAS

(Relación 1) X (Relación 2)

Esta operación da como resultado una relación que tiene $n \times m$ tuplas y todos los atributos de cada relación. Donde:

n = número de tuplas de Relación 1

m = número de tuplas de Relación 2

Los atributos de la relación resultado son los atributos de las relaciones sobre las que se realiza la operación, el nombre de estos esta precedido por el nombre de la relación de la que provienen, de esta forma se sigue cumpliendo el que los nombre de los atributos sean únicos.

Ejemplo:

ALUMNO no_cta	ALUMNO nom_alum	ALUMNO cve_carr	CARRERA cve_carr	CARRERA Carrera
105	Sandra	027	027	Industrial
105	Sandra	027	032	Computación
105	Sandra	027	029	Electrónica
107	José	032	027	Industrial
107	José	032	032	Computación
107	José	032	029	Electrónica
101	Luis	032	027	Industrial
101	Luis	032	032	Computación
101	Luis	032	029	Electrónica
103	Lourdes	027	027	Industrial
103	Lourdes	027	032	Computación
103	Lourdes	027	029	Electrónica
106	Juan	029	027	Industrial
106	Juan	029	032	Computación
106	Juan	029	029	Electrónica
102	Santiago	029	027	Industrial
102	Santiago	029	032	Computación
102	Santiago	029	029	Electrónica
108	Verónica	027	027	Industrial
108	Verónica	027	032	Computación
108	Verónica	027	029	Electrónica

Figura 2.1_5 Producto cruz

En la figura 2.1_5 se observa que en cada columna se indica el nombre de cada tabla seguido por el nombre del atributo.

Unión

Es una operación binaria, con la cual podemos obtener todos las tuplas de las relaciones involucradas. Su sintaxis es la siguiente: Relación 1 U Relación 2

Supongamos, por ejemplo, que se tiene la relación carrera_opción, la unión de la relación carrera y carrera_opción es otra relación definida sobre el mismo esquema de relación y cuya extensión estará construida por el conjunto de tuplas que pertenezcan a carrera_opción o carrera

Ejemplo: CARRERA U CARRERA_OPCIÓN

Cve_carr	Carrera
027	Industrial
032	Computación
039	Electrónica
015	Geólogo
020	Mecánico
035	Topógrafo

Figura 2.1_6 Unión

Resta

Es una operación binaria, con la cual obtenemos todas las tuplas que se encuentran en la primera relación, pero no se encuentran en la segunda.

Su sintaxis es la siguiente: Relación 1 - Relación 2

En la figura 2.1_7 se muestra un ejemplo de diferencia entre las relaciones carrera y carrera_opcion.

Ejemplo: CARRERA - CARRERA_OPCIÓN

Cve_carr	Carrera
027	Industrial
032	Computación
039	Electrónica

Figura 2.1_7 Resta

Intersección

Esta operación es adicional ya que se puede obtener de las operaciones anteriores. El resultado son aquellas tuplas que se encuentran en ambas relaciones.

$$\text{Relación 1} \cap \text{Relación 2}$$

=

$$\text{Relación 1} - (\text{Relación 1} - \text{Relación 2})$$

Esta operación no es conmutativa: $A - B \neq B - A$

Para realizar las operaciones Unión, Resta e Intersección se debe cumplir con dos condiciones:

- Las relaciones involucradas deben tener el mismo número de atributos.
- Los dominios de los atributos de las relaciones involucradas deben ser los mismos.

Vistas

No es conveniente que todos los usuarios vean el modelo conceptual completo. Las condiciones de seguridad pueden requerir que se "escondan" ciertos datos a algunos usuarios. Aparte de las cuestiones de seguridad, se puede querer crear un conjunto de relaciones que correspondan mejor con una cierta imagen que el usuario tiene de la base de datos, que el modelo conceptual.

Cualquier relación que no es parte del modelo conceptual, pero es visible al usuario como una "relación virtual", se llama vista. Es posible tener un gran número de vistas sobre cualquier conjunto de relaciones reales dado.

Las relaciones del modelo conceptual pueden modificarse, es por ello, que no es posible almacenar una relación correspondiente a una vista. En cambio, una vista debe volverse a calcular para cada consulta que se refiera a ella.

Valores Nulos

El símbolo NULL representa un valor no conocido en la base de datos, un valor que no existe.

Un valor nulo lo representa cero o la cadena "", ya que estos, son valores conocidos. Todas las comparaciones que impliquen valores nulos en las operaciones relacionales son falsas por definición. Las operaciones aritméticas que impliquen nulos, dan como resultado un valor desconocido, es decir NULL.

Llave Primaria (PK)

Una llave primaria es un atributo o conjunto de atributos que identifican a las tuplas de una relación. Cada relación debe contar con una llave primaria. Las características para una llave primaria son las siguientes:

- Única
- Conocible en cualquier tiempo

Las características deseables de una llave primaria son las siguientes:

- Estable
- No descriptiva
- Pequeña y simple

Cuando una llave primaria esta formada por más de un atributo, se denomina **llave primaria compuesta**.

Una relación puede tener más de un atributo o combinación de atributos que pueden actuar como llave primaria. A cada una de estas combinaciones o atributos se le llama llave candidato.

Solamente puede existir una llave primaria por relación, las llaves candidatos restantes se denominan llaves alternas.

Llave Foránea (FK)

Una llave foránea en una relación es un atributo o conjunto de atributos que forman la llave primaria de otra o la misma relación. Las llaves foráneas permiten llevar a cabo operaciones binarias para obtener información de un conjunto de relaciones. Los valores de los atributos que conforman una llave foránea deben corresponder a valores existentes en la llave primaria o, tener valor nulo.

Es conveniente que la llave foránea tenga el mismo nombre y tipo de la llave primaria de la que proviene. La llave primaria de una relación puede estar formada en parte por una llave foránea, en este caso, la llave foránea no puede tener valores nulos.

Integridad

El concepto de integridad es sencillo; los datos deben ser válidos. Ejemplos:

- No pueden existir renglones repetidos en una tabla.
- El valor de un atributo de una tabla debe existir en otra tabla cuando exista una relación entre ambas tablas.
- Debe existir consistencia en los datos según el problema que se trate.

Se definen los siguientes tipos de integridad:

- Integridad de entidades: cada tupla en una relación debe ser única.
- Integridad de dominio: cada valor de un atributo pertenece a un dominio previamente definido.
- Integridad referencial: cada llave foránea debe estar asociada a una llave primaria válida, o debe tener un valor nulo.
- Integridad del negocio: los datos de la Base de datos deben cumplir con las reglas del negocio.

Integridad Referencial

Cada llave foránea debe estar asociada a una llave primaria válida, o debe tener un valor nulo.

Para poder mantener esta relación se pueden considerar las siguientes reglas.

Para llave foránea:

"Insert / Update" (Inserción / Actualización)

- No permitir el insert / update si no existe una llave primaria asociada.
- Asignar valor nulo a la llave foránea sino existe la correspondiente llave primaria.
- Asignar valor de "default" (por omisión) a la llave foránea sino existe la correspondiente llave primaria.
- Si la llave primaria asociada a la llave foránea no existe, dar de alta la tupla asociada a la llave primaria.

- Para "Delete" (Borrado) no se debe considerar acción alguna.

Para llave primaria:

• Update:

- Modificar todas las llaves foráneas asociadas.
- Evitar la modificación si existen llaves foráneas asociadas.

• Delete:

- Borrar todas las llaves foráneas asociadas.
- Evitar el borrado si existen llaves foráneas asociadas.
- Asignar NULL a las llaves foráneas asociadas.
- Asignar un valor de default a las llaves foráneas asociadas.

- Para Insert no se debe considerar acción alguna.

2.2 Metodología de Diseño de Bases de Datos Relacionales

En los últimos años se ha dado un significativo avance tecnológico, una gran difusión de los manejadores de bases de datos relacionales, cuyos productos se soportan en cualquier plataforma, desde los superordenadores a los ordenadores personales. Sin embargo, y a pesar del esfuerzo realizado por numerosos investigadores y expertos del tema, la concepción de una base de datos sigue siendo una tarea larga y costosa.

Estas dificultades inherentes al diseño de una base de datos han de afrontarse con procedimientos ordenados y metódicos en el marco de una metodología de diseño de base de datos. En el proceso de diseño de una base de datos se distinguen tres grandes fases:

- **Diseño conceptual**, cuyo objetivo es obtener una buena representación de los recursos de información de la empresa, con independencia de usuarios o aplicaciones en particular, y fuera de consideraciones sobre eficiencia del ordenador.
- **Diseño lógico**, cuyo objetivo es transformar el esquema conceptual obtenido en la etapa anterior, adaptándolo al modelo de datos en el que se apoya el manejador de base de datos que se va a utilizar.
- **Diseño físico**, cuyo objetivo es conseguir una instrumentación, lo más eficiente posible, del esquema lógico.

Es de gran utilidad el obtener, en una primera fase del diseño, un esquema conceptual independiente de las características de los modelos convencionales, como el modelo relacional, y que recogiese la semántica del mundo real.

Un problema que se suele presentar en el diseño de bases de datos es la comunicación entre el diseñador y el usuario, este último conoce bien el dominio de aplicación, lo que en general no ocurre con el diseñador, pero muchas veces no sabe expresarlo de forma correcta y, menos aún, precisa. La utilización del modelo E/R, en una primera fase del diseño, facilita el diálogo entre el informático, conocedor de las técnicas de estructuración de los datos pero ajeno al dominio de la aplicación, y el usuario, que suele conocer bien su mundo real pero que no es capaz de describirlo bajo las premisas que impone un modelo convencional.

El modelo E/R es sencillo, pero a la vez es suficientemente potente, permite entablar un diálogo entre el usuario y el diseñador que facilitará que se despejen dudas y se aclaren aspectos del universo del sistema a modelar. Se facilitará así la colaboración de los especialistas con los usuarios, de manera que estos últimos pueden participar activamente, e incluso ser protagonistas, en el diseño.

Se puede representar esquemáticamente las dos primeras fases (diseño conceptual y diseño lógico) de la metodología como se indica en la figura 2.2_1. en la cual aparece el proceso de diseño de una aplicación industrial; el diseñador observa el mundo real bajo unos ciertos objetivos (Universo del sistema) y, apoyándose en una primera etapa en el modelo E/R, llega a

un esquema conceptual, al cual se le aplicará un conjunto de reglas a fin de transformarlo en una estructura relacional (conjunto de tablas).

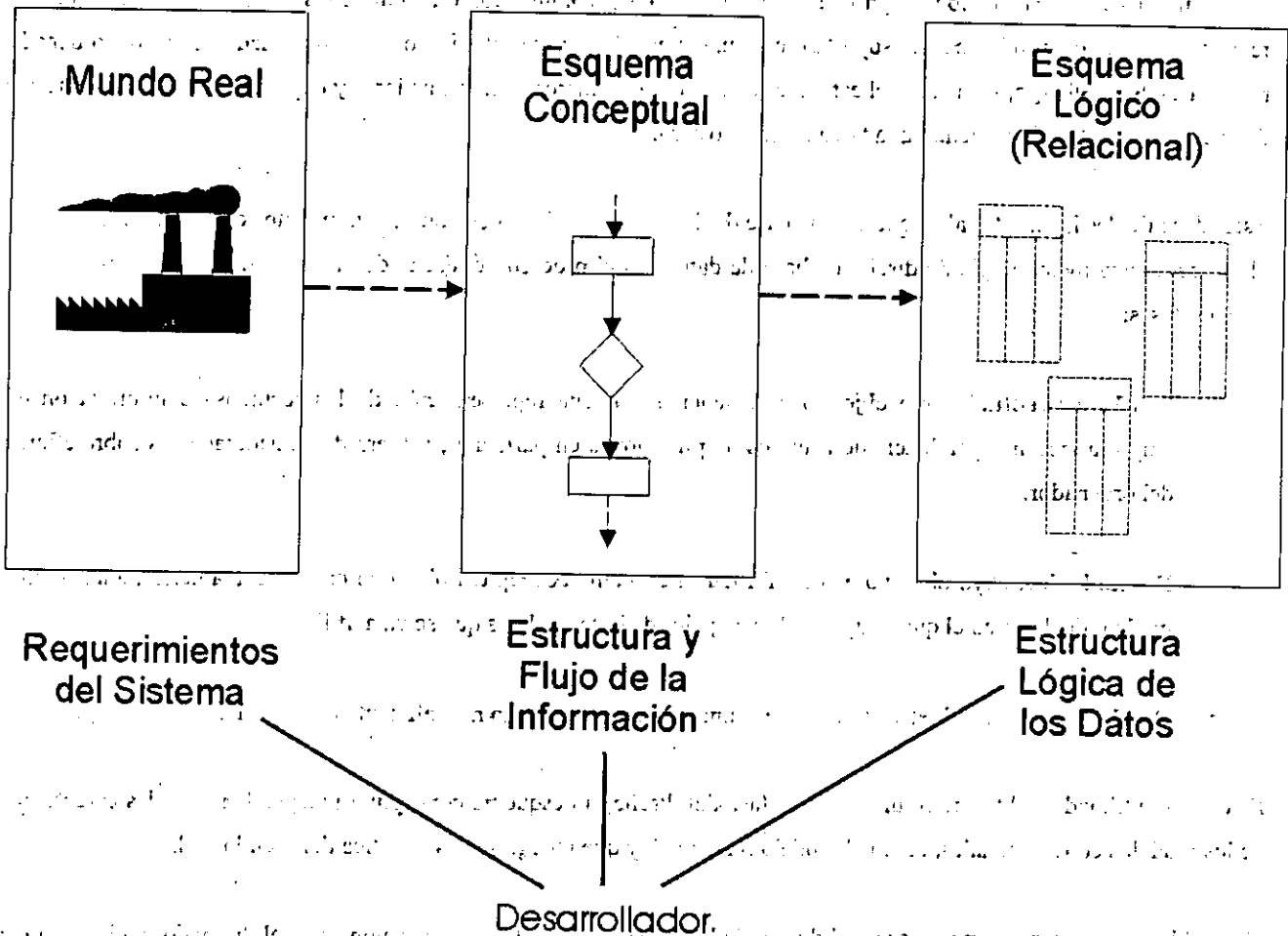


Figura 2.2_1 Representación gráfica de la metodología propuesta para el diseño de Bases de datos relacionales

Una vez adaptada la estructura relacional a las restricciones impuestas por el producto concreto y a las exigencias relativas a su eficiencia, será posible almacenar en la estructura resultante los datos necesarios para la gestión de dicha aplicación.

2.2.1 Transformación del Esquema Conceptual al Relacional

La transformación del esquema consiste básicamente en la generación del Modelo Relacional de la base de datos a partir del Modelo Conceptual. El Modelo Conceptual de la base de datos es independiente del Sistema Manejador de Base de Datos (DBMS) a ser utilizado y también del Modelo de Base de Datos utilizado por este. El diseño relacional se lleva a cabo en dos etapas:

- Mapeo del Diagrama Entidad Relación (DER) a tablas.
- Redefinición del diseño tomando en cuenta consideraciones de rendimiento.

El objetivo de la primera etapa es generar un esquema de tablas con las siguientes características:

- Esquema normalizado (redundancia minimizada).
- Sin pérdidas de información, es decir que toda la información del Modelo Conceptual se pueda obtener relacionamente.

Esta primera etapa se puede llevar a cabo independientemente del DBMS a ser utilizado.

El objeto de la segunda etapa es generar un esquema de tablas que proporcione las condiciones para que las aplicaciones de la base de datos tengan un alto rendimiento. En esta segunda etapa se deben considerar las aplicaciones críticas de la base de datos y algunas características del DBMS a ser utilizado.

Representación de Entidades

- Cada entidad independiente representa una tabla.
- La llave primaria de tablas independientes no contiene llaves foráneas.
- Cada atributo de una entidad representa una columna de la tabla.
- Los atributos con cardinalidad opcional representan columnas que aceptan valores nulos.

ASISTENTE		
<u>NUMEROCTA</u>	<pk>	not null
NOMBRE		not null
DIRECCION		not null
TELEFONO		null
EMPRESA		not null
EDAD		not null

Figura 2.2_2 Representación de una entidad

Representación de Relaciones

Las relaciones representadas en el Modelo Conceptual, se representan con llaves foráneas en las tablas relacionadas, o se generan tablas que mantienen la información de la relación.

Relaciones N:1 (muchos a uno)

En relaciones N:1, no importa la cardinalidad de la misma, la llave foránea se coloca en la entidad con el grado N. La cardinalidad mínima en la entidad de grado N, determina si la llave foránea acepta o no valores nulos.

Relación 1:N con cardinalidad mínima 1:1

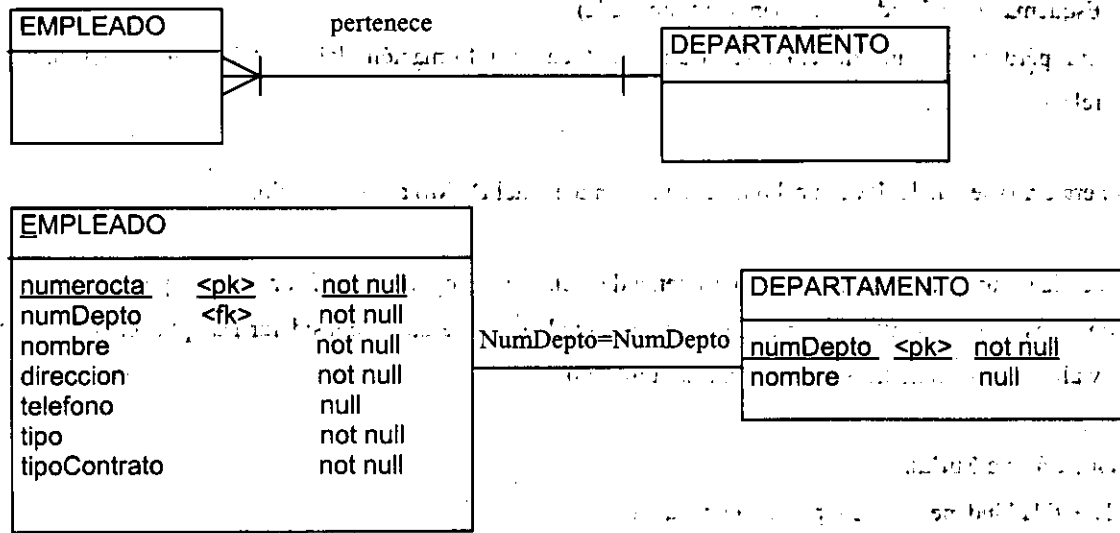


Figura 2.2_3 Relación 1:N con cardinalidad mínima 1:1

Relación 1:N con cardinalidad mínima 1:0

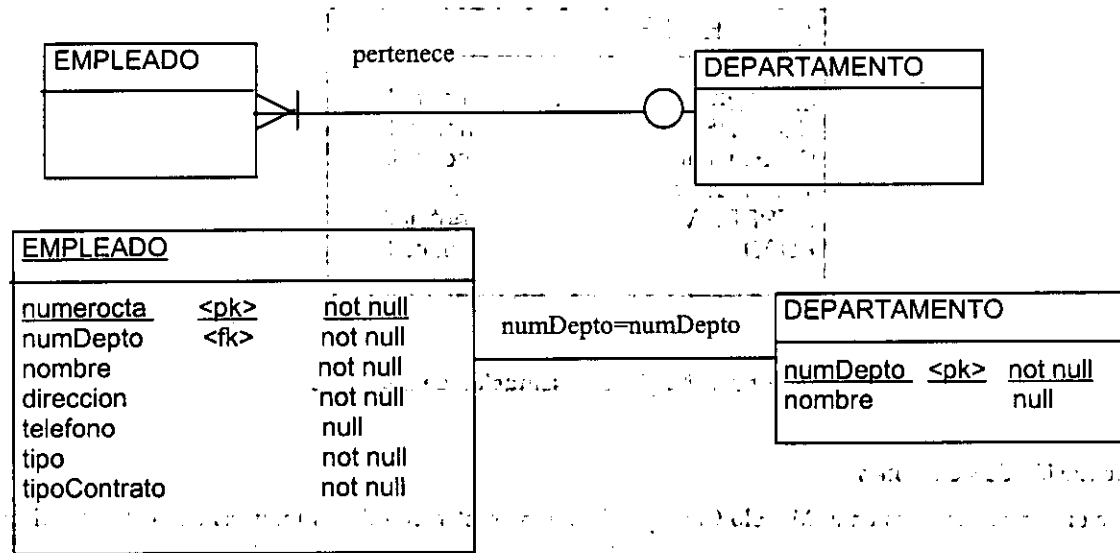


Figura 2.2_4 Relación 1:N con cardinalidad mínima 1:0

Relaciones 1:1 (uno a uno)

La llave foránea se coloca en la entidad con cardinalidad mínima cero:

- Relación 1:1 con cardinalidad mínima 1:0

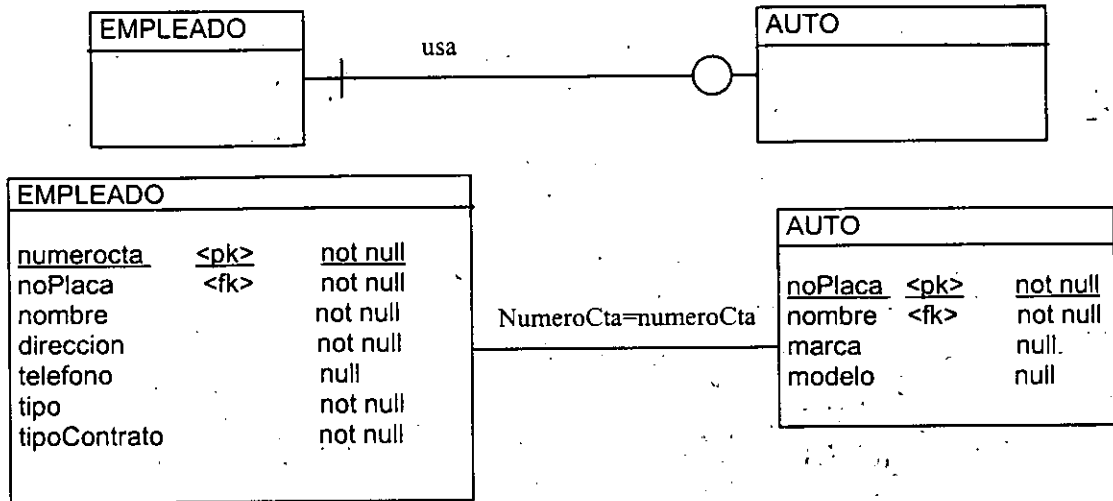


Figura 2.2_5 Relación 1:1 con cardinalidad mínima 1:0

- Relación 1:1 con cardinalidad mínima 0:1

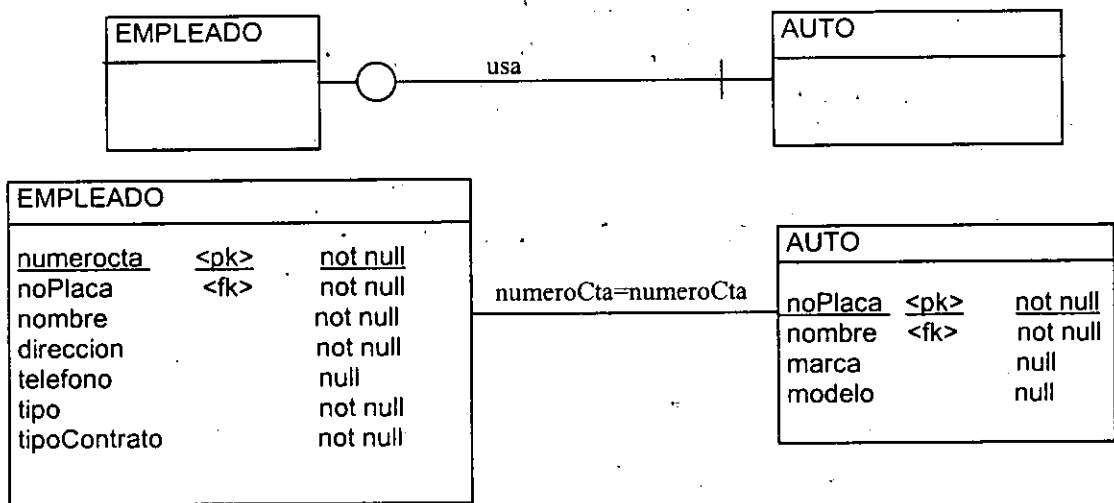


Figura 2.2_6 Relación 1:1 con cardinalidad mínima 0:1

Cuando la cardinalidad mínima es 0:0 se puede colocar la llave foránea en cualquiera de las tablas relacionadas.

Relaciones N:N (muchos a muchos)

En relaciones N:N no importando la cardinalidad mínima, se genera una tabla de relación, cuya llave primaria esta compuesta de las llaves primarias de las tablas que forman parte de la relación.

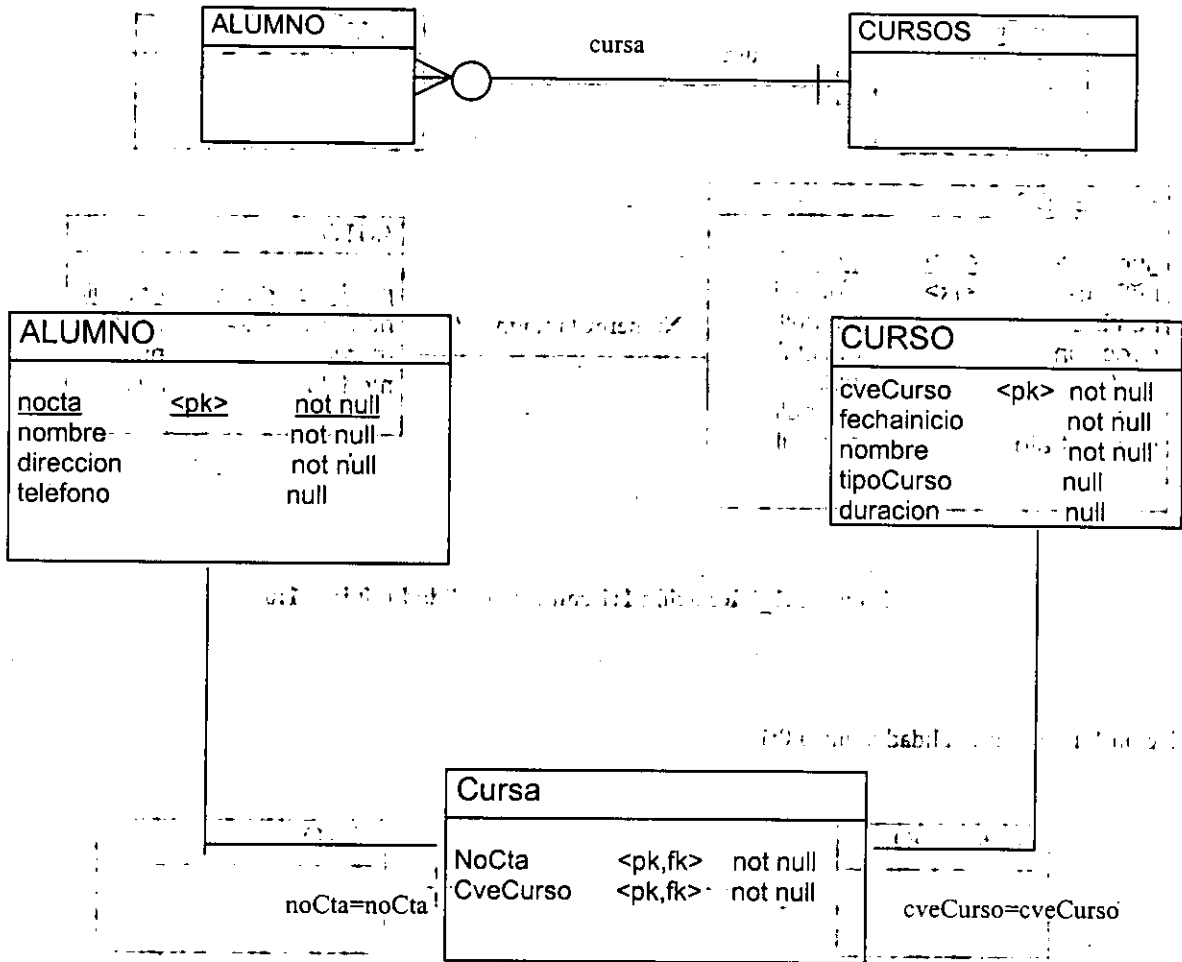


Figura 2.2_7 Relación muchos a muchos

Para relaciones N:N que se modelaron como dos relaciones de dependencia la entidad generada será una tabla, cuya llave primaria contiene las llaves primarias de las entidades que se relacionan:

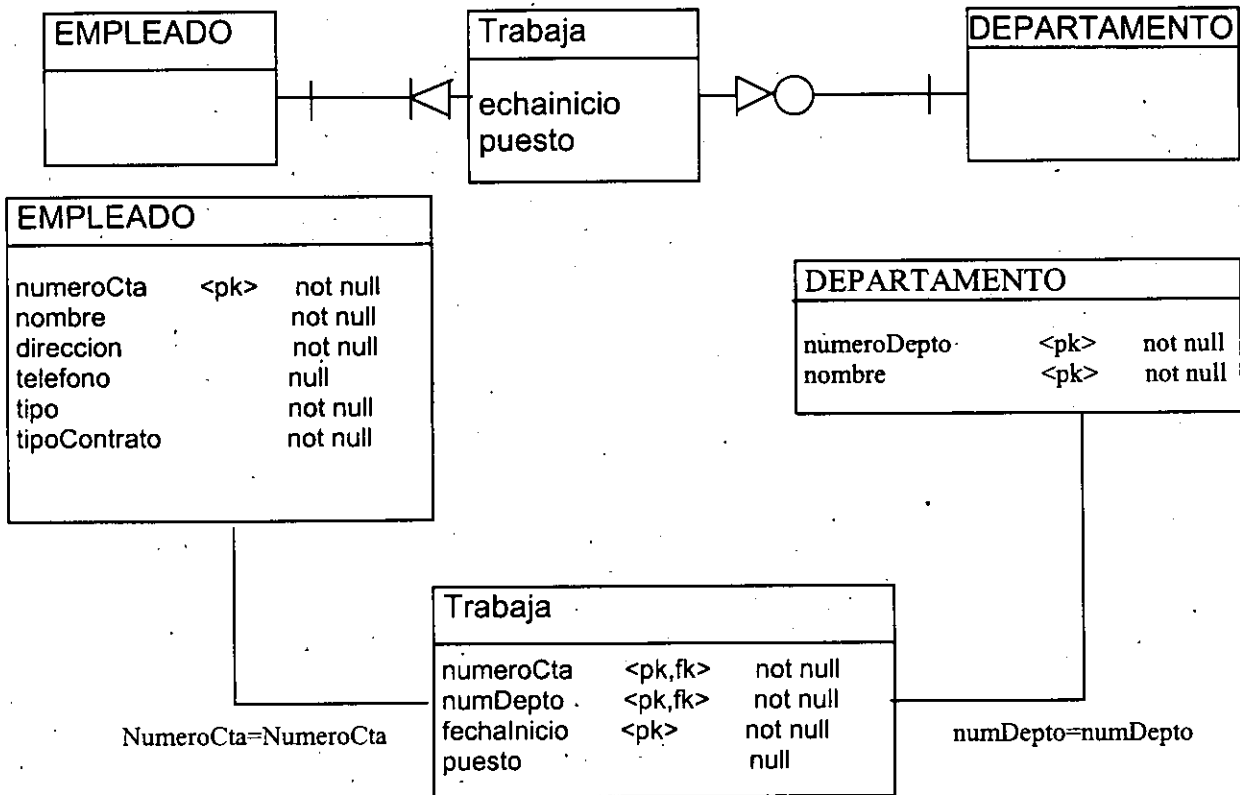


Figura 2.2_8 Relación N:N

Relaciones de Dependencia

Las entidades dependientes, las cuales participan en una relación de dependencia, tienen una llave primaria que contiene la llave primaria de la entidad de la que depende.

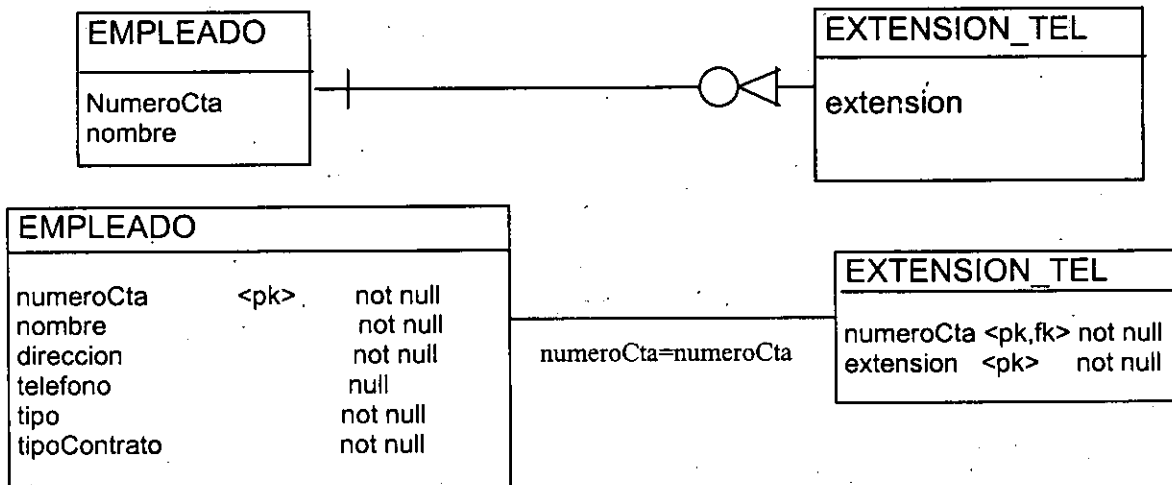


Figura 2.2_9 Relación de dependencia

Relaciones Recursivas

En las relaciones recursivas es importante considerar la cardinalidad máxima de la relación, dependiendo de ella se determina si es necesaria una llave foránea en la tabla o es necesario crear una tabla de relación, de acuerdo a las reglas indicadas para las relaciones binarias.

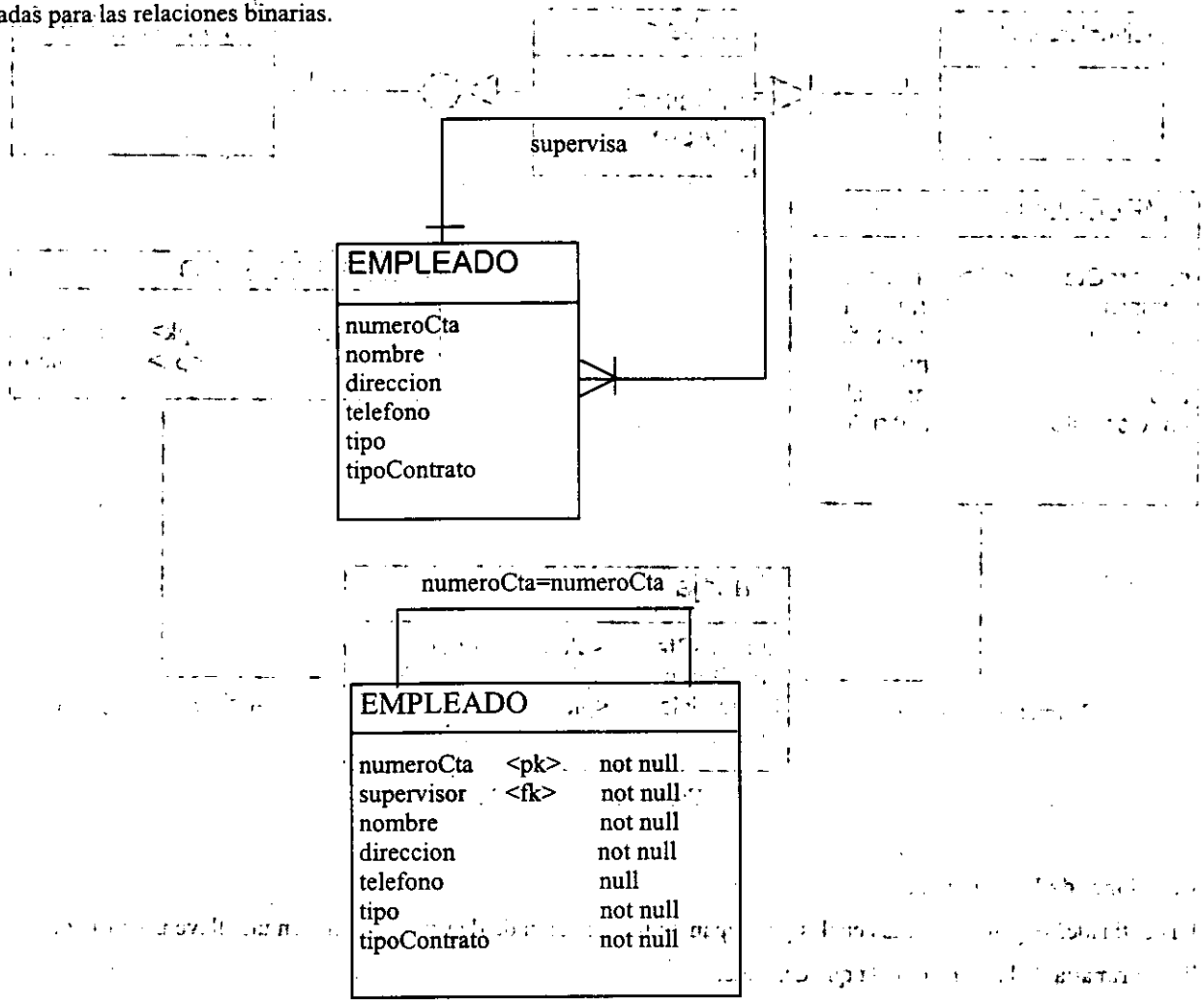


Figura 2.2_10 Relaciones recursivas

Relaciones N - arias

No importan mucho los grados de asociación o de pertenencia, de cualquier forma se debe representar la Asociación y formar una llave primaria con la concatenación de los identificadores de las Entidades que participan.

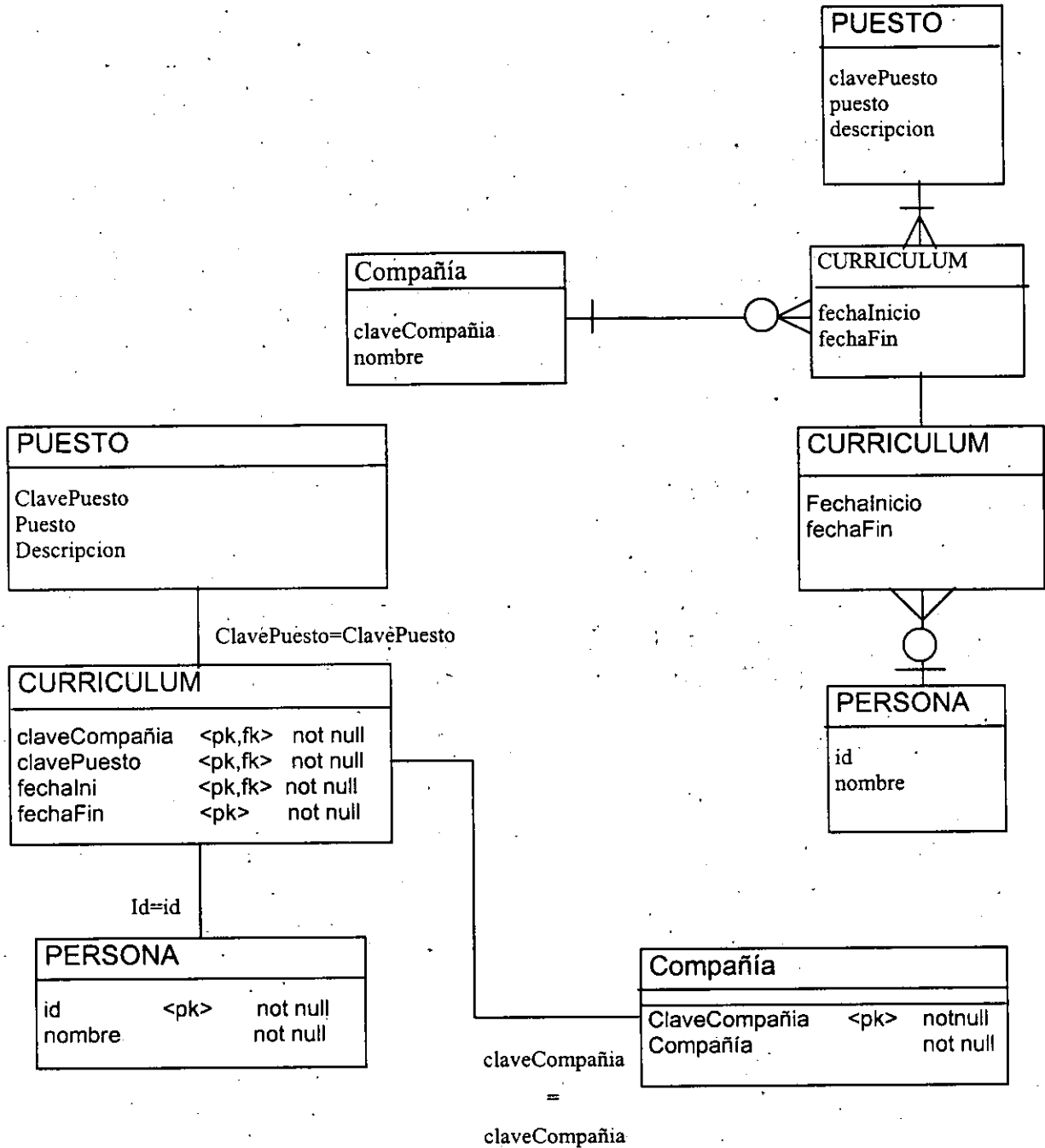


Figura 2.2_11 Relaciones N-Arias

Súper-entidades y Sub-entidades

La llave primaria, atributos y relaciones de una súper-entidad también lo son de las sub-entidades, pero en forma indirecta. Lo contrario no aplica.

La súper-entidad es una tabla que contiene la llave primaria y atributos comunes, además de las relaciones comunes a las sub-entidades. Cada una de las sub-entidades es una entidad dependiente que se representa mediante una tabla que contiene la llave primaria de la súper-entidad y los atributos propios de la sub-entidad. Cada sub-entidad mantiene las relaciones propias de la misma.

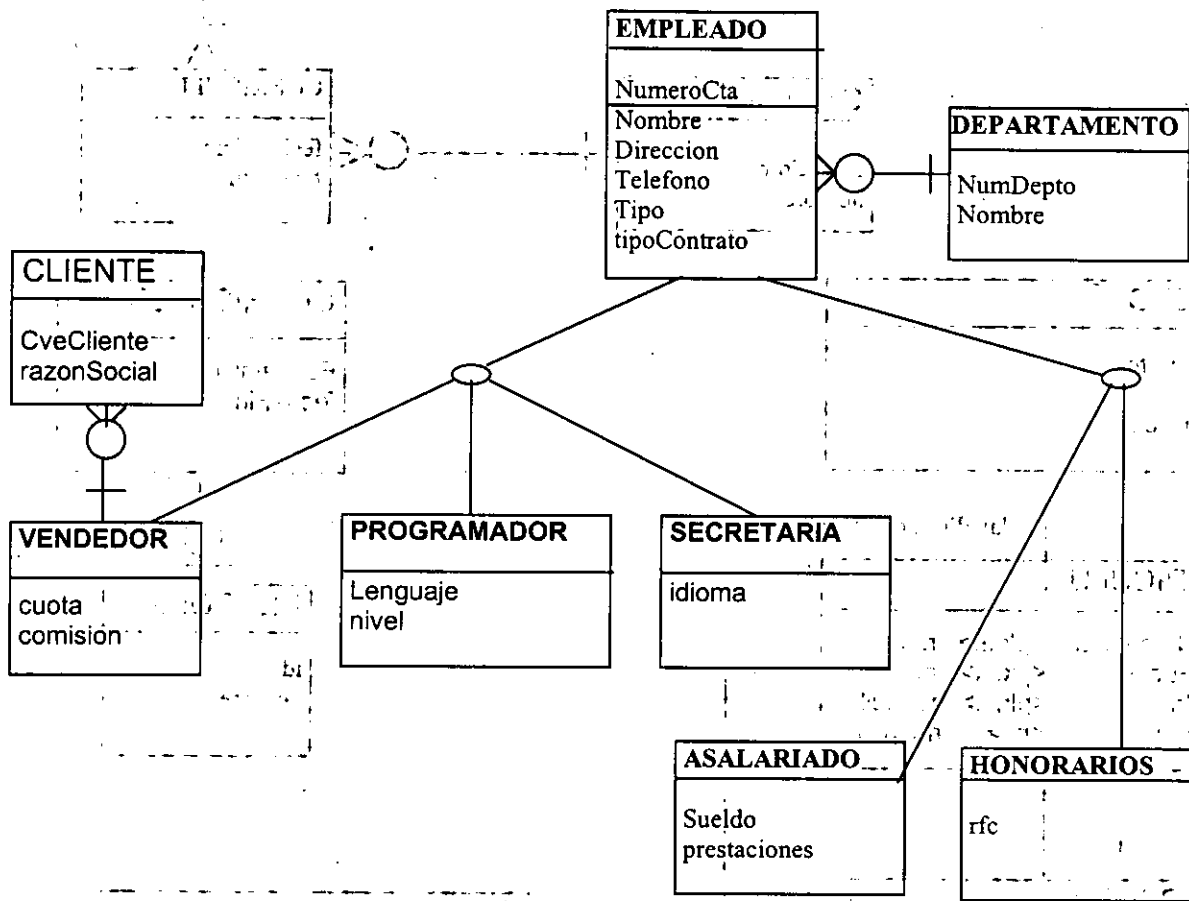


Figura 2.2_12 Representación de sub-entidades

El diseño de sub-entidades se puede llevar a cabo de tres formas:

1. Generalización

Las sub-entidades no generan tablas, solamente se genera una tabla correspondiente a la súper-entidad, la cuál tiene todos los atributos comunes, así como los particulares a las sub-entidades, estos últimos deben indicarse como NULL (aceptan valores nulos).

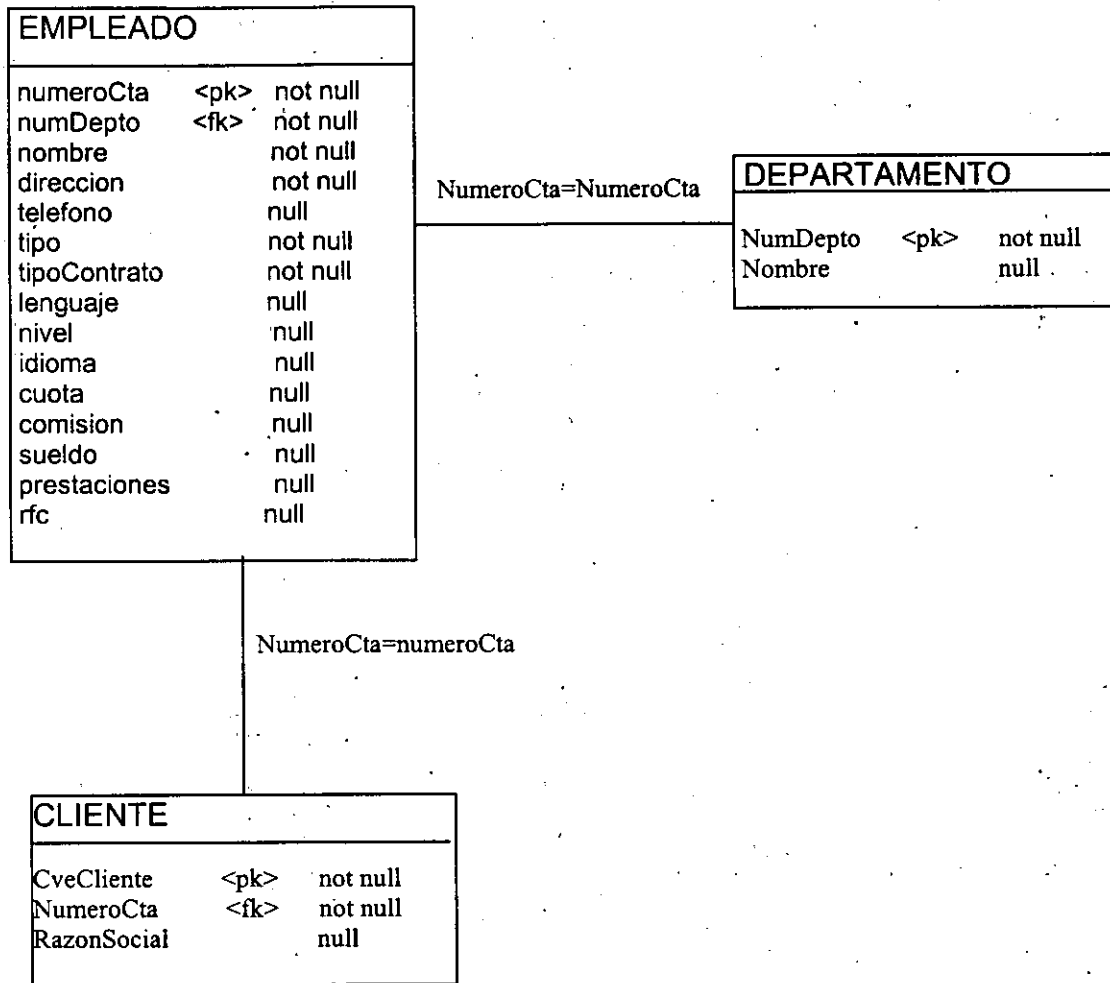


Figura 2.2_13 Generalización de una sub-entidad

2. Especialización Total

Se generan tablas para cada una de las sub-entidades, las cuales heredan los atributos, la llave primaria y las relaciones de la súper-entidad. No se crea una tabla para la súper-entidad.

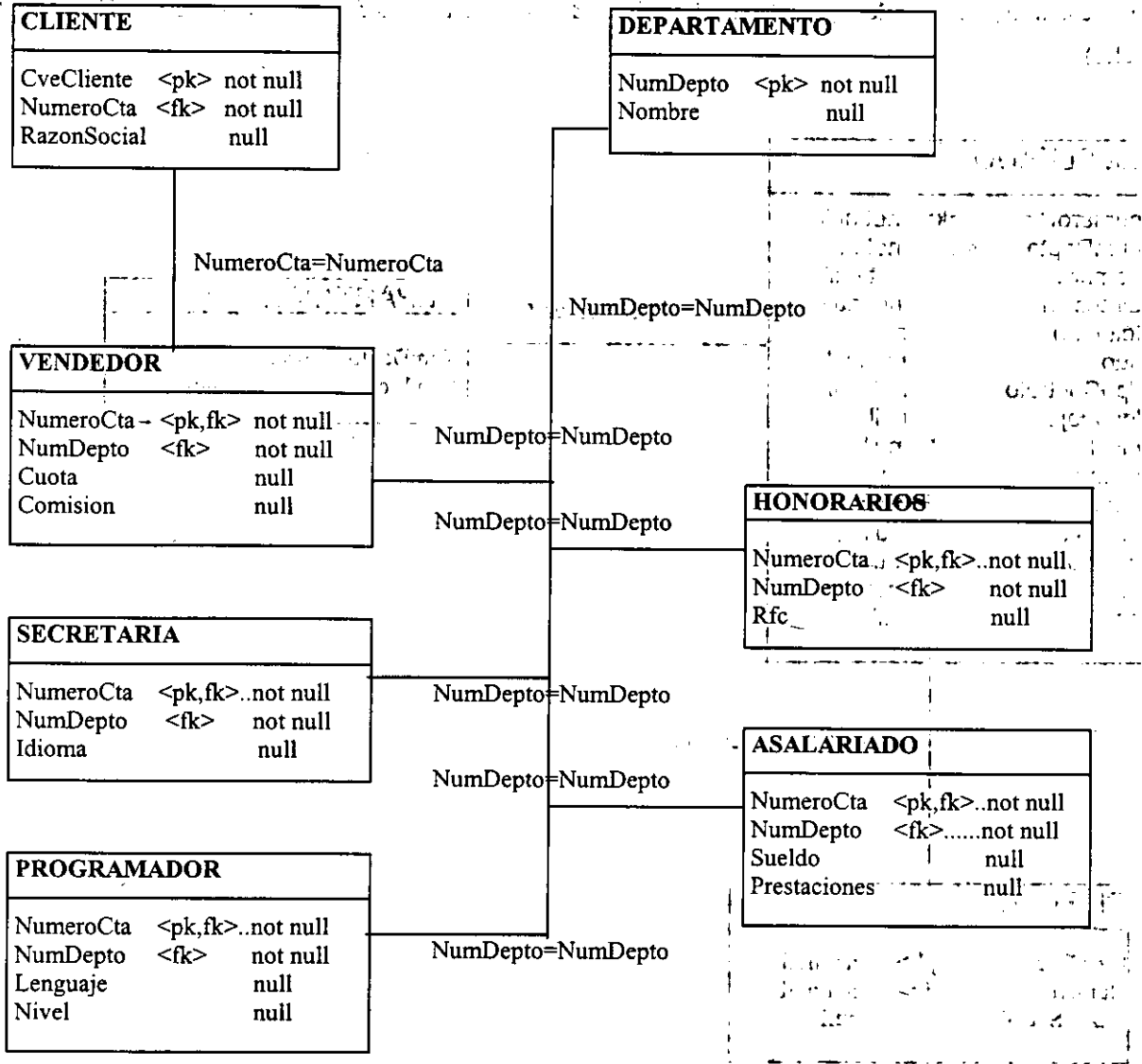


Figura 2.2_14 Especialización de una sub-entidad

3. Especialización Parcial

Para cada sub-entidad se genera una tabla cuyos atributos son aquellos particulares a la sub-entidad. Esta tabla tiene como llave primaria, la llave primaria de la súper-entidad y mantiene solamente las relaciones propias a ella.

Para la súper-entidad se genera una tabla con los atributos y relaciones comunes a las sub-entidades.

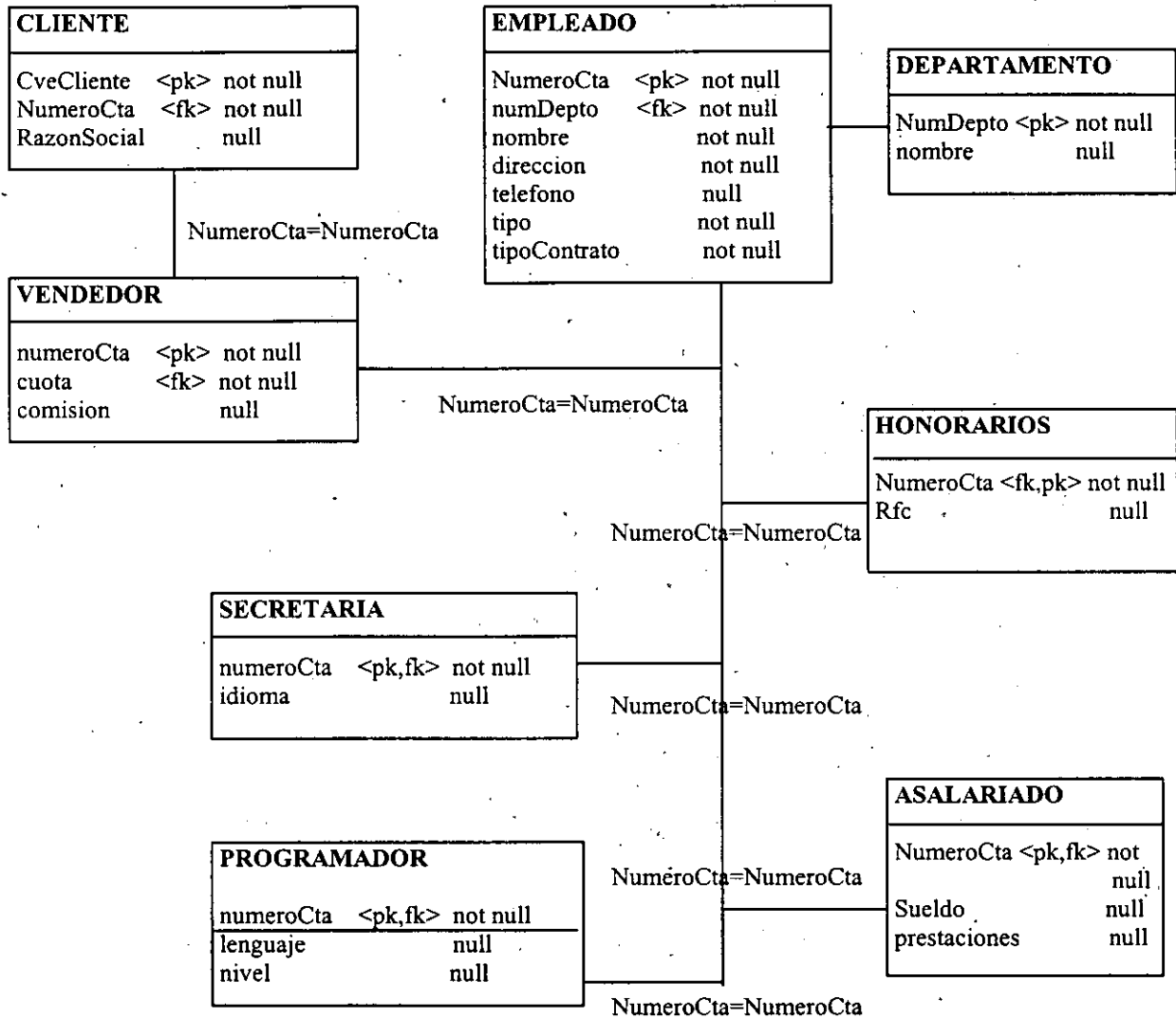


Figura 2.2_15 Especialización parcial de una sub-entidad

2.2.2 Normalización

La normalización es una técnica desarrollada para asegurar que las estructuras de datos sean eficientes. Los beneficios de la Normalización son:

- Minimiza la redundancia
- Libera de dependencias indeseables de inserción, borrado y actualización.
- Minimiza la reestructuración de datos cuando se introduce algo nuevo. Se mejora la independencia de datos, permitiendo que las extensiones a la base de datos tengan poco o ningún efecto sobre los programas o aplicaciones que tiene acceso a ella.
- No se introducen restricciones artificiales a las estructuras de datos

Aunque se han definido más estados de normalización, sólo se han aceptado ampliamente tres: Estos se conocen como primera, segunda y tercera formas normales, o 1NF, 2NF Y 3NF respectivamente. El normalizar es un proceso ascendente, en el que se parte de un universo de relaciones y atributos, y se avanza de forma en forma hasta llegar a la tercera forma normal.

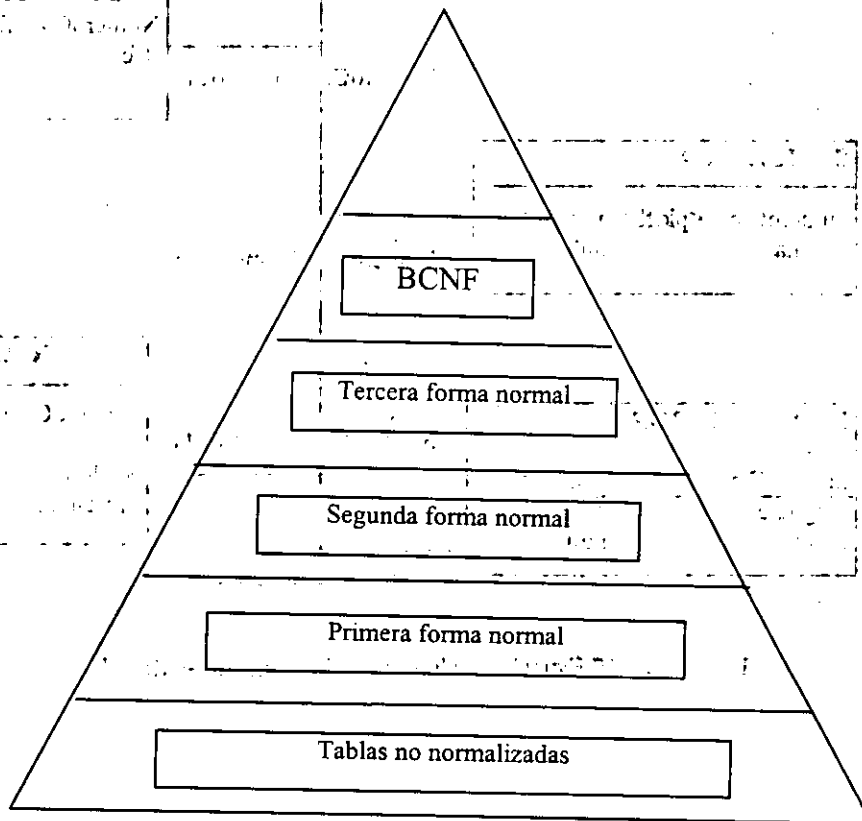


Figura 2.2_16 Proceso de normalización

Primera Forma Normal

Para que una relación R cumpla con la primera forma normal debe cumplir lo siguiente:

Las celdas de la tabla deben poseer valores simples y no se permiten grupos ni arreglos repetidos como valores. Todos los ingresos en cualquier columna (atributo), deben ser del mismo tipo. Cada columna debe tener un nombre único, el orden de las columnas en la tabla no es importante. Dos hileras en una tabla no deben ser idénticas; aunque el orden de las hileras no es importante.

Segunda Forma Normal

Una relación está en segunda forma normal si todos sus atributos que no son claves dependen por completo de la clave. De acuerdo con esta definición, cada relación R, que tiene un atributo único como clave, está en segunda forma normal. La clave es sólo un atributo, en forma predeterminada, cada atributo que no es clave depende por completo de la clave; no puede haber dependencias parciales. La segunda forma normal sólo hace referencia a relaciones con claves compuestas.

Tercera Forma Normal

Una relación R está en tercera forma normal si está en segunda forma normal y no tiene dependencias transitivas.

Las etapas de normalización se muestran adelante con un ejemplo dada la relación no normalizada:

ORDEN (# orden, fecha, proveedor, nombre_proveedor, dirección_proveedor, #producto, descripción_producto, Cantidad_producto, Precio_total_producto, Precio_total_orden)

ORDEN

#orden	Fecha	#proveedor	Nombre proveedor	Dirección proveedor	#producto	Descripción producto	Cantidad producto	Precio_total_producto	Precio_total_orden
10	1999051	10	Rodolfo H.	Azucen #12	12	Impresora	2	\$4000.00	\$8000.00
15	1999061	15	Carlos L.	Arizpe #23	14	Teclado	5	\$300.00	\$1500.00
20	1999081	12	Luis S.	Norte #45	23	CPU	2	\$10000.00	\$20000.00

Figura 2.2_17 Tabla ejemplo para normalizar

Primera Forma Normal

Un registro en primera forma normal no incluye grupos repetidos. Es decir cada uno de sus campos debe tener un solo valor.

En la relación ORDEN , se observa que para una misma orden habrá varios productos, por lo que #producto y otros atributos serán grupos repetidos. En 1NF habría que separar:

ORDEN (#orden, fecha, #proveedor, nombre_proveedor, dirección_proveedor, Precio_total_orden)

#orden	Fecha	#proveedor	Nombre proveedor	Dirección proveedor	Precio_total_orden
10	19990512	10	Rodolfo H.	Azucen #12	\$8000.00
15	19990612	15	Carlos L.	Arizpe #23	\$1500.00
20	19990812	12	Luis S.	Norte #45	\$20000.00

Figura 2.2_18 Primera forma normal

PRODUCTO_ORDENADO (#orden, #producto, descripción_producto, Precio_producto, cantidad_producto, precio_total_producto)

#orden	#producto	Descripción_producto	Precio_producto	Cantidad_producto	Precio_total_orden
10	12	Impresora	\$4000.00	2	\$8000.00
15	14	Teclado	\$300.00	5	\$1500.00
20	23	CPU	\$10000.00	2	\$20000.00

Figura 2.2_19 Primera forma normal

Segunda Forma Normal

Cada atributo depende de la totalidad de la llave, y no de sólo de una parte de ella.

Se puede observar en PRODUCTO_ORDENADO que descripción_producto depende sólo de #producto, y no tiene que ver con #orden. En 2Nf quedaría:

ORDEN (#orden, fecha, #proveedor, nombre_proveedor, dirección_proveedor, precio_total_orden)

#orden	Fecha	#proveedor	Nombre proveedor	Dirección proveedor	Precio_total_orden
10	19990512	10	Rodolfo H.	Azucen #12	\$8000.00
15	19990612	15	Carlos L.	Arizpe #23	\$1500.00
20	19990812	12	Luis S.	Norte #45	\$20000.00

Figura 2.2_20 Segunda forma normal

PRODUCTO (#producto, descripción_producto, precio_producto)

#producto	Descripción_producto	Precio_producto
12	Impresora	\$4000.00
14	Teclado	\$300.00
23	CPU	\$10000.00

Figura 2.2_21 Segunda forma normal

PRODUCTO_ORDENADO (#orden, #producto, cantidad_producto, precio_total_producto)

#orden	#producto	Cantidad_producto	Precio_total_orden
10	12	2	\$8000.00
15	14	5	\$1500.00
20	23	2	\$20000.00

Figura 2.2_22 Segunda forma normal

Tercer Forma Normal

Todos los atributos dependen solamente de la llave y no de otros atributos no-llave.

En la relación ORDEN se presenta este problema:

Nombre_proveedor y dirección_proveedor dependen de #proveedor

De modo que las tablas en 3NF quedarían como:

ORDEN (#orden, fecha, #proveedor, precio_total_orden)

#orden	Fecha	#proveedor	Precio_total_orden
10	19990512	10	\$8000.00
15	19990612	15	\$1500.00
20	19990812	12	\$20000.00

Figura 2.2_23 Tercera forma normal

PROVEEDOR (#proveedor, nombre_proveedor, dirección_proveedor)

#proveedor	Nombre proveedor	Dirección proveedor
10	Rodolfo H.	Azucen #12
15	Carlos L.	Arizpe #23
12	Luis S.	Norte #45

Figura 2.2_24 Tercera forma normal

PRODUCTO (#producto, descripción_producto, precio_producto)

#producto	Descripción producto	Precio producto
12	Impresora	\$4000.00
14	Teclado	\$300.00
23	CPU	\$10000.00

Figura 2.2_25 Tercera forma normal

PRODUCTO_ORDENADO (#orden, #producto, cantidad_producto, precio_total_producto)

#orden	#producto	cantidad_producto	Precio_total orden
10	12	2	\$8000.00
15	14	5	\$1500.00
20	23	2	\$20000.00

Figura 2.2_26 Tercera forma normal

Diseño físico de bases de datos

Implementar una Base de Datos es transformar el Modelo Lógico de la misma en una serie de estructuras físicas que la representen. El diseño Físico es el proceso de determinar las estructuras de datos y mecanismos a ser utilizados para la implementación de la Base de Datos. El Diseño Físico de la Base de Datos es particular a un DBMS. Para llevar a cabo un buen Diseño Físico es conveniente conocer las aplicaciones críticas que tendrán acceso la base de datos.

El Diseño de la Base de Datos incluye conceptos de:

- Distribución de la Base de Datos.
- Almacenamiento de la Base de Datos.
- Tipos de datos.
- Estructuras de acceso directo a los datos.
- Integridad.
- Esquema de seguridad.
- Esquema de respaldo y recuperación.

2.3 SISTEMAS OPERATIVOS

2.3.1 Windows 9x / Windows NT (Server y Workstation)

Un sistema operativo es un grupo de programas desarrollados para simplificar la labor del programador, cuando ésta necesita hacer uso de ciertos recursos que ofrece la computadora. A estos programas se les llama servicios; éstos pueden abarcar el manejo de unidades de disco, el teclado, la pantalla, la impresora, el módem, la conexión a la red y otros periféricos. También pueden ser servicios de manejo de la memoria, las prioridades en las interrupciones que hagan los periféricos cuando necesitan ser atendidos, etc.

Hoy por hoy es común que se tenga una PC en casa, que puede estar ejecutando un sistema operativo como Windows 95/98. Es necesario saber exactamente que ofrece el sistema operativo, los mecanismos de seguridad en el manejo de archivos. Por otra parte, también hay que conocer las posibilidades que brinda el manejo de los equipos periféricos que se conectan al procesador, la capacidad para detectar problemas con estos equipos y la posibilidad de sustitución de los equipos periféricos por otros más eficientes o de más capacidad. A continuación se presentan las características de Windows98 y NT lo que nos permitirá observar las ventajas que tienen cada uno de ellos como sistema operativo.

2.3.1.1 Windows98

Windows98 está diseñado como un sistema operativo de consumo. Ya que Microsoft sigue promoviendo Windows NT para las empresas, aunque tiene soporte más limitado para aplicaciones de DOS y para dispositivos que Windows 95/98. Windows NT es una mejor opción que Windows98 si no se necesita el DOS ni soporte de dispositivos y se desea un SO inherente más estable.

Características de Windows98

- Interfaz.- Windows98 agrega a su escritorio muchas características de integración con el Web, como la Barra de Canales.
- Avances relacionados con las LAPTOPS.- Mejor desempeño de la vida de la batería.
- Multimedia.- Windows98 da un impulso a éste desarrollo con su Broadcast Architecture, que permite ver TV en la PC e incluye muchas otras mejoras de video.
- Herramientas de Desarrollo.- El WSH (Windows Scripting Host) quizá sea el más popular, pero el WDM (Windows Driver Model) tal vez sea más importante, ya que permite a los desarrolladores escribir sólo un conjunto de controladores tanto para Windows98 como para Windows NT.



Figura 2.3_1 Escritorio del Sistema Operativo Windows98

El WSH es el lenguaje de procesamiento por lotes que Windows nunca había tenido. Aunque Microsoft lo ha incluido con el Windows NT Option Pack (una adición para Windows NT Server), el WSH no tenía todas las capacidades que ahora posee Windows98.

En éste último, el WSH puede programar objetos arbitrarios del sistema, como objetos especiales en el shell y el sistema de archivos. Internet Explorer 4.0 incluye el mismo mecanismo de programación.

El programa puede manipular cualquier control programable de ActiveX en el sistema.

El WDM es un modelo para controladores de dispositivos que puede trabajar tanto en Windows98 como en Windows NT. Para Microsoft es importante cambiar a los fabricantes de hardware al WDM a fin de eliminar las brechas de soporte de dispositivos en Windows NT, dado que los controladores de WDM son los mismos archivos para ambos sistemas operativos. El WDM también hace una transición de Windows98 a una versión de Windows NT mucho más suave tanto para los usuarios como para los fabricantes de hardware.

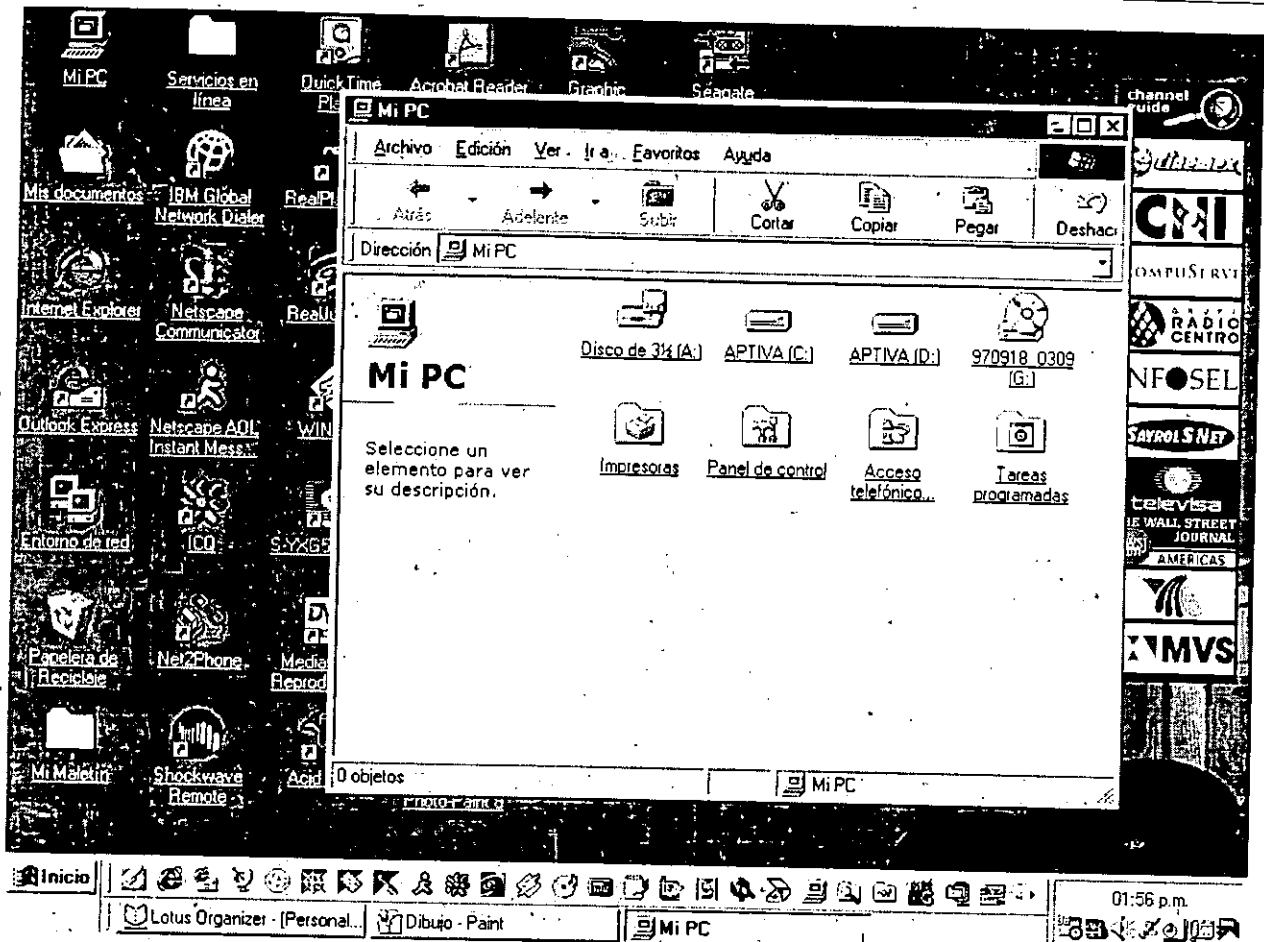


Figura 2.3 2 Dispositivos Disponibles en Windows98.

El WDM soporta cuatro clases de dispositivos. La clase de flujo soporta captura de video, decodificadores MPEG, audio, DVD-ROM. Los controladores de clase HID soportan los dispositivos de entrada como "joysticks" (palancas de mando) y teclados, los controladores de clase USB y los de clase IEEE 1394 soportan hardware controlador para esos buses.

El WDM no soporta todas las clases de dispositivos (por ejemplo, tarjetas para gráficos), pero la mayor parte de los dispositivos no WDM tienen buen soporte de Windows NT.

Se pueden escribir controladores WDM en dos formas. Para la mayor parte de los controladores, puede apalancar un controlador WDM proporcionado por Microsoft y tener un minicontrolador específico para dispositivos, o bien un conjunto de parámetros para que el controlador los emplee al controlar el dispositivo. Si un dispositivo no se adapta a una de las clases predefinidas, será necesario escribir un controlador más tradicional específico de Windows 9x ó Windows NT.

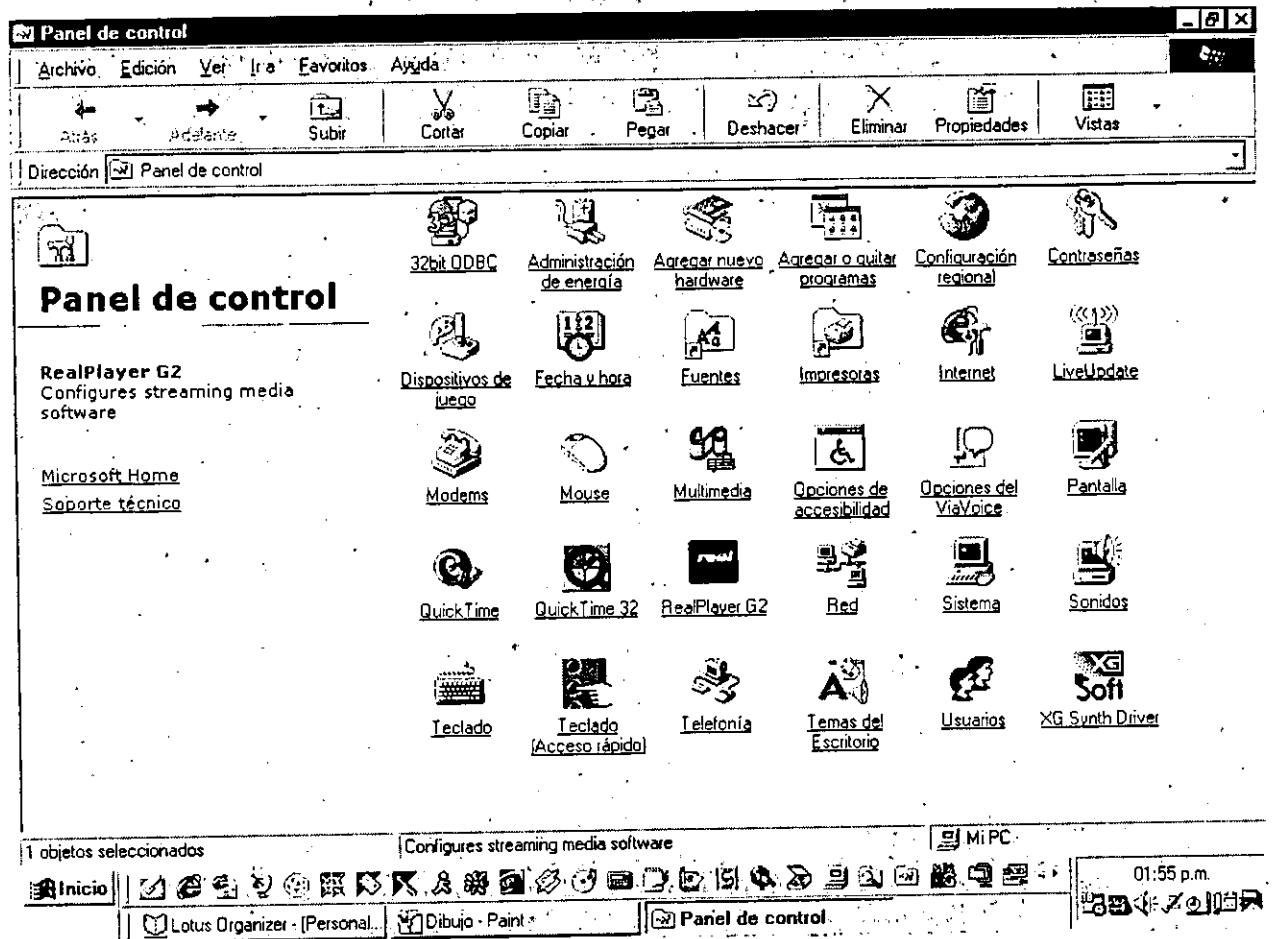


Figura 2.3_3 Panel de Control de Windows98

- **Hardware.**- Windows98 hace mucho más fácil utilizar algunos de los nuevos dispositivos de hardware, como unidades de DVD, escáneres y teclados USB (Universal Serial Bus) y cámaras digitales Fire Wire (IEEE 1394).

USB y Fire Wire son especificaciones que permiten hacer cadenas de margarita con múltiples periféricos hacia la PC por medio de cables externos, sin agregar ningún hardware. Son similares a SCSI, pero pueden soportar muchos dispositivos. USB puede manejar hasta 127 dispositivos, y trabajar a velocidades que alcanzan los 12 Mbps. Fire Wire puede manejar 63 dispositivos en una sola PC y es mucho más rápido que el USB. Los puertos Fire Wire actuales más rápidos trabajan a velocidades de hasta 200 Mbps.

Windows98 también soporta ACPI (Advanced Configuration and Power Interface) y APM (Advanced Power Management) especificaciones para controlar la corriente en la PC con mayor eficiencia.

El mejor soporte para controlador integrado para estos nuevos dispositivos de hardware debe facilitar la instalación de hardware, de modo que las PC nuevas con el soporte apropiado pueden maximizar su potencial.

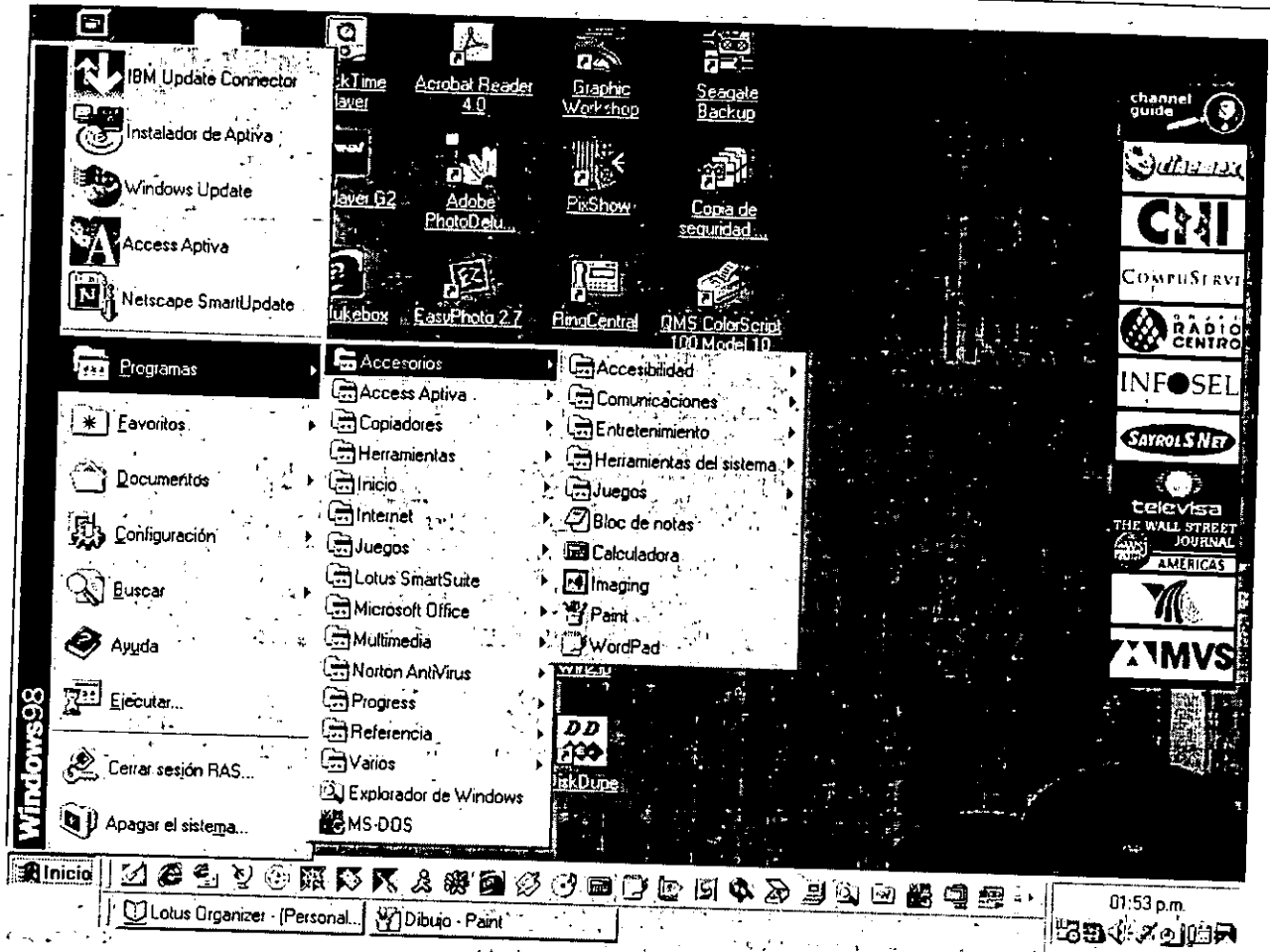


Figura 2.3_4 Inicio de Programas en Windows98

- **Desempeño.-** Con objeto de aprovechar la transferencia más rápida de aplicaciones, es preciso que se convierta el disco a FAT 32, el cual almacena la información en agrupamientos de 4K. Es mucho más eficiente que el antiguo formato de FAT16, que utiliza agrupamientos de hasta 32K. Dependiendo de sus archivos, este método puede ahorrar espacio considerable. Así mismo, puede tener particiones más grandes. Microsoft introdujo FAT32 en Windows 95 de los fabricantes de sistemas pero ésta es la primera versión que puede inicializar los discos existentes.
- **Diagnóstico del sistema.-** Con una infinidad de herramientas nuevas y mejoradas para afinar la PC, Windows98 permite hacer todo, desde programar revisiones rutinarias del sistema hasta solucionar problemas.

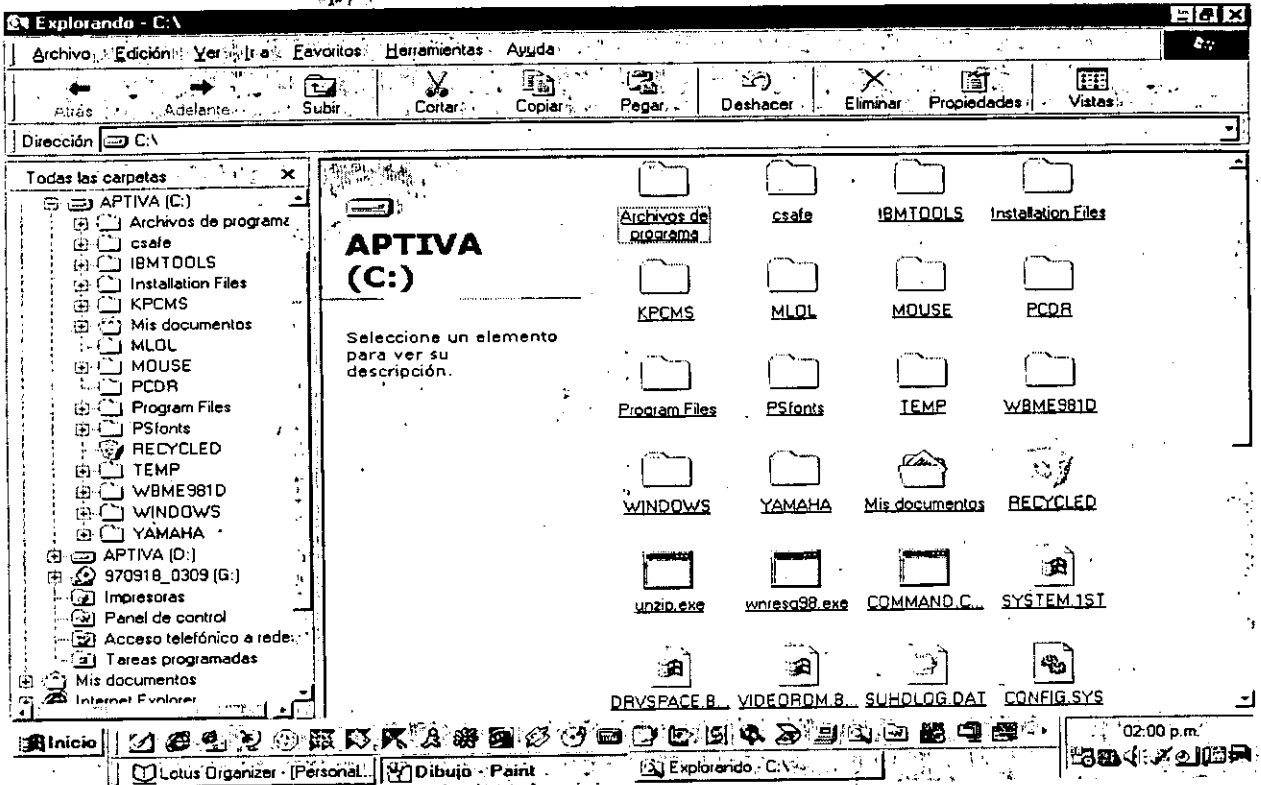


Figura 2.3_5 Explorador de Windows98

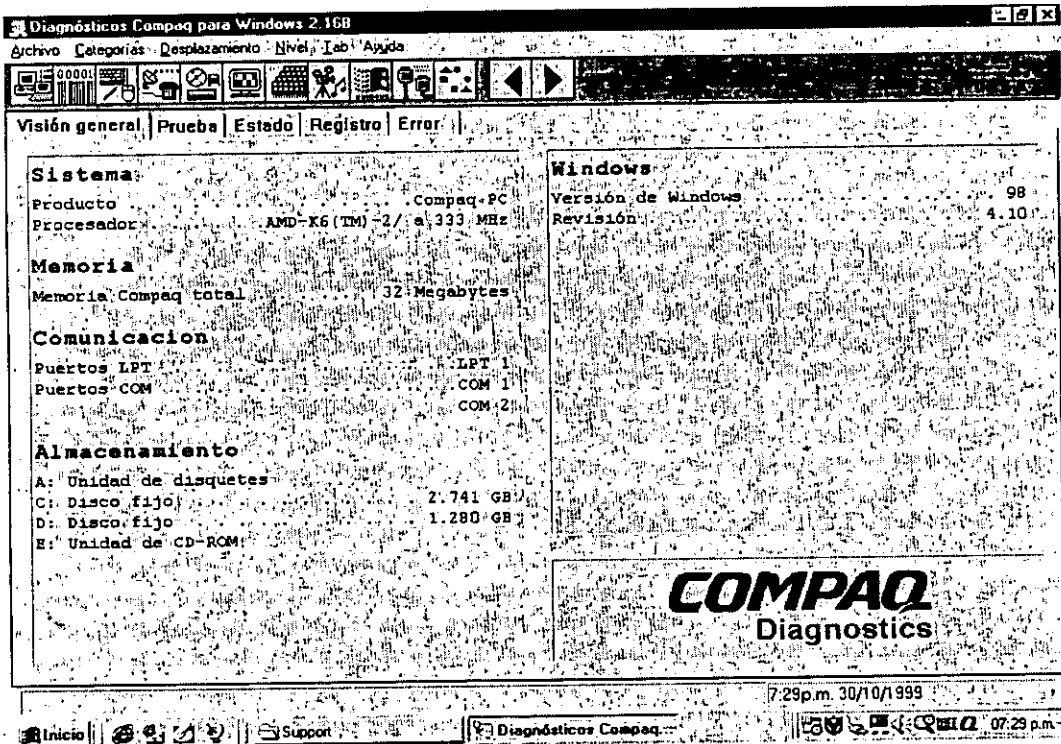


Figura 2.3_6 Diagnósticos del Sistema en Windows98

2.3.1.2 Windows NT

Windows NT, a diferencia de la mayoría de los sistemas operativos, se ha desarrollado de forma independiente de la arquitectura de un procesador. Las partes de programa específicas del procesador se encuentran en conjunto, en una pequeña parte del código de programa denominada HAL ("Hardware Abstraction Layer" o capa de abstracción de Hardware). Basta con modificar esta pequeña parte del sistema operativo para trasladar NT, por ejemplo, de un procesador RISC (en el cual fue desarrollado) a un procesador Intel.

Windows NT es un sistema operativo multitarea, mientras una parte del programa de instalación copia datos en el disco duro, se crean los disquetes de arranque para la instalación. Windows NT utiliza el sistema multitarea real, es decir, que las tareas se ejecutan según la técnica de tiempo compartido y reciben prioridades, como se muestra en la Figura 2.3_7.

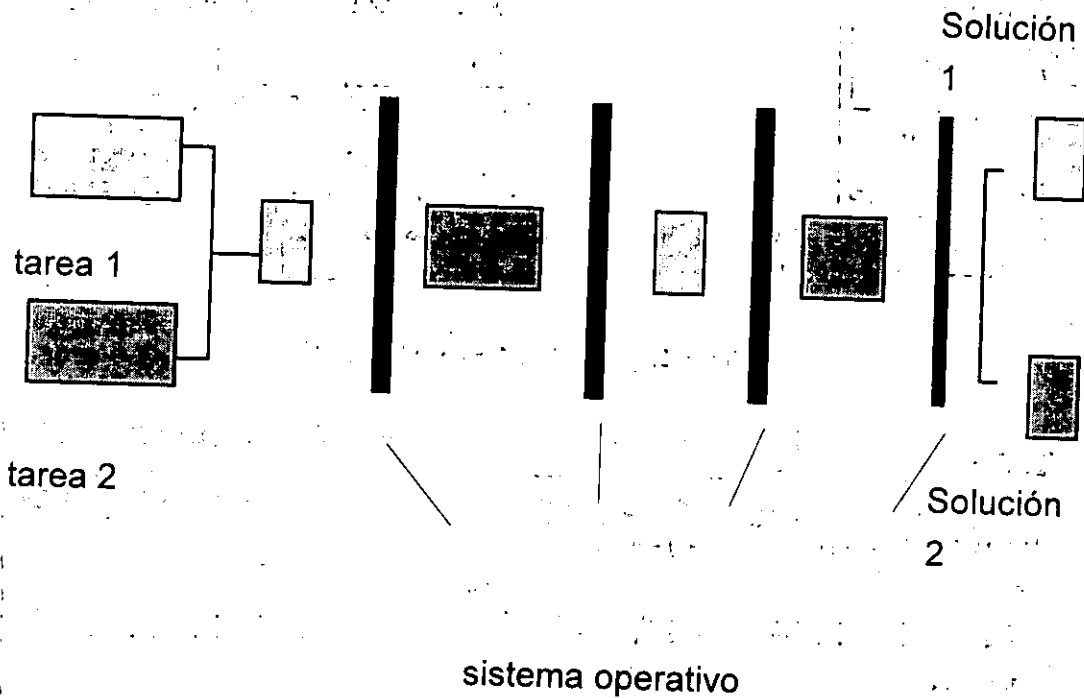


Figura 2.3_7 Multitarea cooperativa: los procesos se ejecutan uno tras otro

El sistema multitarea cooperativa puede asumir y gestionar varios procesos, pero siempre se edita realmente uno sólo. En éste sistema, se indica al sistema operativo que el proceso ha transcurrido de una aplicación a otra, y transmite los recursos que tenía reservados al siguiente proceso.

Características de Windows NT

- Internet Explorer.- La versión 3.0 de Explorador soporta todos los "plug-in" de Netscape, elementos de HTML 3.0 y Java. Además, el Explorer 3.0 contiene la tecnología Active X, un estándar diseñado por Microsoft para ejecutar programas propios a través de Internet.

- Microsoft Exchange.- Microsoft Exchange es un programa muy eficaz con el que se puede controlar y administrar toda la comunicación electrónica (correo electrónico interno, correo electrónico en Internet, correo en CompuServe, fax) desde un solo programa de aplicación. Junto con MS Schedule+, este programa de correo se convierte en un instrumento de planificación y horario dentro de grupos de trabajo.
- Perfiles de Hardware.- Con los perfiles de hardware se pueden guardar distintas configuraciones y activarlas fácilmente a través de sus nombres.
- Servicios Gateway para NetWare.- Con ésta herramienta se puede manejar el ordenador Windows NT en una red 3.1x/4.x y utilizar recursos que se comparten en esa red a través de Bindery o NetWare Directory Services (NDS).
- DCOM (Distributed Component Object Model).- Con esta herramienta es posible utilizar y administrar aplicaciones cliente/servidor repartidas en una red como si se tratara de aplicaciones locales en un ordenador.
- Peer Web Services.- Peer Web Server es un servidor WWW sencillo desarrollado para Windows NT en su versión para estaciones de trabajo. Su objetivo es intercambiar información a través de la WWW en intranets (o sea soluciones internas en redes empresariales) basadas en TCP/IP.
- PPTP (Point-to-Point Tunneling Protocol) Client.- El protocolo punto a punto (PPP) se utiliza para crear redes TCP/IP a través de líneas en serie (teléfono analógico, RDSI). Se emplea generalmente para acceder a un servidor Windows NT o establecer una conexión con un proveedor de servicios de Internet (por ejemplo AOL o CompuServe). Windows NT ofrece con PPTP un protocolo PPP perfeccionado que permite una transferencia de datos segura a través de las líneas telefónicas.
- Multi-Protocol Routing (MPR).- Con el servicio MPR se pueden establecer, a partir de un servidor Windows NT, una interfaz para distintos servicios de red. Así por ejemplo, los datos que llegan al servidor Windows NT a través del protocolo de red Windows NetBEUI pueden ser transmitidos a Internet mediante TCP/IP. Los datos del exterior también pueden ser transmitidos en redes internas. La ventaja esencial de esta organización es que sólo un ordenador tiene conexión directa con el exterior, y en su propia red se necesita un sólo protocolo.

Característica	Windows	NT 4.0 WS (Estándar de trabajo)
Manejo		
Reconocimiento de Hardware	Sí	Sí
Entorno de usuario de Windows 98 orientado por objetos	Sí	Sí
Plug and Play para una instalación fácil y automática de dispositivos periféricos	Sí	Sí (limitado)
Eficiencia		
Multitarea "preemptive"	Sí, en aplicaciones de 32 bits.	Sí
Interfaz de programación Win32 y OLE 2.0 (32 bits)	Sí	Sí
Soporte- GL para la representación de objetos 3D refinados	No	Sí
Funcionalidad en red		
Conexiones peer-to-peer a través de NetBEUI, IPX/SPX o TCP/IP en LANs	Sí	Sí
Soporte de distintos niveles de protocolos y soluciones de red de terceros	Sí	Sí
Acceso remoto a redes RAS	Sí	Sí
Administración		
Administración central de sistema abierta también de terceros proveedores	Sí	Sí
Soporte de estándares de administración de sistema como SNMP y DMI	Sí	Sí
Perfiles de usuarios para el diseño personalizado de escritorio	Sí	Sí
Programas de aplicación		
Aplicación Windows de 16 bits	Sí	Sí en área de dirección separada
Aplicación Win32 y OLE	Sí	Sí
Proceso de aplicación MS-DOS	Sí	sólo en parte, ya que no permite el acceso directo a la memoria
Proceso del IBM Presentation-Manager o de aplicaciones POSIX	Sí	Sí
Estabilidad		
Protección de aplicaciones Win 16 bloqueadas	No, ya que están en la misma área de memoria	Sí, ya que están en áreas de memoria separadas
Reconstrucción automática tras un bloqueo del sistema	No	Sí
Seguridad		
El acceso al sistema correspondiente al estándar C2	No (posibilidades de acceso al sistema FAT, acceso directo a la memoria de las aplicaciones MS-DOS)	Sí. (sistema de archivos NTFS seguro, no hay acceso directo a la memoria.

Tabla 2.3_1 Similitudes y diferencias entre los sistemas operativos

Característica	MS-DOS	Windows (Estación de trabajo)
Seguridad (Cont.)		
El acceso al sistema correspondiente al estándar C2	No (posibilidades de acceso al sistema FAT, acceso directo a la memoria de las aplicaciones MS-DOS)	Sí. (sistema de archivos NTFS seguro, no hay acceso directo a la memoria.
Acceso protegido al escritorio debido a los perfiles de usuario y a las configuraciones de seguridad	No	Sí
Soporte de sistemas de archivos HPFS y NTFS seguros	No	Sí
Compatibilidad		
Utilización de controladores de dispositivos MS-DOS	Sí	No
Utilización de controladores de dispositivos Win216	Sí	No
Soporte de comprensión de disco duro	Sí (a través del paquete Plus)	Sí, integrado en el sistema operativo
Ejecutable en ordenador Power PC, sistemas MIPS, Dec Alpha o RISC	No	Sí
Microsoft Service		
Quick-Fix Teams para la solución de problemas	No	Sí
Paquetes Service en servicios en línea, en Internet y en CD-ROM	Sí	Sí

Tabla 2.3_1 (Cont.) Similitudes y diferencias entre los sistemas operativos

Windows NT 4.0 ha sido creado como sistema operativo para ordenadores de gran rendimiento y pretende promocionar una potencia del sistema también suficiente para los usuarios profesionales. Por ello, las exigencias de NT respecto al ordenador en el que se instala son claramente más altas que en el caso de otros sistemas operativos, como Windows98 o Dos / Windows 3.x.

Los sistemas operativos que pueden ser instalados en paralelo son los siguientes:

- Windows NT 4.0 (Estación de trabajo o Servidor).
- Versiones anteriores de Windows NT (Estación de trabajo o Servidor).
- DOS/Windows 3.x.
- Windows98.
- NeXT.
- Unixware.
- OS/2
- Linux.

Los administradores de iniciación disponibles son:

- El **Boot Manager** o Administrador de Iniciación de NT
- El **Boot Manager** o Administrador de Iniciación de OS/2, y
- Otros administradores de iniciación, como el **Boot Manager Lilo** de Linux.

Además se puede utilizar la instalación "Dual Boot" (inicio múltiple) de Windows 95/98.

En la Tabla 2.3_2 se muestran los requerimientos mínimos para instalar una estación de trabajo y un servidor.

	Windows NT: Estación de Trabajo	Windows NT: Servidor: equipamiento mínimo	Windows NT Servidor: equipamiento razonable
Procesador	Pentium 90; o superior; Alpha, MIPS, Power.PC	Pentium 90	Pentium PRO o Dual Pentium
Memoria RAM	a partir de 12 MB (Intel), o 16 MB (RISC)	a partir de 12 MB (Intel) o 158 MB (RISC)	64 MB o más
Disco duro	a partir de 117 MB (Intel), o 124 MB (RISC) temporal min. 50 MB	a partir de 148 MB (Intel) o 158 MB (RISC) temporal min. 50 MB	dos discos duros reflejados de 4GB
Controladora	E-IDE o SCSI	E-IDE o SCSI	SCSI
Tarjeta gráfica	Tarjeta VGA accelerator con un mínimo de 2MB de memoria	Sólo administrador de sistema en el servidor: tarjeta S-VGA; si no, algo mejor	Sólo administración del sistema en el servidor: tarjetas S-VGA; si no, algo mejor
Unidad de CD	ATAPI; mejor SCSI	ATAPI; mejor SCSI	ATAPI; mejor SCSI

Tabla 2.3_2 Requerimientos de Windows NT Estación de trabajo y Windows NT Servidor

Los diferentes sistemas operativos, utilizan los siguientes sistemas de archivo:

- **FAT.**- Es el sistema de archivo que subdivide el disco duro en bloques.
- **HPFS.**- Se utiliza en ordenadores OS/2, aunque su desarrollo original fue en Microsoft. Los sistemas de archivos HPFS son mucho más seguros, estables, y rápidos que los sistemas de archivos FAT.
- **NTFS.**- El sistema NTFS es todavía más seguro y estable que HPFS, aunque a cambio necesita mayor tiempo de acceso a los archivos.
- Otros sistemas de archivo de los dialectos UNIX. Solo se puede acceder a estos datos desde otro sistema de archivo UNIX o utilizando programas especiales adicionales.

En la tabla 2.3_3 se muestra un resumen de qué sistemas operativos trabajan con qué sistemas de archivo, y si pueden leer desde o escribir a dichos sistemas de archivo.

	FAT	HPFS	NTFS	Linux
Windows NT 4.0	leer y escribir	convertir	leer y escribir	-
Windows Nt 3.x	leer y escribir	leer y escribir	leer y escribir	-
DOS/Windows 3.x	leer y escribir	-	-	-
Windows98	leer y escribir	-	leer a través de redes	-
OS/2	leer y escribir	leer y escribir	-	-
Linux	leer y escribir	-	-	leer y escribir

Tabla 2.3_3 Sistemas de Archivo y su Compatibilidad con Diferentes Sistemas Operativos

2.4 TIPOS DE REDES

2.4.1 Antecedentes

El término "red" por lo general significa un conjunto de computadoras y periféricos, que están conectados por algún medio, esta conexión puede ser directa (por medio de un cable) o indirecta (a través de un módem).

Los dispositivos reales y la forma en cómo están conectados se llama topología de la red. Si los dispositivos de una red están en una sola ubicación, como un edificio, se les llama red de área local o LAN (Local Area Network).

Las redes se clasifican en categorías de acuerdo a su topología o forma. A continuación se describen las topologías de uso más frecuentes en las LAN.

2.4.2 Redes de Computadoras, Tipos y Configuraciones

2.4.2.1 Topologías

2.4.2.1.1 Configuración de Estrella

Las redes usan una topología de estrella si todas las computadoras se conectan a un punto central, en la figura 2.4_1 se ilustra el concepto.

En general, el centro de la red de estrella se llama concentrador. Un concentrador característico consta de un dispositivo electrónico que acepta datos de un transmisor y los entrega al destino adecuado.

En la Figura 2.4_1 se muestra una red de estrella ideal. En la práctica, las redes de estrella pocas veces tienen una forma simétrica en la que el concentrador se localiza a igual distancia de todas las computadoras, en su lugar, con frecuencia, reside en una localidad distinta a las computadoras conectadas.

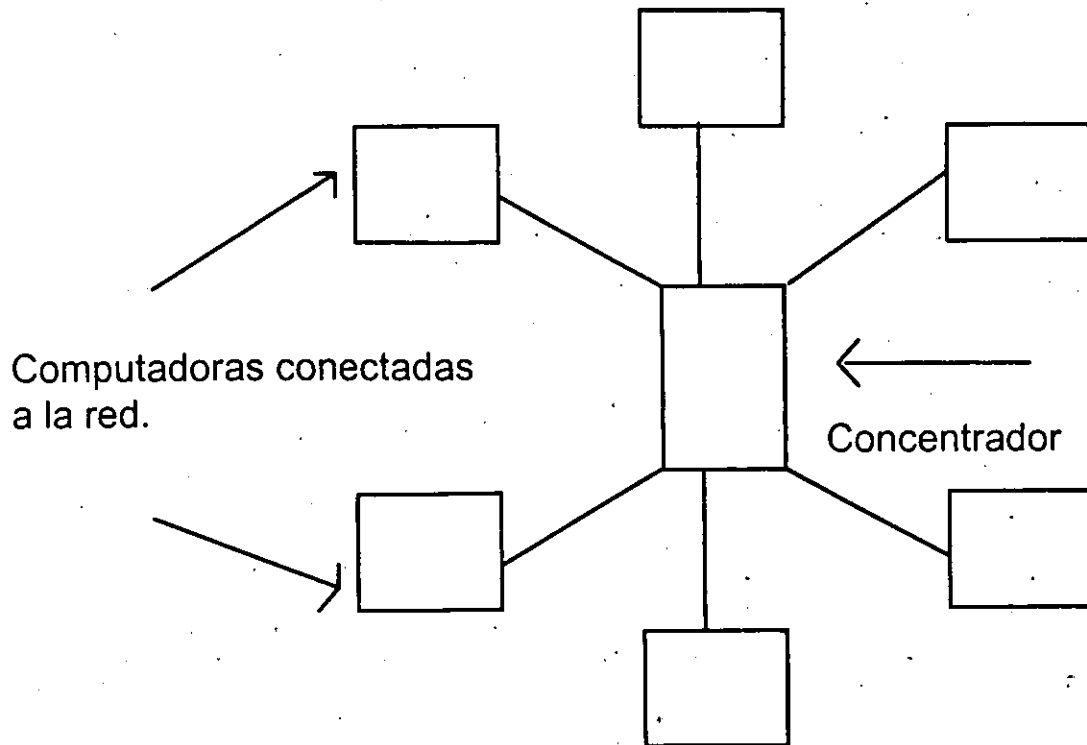


Figura 2.4_1 Topología de Estrella

2.4.2.1.2 Configuración de Anillo

Las redes que utilizan la topología de anillo, disponen la conexión de las computadoras en un ciclo cerrado mediante cables, se conecta la primera computadora a la segunda, la segunda a la tercera, etc., hasta conectar la computadora final con la primera. El nombre de anillo proviene de que las computadoras y cables que las conectan se disponen en círculo, como se ilustra en la Figura 2.4_2.

En la topología tipo anillo, como en la de estrella, se refiere a las conexiones lógicas entre computadoras, no a la orientación física. Las computadoras y las conexiones de una red de anillo no tienen que estar dispuestas en círculo. El cable entre dos computadoras de una red de anillo puede seguir un pasillo o subir al piso superior.

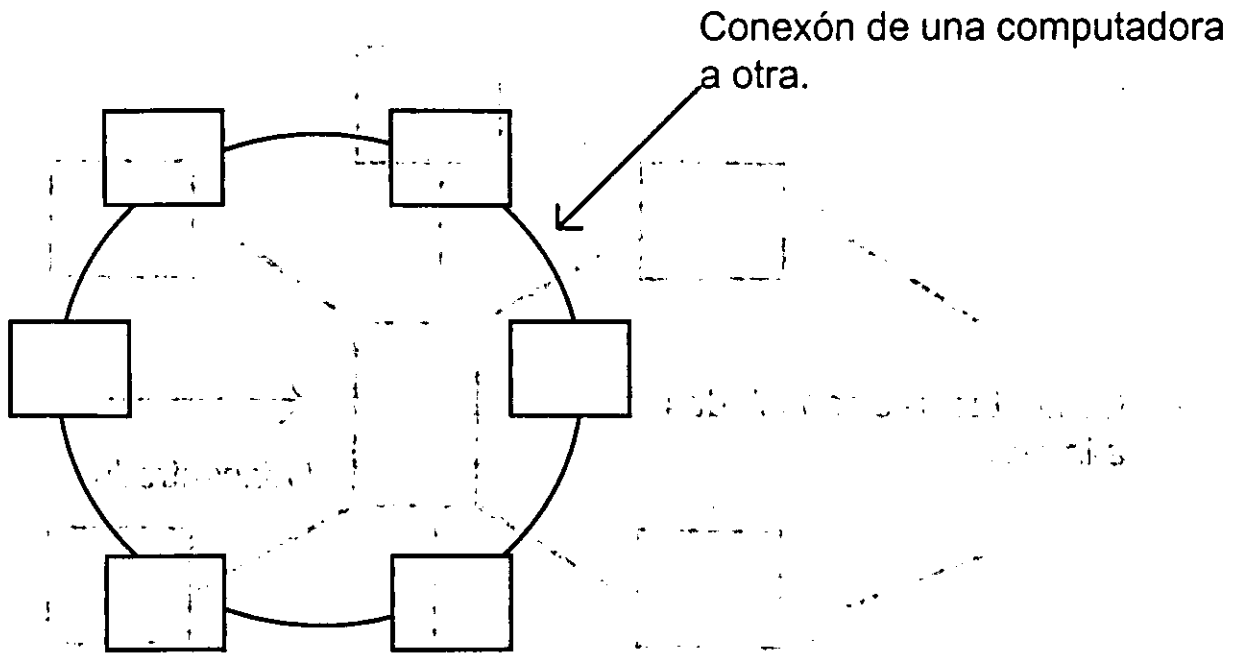


FIGURA 2.4_2 Topología de Anillo

2.4.2.1.3 Configuración de Bus

Las redes que utilizan topología de **Bus** (canal) constan por lo normal de un cable largo al que se conectan las computadoras. Cualquier computadora conectada a un BUS puede enviar señales por el cable y todas las demás las recibirán. En la Figura 2.4_3 se ilustra esta topología. Debido a que todas las computadoras conectadas al cable pueden detectar señales eléctricas, cualquiera de ellas puede transmitir datos a otra. Las computadoras conectadas a una red BUS deben coordinarse para asegurar que sólo una computadora envía las señales a la vez; de lo contrario, se produciría un caos.

Canal (cable compartido)

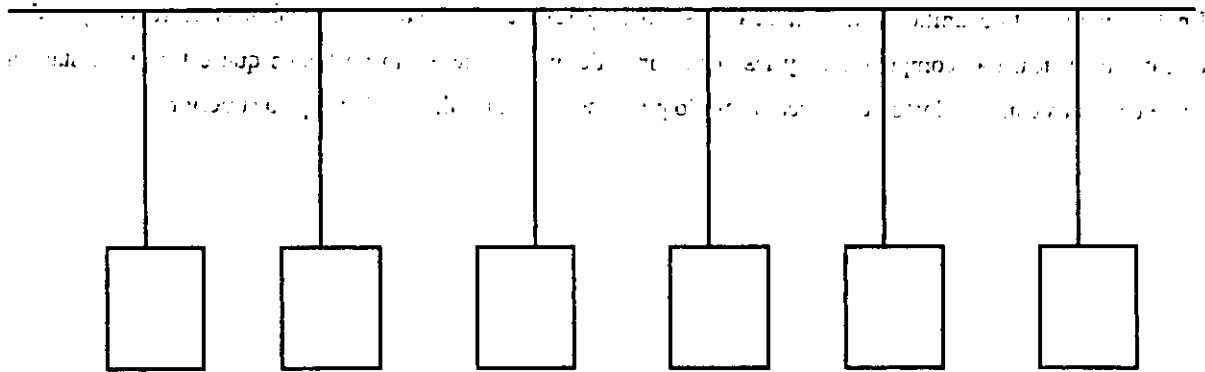


Figura 2.4_3 Topología de Bus

2.4.2.1.4 Configuración de Malla

Las redes tipo malla tienen la siguiente característica: todos sus dispositivos están conectados entre sí.

El número de conexiones punto a punto en una red de malla depende del número de dispositivos que integran la red. Tres dispositivos que integran una red necesitan tres conexiones punto a punto, cuatro dispositivos que integren una red de malla necesitan seis conexiones punto a punto.

Cada dispositivo está comunicado con todos los demás dispositivos. Desde un dispositivo en una red de malla existen múltiples rutas de acceso hacia otros dispositivos.

Si la ruta de acceso directo entre dos dispositivos no está disponible, el mensaje puede ser encaminado a través de los otros dispositivos. Algunas redes híbridas utilizan red de malla entre su estructura.

La figura 2.4_4 muestra un esquema típico de una red de malla.

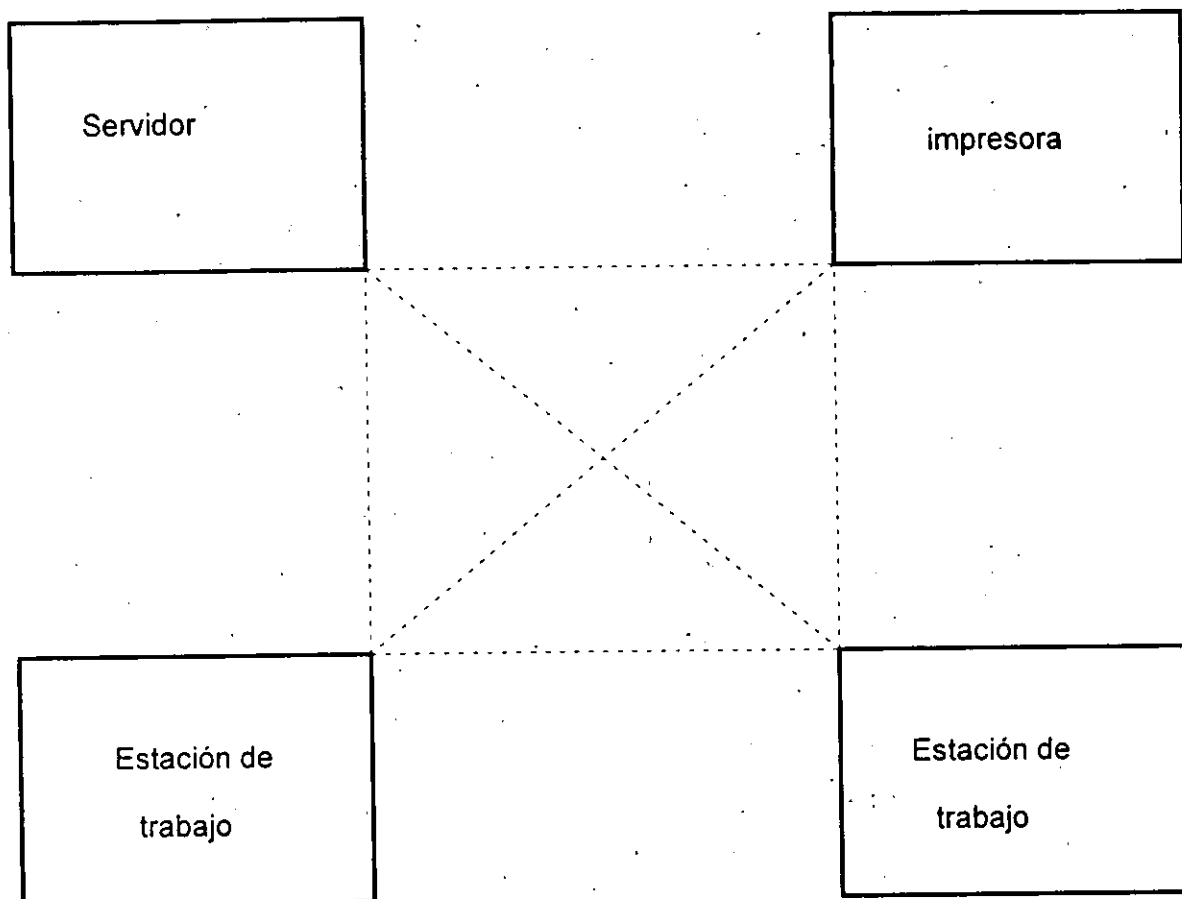


Figura 2.4_4 Esquema de una red de malla.

2.4.2.2 Redes LAN

Ejemplo de red de anillo: Tóken Ring IBM

Casi todas las LAN que usan topología de anillo se valen de un mecanismo de acceso llamado pase de ficha. La red de anillo de pase de fichas se abrevia como anillo de fichas o token ring. Un token ring opera como un solo medio compartido. Cuando una computadora necesita enviar datos, debe esperar un permiso antes de acceder a la red. Una vez obtenido el permiso, el transmisor tiene el control completo del anillo, no puede ocurrir ninguna transmisión simultánea. Cuando el transmisor envía una trama, los bits pasan sucesivamente a las siguientes computadoras, hasta que la estación de destino hace una copia de la trama y este da vuelta al anillo y regresa al transmisor. En la Figura 2.4_5 se muestra el concepto.

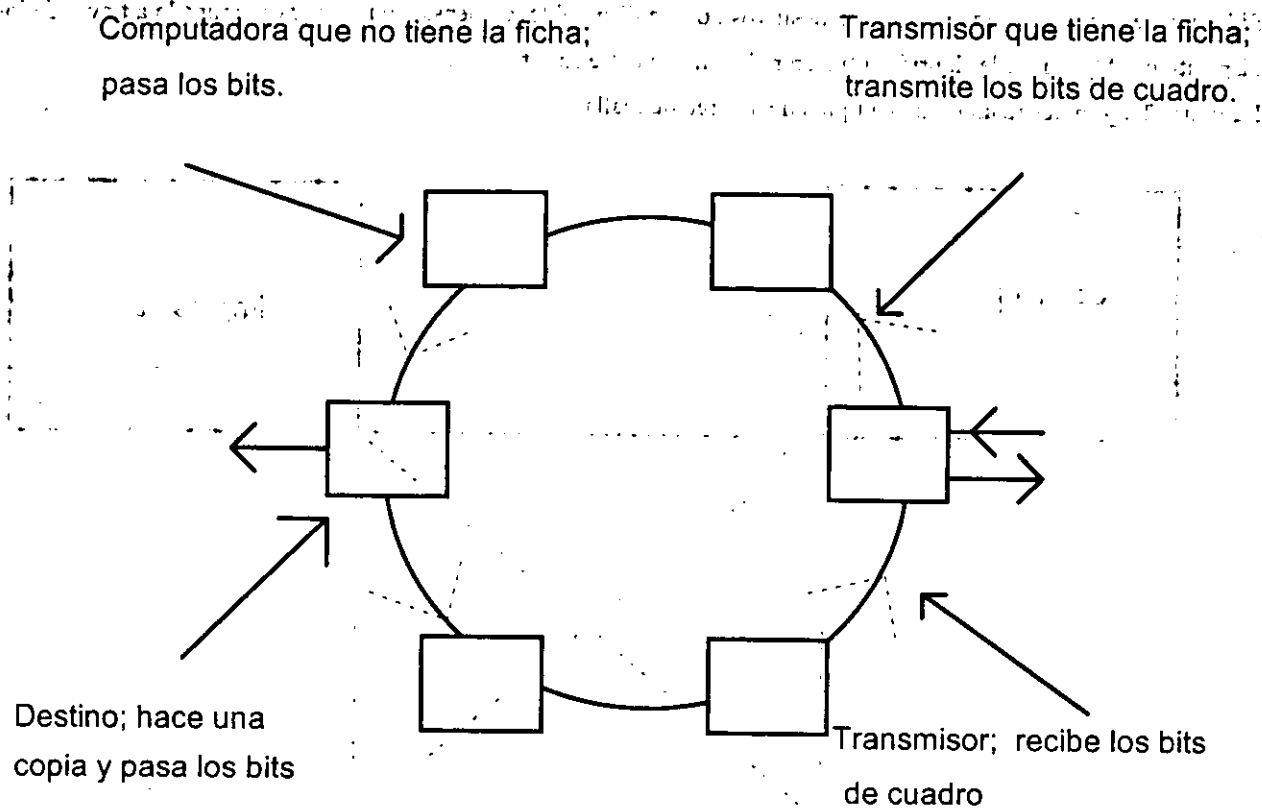


Figura 2.4_5 Flujo de bits durante la transmisión en una red token ring

Como se muestra en la figura 2.4_5, todas las estaciones, excepto el transmisor, reenvían los bits por el anillo. Por lo tanto, para comprobar que no han ocurrido errores de transmisión, el transmisor puede comparar los datos recibidos con los enviados.

Las demás estaciones vigilan todas las transmisiones. Si una trama está destinada a una computadora dada, ésta hace una copia de la trama al ir pasando los bits por el anillo.

2.4.2.3 Redes WAN

Una WAN (Red de Área Ampla) abarca instalaciones o sitios de varias ciudades, países y continentes. Para formar una WAN se interconecta un grupo de conmutadores de paquete. En general, los conmutadores tienen varios conectores de E/S, lo que permite establecer diferentes topologías. Por ejemplo en la Figura 2.4_6 se ilustra una manera de formar una WAN con las conexiones de cuatro conmutadores de paquetes y ocho computadoras.

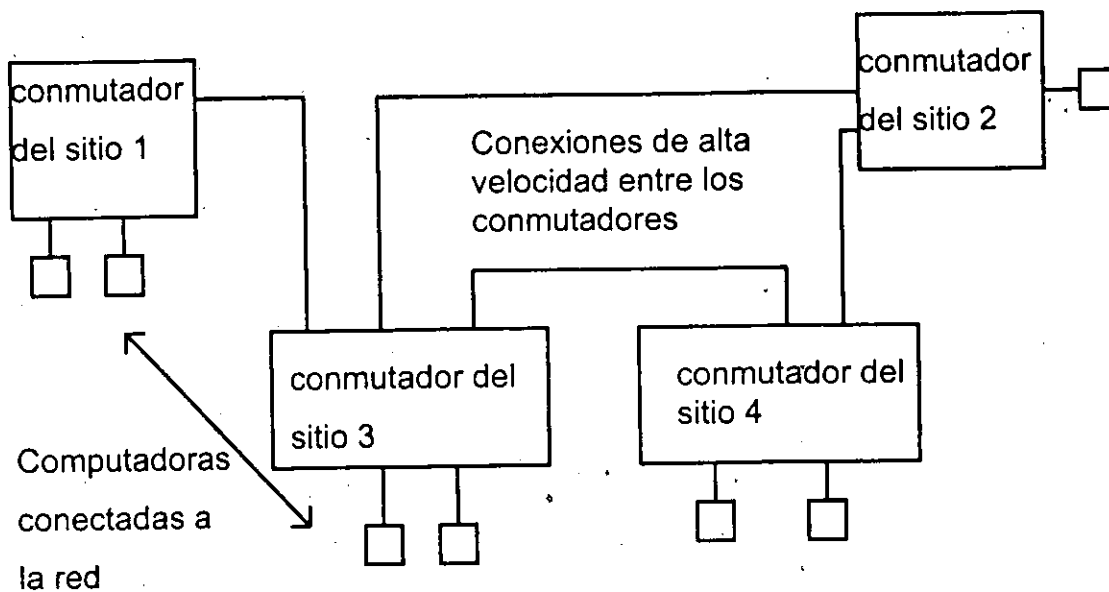


Figura 2.4_6 WAN pequeña formada con la conexión de conmutadores de paquetes

Como se muestra en la figura 2.4_6, no se necesita que la WAN sea simétrica, pues las interconexiones entre los conmutadores y la velocidad a la que operan se seleccionan para ajustarse al tráfico esperado y redundar en caso de fallas. En el ejemplo anterior, el conmutador del sitio 1 se conecta a dos computadoras y tiene una conexión externa hacia otro conmutador. En contraste, el conmutador del sitio 2 sólo se conecta a una computadora y tiene dos conexiones externas a los conmutadores de otros sitios.

El conmutador de paquetes es el bloque de construcción básico de las WAN. Se forma una WAN con la conexión de un grupo de conmutadores de paquetes a los que luego se conectan las computadoras. Se pueden agregar conmutadores e interconexiones para aumentar la capacidad de la WAN. Por lo común, las conexiones entre conmutadores de paquetes operan a mayor velocidad que las conexiones a las computadoras.

2.4.2.4 Redes Ethernet

Ejemplo de la red bus: Ethernet

Ethernet consta de un cable ya sea coaxial, par trenzado o de fibra óptica llamado Ether, al que se conectan varias computadoras. Una red Ethernet con cable coaxial está limitada a 500 metros de extensión, y la norma requiere una separación mínima de 3 metros entre conexiones.

El hardware Ethernet opera a un ancho de banda de 10, 100 y hasta 1,000 megabits por segundo (Mbps). Debido a que utiliza una topología de canal, Ethernet necesita que varias computadoras compartan el acceso al mismo medio.

El transmisor envía una onda portadora modulada, que se propaga del transmisor hacia ambos extremos del cable. En la Figura 2.4_7 se muestra el flujo de datos por una red Ethernet.

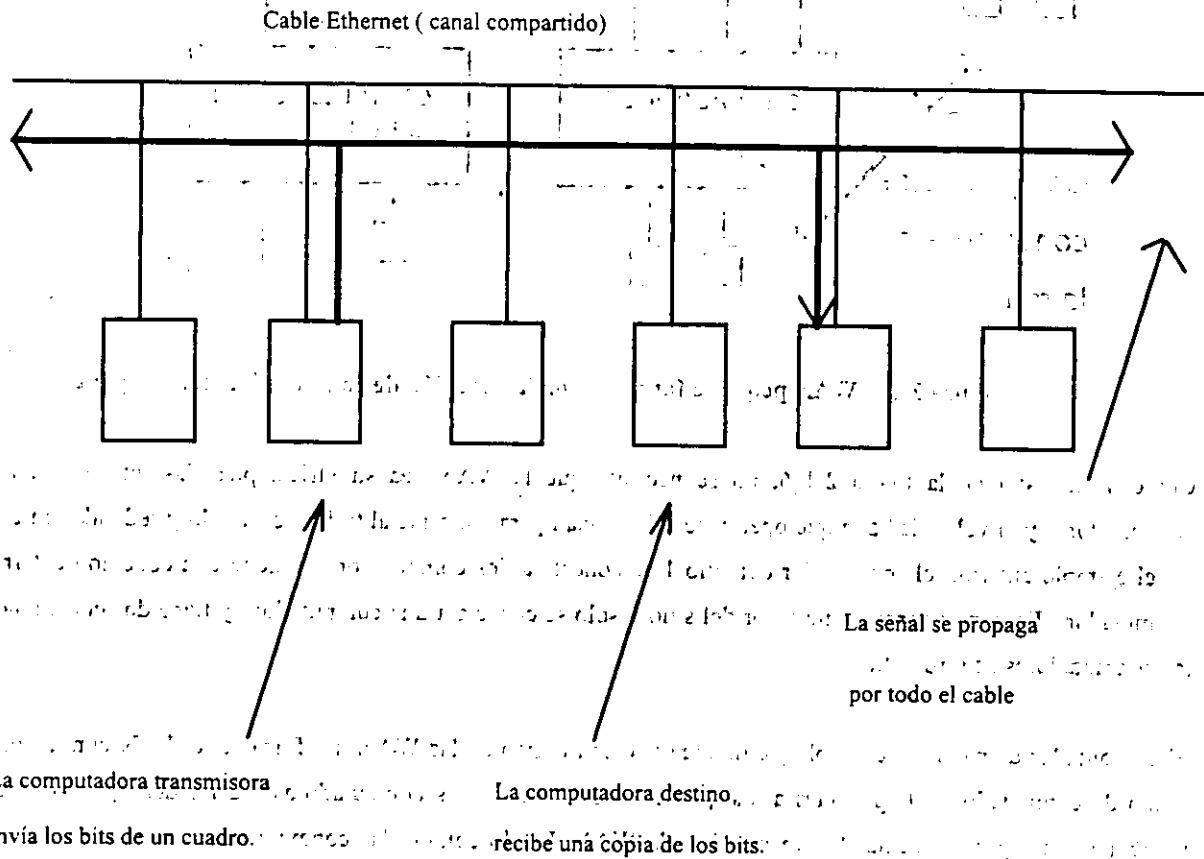


Figura 2.4_7 Flujo de bits por una red Ethernet

Como se muestra en la Figura 2.4_7 la señal se propaga de la computadora transmisora hacia ambos lados del cable compartido. Durante la transmisión de una trama, la computadora transmisora tiene uso exclusivo del cable, las demás deben esperar. Al terminar la transmisión de la trama, la trama compartida queda disponible para el uso de las demás computadoras.

2.4.3 Conectividad de redes

Una de las razones por las cuales una red se amplía, es para que más usuarios puedan tener acceso a ella y se conecten entre sí para poder intercambiar datos con otras redes. Para poder llevar a cabo la conectividad de la red, se utilizan los siguientes dispositivos: Repetidores, Bridges (puentes), Routers (encaminadores), Brouters, enlaces centrales y Gateways (puertas de comunicaciones).

2.4.3.1 Repetidores

Se encargan de amplificar la señal del cable de tal manera que la longitud del cable pueda extenderse. Los repetidores tienen las siguientes características:

- Son utilizados en los sistemas lineales como los Ethernet. Los hubs activos de ARCNET constituyen una forma de repetidor.
- Los repetidores funcionan en el nivel físico, que es el nivel más bajo en la jerarquía de protocolos. Su función es la de amplificar la señal para poder transmitirla sobre un tramo de cable adicional.
- Se utilizan normalmente dentro de un mismo edificio.

2.4.3.2 Bridges

Interconectan dos o más redes pasando los paquetes entre ellos. Los bridges tienen las siguientes características: funcionan en el nivel de control de acceso al medio (MAC). Se pueden conectar dispositivos que usan diferentes protocolos de comunicación.

Un bridge se instala para extender una red existente cuando:

- Se ha llegado a su máxima extensión.
- Se desean eliminar los cuellos de botella que se generan cuando hay muchas estaciones de trabajo conectadas a una única red. Con esto se logra que cada red se cargue con un menor número de usuarios, con lo que se mejora el rendimiento.
- Se requiere conectar entre sí distintos tipos de redes, como la Token Ring y la Ethernet.

2.4.3.3 Routers

Se utilizan fundamentalmente en redes grandes con más de 20 redes locales interconectadas, esto con el fin de que el tráfico fluya eficientemente sobre caminos predefinidos. Se usan en lugar de los bridges cuando:

- Es necesario un filtro avanzado de paquetes.
- La red tiene diversos protocolos y se quiere filtrar el tráfico con unos protocolos específicos a áreas especiales.
- Se tiene la necesidad de utilizar un camino inteligente para mejorar el rendimiento.

- Se utilizan líneas de comunicación remota, lentas y caras.

Un router verifica los campos con información de encaminamiento de los paquetes y determina la mejor ruta para cada uno de ellos. Puede enviar paquetes a redes que no están conectadas directamente a él. Esto debido a que los routers tienen direcciones en el ámbito de estructura de la red que pueden ser vistas por otros dispositivos de la red extendida, incluyendo a otros routers.

Los routers pueden ser específicos para un protocolo o pueden ser multiprotocolo. Este último puede manejar varios protocolos pero suele ser más lento y caro. Los routers multiprotocolo son esenciales en las redes que poseen diversos sistemas y protocolos, y en las que se necesita compartir tráfico entre redes como el sistema de correo electrónico.

2.4.3.4 Brouters

Estos combinan las características de los bridges y los routers. Un brouter selecciona inteligentemente un camino para evitar información como un router, lo que permite se puedan usar uno o más protocolos para encaminar el tráfico en las conexiones con el resto de la red. Algunos brouters pueden llevar a cabo tareas para equilibrar la carga entre las conexiones redundantes y determinar rutas alternativas para cuando existan fallas en las conexiones.

La diferencia principal entre un brouter y un router radica en que el router aplica un protocolo de encaminamiento estándar, mientras que el brouter utiliza un método propio.

2.4.4 Sistemas de Distribución de Redes

Son centros de control de los enlaces que se utilizan para gestionar y monitorear el estado de las conexiones en un edificio completo o en una zona tipo polígono. El sistema consta de dispositivos repetidores que permiten conectar varios tipos de redes a un sistema modular. Generalmente estos dispositivos se conectan a una PC en la que se ejecuta un programa de gestión propietario. Desde la PC los responsables pueden: monitorear el tráfico de la red, redirigir el que pasa por las líneas que fallan a los embotellamientos, esto se realiza a través de un seguimiento de los recursos de toda la red.

2.4.4.1 Gateways

Los gateways permiten conexiones con entornos basados en mini-computadoras y sistemas centrales. A través del gateway cualquier usuario de la red puede acceder al sistema central utilizando el enlace establecido con la red.

Los gateways operan en las tres capas superiores del modelo OSI. Ofrecen el mejor método para conectar segmentos de red y redes mainframes. Un gateway es seleccionado cuando se tiene que interconectar sistemas que se construyeron con base en diferentes arquitecturas de comunicación. Por ejemplo, se utilizaría un gateway para interconectar un TCP/IP a un mainframe SNA ("System Network Architecture") o Arquitectura de Redes de Sistemas).

Un uso frecuente para los gateways es conectar un sistema remoto como una red pública de datos con conmutación de paquetes X.25. (método eficiente de empaquetar datos y enviarlos remotamente). El segmento X.25 cuenta con un protocolo que encamina los paquetes de datos entre dos puntos terminales en la red sin importar qué protocolos pasan por ellos.

En cada extremo de la red el gateway ofrece la conversión de protocolos a los segmentos de red conectados en el otro lado. Los gateways no proporcionan enrutamiento de paquetes dentro de los segmentos de la red, simplemente entregan sus paquetes de datos de tal forma que los segmentos puedan leerlos. Cuando reciben paquetes del segmento, los traducen y encaminan al gateway en el otro extremo, donde los paquetes vuelven a traducirse y entregar al segmento de red en el extremo opuesto.

2.4.4.2 Concentradores

Los concentradores o "Hubs" se encargan de agrupar las conexiones de las terminales de toda la red o parte de ellas. Dependiendo del tipo de hub, sus funciones varían desde repetidor multipuerto (hub para Ethernet) hasta funciones de administración, análisis de protocolos (hub inteligente), etc. Básicamente se pueden agrupar en tres ramas:

- "Stand Alone": simple con su salida de "cascade" (escalado).
- "Stackables": los módulos de "cascade" son de alta velocidad; por eso es más estable o apilable.
- Inteligentes (Switching hub o switch).

2.4.4.3 Concepto de "Switching"

Switching es la retransmisión de cada trama a través del hub, utilizando como base a la dirección MAC (dirección de la tarjeta de red). Dicha retransmisión solamente se realiza a su destinatario, lo cual se conoce como conmutación.

Switches

A estos equipos se les conoce como hubs o concentradores inteligentes, debido a que cada nodo que transmite por este equipo, queda registrado al momento de pasar por su puerto correspondiente; y una vez que todas las estaciones han transmitido al menos una trama, la configuración del switch ha sido completa.

En el entorno de los switches dentro de las aplicaciones futuras, estará presente el contener las comunicaciones multimedia entre usuarios y se requerirá de una transmisión más interactiva y del manejo de prioridades del tráfico.

2.5 Protocolos de Comunicación

2.5.1 El Modelo OSI

En los primeros años de las redes, enviar y recibir datos a través de una red era confuso porque las compañías grandes, tales como IBM, Honeywell, y Digital Equipment Corporation tenían estándares individuales para conectar las computadoras. Los procesos de transmisión y recepción debían "hablar" con los mismos protocolos para comunicarse. Era inverosímil que las aplicaciones que funcionaban en diferentes equipos de diferentes proveedores pudieran comunicarse. Los proveedores, los usuarios, y los grupos de estándares necesitaron convenir en, e implantar una arquitectura estándar que permitiera que los sistemas informáticos intercambiaran información aunque utilizaran programas y equipo de diversos proveedores. En 1978, la Organización Internacional de Estándares (ISO) introdujo un modelo OSI (Open Systems Interconnect) para la interconexión abierta de los sistemas, como un primer paso hacia la estandarización internacional de los diversos protocolos requeridos para la comunicación en red. Este modelo ISO OSI incorpora las siguientes cualidades.

- Ha sido diseñado para establecer los estándares de comunicación de datos que promuevan la interoperabilidad entre múltiples proveedores.
- Consiste en siete capas, con un conjunto específico de funciones de red asignadas a cada capa y con las pautas para la puesta en práctica de las interfaces entre las capas.
- Especifica el sistema de protocolos y de interfaces para poner en ejecución cada capa.

2.5.1.1 Las Capas OSI

Cada capa del modelo OSI existe como módulo independiente. En teoría, se puede sustituir un protocolo con otro en cualquier capa dada sin afectar la operación de capas arriba o abajo. Los principios usados para crear el modelo de siete capas se enumeran abajo.

- Una capa deberá ser creada solamente cuando se requiera un nivel diferente de abstracción.
- Cada capa deberá realizar una función bien definida.
- La función de cada capa se debe elegir con la meta de definir protocolos internacionalmente estandarizados.
- Los límites de la capa se deben elegir para reducir al mínimo el flujo de información a través de las interfaces.
- El número de capas debe ser lo bastante grande para permitir separar funciones distintas, pero lo suficientemente pequeño para evitar que la arquitectura llegue a ser poco manejable.

El diagrama siguiente muestra la enumeración de las capas, comenzando con la capa física, que es la más cercana a los medios físicos de la red.

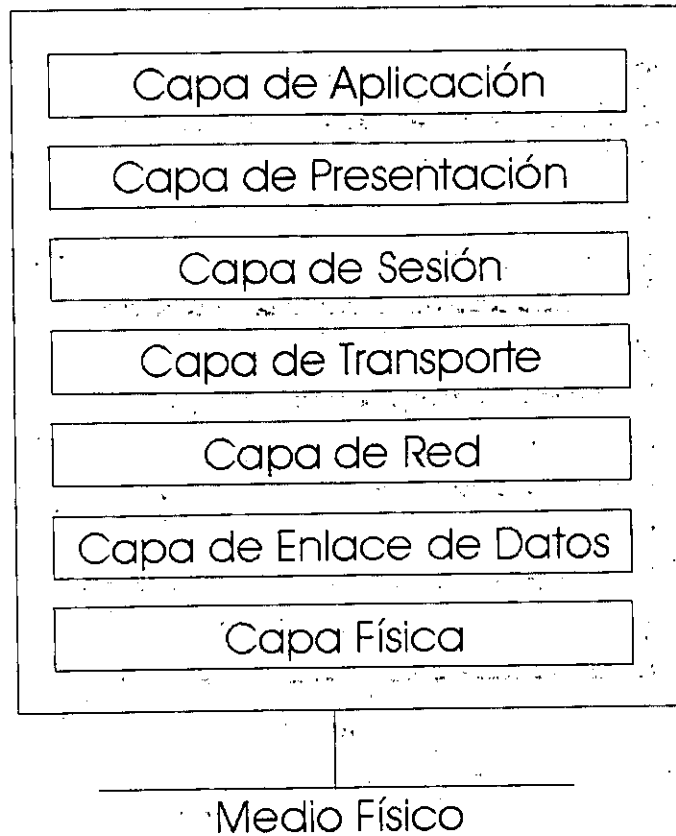


Figura 2.5_1 Capas del modelo OSI

La Capa Física

La capa física es la capa más baja del modelo OSI. Esta capa controla la manera en que los datos brutos y no estructurados del flujo de bits se envían y se reciben a través de un medio físico. Esta capa describe las interfaces eléctrica u óptica, mecánica, y funcionales con el medio físico de la red. La capa física lleva las señales para todas las capas más altas. La codificación de los datos modifica el patrón simple, de la señal digital (1s y 0s) usada por la computadora, para acomodar mejor las características del medio físico y para asistir a la sincronización de bits y de "frames" (tramas).

La codificación de los datos resuelve los siguientes puntos.

- Qué patrón de la señal representa un 1 binario.
- Cómo la estación receptora reconoce cuando un "tiempo de bit" comienza.
- Cómo la estación receptora delimita una trama.

La conexión al medio físico resuelve las siguientes cuestiones.

- Si un transmisor-receptor externo será utilizado para conectar con el medio

- Cuántas patas (pins) tienen los conectadores y para qué se utiliza cada pata.

La técnica de transmisión determina si los bits codificados serán transmitidos por medio de señalización de banda base (digital) o de banda ancha (analógica). La transmisión del medio físico determina si es apropiado transmitir los bits como señales eléctricas u ópticas, basándose en los criterios siguientes.

- Qué opciones de medio físico pueden ser utilizadas.
- Cuántos volts se deben utilizar para representar un estado dado de la señal en el medio físico específico.

La Capa de Enlace de Datos

La capa de enlace de datos ("**Data Link**") proporciona la transferencia sin error de las tramas de datos de una computadora a otra a través de la capa física. Las capas que están arriba de esta capa pueden asumir que la transmisión a través de la red es virtualmente libre de error.

La capa del enlace de datos proporciona las funciones siguientes.

- Establecer y terminar un acoplamiento lógico (conexión de circuito virtual) entre dos computadoras identificadas por sus direcciones únicas de "**Network Interface Cards**" (tarjetas de interfaz de red) o NIC.
- Controlar el flujo de tramas instruyendo a la computadora que transmite para no enviar **buffers** (almacenamiento temporal de datos) de tramas.
- Transmitir y recibir tramas secuencialmente.
- Proveer y esperar "**frame-acknowledgment**" (acuse de recibo de tramas) y detectar y recuperarse de los errores que ocurren en la capa física, por medio de la retransmisión de tramas no reconocidas y de manejar los recibos duplicados de tramas.
- Manejar el acceso a los medios para determinar cuándo se permite a la computadora utilizar el medio físico.
- Delimitar las tramas para crear y reconocer sus límites.
- Tramas de "**error-checking**" (verificación de error) para confirmar la integridad de la trama recibida.
- Examinar la dirección de destino de cada trama recibida y determinar si la trama se debe dirigir a la capa siguiente.

La Capa de Red

La capa de red controla la operación de las sub-redes. Determina qué trayectoria física siguen los datos, basándose en las condiciones de la red, la prioridad del servicio, y otros factores. La capa de red proporciona las funciones siguientes.

- Transferencia de la trama a un "**router**" (encaminador) si la dirección de red del destino no indica la red a la cual la estación está conectada.

- Controlar el tráfico de la sub-red para permitir que un sistema intermedio ordene a una estación que envía datos que no transmita su trama cuando el buffer del router está lleno. Si el router está ocupado, la capa de red puede indicar a la estación transmisora que utilice un router alternativo.
- Permitir que el router fragmente una trama cuando el tamaño de la unidad máxima de transmisión (MTU) de un router en la trayectoria es menor que el tamaño de trama. Los fragmentos de la trama serán ensamblados de nuevo por la estación de destino.
- Resolver, si es necesario, la dirección lógica de la computadora (en la capa de red) por medio de la dirección física de la tarjeta de interfaz de red (NIC) (en la capa de enlace de datos).
- Guardar un registro de estadísticas de las tramas remitidas por el sistema intermedio de sub-red para producir información de facturación.

La capa de red en la computadora que transmite debe construir su encabezado de una manera tal que las capas de red que residen en los sistemas intermedios de la sub-red puedan reconocer el encabezado y utilizarlo para encaminar los datos a la dirección de destino. Esta capa elimina la necesidad de que capas más altas deban saber cualquier cosa sobre la transmisión de datos o las tecnologías intermedias de conmutación usadas para conectar sistemas. La capa de red es responsable de establecer, de mantener, y de terminar la conexión a uno o varios sistemas intermedios en la sub-red de comunicaciones. En la capa de red y las capas debajo de ella, los protocolos de "peer" (par o igual) están entre cada computadora y su vecino inmediato, que no es a menudo la última computadora de destino. Las computadoras de la fuente y del destino pueden estar separadas por muchos sistemas intermedios.

La Capa de Transporte

La capa de transporte se cerciora de que los mensajes sean entregados en el orden en el cual fueron enviados y que no haya pérdida o duplicación. Quita a los protocolos de capa más altos la preocupación por la transferencia de datos entre esas capas más altas y sus iguales (peers). El tamaño y la complejidad de un protocolo de transporte dependen del tipo de servicio que puede conseguir de la capa de red o de la capa de enlace de datos. Para una capa confiable de red o de enlace de datos con capacidad de circuito virtual, tal como la capa de NetBEUIs LLC, se requiere una capa de transporte mínima. Si la capa de red o la capa de enlace de datos no es fiable o apoya solamente datagramas (como lo hacen la capa IP de TCP/IP y la capa IPX de NWLink), la capa de transporte debe incluir el encadenamiento en secuencia y el acuse de recibo (acknowledge) de las tramas, así como la detección y recuperación de error asociadas. Las funciones de la capa de transporte incluyen las tareas siguientes.

- Aceptar mensajes de la capa de arriba y, si es necesario, dividirlos en tramas.
- Proveer la entrega confiable, punto a punto del mensaje, con acuses de recibo (acknowledgments).
- Instruir a la computadora que envía, para que no transmita cuando ningún buffer de recepción esté disponible.
- Multiplexar varias corrientes de mensajes proceso a proceso o sesiones sobre un enlace lógico y sin perder de vista qué mensajes pertenecen a qué sesiones.

La capa de transporte puede aceptar mensajes grandes, pero hay límites estrictos de tamaño impuestos por las capas en el nivel de red y en los más bajos. Por lo tanto, la capa de transporte debe fragmentar los mensajes en unidades más pequeñas, llamadas tramas (frames), y pegar un encabezado a cada trama. Si las capas más bajas no mantienen la secuencia, el encabezado de transporte (TH) debe contener la información de la secuencia, que permite a la capa de transporte en el extremo de recepción presentar datos en la secuencia correcta a la capa más alta siguiente.

A diferencia de las capas más bajas de sub-red, cuyos protocolos están entre nodos o computadoras inmediatamente adyacentes, la capa de transporte y las capas sobre ella son verdaderas capas que van de la fuente al destino. No se preocupan de los detalles de la facilidad de comunicaciones subyacente. El software para las capas en el nivel de transporte y superiores en la computadora fuente lleva una conversación con software similar en la computadora de destino, usando mensajes de control y los encabezados de todos los mensajes, incluyendo los de control.

La Capa de Sesión

La capa de sesión establece una sesión de comunicaciones entre los procesos que funcionan en diferentes computadoras, y puede apoyar la transferencia de datos en modo de mensaje. Las funciones de la capa de sesión incluyen:

- Permitir que los procesos de aplicaciones registren direcciones de proceso únicas, tales como nombres de NetBIOS. Proporciona el medio por el cual estas direcciones de proceso se pueden traducir en las direcciones NIC de la capa de red o de la de enlace de datos, en caso de ser necesario.
- Establecer, monitorear y terminar una sesión de circuito virtual entre dos procesos identificados por sus direcciones únicas de proceso. Una sesión de circuito virtual es un enlace directo que parece existir entre el remitente y el receptor. En realidad, la conexión se establece a través de circuitos.
- Delimitar mensajes, para agregar la información de encabezado que indica donde un mensaje comienza y termina. La capa de sesión receptora puede entonces abstenerse de indicar cualquier dato del mensaje a la aplicación subyacente hasta que se ha recibido el mensaje entero.
- Informar a la aplicación receptora cuando el espacio de buffer es insuficiente para el mensaje entero y que el mensaje es incompleto (llamado sincronización del mensaje). La capa de sesión receptora puede también utilizar una trama de control para informar a la capa de sesión transmisora cuántos bytes (octetos) del mensaje se han recibido con éxito. La capa de sesión transmisora puede entonces reanudar el envío de datos en el byte que sigue al último byte reconocido como recibido. Cuando la aplicación proporciona posteriormente otro buffer, la capa de sesión puede poner el resto del mensaje en ese buffer e indicar a la aplicación que se ha recibido el mensaje entero.
- Realizar otras funciones de soporte que permiten que los procesos se comuniquen a través de la red, tales como la seguridad de la autenticación del usuario y del acceso a recursos.

La Capa de Presentación

La capa de presentación sirve como el traductor de los datos para la red. Esta capa en la computadora transmisora traduce datos del formato enviado por la capa de aplicación a un formato común. En la computadora receptora, la capa de

presentación traduce el formato común a un formato conocido para la capa de aplicación. La capa de presentación proporciona las funciones siguientes.

- Traducción del código de carácter, por ejemplo de ASCII a EBCDIC.
- Conversión de datos, tales como ordenar los bits, de CR a CR/LF, y de número entero a otro de punto flotante.
- Compresión de datos, que reduce el número de bits que necesitan ser transmitidos.
- El cifrado de datos, que hace los datos ilegibles hasta que han sido descifrados, para propósitos de seguridad. Un ejemplo es el encriptamiento de "password" (contraseña).

La Capa de Aplicación

La capa de aplicación sirve como la ventana para que los usuarios y procesos de aplicación tengan acceso a los servicios de red. La capa de aplicación proporciona las siguientes funciones:

- El compartir recursos y redirigir dispositivos.
- Acceso remoto a archivos.
- Soporte a la comunicación entre procesos.
- Soporte para llamado remoto a procedimientos.
- Administración de la red.
- Servicios de directorio.
- Mensajería electrónica, incluyendo "e-mail" (correo electrónico).
- Simulación de terminales virtuales.

2.5.1.2 El Flujo de Datos en el Modelo OSI

El modelo OSI presenta una arquitectura de flujo de datos estándar, con los protocolos especificados de una manera tal que la capa n en la computadora de destino reciba exactamente el mismo objeto que fue enviado por la capa n en la computadora originadora.

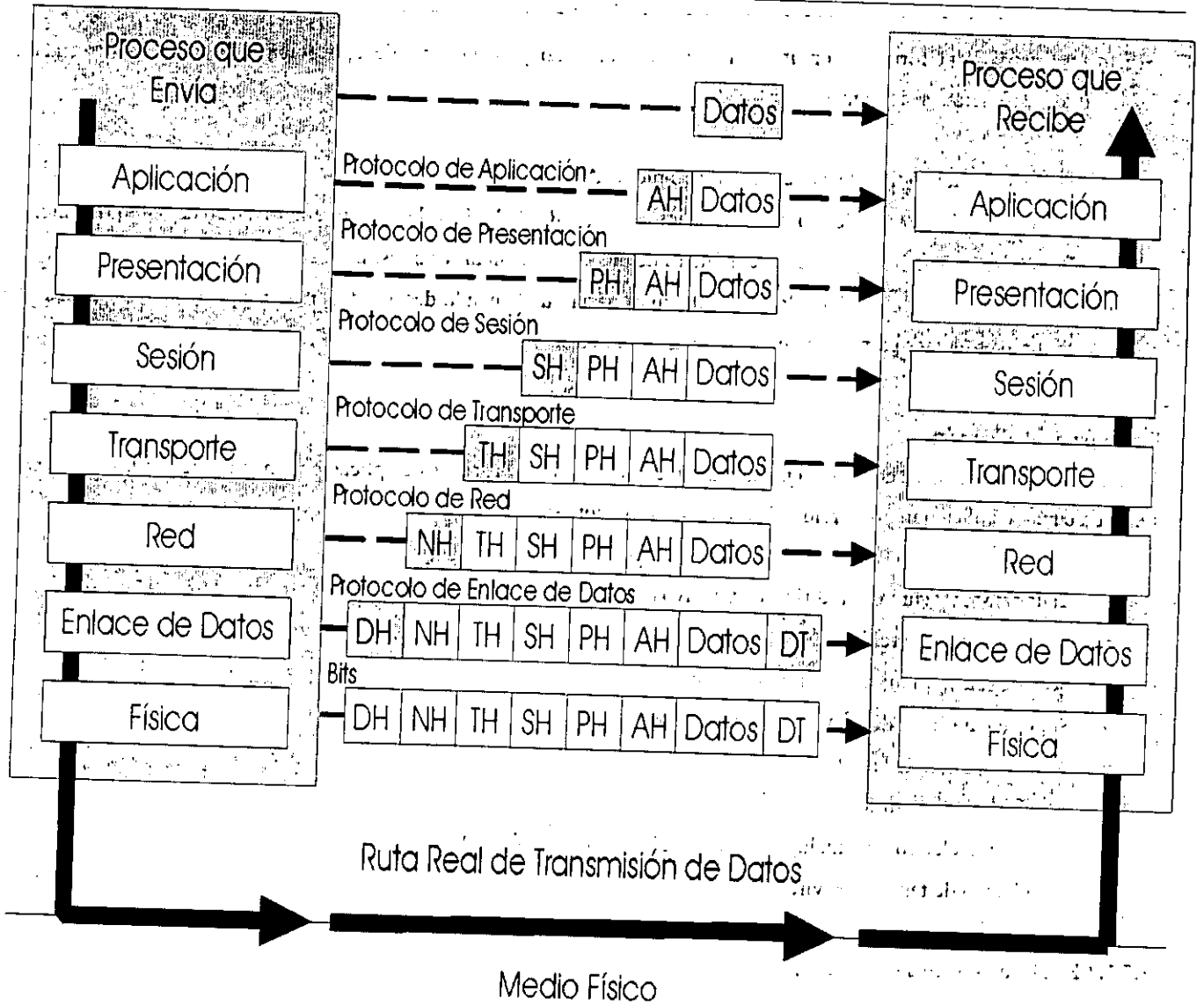


Figura 2.5_2 Modelo OSI de Flujo de Datos

El proceso emisor pasa datos a la capa de aplicación, que agrega un encabezado de aplicación (AH) y después pasa la trama a la capa de presentación.

La capa de presentación puede transformar los datos de varias maneras, en caso de ser necesario, por ejemplo traduciéndolos y agregando un encabezado. Entonces entrega el resultado a la capa de sesión. La capa de presentación no está "enterada" de qué porción (si hay alguna) de los datos recibidos de la capa de aplicación es un AH y qué porción son realmente datos del usuario, porque esa información es irrelevante para el rol de la capa de presentación.

El proceso se repite de capa a capa hasta que la trama alcanza la capa de enlace de datos. Allí, además de un encabezado, se agrega un "data trailer" (rastreador de datos) o DT para ayudar en la sincronización de las tramas. La trama entonces se pasa abajo a la capa física, donde se transmite realmente a la computadora receptora.

En la computadora receptora, los varios encabezados y el DT se eliminan uno por uno conforme asciende la trama por las capas y finalmente alcanza el proceso receptor.

Aunque la transmisión de datos real es vertical, se programa cada capa como si la transmisión fuera horizontal. Por ejemplo, cuando una capa de transporte transmisora obtiene un mensaje de la capa de sesión, agrega un encabezado de transporte (TH) y lo envía a la capa de transporte receptora. El hecho de que el mensaje pasa realmente a la capa de red en su propia máquina es poco importante.

2.5.1.3 La terminología de Interfaz Vertical en el Modelo OSI

Además de definir la arquitectura ideal de siete capas y las funciones de red asignadas a cada capa, el modelo OSI también define un conjunto estándar de reglas y de términos asociados que gobiernan las interfaces entre las capas.

Los elementos activos del protocolo en cada capa se llaman entidades, que se ponen en ejecución típicamente por medio de un proceso de Software. Por ejemplo, la "Suite" (el juego) de protocolos de TCP/IP contiene dos entidades dentro de su capa de transporte: El protocolo de control de transmisión (TCP) y el protocolo de datagrama del usuario (UDP). Las entidades en la misma capa en computadoras diferentes se llaman entidades pares (peer).

La capa directamente debajo de las entidades de la capa "n" pone en ejecución los servicios que serán utilizados por la capa "n". Para los servicios de transferencia de datos, OSI define la terminología para los componentes discretos de los datos pasados a través de la interfaz y entre las entidades pares, como lo describe el ejemplo siguiente.

- La entidad de la capa "n" pasa una unidad de datos de interfaz (IDU) a la entidad de la capa "n-1".
- La IDU consiste en una unidad de datos de protocolo (PDU) y cierta información de control de interfaz (ICI).
- La PDU son los datos que la entidad de la capa "n" desea pasar a través de la red a su entidad par. Consiste en el encabezado de la capa "n" y los datos que la capa "n" recibió de la capa "n+1".
- La PDU de la capa "n" se convierte en la unidad de datos de servicio (SDU) de la capa "n-1", porque es la unidad de datos que será servida por la capa "n".
- La ICI se compone de información de control, tal como la longitud de la SDU, y la información de direccionamiento que la capa de abajo necesita para hacer su trabajo.
- Cuando la capa "n-1" recibe la IDU de la capa "n", le quita y "considera" la ICI, agrega la información de encabezado para su entidad par a través de la red, agrega la ICI para la capa de abajo, y pasa la IDU resultante a la entidad de la capa "n-2".

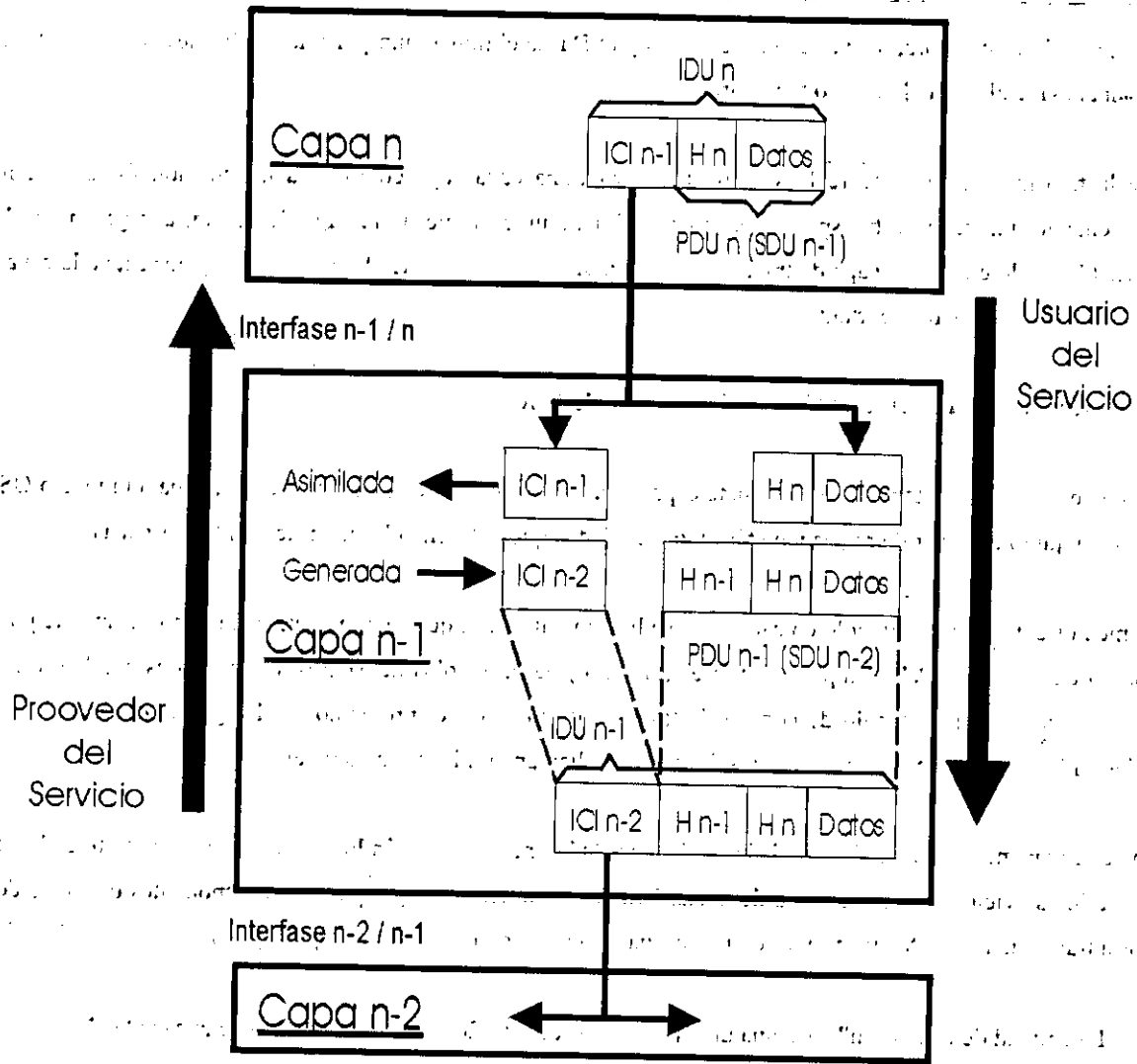


Figura 2.5_3 Entidades de Interfaz Vertical

Las entidades de capa dentro de una computadora pueden llamar a entidades de la capa sobre y debajo de ellos por medio de mecanismos establecidos de llamada de interfaz (tales como una interrupción en MS-DOS o, en Windows NT, una llamada de API-función o sometimiento de IRP) y entonces pasar una estructura definida de IDU.

2.5.1.4 El Estándar 802 de IEEE para los Protocolos de Bajo Nivel

Reconociendo una necesidad de estándares en el mercado de redes de área local (LAN), el IEEE emprendió el Proyecto 802. Nombrado así por el año (1980) y mes (febrero) de su inicio, el proyecto 802 define una familia de estándares de protocolo de bajo nivel en las capas física y de enlace de datos del modelo OSI. Bajo los términos de IEEE 802, la capa de enlace de datos de OSI se divide adicionalmente en dos subcapas: Control Lógico del Enlace (LLC) y Control de Acceso a los Medios (MAC).

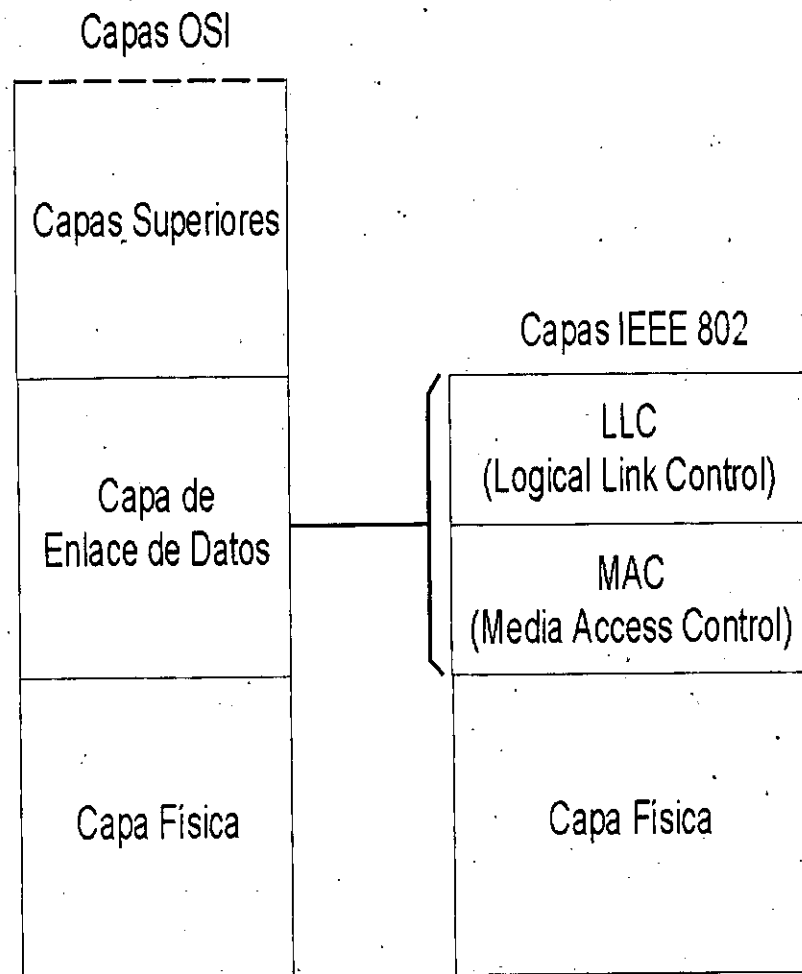


Figura 2.5_4 Comparación de las Capas IEEE y del Modelo OSI

Las funciones de la capa de enlace de datos asignadas a la subcapa del LLC incluyen las siguientes.

- Establecer y terminar enlaces.
- Controlar el tráfico de tramas.
- Ordenar secuencias de tramas.
- Dar acuse de recibo (acknowledging) de tramas.

Las funciones de la capa de enlace de datos asignadas a la subcapa del MAC incluyen las siguientes:

- Administración del acceso a los medios.
- Delimitar tramas.
- Comprobar errores de trama.
- Reconocer direcciones de trama.

Las especificaciones del proyecto 802 incluyen las categorías siguientes.

- La descripción del proyecto 802, incluyendo capas más altas e interconexión de redes (internetworks).
- Control Lógico del Enlace (LLC).
- "Carrier Sense Multiple Access with Collision Detect" (Acceso Múltiple de Percepción de Portadora con Detección de Colisión) (CSMA/CD).
- Token Bus.
- Token Ring.
- "Metropolitan Area Network" (Redes de Area Metropolitana).
- Grupo Consultivo de Tecnologías "Broadband" (de Banda Ancha).
- Grupo Consultivo de Tecnologías de Fibras Ópticas.
- Integración de Voz / datos en LANs.
- Estándar para la Seguridad Interoperable de LAN.

Las especificaciones de protocolos de bajo nivel 802.3 CSMA/CD, 802.4 Token Bus, y 802.5 Token Ring se diferencian en la capa física y en la subcapa del MAC, pero son compatibles en la subcapa de LLC.

Los siguientes organismos han adoptado los estándares 802:

- ANSI, como estándares nacionales de EUA.
- "National Bureau of Standards" (Oficina Nacional de Estándares) o NBS, como estándares del gobierno de EUA.
- ISO, como estándares internacionales (conocidos como ISO 8802).

2.5.1.5 La especificación ANSI FDDI

Relacionado de cerca con los estándares IEEE 802, es un estándar de protocolo de bajo nivel desarrollado más recientemente, "Fiber Distributed Data Interchange" (Intercambio de Datos Distribuidos por Fibra, FDDI). FDDI fue desarrollado por ANSI y se basa en el uso del cable de fibra óptica. FDDI se diferencia de los estándares IEEE 802 en la capa física y en la subcapa del MAC, pero es compatible con los estándares de IEEE en la subcapa de LLC.

2.5.1.6 Servicios de Transferencia de Datos

Las entidades de protocolo dentro de una arquitectura de red proporcionan varios tipos de servicios de transferencia de datos de una capa a las capas sobre ella. Los servicios más frecuentes de transferencia de datos son llamados servicios de conexión confiable ("reliable connection") y servicios sin conexión, no confiables ("unreliable connectionless").

Un servicio de conexión requiere que un circuito virtual o una conexión se establezca desde la computadora fuente hasta una única computadora de destino antes de que cualquier transferencia de datos pueda ocurrir. Una conexión actúa como un tubo: El remitente empuja objetos hacia adentro en un extremo, y el receptor los toma hacia fuera en el otro extremo, en la secuencia apropiada. Como se proporciona una secuencia u orden, un mensaje que es más grande que el tamaño máximo de la trama de transmisión se puede descomponer en tramas múltiples, ser enviado a través de la red, y ser ordenado de nuevo en la secuencia correcta en la computadora receptora.

Un servicio sin conexión no requiere ninguna conexión inicial y no ofrece ninguna garantía de que las unidades de datos llegarán en secuencia. No se requiere ninguna conexión, así que los mensajes se pueden enviar a una o a múltiples estaciones de destino. No se proporciona ningún orden o secuencia, así que los mensajes se pueden enviar solamente en un único tamaño de trama. Un servicio confiable nunca pierde datos porque el receptor acusa el recibo de todos los datos enviados. Si el remitente no recibe el acuse de recibo, vuelve a enviarlos. Con un servicio no fiable, no se envía ningún reconocimiento, así que no hay ninguna garantía de que los datos enviados fueron recibidos alguna vez.

2.5.1.7 Modos de Transferencia de Datos

Diversas entidades de protocolo ofrecen diferentes modos por los cuales se pueden transferir datos de un proceso a otro a través de la red. Según lo indicado en la tabla 2.5_1, algunos protocolos, tales como el NetBEUI NBFP, soportan más de un modo de transferencia de datos.

Modo de Transferencia de Datos	Tipo del Modo	Definición	Protocolo
Conexión Confiable	Modo de Mensaje	Delimitación y sincronización de mensaje	NetBEUI NBFP, TCP/IP NetBT, NWLink NBIPX
	Modo Byte-stream (flujo de bytes)	Ordenamiento en secuencia y acuse de recibo al nivel de Bytes de datos	TCP/IP TCP
	Modo Packetstream (flujo de paquetes)	Ordenamiento en secuencia y acuse de recibo al nivel de Paquetes de datos	NetBEUI, LLC, MSDLC, NWLink SPX
Sin Conexión, no Confiable	Modo Datagrama		NetBEUI NBFP y LLC; MSDLC, TCP/IP UDP e IP; NWLink IPX

Tabla 2.5_1 Definiciones y Protocolos de Modos de Transferencia de Datos

El Modo de Mensaje de Conexión Confiable

Cuando el cliente transmisor somete para ser enviado un mensaje más grande que el tamaño de paquete, la entidad transmisora divide el mensaje en segmentos con un tamaño de trama adecuado al protocolo. Estos incluyen la información delimitadora del mensaje en el encabezado de protocolo, que identifica donde comienza y termina el mensaje sometido por

el cliente. Este proceso permite que el "driver" (controlador) del protocolo receptor reciba el mensaje entero del cliente antes de indicar los datos al cliente receptor.

Si el buffer del cliente receptor se llena antes de que la entidad receptora haya recibido el mensaje entero, esta todavía proporcionará el mensaje parcial al cliente receptor. También indicará que los datos proporcionados son un mensaje parcial y que el cliente receptor debe proveer otro buffer para recibir la porción restante del mensaje.

Cuando la entidad receptora ha recibido el mensaje entero, devuelve un mensaje a la entidad transmisora, acusando el recibo del mensaje entero.

El Modo Byte-stream de Conexión Confiable

Cuando el cliente que envía somete un mensaje más grande que el tamaño de paquete, el controlador de la entidad transmisora divide el mensaje en segmentos pero no incluye la información delimitadora del mensaje en el encabezado del protocolo. La entidad receptora proporciona los datos al cliente receptor cuando este proporciona los buffers para recibirlos, sin ningún respeto al tamaño original del mensaje.

Cuando la receptora proporciona los datos al cliente receptor, devuelve un byte de reconocimiento o acuse de recibo a la entidad transmisora, reconociendo el recibo de todos los datos hasta un byte especificado. El cliente que envía puede entonces comenzar a enviar el mensaje a partir del último byte reconocido.

El modo Packetstream de Conexión Confiable

En este modo, el cliente que envía puede someter solamente mensajes del tamaño de paquete. El protocolo receptor proporciona los datos, paquete por paquete, en secuencia al cliente receptor.

Mientras que la entidad receptora proporciona una trama de datos al cliente receptor, devuelve un mensaje a la entidad transmisora, reconociendo el recibo de todos los paquetes hasta el número de paquete especificado.

El modo Datagrama sin Conexión no Confiable

El cliente que envía puede someter para ser enviados solamente mensajes del tamaño de paquete. Si la unidad de datos recibida es más grande que el siguiente buffer de recepción del cliente receptor, entonces se coloca en el buffer del cliente receptor tanto como quepa de la unidad de datos y se entrega al cliente. La porción de la unidad de datos recibida que no pudo caber en el buffer del cliente se pierde simplemente; no se devuelve ningún error asociado al cliente receptor. Si la unidad de datos recibida es más pequeña que el buffer del cliente, la entidad receptora pondrá la unidad de datos recibida en el buffer del cliente y proporcionará inmediatamente el mensaje, sin esperar más datos que recibir. No se devuelve ningún acuse de recibo a la entidad transmisora.

El diagrama siguiente ilustra el proceso de la transferencia de datos: 'S' (en el derecho del diagrama) indica un código de status (estado) de retorno exitoso ("SUCCESS") en la recepción. 'E' indica un estado de error en la recepción (ERROR_MORE_DATA) en la transferencia de datos del modo de mensaje.

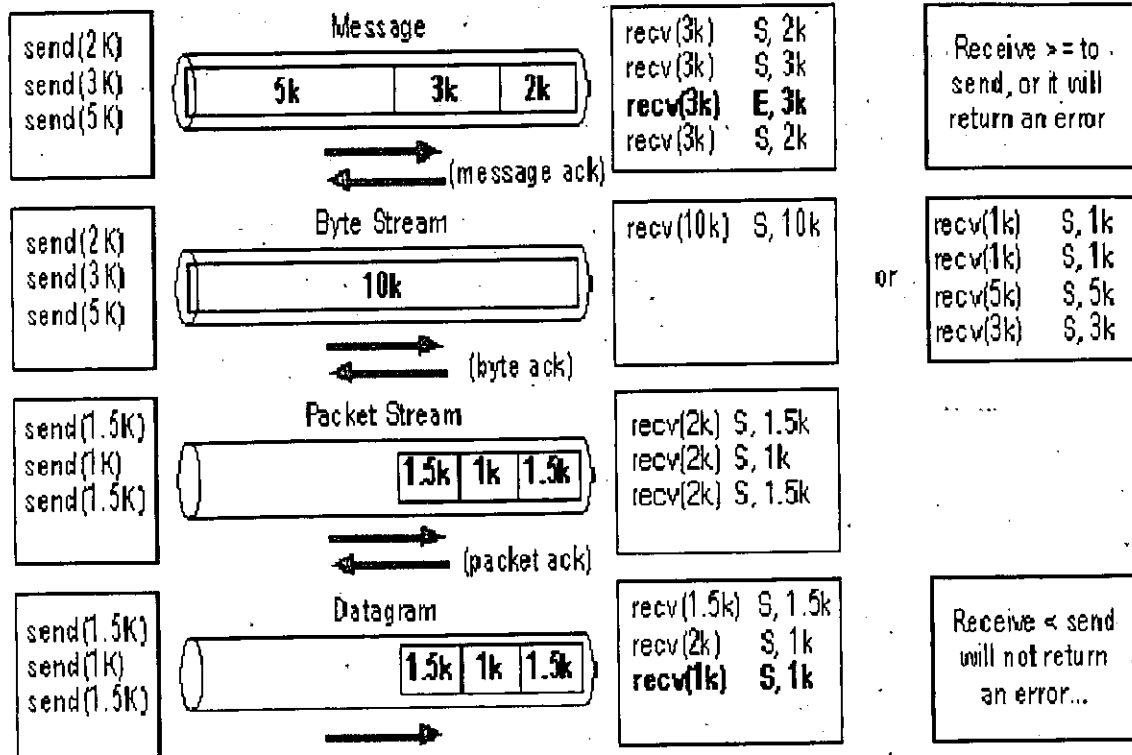


Figura 2.5_5 Modos de Transferencia de Datos

2.5.1.8 Encaminamiento

El proceso de "routing" (encaminamiento) permite que los mensajes (o paquetes de datos) sean entregados de un nodo en una red a un nodo en otra red usando la trayectoria (o ruta) más apropiada y más eficiente. Los ambientes de routing emplean los routers, que funcionan en la capa de red del modelo OSI ya descrito.

Los routers dirigen los paquetes de datos a la red apropiada y los entregan al nodo apropiado en esa red. El router utiliza protocolos de routing, un conjunto de reglas que gobiernan el intercambio de la información entre los nodos, para dirigir los paquetes a su destino. El nodo que envía (designado fuente) y el nodo receptor (designado destino) deben utilizar ya sea los mismos protocolos o un conversor de protocolo para comunicarse. Los routers también permiten que los paquetes de datos sean transmitidos a través de redes disímiles, tales como Ethernet, Token Ring, y FDDI, sin que tengan que ser traducidos.

La determinación de la trayectoria más apropiada y eficiente es específica a cada protocolo. Algunos protocolos determinan la trayectoria por "hop count" (cuenta de salto), que es el número de routers entre la fuente y el destino; otros protocolos determinan la trayectoria analizando el ancho de banda disponible y calculando qué ruta proporciona la mejor calidad del servicio.

En un ambiente de routing, cada destino físico debe ser identificado únicamente. La mayoría de los protocolos de routing se basan en un esquema de direcciones que identifica a cada nodo usando un número de red y de nodo.

Cuando una computadora en una red interconectada desea comunicarse con un nodo en otra red, el software de la capa de red crea un paquete. El paquete contiene los datos que se enviarán, la dirección del remitente (dirección de la fuente), y la dirección del destinatario (dirección del destino). La figura 2.5_6 ilustra cómo los paquetes descomponen la información, tal como el remitente, la dirección del remitente, la dirección de destino, y los datos.

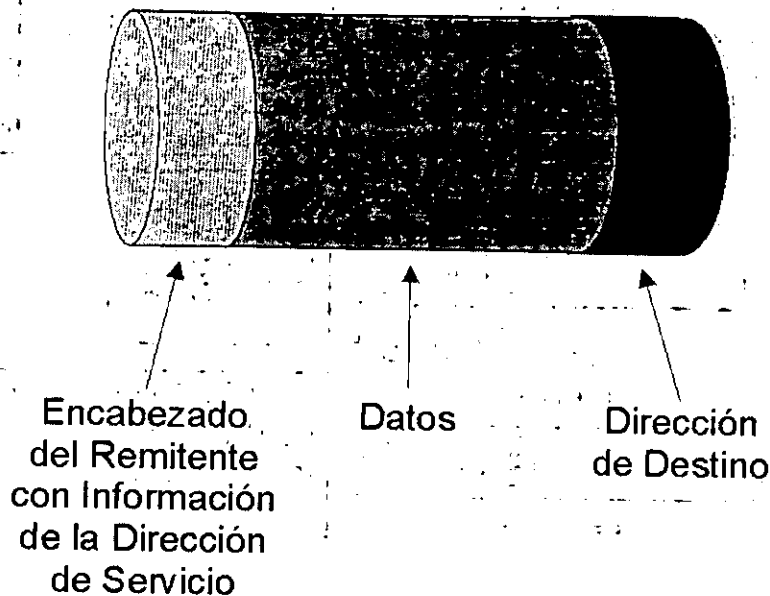


Figura 2.5_6 Estructura de los Paquetes de Datos

El paquete se pone dentro de la trama apropiada para su transmisión a través de la red. El software de la capa de red también determina si el destino reside en la red local o en otra red. Si la dirección de destino está en otra red, el paquete de datos es enviado al router que está conectado a la red local. Cuando el router recibe el paquete de datos, remueve la trama que encapsula al paquete de datos y examina la dirección de destino de los paquetes de la capa de red para encontrar en qué red reside el nodo.

Entonces el router compara la información de la red con su tabla de routing, que contiene direcciones para llegar a varios lugares en la red interconectada. Puede haber un número de redes y de routers entre una fuente y su destino. Cuando la red de destino es encontrada en la tabla, el router pone el paquete de la capa de red en la trama apropiada para la red siguiente sobre la cual será transmitido el paquete, y lo envía a la red siguiente en dirección al nodo de destino. Cada router entre la fuente y la red de destino funciona de esta manera.

Cuando el router conectado a la red que contiene el nodo de destino recibe el paquete, remueve la trama que encapsula el paquete de la capa de red y compara la dirección de red de destino con su tabla de routing. Cuando el router determina que está conectado directamente a la red de destino, examina el número del nodo de destino en el paquete, la cual es comparada con la "Address Mapping Table" (tabla de mapeo de direcciones) del router. Esta tabla correlaciona la dirección de la capa de red del paquete con la dirección de control de acceso de los medios (MAC) del dispositivo. Cuando se encuentra una equivalencia, el router pone los datos del paquete en la trama apropiada y los envía al nodo de destino.

2.5.1.9 Relación de los Principales Protocolos con el Modelo OSI

La tabla 2.5_2 muestra cómo se relacionan algunos de los principales protocolos de comunicación existentes con el modelo OSI. En las siguientes secciones presentaremos específicamente el protocolo TCP/IP.

Capas OSI	Apple Computer	IBM SNA	Microsoft Networking	Novell NetWare	TCP/IP Internet	Protocolos OSI
Aplicación Capa 7	Programas y Protocolos de Aplicación para transferencia de archivos, correo electrónico, etc.					
Presentación Capa 6	AppleTalk Filing Protocol (AFP)	Transaction Services Presentation Services	Server Message Block (SMB)	NetWare Core Protocols (NCP)	Protocolos Especificos de la Aplicación	ISO 8823
Sesión Capa 5	AppleTalk Session Protocol (ASP)	Data Flow Control	Network Basic Input/Output System (NetBIOS)	Network Basic Input/Output System (NetBIOS)	(Telnet, FTP, SNMP, SMTP, ICMP, etc.)	ISO 8327
Transporte Capa 4	AppleTalk Transaction Protocol (ATP)	Transmission Control	Network Basic Extended User Interface (NetBEUI)	Sequenced Packet Exchange (SPX)	Transmission Control Protocol (TCP)	ISO 8073 TP0-4
Red Capa 3	Datagram Delivery Protocol (DDP)	Path Control		Internet Packet Exchange (IPX)	Internet Protocol (IP)	ISO 8473 (CLNP)
Enlace de Datos Capa 2	Tarjetas de Interfaz de Red (NIC): Ethernet, Token-Ring, ARCNET, StarLAN, LocalTalk, FDDI, ATM, etc. Controladores de NIC: Open Datalink Interface (ODI), Network Independent Interface Specification (NDIS)					
Física Capa 1	Medios de Transmisión: Par trenzado, Coaxial, Fibra Óptica, Medios Inalámbricos, etc.					

Tabla 2.5_2 Principales Protocolos y su Relación con el Modelo OSI

2.5.2 TCP/IP

2.5.2.1 Antecedentes

La suite o conjunto llamado "**Transmission Control Protocol / Internet Protocol**" (Protocolo de Control de Transmisión / protocolo de Internet) o **TCP/IP**, es un conjunto estándar de protocolos de red que gobiernan cómo pasan los datos entre las computadoras de la red.

TCP/IP es el protocolo primario de Internet y de su red mundial (World Wide Web). Es también el protocolo primario para muchas redes privadas interconectadas, que son redes que conectan entre sí a redes de área local (LANs). TCP/IP es el transporte preferido para la interconexión de redes y provee interoperabilidad entre redes basadas en TCP/IP, por ejemplo con sistemas UNIX o WINDOWS.

2.5.2.2 Ventajas de usar TCP/IP

TCP/IP ofrece las ventajas siguientes:

- Una arquitectura robusta, escalable, multiplataforma, de tipo cliente-servidor.
- Un protocolo de red estándar, encaminable (**routable**) que es el protocolo más completo y más aceptado que está disponible. Todos los sistemas operativos modernos ofrecen soporte de TCP/IP, y la mayoría de las redes grandes confían en TCP/IP para mucho de su tráfico de red.
- Una tecnología para conectar sistemas disímiles. Muchas utilerías estándar de conectividad están disponibles para tener acceso y para transferir datos entre los sistemas disímiles, incluyendo el Protocolo de Transferencia de Archivo (**FTP**) y el Protocolo de Emulación de Terminal (**Telnet**).
- La tecnología necesaria para habilitar la conexión de las computadoras con el Internet global. El "**Point to Point Protocol**" (Protocolo Punto a Punto) de TCP/IP proporciona la base necesaria para conectarse y para utilizar los servicios de Internet.

2.5.2.3 La Arquitectura TCP/IP

TCP/IP, como la mayoría del software de redes, se modela en capas. Esta representación en capas conduce al término "**protocol stack**" (pila de protocolos) que es sinónimo de Suite de protocolos. Puede ser utilizada para situar (pero no para comparar funcionalmente) la Suite del protocolo de TCP/IP en relación con otras, tales como SNA y OSI. No se pueden extraer comparaciones fácilmente de esto, pues hay diferencias básicas en los modelos de capas usados por las diversas suites de protocolos. Según lo demostrado en la figura 2.5_7, cada capa en el modelo de TCP/IP corresponde a una o más capas del modelo ISO/OSI.



Figura 2.5_7 TCP/IP y el Modelo OSI

2.5.2.4 La Suite de Protocolos TCP/IP

Los protocolos TCP/IP se delinear en un modelo conceptual de cuatro capas: Aplicación, Transporte, Inter-red o Internet, e Interfaz de red. Este modelo se conoce oficialmente como la Suite del protocolo de TCP/IP Internet, pero es referida a menudo como la familia de protocolos de TCP/IP. Cada capa representa un paquete de funciones.

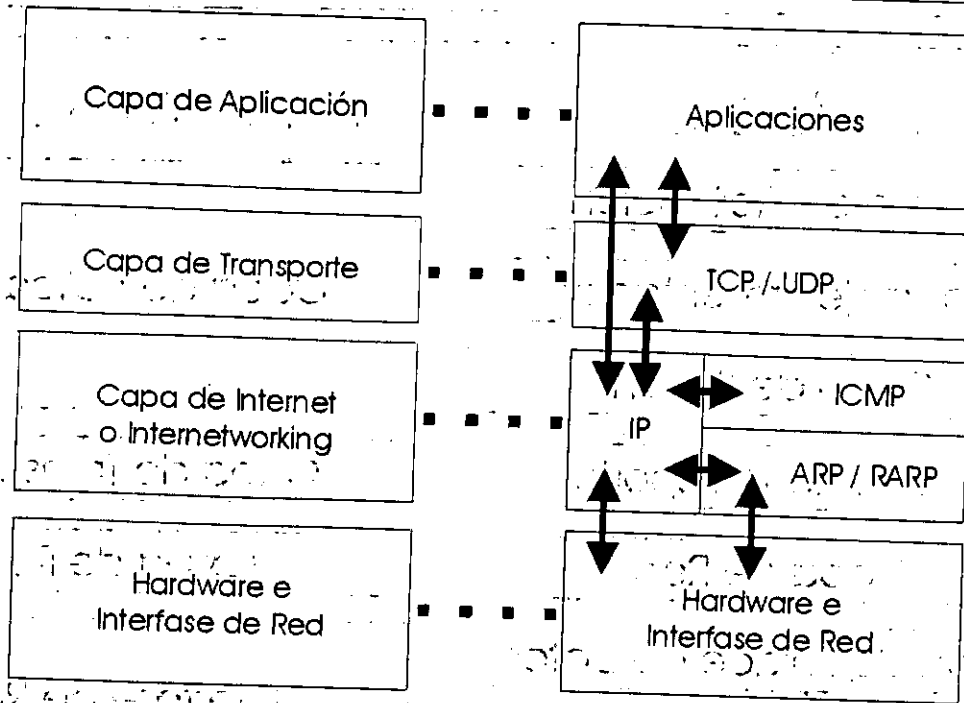


Figura 2.5_8 Suite de Protocolos y Arquitectura TCP/IP

Capa de Aplicación

Es un proceso de usuario cooperando con otro proceso en el mismo o un diverso "host" (anfitrión). Ejemplos de esto son TELNET (un protocolo para la conexión y emulación de terminal remota), FTP (protocolo de transferencia de archivos) y SMTP (protocolo simple de transferencia de correo).

Capa de Transporte

Esta capa proporciona la transferencia de datos "end-to-end" (de extremo a extremo). Ejemplos de esto son los protocolos TCP ("connection-oriented" u orientado a conexión) y UDP ("connectionless" o sin conexión).

Capa de Inter-Red

También llamada la capa de Internet o la capa de red, la capa de inter-red proporciona la imagen de "red virtual" de Internet (es decir, esta capa aísla los niveles más altos de los de la arquitectura de red típica debajo de ella). El protocolo de Internet (IP) es el protocolo más importante de esta capa.

Capa de Interfaz de Red

También llamado la capa de enlace o capa de enlace de datos, la capa de interfaz de red es la interfaz con el hardware real de la red. Esta interfaz puede o puede no proporcionar la entrega confiable, y puede ser orientada a paquetes o "stream oriented" (orientada a corriente de datos). De hecho, TCP/IP no especifica ningún protocolo aquí, sino que puede utilizar casi cualquier interfaz de red disponible, lo cual ilustra la flexibilidad de la capa de IP. Ejemplos de estas interfaces son

IEEE 802,2, X.25 (que es confiable en sí mismo), ATM, FDDI, redes de paquetes de radio (tales como AlohaNet) e incluso SNA.

Las interacciones entre las diversas capas se muestran por las flechas en la figura 2.5_8. Un modelo de capas más detallado se demuestra en la figura 2.5_9 y todos los protocolos mencionados arriba son discutidos en detalle en las siguientes secciones.

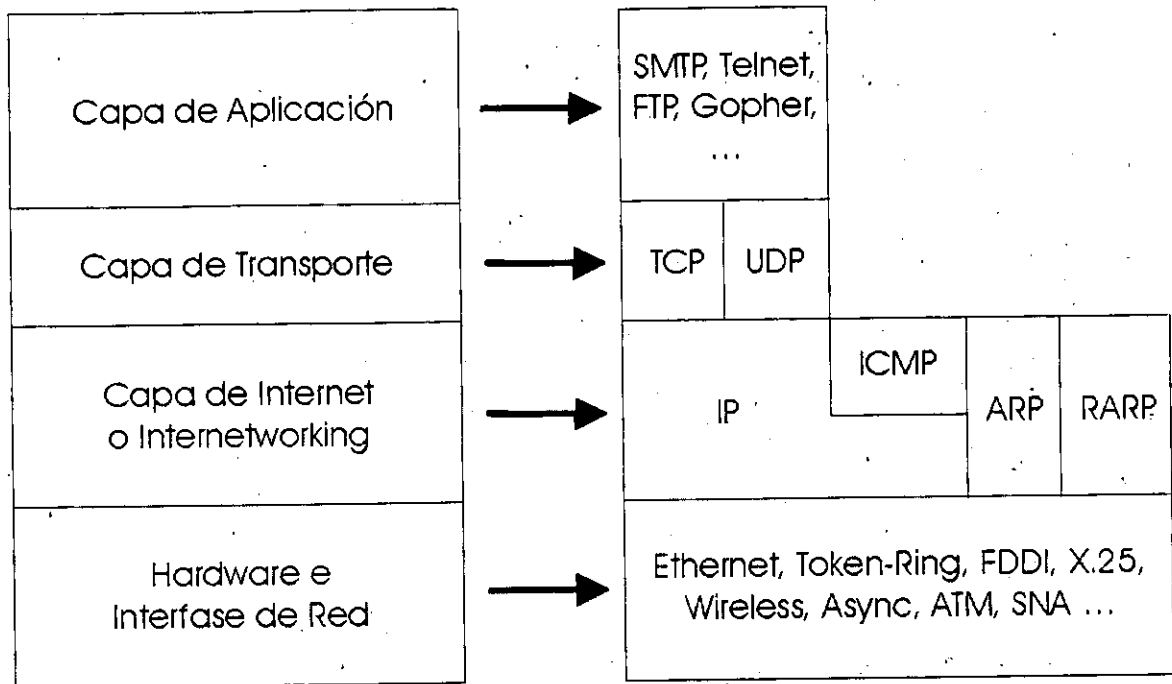


Figura 2.5_9 Modelo Detallado de la Arquitectura TCP/IP

Definido dentro de las cuatro capas de TCP/IP está el juego de protocolos de Internet que dictan cómo se comunican las computadoras y cómo se interconectan las redes. Los más comunes de estos protocolos son el "Transmission Control Protocol" (protocolo de control de transmisión) o TCP, "User Datagram Protocol" (protocolo de datagrama de usuario) o UDP, "Internet Protocol" (protocolo de Internet) o IP, "Address Resolution Protocol" (protocolo de resolución de dirección) o ARP, e "Internet Control Message Protocol" (protocolo de mensajes de control de Internet) o ICMP. Las secciones siguientes introducen estos protocolos.

Protocolo de Control de Transmisión

El protocolo de alto nivel más común de la suite TCP/IP es el protocolo de control de transmisión (TCP). Proporciona un servicio de entrega de paquetes confiable, "connection-oriented" (orientado a conexión), encima (o encapsulado dentro) de IP. TCP garantiza la entrega de paquetes, asegura la secuencia apropiada de los datos, y proporciona una característica de "checksum" (suma de comprobación) que valida la exactitud del encabezado del paquete y de sus datos. Si la red corrompe o pierde un paquete del TCP durante la transmisión, el TCP es responsable de retransmitir el paquete faltante. Esta

confiabilidad hace de TCP el protocolo a elegir para la transmisión de datos basados en sesión, para aplicaciones cliente-servidor, y para servicios críticos, tales como correo electrónico.

Esta confiabilidad tiene un precio. Los encabezados del TCP requieren bits adicionales para proporcionar la secuencia apropiada de la información, así como una suma de comprobación (checksum) obligatoria para asegurar la confiabilidad tanto del encabezado del paquete de TCP como de los datos del paquete. Para garantizar la entrega exitosa de los datos, el protocolo también requiere que el recipiente reconozca (acknowledge) el recibo exitoso de los datos.

Tales reconocimientos (ACKs) generan tráfico adicional en la red, disminuyendo, en favor de la confiabilidad, la velocidad a la cual pasan los datos. Para reducir el impacto en el desempeño, la mayoría de los hosts envían un reconocimiento por cada otro segmento o cuando un intervalo especificado de tiempo ha pasado.

Protocolo de Datagrama de Usuario

Si la confiabilidad no es esencial, el protocolo de datagrama de usuario (UDP), un complemento del TCP, ofrece un servicio de datagrama sin conexión que no garantiza ni la entrega ni la secuencia correcta de los paquetes entregados (muy similar a como lo hace IP). Los protocolos de alto nivel o de las aplicaciones pueden proporcionar mecanismos de confiabilidad en adición a UDP/IP. Son opcionales las sumas de comprobación (checksums) de los datos de UDP, proporcionando una manera de intercambiar datos a través de redes altamente confiables, sin consumir innecesariamente recursos de la red o tiempo de proceso. Cuando se utilizan las sumas de comprobación de UDP, validan la integridad del encabezado y de los datos. El protocolo de UDP no hace cumplir los ACKs; esto se deja a los protocolos de alto nivel. UDP también soporta el envío de datos de un solo remitente a múltiples receptores.

Protocolo de Internet

El protocolo de Internet (IP) proporciona la entrega del paquete para el resto de los protocolos dentro de la Suite TCP/IP. Proporciona un sistema de entrega sin conexión, basada en el mejor esfuerzo, para los datos de la computadora. Es decir, los paquetes de IP no están garantizados para llegar a su destino, ni son garantizados para ser recibidos en la secuencia en la cual fueron enviados. La característica de suma de comprobación del protocolo confirma solamente la integridad del encabezado de IP. IP es un protocolo sin conexión que no asume confiabilidad de las capas más bajas; esto es, no proporciona confiabilidad, control de flujo o recuperación de errores. Estas funciones se deben proporcionar en un nivel más alto, en la capa de transporte usando el TCP como el protocolo de transporte, o en la capa de aplicación si UDP se utiliza como el protocolo de transporte. Así, la responsabilidad de los datos contenidos dentro del paquete de IP (y de su secuencia) es asegurada solamente usando protocolos de alto nivel. Una unidad de mensaje en una red de IP se llama un datagrama de IP. Ésta es la unidad básica de información transmitida a través de redes TCP/IP.

Protocolo de Resolución de Dirección

No relacionado directamente con el transporte de los datos, pero importante no obstante, el protocolo de resolución de dirección (ARP) es uno de los protocolos de mantenimiento que soporta la Suite TCP/IP y es generalmente invisible a los usuarios y a las aplicaciones.

Si dos sistemas deben comunicarse a través de una red TCP/IP, el sistema que envía el paquete debe traducir la dirección de IP del destino final a la dirección física del destino final. IP obtiene esta dirección física difundiendo ("**broadcasting**") un paquete especial de búsqueda (un paquete de petición de ARP) que contiene la dirección de IP del sistema de destino. Todos los sistemas habilitados con ARP en la red local de IP detectan estos mensajes de difusión, y el sistema que posee la dirección de IP en cuestión responde enviando su dirección física al solicitante (en un paquete de contestación de ARP). La dirección física entonces se almacena en el "**cache**" (almacenamiento de alta velocidad) de ARP del sistema solicitante para su uso subsiguiente. Debido a que la contestación del ARP puede también ser difundida a la red, otros sistemas en la red pueden utilizar esta información para poner al día sus propios caches de ARP (se puede utilizar el programa **arp** para examinar las tablas del ARP).

Protocolo de Mensajes de control de Internet

El protocolo de mensajes de control de Internet (ICMP) es otro de los protocolos de mantenimiento. Permite a dos sistemas en una red IP compartir información de error y de status. Esta información se puede utilizar por protocolos de alto nivel para recuperarse de problemas de la transmisión o por los administradores de la red para detectar problemas en la red. Aunque los paquetes de ICMP se encapsulan dentro de los paquetes de IP, no son considerados un protocolo de alto nivel (ICMP se requiere en cada puesta en práctica o implantación de una red de IP.) El programa **ping** utiliza los paquetes de "**echo request**" (petición de eco) y de "**echo reply**" (respuesta de eco) de ICMP para determinar si un sistema IP específico en una red está funcionando. Por esta razón, el programa **ping** es útil para diagnosticar fallas de la red IP o de los routers.

2.5.2.5 El Núcleo de la tecnología de TCP/IP

La suite de protocolos TCP/IP y los servicios relacionados incluyen los siguientes.

- El núcleo de los protocolos de TCP/IP ya comentados: el protocolo de control de transmisión (**TCP**), el protocolo de Internet (**IP**), el protocolo del datagrama del usuario (**UDP**), el protocolo de resolución de dirección (**ARP**), y el protocolo de mensajes de control de Internet (**ICMP**). También se tienen los protocolo punto a punto (**PPP**), e IP de Línea Serial (**SLIP**), que son protocolos usados para el acceso por líneas conmutadas a las redes de TCP/IP, incluyendo el Internet.
- Soporte para la programación de interfaces de red tales como llamadas remotas a procedimientos (**RPC**).
- Utilerías básicas de conectividad de TCP/IP, incluyendo **finger**, **ftp**, **lpr**, **rcp**, **rexec**, **rsh**, **telnet**, y **tftp**. Estas utilerías permiten a usuarios que utilizan diversas plataformas el interactuar con y usar los recursos de diferentes "**hosts**" (servidores).
- Herramientas de diagnóstico de TCP/IP, incluyendo **arp**, **hostname**, **ipconfig**, **lpq**, **nbtstat**, **netstat**, **ping**, **route**, y **tracert**. Se utilizan estas utilerías para detectar y para resolver problemas en una red TCP/IP.
- Servicios y herramientas administrativas relacionadas, incluyendo el servicio de "**Dynamic Host Configuration Protocol**" (Protocolo Dinámico de Configuración del Anfitrión) o **DHCP**, para asignar automáticamente las direcciones de IP que identifican unívocamente cada uno de los ordenadores de una red TCP/IP; el servicio de "**Domain Name System (DNS) Server**" (Servidor del Sistema de Nombres de dominio) para registrar y consultar los nombres de dominio de DNS en una interconexión de redes; y TCP/IP "**Printing**" (impresión) para tener

acceso a las impresoras conectadas a las computadoras de la red o conectadas directamente a la red con un adaptador dedicado de red.

- "Simple Network Management Protocol" (Protocolo Simple de Administración de Red) o **SNMP**. Este componente permite que una computadora sea monitoreada remotamente con herramientas de administración tales como Net Manager de Sun, Open View de HP, o Tivoli de IBM.
- "Simple Mail Transfer Protocol" (Protocolo Simple de Transferencia de Correo) o **SMTP**, para manejo de correo electrónico.
- Protocolos de red simples, incluyendo "Character Generator" (generador de carácter), "Daytime" (hora y día), "Discard" (desechar) y Echo. Estos protocolos permiten que una computadora responda a las peticiones de otros sistemas que apoyen estos protocolos.
- "Path MTU Discovery" (descubrimiento de la trayectoria MTU), que proporciona la capacidad de determinar el tamaño del datagrama para todos los routers, entre las computadoras de la red. También se tiene el "Internet Group Management Protocol" (Protocolo de Administración de Grupos de Internet) o **IGMP**, que es utilizado por productos de software para "workgroups" (grupos de trabajo).

Muchas utilerías y aplicaciones de servicios de servidor ("*daemons*") están disponibles en el dominio público o a través de diferentes proveedores.

2.5.2.6 RFCs

Los "Requests for Comments" (requerimientos de comentarios) o RFCs son una serie en evolución de informes, propuestas de protocolos, y estándares de protocolo usados por la comunidad de Internet. Los estándares de TCP/IP se definen en RFCs publicados por la "Internet Engineering Task Force" (Fuerza de Tarea de Ingeniería de Internet) o IETF y otros grupos de trabajo. La tabla 2.5_3 enumera los principales RFCs.

RFC	Título
768	User Datagram Protocol (UDP)
783	Trivial File Transfer Protocol (TFTP)
791	Internet Protocol (IP)
792	Internet Control Message Protocol (ICMP)
793	Transmission Control Protocol (TCP)
816	Fault Isolation and Recovery
826	Address Resolution Protocol (ARP)
854	Telnet Protocol (TELNET)
862	Echo Protocol (ECHO)
863	Discard Protocol (DISCARD)
864	Character Generator Protocol (CHARGEN)
865	Quote of the Day Protocol (QUOTE)
867	Daytime Protocol (DAYTIME)
894	IP over Ethernet
919, 922	IP Broadcast Datagrams (broadcasting with subnets)
950	Internet Standard Subnetting Procedure

Tabla 2.5_3 Requests for Comments (RFCs) de TCP/IP

RFC	Título
959	File Transfer Protocol (FTP)
1001, 1002	NetBIOS Service Protocols
1034, 1035	Domain Name System (DNS)
1042	IP over Token Ring
1055	Transmission of IP over Serial Lines (IP-SLIP)
1112	Internet Group Management Protocol (IGMP)
1122, 1123	Host Requirements (communications and applications)
1134	Point to Point Protocol (PPP)
1144	Compressing TCP/IP Headers for Low-Speed Serial Links
1157	Simple Network Management Protocol (SNMP)
1179	Line Printer Daemon Protocol
1188	IP over FDDI
1191	Path MTU Discovery
1201	IP over ARCNET
1231	IEEE 802.5 Token Ring MIB (MIB-II)
1332	PPP Internet Protocol Control Protocol (IPCP)
1334	PPP Authentication Protocols
1518	An Architecture for IP Address Allocation with CIDR
1519	Classless Inter-Domain Routing (CIDR): An Address Assignment and Aggregation Strategy
1533	DHCP Options and BOOTP Vendor Extensions 1
1534	Interoperation Between DHCP and BOOTP
1541	Dynamic Host Configuration Protocol (DHCP)
1542	Clarifications and Extensions for the Bootstrap Protocol 2
1547	Requirements for Point to Point Protocol (PPP)
1548	Point to Point Protocol (PPP)
1549	PPP in High-level Data Link Control (HDLC) Framing
1552	PPP Internetwork Packet Exchange Control Protocol (IPXCP)
1553	IPX Header Compression
1570	Link Control Protocol (LCP) Extensions
Borradores de RFCs	NetBIOS Frame Control Protocol (NBFCP); PPP over ISDN; PPP over X.25; Compression Control Protocol

Tabla 2.5_3 (Cont.) Requests for Comments (RFCs) de TCP/IP

2.5.2.7 Direcciones de IP

Para recibir y entregar con éxito los paquetes entre computadoras, TCP/IP requiere que cada computadora esté configurada con tres valores proporcionados por el administrador de la red: una dirección de IP, una máscara de sub-red, y un "gateway" (puerta de comunicaciones) o router por default (por omisión).

Cada dispositivo conectado a una red TCP/IP es identificado por una dirección única de IP (si una computadora tiene adaptadores múltiples de red, cada adaptador tendrá su propia dirección de IP). Esta dirección se representa típicamente en la notación punto-decimal, es decir, con el valor decimal de cada octeto (ocho bits, o un byte) de la dirección separado por un punto. Aquí se tiene un ejemplo de una dirección de IP: 138.57.7.27.

Debido a que las direcciones de IP identifican los dispositivos de una red, cada dispositivo en la red debe tener asignada una dirección única de IP. En los dos siguientes apartados se explica la estructura de la dirección de IP.

Identificación de la Red e identificación del Host

Aunque una dirección de IP es un valor único, contiene dos porciones de información: el "Network ID" (identificación de la red) y el "Host ID" (identificación de host) de la computadora.

El Network ID identifica los sistemas que están situados en la misma red física. Todos los sistemas en la misma red física deben tener el mismo Network ID, y el Network ID debe ser único en la interconexión de redes.

El Host ID identifica una estación de trabajo, servidor, router, u otro dispositivo de TCP/IP dentro de una red. La dirección para cada dispositivo debe ser única para un Network ID dado.

Una computadora conectada a una red TCP/IP utiliza el Network ID y el Host ID para determinar qué paquetes debe recibir o ignorar y para determinar el alcance de sus transmisiones (solamente las computadoras con el mismo Network ID aceptan entre sí mensajes de "broadcast" (difusión) de nivel IP.)

Las redes que se conectan con el Internet público deben obtener un Network ID oficial del "Internet Network Information Center" (centro de información de la red Internet) (InterNIC), para garantizar la unicidad de los Network ID de IP. Después de recibir una identificación de la red, el administrador local de la red debe asignar los Host IDs únicos para las computadoras dentro de la red local. Aunque las redes privadas no conectadas con Internet pueden utilizar su propio identificador de la red (intranet), la obtención de un Network ID válido del InterNIC permite que una red privada se conecte con Internet en el futuro sin reasignar direcciones.

La comunidad de Internet ha definido *clases* de dirección para acomodar redes de tamaños variables. La clase de dirección se puede discernir del primer octeto de una dirección de IP. La tabla 2.5_4 resume la relación entre el primer octeto de una dirección dada y sus campos de Network ID y de Host ID. También identifica el número total de Network IDs y de Host IDs para cada clase de dirección que participa en el esquema de direcciones de Internet. Este ejemplo utiliza w.x.y.z para designar los bytes de la dirección de IP.

Clase	Valores w (1,2)	Network ID	Host ID	Redes Disponibles	Hosts Disponibles por Red
A	1-126	w	x.y.z	126	16,777,214
B	128-191	w.x	y.z	16,384	65,534
C	192-223	w.x.y	z	2,097,151	254

Notas

- 1 - Rango inclusivo para el primer octeto en la dirección de IP.
- 2 - La dirección 127 se reserva para pruebas de "loopback" (retorno de lazo) y para la comunicación entre procesos en la computadora local; no es una dirección de red válida. Las direcciones 224 y superiores están reservadas para protocolos especiales ("Internet Group Management Protocol Multicast" y otros) y no se pueden utilizar como direcciones de Host.

Tabla 2.5_4 Clases de Direcciones de IP

Máscaras de Sub-Red

Las "Subnet masks" (máscaras de sub-red) son valores de 32 bits que permiten que el recipiente de los paquetes de IP distinga la porción de Network ID de la de Host ID en la dirección de IP. Las máscaras de sub-red son creadas asignando un 1 a los bits del Network ID y un 0 a los bits del Host. El valor de 32 bits es entonces convertido a la notación punto-decimal, según lo demostrado en la tabla 2.5_5.

Clase de Dirección	Bits para la Máscara de Sub-Red	Máscara de Sub-Red
Clase A	11111111 00000000 00000000	255.0.0.0
Clase B	11111111 11111111 00000000	255.255.0.0
Clase C	11111111 11111111 11111111	255.255.255.0

Tabla 2.5_5 Máscaras de Sub-Red por Default para Clases Estándar de Direcciones IP

Por ejemplo, cuando la dirección de IP es 138.57.7.27 y la máscara de sub-red es 255.255.0.0, el Network ID es 138.57 y el Host ID es 7.27.

Debido a que la clase de un host se determina fácilmente, configurar un host con una máscara de sub-red puede parecer redundante. Pero las máscaras de sub-red también se utilizan para dividir un Network ID asignado en segmentos adicionales entre varias redes locales. A veces solamente se necesita segmentar porciones de un octeto en segmentos usando solo algunos pocos bits para especificar los subnet IDs (identificaciones de Sub-Red).

Para prevenir problemas de direccionamiento y de routing, todas las computadoras en una red lógica deben utilizar las mismas máscaras de sub-red y un mismo network ID.

2.5.2.8 Encaminamiento de IP

Internet y otras redes TCP/IP son interconectadas por routers, que son los dispositivos que pasan los paquetes de IP de una red a otra.

Para cada computadora en una red IP, se puede mantener una tabla con una entrada para cada otra computadora o red con las cuales la computadora local se comunique. En general, esto no es práctico, y el gateway (router) de default se utiliza en su lugar. El gateway de default es una computadora conectada con la sub-red local y con otras redes. Tiene conocimiento de los Network IDs para otras redes en la interconexión de redes y cómo alcanzarlas. Se necesita solamente para las computadoras que son parte de una interconexión de redes.

Cuando IP se prepara para enviar un paquete, inserta la dirección local (fuente) de IP y la dirección de destino del paquete en el encabezado de IP. Después examina la dirección de destino, la compara con una "route table" (tabla de rutas) mantenida localmente, y toma la acción apropiada basándose en lo que encuentra. Hay tres acciones posibles:

- Puede pasar el paquete hasta una capa de protocolo arriba de IP en el host local.
- Puede remitirlo a través de uno de los adaptadores de red conectados localmente.
- Puede desécharlo.

La búsqueda para una coincidencia de la dirección de destino en la tabla de rutas procede de lo específico a lo general en el orden siguiente:

- Se examina la tabla buscando una coincidencia exacta (ruta del host).
- La porción del host es removida de la dirección de destino, y se examina la tabla buscando una coincidencia (ruta de la sub-red).
- La porción de sub-red es removida de la dirección de destino, y se examina la tabla buscando una coincidencia (ruta de la red).
- Se utiliza el gateway de default.
- Si no se ha especificado un gateway de default, se desecha el paquete.

Debido a que el Gateway de default contiene la información sobre los Network IDs de las otras redes en la interconexión de redes, puede remitir el paquete a otros routers hasta que el paquete se entrega eventualmente a un router conectado con el destino especificado. Este proceso se conoce como *routing* (encaminamiento) y se ilustra en la figura 2.5_10.

Si el gateway de default llega a estar no disponible, la comunicación más allá de la sub-red local puede estropearse. Para prevenir esto, se pueden especificar múltiples gateways de default, o usar la utilería "route" para agregar manualmente rutas a la tabla de rutas para los sistemas o redes muy usados.

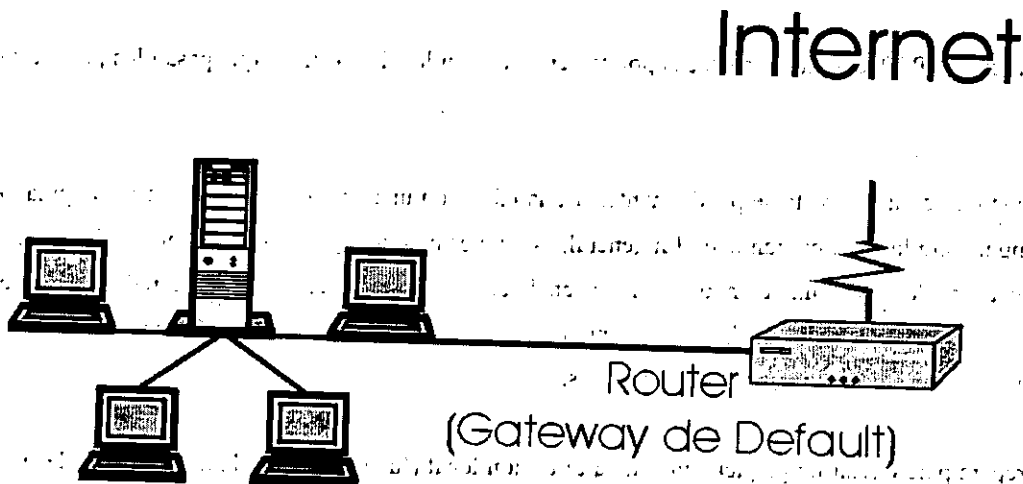


Figura 2.5_10 Routers (gateway de Default)

2.6 Características de Progress y Herramientas de Desarrollo 4GL

Progress cuenta con una Base de Datos Relacional (RDBMS) la cual es de alto rendimiento y escalable desde sistemas Windows 95 de un solo usuario hasta sistema de multiprocesamiento simétrico masivos (SMP) y ccNUMA, soportando miles de usuarios concurrentes. Además de ofrecer el costo mas bajo de propiedad y alta escalabilidad absoluta, un rendimiento inigualable y una portabilidad de plataforma excelente. Con soporte completo para las especificaciones ANSI SQL-92, los productos Progress RDBMS se integran con aplicaciones empresariales, herramientas de terceros y diferentes sistemas de administración de base de datos.

Los tres Productos que integran la línea Progress RDBMS son:

- Progress Enterprise 31 a miles de usuarios
- Progress Workgroup 2-30 usuarios
- Progress Personal 1 usuario

Estos productos le permiten elegir una solución que satisfaga sus objetivos de negocio, de acuerdo al tamaño y requerimientos de su organización. El usuario paga por lo que necesita hoy en día y conforme sus requerimientos crecen, puede actualizarse a una solución más robusta sin generar ningún cambio en el código de los programas.

2.6.1 Progress Enterprise RDBMS

Progress Enterprise RDBMS esta diseñado para grandes ambientes empresariales y el procesamiento de transacciones a través de las aplicaciones más demandadas de hoy día, basadas en procesamiento de transacciones en línea (OLTP) utilizando SQL y Progress 4GL (Ver la figura 2.6_1). Cimentado en una arquitectura flexible, de multienlaces y multiservidores, Progress Enterprise RDBMS es una base de datos empresarial a gran escala, poderosa y abierta, que puede ejecutarse a través de múltiples plataformas de hardware y redes.

Progress Enterprise RDBMS incluye toda la funcionalidad necesaria para satisfacer los requerimientos OLTP más demandantes. Estas capacidades incluyen bloqueo en el ámbito de registro, recuperación "roll-back" y "roll-forward" (hacia atrás y hacia adelante), recuperación "point-in-time" (a un valor de tiempo dado), administración de la base de datos con "two-phase commit", un juego completo de utilerías en línea y soporte completo para ANSI estándar SQL-92. Una combinación de poder, flexibilidad y fácil operación hacen de Progress Enterprise RDBMS un sistema ideal para su amplio rango de aplicaciones comerciales y de procedimiento de datos.

Procesamiento de pedidos

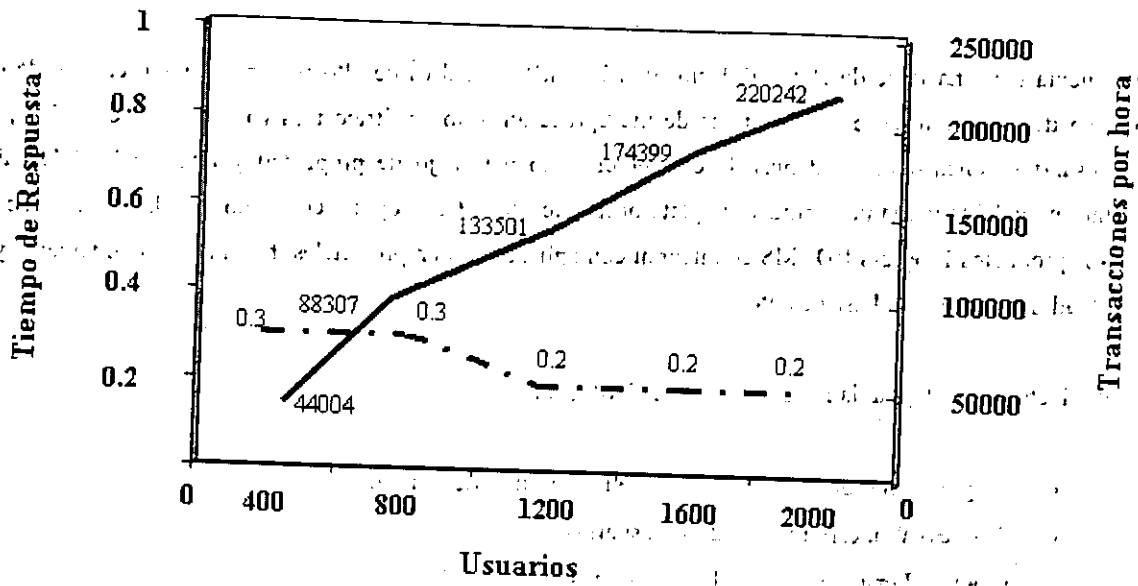


Figura 2.6_1 Benchmark de Transacciones en Línea con HP y Progress

2.6.1.1 Soporta Múltiples Sistemas Operativos y Redes

Para satisfacer los requerimientos de redes empresariales complejas, Progress Enterprise RDBMS corre en un amplio rango de plataformas de hardware y sistemas operativos, que incluyen desde PC's ejecutando Windows NT hasta sistemas de multiprocesadores.

Diferentes Benchmarks (procedimientos de comparación) muestran una escalabilidad comprobada desde 1 hasta 32 procesadores en ambas arquitecturas, SMP y ccNUMA.

2.6.1.2 Características

- Solución altamente escalable que proporciona un rendimiento óptimo.
- Soporta hasta 10,000 usuarios concurrentes y para bases de datos con más de 80 millones de terabytes. Estas características, combinadas con soporte para sistemas operativos de 62 bits, proporcionan una capacidad excepcional para gran escalabilidad y alto rendimiento.
- Soporte para configuraciones de hasta 32 procesadores SMP y ccNUMA. La optimización del sistema de bloqueo de memoria compartida y el "spin locks" de SMP proporcionan un alto procesamiento concurrente para usuarios interactivos en línea al reducir el tiempo de espera de la memoria compartida. Las ventajas de ejecución resultantes permiten a Progress Enterprise RDBMS escalar efectivamente desde el servidor más pequeño hasta los servidores empresariales más grandes.

- Una arquitectura de alto rendimiento, multienlazada y de multiservidor. Esta única arquitectura proporciona un uso eficiente de los recursos del sistema y maximiza la concurrencia al mismo tiempo que minimiza el uso de la red.
- Un conjunto de "buffers" que permite a una porción grande de la base de datos estar en la memoria. Soporta hasta 125 millones de buffers o 128 gigabytes, en un ambiente de sistema operativo de 64 bits, proporciona rápido acceso a la memoria, a las partes utilizadas más frecuentemente de la base de datos y finalmente reduce el acceso al disco.
- Soporte para múltiples sistemas operativos y plataformas de Hardware.
- Conformidad con el ANSI SQL-92, más extensiones.
- APIs para acceso de clientes SQL.
- Asignación de un conjunto dinámico de buffers para procesos en lotes y usuarios interactivos. Permitiendo que un proceso en lotes o un usuario en línea aislado haga un recorrido de tablas o de una pequeña porción del conjunto de "buffers", permitiendo que tareas complicadas de reporte y escaneo no afecten el "pool" de "buffers" de la base de datos. Esto a su vez reduce en gran parte cualquier impacto que pudiera tener el sistema durante una operación a gran escala de escaneo de tablas.
- Almacenamiento virtual de datos ilimitados, su única limitante podría ser el hardware.
- Bloqueo en el ámbito de registro para reducir contención de los recursos y así maximizar el rendimiento. Esta característica permite compartir los datos de una manera eficiente y segura en sistemas de múltiples usuarios al eliminar el bloqueo de una página de información completa del espacio de registros.
- Registro Óptimo de Transacción. Al registrar solo transacciones recuperables, Progress Enterprise RDBMS permite deshacer pequeños registros de bitácora y una reutilización mejorada del espacio de registros.
- Procesos del servidor autoajustables. Características autoajustables, como escritorios asíncronos de páginas, ajustan su comportamiento basado en la actividad de la base de datos. Estas características optimizan el rendimiento aún en instancias que involucran un cambio dramático, como el cambiar de procesamiento en línea a procesamiento en "batch" (por lote).

2.6.1.3 Protege los datos y proporciona alta disponibilidad

- Recuperación automática de caídas. Progress Enterprise RDBMS realiza una recuperación automática cuando la base de datos es re iniciada después de una falla. La recuperación es eficiente y sin esfuerzo.

- Registro "roll-back". Esta característica ofrece protección para la base de datos en caso de cualquier falla. La base de datos utiliza el registro "roll-back" durante la recuperación de caída para aplicar los cambios finalizados a la misma y deshacer todas las transacciones no finalizadas.
 - Registro "roll-forward". La característica "roll-forward", replica las transacciones finalizadas desde el momento del respaldo, permitiendo una recuperación "point-in-time" y soporte a prueba de fallas. El soporte a prueba de fallas para ambas configuraciones "shred-disk" y "shred-nothing".
 - Respaldo en línea completo y en aumento. Para eliminar la necesidad de programar tiempo de inactividad durante el periodo de respaldo, Progress Enterprise RDBMS soporta el respaldo en línea durante el procesamiento normal. La alta disponibilidad lograda por ésta característica permite soportar aplicaciones sin necesidad de tener la operación de los sistemas.
 - Reorganización de tablas en línea e índices. Progress Enterprise RDBMS proporciona la habilidad para reorganizar una tabla mientras que el sistema está en operación y los usuarios están operando en la tabla sin ningún bloqueo de escritura. Los usuarios pueden continuar el procesamiento sin ninguna restricción mientras los índices son reconstruidos.
 - Recuperación "point-in-time". La base de datos puede ser actualizada hacia un punto determinado adelante de un respaldo. Esto permite la recuperación de un error inadvertido del usuario o algún otro tipo de falla no perteneciente al sistema.
- Progress Enterprise RDBMS cumple con el estándar ANSI SQL-92, proporcionando un ambiente abierto que permite la integración eficiente con herramientas de terceros, como herramientas de desarrollo, generadores de reportes y OLAP. Incorpora acceso a clientes SQL, APIs para ODBC, JDBC y SQL/C y provee extensiones SQL para contar con soluciones más robustas.
- El registro de transacciones minimiza el tiempo de espera de los recursos y el uso del espacio. La base de datos conecta solamente aquellas operaciones que necesitan ser completamente recuperables. Por ejemplo, cuando un usuario se conecta a la base de datos para ejecutar un reporte, no hay nada registrado en el registro "roll-back". Debido a que el usuario genera una operación recuperable.
- Adicionalmente, al re escribir consultas antes de una ejecución a través de las características "Predicate Re-write", el sistema SQL mejora el rendimiento y simplifica el entendimiento del desarrollador para la creación de consultas SQL.
- Las extensiones SQL incluidas ofrecen lo siguiente:
- Procedimientos almacenados y Triggers Java.
 - Modelo de seguridad "GRANT/REVOKE".

- Implementación SQL DDL completa incluyendo "ALTER TABLES".
- Vistas actualizables.
- Tablas derivadas.
- Soporte a múltiples esquemas.

2.6.1.4 Simplifica la administración

Además de características autoajustables, Progress Enterprise RDBMS incorpora la herramienta de administración Progress Explorer, la cual administra las bases de datos y servidores a través de la red, simplificando su administración. Progress Explorer soporta generación de guiones de la línea de comandos, permitiendo al administrador de la base de datos almacenarlos en una sola ubicación y utilizarlos para iniciar / parar y administrar sistemas simultáneamente a través de múltiples plataformas.

2.6.1.5 Soporte de Acceso a Base de Datos Distribuidos

Progress Enterprise RDBMS proporciona un acceso fácil y completo a datos almacenados en múltiples bases de datos a través de la red. Las aplicaciones pueden conectarse con 240 bases de datos simultáneamente.

2.6.2 Progress Personal RDBMS

Empaquetado en las herramientas de desarrollo de Progress, se incluyen Progress Provision Plus y Progress Personal RDBMS las cuales son la mejor opción para desarrollar aplicaciones mono usuario, basadas en SQL y 4GL, así como para desarrollar, generar prototipos y probar las aplicaciones.

2.6.3 Especificaciones de Progress RDBMS

Estos parámetros dan un panorama más amplio de la capacidad que tiene Progress RDBMS en las diferentes empresas.

Parámetros máximos de la base de datos	Progress Enterprise RDBMS	Progress Workgroup RDBMS	Progress Personal
Tablas	32,000	32,000	32,000
Índices	32,000	32,000	32,000
Campos x Tabla	32,000	32,000	32,000
Usuarios concurrentes por Base de Datos	Hasta 10,000	Hasta 30	1
Tamaño del Registro en bytes	32,000	32,000	32,000
Conjunto de "buffers" (32 bits)	125,000,000 buffers 128 Gbytes (64 bits) 500,000 buffers 4 gigabytes (32 bits)	500,000 buffers 4Gbytes (32 bits)	500,000 buffers 4 Gigas (32 bits)

Tabla 2.6_1 Especificaciones de Progress RDBMS

En la figura 2.6_2 se muestran dos usuarios remotos que tienen acceso a la base de datos de Progress mediante dos DataServers (servidores de datos), usando uno el 4GL de Progress y el otro Progress SQL-92 Server compartiendo ambos el Shared Storage Manager (Manejador de almacenaje compartido) para una aplicación de Enterprise (empresarial).

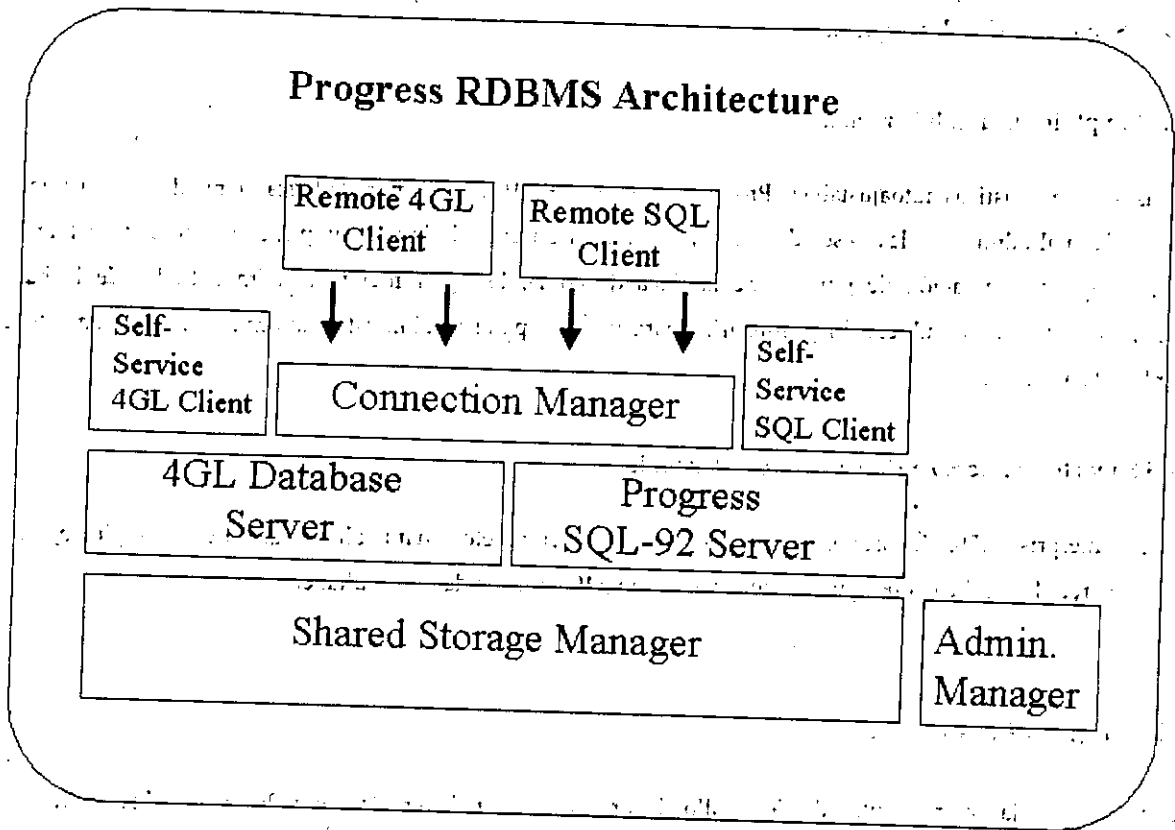


Figura 2.6_2 Arquitectura de Progress RDBMS

Progress puede utilizarse con diferentes Arquitecturas, Base de Datos, Sistemas Operativos y Redes como se muestra en la tabla 2.6_2.

Característica	Estándares y Plataformas Soportadas
User interface	Windows, ASCII Terminal, Windows 9x-NT, MS-DOS
DataBase	Progress, DB2/400, Oracle, MS SQL-Server, ODBC, RMS, C-ISAM, CT-ISAM
Network	TCP/IP, LU 6.2, NetBios, SPX/IPX, SNA, DECnet, PathWorks, Lan Faspath
Operating System	UNIX, Windows 9x-NT, NetWare NLM, OS/400, MS-Dos, VMS, CTOS, OS/2
CPU	Intel, RISC, Alpha, A/S400, PowerPC, Motorola, Vax

Tabla 2.6_2 Productos y Plataformas Compatibles con Progress

La arquitectura del DataServer de Progress proporciona una fácil ejecución a las bases de datos de terceros y cuenta con portabilidad y transparencia en el manejo de datos, los cuales pueden ser utilizados por el 4GL de Progress y el **Data Dictionary** (Diccionario de Datos). Como se muestra en la figura 2.6_3, una aplicación de Progress puede usar diferentes manejadores de Base de Datos.

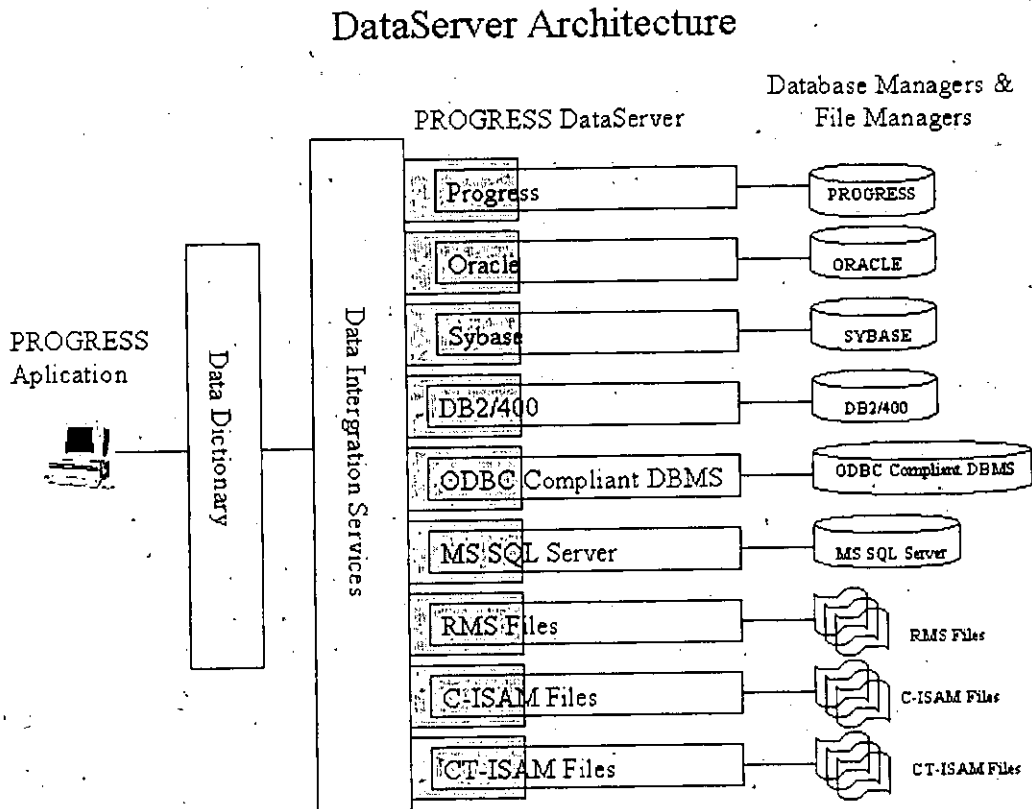


Figura 2.6_3 Arquitectura del Servidor de Datos de Progress

2.6.4 Componentes de Progress 4GL / RDBMS

Los componentes básicos del sistema manejador de bases de datos de Progress son mostrados en la figura 2.6_4.

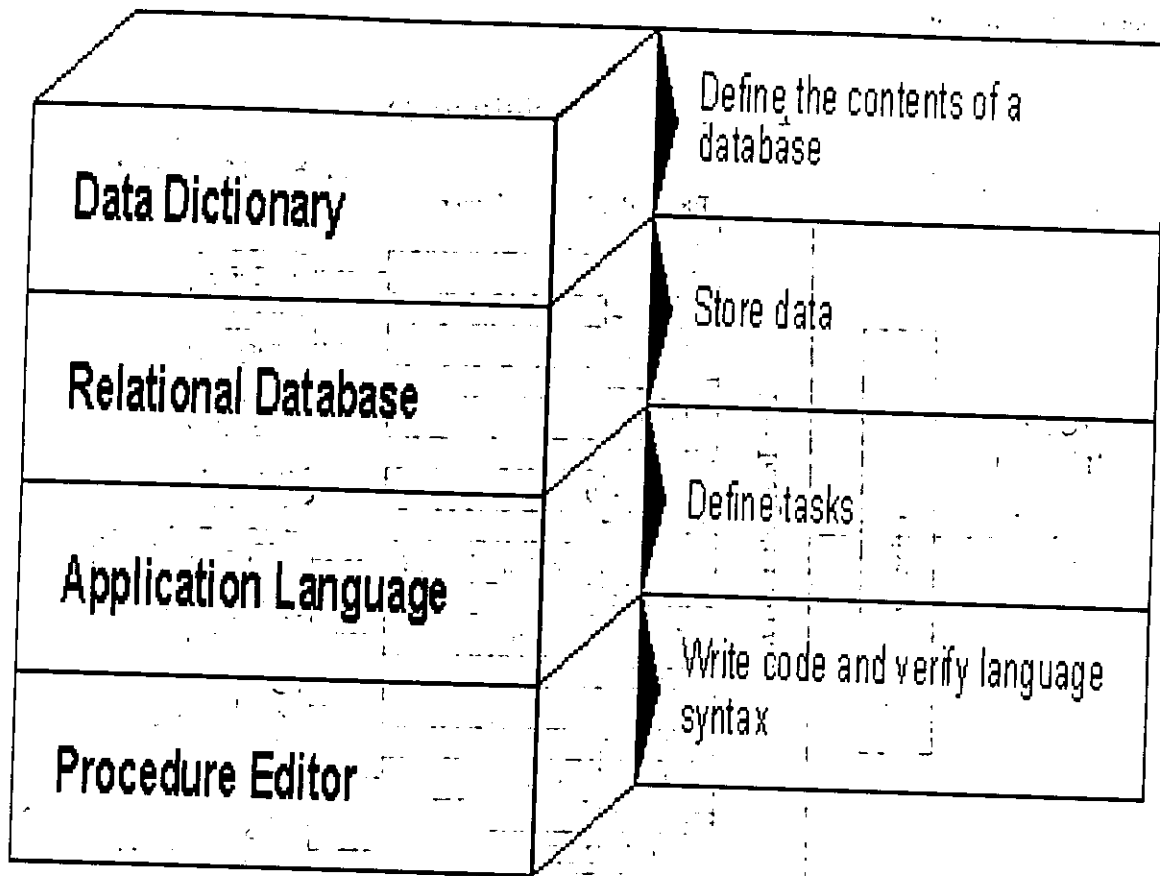


Figura 2.6_4 Componentes de Progress RDBMS

2.6.4.1 Progress 4GL

Es un lenguaje de alto nivel de desarrollo para crear aplicaciones críticas y las transacciones necesarias.

- Reduce la complejidad por ser un lenguaje sencillo para desarrollos en aplicaciones, incluyendo interfaces de usuarios, lógica de transacciones y datos de entrada y salida (input / output).
- El diseñador tendrá una productividad por la reducción de curva de aprendizaje y dará mantenimiento a través de una fácil secuencia de métodos. Con sintaxis en inglés.
- La ayuda provee gran flexibilidad a través de un código portátil.
- Soporta el desarrollo gráfico y de interfaces de caracteres, en tanto el mantenimiento es sencillo y el código fuente es fácil de administrar.
- Reduce esfuerzos de programación por incluir transacciones automáticas de control y bloqueo de registros.
- Incluye un gran poder en los procesos en batch para servidores basados en aplicaciones de despliegue.
- Soporta programación estructurada, orientada a objetos y "event-driven" (accionada por eventos).
- Abierto a DDE, DLL y subrutinas en "C".
- Soporta caracteres internacionales.

En la figura 2.6_5 se tiene la clasificación de Progress 4GL en relación con otros lenguajes de desarrollo y programación.

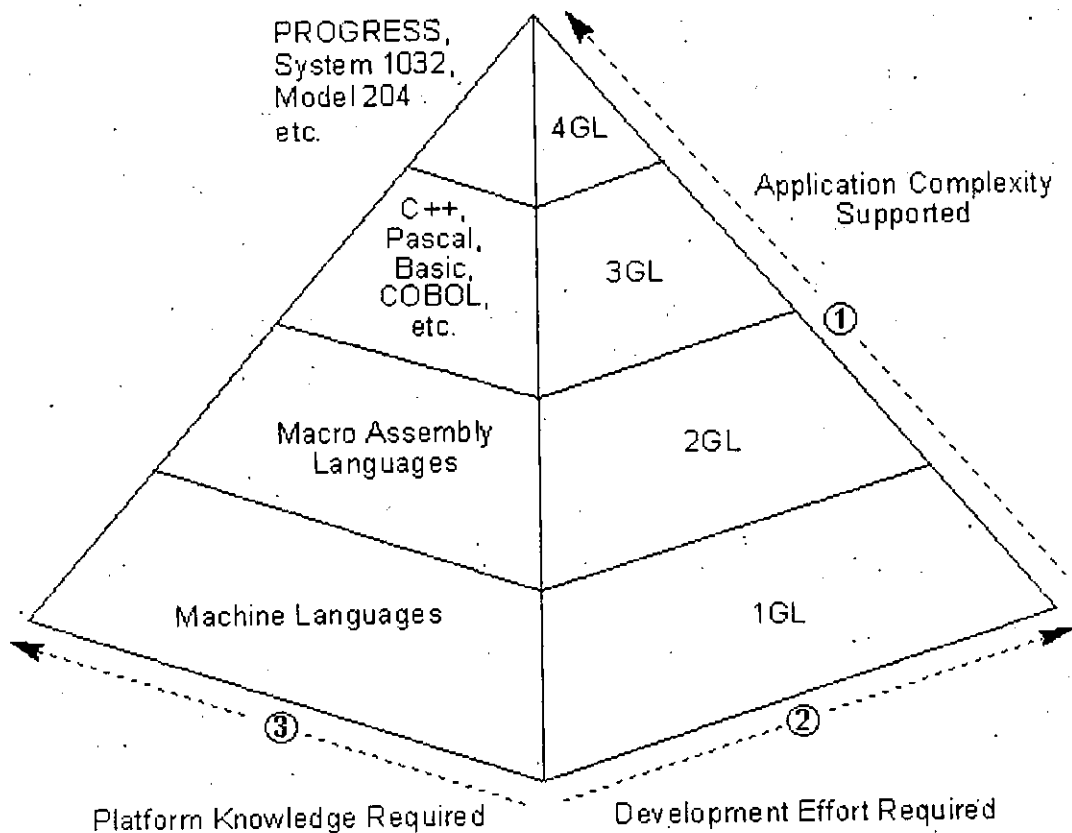


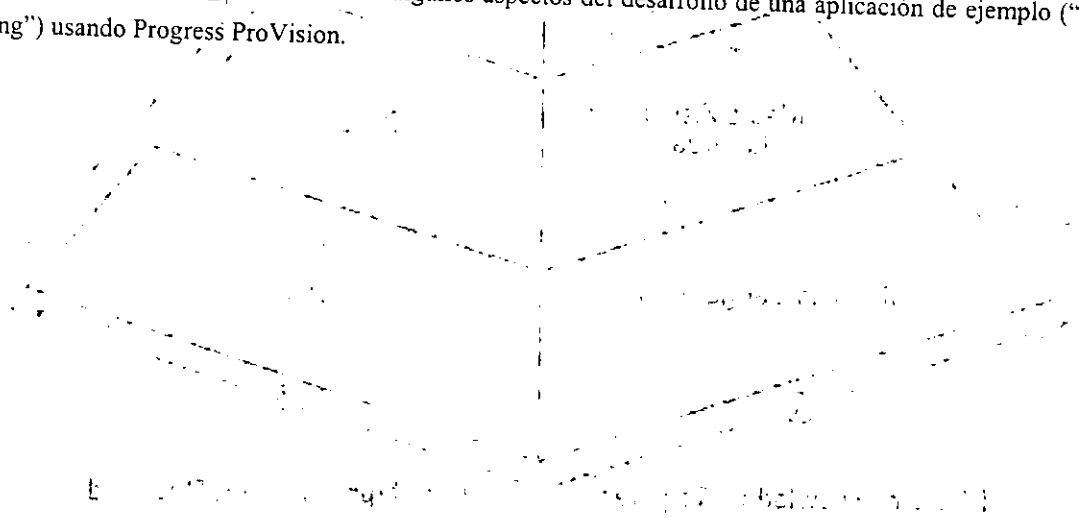
Figura 2.6_5 Progress 4GL y otros Lenguajes de Programación

2.6.4.2 Progress ProVision

Es el núcleo para desarrollar una aplicación visualmente y permita construir, probar y mantener las transacciones orientadas a objetos para soluciones de aplicaciones **Client/Server** (cliente / servidor) y **Host-based** (basadas en anfitriones centrales). Entre sus características se tienen las siguientes.

- Report Builder de 32 bits.
- Data Server Oracle 8.
- Enterprise escalable hasta 4,000 procesos concurrentes.
- Usos del ProUtil.
- Cuenta con el "User Interface Builder" para crear aplicaciones gráficas y de carácter.
- Soporta la automatización OLE para aplicaciones.
- Soporta los controles ActiveX para mayor flexibilidad con las interfaces.
- Integra "SmartObjects" con encapsulamiento y orientación a objetos.
- Incrementa la productividad con un conjunto sencillo de código fuente para desarrollo en modo gráfico y carácter.
- Ofrece la integración de componentes de terceros.
- Reduce la complejidad por incluir Progress 4GL hacia todas direcciones en las necesidades de desarrollo con un lenguaje sencillo.
- Fácil desarrollo y cambios con el editor de procedimientos para cortar, pegar, copiar, buscar, etc.
- Reduce el tiempo de desarrollo e incrementa consistencias con el diccionario de datos.
- Incluye el "Progress Debugger" para localizar errores en lógica.
- Cuenta con ayuda en línea para construir contextos de ayuda.
- Soporta la arquitectura del DataServer para un desarrollo flexible.
- Permite un intercambio de datos con aplicaciones no-Progress usando datos fuente de herramientas de administración de bases de datos.

En las figuras 2.6_6 a 2.6_10 se muestran algunos aspectos del desarrollo de una aplicación de ejemplo ("Customer Order Tracking") usando Progress ProVision.



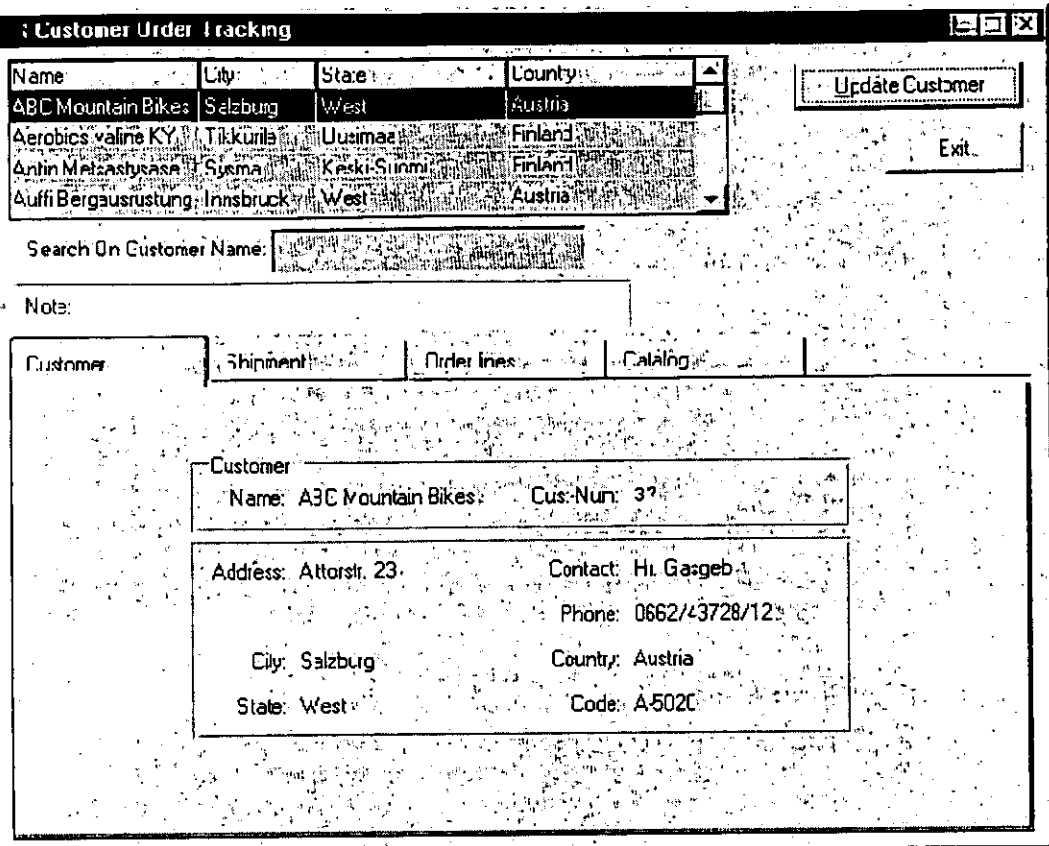


Figura 2.6_6 Desarrollo con Progress ProVision

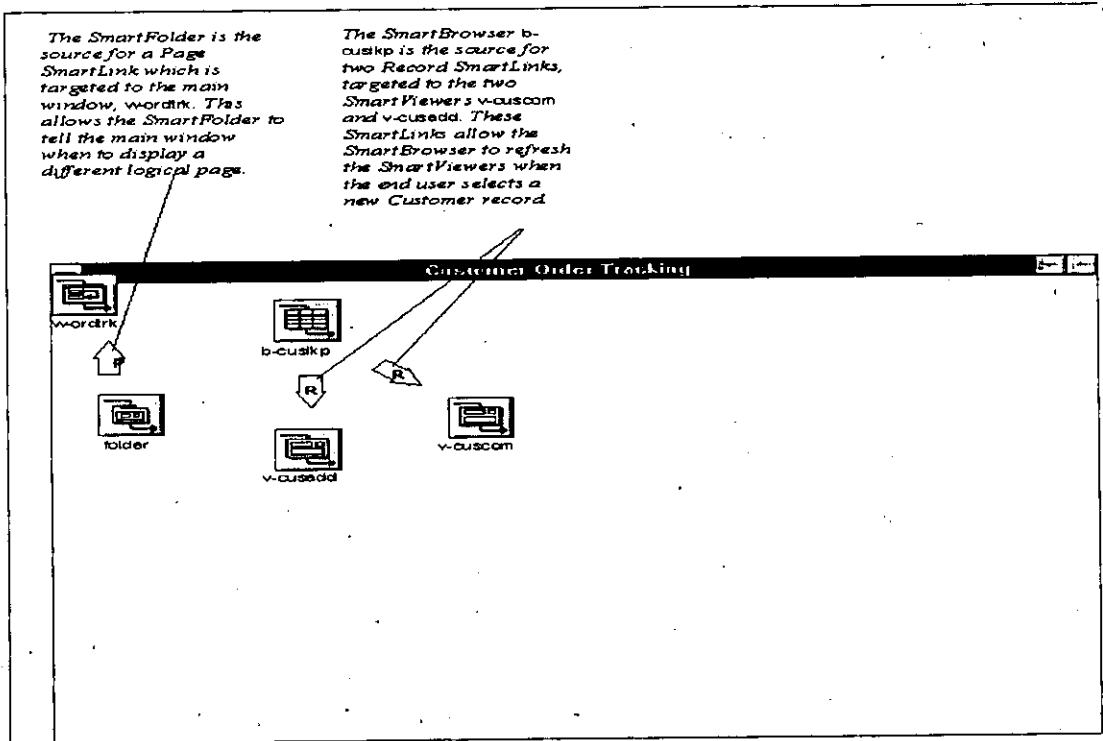
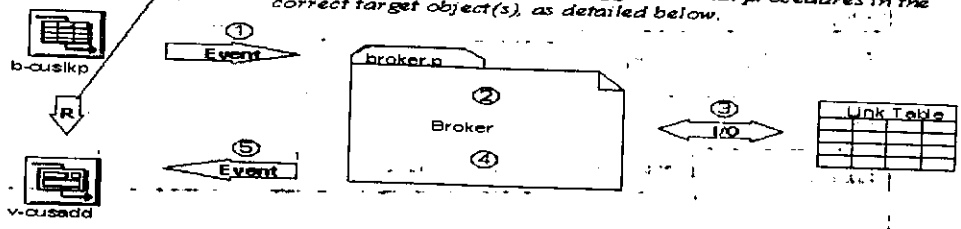


Figura 2.6_7 Desarrollo con Progress ProVision (Cont.)

For practical purposes, you can think of SMARTOBJECTS as communicating directly with each other by means of SmartLinks, as depicted here and in Figures 2-2 and 2-4.

Behind the scenes, a procedure called the Broker, which is stored in a file named broker.p, monitors all running instances of SMARTOBJECTS, detects user-generated events, compares event information to SmartLink information stored in the Link Table, and generates event signals trigger internal procedures in the correct target object(s), as detailed below.



- ① An end-user action (in this case, the selection of a new Customer record) generates an event.
- ② The Broker detects the event.
- ③ The Broker checks the Link Table to see if the SMARTOBJECT that generated the event serves as a SmartLink source of any type is relevant to the event. In this case, the SmartBrowser serves as a Record-source for the Smart Viewer, and the generated event is relevant to that SmartLink.
- ④ In response to the existence of a relevant SmartLink, the Broker generates an event that triggers internal procedures in the Smart Viewer, causing it to update its display fields with information from the newly selected Customer record.
- ⑤ The Broker-generated event is detected by the SmartViewer, which responds as directed.

Figura 2.6_8 Desarrollo con Progress ProVision (Cont.)

The screenshot shows a 'Customer Order Tracking' application window. On the left, there are five icons with corresponding text descriptions:

- worctrlk**: The main window worctrlk is a SmartWindow, which is a type of SmartContainer.
- b-cuslsp**: The SmartBrowser b-cuslsp provides a convenient way for the end user to locate a record in the Customer table.
- v-cuscom**: The SmartViewer v-cuscom displays the contents of the Comments field in the Customer record supplied by b-cuslsp.
- folder**: The SmartFolder, a direct instance of the SMARTOBJECT folder.v, manages the viewing and hiding of other SMARTOBJECTS that are organized into groups called logical pages.
- v-cusadd**: The SmartViewer v-cusadd displays fields from the Customer record supplied by b-cuslsp, which constitute the selected customer's address and phone information.

Annotations on the right side of the window:

- Points to the 'Update Customer and Exit' buttons: *The Update Customer and Exit buttons are not SMARTOBJECTS*
- Points to the 'Customer' data entry fields: *Note.*

The application window contains a table with the following data:

Name	City	State	Country
ABC Mountain Bikes	Salzburg	West	Austria
Aerobics yachts KY	Tikunilo	Uusimaa	Finland
Antin Mekaanisuus	Syama	Keski-Suomi	Finland
Aufli Bergbau-Justing	Innsbruck	West	Austria

Below the table is a search field: 'Search On Customer Name:'. Below that is a 'Note' section with a 'Customer' table:

Customer	Shipment	Order lines	Alarm
Name: ABC Mountain Bikes			
Cust-Num: 37			
Address: Altristr. 23			Contact: Hr. Jasegeb
City: Salzburg			Phone: 0662/43728/12
State: West			Country: Austria
			Code: A-5020

Figura 2.6_9 Desarrollo con Progress ProVision (Cont.)

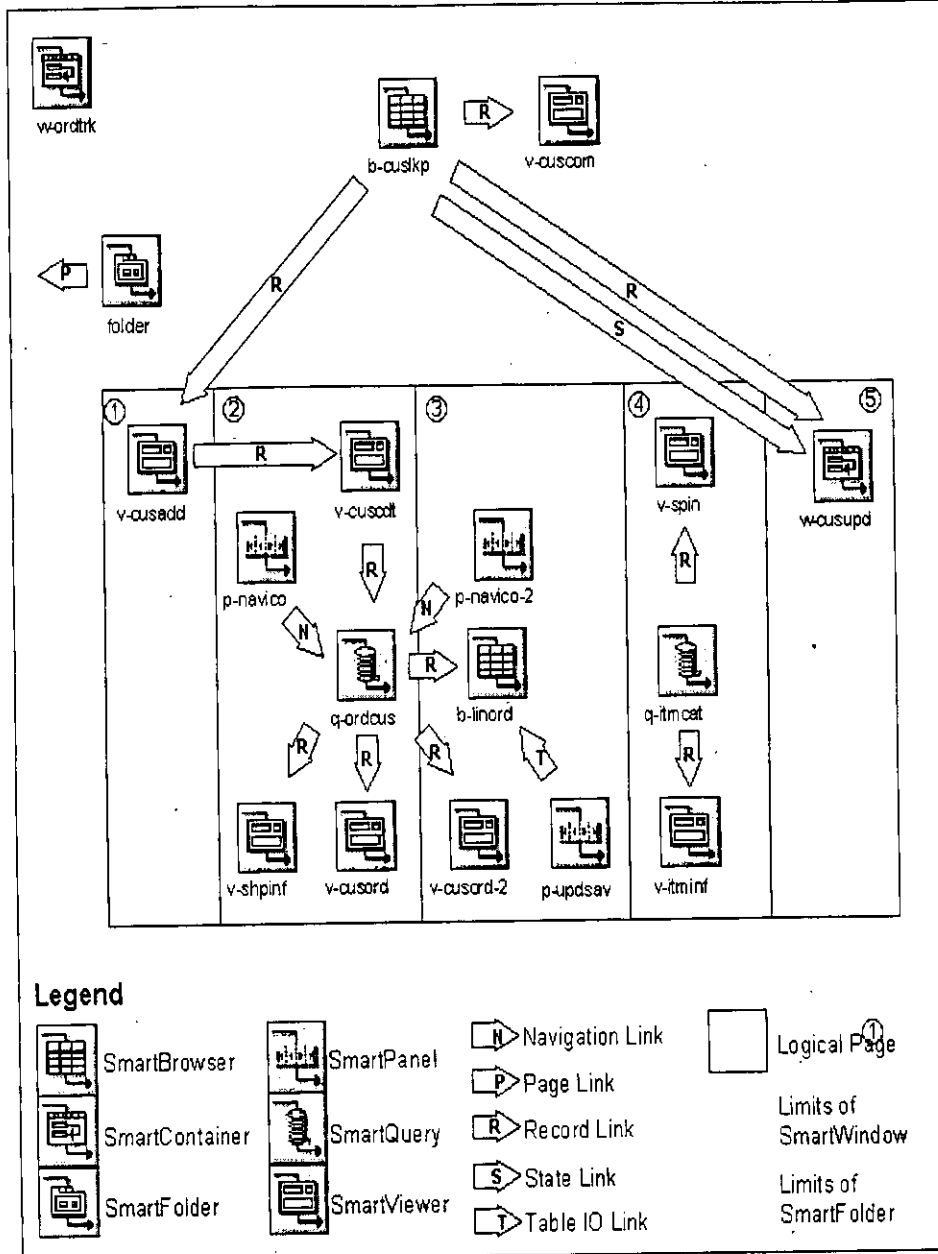


Figura 2.6_10 Desarrollo con Progress ProVision (Cont.)

2.6.4.3 Progress AppServer

Progress facilita la partición de aplicaciones y ofrece el encapsulamiento y distribución lógica "remotely-executable" (con ejecución remota) en un procedimiento de Progress, como se muestra en la figura 2.6_11.

- Remueve la ejecución de un cuello de botella para incrementar la velocidad de la aplicación.
- Provee un poderoso mecanismo de RPC N-tier proporcionando flexibilidad en el despliegado de ayuda y en los cambios a las transacciones que se requieran.
- "Rapid Application Partitioning" (partición rápida de aplicaciones) o RAP, basada en un mismo uso fácil del Progress 4GL.
- Incrementa la productividad del diseñador a través de procedimientos que una vez codificados pueden ser reutilizados en múltiples aplicaciones.

Progress AppServer Architecture

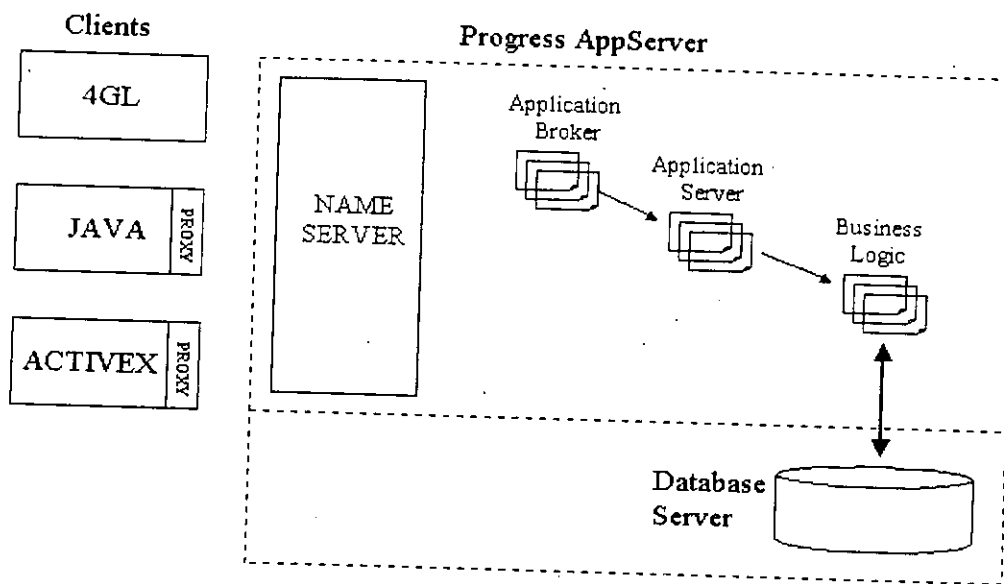


Figura 2.6_11 Arquitectura del Servidor de Aplicaciones de Progress

2.6.4.4 Progress SmartObjects

Esta disponible esta tecnología para desarrolladores que desean construir componentes de aplicaciones empresariales que pueden ser ensamblados dentro de una aplicación simple o compleja de Progress. Los SmartObjects son un conjunto de herramientas automáticas para desarrolladores que requieren de una rápida y fácil creación de código reusable, con alta productividad y en un entorno gráfico. Progress Ver.8 cuenta con User Interface Builder (UIB).

La meta de Progress Software es entregar a los desarrolladores de aplicaciones en un ambiente útil y fácil de usar con componentes basados en gráficos. Para obtener los beneficios de la Orientación a Objetos sin su complejidad, usuarios de Progress, se están actualizando a la Versión 8 de Progress, para poder aprovechar el rápido despliegue y desarrollo de aplicaciones técnicas que usan SmartObjects. En la figura 2.6_12 se muestran las pantallas de las herramientas usadas para construir los SmartObjects.

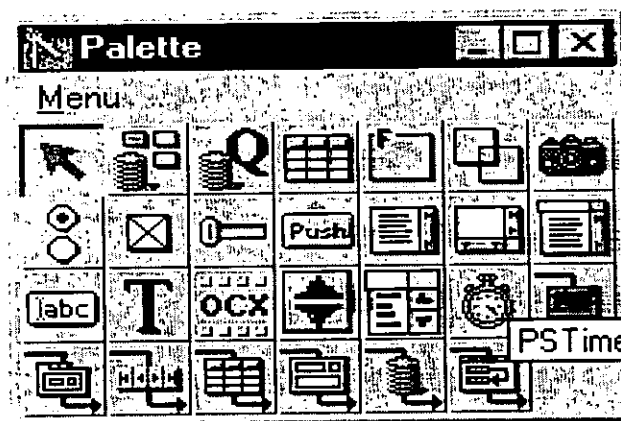
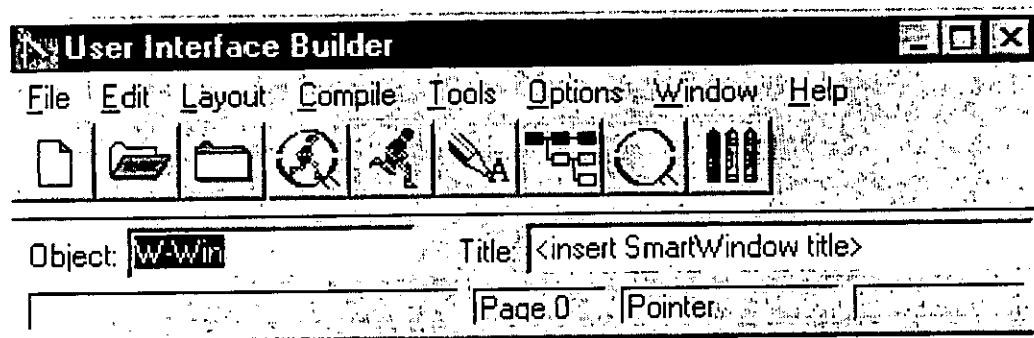


Figura 2.6_12 Progress User Interface Builder

2.6.4.5 Replication Toolkit

La replicación de datos es la duplicidad de información de uno o más lugares. Progress Ver.8.3 provee un "replication Toolkit" (juego de herramientas de replicación) este esta basado en 4GL el cual permite la distribución y la distribución flexible de datos en el nivel de aplicación. Sus características son las siguientes.

- Provee un fácil uso del código 4GL con un extenso soporte de replicación "trigger-based" (activada por disparos).
- Provee redundancia en caso de que el sistema falle.
- El cambio entre una base de datos local y centralizada puede hacerse para recibir información para propósitos de revisión y registro.
- Soporta una replicación síncrona y asíncrona (esto también se conoce como replicación "store and forward").
- Ofrece una alta detección de colisión de datos y resolución de la misma.
- Provee una probada escalabilidad y portabilidad para Windows y una variedad para las principales plataformas de UNIX.

2.6.4.6 Toolkit

Es un conjunto de utilerías, procedimientos y "scripts" para controlar y distribuir aplicaciones, instalaciones y actualizaciones.

- Ayuda a los desarrolladores empaquetando y distribuyendo aplicaciones rápidamente.
- Fácil despliegue para uno o múltiples sitios.
- El despliegue eficaz en función de los costos de aplicaciones modificadas evita la necesidad de comprar licencia de desarrollo para cada sitio.
- Garantiza consistencia de despliegue con el "time-stamp" (registro de tiempo) de versiones.
- Ofrece a desarrolladores un método protegido y seguro para desplegar al usuario final las aplicaciones y sus modificaciones.

2.6.4.7 Client Networking

Es el paquete de software requerido para la conexión de Progress en aplicaciones remotas a la base de datos o DataServers. Posee accesos remotos para clientes en configuraciones distribuidas o Client/Server y también el usuario puede ejecutar reportes creados por el "Report Builder".

La figura 2.6_13 muestra el Software Client Networking de Progress instalado en una PC.

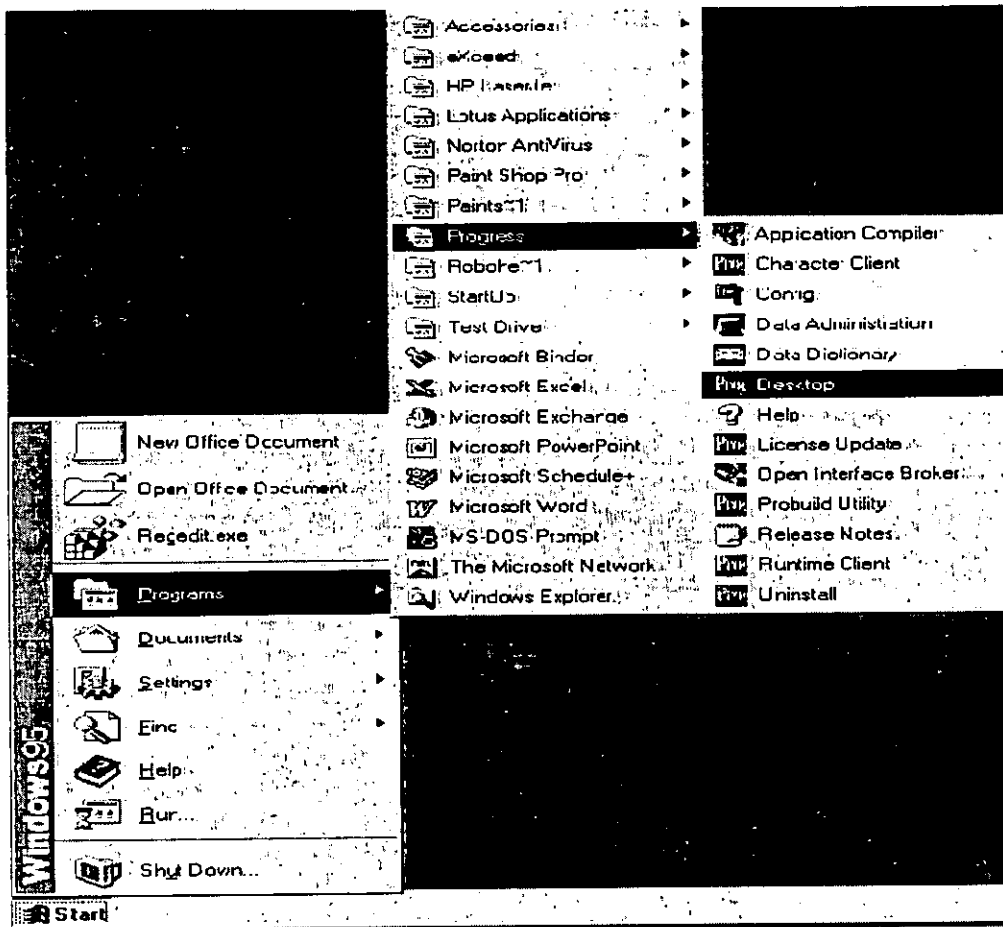


Figura 2.6.13 Progress Client/Networking Instalado en una PC

2.6.4.8 Progress Debugger

Es un monitoreo interactivo, a través de herramientas de examen y análisis para ayuda del profesional, las cuales optimizan la ejecución de los procedimientos de Progress.

- Provee la integración más adecuada con el Progress UIB para fijar los datos y lógica de los errores para entregar calidad en las aplicaciones.
- Ofrece una ejecución completa con capacidad de monitorear y optimizar el código de la aplicación usando "debugging" (depuración):
 - Colocando y quitando "breakpoints" (puntos de interrupción).
 - Mostrando y colocando Buffers.
 - Mostrando y colocando parámetros de aplicación, y variables.
- Permite al programador el uso de la búsqueda "forward" (hacia delante) y "backward" (hacia atrás) en el código de la aplicación que se está examinando.
- Provee una interfase personalizada para el usuario que le proporciona flexibilidad y productividad.

- Soporta un ambiente para examinar múltiples usuarios en una estación de trabajo.
- Soporta el "debugging" de aplicaciones en un ambiente distribuido.
- Permite registrar una sesión de "debug" (depuración) para luego usar y analizar la información registrada.
- Personaliza un proceso de "debugging" con macros y botones de uso definido.
- Soporta un Stack (una pila) de navegación hacia arriba y abajo sin ejecutar múltiples sesiones de "debugging".

Los principales comandos del Debugger de Progress son mostrados en la tabla 2.6_3.

Función	Comandos
STACK Navigation	Down, Run, Up
Breakpoints	Break, Cancel all, Cancel break, Show breaks
Buttons	Button, Cancel all, Cancel button, Show buttons
Debugger Control	Cancel log, Exit, File, Help, Input, Log
Debugging Information	Display file, List buffers, List parameters, List variables, Show breaks, Show buttons, Show macros, Show stack
Macros	Cancel all, Cancel macro, Macro, Show macros
Program Execution	Assign, Break, Continue, Down, Next, Quit, Run, Step, Stop, Up

Tabla 2.6-3 Comandos del Progress Debugger

En la figura 2.6_14 se muestra la pantalla del Debugger de Progress.

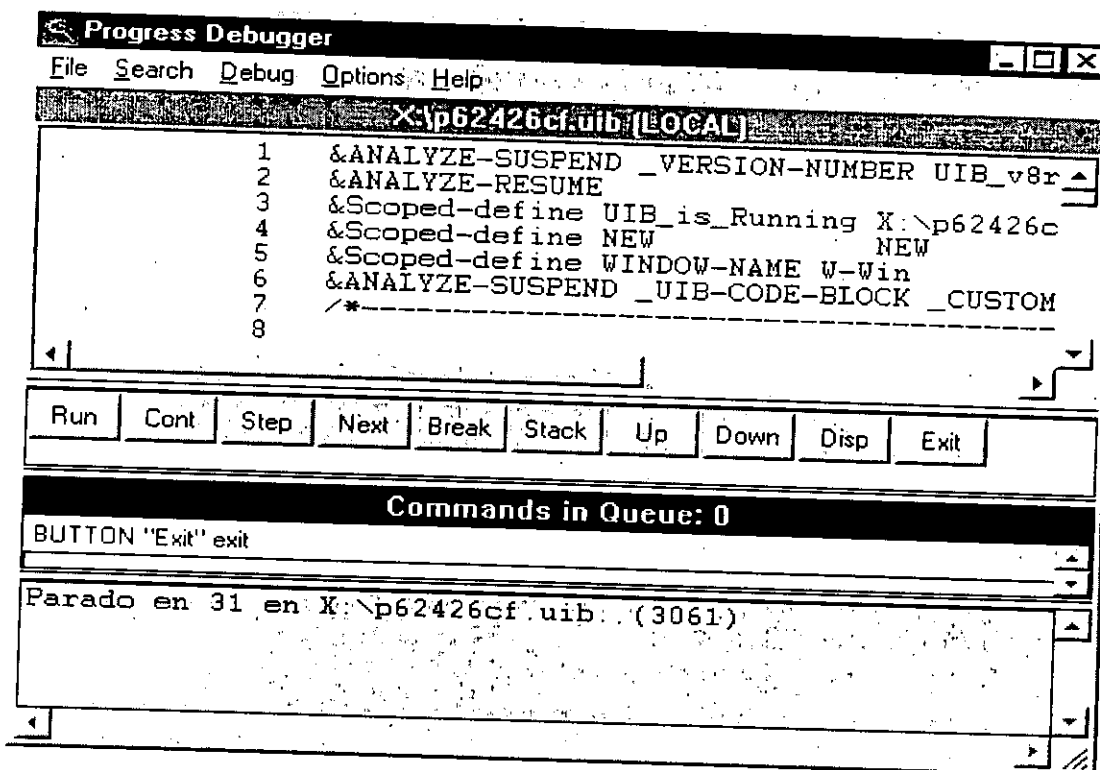


Figura 2.6_14 Debugger de Progress

2.6.4.9 Progress WebSpeed

Progress WebSpeed ofrece a los desarrolladores una solución perfecta y comprobada para construir de manera rápida y segura aplicaciones empresariales – como administración de la cadena de abastecimiento, generación de órdenes, servicio a clientes, control de inventarios y mucho más – e implementarlo ya sea en Internet o en intranets y extranets corporativas.

Progress WebSpeed está formado por Progress WebSpeed Workshop y Progress WebSpeed Transaction Server. Progress WebSpeed Workshop es un conjunto integrado de herramientas de desarrollo para construir de manera rápida aplicaciones ITP, que ofrecen una poderosa conectividad a la base de datos. Progress WebSpeed Transaction Server proporciona una plataforma robusta para aplicaciones ITP que requieren de una alta escalabilidad y velocidad de respuesta rápida.

Progress WebSpeed Workshop es un Ambiente de Desarrollo de Aplicaciones Integrado (IAE) basado en Windows. Incluye un servidor de desarrollo de WebSpeed para compilar y probar aplicaciones. Progress WebSpeed Workshop le permite ejecutar muchas funciones de desarrollo de Internet a través de un “**browser**” (navegador), como administración de archivos y referencia de la base de datos. Se puede utilizar el “**Scripting Lab**”, incluido en el producto, para verificar de manera rápida fragmentos de código.

Progress WebSpeed Workshop está basado en el paradigma del desarrollo por componentes, permitiendo desarrollar y compartir la lógica de procedimientos a través de aplicaciones de Internet y cliente / servidor. Se puede definir y reutilizar Objetos Inteligentes “SmartObjects” en aplicaciones de Internet y cliente / servidor por medio de la herencia de código, encapsulamiento y otras características del desarrollo orientado a objetos.

Algunas de las facilidades de Progress WebSpeed Workshop son:

- “**Wizards**” (asistentes) y plantillas para obtener una mayor productividad en forma instantánea.
- Herramienta de administración de datos para crear las definiciones de la base de datos y las reglas del negocio.
- Progress Personal RDBMS con soporte 4GL y SQL-92, para desarrollo en Internet e integración de herramientas y aplicaciones.
- Diseño y ambiente tipo Windows con manejo local y remoto de archivos.
- Integración con productos RoundTable® SCM para la administración de código fuente y trabajo en grupo.

Progress WebSpeed Transaction Server

Alcanza altos volúmenes transaccionales y tiempos de respuesta rápidos al ejecutar las aplicaciones en un ambiente de procesamiento transaccional óptimo. El poderoso Progress WebSpeed Transaction Server proporciona fácil manejo de altos flujos de información, respuesta transaccional inmediata, balanceo de cargas dinámico y una escalabilidad insuperable para manejar miles de usuarios simultáneos.

El poder deshacer la transacción de bloqueo de registros y el "two-phase commit" permiten que Progress WebSpeed proteja la integridad de la aplicación y de los datos – aún si las transacciones son interrumpidas – permitiendo así garantizar la integridad de las transacciones que tienen acceso a múltiples bases de datos.

Progress WebSpeed Transaction Server permite a los desarrolladores re utilizar la lógica de aplicaciones común para las aplicaciones cliente / servidor y de Internet.

Otras características de Progress WebSpeed Transaction Server son:

- Formar colas de mensajes para mejorar sustancialmente la disponibilidad de los agentes de servicio.
- Manejo eficiente de cargas máximas al iniciar dinámicamente los agentes de servicio de WebSpeed.
- Administración de la ejecución mejorada con las herramientas basadas en Java para configuración por medio de Progress Explorer.
- "Name Server" (servidor de nombres) para una ubicación transparente y balanceo de cargas.
- Capacidades de automatización OLE en el servidor para una mayor interoperabilidad de las aplicaciones.

Características de Progress WebSpeed

- Soporte nativo a las bases de datos, incluyendo acceso a Oracle, DB2/400 y Progress, además de acceso vía ODBC a bases de datos como MS SQL, Sybase, MSAccess, Informix y otras bases de datos.
- Compatible con cualquier ISAPI, NSAPI o Servidor de Internet CGI.
- "SpeedScript" es un lenguaje de escritura de alto nivel para el servidor, que permite desarrollar rápidamente prototipos y desarrollar y mantener la lógica de transacciones de aplicaciones reutilizables. Incluyendo nuevas características de orientación a objetos, super-procedimientos y consultas dinámicas.
- Un modelo de programación flexible le permite estructurar un desarrollo basado en sus requerimientos de proyecto, incluyendo nuevas características específicas como rutinas de validación para las interfaces de usuario HTML para integrar la lógica de aplicaciones a los usuarios HTML.
- "El Scripting Lab" le permite verificar rápidamente fragmentos de código antes de agregarlos a su aplicación de WebSpeed, incrementando la productividad y ayudándole a generar tanto interfaces de usuario con características específicas como rutinas de validación para las interfaces de usuario HTML, utilizando "applets" de JavaScript, VBScript, y JAVA.
- Los "wizards" y las plantillas pueden ser utilizados para generar aplicaciones ITP profesional con múltiples páginas, reduciendo significativamente el tiempo de desarrollo y mantenimiento, además de obtener la habilidad para personalizar su aplicación.

- Una arquitectura abierta que se integra con las soluciones de seguridad que usted elija, incluyendo tecnologías de "firewalls" (barreras de seguridad), validación y codificación.
- Una arquitectura de alto rendimiento y estabilidad que proporciona procesamiento transaccional en un ambiente distribuido de múltiples niveles.
- Una administración del estado de la aplicación ofrece soporte completo y flexible para consultar y actualizar bases de datos utilizando objetos de Internet de tipo "stateless", "state-aware" o "state-reset".
- Cualquier navegador (MS Internet Explorer, AOL Netscape)
 - CGI
 - WebSpeed Messenger
- Application Server
 - WebSpeed Transaction Broker
 - WebSpeed Application Files
 - WebSpeed Agents
- "Data Sources" (datos fuente).

En la figura 2.6_15 se muestran la arquitectura del WebSpeed y las cuatro actividades principales de una transacción a ejecutar.

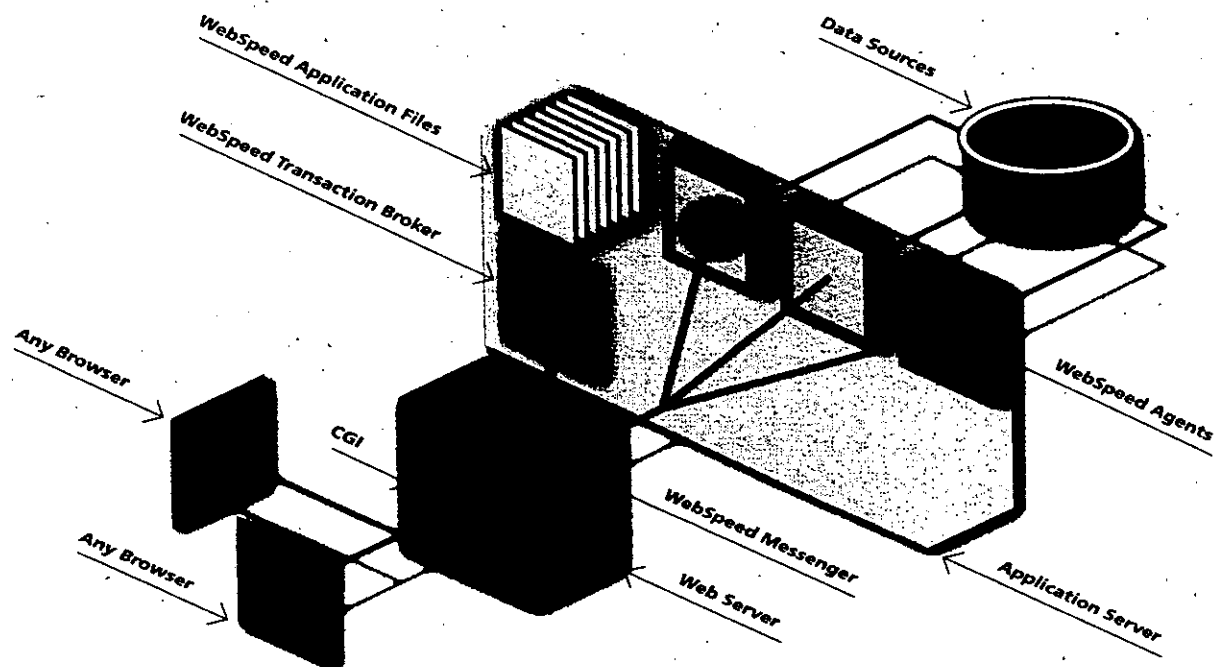


Figura 2.6_15 Arquitectura del WebSpeed de Progress

2.6.5 Requerimientos para el uso de Progress

Los requerimientos para una solución Cliente / Servidor por ejemplo son los siguientes.

Cliente	Requerimientos
PC Pentium, Memoria de 16 Mbytes, 100 Mbytes en Disco Duro.	MS-Windows Ver.9x MS-Windows Ver. NT 4 WorkStation, para Progress Ver. 8.3x
Servidor	
Servidor de red Pentium con Memoria de 32 Mbytes.	MS-Windows Ver. NT 4 Server, para Progress Ver. 8.3x

Tabla 2.6_4 Requerimientos para una Solución Cliente / Servidor de Progress

Plataformas que soporta Progress:

Plataforma	Nivel Mínimo del Sistema Operativo
Windows NT Intel 32-bit Cliente / Servidor	Windows NT Intel 3.51 o Mayor. (WinNT 3.51 requiere Service Pack 5 o Mayor. WinNT 4.0 No requiere Service Pack).
Windows 32-bits cliente	Windows 95 y 98 (Windows 95 requiere Service Pack 1 o Mayor).

Tabla 2.6_5 Plataformas que Soporta Progress

Compatibilidad Cliente / Servidor:

Cliente	Servidor	Soporta
8.3A	7.3x (x=a, b, c...)	TCP/IP, SPX, NETBIOS
8.3A	7.2x (x=a, b, c...)	TCP/IP, SPX, NETBIOS
8.0A	8.3a	TCP/IP
8.0B	8.3a	TCP/IP
8.1A	8.3a	TCP/IP
8.2x (x=A,B,C)	8.3a	TCP/IP
8.3A	8.x (x <=3a)	TCP/IP
WebSpeed 1.0	8.3a	TCP/IP
WebSpeed 2.x (x=0,1)	8.3 ^a	TCP/IP

Tabla 2.6_6 Compatibilidad Cliente / Servidor de Progress

2.6.5.1 Requerimientos Mínimos para ejecutar Progress en Windows NT, Windows 95 y Windows 98

Windows NT / Intel

- Pentium II o Mayor:
- Velocidad de Procesador \geq 350 MHZ
- 32 MB de RAM para Sistema Operativo (64 MB recomendado)
- Suficiente espacio en disco duro (ver tabla 2.6_4)
- Microsoft Windows NT Workstation o Windows NT Server, Versión 3.51 o 4.0
- Service Pack 5 o Mayor para Windows NT Versión 3.51
- Un CD-ROM Drive.

Windows NT / Alpha (Únicamente Server)

- Cualquier sistema Digital AlphaStation, AlphaServer, DEC 2000 AXP o DEC PC capaz de ejecutar Microsoft Windows NT, o cualquier estación de trabajo que use un procesador Digital Alpha y sea capaz de ejecutar Microsoft Windows NT.
- 32 MB de Memoria.
- 65 MB de espacio libre (ver tabla 2.6_4)
- Windows NT Versión 3.51 o 4.0

Nota: La base de datos no puede ser copiada entre Windows NT / Intel y Windows NT / Alpha.

Windows 95 y 98

- Pentium II o Mayor
- Velocidad de Procesador \geq 350 MHZ
- 32 MB de RAM para el sistema operativo (64 MB recomendado)
- Espacio suficiente para la instalación de Progress (ver tabla 2.6_4)
- Microsoft Windows 95 o 98
- Service Pack 1 para Windows 95
- Un CD-ROM Drive.

2.6.5.2 Espacio Aproximado Requerido en Disco Duro

Producto	Total Tamaño Archivo
Character Client	23MB
Client Networking (1)	105MB
DataServer for ORACLE	107MB
DBE Supplement	22MB
Debugger	3.4MB
Enterprise Database	86MB
SQL/C	6.2MB
4GL Development (1)	98MB
International PROMSGS	31MB
ODBC DataServer (1)	79MB
Personal Database (1)	86MB
Progress AppServer	16MB
ProVISION (1)	133MB
Query/RESULTS (1)	90MB
Query/Runtime (1)	94MB
Report Builder (1)	40MB
Toolkit (1)	3.5MB
Translation Manager (1)	19MB
Visual Translator (1)	65MB
Workgroup Database	86MB

Tabla 2.6_4 Requerimientos de Espacio de Progress para Ambiente cliente / Servidor

(1 = No Disponible en Windows NT Alpha 3.51 y 4.0)

2.7 Características del Front-End

Progress es un amplio conjunto de componentes de aplicación basados en un entorno de desarrollo para la construcción, ensamble y distribución de aplicaciones robustas de clase empresarial a través de entornos heterogéneos **Client/Server** (Cliente / Servidor) y **Host-based** (basados en un anfitrión central).

La línea del producto comprende la programación visual de Progress usando "**Application Development Environment**" (ambiente de desarrollo de aplicaciones) o ADE, la arquitectura del Progress "**DataServer**" (servidor de datos), y el sistema de manejo de bases de datos relacionales de Progress (RDBMS).

Progress proporciona herramientas de desarrollo integradas para construir aplicaciones completas de clase empresarial compuestas de componentes altamente interactivos. Los equipos de diseñadores pueden ensamblar fácilmente, compartir y reusar aplicaciones a través de un amplio rango de plataformas, ambientes operativos, configuraciones e interfaces de usuario sin tener que hacer cambios a la lógica de la aplicación. Estas herramientas ofrecen un desarrollo rápido de la aplicación y una mejorada productividad y consistencia en esta, para así disminuir costos de mantenimiento. Las aplicaciones están basadas en Progress 4GL (lenguaje de cuarta generación) que proporciona escalabilidad y flexibilidad a través de la red de la empresa.

Progress tiene una rica variedad de herramientas de implantación que ofrecen un fácil uso de la recuperación de datos y presentación de reportes, así como una traducción altamente intuitiva de productos para la implantación sistemática de aplicaciones en múltiples idiomas.

La Arquitectura de Progress DataServer permite a los diseñadores entregar aplicaciones de bases de datos independientes, portátiles y escalables, que son optimizadas para el acceso de datos desde Progress RDBMS u otros manejadores de datos líderes.

2.7.1 Conquistando el Iceberg Cliente / Servidor

El Software de Progress usa un modelo de iceberg como una forma conceptual de ilustrar las diferencias entre Progress "**Workgroup**" (para grupos de trabajo) y las aplicaciones y herramientas de Progress "**Enterprise**" (de nivel empresarial), tal como se muestra en la figura 2.7_1.

Sobre la "marca de agua" se tiene una interfase de usuario gráfica donde la mayoría de las aplicaciones y herramientas de desarrollo parecen muy similares. Conforme se cava más profundamente hacia los requisitos de las aplicaciones, se observa debajo de la "marca de agua" que las diferencias entre los productos Workgroup y los de nivel empresarial se vuelven mas claras.

Debajo de la marca de agua es donde las aplicaciones cliente / servidor de la empresa requieren modelos de programación múltiple, lógica compleja, manejo de transacciones de alto rendimiento, desarrollo por equipos, localización y procesamiento distribuidos.

El ADE de Progress se dirige a todos los aspectos del iceberg: las interfaces del usuario, lógica y de datos. Además, las aplicaciones basadas en Progress están idealmente ajustadas para enfrentar las complejidades de los sistemas computacionales del iceberg cliente / servidor.

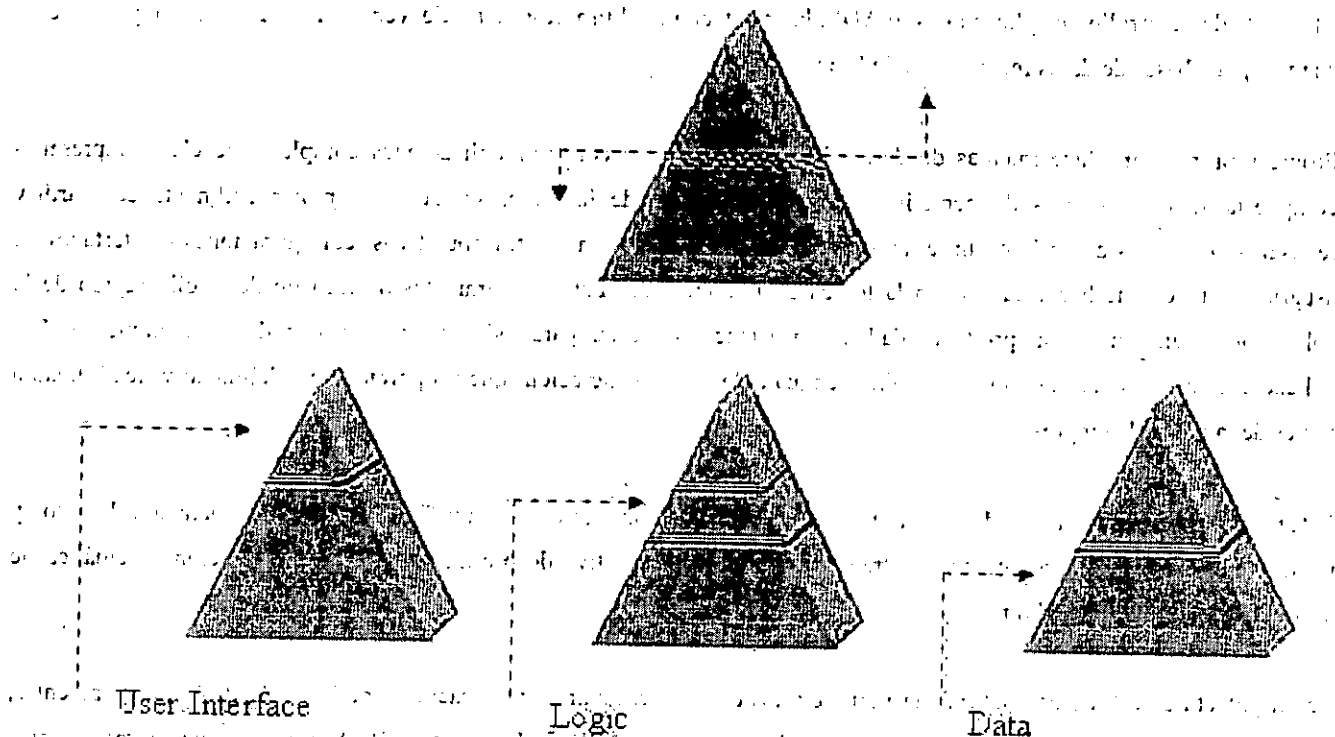


Figura 2.7_1 Modelo del Iceberg Cliente / Servidor

2.7.2 Herramientas de Desarrollo de Progress

Al seleccionar un ambiente de desarrollo de aplicaciones y herramientas, se necesita considerar los requerimientos de desarrollo para el manejo lógico del negocio en plataformas múltiples y sus ambientes de operación.

2.7.2.1 Application Development Environment (ADE)

El Progress ADE es un ambiente de programación orientado a objetos que proporciona a los diseñadores un "mapa visual del camino" para el desarrollo e implantación de aplicaciones complejas de la empresa que sean escalables, portátiles y reconfigurables a través de ambientes heterogéneos cliente / servidor y Host-Based. Dentro del Progress ADE se tiene un conjunto completo de herramientas gráficas de desarrollo que apoyan a un amplio rango de metodologías de desarrollo,

incluyendo: desarrollo estructurado, orientado a procedimientos ("procedural"), manejado por eventos ("event-driven") y orientado a objetos. Las aplicaciones de alto rendimiento pueden ser ensambladas visualmente usando los componentes de aplicación reusables conocidos como Progress SmartObjects.

Progress ADE esta basado en un lenguaje de cuarta generación ("Fourth Generation Language") o 4GL y en un extensivo diccionario de datos que le permite a los diseñadores cubrir los requerimientos de aplicación en misiones críticas. Además, Progress ADE les da a los diseñadores la flexibilidad de controlar totalmente las interfaces, procesos lógicos, y los componentes de manejo de la aplicación requeridos para completar los sistemas de misión crítica. La figura 2.7_2 muestra la pantalla de inicio del Progress ADE y la figura 2.7_3 el modelo de desarrollo de aplicaciones de Progress ADE.

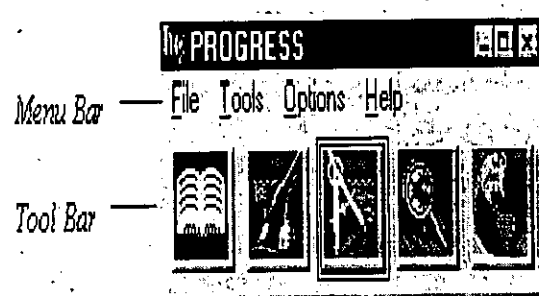


Figura 2.7_2 Pantalla de Inicio de Progress ADE

Application Development Enviroment

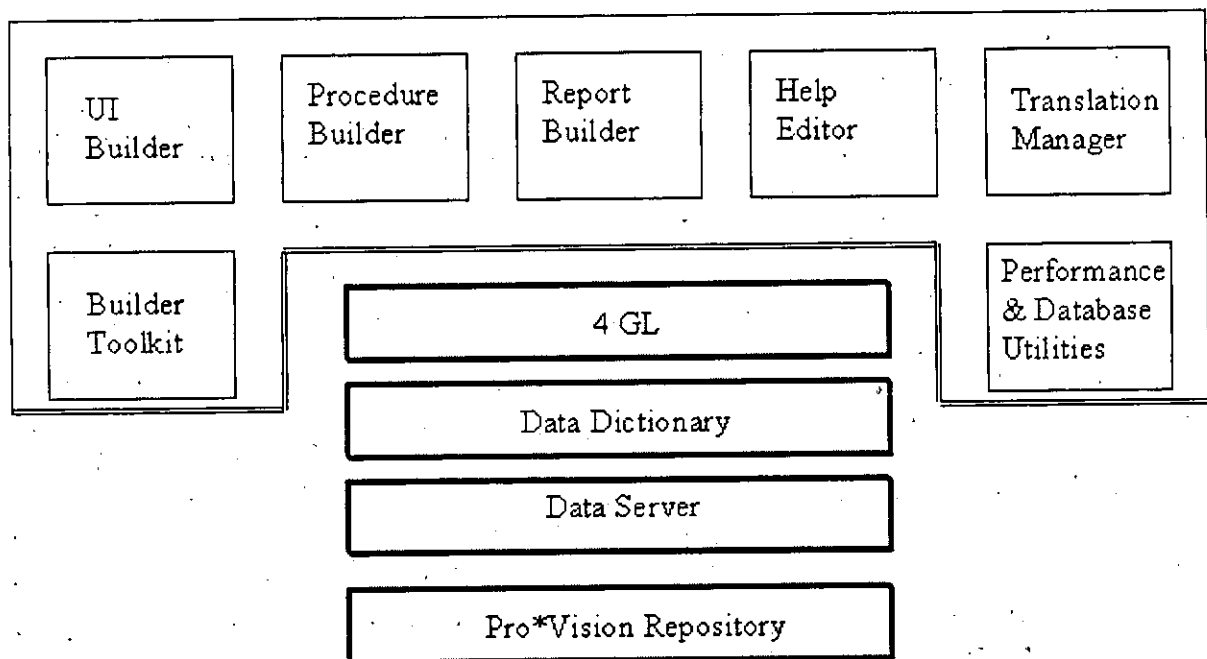


Figura 2.7_3 Modelo de Desarrollo de Aplicaciones ADE

2.7.2.2 Fourth Generation Language (4GL)

Progress 4GL es un lenguaje de alto nivel orientado a objetos para el desarrollo de aplicaciones, que se ejecuta a través de un conjunto de herramientas de Progress ADE. Es un lenguaje de desarrollo robusto y extremadamente eficiente para el desarrollo y la rápida modificación de prototipos de solución.

2.7.2.3 "Data Dictionary" (Diccionario de Datos)

El Progress Data Dictionary es un almacén central para toda la información (sin importar en donde se guarda físicamente) que describe los datos de la aplicación, incluyendo las definiciones de la base de datos, las aplicaciones inmediatas, y reglas del negocio. Cuando se construyen, se actualizan y se distribuyen nuevos componentes de aplicación, las predefiniciones del Data Dictionary y las definiciones inherentes son aplicadas automática y transparentemente.

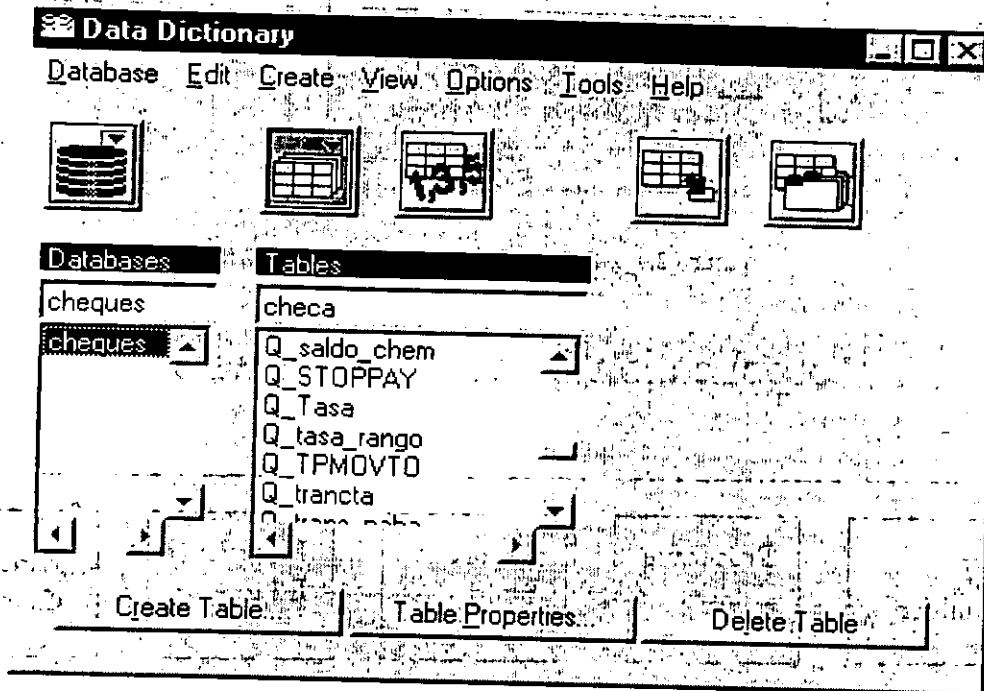


Figura 2.7_4 Progress Data Dictionary

2.7.2.4 User Interface Builder (UIB)

El "User Interface Builder" (constructor de interfaces de usuario) es la herramienta central para crear rápidamente y dar mantenimiento a aplicaciones complejas. El UIB proporciona funcionalidad de "Point-and-Click" (apunta y elige), una paleta de objetos totalmente adaptable, vinculación automática de componentes y mando completo de la interacción de la aplicación con los usuarios finales. Dentro del Progress UIB y con la ayuda de "wizards" (asistentes) de Progress, tarjetas de señal, y diálogos de ayuda, los diseñadores pueden crear y personalizar sus propios SmartObjects como lo muestran las figuras 2.7_5 a 2.7_9.

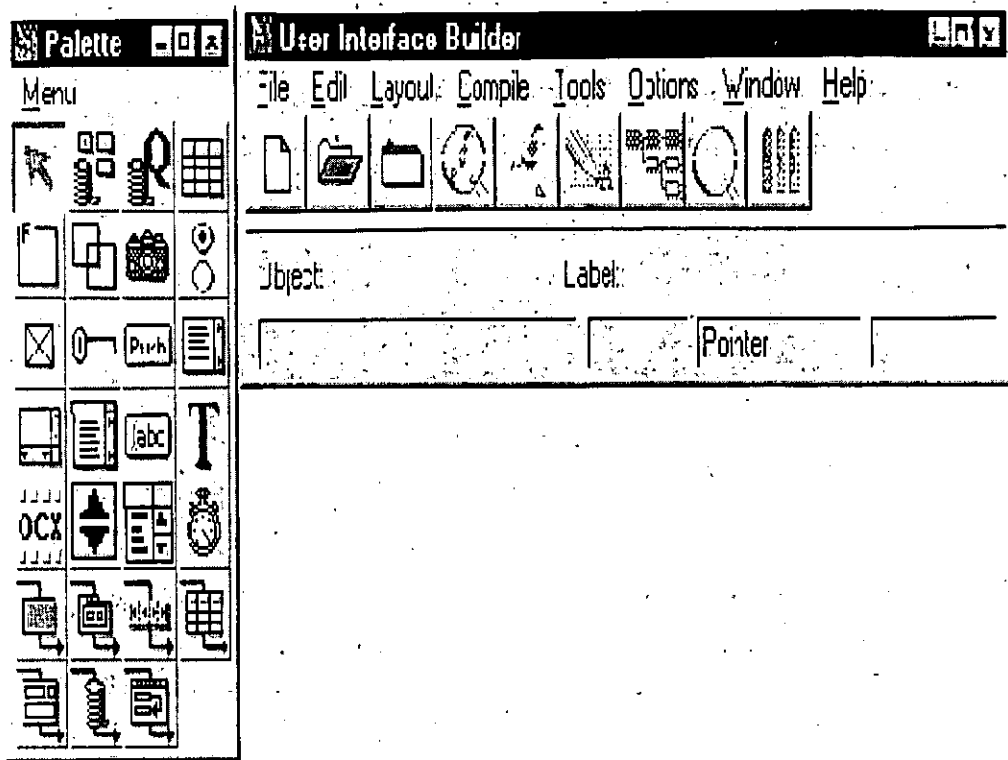


Figura 2.7_5 UIB y la Paleta de Objetos

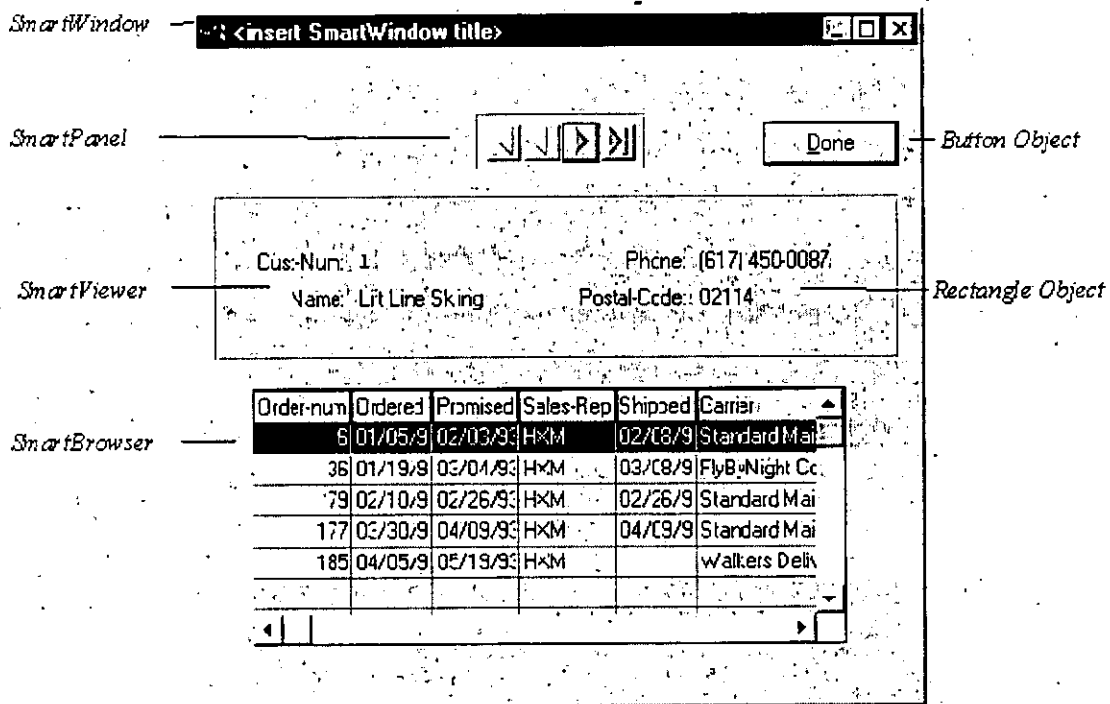


Figura 2.7_6 Apariencia de una Pantalla Ensamblada con SmartObjects y Objetos del UIB

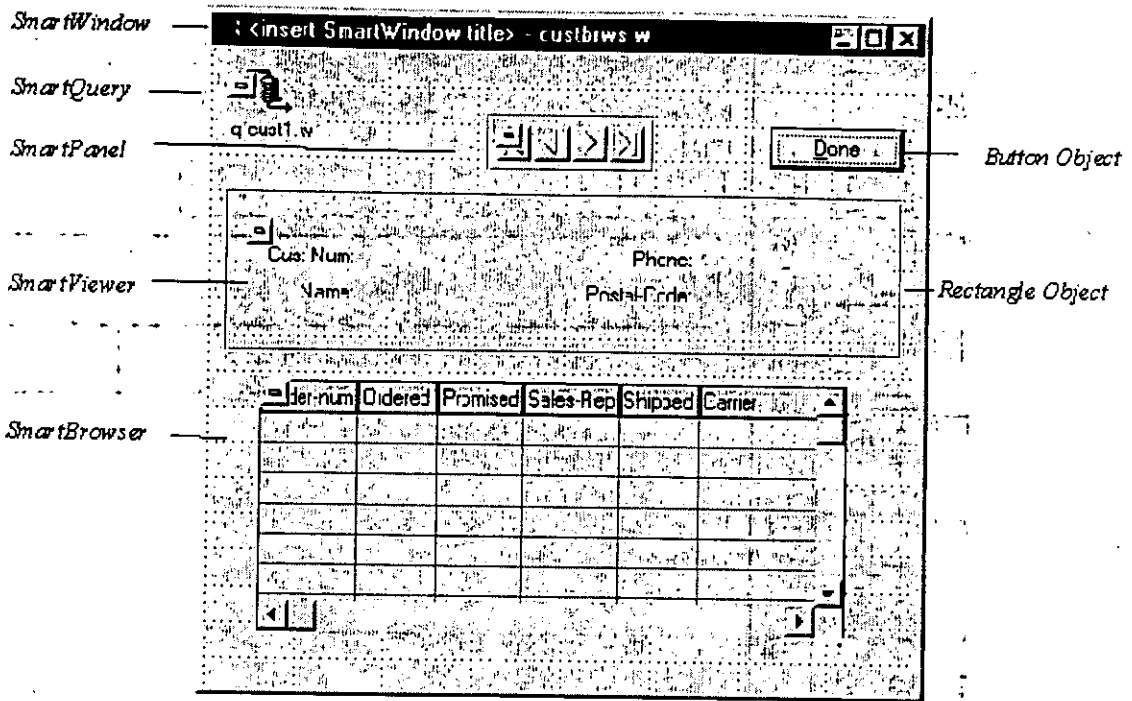


Figura 2.7_7 Ensamble de Componentes de la Pantalla de la Figura 2.7_6

SmartObjects	Botón de la Paleta de Objetos	Descripción
"Browser" (navegador)		Un objeto que despliega los resultados de un "query" (consulta) a una base de datos.
"Button" (botón)		Un objeto del nivel de campo que un usuario selecciona típicamente para ejecutar un "trigger" (disparo). Se tienen disponibles varios tipos de botones en el sub-menú de la paleta de objetos.
"Combo Box"		Un objeto compuesto de un "fill-in" (campo de llenado), una lista de selección y un botón. Cuando el botón es presionado, el valor del "fill-in" puede escogerse de la lista de selección.
"Database Field" (campo de base de datos)		Un objeto que despliega el valor de un campo de un registro de base de datos.
Editor		Un objeto del nivel de campo que permite la edición de campos grandes de caracteres. Por default los editores soportan características tales como cortar, copiar, pegar y "word-wrap" (envolver palabras).
"Fill-in" (campo de llenado)		Un objeto del nivel de campo que acepta la entrada del usuario, la cual típicamente debe ser asignada a un campo de base de datos o a una variable.
"Frame" (marco)		Un área de despliegue dentro de una ventana que Progress usa para desplegar los objetos del nivel de campo.
"Image" (Imagen)		Un objeto del nivel de campo usado para desplegar una imagen de un archivo gráfico.

Figura 2.7_8 Algunos Tipos de SmartObjects

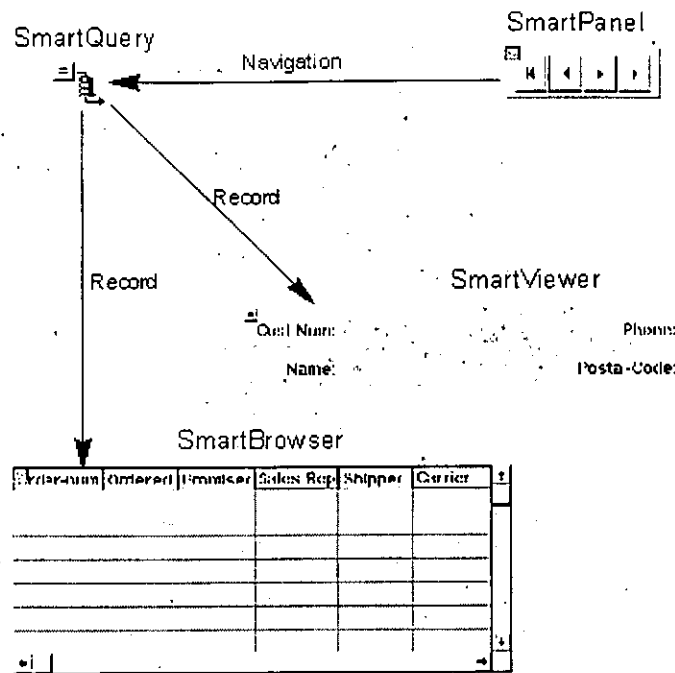


Figura 2.7_9 Enlaces entre los SmartObjects

2.7.2.5 Integración de Herramientas de Desarrollo de Terceros

El ADE de Progress promueve la integración de soluciones de valor agregado de terceros proveedores líderes de herramientas, quienes ofrecen capacidades que complementan, refuerzan, y extienden a Progress como parte de una estrategia de desarrollo de aplicación global. Los productos de estos proveedores de herramientas abarcan rangos de aplicación como el diseño, análisis, modelado, almacenamiento, y prueba de sistemas prototipos, metodología, manejo de procesos, y manejo del proyecto. En las figuras 2.7_10 y 2.7.11 se muestra respectivamente el uso de una Base de Datos de Oracle con Progress y de bases de datos ODBC con Progress.

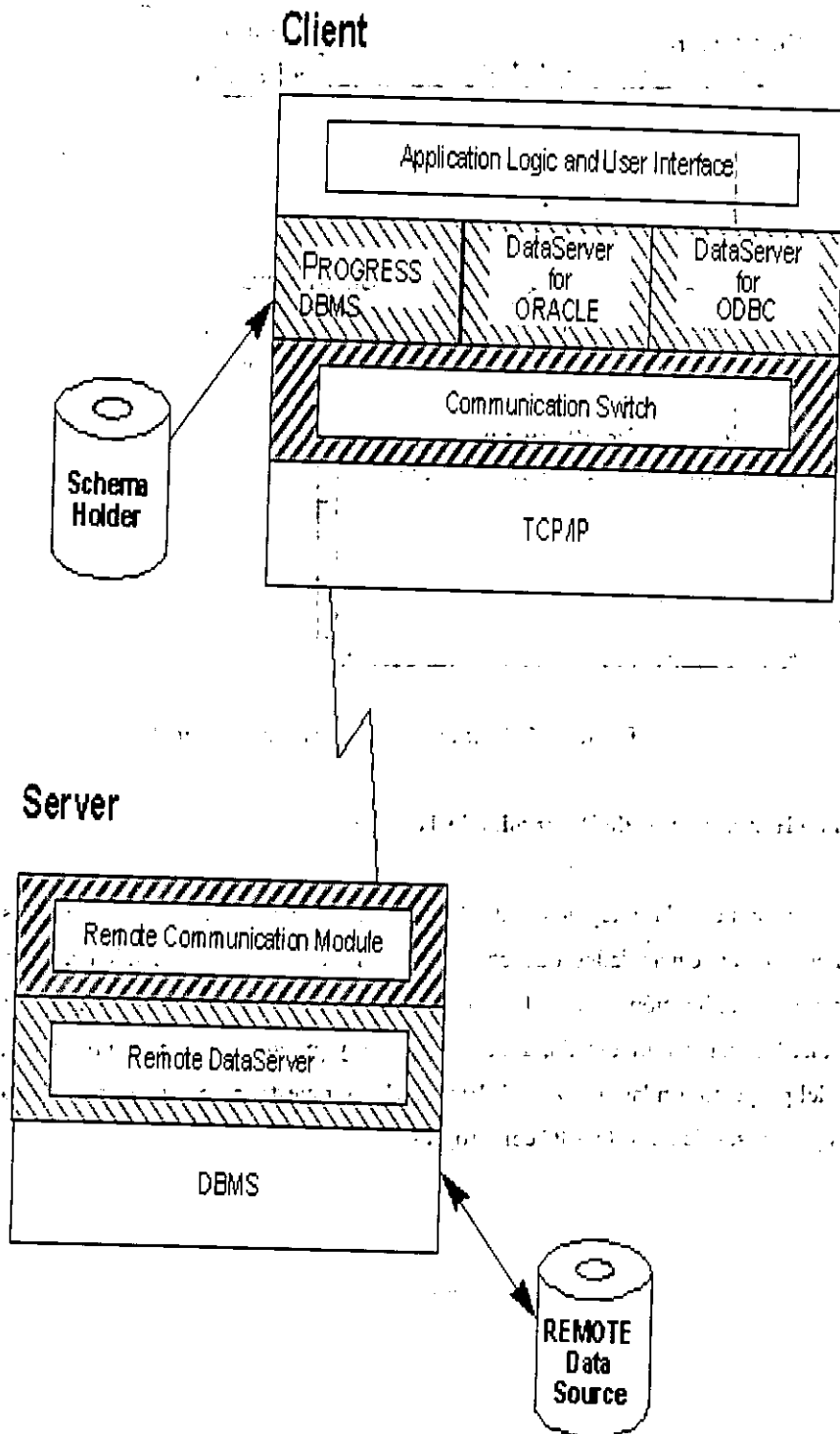


Figura 2.7_10 Progress con Bases de Datos Oracle

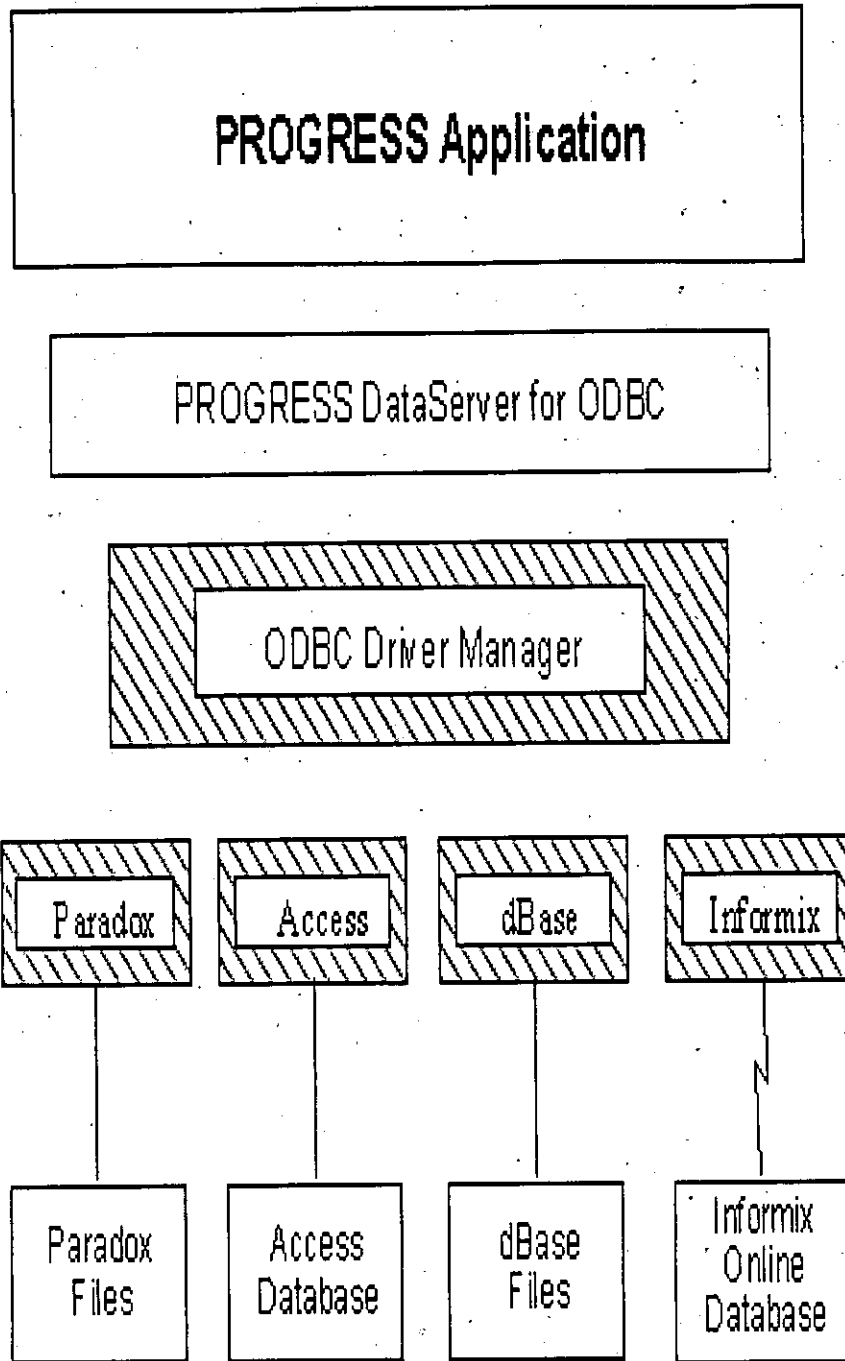


Figura 2.7_11 Progress con Bases de Datos ODBC

2.7.2.6 "On-Line Help" (Ayuda en Línea)

El sistema On-Line Help (únicamente con plataformas de Windows) permite al desarrollador construir su propia Interface de Ayuda usando procesadores de textos populares o sistemas de publicación de escritorio para documentar su sistema, y la ejecuta sin cambios a través de todos los ambientes de soporte de Progress. El sistema On-Line Help también permite al desarrollador incorporar todos los componentes de una referencia en línea, incluyendo vínculos de hipertexto y sus propias notas, facilitando el soporte a las aplicaciones basadas en Progress. En la figura 2.7_12 se muestra la referencia de ayuda de Progress.

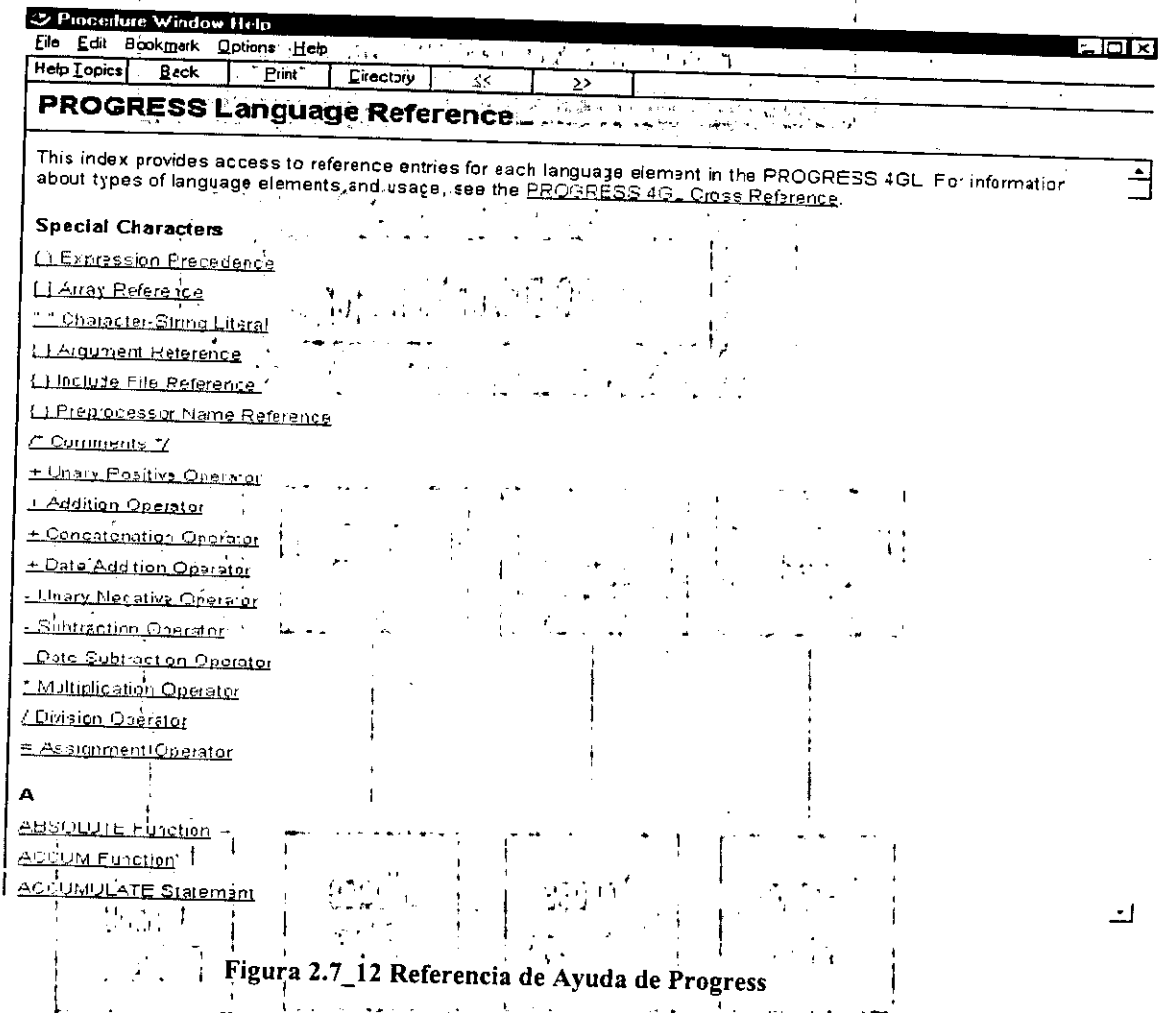


Figura 2.7_12 Referencia de Ayuda de Progress

2.7.2.7 "Procedure Editor" (Editor de Procedimientos)

El editor de procedimientos es una herramienta de edición de código con funciones completas que permite escribir rápidamente, editar, compilar y ejecutar componentes de aplicación de Progress 4GL.

El editor de procedimientos proporciona un rango completo de características para edición incluyendo corte y pegado y búsqueda y reemplazo, lo que le permite hacer rápidamente cambios de largo alcance a diversos programas. Como resultado, el desarrollador puede trabajar concurrentemente en múltiples programas. La figura 2.7_13 muestra la pantalla del editor de procedimientos.

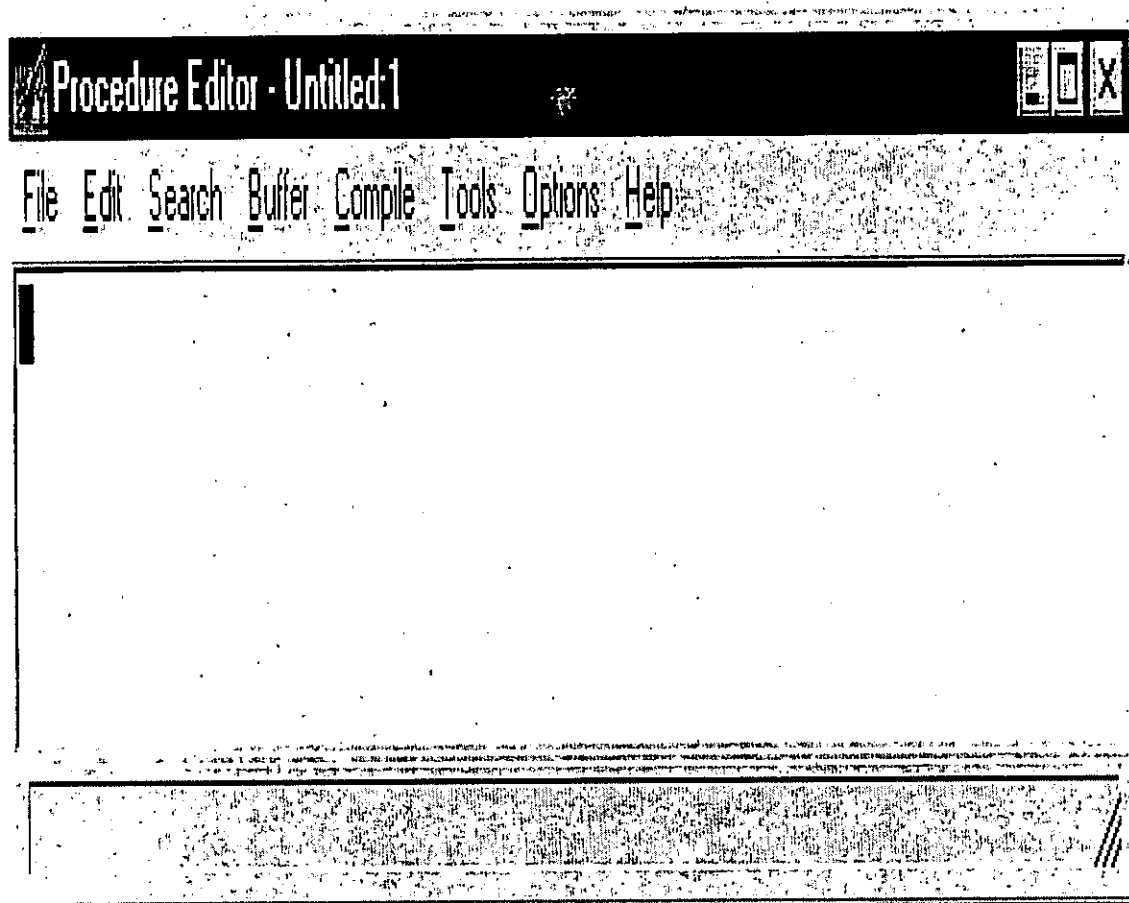


Figura 2.7_13 Editor de Procedimientos de Progress

2.7.2.8 "Application Debugger" (Depurador de Aplicaciones)

El depurador de aplicaciones es una utilidad interactiva que permite controlar y supervisar la ejecución de procedimientos de Progress. Sin tener que modificar el código del procedimiento, el Debugger permite encontrar y corregir errores de datos y de lógica, modificar cualquier código del procedimiento, y mostrar o actualizar información sobre el procedimiento que se está ejecutando. Cuando se invoca al Debugger este corre como un proceso separado en su propia ventana. El Debugger apoya totalmente la supervisión y la depuración de aplicaciones distribuidas construidas con Progress AppServers.

2.7.2.9 "Application Component Environment"

(Ambiente de Componentes de Aplicación) ACE

El ambiente de componente de Progress integra herramientas que permiten a los diseñadores ensamblar visualmente y volver a usar tanto componentes de Progress como componentes externos a Progress.

El ACE es un almacén de desarrollo activo orientado a objetos que define un camino claro para construir y ensamblar una variedad de componentes reusables. El ACE consiste en normas documentadas específicas y la ayuda visual interactiva que permiten la integración del UIB, del 4GL, de Progress SmartObjects, y de componentes de terceros. El marco de trabajo de ACE facilita el ensamble automático de componentes, la vinculación de mensajes entre los componentes, la generación de los procesos de transacción apropiados y el manejo de codificación de datos para completar la aplicación.

2.7.2.10 Progress SmartObjects

Los Progress SmartObjects son una colección de componentes de aplicación comerciales robustos y reusables, que permiten a los diseñadores fabricar y ensamblar rápida y fácilmente desde componentes hasta aplicaciones totalmente funcionales sin escribir ningún código. Trabajando dentro de un ambiente gráfico orientado a objetos, los diseñadores pueden construir aplicaciones de clase empresarial escalables y portátiles usando las plantillas de SmartObjects incluidas en el iniciador de Progress y/o personalizando su propio código reusable. El iniciador de Progress incluye plantillas de SmartViewer, SmartBrowser, SmartQuery, SmartFrame, SmartWindow, SmartPanel, SmartDialog y de SmartFolder que representan la funciones más comunes en ambientes visuales, interactivos, de manejo de datos de aplicaciones cliente / servidor y de transacciones complejas.

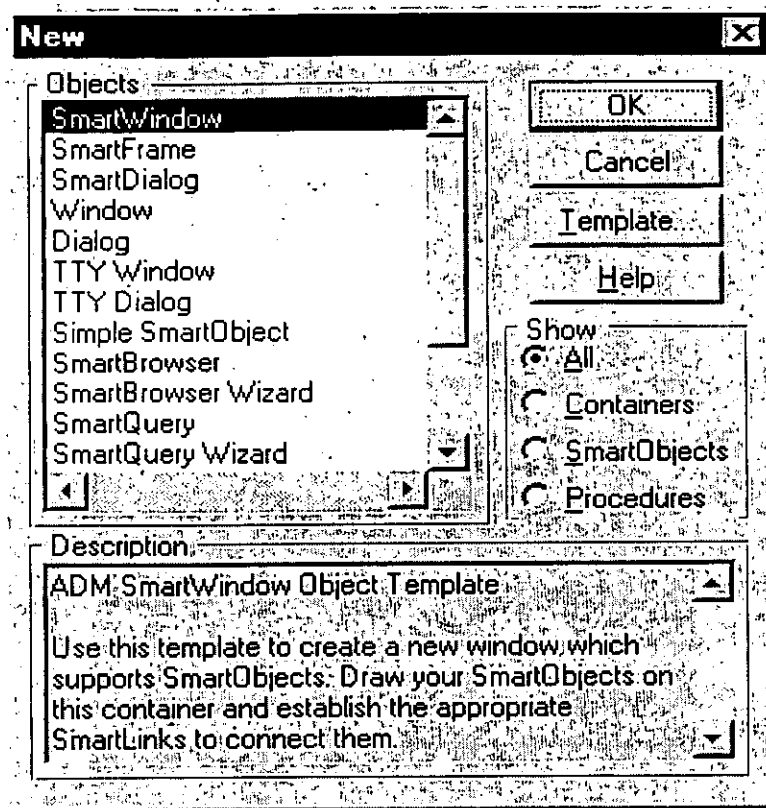


Figura. 2.7.14 SmartObjects Disponibles en el Iniciador de Progress

2.7.2.11 Soporte VBX

Progress UIB apoya el ensamble de aplicaciones Windows de Progress usando las Extensiones Básicas Visuales (VBXs) de terceros. Los diseñadores pueden integrar VBXs rápidamente en una aplicación de Progress desde cualquiera de los iniciadores de Progress o VBXs, usando otros componentes de terceros.

2.7.2.12 Soporte OLE

Progress soporta el ensamble de aplicaciones de Windows que usan el estándar de la industria "Object Linking and Embedding" (vinculación e incrustación de objetos) OLE, tales como Microsoft Excel, Microsoft Word y Novell WordPerfect, y este soporte ha sido incluido en los SmartObjects de Progress. En la figura 2.7_15 se muestra la paleta de objetos de Progress con los ensambles de soporte de VBX y OLE.

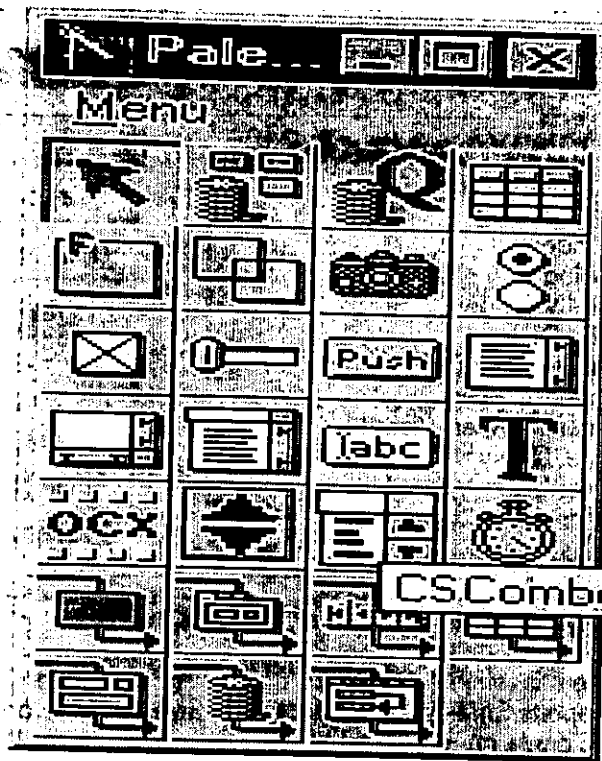


Figura 2.7_15 Paleta de Objetos del UIB Incluyendo Objetos OLE y-VBX

2.7.2.13 Roundtable Total Software Management System

Roundtable es una herramienta de manejo de configuración que promueve el desarrollo por equipos para los proyectos de desarrollo grandes. Roundtable ofrece aplicaciones de almacenamiento automatizado, control de versiones, verificación in/out (entrada / salida), análisis de impacto, esquema de la versión, y seguridad del proyecto. Roundtable está idealmente preparado para reducir los costos de los desarrollos de gran escala asociados con diseño y mantenimiento del GUI o aplicaciones de caracteres en un ambiente de equipo.

2.7.3 Herramientas de Implantación de Progress

Las organizaciones están buscando aplicaciones cliente / servidor de clase empresarial que no sólo respondan a los requerimientos de computación hoy día, sino que buscan el apoyo de los desarrollos del mañana. Progress entrega un armazón poderoso para construir, ensamblar, reusar e instalar componentes de aplicación basados o no en Progress. Con la adición de los Progress AppServers, las aplicaciones pueden ser fácilmente divididas para ejecutarlas a través de múltiples sistemas en red.

El Software de Progress, junto con muchas otras herramientas de desarrollo de terceros ofrece herramientas muy confiables que proporcionan el poder y flexibilidad para soportar aplicaciones de clase empresarial. La figura 2.7_16 muestra como una aplicación empresarial puede ensamblarse con múltiples componentes.

La estrategia de implantación de Progress incluye herramientas del alto desempeño que ofrecen la flexibilidad de implantar aplicaciones de base de datos independientes a través de una amplia gama de ambientes de computación cliente / servidor mezclados con ambientes Host-based. El Tener estas capacidades permite agregar usuarios, mejorar el desempeño, y reaccionar a los adelantos en tecnología con un mínimo de intervención de desarrollo.

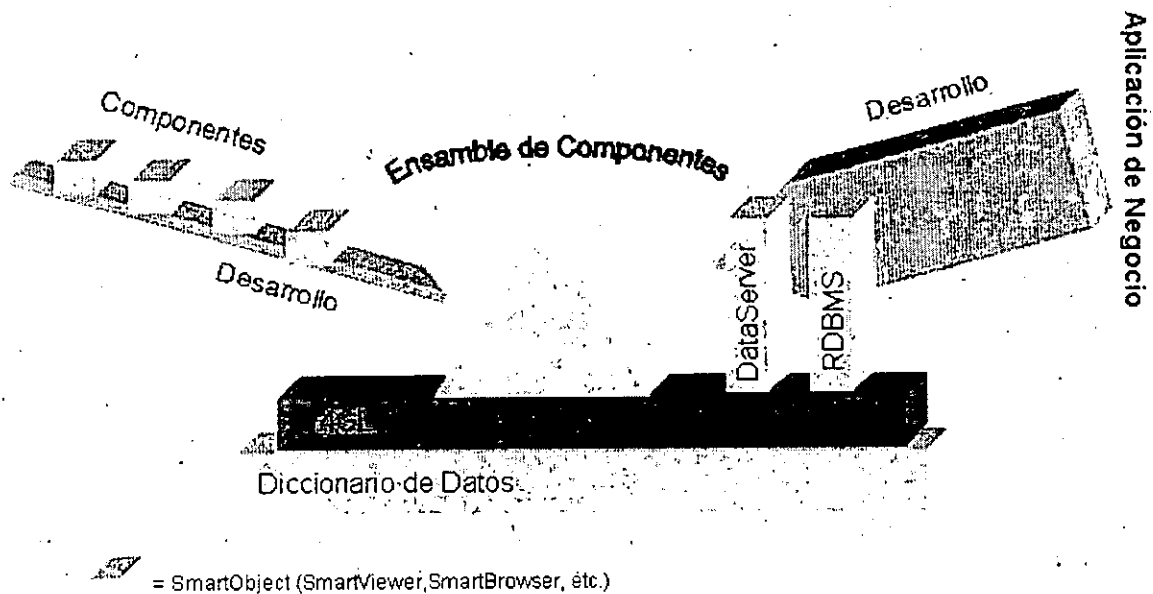


Figura 2.7_16 Ensamble de Aplicaciones en Progress

2.7.3.1 "Progress Results"

Progress Results extiende las características y funcionalidad para que los usuarios finales no técnicos creen rápida y fácilmente sus propios "queries" (consultas) ad-hoc y la generación de reportes personalizados o etiquetas de correos a través de un sistema de navegación de datos Point-and-Click. Pueden exportarse informes en formatos que pueden ser leídos por aplicaciones de hoja de cálculo y procesadores de palabra. Para el diseñador de la aplicación, Progress Results ofrece la flexibilidad para integrar a las aplicaciones de Progress la funcionalidad de los reportes y/o los reportes personalizados. Ejecutándose en cualquiera de los dos ambientes gráfico o de carácter, Progress Results puede tener acceso simultáneamente a múltiples y heterogéneas bases de datos, incluyendo Progress, Oracle, MS SQL-Server, DB2/400, ODBC, y otras. Para los requerimientos de reportes internacionales, Results ésta disponible en muchos lenguajes nativos, incluyendo los lenguajes doble-byte asiáticos, y el conjunto de características internacionales de caracteres, fecha local, y formatos de monedas. La figura 2.7_17 muestra la ventana para construir reportes para la explotación de la base de datos usando Progress Results.

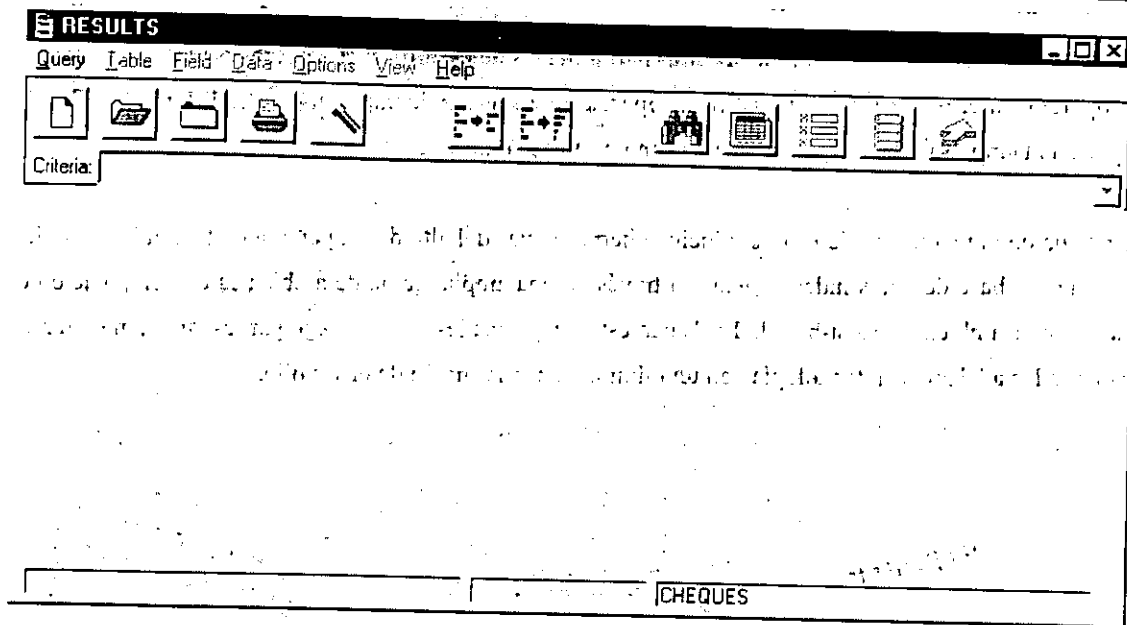


Figura 2.7_17 Progress Results

2.7.3.2 "Report Builder" (Constructor de Reportes)

El Report Builder es una herramienta de escritura de reportes con alta productividad, para los analistas y desarrolladores de aplicaciones. El Report Builder da la flexibilidad para crear formatos de reporte que dan énfasis a la importancia de los datos y mejoran la apariencia visual del informe. Con Report Builder, se puede crear una amplia variedad de reportes de nivel empresarial incluyendo listas, formatos de cartas, etiquetas, estados y declaraciones financieras, catálogos de productos, y muchos otros. Además, los diseñadores de la aplicación pueden desarrollar reportes para sus usuarios incrustando reportes predefinidos en sus aplicaciones basadas en Progress Windows. El Report Builder puede obtener datos de bases de datos múltiples y heterogéneas, incluyendo Progress, Oracle, MS SQL-Server e IBM DB2/400. En la figura 2.7_18 se muestra la ventana para utilizar el Report Builder.

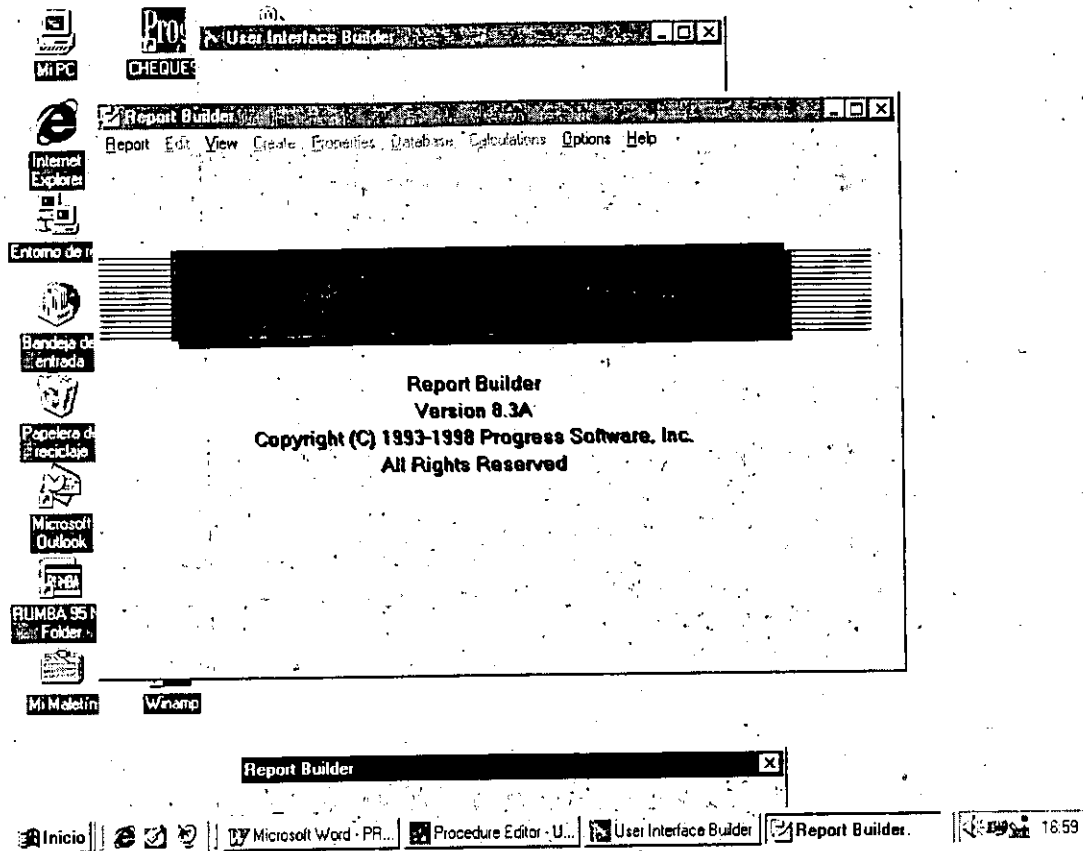


Figura 2.7_18 Constructor de Reportes de Progress

2.7.3.3 "Developer's Toolkit" (Juego de Herramientas del Desarrollador)

El Developer's Toolkit es un juego completo de utilerías, procedimientos y "scripts" (guiones) que los diseñadores pueden usar para empaquetar y distribuir sus aplicaciones basadas en Progress hacia múltiples lugares. El Toolkit les permite controlar muchos aspectos de la distribución e instalación de la aplicación. Ofrece tres formatos de distribución y cada uno proporciona un nivel diferente para el acceso de los usuarios: el código fuente sin encriptamiento para el acceso total del usuario; el código fuente con encriptamiento para seguridad y personalización; y la aplicación del encriptamiento CRC o "r-code" al código fuente para los más altos niveles de seguridad.

Las aplicaciones pueden ser modificadas o actualizadas sin tener que reinstalar una nueva versión de la aplicación. En la figura 2.7_19 se muestra el uso de la herramienta "PRO*tools" del Developer's Toolkit.



Figura 2.7_19 Progress Pro*Tools

2.7.3.4 "Data Administration" (Administración de Datos)

Las facilidades de "Data Administration" permiten realizar una variedad de tareas de mantenimiento a las bases de datos, incluyendo definiciones de carga y descarga de datos, definiciones de intercambio de información con los servidores ajenos a Progress, definir la seguridad de aplicaciones y permisos, y la importación y exportación de datos desde y hacia varias fuentes. La facilidad Data Administration es útil para construir inicialmente una bases de datos de aplicación, así como para el mantenimiento e intercambio de la instalación de base de datos desde el lugar del usuario. La figura 2.7_20 muestra a Progress Data Administration.

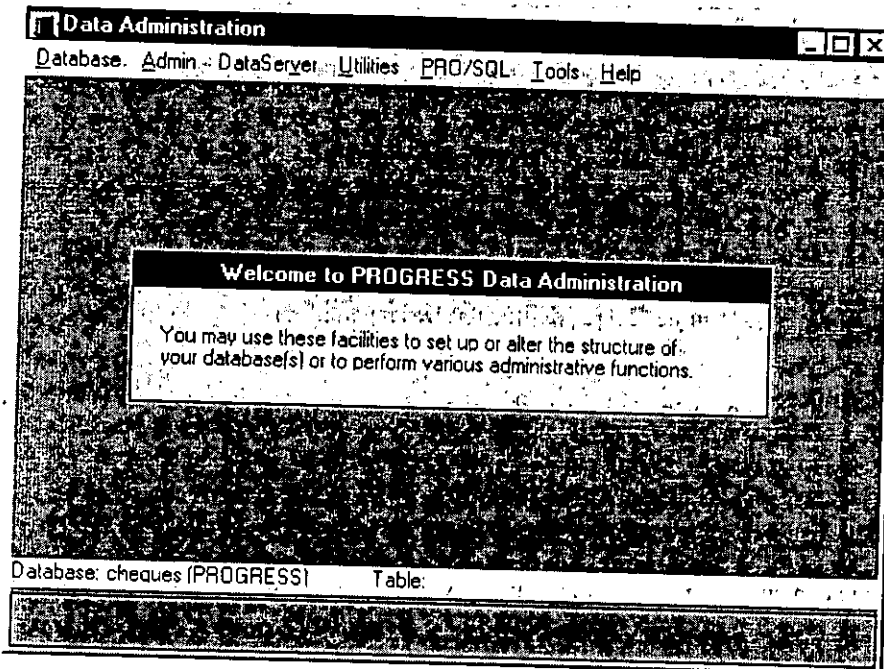


Figura 2.7_20 Progress Data Administration

2.7.3.5 "Translation Manager" (Manejador de Traducciones)

El Sistema manejador de traducciones permite a los diseñadores de una aplicación crear versiones multilingües de su producto, sin tener que volver a revisar copias, o modificar el código fuente original de su aplicación. El Sistema manejador de traducciones consiste en dos componentes: el "Translation Manager" y el "Visual Translator".

El Translation Manager es una herramienta altamente visual que permite a un gerente de proyecto identificar y organizar componentes de aplicación a ser traducidos en múltiples idiomas. El gerente del proyecto detalla controles sobre las frases de texto que son identificadas para la traducción, puede incorporar glosarios de traducción de la compañía y puede dejar activos estos estándares. Incluye adicionalmente un juego de glosarios de Microsoft de términos de software de uso frecuente y las traducciones de Microsoft aceptadas en los idiomas del Oeste y Este europeos. El Translation Manager tiene habilitado el uso de caracteres de doble-byte que permite soportar los idiomas asiáticos.

2.7.3.6 "Visual Translator" (Traductor Visual)

El Visual Translator es una herramienta que proporciona un contexto visual para traducir los componentes de la interfase de usuario (botones, menús, mensajes, etc.) de la aplicación a ser traducida. El traductor puede importar glosarios estándar, puede agregar glosarios predefinidos y puede realizar traducciones globales de todas las frases de texto, como se muestra en la figura 2.7_21.

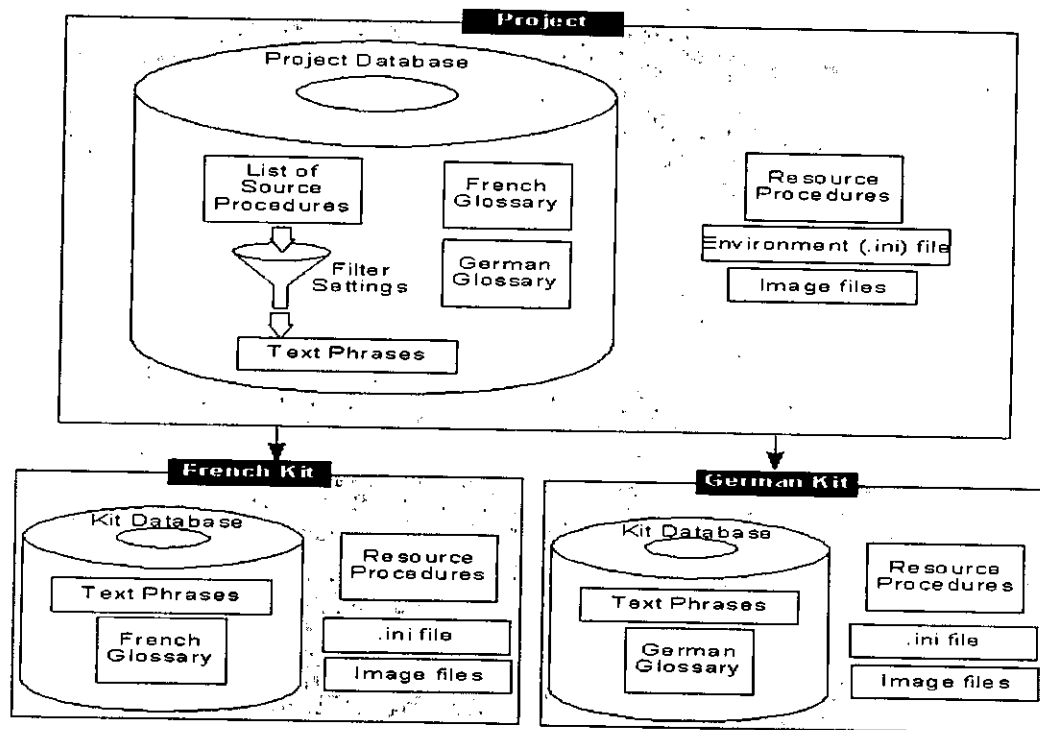


Figura 2.7_21 Traducción de Componentes de la Interfase de Usuario

2.7.3.7 La Arquitectura de Progress DataServer

La Arquitectura de Progress DataServer es un juego de servicios mejorados, interfaces personalizadas, y facilidades estándar de la industria que entregan aplicaciones de bancos de datos independientes a través de un ambiente heterogéneo. La Arquitectura de Progress DataServer también soporta la integración de datos de múltiples fuentes, la portabilidad de la aplicación a manejadores de datos, e interfaces abiertas estándar de la industria, tales como DDE, ODBC, y otros. Todas estas operaciones son manejadas automáticamente y son completamente irrelevantes para la aplicación misma.

2.7.3.8 Progress AppServers

Los "Progress AppServers" (servidores de aplicaciones de Progress) permiten la división de aplicaciones a través de la red de computadoras de una empresa. Soportan la ejecución remota de las aplicaciones lógicas a través de un procedimiento remoto llamado arquitectura basada en Progress 4GL. Así, la aplicación lógica ahora puede ser reubicada desde la máquina del cliente hasta el "Back-End" de la plataforma de bases de datos del servidor, lo que resulta en una ganancia dramática de rendimiento. Los Progress AppServers también pueden implantarse en sistemas dedicados y conectarse transparentemente a otros AppServers remotos para una flexibilidad máxima en la implantación de la aplicación.

Los Progress AppServers entregan mucho más que una arquitectura de implantación "n-tier" (de n grados). Progress proporciona la capacidad de encapsular la lógica de transacciones para que un diseñador codifique una vez y pueda reutilizar los AppServers a través de un rango de servidores basados en UNIX o basados en Windows NT, sin requerir cambios a la aplicación en los servidores de datos o en Progress.

2.7.3.9 "Data Replication" (Réplica de Datos)

La réplica de datos es la distribución de copias de información a uno o más sitios. En una sola empresa existen lugares que abarcan a menudo distintos límites de organización y de región en la empresa, que requieren una repartición oportuna y de manera consistente de los datos de las bases de datos de la empresa.

El desarrollo e implantación de un proceso de replicación exitoso involucra una cuidadosa planificación y el ingreso de los expertos del negocio, diseñadores de la aplicación, y administradores. Progress 4GL soporta la implementación de una replicación "Trigger-Based" (basada en disparos), y proporciona soporte al Data Base Manager para implementar la replicación "log-based" (basada en una bitácora).

2.7.3.10 Progress Relational DataBase Management System (RDBMS)

El Progress Relational DataBase Management System (sistema de manejo de bases de datos relacionales de Progress) o RDBMS, es completamente relacional, es un sistema de manejo de base de datos que cumple con el estándar SQL y que entrega una ejecución escalable y de alto desempeño para aplicaciones con transacciones complejas e intensivas. Progress RDBMS ofrece una interoperabilidad a través de plataformas, pudiendo ser ejecutado en más de 100 combinaciones diferentes de hardware y sistemas operativos, yendo desde PC's con sistema operativo Windows o Windows NT hasta los grandes sistemas de multiprocesamiento.

Progress RDBMS proporciona un juego de características específicamente dirigidas para el soporte de aplicaciones tanto Host-Based como cliente / servidor, en donde sean requeridas la ejecución de transacciones a altas tasas, los controles automáticos de divisas, la integración garantizada de los datos, y la administración mínima de las bases de datos. Se proporciona el máximo rendimiento a través del bloqueo en el nivel de registro, la estrategia de optimización de "queries" (consultas), y el almacenamiento inteligente de registros. Para garantizar la integridad de las transacciones que se separan

en las múltiples bases de datos, Progress RDBMS ofrece la recuperación "roll-forward" (hacia adelante) y recuperación "roll-back" (hacia atrás) de problemas en los sistemas, "commit" (aceptación permanente de cambios) de dos fases para transacciones distribuidas, un índice en línea y utilerías de respaldo sin interrupción de operaciones. Ofreciendo funcionalidad automatizada, el Progress RDBMS es excepcionalmente fácil de instalar, mantener y administrar.

2.7.3.11 Progress DataServers

Los "Progress DataServers" (servidores de datos de Progress) facilitan la ejecución de aplicaciones que acceden a bancos de datos de terceros. Se puede escoger entre una variedad de base de datos relacionales y sistemas de archivos, incluyendo Progress RDBMS. También se puede seleccionar una variedad de configuraciones de instalación, diferentes interfaces de usuario, opciones de redes de computadoras, protocolos, sistemas operativos y plataformas de hardware.

Para usuarios que desean tener acceso a datos o ejecutar aplicaciones con sistemas de manejo de datos diferentes de Progress, los Progress DataServers integran el ambiente Progress con varios de los populares sistemas de archivos de terceros, como DBMS, Oracle, MS SQL-Server, DB2/400, y otros. Trabajando con el diccionario de datos, cada servidor de datos optimiza las solicitudes de datos a Progress para un manejo específico, proporciona un manejo de datos transparente, y la portabilidad de la aplicación a través de diferentes manejadores de datos.

2.7.3.12 Amplio Rango de Portabilidad y Soporte

Progress se ejecuta en un impresionante rango de plataformas de Hardware, ambientes operativos e interfaces de usuario, y da soporte transparentemente a varios protocolos de red para dar a los diseñadores la flexibilidad de trabajar dentro del ambiente mas ajustado a sus propias necesidades de desarrollo e implantación.

Progress también soporta todas las bases de datos y sistemas de archivos líderes de terceros, con alto rendimiento en el manejo de datos, dando portabilidad a la aplicación y soportando toda las características específicas de las bases de datos.

En la figura 2.7_22 se muestran las diferentes plataformas y arquitecturas que soporta Progress.

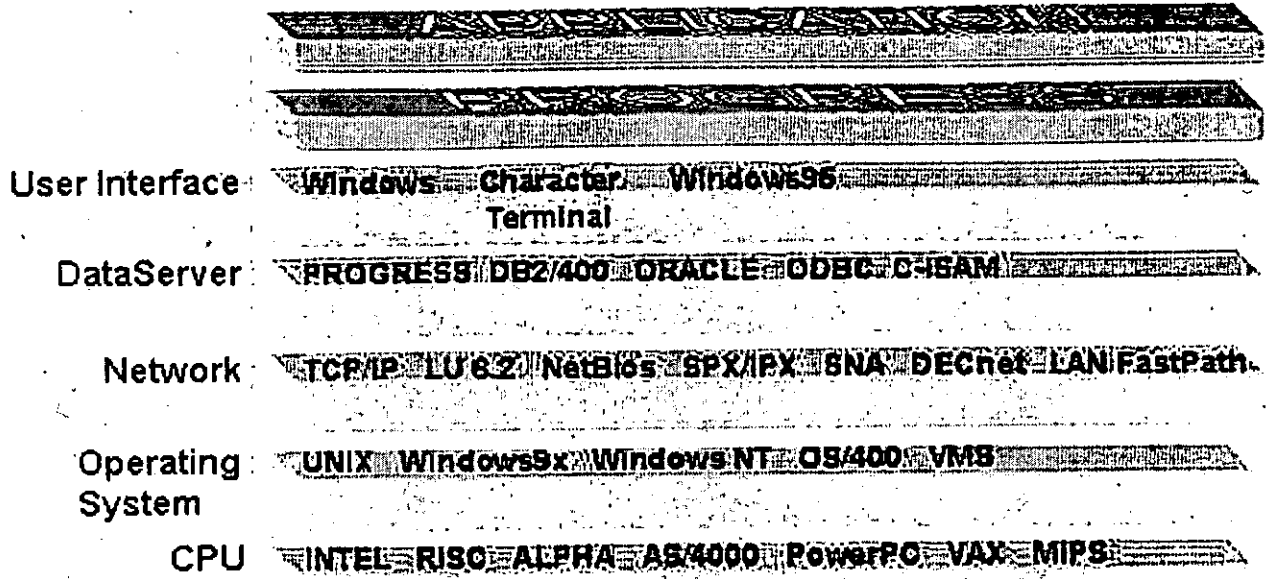


Figura 2.7_22 Rango de Portabilidad de Progress

CAPITULO 3
PLANTEAMIENTO DEL PROBLEMA
Y PROPUESTA DE SOLUCIÓN

3.1 Definición del Problema

3.1.1 Antecedentes

No hay duda de que ninguna economía se mantiene fuera de los procesos de globalización mundial. Directa o indirectamente, el proceso de acumulación de capital influye en los procesos nacionales en distintos grados, dependiendo de su integración y del grado de desarrollo de los países.

A partir de la década de los setenta, la industria en México era una industria protegida por el Estado en la que la obsolescencia se hizo presente, debido a que no tenía que renovar permanentemente su planta productiva ni competir con el exterior. Mientras tanto, enfrentadas a crisis de mercados o del petróleo, las naciones más desarrolladas del mundo capitalista se veían obligadas, como siempre, a renovar su tecnología. La ciencia jugó un papel básico en ese desarrollo, al mismo tiempo que nuestro país se fue rezagando. El desarrollo internacional fue entrelazando las economías de todos los países en los movimientos de capitales, mercancías y servicios. La globalización era ya una realidad.

En el centro de ese proceso se encuentra la tendencia de nivel mundial a la acumulación de capital. El capital siempre buscará dónde obtener mayores tasas y masas de ganancia y recursos que explotar -sean éstos humanos, naturales o técnicos-, además de buscar las facilidades y perspectivas de ganancias cuantiosas. Por supuesto, entre esas facilidades se cuentan prioritariamente las alianzas con Estados que muestran fehacientemente que no impedirán el desarrollo de las fuerzas productivas y que intervendrán sólo lo estrictamente necesario e indispensable en la economía, además de que esté dispuesto a vender las empresas públicas de su ramo que les podrían hacer competencia. En suma, se necesita encontrar Estados que están ansiosos por recibir inversiones extranjeras (ver la figura 3.1_1).

Estas son las condiciones externas. Las condiciones internas están dadas por el deterioro y el atraso del aparato productivo en México, que hace más costosas las relaciones de intercambio con el exterior.

Inversión extranjera directa ene-sep (Millones de dólares)

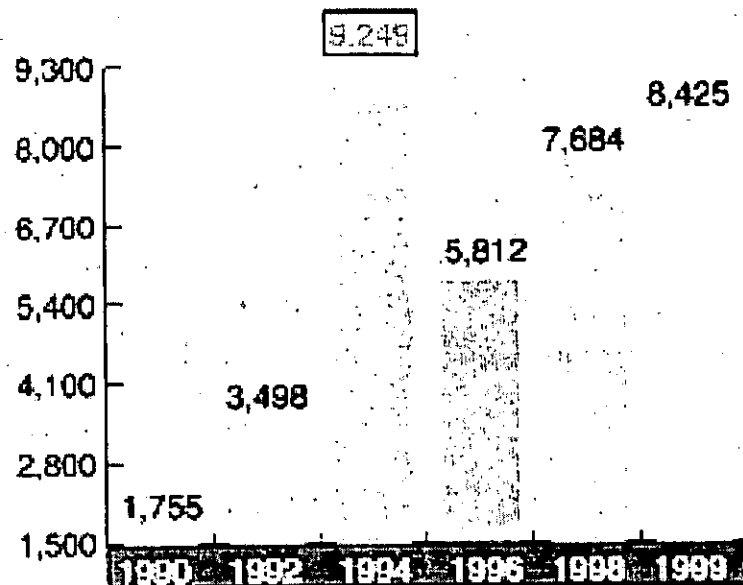


Figura 3.1_1 Inversión Extranjera Directa

En el nivel mundial actualmente hay una lucha feroz por atraer capitales y México no es una excepción; los elementos capital y trabajo determinarán, en gran medida, el futuro de la economía en México. Por otra parte, nuestro país está avanzando en su integración con EUA y Canadá a través del Tratado de Libre Comercio para Norteamérica (TLCNA). Con el TLCNA, y con el recientemente acordado Tratado de Libre Comercio con la Unión Europea (TLCUE), el proceso de fusión de los capitales nacionales con los internacionales se acentuará, con toda seguridad. Así, la economía en México marcha inevitablemente hacia una mayor integración con los EUA, Canadá y Europa, en donde su fuerza de trabajo barata es un incentivo para una inversión externa mayor.

En 1998, el comercio entre los tres países del Tratado de Libre Comercio ascendió a 511 mil 360 millones de dólares, casi cuatro veces más que antes de que entrara en vigor el TLCNA el 1 de enero de 1994. Las exportaciones de EUA hacia México y Canadá han aumentado casi 66 por ciento y superan un valor de 235 mil millones de dólares. Estados Unidos, por ejemplo, vende a Canadá más productos que a todas los países del este de Asia, y a México cinco veces más que a China.

Con respecto a la Banca en México, ésta sufre las transformaciones que se están dando en la intermediación bancaria en el país y en el nivel mundial. Entre las más significativas figura el que la Banca deja de jugar únicamente el rol de recibir y prestar depósitos, pues ahora es también emisora de valores, en combinación con empresas privadas y el gobierno. Estas empresas recurren cada vez más para financiar sus necesidades a la emisión de bonos y obligaciones y menos al crédito bancario. Por otra parte, los cambios que están buscando los gobiernos de Japón y E.U.A. levantarán las restricciones que

les impide funcionar como Banca universal en sus mercados domésticos. Esto confirma la tendencia mundial hacia la homogeneización de las instituciones financieras.

Ya el pasado mes de Noviembre la Cámara de Representantes aprobó una reforma general del sistema financiero y bancario de Estados Unidos que reemplaza las leyes que desde la Gran Depresión de los años 30 impusieron barreras a los servicios de los bancos. La principal reforma en los servicios financieros es la eliminación de las barreras que existen entre los bancos, las empresas de acciones y valores, y las compañías de seguros, lo cual abre la puerta a las fusiones y la creación de compañías gigantescas que ofrezcan todos los servicios.

En tal sentido, las reformas hechas en los últimos años por el gobierno de E.U.A. buscan otorgar mayores poderes a las grandes instituciones bancarias para operar en la economía y el sistema financiero del país. Estos cambios, que se operan en E.U.A. y se han incluido en el TLCNA, buscando la liberalización de los servicios financieros, obligaron al Estado Mexicano a reprivatizar la Banca, la cual debe ponerse al día, pues en el momento actual, no está en capacidad de competir con los poderosos bancos norteamericanos y japoneses ni (debido al TLCUE) con los bancos europeos.

Asimismo, La reciente firma de un acuerdo de libre comercio entre México y la Unión Europea generará cambios importantes en la estructura del sistema financiero mexicano y se espera que sea a partir del año 2001 cuando se empiece a observar un mayor flujo de recursos de inversionistas europeos hacia ese sector. Debido a la incertidumbre que se ha observado en los últimos meses en los mercados bursátil y cambiario, se frenó la llegada de recursos a los bancos mexicanos; el ingreso de inversión europea a los bancos nacionales podría empezar con mayor dinamismo en el año 2001 y el tratado de libre comercio con la Unión Europea provocará que los grandes bancos de este continente se empiecen a fijar en México.

Como la apertura financiera es total, ya en 1991 se tenían oficinas de 64 bancos extranjeros - como cabezas de playa - para iniciar operaciones en el mercado nacional. A la larga, los grandes bancos mexicanos se reorganizarán y remontarán su actual baja productividad, para competir en el nuevo marco de apertura de los servicios financieros internacionales. Algunos bancos no podrán competir y tendrán que desaparecer.

Por otra parte, la Banca Comercial en operación regular en México ha disminuido su participación como intermediario del ahorro del sector privado, según cifras del Banco de México (Banxico), ya que el saldo de sus activos totales como proporción del PIB pasó de 82.5% a 60% entre diciembre de 1994 y junio de 1999. Banxico argumenta que pese al financiamiento decreciente de la banca, la actividad económica se ha apoyado en fuentes no bancarias, lo que ha permitido en parte compensar la falta de intermediación bancaria (ver, la figura 3.1_2).

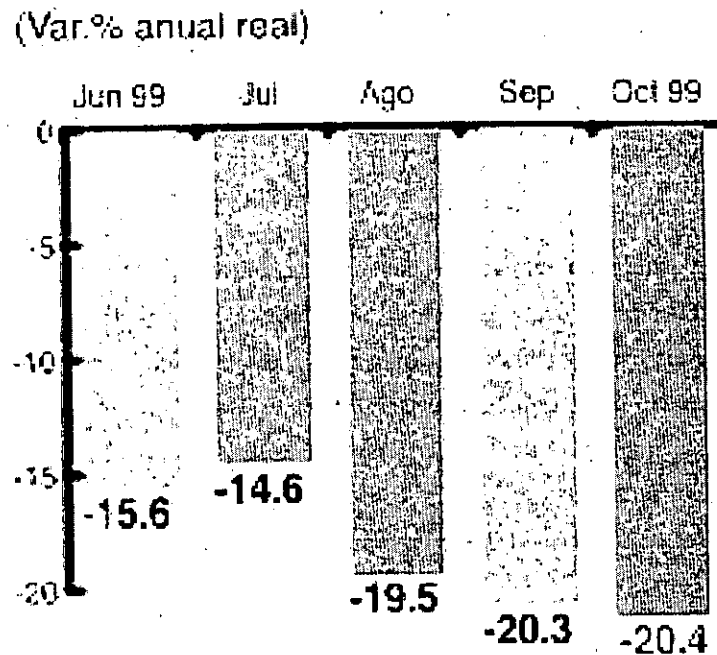


Figura 3.1_2 Financiamiento de la Banca Mexicana al Sector Privado

Tan sólo el financiamiento no bancario y del exterior a las empresas y personas físicas del país creció en 7.6% en términos reales en el último año, mientras que en la banca este segmento decreció en 8.2% real en el mismo lapso. En el caso del financiamiento no bancario al consumo éste aumentó 12.5%, mientras que el crédito hipotecario creció en 35%. En los bancos este último está totalmente cerrado.

Tanto los efectos de la privatización bancaria, como los de la apertura financiera al exterior, traerán la baja de las tasas de interés y el que el dinero se hará más barato para el inversionista y el industrial. En el caso de México, la competencia financiera provocará una lucha abierta por la búsqueda de integración de capitales; según sea rápida o lenta la apertura en el rubro. Si es rápida, las tasas de interés tenderán a caer, en forma más rápida que si la apertura es gradual.

En conclusión, con la llegada de instituciones bancarias extranjeras al país, las entidades financieras nacionales ven amenazada su participación en el mercado nacional, teniendo que superarse en todos los aspectos referentes a los servicios que puedan proporcionar a sus clientes. De no ser así, la institución financiera será desplazada del mercado nacional y no tendrá oportunidad de crecer en el mercado exterior. Toda institución financiera debe tener la capacidad de adaptarse para seguir existiendo o no será competitiva con las demás instituciones. Para estar a la vanguardia en competitividad en esta era de avances tecnológicos y de globalización, las instituciones financieras están obligadas a mejorar sus servicios y a satisfacer las necesidades bancarias de sus clientes con nuevos servicios.

3.1.2 Definición del Problema

Con el fin de cumplir con las estrategias mencionadas arriba, el área de negocios de una entidad financiera busca captar divisas y tener fondeo de dólares para apoyar a otras áreas internacionales y a las áreas de cambios para la venta de dólares por pesos o por otra divisa.

Como parte de este objetivo, y para proporcionar servicios más competitivos con los de los bancos nacionales y extranjeros, la institución ofrecerá a sus clientes un nuevo servicio, Chequera en Dólares Americanos (USD), teniendo como base el cobro / depósito en México y los Estados Unidos de Norteamérica. Se pretende ofrecer este servicio a personas físicas con giro empresarial con fuerte movimiento de dólares y también para las empresas o personas morales que tengan grandes movimientos de exportación e importación de productos. Los principales servicios que se ofrecerán son: depósito en dólares, pagos en dólares y transferencias de pagos en forma electrónica hacia cualquier banco del mundo, además de que se le pagará al cliente un interés en dólares por el saldo promedio que mantenga en su cuenta.

La institución financiera tiene centros financieros en las ciudades más importantes de México (D.F., Guadalajara, Monterrey, Puebla y Tijuana) y en los Estados Unidos estará respaldada por el Barclays Bank, con sede en Miami, Florida, a través de sus servicios de "Payable Through". Los clientes que requieran de este nuevo servicio serán captados en los centros regionales por los promotores. La institución invertirá su captación por medio de su tesorería en "Overnight" en bancos nacionales o extranjeros en el caso de que se encuentre una situación de "Largo" (entrada excesiva de depósitos); en el caso contrario, en una situación de "Corto" (que haya más salida de pagos de la Tesorería), se liquidarán los activos que se tengan en inversión.

Cabe mencionar que para un buen funcionamiento de la chequera serán las cuentas concentradoras del Barclays Bank y del Chase Bank en EUA las que permitirán el depósito de apertura y el envío de los pagos de los clientes.

Como resultado de este requerimiento de negocios de la institución financiera, se tiene la necesidad de desarrollar e implantar la aplicación o sistema de cómputo que soporte el servicio mencionado de chequeras para la captación de dólares americanos.

Dicha aplicación debe ser acorde con las necesidades de los bancos concentradores así como del manejo de los centros financieros. Estas necesidades son:

- Apertura de cuentas de cheques en forma expedita
- Administrar las chequeras en dólares de los clientes y llevar un control de las aperturas y cancelaciones así como de los depósitos y retiros efectuados
- Control integral de todos los movimientos realizados en cada cuenta de cheques
- Generación de cuentas de cheques con algoritmo especial
- Solicitud de chequeras en forma electrónica al proveedor de chequeras

- Poder realizar pagos a sus proveedores en el ámbito mundial
- Transferencias electrónicas a bancos del extranjero
- Seguridad de la información
- Entrega oportuna de documentación a bancos proveedores (Barclays Bank)
- Entrega oportuna de la información a los clientes.

Esta aplicación deberá permitir a la institución financiera una mayor eficiencia en el manejo y administración de las chequeras, así como aplicar mejor sus recursos humanos y monetarios al tener su contabilidad al día. De igual forma se trata de mejorar los procesos actuales con el nuevo sistema dando un apoyo más preciso para la entrega de la información. Además, debe dar una mejor atención a los clientes al procesar en forma confiable, confidencial y más rápida sus operaciones.

3.2 Situación Actual

En la actualidad y como parte de la falta de competitividad con los bancos e instituciones financieras extranjeras, muchos de los sistemas de las instituciones financieras en México y de la que nos ocupa en particular, están funcionando con hojas electrónicas, memos en procesador de palabras y operaciones manuales, o en su defecto trabajan con sistemas aplicativos poco cohesivos, lo que hace muy difícil la integración de todos los movimientos contables.

Esto trae como consecuencia un mal servicio al cliente porque se le proporciona un servicio lento y con posibilidad de errores, además de que no se cuenta con la información en línea dentro de las sucursales, por lo que los promotores a su vez no pueden proporcionar rápidamente la información a sus clientes.

3.3 Requerimientos del Usuario, Recopilación y Análisis de la Información

3.3.1 Especificación Preliminar de los Requerimientos

Con la problemática que actualmente se tiene al realizar la apertura de chequera en dólares, es necesario diseñar un sistema automático, en el cual el usuario no tenga que realizar muchas operaciones manuales y el tiempo requerido para la asignación de la chequera disminuya.

Con el desarrollo del sistema se pretende disminuir los tiempos de espera, tanto para el cliente como para los asesores financieros, que son las personas encargadas de atraer a los clientes susceptibles de otorgarles la cuenta de cheques, para que estos puedan brindar a los clientes los servicios con que cuentan la institución.

En la asignación de chequeras se ven involucradas varias áreas además de los asesores financieros y son las que en un momento dado pueden determinar si es viable que se asigne la chequera a los clientes, además de los servicios con que van a contar, por los que son los primeros usuarios en requerir dicho sistema.

Además se pretende:

- Controlar todos los pagos y depósitos de cada una de las cuentas, así como también el poder brindar la información necesaria a todos los clientes.
- Tener el control de los depósitos y cobros que se realizan en los centros financieros como también los que se hacen en todo el mundo.
- El poder enviar a los clientes sus estados de cuentas oportunos y detallados conteniendo todos los movimientos que tuvieron durante el mes, además de poder contar con la información de los intereses que obtuvieron durante el mes.
- Poder brindar información a los asesores financieros de la situación en que se encuentran sus clientes, de los movimientos que han tenido durante el día, de la cancelación de cheques que el cliente solicite y de los cheques que hayan sido rebotados.
- Solicitar la reposición de chequeras o bien la cancelación de las mismas, ya sea por sobregiro o pérdida de la misma.

Ante tal situación se requiere de un sistema que agilice dichos tramites tanto del Asesor Financiero como de las demás áreas involucradas, así como la asignación de las chequeras, además de que la institución tenga un control de las mismas y se cuente con toda la información que se requiera.

El sistema se manejará de manera centralizada debido a las políticas que presenta la institución para la asignación de chequeras y demás procesos; solamente los centros regionales podrán hacer uso de cierta información que los clientes soliciten.

Este sistema deberá estar realizado en modo gráfico y correr bajo la plataforma de ventanas; deberá estar integrado por un menú principal y cada opción del menú contendrá un submenú. Este sistema tendrá toda la funcionalidad que se pretende dar a la institución financiera, por lo cual deberá ser lo más amigable posible para el usuario, de manera que este no tenga que realizar demasiadas funciones tanto para la apertura de cuentas como para obtener la información que requieran las diferentes áreas. El sistema deberá contar con claves de acceso y de acuerdo con estas, se podrán realizar ciertas actividades dentro del sistema.

3.3.2 Especificación Detallada de los Requerimientos de Software

El menú que se pretende tener en el sistema, contendrá como principales funciones las siguientes:

- Manejo de operaciones diarias.
- Cierre Mensual de operaciones.
- Control de Cuentas.
- Parámetros de modificaciones para el control de procesos.
- Generación de documentos por las nuevas aperturas de cuentas.
- Control de movimientos de Chemical Bank y Barclays Bank.
- Pagos.

El Manejo de operaciones diarias tendrá como procesos principales que se realizaran diariamente los siguientes:

- Movimientos que se llevan a cabo con Barclays Bank.
- Cancelación y modificación de movimientos.
- Control diario de sobregiros de las cuentas.
- Bloqueo de cuentas sobregiradas.
- Consultas de saldos y reportes de los saldos y de los movimientos.
- Pagos diarios que se dan en las chequeras.
- Altas de remesas, liberación e impresión de las mismas.

CAPITULO 3. PLANTEAMIENTO DEL PROBLEMA Y PROPUESTA DE SOLUCIÓN

El cierre Mensual de Operaciones deberá contener como funciones principales:

- Cálculo de interés diario y de saldo promedio de las cuentas, así como las impresiones de los estados de cuentas para los clientes, especificando a detalle los movimientos que tuvieron durante el mes.
- Generación de los reportes conteniendo los movimientos que se dieron en las cuentas a causas del saldo que mantuvieron durante un periodo.
- Cancelación de movimientos durante el mes.
- Reportes de ingresos, gastos y utilidades que se generan durante el cierre de mes.
- Reportes de los estados de las cuentas.
- Aplicación de los movimientos para la contabilidad de la institución.

El control de cuentas deberá contener como funciones principales las siguientes:

- Captura de datos confidenciales de los clientes.
- Procesar altas, bajas y cambios las cuentas.
- Asignación automática para la apertura de cuentas nuevas.
- Congelación de cheques así como de chequeras debido al extravío o robo de los mismos.
- Reportes de cuentas y cheques bloqueados y extraviados.
- Reportes de apertura de cuentas.
- Reportes de Cancelación de cuentas.
- Reporte para estadísticas de los montos que se manejan en las chequeras.
- Reporte de acuerdo a la clasificación de las cuentas.

Las funciones que deben controlar los Parámetros de Modificaciones para el Control de Procesos deberán ser abiertas. Esto es, que el usuario sea el que deba modificar ciertos parámetros de acuerdo a las necesidades que se le presenten, como son:

- La impresión de los reportes.
- Poder cambiar ciertas características de los movimientos sin que esto repercuta en el sistema.
- Modificar los parámetros de Saldos mínimos con que deben de contar las cuentas así como el saldo mínimo que deben de tener las cuentas para su apertura.
- El mantenimiento de las tasas de interés que se pagaran a las cuentas de acuerdo al monto que presentaron, así como también la multa que se deberá de cobrar por el sobregiro si fuera el caso.
- Diferentes tipos de reportes de los diferentes centros regionales, así como qué características tendrán las cuentas de los clientes registrados por ese centro, de acuerdo a la región en donde se encuentre ubicado.

En la generación de documentos de las cuentas se tendrá como funciones principales:

- La generación de los memos que se tienen que hacer para la solicitud de las chequeras, así como también el poder autorizar el cobro de cheques expedidos por los clientes desde cualquier centro financiero del país.
- El poder generar información de los cheques que son cobrados en los diferentes centros financieros con que cuenta la institución. De igual manera la información de los mismos que son rebotados.
- Generar toda la información de los saldos que posee cada una de las chequeras para el control interno de la institución, así como informarle a cada uno de los asesores financieros el estado que presentan sus clientes con relación a sus cuentas.
- Generar información de todos los depósitos que se realicen dentro de las filiales que tiene la institución como son: Barclays y Chemical Bank, así como también poder contar con consultas en línea sobre el estado que presentan los movimientos de cada una de las cuentas.

Otras funciones del sistema son:

- El poder Consultar los saldos de cada una de las cuentas que se tienen con las filiales, además de poder consultar todos los movimientos que se han tenido en estas.
- Contar con procesos que actualicen los saldos de cada una de las cuentas que presenten movimientos de las diferentes filiales para poder así tener la información en línea.
- Poder consultar los movimientos cancelados de las cuentas que fueron rechazados por las filiales. Además de contar con información de los movimientos que están pendientes de ser abonados o cargados de cada una de las cuentas.

También se deberá de poder contar con un control de los diferentes pagos que se realizan y que a su vez esté desglosado por diferentes conceptos que pueden ser:

- Pagos por área, plaza y ciudad
- Pagos de los promotores
- Pagos que se realicen en los diferentes países.
- Pagos que se realizan en las diferentes instituciones bancarias
- Validación de los pagos.
- Reportes de los diferentes pagos que se realicen
- Reportes de generales de los pagos realizados en los diferentes centros financieros.
- Reportes de los cheques cobrados en cada uno de los centros financieros.
- Reportes de pagos rechazados en los diferentes centros financieros.
- Reportes de las cuentas que se encuentran en cada uno de los diferentes centros regionales.

Se debe generar información para las diferentes áreas, dentro de la institución las cuales la analizarán y tomarán decisiones con base en la misma.

3.3.2.1 Reportes a Generar

Se debe de generar un memo indicando el depósito que se realiza en Chemical Bank en el momento en que el cliente realiza la apertura de cuenta, en el cual se indica la cantidad del depósito, y con qué número de cheque se genera dicho depósito para que sea cobrada por la institución; con este documento se ampara la institución para cualquier aclaración que se dé sobre el mismo.

También se genera una ficha de depósito, la cual indica al Barclays Bank que verifique la información correspondiente al depósito realizado, el importe del monto así como la cuenta a la que corresponde dicho depósito. Esto tiene como finalidad llevar el control de los depósitos realizados y que estos cuadren con la información con que cuenta la institución en sus registros.

Como el sistema es centralizado, se genera información global de todos los cheques cobrados en los diferentes centros financieros. La información muestra el número de cuenta, el número de cheque y el importe del monto con que se cobro dicho cheque, la clave del centro financiero y nombre del mismo y el lugar donde fue cobrado, así como el destino que tuvo dicho cheque.

El reporte por tipo de movimiento tiene como finalidad tener un control de los movimientos que ha tenido cada una de las cuentas, así como del propietario de la misma y el importe con que se realizan los movimientos de una determinada fecha, para tener clasificados a los diferentes clientes con que cuenta la institución.

Los reportes de los estados de cuentas globales representan los movimientos que se realizan durante una fecha dada, debido a que esta información es necesaria para los centros financieros al igual que para el área de tesorería de la institución, ya que los movimientos se describen a detalle. Es decir, si el movimiento fue un cobro o un depósito o bien si es el pago de interés del monto que se encuentra depositado en la cuenta. Además describe que clientes son los que han realizado movimientos en sus cuentas e indica el importe de los mismos.

Este tipo de reporte es muy útil para las distintas áreas involucradas y sobre todo para la tesorería, debido a que muestra un balance de cuanto dinero ha sido depositado o bien retirado por los clientes, y con base en esta información es que se toma una decisión para el fondeo que se realiza con las instituciones bancarias del extranjero.

Además se debe contar con información de todos aquellos movimientos que se han generado en el transcurso del día de operación, ya sea de cargo o de abono, conociendo también el nombre del cliente que realiza el movimiento, además de conocer las características de éste. Deberá estar disponible un resumen de cuántos movimientos se realizaron en el día, que muestre también el importe del monto de cargos o de abonos con que termina un día de operación.

De igual manera se deberá contar con información de la situación que presente Chemical Bank, detallando el número de movimientos que han efectuado los diferentes clientes, indicando el importe de cada uno, y conteniendo un resumen de cómo queda Chemical Bank ante tales movimientos.

Todos los inversionistas presentan cierto perfil que interesa a ciertas áreas de la institución, por lo que se necesita tener información de tales clientes, como el nombre del cliente, cual es el estado que presentan sus movimientos durante un periodo de tiempo, el saldo promedio y el saldo final que presenten dichas cuentas, además de estar clasificada y ordenada por los diferentes centros financieros y zonas que tiene clasificadas la institución.

Otra información que se requiere dentro de la institución y que deberá ser proporcionada por el sistema, es la referente a todas las cuentas que se encuentran registradas dentro de la misma, como su saldo inicial, el saldo de los movimientos que se registraron en el día, además del saldo con que cerró la cuenta y otros movimientos que se hayan efectuado.

El sistema también generará información sobre la diferencia que existe en el saldo de las chequeras, debido a que es información que requieren los asesores financieros de los centros. Esta información deberá mostrar la cuenta, el nombre del cliente, el estado en que se encuentra, ya sea una alta o bien que esté cancelada, el saldo disponible que presenta la cuenta, el saldo cancelado y la fecha en que se realizó la alta de la cuenta.

Como información que debe proporcionar la institución hacia sus clientes, se tienen los estados de cuenta en donde vendrán descritos los movimientos que se realizaron durante un periodo. Estos estados de cuenta deberán de contar con la siguiente información: la fecha en que se realizaron sus movimientos, su identificación, la descripción del movimiento, los cargos (débitos), los abonos (créditos) y el importe de dichos movimientos. Además contienen un resumen del importe total de créditos y abonos, el total del saldo de que se dispone y el saldo total de que se puede hacer uso en la cuenta.

Otra información que requiere la institución para realizar muestreos de la clase de clientes con que cuenta, de acuerdo al monto de su inversión; deberá mostrar el número de cuenta, el nombre del cliente y el monto de la inversión con que cuenta hasta ese momento. Esta información deberá obtenerse de manera flexible, es decir, variando el rango de inversión se obtendrá a todo los clientes que se encuentren dentro de ese rango.

También se requiere obtener información referente a los cobros de cheques que se realizan en los diferentes centros financieros de la institución. Esta información contendrá un número secuencial de movimientos, una clave que describe el cobro y el importe del mismo. Así mismo mostrará un resumen por centro financiero, el importe total de cobros de cheques y cuantos movimientos fueron realizados.

Como existen varios movimientos que afectan a los clientes de la institución y para que esta tenga el control de los mismos, se quiere conocer para cada uno que tipo de movimiento es, por lo que es necesario que el sistema produzca esa información para que las áreas de contabilidad la conozcan. La información deberá contener los siguientes datos: una clave, la cuenta a la que va a afectar, el importe de la afectación, la fecha de cuando se realiza el movimiento y el tipo de movimiento que se realiza.

CAPITULO 3. PLANTEAMIENTO DEL PROBLEMA Y PROPUESTA DE SOLUCIÓN

Además se debe de contar con información que indique que cheques han sido cancelados para su cobro, por parte del cliente o bien por estar alterados. Por lo anterior se requiere que la información contenga los siguientes datos: la cuenta y el nombre del cliente, el número de cheque inicial y final a cancelar y la fecha en que se realiza la cancelación de los mismos.

El presente documento tiene como finalidad describir el problema que se plantea y la solución que se propone para resolverlo. El problema que se plantea es la falta de información que indique que cheques han sido cancelados para su cobro, por parte del cliente o bien por estar alterados. Por lo anterior se requiere que la información contenga los siguientes datos: la cuenta y el nombre del cliente, el número de cheque inicial y final a cancelar y la fecha en que se realiza la cancelación de los mismos.

La solución que se propone es la implementación de un sistema de cancelación de cheques que permita al cliente cancelar los cheques que desea cobrar y que permita al sistema registrar la cancelación de los cheques. El sistema de cancelación de cheques debe de tener los siguientes requisitos: que permita al cliente cancelar los cheques que desea cobrar y que permita al sistema registrar la cancelación de los cheques.

El sistema de cancelación de cheques debe de tener los siguientes requisitos: que permita al cliente cancelar los cheques que desea cobrar y que permita al sistema registrar la cancelación de los cheques. El sistema de cancelación de cheques debe de tener los siguientes requisitos: que permita al cliente cancelar los cheques que desea cobrar y que permita al sistema registrar la cancelación de los cheques.

El sistema de cancelación de cheques debe de tener los siguientes requisitos: que permita al cliente cancelar los cheques que desea cobrar y que permita al sistema registrar la cancelación de los cheques. El sistema de cancelación de cheques debe de tener los siguientes requisitos: que permita al cliente cancelar los cheques que desea cobrar y que permita al sistema registrar la cancelación de los cheques.

El sistema de cancelación de cheques debe de tener los siguientes requisitos: que permita al cliente cancelar los cheques que desea cobrar y que permita al sistema registrar la cancelación de los cheques. El sistema de cancelación de cheques debe de tener los siguientes requisitos: que permita al cliente cancelar los cheques que desea cobrar y que permita al sistema registrar la cancelación de los cheques.

El sistema de cancelación de cheques debe de tener los siguientes requisitos: que permita al cliente cancelar los cheques que desea cobrar y que permita al sistema registrar la cancelación de los cheques. El sistema de cancelación de cheques debe de tener los siguientes requisitos: que permita al cliente cancelar los cheques que desea cobrar y que permita al sistema registrar la cancelación de los cheques.

El sistema de cancelación de cheques debe de tener los siguientes requisitos: que permita al cliente cancelar los cheques que desea cobrar y que permita al sistema registrar la cancelación de los cheques. El sistema de cancelación de cheques debe de tener los siguientes requisitos: que permita al cliente cancelar los cheques que desea cobrar y que permita al sistema registrar la cancelación de los cheques.

3.4 Descomposición funcional

3.4.1 Estructura Jerárquica de la Institución

La estructura jerárquica de una Institución Financiera está constituida principalmente por un Director General y Subdirectores de Productos, Tesorería y Créditos, como se muestra en la figura 3.4_1.

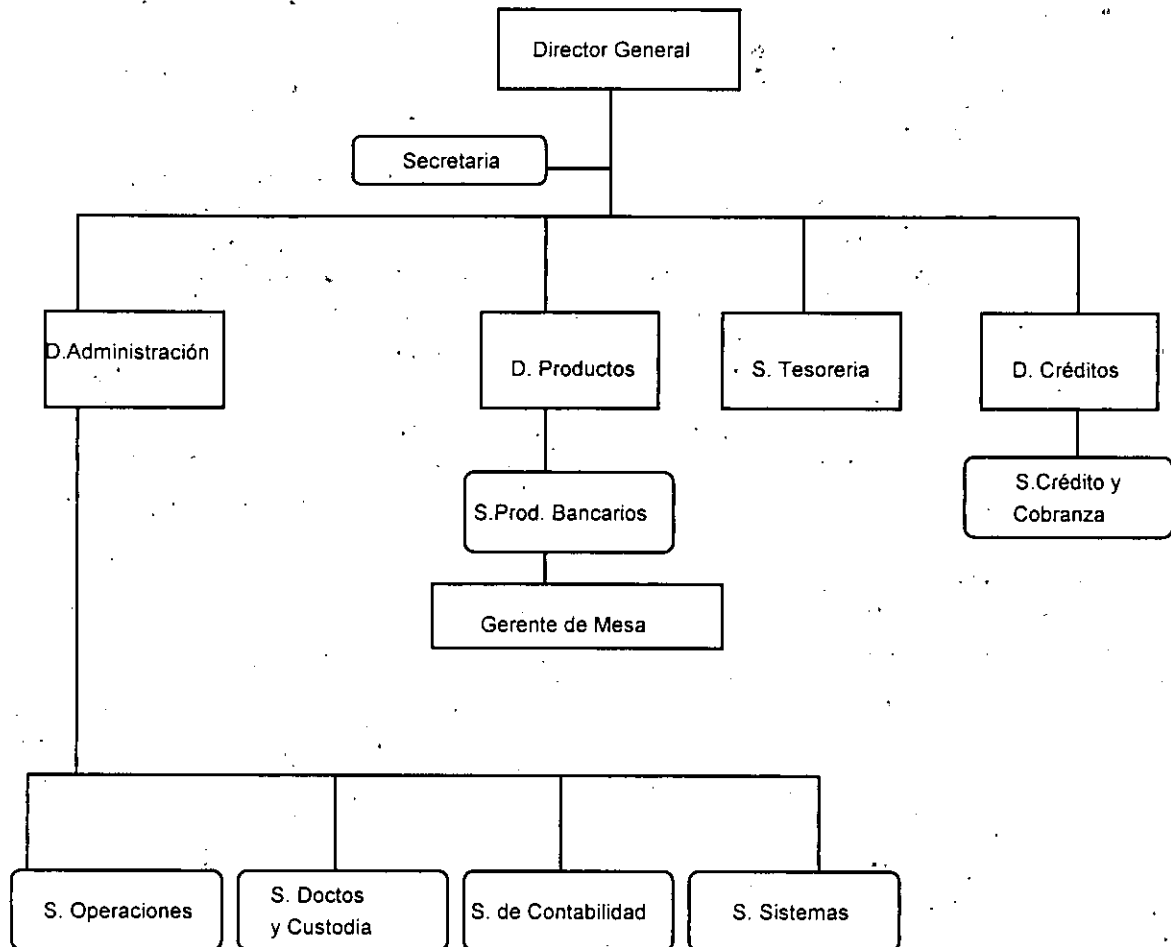


Figura 3.4_1 Estructura Jerárquica de una Institución Bancaria

El Director General dirige y administra las operaciones de las Subdirecciones y Direcciones. La responsabilidad del desarrollo de las actividades principales recae en Administración, los asesores de crédito y los subdirectores de documentación y custodia.

Las funciones de cada uno de ellos se describen a continuación.

3.4.1.1 Subdirector de Operaciones y Chequeras

Dirige la operación del área de chequeras, verificando que se cumplan las políticas y filtros correspondientes.

Dirige la liberación de pagos en el área de operaciones y verifica el adecuado fondeo de las cuentas de la Institución Financiera.

Actividades específicas:

- Dirige la actualización de saldos en cuentas de cheques a través de los reportes internacionales, para que la operación fluya de manera constante.
- Dirige el cobro de remesas internacionales para vigilar la liberación de fondos en cuentas de cheques y dar seguimiento a aquellas remesas que sean rechazadas.
- Dirige el pago de cheques a través de la red de sucursales, mediante una clave de autorización y coordina la liquidación de los mismos, con giros, remesas y cheques de viajero.
- Verifica que se cumplan los filtros necesarios para realizar transferencias de fondos, inversiones, pagos de tarjeta de crédito o liquidaciones de créditos con cargo a cuenta de cheques.
- Verifica que se cumplan las instrucciones de pago de los clientes al vencimiento de las inversiones correspondientes.

3.4.1.2 Subdirector de Documentación y Custodia

Es el responsable de la administración, supervisión y control de la apertura y documentación de cuentas en la Institución.

Actividades Específicas:

- Coordina las funciones de apertura, contratación, actualización, restricción, reasignación y asociación de cuentas.
- Define y comunica el marco legal que rige las cuentas en coordinación con el área jurídica.
- Diseña formatos de documentación de apertura y control interno.
- Revisa la documentación necesaria para la apertura de cuentas.
- Detecta apertura de cuentas sospechosas.
- Elabora el reporte del estado que mantienen las cuentas vigentes, contra las cuentas documentadas, así como de cuentas nuevas.
- Analiza la información contenida en la documentación de cuentas recién abiertas.
- Elabora reportes de control de cuentas (altas y cancelaciones).
- Supervisa y coordina la impresión remota de los estados de cuenta.

3.4.1.3 Subdirector de Contabilidad

Es el responsable de coordinar y supervisar que se realice una correcta contabilidad de las operaciones de la Institución.

Actividades Específicas:

- Supervisa la contabilidad en la Institución Financiera.
- Coordina el cierre contable.
- Controla Recursos.
- Realiza certificaciones contables para la recuperación de crédito.
- Realiza y presenta informes financieros para las áreas de la Institución, así como para los organismos reguladores.

3.4.1.4 Subdirección de Sistemas

Es el responsable de dirigir y coordinar el desarrollo e implantación de sistemas, para apoyar las necesidades de información de las diferentes áreas de operación.

Actividades Específicas:

- Establece la coordinación con las diferentes áreas de sistemas del grupo financiero.
- Determina los aspectos óptimos para el manejo y distribución del equipo de cómputo.
- Coordina el análisis de sistemas, para detectar las necesidades de información, con el fin de maximizar el aprovechamiento del tiempo y del recurso humano.
- Define responsabilidades y fija compromisos de sistemas.

3.4.1.5 Subdirector de Productos Bancarios

Es el responsable de dirigir, administrar y coordinar el proceso de cotización y captación relacionado con las áreas de promoción, así como con la tesorería internacional Bancaria.

Actividades Específicas:

- Coordina el proceso de cotización y captación, junto con la Tesorería Internacional.
- Es responsable de la captación y costos de fondeo.
- Es responsable del funcionamiento de los productos bancarios en la red de promoción, asegurando la calidad en los procesos.
- Realiza estadísticas y análisis del pasivo.
- Valida ingresos a la red de promoción.

3.4.1.6 Gerente de Mesa

Es responsable de establecer las tasas de interés de captación y colocación.

Actividades Específicas:

- Cotiza y asigna tasas.
- Realiza la captación de recursos.
- Da atención y asesoría a la red de promoción.
- Asigna ingresos a ejecutivos.
- Analiza y establece alternativas de inversión.

3.4.1.7 Administrador de Tesorería

Asegura la correcta administración de los recursos sobre la base de la información de las áreas involucradas.

Actividades Específicas:

- Concentra presupuestos.
- Analiza variaciones presupuestales.
- Diseña modelos financieros.
- Efectúa planeación financiera.
- Controla gastos y adquisiciones.

3.4.1.8 Director de Créditos

Es responsable de dirigir y coordinar la operación, administración, seguimiento y cobranza de la cartera de crédito, vigilando el apego a las políticas de crédito.

Actividades Específicas:

- Da apoyo y coordinación a las áreas de recuperación, las cuales tienen a su cargo la recuperación o reestructuración de los créditos vencidos.
- Se coordina con las otras direcciones de la Institución Financiera, en las operaciones y proyectos a realizar.
- Coordina la correcta aplicación de las características generales de la operación, así como las garantías inherentes en su caso, con el fin de minimizar riesgos.

3.4.1.9 Subdirector de Crédito y Cobranza

Es el responsable de la administración y seguimiento de la cartera de crédito, coordinándose con las áreas de contabilidad y tesorería.

Actividades Específicas:

- Proporciona información a las distintas áreas de recuperación del historial del acreditado, con el fin de conocer y estructurar las negociaciones que se realicen dentro de la recuperación.
- Custodia los documentos de los créditos vigentes, derivados de las diferentes operaciones (pagares, contratos, fideicomisos, etc.).
- Coordina la correcta aplicación de las operaciones realizadas por la Institución, con el fin de vigilar el correcto a la normatividad de crédito vigente.
- Se relaciona con diversas áreas con el fin de eliminar al máximo el riesgo de cobranza.

3.4.2 Diagrama estructural de la Problemática de una Institución Financiera

A continuación, en la figura 3.4_2 se muestran los problemas esenciales en el desarrollo funcional de una Institución Financiera.

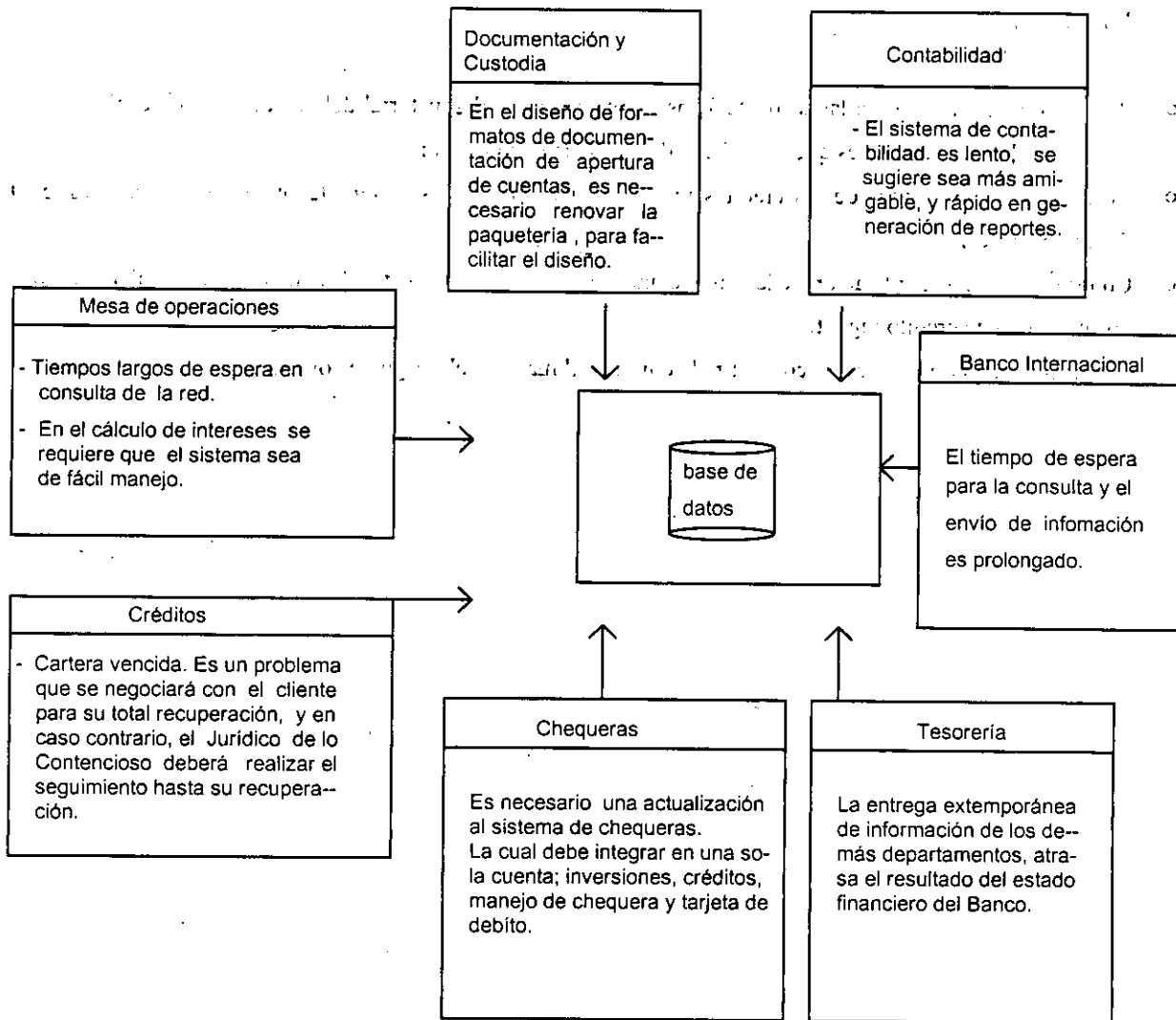


Figura 3.4_2 Diagrama Estructural de la problemática de una Institución Financiera

3.4.3 Apertura de una cuenta de cheques

A continuación en la figura 3.4_3 se describe el procedimiento para la apertura de una cuenta de cheques.

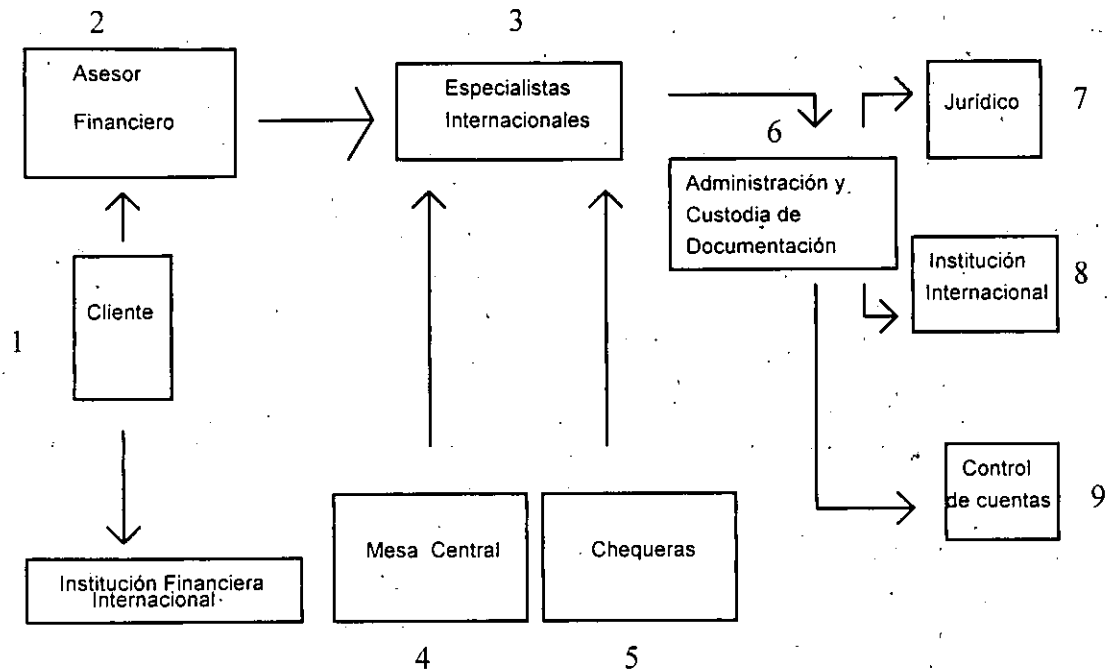


Figura 3.4_3 Diagrama Funcional de la Apertura de una cuenta de cheques

El cliente (1):

- Recibe promoción y solicita la apertura de la cuenta de cheques.
- Se le proporciona papelería y firma la documentación.
- Efectúa depósito inicial en la cuenta de cheques, haciéndose el depósito en una Institución Financiera Internacional.

El asesor financiero (2):

- Solicita formatos y valida la documentación del cliente.
- Captura la apertura e imprime el contrato.
- Recaba firmas en papelería.

Los especialistas Internacionales (3):

- Captan los recursos.
- Verifican que estén los datos completos y correctos en la documentación de apertura.
- Distribuyen copia del contrato a asesores.
- Dan seguimiento a la apertura.
- Envían el expediente a Administración y Custodia.

- Una vez verificados los datos, envían el expediente del cliente, copia de los contratos y chequera al asesor financiero, para que este a su vez los entregue al cliente.

La mesa central (4):

- Asigna número de cuenta.
- Valida el depósito inicial.

Chequeras (5):

- Da de alta la carpeta de apertura.
- Envía la documentación y papelería completas a los especialistas internacionales.

La Administración y custodia de documentación (6):

- Verifica la documentación a detalle.
- Autoriza la apertura.
- Regresa copia del contrato.
- Digitaliza el contrato y la documentación.
- Envía una chequera provisional.
- Solicita la dotación de chequeras.
- Solicita la tarjeta de débito.
- Envía su dictamen al servicio jurídico.
- Envía copia de las tarjetas de firmas a la Institución Financiera Internacional.
- Envía las tarjetas de firmas al control de cuentas.

El Servicio Jurídico (7):

- Verifica poderes.
- Verifica actas constitutivas.
- Dictamina.

La Institución Financiera Internacional (8):

- Archiva copia de tarjetas firmadas.

Control de Cuentas (9):

- Digitaliza firmas.
- Microfilma las tarjetas de firmas.
- Distribuye las microfichas a los centros financieros.

3.5 Opciones de Solución

3.5.1 Sistemas Manejadores de Bases de Datos

El Sistema Manejador de Bases de Datos (SMBD) es la herramienta que nos permite trabajar con los datos y está compuesto por un conjunto de rutinas que nos permiten definir, crear, acceder, respaldar y administrar la base de datos, garantizando la seguridad y protección de los datos. Además, se encarga de sincronizar el acceso a múltiples aplicaciones. El SMBD también impone ciertas reglas para el modelado de las bases de datos y para el acceso de las mismas.

Un SMBD, realiza la función de interfaz entre los datos de bajo nivel almacenados en la base de datos y los distintos programas y consultas realizadas al sistema. Entre las facilidades del SMDB se tienen las siguientes.

- **Seguridad**

Un manejador de bases de datos debe proporcionar la seguridad en los datos. Esto es, que sólo tengan acceso a la información usuarios autorizados así como restringir la manipulación de dicha información a los usuarios no autorizados.

- **Protección**

Dado que los sistemas de cómputo están sujetos a posibles fallas (fallas eléctricas, de software, de los mecanismos del sistema de cómputo, etc.), es necesario tener algún método que detecte estas fallas y restaure los datos existentes hasta antes que se haya presentado la falla, lo cual se logra con procesos de respaldo y protección.

- **Integridad**

Esto se refiere a que los datos almacenados sean los correctos y que la base de datos sea consistente.

- **Concurrencia**

Como una base de datos puede ser consultada por varios usuarios a la vez, se debe tener un control que permita asegurar la interrelación entre los usuarios concurrentes.

Los sistemas manejadores de bases de datos actuales permiten el manejo de bases de datos basadas en el modelo relacional. En la actualidad existen más de cien productos relacionales que pueden correr sobre cualquier combinación de hardware y sistemas operativos.

3.5.1.1 Manejadores de Bases de Datos Relacionales (RDBMS)

El RDBMS contiene lo que se conoce como LDD (Lenguaje de Definición de Datos). El Administrador de la Base de Datos (ABD) es el encargado de utilizar este lenguaje para establecer la definición de las estructuras de las tablas y de las

columnas que componen estas tablas. Las funciones principales que permiten la definición de los datos son la creación y la eliminación de las tablas, vistas e índices de la base de datos.

Un RDBMS tiene la facilidad de crear vistas y un diccionario de datos en el momento en que el ABD crea la base de datos. El diccionario de datos guarda información que describe a la base de datos.

Una vez creada una base de datos es necesario tener una herramienta que nos permita actualizarla, para lo cual el RDBMS posee un Lenguaje de Manipulación de Datos (LMD).

Una de las funciones más importantes de un RDBMS es el controlar de una manera transparente la concurrencia. Dado que es muy importante que el estado de los datos sea siempre el mismo para todos los usuarios, el RDBMS emplea un mecanismo de bloqueo, que proporciona un control de la información cuando un usuario accede a determinados datos. Los mecanismos de bloqueo se pueden realizar por tablas, por registros o a una base de datos completa.

3.5.2 Análisis de los Posibles RDBMS a Utilizar

En este apartado se realizará una evaluación y comparación de las características de algunos de los principales Manejadores de Bases de Datos Relacionales (RDBMS) que se encuentran en el mercado. Se analizarán los siguientes RDBMS:

- ORACLE
- Sybase
- Progress

Se seleccionaron estos tres RDBMS debido a las siguientes razones:

- Las necesidades de la empresa y sus políticas de trabajo, como es el uso de aplicaciones con interfaces de usuario de tipo gráfico ("front-end").
- Los tres RDBMS soportan BLOB's ("binary large objects") y debido a que tiene un mejor rendimiento en el manejo masivo de información, tienen una alta seguridad en el manejo de la misma.
- Cuentan con:
 - Optimizadores de consulta
 - Manejo de "triggers" (disparos)
 - Vistas
 - Integridad declarativa y
 - Soporte para bases de datos distribuidas.

- Cubren las nuevas tendencias como son:
 - El soporte para Internet
 - Llamadas a procedimientos remotos (RPC's) para conectividad y distribución
 - Completo aprovechamiento de las plataformas con múltiples procesadores
 - Replicación de bases de datos

- Son los RDBMS que tiene la mayor participación en el mercado de las bases de datos "embedded" (incrustadas).

En términos generales, los RDBMS anteriores ofrecen:

- Portabilidad
- Compatibilidad
- Integración
- Arquitectura Cliente / Servidor
- Soporte SQL
- Desarrollo de interfaces gráficas.

A continuación se expondrán las características generales de cada uno de los RDBMS citados.

3.5.2.1 ORACLE

Es un RDBMS que soporta SQL ("Structured Query Language" o Lenguaje Estructurado de Consulta) para establecer comunicaciones con computadoras mayores. Entre sus muchas ventajas adicionales Oracle soporta múltiples sistemas operativos y tiene una gran capacidad en cuanto al manejo de aplicaciones grandes debido a su ejecución en memoria extendida.

Bases de Datos Grandes y Control de Espacios

ORACLE soporta potencialmente Bases de Datos mayores a cientos de gigabytes en tamaño. Para hacer un uso eficiente de los costosos dispositivos de hardware, permite el control total del espacio de almacenamiento.

Usuarios concurrentes en la Base de Datos

ORACLE, soporta un número muy grande de usuarios concurrentes, ejecutando una variedad de aplicaciones de Bases de Datos, operando sobre el mismo dato. Esto minimiza argumentos y garantiza la concurrencia de datos.

Alto Desarrollo en el Proceso de Transacciones

Mantiene características anteriores con un alto grado de desarrollo de sistemas completos.

Alta Disponibilidad

Trabaja 24 horas al día sin pérdidas de tiempo que limiten el rendimiento de la Base de Datos. La operación normal del sistema como Base de Datos está respaldada y las fallas parciales del sistema no interrumpen las consultas a la Base de Datos.

Disponibilidad Controlada

Puede controlar selectivamente la disponibilidad de los datos, según el nivel y subnivel de la Base de Datos.

Estándares Aceptados Industrialmente

Soporta los estándares aceptados industrialmente para el lenguaje de acceso de datos, sistemas operativos, interfaces de usuarios, y protocolos de redes de comunicaciones. Es un sistema "abierto" e incluye "banderas" para señalar el uso del SQL no estándar.

Manejo de Seguridad

Para proteger del uso y acceso no autorizado a las Bases de Datos, ORACLE está provisto de seguridad contra fallas, para limitar y monitorear el acceso a datos. Esta característica hace fácil el manejo hasta del diseño más complejo para el acceso de datos.

Integridad de las Bases de Datos

Refuerza la integridad de cada dato, siguiendo "reglas del negocio" que dictan el estándar para aceptar el dato.

Ambiente Cliente / Servidor (Proceso Distribuido)

Toda la responsabilidad de compartir el manejo de datos puede ser procesada por el manejador de sistemas de Bases de Datos, mientras las estaciones de trabajo corren su aplicación, concretándose a la interpretación y despliegue de los datos.

Sistema de Bases de Datos distribuidas

Para ambientes computacionales que están conectados vía red, ORACLE combina los datos de diferentes computadoras y localidades físicas dentro de una Base de Datos lógica, que puede ser accedida por todos los usuarios de red. El sistema distribuido tiene el mismo grado de transparencia y consistencia en los datos que un sistema no distribuido, teniendo incluso las mismas ventajas de un manejador de Base de Datos local.

Portabilidad

Las aplicaciones desarrolladas para ORACLE pueden ser transportadas a otro sistema operativo con muy pocas modificaciones o incluso con ninguna. Implementado en Lenguaje C, ORACLE, funciona sobre "mainframes" mini y microcomputadoras. Proporciona un entorno operativo de Base de Datos completo y consistente, e independiente de la máquina y sistema operativo sobre los que se está ejecutando.

Compatibilidad

El software de ORACLE es compatible con los estándares industriales, incluyendo los de los diversos sistemas operativos, como se menciono anteriormente. El desarrollo de aplicaciones para ORACLE puede ser usado en cualquier sistema virtual con pocas modificaciones o sin ninguna modificación, ya que sus interfases de usuario SQL son idénticas a las de SQL/DS y DB2.

Conectividad

ORACLE posee los productos de conectividad SQL, NET SQL y CONNECT para interactuar con diferentes plataformas.

Capacidad

ORACLE proporciona herramientas de desarrollo y de soporte para la toma de decisiones, como:

- Formateador de pantallas.
- Generador de informes.
- Generador de menús.
- Gráficos en color.
- Preparación de documentos.
- Hojas de cálculo
- Etc.

Replicación

ORACLE soporta la replicación a nivel tabla o pagina.

Facilidad de Implantación en Internet

Posee las herramientas para desarrollar en "WEB".

3.5.2.2 Sybase

El RDBMS de Sybase proporciona servicios cliente / servidor y para Internet. Sybase esta enfocado a cuatro grandes segmentos del mercado:

- Procesamiento de transacciones en línea.
- Data warehousing.
- Despliegue masivo.
- Comercio electrónico en línea.

Sybase proporciona productos y servicios bajo dos marcas. Con la marca Sybase se tienen productos enfocados a bases de datos escalables y soluciones de "middleware". Con la marca Powersoft existen los productos enfocados al mercado de las herramientas de desarrollo.

Bases de Datos "Enterprise" (Empresariales)

- Sybase SQL Server11. Es una base de datos que ofrece gran calidad, rapidez y capacidad de crecimiento, además de una mayor flexibilidad y rendimiento a una mayor cantidad de usuarios.
- Sybase MPP. Realiza la gestión de bases de datos relacionales en paralelo.

Bases de Datos "Workplace" (Departamentales)

- Sybase SQL Anywhere. Es un RDBMS escalable que se ejecuta en diversas plataformas PC y que, al mismo tiempo, satisface las exigencias de los entornos corporativos.
- Sybase SQL Server para Windows NT. Administra contenidos integrados para aplicaciones Web y cliente / servidor.

Productos de "Data Warehouse" (almacén de datos)

- Sybase IQ. Para aplicaciones que requieran velocidad de acceso muy alta, a fin de ejecutar consultas interactivas complejas.

Productos de Conectividad

Sybase tiene una gran variedad de productos en este rubro, como:

- Replication Server11. Aporta un entorno para la gestión y sincronización de datos y transacciones en emplazamientos distribuidos.
- Replication Agent. Trabajando conjuntamente con el Replication Server, el Replication Agent proporciona una replicación asíncrona para información externa a Sybase en fuente de datos de Oracle, DB2, IMS y VSAM.
- Sybase DBQ. Proporciona un sistema de "querying" (consulta) de bases de datos que integra aplicaciones distribuidas independientes con transacciones "fire and forget".
- Sybase Web SQL. Ofrece conexiones abiertas de alto rendimiento a bases de datos e incluye sentencias de SQL incluidas en una página HTML permitiendo la creación de contenidos dinámicos personalizados.
- Replication Toolkit para MVS. Permite desarrollar y ejecutar módulos MVS para el entorno del Replication Server, proporcionándoles un "log" (bitácora) común, un fichero de configuración y una interfase para la replicación.

- Replication Driver para ODBC. Es un componente del Replication Server que permite la replicación de transacciones para bases de datos "**desktop**" (de escritorio) soportadas por ODBC, como Lotus Notes, Microsoft Access y dBase 5.x.
- OnmiConnect. Ofrece acceso a más de 25 fuentes de datos con total transparencia de localización, fabricante y base de datos. Permite que los usuarios accedan a múltiples fuentes de datos como si todos los datos estuvieran almacenados en una única base de datos y les permite llevar a cabo enlaces heterogéneos entre mainframes, sistemas intermedios ("**midrange systems**") y plataformas cliente / servidor, e Internet / intranet.
- DirectConnect. Permite el acceso de lectura / escritura a datos de una única fuente sin importar su localización. Incluye acceso a datos MVS (con acceso a DB2 incluido), AS/400, Oracle e Informix. Ofrece movimientos bidireccionales de los datos entre mainframes y RDBMS en redes de área local, e incorpora herramientas de gestión de sistemas remotos y capacidades de motorización.
- Open ClientConnect. Permite a mainframes en entornos CICS, IMS/TM o MVS actuar como un cliente inicializador con servidores basados en LANs.
- Open ServerConnect. Permite que populares aplicaciones cliente / servidor consigan acceso de alto rendimiento a datos de mainframe y a aplicaciones de producción en línea en entornos CICS, IMS/TM o MVS. Permite que el mainframe actúe como servidor de datos / aplicaciones en entornos distribuidos.
- Open Client. Es un juego de aplicación cliente para la construcción de interfases a fuentes de datos Sybase, incluyendo SQL Server, Open server, Replication Server y OnmiConnect entre otras. Ofrece a los desarrolladores una serie de interfaces de programación basados en estándares que se ofrecen a través de una amplia variedad de plataformas de hardware.
- Open Server. Ofrece una potente y exclusiva API ("**Application Program Interface**" o Interfase de Programas de Aplicación) de servidor para sistemas de bases de datos distribuidas. Trabaja para integrar por completo cualquier cliente (CT-Lib, DB-Lib, ODBC, JDBC, Web), cualquier fuente de datos SQL o no SQL y cualquier servicio de aplicación, independientemente del entorno de hardware o Sistema Operativo.

3.5.2.3 Progress

Progress es un sistema de manejo de bases de datos relacionales que entre otras dispone de las siguientes características y facilidades.

- Cliente / Servidor
- Multi-plataforma
- Multi-base de datos
- Seguridad
- Copias incrementales
- Flexibilidad
- Alto rendimiento y escalamiento
- Multiprocesamiento simétrico masivo

- Portátil a través de Hardware y sistemas operativos
- Alta disponibilidad y excelente confiabilidad
- Soporta miles de usuarios concurrentes
- Soporte completo para ANSI SQL-92
- Integración con diferentes sistemas de administración de bases de datos diferentes a Progress
- Fácilmente adaptable.

Ya antes en las secciones 2.6 y 2.7 se presentaron ampliamente las características de Progress.

3.5.3 Evaluación

Cada uno de los productos revisados en esta sección tiene excelentes características en algunas áreas mientras que en otras no las tiene tanto. Lo importante al evaluar un RDBMS consiste en determinar que conjunto de características es importante para las aplicaciones que se van a desarrollar, ponderar estas características de acuerdo a su importancia y finalmente comparar los productos de acuerdo a los requerimientos.

A continuación se resumen en la tabla 3.5_1 las calificaciones de estos tres RDBMS obtenidas en una encuesta realizada por la revista "Computer World" de México a 270 usuarios. En la tabla 3.5_2 se dan las características técnicas de los tres RDBMS a evaluar.

PROVEEDOR DE BASE DE DATOS	ORACLE	PROGRESS	SYBASE
Eficiencia y recuperación de caídas	5	5	4
Capacidad de transacciones en línea	5	5	5
Integridad	4	5	5
Herramientas de programación	4	4	5
Administración del sistema	4	4	4
Herramientas del usuario	5	4	4
Seguridad del sistema operativo	4	5	4
Seguridad multi-nivel	5	5	4
Soporte SQL estándar	5	5	5
Aplicaciones de soporte en toma de decisiones	4	5	3
Funciones de auditoria	4	4	3
Facilidades de migración	4	5	4
Extensiones SQL	5	4	4
Soporte SQL estándar	5	5	5
Actualización, recuperación y administración remota	3	5	4
Integración de herramientas CASE	5	4	4

Tabla 3.5_1 Calificaciones de los tres RDBMS a Evaluar (5 es la más alta)

	ORACLE	SYBASE	PROGRESS
REQUERIMIENTOS DE HARDWARE	<p>Procesador Pentium o mayor. 32 MB en RAM (64 MB recomendados). De 25 a 50 MB en disco. Unidad CD-ROM.</p> <ul style="list-style-type: none"> -IBM RS/600 -SEQUENT -DYNIX/ptx -Sun -4/Sparc/Soulborne -Solaris -Apple Mac, A UX. -Bull DPX20 AIX v7 -Control Data Server 4000(CDC) -DEC Ultrix RISC -VAX MVS 	<p>De 72 MB a 111 MB en disco. Por lo menos 32MB en RAM (recomendable 64 MB).</p> <p>Sybase corre en las plataformas más utilizadas del mercado, tales como: IBM, HP, Intel, Digital, Sequent, Data General, AT&T, Silicon Graphics, Tandem, Unisys, Pyramid, Siemens y Stratus.</p>	<p>De 18 MB a 25 MB en disco. Procesador Pentium o mayor. Estaciones de trabajo basadas en la tecnología RISC o incluso equipos como la ALPHA 10000. De 500 KB a 4 MB de memoria para el usuario. Los procesadores pueden ser de tecnología Intel, RISC, Motorola, AS/400 y VAX.</p>
PLATAFORMAS QUE SOPORTA	<p>MACINTOSH OS/2 MS-DOS DG/UX Open VMS Ultrix HP-UX Solaris DYNIX SCO UNIX IBM VM/CMS/VSE/SP DEC ADABAS, EDMS, IMS, SQL/DS, VSAM Cualquier fuente de datos APPC AS/400 DRDA, DB2 SYBASE, RDB RMS, INFORMIX, CA-INGRES, SQL SERVER TERADATA</p>	<p>UNIX OS/2 IRIX MVS VSE /VMS DG/UX Solaris PC/UNIX MS-DOS Adabas AS/400,DB2 IDMS, IMS INFORMIX, INGRES ISAM MICROSOFT SQL SERVER ORACLE RMS, Archivos secuenciales SQL/DS Teradata VSAM</p>	<p>IBM AIX HP-UX NCROS Sun OS Sun Solaris. DEC Alpha OSF-1 DEC- Open VMS ULTRIX SCO-UNIX MS-DOS UNIX -V PTX AUX DG UX IRIX Hp 3000 Alpha Vax Hp 9000 AS/ 400 PC DIGITAL UNIX Motorola AIX NCR Intel UNIX SCO Unixware Sequent Sun (SPARC) Solaris Unisys UNIX ICL Intel Sequent PTX Siemens - RM</p>

Tabla 3.5_2 Características Técnicas de los RDBMS

	ORACLE	SYBASE	PROGRESS
SISTEMAS OPERATIVOS	Novell Windows NT UNIX OS/2	Nóvell Windows NT UNIX OS/2 VMS	Nóvell Windows NT SCO-UNIX Unixware AIX OS/2 IBM AS/400
INTERFASE GRAFICA DEL SISTEMA OPERATIVO	WINDOWS MOTIF MACINTOSH	WINDOWS MOTIF MACINTOSH OS/2	WINDOWS NT / INTEL 4.0 WINDOWS 95/98 MOTIF IBM-AIX SUN SOLARIS 2.6 HP-UX CITRIX X-WINDOWS Carácter UNIX
SISTEMA MANEJADOR DE LA BASE DE DATOS Y/O LENGUAJES	RDBMS SQL PL/SQL C FORTRAN COBOL C++ NET TCP/IP PL1	RDBMS ANSI SQL TRANSACTION SQL	RDBMS HLC ESQL/C SQL ODBC DDE C
HERRAMIENTAS PARA DESARROLLO DE APLICACIÓN	CDE2 TOOL ORACLE FORMS 4.5 ORACLE GRAPHICS 2.5 ORACLE DESIGNER ORACLE DEVELOPER ORACLE REPORTS PROAMMER/2000 SQL* PLUS ORACLE PROCEDURE BUILDER ORACLE EXPRES OBJECTS ORACLE MOBILE AGENTS DEVELOPER 'S KIT CASE * EXCHANGE	SYBASE SQL MANAGER SQL MONITOR POWER BUILDER SQL ANYWHERE Cuenta con más de 125 herramientas de desarrollo.	REPORT BUILDER PROGRESS RESULTS, DEVELOPER'S TOOLKIT, DATABASE ADMINISTRADOR DE BASE DE DATOS, ADMINISTRADOR DE TRADUCCIÓN, TRADUCTOR VISUAL HERRAMIENTAS PARA EL WEB(WEBSPED)
MODULOS DE CONECTIVIDAD PREESTABLECIDOS	DB2 SQL RDB SERVER DE MICROSOFT	Sybase tiene más de mil asociados en aplicaciones y herramientas de desarrollo; para una gran variedad de mercados.	ORACLE, RMS EDB-VMS C-ISAM, CT-ISAM DB2 Object Access, Object Store, Sybase

Tabla 3.5_2 Características Técnicas de los RDBMS (Cont.)

Por último, en la tabla 3.5_3 se exponen las características funcionales de los tres RDBMS a evaluar.

	ORACLE	SYBASE	PROGRESS
Modelo de Datos Relacional			
Dominios	NO	NO	No
Opciones de violación de integridad referencial	Restringidas, excepto el borrado en cascada	Sólo restringidas	Restringidas
Personalizar mensajes referenciales	No	No	No
Cláusulas WHERE referenciales	No	No	No
Vistas actualizables	Sí	Sí	Sí
Objetos de bases de datos			
Tipos de datos definidos por el usuario	Sí	Sí	Sí
BLOB's			
Tipos de datos soportados	Imágenes, video, texto, mensajes, tipos de datos espaciales	Binarios, imágenes, monedas, bit, texto, variables binarias	Carácter, Fecha, Decimal, Número, Entero, Lógica, Image
Estructuras de tablas	"Heap" y "clustered"	"Heap" y "clustered"	"Heap" y "clustered"
Estructuras de índices	B-tree, bitmap y hash	B-tree	B-tree y Hash
Facilidades de optimización ("tuning")	Distribución de índices y tablas	Índices, prefetch, I/O buffer cache, tamaño de bloque, partición de tablas	Distribución de índices y tablas
Nivel de "Triggers" (disparos)	Basado en registro y basado en conjuntos de registros	Basado en conjuntos de registros	Basado en registro y basado en conjuntos de registros
Ejecución del trigger	Antes y después de la operación que disparo el trigger	Después de la operación que disparo el trigger	Antes y después de la operación que disparo el trigger
Anidamiento de triggers	Sí	Sí	Sí
Lenguaje empleado en procedimientos almacenados	PL/SQL	TRANSACT-SQL	SQL, PROGRESS 4GL
Anidamiento de procedimientos almacenados	Sí	Sí	Sí
Cursores en procedimientos almacenados	Sí	Sí	Sí
Llamadas externas de procedimientos almacenados	RPC	RPC	RPC
Eventos en procedimientos almacenados	Sí	Controlados por tiempo	Sí
Consultas			
Nivel de bloqueo	Tabla registro	Tabla, página	Tabla, registro

Tabla 3.5_3 Características Funcionales de los RDBMS

	ORACLE	SYBASE	PROGRESS
Cumplimiento del estándar ANSI SQL	SQL-92	SQL-92	SQL-92
Cursores	Hacia adelante	Hacia adelante	Hacia adelante
Outer joins			
Sintaxis ANSI de outer joins	No	No	Si
API's	ODBC	DBLIB, CTLIB, ODBC	ANSI SQL-92, 4GL, ODBC 3.0, JDBC, SQLIC incrustado
Administración de Bases de Datos			
Soporte a SNMP	Si	Si	SMP, ccNUMA, VLM y Clusters
Seguridad	C2	C2	C2
Respaldos parciales y recuperación	Configurable	Configurable	Configurable y en línea
Internet			
Soporte a Internet	Oracle WebServer	Web SQL	WebSpeed
Conectividad, Distribución y Middleware			
Bases de Datos Distribuidas	Incluido como parte del producto	OmniConnect	Incluido como parte del producto
Protocolo 2PC	Si	Si	Si
Esquema heterogéneo de Bases de Datos	A través de gateways	A través de DirectConnect	A través de gateways
Optimización	Si	Si	Si
RPC	Si	Si	Si
Replicación			
Registro	Replicación de Log y de Triggers	Empleando el Log	Replicación de Log y de Triggers
Hot-standby	Si	Si	Si
Punto a Punto	Si	Si	Si
Replicación hacia otros RDBMS	A través de gateway	A través de DirectConnect	A través de gateway
Replicación en cascada	Si	Si	Si
Otras características adicionales			
Longitud de nombres de objetos	30	30	30
Número máximo de columnas	254	250	32000
Tamaño de las columnas	2 GB	1962 MB	
Número de tablas	N/D	2 billones	32000
Tamaño de las tablas	N/D	Depende del espacio en el disco	4 GB

Tabla 3.5_3 Características Funcionales de los RDBMS (Cont.)

3.5.4 Selección del RDBMS

Haciendo referencia a la tabla 3.5_2, se observa lo siguiente.

- Requerimientos de hardware: Se observa que ORACLE tiene requerimientos mayores y más costosos que PROGRESS o SYBASE.
- Clientes que soporta: En este punto el que mayor número de clientes soporta es PROGRESS.
- Sistema operativo: Nuevamente PROGRESS es el que maneja un mayor número de sistemas operativo en red.
- Interface gráfica del sistema operativo en red: PROGRESS lleva la vanguardia.
- Sistema manejador de la base de datos y/o lenguajes: En este rublo ORACLE maneja más lenguajes de desarrollo.
- Herramientas para desarrollar la aplicación: SYBASE es el que tiene más herramientas de desarrollo.
- Módulos de conectividad preestablecidos: El manejador que mayor número de módulos de conectividad maneja es PROGRESS.

Se seleccionó PROGRESS como el RDBMS a utilizar en el desarrollo del proyecto con base en las características técnicas y funcionales mostradas en las tablas 3.5_1 a 3.5_3. A continuación se resumen sus principales características técnicas y facilidades.

- Progress realiza procesamiento de transacciones en línea (OLPT) utilizando SQL y Progress 4GL. Posee las capacidades de bloqueo en el nivel de registro, recuperación "roll-back" y "roll-forward", recuperación "point-in-time", administración de la base de datos en línea, soporte completo para el estándar ANSI SQL-92 y una interfase gráfica para la administración del sistema.
- Es el que tiene los más bajos costos de administración, un bajo costo inicial por licencias y costos mínimos por actualización. Proporciona una ventaja significativa en el costo de propiedad - aproximadamente de 4 a 1 - sobre los otros productos de bases de datos analizados.
- Es altamente, escalable.
- Soporta configuraciones de hasta 32 procesadores SMP, ccNUMA, VLM y Clusters
- Tiene una arquitectura de alto rendimiento multi-enlazado y múltiples servidores por cada base de datos (hasta 256).
- Soporta múltiples sistemas operativos y plataformas de hardware.
- Tiene APIs para acceso de clientes SQL.
- Soporta 10,000 usuarios concurrentes.
- En su conjunto de "buffers" maneja hasta 125 millones de buffers o 128 GB.
- Bloqueo en el nivel de registro para reducir contención de los recursos y así maximizar el rendimiento.
- Registro Optimo de Transacciones.
- Procesos del servidor auto-ajustables.
- Asignación de un conjunto dinámico de buffers para proceso en lotes y usuarios interactivos.
- Recuperación automática de caídas.

- Registro "Roll-back" y "roll-forward".
- Respaldo en línea completo y en aumento.
- Reorganización de tablas e índices en línea.
- "Two-phase commit". Protocolo de dos fases para asegurar la integridad y consistencia de las transacciones.
- Recuperación "Point- in- time". Actualización de la base de datos desde un punto determinado.
- Soporta ActiveX / OCX.
- Soporta datos del tipo carácter, fecha, decimal, número entero y lógicos.
- Permite hasta 32,000 tablas.
- Soporta 32,000 índices.
- Soporta hasta 32,000 campos por tabla.
- Tamaño del registro de 32,000 bytes.
- El ambiente de Progress usa un conjunto de herramientas de desarrollo y gráficas que soportan tanto los modelos estructurales como los orientados a objetos.
- Progress es la base de datos "embedded" (incrustada) número 1 en aplicaciones comerciales.

Deben agregarse factores adicionales en la evaluación, tales como costos. En la tabla 3.5_4 se presentan los resultados de un estudio realizado por Aberdeen Group, en el que se compara Oracle con Progress.

Costos Asociados con la Propiedad de la Base de Datos y su Uso	Oracle Corp. Oracle 7.3 Workgroup Server	Progress Software Progress Workgroup
Licencias de la base de datos	\$2,950	\$1,800
Herramientas del cliente	\$3,995	\$1,800
Red e integración del cliente	\$990	\$1,150
Herramientas de monitoreo	Incluidas	Incluidas
Cargos de implementación	\$35,000-\$100,000	\$10,000-\$15,000
Costos de mantenimiento interno	\$126,000 (5 años)	\$13,165 (5 años)
Costos de entrenamiento	\$15,000	\$4,000
Actualizaciones	\$11,112 (5 años)	\$7,600 (5 años)
Costo total de la base de datos para cinco años	\$160,047	\$29,515

**Tabla 3.5_4 Oracle Workgroup server y Progress Workgroup.
Costos para 10 usuarios en 5 años (en Dólares)**

La tabla 3.5_4 fundamenta la elección de Progress, ya que la diferencia en costos es muy contrastante. Además, la empresa ya posee el producto y las licencias del producto Progress, lo cual es una razón adicional muy importante para su elección en este proyecto.

CAPITULO 4

DESARROLLO E

IMPLEMENTACIÓN DEL SISTEMA

4.1 Diseño de la Aplicación

4.1.1 Diseño Preliminar

El diseño final de la aplicación al cual se llegó con los usuarios posee las siguientes opciones principales.

1. Diario: En esta opción es donde se llevan a cabo todos los procesos que diariamente se realizan en la institución financiera.
2. Mensual: Son todos los procesos que se corren a fin de mes para el pago de intereses de los clientes así como la generación de los estados de cuenta de los mismos.
3. Cuentas: En los submenús que posee esta opción es donde se realizan los chequeos de las cuentas concentradoras y demás cuentas importantes para la contabilidad de la institución.
4. Parámetros de modificaciones para el control de procesos: En esta opción se capturan o modifican ciertos parámetros que son necesarios para el cálculo de interés y demás comisiones que realice la institución a sus diferentes cuentas.
5. Memos: Son documentos que se generan para las diferentes áreas de la institución.
6. Chemical: En esta opción se lleva a cabo el monitoreo de las cuentas de Chemical Bank y Barclays Bank
7. Pagos: Esta opción contempla todos los pagos que se tengan que realizar por los diferentes servicios con que cuenta la institución.

Cada una de estas opciones que integran al sistema contienen a su vez varias sub-opciones que son las siguientes.

CAPITULO 4. DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA

En el Manejo de operaciones diarias se tienen como procesos:

- Movimientos Barclays Bank.
- Otros Movimientos.
- Cancelación y Baja de Movimientos.
- Listados de Sobregiros.
- Reporte de Movimientos por Tipo.
- Consultas de Saldos.
- Instrucciones de Desembolso.
- Reporte de Estado de Cuenta Global.
- Pago Diario de Cheques.
- Consulta de Movimientos Automáticos.
- Alta de Remesas.
- Liberación de Remesas
- Listados de las Remesas.

En la opción "Mensual" se tienen como sub-opciones:

- Listado de Saldos en Cuentas.
- Movimientos de Cierre de Mes.
- Cancelación Masiva de Movimientos.
- Emisión de Estado Cuenta.
- Listado de Ingresos y Gastos.
- Cierre de Mes.
- Listado de Perfil de la Clientela.
- Contabilizar Movimientos.
- Listado para Cuadrar Contabilidad.
- Total de Débitos / Créditos por Rango.

La opción Cuentas contiene las siguientes sub-opciones:

- ABC (altas, bajas y cambios) de cuentas.
- Estadísticas por cliente.
- Congelar Cheques.
- Congelar Cuentas.
- Registro y Consulta de Chequeras.
- Listado de Cuentas y Cheques Congelados.
- Catalogo de Cuentas.

- Listado de Estatus de Cuenta (por Fecha).
- Listado de Estatus de Cuenta (por Fecha, Plaza y Asesor).

La opción "Parámetros" contienen las siguientes sub-opciones:

- Parámetros Generales.
- ABC de Tipos de Movimientos.
- ABC de OverNight de Barclays.
- Chequera Congelada.
- Total de Saldo en Chequera.
- Alta de Instrumentos de Desembolso.
- Listado de Movimientos Automáticos.
- Listado de Cheques por Centro Financiero.
- Total de Pagos y Movimientos.
- Listado de Saldos por Rango.

La opción "Memos" contiene las siguientes sub-opciones:

- Test de Barclays.
- Acuse de Chequeras Solicitadas.
- Fondo de Casa de Cambio.
- Autorización de Cheques.
- Resumen de Casa de Cambio.
- Cheques Cobrados por Centro Regional.
- Acuse de Recibo de Cheque.
- Carta de Cheques Rebotados.
- Cheques Cobrados en Centros Financieros.
- Diferencia de Saldos de Cheques.
- Depósito en Barclays.
- Depósito en Chemical.

En la opción "Chemical" se tienen las siguientes sub-opciones:

- Movimientos de Cuentas de Chemical.
- Movimientos Globales de Chemical.
- Cancelación de Movimientos de Clientes de Chemical.
- Cancelación Global de Movimientos de Chemical.
- Consulta de Saldos de Chemical.

CAPITULO 4. DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA

- Listado de Movimientos de Chemical.
- Memo de Traspaso de Chemical.
- Contabilizar Movimientos de Chemical.
- Des-contabilizar Movimientos de Chemical.
- Actualización Saldo de Chemical.

La opción "Pagos" contiene las siguientes sub-opciones:

- ABC de Área, División, Plaza y Sucursal.
- ABC de Promotores.
- ABC de País.
- ABC de días Inhábiles.
- ABC de Bancos.
- ABC de Centros Financieros.
- ABC de Asesores.
- ABC de Preformatos.
- Generación de Cuentas De Clientes.
- Listado de Cheques Cobrados.
- Validación de Pagos.
- Solicitud de Chequera Provisional.
- Test-Key.
- Eliminación de Test-Key.

4.1.2 Diagrama de Contexto y de Flujo de datos.

Para el diseño de un sistema de información es necesario establecer el curso que siguen los Datos de ciertos puntos de proceso, mismos en los que se puede dividir el sistema. Estos puntos de proceso pueden ser tantos como el nivel de detalle del flujo de datos que se quiera alcanzar.

Para cada proceso que es alimentado de datos se realiza una función determinada que generará otros datos, que a su vez pueden ser la entrada para otros procesos del sistema.

El diagrama de flujo de datos expone la relación entre los procesos que intervienen en el sistema. Todas las posibles fuentes de datos externas deben ser consideradas en este diagrama, ya que son las que determinan en gran medida el comportamiento de los procesos que tienen el primer contacto con ellas. También deben incluirse los datos que produzca el sistema en el nivel global, ya que pueden tomarse como datos de entrada para otro sistema.

Las necesidades especificadas por la Institución indican que requiere de un sistema para apertura de cuenta de cheques, la cual debe iniciarse con un depósito y una administración de la cuenta, como se muestra en la figura 4.1_1. En las siguientes páginas se muestran los diagramas de flujo de datos de:

- Nivel 0 (diagrama de contexto)
- Nivel 1 para todos los procesos del Nivel 0
- Nivel 2 para todos los procesos del Nivel 1
- Nivel 3 para los subprocesos 5.3 a 5.8, 5.10 y 5.11, pertenecientes al proceso No. 5 (administración de la chequera).

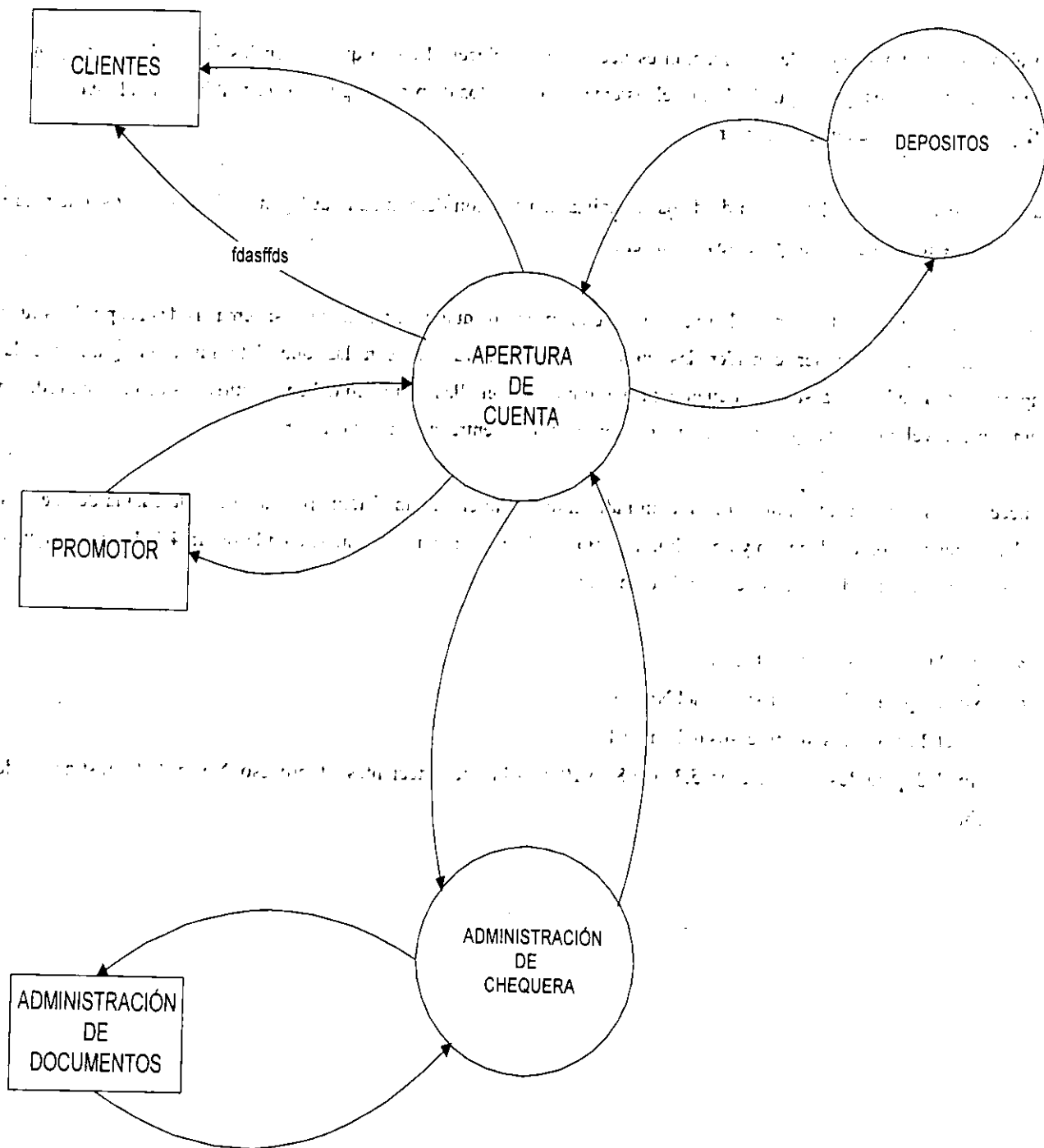


Figura 4.1_1 Diagrama de contexto de Nivel 0

El área de chequeras es la parte fundamental de nuestro sistema, desde el momento de promocionar la cuenta, hacer el depósito de apertura, dar de alta la cuenta y administrar la chequera, se está creando el directorio que nos servirá para cada proceso del sistema, el cual también podrá ser utilizado por diferentes áreas de la Institución. Este proceso se muestra en la figura 4.1_2.

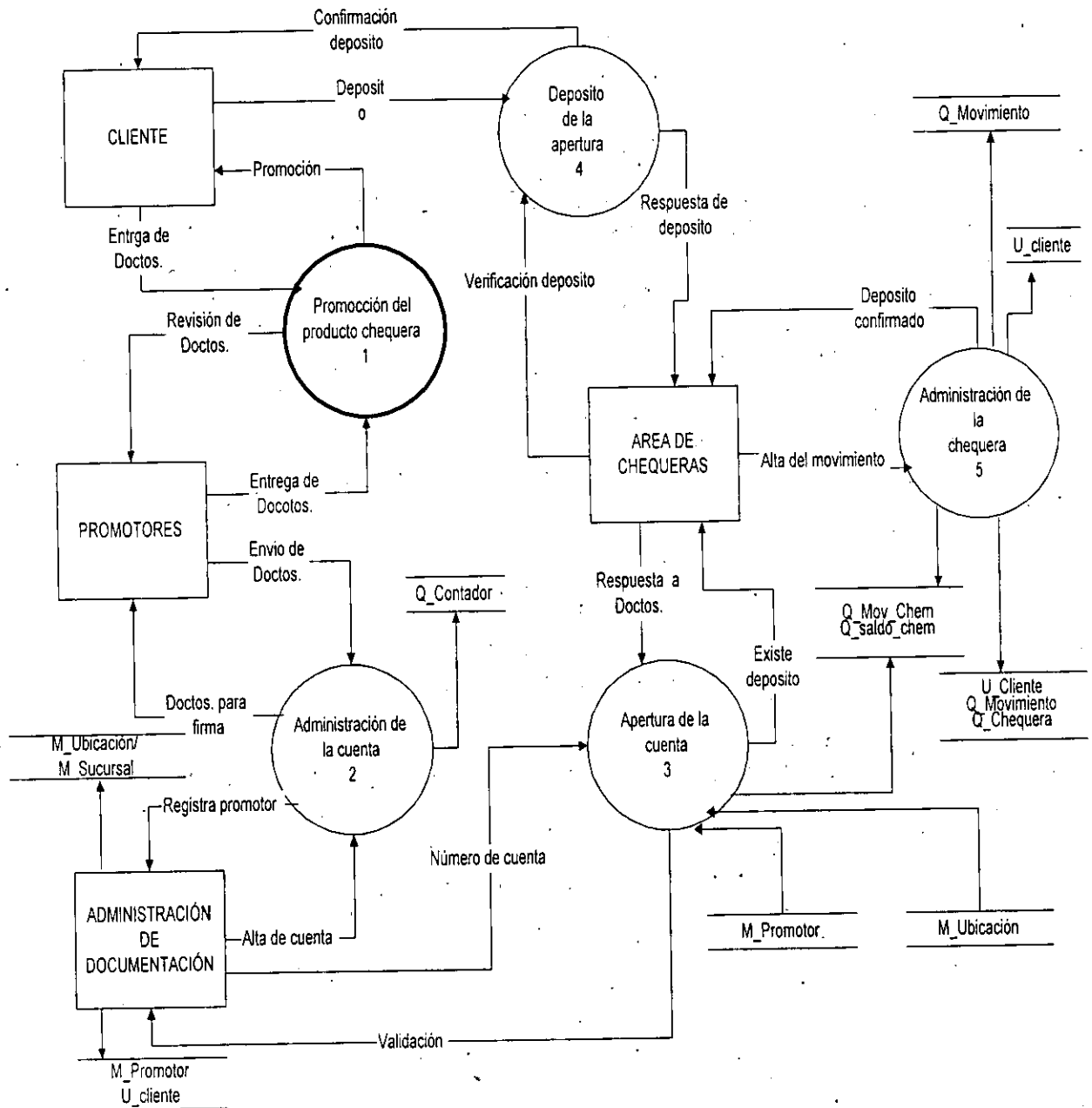


Figura 4.1_2 Diagrama de Flujo de Nivel 1

Promoción del Producto Chequera (1.0)

Al promocionar el producto chequera, se crean las tablas y los campos que se muestran en la figura 4.1_3, enlazándose con los procesos del área de administración de cuentas y de promoción de cuentas.

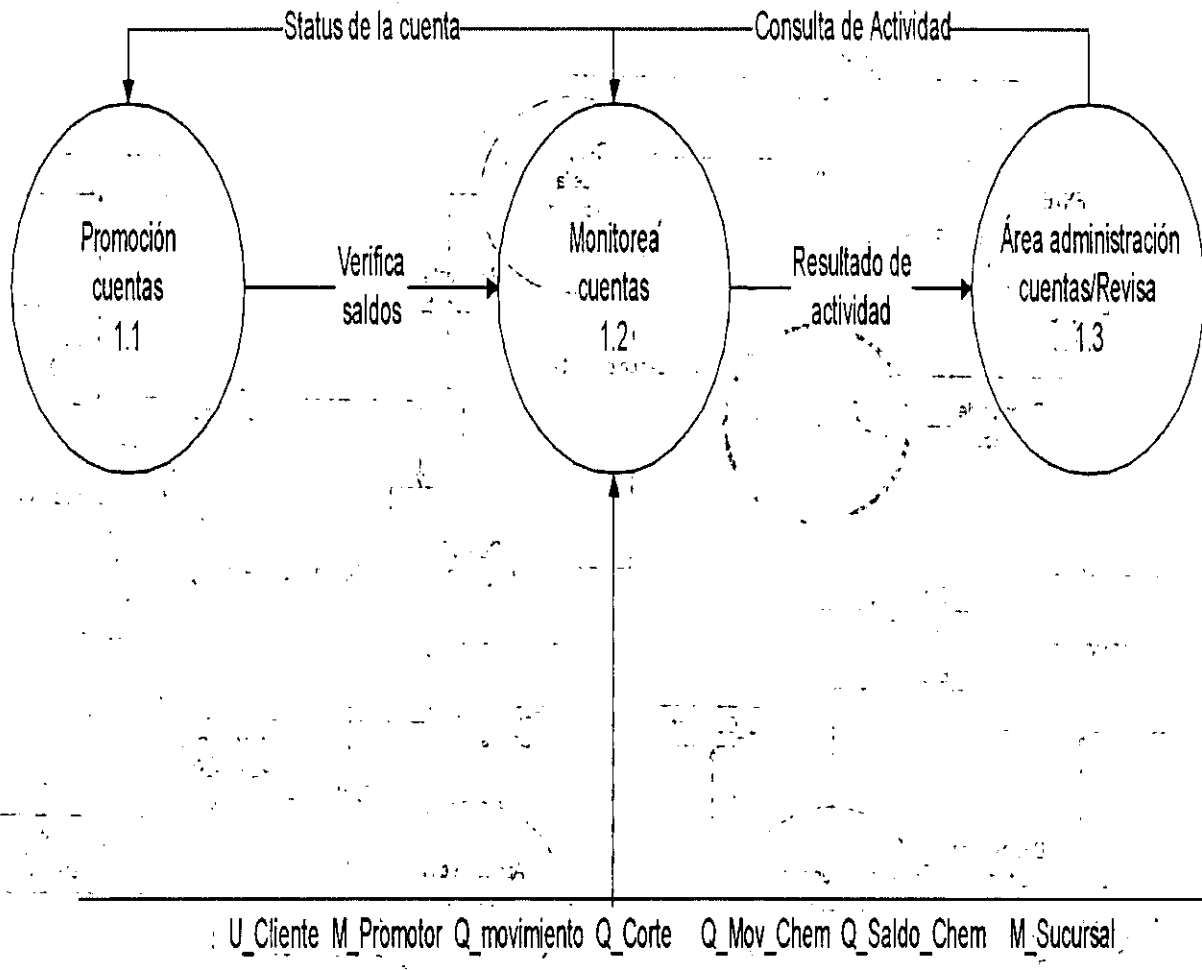


Figura 4.1_3 Diagrama de flujo de nivel 2

Administración de la Cuenta (2.0).

La administración de cuenta es una parte importante del menú de nuestro sistema, ya que válida la documentación. Observamos que en el proceso de alta de promotores donde es requerido nuestro sistema, valida los campos de referencia (ver la figura 4.1_4).

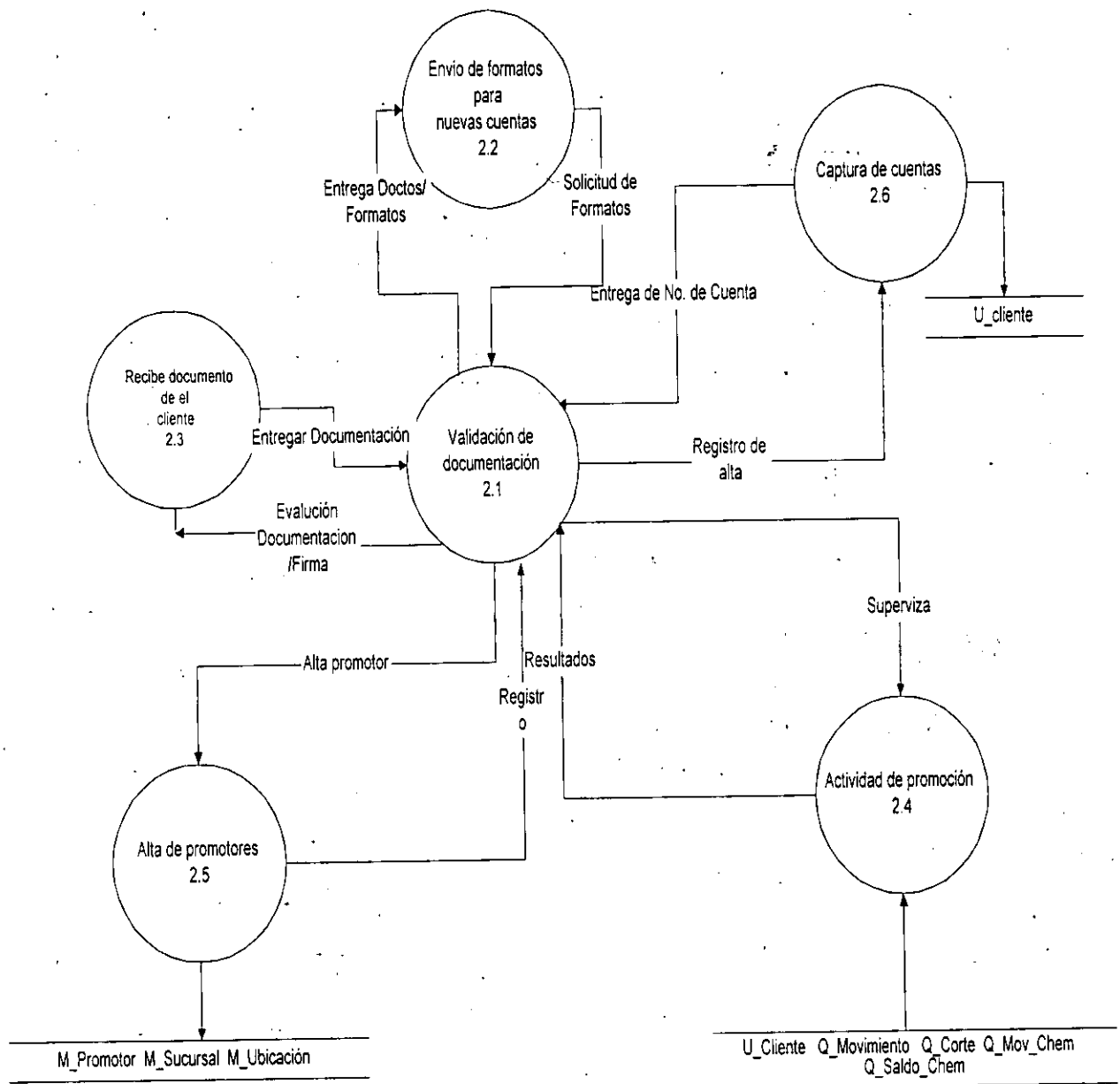
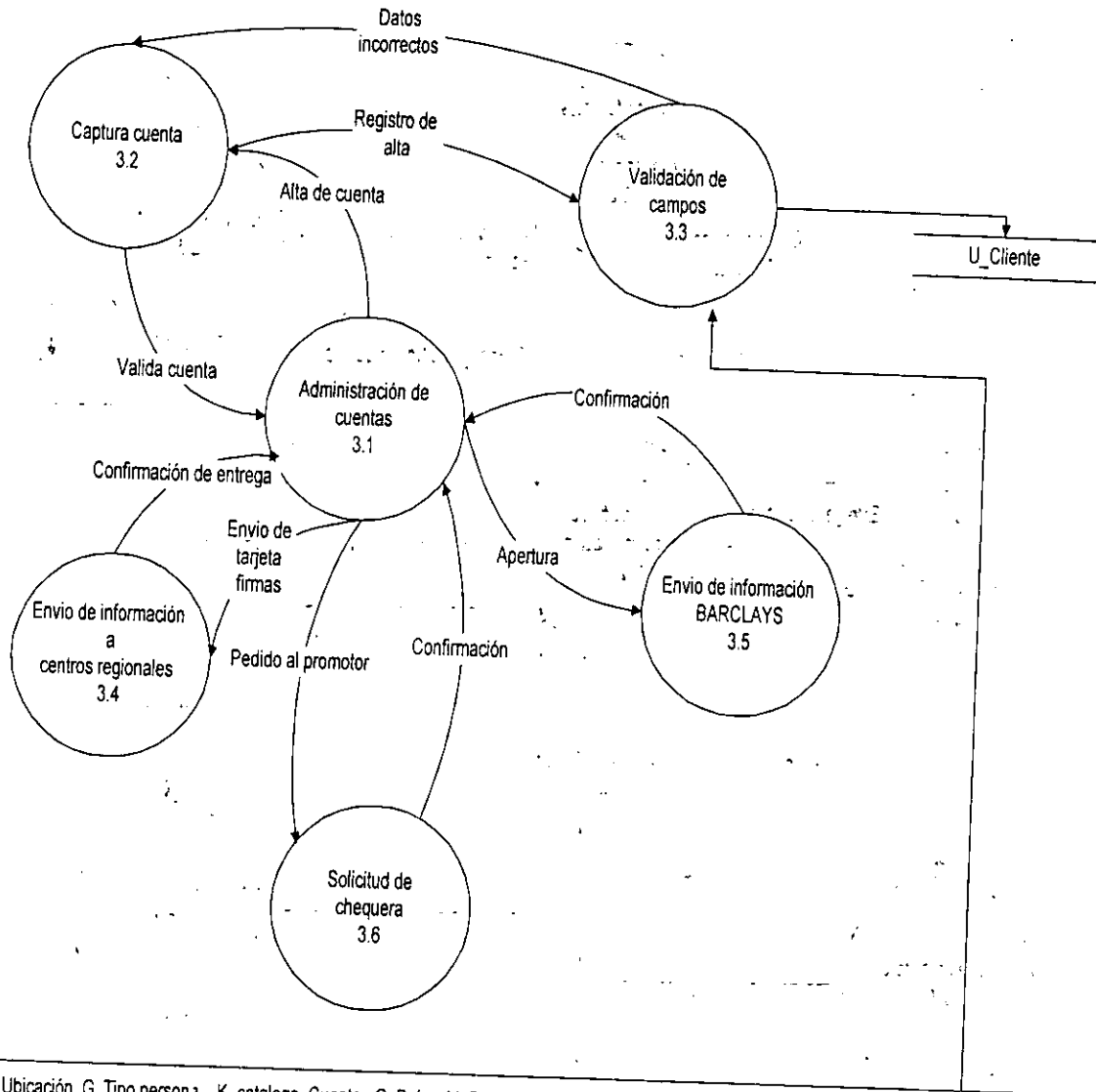


Figura 4.1_4 Diagrama de Flujo de Nivel 2

Apertura de Cuenta (3.0)

La apertura de cuenta de cheques se da al concluir la validación de datos y documentos del cliente, enviando información a los bancos extranjeros como se muestra en la figura 4.1_5.



M_Ubicación G_Tipo persona K_catalogo_Cuenta G_País M_Promotor M_Ubicación G_País C_Segmento U_Cod_dist G_Estado S_Usuario

Figura 4.1_5 Diagrama de Flujo de nivel 2

Deposito de la apertura (4.0)

La figura 4.1_6 muestra el procedimiento del deposito de apertura de cuentas.

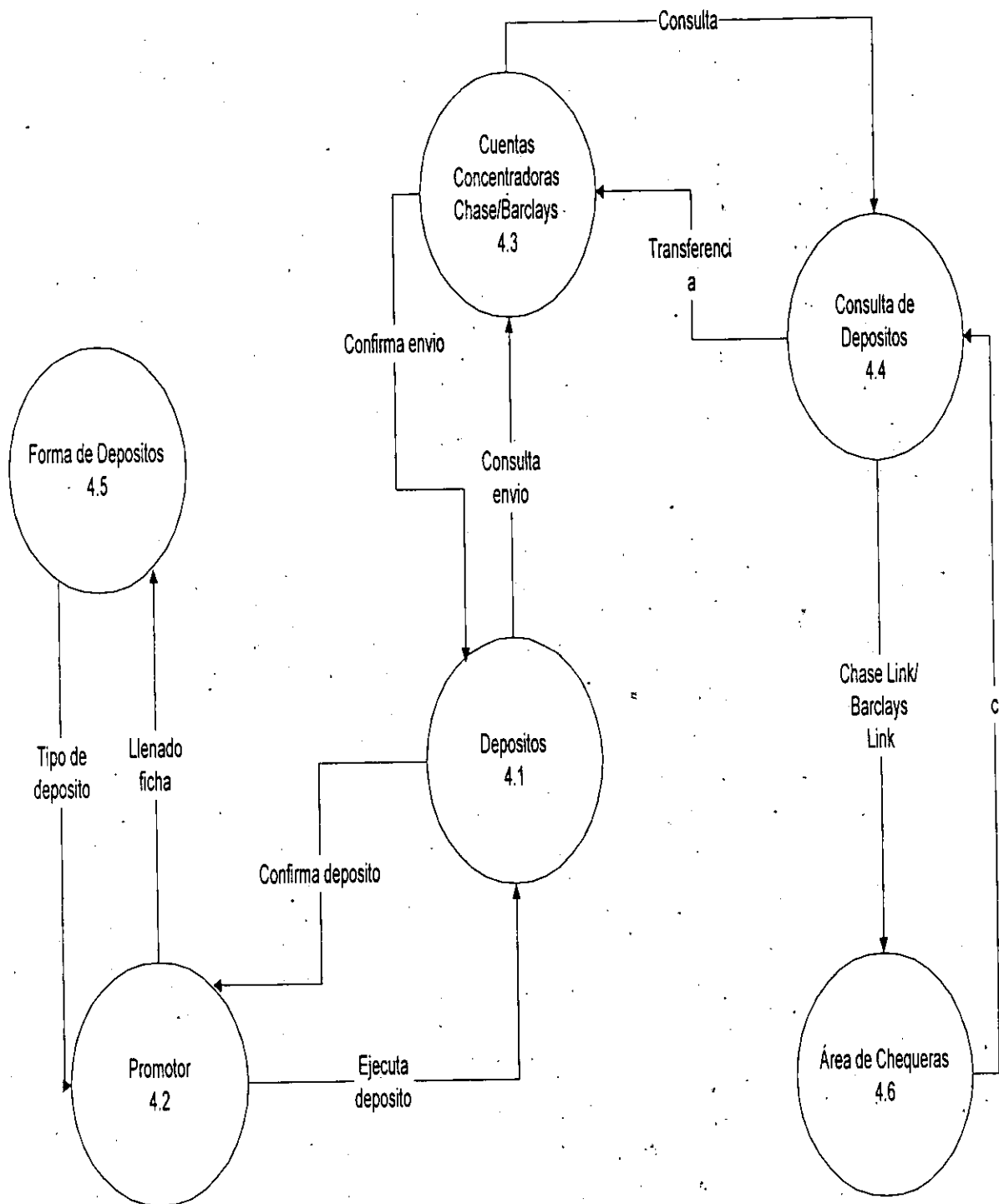


Figura 4.1_6 Diagrama de flujo de nivel 2

Administración de la chequera (5.0)

La figura 4.1_7 muestra la administración de la chequera. Es el proceso central del sistema, ya que encontramos el ABC (Altas, Bajas y Cambios) del cliente y de los movimientos realizados, como son: las transferencias, pagos y liberaciones de dinero, así como los movimientos hechos con otros bancos.

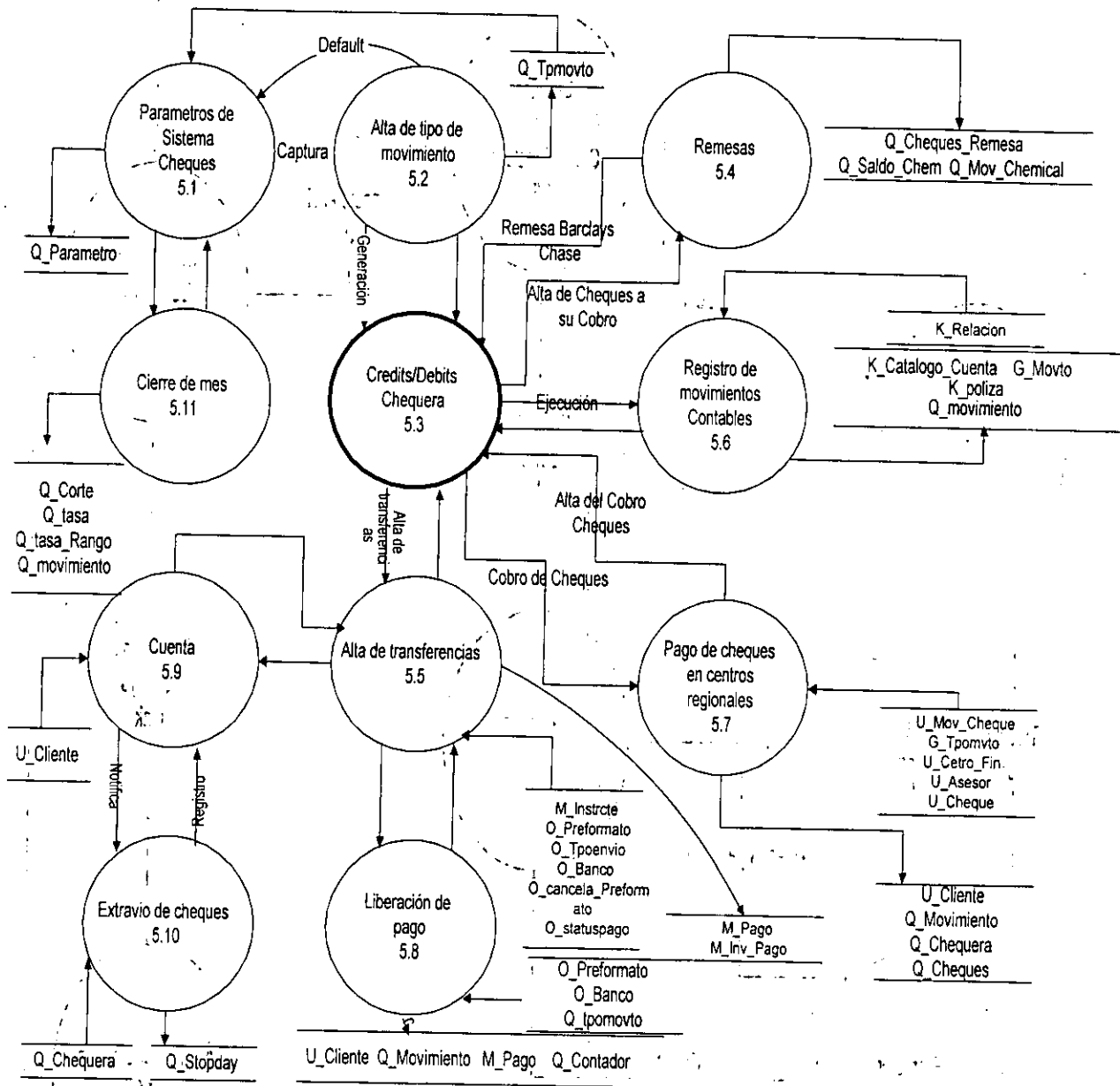


Figura 4.1_7 Diagrama de flujo de nivel 2

Credits/Debits (Créditos / Débitos) de la chequera (5.3)

En este proceso se deben capturar todos los tipos de movimientos efectuados a través de la cuenta de chequeras, como son movimientos con Chemical Bank, depósitos, transferencias, cobro por manejo de cuentas, movimientos de estados de cuenta, etc. Este proceso se muestra en la figura 4.1_8.

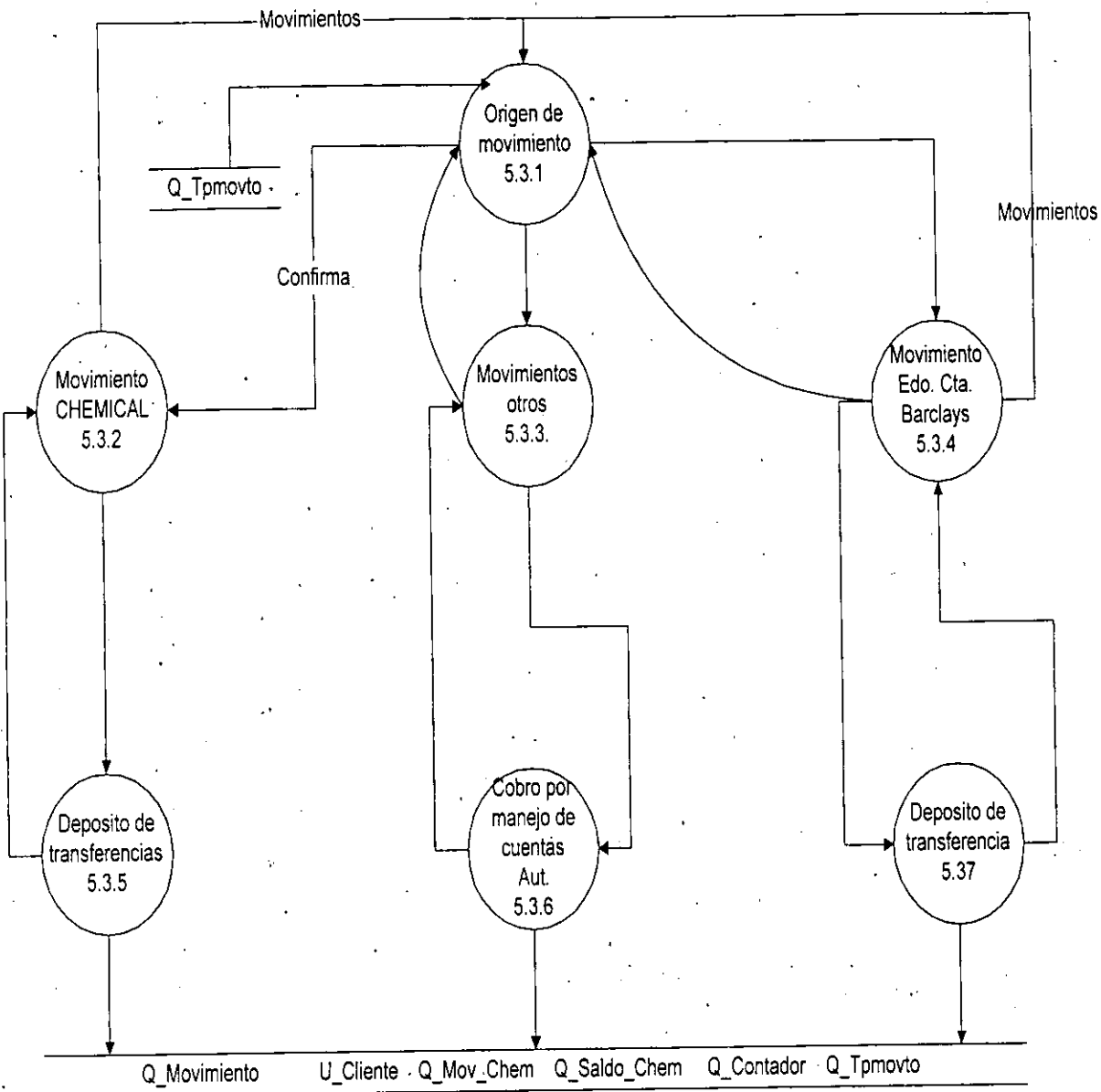


Figura 4.1_8 Diagrama de flujo de nivel 3

Remesas (Cobro de cheques de otros bancos a cuentas del cliente) (5.4)

El control de remesas, crea nuevas tablas, este proceso se observa en la figura 4.1_9. En ésta se controlan las precedencias de los cheques y los diferentes movimientos referentes a dichos cheques.

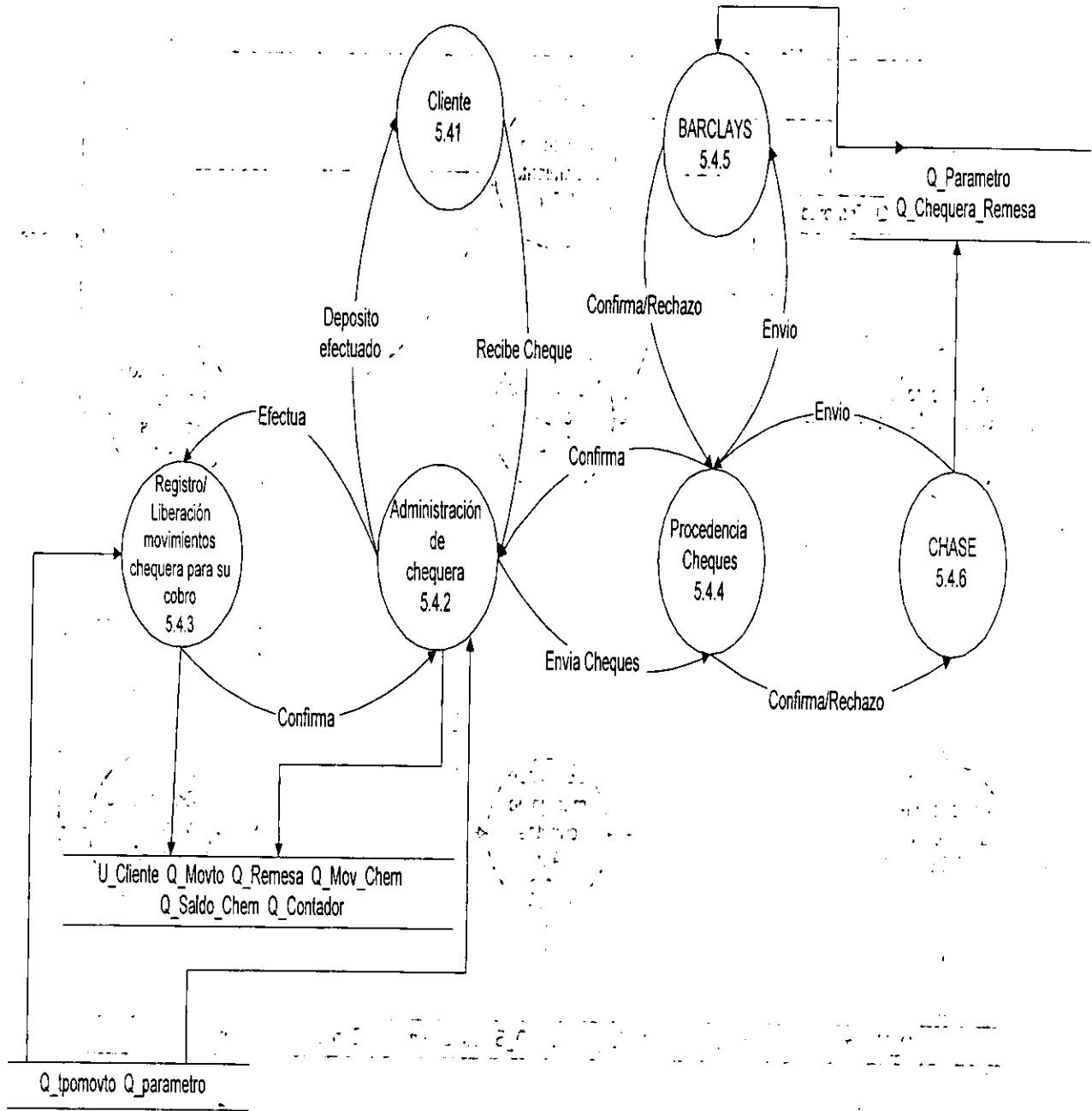


Figura 4.1_9 Diagrama de flujo de nivel 3

Alta de transferencias (5.5)

El proceso de alta de transferencias se muestra en la figura 4.1_10. Este procedimiento registra y capta todo tipo de movimiento de transferencia de cuentas.

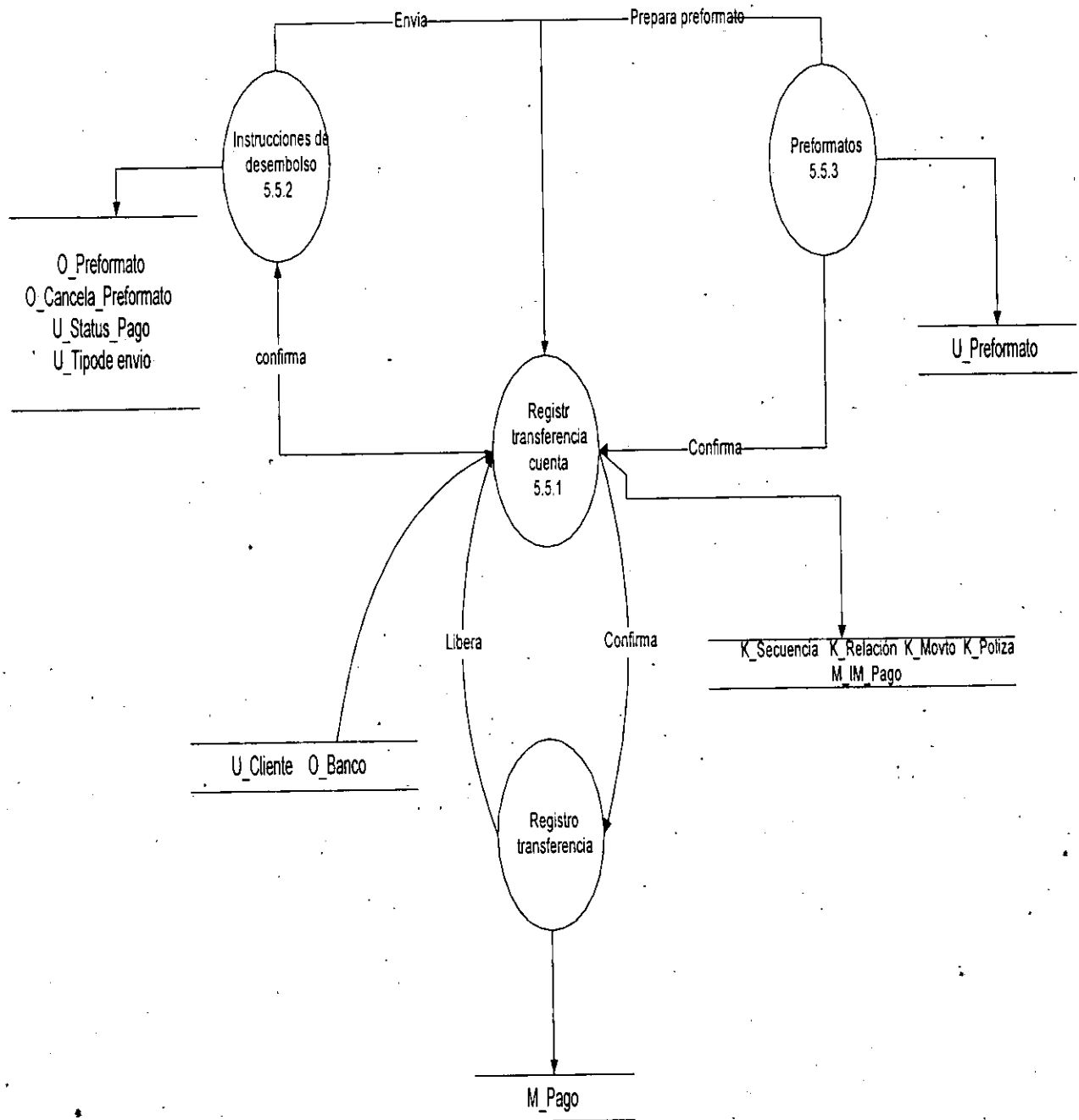


Figura 4.1_10 Diagrama de flujo de nivel 3

Registro de movimientos Contables (5.6)

El departamento de contabilidad se apoya de éste sistema para la generación de su reporte de conciliación y cierre del día.
 Figura 4.1_11.

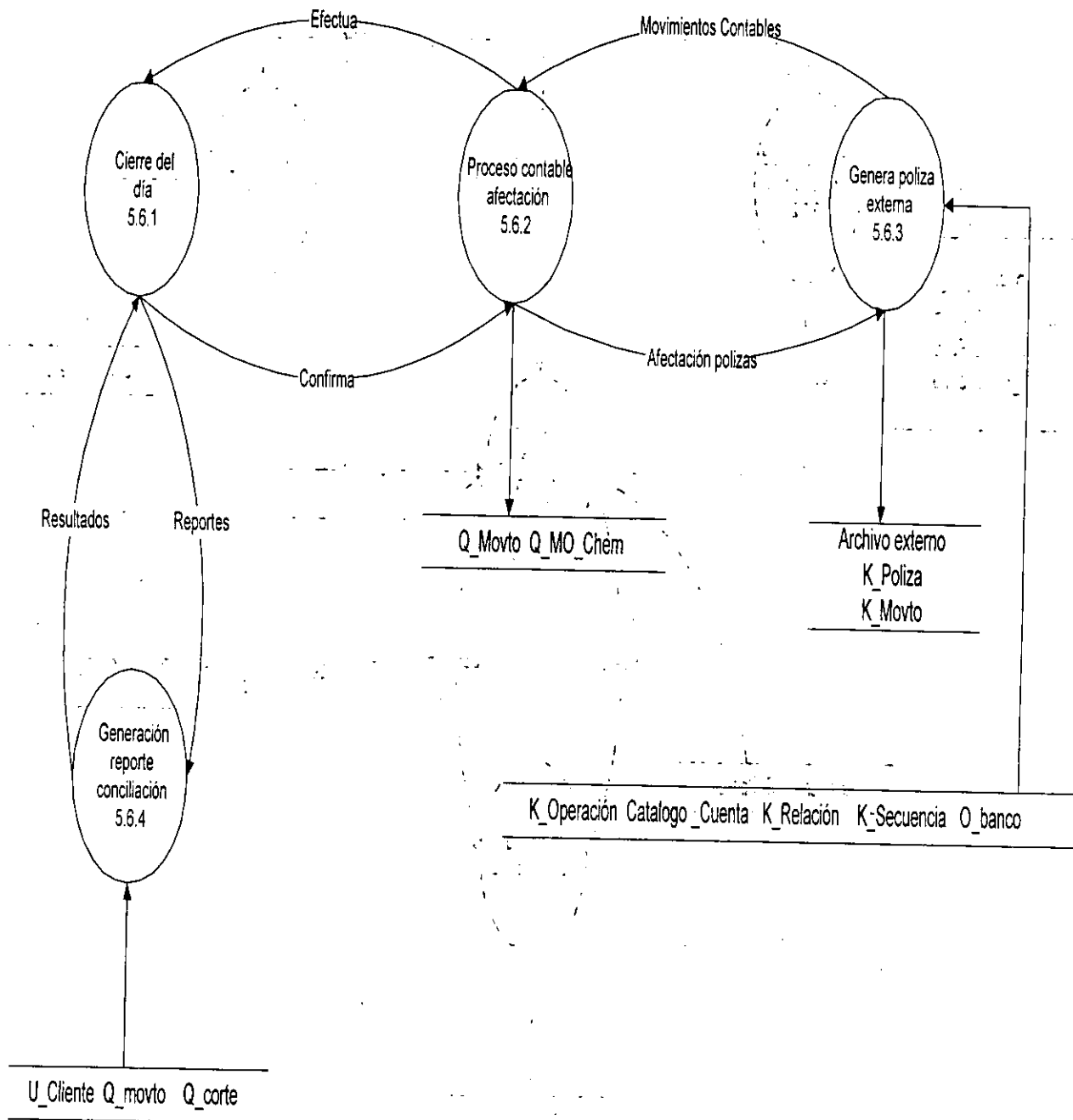


Figura 4.1_11 Diagrama de flujo de nivel 3

Pago de Cheques en centros regionales (5.7)

En la figura 4.1_12 se muestra el procedimiento para el pago de cheques de los centros regionales, en el cual se pueden hacer consultas mediante el sistema, para validar los cheques.

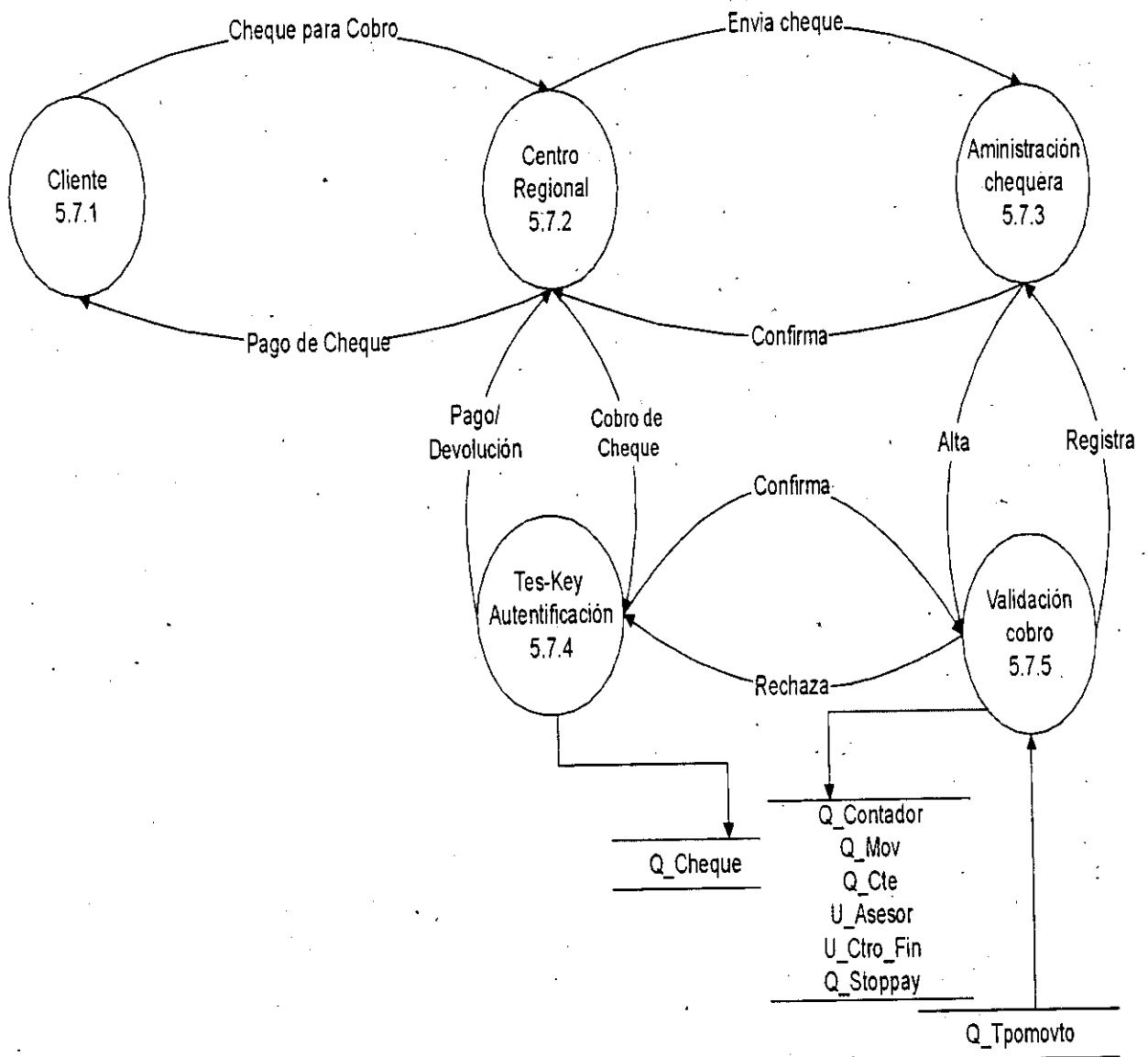


Figura 4.1_12 Diagrama de flujo nivel 3

Las figuras 4.1_13, 4.1_14 y 4.1_15 muestran los procesos, de liberación de pago, extravío de chequeras y cierre del mes, en donde observamos los diferentes campos generados y consultados.

Liberación del Pago (5.8)

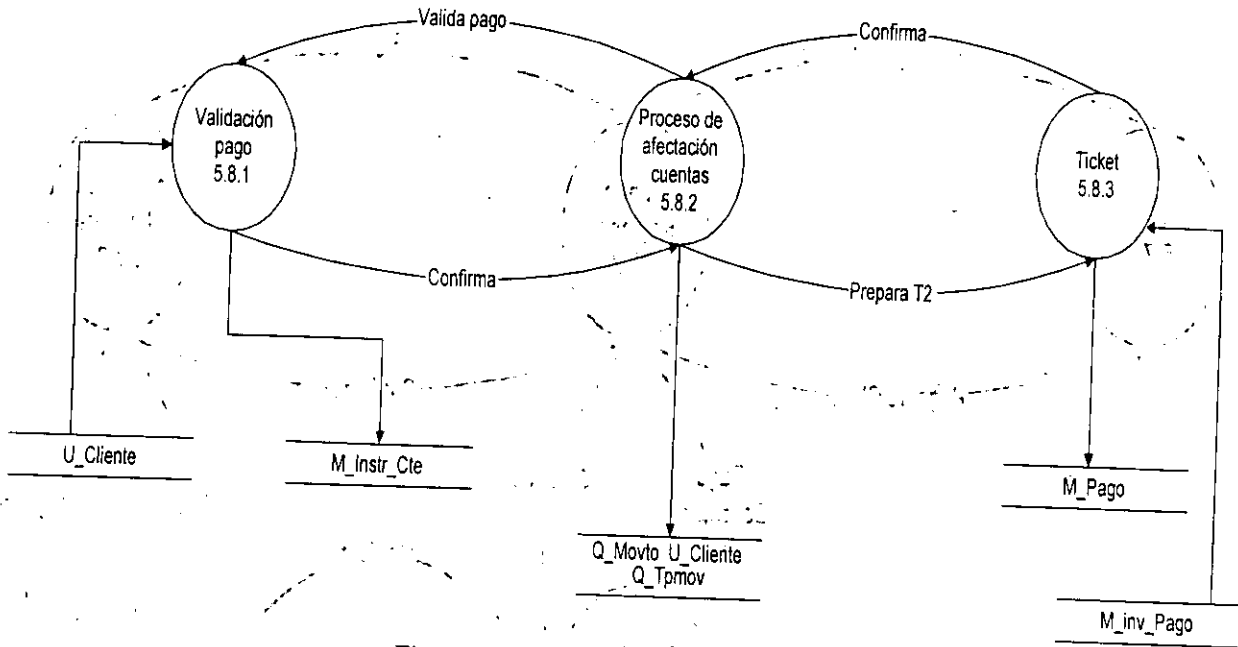


Figura 4.1_13 Diagrama de flujo nivel 3

Extravío de chequera (5.10)

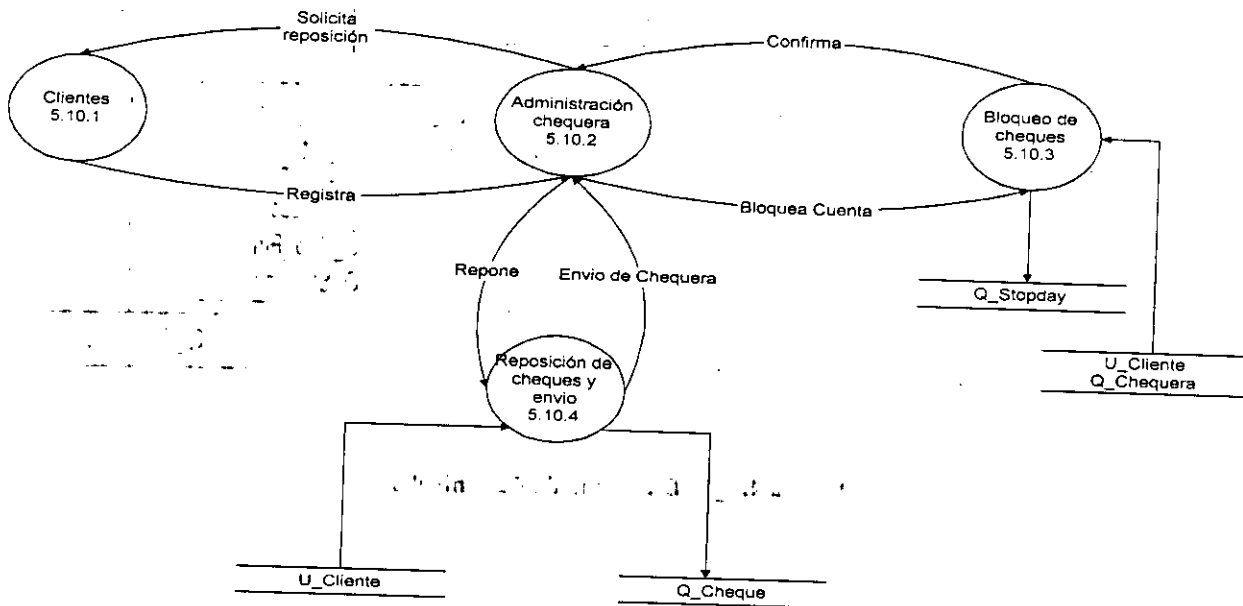


Figura 4.1_14 Diagrama de flujo de nivel 3

Cierre del mes (Impresión de Estado de cuenta) (5.11)

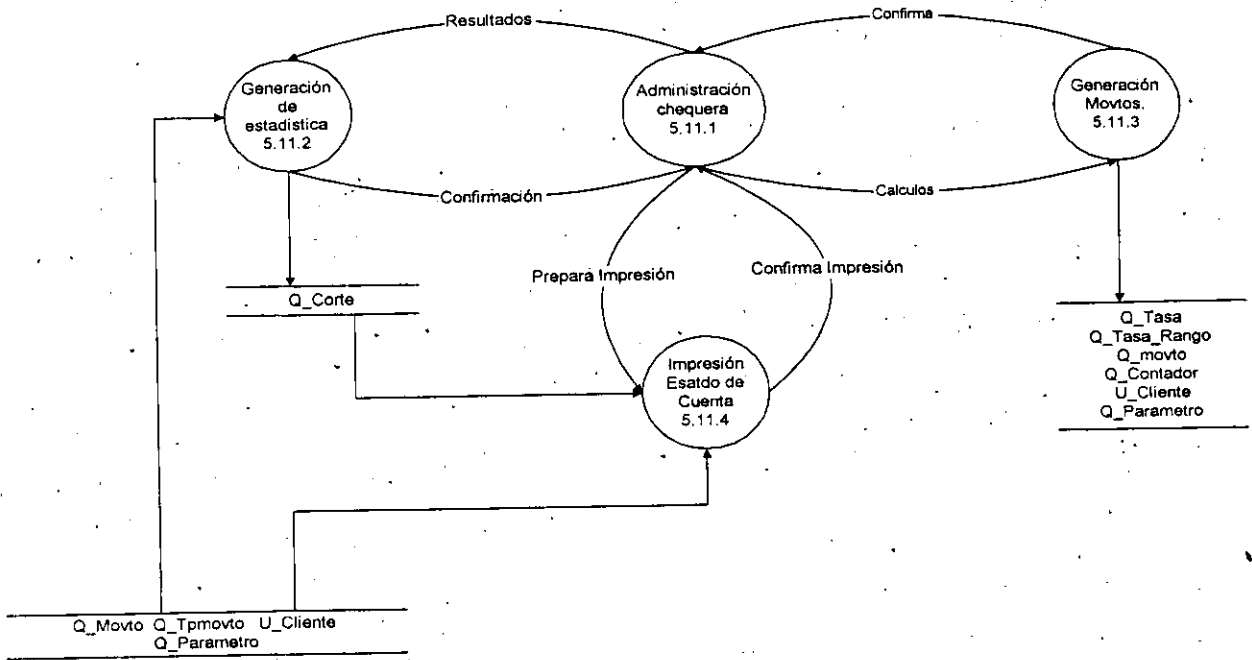


Figura 4.1_15 Diagrama de flujo de nivel 3

4.1.3 Diagrama Entidad Relación

La estructura lógica general de la base de datos se puede expresar en forma gráfica por medio de un diagrama E-R que se constituye por los siguientes elementos:

- El encabezado de los rectángulos, representa al conjunto de entidades o tablas.
- El contenido de los rectángulos representa los atributos de cada una de las tablas.
- Las líneas continuas representan las llaves primarias de una entidad que va a relacionar a otra entidad, donde esta relación pasa a formar parte como llave primaria de esta última.
- Las líneas semi-continuas representan las llaves foráneas de una entidad que va a relacionar a otra entidad, donde esta relación pasa a formar parte como atributos de esta última.

El diagrama de entidad-relación correspondiente a la base de datos del proyecto que nos ocupa aparece al final de este documento como Anexo 1.

En el Anexo 1 se muestra el diagrama Entidad-Relación en el cual se muestran las relaciones y la cardinalidad de las entidades. Así como los atributos que conforma a cada una de las entidades.

4.1.4 Normalización

Se llevó a cabo el proceso de normalización hasta la tercera forma normal (BCNF) para todas las tablas del sistema que nos ocupa (Sistema de Transferencia de Depósitos a la Vista en Dólares). Este proceso consiste en:

- Eliminar grupos que se repiten (primera forma normal).
- Eliminar datos redundantes (segunda forma normal).
- Eliminar columnas no dependientes en la llave (tercera forma normal).

A continuación se muestra dicho proceso para dos de las tablas “centrales” o que serán usadas más frecuentemente en el sistema: las tablas Q_Corte y Q_Movimiento.

- Q_Corte es la tabla que guarda estadísticas (para cada producto financiero, cliente y fecha de corte) tales como saldos, débitos, créditos, intereses, etc.
- Q_Movimiento es la tabla de movimientos en las cuentas de cheques.

Inicialmente se tenía en vez de las dos tablas indicadas una tabla única llamada Q_Estadísticas, la cual se diseñó pensando en integrar toda la información o estadísticas sobre las transacciones de cada cliente: saldos, cheques emitidos, créditos, débitos, intereses, movimientos, etc. En base a esto, los campos que se incluyeron en dicha tabla y su descripción son los que se muestran a continuación (aunque adelante en la sección “Diccionario de Datos” y en el apéndice del mismo nombre se muestran todas las tablas y campos del sistema en su versión final, se explican aquí los de la tabla Q_Estadísticas para fines de comprensión del proceso de normalización que se le aplicó, así como debido a que sus campos no coinciden con la versión final).

4.1.4.1 Tabla Q_ESTADISTICAS

Nombre de Campo o Columna	¿Llave Primaria? (1)
id cliente	1
año mes corte	2
f movto	3
Cronologico	4
Cta Cheques	
Otra Cuenta	
Saldo Inicial	
Saldo Final	
Saldo Promedio	
n cheques rebotados	
n cheques emitidos	
Tot Che Emitidos	
n Debits	
Tot Debits	
n Credits	
Tot Credits	
Interes	
Tasa interes	
Interes Anual	
Consecutivo	
id Movto	
Descrip Movto	
Referencia	
Monto	
Afecto Disponible	
Contabilizado	
id instrum	
n inversion	
n reinversion	
id cta che	
Notas (1) Los diferentes números asignados a los atributos de llave primaria indican el orden en que se incluyen en la llave primaria compuesta.	

Tabla 4.1_1 Tabla Q_ESTADISTICAS

4.1.4.2 Descripción de los Campos de Q_Estadísticas

id_cliente

Descripción: Identifica la Cuenta de Cheques.

año_mes_corte

Descripción: Año (incluye siglo) y mes del corte.

f_movto

Descripción: Fecha Valor del movimiento.

Help: Proporcione la fecha del movimiento.

Cronologico

Descripción: Orden cronológico de movimientos.

Cta_Cheques

Descripción: Indica si el registro es para una cuenta de cheques en dólares o no.

Otra_Cuenta

Descripción: Indica si el registro es para otro tipo de cuenta. Su valor debe ser siempre el contrario de Cta_Cheques.

Saldo_Inicial

Descripción: Saldo al inicio de mes del periodo.

Saldo_Final

Descripción: Saldo final al cierre de mes.

Saldo_Promedio

Descripción: Saldo Promedio del mes.

n_cheques_rebotados

Descripción: Número de cheques sobregirados.

n_cheques_emitidos

Descripción: Número de cheques emitidos en el mes

Tot_Che_Emitidos

Descripción: Total de cheques emitidos

n_Debits

Descripción: Numero de debitos en el periodo.

Tot_Debits

Descripción: Monto total de Debitos en el mes.

n_Credits

Descripción: Numero de Creditos en el periodo.

Tot_Credits

Descripción: Monto total de Creditos en el mes.

Interes

Descripción: Intereses pagados en el periodo.

Tasa_interes

Descripción: Tasa de interes pagada en el periodo.

Interes_Anual

Descripción: Interés anual acumulado.

Consecutivo

Descripción: Número consecutivo de corte para cada producto, cliente y mes de corte.

id_Movto

Descripción: Identificador del tipo de movimiento.

Descrip_Movto

Descripción: Descripción del tipo de movimiento.

Referencia

Descripción: Número de cheque o referencia del movimiento.

Monto

Descripción: Monto del movimiento.

Afecto_Disponible

Descripción: La cuenta ya afecto el saldo disponible ?

Contabilizado

Descripción: Ese movimiento ya fue contabilizado ?

id_instrum

Descripción: Identificador del instrumento de inversion.

n_inversion

Descripción: Numero de Inversion.

n_reinversion

Descripción: Numero de reinversion.

Descripción de los Campos de Q_Estadísticas (Cont.)

id_cta_che

Descripción: Numero de cuenta de cheques.

4.1.4.3 Primera Forma Normal

Usando la primera regla de normalización se eliminan los grupos repetitivos.

En la tabla Q_Estadísticas tiene el atributo Cta_Cheques para indicar si las estadísticas y movimiento de cada registro corresponden a una cuenta de cheques en dólares; este es el valor por omisión. Sin embargo, tiene también el atributo Otra_Cuenta para indicar si por el contrario, se trata de algún otro tipo de cuenta. Este atributo se agregó para permitir ampliar el sistema con otro tipo de instrumentos financieros. La llave primaria de esta tabla es:

id_cliente.año_mes_corte.f_movto.Cronologico.

Se puede ver que Cta_Cheques y Otra_Cuenta son campos que almacenan datos similares. Además, si se desea agregar más de un tipo de instrumentos adicionales, se requerirán más campos y por tanto modificaciones a los programas y a la tabla, por lo que esta no es adecuada para un número dinámico de instrumentos financieros. Aplicando la primera regla de normalización, los diferentes tipos de cuentas o instrumentos se movieron a una nueva tabla, llamada Tipos_Cuentas, la cual se puede ligar a Q_Estadísticas mediante la llave id_producto. En las tablas 4.1_2 y 4.1_3 se muestran junto con sus llaves primarias la nueva tabla Tipos_Cuentas y la tabla Q_Estadísticas en su primera forma normal.

Nombre de Campo o Columna	¿Llave Primaria?
id_producto	1
Descrip	

Tabla 4.1_2 Nueva Tabla TIPOS_CUENTAS

Nombre de Campo o Columna	¿Llave Primaria?
id producto	1
id cliente	2
año mes corte	3
f movto	4
Cronologico	5
Cta Cheques	
Otra Cuenta	
Saldo Inicial	
Saldo Final	
Saldo Promedio	
n cheques rebotados	
n cheques emitidos	
Tot Che Emitidos	
n Debits	
Tot Debits	
n Credits	
Tot Credits	
Interes	
Tasa interes	
Interes Anual	
Consecutivo	
id Movto	
Descrip Movto	
Referencia	
Monto	
Afecto Disponible	
Contabilizado	
id instrum	
n inversion	
n reinversion	
id cta che	

Tabla 4.1_3 Tabla Q_ESTADISTICAS en la Primera Forma Normal

4.1.4.4 Segunda Forma Normal

Al aplicar la segunda regla de normalización se eliminan los datos redundantes al remover los atributos o campos que dependen de solo una parte de la llave.

En la tabla Q_Estadísticas se tiene un registro o renglón por cada movimiento de cada cliente y dicho renglón incluye siempre las estadísticas (saldos, número de cheques, débitos, créditos, intereses y No. Consecutivo) del período de corte correspondiente a la fecha del movimiento.; la llave primaria de estos registros es:

id_producto.id_cliente.año_mes_corte.f_movto.Cronologico.

Se puede ver que las estadísticas de un cliente no dependen de toda la llave, sino solamente de "id_producto.id_cliente.año_mes_corte". Si un cliente tiene más de un movimiento por período de corte (lo cual es el caso más común) se tendrán renglones con los campos de estadísticas repetidos. Además, si un cliente no hace ningún movimiento durante un período de corte, no se tendrán estadísticas para ese cliente y período de corte.

Por tanto se pueden tener anomalías de actualización y borrado, las cuales se evitan separando Q_Estadísticas en las dos tablas Q_Corte (tabla de estadísticas de clientes) y Q_Movimiento (tabla de movimientos de los clientes) mencionadas al inicio de esta sección y que se muestran junto con sus llaves primarias en las tablas 4.1_4 y 4.1_5.

Nombre de Campo o Columna	¿Llave Primaria?
id producto	1
id cliente	2
año mes corte	3
Saldo Inicial	
Saldo Final	
Saldo Promedio	
n cheques rebotados	
n cheques emitidos	
Tot Che Emitidos	
n Debits	
Tot Debits	
n Credits	
Tot Credits	
Interes	
Tasa interés	
Interes Anual	
Consécutivo	

Tabla 4.1_4 Tabla Q_CORTE en la Segunda Forma Normal

Nombre de Campo o Columna	¿Llave Primaria?
id producto	1
id cliente	2
f movto	3
Cronologico	4
id Movto	
Descrip Movto	
Referencia	
Monto	
Afecto Disponible	
Contabilizado	
id instrum	
n inversion	
n reinversion	
Id cta che	

Tabla 4.1_5 Tabla Q_MOVIMIENTO en la Segunda Forma Normal

4.1.4.5 Tercera Forma Normal

En esta etapa las dependencias transitivas (dependencias en atributos o campos que no son parte de la llave) son removidas por descomposición adicional, resultando la tercera forma normal o la llamada forma normal de Boyce Codd (BCNF por sus siglas en inglés).

En la tabla Q_Movimiento, la descripción del movimiento (Descrip_Movto) no depende de la llave (id_producto.id_cliente.f_movto.Cronologico), sino del tipo de movimiento (id_Movto). Por tanto aquí también se pueden tener anomalías de actualización y borrado, las cuales se evitan moviendo el atributo Descrip_Movto a una nueva tabla, llamada Q_TPMOVTO.

A continuación se muestran las dos tablas Q_Movimiento y Q_TPMOVTO resultantes. En este punto las dos tablas Q_Corte y Q_Movimiento están ya en la tercera forma normal o BCNF (la tabla Q_Corte no sufrió ningún cambio, por lo que su tercera forma normal es la misma que la segunda forma normal).

Nombre de Campo o Columna	¿Llave Primaria?
id producto	1
id cliente	2
f movto	3
Cronologico	4
id Movto	
Descrip Movto	
Referencia	
Monto	
Afecto Disponible	
Contabilizado	
id instrum	
n inversion	
n reinversion	
Id cta che	

Tabla 4.1_6 Tabla Q_MOVIMIENTO en la Tercera Forma Normal

Nombre de Campo o Columna	¿Llave Primaria?
id Movto	1
Descrip Movto	

Tabla 4.1_7 Nueva Tabla Q_TPMOVTO

Eventualmente, la tabla Q_TPMOVTO fue ampliada con otros campos, como Descripción corta y larga en vez de "Descrip_Movto", naturaleza, etc. Su versión final se puede consultar en la sección "Diccionario de Datos".

4.1.5 Diccionario de Datos

4.1.5.1 Introducción

El diccionario de datos se utiliza para definir el significado, uso, características y otros datos relevantes de todos los campos, entidades o referencias cruzadas y las relaciones que existen entre ellos. En él se almacenan lo que en el medio se conocen como metadatos (datos definidos por datos) para todos los objetos que residen en nuestra base de datos. Este diccionario se almacena en una ubicación especial aparte del resto de las tablas de trabajo o datos y es utilizada exclusivamente por el kernel (núcleo) del manejador de bases de datos que utilizamos.

La tabla 4.1_8 nos da la descripción de las tablas usadas para el Sistema de Chequeras.

Nombre de Tabla	Descripción
C segmento	Segmento en el que se encuentra el giro del cliente
G contador	Tabla de números consecutivos.
G dia inhabil	Días inhábiles
G estado	Tabla de estados de la república
G impresora	Catalogo de impresoras
G pais	Países que tienen fechas inhábiles.
G_tipo_persona	Guarda los diferentes tipos de persona que puede haber para clasificar un cliente
K catalogo cta	Catalogo de cuentas contables
K movimiento	Movimientos de cada póliza contable
K operación	Tabla de operaciones posibles en contabilidad
K poliza	Archivo de pólizas que se integran al Paquete de Contabilidad
K relacion	Contiene la Relación que equivale a la subcuenta contable
K secuencia	Secuencias de contabilización
M area gral	
M cont ubic	Contadores de ultimas áreas, divisiones, plazas y sucursales.
M instr cte	Instrucciones de pago de clientes
M inv pago	Pagos registrados para desembolso
M pago	Pagos a realizar por desembolsos generados por chequera.
M promotor	Catalogo de promotores
M sucursal	Sucursales que reciben inversiones
M ubicacion	Tabla de ubicaciones (área, división y plaza)
O_banco	Contiene el catalogo de bancos que se importa al sistema del Chemical Bank.
O cancela preformato	Contiene información de los preformatos cancelados.
O estatus pago	Estatus de todos los pagos.
O preformato	Preformatos de captura para carga de los pagos.
O tipo envio	Descripción de los tipos de envío.
Q chequera	Chequeras impresas por cliente.
Q cheque remesa	Registro inicial de los cheques enviados a Barclays y Chase que serán los foráneos y locales.
Q contador	Contadores para llevar el orden cronológico en el registro de Q movimiento.
Q corte	Archivo que guarda los cortes, saldos finales en cada mes y estadísticas.

Tabla 4.1_8 Tablas en la Base de Datos del Sistema de Chequeras

Nombre de Tabla	Descripción
Q movimiento	Registro de movimientos de las Cuentas de Cheques.
Q mov chem	Alta de Movs. De la Cuenta Concentradora del Chemical.
Q on chequera	Registro de OverNigth de Chequera en Inversión.
Q parametro	Parámetros generales del sistema.
Q saldo_chem	Archivo que controla los saldos iniciales y finales de los movimientos de Débito o de Crédito de la cuenta concentradora del Chemical.
Q stoppay	Registro del Stop Payments. (Cheques Cancelados)
Q tasa	Tasas para el pago de intereses en el cierre del mes.
Q tasa rango	Tasas de interés aplicables a los saldos de las chequeras.
Q tpmovto	Archivo de tipos de movimiento del sistema de cheques.
S area	Area en la que se encuentra el usuario de uso de los programas.
S hist_passw	Archivo que contiene el histórico de passwords entrada del sistema.
S nivel	Niveles de jerarquía dentro de un área interna del Banco.
S programa	Programas usados por el sistema que se está ejecutando.
S tab cod imp	Tabla de códigos de impresión.
S usuario	Tabla de usuarios validos para acceder al sistema.
U asesor	Archivo que contiene los asesores (centros regionales)
U_cat_dist	Catalogo de código de distribución para el envío de los Estados de Cuenta.
U cheque	Archivo para el registro de pagos de cheques en centros regionales.
U cliente	Catalogo de clientes.
U ctro fin	Archivo para registrar los centros financieros
U memo	Envío de Memos para Microfichas y Barclays Bank (Alta).
U_movto_cheque	Catalogo para identificar lo que se puede hacer con un cheque cobrado en centro financiero por clientes de cuenta cheques. Ejemplo: Hacer efectivo, comprar transferencia, cheques de viajero, etc.

Tabla 4.1_8 (Cont.) Tablas en la Base de Datos del Sistema de Chequeras (Cont.)

4.1.5.2 Descripción del Reporte del Diccionario de Datos

Antes de presentar el reporte del diccionario de datos con las definiciones hechas en él, en las tablas 4.1_9 a 4.1_12 se muestra el significado de los nombres de columnas de cada sección del reporte. Los campos de la Sección "Field Details" (Detalle de Campos) son usados por Progress al momento de definir y desplegar componentes de aplicación que utilizan esos campos. En la sección 4.1.5.3 se incluye dicho reporte para algunas de las tablas usadas en la base de datos del sistema de chequeras (tablas C_Segmento a M_Pago). En el apéndice C se tiene el mismo reporte para todas las tablas restantes del sistema de chequeras (tablas M_Promotor a U_movto_cheque).

Table Name	Nombre de tabla
Table Dump	Nombre del "Dump" (vaciado de memoria)
Table Flags	Banderas de tablas
Count Field	Número de campos
Count Index	Número de índices
Table Label	Etiqueta de la tabla

Tabla 4.1_9 Sección "Table" (Tabla) del Diccionario de Datos

Ejemplos de banderas de tabla: "f" = "frozen" (congelada), "s" = tabla SQL

Order	Secuencia del campo
Field Name	Nombre del campo
Data Type	Tipo de dato
Flags	Banderas
Format	Formato
Initial	Valor Inicial
Label	Etiqueta del campo
Column Label	Etiqueta de columna

Tabla 4.1_10 Sección "Field Summary" (Sumario de Campos) del Diccionario de Datos

Ejemplo de banderas de campos: <c>ase sensitive, <i>index component, <m>andatory, <v>iew component

Flags	Banderas
Index name	Nombre del Índice
Cnt	Contador de Índices
Field name	Campo indexado

Tabla 4.1_11 Sección "Index Summary" (Sumario de Índices) del Diccionario de Datos

Ejemplo de banderas del sumario: <p>primary, <u>nique, <w>ord, <a>bbreviated, <i>nactive, + asc, - desc

Field Name	Nombre del campo
Description	Descripción del campo
Help	Mensaje de Ayuda
Val-Exp	Expresión de Validación
Val-Msg	Mensaje de Validación

Tabla 4.1_12 Sección "Field Details" (Detalle de Campos) del Diccionario de Datos

El detalle de los campos se muestra en la siguiente tabla. Este detalle incluye información sobre el tipo de dato, el formato, el valor inicial, la etiqueta del campo y la etiqueta de columna. La información se muestra en un formato que facilita la lectura y el análisis de los datos.

4.1.5.3 Reporte del Diccionario de Datos

===== Table: C_segmento =====

Table Name	Dump Name	Table Flags	Field Count	Index Count	Table Label
C_segmento	c_segmen		2	1	?

===== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
10	id_segmento	inte	im	99	0
20	descripcion	char	m	x(40)	

Field Name	Label	Column Label
id_segmento	id_segmento	id_segmento
descripcion	Descripcion del segmento	Descripcion del segmento

===== index summary =====

Flags	Index Name	Cnt	Field Name
pu	segmento	1	+ id_segmento

===== field details =====

- ** Field Name: id_segmento
 Description: Clave de segmentos. (giros)
 Help: Teclee la clave del segmento a la que pertenece el cliente.
- ** Field Name: descripcion
 Description: Describe el tipo de segmento. (giro)
 Help: Teclee la descripción del segmento.

===== Table: G_contador =====

Table Name	Dump Name	Table Flags	Field Count	Index Count	Table Label
G_contador	contador		2	1	?

===== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
10	cve_contador	char	i	X(17)	
20	consecutivo	inte		>>>, >>9	0

Field Name	Label	Column Label
cve_contador	?	?
consecutivo	?	?

===== index summary =====

Flags	Index Name	Cnt	Field Name
pu	cve_contador	1	+ cve_contador

===== field details =====

- ** Field Name: cve_contador
Val-Msg: La clave del contador es requerida
Val-Exp: INPUT cve_contador <> ""
- ** Field Name: consecutivo
Description: Valor actual del consecutivo. Se asigna por programa.

===== Table: G_dia_inhabil =====

Table Name	Dump Name	Table Flags	Field Count	Index Count	Table Label
G_dia_inhabil	dia_inha		3	2 (pk)	

===== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
10	id_pais	char	im	!!	
20	f_inhabil	date	im	99/99/9999	?
30	motivo	char	m	x(30)	

Field Name	Label	Column Label
id_pais	?	?
f_inhabil	?	?

motivo ? ?

===== index summary =====

Flags	Index Name	Cnt	Field Name
	fecha	1	+ f_inhabil
pu	llave_dia	2	+ id_pais + f_inhabil

===== field details =====

- ** Field Name: id_pais
 Description: Identificación del país.
 Help: Teclee la identificación del país.
 Val-Msg: Este país no existe.
 Val-Exp: can-find (G_pais of G_dia_inhabil)
- ** Field Name: f_inhabil
 Description: Fecha inhábil en que no puede haber operaciones.
 Help: Teclee la fecha inhábil.
 Val-Msg: La fecha dada es fin de semana...
 Val-Exp: weekday(input f_inhabil) <> 1 and weekday(input f_inhabil) <> 7
- ** Field Name: motivo
 Description: Causa por la que el día es inhábil.
 Help: Teclee el motivo del día inhábil.
 Val-Msg: Debe de haber un motivo.
 Val-Exp: input motivo <> ""

===== Table: G_estado =====

Table Name	Dump Name	Table Flags	Field Count	Index Count	Table Label
g_estado	g_estado		2	1	?

===== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
150	id_estado	inte	im	>9	
160	nombre	char	m	x(25)	

Field Name	Label	Column Label
id_estado	?	?
nombre	?	?

===== index summary =====

Flags	Index Name	Cnt	Field Name
pu	id_estado	1	+ id_estado

===== field details =====

- ** Field Name: id_estado
Help : Teclee la clave del estado.
- ** Field Name: nombre
Help : Teclee el nombre del estado.

===== Table: G_impresora =====

Table Name	Dump Name	Table Field Flags	Count	Index Count	Table Label
g_impresora	impresor		9	1	?

===== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
10	id_impresora	char	im	x(8)	
12	Estatus	logi	m	Activa/Inactiva	Activa
17	Comentario	char[2]		x(60)	
20	comando	char[3]		x(65)	
40	accesan	char	m	x(65)	
50	cola	char	m	x(30)	
60	acceso_def	char	m	x(65)	
70	owner	char		x(20)	

80 tipo_impresion char m x(10)

Field Name	Label	Column Label
id_impresora	Impresora	Impresora
Estatus	Estatus	Estatus
Comentario	Comentario	Comentario
comando	Comando	Comando
accesan	Accesan	Accesan
Cola	Cola	Cola
Acceso_def	Acceso def	Acceso def
owner	Owner	Owner
tipo_impresion	Tipo Impresión	Tipo Impresión

===== index summary =====

Flags	Index Name	Cnt	Field Name
pu	impresora	1	+ id_impresora

===== field details =====

** Field Name: id_impresora
 Help : Teclee la clave de la impresora
 Val-Msg : Debe proporcionar la clave de la impresora
 Val-Exp : id_impresora <> "

** Field Name: Estatus
 Help : Indique sí esta activa/inactiva

** Field Name: comando
 Help : Teclee los comandos de asignación

** Field Name: accesan
 Help : Indique que usuarios la accesan

** Field Name: Cola
 Help : Teclee el nombre de la cola
 Val-Msg : Debe de proporcionar el nombre de la cola
 Val-Exp : cola <> "

** Field Name: Acceso_def
 Help : Indique que usuarios la accesan por default

** Field Name: tipo_impresion
 Description : Tipo de impresión a utilizar.
 Help : Teclee el tipo de impresión a utilizar.

=====**Table: G_pais**=====

Table Name	Dump Name	Table Flags	Field Count	Index Count	Table Label
------------	-----------	-------------	-------------	-------------	-------------

G_pais	pais		2	1	?
--------	------	--	---	---	---

=====**field summary**=====

Order	Field Name	Data Type	Flags	Format	Initial
-------	------------	-----------	-------	--------	---------

11	id_pais	char	im	!!	
20	nombre	char	m	X(40)	

Field Name	Label	Column Label
------------	-------	--------------

id_pais	?	?
nombre	?	?

=====**index summary**=====

Flags	Index Name	Cnt	Field Name
-------	------------	-----	------------

pu	id_pais	1	+ id_pais
----	---------	---	-----------

=====**field details**=====

** Field Name: id_pais

Description : Identificación del país.

Help : Teclee la identificación del país.

Val-Msg : Debe proporcionarse la identificación del país

Val-Exp : input id_pais <> ""

** Field Name: nombre

Description : País o ciudad que tiene días inhábiles.

Help : Teclee el nombre del país o ciudad.

Val-Msg : Debe de proporcionarse el nombre del país o ciudad.

Val-Exp : input nombre <> ?

=====**Table: G_tipo_persona**=====

Table Name	Dump Name	Table Flags	Field Count	Index Count	Table Label
------------	-----------	-------------	-------------	-------------	-------------

G_tipo_persona	g_tipo_p		2	1	?
----------------	----------	--	---	---	---

===== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
10	Tipo_persona	inte	im	9	0
20	nombre	char	m	x(12)	

Field Name	Label	Column Label
Tipo_persona	?	?
nombre	?	?

===== index summary =====

Flags	Index Name	Cnt	Field Name
pu	tipo_persona	1	+ Tipo_persona

===== field details =====

** Field Name: Tipo_persona

Description : Tipo de persona. fisica, banco, subsid., persona moral, fideicomiso.

Help : Teclee el tipo de persona.

Val-Msg : Debe proporcionarse el tipo de persona.

Val-Exp : input Tipo_persona < 6

** Field Name: nombre

Description : Pais o ciudad que tiene días inhábiles.

Help : Teclee el nombre del país o ciudad.

Val-Msg : Debe de proporcionarse el nombre del país o ciudad.

Val-Exp : input nombre <> ?

===== Table: K_catalogo_cta =====

Table Name	Dump Name	Table Flags	Field Count	Index Count	Table Label
K_catalogo_cta	catalogo		6	1	?

===== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
10	cta	inte	im	9999	0
20	subcta	inte	im	99	0

30	subsubcta	inte	im	99	0
40	subsubsubcta	inte	im	99	0
50	nombre	char	m	x(30)	
60	catalogo_de	char	i	x(3)	

Field Name	Label	Column Label
cta	Cuenta	Cta.
subcta	Subcuenta	Subcta.
subsubcta	Subsubcta.	Subsubcta.
subsubsubcta	Subsubsubcta.	Subsubsubcta.
nombre	Descripción	Desc.
catalogo_de	?	?

===== index summary =====

Flags Index Name	Cnt Field Name
pu cuenta	5 + catalogo_de + cta + subcta + subsubcta + subsubsubcta

===== field details =====

- ** Field Name: cta
Description : Número de cuenta contable.
Help : Teclee el número de cuenta.
- ** Field Name: subcta
Description : Subcuenta contable.
Help : Teclee la subcuenta.
- ** Field Name: subsubcta
Help : Teclee la subsubcuenta.
- ** Field Name: subsubsubcta
Help : Teclee la sub-sub-subcuenta.
- ** Field Name: nombre
Help : Teclee la descripción de la cuenta.
- ** Field Name: catalogo_de
Help : Teclee de quien es el catalogo (DT=Dynatrúst, DBT=Dyna,etc.)

```
===== Table: K_movimiento =====
```

Table Name	Dump Name	Table Field Flags	Field Count	Index Count	Table Label
K_movimiento	movimien		6	1	?

```
===== field summary =====
```

Order	Field Name	Data Type	Flags	Format	Initial
10	cuenta	char	m	x(14)	
20	n_poliza	inte	im	999999	0
30	referencia	char		x(8)	
40	tipo_mov	inte	m	9	1
50	importe	deci-2	m	-999999999999.9	0
70	n_movimiento	inte	im	99	0

Field Name	Label	Column Label
cuenta	?	?
n_poliza	?	?
referencia	?	?
tipo_mov	?	?
importe	?	?
n_movimiento	?	?

```
===== index summary =====
```

Flags	Index Name	Cnt	Field Name
pu	movimiento	2	+ n_poliza + n_movimiento

```
===== field details =====
```

** Field Name: tipo_mov
Description : 1=cargo , 2=abono

```
===== Table: K_operacion =====
```

Table Name	Dump Name	Table Field Flags	Field Count	Index Count	Table Label
K_operacion	t_operac		2	1	?

===== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
10	id_operacion	inte	m	999	0
20	descripcion	char	m	x(7)	

Field Name	Label	Column Label
id_operacion	Id.Operación	Id.Operación
descripcion	Descripción	Desc.

===== index summary =====

Flags	Index Name	Cnt	Field Name
pu	t_operacion	1	+ id_operacion

===== field details =====

- ** Field Name: id_operacion
Help : Teclee el identif. De la operación contable.
- ** Field Name: descripcion
Help : Teclee la descripción de la operación a contabilizar.

===== Table: K_poliza =====

Table Name	Dump Name	Table Field Flags	Field Count	Index Count	Table Label
K_poliza	poliza		11	3	?

===== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
10	id_instrum	char	m	!!	
20	f_poliza	date	im	99/99/9999	today
40	tipo_poliza	inte	m	9	3
60	n_poliza	inte	im	999999	0
80	clase	inte		9	1
100	diario	inte		999	0
120	concepto	char		x(30)	
140	f_fin_acept	date		99/99/9999	today
150	f_pol_lantek	date	m	99/99/9999	?

```

160 referencia      char      i      x(17)      ?
170 catalogo_de    char      i      x(3)

```

Field Name	Label	Column Label
id_instrum	Fid.	Fid.
f_poliza	?	?
tipo_poliza	?	?
n_poliza	?	?
clase	?	?
diario	?	?
concepto	?	?
f_fin_acept	?	?
f_pol_lantek	Fecha Envío	Fecha Envío
referencia	?	?
catalogo_de	?	?

===== index summary =====

Flags Index Name	Cnt Field Name
f_poliza_cat	2 + f_poliza + catalogo_de
pu poliza	1 + n_poliza
u referencia	1 + referencia

===== field details =====

```

** Field Name: id_instrum
  Description : Identificador del instrumento que se contabiliza.
** Field Name: tipo_poliza
  Description : Tipo de póliza 1=ingresos,2=egresos,3=diario,4=orden,5=estad.
** Field Name: clase
  Description : clase 1=normal,2=permanente,3=perm.sin afectar,4=sin afectar
** Field Name: concepto
  Description : concepto de la póliza
** Field Name: f_fin_acept
  Description : fecha de fin de aceptación, si la póliza es permanente.
** Field Name: f_pol_lantek
  Description : Fecha en que se pasa al archivo ASCII de Lantek/C
** Field Name: referencia
  Description : Referencia hacia los archivos de inversiones, pagos, etc.
** Field Name: catalogo_de
  Help      : Teclee de quien es el catalogo (DT=Dynatrust, DBT=Dyna,etc.)

```


=====**Table: K_relacion**=====

Table Name	Dump Name	Table Field Flags	Count	Index Count	Table Label
------------	-----------	-------------------	-------	-------------	-------------

K_relacion	k_relaci		2	1	?
------------	----------	--	---	---	---

=====**field summary**=====

Order	Field Name	Data Type	Flags	Format	Initial
10	relacion	inte	im	99	0
20	nombre	char	m	x(15)	
	Field Name	Label		Column Label	
	relacion	?	?	?	
	nombre	?	?	?	

=====**index summary**=====

Flags	Index Name	Cnt	Field Name
pu	relacion	1	+ relacion

=====**field details**=====

** Field Name: relacion
 Description : Related, Unrelated, within, abroad
 Help : Teclee la clave de relación

** Field Name: nombre
 Description : Teclee el nombre de la relación

=====**Table: K_secuencia**=====

Table Name	Dump Name	Table Field Flags	Count	Index Count	Table Label
------------	-----------	-------------------	-------	-------------	-------------

K_secuencia	secuenci		12	1	?
-------------	----------	--	----	---	---

===== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
10	id_operacion	inte	im	999	0
20	n_secuencia	inte	im	999	1
30	id_instrum	char	im	!!	
40	cta	inte	m	9999	0
50	subcta	inte		99	0
60	subsubcta	inte		99	0
70	subsubsubcta	inte		99	0
80	tipo_mov	inte	m	9	1
100	campo	char		x(15)	
110	signo	inte		-9	1
130	naturaleza	char		x(2)	
140	id_instrum_proced	char	i	!!	

Field Name	Label	Column Label
id_operacion	Id.Operación	Id.Operación
n_secuencia	Secuencia	Sec.
id_instrum	Abreviación	Abrev.
cta	Cuenta	Cta.
subcta	Subcuenta	Subcta.
subsubcta	Subsubcta.	Subsub!cta.
subsubsubcta	Subsubsubcta.	Subsub!subcta.
tipo_mov	Cgo./Abo.	Cgo./Abo.
campo	Campo	Campo
signo	Signo	Signo
naturaleza	?	?
id_instrum_proced	?	?

===== index summary =====

Flags	Index Name	Cnt	Field Name
pu	secuencia	4	+ id_operacion + id_instrum + id_instrum_proced + n_secuencia

===== field details =====

** Field Name: id_operacion
 Help : Teclee el identif. De la operación contable.
 Val-Msg : El identificador de la operación no existe.

** Field Name: n_secuencia

- Description : Numero de secuencia dentro de la operación.
 Help : Teclee la secuencia.
- ** Field Name: id_instrum
 Description : instrumento que se contabiliza.
 Help : Teclee la abreviación que usa en la tabla de subcuenta.
- ** Field Name: cta
 Description : Número de cuenta contable.
 Help : Teclee el número, de cuenta.
- ** Field Name: subcta
 Description : Subcuenta contable.
 Help : Teclee la subcuenta.
- ** Field Name: subsubcta
 Help : Teclee la subsubcuenta.
- ** Field Name: subsubsubcta
 Help : Teclee la sub-sub-subcuenta.
- ** Field Name: tipo_mov
 Description : 1=cargo , 2=abono
 Help : Teclee 1=Cargo, 2= Abono.
 Val-Msg : Los valores válidos son 1 y 2.
 Val-Exp : INPUT tipo_mov = 1 or INPUT tipo_mov = 2
- ** Field Name: campo
 Description : Campo que contiene el importe a cargar o abonar.
 Help : Teclee el campo que contiene el importe.
- ** Field Name: signo
 Description : Signo por el que se multiplica el importe. 1=positivo -1negativo
 Help : Teclee el signo (1=positivo, -1=negativo).
 Val-Msg : Los valores validos son 1 y -1.
 Val-Exp : INPUT signo = 1 or INPUT signo = -1

===== Table: M_area_gral =====

Table Name	Dump Name	Table Flags	Field Count	Index Count	Table Label
M_area_gral	m_area_g		3	2	

===== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
10	id_gral	inte	i	99	0
20	id_area	inte		>9	0
30	nombre	char		x(15)	

Field Name	Label	Column Label
id_gral	?	?
id_area	?	?
nombre	?	?

===== index summary =====

Flags	Index Name	Cnt	Field Name
u	id_area	1	+ id_area
pu	id_area_gral	2	+ id_gral + id_area

===== field details =====

** Field Name: id_area

Description : Area del cliente. En esta tabla se asigna por programa.

===== Table: M_cont_ubic =====

Table Name	Dump Name	Table Field Flags	Index Count	Table Count	Table Label
M_cont_ubic	cont_ubi		5	1	?

===== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
10	tipo	char	i	!	
20	area	inte	im	>>9	0
30	division	inte	im	>>9	0
40	plaza	inte	im	>>9	0
50	sucurs	inte	m	>>9	0

Field Name	Label	Column Label
tipo	?	?
area	?	?
division	?	?
plaza	?	?
sucurs	?	?

===== index summary =====

Flags	Index Name	Cnt	Field Name
pu	llave_cont_u	4	+ tipo + area + division + plaza

===== field details =====

- ** Field Name: tipo
Description : Tipo de contador. a=áreas, d=divis, p=plazas, s=sucurs
- ** Field Name: area
Description : Sig. valor del contador de áreas
- ** Field Name: division
Description : Sig. valor del contador de divisiones
- ** Field Name: plaza
Description : Sig. valor del contador de plazas
- ** Field Name: sucurs
Description : Sig. valor del Contador de sucursales.

===== Table: M_instr_cte =====

Table Name	Dump Name	Table Field	Index Count	Table Index Count	Label
M_instr_cte	m_instr_	27	2	?	

===== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
5	id_sistema	char	cim		M
8	id_producto	char	i		
10	id_cliente	inte	im	<<>>>>9	0
15	n_pago_cte	inte	i	>9	1
16	instruc_pago	char	m	XXX	TRS
18	n_preformato	inte	i	>>9	0
20	your_reference	char		x(16)	
30	related_reference	char		x(16)	
40	cr_party_type	inte	9		0
50	cr_party_id	char		x(16)	
55	cr_party_name	char		x(35)	
58	cr_party_addr	char[3]		x(35)	
59	acc_party_type	inte		9	0
60	acc_party_id	char		x(34)	

70	acc_party_name	char	x(35)	
80	acc_party_addr	char[3]	x(35)	
100	pay_details	char[2]	x(35)	
110	bank_information	char[2]	x(35)	
120	advice_method	char	!	
130	charges	char	!	O
132	ordering_party_name	char	x(35)	
133	ordering_party_addr	char[3]	x(35)	
135	ult_benef_acc	char	x(34)	
140	ult_benef_name	char	x(35)	
145	ult_benef_addr	char[3]	x(35)	
180	last_beneficiary	char	!	N
190	st_instruc	char	x(3)	REG

Field Name	Label	Column Label
id_sistema	Depto. que genera	Dese. Depto.
id_producto	Id_producto	Id_producto
id_cliente	?	?
n_pago_cte	?	?
instruc_pago	?	?
n_preformato	?	?
your_reference	Your Reference	Your Ref
related_reference	Related Reference	Rel Ref
cr_party_type	?	?
cr_party_id	?	?
cr_party_name	?	?
cr_party_addr	?	?
acc_party_type	?	?
acc_party_id	?	?
acc_party_name	?	?
acc_party_addr	?	?
pay_details	Pay Details	Pay Details
bank_information	Bank Information	Bank Info
advice_method	?	?
charges	?	?
ordering_party_name	?	?
ordering_party_addr	?	?
ult_benef_acc	?	?
ult_benef_name	?	?
ult_benef_addr	?	?
last_beneficiary	?	?
st_instruc	?	?

===== index summary =====

Flags	Index Name	Cnt	Field Name
	prefor	1	+ n_preformato
pu	sist_con_cliente	4	+ id_sistema + id_producto + id_cliente + n_pago_cte

===== field details =====

** Field Name: id_sistema

Description : Indica que sistema genera la salida de Dinero en Operaciones

Help : Teclee la letra inicial del Sistema que genera el Desembolso

Val-Msg : La clave del sistema es incorrecta

Val-Exp : id_sistema = 'T' OR id_sistema = 'M' OR id_sistema = 'C'

** Field Name: id_producto

Description : Sistema al que pertenece el cliente.

Help : Sistema al que pertenece el cliente

Val-Msg : Dar de alta (M=Mesa,U=Univ.)

Val-Exp : INPUT id_producto = "Q" OR INPUT id_producto = "U"

** Field Name: id_cliente

Description: Número consecutivo asignado a cada cliente.

Help : Teclee la clave del cliente.

Val-Msg : Debe proporcionarse el numero de cliente.

Val-Exp : input id_cliente <> "

** Field Name: n_pago_cte

Description : Numero de pago a efectuar por la instrucc. Al vencimiento.

** Field Name: instruc_pago

Description : Tipo de Pago (OPA) (TRS) (CHE).

Help : Teclee el tipo de pago. 'OPA, TRS, CHE'.

Val-Msg : Forma de pago incorrecto.

Val-Exp : input M_instr_cte.instruc_pago = 'TRS' or input M_instr_cte.instruc_pago = 'OPA' or input M_instr_cte.instruc_pago = 'CHE'

** Field Name: n_preformato

Description : Número del preformato

** Field Name: your_reference

Description : Tu Referencia

Help : Your Reference

** Field Name: related_reference

Description : Related Reference

Help : Related Reference

** Field Name: cr_party_type

Description : Env.Pag.1=Name.2=Acco.Num.3=CHIPSunid.4=CHIPSpart.5=FEDWIRE
ba.6=SWIFT

Val-Msg : El valor es de tipo Numérico y va de 1 a 6.

- Val-Exp : input cr_party_type >= 1 AND input cr_party_type <= 6
- ** Field Name: cr_party_id
Description : Número de vía.
Help : Teclee la vía de pago.
Val-Msg : Debe proporcionarse como se pagara
Val-Exp : input cr_party_id <> "
- ** Field Name: cr_party_name
Description : Credit Party Name
Help : Teclee el credit party name
Val-Msg : Debe teclearse el credit party name
- ** Field Name: cr_party_addr
Description : Credit Party addr
Help : Teclee el credit party address
Val-Msg : Debe teclearse la dirección
Val-Exp : cr_party_addr[1] <> "
- ** Field Name: acc_party_type
Description : Id.Cuenta 1=Name. 2=Account Num. 3=CHIPS UID. 6=SWIFT ID.
7=NONE.
Val-Msg : Identificador de la cuenta de Partida.
Val-Exp : (input acc_party_type >= 1 AND input acc_party_type <= 3)
OR (input acc_party_type = 6) OR (input acc_party_type = 7)
- ** Field Name: acc_party_id
Description : Cuenta Receptora
Help : Teclee el numero de acc party account
Val-Msg : Debe proporcionarse el numero de cuenta.
Val-Exp : acc_party_id <> "
- ** Field Name: acc_party_name
Description : A nombre de quien va a salir el pago.
Help : Teclee a nombre de quien es el pago.
Val-Msg : Debe teclearse el nombre.
Val-Exp : acc_party_name <> "
- ** Field Name: acc_party_addr
Description : Dirección del beneficiario
Help : Teclee la acc. party address
- ** Field Name: pay_details
Description : Pay Details
Help : Teclee la descripción del pago o [F1] = Terminar.
- ** Field Name: bank_information
Description : Bank Information
Help : Teclee la información del banco o [F1] = Terminar.
- ** Field Name: advice_method
Description : Métodos de Aviso : M.= Mail P = Phone T =.Telex.
Help : Métodos de Aviso: M=Mail, P=Phone, T=Telex o [F1]=Para Terminar
Val-Msg : Debe Proporcionarse el Advice Method.
Val-Exp : (INPUT advice_method = 'M') OR (INPUT advice_method = 'P') OR (INPUT
advice_method = 'T')
- ** Field Name: charges
Description : Cargos: B = Beneficiary (Beneficiario), O = Our (Nosotros).
Help : Cargos: B = Beneficiario, O = Nosotros o [F1] = Terminar.

- Val-Msg : Indicador a quien se le hacen los Cargos de las Transacciones.
- Val-Exp : INPUT charges = 'B' OR INPUT charges = 'O'
- ** Field Name: ordering_party_name
Description : Ordering Party Name
Help : Ordering party name o [F1] = Terminar
- ** Field Name: ordering_party_addr
Description : Ordering Party Addr
Help : Ordering party addr o [F1] = Terminar.
- ** Field Name: ult_benef_acc
Description : Ult. benef acc
Help : Teclee la cuenta del beneficiario final o [F1] = Terminar.
- ** Field Name: ult_benef_name
Description : Ult. benef name
Help : Teclee el nombre del beneficiario final o [F1] = Terminar.
- ** Field Name: ult_benef_addr
Description : Ult. benef addr
Help : Teclee la dirección del beneficiario final o [F1] = Terminar.
- ** Field Name: last_beneficiary
Description : Se teclea Banco B = Bank (Banco) N = Non-Bank (No-Banco)
Help : Id. B = Bank (Banco) N = Non-Bank (No-Banco) o [F1]=Terminar.
Val-Msg : Se utiliza este campo para dar alta Banco (B) o (N) No-Banco
Val-Exp : INPUT last_beneficiary = 'B' OR INPUT last_beneficiary = 'N'
- ** Field Name: st_instruc
Description : STATUS DE LA INSTRUCCION REGISTRADA, APROBADA, INACTIVA

===== Table: M_inv_pago =====

Table Name	Dump Name	Table Flags	Field Count	Index Count	Table Label
M_inv_pago	inv_pago		7	3	?

===== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
5	id_sistema	char	cim		M
10	id_instrum	char	i	X(3)	DT
20	n_consec	inte	im	>>>>9	0
30	n_reinvers	inte	i	9999	0
40	n_pago	inte	im	>9	1
50	monto	deci-2	m	>>>>>>9.99	0
60	n_ctrl_pago	inte	i	>>>>9	0

Field Name	Label	Column Label
id_sistema	Depto. que	Genera Dese
id_instrum		Depto.

n_consec	?	?
n_reinvers	?	?
n_pago	?	?
monto	?	?
n_ctrl_pago	?	?

===== index summary =====

Flags	Index Name	Cnt	Field Name
pu	inv_pag_llav	4	+ id_instrum + n_consec + n_reinvers + n_pago
	n_ctrl_pago	1	+ n_ctrl_pago
	sist_folio	4	+ id_sistema + id_instrum + n_consec + n_reinvers

===== field details =====

** Field Name: id_sistema
 Description : Indica que sistema genera la salida de Dinero en Operaciones
 Help : Teclee la letra inicial del Sistema que genera el Desembolso
 Val-Msg : La clave del sistema es incorrecta
 Val-Exp : id_sistema = 'T' OR id_sistema = 'M' OR id_sistema = 'C'

** Field Name: id_instrum
 Description : Identificador del instrumento. Se asigna por programa.

** Field Name: n_consec
 Description : Numero consecutivo de la inversión.

** Field Name: n_reinvers
 Description : Numero consecutivo de reinversión.

** Field Name: n_pago
 Description : Numero de pago a efectuar por la instrucc. al vencimiento.

** Field Name: monto
 Description : Monto a pagar.
 Help : Teclee el monto que se va a pagar.
 Val-Msg : El monto a pagar debe de ser mayor que cero.
 Val-Exp : input monto > 0

** Field Name: n_ctrl_pago
 Description : Numero de control de pago.

Table: M_pago

Table Name	Dump Name	Table Field Index	Table Flags	Table Count	Table Label
M_pago	m_pago	16	9	?	

field summary

Order	Field Name	Data Type	Flags	Format	Initial
3	id_sistema	char	ci		M
4	id_producto	char	i		
6	id_cliente	inte	i	>>>>9	0
15	n_pago_cte	inte	im	>9	0
20	n_ctrl_pago	inte	i	>>>>9	0
35	n_cuenta	char		x(20)	
60	monto	deci-2	m	>>>, >>>, >>>9.99	0
110	n_t2	inte	i	>>, >>9	0
120	f_liberacion	date	i	99/99/9999	?
150	f_a_pagar	date	i	99/99/9999	?
160	id_instrum	char	m	x(4)	
210	tipo_pago	inte		9	0
400	agrupado_en	inte	i	>>>>9	?
480	your_reference	char	m	x(16)	
490	related_reference	char	m	x(16)	
500	st_envio	logi	i	yes/no	no

Field Name	Label	Column Label
id_sistema	Depto. que	genera Dese Depto.
id_producto	Id_producto	Id_producto
id_cliente	?	?
n_pago_cte	?	?
n_ctrl_pago	?	?
n_cuenta	Cta.	Cta.
monto	?	?
n_t2	?	?
f_liberacion	?	?
f_a_pagar	?	?
id_instrum	?	?
tipo_pago	?	?
agrupado_en	?	?
your_reference	Your Reference	Your Ref
related_reference	Related Reference	Rel Ref
st_envio	st_envio	st_envio

===== index summary =====

Flags	Index Name	Cnt	Field Name
	agrupado_en	1	+ agrupado_en
	agrup_f_apag	2	+ agrupado_en + f_a_pagar
	agrup_nt2	2	+ agrupado_en + n_t2
	cte_instruc	3	+ id_producto + id_cliente + n_pago_cte
	f_a_pagar	1	+ f_a_pagar
	nt2_f_apag	2	+ n_t2 + f_a_pagar
	nt2_f_pagar_lib	4	+ n_t2 + f_a_pagar + f_liberacion + st_envio
u	n_ctrl_pago	1	+ n_ctrl_pago
p	pago_por_sist	4	+ id_sistema + id_producto + id_cliente + f_a_pagar

===== field details =====

**** Field Name: id_sistema**

Description : Indica que sistema genera la salida de Dinero en Operaciones

Help : Teclee la letra inicial del Sistema que genera el Desembolso

Val-Msg : La clave del sistema es incorrecta

Val-Exp : id_sistema = 'M' OR id_sistema = 'T' or id_sistema = 'C'

**** Field Name: id_producto**

Description : Sistema al que pertenece el cliente.

Help : Sistema al que pertenece el cliente

Val-Msg : Dar de alta (Q=Cheq, U=Univ.)

Val-Exp : INPUT id_producto = "Q" OR INPUT id_producto = "U"

**** Field Name: id_cliente**

Description : Numero consecutivo asignado a cada cliente.

Help : Teclee la clave del cliente.

- Val-Msg : Debe proporcionarse el numero de cliente.
Val-Exp : input id_cliente > 0
- ** Field Name: n_pago_cte
Description : Numero de pago a efectuar por la instrucc. al vencimiento.
- ** Field Name: n_ctrl_pago
Description : Numero de control de pago.
- ** Field Name: n_cuenta
Help : Teclee el numero de cuenta, en la cual se debita el pago.
Val-Exp : CAN-FIND (T_cuenta where T_cuenta.id_banco = INPUT M_pago.id_banco AND T_cuenta.n_cuenta = INPUT M_pago.n_cuenta)
- ** Field Name: monto
Description : Monto a pagar.
Help : Teclee el monto que se va a pagar.
Val-Msg : El monto a pagar debe de ser mayor que cero.
Val-Exp : input monto > 0
- ** Field Name: n_t2
Description : Numero de T2 que respalda al pago. Asignado por sistema.
- ** Field Name: f_liberacion
Description : Fecha de liberación del pago en el M.H.T.
Help : Teclee la fecha en que se libero el pago.
Val-Msg : Debe proporcionarse la fecha de liberación.
Val-Exp : weekday(input f_liberacion)>1 and weekday(f_liberacion) < 7
- ** Field Name: f_a_pagar
Description : Fecha valor del pago.
Help : Teclee la fecha en que se va a realizar el pago.
Val-Msg : La fecha de pago cae en fin de semana
Val-Exp : weekday(input f_a_pagar)<> 1 and weekday(input f_a_pagar)<> 7.
- ** Field Name: id_instrum
Description : Identificador del instrumento de inversión.
Help : Teclee la clave del instrumento de la inversión.
Val-Msg : Debe proporcionarse el tipo de instrumento.
- ** Field Name: tipo_pago
Description : Tipo de pago. 1=inv con pago de inter por periodos, 0=int al enc.
- ** Field Name: your_reference
Description : Tu Referencia
Help : Your Reference
- ** Field Name: related_reference
Help : Related Reference
- ** Field Name: st_envio
Description : Este pago ya se libero.

4.2 Diseño y Construcción del "Front-End"

Las aplicaciones en un ambiente gráfico (Windows) se pueden desarrollar mediante técnicas, herramientas de programación y con el conocimiento de algún lenguaje de programación de alto nivel. Para nuestro caso particular, desarrollamos nuestro "Front-End" (interfase con el usuario final) en un ambiente gráfico, utilizando como herramienta de desarrollo Progress Ver. 8.3 que es un lenguaje "4GL", con el objetivo de hacer fácil al usuario final la interacción con el sistema.

En un entorno gráfico el usuario interactúa con los objetos que hay en la pantalla, ejecutando con ellos un evento o una acción que hace referencia a un procedimiento obteniendo así una respuesta de acuerdo a la acción realizada.

4.2.1 Fases para el Desarrollo de Aplicaciones Gráficas con Progress

De manera genérica en el desarrollo de una aplicación gráfica en Progress, se utilizan las siguientes fases:

- Establecer la ventana y las propiedades que va a poseer.
- Añadir objetos (controles) y las propiedades que poseen.
- Escribir el código de los objetos que contendrán la ventana.
- Desarrollar los procedimientos asociados a cada uno de los eventos de los controles y de la ventana misma.
- Integrar las formas utilizadas en el sistema.
- Compilación del programa.

Las herramientas con que cuenta el lenguaje para el desarrollo del Front-End se muestran en la figura 4.2_1. Consisten de un menú principal y de una paleta de objetos que puede poseer cada una de las ventanas que se desarrollen.

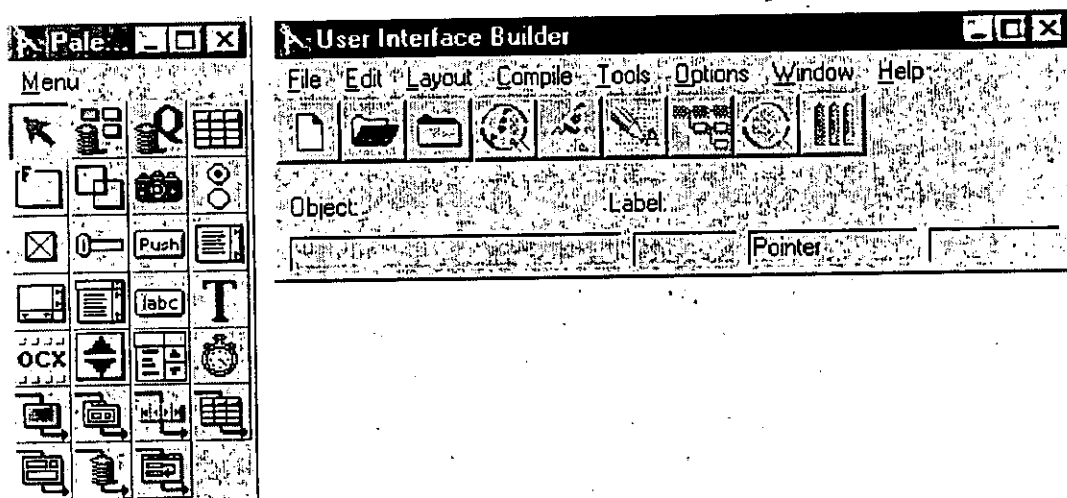


Figura 4.2_1 Herramienta Progress UIB para Desarrollo Gráfico

Para crear una ventana se selecciona del menú principal una forma nueva; aparece un grupo de objetos que se pueden generar y se selecciona "Window" (ventana) o "SmartWindow" (ventana inteligente), tal y como se muestra en la figura 4.2_2.

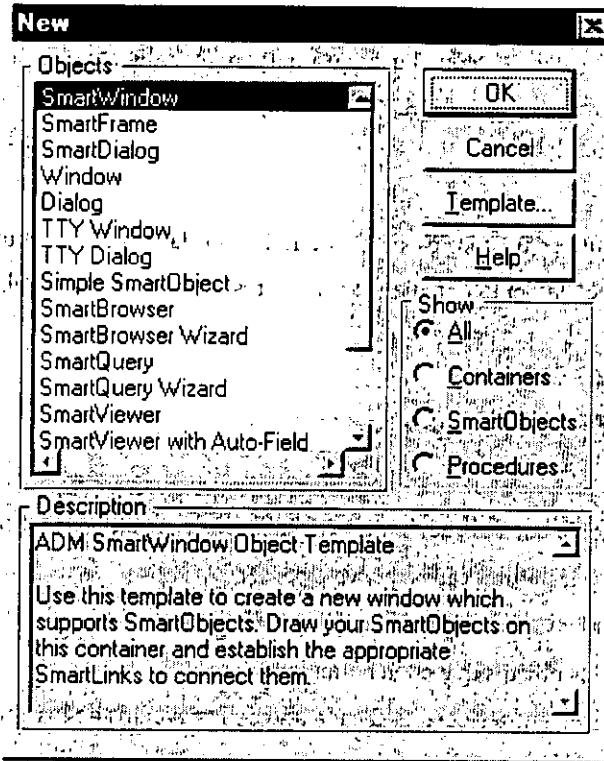


Figura 4.2_2 Diferentes Objetos que se pueden crear desde Progress.UIB

4.2.1.1 "Text" (Texto)

Son objetos en los cuales se pueden colocar textos fijos (etiquetas) y que además no realizarán ninguna acción. Sus propiedades se muestran en la figura 4.2_4.

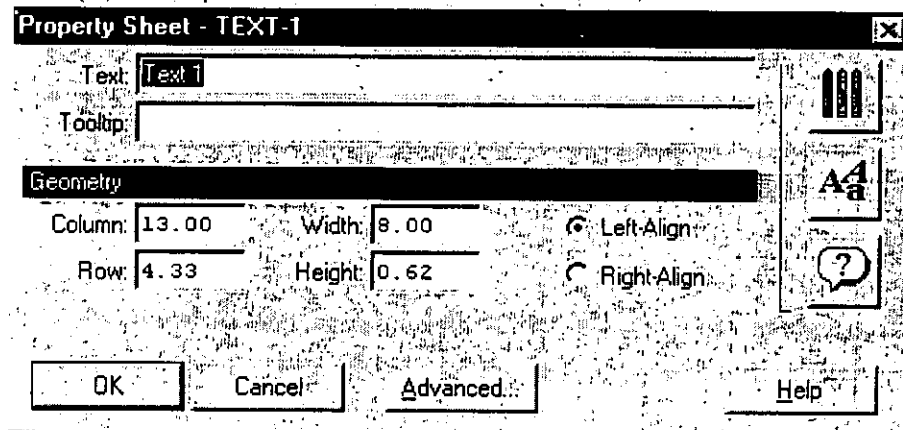


Figura 4.2_4 Propiedades de los Textos o Etiquetas

Propiedades

- **Text:** Se refiere a la etiqueta o título que debe mostrarse.
- **Tooltip:** Indica que cuando el cursor pase sobre esta etiqueta se muestre una pequeña línea de ayuda al usuario, también conocida como TAB.
- **Geometry:** Nos indica en que posición se encuentra ubicado dicho texto. Además contiene la altura y el ancho del objeto texto y que justificación se requiere para el texto, es decir: justificado a la izquierda, derecha o bien centrado.
- **Otros botones:** El primer botón (colores) define los colores que se van a usar en el frente y el fondo del texto. El segundo botón define el tipo de letra que se va a usar para el texto.

4.2.1.1 "Windows" (Ventanas)

Las ventanas son la parte contenedora de los objetos que van a estar colocados dentro de esta. Sus propiedades se muestran en la figura 4.2_3:

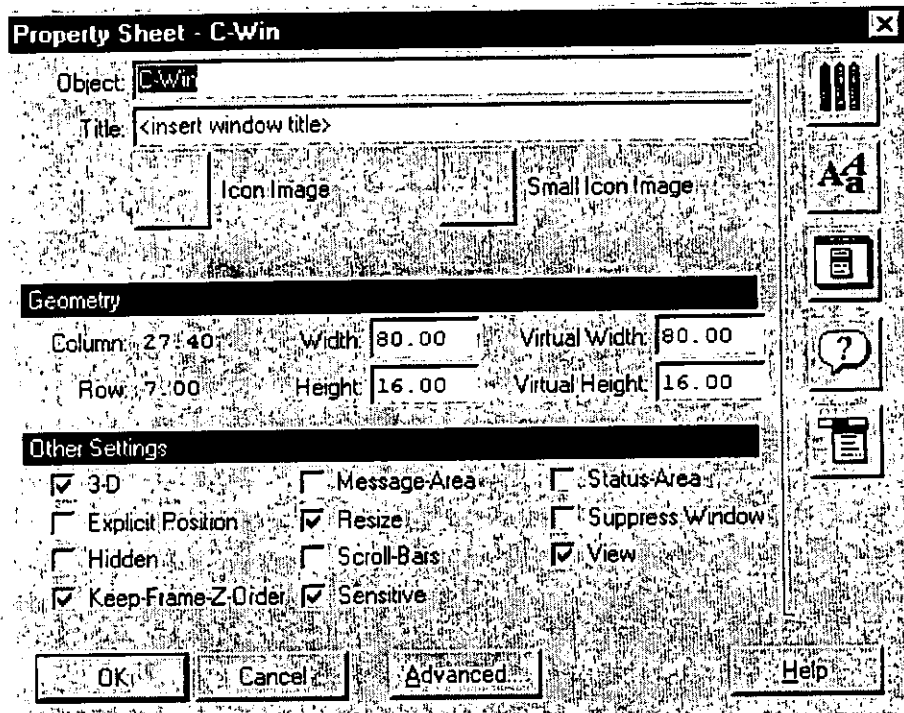


Figura 4.2_3 Propiedades de las Ventanas de Progress

Propiedades

- **Object:** Es el nombre que se le da al objeto. Por "default" (por omisión) Progress define un nombre que empieza con "C-Win", el cual se puede cambiar opcionalmente.
- **Title:** Es el título que se pondrá a la ventana una vez que esté ejecutándose.
- **Iconos:** Son los que se muestran en la parte superior izquierda de la ventana.

- **Geometry:** Es donde se define el tamaño de la ventana. Es decir, se define el ancho y largo de la ventana, así como la posición en la cual se encuentra o se encontrará una vez que se ejecute.
- **Other Settings:** Son las características que va a contener la ventana una vez que este en ejecución, como son: desplegar una barra de ayuda una vez que se estén ejecutando los objetos que posee; que posea una "scroll bar" (barra con desplazamiento); que permita maximizar o minimizar la ventana, o que muestre una ventana de ayuda.
- **Otros botones:** El primer botón (colores) define los colores que se van a usar en el "Foreground" (frente) y el "Background" (fondo) de la ventana. El segundo botón define el tipo de letra que se va a usar para la ventana.

4.2.1.3 "Buttons" (Botones)

Son los encargados de realizar una acción, una vez que se le agrega código dentro de los eventos que posee. Sus propiedades se muestran en la figura 4.2_5.

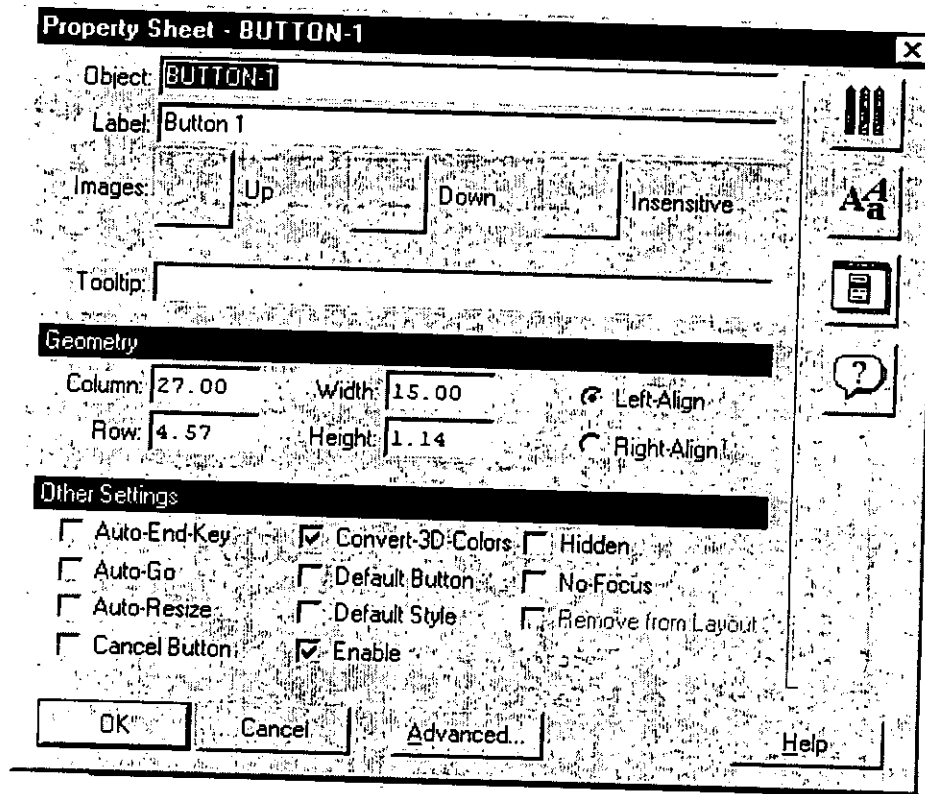


Figura 4.2_5 Propiedades de los Botones

Propiedades

- **Object:** Es el nombre que se asocia al botón. Por default inicia con "BUTTON-n", donde n es el número de botones utilizados. Sin embargo este se puede cambiar.
- **Label:** Es la etiqueta que se colocara al objeto. Por ejemplo "Salir".
- **Images:** Son los "bitmaps" (mapas de bits) que se pueden asociar a este botón.

- **Tooltip:** Indica que cuando el cursor pase sobre este botón se muestre una pequeña línea de ayuda al usuario, también conocida como TAB.
- **Geometry:** Define el tamaño del objeto y su posición dentro de la ventana.
- **Other Settings:** Es donde se especifican las propiedades de habilitar o deshabilitar este botón, que se cierre la ventana una vez que se presione el botón ("Auto-End-Key"), que esté en espera de realizar una acción ("Auto-Go"), o bien que se pueda ajustar automáticamente su tamaño ("Auto-Resize").
- **Otros botones:** El primer botón (colores) define los colores que se van a usar en el frente y el fondo del objeto "button". El segundo botón define su tipo de letra.

4.2.1.4 "Rectangle" (Rectángulo)

Es un cuadro de cierto tamaño dentro de la ventana. Sus propiedades se muestran en la figura 4.2_6.

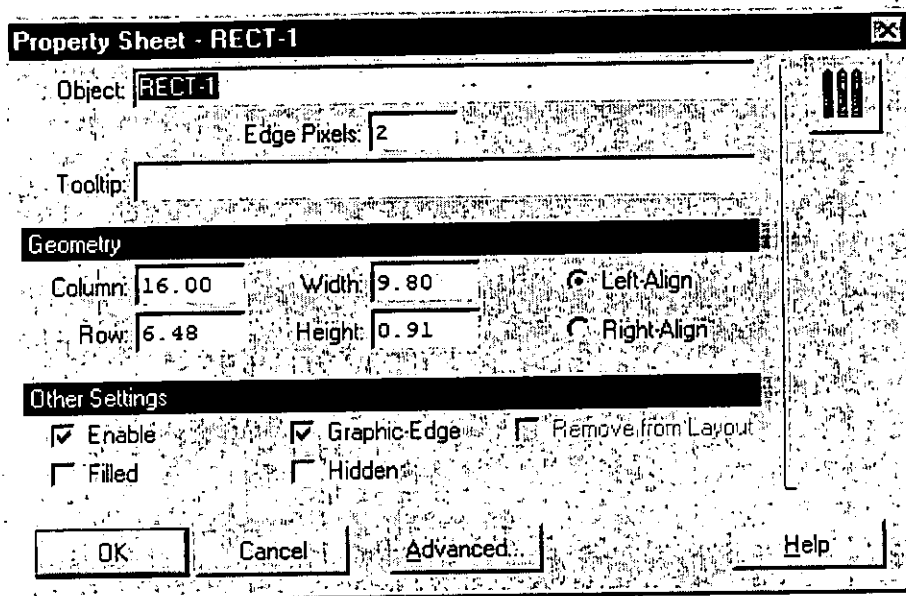


Figura 4.2_6 Propiedades de los Rectángulos

Propiedades

- **Object:** Es el nombre que se asocia al objeto "Rectangle". Por default Progress le pone un nombre que comienza con "RECT-n", donde n indica el número de objetos utilizados de la misma clase. Sin embargo este se puede cambiar.
- **Tooltip:** Indica que cuando el cursor pase sobre el rectángulo se muestre una pequeña línea de ayuda al usuario, también conocida como TAB.
- **Geometry:** Es donde se define el tamaño del objeto (el ancho y largo del rectángulo) así como la posición en la cual se encuentra dentro de la ventana.
- **Other Settings:** Es donde se definen las propiedades de llenado del objeto "rectangle" y la acción de habilitarlo o deshabilitarlo.
- **Otros botones:** El primer y único botón (colores) define los colores que se van a usar en el frente y el fondo del objeto "rectangle".

4.2.1.5 "Fill-In" (Campo de Llenado)

Esta clase de objetos es muy utilizada, debido a que son los campos a capturar por el usuario. Sus propiedades se muestran en la figura 4.2_7.

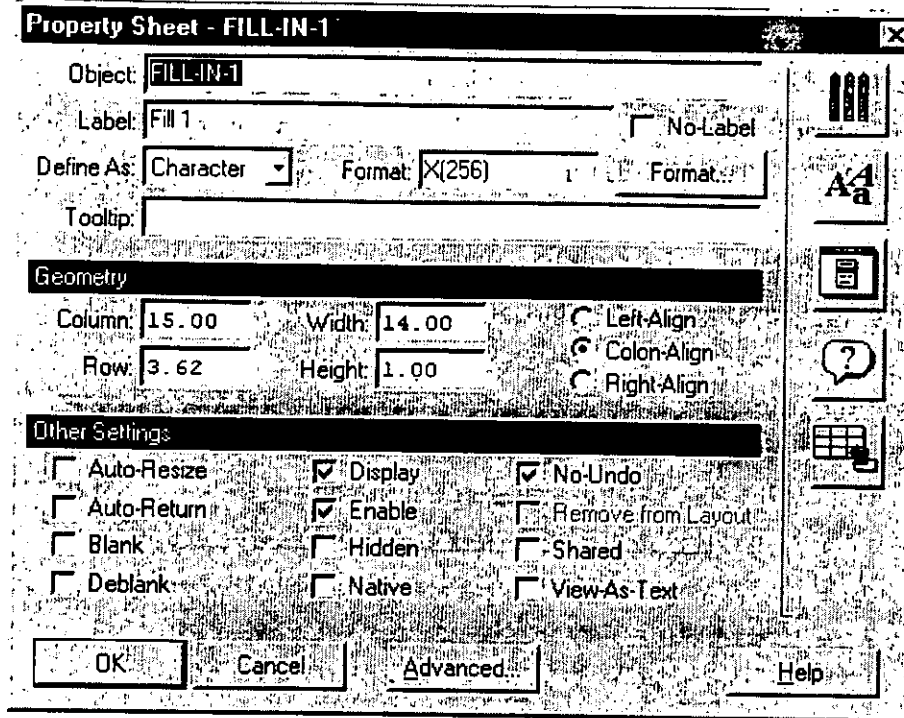


Figura 4.2_7 Propiedades de los Campos "Fill-In" (Campos de Llenado)

Propiedades

- **Object:** Es el nombre que se asocia al objeto "Fill-In". Por default Progress le pone un nombre que comienza con "FILL-IN-n", donde n indica el número de objetos utilizados de la misma clase. Sin embargo este se puede cambiar.
- **Label:** Es la etiqueta que se colocará al objeto, por ejemplo "Fecha corte". Esta se puede deshabilitar.
- **Define As:** Se selecciona el tipo de dato que va a recibir dicho objeto como por ejemplo: importes, fechas, texto, etc.
- **Format:** Es el formato con que se va a permitir capturar la información. Por ejemplo para fechas su formato puede ser "mm/dd/aaaa".
- **Tooltip:** Indica que cuando el cursor pase sobre el campo "Fill-In" se muestre una pequeña línea de ayuda al usuario, también conocida como TAB.
- **Geometry:** Es donde se define el tamaño del objeto (el ancho y largo del campo "Fill-In"), así como la posición en la cual se encuentra dentro de la ventana.
- **Other Settings:** Es donde se definen las propiedades de habilitar o deshabilitar este campo "Fill-In", que no se muestre la información al momento de capturarla, o que solamente se despliegue la información.
- **Otros botones:** El primer botón (colores) define los colores que se van a usar en el frente y el fondo del objeto "Fill-In". El segundo botón define el tipo de letra que se va a usar para el mismo.

4.2.1.6 "Frame" (Marco)

Este es un contenedor que va a alojar a otros objetos. Cuando se crea una ventana automáticamente se le asocia este objeto contenedor. Sus propiedades se muestran en la figura 4.2_8.

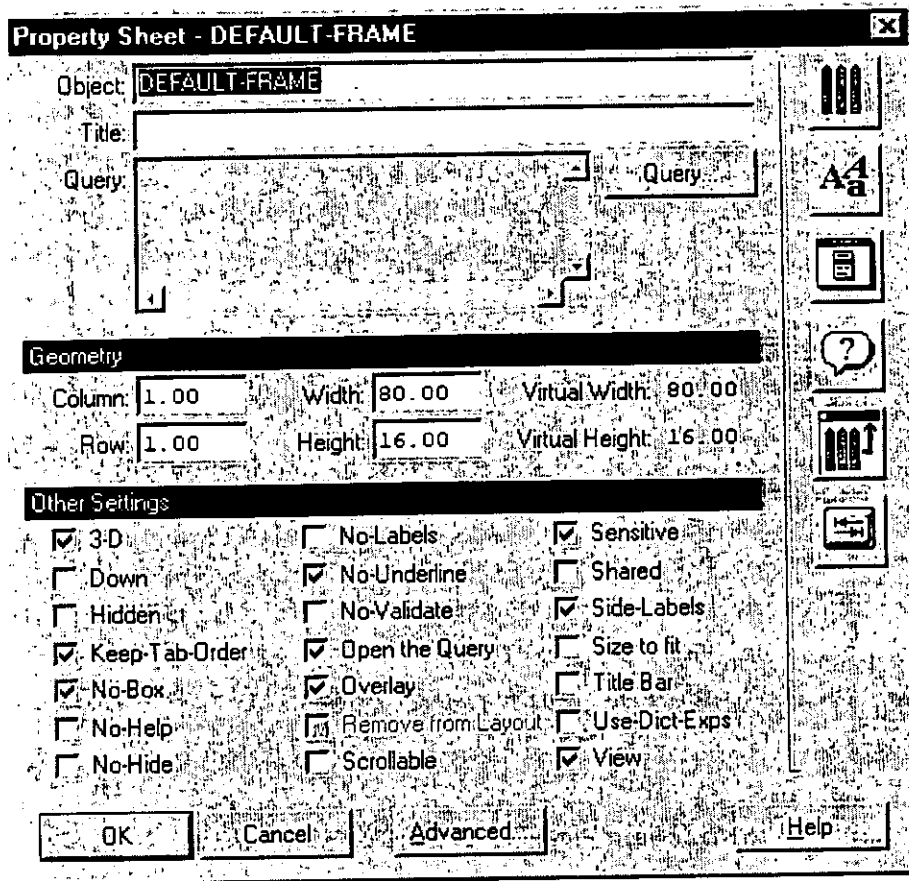


Figura 4.2_8 Propiedades de los "Frames"

Propiedades

- **Object:** Es el nombre que se asocia al objeto "frame". Por default Progress le pone un nombre que comienza con "DEFAULT-FRAME". Este puede ser modificado.
- **Title:** Es el título que se pondrá a este objeto.
- **Other Settings:** Es donde se definen las propiedades de habilitar o deshabilitar este "frame", que se pueda hacer "scroll" (desplazamiento) a la información que se muestre, o bien que se muestre el objeto dentro de un marco.
- **Otros botones:** El primer botón (colores) define los colores que se van a usar en el frente y el fondo del objeto "frame". El segundo botón define el tipo de letra que se va a usar para el mismo.

4.2.1.7 "SmartPanel" (Panel Inteligente)

Son objetos con un comportamiento ya definido por Progress, los cuales tienen la finalidad de poder actualizar las tablas a las que estén ligados dichos objetos, así como poder navegar dentro de la información que contiene cada una de las tablas a las que estén enlazados dicho objetos. La figura 4.2_9 muestra dos objetos "SmartPanel". El primero es un panel de actualización, que realiza modificaciones y actualizaciones directamente en las tablas de la base de datos. El segundo es un panel de navegación, el cual permite navegar dentro de la tabla a la cual esté ligado.

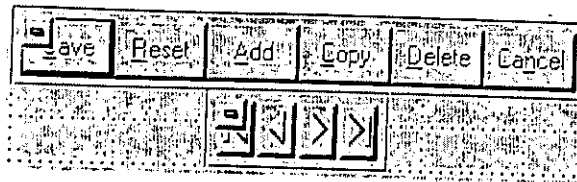


Figura 4.2_9 "SmartPanels" de Actualización y Navegación

4.2.1.8 "SmartViewer" (Visor Inteligente)

El objeto SmartViewer se construye con información de una tabla o bien de diferentes tablas. Una vez que se han seleccionado las tablas a las que hará referencia, se desplegarán los campos de dichas tablas, para que se seleccionen los campos que sean necesarios solo para visualización, o bien los que vayan a sufrir alguna modificación en su contenido. Dentro de la definición del SmartViewer automáticamente se agregarán las validaciones definidas para los campos en el diccionario de datos.

La figura 4.2_10 muestra la creación de un SmartViewer con campos de la tabla U_banco, los cuales se podrán desplegar y/o modificar.

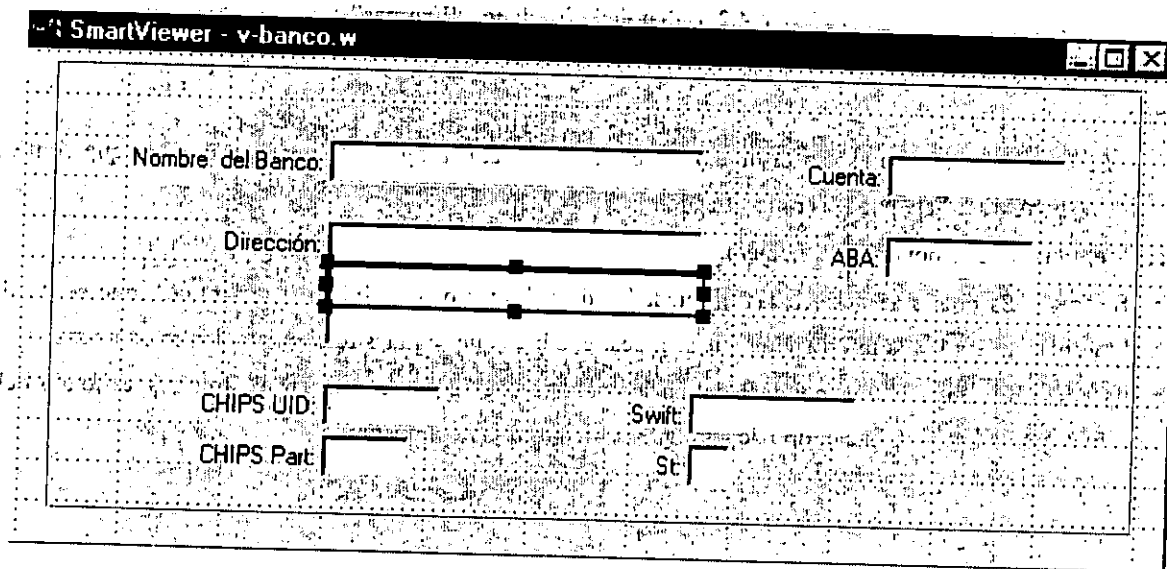


Figura 4.2_10 Creación de un "SmartViewer"

4.2.2 Pantallas de la Aplicación

4.2.2.1 Pantalla del Menú inicial

Esta pantalla muestra las opciones iniciales de que consta el sistema. Véase la figura 4.2_11. En cada una de estas opciones se llama a otras pantallas que a su vez contendrán los submenús de que se compone el sistema. En esta pantalla se utilizaron los objetos text, rectangle, button, frame y window. Cuando el usuario presiona cada uno de los botones estos a su vez mandarán a ejecutar los eventos asociados a cada uno de ellos, por lo que abrirán las ventanas correspondientes. Los usos de cada objeto en este caso son los siguientes.

- **Window.** Es el repositorio principal en donde se encuentran alojados los objetos
- **Frame.** Es el contenedor que esta asociado con la ventana y donde se colocarán todos los objetos que se usen.
- **Text.** Se emplea para colocar una etiqueta que describe el titulo del menú (MENÚ PRINCIPAL).
- **Rectangle.** Se emplea para dibujar un marco que aparece dentro de la ventana.
- **Buttons.** Se emplean para seleccionar cada uno de los submenús cuando dichos botones son presionados.

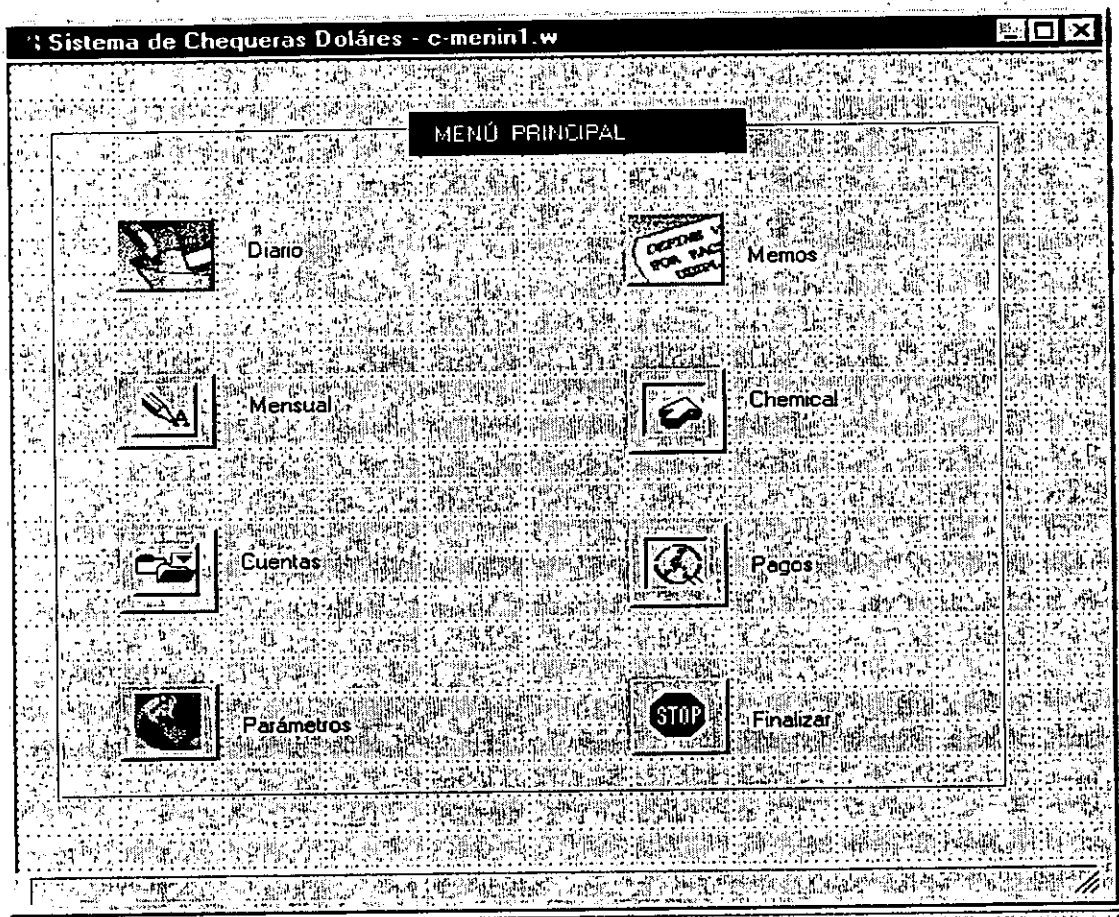


Figura 4.2_11 Pantalla del aMenú Principal del Sistema

4.2.2.2 Pantalla de ABC de Bancos

En esta ventana de "ABC" (altas, bajas y cambios) se podrá actualizar el catalogo de Bancos. Es decir, se darán de alta nuevos registros, se modificarán algunos de sus atributos que ya existen registrados o bien se darán de baja, lo que implica ser eliminados de la tabla en que se encuentran. Cada una de estas opciones vienen ya implementadas por el propio lenguaje Progress, lo que facilita grandemente la creación de la pantalla y sus componentes. Esta pantalla se puede observar en la figura 4.2_12.

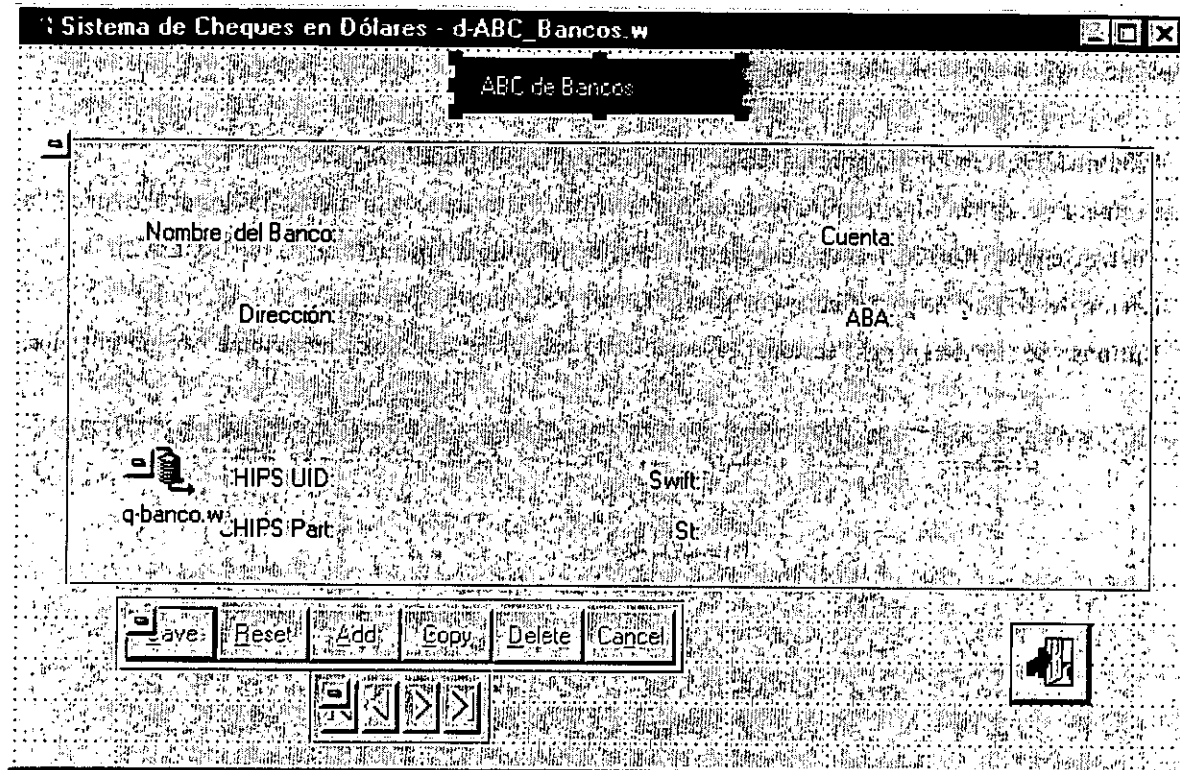


Figura 4.2_12 Pantalla de ABC de Bancos

Las opciones de que consta esta pantalla son: guardar ("save") el registro que se ha dado de alta; actualizar los índices de la tabla ("reset") y actualizar la información que contiene la tabla; agregar un registro para capturar la información que se va a dar de alta en la tabla; hacer una copia ("copy") de un registro dentro de la tabla; borrar ("delete") un registro que se encuentra ya dentro de la tabla, o bien cancelar ("cancel") cualquier modificación que se intente hacer sobre la tabla.

La pantalla también posee una barra de navegación para moverse dentro de la tabla. Es decir se puede posicionar ya sea en el primer registro o el ultimo, así como avanzar o retroceder un registro a la vez.

La pantalla esta compuesta de los siguientes objetos: window, frame, text, rectangle, button, dos SmartPanels, un SmartQuery (consulta inteligente) y un SmartViewer.

Los usos de cada objeto en este caso son los siguientes.

- **Window.** Es el repositorio principal en donde se encuentran alojados los objetos
- **Frame.** Es el contenedor que esta asociado con la ventana y donde se colocarán todos los objetos que se usen.
- **Text.** Se emplea para colocar una etiqueta que describe el titulo del menú (ABC Bancos).
- **Rectangle.** Se emplea para dibujar un marco que aparece dentro de la ventana.
- **Button.** Se emplea para salir de la ventana una vez que sea presionado.
- **SmartPanel.** Son los objetos que dan la funcionalidad de poder navegar dentro de la tabla y de poder hacer operaciones dentro de la misma, es decir agregar información a la tabla, borrar, o modificar la misma.
- **SmartViewer.** Muestra la información de la tabla con la cual está ligado y permite su modificación.
- **SmartQuery.** Es un objeto no visible el cual, cuando se usa el SmartPanel de navegación proporciona al SmartViewer el registro con los campos que este debe desplegar; por lo mismo, además de estar ligado con el SmartViewer lo está con dicho panel de navegación.

4.2.2.3 Pantalla de Consulta de Saldo de Cuenta

En esta pantalla (ver la figura 4.2_13) el usuario digita el número de cuenta y el producto y la pantalla mostrara el saldo disponible y el saldo total que tiene dicha cuenta, así como todos los movimientos que haya tenido el cliente en esa cuenta. Todas estas consultas son realizadas en línea.

The screenshot shows a window titled "Sistema de Cheques en Dólares". It contains the following fields and labels:

- Producto: Cuenta: 11111
- Saldo Disp
- Cliente: Saldo Tot
- Fecha del Movimiento a Cancelar/Modificar:

Below these fields is a table with the following headers:

Interno	Movto	Descripción	Refer	Debits	Credits
[The table body is mostly obscured by noise and is empty of data.]					

At the bottom of the window, there is a status bar that reads "Sistema al que pertenece el cliente" and a small icon.

Figura 4.2_13 Pantalla de Consulta de Saldo de Cuenta

La pantalla de consulta de saldo de cuenta esta compuesta de los siguientes objetos: window, frame, rectangle, button, un "SmartBrowser" (navegador inteligente) y varios campos Fill-in. Los usos de cada objeto en este caso son los siguientes.

- **Window.** Es el repositorio principal en donde se encuentran alojados los objetos.
- **Frame.** Es el contenedor que esta asociado con la ventana y donde se colocarán todos los objetos que se usen.
- **Rectangle.** Se emplea para dibujar un marco que aparece dentro de la ventana.
- **Button.** Se emplea para salir de la ventana una vez que sea presionado.
- **SmartBrowser.** Es el encargado de mostrar a la vez la información de varios registros de la tabla con la cual está ligado, permitiendo también su modificación. Se emplea para desplegar toda la información de los movimientos que se encuentran relacionados con el producto y número de cuenta que fueron digitados por el usuario.
- **Campos Fill_in.** Se emplean para capturar el número de la cuenta que se quiere consultar y el tipo de producto con el que se encuentra relacionada la cuenta. Otros campos Fill-In son empleados para visualizar la información restante de dicha cuenta, como el saldo disponible, el saldo total y el nombre del cliente.

4.3 Generación de Código

En esta sección se muestra la generación de código 4GL para una de las pantallas de la aplicación. En casos como este en que se utilizan los SmartObjects, dicho código se genera en forma automática por Progress sin que el desarrollador codifique ninguna línea de programación.

Debido a esto, en el apéndice D se incluye todo el código fuente en lenguaje 4GL que tuvo que agregarse al código generado por Progress para poder realizar funciones específicas de la aplicación; sin embargo el código generado automáticamente no se muestra ya que es similar al presentado en esta sección y se repite para todos los programas generados de esta forma.

4.3.1 Creación y Ejecución de una Pantalla con SmartObjects

La figura 4.3_1 muestra la construcción de una pantalla usando Progress en modo gráfico con el uso del SmartObjects. Se trata de la pantalla de "ABC" (altas, bajas y cambios) de los tipos de movimientos para el uso de la chequera. Este pantalla es un objeto contenedor que se llama "w-tpmov.w" y contiene los componentes "q-tpmov.w" y "v-tpmov.w". A continuación se describen estos tres componentes.

- q-tpmov.w, es un SmartQuery o componente de Query o consulta, que proporciona los registros de la tabla que debe mostrar el visor v-tpmov.w.
- v-tpmov.w, es un SmartViewer o visor que despliega los campos de la tabla y permite su modificación.
- w-tpmov.w, es un SmartWindow, o la ventana contenedora de los objetos anteriores, así como de los SmartPanels de actualización (Save, Reset, Copy, Add, Delete, Cancel) y de navegación.

La figura 4.3.2 muestra la misma pantalla al ser ejecutada.

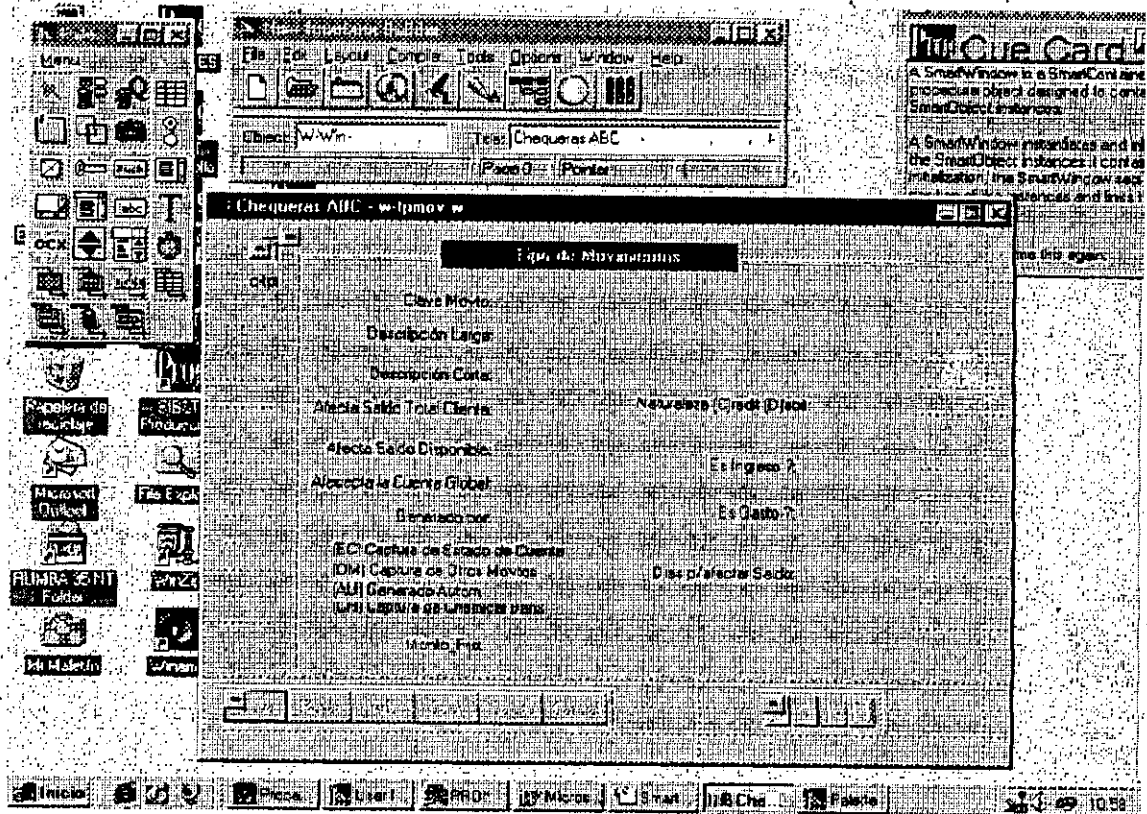


Figura 4.3_1 Creación de la Pantalla de ABC de Movimientos

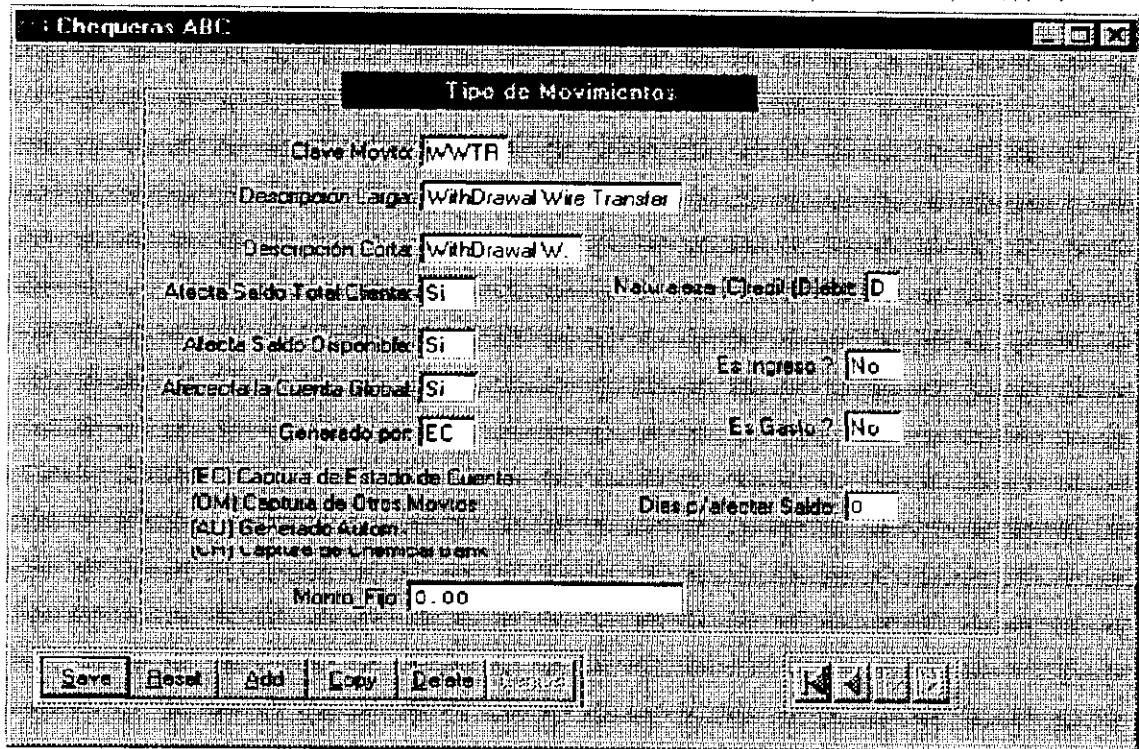


Figura 4.3_2 Ejecución de la Pantalla de ABC de Movimientos

Parte del código fuente generado por Progress para la ventana "w-tpmov.w" se muestra en la figura 4.3_3, en el editor de procedimientos de Progress. En las siguientes dos secciones se lista todo el código generado por Progress para el SmartQuery "q-tpmov.w" y para el SmartViewer "v-tpmov.w".

```

Procedure - C:\pola\grafico\w-tpmov.w
File Edit Search Compile Help

&ANALYZE-SUSPEND _PROCEDURE-SETTINGS
/* Settings for THIS-PROCEDURE
   Type: SmartWindow
   Allow: Basic,Browse,DB-Fields,Query,Smart,Window
*/
&ANALYZE-RESUME _END-PROCEDURE-SETTINGS

/* ***** Create Window ***** */

&ANALYZE-SUSPEND _CREATE-WINDOW
IF SESSION:DISPLAY-TYPE = "GUI":U THEN
  CREATE WINDOW W-Win ASSIGN
    HIDDEN           = YES
    TITLE            = "Chequeras ABC"
    HEIGHT           = 21
    WIDTH            = 113.4
    MAX-HEIGHT       = 29.11
    MAX-WIDTH        = 159.6
    VIRTUAL-HEIGHT   = 29.11
    VIRTUAL-WIDTH    = 159.6
    RESIZE           = no
    SCROLL-BARS      = no
    STATUS-AREA      = no
    BGCOLOR          = ?
    FGCOLOR          = ?
    THREE-D          = yes
    MESSAGE-AREA     = no
    SENSITIVE        = yes.
ELSE {&WINDOW-NAME} = CURRENT-WINDOW.
/* END WINDOW DEFINITION */
&ANALYZE-RESUME
&ANALYZE-SUSPEND _UIB-CODE-BLOCK_XFTR "SmartWindowCues" W-Win_INLINE
/* Actions: adecomm/_so-cue.w ? adecomm/_so-cued.p ? adecomm/_so-cue.w.p */
/* SmartWindow,uib,49271
Destroy on next read */
/* _UIB-CODE-BLOCK-END */
&ANALYZE-RESUME

```

Figura 4.3_3 Código Generado por Progress para la ventana w-tpmov.w

4.3.2 Código Fuente Generado para el SmartQuery "q-tpmov.w"

```

&ANALYZE-SUSPEND _EXPORT-NUMBER UIB_v8r12,
&ANALYZE-RESUME

/* Connected Databases
*/

&Scoped-define WINDOW-NAME W-Win

&ANALYZE-SUSPEND _UIB-PREPROCESSOR-BLOCK

/* ***** Preprocessor Definitions ***** */

&Scoped-define PROCEDURE-TYPE SmartWindow

&Scoped-define ADM-CONTAINER WINDOW

/* Custom List Definitions */
/* List-1,List-2,List-3,List-4,List-5,List-6 */

/* _UIB-PREPROCESSOR-BLOCK-END */
&ANALYZE-RESUME

/* ***** Control Definitions ***** */

/* Define the widget handle for the window */
DEFINE VAR W-Win AS WIDGET-HANDLE NO-UNDO.

/* Definitions of handles for SmartObjects */
DEFINE VARIABLE h_q-tpmov AS HANDLE NO-UNDO.

/* ***** Frame Definitions ***** */

/* ***** Runtime Attributes and UIB Settings ***** */

&ANALYZE-SUSPEND _RUN-TIME-ATTRIBUTES
/* SETTINGS FOR WINDOW W-Win
VISIBLE,,RUN-PERSISTENT */
IF SESSION:DISPLAY-TYPE = "GUI":U AND VALID-HANDLE(W-Win)
THEN W-Win:HIDDEN = yes.

```

```
/* _RUN-TIME-ATTRIBUTES-END */
&ANALYZE-RESUME

&ANALYZE-SUSPEND _EXPORT-CODE-BLOCK _ADM-CREATE-OBJECTS
PROCEDURE adm-create-objects :
  DEFINE VARIABLE adm-current-page AS INTEGER NO-UNDO.

  RUN get-attribute IN THIS-PROCEDURE ('Current-Page':U).
  ASSIGN adm-current-page = INTEGER(RETURN-VALUE).

  CASE adm-current-page:
    WHEN ? THEN DO:
      RUN init-object IN THIS-PROCEDURE (
        INPUT 'q-tpmov.w':U,
        INPUT FRAME ?,
        INPUT ":U,
        OUTPUT h_q-tpmov ).
      RUN set-position IN h_q-tpmov ( 1.53 , 6.00 ) NO-ERROR.
      /* Size in UIB: ( 2.26 , 11.60 ) */

      /* Adjust the tab order of the smart objects. */
      RUN adjust-tab-order IN adm-broker-hdl ( h_q-tpmov ,
        h_p-navico , 'AFTER':U ).
      END. /* Page 0 */

  END CASE.

END PROCEDURE.
&ANALYZE-RESUME
```

4.3.3 Código Fuente Generado para el SmartViewer "v-tpmov.w"

```

&ANALYZE-SUSPEND _EXPORT-NUMBER UIB_v8r12
&ANALYZE-RESUME
/* Connected Databases
*/
&Scoped-define WINDOW-NAME W-Win

&ANALYZE-SUSPEND _UIB-PREPROCESSOR-BLOCK

/* ***** Preprocessor Definitions ***** */

&Scoped-define PROCEDURE-TYPE SmartWindow

&Scoped-define ADM-CONTAINER WINDOW

/* Custom List Definitions */
/* List-1,List-2,List-3,List-4,List-5,List-6 */

/* _UIB-PREPROCESSOR-BLOCK-END */
&ANALYZE-RESUME

/* ***** Control Definitions ***** */

/* Define the widget handle for the window */
DEFINE VAR W-Win AS WIDGET-HANDLE NO-UNDO.

/* Definitions of handles for SmartObjects */
DEFINE VARIABLE h_v-tpmov AS HANDLE NO-UNDO.

/* ***** Frame Definitions ***** */

/* ***** Runtime Attributes and UIB Settings ***** */

&ANALYZE-SUSPEND _RUN-TIME-ATTRIBUTES
/* SETTINGS FOR WINDOW W-Win
VISIBLE,,RUN-PERSISTENT */
IF SESSION:DISPLAY-TYPE = "GUI":U AND VALID-HANDLE(W-Win)
THEN W-Win:HIDDEN = yes.

```

```
/* _RUN-TIME-ATTRIBUTES-END */
&ANALYZE-RESUME

&ANALYZE-SUSPEND _EXPORT-CODE-BLOCK _ADM-CREATE-OBJECTS
PROCEDURE adm-create-objects :
  DEFINE VARIABLE adm-current-page AS INTEGER NO-UNDO.

  RUN get-attribute IN THIS-PROCEDURE ('Current-Page':U).
  ASSIGN adm-current-page = INTEGER(RETURN-VALUE).

  CASE adm-current-page:

    WHEN ? THEN DO:
      RUN init-object IN THIS-PROCEDURE (
        INPUT 'v-tpmov.w':U ,
        INPUT FRAME ? ,
        INPUT 'Layout = ':U ,
        OUTPUT h_v-tpmov ).
      RUN set-position IN h_v-tpmov ( 1.26 , 11.00 ) NO-ERROR.
      /* Size in UIB: ( 17.37 , 90.00 ) */

      /* Adjust the tab order of the smart objects. */
      END. /* Page 0 */

    END CASE.

  END PROCEDURE.
&ANALYZE-RESUME
```


4.4 Diseño y Estrategia de Pruebas

4.4.1 Fundamentos de las Pruebas de Software

En el momento en que se pone un sistema en operación para que pueda ser utilizado por el cliente, se presentan diferentes dudas:

- ¿ El sistema hace los que se pidió?
- ¿ Soportará la carga de trabajo planeada?
- ¿ El sistema puede fallar?

Estas son las preguntas más comunes que se plantean al momento de instalar un sistema por primera vez. Es por lo anterior que se realizan pruebas que disminuirán los riesgos de falla del sistema. La prueba es uno de los pasos de la ingeniería de software que se puede ver como destructivo en lugar de constructivo.

Objetivo y reglas de las pruebas

- La prueba es un proceso de ejecución de un programa con la intención de descubrir un error.
- Un buen caso de prueba es aquel que tiene una alta probabilidad de mostrar un error no descubierto hasta entonces.
- Una prueba tiene éxito si descubre un error no detectado hasta entonces.
- El objetivo es diseñar pruebas que sistemáticamente saquen a la luz diferentes clases de errores, haciéndolo con la menor cantidad de tiempo y esfuerzo posible.

Las pruebas también demuestran hasta qué punto el software cumple con las especificaciones y requerimientos de rendimiento planteados desde el análisis. La prueba no puede asegurar la ausencia de defectos, sólo puede demostrar que existen defectos en el software.

Las pruebas tienen como objetivo principal el de entregar un producto de calidad, que ahorre costos de tiempo, esfuerzo y dinero al momento de ser implantado, así como al efectuarse un menor mantenimiento del sistema.

4.4.2 Pasos en la Prueba de los Sistemas de Software

Los documentos que se utilizan para la fase de pruebas son:

- Especificación de los requerimientos del software.
- Especificación del diseño y el código fuente.
- Plan y procedimientos de prueba.

Se realizan los diferentes tipos de pruebas en el siguiente orden:

- Pruebas de unidad.
- Pruebas de integración.
- Pruebas de Validación y de Sistema.

Para cada tipo de prueba se llevan a cabo los siguientes pasos:

- Se realiza la prueba y se evalúan los resultados.
- Se comparan los resultados de la prueba con los esperados.
- Cuando se descubren errores se comienza a depurar el sistema.
- Un error puede llevarse un minuto, una hora, un día o un mes en corregirlo. Esto hace difícil planificar la prueba de forma eficaz.

Si el funcionamiento del SW parece ser correcto y los errores que se encuentran son fácilmente corregibles, se puede obtener alguna de las siguientes conclusiones:

- La calidad y la fiabilidad del SW son aceptables.
- Las pruebas son inadecuadas para descubrir serios errores.
- Si la prueba no descubre errores, quedará la sospecha de que no se ha pensado cuidadosamente la configuración de la prueba y de que los errores están escondidos en el SW.

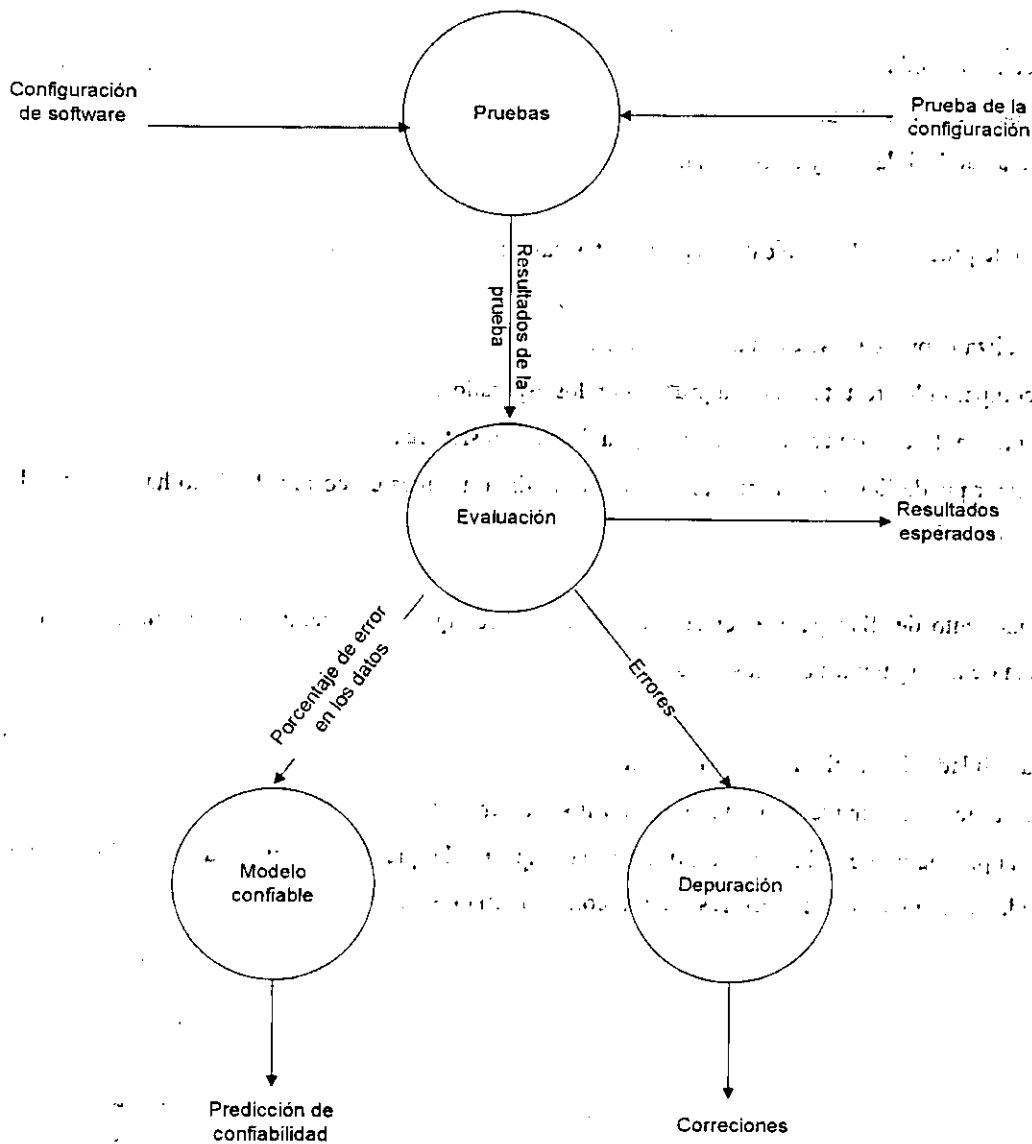


Figura 4.4_1 Flujo de Información para las Pruebas

Un plan de pruebas bien elaborado dará como resultado la detección de errores y de esta manera se procederá a la depuración de los mismos. El plan debe indicar el nombre de la prueba, los datos de entrada, el objetivo de la prueba y los datos de salida.

En la figura 4.4_1 se tienen dos entradas: la configuración del software que incluye la especificación de requisitos, la especificación del diseño y el código fuente, la configuración de prueba que incluye un plan y un procedimiento de prueba así como los casos de prueba a utilizar y los resultados esperados. Se realizan las pruebas y se evalúan los resultados esperados contra los generados. Cuando se descubren errores se inicia el proceso de depuración.

A continuación en el apartado 4.4.3 se presentan los diferentes tipos de pruebas que se tienen y en los apartados 4.4.4 y 4.4.5 se presentan las estrategias de pruebas que se diseñaron para el sistema de cheques en dólares. Finalmente, en la siguiente sección (4.5) se reportan los resultados de las pruebas realizadas siguiendo dichas estrategias.

4.4.3 Diferentes Tipos de Pruebas

4.4.3.1 Pruebas Unitarias y de Integración

La prueba unitaria o de unidad se realiza al haber terminado de codificar un módulo y pretende descubrir los errores inherentes a ese módulo. Dentro de este tipo de prueba normalmente se tienen las pruebas de "Caja Blanca" y las pruebas de interfase gráfica.

La prueba de integración se realiza al juntar varios módulos y su objetivo será descubrir los errores que surgieron al integrar dichos módulos. Para este tipo de pruebas existen dos formas de atacarlas: en forma ascendente y en forma descendente. Dentro de este tipo de pruebas normalmente se tienen las pruebas de "Caja Negra".

Pruebas de Caja Blanca

- Es un método de diseño de casos de prueba que usa la estructura de control del diseño de procedimientos ("procedural") para derivar los casos de prueba.
- Mediante los métodos de prueba de la caja blanca se pueden derivar casos de prueba que garanticen que:
 1. Se ejercitan por los menos una vez todos los caminos independientes de cada módulo.
 2. Se realizan todas las decisiones lógicas en sus valores verdadero y falso.
 3. Se ejecutan todas las iteraciones en sus límites y con sus límites operacionales.
 4. Se llenan las estructuras de datos internas para asegurar su validez.

Pruebas de Interfase Gráfica

Durante estas pruebas se comprueba que la información fluya de forma adecuada hacia y desde la unidad o programa que está siendo probado. Se examinan las estructuras de los datos locales para asegurar que los datos que se mantienen temporalmente conservan su integridad durante todos los pasos de ejecución. Se prueban las condiciones límite para asegurar que el programa funciona correctamente en los límites establecidos como restricciones de procesamiento.

Pruebas de Caja Negra

Entre los tipos de pruebas de "Caja Negra" se tienen:

- **Pruebas de Trayectoria Normal**

Se derivan del diccionario de datos. Se hace una prueba por cada flujo de datos que entre al sistema (lo que ve el usuario). Cada prueba se hace con un número válido de entradas que estén de acuerdo con lo definido en el

diccionarios de datos. Para cada ítem se toman tres valores: los extremos y un valor medio. Para cada flujo de datos se buscan las permutaciones de ítems que sean válidas.

- **Pruebas de Trayectorias de Excepción**

Se intenta generar un conjunto de entradas para probar la forma del manejo de errores. Se seleccionan valores inválidos para cada ítem. Estos valores se usan junto con los flujos de datos de la prueba normal. Cada prueba de excepción involucrará un solo ítem incorrecto.

- **Pruebas de Estado Transitorio**

En este tipo de prueba se trata de introducir secuencias de flujo de datos (altas, bajas, cambios, etc.). Los flujos de datos para crear estas secuencias se pueden obtener de los creados durante las pruebas normales.

4.4.3.2 Pruebas de Sistema

La prueba del sistema, realmente, está constituida por una serie de pruebas diferentes cuyo propósito primordial es probar profundamente el sistema. Aunque cada prueba tiene un propósito diferente, todas trabajan para verificar que se han integrado adecuadamente todos los elementos del sistema y que realizan las funciones apropiadas. Estas pruebas incluyen:

- Pruebas de resistencia
- Pruebas de rendimiento
- Pruebas de seguridad
- Pruebas de recuperación
- Pruebas especiales.

Pruebas de Resistencia

Las pruebas de resistencia ejecutan un sistema en una forma que demande recursos en cantidad, frecuencia o volumen anormales. Por ejemplo, se puede incrementar las frecuencias de entradas de datos para evaluar como responden las funciones de entrada, realizar pruebas que requieran un máximo de memoria u otros recursos, hacer pruebas en varios sistemas operativos, o pruebas que requieran excesivas búsquedas de datos en disco.

Pruebas de Rendimiento

Con estas pruebas se mide el sistema en términos del rendimiento esperado en el documento de objetivos. Se hará una prueba por cada factor de rendimiento a evaluar (tiempo de respuesta, capacidad de volumen, tiempo de CPU, espacio en disco, etc.).

Se asegura el rendimiento desde los módulos individuales hasta la integración total de los mismos, que es realmente el momento en que se puede evaluar el rendimiento total del sistema. Se pueden medir los ciclos de procesador así como el tiempo que los módulos tardan en realizarse.

Pruebas de Seguridad

Las pruebas de seguridad buscan verificar que los mecanismos de protección incluidos en el sistema de hecho lo protegerán de una penetración inapropiada.

Pruebas de Recuperación

Son pruebas que fuerzan al Software a fallar en una variedad de formas y verifican que la recuperación se realiza apropiadamente.

Pruebas Especiales

Son todas aquellas pruebas, que no estando dentro de las anteriormente descritas, dependen de la naturaleza del sistema, como lo pueden ser: puntos de reinicio (simulación de fallas en el equipo), reprocesamiento y recuperación.

4.4.3.3 Pruebas de Validación

El propósito de la prueba de validación (también llamada de aceptación) es:

- Verificar que el software desarrollado funciona de acuerdo a las expectativas razonables del cliente.
- Evitar sorpresas cuando un sistema pasa a la fase de implantación.
- Evaluar el trabajo de análisis, no del diseño ni de la implementación.

El resultado de esta prueba es binario: se acepta o se rechaza el sistema. No forma parte de un proceso de corrección de errores. La prueba de validación se deriva única y exclusivamente del documento de objetivos. Dentro de las pruebas de validación se tienen las pruebas alfa y beta.

Pruebas Alfa

Para llevar a cabo la prueba alfa se invita a un usuario al lugar del desarrollo y en presencia del desarrollador, este registro errores y problemas de uso. Es decir, todo esto se lleva a cabo en un entorno controlado, en el área de desarrollo.

Prueba Beta

Para llevar a cabo la prueba beta se lleva el sistema a un lugar donde el encargado del desarrollo no esté presente y donde estará funcionando el sistema.

4.4.4 Diseño de Pruebas Unitarias

Como se explicó en las diferentes secciones que hablan de las facilidades de Progress para desarrollo gráfico de aplicaciones, los componentes gráficos de la arquitectura ADM (“**Application Development Model**”) de Progress comprenden los componentes estándar y los “**SmartObjects**” y tienen ya integradas las siguientes funciones que son las más comunes en una aplicación gráfica. A continuación se listan ambos tipos de componentes.

- Componentes Gráficos Estándar:

- "Browsers" o navegadores.
- "Viewers" o visores.
- "Querys" o consultas.
- Campos "Fill-In" para despliegue y asignación de campos de tablas y variables.
- Botones.
- Campos de texto.
- Rectángulos.
- Contenedores para elementos como los anteriores:
 - "Windows" o ventanas.
 - "Frames" o marcos.

- SmartObjects:

- "SmartBrowsers" o navegadores.
- "SmartViewers" o visores.
- "SmartQuerys" o consultas.
- "SmartPanels" de navegación en tablas.
- "SmartPanels" de actualización, para agregar, borrar y cambiar registros de tablas.
- "SmartFolders" para mostrar diferentes páginas en cada pestaña.
- "SmartContainers":
 - "SmartWindows" o ventanas.
 - "SmartDialogs" o ventanas de diálogo con botones estándar para Aceptar y Salir.

Los componentes gráficos de ADM son elementos estándar del sistema de desarrollo de Progress, cuyo código 4GL ya ha sido diseñado y probado como parte de su integración en Progress y dicho código solo requiere ser modificado o complementado para realizar algún proceso específico que no contemple ADM. Los SmartObjects son componentes gráficos similares a los componentes estándares, pero ya incluyen el código necesario para realizar todas sus funciones. En el proyecto que nos ocupa, el código adicional (en lenguaje 4GL) se incluyó para iniciar y controlar la ejecución de diversas rutinas utilizando botones gráficos.

Por lo anterior, se dividieron los procedimientos de la aplicación en dos tipos:

- Módulos que solo tienen código estándar generado por ADM de Progress.
- Módulos que además tienen código 4GL insertado para algún proceso específico.

A continuación se muestra parte del código estándar generado por Progress y del código que se agregó (el cual se indica en negritas precedido por la línea "Código Adicional") para el módulo c_genmovcierre.w (en el apéndice E se presenta el código 4GL adicional de todos aquellos módulos en que se requirió).

4.4.4.1 Código 4GL (Generado por Progress y Adicional) de un Módulo

```

&ANALYZE-SUSPEND _VERSION-NUMBER UIB_v8r12 GUI
&ANALYZE-RESUME
/* Connected Databases
    cheques    PROGRESS
*/
&Scoped-define WINDOW-NAME C-Win
&ANALYZE-SUSPEND _UIB-CODE-BLOCK _CUSTOM_DEFINITIONS C-Win
/*-----
File:
Description:
Input Parameters:
    <none>
Output Parameters:
    <none>
Author:
Created:
-----*/
/*      This .W file was created with the Progress UIB.      */
/*-----*/
/* Create an unnamed pool to store all the widgets created
   by this procedure. This is a good default which assures
   that this procedure's triggers and internal procedures
   will execute in this procedure's storage, and that proper
   cleanup will occur on deletion of the procedure. */

.CREATE WIDGET-POOL.

/* ***** Definitions ***** */

DEFINE NEW SHARED VARIABLE P_prod    LIKE U_cliente.id_producto.
DEFINE NEW SHARED VARIABLE P_cuenta  LIKE U_cliente.id_cliente.
DEFINE NEW SHARED VARIABLE P_F_inicial AS DATE FORMAT "99/99/9999".

(Código adicional)
DEFINE BUFFER cta_global FOR U_cliente.

/* Parameters Definitions --- */
/* Local Variable Definitions --- */
/* _UIB-CODE-BLOCK-END */
&ANALYZE-RESUME

&ANALYZE-SUSPEND _UIB-PREPROCESSOR-BLOCK

/* ***** Preprocessor Definitions ***** */

&Scoped-define PROCEDURE-TYPE Window

```


CAPITULO 4. DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA

```
/* Name of first Frame and/or Browse and/or first Query
&Scoped-define FRAME-NAME DEFAULT-FRAME

/* Internal Tables (found by Frame, Query & Browse Queries)
&Scoped-define INTERNAL-TABLES Q_Corte Q_Tasa

/* Definitions for FRAME DEFAULT-FRAME
&Scoped-define OPEN-QUERY-DEFAULT-FRAME OPEN QUERY DEFAULT-FRAME FOR EACH Q_Corte SHARE-
LOCK, ~
    EACH Q_Tasa WHERE TRUE /* Join to Q_Corte incomplete */ SHARE-LOCK.
&Scoped-define TABLES-IN-QUERY-DEFAULT-FRAME Q_Corte Q_Tasa
&Scoped-define FIRST-TABLE-IN-QUERY-DEFAULT-FRAME Q_Corte

/* Standard List Definitions
&Scoped-Define ENABLED-OBJECTS RECT-4 RECT-5 k_ano_mes_corte k_tasa_Fisica ~
k_Tasa_Moral BUTTON-2 BUTTON-3
&Scoped-Define DISPLAYED-OBJECTS k_ano_mes_corte k_tasa_Fisica ~
k_Tasa_Moral

/* Custom List Definitions
/* List-1,List-2,List-3,List-4,List-5,List-6

/* _UIB-PREPROCESSOR-BLOCK-END */
&ANALYZE-RESUME

/* ***** Control Definitions ***** */

/* Define the widget handle for the window
DEFINE VAR C-Win AS WIDGET-HANDLE NO-UNDO.

/* Definitions of the field level widgets
DEFINE BUTTON BUTTON-2 AUTO-GO
    IMAGE-UP FILE "adeicon\run":U
    LABEL "Button 2"
    SIZE 8 BY 2.1.

DEFINE VARIABLE k_ano_mes_corte AS INTEGER FORMAT ">>>>>>".INITIAL 0.
    LABEL "Periodo( aaaa,mm)"
    VIEW-AS FILL-IN
    SIZE 14 BY 1 NO-UNDO.

DEFINE RECTANGLE RECT-4
    EDGE-PIXELS 2 GRAPHIC-EDGE NO-FILL
    SIZE 101 BY 11.52.

/* Query definitions
&ANALYZE-SUSPEND
DEFINE QUERY DEFAULT-FRAME FOR
    Q_Corte,
    Q_Tasa SCROLLING.
&ANALYZE-RESUME
```

```
/* ***** Frame Definitions ***** */
```

```
DEFINE FRAME DEFAULT-FRAME
```

```
  k_ano_mes_corte AT ROW 5.19 COL 39 COLON-ALIGNED HELP
    "Debe gitar mes Año y Mes" AUTO-RETURN
  k_tasa_Fisica AT ROW 6.81 COL 39 COLON-ALIGNED HELP
    "Tasa de interes a la Persona Física" AUTO-RETURN
  k_Tasa_Moral AT ROW 8.62 COL 39 COLON-ALIGNED HELP
    "Tasa de Intereses a Persona Moral" AUTO-RETURN
  BUTTON-2 AT ROW 14.33 COL 90 HELP
    "Esta Seguro de Iniciar el Proceso"
  BUTTON-3 AT ROW 14.33 COL 99 HELP
    "Salir de este Proceso"
  RECT-4 AT ROW 2.33 COL 7
    " Generar Movimientos de Cierre del Mes" VIEW-AS TEXT
    SIZE 42 BY 1.05 AT ROW 1.95 COL 36
    BGCOLOR 9 FGColor 14
```

```
/* ***** Procedure Settings ***** */
```

```
&ANALYZE-SUSPEND _PROCEDURE-SETTINGS
```

```
/* Settings for THIS-PROCEDURE
```

```
  Type: Window
```

```
  Allow: Basic,Browse,DB-Fields,Window,Query
```

```
  Other Settings: COMPILE
```

```
*/
```

```
&ANALYZE-RESUME _END-PROCEDURE-SETTINGS
```

```
/* ***** Create Window ***** */
```

```
&ANALYZE-SUSPEND _CREATE-WINDOW
```

```
IF SESSION:DISPLAY-TYPE = "GUI":U THEN
```

```
  CREATE WINDOW C-Win ASSIGN
```

```
    HIDDEN          = YES
```

```
    TITLE           = "Sistema de Chequera en Dólares"
```

```
    HEIGHT          = 16.43
```

```
    WIDTH           = 111.4
```

```
ELSE {&WINDOW-NAME} = CURRENT-WINDOW.
```

```
/* END WINDOW DEFINITION
```

```
*/
```

```
&ANALYZE-RESUME
```

```
/* ***** Runtime Attributes and UIB Settings ***** */
```

```
&ANALYZE-SUSPEND _RUN-TIME-ATTRIBUTES
```

```
/* SETTINGS FOR WINDOW C-Win
```

```
  VISIBLE,,RUN-PERSISTENT
```

```
*/
```

```
/* SETTINGS FOR FRAME DEFAULT-FRAME
```

```
*/
```

(Código Adicional)

ASSIGN

k_anio_mes_corte:AUTO-RESIZE IN FRAME DEFAULT-FRAME = TRUE

k_anio_mes_corte:PRIVATE-DATA IN FRAME DEFAULT-FRAME =

"k_anio_mes_corte = t_periodo."

IF SESSION:DISPLAY-TYPE = "GUI":U AND VALID-HANDLE(C-Win)

THEN C-Win:HIDDEN = no.

/* _RUN-TIME-ATTRIBUTES-END */

&ANALYZE-RESUME

/* Setting information for Queries and Browse Widgets fields */

&ANALYZE-SUSPEND _QUERY-BLOCK FRAME DEFAULT-FRAME

/* Query rebuild information for FRAME DEFAULT-FRAME

_TblList = "cheques.Q_Corte,cheques.Q_Tasa WHERE cheques.Q_Corte ..."

_Query is OPENED

/ / FRAME DEFAULT-FRAME */

&ANALYZE-RESUME

/* ***** Control Triggers ***** */

&Scoped-define SELF-NAME C-Win

&ANALYZE-SUSPEND _UIB-CODE-BLOCK _CONTROL C-Win C-Win

ON END-ERROR OF C-Win /* Sistema de Chequera en Dólares */

OR ENDKEY OF {&WINDOW-NAME} ANYWHERE DO:

/* This case occurs when the user presses the "Esc" key.

In a persistently run window, just ignore this. If we did not, the
application would exit. */

IF THIS-PROCEDURE:PERSISTENT THEN RETURN NO-APPLY.

END.

/* _UIB-CODE-BLOCK-END */

&ANALYZE-RESUME

&ANALYZE-SUSPEND _UIB-CODE-BLOCK _CONTROL C-Win C-Win

ON WINDOW-CLOSE OF C-Win /* Sistema de Chequera en Dólares */

DO:

/* This event will close the window and terminate the procedure. */

APPLY "CLOSE":U TO THIS-PROCEDURE.

RETURN NO-APPLY.

END.

/* _UIB-CODE-BLOCK-END */

&ANALYZE-RESUME

&Scoped-define SELF-NAME BUTTON-2

&ANALYZE-SUSPEND _UIB-CODE-BLOCK _CONTROL BUTTON-2 C-Win

(Código Adicional)

ON CHOOSE OF BUTTON-2 IN FRAME DEFAULT-FRAME /* Button 2 */

DO:

IF k_tasa_moral = 0 OR k_tasa_fisica = 0 OR k_anio_mes_corte = 0 THEN

```

MESSAGE "FINALIZO EL PROCESO".
APPLY 'CLOSE' TO THIS-PROCEDURE.
END.

```

```

/* _UIB-CODE-BLOCK-END */
&ANALYZE-RESUME

```

```

&Scoped-define SELF-NAME k_anio_mes_corte
&ANALYZE-SUSPEND _UIB-CODE-BLOCK_CONTROL k_anio_mes_corte C-Win
ON LEAVE OF k_anio_mes_corte IN FRAME DEFAULT-FRAME /* Periodo( aaaa,mm) */
(Código Adicional)
DO:

```

```

    k_anio_mes_corte = INPUT k_anio_mes_corte.
    t_periodo       = k_anio_mes_corte.
    DISPLAY k_anio_mes_corte
        WITH frame default-frame.

```

```

END.

```

```

/* _UIB-CODE-BLOCK-END */
&ANALYZE-RESUME

```

```

&Scoped-define SELF-NAME k_Tasa_Moral
&ANALYZE-SUSPEND _UIB-CODE-BLOCK_CONTROL k_Tasa_Moral C-Win
ON LEAVE OF k_Tasa_Moral IN FRAME DEFAULT-FRAME /* Tasa p/Persona Moral */
(Código Adicional)
DO:

```

```

    k_tasa_moral = INPUT k_tasa_moral.

```

```

END.

```

```

/* _UIB-CODE-BLOCK-END */
&ANALYZE-RESUME

```

```

&UNDEFINE SELF-NAME

```

```

&ANALYZE-SUSPEND _UIB-CODE-BLOCK_CUSTOM_MAIN-BLOCK C-Win

```

```

/* ***** Main Block ***** */

```

```

/* Set CURRENT-WINDOW: this will parent dialog-boxes and frames. */
ASSIGN CURRENT-WINDOW = {&WINDOW-NAME}
THIS-PROCEDURE:CURRENT-WINDOW = {&WINDOW-NAME}.

```

```

/* The CLOSE event can be used from inside or outside the procedure to */
/* terminate it. */
ON CLOSE OF THIS-PROCEDURE
    RUN disable_UI.

```

```

/* Best default for GUI applications is... */
PAUSE 0 BEFORE-HIDE.

```

```

/* Now enable the interface and wait for the exit condition. */

```

```

/* (NOTE: handle ERROR and END-KEY so cleanup code will always fire. */
MAIN-BLOCK:
DO ON ERROR UNDO MAIN-BLOCK, LEAVE MAIN-BLOCK
ON END-KEY UNDO MAIN-BLOCK, LEAVE MAIN-BLOCK:
RUN enable_UI.
IF NOT THIS-PROCEDURE:PERSISTENT THEN
WAIT-FOR CLOSE OF THIS-PROCEDURE.

```

(Código Adicional)

```

<FIND FIRST Q_Corte USE-INDEX anio_mes NO-ERROR.
IF AVAILABLE Q_Corte THEN DO:

```

END.

```

/* _UIB-CODE-BLOCK-END */
&ANALYZE-RESUME

```

```

/* ***** Internal Procedures ***** */

```

```

&ANALYZE-SUSPEND _UIB-CODE-BLOCK _PROCEDURE disable_UI C-Win _DEFAULT-DISABLE
PROCEDURE disable_UI:

```

```

/*-----
Purpose:  DISABLE the User Interface
Parameters: <none>
Notes:   Here we clean-up the user-interface by deleting
dynamic widgets we have created and/or-hide
frames. This procedure is usually called when
we are ready to "clean-up" after running.
-----*/

```

```

/* Delete the WINDOW we created */
IF SESSION:DISPLAY-TYPE = "GUI":U AND VALID-HANDLE(C-Win)
THEN DELETE WIDGET C-Win.
IF THIS-PROCEDURE:PERSISTENT THEN DELETE PROCEDURE THIS-PROCEDURE.
END PROCEDURE.

```

```

/* _UIB-CODE-BLOCK-END */
&ANALYZE-RESUME

```

```

&ANALYZE-SUSPEND _UIB-CODE-BLOCK _PROCEDURE enable_UI C-Win _DEFAULT-ENABLE
PROCEDURE enable_UI:

```

```

/*-----
Purpose:  ENABLE the User Interface
Parameters: <none>
Notes:   Here we display/view/enable the widgets in the
user-interface. In addition, OPEN all queries
associated with each FRAME and BROWSE.
These statements here are based on the "Other
Settings" section of the widget Property Sheets.
-----*/

```

```

{&OPEN-QUERY-DEFAULT-FRAME}
GET FIRST DEFAULT-FRAME.
DISPLAY k_anio_mes_corte k_tasa_Fisica k_Tasa_Moral *
WITH FRAME DEFAULT-FRAME IN WINDOW C-Win.
ENABLE RECT-4 RECT-5 k_anio_mes_corte k_tasa_Fisica k_Tasa_Moral BUTTON-2
BUTTON-3

```

```

WITH FRAME DEFAULT-FRAME IN WINDOW C-Win.
{&OPEN-BROWSERS-IN-QUERY-DEFAULT-FRAME}
VIEW C-Win.
END PROCEDURE.

```

```

/* _UIB-CODE-BLOCK-END */
&ANALYZE-RESUME

```

4.4.4.2 Pruebas Unitarias Generales

Para ambos tipos de módulos (módulos con código estándar generado por Progress y módulos que además tienen código adicional) se diseñaron pruebas unitarias de caja negra y pruebas unitarias de interfase gráfica.

Estas pruebas, que en la mayoría de los casos se realizan como parte de las pruebas de integración, se diseñaron aquí como parte de las pruebas unitarias aplicables a todos los módulos, debido a que se probarán módulos generados automáticamente por Progress, con objetos reusables y probados que tienen un comportamiento predefinido, para los cuales no tiene caso realizar pruebas de caja blanca (a menos que se deseara modificar el software Progress, lo cual no es parte del objetivo de este proyecto). Las pruebas unitarias generales que deben hacerse para cada módulo son las siguientes.

a) Pruebas de Interfase Gráfica

- Browsers, visores y ventanas desplegando los campos de tablas definidos en cada uno.
- Folders mostrando en cada pestaña los diferentes visores y ventanas.
- Campos "Fill-In" en la posición correcta.
- Botones realizando sus funciones.
- Campos "Fill-In", de texto y rectángulos desplegados en la posición correcta.
- Ortografía correcta. Todos los mensajes y pantallas deben estar escritos correctamente.
- Nombre del menú. Cuando se selecciona un menú se mantiene el nombre de éste en la parte superior izquierda de la pantalla, con el fin de saber en todo momento la opción en la que nos encontramos.
- Opciones del menú. Se muestran todas las opciones que comprenden el menú seleccionado en forma de iconos para una fácil aceptación y comprensión.
- Descripción de la opción seleccionada. Cuando el puntero del Mouse esta posesionado en un botón icono aparece una ayuda de función del mismo.
- Tecla de aceptar. El usuario controla el flujo de operaciones del sistema al activar el botón de aceptación, cancelación o selección.
- Se permite actuar en el sistema mediante el teclado o el Mouse.

b) Pruebas de Caja Negra

- Browsers funcionando en forma similar a los visores y además permitiendo navegar por diferentes registros de una tabla.
- Querys entregando a los visores y Browsers los registros de tablas especificados.
- Paneles de navegación y actualización cambiando los registros de tablas en queries, visores y browsers.
- Campos "Fill-In" mostrando y asignando campos de tablas y variables correctamente.
- Pruebas de Partición Equivalente
- Análisis de Valores Límite.
- Manejo de errores. Se debe comprobar que no se presente ninguna de las siguientes fallas al manejar errores:
 - Descripción ininteligible del error.
 - El error señalado no corresponde con el error encontrado.
 - El error hace que el sistema intervenga antes que el mecanismo de errores.
 - La condición de error no proporciona suficiente información para ayudar a la localización del error.
 - La prueba de límites. Verifica los valores máximos y mínimos permitidos. Los valores de los datos por debajo y por encima de los máximos y mínimos son apropiados para descubrir estos errores.

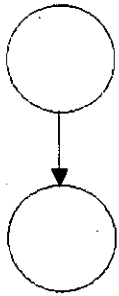
4.4.4.3 Pruebas de Caja Blanca

Estas pruebas aplican solo para los módulos del segundo tipo (con código 4GL adicional al estándar generado por Progress). Las pruebas de este tipo que deben realizarse son las de método de prueba de caminos o rutas básicas, que se describen en los siguientes apartados.

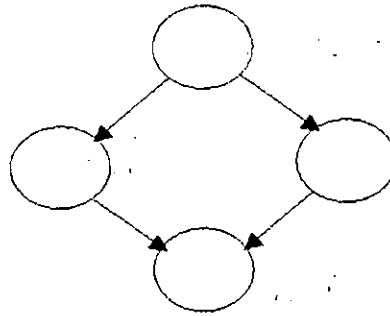
Prueba de Rutas Básicas

El método del camino básico permite obtener una medida de la complejidad lógica de un diseño de procedimiento ("procedural"). Los casos de prueba obtenidos del conjunto básico garantizan que durante la prueba se ejecuta por lo menos una vez cada sentencia del programa. Antes de considerar el método del camino básico se indicara una sencilla notación gráfica para la representación del flujo de control, denominada grafo de flujo, el cual representa el flujo de control lógico. Cada instrucción estructurada tiene su correspondiente simbolo en el grafo de flujo como se muestra en la figura 4.4_2.

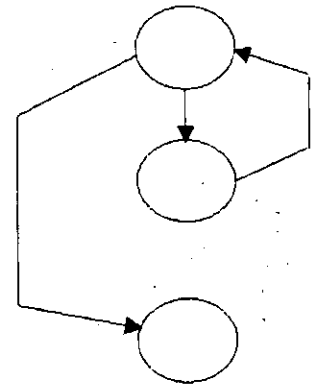
Secuencia



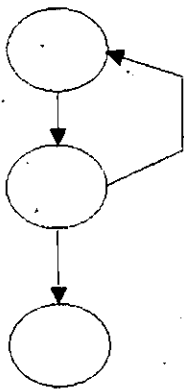
Condición (IF)



Bucle (While)



Bucle (Until)



Selectiva Multiple (Case)

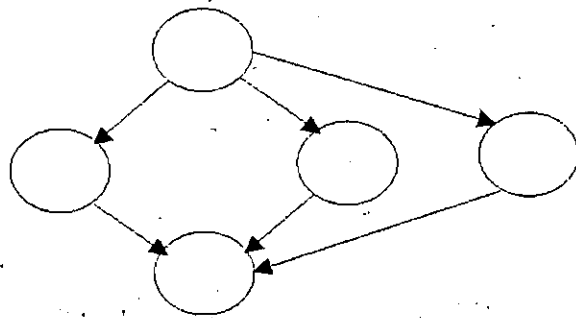


Figura 4.4_2 Notación de Grafo de Flujo

Cada círculo, denominado nodo del grafo de flujo, representa una o más sentencias de procedimiento. Las flechas del grafo de flujo, denominadas aristas o enlaces, representan flujo de control. Una arista debe terminar en un nodo, incluso aunque el nodo no represente ninguna sentencia de procedimiento. Las áreas delimitadas por aristas y nodos se denominan regiones.

Para condiciones compuestas, como es el caso cuando aparecen operadores lógicos (OR, AND, NAND y NOR) en una sentencia condicional, se agregan nodos "if" en cascada. En la figura 4.4_3 se presenta una condición compuesta.

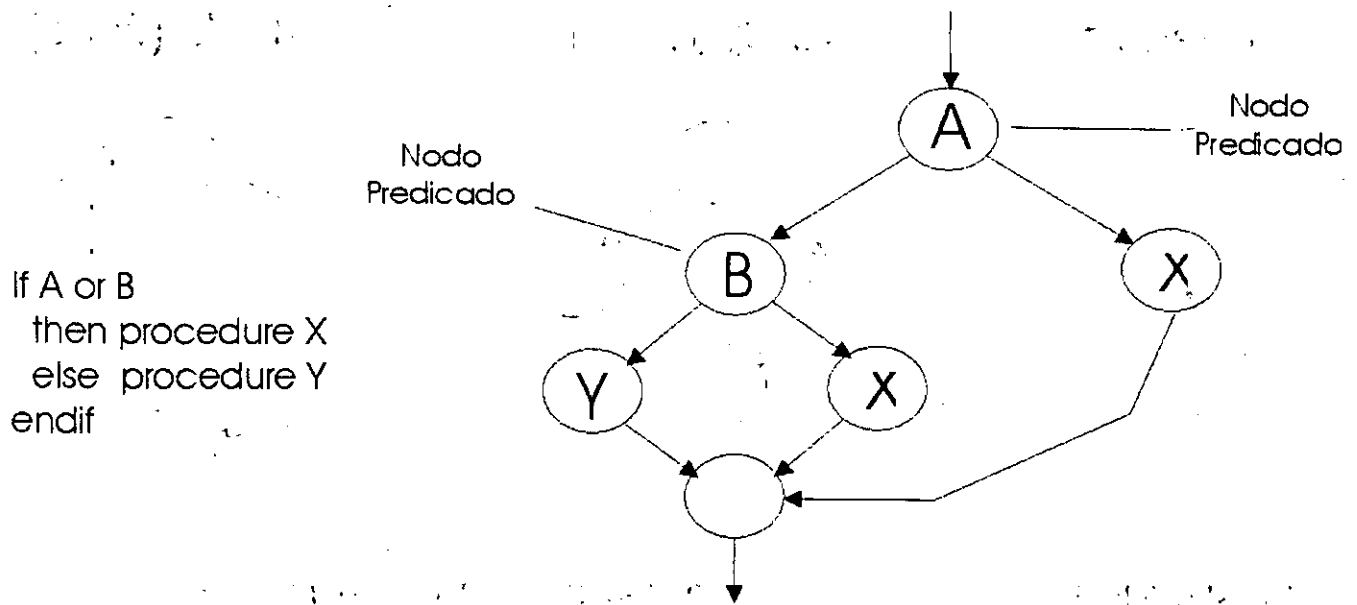


Figura 4.4_3 Notación de Lógica Compuesta

Nodo predicado

Cada nodo que contiene una condición se denomina nodo predicado y está caracterizado porque dos o más aristas emergen de él.

Complejidad ciclomática

Es una métrica del software que proporciona una medida de la complejidad lógica de un programa. Define el número de caminos independientes del conjunto básico de un programa y proporciona un límite superior para el número de pruebas que se deben realizar para asegurar que se ejecuta cada sentencia al menos una vez.

Un camino independiente es cualquier camino del programa que introduce por lo menos un nuevo conjunto de sentencias de proceso o una nueva condición. En términos del grafo de flujo, un camino independiente está constituido por lo menos por una arista que no haya sido recorrida anteriormente a la definición del camino.

Calculo de la complejidad ciclomática

Se basa en la teoría de grafos y proporciona una métrica del software muy útil. La complejidad ciclomática se puede calcular de tres formas:

1. El numero de regiones del grafo de flujo coincide con la complejidad ciclomática.
2. La complejidad ciclomática, $V(G)$, de un grafo de flujo G se define como

$$V(G) = A - N + 2$$

Donde A es el número de aristas del grafo de flujo y N es el número de nodos.

3. La complejidad ciclomática, $V(G)$, de un grafo de flujo G también se define como

$$V(G) = P + 1$$

Donde P es el número de nodos predicado contenidos en el grafo de flujo G .

El valor $V(G)$ proporciona el límite superior para el número de caminos independientes que componen el conjunto básico y, por lo tanto un valor límite superior para el número de pruebas que se deben diseñar y ejecutar para garantizar que se cubran todas las sentencias del programa.

4.4.5 Diseño de Pruebas de Integración

En estas pruebas se debe verificar lo siguiente.

- Un programa no debe tener efectos adversos o inadvertidos sobre otros.
- Las funciones cuando se combinan, pueden no producir la función principal deseada.
- La estructura de datos globales puede presentar problemas.
- Los datos se pueden perder en alguna interface.

Se deben probar características tales como:

- Fácil uso por parte del usuario.
- El fácil acceso utilizando los dispositivos de entrada: el Mouse y el teclado.
- Interacción con el usuario:
 - Selección de órdenes.
 - Representación visual.
- Procesamiento y representación de errores.
- Manipulación y análisis de datos.
- Procesamiento y generación de información visual.
- Reportes.
- Estructura y contenido de la base de datos

Se diseñaron estas pruebas siguiendo una estrategia de integración ascendente (“**Bottom-Up**”), en la que se deben realizar pruebas de caja negra con el fin de verificar que el control de flujo entre los diferentes menús y botones de la aplicación funcione como se diseñó.

El método de integración ascendente, como su nombre lo indica, empieza la construcción y la prueba con los módulos atómicos (es decir, módulos de los niveles más bajos de la estructura del programa). Dado que los módulos se integran de abajo hacia arriba, los procesos requeridos de los módulos subordinados a un nivel dado siempre están disponibles y se elimina la necesidad de resguardos (programas ficticios o "dummy" que simulan las funciones de los programas faltantes).

Para realizar estas pruebas con la estrategia de integración ascendente se deben seguir los siguientes pasos:

- Combinar los módulos de más bajo nivel en grupos que realicen una sub-función específica del software.
- Mediante un controlador se coordina la entrada y salida de los casos de prueba.
- Se prueba el grupo.
- Se eliminan los controladores y se combinan los grupos moviéndose hacia arriba por la estructura del programa.

La integración ascendente sigue el esquema mostrado en la figura 4.4_4. Se combinan los módulos inferiores para formar los grupos 1, 2 y 3. Cada uno de los grupos se somete a prueba mediante un controlador (mostrado como un cuadro punteado). Los módulos de los grupos 1 y 2 son subordinados de Ma. Los controladores D1 y D2 se eliminan y los grupos interaccionan directamente con Ma. De forma similar, se elimina el controlador D3 del grupo 3 antes de la integración con el módulo Mc y así sucesivamente.

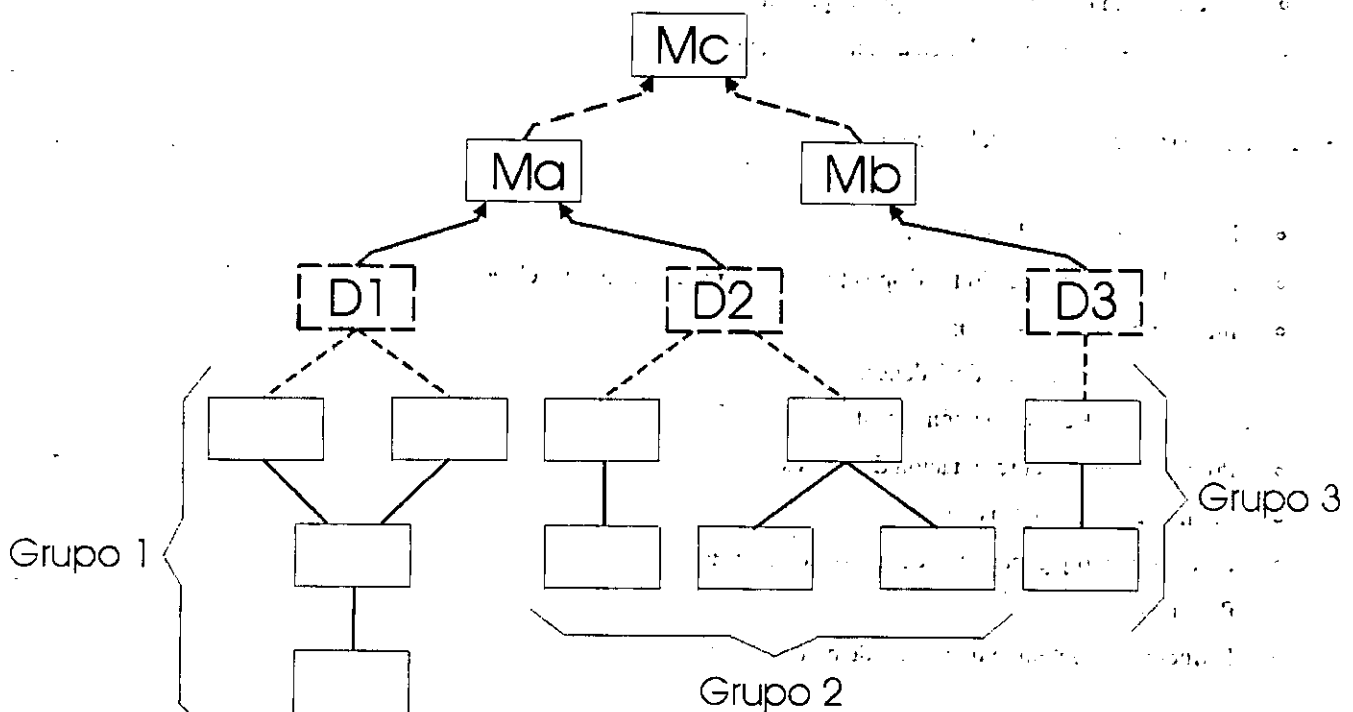


Figura 4.4_4 Integración Ascendente

La figura 4.4_5 muestra una de las categorías de los módulos o programas controladores.

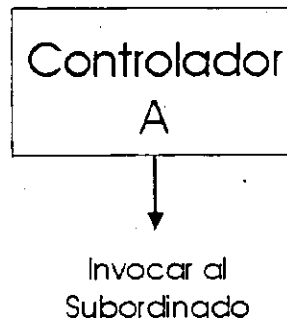


Figura 4.4_5 Controlador o Conductor

En nuestro caso la jerarquía de control de la aplicación solo tiene tres niveles (menú principal, submenús y las opciones de estos) y cada elemento o módulo de esa jerarquía regresa el control al elemento de nivel superior que lo llamó. Además, esos elementos no se pasan parámetros unos a otros; toda la información que requieren es tomada de los datos que ingresa el usuario para cada módulo y/o de la base de datos de la aplicación.

Por lo tanto la complejidad de la interfaz entre módulos, de los puntos en que se da entrada o referencia a cada módulo y los datos que se pasan a través de las interfases es muy baja; se tiene un acoplamiento muy bajo entre los módulos en general, sin acoplamiento directo de datos y con acoplamiento de control muy sencillo. Esta conectividad simple entre los módulos resulta en Software que es más fácil de entender y evita que los errores en una ubicación dada se propaguen a través de todo el sistema.

Por lo anterior, las pruebas de integración diseñadas para esta aplicación son muy sencillas. Consisten en realizar pruebas de caja negra siguiendo la estrategia de integración incremental "Bottom-Up" (ascendente), para verificar básicamente que el control de flujo entre los diferentes menús y botones de la aplicación funcione como se diseñó.

4.5 Resultado de las Pruebas

En esta sección se presentan los resultados de las pruebas realizadas en base al diseño y estrategia indicados en la sección anterior ("4.4 Diseño y Estrategia de Pruebas").

4.5.1 Pruebas Unitarias

4.5.1.1 Pruebas Unitarias Generales

Como se comentó en la sección 4.4, se diseñaron pruebas de caja negra y de interfaz gráfica en todos los módulos accesibles desde los menús de la aplicación. Los puntos que se verificaron siguiendo dicho diseño fueron:

a) Pruebas de Interfase Gráfica

- Browsers, visores y ventanas.
- Fólderres.
- Botones.
- Campos "Fill-In", de texto y rectángulos.
- Ortografía correcta.
- Nombre del menú.
- Opciones del menú.
- Descripción de la opción seleccionada.
- Tecla de aceptar.
- Uso del teclado y del "Mouse" (apuntador).

b) Pruebas de Caja Negra

- Browsers.
- Querys.
- Paneles de navegación y actualización.
- Campos "Fill-In".
- Pruebas de Partición Equivalente.
- Análisis de Valores Límite.
- Manejo de errores.

Para una descripción detallada de lo que se verificó en cada punto véase la sección 4.4.

A continuación se presentan los errores que se detectaron durante las pruebas unitarias generales, mismos que ya fueron corregidos.

Pantalla o Menú	Opción o Submenú	Error
Pantalla de "Login"		<ul style="list-style-type: none"> • Error de ortografía (BIENBENIDO) • Agregar menú inicial para el administrador que incluya las opciones de todos los usuarios más un ABC de usuarios
Menú de Diario	Mov. Barclays Bank	Al ejecutarse y no tener capturada la fecha marca mensaje fe fecha no encontrada; al aceptar el mensaje interrumpe la aplicación.
	Consulta de saldos	Al ejecutarse no se visualizan los resultados o no menciona que es lo que realizó.
	Débito de la chequera	Después de ejecutarse el proceso aparece mensaje "M inv pago..."
	Diario de pago de cheques	El reporte no tiene formato.
	Listado de remesas	<ul style="list-style-type: none"> • El titulo se repite. • Se introducé la fecha inicial y aparece mensaje "no se puede usar el campo fecha". • Falta un texto para indicar al usuario como digitar (aplicado/no aplicado).
Menú Mensual	Movtos de cierre de mes	La respuesta no se entiende.
	Cancelación masiva de movtos	Falta un catálogo o lista de selección de movimientos para no tener que aprender de memoria todos.
	Cierre del mes	Revisar la ortografía.
	Contabilizar	Sigue pidiendo las fechas una vez capturadas.
	Listado total de débitos	No permite crear el archivo en la ruta deseada.
Menú Cuentas	ABC de Cuentas	<ul style="list-style-type: none"> • Mover Status de Alta a la primera página. • Mover Área junto a Div/Plaza/Suc.
	Listado cuentas y cheques congelados	<ul style="list-style-type: none"> • Limpiar campos al iniciar el modulo. • Al cambiar tipo de producto, aparece el mensaje "los campos fecha", y no permite continuar.
Menú Parámetros	Parámetros generales	<ul style="list-style-type: none"> • Faltan títulos. • Falta botón para salir. • Mover al menú inicial del administrador para que no puedan ejecutarlo los demás usuarios.
	ABC de tipos de movimientos	<ul style="list-style-type: none"> • Faltan títulos. • Mover al menú inicial del administrador para que no puedan ejecutarlo los demás usuarios.
	ABC de "overnigth"	Centrar despliegue de objetos.
	Cancelar cuentas	Después de invocarlo y contestar su mensaje, suspende con un error la ejecución de toda la aplicación.
Menú Memos	Memo acuse de chequera	Error de ortografía (provisional).
Menú Chemical	Generación de cuentas de cliente	No permite crear el archivo de cuentas, sino que debe ya haber sido creado.
	ABC promotores	Envía mensaje de error y termina la ejecución.
	Validación de pagos	Después de ejecutarse aparece mensaje: Q-tpmovto...

Tabla 4.5_1 Errores Detectados en las Pruebas Unitarias Generales

Pantalla o Menú	Opción o Submenú	Error
Menú Pagos	Validación de Pagos	Envía el mensaje "Tipo de Movto. no encontrado" en todos los casos.
Todos los Menús	Submenús de Listados excepto: • Listados de Pagos y Montos de Preformato. • Cheques Cobrados Cuentas Regionales.	Revisar rangos de fechas (la fecha final debe ser mayor que la inicial).
	Todas las pantallas	Habilitar maximización de ventanas.

Tabla 4.5_1 (Cont.) Errores Detectados en las Pruebas Unitarias Generales

4.5.1.2 Pruebas de Caja Blanca

Como también se comentó en la sección 4.4, una vez corregidos los errores encontrados en las pruebas unitarias generales (pruebas de caja negra y de interfase gráfica), se realizaron pruebas de caja blanca solo en los módulos que tienen código 4GL adicional al estándar generado por Progress. Estas fueron pruebas de rutas básicas (que se describen en la sección 4.4) siguiendo el siguiente procedimiento:

- Revisar las líneas de código adicional y asignar números consecutivos a las sentencias correspondientes a estructuras de control básicas (if, and, or, while, until, case).
- Obtener del punto anterior la gráfica de flujo.
- Determinar de la gráfica de flujo las rutas básicas (y la complejidad ciclomática).
- Preparar los casos de prueba para forzar la ejecución de cada ruta en el conjunto básico.
- Ejecutar los casos de prueba utilizando la herramienta "Debugger" (depurador) de Progress, la cual permite controlar la ejecución paso a paso de cada procedimiento, asignar valores a las variables e invocar a cualquier otro procedimiento.
- Corregir los errores encontrados.

Como ejemplo de esto, a continuación se presenta el código a probar (código 4GL adicional al estándar), la gráfica de flujo y el resultado de la prueba de trayectorias correspondientes a los dos módulos siguientes:

- c-valpag.w
- c-genmovcierre.w

Dentro del código 4GL a probar, los nodos de la gráfica de flujo se indican como (xx), donde "xx" = número de nodo.

4.5.1.3 Pruebas de Caja Blanca para c-valpag.w

Código a Probar e Identificación de los Nodos de la Gráfica de Flujo

```

/* ***** Control Triggers ***** */

&Scoped-define SELF-NAME BUTTON-2
&ANALYZE-SUSPEND UIB-CODE-BLOCK_CONTROL BUTTON-2 C-Win
(1) ON CHOOSE OF BUTTON-2 IN FRAME DEFAULT-FRAME /* Button 2 */
(2) DO:
    DEF VAR t_bandera AS LOGICAL FORMAT "Si/No" NO-UNDO.
    DEF VAR t_saldo_fin_ant LIKE Q_saldo_chem.saldo_fin NO-UNDO.
    DEF VAR t_inicio AS LOGICAL FORMAT "Si/No" NO-UNDO.
    DEF VAR t_passw AS CHARACTER NO-UNDO.
    DEF FRAME fr_passw
        t_passw FORMAT "X(7)" BLANK LABEL "PassWord"
        WITH FRAME fr_passw
            CENTERED
            SIDE-LABELS
            OVERLAY
            ROW 12.58
            THREE-D
            NO-BOX.

(4) REPEAT:
(7) IF WEEKDAY(TODAY) = 2 THEN
(8)     t_fecha_mov = TODAY - 3.
    ELSE
(9)     t_fecha_mov = TODAY - 1.
(10)  FIND PREV Q_saldo_chem
        WHERE Q_saldo_chem.f_mov <= t_fecha_mov - 1
        NO-LOCK NO-ERROR.
(11)  IF NOT AVAILABLE Q_saldo_chem THEN DO:
(12)  FIND FIRST Q_saldo_chem NO-LOCK.
        MESSAGE "No existe esa Fecha, inicia con " +
            STRING(Q_saldo_chem.f_mov) VIEW-AS ALERT-BOX.
        UPDATE "Deseas Iniciar el proceso (S/N)" t_inicio
            WITH FRAME FR_KK
                CENTERED
                NO-LABELS
                OVERLAY
                NO-BOX
                THREE-D.
(15)  IF NOT t_inicio THEN
(16)  UNDO, RETRY.
(17)  t_saldo_fin_ant = 0.
    END.
    ELSE
(13)  t_saldo_fin_ant = Q_saldo_chem.saldo_fin.
(14)  UPDATE t_passw FORMAT "X(7)" BLANK
        WITH FRAME fr_passw.
(18)  IF t_passw <> "cheques" THEN DO:
(19)  MESSAGE "Repita el Password no corresponde"
        VIEW-AS ALERT-BOX.
        UNDO, RETRY.
    END.
(20)  HIDE FRAME fr_passw.

```



```

(21) FOR EACH Q_saldo_chem
      WHERE Q_saldo_chem.f_mov >= t_fecha_mov:
      DISPLAY
          t_saldo_fin_ant   FORMAT ">,999,999.99-" LABEL "SdoAnt"
          Q_saldo_chem.saldo_ini FORMAT ">,999,999.99-" LABEL "SdoIni"
          Q_saldo_chem.credits  FORMAT ">>,999,999.99" LABEL "Cred."
          Q_saldo_chem.debits   FORMAT ">>,999,999.99-" LABEL "Debit"
          Q_saldo_chem.saldo_fin FORMAT ">,999,999.99-" LABEL "SdoFin"
          Q_saldo_chem.f_mov    FORMAT "99/99/9999"   LABEL "FMov"
      WITH NO-LABEL no-box.
      Q_saldo_chem.saldo_ini = t_saldo_fin_ant.
      Q_saldo_chem.saldo_fin = Q_saldo_chem.saldo_ini + Credits - Debits.
      t_saldo_fin_ant      = q_saldo_chem.saldo_fin.
(22) END.
(23) MESSAGE "Finalizo todo el Proceso" VIEW-AS ALERT-BOX.
(5) LEAVE.
(6) END. /*REPEAT*/
      RETURN.
(3) END. /*DO*/
      /* _UIB-CODE-BLOCK-END */
      &ANALYZE-RESUME

&Scoped-define SELF-NAME t_fecha_mov
&ANALYZE-SUSPEND _UIB-CODE-BLOCK_CONTROL t_fecha_mov C-Win
(24) ON LEAVE OF t_fecha_mov IN FRAME DEFAULT-FRAME /* Fecha Act. Saldo */
(25) DO:
      ASSIGN t_fecha_mov.
      t_fecha_mov = TODAY.
      DISPLAY t_fecha_mov
      WITH FRAME DEFAULT-FRAME.
(26) END.

/* _UIB-CODE-BLOCK-END */
&ANALYZE-RESUME

```

Cálculo de la Complejidad Ciclomática de c-valpag.w

En la figura 4.5_1 se muestra la gráfica de flujo correspondiente al módulo c-valpag.w (los nodos predicado se muestran sombreados). En ella se ve que:

- La parte correspondiente al "control trigger" (disparador de control) de "BUTTON-2" tiene 9 regiones y por tanto 9 trayectorias básicas a probar.

La parte de la gráfica correspondiente al "control trigger" de "t_fecha.mov" tiene 2 regiones y 2 trayectorias básicas.

- Este resultado también se obtiene si usamos la fórmula: $V(G) = A - N + 2$ (donde A es el número de aristas del grafo de flujo y N es el número de nodos).

La parte de la gráfica para el "control trigger" de "BUTTON-2" tiene 30 aristas y 23 nodos, por lo que $V(G)_1 = 30 - 23 + 2 = 9$.

La parte de la gráfica para el "control trigger" de "t_fecha.mov" tiene 3 aristas y 3 nodos, por lo que $V(G)_2 = 3 - 3 + 2 = 2$.

Resultado de la Prueba de Rutas Básicas en c-valpag.w

Usando el "Debugger" de Progress se ejecutó cada una de las rutas básicas. Se encontró que el bloque "REPEAT" delimitado por los nodos (4) y (6) es innecesario, ya que la última sentencia dentro del mismo (sentencia "LEAVE" en el nodo (5)) ocasiona que el proceso salga de dicho bloque "REPEAT" al terminar la primera iteración; por lo tanto, se eliminaron esas sentencias en los nodos (4) y (6).

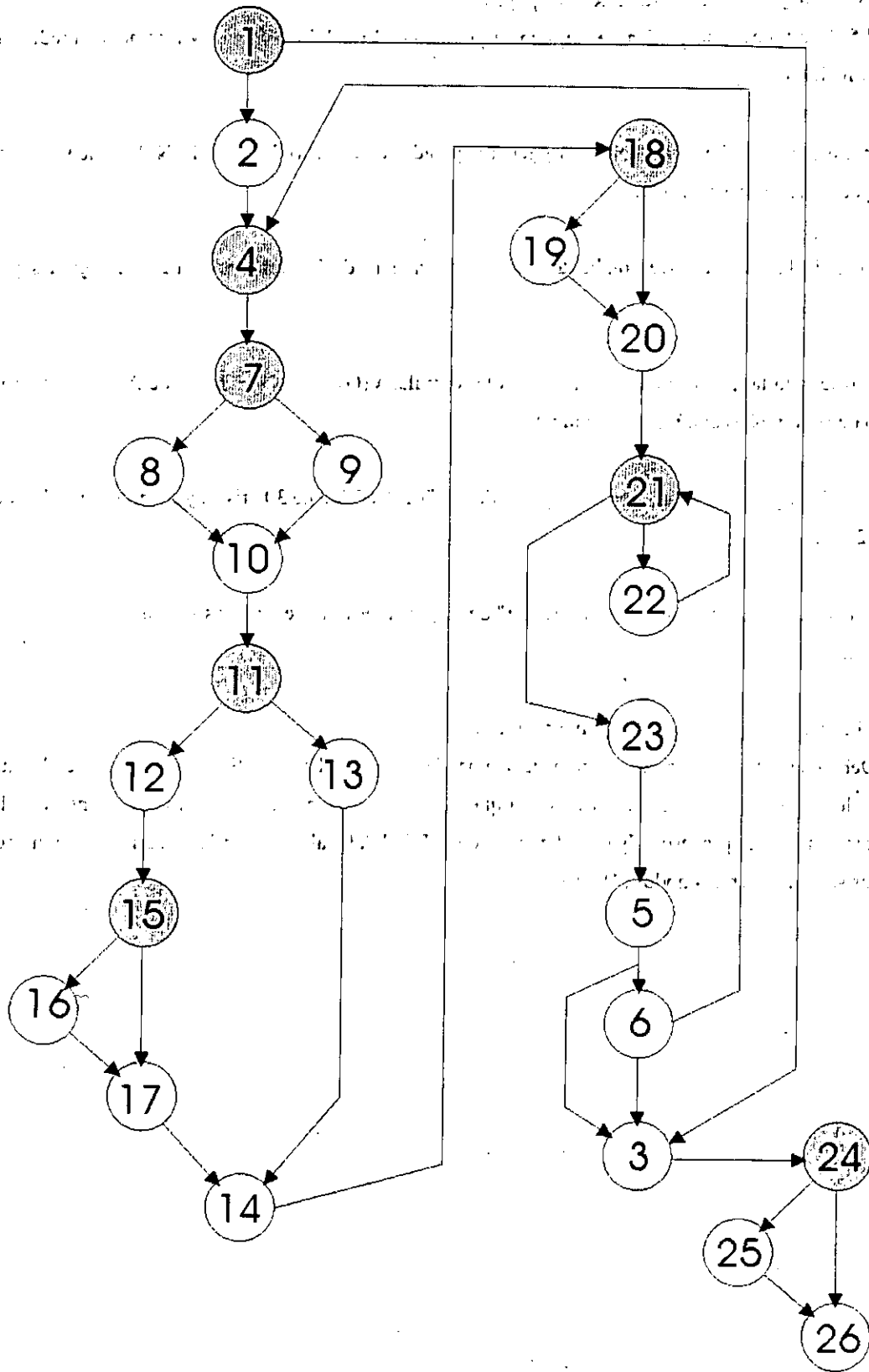


Figura 4.5_1 Gráfica de Flujo del Módulo c-valpag.w

4.5.1.4 Pruebas de Caja Blanca para genmovcierre.w

Código a Probar e Identificación de los Nodos de la Gráfica de Flujo

```

/* ***** Control Triggers ***** */

&Scoped-define SELF-NAME BUTTON-2
&ANALYZE-SUSPEND _UIB-CODE-BLOCK_CONTROL BUTTON-2 C-Win
ON CHOOSE OF BUTTON-2 IN FRAME DEFAULT-FRAME /* Button 2 */
(0) DO:
(1)   IF k_tasa_moral = 0 OR k_tasa_fisica = 0 OR k_anio_mes_corte = 0 THEN DO
(2)     MESSAGE "Teclear los Campos".
     RETURN NO-APPLY.
     END.
     t_periodo = k_anio_mes_corte.
     t_tasa_fisica = k_tasa_fisica.
     t_tasa_moral = k_tasa_moral.

(3)   DISPLAY "Generando Movimientos. Espere un momento .... "
     WITH FRAME fr_x OVERLAY THREE-D
     ROW 10 COL 8 COLOR WHITE/BLUE.

     FIND FIRST Q_p parametro NO-LOCK NO-ERROR.

(4)   DO TRANSACTION:
     FIND Q_Tasa
     WHERE Q_Tasa.periodo = t_periodo
     EXCLUSIVE-LOCK NO-ERROR.

(5)     IF NOT AVAILABLE Q_Tasa THEN
(5a)       CREATE Q_tasa.
(5b)       ASSIGN Q_tasa.periodo = t_periodo
     Q_tasa.tasa_fisica = t_tasa_fisica
     Q_tasa.tasa_moral = t_tasa_moral.
     FIND cta_global
     WHERE cta_global.id_producto = 'Q' AND cta_global.id_cliente =
     Q_Parametro.cuenta_global
     EXCLUSIVE-LOCK.

(6)   END.

(7)   FOR EACH U_cliente
     WHERE U_cliente.id_producto = 'Q' EXCLUSIVE-LOCK:

     val_periodo = ( YEAR( U_cliente.f_alta ) * 100 ) + MONTH( U_cliente.f_alta ).

(8)     IF U_cliente.id_cliente <> Q_Parametro.cuenta_global AND
(9)     U_cliente.Stat_Alta AND
(10)    val_periodo <= t_periodo THEN DO:

(11)   P_cuenta = U_cliente.id_cliente.
     P_prod = U_cliente.id_producto.

     RUN qfestad0.

     t_tiene_inv_vig = FALSE.
     t_calculo_int = FALSE.
     t_calculo_bajoprom = FALSE.

```

```

/* t_talonario = TRUE. */

FIND FIRST M_inversion
WHERE M_inversion.id_producto = U_cliente.id_producto AND
      M_inversion.id_cliente = U_cliente.id_cliente AND
      M_inversion.f_valor <= P_f_final AND
      M_inversion.f_venc >= P_f_inicial AND
      M_inversion.st_impresion <> "EL" AND
      M_inversion.st_inversion > 1
NO-LOCK NO:ERROR:

(12) IF AVAILABLE M_inversion AND
(13) U_cliente.id_producto = "U" THEN,
(14) t_tiene_inv_vig = TRUE.

/*----- Verifica que tenga Talonario -----*/

(15) IF U_cliente.id_talonario THEN DO:
(15a) IF t_tiene_inv_vig THEN DO:
(15b) IF ( U_cliente.tipo_persona = 4 AND /* fisica */
(15c) P_SalPro >= Q_Parametro.minimo_fisica ) OR
(15d) ( U_cliente.tipo_persona <> 4 AND /* moral */
(15e) P_SalPro >= Q_Parametro.minimo_moral ) THEN DO:
(15f) t_calculo_int = TRUE.
      END.
      ELSE DO:
(15g) t_calculo_bajoprom = TRUE.
      END.
      ELSE DO:
(15h) IF ( U_cliente.tipo_persona = 4 AND /* fisica */
(15i) P_SalPro >= Q_Parametro.sdo_min_ctalonario_fisica ) OR
(15j) ( U_cliente.tipo_persona <> 4 AND /* moral */
(15k) P_SalPro >= Q_Parametro.sdo_min_ctalonario_moral ) THEN DO:
(15l) t_calculo_int = TRUE.
      END.
      ELSE DO:
(15m) t_calculo_bajoprom = TRUE.
      END.
      END.
      ELSE DO:
(15n) IF t_tiene_inv_vig THEN DO:
(15o) IF ( U_cliente.tipo_persona = 4 AND /* fisica */
(15p) P_SalPro >= Q_Parametro.minimo_fisica ) OR
(15q) ( U_cliente.tipo_persona <> 4 AND /* moral */
(15r) P_SalPro >= Q_Parametro.minimo_moral ) THEN DO:
(15s) t_calculo_int = TRUE.
      END.
      ELSE DO:
(15t) t_calculo_int = FALSE.
      END.
      END.
      ELSE DO:
(15u) IF ( U_cliente.tipo_persona = 4 AND /* fisica */
(15v) P_SalPro >= Q_Parametro.minimo_fisica ) OR
(15w) ( U_cliente.tipo_persona <> 4 AND /* moral */
(15x) P_SalPro >= Q_Parametro.minimo_moral ) THEN DO:
(15y) t_calculo_int = TRUE.

```

```

        END.
        ELSE DO:
(15z)           t_calculo_bajoprom = TRUE.
        END.
(16)           END.
(17)           IF U_cliente.cuenta_especial THEN DO:
                t_calculo_bajoprom = FALSE.
(18)           IF ( U_cliente.tipo_persona = 4      AND /* fisica */
(18a)             P_SalPro >= Q_Parametro.minimo_fisica ) OR
(18b)             ( U_cliente.tipo_persona <> 4      AND /* moral */
(18c)             P_SalPro >= Q_Parametro.minimo_moral ) THEN DO:
(18d)             t_calculo_int = TRUE.
                END.
                ELSE DO:
(18e)             t_calculo_int = FALSE.
                END.
(19)           END.

/*--- eliminar posteriormente ---*/
/*
(20)           display stream imp
                U_cliente.id_producto
                U_cliente.id_cliente
                U_cliente.cuenta_especial
                U_cliente.tipo_persona
                t_tiene_inv_vig
                t_calculo_int
                t_calculo_bajoprom
                t_talonario
                p_salpro
                with 3 col.
                */

/*----- Calculo de Intereses -----*/
(21)           IF t_calculo_int THEN DO:
(21b)           FIND FIRST Q_tpmovto
                WHERE Q_tpmovto.id_movto = Q_parametro.id_int_pasivo
                NO-LOCK NO-ERROR.
(22)           IF U_cliente.tipo_persona = 4 THEN DO: /* fisica */
(22a)             IF U_cliente.f_alta > p_f_inicial AND
(22b)             U_cliente.f_alta <= p_f_final THEN
(22c)             x_monto = ROUND(( P_SalPro * ( p_f_final - U_cliente.f_alta +
                1 ) * t_tasa_fisica ) / 36000 , 2 ).
                ELSE
(22d)             x_monto = ROUND(( P_SalPro * ( p_f_final - p_f_inicial + 1 )
                * t_tasa_fisica ) / 36000 , 2 ).
                END.
                ELSE DO:
(22e)             IF U_cliente.f_alta > p_f_inicial AND
(22f)             U_cliente.f_alta <= p_f_final THEN
(22g)             x_monto = ROUND(( P_SalPro * ( p_f_final - U_cliente.f_alta +
                1 ) * t_tasa_moral ) / 36000 , 2 ).
                ELSE

```

CAPITULO 4. DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA

```

(22h)      x_monto = ROUND(( P_SalPro * ( p_f_final - p_f_inicial + 1 )
              * t_tasa_moral ) / 36000 , 2 ).
      END.

(23)      IF x_monto > 0 THEN DO:
(24)      RUN QFGETCNO (INPUT p_f_final, OUTPUT t_consec).

      CREATE Q_movimiento.
      ASSIGN Q_movimiento.id_movto = Q_parametro.id_int_pasivo
      Q_movimiento.afecto_disponible = Q_tpmovto.Afec_saldo_disp
      Q_movimiento.cronologico = t_consec
      Q_movimiento.id_producto = U_cliente.id_producto
      Q_movimiento.id_cliente = U_cliente.id_cliente
      Q_movimiento.f_movto = p_f_final
      Q_movimiento.monto = x_monto.

      {qtgeint0.i1} /* afecta cuentas */

      /*
      U_cliente.Interes_Anual = U_cliente.Interes_anual + x_monto.
      */

(25)      END.

(26)      END. /* genera intereses */

      /*----- Cargo por bajo promedio -----*/

(27)      IF t_calculo_bajoprom THEN DO:

(28)      FIND FIRST Q_tpmovto
      WHERE Q_tpmovto.id_movto = Q_parametro.id_cargo_promedio
      NO-LOCK NO-ERROR.

      RUN QFGETCNO (INPUT p_f_final, OUTPUT t_consec).

      x_monto = Q_tpmovto.monto_fijo.

      CREATE Q_movimiento.
      ASSIGN Q_movimiento.id_movto = Q_parametro.id_cargo_promedio
      Q_movimiento.afecto_disponible = Q_tpmovto.Afec_saldo_disp
      Q_movimiento.cronologico = t_consec
      Q_movimiento.id_producto = U_cliente.id_producto
      Q_movimiento.id_cliente = U_cliente.id_cliente
      Q_movimiento.f_movto = p_f_final
      Q_movimiento.monto = x_monto.

      {qtgeint0.i1} /* afecta cuentas */

(29)      END. /* genera cargo por bajo promedio */

      /* Cargo por cheques emitidos */

(30)      IF p_n_che_emit > 0 AND
(31)      NOT U_cliente.cuenta_especial THEN DO:

(32)      FIND FIRST Q_tpmovto
      WHERE Q_tpmovto.id_movto = Q_parametro.id_cargo_cheque
      NO-LOCK NO-ERROR.
    
```

```
RUN QFGETCNO (INPUT p_f_final, OUTPUT t_consec).
```

```
x_monto = Q_tpmovto.monto_fijo * p_n_che_emit.
```

```
CREATE Q_movimiento.
```

```
ASSIGN Q_movimiento.id_movto = Q_parametro.id_cargo_cheque
```

```
Q_movimiento.afecto_disponible = Q_tpmovto.Afec_saldo_disp
```

```
Q_movimiento.cronologico = t_consec
```

```
Q_movimiento.id_producto = U_cliente.id_producto
```

```
Q_movimiento.id_cliente = U_cliente.id_cliente
```

```
Q_movimiento.f_movto = p_f_final
```

```
Q_movimiento.monto = x_monto.
```

```
{qtgeint0.i1} /* afecta cuentas */
```

```
(33) END. /* genera cargo por cheques emitidos */
```

```
(34) END. /* no es cuenta global */
```

```
(35) END. /* for each cuenta */
```

```
(36) HIDE FRAME fr_x NO-PAUSE.
```

```
MESSAGE "FINALIZO EL PROCESO".
```

```
APPLY 'CLOSE' TO THIS-PROCEDURE.
```

```
(37) END.
```

```
/* _UIB-CODE-BLOCK-END */
```

```
&ANALYZE-RESUME
```

Cálculo de la Complejidad Ciclomática de genmovcierre.w

En las figuras 4.5_2a a 4.5_2c se muestra la gráfica de flujo correspondiente al módulo genmovcierre.w. De ellas se puede obtener su número de trayectorias básicas de prueba (complejidad ciclomática), esta vez usando la fórmula: $V(G) = P + 1$ (donde P es el número de nodos predicado contenidos en el grafo de flujo G).

La gráfica de flujo de genmovcierre.w tiene 8 nodos predicado en la figura 4.5_2a, 19 nodos predicado en la figura 4.5_2b y 15 nodos predicado en la figura 4.5_2c. Por lo tanto, el número de trayectorias básicas de prueba para este módulo es:

$$V(G) = 8 + 19 + 15 = 43.$$

Resultado de las Pruebas de Rutas Básicas en genmovcierre.w

Usando el "Debugger" de Progress se ejecutó cada una de las rutas básicas del módulo. Se encontró que entre los nodos (5) (inicio de una estructura "IF") y (6) (fin de bloque "DO" exterior al "if") se debe definir otro bloque "DO" que incluya todas las sentencias representadas por los nodos (5a) y (5b) ("CREATE", "ASSIGN" y "FIND"), ya que en la forma en que aparece arriba el código del módulo, en caso de ser verdadera la condición evaluada por el "IF" se ejecuta la sentencia (5a) y después las correspondientes a (5b); mientras que en caso de ser falsa dicha condición de todas formas se ejecutan

indebidamente las sentencias correspondientes a (5b). Por tanto se agregaron sentencias "DO", y "END" después del nodo (5) y antes del nodo (6) respectivamente.

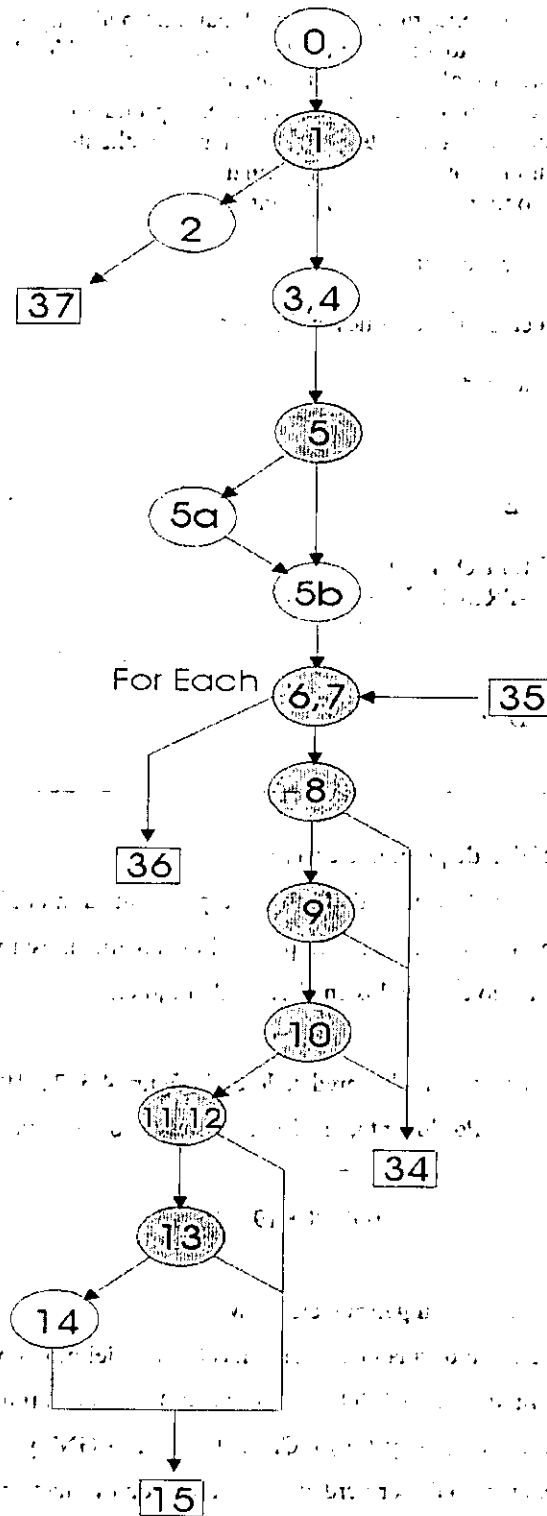


Figura 4.5_2a Gráfica de Flujo del Módulo genmovcierre.w

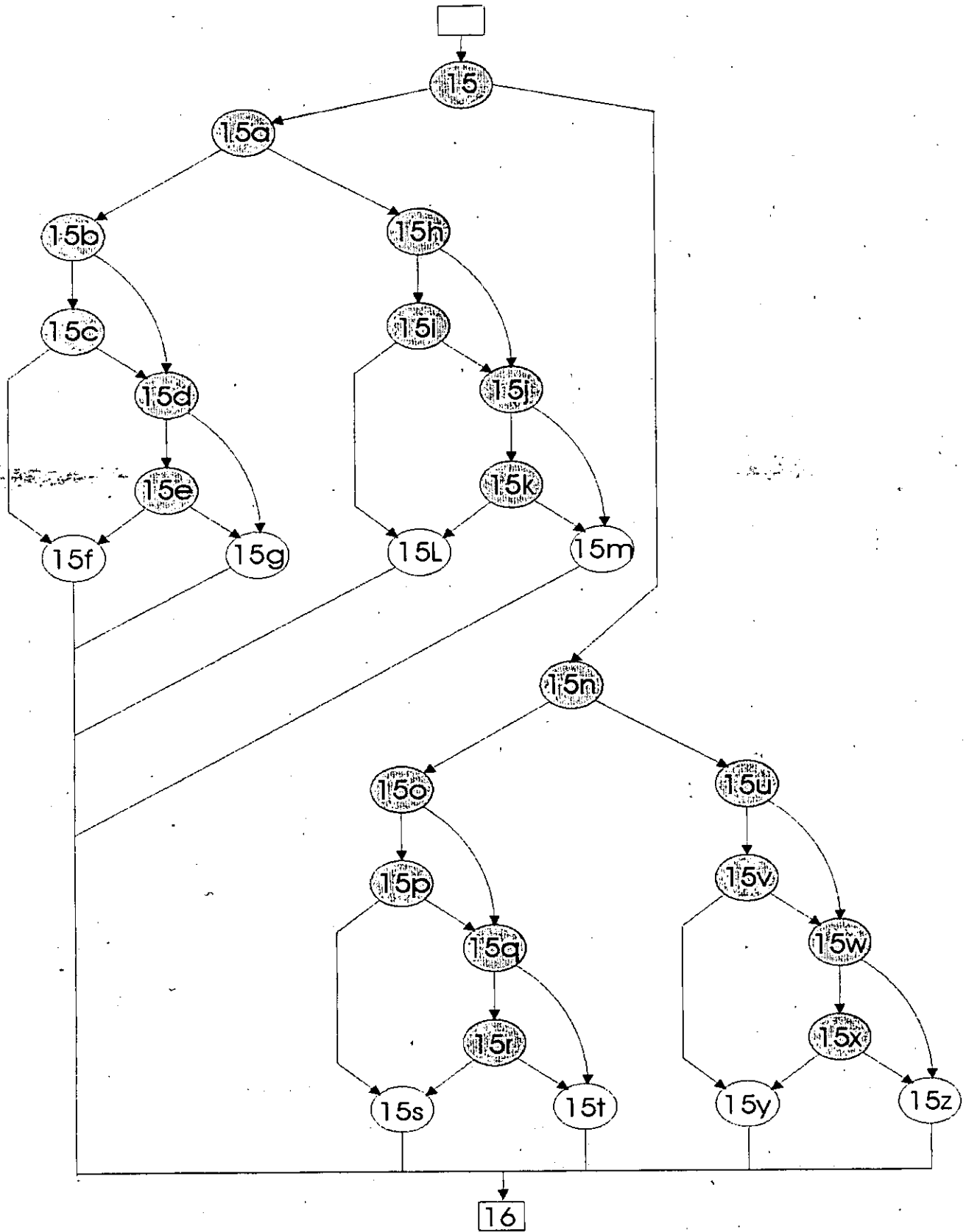


Figura 4.5_2b Gráfica de Flujo del Módulo genmovcierre.w (Cont.)

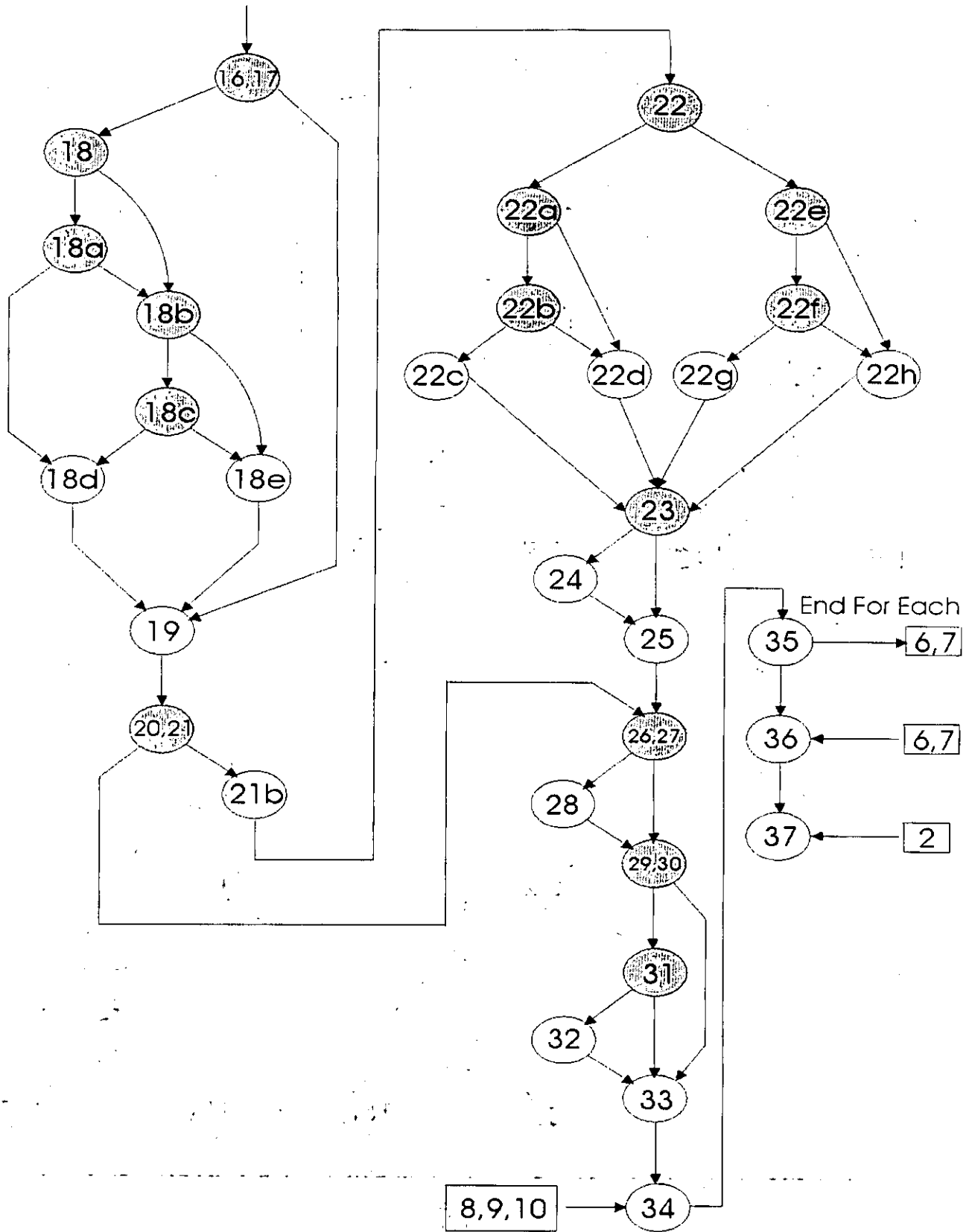


Figura 4.5_2c Gráfica de Flujo del Módulo genmovcierre.w (Cont.)

4.5.2 Pruebas de Integración

Como se comentó en la sección 4.4, las pruebas de integración diseñadas para esta aplicación son muy sencillas. Consisten en realizar pruebas de caja negra siguiendo la estrategia de integración incremental “**Bottom-Up**” (ascendente), para verificar básicamente que el control de flujo entre los diferentes menús y botones de la aplicación funcione como se diseñó.

A continuación se presentan los errores que se detectaron y que fueron corregidos en esta etapa de las pruebas.

Pantalla o Menú	Opción o Submenú	Error
Menú de Diario	Alta de remesas	<ul style="list-style-type: none"> • Aparece mensaje de error “c:\jpol\grafico... no lo encuentra”. • Al ejecutar directamente el proceso w-abcremasas.w al que llama, este no realiza ninguna función.
Menú Mensual	Interés diario promedio	Aparece como opción deshabilitada.
	Movtos de cierre de mes	El botón de salida no se activa.
Menú Cuentas	Registro y consulta de cheques	Opción deshabilitada.
	Listado de status de cuenta por fecha	No corresponde el titulo del modulo con el del menú.
	Listado de status de cuenta por FPA	No corresponde el titulo del modulo con el del menú.
Menú Parámetros	Parámetros generales	Falta botón de salida.
	Cancelar cuenta de cheques	Sobra. El modulo ABC de cuentas realiza las mismas funciones.
Menú Chemical	Validación de pagos	No corresponde el titulo del modulo con el del menú.
Menú Pagos	Validación de Pagos	No corresponde el titulo del modulo con el del menú.

Tabla 4.5_2 Errores Detectados en las Pruebas de Integración

Otros errores se encontraron al compilar todos los módulos como un conjunto. Estos se muestran en el reporte 4.5_1 que presenta los resultados de la primera compilación.

Reporte 4.5_1. Resultados de la Primera Compilación

```

11:43:57 Looking for *.p in d:\progre~1\appl_c~1\fuentes
Compiling C-CONTAB.P
Compiling C-CONTABDESCOTAB.P
Compiling C-DESCON.P
Compiling C-LCATALOGOCTA.P
Compiling C-LCTASALTASCANC.P
Compiling C-LCTASALTASCANCPROM.P
Compiling C-LCTASCHEQCONG.P
Compiling C-LEDOCT.P
** "vars_imp.i" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-LEDOCT.P Could not understand line 52. (193)
Compiling C-LINGGASUTI.P
Compiling C-LPERFILCTE.P
Compiling C-PREFOR.P
Compiling C-VALPAG.P
** "MMVALPA0.I3" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-VALPAG.P Could not understand line 34. (193)
** "OLINSCL0.VAR" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-VALPAG.P Could not understand line 35. (193)
** Unknown Field or Variable name - t_f_control. (201)
** d:\progre~1\appl_c~1\fuentes\C-VALPAG.P Could not understand line 225. (196)
Compiling EJEMPLO1.P
** Unable to understand after -- "&SCOTED-DEFINE". (247)
** d:\progre~1\appl_c~1\fuentes\EJEMPLO1.P Could not understand line 1. (198)
11:43:59 Looking for *.w in d:\progre~1\appl_c~1\fuentes
Compiling B-ABC-PAIS.W
Compiling B-REMESA.W
Compiling B-TABLA_Q_CORTE.W
Compiling C-ACTSDC.W
Compiling C-BAJAMOVCHEMGLO.W
** "CREAR1.I" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-BAJAMOVCHEMGLO.W Could not understand line 506. (193)
** "EDITAR1.I" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-BAJAMOVCHEMGLO.W Could not understand line 517. (193)
Compiling C-BAJAMOVFAX.W
** "CREAR1.I" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-BAJAMOVFAX.W Could not understand line 730. (193)
** "EDITAR1.I" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-BAJAMOVFAX.W Could not understand line 741. (193)
Compiling C-BORRARPAGOCHEQCTROFIN.W
Compiling C-CAMOVCHCTE.W
Compiling C-CAMOVCHGEN.W
Compiling C-CANCELACTA.W
Compiling C-CANCMOVMAS.W
Compiling C-CANMOVCHC.W
Compiling C-CANMOVCHG.W
Compiling C-CAPINSTPAGOCTE.W
** "CREAR1.I" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-CAPINSTPAGOCTE.W Could not understand line 1077. (193)
** "EDITAR1.I" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-CAPINSTPAGOCTE.W Could not understand line 1090. (193)
** "BORRAR1.I" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-CAPINSTPAGOCTE.W Could not understand line 1101. (193)
Compiling C-CAPMOVCHEMCTE.W
** "CREAR1.I" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-CAPMOVCHEMCTE.W Could not understand line 701. (193)
** "EDITAR1.I" was not found. (293)

```

```
** d:\progre~1\appl_c~1\fuentes\C-CAPMOVCHEMCTE.W Could not understand line 712. (193)
** "BORRAR1.I" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-CAPMOVCHEMCTE.W Could not understand line 723. (193)
Compiling C-CAPMOVCHEMGLO.W
** "CREAR1.I" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-CAPMOVCHEMGLO.W Could not understand line 711. (193)
** "EDITAR1.I" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-CAPMOVCHEMGLO.W Could not understand line 722. (193)
** "BORRAR1.I" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-CAPMOVCHEMGLO.W Could not understand line 733. (193)
Compiling C-CAPMOVFAV.W
** "CREAR1.I" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-CAPMOVFAV.W Could not understand line 727. (193)
** "EDITAR1.I" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-CAPMOVFAV.W Could not understand line 738. (193)
** "BORRAR1.I" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-CAPMOVFAV.W Could not understand line 749. (193)
Compiling C-CAPOTROSMOVTOS.W
Compiling C-CIERREMES.W
Compiling C-CONEDOCTACHEM.W
Compiling C-CONGELARCTA.W
Compiling C-CONMOVCHG.W
Compiling C-CONMOVEDOCTA.W
Compiling C-DEBOCHECHEM.W
** Unknown Field or Variable name - x_bien. (201)
** d:\progre~1\appl_c~1\fuentes\C-DEBOCHECHEM.W Could not understand line 428. (196)
Compiling C-DEBOCHECTE.W
Compiling C-ESTADCLIENTE.W
Compiling C-GENMOVCIERRE.W
** "qtgeint0.i1" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-GENMOVCIERRE.W Could not understand line 477. (193)
** "qtgeint0.i1" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-GENMOVCIERRE.W Could not understand line 507. (193)
** "qtgeint0.i1" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-GENMOVCIERRE.W Could not understand line 533. (193)
Compiling C-GENNUMCHEQ.W
Compiling C-KK.W
Compiling C-LCARTACHEQREB.W
Compiling C-LCHEQCOBCTRFIN.W
Compiling C-LCHEQCOBRACROSFINAN.W
Compiling C-LCHEQCTROFINAN.W
Compiling C-LCUADRARCONTAB.W
Compiling C-LIBPAGOCHEM.W
Compiling C-LIMPEDOCTASCTES.W
** "qledcta0.i2" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-LIMPEDOCTASCTES.W Could not understand line 625. (193)
** "qledcta0.i1" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-LIMPEDOCTASCTES.W Could not understand line 627. (193)
** "qledcta0.i1" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-LIMPEDOCTASCTES.W Could not understand line 664. (193)
** "qledcta0.i1" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-LIMPEDOCTASCTES.W Could not understand line 699. (193)
** "qledcta0.i1" was not found. (293)
** d:\progre~1\appl_c~1\fuentes\C-LIMPEDOCTASCTES.W Could not understand line 740. (193)
** d:\progre~1\appl_c~1\fuentes\C-LIMPEDOCTASCTES.W Could not understand line 740. (193)
Compiling C-LMEMOACUSECHEQ.W
Compiling C-LMEMOAUTOCHQ.W
Compiling C-LMEMOFONDOSCASA.W
Compiling C-LMEMOTESTBAR.W
```

CAPITULO 4. DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA

- Compiling C-LMOVAUTO.W
- Compiling C-LOGINISTEMA.W
- Compiling C-LPAGOCHEQ.W
- Compiling C-LPAGOPREFORMA.W
- Compiling C-LREMESALIB.W
- Compiling C-LRESUCASACAM.W
- Compiling C-LSALDOCUENTAS.W
- Compiling C-LSALDOSRANGO.W
- Compiling C-LSDOCTACHES.W
- Compiling C-LSOBREGIROS.W
- Compiling C-LSOLICCHEQPROV.W
- Compiling C-LTIPOMOV.W
- Compiling C-LTOTALDEBIT.W
- Compiling C-LVALIDAT2.W
- ** "mmvalin0.i2" was not found. (293)
- ** d:\progre~1\appl_c~1\fuentes\C-LVALIDAT2.W Could not understand line 137: (193)
- ** Unknown Field or Variable name - t_n_reimpresion. (201)
- ** d:\progre~1\appl_c~1\fuentes\C-LVALIDAT2.W Could not understand line 422. (196)
- Compiling C-MANTREMESA.W
- Compiling C-MENCHE.W
- Compiling C-MENCTA.W
- Compiling C-MENDIA.W
- Compiling C-MENINI.W
- Compiling C-MENMEM.W
- Compiling C-MENMEN.W
- Compiling C-MENPAG.W
- Compiling C-MENPAR.W
- Compiling C-MOVAUTO.W
- Compiling C-PAGOCHEQCTROFIN.W
- Compiling C-REMESALIB.W
- Compiling C-STOPPAMENT.W
- Compiling C-VALPAG.W
- Compiling COPIA DE C-CONMOVCHG.W
- Compiling D-ABC_AREA_DIV_ETC.W
- Compiling D-ABC_ASESORES.W
- Compiling D-ABC_BANCOS.W
- Compiling D-ABC_CENTROS_FINANCIEROS.W
- Compiling D-ABC_CUENTAS.W
- Compiling D-ABC_DIAS_INHABILES.W
- Compiling D-ABC_OVERNIGHT_DE_CHEQUERAS.W
- Compiling D-ABC_PARAMETROS_DE_OPERACION.W
- Compiling D-ABC_PREFORMATOS.W
- Compiling D-ABC_PROMOTORES.W
- Compiling D-ABC_REMESA.W
- Compiling D-ABC_SUCURSALES.W
- Compiling D-ABC_TIPOS_DE_MOVIMIENTOS.W
- Compiling D-ESTADÍSTICAS POR CLIENTE.W
- Compiling D-IMPRESIÓN DE CTAS. CON CH. CONGELADOS.W
- Compiling D-OPCIONES DEL SISTEMA DE CHEQUES.W
- Compiling Q-ABC-PAIS.W
- Compiling Q-BANCO.W
- Compiling Q-CTROFIN.W
- Compiling Q-INHABIL.W
- Compiling Q-JOIN_M_SUCURSAL_M_UBICACION.W
- Compiling Q-JOIN_U_ASESOR_U_CTRO_FIN.W
- Compiling Q-ONCHEQ.W
- Compiling Q-PAIS.W
- Compiling Q-PARAM.W
- Compiling Q-PREFOR.W

```
Compiling Q-PREFORMATO.W
Compiling Q-PROMO.W
Compiling Q-REMESA.W
Compiling Q-SUCU.W
Compiling Q-TABLA_M_SUCURSAL.W
Compiling Q-TABLA_U_ASESOR.W
Compiling Q-TABLA_U_CLIENTE.W
Compiling Q-TPMOV.W
Compiling Q-UBICA.W
Compiling V-ABC-PAIS.W
Compiling V-BANCO.W
Compiling V-CTROFIN.W
Compiling V-ID_PRODUCT_Y_ID_CLIENTE.W
Compiling V-IMPRCTASCONCHCONGXCLIENTE.W
Compiling V-IMPRCTASCONCHCONGXFECHAS.W
Compiling V-INHABIL.W
Compiling V-ONCHEQ.W
Compiling V-PAIS.W
Compiling V-PARAM.W
Compiling V-PREFOR.W
Compiling V-PREFORMATO.W
Compiling V-PREFORMATO2.W
Compiling V-PREFORMATO3.W
Compiling V-PROMO.W
** "PROM_V4.V" was not found. (293)
** C:\ARCHIV~1\PROGRESS\src\adm\method\tableio.i Could not understand line 1. (193)
** Unable to analyze dictionary validate expression for field %. (575)
** C:\ARCHIV~1\PROGRESS\src\adm\method\tableio.i Could not understand line 1085. (196)
Compiling V-TABLA_M_SUCURSAL.W
Compiling V-TABLA_M_UBICACION.W
Compiling V-TABLA_U_ASESOR.W
Compiling V-TABLA_U_CLIENTE_PARTE1.W
Compiling V-TABLA_U_CLIENTE_PARTE2.W
Compiling V-TABLA_U_CLIENTE_PARTE3.W
Compiling V-TABLA_U_CTRO_FIN.W
Compiling V-TPMOV.W
Compiling W-ABC-PAIS.W
Compiling W-ABCREMESAS.W
Compiling W-ONCHEQ.W
Compiling W-PARAM.W
Compiling W-SUCU.W
Compiling W-TPMOV.W
Compiling W-UBICA.W
11:44:34 Looking for *.i in d:\progre~1\appl_c~1\fuentes
Compiling C-CANMOVCHC.I
** Unknown Field or Variable name - x_bien. (201)
** d:\progre~1\appl_c~1\fuentes\C-CANMOVCHC.I Could not understand line 9. (196)
Compiling C-CANMOVCHG.I
** Unknown Field or Variable name - x_bien. (201)
** d:\progre~1\appl_c~1\fuentes\C-CANMOVCHG.I Could not understand line 9. (196)
Compiling C-REMESALIB.I
** Unknown Field or Variable name - t_band. (201)
** d:\progre~1\appl_c~1\fuentes\C-REMESALIB.I Could not understand line 1. (196)
Compiling IMP.I
```

Compilation Finished.

Para todos los mensajes "xxxxxx was not found" y "Unknown Field or Variable name - xxxxxx", se eliminaron las variables y nombres a las que se hacía referencia, ya que se trataba de variables de prueba que no se requerían más. En el reporte 4.5_2 se muestran los resultados de la compilación final sin errores.

Reporte 4.5_2. Resultados de la Compilación Final

13:46:22 Looking for *.p in d:\progre~1\appl_c~1\fuentes
Compiling C-CONTAB.P
Compiling C-CONTABDESCOTAB.P
Compiling C-DESCON.P
Compiling C-LCATALOGOCTA.P
Compiling C-LCTASALTASCANC.P
Compiling C-LCTASALTASCANCPROM.P
Compiling C-LCTASCHEQCONG.P
Compiling C-LEDOCT.P
Compiling C-LINGGASUTI.P
Compiling C-LPERFILCTE.P
13:46:24 Looking for *.w in d:\progre~1\appl_c~1\fuentes
Compiling B-ABC-PAIS.W
Compiling D-ABC_CUENTAS.W
Compiling C-ACTSDC.W
Compiling C-BAJAMOVCHEMGLO.W
Compiling C-BAJAMOVFAX.W
Compiling C-BORRARPAGOECHEQCTROFIN.W
Compiling C-CANCELACTA.W
Compiling C-CANCMOVMAS.W
Compiling C-CAPINSTPAGOCTE.W
Compiling C-CAPMOVCHEMCTE.W
Compiling C-CAPMOVCHEMGLO.W
Compiling C-CAPMOVFAX.W
Compiling C-CAPOTROSMOVTOS.W
Compiling C-CIERREMES.W
Compiling C-CONGELARCTA.W
Compiling C-CONMOVEDOCTA.W
Compiling C-DEBOCHECTE.W
Compiling C-ESTADCLIENTE.W
Compiling C-GENMOVCIERRE.W
Compiling C-GENNUMCHEQ.W
Compiling C-LCARTACHEQREB.W
Compiling C-LCHEQCOBCTRFIN.W
Compiling C-LCHEQCOBRACROSFINAN.W
Compiling C-LCHEQCTROFINAN.W
Compiling C-LCUADRARCONTAB.W
Compiling C-LIBPAGOCHেম.W
Compiling C-LIMPEDOCTASCTES.W
Compiling C-LMEMOACUSECHEQ.W
Compiling C-LMEMOAUTOCHেম.W
Compiling C-LMEMOFONDOSCASA.W
Compiling C-LMEMOTESTBAR.W
Compiling C-LMOVAUTO.W
Compiling C-LOGINISTEMA.W
Compiling C-LPAGOCHেম.W
Compiling C-LPAGOPREFORMA.W
Compiling C-LREMESALIB.W
Compiling C-LRESUCASACAM.W
Compiling C-LSALDOCUENTAS.W
Compiling C-LSALDOSRANGO.W

Compiling C-LSDOCTACHES.W
Compiling C-LSOBREGIROS.W
Compiling C-LSOLICCHEQPROV.W
Compiling C-LTIPOMOV.W
Compiling C-LTOTALDEBIT.W
Compiling C-LVALIDAT2.W
Compiling C-MENCHE.W
Compiling C-MENCTA.W
Compiling C-MENDIA.W
Compiling C-MENIN1.W
Compiling C-MENMEM.W
Compiling C-MENMEN.W
Compiling C-MENPAG.W
Compiling C-MENPAR.W
Compiling C-MOVAUTO.W
Compiling C-PAGOCHEQCTROFIN.W
Compiling C-REMESALIB.W
Compiling C-STOPPAYMENT.W
Compiling D-ABC_AREA_DIV_ETC.W
Compiling D-ABC_ASESORES.W
Compiling D-ABC_BANCOS.W
Compiling D-ABC_CENTROS_FINANCIEROS.W
Compiling D-ABC_DIAS_INHABILES.W
Compiling D-ABC_OVERNIGHT_DE_CHEQUERAS.W
Compiling D-ABC_PROMOTORES.W
Compiling D-ABC_TIPOS_DE_MOVIMIENTOS.W
Compiling Q-ABC-PAIS.W
Compiling Q-JOIN_U_ASESOR_U_CTRO_FIN.W
Compiling Q-TABLA_U_ASESOR.W
Compiling V-ABC-PAIS.W
Compiling V-TABLA_U_ASESOR.W
Compiling V-TABLA_U_CTRO_FIN.W
Compiling V-TPMOV.W
Compiling W-ABC-PAIS.W
Compiling W-ABCREMESAS.W
Compiling W-PARAM.W
Compiling B-TABLA_Q_CORTE.W
Compiling B-REMESA.W
Compiling V-TABLA_U_CLIENTE_PARTE3.W
Compiling D-ABC_PREFORMATOS.W
Compiling D-ABC_REMESA.W
Compiling D-ABC_SUCURSALES.W
Compiling D-ESTADÍSTICAS POR CLIENTE.W
Compiling D-IMPRESIÓN DE CTAS. CON CH. CONGELADOS.W
Compiling D-OPCIONES DEL SISTEMA DE CHEQUES.W
Compiling Q-BANCO.W
Compiling Q-CTROFIN.W
Compiling Q-INHABIL.W
Compiling Q-JOIN_M_SUCURSAL_M_UBICACION.W
Compiling Q-ONCHEQ.W
Compiling Q-PAIS.W
Compiling Q-PARAM.W
Compiling Q-PREFOR.W
Compiling Q-PREFORMATO.W
Compiling Q-PROMO.W
Compiling Q-REMESA.W
Compiling Q-SUCU.W
Compiling Q-TABLA_M_SUCURSAL.W
Compiling Q-TABLA_U_CLIENTE.W

Compiling Q-TPMOV.W
Compiling Q-UBICA.W
Compiling V-BANCO.W
Compiling V-CTROFIN.W
Compiling V-ID_PRODUCT_Y_ID_CLIENTE.W
Compiling V-IMPRCTASCONCHCONGXCLIENTE.W
Compiling V-IMPRCTASCONCHCONGXFECHAS.W
Compiling V-INHABIL.W
Compiling V-ONCHEQ.W
Compiling V-PAIS.W
Compiling V-PARAM.W
Compiling V-PREFOR.W
Compiling V-PREFORMATO.W
Compiling V-PREFORMATO2.W
Compiling V-PREFORMATO3.W
Compiling V-PROMO.W
Compiling V-TABLA_M_SUCURSAL.W
Compiling V-TABLA_M_UBICACION.W
Compiling V-TABLA_U_CLIENTE_PARTE1.W
Compiling V-TABLA_U_CLIENTE_PARTE2.W
Compiling D-ABC_PARAMETROS_DE_OPERACION.W

Compilation Finished.

4.5.3 Pruebas del Sistema

4.5.3.1 Pruebas de Recuperación

Se llevaron a cabo estas pruebas repitiendo el siguiente procedimiento en sucesivas ocasiones.

- Se obtuvo una copia de respaldo de la base de datos.
- Se comenzó a ingresar datos en algún módulo escogido al azar dentro de los menús de la aplicación.
- Antes de terminar de ingresar todos los datos al módulo se forzó a la aplicación a fallar de las siguientes formas:
 - Interrupción de la aplicación mediante el administrador de tareas del sistema operativo (Ctrl+Alt+Del+Finalizar Tarea).
 - Apagado del equipo directamente, sin cerrar antes ninguna aplicación ni tampoco el sistema operativo, simulando una pérdida de energía eléctrica o una falla en el servidor o en la computadora del usuario (cliente).
- Al reiniciar la aplicación y el sistema se obtuvo otra copia de la base de datos.

- Se verificó la integridad de la base de datos, comparando ambas copias mediante el programa "DiskDupe", que es un "shareware" (programa de uso compartido) de comparación y duplicación.

Como resultado se encontró que los mecanismos de inicialización y recuperación del estado del sistema, de recuperación de datos y de reinicio son aceptables. En todos los casos se encontró que ambas versiones o copias de la base de datos eran idénticas y por tanto que los mecanismos de recuperación de Progress son suficientes para garantizar la integridad y recuperación de los datos.

4.5.3.2 Pruebas de Seguridad

Los mecanismos de seguridad que proporciona la aplicación se basan en la pantalla de "login" (registro de usuario) y en la tabla correspondiente de usuarios autorizados.

Cada usuario autorizado debe conectarse a la base de datos (lo cual hace la pantalla de "login") antes de poder tener acceso a ella mediante programas de la aplicación o de Progress; después de esto la pantalla de "login" pasa el control al menú principal, de forma que el usuario solo pueda navegar por los menús predefinidos de la aplicación.

Sin embargo, la mayor parte de la seguridad y protección de los datos y programas de la aplicación (y de Progress), reside en la administración de recursos y seguridad del sistema operativo servidor (Windows NT). Para probar ambos tipos de seguridad se realizó el siguiente procedimiento:

- Se definieron los usuarios TESIS1 y TESIS2 en la tabla de usuarios autorizados de la aplicación.
- Se crearon en el sistema operativo del servidor los mismos usuarios TESIS1 y TESIS2 y se definieron en el grupo "TESIS". Se dio acceso a este grupo a los programas y base de datos de la aplicación, así como a los programas de Progress.
- Se creó otro grupo llamado "Default" sin acceso a ninguno de estos recursos.
- Usando alguno de los nombres de usuarios autorizados, se intentó ejecutar los diferentes módulos de la aplicación sin pasar por la pantalla de "login". Sólo se pudo hacer en los casos en que se tenía instalado el Software de desarrollo de aplicaciones de Progress (en el servidor), haciendo la conexión a la base de datos directamente mediante sus utilerías de bases de datos.
- Usando alguno de los nombres de usuarios **no** autorizados se intentó ejecutar los diferentes módulos de la aplicación, tanto a través de la pantalla de "login" como directamente, siendo impedido en todos los casos por el sistema operativo.

Conclusiones y Recomendaciones Relativas a la Seguridad

- Un usuario autorizado puede ejecutar los programas de la aplicación y de Progress, así como tener acceso a la base de datos de la aplicación y modificarla, sin usar la pantalla de "login", siempre y cuando tenga instalado y conozca el Software y utilerías de Progress y conozca también los nombres de los módulos de la aplicación, por lo que estos

recursos no deben estar disponibles más que para el administrador del sistema, el administrador de seguridad y/o el administrador de los datos.

- Se debe dar acceso a los usuarios a los programas de la aplicación y de Progress solo para lectura.
- Se deben habilitar las facilidades de auditoria de seguridad tanto del sistema operativo servidor como del de los clientes, para monitorear cualquier intento exitoso o no de modificar la base de datos.
- Se recomienda que el sistema operativo de los clientes de la red sea uno que tenga mecanismos de seguridad similares a los del sistema operativo servidor, por ejemplo: Windows NT Workstation.
- No hay forma de que los usuarios no autorizados tengan acceso a recursos de la aplicación, a menos que lleguen a conocer el nombre y clave de algún usuario autorizado.
- La protección contra acciones indebidas del administrador del sistema, de la seguridad y/o de los datos la proporciona el sistema operativo al comunicar al usuario propietario de los recursos (en este caso programas o base de datos de la aplicación) cuando el administrador ha tomado posesión de un recurso suyo y ha tenido acceso a él. Por tanto, se deben manejar dos copias de respaldo de la aplicación bajo la responsabilidad cada una de un administrador diferente, como por ejemplo el administrador del sistema y el administrador de los datos.

4.5.3.3 Pruebas de "Stress" (Resistencia) y de Rendimiento

Se hicieron dos tipos de pruebas. El primero consistió en oprimir teclas y combinaciones de teclas que se supone no deben usarse en la aplicación, de forma continua y rápida. La única ocasión en que se logró interferir con el funcionamiento de la aplicación fue al usar la combinación de teclas Ctrl+Alt+Del, que invoca al administrador de tareas del sistema operativo y permite finalizar la aplicación en forma anormal.

El segundo tipo de pruebas fue abrir en la computadora de un usuario (cliente de la red) diferentes ventanas y aplicaciones (procesadores de texto, hojas de cálculo, acceso a Internet, navegadores, reproductores de medios, interfaz de comandos), al mismo tiempo que la aplicación que nos ocupa. No se encontró ninguna falla en la aplicación, excepto que conforme se continuó utilizando nuevas ventanas y aplicaciones, su tiempo de respuesta se degradó al mismo tiempo que para el resto de los programas.

Se midió durante este proceso la utilización del sistema usando el monitor del sistema, que es una facilidad del sistema operativo que mide en tiempo real la utilización de CPU, memoria y dispositivos de E/S. Dichas utilizaciones tuvieron picos debajo del 30% la mayor parte del tiempo, saltando hasta el 100% solo cuando el número de programas y ventanas abiertos hizo que todos los programas se congelaran. Estos resultados pueden esperarse en cualquier computadora que se utilice como cliente de red con los recursos necesarios para correr el sistema operativo Windows98 (usado en estas pruebas) o Windows NT Workstation.

4.5.4 Configuración de los Equipos de Prueba

Las características relevantes para este proyecto de las máquinas utilizadas en las pruebas anteriores son las siguientes.

Servidor

Procesador	- Pentium II MMX @ 350Mhz
Memoria Cache	- L2 de 512KB
Memoria	- 64MB RAM
Disco Duro	- Partición de 3GB con sistema de archivos NTFS
Sistema Operativo	- Windows NT Server, versión 4.0
Adaptador de red	- HP Ethernet 10/100

Cliente

Procesador	- Pentium II MMX @ 233Mhz
Memoria Cache	- L2 de 512KB
Memoria	- 32MB RAM
Disco Duro	- 3.2GB con sistema de archivos FAT
Sistema Operativo	- Windows 98, versión 4.10
Adaptador de red	- HP Ethernet 10/100

4.6 Factibilidad Técnica y Operativa

El sistema de cuenta de cheques en dólares, es una cuenta eje para el acceso a todos los productos y servicios de una Institución Financiera del ámbito internacional, a la que se le pueden instruir gastos o abonos para tarjeta de crédito, inversiones a plazo o crédito.

Esta cuenta de cheques, está denominada en dólares americanos, es pagadera en Estados Unidos de Norteamérica a través de Barclay's Bank en Miami Florida y paga intereses de mercado mensualmente sobre saldo promedio mínimo.

Para satisfacer las necesidades que la empresa nos solicitaba fue necesario hacer un estudio de factibilidad técnica / operativa el cual fue realizado, por una área especializada del banco.

El sistema de cuenta de cheques en dólares requiere que para cada cliente (usuario que tiene acceso al sistema) se cubran los siguientes puntos para su implementación.

- Primeramente a los clientes se les debe de instalar el acceso telefónico o bien el de red para poder conectarse al servidor que contiene la aplicación; si no cuentan con tales recursos no podrán hacer uso de la aplicación.
- A cada usuario se le deben dar todas las especificaciones como son: la dirección de red, el "login" (identificación) y el "password" (clave de acceso) para el servidor que les dará entrada al sistema, además de contar con una infraestructura de comunicaciones.
- Todo usuario que intente entrar al sistema deberá de poseer un "login" y "password" válidos para el sistema.
- Se debe tener instalado el software "run-time" (para usar en tiempo de ejecución) de Progress, que es necesario para que los clientes puedan hacer uso de la aplicación. De no contar con este software será imposible que puedan usarla.

Todo lo anterior se debe tener contemplado para que los clientes puedan hacer uso de la aplicación. En caso de faltar alguno de estos requerimientos, será casi imposible que los usuarios puedan conectarse al servidor y hacer uso de la aplicación.

Por parte del servidor los mecanismos con que se debe contar para hacer funcionar la aplicación son los siguientes.

- El servidor debe de tener instalado el sistema Windows NT Server como sistema operativo, debido a que la licencia que se tiene de Progress es para este sistema.

- Tener instalado el Software del Service Pack (paquete de servicio) 3.+ o superior para Windows NT, que es el que requerimos para poder realizar la comunicación con los clientes remotos además de contar con una infraestructura de comunicaciones.
- Una vez que se ha instalado el anterior software y se ha verificado su correcto funcionamiento, se deben crear los diferentes grupos de usuarios que tendrán acceso al sistema.
- Una vez creados los grupos de trabajo, se procederá a definir todos los usuarios permitidos que harán uso de la aplicación, donde a cada uno se le dará un "login" y un "password" para poder conectarse. Los usuarios serán dados de alta de acuerdo al perfil de cada uno de los grupos de trabajo con que fueron creados.
- Se procederá a realizar la instalación de Progress usando la licencia con que se cuenta. Para esto se debe crear una estructura en el servidor para contener el producto sin mezclar su ambiente de operación con el de otro software que se tenga, para tener así una administración ordenada del Software instalado en el servidor.
- Una vez instalado todo el manejador de base de datos de Progress se procederá a instalar la aplicación (sistema de cheques en dólares) en el servidor. De igual manera se creará una estructura de directorios en el servidor en donde ubicar: las base de datos, los programas fuentes, los programas compilados y los documentos a los cuales se haga referencia.

Una vez que, tanto el software de servidor como de cliente ya están instalados con todos los componentes que se requieren para hacer uso de la aplicación, se deben realizar las pruebas de comunicación de cada uno de los usuarios que fueron dados de alta en el servidor.

La infraestructura de la red dentro de las sucursales y dentro de la red del área de Sistemas en donde se instalará el servidor ya esta implementada y cumple con la normatividad establecida por la Institución para las redes que utilicen Windows NT con sistema operativo de red dentro de la Institución.

Dada esta normatividad, la implementación del servidor a utilizar para el sistema de cuenta de cheques en dólares debe cumplir en su totalidad con la misma.

A continuación se presenta la normatividad de la empresa para redes con sistema operativo Windows NT.

4.6.1 Normatividad para Redes bajo Windows NT como Sistema Operativo de Red en la Institución

Esta normatividad deberá seguirse por todas aquellas personas encargadas directamente de la instalación y puesta a punto de servidores Windows NT Server. El cumplimiento de la misma, garantiza la operación óptima del servidor y evita cualquier problema con la interconexión, además de permitir la creación de una plataforma eficiente y estandarizada para proporcionar los servicios de red corporativos. La presente normatividad esta enfocada fuertemente a lograr reducir al máximo las labores relacionadas con la administración de redes.

El cumplimiento de está normatividad permitirá:

- Una operación óptima de los servidores
- Evitar conflictos de operación
- La integración e interconexión en la red
- Facilitar actualizaciones futuras
- Simplificar la estructura de los parámetros de operación
- Facilitar el soporte técnico.

4.6.1.1 Requerimientos del Servidor

La parte más importante y crítica de una red la constituye el servidor. La función de éste es la de compartir sus recursos con la red, ya sean estos servicios de disco, impresión o aplicaciones cliente-servidor. También debe proporcionar una plataforma robusta para la ejecución de aplicaciones de misión crítica. El servidor departamental esta enfocado a la ejecución de los servicios básicos de red (archivos, impresión y software institucional).

Por lo anterior, se hace necesario que la computadora destinada a ser servidor, cumpla con los requerimientos para realizar en forma eficiente todas las funciones anteriormente mencionadas. Estos incluyen los requerimientos mínimos de hardware y software, sus requerimientos eléctricos y su ubicación física.

Requerimientos Mínimos de Hardware del servidor

- Servidor Pentium III (500 Mhz)
- Memoria RAM ECC de 128 MB
- Unidad CD-ROM 32x(SCSI) interna iniciadora ("bootable")
- Unidad de respaldo de 24 GB (SCSI) - interna
- Dos Discos duros de 9 GB (SCSI)
- Bahías disponibles para crecimiento futuro
- Ranuras de expansión libres para crecimiento futuro

- Ranuras de expansión libres PCI
- Tarjeta de red PCI (Operación Dual 10BASET / 100BASETX)
- Puertos seriales y 1 puerto paralelo
- Mouse (bus)
- Monitor SVGA a color 15"
- UPS con interfase al servidor (1200 Watts)

Es importante señalar que las marcas y modelos a adquirir (CPU, UPS, tarjetas de red) estén registrados en el HCL (Hardware Compatibility List) de Windows NT Server, además de ser el estándar en este momento de la Institución Bancaria.

Requerimientos Mínimos de Software del servidor

- Windows NT Server v4.x -English-
- Windows NT Service Pack4 -English-

Requerimientos eléctricos del servidor

El servidor debe estar alimentado por un UPS, y a su vez contar con la interfaz a este para protegerlo de forma automática en caso de falla en el suministro eléctrico. Gracias a esta interfaz, es posible salvaguardar en forma óptima el servidor, evitando así pérdida de información y posibles daños al equipo.

Ubicación física del servidor

El servidor debe ser colocado en un área exclusiva y libre del paso (de preferencia en un lugar cerrado, y con aire acondicionado), con el fin de que pueda ser operado sin dificultad cuando sea necesario.

4.6.1.2 Requerimientos para las Estaciones de Trabajo

La estación de trabajo es la computadora donde el usuario ejecuta aplicaciones y desde donde tiene acceso y utiliza los recursos del servidor. Para ejecutar todo el software institucional de manera rápida y eficiente las estaciones de trabajo deben cumplir con ciertos requerimientos mínimos de hardware y de software, aprobados por el Departamento de estandarización del Banco.

Requerimientos Mínimos de Hardware para las Estaciones de Trabajo

- Procesador Pentium II a 366 Mhz.
- Memoria RAM de 64 MB
- Disco duro de 8GB
- ranuras de expansión libres ISA
- ranuras de expansión libres PCI
- Tarjeta de red PCI (Operación Dual 10BASET / 100BASETX)

- Monitor a color SVGA 14".
- Mouse (bus)
- puertos seriales y 1 puerto paralelo

Requerimientos Mínimos de Software para las Estaciones de Trabajo

- Windows NT Workstation 4.x ó Windows 98.
- Asimismo deberán utilizar la configuración de red en modo protegido (32 bits).

4.6.1.3 Impresoras

Aquellas impresoras que sean conectadas directamente a la red (impresoras con tarjeta de red a través de una interfaz externa) deberán soportar (y por lo tanto configurarse) con el protocolo TCP/IP. Las impresoras y los respectivos adaptadores de red tienen que estar en el HCL de Windows NT y Windows 98.

4.6.1.4 Puesta a Punto

Una vez que se tienen listos el servidor, las estaciones de trabajo e impresoras con sus respectivas tarjetas de red, configuradas e interconectadas a través del cableado, se debe realizar la instalación y configuración inicial del servidor.

4.6.1.5 Instalación

Los siguientes son los primeros parámetros que deberán ajustarse:

Directorio de Instalación de Windows NT	Modo de licenciamiento
C:\WINNT	Per SEAT

Tabla 4.6_1 Parámetros de instalación

Se requiere que la partición del sistema (C) tenga al menos 2GB de espacio.

Antes de proceder con la instalación, se deberá contar con un nombre y rol asignados, y la configuración TCP/IP correcta. Se deberán conocer asimismo los servicios a utilizar y asegurarse de que el servidor este apropiadamente conectado a un nodo operacional.

Nombre del Servidor y Rol Asignado

El servidor debe contar con un nombre único que lo identifique y evite problemas de comunicación, hacia otras redes.

Un servidor Windows NT puede configurarse bajo tres esquemas diferentes de seguridad como PDC ("Primary Domain Controller" o controlador primario del dominio), BCD ("Backup Domain Controller" o controlador de dominio de respaldo) y como Server ("Stand-alone Server" o servidor único). Este esquema de seguridad es conocido como Dominio. El nombre del servidor, así como el rol asignado deberá solicitarse y registrarse en la Gerencia de Sistemas.

Nombres de las Estaciones de Trabajo y Grupo de Trabajo

El nombre asignado a las estaciones de trabajo (nombre NETBIOS) debe ser igual al "login-name" del usuario. Si algún usuario tiene asignado más de un equipo se deberán utilizar números consecutivos, empezando por el número dos.

Nombres de Cuenta de Usuarios ("login-name")

El nombre del usuario ("login-name") se formará con un máximo de 8 caracteres (todos ellos deben ser caracteres ASCII de 7 bits, esto significa que deberán utilizar caracteres con la "ñ" y letras acentuadas).

Nombres de Recursos

Un recurso es cualquier dispositivo interconectado a la red, como una impresora, disco duro, unidad de CD-ROM y que puede ser compartido.

Ahora bien, con el fin de asegurar el acceso homologado a ciertos recursos de los servidores Windows NT, se deberán utilizar los nombres que a continuación se señalan.

CDROM	CDROM2	EVERYONE EVERYBODY
Primera unidad de CD-ROM	Segunda unidad de CD-ROM	Directorios de acceso común (\groups\everyone\everybody)

Tabla 4.6_2 Nombres de los recursos reservados

Para el caso de los nombres de recursos asignados a impresoras la terminología se deja abierta.

Dominios

En Windows NT, al conjunto de servidores que comparten una base de datos de usuarios común se llama dominio. Un dominio provee acceso centralizado a las cuentas de usuarios y de grupos. Todo esto es controlado por un administrador de dominio. Cada dominio debe tener un nombre único.

Ahora bien, el configurar servidores NT bajo un mismo dominio es una poderosa herramienta que facilita la administración del sistema, ya que posibilita "ver" a un conjunto de servidores NT como si fueran un sistema único.

En una seguridad basada en dominios, un servidor NT puede jugar uno de tres papeles: como PDC (Primary Domain Controller), BDC (Backup Domain Controller) y como Server (Stand-alone Server). Solo puede haber un PDC en un dominio. Todos los dominios, así como el rol desempeñado por cada servidor Windows NT instalado se deberán solicitar y registrar en la Gerencia de Sistemas.

4.6.1.6 Configuración TCP/IP

Uno de los puntos claves en la arquitectura bajo la cual se diseñó la red de servidores corporativos utilizando Windows NT Server es la capacidad de este sistema de realizar todos sus servicios a través del protocolo TCP/IP. Esto está estrechamente relacionado con los proyectos que permiten construir una red WAN uni-protocolo, lo cual representa incrementar sustancialmente la red y reducir los tiempos de administración de ésta al simplificar enormemente los parámetros de operación y configuración, para lograr así una red más robusta, segura y eficiente.

Por tal motivo, solo deberá habilitarse el protocolo TCP/IP en el servidor. Otros protocolos como IPX/SPX, NETBEUI, DLC no deberán instalarse.

Los servidores Windows NT deben tener las direcciones de "hosts" en el rango del 245 al 249, configurarse bajo el dominio DNS Internet <nombre de la institución> .net, con un "hostname" igual al nombre NetBIOS ("computer name") del servidor y apuntar hacia los servidores de resolución de nombres WINS / DNS registrados.

Servicios DHCP / WINS / DNS

El servicio de DHCP ("Dynamic Host Configuration Protocol") ofrece la posibilidad de configurar automáticamente todos los parámetros TCP/IP en las estaciones de trabajo (Windows NT Workstation y Windows 98). DHCP provee una manera segura, confiable y simple para configurar TCP/IP en la red a la vez que ayuda a prevenir conflictos de direcciones, ayudando así a conservar el uso de las direcciones IP y posibilitando la administración centralizada de esta función. Actualmente el rango de direcciones de 13.x.x.120 a 13.x.x.219 esta reservada para la asignación de direcciones dinámicas.

WINS (Windows Internet Name Service) y DNS son los servicios de resolución de nombres. Estos se utilizan para traducir nombres NETBIOS (los nombres utilizados por los servidores y estaciones de trabajo) a direcciones IP.

Configuración de las estaciones de trabajo

Los clientes que se recomiendan para trabajar en redes NT son Windows NT Workstation y Windows 98. Es importante que estos sistemas se configuren con el software de red apropiado, es decir, que se utilicen "drivers" (controladores) de red en modo protegido (32 bits) específicamente diseñados para el sistema operativo que se use.

Todos los clientes deben configurarse con la suite de TCP/IP y de preferencia configurarse a través de los servicios DHCP (siempre que sea posible).

Así mismo, para asegurar la correcta resolución de nombres de la red se deberán configurar los servidores primarios y secundarios de WINS y DNS.

4.6.2 Puesta a punto del servidor del Sistema de Cheques en Dólares

La puesta a punto del servidor donde se instalará la aplicación contempla todas y cada una de las normas establecidas y presentadas anteriormente. El nombre asignado al servidor tentativamente será el siguiente: CHEQUES.

De forma general los requisitos mínimos de hardware y software son cumplidos satisfactoriamente. La ubicación del servidor será la misma en donde esta actualmente el servidor del área de sistemas y que cumple con las condiciones de clima y eléctricas establecidas en la normatividad.

La estructura de directorios y nombres de dispositivos también cumple con las normas establecidas.

4.6.3 Instalación de Progress

La instalación de Progress v8.3 es sumamente sencilla ya que cuenta con un "script" (guión) de instalación, que se inicia automáticamente a través del archivo setup.exe. Se deberá realizar la carga total del software.

La instalación del sistema de cheques en dólares, consistirá en pasar del equipo de prueba actualmente utilizado y en donde actualmente está el sistema, al servidor CHEQUES una vez que se haya puesto a punto.

La instalación del software Progress y el Sistema de Cheques en Dólares, en cumplimiento con la normatividad para los servidores Windows NT, se instalará bajo el directorio " \APPS " bajo los directorios PRG (PROGRESS) y CHEQUES (Sistema de Cheques en Dólares).

4.6.4 Mantenimiento

Una parte fundamental para el buen funcionamiento y posible crecimiento de un sistema se basa en el mantenimiento que a este se le brinde, a continuación se presenta la descripción de algunos tipos de mantenimiento.

4.6.4.1 Mantenimiento Perfectivo

Comprende los cambios solicitados al programador del sistema.

A medida que se usa el software, se reciben de los usuarios recomendaciones sobre nuevas posibilidades acerca de modificaciones a funciones ya existentes. Para satisfacer estas peticiones se lleva a cabo el mantenimiento perfectivo.

4.6.4.2 Mantenimiento Preventivo

Previene errores. Este mantenimiento se proporciona depurando el sistema cada cierto tiempo con el fin de mejorar algún proceso.

Se puede considerar como mantenimiento preventivo el que se da a la información que se maneja para garantizar que los resultados dados por el sistema sean los correctos.

4.6.4.3 Mantenimiento Adaptativo

Se debe a cambios en el ambiente del programa, y a la adaptación de nuevas unidades o módulos, así como al cambio en la tecnología de hardware y software.

La vida útil estimada del software de aplicación puede fácilmente sobrepasar los diez años, pero considerando la evolución del ambiente, en la práctica éste puede volverse obsoleto en un par de años.

De este mantenimiento pueden derivarse los siguientes mantenimientos:

- **Mantenimiento aumentativo.** Se da cuando se incluyen nuevas funciones que no se contemplan al inicio del desarrollo del sistema y surgen como una necesidad del usuario.
- **Mantenimiento tecnológico.** Se da debido a los cambios importantes de Hardware y Software, es decir, al avance de los sistemas operativos, lenguajes de programación y nuevas generaciones de hardware, con los cuales se mejoran las herramientas de trabajo, las versiones de software, equipos periféricos y otros elementos de sistemas.

4.6.4.4 Mantenimiento correctivo

El proceso que incluye el diagnóstico y corrección de uno o más errores en el sistema, se denomina mantenimiento correctivo. Durante el uso del sistema se encontrarán errores, los cuales deben ser informados al equipo de desarrollo.

La primera actividad del mantenimiento se da ocasionalmente cuando la prueba del software no haya descubierto todos los errores latentes de un sistema.

4.6.5 Mecanismos de Respaldo y Protección

Dentro de la problemática que se suele presentar en todo sistema computacional, la relacionada con los casos de contingencia o bien con los casos de fallos del mismo se debe tener en cuenta. Para contar con mecanismos de protección de la información y del sistema se debe tener presente lo siguiente.

- Como la información es muy valiosa y es vital para la institución financiera se deben tener mecanismos de respaldo de la misma. Dado que la base de datos contiene toda la información de los cuenta-habientes, es necesario contar con mecanismos de replicación de todos los movimientos que se estén realizando en el servidor hacia otros servidores, de manera que si se llegasen a presentar problemas con este, se cuente con otro servidor de respaldo con la información casi completa, a excepción de las últimas transacciones que se estuvieran realizando en el sistema y además de que no se deje de prestar el servicio a los usuarios.
- De igual manera, al contar con servidores de replicación se corre el riesgo de que las últimas transacciones que se estaban realizando en la base de datos se llegasen a perder, por lo que se debe hacer un chequeo de la información por parte de los usuarios, ya que son los encargados de conocer que información era la que estaban actualizando.
- Una vez que se ha terminado de operar el sistema, se deben realizar respaldos en cinta de la base de datos y de los programas con que cuenta la aplicación. Estos respaldos se deben hacer todos los días. Debido a que por cualquier causa no se pueda operar en ese servidor, se contará con la información anterior que es la que se respaldó, y esta misma se podrá montar en cualquier otro servidor para poder trabajar con el sistema.

Como en cualquier sistema que posee una base de datos se tienen límites en la capacidad del hardware, en el sentido de poder almacenar toda la información con que se cuenta, debido a que esta va creciendo día con día y es imposible mantener toda la información en línea. Además se requiere que cierta información se conserve para aclaraciones posteriores de los clientes ante la institución.

Por tales razones se cuenta además con una base de datos histórica que contiene algunas de las tablas principales del sistema, donde se grabará la información en periodos vencidos. Es decir, mensualmente se pasará la información a la base de datos histórica, que contendrá los movimientos de los clientes que soliciten sus estados de cuentas de meses anteriores.

Aún este mecanismo de respaldo de historia no basta para evitar que se consuman todos los recursos del servidor. Además, el contar con toda esa información en línea ocasionaría que las consultas a la base de datos histórica se hicieran mucho más lentas y los tiempos de espera se harían más grandes, lo que haría muy problemático el poder explotar la base de datos histórica.

Por tal motivo, cada año se tiene la necesidad de obtener respaldos de la información en otros medios que sean de mayor capacidad, para poder almacenar toda la información histórica. Ante tal necesidad los respaldos se deben de hacer en discos ópticos, que son los de mayor capacidad y a su vez muy fáciles de manejar. Una vez que se haya realizado el respaldo en estos medios y se haya comprobado que la información grabada está correcta, se procederá a depurar la información de acuerdo a lo que se respaldó.

Además de los anteriores mecanismos de seguridad, en caso de fallas externas al sistema, Progress cuenta con herramientas que se encargan de hacer un chequeo de la integridad de la información una vez que se ha restablecido el sistema.

4.6.6 Factibilidad Operativa

Un sistema puede tener éxito o fracaso determinado con tan solo la forma en que se opera y se usa, por lo tanto la capacitación o entrenamiento de los distintos usuarios es fundamental para el buen funcionamiento del sistema.

La capacitación también requiere la familiarización con los procedimientos de ejecución, lo cual implica trabajar a través de la sucesión de actividades necesarias para usar un nuevo sistema.

Existen varias técnicas para instruir y capacitar al personal en el uso y la operación del nuevo sistema entre las cuales figuran las siguientes:

- Seminarios e instrucción en grupo.
- Capacitación con base en procedimientos.
- Capacitación Personalizada.
- Simulación.
- Entrenamiento directo.
- Utilizando metodologías Modernas (video, Internet, multimedia).

En nuestro caso, la capacitación en el uso y manejo del sistema de Cheques en Dólares se estableció bajo los siguientes lineamientos:

- Al distribuir este sistema a cada sucursal, se incluye un manual de usuario en donde se le lleva de la mano, desde la instalación, manejo y soluciones de problemas.
- Se proporciona atención vía telefónica a las sucursales que solicitan ayuda al instalar o bien, al operar el sistema.
- Se deben organizar periódicamente cursos, en donde acude el personal de las sucursales a recibir la instrucción necesaria.

4.6.6.1 Capacitación a usuarios y administrador del sistema

En la capacitación a los usuarios se contemplan los siguientes puntos:

- Problemática en los procesos que desarrollan actualmente.
- Solución y/o sistema propuesto.

- Ventajas que proporciona el sistema.
- Funcionamiento del sistema de Cheques (Manual de Usuario).

El administrador del sistema deberá tomar la capacitación que se les brinde a los usuarios. Además, al administrador y a dos personas más de las áreas involucradas en el proceso directo del sistema de Cheques se les brindara la siguiente capacitación.

- Características y ventajas de Progress.
- Desarrollo e implementación del sistema.
- Administración y mantenimiento del programa.
- Manual Técnico.

A continuación se muestra el análisis efectuado del tiempo invertido en la capacitación.

Concepto	Días empleados	Unidades Requeridas	Unidad de Medición	Tiempo Total
Usuario	3	8	Hora	24 horas
Analista	3	4	Hora	12 horas
Administrador	3	4	Hora	12 horas
Total				48 horas

Tabla 4.6_3 Análisis del tiempo invertido en la capacitación.

4.6.6.2 Soporte Técnico

Durante los primeros tres meses de la puesta en marcha del sistema de Cheques, los desarrolladores del sistema brindaremos el soporte técnico necesario. Este se realizará a través del administrador del sistema. Es decir, si se encuentran errores de funcionalidad, los usuarios deberán informar al administrador del sistema con el fin de que los resuelva y en caso de no poder resolverlos, éste consultará con nosotros para que se le brinde la asesoría necesaria y/o realizar el mantenimiento correctivo y dar solución inmediata a los problemas presentados.

4.6.7 Análisis del costo y venta del sistema

El costo total del sistema puede calcularse sobre la base de los siguientes puntos.

- Por el número de líneas de código.

- Por hora-hombre de:
 - Analista
 - Diseñador
 - Programador.
- Por el tiempo de desarrollo.

El costo de una línea de código oscila entre los 25 y 35 dólares de EUA; para el sistema de Cheques se tienen aproximadamente 2,000 líneas de código. El costo del sistema de Cheques obtenido mediante esta opción y tomando el valor en línea de 35 dólares es de 70,000 dólares.

Para obtener el costo del sistema basándose en las horas-hombre utilizadas para el análisis, desarrollo e implementación del sistema, fueron considerados los siguientes puntos:

- El tiempo de desarrollo del sistema de Cheques fue de 3.5 meses desde el análisis, diseño y programación hasta la implementación del sistema.
- El analista y el diseñador del sistema trabajaron durante 1 mes, cuatro horas al día.
- Considerando los días hábiles únicamente, se tienen por mes 22 días, por lo tanto el total de horas-hombre por mes es de 88.
- Los programadores (2) trabajaron durante 2.5 meses, 4 horas al día, por lo tanto el número de horas por los 2.5 meses laborados, son 220 horas-hombre, por cada uno de ellos, dando un total de 440 horas-hombre.
- El costo por hora-hombre para los analistas, diseñadores, programadores y administradores de sistemas varía entre los 70 y 120 Dólares.

A continuación se presenta el análisis de precios realizado con base en horas-hombre.

Concepto	Costo unitario	Unidad de medición	Cantidad	Costo por concepto
Analista	\$120.00	Hora-hombre	\$88.00	\$10,560.00
Diseñador	\$120.00	Hora-hombre	\$88.00	\$10,560.00
Programadores(2)	\$120.00	Hora-hombre	\$440.00	\$52,800.00
Total				\$73,920.00

Tabla 4.6_4 Análisis de Costos por Hora-Hombre (en Dólares)

Realizando un ajuste en los costos anteriores y basándonos en el tiempo de desarrollo del sistema, tenemos el siguiente cálculo.

Concepto	Costo unitario	Unidad de Medición	Cantidad	Costo por concepto
Tiempo de desarrollo (Análisis, Diseño, Programación e Implementación)	\$21,428.00	Mes	3.5	\$75,000.00
Total				\$75,000.00

Tabla 4.6_5 Análisis de Costos por Tiempo de Desarrollo (en Dólares)

La Institución a la cual se le esta desarrollando el Sistema de Cheques, ya tiene la infraestructura de red necesaria, para la implantación del sistema, así como el hardware y el software necesarios, por lo que no se incluyen estos conceptos en el análisis de costos.

Para empresas que no cuenten con la estructura de red, el hardware (servidor y estaciones de trabajo) y el software (Progress) necesarios para la implementación del sistema de Cheques, estos conceptos se agregarán al costo del sistema.

La operación de compra-venta del sistema puede realizarse de las siguientes formas:

- Venta del sistema por licencia.
- Venta total del sistema.
- El costo de venta por licencia sería de 650 dólares al tipo de cambio actual.
- El costo total del sistema sería de 75 mil dólares al tipo de cambio actual.

La Institución para la cual se desarrolló el sistema de Cheques tiene como políticas el adquirir totalmente los derechos de los programas realizados, es decir requiere que le sea entregada la totalidad del código del sistema, manuales de usuario y técnico.

Por tanto la operación de compra-venta del Sistema de Cheques deberá realizarse en forma total.

CONCLUSIONES

- La tecnología de información permite a las empresas mexicanas modernizarse y estar al mismo nivel tecnológico que sus competidores.
- Las herramientas RAD ("Rapid Application Development") como las utilizadas en este proyecto reducen la duración de las fases de diseño, desarrollo y mantenimiento de las aplicaciones.
- El uso de sistemas y aplicaciones en plataformas de diferentes proveedores es ya una realidad, y este proyecto muestra como el emplear estándares internacionales o de la industria (bases de datos relacionales, sistemas operativos Windows y Unix, TCP/IP, Ethernet, etc.) ampliamente difundidos y que preferentemente sean abiertos facilita la integración de dichos sistemas.
- Para la realización de este trabajo de tesis, pudimos aplicar conocimientos vistos en las materias: Ingeniería de sistemas, Programación Estructurada, Redes de Computadoras, Temas especiales de Computación, así como el conocimiento basado en la experiencia profesional de cada uno de nosotros.
- El ingresar al PAT es un gran proyecto para las personas que ya tienen mucho tiempo de haber terminado la carrera, así como el contar con un asesor que apoya con sus conocimientos y experiencia en dirigir tesis, concluyendo con el examen profesional.
- El trabajar con Progress nos ayudó en nuestra formación profesional ya que la mayoría de los integrantes no lo conocíamos y pudimos darnos cuenta de las bondades de este sistema de base datos relacionales. Aprovechando, lo mejor de Progress se hace cada vez más fácil la programación y la creación de los "Front-End".
- Con el desarrollo del trabajo antes descrito, podemos conocer de las diferentes políticas que existen dentro de cada una de las entidades financieras para el trámite que el usuario debe de realizar para poder contratar una cuenta en dólares, así como del trámite que a su vez debe realizar la misma institución para la asignación de las cuentas de cheques en dólares hacia sus clientes.
- Este diseño e implementación del sistema de cheques nos ha permitido conocer más de la Administración y el "Back-Office" de las Chequeras, así como de la normatividad por parte del Banco de México, Comisión Nacional Bancaria y

CONCLUSIONES

de Valores y Secretaria de Hacienda y Crédito Público. Nos permitió conocer los servicios de Bancos Americanos (Barclays Bank y Chase Bank), apertura de cuentas y el "Payable Through".

- El Front-End de Progress que fue utilizado es un producto abierto que nos permite desarrollar muy fácil y eficientemente aplicaciones robustas por el gran contenido de herramientas para su desarrollo. Este Front-End es un Lenguaje de Cuarta Generación (4GL) y además esta construido para ejecución multiplataforma.
- El Back-End de Progress nos ha permitido conocer mas acerca de una Base de Datos Relacional (RDBMS), con su Diccionario de Datos y Administración de Datos que se encuentra a un mismo nivel de Oracle, Sybase e Informix. La base de datos de Progress Enterprise tiene características que son muy importantes: Recuperación automática de caídas; Registro "Roll-back"; Registro "Roll-Forward"; Respaldo en línea completo y en aumento; Reorganización de tablas e índices en línea; "Two-phase commit" y Recuperación "Point-in-time".
- El haber trabajado con un lenguaje 4GL para el desarrollo de la tesis nos amplio el conocimiento respecto a estos mismos, ya que ofrece grandes ventajas al desarrollador de sistemas, debido a la gran facilidad para poder programar con el mismo. Además de que ofrece ciertas ventajas al ser multiplataforma, es decir que puede correr bajo cualquier sistema operativo y puede conectarse con diferentes manejadores de base de datos.
- Al trabajar con un lenguaje 4GL como cualquier otro lenguaje de su clase, los conocimientos deben de ser más amplios para el programador debido a que para poder programar se involucran conceptos de programación orientada a objetos. Y como todo lenguaje de este tipo, Progress no posee todo lo necesario, ya que una vez que se cuenta con el producto se hace depender demasiado de él, debido a que no se cuenta con los productos para utilizar a manejadores de bases distintos al nativo, por lo que estos se deben adquirir aparte para poder trabajar con ellos, aunque se tenga la facilidad de conectarse a ellos con el mencionado producto.

BIBLIOGRAFÍA

1. "Breve Estudio sobre el cheque"
José Abel Recinos Sandoval
2. "Contrato de Cheque"
Walterio Duque Juárez
3. "Cheque post-datado ante la doctrina y ante la legislación"
Joaquín Gamboa Pascoe
4. "El cheque y el fraude"
Xavier González Ortega
5. "Stop payment orders"
National Association of Bank Auditors and Controllers
www.banxico.org.mx/
www.shcp.org.mx/
6. "Manual para prevenir, detectar y reportar operaciones con recursos de procedencia ilícita"
Banca Serfin, S.A., Institución de Banca Múltiple, Grupo Financiero Serfin
7. "Marco Legal"
Banca Serfin, S.A., Institución de Banca Múltiple, Grupo Financiero Serfin
8. "Principios de Contabilidad"
Instituto Mexicano de Contadores, AC
9. "Contabilidad General"
Máximo Anzures
Editorial TRILLAS. Segunda Edición, 1966
10. "Fundamentos de Contabilidad"
Vartkes Hatzacorsian Hovsepian
Editorial ECAFSA. Segunda Edición, 1999
11. "Principios de Contabilidad"
Alejandro Prieto
Editorial Banca y Comercio, S.A. Décima Sexta Edición, 1981

12. "Introducción a los Sistemas de Bases de Datos"
— Jefferey D. Ullman, Jennifer Widom. Editorial Prentice Hall
13. "Fundamentos y Modelos de Bases de Datos"
Miguel Mário Piattini de Adoración.
14. "Introducción a los Sistemas de Bases de Datos"
C.J.DATE.
15. "Procesamiento de Bases de Datos"
"Fundamentos, Diseño e Instrumentación"
David M. Kroenke.
16. "A SYNTETHIC METHOD FOR NORMALIZING RELATIONAL DATABASES"
Jamal Munshi. Sonoma State University, 1993
munshi.sonoma.edu/working/normalize.html
17. "Database Normalization Basics"
Microsoft Access Knowledge Base. #Q100139
support.microsoft.com/support/kb/articles/Q100/1/39.asp
18. "Database Normalization"
Reid Software Development Site
www.ariasoft.com/normalize.html
19. "DATABASE MANAGEMENT PRINCIPLES AND APPLICATIONS."
"Overview of Normalization".
R.E. Wyllys. The University of Texas at Austin. Graduate School of Library and
Information Science
20. "Sistemas Operativos Modernos"
Andrew S. Tane Baum. Editorial Prentice Hall.
21. "Todo sobre Windows NT 4.0"
B. Kretshmer, C. Schnerder. Editorial Maracambo Boixareu.
22. "El Gran Libro de Windows NT"
Bernd Kretschmer. Marcombo, 1996
23. "Manual de Usuario Windows 98"
Microsoft Corporation.
24. "Redes de Computadoras, Internet e Interredes"
Douglas E. Comer. Editorial Prentice Hall.

25. "Guía de Redes de Alta Velocidad"
Tere Parnell. Editorial Mc Graw Hill,
26. "Clark's Networking Repository of Knowledge"
Heinze Institute of Applied Computer Technology
www.lebarge.com/clark/default.htm
27. "Aprenda la red (www.learnthenet.com) "
Michael Lerner Productions
www.learnthenet.com/spanish/
28. "TCP/IP Tutorial and Technical Overview"
IBM Corporation
www.auggy.mlnet.com/ibm/3376fm.html
29. "COMPUTER NETWORKS"
"Protocols, Standards, and Interfaces"
Uyless Black. Prentice Hall, 1996
30. "Progress 8.2 from Progress Software"
An extract from the Bloor Research report entitled "Database : An Evaluation and Comparison"
Bloor Research
31. "Progress Unfolds" Products, Technology & Business for Solution Selling"
VarBusiness March 15, 1997
32. "Unix and Advanced Operating Enviroments"
International Data Corporation
33. "Powered by Progress"
Dataquest analysis of the Embedded Data Marquet.
www.progress.com
34. "Database Cost of Ownership Study"
Boletín informativo de Aberdeen Group, Inc.
Code 3435 Aberdeen RDBMS Study.
35. Progress Software
[Http://www.progress.com/](http://www.progress.com/)
36. Sybase Software
[Http://www.sybase.com/](http://www.sybase.com/)
37. Oracle Software
[Http://www.oracle.com/](http://www.oracle.com/)

38. "Software Assurance Hotlist"
Reliable Software Technologies
www.rstcorp.com/hotlist/complete.html
39. "Frequently Asked Questions (FAQ) "
comp.software.testing
www.lib.ox.ac.uk/internet/news/faq/archive/software-eng.testing-faq.htm
40. "Software Engineering, a Practitioner's Approach"
Roger S. Pressman. McGraw-Hill International Editions, 1995.
41. "Ingeniería del Software"
"Un enfoque práctico"
Roger S. Pressman. McGraw-Hill International, 1997
42. "Code Coverage Analysis"
Steve Cornett. Bullseye Testing Technology
www.bullseye.com/webCoverage.html
43. "Technology Audit"
Butler Group. August 1999
44. "Disminución de actividad como intermediario del ahorro del sector privado"
Momento Económico, Boletín Electrónico. Sección Finanzas, Banca Comercial.
Instituto de Investigaciones Económicas, UNAM. Octubre de 1999, Volumen 5, Ejemplar 10
www.unam.mx/iie/
45. "Privatización de la Banca en México"
Gilberto Calderón Ortiz. Gestión y Estrategia Edición Internet, 8-Nov-97
www-azc.uam.mx/gestion/num1/doc3.html
46. "Consideran controvertidos los resultados del TLCAN"
Finanzas. El Universal, 9-Nov-1999
47. "Aprueban en EU reforma financiera"
Finanzas. El Universal, 6-Nov-1999.
48. "Obligarán a invertir a mexicanos"
Finanzas. El Universal, 25-Nov-1999
49. "Cayó 20.4 por ciento el financiamiento"
Finanzas. El Universal, 23-Nov-1999

APÉNDICE A
Banco de México

A.1 Operaciones pasivas

A.12. Características de las operaciones en dólares de los EE.UU.A.

Las instituciones en la contratación de estas operaciones a través de sus oficinas en México habrán de sujetarse a los términos y condiciones que a continuación se indican y a las demás disposiciones que resulten aplicables.

A.12.1 Depósitos a la vista en cuenta de cheques pagaderos en la república mexicana.

A.12.11. Cuenta habientes.

Estos depósitos podrán recibirse de:

A.12.11.1 Personas físicas domiciliadas en poblaciones localizadas en una franja de 20 kilómetros, paralela a la línea divisoria internacional norte del país o en los Estados de Baja California y Baja California Sur.

A.12.11.2 Personas morales domiciliadas en cualquier parte del país.

A.12.11.3 Representaciones oficiales de gobiernos extranjeros, organismos internacionales e instituciones análogas, ciudadanos extranjeros que presten sus servicios en tales representaciones, organismos e instituciones, así como de corresponsales extranjeros. En todos los casos, los interesados deberán estar acreditados en México ante la secretaría de estado que corresponda.

A.12.12. Abonos.

Estos depósitos sólo podrán ser constituidos o incrementados mediante: a) trasposos de fondos de depósitos bancarios denominados y pagaderos en moneda extranjera; b) la transmisión de documentos a la vista denominados en moneda extranjera y pagaderos sobre el exterior, o c) la entrega de moneda de curso legal de los EE.UU.A.

A.12.13. Montos.

Las instituciones podrán determinar libremente los montos mínimos a partir de los cuales estén dispuestas a recibir depósitos para la constitución y abono de estas cuentas.

A.12.14. Rendimientos.

Estos depósitos podrán o no devengar intereses. Las instituciones podrán pactar libremente las tasas de interés que, en su caso, devenguen los depósitos, las cuales podrán ser distintas para cada inversionista.

En los depósitos con interés, las instituciones deberán reservarse invariablemente el derecho de revisar y ajustar diariamente la tasa de interés pactada.

Los intereses se calcularán sobre promedios de saldos diarios del período en el cual hayan estado vigentes y se pagarán o capitalizarán libremente, según se establezca en el contrato respectivo.

A.12.15. Documentación.

A.12.15.1 Las instituciones deberán documentar estos depósitos en contratos que como mínimo establezcan lo siguiente:

- a) En caso de que devenguen intereses, la tasa aplicable o los elementos necesarios para calcularla;
- b) La obligación por parte de la institución depositaria de pagar los cheques de acuerdo con lo previsto en el último párrafo del artículo 8vo. de la Ley Monetaria de los Estados Unidos Mexicanos y a la tercera de las Reglas de carácter general a las que deberán sujetarse las instituciones de crédito para recibir depósitos en cuenta de cheques en dólares de los Estados Unidos de América, publicadas en el Diario Oficial de la Federación el 11 de septiembre de 1986, modificadas por Resoluciones publicadas en dicho Diario el 17 de junio de 1991 y el 26 de marzo de 1993, mismas que se acompañan a la presente Circular como Anexo 2;
- c) La prohibición para el depositante de ceder los derechos que para él se deriven del contrato respectivo, y
- d) Que el pago de los cheques referido en A.12.16. se efectúe, a elección del beneficiario, mediante: a) situaciones de fondos en cuentas de depósitos bancarios denominados y pagaderos en moneda extranjera; b) la entrega de documentos a la vista denominados en dólares de los EE.UU.A. y pagaderos sobre el exterior, o c) la entrega de moneda de curso legal de los EE.UU.A: La última de las formas de pago citadas estará en todo momento condicionada a la disponibilidad de billetes y monedas metálicas con curso legal en los Estados Unidos de América, por parte de la oficina en la que el beneficiario pretenda cobrar el cheque de que se trate.

A.12.15.2 Los esqueletos para la expedición de los cheques contendrán, en el anverso, la leyenda siguiente: "Este título se pagará precisamente en dólares de los EE.UU.A.". Esta leyenda deberá tener una dimensión que, no podrá ser inferior a uno por cinco centímetros.

A.12.16. Retiros.

Serán retirables a la vista mediante el libramiento de cheques a cargo de oficinas que las instituciones tengan establecidas en las poblaciones señaladas en A.12.11.1, tratándose de los depósitos que mantengan las personas físicas mencionadas en dicho numeral, y en toda la República tratándose de los depósitos de los cuenta-habientes referidos en A.12.11.2 y A.12.11.3. Al efecto, los esqueletos para la expedición de los cheques provenientes de cuentas abiertas por personas físicas,

a que se refiere A.12.11.1, deberán indicar en el reverso las plazas en las que los cheques podrán ser presentados para su cobro.

A.12.17. Otras disposiciones.

A.12.17.1 Las instituciones podrán determinar libremente las comisiones que, en su caso, aplicarán por el manejo de estos depósitos.

A.12.17.2 Previamente a la constitución de estos depósitos, las instituciones deberán requerir al solicitante de que se trate, la documentación que a su juicio demuestre que cumple con el requisito indicado en A.12.11.1, en A.12.11.2 o A.12.11.3, según corresponda. En todos los casos la institución deberá guardar constancia del cumplimiento de los requisitos señalados.

A.12.17.3 Los depósitos deberán quedar constituidos en dólares de los EE.UU.A., por lo que si la recepción se efectuara en otra divisa, la institución depositaria deberá realizar la conversión correspondiente

A.12.2 Depósitos a plazo pagadero sobre el exterior.

A.12.21. Cuenta habientes.

Estos depósitos sólo podrán recibirse de personas morales domiciliadas en el país

A.12.22. Abonos.

Estos depósitos sólo podrán ser constituidos o incrementados mediante: a) trasposos de fondos de depósitos bancarios denominados y pagaderos en moneda extranjera; b) la transmisión de documentos a la vista denominados en moneda extranjera y pagaderos sobre el exterior, o c) la entrega de moneda de curso legal de los EE.UU.A.

A.12.23. Montos.

Las instituciones podrán determinar libremente los montos mínimos a partir de los cuales estén dispuestas a recibir depósitos para la constitución y abono de estas cuentas

A.12.24. Rendimientos.

Estos depósitos devengarán intereses a la tasa que convengan libremente las instituciones depositarias con los depositantes. Una vez pactada la tasa, se mantendrá fija durante toda la vigencia del depósito, no procediendo revisión alguna de la misma.

Las instituciones podrán pactar libremente con su clientela la periodicidad con la que vayan a pagar los intereses, los cuales serán cubiertos en los términos previstos en A.12.27.

A.12.25. Documentación.
Los depósitos a plazo deberán documentarse en certificados no negociables o constancias de depósito en las que se especifique que no podrán transferirse los derechos correspondientes a los depósitos que documenten.

En los contratos, certificados o constancias en que se documenten los depósitos de que se trata, deberá estipularse la forma de pago indicada en A.12.27.

En las constancias de depósito podrá estipularse la renovación automática de la operación, en caso de no ser retirado el depósito respectivo a su vencimiento.

A.12.26. Plazo.
Al constituirse estos depósitos, las partes pactarán libremente, en cada caso, el plazo de los mismos, no debiendo ser menor a un día y será forzoso para ambas partes.

A.12.27. Retiros.
Los depósitos serán pagaderos a elección del depositante, mediante: a) situación de fondos en cuentas de depósitos bancarios denominados y pagaderos en moneda extranjera, o b) la entrega de documentos a la vista denominados en dólares de los EE.UU.A. y pagaderos sobre el exterior.

A.12.28. Otras disposiciones.
A estos depósitos les será aplicable, en lo conducente, lo señalado en A.11.82.1, A.11.83. y A.11.84.

A.12.29. Los depósitos deberán quedar constituidos en dólares de los EE.UU.A., por lo que si la recepción se efectuara en otra divisa, la institución depositaria deberá realizar la conversión correspondiente.

A.12.3 Disposiciones generales.
A los depósitos bancarios en dólares de los EE.UU.A. les será aplicable, en lo conducente, lo señalado en A.11.81, A.11.82.3, A.11.82.4, A.11.85, A.11.91. y A.11.92.
(Modificado por la Circular-Telefax 22/99)

A.12.4 Bonos bancarios.
Las instituciones podrán emitir bonos bancarios en moneda extranjera de los previstos en el artículo 63 de la Ley de Instituciones de Crédito en términos de lo dispuesto en A.11.3.

A estos títulos les será aplicable, en lo conducente, lo dispuesto en A.11.8 y A.11.9. El pago de los bonos bancarios se efectuará a elección del beneficiario respectivo, mediante: a) situaciones de fondos en cuentas de depósitos bancarios denominados y pagaderos en moneda extranjera, o b) la entrega de documentos a la vista denominados en moneda extranjera y pagaderos en el extranjero.

A.12.5 Obligaciones subordinadas.

Las instituciones podrán emitir obligaciones subordinadas en moneda extranjera de las previstas en el artículo 64 de la Ley de Instituciones de Crédito, previa autorización del Banco de México, en términos de lo dispuesto en A.11.4 las cuales podrán ser colocadas por la institución emisora sin intermediación de casas de bolsa.

A estos títulos les será aplicable, en lo conducente, lo dispuesto en A.11.8 y A.11.9.

1912

...

...

...

LEY PARA REGULAR LAS AGRUPACIONES FINANCIERAS

(Publicada en el Diario Oficial de la Federación el 18 de julio de 1990, reformada y adicionada por los Decretos publicados en los Diarios Oficiales de la Federación del 9 de junio de 1992, 23 de julio y 23 de diciembre de 1993, 15 de febrero de 1995, 30 de abril y 23 de mayo de 1996 y 19 de enero de 1999).

Al margen un sello con el Escudo Nacional, que dice: Estados Unidos Mexicanos.- Presidente de la República.

ERNESTO ZEDILLO PONCE DE LEON, Presidente Constitucional de los Estados Unidos Mexicanos, a sus habitantes, sabed:

Que el H. Congreso de la Unión se ha servido dirigirme el siguiente:

DECRETO:

"El Congreso de los Estados Unidos Mexicanos, decreta":

LEY PARA REGULAR LAS AGRUPACIONES FINANCIERAS

TITULO PRIMERO

DE LAS DISPOSICIONES PRELIMINARES

ARTICULO 1o.- La presente Ley tiene por objeto regular las bases de organización y funcionamiento de los grupos financieros; establecer los términos bajo los cuales habrán de operar, así como la protección de los intereses de quienes celebren operaciones con los integrantes de dichos grupos.

ARTICULO 2o.- Las autoridades financieras, cada una en la esfera de su respectiva competencia, ejercerán sus atribuciones procurando: el desarrollo equilibrado del sistema financiero del país, con una apropiada cobertura regional; una adecuada competencia entre los integrantes de dicho sistema; la prestación de los servicios integrados conforme a sanas prácticas y

usos financieros; el fomento del ahorro interno y su adecuada canalización hacia actividades productivas; así como, en general, que el sistema citado contribuya al sano crecimiento de la economía nacional.

ARTICULO 30.- Las entidades financieras no deberán utilizar denominaciones iguales o semejantes a las de otras entidades financieras; actuar de manera conjunta; ofrecer servicios complementarios ni, en general, ostentarse en forma alguna como integrantes de grupos financieros, salvo cuando dichos grupos se encuentren constituidos y funcionen conforme a las disposiciones de la presente Ley.

Sin perjuicio de lo dispuesto en el párrafo anterior, las entidades financieras y sus filiales podrán utilizar denominaciones iguales o semejantes, actuar de manera conjunta y ofrecer servicios complementarios, cuando así lo prevean las leyes especiales que las rijan y con sujeción a las disposiciones contenidas en dichos ordenamientos.

* Reformado 23-07-93.

ARTICULO 40.- En lo no previsto por la presente Ley, se aplicarán supletoriamente, en el orden siguiente:

- I. La legislación mercantil;
- II. Los usos y prácticas mercantiles;
- III. El Código Civil para el Distrito Federal, y
- IV. Código Fiscal de la Federación, para efectos de las notificaciones y del recurso a que se refiere el artículo 27 de esta Ley.

Cada entidad financiera integrante de los grupos se registrará por lo dispuesto en las respectivas leyes que le sean aplicables.

* Adicionado 09-06-92.

ARTICULO 50.- El Ejecutivo Federal, a través de la Secretaría de Hacienda y Crédito Público, podrá interpretar para efectos administrativos, los preceptos de esta Ley.

TITULO SEGUNDO

DE LA CONSTITUCION E INTEGRACION DE LOS GRUPOS.

ARTICULO 60.- Se requerirá autorización de la Secretaría de Hacienda y Crédito Público para la constitución y funcionamiento de grupos financieros. Estas autorizaciones serán otorgadas o denegadas discrecionalmente por dicha Secretaría, oyendo la opinión del Banco de México y, según corresponda, en virtud de los integrantes del grupo que pretenda constituirse, de las Comisiones Nacionales Bancaria, de Valores y de Seguros y Fianzas.

Por su naturaleza, dichas autorizaciones serán intransmisibles.

ARTICULO 7o.- Los grupos a que se refiere la presente Ley estarán integrados por una sociedad controladora y por algunas de las entidades financieras siguientes: almacenes generales de depósito, arrendadoras financieras, empresas de factoraje financiero, casas de cambio, instituciones de fianzas, instituciones de seguros, sociedades financieras de objeto limitado, casas de bolsa, instituciones de banca múltiple, así como sociedades operadoras de sociedades de inversión y administradoras de fondos para el retiro.

El grupo financiero podrá formarse con cuando menos dos tipos diferentes de las entidades financieras siguientes: instituciones de banca múltiple, casas de bolsa e instituciones de seguros. En los casos en que el grupo no incluya a dos de las mencionadas entidades, deberá contar por lo menos con tres tipos diferentes de entidades financieras de las citadas en el párrafo anterior que no sean sociedades operadoras de sociedades de inversión o administradoras de fondos para el retiro.

La Secretaría de Hacienda y Crédito Público, mediante disposiciones de carácter general, podrá autorizar que otras sociedades puedan formar parte de estos grupos.

* Reformado 23-07-93.

* Reformado 23-12-93.

* Reformado 23-05-96.

ARTICULO 8o.- Las entidades financieras que formen parte de un grupo de los previstos en esta Ley podrán:

I. Actuar de manera conjunta frente al público, ofrecer servicios complementarios y ostentarse como integrantes del grupo de que se trate;

II. Usar denominaciones iguales o semejantes que los identifiquen frente al público como integrantes de un mismo grupo, o bien, conservar la denominación que tenían antes de formar parte de dicho grupo, en todo caso deberán añadirle las palabras "Grupo Financiero" y la denominación del mismo, y

III. Llevar a cabo operaciones de las que le son propias a través de oficinas y sucursales de atención al público de otras entidades financieras integrantes del grupo, de conformidad con las reglas generales que dicte la Secretaría de Hacienda y Crédito Público.

En ningún caso podrán realizarse operaciones propias de las entidades financieras integrantes del grupo a través de las oficinas de la controladora.

ARTICULO 9o.- Las solicitudes de autorización para constituirse y funcionar como grupo deberán presentarse a la Secretaría de Hacienda y Crédito Público, acompañadas de la documentación siguiente:

- I. Proyecto de estatutos de la controladora, que deberá contener los criterios generales a seguir para evitar conflictos de interés entre los participantes del grupo, así como la estipulación por la cual los socios aceptan el procedimiento que, para dar en garantía las acciones emitidas por la controladora prevé el artículo 29 de esa Ley;
- II. Relación de socios que constituirían la controladora y el capital que cada uno de ellos aportaría, así como de los consejeros y funcionarios de los dos primeros niveles que integrarían la administración;
- III. Proyecto de estatutos de las entidades financieras que integrarán el grupo respectivo. Tratándose de entidades ya constituidas, escritura otorgada ante notario público que contenga los estatutos vigentes, así como los proyectos de modificaciones que se efectuarían con motivo de la creación del grupo;
- IV. Proyecto del convenio de responsabilidades a que se refiere el artículo 28 de esta Ley;
- V. Programa y convenios conforme a los cuales la controladora adquiriría las acciones representativas del capital pagado de las entidades financieras de que se trate, para dar cumplimiento a lo dispuesto en el artículo 15 de esta Ley, y;
- VI. La demás documentación que, en su caso solicite la Secretaría de Hacienda y Crédito Público.

En el evento de que la controladora pretendiere adquirir acciones de empresas que prestaren servicios complementarios o auxiliares a la propia controladora o a los demás integrantes del grupo, incluyendo inmobiliarias propietarias de bienes destinados a oficinas de la controladora o de los demás integrantes del grupo, también deberán presentar, según corresponda, los proyectos de estatutos de tales empresas, o los estatutos vigentes con el proyecto de sus modificaciones, así como el programa y convenios para la adquisición de las acciones respectivas.

ARTICULO 10.- La incorporación de una nueva sociedad a un grupo ya constituido, la fusión de dos o más grupos, así como la fusión de dos o más entidades participantes en un mismo grupo, o de una entidad financiera con cualquier sociedad, requerirá autorización de la Secretaría de Hacienda y Crédito Público, oyendo la opinión del Banco de México y, según corresponda, de las Comisiones Nacionales Bancaria y de Valores, de Seguros y Fianzas o del Sistema de Ahorro para el Retiro.

La incorporación o la fusión, según se trate, se llevará a cabo conforme a las bases siguientes:

- I. A la solicitud respectiva deberán adjuntarse: los proyectos de los acuerdos de las asambleas de accionistas de las sociedades que se incorporan o fusionan, así como de las modificaciones que, en su caso, correspondería realizar a los estatutos de las propias sociedades y al convenio de responsabilidades relativos; los estados financieros que presenten la situación de la sociedad a ser incorporada, de la o las controladoras de que se trate, y de los demás integrantes del o de los grupos respectivos; los convenios conforme a los cuales la correspondiente controladora realizaría la adquisición de las acciones que tuviere que efectuar; los programas conforme a los que se llevaría a cabo la incorporación o la fusión; así como la demás documentación que, en su caso, solicite la Secretaría de Hacienda y Crédito Público;

II. (Derogada).

III. La propia Secretaría, al autorizar la incorporación o la fusión, cuidará en todo tiempo la adecuada protección de los intereses de quienes tuviéren celebradas operaciones con las respectivas entidades financieras;

IV. La incorporación o la fusión surtirán efectos a partir de la fecha en que la autorización a que se refiere este artículo, así como los acuerdos de incorporación o de fusión adoptados por las respectivas asambleas de accionistas, se inscriban en el Registro Público de Comercio;

V. Una vez hecha la inscripción anterior, los acuerdos de incorporación o fusión mencionados, se publicarán en el Diario Oficial de la Federación y en dos periódicos de amplia circulación en la plaza en que tengan su domicilio las sociedades, y

VI. Durante los noventa días naturales siguientes a partir de la fecha de publicación, los acreedores de cualquiera de las sociedades, incluso de las demás entidades financieras integrantes del o de los grupos respectivos, podrán oponerse judicialmente, con el único objeto de obtener el pago de sus créditos, sin que esta oposición suspenda la incorporación o la fusión.

* Reformado 23-07-93.

* Reformado 23-05-96.

ARTICULO 11.- La separación de alguno o algunos de los integrantes de un grupo, así como la disolución de este último, deberán ser autorizadas por la Secretaría de Hacienda y Crédito Público, oyendo la opinión del Banco de México y, según corresponda, de las Comisiones Nacionales Bancaria y de Valores, de Seguros y Fianzas o del Sistema de Ahorro para el Retiro.

La separación o disolución mencionadas, surtirán efectos a partir de la fecha en que la autorización a que se refiere este artículo, así como los respectivos acuerdos de las asambleas de accionistas, se inscriban en el Registro Público de Comercio. Tratándose de la separación, además será aplicable lo dispuesto en las fracciones V y VI del artículo anterior. Al surtir efectos la separación, las entidades financieras deberán dejar de ostentarse como integrantes del grupo respectivo.

Cuando el Fondo Bancario de Protección al Ahorro o el Fondo de Apoyo al Mercado de Valores suscriba o adquiera el cincuenta por ciento o más del capital social de cualquier entidad financiera integrante de un grupo, no se observará lo dispuesto en el primero y segundo párrafos del presente artículo, así como en las fracciones V y VI del artículo 10 de esta Ley. La separación del grupo tendrá efectos a partir de dicha suscripción o adquisición, por lo que se tendrá por modificado el convenio único de responsabilidades en este sentido.

La separación de las entidades financieras se llevará a cabo sin perjuicio de que las responsabilidades de la controladora a que se refiere el artículo 28 de esta Ley, subsistan en tanto no queden totalmente cumplidas todas las obligaciones contraídas por dichas entidades con anterioridad a su separación del grupo.

La controladora sólo podrá disolverse una vez cumplidas todas las obligaciones contraídas por cada una de las entidades financieras con anterioridad a la disolución del grupo.

* Adicionado 30-04-96:

* Reformado 23-05-96.

ARTICULO 12.- La Secretaría de Hacienda y Crédito Público, oyendo al Banco de México y, según corresponda, a las Comisiones Nacionales Bancaria y de Valores, de Seguros y Fianzas o del Sistema de Ahorro para el Retiro, así como a la controladora del grupo financiero afectado, podrá revocar la autorización a que se refiere el artículo 6o. de esta Ley, si la controladora o alguno de los demás integrantes del grupo incumplen lo dispuesto en la presente Ley o en las normas que de ella emanen.

Al revocarse la autorización, los integrantes deberán dejar de ostentarse como miembros del grupo respectivo. Las responsabilidades de la controladora a que se refiere el artículo 28 de esta Ley, subsistirán en tanto no queden totalmente cumplidas todas las obligaciones contraídas por cada una de las entidades financieras que formaban el grupo, con anterioridad a la revocación. La controladora se disolverá en los términos de lo dispuesto por el último párrafo del artículo anterior.

* Reformado 23-05-96.

ARTICULO 13.- Las autorizaciones de la Secretaría de Hacienda y Crédito Público a que se refieren los artículos 6o., 10 y 11, así como la revocación a que alude el artículo 12 de esta Ley, deberán publicarse en el Diario Oficial de la Federación.

ARTICULO 14.- La Secretaría de Hacienda y Crédito Público expedirá las reglas generales que regulen los demás términos y condiciones para la constitución y funcionamiento de grupos financieros, de conformidad con lo dispuesto en la presente Ley.

LEY DE INSTITUCIONES DE CREDITO

(Publicada en el Diario Oficial de la Federación el 18 de julio de 1990, reformada y adicionada por los Decretos publicados en los Diarios Oficiales de la Federación del 9 de junio de 1992, 23 de julio y 23 de diciembre de 1993, 22 de julio de 1994, 15 de febrero, 28 de abril y 17 de noviembre de 1995, 30 de abril y 23 de mayo de 1996, 7 de mayo de 1997 y 18 y 19 de enero de 1999).

Al margen un sello con el Escudo Nacional, que dice: Estados Unidos Mexicanos.- Presidencia de la República.

ERNESTO ZEDILLO PONCE DE LEON, Presidente Constitucional de los Estados Unidos Mexicanos, a sus habitantes, sabed:

Que el H. Congreso de la Unión, se ha servido dirigirme el siguiente:

DECRETO:

"El Congreso de los Estados Unidos Mexicanos, decreta":

LEY DE INSTITUCIONES DE CREDITO

TITULO PRIMERO

DE LAS DISPOSICIONES PRELIMINARES

ARTICULO 1o.- La presente Ley tiene por objeto regular el servicio de banca y crédito; la organización y funcionamiento de las instituciones de crédito; las actividades y operaciones que las mismas podrán realizar; su sano y equilibrado desarrollo; la protección de los intereses del público; y los términos en que el Estado ejercerá la rectoría financiera del Sistema Bancario Mexicano.

ARTICULO 2o.- El servicio de banca y crédito sólo podrá prestarse por instituciones de crédito, que podrán ser:

I. Instituciones de banca múltiple, y

II. Instituciones de banca de desarrollo.

Para efectos de lo dispuesto en la presente Ley, se considera servicio de banca y crédito la captación de recursos del público en el mercado nacional para su colocación en el público, mediante actos causantes de pasivo directo o contingente, quedando el intermediario obligado a cubrir el principal y, en su caso, los accesorios financieros de los recursos captados.

No se considerarán operaciones de banca y crédito aquéllas que, en el ejercicio de las actividades que les sean propias, celebren intermediarios financieros distintos a instituciones de crédito que se encuentren debidamente autorizados conforme a los ordenamientos legales aplicables. Dichos intermediarios en ningún caso podrán recibir depósitos irregulares de dinero en cuenta de cheques.

* Reformado 30-04-96.

ARTICULO 3o.- El Sistema Bancario Mexicano estará integrado por el Banco de México, las instituciones de banca múltiple, las instituciones de banca de desarrollo, el Patronato del Ahorro Nacional y los fideicomisos públicos constituidos por el Gobierno Federal para el fomento económico, así como aquéllos que para el desempeño de las funciones que la ley encomienda al Banco de México, con tal carácter se constituyan.

ARTICULO 4o.- El Estado ejercerá la rectoría del Sistema Bancario Mexicano, a fin de que éste oriente fundamentalmente sus actividades a apoyar y promover el desarrollo de las fuerzas productivas del país y el crecimiento de la economía nacional, basado en una política económica soberana, fomentando el ahorro en todos los sectores y regiones de la República y su adecuada canalización a una amplia cobertura regional que propicie la descentralización del propio Sistema, con apego a sanas prácticas y usos bancarios.

Las instituciones de banca de desarrollo atenderán las actividades productivas que el Congreso de la Unión determine como especialidad de cada una de éstas, en las respectivas leyes orgánicas.

ARTICULO 5o.- El Ejecutivo Federal, a través de la Secretaría de Hacienda y Crédito Público, podrá interpretar para efectos administrativos, los preceptos de esta Ley.

ARTICULO 6o.- En lo no previsto por la presente Ley y por la Ley Orgánica del Banco de México, a las instituciones de banca múltiple se les aplicarán en el orden siguiente:

- I. La legislación mercantil;
- II. Los usos y prácticas bancarias y mercantiles;
- III. El Código Civil para el Distrito Federal, y
- IV. El Código Fiscal de la Federación, para efectos de las notificaciones y los recursos a que se refieren los artículos 25 y 110 de esta Ley.

Las instituciones de banca de desarrollo, se regirán por su respectiva ley orgánica y, en su defecto, por lo dispuesto en este artículo.

* Adicionado 09-06-92.

ARTICULO 7o.- La Secretaría de Hacienda y Crédito Público podrá autorizar, el establecimiento en el territorio nacional de oficinas de representación de entidades financieras del exterior. Estas oficinas no podrán realizar en el mercado nacional ninguna actividad de intermediación financiera que requiera de autorización por parte del Gobierno Federal, y por tanto se abstendrán de actuar, directamente o a través de interpósita persona, en operaciones de captación de recursos del público, ya sea por cuenta propia o ajena, y de proporcionar información o hacer gestión o trámite alguno para este tipo de operaciones.

Las actividades que realicen las oficinas de representación se sujetarán a las reglas que expida la Secretaría de Hacienda y Crédito Público, y a las orientaciones que de acuerdo con la política financiera señalen la propia Secretaría y el Banco de México.

La propia Secretaría podrá autorizar el establecimiento en la República de sucursales de bancos extranjeros de primer orden, cuyas operaciones activas y pasivas podrán efectuarse exclusivamente con residentes fuera del país.

El establecimiento de las mencionadas sucursales se sujetará a las reglas de carácter general que dicte la Secretaría de Hacienda y Crédito Público, y las operaciones que a éstas autorice la propia Dependencia se sujetarán a las disposiciones que emita el Banco de México.

Los bancos extranjeros de referencia, sin perjuicio de la obligación de responder ilimitadamente con todos sus bienes, por las operaciones que practiquen en la República, mantendrán afecto a las sucursales citadas el capital mínimo que determine la Secretaría de Hacienda y Crédito Público, tomando en cuenta los usos internacionales relativos a esas operaciones.

La Secretaría de Hacienda y Crédito Público podrá declarar la revocación de las autorizaciones correspondientes, cuando las referidas sucursales y oficinas no se ajusten a las disposiciones a que se refiere este artículo, sin perjuicio de la aplicación de las sanciones establecidas en la presente Ley y en los demás ordenamientos legales.

Al ejercer las atribuciones que le confiere este artículo la Secretaría de Hacienda y Crédito Público escuchará la opinión del Banco de México y de la Comisión Nacional Bancaria.

Las oficinas y sucursales se sujetarán a la inspección y vigilancia de la Comisión Nacional Bancaria, y cubrirán las cuotas que por estos conceptos determine la Secretaría de Hacienda y Crédito Público.

LEY DE LA COMISION NACIONAL BANCARIA Y DE VALORES

(Publicada en el Diario Oficial de la Federación el 28 de abril de 1995, reformada y adicionada por los Decretos publicados en los Diarios Oficiales de la Federación del 23 de enero de 1998 y 18 de enero de 1999).

Al margen un sello con el Escudo Nacional, que dice: Estados Unidos Mexicanos.- Presidencia de la República.

ERNESTO ZEDILLO PONCE DE LEON, Presidente Constitucional de los Estados Unidos Mexicanos, a sus habitantes sabed:

Que el H. Congreso de la Unión, se ha servido dirigirme el siguiente

DECRETO:

"El Congreso de los Estados Unidos Mexicanos, decreta":

LEY DE LA COMISION NACIONAL BANCARIA Y DE VALORES

TITULO PRIMERO

DE LA NATURALEZA, OBJETO Y FACULTADES

CAPITULO I

DE LA NATURALEZA Y OBJETO

ARTICULO 1.- Se crea la Comisión Nacional Bancaria y de Valores como órgano desconcentrado de la Secretaría de Hacienda y Crédito Público, con autonomía técnica y facultades ejecutivas en los términos de esta Ley.

ARTICULO 2.- La Comisión tendrá por objeto supervisar y regular, en el ámbito de su competencia, a las entidades financieras, a fin de procurar su estabilidad y correcto funcionamiento, así como mantener y fomentar el sano y equilibrado desarrollo del sistema financiero en su conjunto, en protección de los intereses del público.

También será su objeto supervisar y regular a las personas físicas y demás personas morales, cuando realicen actividades previstas en las leyes relativas al citado sistema financiero.

ARTICULO 3.- Para los efectos de la presente Ley se entenderá por:

I. Comisión, a la Comisión Nacional Bancaria y de Valores;

II. Junta de Gobierno o Junta, a la Junta de Gobierno de la Comisión;

III. Presidente, al Presidente de la Comisión, y

IV. Entidades del sector financiero o entidades, a las sociedades controladoras de grupos financieros, instituciones de crédito, casas de bolsa, especialistas bursátiles, bolsas de valores, sociedades operadoras de sociedades de inversión, sociedades de inversión, almacenes generales de depósito, uniones de crédito, arrendadoras financieras, empresas de factoraje financiero, sociedades de ahorro y préstamo, casas de cambio, sociedades financieras de objeto limitado, instituciones para el depósito de valores, instituciones calificadoras de valores, sociedades de información crediticia, así como otras instituciones y fideicomisos públicos que realicen actividades financieras y respecto de los cuales la Comisión ejerza facultades de supervisión.

CAPITULO II

DE LAS FACULTADES

ARTICULO 4.- Corresponde a la Comisión:

- I. Realizar la supervisión de las entidades, así como de las personas físicas y demás personas morales cuando realicen actividades previstas en las leyes relativas al sistema financiero;
- II. Emitir en el ámbito de su competencia la regulación prudencial a que se sujetarán las entidades;
- III. Dictar normas de registro de operaciones aplicables a las entidades;
- IV. Fijar reglas para la estimación de los activos y, en su caso, de las obligaciones y responsabilidades de las entidades, en los términos que señalan las leyes;
- V. Expedir normas respecto a la información que deberán proporcionarle periódicamente las entidades;
- VI. Emitir disposiciones de carácter general que establezcan las características y requisitos que deberán cumplir los auditores de las entidades, así como sus dictámenes;
- VII. Establecer los criterios a que se refiere el artículo 2o. de la Ley del Mercado de Valores, así como aquéllos de aplicación general en el sector financiero acerca de los actos y operaciones que se consideren contrarios a los usos mercantiles, bancarios y bursátiles o sanas prácticas de los mercados financieros y dictar las medidas necesarias para que las entidades ajusten sus actividades y operaciones a las leyes que les sean aplicables, a las disposiciones de carácter general que de ellas deriven y a los referidos usos y sanas prácticas;
- VIII. Fungir como órgano de consulta del Gobierno Federal en materia financiera;
- IX. Procurar a través de los procedimientos establecidos en las leyes que regulan al sistema financiero, que las entidades cumplan debida y eficazmente las operaciones y servicios, en los términos y condiciones concertados, con los usuarios de servicios financieros;
- X. (Derogada por el Transitorio Segundo del Decreto que expide la Ley de Protección y Defensa al Usuario de Servicios Financieros, publicado en el Diario Oficial de la Federación el 18-01-99).
- XI. Autorizar la constitución y operación, así como determinar el capital mínimo, de aquellas entidades que señalan las leyes;

* Derogada 18-01-99.

- XII. Autorizar o aprobar los nombramientos de consejeros, directivos, comisarios y apoderados de las entidades, en los términos de las leyes respectivas;
- XIII. Determinar o recomendar que se proceda a la amonestación, suspensión, veto o remoción y, en su caso, inhabilitación de los consejeros, directivos, comisarios, delegados fiduciarios, apoderados, funcionarios y demás personas que puedan obligar a las entidades, de conformidad con lo establecido en las leyes que las rigen;
- XIV. Ordenar la suspensión de operaciones de las entidades de acuerdo a lo dispuesto en esta Ley;
- XV. Intervenir administrativa o gerencialmente a las entidades, con objeto de suspender, normalizar o resolver las operaciones que pongan en peligro su solvencia, estabilidad o liquidez, o aquellas violatorias de las leyes que las regulan o de las disposiciones de carácter general que de ellas deriven, en los términos que establecen las propias leyes;
- XVI. Investigar aquellos actos de personas físicas, así como de personas morales que no siendo entidades del sector financiero, hagan suponer la realización de operaciones violatorias de las leyes que rigen a las citadas entidades, pudiendo al efecto ordenar visitas de inspección a los presuntos responsables;
- XVII. Ordenar la suspensión de operaciones, así como intervenir administrativa o gerencialmente, según se prevea en las leyes, la negociación, empresa o establecimientos de personas físicas o a las personas morales que, sin la autorización correspondiente, realicen actividades que la requieran en términos de las disposiciones que regulan a las entidades del sector financiero, o bien proceder a la clausura de sus oficinas;
- XVIII. Investigar presuntas infracciones en materia de uso indebido de información privilegiada, de conformidad con las leyes que rigen a las entidades;
- XIX. Imponer sanciones administrativas por infracciones a las leyes que regulan las actividades, entidades y personas sujetas a su supervisión, así como a las disposiciones que emanen de ellas;
- XX. Conocer y resolver sobre el recurso de revocación que se interponga en contra de las sanciones aplicadas, así como condonar total o parcialmente las multas impuestas;
- XXI. Intervenir en los procedimientos de liquidación de las entidades en los términos de ley;
- XXII. Determinar los días en que las entidades deberán cerrar sus puertas y suspender sus operaciones;
- XXIII. Elaborar y publicar estadísticas relativas a las entidades y mercados financieros;
- XXIV. Celebrar convenios con organismos nacionales e internacionales con funciones de supervisión y regulación similares a las de la Comisión, así como participar en foros de consulta y organismos de supervisión y regulación financieras a nivel nacional e internacional;
- XXV. Proporcionar la asistencia que le soliciten las instituciones supervisoras y reguladoras de otros países, para lo cual en ejercicio de sus facultades de inspección y vigilancia, podrá recabar respecto de cualquier persona la información y documentación que sea objeto de la solicitud;
- XXVI. Intervenir en la emisión, sorteos y cancelación de títulos o valores de las entidades, en los términos de ley, cuidando que la circulación de los mismos no exceda de los límites legales;
- XXVII. Aplicar a los servidores públicos de las instituciones de banca múltiple, en las que el Gobierno Federal tenga el control por su participación accionaria y de las instituciones de banca de desarrollo las disposiciones, así como las

sanciones previstas en la Ley Federal de Responsabilidades de los Servidores Públicos que correspondan a las contralorías internas, sin perjuicio de las que en términos de la propia Ley, compete aplicar a la Secretaría de Contraloría y Desarrollo Administrativo;

XXVIII. Llevar el Registro Nacional de Valores e Intermediarios y certificar inscripciones que consten en el mismo;

XXIX. Autorizar, suspender o cancelar la inscripción de valores y especialistas bursátiles en el Registro Nacional de Valores e Intermediarios, así como suspender la citada inscripción por lo que hace a las casas de bolsa;

XXX. Supervisar a los emisores de valores inscritos en el Registro Nacional de Valores e Intermediarios, respecto de las obligaciones que les impone la Ley del Mercado de Valores;

XXXI. Dictar las disposiciones de carácter general relativas a la forma y términos en que las sociedades emisoras que dispongan de información privilegiada tendrán la obligación de hacerla de conocimiento del público;

XXXII. Expedir normas que establezcan los requisitos mínimos de divulgación al público que las instituciones calificadoras de valores deberán satisfacer sobre la calidad crediticia de las emisiones que éstas hayan dictaminado y sobre otros aspectos tendientes a mejorar los servicios que las mismas prestan a los usuarios;

XXXIII. Emitir reglas a que deberán sujetarse las casas de bolsa al realizar operaciones con sus accionistas, consejeros, directivos y empleados;

XXXIV. Autorizar y vigilar sistemas de compensación, de información centralizada, calificación de valores y otros mecanismos tendientes a facilitar las operaciones o a perfeccionar el mercado de valores;

XXXV. Ordenar la suspensión de cotizaciones de valores, cuando en su mercado existan condiciones desordenadas o se efectúen operaciones no conformes a sanos usos o prácticas;

XXXVI. Emitir las disposiciones necesarias para el ejercicio de las facultades que esta Ley y demás leyes le otorgan y para el eficaz cumplimiento de las mismas y de las disposiciones que con base en ellas se expidan, y

XXXVII. Las demás facultades que le estén atribuidas por esta Ley, por la Ley Reglamentaria de la Fracción XIII Bis del Apartado B, del Artículo 123 de la Constitución Política de los Estados Unidos Mexicanos y por otras leyes.

ARTICULO 5.- La supervisión que realice la Comisión se sujetará al reglamento que al efecto expida el Ejecutivo Federal y comprenderá el ejercicio de las facultades de inspección, vigilancia, prevención y corrección que le confieren a la Comisión esta Ley, así como otras leyes y disposiciones aplicables.

La supervisión de las entidades financieras tendrá por objeto evaluar los riesgos a que están sujetas, sus sistemas de control y la calidad de su administración, a fin de procurar que las mismas mantengan una adecuada liquidez, sean solventes y estables y, en general, se ajusten a las disposiciones que las rigen y a los usos y sanas prácticas de los mercados financieros. Asimismo, por medio de la supervisión se evaluarán de manera consolidada los riesgos de entidades financieras agrupadas o que tengan vínculos patrimoniales, así como en general el adecuado funcionamiento del sistema financiero.

La inspección se efectuará a través de visitas, verificación de operaciones y auditoría de registros y sistemas, en las instalaciones o equipos automatizados de las entidades financieras, para comprobar el estado en que se encuentran estas últimas.

La vigilancia se realizará por medio del análisis de la información económica y financiera, a fin de medir posibles efectos en las entidades financieras y en el sistema financiero en su conjunto.

La prevención y corrección se llevarán a cabo mediante el establecimiento de programas, de cumplimiento forzoso para las entidades financieras, tendientes a eliminar irregularidades. Asimismo, dichos programas se establecerán cuando las entidades presenten desequilibrios financieros que puedan afectar su liquidez, solvencia o estabilidad, pudiendo en todo caso instrumentarse mediante acuerdo con las propias entidades. El incumplimiento de los programas podrá dar lugar al ejercicio de la facultad contenida en la fracción XV del artículo 4 de esta Ley, sin perjuicio de las sanciones contempladas en el artículo 108 de la Ley de Instituciones de Crédito.

La supervisión que efectúe la Comisión respecto de las personas físicas y demás personas morales, cuando realicen actividades previstas en las leyes relativas al sistema financiero, tendrá por propósito que tales personas observen debidamente las citadas leyes, así como las disposiciones que emanen de ellas.

ARTICULO 6.- Para los efectos de la fracción II del artículo 4, la Comisión, de conformidad con lo que establezcan las leyes relativas al sistema financiero, emitirá normas de carácter prudencial orientadas a preservar la liquidez, solvencia y estabilidad de las entidades financieras.

ARTICULO 7.- La Comisión en uso de la facultad a que se refiere la fracción XIV del artículo 4, podrá ordenar la suspensión temporal de todas o algunas de las operaciones de las entidades financieras cuando infrinjan de manera grave o reiterada la legislación que les resulta aplicable, así como las disposiciones que deriven de ella. Dicha facultad no comprenderá la suspensión de operaciones que de conformidad con las leyes corresponda ordenar al Banco de México.

ARTICULO 8.- La Comisión a efecto de llevar a cabo visitas de inspección en los términos de la fracción XVI del artículo 4, así como de cumplir eficazmente las resoluciones de clausura, intervención administrativa o gerencial que emita en ejercicio de sus facultades, podrá solicitar el auxilio de la fuerza pública.

ARTICULO 9.- El ejercicio de la facultad prevista en la fracción XXV del artículo 4, se sujetará, en su caso, a los términos del acuerdo previamente suscrito al efecto con las entidades supervisoras y reguladoras de otros países y siempre que prevalezca el principio de reciprocidad.

La Comisión podrá solicitar a otras autoridades y dependencias nacionales la información y documentación que obre en su poder, a fin de atender las solicitudes de asistencia correspondientes.

APÉNDICE C
Diccionario de Datos

==== Table: M_promotor =====

Table Name	Dump Name	Table Field Flags	Count	Index Count	Table Label
M_promotor	promotor	7	2	?	

==== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
10	id_area	inte	im	>9	0
20	id_division	inte	im	>9	0
30	id_plaza	inte	im	>9	0
40	id_sucursal	inte	im	>>>9	0
50	id_promotor	inte	im	>>>>9	0
60	nombre	char		m	X(40)
90	origen	char		m	! S

Field Name	Label	Column Label
id_area	?	?
id_division	?	?
id_plaza	?	?
id_sucursal	?	?
id_promotor	?	?
nombre	?	?
origen	?	?

==== index summary =====

Flags	Index Name	Cnt	Field Name
pu	id_promotor	1	+ id_promotor
	ubic_sucurs	4	+ id_area + id_division + id_plaza + id_sucursal

==== field details =====

** Field Name : id_area

Description : Area que comprende divisiones y plazas.
 Help : Teclee el area del promotor.
 Val-Msg : No existe esta area.
 Val-Exp : can-find (first M_ubicacion where M_ubicacion.id_area = input M_promotor.id_area)
**** Field Name : id_division**
 Description : Division comprendida en areas y que comprende plazas.
 Help : Teclee la division del promotor.
 Val-Msg : No existe esta division.
 Val-Exp : can-find (first M_ubicacion where M_ubicacion.id_area = input M_promotor.id_area and M_ubicacion.id_division = input M_promotor.id_division)
**** Field Name : id_plaza**
 Description : Numero de plaza que esta en una area y division.
 Help : Teclee la plaza del promotor.
 Val-Msg : No existe esta plaza.
 Val-Exp : can-find (M_ubicacion where M_ubicacion.id_area = input M_promotor.id_area and M_ubicacion.id_division = input M_promotor.id_division and M_ubicacion.id_plaza = input M_promotor.id_plaza)
**** Field Name : id_sucursal**
 Description : Sucursal a la que pertenece el promotor.
 Help : Teclee la sucursal del promotor.
 Val-Msg : No existe esta sucursal.
 Val-Exp : {PROM_V4.V}
**** Field Name : id_promotor**
 Description : Numero que identifica a cada promotor.
 Help : Teclee el numero de promotor.
 Val-Msg : Debe proporcionarse el numero de promotor
 Val-Exp : input id_promotor > 0
**** Field Name : nombre**
 Description : Nombre del promotor.
 Help : Teclee el nombre del promotor.
 Val-Msg : Debe proporcionarse el nombre.
 Val-Exp : input nombre <> ?
**** Field Name : origen**
 Description : Origen del promotor. dbt o serfin.
 Help : Teclee el origen del promotor. S = serfin, D = DBT
 Val-Msg : Debe teclearse 'S' (serfin) o 'D' (DBT) o "O" Obsa
 Val-Exp : input origen = 'S' or input origen = 'D' or input origen = "O"

=====
 Table: M_sucursal
 =====

Table Name	Dump Name	Table Field Index	Table Label
		Flags Count Count	
M_sucursal	sucursal	.15 1- ?	

=====
 field summary
 =====

Order	Field Name	Data Type	Flags	Format	Initial
10	id_area	inte	im	>9	0
20	id_division	inte	im	>9	0
30	id_plaza	inte	im	>9	0
40	id_sucursal	inte	im	>>>9	0
50	nombre	char	m	x(25)	
60	calle	char	m	x(40)	
70	colonia	char	m	x(40)	

80 ciudad	char	m	x(30)	
90 c_p	inte		99999	0
100 id_pais	char	m	!!	ME
110 tel_1	char		(XXX) XXX-XX-XX	
120 tel_2	char		(XXX) XXX-XX-XX	
125 tel_3	char		(XXX) XXX-XX-XX	
140 fax	char		(XXX) XXX-XX-XX	
150 id_estado	inte	m	>9	9

Field Name	Label	Column Label
id_area	?	?
id_division	?	?
id_plaza	?	?
id_sucursal	?	?
nombre	?	?
calle	?	?
colonia	?	?
ciudad	?	?
c_p	?	?
id_pais	?	?
tel_1	?	?
tel_2	?	?
tel_3	?	?
fax	?	?
id_estado	?	?

=====
index summary
=====

Flags Index Name	Cnt	Field Name
pu ubic_sucurs	4	+ id_area
		+ id_division
		+ id_plaza
		+ id_sucursal

=====
field details
=====

- ** Field Name : id_area
 Description : Area en donde se ubica la sucursal.
 Help : Teclee el area de la sucursal.
 Val-Msg : No existe esta area.
 Val-Exp : {P_V3.V}
- ** Field Name : id_division
 Description : Division en donde se encuentra la sucursal.
 Help : Teclee la division de la sucursal.
 Val-Msg : No existe esta division.
 Val-Exp : {P_V1.V}
- ** Field Name : id_plaza
 Description : Numero de plaza en donde se encuentra la sucursal.
 Help : Teclee la plaza de la sucursal.
 Val-Msg : No existe esta plaza.
 Val-Exp : {P_V2.V}
- ** Field Name : id_sucursal
 Description : Numero de la sucursal.
 Help : Teclee el número de la sucursal.

```

Val-Msg      : Debe proporcionarse la sucursal.
Val-Exp      : input id_sucursal > 0
** Field Name : nombre
Description  : Nombre asignado a la sucursal.
Help         : Teclee el nombre de la sucursal.
Val-Msg      : Debe proporcionarse el nombre.
Val-Exp      : input nombre <> ""
** Field Name : calle
Description  : Calle donde se ubica la sucursal.
Help         : Teclee la calle donde se ubica la sucursal.
Val-Msg      : Debe proporcionarse la calle de la sucursal.
Val-Exp      : input calle <> ?
** Field Name : colonia
Description  : Colonia donde se ubica la sucursal.
Help         : Teclee la colonia donde se ubica la sucursal.
Val-Msg      : Debe proporcionarse la colonia de la sucursal.
Val-Exp      : input colonia <> ?
** Field Name : ciudad
Description  : Ciudad en donde se ubica la sucursal.
Help         : Teclee la ciudad en donde se ubica la sucursal.
Val-Msg      : Debe proporcionarse el nombre de la ciudad de la sucursal.
Val-Exp      : input ciudad <> ?
** Field Name : c_p
Description  : Código postal de la sucursal.
Help         : Teclee el código postal de la sucursal.
** Field Name : id_pais
Description  : Identificación del país.
Help         : Teclee la identificación del país.
Val-Msg      : Este país no existe.
Val-Exp      : can-find (G_pais where G_pais.id_pais = M_sucursal.id_pais)
** Field Name : tel_1
Description  : Primer teléfono de la sucursal.
Help         : Teclee el primer teléfono de la sucursal.
** Field Name : tel_2
Description  : Segundo teléfono de la sucursal.
Help         : Teclee el segundo teléfono de la sucursal.
** Field Name : tel_3
Description  : Tercer teléfono de la sucursal.
Help         : Teclee el tercer teléfono de la sucursal.
** Field Name : fax
Description  : Numero de fax de la sucursal.
Help         : Teclee el numero de fax de la sucursal.
** Field Name : id_estado
Help         : Teclee la clave del estado.
Val-Msg      : No existe el estado.
Val-Exp      : can-find(g_estado of m_sucursal)
    
```

==== Table: M_ubicacion =====

Table Name	Dump Name	Table Field Flags	Index Count	Table Count	Label
M_ubicacion	ubicacio		4	1	?

==== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
10	id_area	inte	im	>9	00
20	id_division	inte	im	>9	00
30	id_plaza	inte	im	>9	00
40	nombre	char	m	x(25)	

Field Name	Label	Column Label
id_area	?	?
id_division	?	?
id_plaza	?	?
nombre	?	?

=====
=====
index summary
=====

Flags	Index Name	Cnt	Field Name
pu	ubicacion	3	+ id_area + id_division + id_plaza

=====
=====
field details
=====

```

** Field Name : id_area
Description   : Area que comprende divisiones y plazas.
Help         : Teclee el area.
Val-Msg      : Debe proporcionarse numero de area.
Val-Exp      : input id_area > 0
** Field Name : id_division
Description   : Division que comprende plazas.
Help         : Teclee el número de división.
Val-Msg      : Debe proporcionarse la división.
Val-Exp      : input id_division > 0
** Field Name : id_plaza
Description   : Numero de plaza que esta en una area y division.
Help         : Teclee el número de plaza.
Val-Msg      : Debe proporcionarse la plaza.
Val-Exp      : input id_plaza <> ?
** Field Name : nombre
Description   : Nombre de la division, plaza o sucursal.
Help         : Teclee el nombre.
Val-Msg      : Debe proporcionarse el nombre.
Val-Exp      : input nombre <> ""
    
```

=====
=====
Table: O_banco
=====

Table Name	Dump Table Name	Field Flags	Index Count	Table Count	Table Label
O_banco	o_banco		8	6	?

=====
=====
field summary
=====

Order	Field Name	Data Type	Flags	Format	Initial
-------	------------	-----------	-------	--------	---------

10 nombre	char	im	x(35)
20 direccion	char[3]	m	x(35)
30 n_cuenta	char	i	x(12)
40 aba	char	i	9(9)
50 chips_uid	char	i	9(6)
60 chips_partic	char	i	9(3)
70 swift	char	i	!(6)NNXXX
80 stat	char		x(1)

Field Name	Label	Column Label
nombre	Nombre del Banco	Banco
direccion	Dirección del Banco	Dirección
n_cuenta	Número de Cuenta	Cuenta
aba	Fedwire ABA number	ABA
chips_uid	CHIPS UID number	CHIPS UID
chips_partic	CHIPS Participant no	CHIPS Part.
swift	Swift Address number	Swift
stat	St	St

==== index summary =====

Flags Index Name	Cnt	Field Name
	1	+ aba
	1	+ chips_partic
	1	+ chips_uid
	1	+ nombre
p n_cuenta	1	+ n_cuenta
	1	+ swift

==== field details =====

- ** Field Name : nombre
 Description : Contiene la razón social del banco.
 Help : Teclee el nombre del Banco.
 Val-Msg : Debe teclearse el nombre del Banco...
 Val-Exp : nombre <> "
- ** Field Name : direccion
 Description : Contiene la dirección del Banco.
 Help : Teclee la dirección completa del Banco.
 Val-Msg : Debe proporcionarse la dirección del Banco...
 Val-Exp : direccion <> "
- ** Field Name : n_cuenta
 Description : Contiene el número de cuenta del Chemical donde el titular es el Banco...
 Help : Teclee el número de cuenta del Banco en el Chemical Bank.
- ** Field Name : aba
 Description : Contiene el número del Fedwire ABA del Banco.
 Help : Teclee el número de Fedwire ABA del Banco.
- ** Field Name : chips_uid
 Description : Contiene el número de CHIPS UID del Banco...
 Help : Teclee el número de CHIPS UID del Banco.

** Field Name : chips_partic
 Description : Contiene el número de CHIPS Participant del Banco.
 Help : Teclee el número de CHIPS Participant del Banco.
 ** Field Name : swift
 Description : Contiene el número del Swift Address del Banco.
 Help : Teclee el número de Swift Address del Banco.
 ** Field Name : stat
 Description : Estatus de la información del banco actual.

==== Table: O_cancela_preformato =====

Table Name	Dump Name	Table Flags	Field Count	Index Count	Table Label
O_cancela_preformato		o_cancel	4	1	?

==== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
10	n_preformato	inte	>>9		0
20	nombre	char	m	x(40)	
30	fec_can	date	m	99/99/9999	?
40	id_usuario	char	m	x(8)	

Field Name	Label	Column Label
n_preformato	?	?
nombre	Nombre	Nombre
fec_can	Fecha Cancelación	Fecha de !Cancelación
id_usuario	?	Usuario Responsable

==== index summary =====

Flags	Index Name	Cnt	Field Name
p	default	0	

==== field details =====

** Field Name : n_preformato
 Description : Numero del preformato
 ** Field Name : nombre
 Help : Teclee el nombre del preformato.
 ** Field Name : fec_can
 Help : Fecha en que se cancelo el preformato.
 ** Field Name : id_usuario
 Description : Identifica al responsable quien cancela el preformato.
 Help : Teclee el responsable.

==== Table: O_estatus_pago =====

Table Name	Dump Name	Table Flags	Field Count	Index Count	Table Label
------------	-----------	-------------	-------------	-------------	-------------

APÉNDICE C. Diccionario de Datos

O_estatus_pago o_estatu 3 2 ?

=====
field summary
=====

Order	Field Name	Data Type	Flags	Format	Initial
9	n_ctrl_pago	inte	im	>>>>9	0
15	estatus	logi	im	SI/NO	no
20	f_valor	date	im	99/99/9999	?

Field Name	Label	Column Label
n_ctrl_pago	?	?
estatus	?	?
f_valor	?	?

=====
index summary
=====

Flags	Index Name	Cnt	Field Name
	Estatus	2	+ estatus + f_valor
pu	N_ctrl_pago	1	+ n_ctrl_pago

=====
field details
=====

** Field Name : n_ctrl_pago
 Description : Numero de control de pago.
 Val-Msg : Debe tener numero de pago
 Val-Exp : n_ctrl_pago <> ? and n_ctrl_pago <> 0.
 ** Field Name : estatus
 Description : indicador para saber si el pago ya afecto la chequera
 ** Field Name : f_valor
 Description : Fecha valor del pago.

=====
Table: O_preformato
=====

Table Name	Dump Name	Lable	Field Flags	Index Count	Table Count	Label
O_preformato	o_prefor	22	2	?		

=====
field summary
=====

Order	Field Name	Data Type	Flags	Format	Initial
5	n_preformato	inte	i	>>>9	0
10	nombre	char	im	x(40)	
20	your_reference	char	m	x(16)	
30	related_reference	char		x(16)	

40 cr_party_type	inte	m	9	0
50 cr_party_id	char	m	x(16)	
60 cr_party_name	char		x(35)	
70 cr_party_addr	char[3]		x(35)	
80 acc_party_type	inte	m	9	0
90 acc_party_id	char		x(34)	
100 acc_party_name	char		x(35)	
110 acc_party_addr	char[3]		x(35)	
120 pay_details	char[2]		x(35)	
130 bank_information	char[2]		x(35)	
140 advice_method	char		!	
150 charges	char		!	O
160 ordering_party_name	char			x(35)
170 ordering_party_addr	char[3]			x(35)
180 ult_benef_acc	char		x(34)	
190 ult_benef_name	char		x(35)	
200 ult_benef_addr	char[3]		x(35)	
210 last_beneficiary	char		!	N

Field Name	Label	Column Label
n_preformato	?	?
nombre	?	?
your_reference	Your Reference	Your Ref
related_reference	Related Reference	Related Ref.
cr_party_type	?	?
cr_party_id	?	?
cr_party_name	?	?
cr_party_addr	?	?
acc_party_type	?	?
acc_party_id	?	?
acc_party_name	?	?
acc_party_addr	?	?
pay_details	Pay Details	Pay Details
bank_information	Bank Information	Bank Info
advice_method	?	?
charges	?	?
ordering_party_name		?
ordering_party_addr		?
ult_benef_acc	?	?
ult_benef_name	?	?
ult_benef_addr	?	?
last_beneficiary	?	?

==== index summary =====

Flags Index Name	Cnt Field Name
pu nombre	1 + nombre
u n_preformato	1 + n_preformato

==== field details =====

- ** Field Name : n_preformato
- Description : Número del preformato
- ** Field Name : nombre

- Description : Escriba el nombre del Preformato.
 Help : Teclee el nombre del preformato
 Val-Msg : Debe de proporcionarse el nombre del preformato
 Val-Exp : input nombre <> "
- ** Field Name : your_reference
 Description : Your Reference
 Help : Your Reference
- ** Field Name : related_reference
 Description : Related Reference
- ** Field Name : cr_party_type
 Description : Env.Pag.1=Name.2=Acco.Num.3=CHIPSunid.4=CHIPSpart.5=FEDWIRE
 ba.6=SWIFT
 Val-Msg : El valor es de tipo Numérico y va de 1 a 6.
 Val-Exp : input cr_party_type >= 1 AND input cr_party_type <= 6
- ** Field Name : cr_party_id
 Description : Número de vía de pago.
 Help : Teclee la vía de pago.
 Val-Msg : Debe proporcionarse como se pagara
 Val-Exp : input cr_party_id <> "
- ** Field Name : cr_party_name
 Description : Credit Party Name
 Help : Teclee el credit party name
 Val-Msg : Debe teclearse el credit party name
- ** Field Name : cr_party_addr
 Description : Credit Party Address
 Help : Teclee el credit party address
- ** Field Name : acc_party_type
 Description : Id.Cuenta 1=Name. 2=Account Núm. 3=CHIPS UID. 6=SWIFT ID.
 7=NONE.
 Val-Msg : Identificador de la cuenta de Partida.
 Val-Exp : (input acc_party_type >= 1 AND input acc_party_type <=3)
 OR (input acc_party_type = 6) OR (input acc_party_type = 7)
- ** Field Name : acc_party_id
 Description : Cuenta Receptora
 Help : Teclee el numero de acc party account
- ** Field Name : acc_party_name
 Description : A nombre de quien va a salir el pago.
 Help : Teclee a favor de quien es el pago.
- ** Field Name : acc_party_addr
 Description : Dirección del beneficiario
 Help : Teclee la acc party address
- ** Field Name : pay_details
 Description : Descripción de Pago.
 Help : Teclee la descripción del pago.
- ** Field Name : Bank_information
 Description : Bank Information
 Help : Teclee la información del banco.
- ** Field Name : advice_method
 Description : Se da la clave del campo M = Mail. P = Phone. T = Telex:
 Help : Métodos de Aviso: M = Mail P = Phone T = Telex (M,P,T).
 Val-Msg : Debe Proporcionarse el Advice Method.
 Val-Exp : (INPUT advice_method = 'M') OR (INPUT advice_method = 'P')
 OR (INPUT advice_method = 'T')
- ** Field Name : charges
 Description : Ind. Trans. B = Beneficiary (Beneficiario), O = Our (Nosotros)
 Help : Cargos: B = Beneficiary (Beneficiario), O = Our (Nosotros).
 Val-Msg : Indicador a quien se le hacen los Cargos de las Transacciones.
 Val-Exp : INPUT charges = 'B' OR INPUT charges = 'O'

**** Field Name** : ordering_party_name
Description : Ordering Party Name
**** Field Name** : ordering_party_addr
Description : Ordering Party addr
**** Field Name** : ult_benef_acc
Description : Cuenta del Ultimo Beneficiario.
Help : Teclee la cuenta del beneficiario final.
**** Field Name** : ult_benef_name
Description : Nombre del Ultimo beneficiario.
Help : Teclee el nombre del beneficiario final.
**** Field Name** : ult_benef_addr
Description : Ultimo beneficiario dirección
Help : Teclee la dirección del beneficiario final
**** Field Name** : last_beneficiary
Description : Id. Banco B = Bank (Banco) N = Non-Bank (No-Banco).
Help : Id. Banco B = Bank (Banco) N = Non-Bank (No-Banco).
Val-Msg : Se utiliza este campo para dar alta Banco (B) o no (N) No-Banco
Val-Exp : INPUT last_beneficiary = 'B' OR INPUT last_beneficiary = 'N'.

==== Table: O_tipo_envio =====

Table Name	Dump Name	Table Flags	Field Count	Index Count	Table Label
O_tipo_envio	o_tipo_e		2	1	?

==== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
10	party_type	inte	im	9	0
20	nombre	char		x(25)	

Field Name	Label	Column Label
party_type	?	?
nombre	?	?

==== index summary =====

Flags	Index Name	Cnt	Field Name
pu	party_type	1	+ party_type

==== field details =====

**** Field Name** : party_type
Description : Se de teclear el tipo de pago (1-7).
Help : Teclear el tipo de pago.
Val-Msg : Se de teclear Unicamente valores de 1 al 7.

Val-Exp : INPUT party_type >= 1 AND INPUT party_type <= 7
 ** Field Name : nombre
 Description : Descripción del tipo de envío.
 Help : Teclee el nombre de la descripción del tipo de envío.
 Val-Msg : El nombre no puede ser bancos
 Val-Exp : INPUT nombre <> ""

=====
 Table: Q_chequera
 =====

Table Name	Dump Name	Table Flags	Field Count	Index Count	Table Label
------------	-----------	-------------	-------------	-------------	-------------

Q_chequera	q_cheque		8	2	?
------------	----------	--	---	---	---

=====
 field summary
 =====

Order	Field Name	Data Type	Flags	Format	Initial
5	id_producto	char	im	!	Q
8	id_cliente	inte	im	>>>>9	0
20	F_Chequera	date	i	99/99/9999	?
30	Cheque_Inicial	inte	i	>>>>>>>	0
40	Cheque_Final	inte		>>>>>>>	0
50	n_cheque_cta	inte		>>>>>9	0
60	n_cheques_chequera	inte		>>>>>9	0
70	nombres_cheq	char[4]		x(30)	

Field Name	Label	Column Label
id_producto	Id_producto	Id_producto
id_cliente	Cliente	Cte
F_Chequera	?	?
Cheque_Inicial	?	?
Cheque_Final	?	?
n_cheque_cta	Chequera por Cuenta	?
n_cheques_chequera	Cheques_Chequera	?
nombres_cheq	Nombre a Cheq.	?

=====
 index summary
 =====

Flags	Index Name	Cnt	Field Name
pu	chequera	4	+ id_producto + id_cliente + F_Chequera + Cheque_Inicial
	cliente	2	+ id_producto + id_cliente

=====
 field details
 =====

** Field Name : id_producto
 Description : Sistema al que pertenece el cliente.
 Help : Sistema al que pertenece el cliente

```

Val-Msg      : Dar de alta (Q = Chequera, U=Univ.)
Val-Exp      : INPUT id_producto = "Q" OR INPUT id_producto = "U"
** Field Name : id_cliente
Description  : Identifica la Cuenta de Cheques.
Help        : Teclee el número de Cuenta.
Val-Msg      : No existe esa cuenta.
Val-Exp      : CAN-FIND (U_cliente WHERE U_cliente.id_producto = input
              Q_chequera.id_producto AND U_cliente.id_cliente = input Q_chequera.id_cliente)
** Field Name : F_Chequera
Description  : Fecha en que se da de alta
Help        : Proporcione la fecha de alta de la chequera
** Field Name : Cheque_Inicial
Description  : Número de cheque inicial.
Help        : Proporcione el número de cheque inicial.
** Field Name : Cheque_Final
Description  : Numero de cheque final
Help        : Proporcione el número de cheque final.
** Field Name : n_cheque_cta
Description  : Chequeras por Cuenta.
Help        : Teclee las Chequeras por Cuenta.
** Field Name : n_cheques_chequera
Description  : Cheques por Chequera.
Help        : Teclee los Cheques por Chequera.
** Field Name : nombres_cheq
Description  : Nombres de las Chequeras.
Help        : Teclee los nombres de las Chequeras.
    
```

Table: Q_cheque_remesa

Table Name	Dump Name	Table Field Name	Index	Table Count	Field Count	Label
Q_cheque_remesa	q_chreme			13	3	?

field summary

Order	Field Name	Data Type	Flags	Format	Initial
5	contador	inte	im	>>9	0
10	n_remesa	inte	im	>>>>9	0
20	id_producto	char	im	!	
30	id_cliente	inte	im	>>>>9	0
40	f_movto	date	m	99/99/9999	?
50	f_afectac	date		99/99/9999	?
60	Referencia	inte		>>>>>	0
70	estatus_imp	logi		Imp-Env/NoImp-N	no
80	estatus_env	logi	i	Aplicado/No-Apl	no
90	Monto	deci-2	m	>>>, >>>, >>>, >>9	0
100	n_cheque	inte		>>>>>9	0
110	foraneo_local	logi		Foraneo/Local	no
120	id_banco	char	im	!(3)	

Field Name	Label	Column Label
contador	Cont	Cont

APÉNDICE C. Diccionario de Datos

n_remesa	No.Remesa	?
id_producto	Id_producto	Id_producto
id_cliente	?	?
f_movto	?	?
f_afectac	Fecha Afect.	?
Referencia	?	?
estatus_imp	Estatus_imp	
estatus_env	Envío de Remesas	
Monto	?	?
n_cheque	No.Cheque	?
foraneo_local	Foraneo-Local	?
id_banco	Id.Banco	?

=====
=====
index summary
=====

Flags Index Name	Cnt	Field Name
pu contador	2	+ n_remesa
		+ contador
id_banco	3	+ id_banco
		+ n_remesa
		+ estatus_env
n_remesa	3	+ n_remesa
		+ id_producto
		+ id_cliente

=====
=====
field details
=====

- ** Field Name : contador
- Description : No. Registros en cada remesa
- ** Field Name : n_remesa
- Description : Numero de Remesa
- Help : Teclee el numero de remesa para el envío de cheques.
- Val-Msg : Debe ser mayor que cero.
- Val-Exp : INPUT n_remesa > 0
- ** Field Name : id_producto
- Description : Sistema al que pertenece el cliente.
- Help : El cliente al que pertenece
- Val-Msg : Dar de alta (U = Univ., Q = Chequera)
- Val-Exp : INPUT id_producto = 'U' OR INPUT id_producto = 'Q'
- ** Field Name : id_cliente
- Description : Identifica a los diferentes clientes de D.B.T.
- Help : Teclee el numero de cliente.
- Val-Msg : Teclee un numero de cliente valido.
- Val-Exp : INPUT id_cliente > 0
- ** Field Name : f_movto
- Description : Fecha Valor del movimiento.
- Help : Proporcione la fecha del movimiento.
- ** Field Name : f_afectac
- Description : Afectación
- Help : Fecha que se esta afectación
- ** Field Name : Referencia
- Description : Referencia del movimiento
- Help : Proporcione la fecha de MMDD
- ** Field Name : Estatus_imp
- Description : Estatus de Impresión de la remesa.

Help : Este valor va ser True Cuando se haya impreso
 ** Field Name : estatus_env
 Description : Estatus de envío de remesa.
 Help : Estatus de envío de la remesa.
 ** Field Name : Monto
 Description : Monto del movimiento.
 Help : Proporcione el monto del movimiento
 Val-Msg : El monto no puede ir en CERO.
 Val-Exp : Monto < 0
 ** Field Name : n_cheque
 Description : El numero de cheque.
 Help : Teclee el numero de Cheques.
 ** Field Name : foraneo_local
 Description : Cheques Foráneos/Locales
 Help : Cheque Foraneo fuera de USA / Cheque Local dentro de USA
 ** Field Name : id_banco
 Description : Banco en donde se deposita.
 Help : Teclee la clave del banco que recibe el deposito.
 Val-Msg : El banco no existe
 Val-Exp : INPUT id_banco = 'BCL' OR INPUT id_banco = 'CHA'

=====
 ===== Table: Q_contador =====
 =====

Table Name	Dump Name	Table Flags	Field Count	Index Count	Table Label
Q_contador	q_contad	2	1	?	

=====
 ===== field summary =====
 =====

Order	Field Name	Data Type	Flags	Format	Initial
10	f_contador	date	i	99/99/9999	?
20	Consecutivo	inte		>>>>>>>	0

Field Name	Label	Column Label
f_contador	?	?
Consecutivo	?	?

=====
 ===== index summary =====
 =====

Flags	Index Name	Cnt	Field Name
pu	fecha	1	+ f_contador

=====
 ===== field details =====
 =====

** Field Name : f_contador
 Description : Fecha del movimiento
 ** Field Name : Consecutivo
 Description : Consecutivo cronológico por fecha.

=====
 ===== Q_corte =====
 =====

Table Name	Dump Name	Table Flags	Field Count	Index Count	Table Label
Q_corte	q_corte	17	3	?	

==== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
5	id_producto	char	i	! . Q	
8	id_cliente	inte	im	>>>>9	0
20	anio_mes_corte	inte	im	>>>>, >>	0
25	Saldo_Inicial	deci-2		>>>, >>>, >>>, >>>	>>>9 0
30	Saldo_Final	deci-2		>>>, >>>, >>>, >>>	>>>9 0
40	Saldo_Promedio	deci-2		>>>, >>>, >>>, >>>	>>>9 0
50	n_cheques_rebotados	inte		>>>>	0
60	n_cheques_emitidos	inte		>>>>9	0
65	Tot_Che_Emitidos	deci-2		>>>, >>>, >>>, >>>	>>>9 0
67	n_Debits	inte		>>>>9	0
70	Tot_Debits	deci-2		>>>, >>>, >>>, >>>	>>>9 0
75	n_Credits	inte		>>>>9	0
80	Tot_Credits	deci-2		>>>, >>>, >>>, >>>	>>>9 0
90	Interes	deci-2		>>>, >>>, >>>, >>>	>>>9 0
100	Tasa_interes	deci-5		>>9.99999	0
110	Interes_Anual	deci-2		>>>, >>>, >>>, >>>	>>>9 0
120	Consecutivo	inte		>>>>>>>	0

Field Name	Label	Column Label
id_producto	Id_producto	Id_producto
id_cliente	Cliente	Cte
anio_mes_corte	?	?
Saldo_Inicial	?	?
Saldo_Final	?	?
Saldo_Promedio	?	?
n_cheques_rebotados	?	?
n_cheques_emitidos	?	?
Tot_Che_Emitidos	?	?
n_Debits	?	?
Tot_Debits	?	?
n_Credits	?	?
Tot_Credits	?	?
Interes	?	?
Tasa_interes	?	?
Interes_Anual	?	?
Consecutivo	?	?

==== index summary =====

Flags	Index Name	Cnt	Field Name
	anio_mes	1	+ anio_mes_corte
	cliente	2	+ id_producto

```

+ id_cliente
pu corte      3      + id_producto
               + id_cliente
               + anio_mes_corte
    
```

===== field details =====

```

** Field Name : id_producto
Description   : Sistema al que pertenece el cliente.
Help          : Sistema al que pertenece el cliente
Val-Msg      : Dar de alta (Q =Chequera, U =Univ.)
Val-Exp      : INPUT id_producto = "Q" OR INPUT id_producto = "U"
** Field Name : id_cliente
Description   : Identifica la Cuenta de Cheques.
Help          : Teclee el número de Cuenta.
Val-Msg      : No existe la Cuenta especificada.
Val-Exp      : CAN-FIND( U_cliente OF Q_corte)
** Field Name : anio_mes_corte
Description   : Año (incluye siglo) y mes del corte.
** Field Name : Saldo_Inicial
Description   : Saldo Inicial al mes del periodo.
** Field Name : Saldo_Final
Description   : Saldo final al cierre de mes.
Help          : Proporcione el saldo final.
** Field Name : Saldo_Promedio
Description   : Saldo Promedio del mes.
Help          : Proporcione el Saldo Promedio del mes.
** Field Name : n_cheques_rebotados
Description   : Número de cheques sobregirados.
** Field Name : n_cheques_emitidos
Description   : Número de cheques emitidos en el mes
** Field Name : Tot_Che_Emitidos
Description   : Total de cheques emitidos
** Field Name : n_Debits
Description   : Numero de otros debits
** Field Name : Tot_Debits
Description   : Total de Debits en el mes.
** Field Name : n_Credits
Description   : Numero de Credits en el periodo.
** Field Name : Tot_Credits
Description   : Total de Credits en el mes.
** Field Name : Interes
Description   : Intereses pagados en el periodo.
** Field Name : Tasa_interes
Description   : Tasa de interés pagada en el periodo.
** Field Name : Interes_Anual
Description   : Interés anual acumulado.
Help          : Interes anual acumulado.
    
```

===== Table: Q_movimiento =====

Table Dump Table Field Index Table

APÉNDICE C. Diccionario de Datos

Name	Name	Flags	Count	Count	Label
Q_movimiento	q_movimi		13	9?	

field summary

Order	Field Name	Data Type	Flags	Format	Initial
5	id_producto	char	i !		Q
8	id_cliente	inte	im	>>>>9	0
10	f_movto	date	im	99/99/9999	TODAY
20	Cronologico	inte	im	>>>>>>>	0
40	id_Movto	char	im	!!!!	
50	Referencia	inte		>>>>>>	0
60	Monto	deci-2	m	>>>>>>>>>>>>	0
70	Afecto_Disponible	logi	i	Si/No	no
80	Contabilizado	logi		Si/No	no
90	id_instrum	char	i	!!	
100	n_inversion	inte	i	>>>>9	0
110	n_reinversion	inte	i	>>>>9	0
120	id_cta_che	inte		>>>>9	0

Field Name	Label	Column Label
id_producto	Id_producto	Id_producto
id_cliente	Cliente	Cte
f_movto	?	?
Cronologico	?	?
id_Movto	?	?
Referencia	?	?
Monto	?	?
Afecto_Disponible	?	?
Contabilizado	?	?
id_instrum	?	?
n_inversion	Inversión	Inversión
n_reinversion		
id_cta_che	Id_cta_che	Id_cta_che

index summary

Flags	Index Name	Cnt	Field Name
	afecto	1	+ Afecto_Disponible
	cliente	2	+ id_producto + id_cliente
u	clie_cro	4	+ id_producto + id_cliente + f_movto + Cronologico
	contab	2	+ f_movto + id_Movto
p	cuenta	3	+ id_producto + id_cliente + f_movto

```

Edo_Global      2      + f_movto
                + Cronologico
id_movto        3      + id_Movto
                + f_movto
                + Cronologico
mvto_inver      7      + id_Movto
                + id_producto
                + id_cliente
                + f_movto
                + id_instrum
                + n_inversion
                + n_reinversion
valida_idmovto  1      + id_Movto
    
```

=====
===== field details =====

```

** Field Name : id_producto
Description   : Sistema al que pertenece el cliente.
Help          : Sistema al que pertenece el cliente
Val-Msg      : Dar de alta (Q = Chequera ,U =Univ.)
Val-Exp      : INPUT id_producto = "Q" OR INPUT id_producto = "U"
** Field Name : id_cliente
Description   : Identifica la Cuenta de Cheques.
Help          : Teclee el número de Cuenta.
Val-Msg      : No existe la Cuenta especificada.
Val-Exp      : CAN-FIND( U_cliente WHERE U_cliente.id_producto = INPUT
              Q_movimiento.id_producto AND U_cliente.id_cliente = INPUT
              Q_movimiento.id_cliente )
** Field Name : f_movto
Description   : Fecha Valor del movimiento.
Help          : Proporcione la fecha del movimiento.
** Field Name : Cronologico
Description   : Orden cronológico de movimientos.
** Field Name : id_Movto
Description   : Identificador del tipo de movimiento
Help          : Proporcione el identificador del tipo de movimiento.
Val-Msg      : No esta registrado el tipo de movimiento.
Val-Exp      : CAN-FIND ( Q_Tpmovto WHERE Q_tpmovto.id_movto = Q_Movimiento.id_movto
              USE-INDEX tpmovto )
** Field Name : Referencia
Description   : Referencia del movimiento
Help          : Proporcione el número de cheque o referencia.
** Field Name : Monto
Description   : Monto del movimiento.
Help          : Proporcione el monto del movimiento
Val-Msg      : El monto no puede ir en CERO.
Val-Exp      : Monto < 0
** Field Name : Afecto_Disponible
Description   : La cuenta ya afecto el saldo disponible?
** Field Name : Contabilizado
Description   : Ese movimiento ya fue contabilizado?
** Field Name : id_instrum
Description   : Identificador del instrumento de inversión.
Help          : Teclee la clave del Instrumento de la Inversión.
Val-Msg      : El tipo de Instrumento no existe.
Val-Exp      : CAN-FIND (T_teso_fid WHERE T_teso_fid.id_instrum = INPUT
              Q_movimiento.id_instrum)
    
```

** Field Name : n_inversion
 Description : Numero de Inversión.
 Help : Teclee el No. De Inversión a Registrar.
 ** Field Name : n_reinversion
 Description : Numero de reinversión.
 Help : Teclee el numero de reinversión para registrar
 ** Field Name : id_cta_che
 Description : Numero de cuenta de cheques.
 Help : Teclee el numero de cuenta de Cheques.
 Val-Msg : Debe proporcionar el numero de cuenta de cheques.
 Val-Exp : INPUT id_cta_che < 0

===== Table: Q_mov_chem =====

Table Dump Table Field Index Table
 Name Name Flags Count Count Label

 Q_mov_chem q_mov_ch 13 6 ?

===== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
5	id_producto	char		i !(1)	
8	id_cliente	inte		im >>>>>9	0
10	f_movto	date	im	99/99/9999	?
20	Cronologico	inte	im	>>>>>>>	0
40	id_Movto	char	im	!!!!	
50	Referencia	inte		>>>>>	0
60	Monto	deci-2	m	>>>, >>>, >>>, >>>9	0
70	Afecto_Disponible	logi		i Si/No	no
80	Contabilizado	logi		Si/No	No
90	id_instrum	char		!!	
100	n_inversion	inte		>>>>9	0
110	n_reinversion	inte		>>>>9	0
120	id_cta_che	inte		>>>>9	0

Field Name Label Column Label

Field Name	Label	Column Label
id_producto	Id_producto	Id_producto
id_cliente	?	?
f_movto	?	?
Cronologico	?	?
id_Movto	?	?
Referencia	?	?
Monto	?	?
Afecto_Disponible	?	?
Contabilizado	Contabilizado	Contabilizado
id_instrum	?	?
n_inversion	Inversión	Inversión
n_reinversion	?	?
id_cta_che	id_cta_che	id_cta_che

===== index summary =====

Flags	Index Name	Cnt	Field Name
	afecto	1	+ Afecto_Disponible
u	clie_pro	4	+ id_producto + id_cliente + f_movto + Cronologico
	contab	2	+ f_movto + id_Movto
p	cuenta	3	+ id_producto + id_cliente + f_movto
	edo_global	2	+ f_movto + Cronologico
	id_movto	3	+ id_Movto + f_movto + Cronologico

=====
field details
=====

**** Field Name** : id_producto
Description : Sistema al que pertenece el cliente.
Help : Sistema al que pertenece el cliente
Val-Msg : Alta Unic. (Q = Chequera, U=Univ.)
Val-Exp : INPUT id_producto = 'Q' OR INPUT id_producto = 'U'

**** Field Name** : id_cliente
Description : Cuenta a afectar.
Help : Proporcione la cuenta a afectar.
Val-Msg : La cuenta especifica no esta registrada.
Val-Exp : CAN-FIND(U_cliente WHERE U_cliente.id_producto = INPUT Q_mov_chem.id_producto AND U_cliente.id_cliente = INPUT Q_mov_chem.id_cliente)

**** Field Name** : f_movto
Description : Fecha Valor del movimiento.
Help : Proporcione la fecha del movimiento.

**** Field Name** : Cronologico
Description : Orden cronológico de movimientos.

**** Field Name** : id_Movto
Description : Identificador del tipo de movimiento
Help : Proporcione el identificador del tipo de movimiento.
Val-Msg : No esta registrado el tipo de movimiento.
Val-Exp : CAN-FIND (Q_tpmovto WHERE Q_tpmovto.id_movto = Q_mov_chem.id_movto USE-INDEX tpmovto)

**** Field Name** : Referencia
Description : Referencia del movimiento
Help : Proporcione el numero de cheque o referencia.

**** Field Name** : Monto
Description : Monto del movimiento.
Help : Proporcione el monto del movimiento
Val-Msg : El monto no puede ir en CERO.
Val-Exp : Monto < 0

**** Field Name** : Afecto_Disponible
Description : La cuenta ya afecto el saldo disponible?

**** Field Name** : Contabilizado
Description : Describe si un movimiento esta contabilizado.

**** Field Name** : id_instrum
Description : Identificador del instrumento de Inversión.

Help : Teclee la clave del instrumento de la Inversión.
 Val-Msg : El tipo de instrumento no existe.
 Val-Exp : CAN-FIND (T_teso_fid WHERE T_teso_fid.id_instrum = INPUT
 Q_movimiento.id_instrum)
 ** Field Name : n_inversion
 Description : No. De Inversión.
 Help : Teclee el No. de Inversión a registrar
 ** Field Name : n_reinversion
 Description : Numero de reinversión.
 Help : Teclee el numero de Reinversión para registrar.
 ** Field Name : id_cta_che
 Description : Proporcione el Numero de Cuenta de Cheques.
 Help : Teclee el No. de cuenta de Cheques.
 Val-Msg : Debe proporcionar el No. de Cuenta de Cheques.
 Val-Exp : INPUT id_cta_che <> "

==== Table: Q_on_chequera =====

Table Name	Dump Name	Table Flags	Field Count	Index Count	Table Label
Q_on_chequera	q_on_che		11	1	?

==== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
10	id_producto	char	i !		C
20	id_cliente	inte	im >>>>9		0
30	f_movto	date	im 99/99/9999		TODAY
40	plazo	inte	m >>>9		0
50	tasa	deci-4	m >>.9999		0
60	monto	deci-2	m >>>, >>>, >>>9.99		0
70	f_venc	date	m 99/99/9999		?
80	f_venc_ch	date	m 99/99/9999		?
90	monto_ch	deci-2	m >>>, >>>, >>>9.99		0
100	plazo_ch	inte	m >>>9		0
110	tasa_ch	deci-4	m >>.9999		0

Field Name	Label	Column Label
id_producto	Id_producto	Id_producto
id_cliente	?	?
f_movto	?	?
plazo	Plazo	Plazo
tasa	Tasa	Tasa
monto	Monto	Monto
f_venc	Fecha Venc.	Fecha Venc.
f_venc_ch	Fecha Venc.	Fecha Venc.
monto_ch	Monto	Monto
plazo_ch	Plazo	Plazo
tasa_ch	Tasa	Tasa

==== index summary =====

Flags	Index Name	Cnt	Field Name
p	cliente	3	+ id_producto + id_cliente + f_movto

==== field details =====

- ** Field Name : id_producto
Description : Sistema al que pertenece.
Help : Sistema al que pertenece.
Val-Msg : Dar de alta (Q = Chequera, U=Univ.)
Val-Exp : INPUT id_producto = "Q" OR INPUT id_producto = "U"
- ** Field Name : id_cliente
Description : Clave del Cliente
Help : Teclee la clave del cliente.
- ** Field Name : f_movto
Description : Fecha Valor del movimiento.
Help : Proporcione la fecha del movimiento.
- ** Field Name : plazo
Help : Teclee del Overnith de Chequera.
- ** Field Name : tasa
Description : Tasa de interés asignada.
Help : Escriba la tasa asignada.
- ** Field Name : monto
Description : Monto a invertir.
Help : Teclee el monto.
- ** Field Name : f_venc
Description : Fecha de vencimiento de la inversión.
Help : Teclee la fecha de vencimiento de la inversión.
- ** Field Name : f_venc_ch
Description : Fecha de vencimiento de la inversión.
Help : Teclee la fecha de vencimiento de la inversión.
- ** Field Name : monto_ch
Description : Monto a invertir.
Help : Teclee el monto.
- ** Field Name : plazo_ch
Help : Teclee del Overnith de Chequera.
- ** Field Name : tasa_ch
Description : Tasa de interes asignada.
Help : Digite la tasa asignada.

==== Table: Q_parametro =====

Table Name	Dump Name	Table Flags	Field Count	Index Count	Table Label
Q_parametro	q_parame		28	1	?

==== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
10	Nombre_Banco	char	m	x(30)	
20	Direccion	char[3]		x(30)	

30 Tronco	char	m	x(10)		
50 Minimo_Fisica	deci-2		>>>>>>>>>>>>9 0		
60 Minimo_Moral	deci-2		>>>>>>>>>>>>9 0		
70 id_cheques	char[5]		x(4)		
80 id_int_Pasivo	char	m	x(4)		
90 id_cargo_cheque	char	m	x(4)		
100 id_cargo_promedio	char	m	x(4)		
110 f_sistema	date		-99/99/9999 ?		
120 id_ret_check	char	m	!!!		
130 id_alt_inv_deb	char	m	x(4)		
140 id_alt_inv_cre	char	m	x(4)		
150 id_pag_inv_deb	char	m	x(4)		
160 id_pag_inv_cre	char	m	x(4)		
170 id_alt_cred_deb	char	m	x(4)		
180 id_alt_cred_cre	char	m	x(4)		
190 id_pag_cred_deb	char	m	x(4)		
200 id_pag_cred_cre	char	m	x(4)		
210 cta_aut_tar	inte	m	>>>>9 0		
220 id_pag_cheq_deb	char	m	x(4)		
230 id_pag_aut_tard	char	m	x(4)		
240 id_pag_aut_tarc	char	m	x(4)		
250 cuenta_global	inte	m	>>>>9 0		
260 limite_cobro	deci-2	m	>>>>>>9.99 9000		
270 id_otro_pago	char	m	x(4)		
280 sdo_min_ctalonario_fisica	deci-2		>>>>>>>>>>>>9 0		
290 sdo_min_ctalonario_moral	deci-2		>>>>>>>>>>>>9 0		

Field Name	Label	Column Label	
Nombre_Banco		?	?
Direccion	?	?	
Tronco	?	?	
Minimo_Fisica	?	?	
Minimo_Moral	?	?	
id_cheques	?	?	
id_int_Pasivo	?	?	
id_cargo_cheque	?	?	
id_cargo_promedio	?	?	
f_sistema	?	?	
id_ret_check	?	?	
id_alt_inv_deb	?	?	
id_alt_inv_cre	?	?	
id_pag_inv_deb	?	?	
id_pag_inv_cre	?	?	
id_alt_cred_deb	?	?	
id_alt_cred_cre	?	?	
id_pag_cred_deb	?	?	
id_pag_cred_cre	?	?	
cta_aut_tar	Cta.Aut.Tar	Cta.Aut.Tar	
id_pag_cheq_deb	?	?	
id_pag_aut_tard	?	?	
id_pag_aut_tarc	?	?	
cuenta_global			
limite_cobro	?	?	
id_otro_pago	?	?	
sdo_min_ctalonario_fisica	SdoMinCTaloFisica	SdoMinCTaloFisica	
sdo_min_ctalonario_moral	SdoMinCTaloMoral	SdoMinCTaloMoral	

=====
 ===== index summary =====
 =====

Flags	Index Name	Cnt	Field Name
-------	------------	-----	------------

p	default	0	
---	---------	---	--

=====
 ===== field details =====
 =====

**** Field Name** : Nombre_Banco
Description : Nombre del Banco. (DBT)
Help : Proporcione el nombre del banco.

**** Field Name** : Direccion
Description : Dirección del banco
Help : Proporcione la dirección del banco.

**** Field Name** : Tronco
Description : Tronco común de todas las cuentas.
Help : Proporcione el tronco común a todas las cuentas.

**** Field Name** : Minimo_Fisica
Description : Monto mínimo para personas físicas.
Help : Proporcione el monto mínimo en Cuenta personas físicas.

**** Field Name** : Minimo_Moral
Description : Monto mínimo en cuenta para personas morales
Help : Proporcione el monto mínimo para personas morales.

**** Field Name** : id_cheques
Description : Claves de cheques emitidos.
Help : Proporcione las claves para identificar cheques emitidos.

**** Field Name** : id_int_Pasivo
Description : Clave de tipo de movto. Para intereses pasivos.
Help : Proporcione la clave para identificar intereses (Pasivos)
Val-Msg : No existe ese tipo de movimiento.
Val-Exp : CAN-FIND(Q_tpmovto WHERE Q_tpmovto.id_movto = INPUT
 Q_parametro.id_int_pasivo)

**** Field Name** : id_cargo_cheque
Description : Clave de tipo de movto. Para cheques girados.
Help : Proporcione el id. Para cargo por cheques girados.
Val-Msg : No existe ese tipo de movimiento.
Val-Exp : CAN-FIND(Q_tpmovto WHERE Q_tpmovto.id_movto = INPUT
 Q_parametro.id_cargo_cheque)

**** Field Name** : id_cargo_promedio
Description : Clave de tipo de movto para cargo por bajo promedio.
Help : Proporcione la clave para cargos por bajo promedio.
Val-Msg : No esta registrado ese tipo de movimiento.
Val-Exp : CAN-FIND(Q_tpmovto WHERE Q_tpmovto.id_movto = INPUT
 Q_parametro.id_cargo_promedio)

**** Field Name** : f_sistema
Description : Fecha del sistema.

**** Field Name** : id_ret_check
Description : Identificador de cargo por cheques rebotados.
Help : Proporcione el id. del cargo por cheques regresados.
Val-Msg : No existe ese tipo de movimiento.
Val-Exp : CAN-FIND (Q_tpmovto WHERE Q_tpmovto.id_movto = INPUT
 Q_parametro.id_ret_check)

**** Field Name** : id_alt_inv_deb
Description : Clave de tipo de Movto. para alta Inv. debit.
Help : Proporcione la Clave para identificar el alta de Inversión Deb.
Val-Msg : No existe ese tipo de Movimiento.

Val-Exp	: CAN-FIND (Q_tpmovto WHERE Q_tpmovto.id_movto = INPUT Q_parametro.id_alt_inv_deb)
** Field Name	: id_alt_inv_cre
Description	: Clave de tipo de Movto. para alta Inv. Credit.
Help	: Proporcione la Clave para identificar el alta de Inversión Cre.
Val-Msg	: No existe ese tipo de Movimiento.
Val-Exp	: CAN-FIND (Q_tpmovto WHERE Q_tpmovto.id_movto = INPUT Q_parametro.id_alt_inv_cre)
** Field Name	: id_pag_inv_deb
Description	: Clave de tipo de Movto. para alta Inv. Debit.
Help	: Proporcione la Clave para identificar el Pago de Inversión Deb.
Val-Msg	: No existe ese tipo de Movimiento.
Val-Exp	: CAN-FIND (Q_tpmovto WHERE Q_tpmovto.id_movto = INPUT Q_parametro.id_pag_inv_deb)
** Field Name	: id_pag_inv_cre
Description	: Clave de tipo de Movto. para Pago Inv. Credit.
Help	: Proporcione la Clave para identificar el Pago de Inversión Cred
Val-Msg	: No existe ese tipo de Movimiento.
Val-Exp	: CAN-FIND (Q_tpmovto WHERE Q_tpmovto.id_movto = INPUT Q_parametro.id_pag_inv_cre)
** Field Name	: id_alt_cred_deb
Description	: Clave de tipo de Movto. Para alta Cred. Debit.
Help	: Proporcione la Clave para identificar el alta de Crédito Debit.
Val-Msg	: No existe ese tipo de Movimiento.
Val-Exp	: CAN-FIND (Q_tpmovto WHERE Q_tpmovto.id_movto = INPUT Q_parametro.id_alt_cred_deb)
** Field Name	: id_alt_cred_cre
Description	: Clave de tipo de Movto. Para alta Crédito Credit.
Help	: Proporcione la Clave para identificar el alta de Crédito Credit
Val-Msg	: No existe ese tipo de Movimiento.
Val-Exp	: CAN-FIND (Q_tpmovto WHERE Q_tpmovto.id_movto = INPUT Q_parametro.id_alt_cred_cre)
** Field Name	: id_pag_cred_deb
Description	: Clave de tipo de Movto. Para Pago Crédito Debit.
Help	: Proporcione la Clave para identificar el Pago de Crédito Debit.
Val-Msg	: No existe ese tipo de Movimiento.
Val-Exp	: CAN-FIND (Q_tpmovto WHERE Q_tpmovto.id_movto = INPUT Q_parametro.id_pag_cred_deb)
** Field Name	: id_pag_cred_cre
Description	: Clave de tipo de Movto. Para alta Crédito Credit.
Help	: Proporcione la Clave para identificar el Pago de Crédito Credit
Val-Msg	: No existe ese tipo de Movimiento.
Val-Exp	: CAN-FIND (Q_tpmovto WHERE Q_tpmovto.id_movto = INPUT Q_parametro.id_pag_cred_cre)
** Field Name	: cta_aut_tar
Description	: Cuenta de tarjeta de crédito.
Help	: Teclee la cuenta de tarjeta de Crédito.
Val-Msg	: La cuenta debe ser mayor que cero.
Val-Exp	: INPUT cta_aut_tar > 0
** Field Name	: id_pag_cheq_deb
Description	: Clave de tipo de Movto. Para Pago Cheq. Debit.
Help	: Proporcione la Clave para identificar el Pago de Cheq. Debit.
Val-Msg	: No existe ese tipo de Movimiento.
Val-Exp	: CAN-FIND (Q_tpmovto WHERE Q_tpmovto.id_movto = INPUT Q_parametro.id_pag_cheq_deb)
** Field Name	: id_pag_aut_tard
Description	: Clave de tipo de Movto. Para Pago Automático de Tarj. Crédito Debit.
Help	: Proporcione la Cve para ident. El Pago de Aut. Tarjeta Debit.

Val-Msg : No existe ese tipo de Movimiento.
 Val-Exp : CAN-FIND (Q_tpmovto WHERE Q_tpmovto.id_movto = INPUT
 Q_parametro.id_pag_aut_tard)

** Field Name : id_pag_aut_tarc
 Description : Clave de tipo de Movto. Para Pago Automático de Tarjeta de Cred. Credi
 Help : Proporcione la Clave para identificar el Pago de Aut. TarjCred
 Val-Msg : No existe ese tipo de Movimiento.
 Val-Exp : CAN-FIND (Q_tpmovto WHERE Q_tpmovto.id_movto = INPUT
 Q_parametro.id_pag_aut_tarc)

** Field Name : cuenta_global
 Description : Cuenta Global.
 Help : Proporcione la Cuenta Global.
 Val-Msg : La cuenta especificada no esta registrada.
 Val-Exp : CAN-FIND (U_cliente WHERE U_cliente.id_producto = "Q" AND
 U_cliente.id_cliente = INPUT cuenta_global)

** Field Name : limite_cobro
 Help : Teclee el importe máximo a cobrar en cheques por día
 Val-Msg : Debe proporcionar el limite de cobro por cheques
 Val-Exp : limite_cobro > 0

** Field Name : id_otro_pago
 Help : Teclee la clave del movimiento de otros pagos
 Val-Msg : No existe ese tipo de movimiento
 Val-Exp : can-find (q_tpmovto where q_tpmovto.id_movto = input
 q_parametro.id_otro_pago)

** Field Name : sdo_min_ctalonario_fisica
 Description : Proporcione el monto mínimo en Cta. Personas Físicas C/Talonario.
 Help : Monto mínimo para personas físicas con talonario.

** Field Name : sdo_min_ctalonario_moral
 Description : Proporcione el monto mínimo en Cta. Personas Morales C/Talonario.
 Help : Monto Mínimo para personal Moral con Talonario.

==== Table: Q_saldo_chem =====

Table Name	Dump Name	Table Flags	Field Count	Index Count	Table Label
Q_saldo_chem	q_saldo_		7	1	?

==== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
10	saldo_inic	deci-2	->, >>>, >>>, >>>	>>>9.99	0
20	credits	deci-2	->>>, >>>, >>>	>>>9.99	0
30	debits	deci-2	->>>, >>>, >>>	>>>9.99	0
40	saldo_final	deci-2	>>>, >>>, >>>	>>>9.99-	0
50	f_mov	date	i	99/99/9999	?
60	stat_conf	logi		Si/No	no
70	saldo_tras	deci-2	->, >>>, >>>, >>>	>>>9.99	0

Field Name	Label	Column Label
saldo_inic	Saldo Inic.	Saldo Inic.
credits	Credits	Credits
debits	Debits	Debits
saldo_final	Saldo Final	Saldo!Final
f_mov	Fecha Mov.	Fecha Mov.

stat_conf ? ?
 saldo_tras Saldo Traspaso Saldo Traspaso,

=====
 index summary
 =====

Flags Index Name Cnt Field Name

pu f_mov 1 + f_mov

=====
 field details
 =====

- ** Field Name: saldo_inic
 Description: Saldo Inicial Cuenta Chemical.
 Help: Teclee el saldo Inicial de la Cuenta del Chemical.
- ** Field Name: credits
 Description: Credits a la Cuenta del Chemical.
 Help: Credits a la Cuenta del Chemical.
- ** Field Name: debits
 Description: Debits de Cuenta del Chemical.
 Help: Debits de Cuenta del Chemical.
- ** Field Name: saldo_final
 Description: Saldo final de la Cuenta del Chemical.
 Help: Saldo final de la Cuenta del Chemical.
- ** Field Name: f_mov
 Description: Fecha de Movimiento de la Cuenta del Chemical
 Help: Fecha de Movimiento
- ** Field Name: stat_conf
 Description: Estatus de Cuenta Chemical.
 Help: Estatus de Confirmación de Cuenta Chemical.
- ** Field Name: saldo_tras
 Description: Saldo a Traspasar:
 Help: Teclee el saldo a Traspasar a la Cuenta del Chemical.

=====
 Table: Q_stoppay
 =====

Table Name	Dump Name	Table Field Name	Index	Table Count	Field Count	Label
------------	-----------	------------------	-------	-------------	-------------	-------

Q_stoppay	q_stoppa		6	1	?	
-----------	----------	--	---	---	---	--

=====
 field summary
 =====

Order	Field Name	Data Type	Flags	Format	Initial
5	id_producto	char	im	!	Q
8	id_cliente	inte	im	>>>>9	0
20	F_Stop	date	m	99/99/9999	?
30	Cheque_Inicial	inte		>>>>>>>	0
40	Cheque_Final	inte		>>>>>>>	0
50	hora_aviso	inte	m	99999	0

Field Name	Label	Column Label
------------	-------	--------------

id_producto	Id_producto	Id_producto
-------------	-------------	-------------

id_cliente	Cliente	Cte
F_Stop	?	?
Cheque_Inicial	?	?
Cheque_Final	?	?
hora_aviso	?	?

=====
index summary
=====

Flags	Index Name	Cnt	Field Name
p	cliente	2	+ id_producto + id_cliente

=====
field details
=====

- ** Field Name: id_producto
Description: Sistema al que pertenece el cliente.
Help: Sistema al que pertenece el cliente
Val-Msg: Dar de alta (Q = Chequera, U=Univ.)
Val-Exp: INPUT id_producto = "Q" OR INPUT id_producto = "U"
- ** Field Name: id_cliente
Description: Identifica la Cuenta de Cheques.
Help: Teclee el número de Cuenta.
Val-Msg: No existe la Cuenta especificada.
Val-Exp: CAN-FIND(U_cliente OF Q_stoppay)
- ** Field Name: F_Stop
Description: Fecha en que se da de alta el registro.
Help: Proporcione la fecha de Congelación.
- ** Field Name: Cheque_Inicial
Description: Número de cheque inicial.
Help: Proporcione el número de cheque inicial.
- ** Field Name: Cheque_Final
Description: Numero de cheque final
Help: Proporcione el número de cheque final.
- ** Field Name: hora_aviso
Description: Registro de Cheque Congelado.
Help: Hora de Registro del cheque Congelado

=====
Table: Q_tasa
=====

Table Name	Dump Name	Table Field Flags	Index Count	Table Count	Label
Q_Tasa	q_tasa		3	1?	

=====
field summary
=====

Order	Field Name	Data Type	Flags	Format	Initial
5	periodo	inte	i	>>>>, >>	0
10	Tasa_Fisica	deci-5		>>9.99999	0
20	Tasa_Moral	deci-5		>>9.99999	0

Field Name	Label	Column Label

APÉNDICE C. Diccionario de Datos

```
-----
periodo          ?          ?
Tasa_Fisica     ?          ?
Tasa_Moral      ?          ?
-----
```

=====
===== index summary =====

```
-----
Flags Index Name      Cnt Field Name
-----
```

```
pu tasa              1 + periodo
-----
```

=====
===== field details =====

```
** Field Name: periodo
Description: Periodo que cubren las tasas.
** Field Name: Tasa_Fisica
Description: Tasa de interés pagado a Personas Físicas.
** Field Name: Tasa_Moral
Description: Tasa de interés pagado a personas Morales.
-----
```

=====
===== Table: Q_tasa_rango =====

```
Table      Dump  Table Field Index Table
Name       Name  Flags Count Count Label
-----
```

```
Q_tasa_rango  q_tasa_p  5  1 ?
-----
```

=====
===== field summary =====

```
Order Field Name      Data Type  Flags Format      Initial
-----
```

```
10 Periodo            inte      im 9999.99      0
20 Minimo              deci-2   im ->>>, >>>, >>>9.99 0
30 Maximo              deci-2   m  ->>>, >>>, >>>9.9 0
40 tasa                deci-5   m  >>9.99999  0
50 tipo_persona        inte      im >9          4
-----
```

```
Field Name      Label      Column Label
-----
```

```
Periodo          Periodo    -Periodo
Minimo           Mínimo    Mínimo
Maximo           Máximo    Máximo
tasa             Tasa      Tasa
tipo_persona     Tipo Persona  Tipo!Persona
-----
```

=====
===== index summary =====

```
-----
Flags Index Name      Cnt Field Name
-----
```

```
pu tasa              3 + Periodo
                    + tipo_persona
                    + Minimo
-----
```

=====
===== field details =====

- ** Field Name: Periodo
Description: Periodo que cubren las tasas.
Help: Teclee el periodo de las tasas.
Val-Msg: Debe proporcionar el periodo de la tasa.
Val-Exp: Periodo > 0
- ** Field Name: Minimo
Help: Teclee el importe minimo
- ** Field Name: Maximo
Help: Teclee el máximo
Val-Msg: El maximo debe de ser mayor que el minimo
Val-Exp: Maximo > Minimo
- ** Field Name: tasa
Help: Escriba la tasa asignada a rango
- ** Field Name: tipo_persona
Help: Teclee el tipo de persona
Val-Msg: El tipo de persona no esta catalogado
Val-Exp: CAN-FIND(g_tipo_persona OF q_tasa_rango)

=====
Table: Q_tpmovto
=====

Table Name	Dump Name	Table Field Flags	Index Count	Table Count	Label
Q_tpmovto	q_tpmovt	14	1	?	

=====
field summary
=====

Order	Field Name	Data Type	Flags	Format	Initial
10	id_Movto	char	im	!!!!	
20	Descrip_Larga	char	m	x(25)	
30	Descrip_Corta	char	m	x(15)	
40	Naturaleza	char	m	!	
50	Afec_Saldo_Total	logi	m	Si/No	no
60	Afec_Saldo_Dis	logi	m	Si/No	no
70	No_Dias_Dis	inte		>>9	0
80	Afec_Cta_Global	logi	m	Si/No	no
90	Es_Ingreso	logi	m	Si/No	no
100	Es_Gasto	logi	m	Si/No	no
110	Generado	char	m	!!	
120	Monto_Fijo	deci-2		>>>, >>>, >>>, >>9	0
130	Cuenta_Cargar	char		9999-99-XX-99	
140	Cuenta_Abonar	char		9999-99-XX-99	

Field Name	Label	Column Label
id_Movto	?	?
Descrip_Larga	?	?
Descrip_Corta	?	?
Naturaleza	?	?
Afec_Saldo_Total	?	?
Afec_Saldo_Dis	?	?
No_Dias_Dis	?	?
Afec_Cta_Global	?	?
Es_Ingreso	?	?
Es_Gasto	?	?

Generado	?	?	
Monto_Fijo	?	?	
Cuenta_Cargar	Cuenta Cargar	Cuenta Abonar	
Cuenta_Abonar	Cuenta Abonar	Cuenta Abonar	

=====
index summary
=====

Flags Index Name	Cnt	Field Name
pa tpmovto	1	id_Movto

=====
field details
=====

- ** Field Name: id_Movto
Description: Identificador del tipo de movimiento
Help: Proporcione el identificador del tipo de movimiento.
Val-Msg: El id. del movimiento no puede ir en blancos
Val-Exp: id_movto <> ""
- ** Field Name: Descrip_Larga
Description: Descripción usada para reportes con suficiente espacio
Help: Proporcione Una descripción completa del tipo de movimiento.
- ** Field Name: Descrip_Corta
Description: Descripción breve para pantallas con escaso espacio.
Help: Proporcione una breve descripción del tipo de movimiento
- ** Field Name: Naturaleza
Description: Naturaleza del movimiento
Help: Proporcione un "C" si es un CREDIT o una "D" para DEBIT
Val-Msg: La naturaleza debe ser "C"redit o "D"ebit
Val-Exp: Naturaleza = "C" OR Naturaleza = "D"
- ** Field Name: Afec_Saldo_Total
Description: El tipo de movimiento afecta el saldo total del cliente ?
Help: Teclee S si se afecta el saldo TOTAL del Cliente
- ** Field Name: Afec_Saldo_Dis
Description: El tipo de movto. afecta inmediatamente el saldo Disponible
Help: Teclee "S" si se afecta inmediatamente el Saldo Disponible
- ** Field Name: No_Dias_Dis
Description: Número de días que tarda el movto. en afectar el saldo disp.
Help: Número de días para afectar saldo disponible (En caso de No afe
- ** Field Name: Afec_Cta_Global
Description: El tipo de movimiento afecta la cuenta global.
Help: Teclee "S" si el tipo de movto. afecta la cuenta Global.
- ** Field Name: Es_Ingreso
Description: El tipo es un ingreso. (para reporte de utilidades)
Help: Proporcione una "S" si el Movimiento es una ingreso.
- ** Field Name: Es_Gasto
Description: El tipo de movto. es un gasto ?.
Help: Proporcione una "S" si el tipo de movto. es un gasto.
- ** Field Name: Generado
Description: Módulo en el que es generado este tipo de movimiento
Help: Teclee "EC" para Captura del FAX, "OM" Otros Movtos, "AU" Automáticos.
Val-Msg: Valores válidos "EC", "OM" y "AU"
Val-Exp: Generado = "EC" OR Generado = "OM" OR Generado = "AU"
- ** Field Name: Monto_Fijo
Description: Monto fijo por ese tipo de movto o 0 para ninguno.

Help: Monto del tipo de movimiento o 0 para dejar monto abierto

** Field Name: Cuenta_Cargar

Description: Cuenta donde se carga ese tipo de movimiento (contabilidad)

** Field Name: Cuenta_Abonar

Description: Cuenta donde se abona el tipo de movimiento (contabilidad)

===== Table: S_area =====

Table Name	Dump Name	Table Field Flags	Index Count	Table Count	Label
S_area	s_area	2	1	?	

===== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
10	id_area	char	i	x(4)	
20	nombre_area	char		x(20)	

Field Name	Label	Column	Label
id_area	?	?	
nombre_area	Nombre		Nombre

===== index summary =====

Flags	Index Name	Cnt	Field Name
pu	id_area	1	id_area

===== field details =====

** Field Name: id_area

Description: Campo que almacena el area donde se localiza al usuario

Help: Teclee el area donde se localiza al usuario

Val-Msg: El area debe ser de 4 caracteres como máximo

Val-Exp: id_area <> ""

** Field Name: nombre_area

Description: Nombre del area donde se localiza al usuario

Help: Teclee el nombre del area donde se localiza al usuario

Val-Msg: El nombre del area debe ser distinto a cero

Val-Exp: nombre_area <> ""

===== Table: S_hist_passw =====

Table Name	Dump Name	Table Field Flags	Index Count	Table Count	Label
S_hist_passw	s_hist_p	3	1	?	

=====
field summary
=====

Order	Field Name	Data Type	Flags	Format	Initial
10	f_cambio	date		99/99/9999	TODAY
20	Id_usuario	char		x(8)	
30	password	char		x(8)	

Field Name	Label	Column Label
f_cambio	?	?
Id_usuario	?	?
password	?	?

=====
index summary
=====

Flags	Index Name	Cnt	Field Name
p	default	0	

=====
field details
=====

** Field Name: f_cambio

Description: Fecha de cambio del password. Se asigna por programa

** Field Name: Id_usuario

Description: Identificación del usuario

Help: Teclee la identificación del usuario

Val-Msg: Ese usuario no existe

Val-Exp: INPUT id_usuario <> " "

** Field Name: password

Description: Password que se cambio en la fecha indicada.

Help: Teclee el password para acceder el programa.

=====
Table: S_nivel
=====

Table Name	Dump Name	Table Field	Index	Table
		Flags	Count	Count Label
S_nivel	s_nivel	4	1?	

=====
field summary
=====

Order	Field Name	Data Type	Flags	Format	Initial
10	id_depto	inte	m	>9	0
20	id_nivel	inte	im	>9	0
30	nombre	char	m	X(15)	
40	id_area	char	im	x(8)	

Field Name	Label	Column Label
id_depto	?	?
id_nivel	?	?
nombre	?	?
id_area	?	?

=====
index summary
=====

Flags Index Name	Cnt Field Name
pu S_llave	2 + id_area + id_nivel

=====
field details
=====

- ** Field Name: id_depto
Description: Area interna de dbt.
Help: Teclee el departamento de DBT a la que pertenece el nivel.
Val-Msg: No existe este departamento.
Val-Exp: can-find (S_depto of S_nivel)
- ** Field Name: id_nivel
Description: Nivel dentro de la jerarquia de un area interna de DBT.
Help: Teclee el numero de nivel.
Val-Msg: Debe proporcionarse el numero de nivel.
Val-Exp: input id_nivel <>?
- ** Field Name: nombre
Description: Nombre del nivel.
Help: Teclee el nombre del nivel.
Val-Msg: Debe proporcionarse el nombre del nivel.
Val-Exp: input nombre <>?
- ** Field Name: id_area
Description: Area del cliente. En esta tabla se asigna por programa.
Help: Teclee el area a la que pertenece ese nivel.
Val-Msg: El area no existe
Val-Exp: CAN-FIND (S_area of S_nivel)

=====
Table: S_programa
=====

Table Name	Dump Name	Table Field Flags	Index Count	Table Count	Label
S_programa	s_progra	13	2	?	

=====
field summary
=====

Order	Field Name	Data Type	Flags	Format	Initial
10	id_proyecto	inte	im	>9	0
20	id_modulo	inte	im	>9	0

30 id_programa	char	m	x(8)	
40 descripcion	char	m	x(60)	
50 password	char		x(8)	
70 hora_inicial	deci-2		>9.99	9.00
80 hora_final	deci-2		>9.99	12.00
90 dia_semana	logi[7]	m	si/no	no
140 id_area	char		x(4)	
150 id_autor	char		x(8)	
160 f_creacion	date		99/99/9999	?
170 parametros	char		x(30)	
180 compilado	logi	m	si/no	si

Field Name	Label	Column Label
id_proyecto	?	?
id_modulo	?	?
id_programa	?	?
descripcion	?	?
password	?	?
hora_inicial	?	?
hora_final	?	?
dia_semana	?	?
id_area	Area	Area
id_autor	?	?
f_creacion	?	?
parametros	?	?
compilado	?	?

=====
 ===== index summary =====
 =====

Flags	Index Name	Cnt	Field Name
u	id_programa	1	+ id_programa
pu	S_llave_prog	3	+ id_proyecto + id_modulo + id_programa

=====
 ===== field details =====
 =====

- ** Field Name: id_proyecto
 Description: Proyecto de sistemas.
 Help: Teclee el numero de proyecto.
 Val-Msg: No existe el proyecto.
- ** Field Name: id_modulo
 Description: Numero asignado a un modulo dentro de un proyecto.
 Help: Teclee el numero de modulo.
 Val-Msg: No existe este modulo.
- ** Field Name: id_programa
 Description: Nombre del programa.
 Help: Teclee el nombre del programa.
 Val-Msg: Debe proporcionarse el nombre del programa.
- ** Field Name: descripcion
 Description: Descripción del programa.
 Help: Teclee la descripción del programa.
 Val-Msg: Debe proporcionarse la descripción del programa.
 Val-Exp: input descripcion <>?

- ** Field Name: password
Description: Password requerido para acceder el programa.
Help: Teclee el password para acceder el programa.
- ** Field Name: hora_inicial
Description: Hora desde la cual se puede acceder el programa.
Help: Teclee la hora desde la cual se puede acceder el programa.
Val-Msg: El horario de inicio puede ser desde las 7 a.m.
Val-Exp: input hora_inicial >= 7
- ** Field Name: hora_final
Description: Hora hasta la cual se puede acceder el programa.
Help: Teclee la hora hasta la que se permite acceder el programa.
Val-Msg: La hora de terminación puede ser hasta las 18 máximo.
Val-Exp: input hora_final > input hora_inicial
- ** Field Name: dia_semana
Help: Teclee si o no dependiendo de si este día se trabaja.
- ** Field Name: id_area
Description: Indica el area donde se localiza el usuario
Help: Teclee el area donde se localiza el usuario
- ** Field Name: id_autor
Description: Autor del programa
Help: Teclee el autor del programa
- ** Field Name: f_creacion
Description: Fecha de creación
Help: Teclee la fecha de creación del programa
- ** Field Name: parametros
Description: Parametros del programa
Help: Teclee los parametros
- ** Field Name: compilado
Help: Teclee si el programa debe estar o no compilado

Table: S_tab_cod_imp

Table Name	Dump Name	Table Field Flags	Index Count	Table Count	Label
S_tab_cod_imp	tab_cod_	3	1	?	

field summary

Order	Field Name	Data Type	Flags	Format	Initial
10	tipo_impresion	char	i	x(10)	
20	nombre_codigo	char	i	x(15)	
40	asc_code	char		x(75)	

Field Name	Label	Column Label
tipo_impresion	Tipo Impresión	Tipo Impresión
nombre_codigo	Nombre Código	Nombre Código
asc_code	Código ASCII	Código ASCII

index summary

Flags	Index Name	Cnt	Field Name
pu	tipo_impresion	2	+ tipo_impresion + nombre_codigo

=====
 ===== field details =====
 =====

- ** Field Name: tipo_impresion
 Description: Tipo de impresión a utilizar.
 Help: Teclee el tipo de impresión a utilizar.
- ** Field Name: nombre_codigo
 Description: Nombre del Código de impresión.
 Help: Teclee el nombre del Código de impresión.
- ** Field Name: asc_code
 Description: Código ASCII del tipo de impresión. El usuario no lo debe de capturar.
 Help: Código ASCII del tipo de impresión.

=====
 ===== Table: S_usuario =====
 =====

Table Name	Dump Name	Table Field Flags	Index Count	Table Count	Label
S_usuario	s_usuari	11	2	?	

=====
 ===== field summary =====
 =====

Order	Field Name	Data Type	Flags	Format	Initial
10	id_usuario	char	im	x(8)	
30	id_nivel	inte	m	>9	0
40	nombre	char	m	X(40)	
50	password	char	m	x(8)	
60	f_cambio_psw	date		99/99/9999	?
70	duracion_psw	inte	m	>>9	0
80	f_alta	date		99/99/9999	TODAY
90	id_area	char	i	x(4)	
100	id_programa	char		x(8)	
110	menu	char		x(8)	
120	id_promotor	inte		>>>>9	0

Field Name	Label	Column Label
id_usuario	?	?
id_nivel	?	?
nombre	?	?
password	?	?
f_cambio_psw	?	?
duracion_psw	?	?
f_alta	F. alta	F. alta
id_area	?	?
id_programa	?	?
menu	?	?
id_promotor	?	?

=====
 ===== index summary =====
 =====

Flags	Index Name	Cnt	Field Name
	ID_AREA	1	id_area

pu id_user 1 + id_usuario

=====
=====
field details
=====

- ** Field Name: id_usuario
Description: Identificación del usuario del proyecto.
Help: Teclee la identificación del usuario.
Val-Msg: Debe proporcionarse la identificación del usuario.
Val-Exp: input id_usuario <>?
- ** Field Name: id_nivel
Description: Nivel dentro de la jerarquía de un area interna de DBT.
Help: Teclee el numero de nivel del usuario.
Val-Msg: No existe este nivel.
- ** Field Name: nombre
Description: Nombre del usuario..
Help: Teclee el nombre del usuario.
Val-Msg: Debe proporcionarse el nombre del usuario.
Val-Exp: input nombre <>?
- ** Field Name: password
Description: Password del usuario.
Help: Teclee el password del usuario.
Val-Msg: Debe proporcionarse un password.
Val-Exp: input password <>?
- ** Field Name: f_cambio_psw
Description: Fecha del ultimo cambio de password. Se asigna por programa.
- ** Field Name: duracion_psw
Description: Vigencia del password en días.
Help: Teclee la duración del password en días.
Val-Exp: input duracion_psw > 0
- ** Field Name: f_alta
Description: Fecha de alta del usuario. Dada mediante un programa.
Val-Msg: La fecha de alta debe ser = hoy.
Val-Exp: INPUT f_alta = TODAY
- ** Field Name: id_area
Description: Area del usuario.
Help: Teclee el Area en que se localiza al usuario.
Val-Msg: El area del usuario debe existir.
- ** Field Name: id_programa
Description: Nombre del programa que se corre al inicio
Help: Teclee el programa inicial que corre el usuario
Val-Msg: El programa no existe
Val-Exp: Can-find (s_programa where s_programa.id_programa = input s_usuario.id_programa)
- ** Field Name: menu
Help: Teclee el nombre del menu que corre el usuario.
- ** Field Name: id_promotor
Description: Numero de promotor en su caso.

=====
=====
Table: U_asesor
=====

Table Name	Dump Name	Table Field Flags	Index Count	Table Count	Label
U_asesor	u_asesor		4	3	?

=====
=====
field summary
=====

APENDICE C. Diccionario de Datos

Order	Field Name	Data Type	Flags	Format	Initial
10	id_asesor	inte	im	>>>>9	0
20	nomb_asesor	char	im	x(30)	
30	id_ctro_fin	inte	im	>>>>9	0
40	Stat_Alta	logi	m	Alta/Cancelada	Alta

Field Name	Label	Column Label
id_asesor	Id_asesor	Id_asesor
nomb_asesor	Nomb.Asesor	Nomb.Asesor
id_ctro_fin	Ctro.Finan.	Ctro.Finan.
Stat_Alta	?	?

=====
index summary
=====

Flags	Index Name	Cnt	Field Name
pu	asesor	1	+ id_asesor
u	ctro_fin	2	+ id_ctro_fin + id_asesor
u	Nombre	1	+ nomb_asesor

=====
field details
=====

** Field Name: id_asesor
Description: Clave del Asesor.
Help: Teclee la clave del Asesor.
Val-Msg: El cliente debe ser mayor a Cero.
Val-Exp: INPUT id_asesor <> 0

** Field Name: nomb_asesor
Description: Nombre del asesor.
Help: Teclee el nombre del asesor.
Val-Msg: Dar de alta el nombre del asesor.
Val-Exp: INPUT nomb_asesor <> "

** Field Name: id_ctro_fin
Description: Centro financiero.
Help: Teclee el centro financiero.
Val-Msg: No existe el centro financiero el catalogo.
Val-Exp: CAN-FIND (U_ctro_fin WHERE U_ctro_fin.id_ctro_fin = INPUT U_asesor.id_ctro_fin)

** Field Name: Stat_Alta
Description: Estatus de la cuenta.
Help: Teclee "A" para dar la cuenta de alta.

=====
Table: U_cat_dist
=====

Table Name	Dump Name	Table Field	Index	Table
		Flags	Count	Count Label
U_cat_dist	u_cat_di	3	1?	

field summary

Order	Field Name	Data Type	Flags	Format	Initial
10	cod_distrib	char	i	999	
20	nombre	char	m	x(40)	?
30	stat_alta	logi	m	Alta/Cancelada	Alta

Field Name	Label	Column Label
cod_distrib	Cod. Distrib.	?
nombre	Nomb.Codigo	?
stat_alta	?	?

index summary

Flags	Index Name	Cnt	Field Name
pu	cod_dist	1	+ cod_distrib

field details

- ** Field Name: cod_distrib
Description: Código de Distribución "AreaDivPla" 999.
Help: Tecleé Código de Distribución.
Val-Msg: No se permite Código con Ceros.
Val-Exp: INPUT cod_distrib <> "000"
- ** Field Name: nombre
Description: Nombre del Código del Centro de Distr.
Help: Teclee el nombre del código.
Val-Msg: Debe proporcionarse el nombre del Centro de Distr.
Val-Exp: nombre <>?
- ** Field Name: stat_alta
Description: Status de la Código de Distribución.
Help: Teclee 'A' para dar la Cod. De Distrib.

Table: U_cheque

Table Name	Dump Name	Table Field	Index	Table
		Flags	Count	Count Label
U_cheque	u_cheque	9	5	?

field summary

Order	Field Name	Data Type	Flags	Format	Initial
10	id_ctro_fin	inte	im	>>>>9	0
20	id_asesor	inte	im	>>>>9	0
30	id_producto	char	i	!(1)	
40	id_cliente	inte	im	>>>>9	0
50	n_cheque	inte	i	>>>>9	0
60	f_cheque	date	i	99/99/9999	today

APÉNDICE C. Diccionario de Datos

```

90 n_testkey      inte  m  >>>>9      0
100 monto         deci-2 m  >>>>>>9.99 0
110 id_mov_che    char  im x(4)
    
```

Field Name	Label	Column Label
id_ctro_fin	Ctro.Finan.	Ctro.Finan.
id_asesor	Id_asesor	Id_asesor
id_producto	Id_producto	Id_producto
id_cliente	?	?
n_cheque	?	?
f_cheque	F.Cheque	F.Cheque
n_testkey	Testk	Testk
monto	Monto	Monto
id_mov_che	?	?

==== index summary =====

Flags Index Name	Cnt Field Name
asesor	1 + id_asesor
pu cliente	3 + id_producto + id_cliente + n_cheque
cte_fecha	3 + id_producto + id_cliente + f_cheque
ctro_fecha	2 + id_ctro_fin + f_cheque
mov_che	1 + id_mov_che

==== field details =====

** Field Name: id_ctro_fin

Description: Centro financiero.

Help: Teclee el centro financiero.

Val-Msg: No existe el centro financiero el catalogo.

Val-Exp: CAN-FIND (U_ctro_fin WHERE U_ctro_fin.id_ctro_fin = INPUT
U_cheque.id_ctro_fin)

** Field Name: id_asesor

Description: Clave del Asesor.

Help: Teclee la clave del Asesor.

Val-Msg: El numero del asesor no corresponde.

Val-Exp: CAN-FIND (U_asesor WHERE U_asesor.id_asesor = INPUT
U_cheque.id_asesor).

** Field Name: id_producto

Description: Sistema al que pertenece el cliente (U=Universal).

Help: Sistema al que pertenece el cliente

Val-Msg: Alta Unic. (U=Univ.)

Val-Exp: INPUT id_producto = "U"

** Field Name: id_cliente

Description: Identifica a los diferentes clientes de Banco.

Help: Teclee el numero de cliente.

Val-Msg: Teclee un numero de cliente valido.

Val-Exp: INPUT id_cliente > 0

- ** Field Name: n_cheque
Description: Numero de cheque.
Help: Teclee el numero de Cheque.
- ** Field Name: f_cheque
Description: Fecha del Cheque.
Help: Teclee la fecha del Cheque.
- ** Field Name: n_testkey
Description: Clave del test_key.
Help: Teclee el Test-Key.
- ** Field Name: monto
Description: Monto del cheque
Help: Teclee el monto del cheque
Val-Msg: Debe proporcionar el monto del cheque
Val-Exp: Monto < 0
- ** Field Name: id_mov_che
Description: Destino del cheque
Help: Indique cual es el destino del cheque
Val-Msg: No existe el tipo de movimiento
Val-Exp: can-find(u_movto_cheque OF u_cheque)

==== Table: U_cliente =====

Table Name	Dump Name	Table Flags	Field Count	Index Count	Table Label
U_cliente	u_client		42	10	?

==== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
4	id_producto	char	im	!	
8	id_cliente	inte	im	>>>>9	0
30	atencion	char	m	x(50)	
40	nombre	char	im	x(50)	?
60	pais	char		x(20)	
70	telefonos	char[4]		X(19)	
80	rfc	char		x(14)	
90	tipo_persona	inte	im	99	0
95	direccion	char[4]		x(50)	
97	c_p	char	m	99999	
100	direc_edo_cta	char[5]		x(35)	
110	f_alta	date	m	99/99/9999	TODAY
120	f_cancelacion	date		99/99/9999	?
130	cuenta_especial	logi	m	SI/NO	no
140	congelada	logi	m	Si/No	no
150	stat_alta	logi	m	Alta/Cancelada	Alta
160	saldo_total	deci-2		>>>,>>>,>>>,>>9	0
170	saldo_disponible	deci-2		>>>,>>>,>>>,>>9	0
190	person_giran	char[4]		x(30)	
200	giro	char		x(35)	
205	grupo	inte	i	>>>>9	0
210	accionistas	char[5]		x(30)	
220	administradores	char[5]		x(30)	
230	califica	char		!	
240	id_usuario	char		x(8)	

APÉNDICE C. Diccionario de Datos

250 id_promotor	inte	im >>>>9	0
260 id_responsab	inte	im >>>>9	0
270 id_area	inte	>9	0
280 id_division	inte	i >9	0
290 id_plaza	inte	i >9	0
300 id_sucursal	inte	i >9	0
310 id_estado	inte	m >9	0
320 id_segmento	inte	i 99	0
325 st_entrega	char	m x(2)	
330 relacion	inte	99	0
340 subsubsubcta	inte	99	0
390 cte_credito	logi	Si/No	no
400 cte_inver	logi	Si/No	no
440 cte_tarjeta	logi	Si/No	no
450 id_cheq	logi	m Mancomunada/Ind no	
460 cod_distrib	char	i 999	
470 id_talonario	logi	Si/No	Si

Field Name Label Column Label

Field Name	Label	Column Label
id_producto	Id_producto	Id_producto
id_cliente	?	?
atencion	?	?
nombre	?	?
pais	?	?
telefonos	?	?
rfc	?	?
tipo_persona	?	?
direccion	Direccion	Direccion
c_p	C.P.	C.P.
direc_edo_cta	?	?
f_alta	?	?
f_cancelacion	?	?
cuenta_especial	?	?
congelada	?	?
stat_alta	?	?
saldo_total	?	?
saldo_disponible	?	?
person_giran	?	?
giro	?	?
grupo	?	?
accionistas	?	?
administradores	?	?
califica	?	?
id_usuario	?	?
id_promotor	?	?
id_responsab	?	?
id_area	?	?
id_division	?	?
id_plaza	?	?
id_sucursal	?	?
id_estado	?	?
id_segmento	?	?
st_entrega	?	?
relacion	?	?
subsubsubcta	?	?
cte_credito	Cte.Credito	?
cte_inver	?	?

cte_tarjeta	Cte.tarjeta	?
id_cheq	?	?
cod_distrib	?	?
id_talonario	Talonario	Talonario

=====
===== index summary =====

Flags	Index Name	Cnt	Field Name
pu	cliente	2	+ id_producto + id_cliente
u	clie_per	3	+ id_producto + id_cliente + tipo_persona
	cod_dist	1	+ cod_distrib
	grupo	1	+ grupo
	id_promotor	1	+ id_promotor
	id_responsab	1	+ id_responsab
a	nombre	1	+ nombre
	perfil	4	+ id_division + id_plaza + id_sucursal + id_cliente
	prod_nombre	2	+ id_producto + nombre
	segmento	1	+ id_segmento

=====
===== field details =====

- ** Field Name: id_producto
Description: Sistema al que pertenece el cliente.
Help: Sistema al que pertenece el cliente
Val-Msg: Dar de alta (M=Mesa,Q=Chequera,U=Univ.)
Val-Exp: INPUT id_producto = "M" OR INPUT id_producto = "Q" OR INPUT
id_producto = "C" OR INPUT id_producto = "V" OR INPUT
id_producto = "U"
- ** Field Name: id_cliente
Description: Identifica a los diferentes clientes de D.B.T..
Help: Teclee el numero de cliente.
Val-Msg: Teclee un numero de cliente valido.
Val-Exp: INPUT id_cliente > 0
- ** Field Name: atencion
Description: Teclee el nombre de la persona que atiende en esa empresa.
Help: Teclee el nombre de la persona que atiende en esa empresa.
Val-Msg: El nombre de la persona que atiende es requerido.
- ** Field Name: nombre
Description: Nombre del Cliente.
Help: Teclee el nombre del cliente.
Val-Msg: Debe proporcionarse el nombre.
Val-Exp: nombre <> "
- ** Field Name: pais
Description: País donde radica el cliente.
Help: Teclee el país donde radica el cliente.
- ** Field Name: telefonos
Description: Teléfono del cliente.

- Help: Proporcione el teléfono del cliente.
- ** Field Name: rfc
Description: Registro Federal de Contribuyentes.
Help: Teclee el Registro Federal de Contribuyentes.
- ** Field Name: tipo_persona
Description: Tipo de Persona.
Help: Debe ser mayor que cero
Val-Msg: Debe proporcionarse el tipo de persona mayor que cero.
Val-Exp: INPUT Tipo_persona > 0
- ** Field Name: direccion
Description: Dirección del Cliente.
Help: Teclee la Dirección del Cliente.
- ** Field Name: c_p
Description: Código Postal.
Help: Teclee el código postal.
- ** Field Name: direc_edo_cta
Description: Dirección que aparece en el edo. de cta..
Help: Proporcione la leyenda que aparece en el estado de cuenta.
- ** Field Name: f_alta
Description: Fecha de ingreso del cliente.
Help: Teclee la fecha de ingreso como cliente.
Val-Msg: Debe ser una fecha Hábil.
Val-Exp: WEEKDAY(INPUT F_ALTA) < 1 AND WEEKDAY(INPUT F_ALTA) < 7
- ** Field Name: f_cancelacion
Description: Fecha en la que se cancela la cuenta.
Help: Proporcione la fecha de cancelación en su caso.
Val-Msg: Debe digitar la fecha de canc. menor o igual a la de ahora
Val-Exp: INPUT f_cancelacion <= TODAY
- ** Field Name: cuenta_especial
Description: Cuenta especial.
Help: Teclee "S" si se trata de una Cuenta Especial.
- ** Field Name: congelada
Description: Esta la cuenta congelada?
Help: Teclee una "S" para CONGELAR la cuenta.
- ** Field Name: stat_alta
Description: Status de la cuenta.
Help: Teclee "A" para dar la cuenta de alta.
- ** Field Name: saldo_total
Description: Saldo total final.
Help: Saldo total de la cuenta.
- ** Field Name: saldo_disponible
Description: Saldo Disponible.
Help: Saldo disponible.
- ** Field Name: person_giran
Description: Nombre de las personas que Giran.
Help: Proporcione el nombre de las personas que Giran.
- ** Field Name: giro
Description: Ocupación o Giro de la empresa.
Help: Proporcione la Ocupación o Giro de la empresa.
- ** Field Name: grupo
Description: Grupo al que pertenece el cliente.
Help: Teclee el numero de grupo al que pertenece el cliente.
- ** Field Name: accionistas
Description: Accionistas.
Help: Teclee los accionistas.
- ** Field Name: administradores
Description: Administradores del cliente.
Help: Teclee los administradores.

- ** Field Name: califica**
 Description: Nivel de Riesgo que representa el cliente para el Grupo.
 Help: Teclee el nivel de Riesgo o Calificación del cliente(A...E).
- ** Field Name: id_usuario**
 Description: Identifica al responsable que llenó la carátula.
 Help: Teclee el responsable.
- ** Field Name: id_promotor**
 Description: Numero que identifica que promotor atiende al cliente.
 Help: Teclee el numero de promotor que atiende al cliente.
 Val-Msg: Este promotor no existe.
 Val-Exp: CAN-FIND(M_promotor OF U_cliente)
- ** Field Name: id_responsab**
 Description: Responsable de las operaciones de este cliente.
 Help: Teclee el numero de Promotor responsable del cliente.
 Val-Msg: El responsable no existe.
 Val-Exp: CAN-FIND (M_promotor WHERE M_promotor.id_promotor = U_cliente.id_responsab)
- ** Field Name: id_area**
 Description: Area del cliente.
 Help: Area en la que se encuentra ubicado el cliente.
 Val-Msg: No existe area especificada.
 Val-Exp: CAN-FIND (FIRST M_ubicacion WHERE M_ubicacion.id_area = INPUT U_cliente.id_area AND M_ubicacion.id_division = 0.
- ** Field Name: id_division**
 Description: Division del Cliente.
 Help: Division del cliente. En esta tabla se asigna por programa.
 Val-Msg: La división no existe.
 Val-Exp: CAN-FIND(FIRST M_ubicacion WHERE M_ubicacion.id_area = INPUT U_cliente.id_area AND M_ubicacion.id_division = INPUT U_cliente.id_division) and INPUT id_division <> 0
- ** Field Name: id_plaza**
 Description: Plaza del cliente.
 Help: Plaza del cliente. En esta tabla se asigna por programa.
 Val-Msg: La plaza no existe.
 Val-Exp: can-find(first M_ubicacion OF U_cliente) and INPUT id_plaza <> 0
- ** Field Name: id_sucursal**
 Description: Sucursal del Cliente.
 Help: Sucursal del cliente. En esta tabla se asigna por programa.
 Val-Msg: La sucursal no existe.
 Val-Exp: Can-find(first M_sucursal WHERE M_sucursal.id_area = INPUT id_area AND M_sucursal.id_division = INPUT id_division AND M_sucursal.id_plaza = INPUT U_cliente.id_plaza AND M_sucursal.id_sucursal=INPUT id_sucursal)
- ** Field Name: id_estado**
 Description: Clave del estado.
 Help: Teclee la clave del estado.
 Val-Msg: No existe el estado.
 Val-Exp: CAN-FIND (G_estado OF U_cliente)
- ** Field Name: id_segmento**
 Description: Clave de segmentos (Giros).
 Help: Teclee la clave de segmento al que pertenece el cliente.
 Val-Msg: No existe ese numero de segmento.
 Val-Exp: CAN-FIND (FIRST C_segmento WHERE C_segmento.id_segmento = U_cliente.id_segmento)
- ** Field Name: st_entrega**
 Description: CO=Correo, RE=Retener, VA=Valija.
 Help: Teclee "CO" correo, "RE" retener, "VA" valija.

Val-Msg: Teclee "CO" correo, "RE" retener, "VA" valija.
 Val-Exp: LOOKUP(ST_ENTREGA, "CO,RE,VA") <> 0
**** Field Name:** relacion
 Description: Related, Unrelated, Within, Abroad.
 Help: Teclee la clave de relacion.
 Val-Msg: No corresponde esa relacion con el Catalogo.
 Val-Exp: CAN-FIND (K_relacion WHERE K_relacion.relacion =
 U_cliente.relacion)

**** Field Name:** subsubsubcta
 Description: Sub-sub-subcuenta
 Help: Teclee la sub-sub-subcuenta.

**** Field Name:** cte_credito
 Description: Si el Cliente tiene Crédito.
 Help: Teclee Si=Si tiene Créditos.

**** Field Name:** cte_inver
 Description: Si el Cliente tiene Inversiones.
 Help: Teclee Si=Si tiene Inversiones.

**** Field Name:** cte_tarjeta
 Description: Si el Cliente tiene Tarjeta de Crédito.
 Help: Teclee Si=Si tiene Tarjeta de Crédito

**** Field Name:** id_cheq
 Help: Indique si la cuenta es Mancomunada o Indistinta

**** Field Name:** id_talonario
 Description: Talonario Si = Tiene Talonario No = No Tiene Talonario.
 Help: Las Cuentas que tienen Talonario para el cobro de comisiones.

=====
 Table: U_ctro_fin
 =====

Table Name	Dump Name	Table Field Index	Flags	Count	Count	Label
------------	-----------	-------------------	-------	-------	-------	-------

U_ctro_fin	u_ctro_f	2	2?			
------------	----------	---	----	--	--	--

=====
 field summary
 =====

Order	Field Name	Data Type	Flags	Format	Initial
-------	------------	-----------	-------	--------	---------

10	id_ctro_fin	inte	im	>>>>9	0
20	nomb_cf	char	im	x(30)	

Field Name	Label	Column Label
------------	-------	--------------

id_ctro_fin	Ctro.Finan.	Ctro.Finan.
nomb_cf	Nomb.Cf.	Nomb.Cf.

=====
 index summary
 =====

Flags	Index Name	Cnt	Field Name
-------	------------	-----	------------

pu	ctro_fin	1	+ id_ctro_fin
u	ctro_nom	1	+ nomb_cf

=====
 field details
 =====

- ** Field Name: id_ctro_fin
Description: Centro financiero.
Help: Teclee el centro financiero.
Val-Msg: El centro financiero debe ser mayor a cero.
Val-Exp: INPUT id_ctro_fin > 0
- ** Field Name: nomb_cf
Description: Nombre del centro financiero.
Help: Teclee el nombre del centro financiero.
Val-Msg: El nombre del centro financiero debe ser dif. a blancos.
Val-Exp: INPUT nomb_cf <> "

==== Table: U_memo =====

Table Name	Dump Name	Table Field Flags	Index Count	Table Count	Index Label
U_memo	u_memo		4	1	?

==== field summary =====

Order	Field Name	Data Type	Flags	Format	Initial
10	Id_cliente	inte	i	>>>>9	0
20	F_recep_microf	date		99/99/9999	?
30	F_envio_miami	date		99/99/9999	?
40	F_envio_microf	date		99/99/9999	?

Field Name	Label	Column Label
Id_cliente	Cliente	Cliente
F_recep_microf	Fecha Microfilm	Fecha!Microfilm
F_envio_miami	Fecha Miami	Fecha Miami
F_envio_microf	Fecha Envío Microf	Fecha Envío Microf

==== index summary =====

Flags	Index Name	Cnt	Field Name
pu	cliente	1	Id_cliente

==== field details =====

- ** Field Name: F_recep_microf
Help: Teclee la fecha de recepción de microfilmación
- ** Field Name: F_envio_miami
Help: Teclee la fecha de envío a Miami
- ** Field Name: F_envio_microf
Description: Fecha en que se envía a microfilmación

==== Table: u_movto_cheque =====

Table Name	Dump Name	Table Field Flags	Index Count	Table Count	Index Label
------------	-----------	-------------------	-------------	-------------	-------------

u_movto_cheque u_movto_ 3 1 ?

=====
field summary
=====

Order	Field Name	Data Type	Flags	Format	Initial
10	id_mov_che	char	im	!!!!	
20	Movimiento	char	m	x(20)	
30	afecta_monto	logi	m	SI/NO	NO

Field Name	Label	Column Label
id_mov_che	?	?
Movimiento	?	?
afecta_monto	?	?

=====
index summary
=====

Flags	Index Name	Cnt	Field Name
pu	movtos_cheque	1	id_mov_che

=====
field details
=====

- ** Field Name: id_mov_che
Description: Identificador del tipo de movimiento
Help: Proporcione el identificador del tipo de movimiento.
Val-Msg: Debe proporcionar el tipo de movimiento
Val-Exp: id_mov_che <> "
- ** Field Name: Movimiento
Description: Descripcion de lo que se va a hacer con el cheque
Help: Teclee la descripcion del movimiento
Val-Msg: Debe proporcionar el texto del movimiento
Val-Exp: movimiento <> "
- ** Field Name: afecta_monto
Description: Indicador para saber si el movto afecta el monto diario disponible
Help: Indique si el movimiento afecta el monto diario disponible
Val-Msg: Debe indicar si afecta o no al monto disponible diario
Val-Exp: afecta_monto <> ?

=====
end sequences
=====

APÉNDICE D
Código 4GL Adicional
al Generado por Progress

```

/*
*****
*****
Control
Definitions
***** */
/* Define the
widget handle
for the window
*/
DEFINE VAR C-
Win AS WIDGET-
HANDLE NO-UNDO.

/* Definitions
of the field
level widgets
*/
DEFINE BUTTON
BUTTON-2 AUTO-
GO
IMAGE-UP
FILE
"adeicon\run":U
LABEL
"Button 2"
SIZE 8 BY
2.11.

DEFINE BUTTON
BUTTON-3 AUTO-
END-KEY
IMAGE-UP
FILE
"adeicon\exit-
au":U
LABEL
"Button 3"
SIZE 8 BY
2.11.

DEFINE VARIABLE
t_fecha_mov AS
DATE FORMAT
"99/99/9999":U
LABEL
"Fecha Corte"
VIEW-AS
FILL-IN
SIZE 16 BY
1 TOOLTIP
"Fecha del
Listado" NO-
UNDO.

DEFINE VARIABLE
t_poliza AS
INTEGER FORMAT
"999,999":U
INITIAL 0

```

```

LABEL "No.
de Póliza"
VIEW-AS
FILL-IN
SIZE 16 BY
1 TOOLTIP
"Poliza
Contable
Entero" NO-
UNDO.

DEFINE
RECTANGLE RECT-
4
EDGE-
PIXELS 2
GRAPHIC-EDGE
NO-FILL
SIZE 100
BY 11.05.

DEFINE
RECTANGLE RECT-
5
EDGE-
PIXELS 2
GRAPHIC-EDGE
NO-FILL
SIZE 100
BY 2.63.

/*
*****
*****
Frame
Definitions
***** */

DEFINE FRAME
DEFAULT-FRAME
t_fecha_mov AT
ROW 4:16 COL 44
COLON-ALIGNED
t_poliza
AT ROW 6 COL 44
COLON-ALIGNED
AUTO-RETURN
BUTTON-2
AT ROW 13.9 COL
88
BUTTON-3
AT ROW 13.9 COL
97
RECT-5 AT
ROW 13.63 COL 7
RECT-4 AT
ROW 2.32 COL 7
"
Contabilizar

```

```

Mov. Chemical"
VIEW-AS TEXT
SIZE
32 BY 1.05 AT
ROW 1.79 COL 43
BGCOLOR 9
FGCOLOR 14
WITH 1-DOWN
NO-BOX KEEP-
TAB-ORDER
OVERLAY
SIDE-
LABELS NO-
UNDERLINE
THREE-D
AT COL
1 ROW 1
SIZE
112.4 BY 16.84.

/*
*****
*****
Procedure
Settings
***** */

&ANALYZE-
SUSPEND
PROCEDURE-
SETTINGS
/* Settings for
THIS-PROCEDURE
Type: Window
Allow:
Basic, Browse, DB
Fields, Window, Q
uery
Other
Settings:
COMPILE
*/
&ANALYZE-RESUME
_END-PROCEDURE-
SETTINGS

/*
*****
*****
Create Window
***** */

&ANALYZE-
SUSPEND
_CREATE-WINDOW
IF
SESSION:DISPLAY

```

```

-TYPE = "GUI":U
THEN
CREATE WINDOW
C-Win ASSIGN
HIDDEN
= YES
TITLE
= "Sistema de
Chequera en
Dólares"
HEIGHT
= 16.42
WIDTH
= 111.4
MAX-
HEIGHT
= 17.84
MAX-
WIDTH
= 114.4
VIRTUAL-HEIGHT
= 17.84
VIRTUAL-WIDTH
= 114.4
RESIZE
= yes
SCROLL-BARS
= no
STATUS-AREA
= yes
BGCOLOR
= ?
FGCOLOR
= ?
KEEP-
FRAME-Z-ORDER =
yes
THREE-
D
=
yes
MESSAGE-AREA
= no
SENSITIVE
= yes.
ELSE {&WINDOW-
NAME} =
CURRENT-WINDOW.
/* END WINDOW
DEFINITION
*/
&ANALYZE-RESUME

/*
*****
*****

```

```

Runtime
Attributes and
UIB Settings
***** */

&ANALYZE-
SUSPEND RUN-
TIME-ATTRIBUTES
/* SETTINGS FOR
WINDOW C-Win
VISIBLE,,RUN-
PERSISTENT
*/
/* SETTINGS FOR
FRAME DEFAULT-
FRAME
*/
ASSIGN
t_fecha_mov:AUT
O-RESIZE IN
FRAME DEFAULT-
FRAME
=
TRUE
t_fecha_mov:PRI
VATE-DATA IN
FRAME DEFAULT-
FRAME
=
"Today".
ASSIGN
t_poliza:AUTO-
RESIZE IN FRAME
DEFAULT-FRAME
= TRUE.
IF
SESSION:DISPLAY
-TYPE = "GUI":U
AND VALID-
HANDLE(C-Win)
THEN C-
Win:HIDDEN =
no.

/* --RUN-TIME-
ATTRIBUTES-END
*/
&ANALYZE-RESUME

/*
*****
*****

```

```

Control      APPLY
Triggers     "CLOSE":U TO
*****      THIS-PROCEDURE.
*****      RETURN NO-
*****      APPLY.
             END.
&Scoped-define
SELF-NAME C-Win
&ANALYZE-
SUSPEND _UIB-
CODE-BLOCK
_CONTROL C-Win
C-Win
ON END-ERROR OF
C-Win /*
Sistema de
Chequera en
Dólares */
OR ENDKEY OF
{&WINDOW-NAME}.
ANYWHERE DO:
/* This case
occurs when the
user presses
the "Esc" key.
In a
persistently
run window,
just ignore
this. If we
did not, the
application
would exit. */
IF THIS-
PROCEDURE:PERSI-
STENT THEN
RETURN NO-
APPLY.
END.
/* _UIB-CODE-
BLOCK-END */
&ANALYZE-RESUME
&ANALYZE-
SUSPEND _UIB-
CODE-BLOCK
_CONTROL C-Win
C-Win
ON WINDOW-CLOSE
OF C-Win /*
Sistema de
Chequera en
Dólares */
DO:
/* This event
will close the
window and
terminate the
procedure. */

```

```

APPLY
IF NOT
AVAILABLE
Q_mov_chem THEN
DO:
MESSAGE
"No existen
movimientos en
esa fecha para
ser
contabilizados"
PAUSE 3.
RETURN.
END.
/*-----
-----*/
validacion de
movimientos a
esa fecha
-----*/
DISPLAY
"Verificando
fechas. Espere
un momento
....."
WITH
FRAME X
CENTERED
ROW 15.
NO-LABELS.
FOR EACH
Q_mov_chem
WHERE
Q_mov_chem.f_mo
vto =
Po_fechaini.
USE-INDEX
contab.
FIND
FIRST Q_Tpmovto
WHERE
Q_Tpmovto.id_mo
vto =
Q_mov_chem.id_m
ovto
USE-INDEX
tpmovto.
IF
Q_tpmovto.cuent
a_cargar <> ""
AND
Q_tpmovto.cuent

```

```

a_abonar <> ""
AND
NOT
Q_mov_chem.cont
abilizado THEN
n_movtos =
n_movtos + 1.
END.
IF n_movtos =
0 THEN DO:
MESSAGE
"Los
movimientos a
esta fecha ya
fueron
contabilizados"
APPLY
'CLOSE' TO
THIS-PROCEDURE.
END.
DISPLAY
"Generando el
archivo.
Espere un
momento"
WITH
CENTERED
ROW 15
COLOR
WHITE/BLUE
NO-LABELS.
dato =
STRING(Po_fecha
ini).
archivo =
"C:\jpol\txt\ch
" +
SUBSTRING(dato,
1,2) +
SUBSTRING(dato,
4,2) +
SUBSTRING(dato,
7,2) + ".txt".
fecha =
INTEGER(SUBSTRI
NG(dato,4,2) +
SUBSTRING(dato,
1,2) +

```

```

SUBSTRING(dato,
7,2)).
/*DOS SILENT
VALUE("DEL " +
archivo). jpol.
oct 21 98*/
OUTPUT STREAM
report TO
VALUE(archivo).
Po_concepto =
"ACCOUNT
ENTRIES 22809
CHEMICAL "
DISPLAY
STREAM report
reg_pol
fecha
Po_tpol
Po_poliza
Po_clase
Po_diarioagr
Po_concepto
WITH
FRAME A NO-
LABEL NO-BOX.
WIDTH 64
STREAM-IO.
Tot_fisica =
0.
Tot_Moral =
0.
FOR EACH
Q_mov_chem
WHERE
Q_mov_chem.f_mo
vto =
Po_fechaini AND
NOT
Q_mov_chem.cont
abilizado
USE-INDEX
contab.
EACH
Q_Tpmovto
WHERE
Q_Tpmovto.id_mo
vto =

```

Q_mov_chem.id_m ovto AND	gar,7,2) = "XX" THEN DO:	cta_abo_moral = cuenta_abonar. END.	Mo_diarioesp WITH FRAME C NO-LABEL NO-BOX	WIDTH 32 STREAM-IO. IF
Q_tpmovto.cuent a_cargar <> "" AND	cta_carg_fisica = SUBSTRING(cuent a_cargar,1,6) + "04" +	desc_mov = Q_tpmovto.Descr ip_larga.	WIDTH 49 STREAM-IO.	cta_abo = "" then cta_abo = cta_abo_moral.
Q_tpmovto.cuent a_abonar <> "" USE-INDEX tpmovto, EACH	SUBSTRING(cuent a_cargar,9,2) .	IF Tot_fisica > 0 THEN DO:	DISPLAY STREAM report	DISPLAY STREAM report
U_cliente	cta_carg_moral = SUBSTRING(cuent a_cargar,1,6) + "03" +	DISPLAY STREAM report	reg_descmov	reg_mov
WHERE	SUBSTRING(cuent a_cargar,9,2) . END.	cta_carg_fisica	desc_mov	cta_abo_moral
U_cliente.id_pr oducto =	SUBSTRING(cuent a_cargar,9,2) . END.	reg_mov	WITH FRAME B1 NO-LABEL NO-BOX	Mo_ref
Q_mov_chem.id_p roducto AND	ELSE DO:	cta_carg_fisica	WIDTH 32 STREAM-IO.	Mo_abono
U_cliente.id_cl iente =	cta_carg_fisica = cuenta_cargar.	Mo_ref	Tot_fisica = 0. END.	Tot_moral
Q_mov_chem.id_c liente	cta_carg_moral = cuenta_cargar. END.	Mo_cargo	IF Tot_moral > 0 THEN DO:	Mo_diarioesp
USE-INDEX cliente	IF SUBSTRING(Q_tpm ovto.cuenta_abo nar,7,2) = "XX" THEN DO:	Tot_fisica	DISPLAY STREAM report	WITH FRAME E NO-LABEL NO-BOX
BREAK BY	IF	Mo_diarioesp	Tot_moral > 0 THEN DO:	WIDTH 49 STREAM-IO.
Q_mov_chem.id_m ovto:	IF SUBSTRING(Q_tpm ovto.cuenta_abo nar,7,2) = "XX" THEN DO:	WITH FRAME B NO-LABEL NO-BOX	DISPLAY STREAM report	DISPLAY STREAM report
IF	IF	WIDTH 49 STREAM-IO.	reg_mov	reg_descmov
U_cliente.tipo_ persona = 4 THEN /* Persona fisica */	cta_abo_fisica = SUBSTRING(cuent a_abonar,1,6) + "04" + SUBSTRING(cuent a_abonar,9,2) .	DISPLAY STREAM report	cta_carg_moral	desc_mov
Tot_fisica = Tot_fisica + Q_mov_chem.mont o.	cta_abo_moral = SUBSTRING(cuent a_abonar,1,6) + "03" + SUBSTRING(cuent a_abonar,9,2) . END.	reg_descmov	Mo_ref	WITH FRAME B1 NO-LABEL NO-BOX
ELSE	END.	desc_mov	Mo_cargo	WIDTH 32 STREAM-IO.
Tot_moral = Tot_moral + Q_mov_chem.mont o.	cta_abo_fisica = cuenta_abonar. END.	WITH FRAME B1 NO-LABEL NO-BOX	Tot_moral	Tot_moral = 0. END.
IF LAST- OF(Q_mov_chem.i d_movto) THEN DO:	SUBSTRING(cuent a_abonar,9,2) . END.	WIDTH 32 STREAM-IO.	Mo_diarioesp	END. /*-----
cta_carg = "".	IF SUBSTRING(Q_tpm ovto.cuenta_car	DISPLAY STREAM report	WITH FRAME D NO-LABEL NO-BOX	----- -- FIN DEL LAST-OF ----- ---*/
cta_abo = "".	DO:	reg_mov	DISPLAY STREAM report	
IF	ELSE	cta_abo_fisica	reg_descmov	ASSIGN Q_mov_chem.cont abilizado = YES.
SUBSTRING(Q_tpm ovto.cuenta_car	cta_abo_fisica = cuenta_abonar. END.	Mo_ref	desc_mov	END.
	cta_abo_fisica = cuenta_abonar. END.	Mo_abono	WITH FRAME B1 NO-LABEL NO-BOX	
		Tot_fisica		

<pre>OUTPUT STREAM report CLOSE. MESSAGE "FINALIZO LA CONTABILIZACION " APPLY !CLOSE! TO THIS- PROCEDURE. END. /*DO*/ /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME &Scoped-define SELF-NAME t_fecha_mov &ANALYZE- SUSPEND _UIB- CODE-BLOCK _CONTROL t_fecha_mov C- Win. ON LEAVE OF t_fecha_mov IN FRAME DEFAULT- FRAME /* Fecha Corte */ DO: t_fecha_mov = INPUT t_fecha_mov. END. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME &Scoped-define SELF-NAME t_poliza &ANALYZE- SUSPEND _UIB- CODE-BLOCK _CONTROL t_poliza C-Win ON LEAVE OF t_poliza IN FRAME DEFAULT- FRAME /* No. de Póliza */ DO: t_poliza = INPUT t_poliza. END. /* _UIB-CODE- BLOCK-END */</pre>	<pre>&ANALYZE-RESUME condition. /* (NOTE: handle ERROR- and END-KEY so cleanup code will always fire. */ MAIN-BLOCK: DO ON ERROR UNDO MAIN- BLOCK, LEAVE MAIN-BLOCK ON END-KEY: UNDO MAIN- BLOCK; LEAVE MAIN-BLOCK; RUN enable_UI. IF NOT THIS- PROCEDURE:PERSI- STENT THEN WAIT-FOR CLOSE OF THIS- PROCEDURE. END. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME /* The CLOSE event can be used from inside or outside the procedure to */ /* terminate it. ON CLOSE OF THIS-PROCEDURE RUN disable_UI. /* Best default for GUI applications is. */ PAUSE 0 BEFORE- HIDE. /* Now enable the interface and wait for the exit.</pre>	<pre>interface by deleting dynamic widgets we have created and/or hide frames. This procedure is usually called when we are ready to "clean-up" after running. ----- These statements here are based on the "Other Settings" section of the widget Property Sheets. ----- /* Delete the WINDOW we created */ IF SESSION:DISPLAY -TYPE = "GUI":U AND VALID- HANDLE(C-Win) THEN DELETE WIDGET C-Win. IF THIS- PROCEDURE:PERSI- STENT THEN: DELETE PROCEDURE:THIS- PROCEDURE. END PROCEDURE. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME disable_UI C- Win DEFAULT- DISABLE PROCEDURE disable_UI: /*----- ----- Purpose: DISABLE the User Interface Parameters: <none> Notes: Here we clean- up the user-</pre>	<pre>Notes: Here we display/view/en- able the widgets in the user-interface. In addition, OPEN all queries associated with each FRAME and BROWSE. These statements here are based on the "Other Settings" section of the widget Property Sheets. ----- /* Delete the WINDOW we created */ IF SESSION:DISPLAY -TYPE = "GUI":U AND VALID- HANDLE(C-Win) THEN DELETE WIDGET C-Win. IF THIS- PROCEDURE:PERSI- STENT THEN: DELETE PROCEDURE:THIS- PROCEDURE. END PROCEDURE. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME enable_UI C-Win DEFAULT-ENABLE PROCEDURE enable_UI : /*----- ----- Purpose: ENABLE the User Interface Parameters: <none></pre>
--	---	---	---

```

*****
**/
/* Programa : QFCUADRO.P
*/
/* Objetivo : Dibuja un cuadro en la Laser Jet
*/
/* Autor : Equipo de Trabajo de TESIS
*/
/* Fecha : 01/31/2000
*/
/* Parametros :
x_ULRow Upper Left Row
*/
/* x_ULCol Upper Left Column
*/
/* x_RowTitle Ren glon de raya que separa titulo*/
/* x_HorWidth Horizontal Width
*/
/* x_VerWidth Vertical Width
*/
/* x_shadow Porcentaje a sombrear
*/
/*****
*****
*****
*****
**/

DEFINE INPUT PARAMETER x_ULRow AS INTEGER.
DEFINE INPUT PARAMETER x_ULCol AS INTEGER.
DEFINE INPUT PARAMETER x_RowTitle AS CHARACTER FORMAT "X(100)".
DEFINE SHARED STREAM impre.
/* Sombreado
*/
IF x_shadow <> 0 THEN
DO:
RUN
qfposit0 (INPUT x_ULRow , INPUT x_ULCol ).
x_str = " *c" + TRIM( STRING( x_HorWidth , ">>>>>>>" ) ) + "H".
x_str = x_str + " *c" + TRIM( STRING( x_RowTitle , ">>>>>>>" ) ) + "V".
x_str = x_str + " *c" + TRIM( STRING( x_shadow , ">>>>" ) ) + "G".
x_str = x_str + " *c2P".
PUT STREAM impre CONTROL x_str.
END.

/* RAYA DE ARRIBA y DERECHA SENCILLAS */
RUN qfraya0 ( INPUT x_ULRow , INPUT x_ULCol , INPUT 0 , INPUT x_HorWidth ).
RUN qfraya0 ( INPUT (x_ULRow + x_RowTitle) , INPUT x_ULCol , INPUT 0 , INPUT x_VerWidth ).
RUN qfposit0 (INPUT x_ULRow , INPUT (x_ULCol - 25) ).
x_str = " *c25H".
x_str = x_str + " *c" + TRIM( STRING( x_VerWidth , ">>>>>>>" ) ) + "V".
x_str = x_str + " *c0P".
PUT STREAM impre CONTROL x_str.
RUN qfposit0 (INPUT ( x_ULRow + x_VerWidth ) , INPUT (x_ULCol - 25) ).
x_str = " *c" + TRIM( STRING ( x_HorWidth + 29 ) , ">>>>>>" ) ) + "H".
x_str = x_str + " *c25V".
x_str = x_str + " *c0P".

PUT STREAM impre CONTROL x_str.
RETURN.
/*****
*****
*****
**/
/* Programa : QFESTADO.P
*/
/* Objetivo : Rutina común para cálculo de Estadísticas.
*/
/* Autor : Equipo de Trabajo de TESIS
*/
/* Fecha : 01/31/2000
*/
/*****
*****
*****
**/
DEFINE SHARED VARIABLE P_prod LIKE U_cliente.id_producto.
DEFINE SHARED VARIABLE P_cuenta LIKE U_cliente.id_cliente.
DEFINE SHARED VARIABLE P_F_inicial AS DATE FORMAT "99/99/9999".
DEFINE SHARED VARIABLE P_F_final AS DATE FORMAT "99/99/9999".
DEFINE SHARED VARIABLE p_SalIni LIKE Q_Corte.Saldo_Inicial INITIAL 0.

```



```

Q_Corte.Interes
INITIAL 0.
DEFINE SHARED
VARIABLE
P_Interes_anual
LIKE
Q_Corte.Interes
_Anuual INITIAL
0.
DEFINE SHARED
VARIABLE
P_Tasa_Int
LIKE
Q_Corte.Tasa_In
teres INITIAL
0.

DEFINE VARIABLE
x_mes
AS INTEGER
NO-UNDO.
DEFINE VARIABLE
x_anio
AS INTEGER
NO-UNDO.
DEFINE VARIABLE
x_anio_mes
AS INTEGER
NO-UNDO.
DEFINE VARIABLE
x_f_aux
AS DATE
FORMAT
"99/99/9999"
NO-UNDO.
DEFINE VARIABLE
x_encuentra
AS LOGICAL
NO-UNDO.
DEFINE VARIABLE
x_suma
LIKE
Q_Corte.Saldo_P
romedio INITIAL
0.
DEFINE VARIABLE
i
AS INTEGER.
DEFINE VARIABLE
x_cheque
AS LOGICAL.

/* Verifica si
las fechas es
un periodo
completo */

FIND U_cliente
WHERE
U_cliente.id_pr
oducto = p_prod
AND
U_cliente.id_cl
iente =
p_cuenta NO-
LOCK.
x_f_aux =
p_f_inicial +
32.
x_f_aux =
x_f_aux -
DAY(x_f_aux).
P_Tasa_Int =
0.
IF
YEAR(p_f_inicia
l) =
YEAR(p_f_final)
AND
MONTH(p_f_inici
al) =
MONTH(p_f_final
) AND
DAY(p_f_inicial
) = 1 AND
DAY(p_f_final)
= DAY(x_f_aux
) THEN
DO:
x_anio_mes
=
YEAR(p_f_inicia
l) * 100 +
MONTH(p_f_inici
al).
FIND
Q_Corte
WHERE
Q_corte.id_prod
ucto = p_prod
AND
Q_Corte.id_clie
nte = p_cuenta
AND
Q_Corte.anio_me
s_corte =
x_anio_mes
USE
INDEX corte NO-
LOCK NO-ERROR.
IF
AVAILABLE
Q_Corte THEN
DO:
P_SalIni
=
Q_Corte.Saldo_I
nicial.
P_SalFin
=
Q_Corte.Saldo_F
inal.
P_SalPro
=
Q_Corte.Saldo_P
romedio.
P_n_che_rebot
=
Q_Corte.n_chequ
es_rebotados.
P_n_che_emit
=
Q_Corte.n_chequ
es_emitidos.
P_Tot_Cheques
=
Q_Corte.Tot_Che
_Emitidos.
P_n_Debits
=
Q_Corte.n_Debit
s.
P_Tot_Debits
=
Q_Corte.Tot_Deb
its.
P_n_Credits
=
Q_Corte.n_Credi
ts.
P_Tot_Credits
=
Q_Corte.Tot_Cre
dits.
P_Interes
=
Q_Corte.Interes
_Anuual.
P_Interes_anual
=
Q_corte.Interes
_Anuual.
P_Tasa_Int
=
Q_Corte.Tasa_In
teres.
RETURN.
END.
FIND Q_Tasa
WHERE
Q_Tasa.Periodo
= x_anio_mes.
USE-INDEX tasa
NO-LOCK NO-
ERROR.
IF
AVAILABLE
Q_Tasa
THEN
IF
U_cliente.Tipo_
Persona = 4 /*
Fisica */
THEN P_Tasa_Int
= Tasa_Fisica.
ELSE P_Tasa_Int
= Tasa_Moral.
END.
IF
MONTH(p_f_inici
al) = 1 THEN
DO:
x_mes =
12.
x_anio =
YEAR(p_f_inicia
l) - 1.
END.
ELSE
DO:
x_mes =
MONTH(p_f_inici
al) - 1.
x_anio =
YEAR(p_f_inicia
l).
END.
x_anio_mes = (
x_anio * 100 )
+ x_mes.
FIND FIRST
Q_Parametro NO-
LOCK.
FIND LAST
Q_Corte
WHERE
Q_corte.id_prod
ucto = p_prod
AND
Q_Corte.id_clie
nte = p_cuenta
AND
Q_Corte.anio_me
s_corte <=
x_anio_mes
USE-INDEX
corte NO-LOCK
NO-ERROR.
IF NOT
AVAILABLE
Q_corte THEN
DO:
p_interes_Anuual
= 0.
p_SalFin =
0.
FIND FIRST
Q_movimiento
WHERE
Q_movimiento.id
_producto =
p_prod AND
Q_movimiento.id
_cliente =
p_cuenta AND
Q_Movimiento.f
movto <
p_f_inicial
USE-
INDEX cuenta
NO-LOCK NO-
ERROR.
END.
ELSE
DO:
p_SalFin =
Q_Corte.Saldo_F
inal.
IF
TRUNCATE(
Q_Corte.anio_me
s_corte / 100
0 ) = YEAR(
p_f_final )
THEN
P_Interes_Anuual
=
Q_Corte.Interes
_Anuual.
ELSE
P_Interes_Anuual
= 0.
IF (
Q_corte.anio_me
s_corte MODULO
100 ) = 12 THEN
DO:

```

```

x_mes
= 1.
x_anio
= TRUNCATE ((
Q_Corte.anio_me
s_corte / 100 )
, 0 ) + 1.
END.
ELSE
DO:
x_mes
= (
Q_corte.anio_me
s_corte MODULO
100 ) + 1.
x_anio
= TRUNCATE (
Q_Corte.anio_me
s_corte / 100
, 0 ).
END.
x_f_aux =
DATE( x_mes , 1
, x_anio ).
FIND FIRST
Q_Movimiento
WHERE
Q_movimiento.id
_producto =
p_prod AND
Q_movimiento.id
_cliente =
p_cuenta AND
Q_movimiento.f_
movto >=
x_f_aux AND
Q_Movimiento.f_
movto <
p_f_inicial
USE-
INDEX cuenta
NO-LOCK NO-
ERROR.
END.
DO WHILE
AVAILABLE
Q_Movimiento:
FIND
Q_tpmovto WHERE
Q_tpmovto.id_mo
vto =
Q_movimiento.id
_movto
USE-INDEX
tpmovto NO-
LOCK.
IF
Q_Tpmovto.afec_
saldo_total
THEN
DO:
IF
Q_Tpmovto.Natur
aleza = "C"
THEN p_SalFin =
p_SalFin +
Q_movimiento.Mo
nto.
ELSE p_SalFin =
p_SalFin -
Q_movimiento.Mo
nto.
END.
IF YEAR(
Q_Movimiento.f_
movto ) = YEAR(
p_f_final ) AND
Q_Movimiento.id
_movto =
Q_Parametro.id_
int_pasivo
THEN
P_interes_anual
=
P_interes_Anual
+
Q_Movimiento.mo
vto.
FIND NEXT
Q_Movimiento
WHERE
Q_movimiento.id
_producto =
p_prod AND
Q_Movimiento.id
_cliente =
p_cuenta AND
Q_Movimiento.f_
movto <
p_f_inicial
USE-
INDEX cuenta
NO-LOCK NO-
ERROR.
END.
P_SalIni
= P_SalFin.
x_f_aux
= p_f_inicial -
1.
x_Suma
= 0.
P_n_che_rebot
= 0.
P_n_che_emit
= 0.
P_Tot_Cheques
= 0.
P_n_Debits
= 0.
P_Tot_Debits
= 0.00.
P_n_Credits
= 0.
P_Tot_Credits
= 0.00.
P_Interes
= 0.00.
FOR EACH
Q_Movimiento
WHERE
Q_movimiento.id
_producto =
p_prod AND
Q_Movimiento.id
_cliente =
p_cuenta AND
Q_Movimiento.F_
Movto >=
p_f_inicial AND
Q_Movimiento.F_
Movto <=
p_f_Final
USE-INDEX
cuenta NO-LOCK
BREAK BY
Q_Movimiento.F_
Movto.
FIND
Q_Tpmovto WHERE
Q_Tpmovto.id_mo
vto =
Q_Movimiento.id
_movto
USE-INDEX
Tpmovto NO-
LOCK.
IF FIRST-
OF(Q_Movimiento
.F_Movto) THEN
DO:
x_Suma
= x_Suma +
( (
Q_Movimiento.f_
movto - x_f_aux
- 1 ) *
p_SalFin ).
x_f_aux =
Q_Movimiento.f_
movto.
END.
x_cheque =
No.
buscaChe:
DO i = 1 TO
5:
IF
Q_Movimiento.id
_movto =
Q_Parametro.id_
cheques[i] THEN
DO:
p_n_che_emit
= p_n_che_emit
+ 1.
P_Tot_Cheques
= P_Tot_Cheques
+
Q_Movimiento.Mo
vto.
x_cheque = Yes.
LEAVE buscaChe.
END.
IF
Q_Tpmovto.afec_
saldo_total
THEN
DO:
IF
Q_Tpmovto.Natur
aleza = "C"
THEN
DO:
p_salFin
= p_salFin +
Q_movimiento.Mo
nto.
p_Tot_Credits
= p_Tot_Credits
+
Q_Movimiento.Mo
nto.
END.
ELSE
DO:
p_salFin =
p_salFin -
Q_Movimiento.Mo
nto.
IF NOT x_cheque
THEN
DO:
p_Tot_Debits =
p_Tot_Debits +
Q_Movimiento.Mo
nto.
P_n_Debits =
P_n_Debits + 1.
END.
END.
IF
Q_Movimiento.id
_movto =
Q_Parametro.id_
Ret_Check
THEN
p_n_che_rebot =
p_n_che_rebot +
1.
IF
Q_Movimiento.id
_movto =
Q_Parametro.id_
int_Pasivo THEN
DO:
p_Interes =
P_Interes +
Q_Movimiento.Mo
nto.
IF
YEAR(
Q_Movimiento.f_
movto ) = YEAR(
P_f_final )
THEN
P_Interes_Anual
=

```

```

P_Interes_Anual  TESIS
+                */
Q_Movimiento.mo /*      Fecha
nto.             : 01/31/2000
                */
                END.
                /******
                *****
                *****
                *****
                *****
                **/
                DEFINE SHARED
                VARIABLE P_prod
                LIKE
                U_cliente.id_pr
                oducto.
                DEFINE SHARED
                VARIABLE P_Cuenta
                LIKE
                U_cliente.id_cl
                iente.
                DEFINE SHARED
                VARIABLE
                P_F_inicial
                AS DATE FORMAT
                "99/99/9999".
                DEFINE SHARED
                VARIABLE
                P_F_final
                AS DATE FORMAT
                "99/99/9999".
                DEFINE SHARED
                VARIABLE
                p_SalIni
                LIKE
                Q_Corte.Saldo_I
                nicial INITIAL
                0.
                DEFINE SHARED
                VARIABLE
                P_SalFin
                LIKE
                Q_Corte.Saldo_F
                inal INITIAL
                0.
                DEFINE SHARED
                VARIABLE
                P_SalPro
                LIKE
                Q_Corte.Saldo_P
                romedio INITIAL
                0.
                DEFINE SHARED
                VARIABLE
                P_n_che_rebot
                LIKE
                Q_Corte.n_chequ
                es_rebotados
                INITIAL 0.
                DEFINE SHARED
                VARIABLE
                P_n_che_emit
                LIKE
                Q_Corte.n_chequ
                es_emitidos
                INITIAL 0.
                DEFINE SHARED
                VARIABLE
                P_n_Debits
                LIKE
                Q_Corte.n_Debit
                s INITIAL 0.
                DEFINE SHARED
                VARIABLE
                P_Tot_Debits
                LIKE
                Q_Corte.Tot_Deb
                its INITIAL 0.
                DEFINE SHARED
                VARIABLE
                P_n_Credits
                LIKE
                Q_Corte.n_Credi
                ts INITIAL 0.
                DEFINE SHARED
                VARIABLE
                P_Tot_Credits
                LIKE
                Q_Corte.Tot_Cre
                dits INITIAL 0.
                DEFINE SHARED
                VARIABLE
                P_Interes
                LIKE
                Q_Corte.Interes
                INITIAL 0.
                DEFINE SHARED
                VARIABLE
                P_Tasa_Int
                LIKE
                Q_Corte.Tasa_In
                teres INITIAL
                0.
                DEFINE VARIABLE
                x_mes
                AS INTEGER NO-
                UNDO.
                DEFINE VARIABLE
                x_anio
                AS INTEGER NO-
                UNDO.
                DEFINE VARIABLE
                x_anio_mes
                AS INTEGER NO-
                UNDO.
                DEFINE VARIABLE
                x_f_aux
                AS DATE
                NO-
                UNDO.
                DEFINE VARIABLE
                x_encuentra
                AS LOGICAL
                NO-
                UNDO.
                DEFINE VARIABLE
                x_Suma
                LIKE
                Q_Corte.Saldo_P
                romedio INITIAL
                0.
                DEFINE VARIABLE
                i
                AS INTEGER.
                DEFINE VARIABLE
                x_cheque
                AS LOGICAL.
                FIND U_cliente
                WHERE
                U_cliente.id_pr
                oducto = p_prod
                AND
                U_cliente.id_cl
                iente =
                p_cuenta NO-
                LOCK.
                /* Verifica si
                las fechas es
                un periodo
                completo */
                x_f_aux =
                p_f_inicial +
                32.
                x_f_aux =
                x_f_aux -
                DAY(x_f_aux).
                P_Tasa_Int =
                0.
                IF
                YEAR(p_f_inicia
                l) =
                YEAR(p_f_final)
                AND
                MONTH(p_f_inici
                al) =
                MONTH(p_f_final.
                ) AND
                DAY(p_f_inicial
                ) = 1 AND
                DAY(p_f_final)
                = DAY( x_f_aux
                ) THEN
                DO:
                x_anio_mes =
                YEAR(p_f_inicia
                l) * 100 +
                MONTH(p_f_inici
                al).
                FIND
                Q_Corte
                WHERE
                Q_corte.id_prod
                ucto = p_prod
                AND
                Q_Corte.id_clie
                nte = p_cuenta
                AND
                Q_Corte.anio_me
                s_corte =
                x_anio_mes
                USE-
                INDEX corte NO-
                LOCK NO-ERROR.
                IF
                AVAILABLE
                Q_Corte THEN
                DO:
                P_SalIni
                =
                Q_Corte.Saldo_I
                nicial.
                P_SalFin
                =
                Q_Corte.Saldo_F
                inal.
                P_SalPro
                =
                Q_Corte.Saldo_P
                romedio.
                P_n_che_rebot
                =
                Q_Corte.n_chequ
                es_rebotados.
                P_n_che_emit
                =
                Q_Corte.n_chequ
                es_emitidos.
                P_Tot_Cheques
                =
                Q_Corte.Tot_Che
                _Emitidos.
                P_n_Debits
                =
                Q_Corte.n_Debit
                s.
                P_Tot_Debits
                =
                Q_Corte.Tot_Debit
                s.
                END.
                IF LAST-
                OF(Q_Movimiento
                .F_Movto) THEN
                x_Suma =
                x_Suma +
                p_SalFin.
                END.
                x_Suma =
                x_Suma + ( (
                p_f_Final -
                x_f_aux ) *
                p_SalFin ).
                IF
                U_cliente.f_alt
                a > p_f_inicial
                AND
                U_cliente.f_alt
                a <= p_f_final
                THEN
                p_salpro =
                x_Suma / (
                p_f_final -
                U_cliente.f_alt
                a + 1 ).
                ELSE
                p_salpro =
                x_Suma / (
                p_f_final -
                p_f_inicial + 1
                ).
                RETURN.
                /******
                *****
                *****
                *****
                *****
                **/
                /* Programa:
                QFESTGLO.P
                */
                /* Objetivo
                : Rutina para
                c lculo de
                Estadisticas en
                Cta Global.
                */
                /* Autor
                : Equipo de
                Trabajo de
    
```

```

Q_Corte.Tot_Deb     .   x_mes =          s_corte / 100 )
its.                MONTH(p_f_inici    , 0 ) + 1.
                    al) - 1.          END.
P_n_Credits         =   x_anio =          ELSE
Q_Corte.n_Credi    YEAR(p_f_inicia    DO:
ts.                 1).              x_mes
                    END.            = (
P_Tot_Credits      x_anio_mes = (      Q_corte.anio_me
=                   x_anio * 100 )   s_corte MODULO
Q_Corte.Tot_Cre    + x_mes.         100 ) + 1.
dits.              FIND LAST        x_anio
                    Q_Corte         = TRUNCATE (
P_Interes          Q_Corte         Q_Corte.anio_me
=                   WHERE           s_corte / 100
Q_Corte.Interes   Q_corte.id_prod        , 0 ).
                    ucto = p_prod    END.
P_Tasa_Int         AND              x_f_aux =
=                   Q_Corte.id_clie  DATE( x_mes , 1
Q_Corte.Tasa_In   nte = p_cuenta        , x_anio ).
teres.             AND              FIND FIRST
                    Q_Corte:anio_me  Q_Movimiento
RETURN.            s_corte <=      WHERE
END.                x_anio_mes      Q_movimiento.f_
                    USE-INDEX       movto >=
                    corte NO-LOCK    x_f_aux AND
                    NO-ERROR.       Q_Movimiento.f_
NO-LOCK NO-       IF NOT              USE-INDEX
ERROR.            AVAILABLE          edo_global NO-
                    IF              LOCK NO-ERROR.
AVAILABLE         Q_corte THEN      END.
Q_Tasa           DO:
                    p_SalFin =
THEN              0.
                    FIND FIRST
IF                Q_movimiento    DO WHILE
U_cliente.Tipo_  WHERE           AVAILABLE
Persona = 4 /*   Q_Movimiento.f_      Q_Movimiento:
fisica */        movto <
THEN P_Tasa_Int  p_f_inicial
= Tasa_Fisica.  USE-INDEX
ELSE P_Tasa_Int  edo_global NO-
= Tasa_Moral.   LOCK NO-ERROR:
END.             END.
IF              ELSE
MONTH(p_f_inici  DO:
al) = 1 THEN    p_SalFin =
DO:             Q_Corte.Saldo_F
x_mes =        inal.
12.            IF (
x_anio =       Q_corte.anio_me
YEAR(p_f_inicia s_corte MODULO
1) - 1.        100 ) = 12 THEN
END.           DO:
ELSE           x_mes
DO:           = 1.
              x_anio
              = TRUNCATE ((
              Q_Corte.anio_me

```

```

IF
Q_Tpmovto.afec_
cta_global THEN
DO:
IF
Q_Tpmovto.Natur
aleza = "C"
THEN
DO:
p_salFin'
= p_salFin +
Q_movimiento.Mo
nto.
p_Tot_Credits
= p_Tot_Credits
+
Q_Movimiento.Mo
nto.
P_n_Credits
= P_n_Credits +
1.
END.
ELSE
DO:
p_salFin =
p_salFin -
Q_Movimiento.Mo
nto.
IF NOT x_cheque
THEN
DO:
p_Tot_Debits =
p_Tot_Debits +
Q_Movimiento.Mo
nto.
P_n_Debits =
P_n_Debits + 1.
END.
END.
IF
Q_Movimiento.id
_movto =
Q_Parametro.id_
Ret_Check
THEN
p_n_che_rebot =
p_n_che_rebot +
1.

```

```

IF
Q_Movimiento.id
_movto =
Q_Parametro.id_
int_Pasivo
THEN
p_Interes =
P_Interes +
Q_Movimiento.Mo
nto.
IF LAST-
OF(Q_Movimiento
.F_Movto) THEN.
x_Suma =
x_Suma +
p_SalFin.
END.
x_Suma =
x_Suma + (-((
p_f_Final -
x_f_aux ) *
p_SalFin -
p_salpro =
x_Suma / (
p_f_final -
p_f_inicial + 1
).
RETURN.
/*****
*****
*****
*****
*/
Programa
: QFGETCNO.P
*/
Objetivo
: Obtener el
siguiente
consecutivo
p/file
q_movimiento */
Autor
: Equipo de
Trabajo de
TESIS
*/
Fecha
: 01/31/2000
*/
Parametros
:
inp x_fecha

```

```

fecha del
movimiento
*/
out
x_consecutivo
numero
crónológico de
captura */
/*****
*****
*****
*****
*/
DEFINE INPUT
PARAMETER
x_fecha
AS DATE FORMAT
"99/99/9999"
DEFINE OUTPUT
PARAMETER
x_consecutivo
AS INTEGER.
FIND LAST
Q_Movimiento
WHERE
Q_Movimiento.f
movto = x_fecha
USE-INDEX
edo_global NO-
LOCK NO-ERROR.
IF AVAILABLE
Q_Movimiento.
THEN
x_consecutivo =
Q_Movimiento.Cr
onológico + 1.
ELSE
x_consecutivo =
1.
RETURN.
/*****
*****
*****
*****
*/
Programa
: QFPOSITO.P
*/
Objetivo
: Posiciona el
cursor en la HP
Laser Jet
*/

```

```

/*
Autor
: Equipo de
Trabajo de
TESIS
*/
Fecha
: 01/31/2000
*/
Parametros
:
x_ULRow
Upper Left Row
*/
x_ULCol
Upper Left
Column
*/
/*****
*****
*****
*****
*/
DEFINE INPUT
PARAMETER
x_ULRow AS
INTEGER.
DEFINE INPUT
PARAMETER
x_ULCol AS
INTEGER.
DEFINE VARIABLE
x_str AS
CHARACTER
FORMAT
"X(100)".
DEFINE SHARED
STREAM impre.
/*
posiciona
el cursor en el
lugar
especificado */
x_str = "&a" +
TRIM( STRING(
x_ULRow ,
">>>>>>>" ) )
+ "v".
x_str = x_str +
"&a" + TRIM(
STRING( x_ULCol
, ">>>>>>>" )
) + "H".
PUT STREAM
impre CONTROL
x_str.

```

```

RETURN.
/*****
*****
*****
*****
*/
Programa
: QFPOSITO.P
*/
Objetivo
: Dibuja una
raya en HP
laser Jet
*/
Autor
: Equipo de
Trabajo de
TESIS
*/
Fecha
: 01/31/2000
*/
Parametros
:
x_ULRow
Upper Left Row
*/
x_ULCol
Upper Left
Column
*/
Direc
Direccion 0
Horizontal
*/
1 Vertical
*/
x_Lon
Longitud de la
raya
*/
/*****
*****
*****
*****
*/
DEFINE INPUT
PARAMETER
x_ULRow AS
INTEGER.
DEFINE INPUT
PARAMETER
x_ULCol AS
INTEGER.

```



```

was created AS INTEGER
with the FORMAT
Progress UIB. "99999999"
*/ DEF VAR
/*----- Po_clase
----- AS INTEGER
----- FORMAT "9"
----- INITIAL 1.
-----*/ DEF VAR
/* Create an AS INTEGER
unnamed pool AS INTEGER
to store all the FORMAT "999"
widgets created INITIAL 0.
by this DEF VAR
procedure. This Po_concepto
is a good AS CHARACTER
default which FORMAT "X(30)"
assures DEF VAR
that this Po_fechafin
procedure's AS DATE
triggers and FORMAT
internal "99/99/9999"
DEF VAR reg_mov
procedures AS CHARACTER
will AS CHARACTER
execute in this FORMAT "X(1)"
procedure's INITIAL "M".
storage, and DEF VAR
that proper cta_carg
cleanup AS CHARACTER
will occur on FORMAT
deletion of the "9999999999999999"
procedure. */
CREATE-WIDGET-
POOL.
DEF VAR cta_abo
AS CHARACTER
FORMAT
"9999999999999999"
DEF VAR cta_abo
AS CHARACTER
FORMAT
"9999999999999999"
DEF VAR Mo_ref
AS INTEGER
FORMAT
"99999999"
DEF VAR
Mo_cargo
AS INTEGER
FORMAT "9"
INITIAL 1.
DEF VAR
Mo_abono
AS INTEGER
FORMAT "9"
INITIAL 2.
DEF VAR
Mo_diarioesp
AS INTEGER
FORMAT "999"
INITIAL 0.
DEF VAR
reg_descmov
AS CHARACTER
FORMAT "X(1)"
INITIAL "D".
DEF VAR
desc_mov
AS CHARACTER
FORMAT "X(28)"
DEF VAR
Tot_fisica
AS DECIMAL
FORMAT "-
999999999999.99"
DEF VAR
Tot_moral
AS DECIMAL
FORMAT "-
999999999999.99"
DEF VAR fis
AS CHARACTER
FORMAT "99"
INITIAL "04".
DEF VAR mor
AS CHARACTER
FORMAT "99"
INITIAL "03".
DEF VAR
cta_carg_fisica
AS CHARACTER
FORMAT
"9999999999999999"
DEF VAR
cta_carg_moral
AS CHARACTER
FORMAT
"9999999999999999"
DEF VAR
cta_abo_fisica
AS CHARACTER
FORMAT
"9999999999999999"
DEF VAR
cta_abo_moral
AS CHARACTER
FORMAT
"9999999999999999"
DEF VAR
n_movtos
AS INTEGER
FORMAT ">>>>9"
DEF VAR dato
AS CHARACTER
FORMAT "X(08)"
DEF VAR archivo
AS CHARACTER
FORMAT "X(40)"
DEF VAR t_flag
AS LOGICAL
NO-UNDO.
DEF STREAM
report.
/* Parameters
Definitions ---
*/
/* Local
Variable
Definitions ---
*/
/* _UIB-
PREPROCESSOR-
BLOCK-END */
/* _UIB-CODE-
BLOCK-END */
&ANALYZE-RESUME
&ANALYZE-
SUSPEND _UIB-
PREPROCESSOR-
BLOCK
/*
*****
*****
Control
Definitions
***** */
Preprocessor
Definitions
***** */
&Scoped-define
PROCEDURE-TYPE
Window
/* Name of
first Frame
and/or Browse
and/or first
Query
*/
&Scoped-define
FRAME-NAME
DEFAULT-FRAME
/* Standard
List
Definitions
*/
&Scoped-Define
ENABLED-OBJECTS
RECT-14 RECT-4
RECT-5 RECT-13
t_contab -
t_fecha
t_poliza
t_fecha_descon
&Scoped-Define
DISPLAYED-
OBJECTS
t_contab
t_fecha
t_poliza
t_fecha_descon
/* Custom List
Definitions.
*/
/* List-1,List-
2,List-3,List-
4,List-5,List-6
*/
/* _UIB-
PREPROCESSOR-
BLOCK-END */
&ANALYZE-RESUME
/*
*****
*****
Control
Definitions
***** */
/* Define the
widget handle
for the window.
*/
DEFINE VAR C-
Win AS WIDGET-
HANDLE NO-UNDO.
/* Definitions
of the field
level widgets
*/
DEFINE BUTTON
BUTTON-2 AUTO-
GO
IMAGE-UP
FILE
"adeicon\run":U
LABEL
"Button 2"
SIZE 8, BY
2.1.
DEFINE BUTTON
BUTTON-3 AUTO-
END-KEY
IMAGE-UP
FILE
"adeicon\exit-
au":U
LABEL
"Button 3"

```

2.1.	SIZE 8 BY GRAPHIC-EDGE NO-FILL	t_poliza AT ROW 8.62 COL 33 COLON-	RECT-5 AT ROW 14.14 COL 7	HIDDEN = YES
DEFINE VARIABLE t_contab AS LOGICAL FORMAT "Contabilidad/D escontabilidad" :U INITIAL NO LABEL "Contabilizaci6 n (Contabilidad/D escontabilidad) "	4.48. SIZE 43 BY DEFINE RECTANGLE RECT- 14 EDGE- PIXELS 2 GRAPHIC-EDGE NO-FILL	"El valor debe ser mayor que cero" AUTO-RETURN	RECT-13 AT ROW 6.24 COL 11 WITH 1 DOWN NO-BOX KEEP- TAB-ORDER OVERLAY	TITLE = "Sistema de Chequera en D6lares" HEIGHT = 16.91 WIDTH = 113 MAX- HEIGHT = 27.95 MAX- WIDTH = 159.6
VIEW-AS FILL-IN SIZE 19 BY 1 NO-UNDO.	4.48. SIZE 48 BY DEFINE RECTANGLE RECT- 4 EDGE- PIXELS 2 GRAPHIC-EDGE NO-FILL	"Fecha a Descontabilizar " AUTO-RETURN BUTTON-2 AT ROW 14.33 COL 88 HELP	112.4 BY 16.84. /* ***** ***** Procedure Settings ***** ***** */	VIRTUAL-HEIGHT = 27.95 VIRTUAL-WIDTH = 159.6 RESIZE = yes SCROLL-BARS = no
DEFINE VARIABLE t_fecha AS DATE FORMAT "99/99/9999":U LABEL "Fecha a Contabilizar" VIEW-AS FILL-IN SIZE 16 BY 1 NO-UNDO.	BY 11.52. DEFINE RECTANGLE RECT- 5 EDGE- PIXELS 2 GRAPHIC-EDGE NO-FILL	"Salir de este Proceso" "Descontabiliza r" VIEW-AS TEXT SIZE 15 BY .67 AT ROW 6 COL 75	/* Settings for THIS-PROCEDURE Type: Window Allow: Basic,Browse,DB Fields,Window,Q uery Other Settings: COMPILE */ &ANALYZE-RESUME _END-PROCEDURE- SETTINGS	STATUS-AREA = yes BGCOLOR = ? FGCOLOR = ? KEEP- FRAME-Z-ORDER = yes THREE- D = yes MESSAGE-AREA = no
DEFINE VARIABLE t_fecha_descon AS DATE FORMAT "99/99/9999":U LABEL "Fecha para Descontabilizar " VIEW-AS FILL-IN SIZE 16 BY 1 NO-UNDO.	BY 2.62. /* ***** ***** Frame Definitions ***** ***** */ DEFINE FRAME DEFAULT-FRAME t_contab AT ROW 4.1 COL 54 COLON-	BGCOLOR 9 FGCOLOR 14 "Contabilizar" VIEW-AS TEXT SIZE 12 BY .67 AT ROW 6 COL 30	/* ***** ***** Create Window ***** ***** */ &ANALYZE- SUSPEND _CREATE-WINDOW IF SESSION:DISPLAY -TYPE = "GUI":U THEN CREATE WINDOW C-win ASSIGN	SENSITIVE = yes. ELSE {&WINDOW- NAME} = CURRENT-WINDOW. /* END WINDOW DEFINITION */ &ANALYZE-RESUME
DEFINE VARIABLE t_poliza AS INTEGER FORMAT ">, >>9":U INITIAL 0 LABEL "N6mero de Poliza" VIEW-AS FILL-IN SIZE 16 BY 1 NO-UNDO.	ALIGNED HELP "Teclee Contabilizar 6 Descontabilizar " AUTO-RETURN t_fecha AT ROW 7.33 COL 33 COLON-ALIGNED HELP	BGCOLOR 9 FGCOLOR 14 RECT-14 AT ROW 6.24 COL 56 " Contabilidad / Descontabilidad " VIEW-AS TEXT SIZE 34 BY 1.05 AT ROW 1.71 COL 40	/* ***** ***** Create Window ***** ***** */ &ANALYZE- SUSPEND _CREATE-WINDOW IF SESSION:DISPLAY -TYPE = "GUI":U THEN CREATE WINDOW C-win ASSIGN	Runtime Attributes and UIB Settings
DEFINE RECTANGLE RECT- 13 EDGE- PIXELS 2	"Teclee la Fecha" AUTO- RETURN	BGCOLOR 9 FGCOLOR 14 RECT-4 AT ROW 2.33 COL 7		


```

*****
*/
&ANALYZE-
SUSPEND _RUN-
TIME-ATTRIBUTES
/* SETTINGS FOR
WINDOW C-Win
VISIBLE, RUN-
PERSISTENT.
*/
/* SETTINGS FOR
FRAME DEFAULT-
FRAME
*/
ASSIGN
t_contab:AUTO-
RESIZE IN FRAME
DEFAULT-FRAME
= TRUE;
ASSIGN
t_fecha:AUTO-
RESIZE IN FRAME
DEFAULT-FRAME
= TRUE.
ASSIGN
t_fecha_descon:
AUTO-RESIZE IN
FRAME DEFAULT-
FRAME
= TRUE.
ASSIGN
t_poliza:AUTO-
RESIZE IN FRAME
DEFAULT-FRAME
= TRUE.
IF
SESSION:DISPLAY
-TYPE = "GUI":U
AND VALID-
HANDLE(C-Win).
THEN C-
Win:HIDDEN. =
no.
/* _RUN-TIME-
ATTRIBUTES-END
*/
&ANALYZE-RESUME
terminate the
procedure.
/*
APPLY:
"CLOSE":U
THIS-PROCEDURE.
RETURN:NO-
APPLY.
END.
/* _UIB-CODE-
BLOCK-END */
&ANALYZE-RESUME
&Scoped-define
SELF-NAME C-Win
&ANALYZE-
SUSPEND _UIB-
CODE-BLOCK
CONTROL C-Win
C-Win
ON END-ERROR OF
C-Win /*
Sistema de
Chequera en
Dólares */
OR ENDKEY OF
{&WINDOW-NAME}
ANYWHERE DO:
/* This case
occurs when the
user presses
the "Esc" key.
In a
persistently
run window,
just ignore
this. If we
did not, the
application
would exit.*/
IF THIS-
PROCEDURE:PERSI
STENT THEN
RETURN NO-
APPLY.
END.
/* _UIB-CODE-
BLOCK-END */
&ANALYZE-RESUME
&ANALYZE-
SUSPEND _UIB-
CODE-BLOCK
CONTROL C-Win
C-Win
ON WINDOW-CLOSE
OF C-Win /*
Sistema de
Chequera en
Dólares */
DO:
/* This event
will close the
window and
INDEX contab.
FIND
FIRST Q_Tpmovto
WHERE
Q_Tpmovto.id_mo
vto =
Q_Movimiento.id
_movto
USE-INDEX
tpmovto.
IF
Q_tpmovto.cuent
a_cargar <> ""
AND
Q_tpmovto:cuent
a_abonar <> ""
AND
NOT
Q_Movimiento.co
ntabilizado.
THEN
n_movtos =
n_movtos + 1.
END.
IF
n_movtos = 0
THEN DO:
MESSAGE
"Los
movimientos a
esta fecha ya
fueron
contabilizados"
RETURN.
NO-APPLY.
* **END.
DISPLAY
"Generando el
archivo.
Espere un
momento..."
WITH
COL 3
ROW
12 OVERLAY
THREE-D
COLOR
WHITE/BLUE
NO-
LABELS.
dato =
string(Po_fecha
ini).
archivo =
"c:\jpol\txt\p"
+
SUBSTRING(dato,
1,2) +
SUBSTRING(dato,
4,2), +
SUBSTRING(dato,
7,2) + ".txt".
fecha =
INTEGER(SUBSTRIN
G(dato,4,2) +
SUBSTRING(dato,
1,2) +
SUBSTRING(dato,
7,2)):
/* DOS
SILENT
VALUE("DEL-" +
archivo).
OUTPUT ;
STREAM report
TO
VALUE(archivo).
Po_concepto =
"DYNAMIC
ACCOUNT.ENTRIES
";
DISPLAY
STREAM report
reg_pol
fecha
Po_tpol
Po_poliza
Po_clase
Po_diarioagr
Po_concepto
WITH FRAME
A NO-LABEL NO-
BOX
WIDTH
64 STREAM-IO.
Tot_fisica
= 0.

```

<pre> Tot_Moral = 0. FOR EACH Q_Movimiento WHERE Q_Movimiento.f_ movto = Po_fechaini AND NOT Q_movimiento.co ntabilizado USE- INDEX contab , EACH Q_Tpmovto WHERE Q_Tpmovto.id_mo vto = Q_Movimiento.id _movto AND Q_tpmovto.cuent a_cargar <> "" AND Q_tpmovto.cuent a_abonar <> "" USE- INDEX tpmovto ; EACH U_cliente WHERE U_cliente.id_pr oducto = Q_Movimiento.id _producto AND U_cliente.id_cl iente = Q_Movimiento.id _cliente USE- INDEX cliente BREAK BY Q_movimiento.id _movto: IF U_cliente.tipo_ persona = 4 THEN /* Persona fisica */ Tot_fisica = Tot_fisica + Q_Movimiento.mo vto. ELSE Tot_moral = Tot_moral + </pre>	<pre> Q_Movimiento.mo vto. IF LAST- OF(Q_Movimiento .id_movto) THEN DO: cta_carg = "". cta_abo = "". IF SUBSTRING(Q_tpm ovto.cuenta_car gar,7,2) = "XX" THEN DO: cta_carg_fisica = SUBSTRING(cuent a_cargar,1,6) + "04" + SUBSTRING(cuent a_cargar,9,2) . cta_carg_moral = SUBSTRING(cuent a_cargar,1,6) + "03" + SUBSTRING(cuent a_cargar,9,2) . END. ELSE DO: cta_carg_fisica = SUBSTRING(cuent a_cargar,1,6) + "04" + SUBSTRING(cuent a_cargar,9,2) . cta_carg_moral = SUBSTRING(cuent a_cargar,1,6) + "03" + SUBSTRING(cuent a_cargar,9,2) . END. IF SUBSTRING(Q_tpm ovto.cuenta_abo nar,7,2) = "XX" THEN DO: cta_abo_fisica = SUBSTRING(cuent a_abonar,1,6) + </pre>	<pre> "04" + SUBSTRING(cuent a_abonar,9,2) . cta_abo_moral = SUBSTRING(cuent a_abonar,1,6) + "03" + SUBSTRING(cuent a_abonar,9,2) . END. ELSE DO: cta_abo_fisica = cuenta_abonar. cta_abo_moral = cuenta_abonar. END. desc_mov = Q_tpmovto.Descr ip_larga. IF Tot_fisica > 0 THEN DO: DISPLAY STREAM report reg_mov cta_carg_fisica Mo_ref Mo_cargo Tot_fisica Mo_diarioesp WITH FRAME B NO-LABEL NO-BOX WIDTH 49 STREAM-IO. DISPLAY STREAM report reg_descmov desc_mov WITH FRAME B1 NO-LABEL NO-BOX WIDTH 32 STREAM-IO. Tot_fisica = 0. END. IF Tot_moral > 0 THEN DO: DISPLAY STREAM report reg_descmov desc_mov WITH FRAME B1 NO-LABEL NO-BOX WIDTH 49 STREAM-IO. Mo_ref Mo_cargo Tot_moral Mo_diarioesp </pre>	<pre> WITH FRAME B1 NO-LABEL NO-BOX WIDTH 32 STREAM-IO. DISPLAY STREAM report reg_mov cta_abo_fisica Mo_ref Mo_abono Tot_fisica Mo_diarioesp WITH FRAME C NO-LABEL NO-BOX WIDTH 49 STREAM-IO. DISPLAY STREAM report reg_descmov desc_mov WITH FRAME B1 NO-LABEL NO-BOX WIDTH 32 STREAM-IO. DISPLAY STREAM report reg_descmov desc_mov WITH FRAME E NO-LABEL NO-BOX WIDTH 49 STREAM-IO. DISPLAY STREAM report reg_descmov desc_mov WITH FRAME B1 NO-LABEL NO-BOX WIDTH 32 STREAM-IO. Mo_ref Mo_cargo Tot_moral Mo_diarioesp </pre>	<pre> WITH FRAME D NO-LABEL NO-BOX WIDTH 49 STREAM-IO. DISPLAY STREAM report reg_descmov desc_mov WITH FRAME B1 NO-LABEL NO-BOX WIDTH 32 STREAM-IO. IF cta_abo = "" then cta_abo = cta_abo_moral. DISPLAY STREAM report reg_mov cta_abo_moral Mo_ref Mo_abono Tot_moral Mo_diarioesp WITH FRAME E NO-LABEL NO-BOX WIDTH 49 STREAM-IO. DISPLAY STREAM report reg_descmov desc_mov WITH FRAME B1 NO-LABEL NO-BOX WIDTH 32 STREAM-IO. Tot_moral = 0. END. END. /*----- </pre>
--	---	---	---	--

```

----- FIN
DEL LAST-OF -----
-----*/
        ASSIGN
Q_movimiento.co
ntabilizado =
YES.
        t_flag
= TRUE.
        END.
        END. /* IF
t_contab */:
        ELSE DO:
                DISPLAY
"Actualizando
registros.
Espere un
momento"
WITH THREE-D 3
COL
ROW 12 COLOR.
WHITE/BLUE NO-
LABELS.
        FOR EACH
Q_Movimiento
WHERE
Q_Movimiento.f_
movto =
Po_fechaini
        USE-
INDEX contab.
ASSIGN
Q_Movimiento.co
ntabilizado =
NO.
        t_flag =
TRUE.
        END.
        OUTPUT STREAM
report CLOSE.
        IF t_flag
THEN
        MESSAGE
"FINALIZO
PROCESO".
        ELSE
        MESSAGE
"NO SE GENERO
EL PROCESO".
        APPLY 'CLOSE'
TO THIS-
PROCEDURE.
        END.
        /* _UIB-CODE-
BLOCK-END */
&ANALYZE-RESUME
        &Scoped-define
SELF-NAME
t_contab
&ANALYZE-
SUSPEND _UIB-
CODE-BLOCK
_CONTROL
t_contab C-Win
ON LEAVE OF
t_contab IN
FRAME DEFAULT-
FRAME /*
Contabilización
(Contabilidad/D
escontabilidad)
*/
DO:
        t_contab =
INPUT t_contab.
        END.
        /* _UIB-CODE-
BLOCK-END */
&ANALYZE-RESUME
        &Scoped-define
SELF-NAME
t_fecha_descon
&ANALYZE-
SUSPEND _UIB-
CODE-BLOCK
_CONTROL
t_fecha_descon
C-Win
ON LEAVE OF
t_fecha_descon
IN FRAME
DEFAULT-FRAME
/* Fecha para
Descontabilizar
*/
DO:
        t_fecha_descon
= INPUT
t_fecha_descon.
Po_fechaini
=
t_fecha_descon.
        IF t_contab =
TRUE THEN DO:
                IF
t_fecha_descon
<> ? THEN DO:
                        MESSAGE
"Se debe
teclear Fecha y
No. Poliza".
                        RETURN
NO-APPLY.
                END.
                FIND FIRST
Q_movimiento
WHERE
Q_Movimiento.f_
movto =
Po_fechaini
        USE-
INDEX contab.
NO-ERROR.
        IF NOT
AVAILABLE
Q_movimiento.
THEN DO:
                MESSAGE
"No Existen
Movimientos en
esa fecha para
ser
contabilizados"
                RETURN
NO-APPLY.
                END.
                END.
                /* _UIB-CODE-
BLOCK-END */
&ANALYZE-RESUME
        &Scoped-define
SELF-NAME
t_fecha_descon
&ANALYZE-
SUSPEND _UIB-
CODE-BLOCK
_CONTROL
t_fecha_descon
C-Win
ON LEAVE OF
t_fecha_descon
IN FRAME
DEFAULT-FRAME
/* Fecha para
Descontabilizar
*/
DO:
        t_fecha_descon
= INPUT
t_fecha_descon.
Po_fechaini
=
t_fecha_descon.
        IF t_contab =
TRUE THEN DO:
                IF
t_fecha_descon
<> ? THEN DO:
                        MESSAGE
"Se debe
teclear fecha
de
Descontabilizac
ion".
                        RETURN
NO-APPLY.
                END.
                RETURN
NO-APPLY.
                END.
        WHERE
Q_Movimiento.f_
movto =
Po_fechaini
        USE-
INDEX contab.
NO-ERROR.
        IF t_poliza =
0 THEN DO:
                MESSAGE
"Debe ser Mayor
que cero".
                RETURN NO-
APPLY.
                END.
                /* _UIB-CODE-
BLOCK-END */
&ANALYZE-RESUME
        &UNDEFINE SELF-
NAME
        &ANALYZE-
SUSPEND _UIB-
CODE-BLOCK
_CUSTOM_MAIN-
BLOCK C-Win
        /*
*****
*****
Main Block
*****
***** */
        /* Set CURRENT-
WINDOW: this
will parent
dialog-boxes
and frames.
*/
ASSIGN CURRENT-
WINDOW
= {&WINDOW-
NAME}
        THIS-
PROCEDURE: CURRE
NT-WINDOW =
{&WINDOW-NAME}.
        /* The CLOSE
event can be
used from
inside or
outside the
procedure to
*/
/* terminate
it.
*/
ON CLOSE OF
THIS-PROCEDURE
RUN
disable_UI.
    
```

<pre> /* Best default for GUI applications is... */ PAUSE 0 BEFORE- HIDE. /* Now enable the interface and wait for the exit condition. */ /* (NOTE: handle ERROR and END-KEY so cleanup code will always fire. */ MAIN-BLOCK: DO ON ERROR UNDO MAIN- BLOCK; LEAVE MAIN-BLOCK ON END-KEY UNDO MAIN- BLOCK; LEAVE MAIN-BLOCK: RUN enable_UI. IF NOT THIS- PROCEDURE:PERSI STENT THEN WAIT-FOR CLOSE OF THIS- PROCEDURE. END. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME /* ***** ***** Internal Procedures ***** ***** */ &ANALYZE- SUSPEND _UIB- CODE-BLOCK _PROCEDURE disable_UI C- Win_DEFAULT- DISABLE PROCEDURE disable_UI : /*----- </pre>	<pre> ----- ----- ----- Purpose: DISABLE the User Interface Parameters: <none> Notes: Here we clean- up the user- interface by deleting dynamic widgets we have created and/or hide frames. This procedure is usually called when we are ready to "clean-up" after running. ----- ----- ----- ----- */ /* Delete the WINDOW we created */ IF SESSION:DISPLAY - TYPE = "GUI":U AND VALID- HANDLE(C-Win) THEN DELETE WIDGET C-Win. IF THIS- PROCEDURE:PERSI STENT THEN DELETE PROCEDURE THIS- PROCEDURE. END PROCEDURE. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME &ANALYZE- SUSPEND _UIB- CODE-BLOCK _PROCEDURE enable_UI C-Win DEFAULT-ENABLE </pre>	<pre> PROCEDURE enable_UI : /*----- ----- ----- Purpose: ENABLE the User Interface Parameters: <none> Notes: Here we display/view/en able the widgets in the user-interface. In addition, OPEN all queries associated with each FRAME and BROWSE. These statements here are based on the "Other Settings" section of the widget Property Sheets. ----- ----- ----- ----- */ DISPLAY t_contab t_fecha t_poliza t_fecha_descon WITH FRAME DEFAULT- FRAME IN WINDOW C-Win. ENABLE RECT- 14 RECT-4 RECT- 5 RECT-13 t_contab t_fecha t_poliza t_fecha_descon BUTTON-2 BUTTON-3 </pre>	<pre> WITH FRAME DEFAULT- FRAME IN WINDOW C-Win. {&OPEN- BROWSERS-IN- QUERY-DEFAULT- FRAME} VIEW C-Win. END PROCEDURE. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME /*Programa:c- lcatalocta.p*/ &ANALYZE- SUSPEND _VERSION-NUMBER UIB_v8r12 GUI &ANALYZE-RESUME &Scoped-define WINDOW-NAME C- Win &ANALYZE- SUSPEND _UIB- CODE-BLOCK _CUSTOM _DEFINITIONS C- Win /*----- ----- ----- ----- File: ----- ----- Description: ----- Input Parameters: <none> Output Parameters: <none> Author: Created: ----- ----- ----- /* This .W file was created </pre>	<pre> with the Progress UIB. */ /*----- ----- ----- ----- /* Create an unnamed pool to store all the widgets created by this procedure. This is a good default which assures that this procedure's triggers and internal procedures will execute in this procedure's storage, and that proper cleanup will occur on deletion of the procedure. */ CREATE WIDGET- POOL. /* ***** ***** Definitions ***** ***** */ DEF VAR t_flag AS LOGICAL NO- UNDO. DEFINE VARIABLE C_Inicial LIKE U_cliente.id_cl iente . DEFINE VARIABLE C_Final LIKE U_cliente.id_cl iente . DEFINE VARIABLE n_ctas AS INTEGER FORMAT ">>>>9". </pre>
--	--	--	--	---

DEFINE VARIABLE	AT 1		n_giran[4] .. AT	Definitions
t_ordena	FORMAT "X(134)."	U_cliente.stat_	8	*/
AS INTEGER	"P"	alta	AT	&Scoped-Define
FORMAT "9"	AT 1	58	U_cliente.direc	ENABLED-OBJECTS
INITIAL 1.	"Cta"		_edo_cta[4] AT	RECT-4 RECT-5
DEF VAR	AT 3	U_cliente.tipo_	68	k_tipo_reporte
k_imprimir	"Nombre"	persona	AT	BUTTON-2
AS LOGICAL NO-	AT 10	68		BUTTON-3
UNDO.	"F.alta"		U_cliente.direc	&Scoped-Define
DEF VAR:peque	AT 46	U_cliente.direc	_edo_cta[5] AT	DISPLAYED-
AS CHARACTER	"Status"	_edo_cta[1] AT	68	OBJECTS
FORMAT "X(25)"	AT 56	71	WITH	k_tipo_reporte
INITIAL:	"Tpo"		FRAME	
"_&llX:(10U (s3	AT 66	U_cliente.telef	fr_detalle	/* Custom List
T_(s0B_(s0S_(s1	"Direccion"	onos[1] AT		Definitions
0.00v16.66h0P".	AT 70	107	WIDTH 160 DOWN	*/
DEF.VAR kk_flag	AT 70		STREAM-IO.	/* List-1,List-
AS LOGICAL		U_cliente.id_ar		2,List-3,List-
INITIAL FALSE	"Telefono"	ea	AT	4,List-5,List-6
NO-UNDO.	AT 106	127		*/
	"Ar"		/* Parameters	*/
DEFINE STREAM	AT 123	U_cliente.id_di	vision	*/
report:	"Di"	vision	AT	/* _UIB-
	AT 126	130		PREPROCESSOR-
FORM SKIP(1)	"Pl"		/* Local	BLOCK-END */
'Cuentas	AT 129	U_cliente.id_pl	Definition	&ANALYZE-RESUME
Reportadas :	"Suc"	aza	AT	
AT 54 n_ctas	AT 132	133	/* _UIB-CODE-	/*
WITH FRAME	"Prom."		BLOCK-END */	*****
fr_total NO-	AT 44	U_cliente.id_su	&ANALYZE-RESUME	*****
LABELS WIDTH	"Resp."	cursal	AT	Control
136 STREAM-IO.	AT 54	136		Definitions
		SKIP		*****
FORM HEADER	FILL("_",134)		&ANALYZE-	*****
"Sistema de	AT 1	U_cliente.perso	SUSPEND _UIB-	***** */
Cheques en	FORMAT "X(134)."	n_giran[2] AT	PREPROCESSOR-	/* Define the
Dólares" AT 45	WITH	8	BLOCK	widget handle
SKIP	FRAME			for the window
"	fp_header	U_cliente.id_pr	/*	*/
CATALOGO.DE	WIDTH 150.	omotor AT	*****	DEFINE VAR C-
CUENTAS		44	*****	Win AS WIDGET-
AT 49	PAGE-TOP		Preprocessor	HANDLE NO-UNDO.
(1)	NO-LABELS	U_cliente.id_re	Definitions	
"Fecha de	STREAM-IO.	sponsab AT	*****	/* Definitions
emision :		54	***** */	of the field
AT 1	FORM			level widgets
TODAY	U_cliente.id_pr	U_cliente.direc	&Scoped-define-	*/
FORMAT:	oducto AT	_edo_cta[2] AT	PROCEDURE-TYPE	DEFINE BUTTON
"99/99/9999"	1	68	Window	BUTTON-2 AUTO-
SKIP				GO
"Hora :	U_cliente.id_cl	U_cliente.telef	/* Name of	IMAGE-UP
AT 1	iente AT	onos[2]	first Frame	FILE
	4		and/or Browse	"adeicon\run":U
STRING (TIME,"HH		U_cliente.perso	and/or first	LABEL
:MM:SS")	U_cliente.nombr	n_giran[3] AT	Query	"Button 2"
"PAG."	e AT	8	*/	SIZE 8 BY
AT 126	11 FORMAT		&Scoped-define	2.11.
PAGE-	'x(34)'	U_cliente.direc	FRAME-NAME	
NUMBER(report)		_edo_cta[3] AT	DEFAULT-FRAME	
COLON 131	U_cliente.f_alt	68		DEFINE BUTTON
FORMAT 999	a AT	SKIP	/* Standard	BUTTON-3 AUTO-
	47		List	END-KEY
FILL("_",134)		U_cliente.perso		

```

IMAGE-UP
FILE
"adeicon\exit-
au":U
    LABEL
    "Button 3"
    SIZE 8 BY
    2.11.
DEFINE VARIABLE
k_tipo_reporte
AS INTEGER
INITIAL 1
    VIEW-AS
RADIO-SET
VERTICAL
    RADIO-
    BUTTONS
    "Cuenta ", 1,
    "Nombre ", 2,
    "Promotor", 3,
    "Area-Div-
    Plaza-Suc", 4
    SIZE 26 BY
    4.47 NO-UNDO.
DEFINE
RECTANGLE RECT-
4
    EDGE-
    PIXELS 2
    GRAPHIC-EDGE
    NO-FILL
    SIZE 102
    BY 11.84.
DEFINE
RECTANGLE RECT-
5
    EDGE-
    PIXELS 2
    GRAPHIC-EDGE
    NO-FILL
    SIZE 102
    BY 2.63.
/*
*****
*****
Frame
Definitions
*****
***** */
DEFINE FRAME
DEFAULT-FRAME
k_tipo_reporte
AT ROW 5.21 COL
52 HELP
"Seleccione el
Orden el cual
será en el
Listado" NO-
LABEL
    BUTTON-2
    AT ROW 14.42
    COL 89 HELP
    "Esta
    Seguro de
    Iniciar el
    Proceso"
    BUTTON-3
    AT ROW 14.42
    COL 99 HELP
    "Salir de este
    Proceso"
"Seleccione el
orden del
reporte" VIEW-
AS TEXT
    SIZE
    31 BY 1.58 AT
    ROW 6.26 COL 19
    RECT-4 AT
    ROW 1.79 COL 7
    RECT-5 AT
    ROW 14.16 COL 7
    "
    Listado
    Catalogo de
    Cuentas" VIEW-
    AS TEXT
    SIZE
    32 BY 1.05 AT
    ROW 1.26 COL 42
    BGCOLOR 9
    FGCOLOR 14
    WITH 1 DOWN
    NO-BOX KEEP-
    TAB-ORDER
    OVERLAY
    SIDE-
    LABELS NO-
    UNDERLINE
    THREE-D
    AT COL
    1 ROW 1
    SIZE
    112.4 BY 16.84.
/*
*****
*****
Procedure
Settings
*****
***** */
&ANALYZE-
SUSPEND
_PROCEDURE-
SETTINGS
/* Settings for
THIS-PROCEDURE
Type: Window
Allow:
Basic, Browse, DB
-
Fields, Window, Q
uery
Other
Settings:
COMPILE
*/
&ANALYZE-RESUME
_END-PROCEDURE-
SETTINGS
/*
*****
*****
Create Window
*****
***** */
&ANALYZE-
SUSPEND
_CREATE-WINDOW
IF
SESSION:DISPLAY
- TYPE = "GUI":U
THEN
CREATE WINDOW
C-Win ASSIGN
HIDDEN
= YES
TITLE
= "Sistema de
Chequera en
Dólares"
HEIGHT
= 16.9
WIDTH
= 113
MAX-
HEIGHT
= 27.95
MAX-
WIDTH
= 159.6
VIRTUAL-HEIGHT
= 27.95
VIRTUAL-WIDTH
= 159.6
RESIZE
= yes
SCROLL-BARS
= no
STATUS-AREA
= yes
BGCOLOR
= ?
FGCOLOR
= ?
KEEP-
FRAME-Z-ORDER =
yes
THREE-
D =
yes
MESSAGE-AREA
= no
SENSITIVE
= yes.
ELSE {&WINDOW-
NAME} =
CURRENT-WINDOW.
/* END WINDOW
DEFINITION
*/
&ANALYZE-RESUME
/*
*****
*****
Runtime
Attributes and
UIB Settings
*****
*/
&ANALYZE-
SUSPEND_RUN-
TIME-ATTRIBUTES
/* SETTINGS FOR
WINDOW C-Win
VISIBLE,, RUN-
PERSISTENT
*/
/* SETTINGS FOR
FRAME DEFAULT-
FRAME
*/
IF
SESSION:DISPLAY
- TYPE = "GUI":U
AND VALID-
HANDLE(C-Win)
THEN C-
Win:HIDDEN =
no.
/* _RUN-TIME-
ATTRIBUTES-END
*/
&ANALYZE-RESUME
/*
*****
*****
Control
Triggers
*****
***** */
&Scoped-define
SELF-NAME C-Win
&ANALYZE-
SUSPEND_UIB-
CODE-BLOCK
_CONTROL C-Win
C-Win
ON END-ERROR OF
C-Win /*
Sistema de
Chequera en
Dólares */
OR ENDKEY OF
{&WINDOW-NAME}
ANYWHERE DO:
/* This case
occurs when the
user presses
the "Esc" key.
In a
persistently
run window,
just ignore
this. If we
did not, the
application
would exit. */
IF THIS-
PROCEDURE:PERSI
STENT THEN
RETURN NO-
APPLY.
END.
/* _UIB-CODE-
BLOCK-END */
&ANALYZE-RESUME
&ANALYZE-
SUSPEND_UIB-
CODE-BLOCK
_CONTROL C-Win
C-Win

```

```

ON WINDOW-CLOSE
OF C-Win /*
Sistema de
Chequera en
Dólares */
DO:
  /* This event
  will close the
  window and
  terminate the
  procedure. */
  APPLY
  "CLOSE":U TO
  THIS-PROCEDURE.
  RETURN NO-
  APPLY.
  END.

/*~_UIB-CODE-
BLOCK-END */
&ANALYZE-RESUME

&Scoped-define
SELF-NAME
BUTTON-2
&ANALYZE-
SUSPEND _UIB-
CODE-BLOCK.
_CONTROL
BUTTON-2 C-Win
ON CHOOSE OF
BUTTON-2 IN
FRAME DEFAULT-
FRAME /* Button
2 */
DO:
  SYSTEM-DIALOG
  PRINTER-SETUP
  UPDATE
  k_imprimir.

  IF NOT
  k_imprimir THEN
  DO:
    MESSAGE
    "IMPRESIÓN
    CANCELADA".
    OUTPUT
    STREAM report
    CLOSE.
    RETURN NO-
    APPLY.
    END.
    OUTPUT STREAM
    report TO
    PRINTER-PAGED
    PAGE-SIZE 58.
    PUT STREAM
    report CONTROL
    peque.

    t_flag = .
    TRUE.
    t_ordena =
    k_tipo_reporte.
    n_ctas = 0.
    IF t_ordena =
    1 THEN
      FOR EACH
      U_cliente
      WHERE
      U_cliente.id_pr
      oducto = "Q"
      USE-
      INDEX cliente:
      {qlcatct0.i1}
      END.
    ELSE
      IF t_ordena =
      2 THEN
        FOR EACH
        U_cliente
        WHERE
        U_cliente.id_pr
        oducto = "Q"
        USE-
        INDEX nombre:
        {qlcatct0.i1}
        END.
      ELSE
        IF t_ordena =
        3 THEN
          FOR EACH
          U_cliente
          WHERE
          U_cliente.id_pr
          oducto = "Q"
          USE-
          INDEX
          id_promotor:
          {qlcatct0.i1}
          END.
        ELSE
          IF t_ordena =
          4 THEN
            FOR EACH
            U_cliente
            WHERE
            U_cliente.id_pr
            oducto = "Q"
            USE-
            INDEX perfil:
            {qlcatct0.i1}
            END.
            DISPLAY STREAM
            report

            n_ctas
            WITH
            FRAME fr_total
            DOWN STREAM
            report WITH
            FRAME fr_total.
            OUTPUT STREAM
            report CLOSE.
            IF t_flag THEN
            MESSAGE
            "FINALIZO EL
            LISTADO".
            ELSE
            MESSAGE "NO
            SE GENERO
            LISTADO".
            APPLY 'CLOSE'
            TO THIS-
            PROCEDURE.
            END.
            /*~_UIB-CODE-
            BLOCK-END */
            &ANALYZE-RESUME

            &Scoped-define
            SELF-NAME
            k_tipo_reporte
            &ANALYZE-
            SUSPEND _UIB-
            CODE-BLOCK.
            _CONTROL
            k_tipo_reporte
            C-Win
            ON VALUE-
            CHANGED OF
            k_tipo_reporte
            IN FRAME
            DEFAULT-FRAME
            DO:
              k_tipo_reporte
              = INPUT
              k_tipo_reporte.
              END.
              /*~_UIB-CODE-
              BLOCK-END */
              &ANALYZE-RESUME

              &UNDEFINE "SELF-
              NAME
              &ANALYZE-
              SUSPEND _UIB-
              CODE-BLOCK

              _CUSTOM_MAIN-
              BLOCK C-Win
              /*
              *****
              *****
              Main Block:
              *****
              ***** */
              /* Set CURRENT-
              WINDOW: this
              will parent
              dialog-boxes
              and frames:
              */
              ASSIGN CURRENT-
              WINDOW
              = {&WINDOW-
              NAME}
              THIS-
              PROCEDURE:CURRE
              NT-WINDOW=
              {&WINDOW-NAME}.
              /* The CLOSE
              event can be
              used from
              inside or
              outside the
              procedure to
              */
              /* terminate
              it.
              */
              ON CLOSE OF
              THIS-PROCEDURE
              RUN
              disable_UI.
              /* Best default
              for GUI
              applications
              is...
              */
              PAUSE 0 BEFORE-
              HIDE.
              /* Now enable
              the interface
              and wait for
              the exit
              condition.
              */
              /* (NOTE:
              handle ERROR
              and END-KEY so
              cleanup code
              will always
              fire.
              */
              MAIN-BLOCK:
              DO ON ERROR
              UNDO MAIN-
              BLOCK, LEAVE
              MAIN-BLOCK
              ON END-KEY
              UNDO MAIN-
              BLOCK, LEAVE
              MAIN-BLOCK:
              RUN
              enable_UI.
              IF NOT THIS-
              PROCEDURE:PERSI-
              STENT THEN
              WAIT-FOR
              CLOSE OF THIS-
              PROCEDURE.
              END.
              /*~_UIB-CODE-
              BLOCK-END */
              &ANALYZE-RESUME

              /*
              *****
              *****
              Internal
              Procedures
              *****
              ***** */

              &ANALYZE-
              SUSPEND _UIB-
              CODE-BLOCK
              _PROCEDURE:
              disable_UI C-
              Win DEFAULT-
              DISABLE
              PROCEDURE
              disable_UI :
              /*-----
              -----
              -----
              -----
              -----
              Purpose:
              DISABLE the
              User Interface.
              Parameters:
              <none>
              Notes:
              Here we clean-
              up the user-
              interface by
              deleting
              dynamic widgets
              we have created
              and/or hide
              frames. This
              procedure is
          
```

usually called when	OPEN all queries	_DEFINITIONS C-Win	deletion of the procedure. */	Q_Corte.n_cheques_rebotados INITIAL 0.
we are ready to "clean-up" after running.	associated with each FRAME and BROWSE.	/----- /----- /-----	CREATE WIDGET-POOL.	DEFINE NEW SHARED VARIABLE P_n_che_emit
----- ----- -----	These statements here are based on the "Other	File:	/* *****	LIKE Q_Corte.n_cheques_emitidos
---*/	Settings"	Description:	***** Definitions ***** */	INITIAL 0. DEFINE NEW SHARED VARIABLE
/* Delete the WINDOW we created */ IF	section of the widget Property Sheets.	Input Parameters: <none>	DEF VAR t_flag AS LOGICAL NO-UNDO.	P_Tot_Cheques LIKE Q_Corte.Tot_Che_Emitidos
SESSION:DISPLAY -TYPE = "GUI":U AND VALID- HANDLE(C-Win) THEN DELETE WIDGET C-Win.	----- ----- ----- -----	Output Parameters: <none>	DEFINE NEW SHARED VARIABLE P_prod LIKE	SHARED VARIABLE P_n_Debits LIKE
IF THIS- PROCEDURE:PERSI STENT THEN	---*/ DISPLAY	Author:	U_cliente.id_pr oducto.	Q_Corte.n_Debits S INITIAL 0.
DELETE PROCEDURE THIS- PROCEDURE. END PROCEDURE.	k_tipo_reporte WITH FRAME DEFAULT- FRAME IN WINDOW C-Win. ENABLE RECT-4 RECT-5	Created:	DEFINE NEW SHARED VARIABLE P_cuenta LIKE	DEFINE NEW SHARED VARIABLE P_Tot_Debits LIKE
/* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME	k_tipo_reporte BUTTON-2 BUTTON-3 WITH FRAME DEFAULT- FRAME IN WINDOW C-Win.	----- ----- -----	U_cliente.id_cl iente.	Q_Corte.Tot_Debits INITIAL 0. DEFINE NEW
&ANALYZE- SUSPEND _UIB- CODE-BLOCK PROCEDURE	{&OPEN- BROWSERS-IN- QUERY-DEFAULT- FRAME}	This .W file was created with the Progress UIB.	SHARED VARIABLE P_F_inicial AS DATE FORMAT "99/99/9999".	SHARED VARIABLE P_n_Credits LIKE
enable_UI C-Win DEFAULT-ENABLE PROCEDURE enable_UI :	VIEW C-Win. END PROCEDURE. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME	/* Create an unnamed pool to store all the widgets created by this procedure. This is a good default which assures	DEFINE NEW SHARED VARIABLE P_F_final AS DATE FORMAT "99/99/9999".	Q_Corte.n_Credits INITIAL 0. DEFINE NEW SHARED VARIABLE P_Tot_Credits LIKE
----- ----- ----- ----- -----	/*Programa:c- ctasaltascanc.p */	/*	DEFINE NEW SHARED VARIABLE p_SalIni LIKE	Q_Corte.Tot_Cre dits INITIAL 0. DEFINE NEW SHARED VARIABLE P_Interes LIKE
Purpose: ENABLE the User Interface Parameters: <none>	&ANALYZE- SUSPEND	that this procedure's triggers and internal procedures	Q_Corte.Saldo_I nicial INITIAL 0.	Q_Corte.Interes INITIAL 0. DEFINE NEW SHARED VARIABLE P_Interes_anual LIKE
Notes: Here we display/view/en able the widgets in the	_VERSION-NUMBER UIB_v8r12 GUI &ANALYZE-RESUME &Scoped-define WINDOW-NAME C- Win &ANALYZE- SUSPEND _UIB- CODE-BLOCK _CUSTOM	will execute in this procedure's storage, and that proper cleanup will occur on	SHARED VARIABLE P_SalFin LIKE Q_Corte.Saldo_F inal INITIAL 0. DEFINE NEW SHARED VARIABLE P_SalPro LIKE	SHARED VARIABLE P_Interes_anual LIKE Q_Corte.Interes Annual INITIAL 0.
user-interface. In addition,			Q_Corte.Saldo_P romedio INITIAL 0. DEFINE NEW SHARED VARIABLE P_n_che_rebot LIKE	DEFINE NEW SHARED VARIABLE P_Tasa_Int LIKE Q_Corte.Tasa_In teres INITIAL 0. DEF VAR t_totsdo LIKE

Code	Definition	Width	Form	Definitions
M_inversion.mon to.	DEF VAR peque AS CHARACTER	150	FORM	Definitions
DEF VAR	FORMAT "X(25)"		"Procesando	*/
t_prosd	LIKE		Información."	&Scoped-Define
M_inversion.mon to.	INITIAL	TOP.STREAM-IO	SKIP	ENABLED-OBJECTS
	"_&11X_(10U_(s3	NO-	"No. de	RECT-4 RECT-5
	T_(s0B_(s0S_(s1	LABELS.	Registros....	k_estatus
	0.00v16.66h0P".		" t_reg	k_f_ini k_f_fin
DEF VAR t_cont	DEF VAR kk_flag	FORM t_cont	WITH FRAME	-
AS INTEGER	AS LOGICAL	COLUMN-LABEL	fr_proc	BUTTON-2
FORMAT ">>>>9"	INITIAL FALSE	"Cont."	ROW 5	BUTTON-3
INITIAL 0	NO-UNDO.		COL 5	&Scoped-Define
NO-UNDO.		U_cliente.id_pr		DISPLAYED-
DEF VAR		oducto	OVERLAY	OBJECTS
t_id_esta AS	DEF VAR	COLUMN-LABEL	NO-	k_estatus
LOGICAL	t_are_prom AS	"Pd."	LABELS	k_f_ini k_f_fin
FORMAT	LOGICAL FORMAT		COLOR	
"Alta/Cancelada	"Ara/Promotor"	U_cliente.id_cl	WHITE/BLUE.	/* Custom List
" INITIAL	DEF STREAM:	iente		Definitions
"Alta" NO-UNDO.	report	COLUMN-LABEL		*/
DEF VAR		"Cte."	/* Parameters	/* List-1, List-
t_id_prod AS	FORM HEADER	U_cliente.nombr	Definitions ---	2, List-3, List-
CHARACTER	"Sistema de	e	*/	4, List-5, List-6
FORMAT "X(1)"	Cheques en	FORMAT "X(25)"	/* Local	*/
INITIAL "Q"	Dólares" AT 50	COLUMN-LABEL	Variable	/* UIB-
NO-UNDO.	SKIP	"Nombre"	Definitions ---	PREPROCESSOR-
DEF VAR t_name	"Estatus	U_cliente.f_alt	*/	BLOCK-END */
AS CHARACTER	de Chequeras"	a	/* UIB-CODE-	&ANALYZE-RESUME
FORMAT "X(12)"	AT 53	COLUMN-LABEL	BLOCK-END */	
NO-UNDO.	t_desc_stat	"Fecha Alta"	&ANALYZE-RESUME	
DEF VAR t_reg	AT 76	U_cliente.f_can		/*
AS INTEGER	"Fecha de	celacion	&ANALYZE-	*****
FORMAT ">>>>9"	emision : "	COLUMN-LABEL	SUSPEND UIB-	*****
INITIAL 0	AT 1	"Fecha Can."	PREPROCESSOR-	Control
NO-UNDO.	TODAY	p_salfin	BLOCK	Definitions
DEF VAR	FORMAT	COLUMN-LABEL		*****
t_desc_stat AS	"99/99/9999"	"Sdo. Inicial"	/*	***** */
CHARACTER	SKIP		*****	/* Define the
FORMAT "X(20)"	"Hora :"	U_cliente.id_pr	Preprocessor	widget handle
NO-UNDO.	AT 1	omotor	Definitions	for the window.
DEF VAR t_f_ini	STRING(TIME, "HH	COLUMN-LABEL	*****	*/
AS DATE FORMAT	:MM:SS")	"ID. Prom."	***** */	DEFINE VAR C-
"99/99/9999"	"Periodo:	M_promotor.nomb		Win AS WIDGET-
DEF VAR t_imp	"	re	&Scoped-define	HANDLE NO-UNDO.
AS LOGICAL	AT 25	FORMAT "X(25)"	PROCEDURE-TYPE	/* Definitions
FORMAT	t_f_ini	COLUMN-LABEL	Window	of the field
"Impresora/Pant	" AL "	"Nombre		level widgets
alla"	t_f_fin	Promotor"	/* Name of	*/
DEF VAR compres	"PAG."		first Frame	DEFINE BUTTON
AS CHARACTER	AT 126	M_ubicacion.nom	and/or Browse	BUTTON-2 AUTO-
INITIAL:	PAGE-	bre	and/or first	GO
	NUMBER(report)	FORMAT "X(12)"	Query	IMAGE-UP
"_&11X_&10S_&10-	COLON 131	COLUMN-LABEL	*/	FILE
l0o2a6dle66F(8	FORMAT 999	"Nombre Area"	&Scoped-define	"adeicon\run":U
U_(s3T_(s0B_(s0	FILL("_", 134)	WITH DOWN	FRAME-NAME	LABEL
S_(s16.00v16.00	AT 1	FRAME fr_imp,	DEFAULT-FRAME	"Button 2"
h0P".	FORMAT "X(134)"	WIDTH 136		SIZE 8 BY
DEF VAR	WITH FRAME-	STREAM-IO..	/* Standard	2.1
k_imprimir	fr_header		List	
AS LOGICAL NO-				
UNDO.				

```

DEFINE BUTTON GRAPHIC-EDGE UNDERLINE MAX- /* SETTINGS FOR
BUTTON-3 AUTO- NO-FILL THREE-D HEIGHT FRAME DEFAULT-
END-KEY SIZE 102 AT COL = 27.95 FRAME
IMAGE-UP BY 2.62. 1 ROW 1 MAX-
FILE SIZE WIDTH /*
"adeicon\exit- 112.4 BY 16.84. = 159.6 ASSIGN
au":U
LABEL *****
"Button 3" ***** /*
SIZE 8 BY Frame *****
2.1. Definitions *****
Procedure
***** */ Settings *****
DEFINE VARIABLE k_estatus AS ***** */
LOGICAL FORMAT DEFINE FRAME
"Alta/Cancelada DEFAULT-FRAME
":U INITIAL NO k_estatus
LABEL AT ROW 4.14 COL
"Estatus 49 COLON-
(Alta/Cancelada ALIGNED AUTO-
)" RETURN
VIEW-AS k_f_ini AT
FILL-IN ROW 6.24 COL 29
SIZE 14 BY COLON-ALIGNED
1 NO-UNDO. AUTO-RETURN
k_f_fin AT
DEFINE VARIABLE ROW 6.24 COL 68
k_f_fin AS DATE COLON-ALIGNED
FORMAT AUTO-RETURN
"99/99/9999":U BUTTON-2
LABEL AT ROW 14.33
"Fecha final" COL 89 HELP
VIEW-AS "Esta
FILL-IN Seguro de
SIZE 16 BY Iniciar el
1 NO-UNDO. Proceso"
BUTTON-3
DEFINE VARIABLE AT ROW 14.33
k_f_ini AS DATE COL 99 HELP
FORMAT Create Window
"99/99/9999":U *****
LABEL "Salir de este *****
"Fecha Inicial" RECT-4 AT /*
VIEW-AS ROW 1.81 COL 6 &ANALYZE-
FILL-IN RECT-5 AT SUSPEND
SIZE 16 BY ROW 14.14 COL 6 _CREATE-WINDOW
1 NO-UNDO. " IF
Listado Cuentas SESSION:DISPLAY
Dadas de Alta -TYPE = "GUI":U
Canceladas" THEN
VIEW-AS TEXT CREATE WINDOW
4 SIZE C-Win ASSIGN
EDGE- 46 BY 1.05 AT HIDDEN
PIXELS 2 ROW 1.24 COL 36 = YES
GRAPHIC-EDGE BGCOLOR 9 TITLE
NO-FILL FGCOLOR 14 = "Sistema de
SIZE 102 WITH 1 DOWN
BY 12.05. NO-BOX KEEP- HEIGHT
DEFINE RECTANGLE RECT- TAB-ORDER = 16.91
5 OVERLAY WIDTH
EDGE- SIDE- = 113
PIXELS 2 LABELS NO-

```

Code Block	Code	Output	Where	Data
<pre> Chequera en Dólares */ OR ENDKEY OF {&WINDOW-NAME} ANYWHERE DO: /* This case occurs when the user presses the "Esc" key. In a persistently run window, just ignore this. If we did not, the application would exit. */ IF THIS- PROCEDURE:PERSI- STENT THEN RETURN NO- APPLY. END. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME &ANALYZE- SUSPEND _UIB- CODE-BLOCK _CONTROL C-Win C-Win ON WINDOW-CLOSE OF C-Win /* Sistema de Chequera en Dólares */ DO: /* This event will close the window and terminate the procedure. */ *APPLY. "CLOSE":U TO THIS-PROCEDURE. RETURN NO- APPLY. END. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME &Scoped-define SELF-NAME BUTTON-2 &ANALYZE- SUSPEND _UIB- </pre>	<pre> CODE-BLOCK _CONTROL BUTTON-2 C-Win ON CHOOSE OF BUTTON-2 IN FRAME DEFAULT- FRAME /* Button 2 */ DO: IF k_f_ini = ? OR k_f_fin = ? THEN DO: MESSAGE "Debe teclear valor-en el campo Fecha". RETURN NO- APPLY. END. t_flag = FALSE. t_id_prod = "Q". t_id_esta = k_estatus. t_f_ini = k_f_ini. t_f_fin = k_f_fin. t_impa = TRUE. IF t_id_esta THEN t_desc_stat = " A l t a ". ELSE t_desc_stat = " C a n c e l a d a ". t_cont = 0. t_reg = 0. /***** Impresora. *****/ SYSTEM-DIALOG PRINTER-SETUP UPDATE: k_imprimir. IF NOT k_imprimir THEN DO: MESSAGE "IMPRESIÓN CANCELADA". </pre>	<pre> OUTPUT STREAM report CLOSE. RETURN NO- APPLY. END. OUTPUT STREAM report TO PRINTER PAGED PAGE-SIZE 58. PUT STREAM report CONTROL peque. IF t_impa THEN DO: /*** Altas *****/ IF t_id_esta THEN DO: FOR EACH U_cliente WHERE U_cliente.id_pr oducto = t_id_prod AND U_cliente.stat_ alta = t_id_esta AND U_cliente.f_alt a >= t_f_ini AND U_cliente.f_alt a <= t_f_fin BY U_cliente.f_alt a BY U_cliente.id_cl iente. VIEW STREAM report FRAME fr_header. kk_flag = TRUE. FIND FIRST M_promotor </pre>	<pre> WHERE M_promotor.id_p romotor = U_cliente.id_pr omotor NO-ERROR. t_cont = t_cont + 1. t_reg = t_cont. P_f_inicial = U_cliente.f_alt a. P_f_final = U_cliente.f_alt a. P_cuenta = U_cliente.id_cl iente. P_prod = U_cliente.id_pr oducto. FIND FIRST Q_MOVIMIENTO WHERE Q_movimiento.id _producto = U_cliente.id_pr oducto AND Q_movimiento.id _cliente = U_cliente.id_cl iente NO-ERROR. IF AVAILABLE Q_movimiento THEN p_salfin = Q_movimiento.mo nto. ELSE p_salfin = 0. </pre>	<pre> t_totsdo = t_totsdo + p_salfin. DISPLAY STREAM report t_cont U_cliente.id_pr oducto U_cliente.id_cl iente U_cliente.nombr e U_cliente.f_alt a U_cliente.f_can celacion p_salfin U_cliente.id_pr omotor M_promotor.nomb re WITH FRAME fr_imp. DISPLAY t_reg WITH FRAME fr_proc. PAUSE 0. DOWN STREAM report WITH FRAME fr_imp. END. /* FOR EACH */ END. /* IF t_id_esta THEN DO:IMPRESORA Y ALTA */ ELSE DO: </pre>

```

/**/ Canceladas
/**/
FOR EACH
U_cliente
WHERE
U_cliente.id_pr
oducto =
t_id_prod AND
U_cliente.stat_
alta =
t_id_esta AND
U_cliente.f_can
celacion >=
t_f_ini AND
U_cliente.f_can
celacion <=
t_f_fin
BY
U_cliente.f_can
celacion
BY
U_cliente.id_cl
iente:
VIEW STREAM
report FRAME
fr_header.
kk_flag = TRUE.
FIND FIRST
M_promotor
WHERE
M_promotor.id_p
romotor =
U_cliente.id_pr
omotor
NO-ERROR.
t_cont =
t_cont + 1.
t_reg =
t_cont.
P_f_inicial =
U_cliente.f_alt
a.
P_f_final =
U_cliente.f_alt
a.
P_cuenta =
U_cliente.id_cl
iente.
P_prod =
U_cliente.id_pr
oducto.
FIND FIRST
Q_MOVIMIENTO
WHERE
Q_movimiento.id
_producto =
U_cliente.id_pr
oducto AND
Q_movimiento.id
_cliente =
U_cliente.id_cl
iente
NO-ERROR.
IF AVAILABLE
Q_movimiento
THEN
p_salfin =
Q_movimiento.mo
nto.
ELSE
p_salfin = 0.
t_totsdo =
t_totsdo +
p_salfin.
DISPLAY STREAM
report
t_cont
U_cliente.id_pr
oducto
U_cliente.id_cl
iente
U_cliente.nombr
e
U_cliente.f_alt
a
U_cliente.f_can
celacion
p_salfin
U_cliente.id_pr
omotor
M_promotor.nomb
re
WITH FRAME
fr_imp.
DISPLAY t_reg
WITH FRAME
fr_proc.
PAUSE 0.
DOWN STREAM
report WITH
FRAME fr_imp.
END. /* FOR
EACH */
t_prosdo =
t_totsdo /
t_cont.
DISPLAY STREAM
report
SKIP(2)
"Total
depositos
iniciales.
: " t_totsdo
SKIP
"Total Promedio
Depos.
Iniciales: "
t_prosdo
WITH NO-LABELS
NO-BOX FRAME
fr_tot STREAM-
IO.
t_totsdo = 0.
PAGE STREAM
report.
END. /*
CANCELADAS Y
PARA IMPRESION
*/
END. /*
IMPRESORA */
OUTPUT
STREAM report
CLOSE.
IF kk_flag
THEN
MESSAGE
"FINALIZO EL
LISTADO".
ELSE
MESSAGE
"NO SE GENERO
LISTADO".
APPLY
'CLOSE' TO
THIS-PROCEDURE.
END.
/* _UIB-CODE-
BLOCK-END */
&ANALYZE-RESUME
&Scoped-define
SELF-NAME
k_f_fin
&ANALYZE-
SUSPEND _UIB-
CODE-BLOCK
_CONTROL
k_f_fin C-Win
ON LEAVE OF
k_f_fin IN.
FRAME DEFAULT-
FRAME /* Fecha
final */
DO:
k_f_fin =
INPUT k_f_fin.
END.
/* _UIB-CODE-
BLOCK-END */
&ANALYZE-RESUME
&Scoped-define
SELF-NAME
k_f_ini
&ANALYZE-
SUSPEND _UIB-
CODE-BLOCK
_CONTROL
k_f_ini C-Win
ON LEAVE OF
k_f_ini IN
FRAME DEFAULT-
FRAME /* Fecha
Inicial */
DO:
k_f_ini =
INPUT k_f_ini.
END.
/* _UIB-CODE-
BLOCK-END */
&ANALYZE-RESUME
&UNDEFINE SELF-
NAME
&ANALYZE-
SUSPEND _UIB-
CODE-BLOCK
_CUSTOM_MAIN-
BLOCK C-Win
/*
*****
*****
Main Block

```

<pre> ***** ***** */ /* Set CURRENT- WINDOW: this will parent dialog-boxes and frames. */ ASSIGN CURRENT- WINDOW- = {&WINDOW- NAME} THIS- PROCEDURE:CURRE NT-WINDOW = {&WINDOW-NAME}. /* The CLOSE event can be used from inside or outside the procedure to */ /* terminate it. */ ON CLOSE OF THIS-PROCEDURE RUN disable_UI. /* Best default for GUI applications is... */ PAUSE 0 BEFORE- HIDE. /* Now enable the interface and wait for the exit condition. */ /* (NOTE: handle ERROR and-END-KEY so cleanup code will always fire. */ MAIN-BLOCK: DO ON ERROR: UNDO MAIN- BLOCK; LEAVE MAIN-BLOCK ON END-KEY UNDO MAIN- BLOCK; LEAVE MAIN-BLOCK: </pre>	<pre> RUN enable_UI. IF NOT THIS- PROCEDURE:PERSI STENT THEN WAIT-FOR CLOSE OF THIS- PROCEDURE. END: /* _UIB-CODE- BLOCK-END.*/ &ANALYZE-RESUME Internal Procedures ***** &ANALYZE- SUSPEND _UIB- CODE-BLOCK PROCEDURE disable_UI C- Win_DEFAULT- DISABLE PROCEDURE disable_UI : Purpose: DISABLE the User Interface Parameters: <none> Notes: Here we clean- up the user- interface by deleting dynamic:widgets we have created and/or hide frames. This procedure is usually called when we are ready to "clean-up" after running: </pre>	<pre> ----- ----- -----*/ /* Delete the WINDOW we created */ IF SESSION:DISPLAY -TYPE' = "GUI":U AND VALID- HANDLE(C-Win) THEN: DELETE C-Win. IF THIS- PROCEDURE:PERSI STENT THEN DELETE PROCEDURE THIS- PROCEDURE. END PROCEDURE. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME &ANALYZE- SUSPEND _UIB- CODE-BLOCK PROCEDURE enable_UI C-Win DEFAULT-ENABLE PROCEDURE enable_UI : /* Purpose: ENABLE the User Interface Parameters: <none> Notes: Here we display/view/en able the widgets in the user-interface: In addition, OPEN all queries associated with each FRAME and BROWSE. These statements here </pre>	<pre> are based on the "Other Settings" section of the widget Property Sheets. ----- ----- -----*/ DISPLAY k_estatus k_f_ini k_f_fin WITH FRAME DEFAULT- FRAME IN WINDOW C-Win. ENABLE, RECT-4 RECT-5 k_estatus k_f_ini k_f_fin BUTTON-2 BUTTON-3 WITH FRAME DEFAULT- FRAME IN WINDOW C-Win. {&OPEN- BROWSERS-IN- QUERY-DEFAULT- FRAME} VIEW C-Win. END PROCEDURE. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME /*Programa:c- lctasaltascancp rom*/ &ANALYZE- SUSPEND _VERSION-NUMBER _UIB_v8r12 GUI &ANALYZE-RESUME &Scoped-define WINDOW-NAME C- Win &ANALYZE- SUSPEND _UIB- CODE-BLOCK _CUSTOM _DEFINITIONS C- Win /* </pre>	<pre> ----- ----- File: Description: Input Parameters: <none> Output Parameters: <none> Author: Created: ----- ----- /* This .W file was created with the Progress UIB. */ /* ----- ----- /* Create an unnamed pool to store all the widgets created by this procedure. This is a good default which assures that this procedure's triggers and internal procedures will execute in this procedure's storage, and that proper cleanup will occur on deletion of the procedure. */ CREATE WIDGET- POOL. </pre>
--	---	---	---	---

/*	LIKE	INITIAL FALSE	
*****	Q_Corte.n_chequ	NO-UNDO.	U_cliente.id_cl
*****	es emitidos		iente
Definitions	INITIAL 0.	DEF STREAM	COLUMN-LABEL
*****	DEFINE NEW	report.	"Cte."
*****	SHARED VARIABLE		
*/	P_Tot_Cheques	FORM HEADER	U_cliente.nomb
DEF VAR t_flag	LIKE	"Sistema de	e
AS LOGICAL NO-	Q_Corte.Tot_Che	Cheques en	FORMAT "X(25)"
UNDO.	_Emitidos	Dólares" AT 40	COLUMN-LABEL
	INITIAL 0.	SKIP	"Nombre"
DEFINE NEW	DEFINE NEW	"Estatus	
SHARED VARIABLE	SHARED VARIABLE	de Chequeras"	U_cliente.f_alt
P_prod	P_n Debits	AT 50	a
LIKE	LIKE	t_desc_stat	COLUMN-LABEL
U_cliente.id_pr	Q_Corte.n_Debit	AT 86	"Fecha Alta"
oducto.	s INITIAL 0:	"Fecha de	U_cliente.f_can
DEFINE NEW	DEFINE NEW	emision : "	celacion
SHARED VARIABLE	SHARED VARIABLE	AT 1	COLUMN-LABEL
P_cuenta	P_Tot_Debits	TODAY	"Fecha Can."
LIKE	LIKE	FORMAT	p_salfin
U_cliente.id_cl	Q_Corte.Tot_Deb	"99/99/9999"	COLUMN-LABEL
iente.	its INITIAL 0.	SKIP	"Sdo. Inicial"
DEFINE NEW	DEFINE NEW	"Hora : "	
SHARED VARIABLE	SHARED VARIABLE	AT 1	U_cliente.id_pr
P_F_inicial	P_n Credits	STRING(TIME, "HH	omotor
AS DATE FORMAT	LIKE	:MM:SS")	COLUMN-LABEL
"99/99/9999"	Q_Corte.n_Credi	"Periodo:	"ID. Prom."
DEFINE NEW	ts INITIAL 0.		
SHARED VARIABLE	DEFINE NEW		M_promotor.nomb
P_F_final	SHARED VARIABLE	AT 25	re
AS DATE FORMAT	P_Tot_Credits	t_f_ini	FORMAT "X(25)"
"99/99/9999"	LIKE	" AL "	COLUMN-LABEL
DEFINE NEW	Q_Corte.Tot_Cre	t_f_fin	"Nombre
SHARED VARIABLE	dits INITIAL 0.	"PAG."	Promotor"
P_SalIni	DEFINE NEW	AT 120	
LIKE	SHARED VARIABLE	PAGE-	M_ubicacion.nom
Q_Corte.Saldo_I	P_Interes	NUMBER(report)	bre
nicial INITIAL	LIKE	AT 130 FORMAT	FORMAT "X(12)."
0.	Q_Corte.Interes	999	COLUMN-LABEL
DEFINE NEW	INITIAL 0.	FILL(" ",134)	"Nombre Area"
SHARED VARIABLE	DEFINE NEW	AT 1 FORMAT	WITH DOWN
P_SalFin	SHARED VARIABLE	"X(140)"	FRAME fr_imp
LIKE	P_Interes_anual	WITH FRAME	WIDTH 150
Q_Corte.Saldo_F	LIKE	fr_header	STREAM-IO.
inal INITIAL 0.	Q_Corte.Interes	WIDTH	
DEFINE NEW	_Anual INITIAL	170	FORM
SHARED VARIABLE	0.	PAGE-	"Procesando
P_SalPro	DEFINE NEW	TOP STREAM-IO	Información"
LIKE	SHARED VARIABLE	NO-	SKIP
Q_Corte.Saldo_P	P_Tasa_Int	LABELS.	"No. de
romedio INITIAL	LIKE		Registros.....
0.	Q_Corte.Tasa_In	FORM t_cont	." t_reg
DEFINE NEW	teres INITIAL	COLUMN-LABEL	WITH FRAME
SHARED VARIABLE	0.	"Cont."	fr_proc
P_n_che_rebot	DEF VAR	U_cliente.id_pr	ROW 5
LIKE	t_totsdo LIKE	oducto	COL 5
Q_Corte.n_chequ	M_inversion.mon	COLUMN-LABEL	
es_rebotados	to.	"Pd."	
INITIAL 0.	DEF VAR		OVERLAY
DEFINE NEW	t_prosdo LIKE		NO-
SHARED VARIABLE	M_inversion.mon		LABELS
P_n_che_emit	to.		

COLOR		"Area/Promotor"	GRAPHIC-EDGE	SIZE
WHITE/BLUE.	/* Custom List Definitions	:U INITIAL NO LABEL	NO-FILL	51 BY 1.05 AT ROW 1.48,COL,31
/* Parameters; Definitions --- */	/* List-1,List-2,List-3,List-4,List-5,List-6 */	"Ordena por (Area/Promotor)."	SIZE 102 BY 2.62	BGCOLOR 9 FGCOLOR 14 WITH 1 DOWN NO-BOX KEEP-TAB-ORDER OVERLAY
/* Local Variable Definitions --- */	/* _UIB- PREPROCESSOR-BLOCK-END */ &ANALYZE-RESUME	VIEW-AS FILL-IN SIZE 14 BY 1 NO-UNDO.	***** Frame Definitions *****	SIDE-LABELS NO-UNDERLINE THREE-D AT COL
/* _UIB-CODE-BLOCK-END */ &ANALYZE-RESUME	/*	DEFINE VARIABLE k_estatus AS LOGICAL FORMAT "Alta/Cancelada":U INITIAL NO LABEL	DEFINE FRAME DEFAULT-FRAME k_estatus AT ROW,4.14 COL 49 COLON-ALIGNED AUTO-RETURN	1 ROW 1.12.4 BY 16.84
&ANALYZE-SUSPEND _UIB- PREPROCESSOR-BLOCK	***** Control Definitions *****	"Estatus (Alta/Cancelada)"	k_areaprom AT ROW 6.24 COL 49 COLON-ALIGNED HELP	/*
/*	*****	VIEW-AS FILL-IN SIZE 14 BY 1 NO-UNDO.	RETURN	*****
*****	/* Define the widget handle for the window */	DEFINE VARIABLE k_f_fin AS DATE FORMAT "99/99/9999":U LABEL "Fecha final"	"Cuentas Vigentes Canceladas Ordenadas por Area/Promotor" AUTO-RETURN	Procedure Settings
Preprocessor Definitions	DEFINE VAR C-Win AS WIDGET-HANDLE NO-UNDO.	VIEW-AS FILL-IN SIZE 16 BY 1 NO-UNDO.	k_f_ini AT ROW 8.38 COL 30 COLON-ALIGNED AUTO-RETURN	*****
*****	/* Definitions of the field level widgets */	DEFINE VARIABLE k_f_ini AS DATE FORMAT "99/99/9999":U LABEL "Fecha Inicial"	k_f_fin AT ROW 8.38 COL 69 COLON-ALIGNED AUTO-RETURN	/* Settings for THIS-PROCEDURE Type: Window Allow: Basic,Browse,DB Fields,Window,Query Other. Settings: COMPILE
&Scoped-define PROCEDURE-TYPE Window.	/*	VIEW-AS FILL-IN SIZE 16 BY 1 NO-UNDO.	BUTTON-2 AT ROW 14.33 COL 89 HELP	*/
/* Name of first Frame and/or Browse and/or first Query.	DEFINE BUTTON BUTTON-2 AUTO-GO	IMAGE-UP FILE "adeicon\run":U LABEL "Button 2" SIZE 8 BY 2.1.	"Esta Seguro de Iniciar el Proceso" BUTTON-3 AT ROW 14.33 COL 99 HELP	&ANALYZE-RESUME _END-PROCEDURE-SETTINGS
/	/ Standard List Definitions	/*	/*	*****
&Scoped-define ENABLED-OBJECTS RECT-4 RECT-5	DEFINE BUTTON BUTTON-3 AUTO-END-KEY	IMAGE-UP FILE "adeicon\exit-au":U LABEL "Button 3" SIZE 8 BY 2.1.	"Salir de este Proceso" RECT-4 AT ROW 1.81 COL 6 RECT-5 AT ROW 14.14 COL 6	/*
/	/	/*	/*	*****
&Scoped-Define RECT-4 RECT-5	IMAGE-UP FILE "adeicon\exit-au":U LABEL "Button 3" SIZE 8 BY 2.1.	DEFINE RECTANGLE RECT-4 EDGE-PIXELS 2 GRAPHIC-EDGE NO-FILL SIZE 102 BY 11.81	Listado Cuentas Alta/Canceladas Ord. Promotor" VIEW-AS TEXT	/*
k_estatus	DEFINE VARIABLE k_areaprom AS LOGICAL FORMAT	EDGE-PIXELS 2		&ANALYZE-SUSPEND _CREATE-WINDOW IF SESSION:DISPLAY -TYPE = "GUI":U THEN
k_areaprom				
k_f_ini k_f_fin				

CREATE WINDOW C-Win ASSIGN HIDDEN = YES TITLE = "Sistema de Chequera en Dólares" HEIGHT = 16.91 WIDTH = 113 MAX- HEIGHT = 27.95 MAX- WIDTH = 159.6 VIRTUAL-HEIGHT = 27.95 VIRTUAL-WIDTH = 159.6 RESIZE = yes SCROLL-BARS = no STATUS-AREA = yes BGCOLOR = ? FGCOLOR = ? KEEP- FRAME-Z-ORDER = yes THREE- D = yes MESSAGE-AREA = no SENSITIVE = yes. ELSE {&WINDOW- NAME} = CURRENT-WINDOW. /* END WINDOW DEFINITION */ &ANALYZE-RESUME /* ***** Runtime Attributes and	UIB Settings ***** */ &ANALYZE- SUSPEND_RUN- TIME-ATTRIBUTES /* SETTINGS FOR WINDOW C-Win VISIBLE,,RUN- PERSISTENT */ /* SETTINGS FOR FRAME DEFAULT- FRAME */ ASSIGN k_areaprom:AUTO -RESIZE IN FRAME DEFAULT- FRAME = TRUE. ASSIGN k_estatus:AUTO- RESIZE IN FRAME DEFAULT-FRAME = TRUE. ASSIGN k_f_fin:AUTO- RESIZE IN FRAME DEFAULT-FRAME = TRUE. ASSIGN k_f_ini:AUTO- RESIZE IN FRAME DEFAULT-FRAME = TRUE. IF SESSION:DISPLAY -TYPE = "GUI":U AND VALID- HANDLE(C-Win) THEN C- Win:HIDDEN = no. /* _RUN-TIME- ATTRIBUTES-END */ &ANALYZE-RESUME	/* ***** Control Triggers ***** */ &Scoped-define SELF-NAME C-Win &ANALYZE- SUSPEND_UIB- CODE-BLOCK CONTROL C-Win C-Win ON END-ERROR OF C-Win /* Sistema de Chequera en Dólares */ OR ENDKEY OF {&WINDOW-NAME} ANYWHERE DO: /* This case occurs when the user presses the "Esc" key. In a persistently run window, just ignore this. If we did not, the application would exit. */ IF THIS- PROCEDURE:PERSI STENT THEN RETURN NO- APPLY. END. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME &ANALYZE- SUSPEND_UIB- CODE-BLOCK CONTROL C-Win C-Win ON WINDOW-CLOSE OF C-Win /* Sistema de Chequera en Dólares */ DO: /* This event will close the	window and terminate the procedure. */ APPLY "CLOSE":U TO THIS-PROCEDURE. RETURN NO- APPLY. END. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME &Scoped-define SELF-NAME BUTTON-2 &ANALYZE- SUSPEND_UIB- CODE-BLOCK CONTROL BUTTON-2 C-Win ON CHOOSE OF BUTTON-2 IN FRAME DEFAULT- FRAME /* Button 2 */ DO: IF k_f_ini = ? OR k_f_fin = ? THEN DO: MESSAGE "Debe teclar valor en el campo Fecha". RETURN NO- APPLY. END. t_flag = FALSE. t_id_prod = "Q". t_id_esta = k_estatus. t_are_prom = k_areaprom. t_f_ini = k_f_ini. t_f_fin = k_f_fin. t_impa = TRUE. IF t_id_esta THEN t_desc_stat = " A l t a". ELSE	t_desc_stat = " C a n c e l a d a". t_cont = 0. t_reg = 0. SYSTEM-DIALOG PRINTER-SETUP UPDATE k_imprimir. IF NOT k_imprimir THEN DO: MESSAGE "IMPRESIÓN CANCELADA". OUTPUT STREAM report CLOSE. RETURN NO- APPLY. END. OUTPUT STREAM report TO PRINTER PAGED PAGE-SIZE 58. PUT STREAM report CONTROL peque. IF t_impa THEN DO: IF t_id_esta THEN /**** Aaltas ****/ IF t_are_prom THEN /**** Altas Ordenado por Area ****/ FOR EACH U_cliente WHERE U_cliente.id_pr oducto = t_id_prod AND U_cliente.stat alta = t_id_esta AND U_cliente.f_alt
--	---	---	---	--

<pre> a >= t_f_ini AND U_cliente.f_alt a <= t_f_fin BY U_cliente.id_ar ea BY U_cliente.f_alt a BY U_cliente.id_cl iente: VIEW STREAM report FRAME fr_header. kk_flag = TRUE. FIND FIRST M_promotor WHERE M_promotor.id_p romotor = U_cliente.id_pr omotor NO-ERROR. FIND FIRST M_ubicacion WHERE M_ubicacion.id_ area = U_cliente.id_ar ea AND M_ubicacion.id_ division = 0 AND M_ubicacion.id_ plaza = 0 NO-ERROR. t_cont = t_cont + 1. </pre>	<pre> t_reg = t_cont. P_f_inicial = U_cliente.f_alt a. P_f_final = U_cliente.f_alt a. P_cuenta = U_cliente.id_cl iente. P_prod = U_cliente.id_pr oducto. FIND FIRST Q_MOVIMIENTO WHERE Q_movimiento.id _producto = U_cliente.id_pr oducto AND Q_movimiento.id _cliente = U_cliente.id_cl iente. NO-ERROR. IF AVAILABLE Q_movimiento THEN p_salfin = Q_movimiento.mo nto. ELSE p_salfin = 0. t_totsdo = t_totsdo + p_salfin. DISPLAY STREAM report t_cont </pre>	<pre> U_cliente.id_pr oducto U_cliente.id_cl iente U_cliente.nombr e U_cliente.f_alt a U_cliente.f_can celacion p_salfin U_cliente.id_pr omotor M_promotor.nomb re WHEN AVAILABLE M_promotor M_ubicacion.nom bre WHEN AVAILABLE M_ubicacion WITH FRAME fr_imp. DISPLAY t_reg WITH FRAME fr_proc. PAUSE 0. DOWN STREAM report WITH FRAME fr_imp. END. ELSE /** Altas Ordenado por Promotor ***/ FOR EACH U_cliente WHERE U_cliente.id_pr oducto = t_id_prod AND U_cliente.stat </pre>	<pre> alta = t_id_esta AND U_cliente.f_alt a >= t_f_ini AND U_cliente.f_alt a <= t_f_fin BY U_cliente.id_pr omotor BY U_cliente.f_alt a BY U_cliente.id_cl iente: BY U_cliente.id_cl iente: VIEW STREAM report FRAME fr_header. kk_flag = TRUE. FIND FIRST M_promotor WHERE M_promotor.id_p romotor = U_cliente.id_pr omotor NO-ERROR. FIND FIRST M_ubicacion WHERE M_ubicacion.id_ area = U_cliente.id_ar ea AND M_ubicacion.id_ division = 0 AND M_ubicacion.id_ plaza = 0 NO-ERROR. t_cont = t_cont + 1. </pre>	<pre> t_reg = t_cont. P_f_inicial = U_cliente.f_alt a. P_f_final = U_cliente.f_alt a. P_cuenta = U_cliente.id_cl iente. P_prod = U_cliente.id_pr oducto. FIND FIRST Q_MOVIMIENTO WHERE Q_movimiento.id_ _producto = U_cliente.id_pr oducto AND Q_movimiento.id _cliente = U_cliente.id_cl iente NO-ERROR. IF AVAILABLE Q_movimiento THEN p_salfin = Q_movimiento.mo nto. ELSE p_salfin = 0. t_totsdo = t_totsdo + p_salfin. DISPLAY STREAM report t_cont </pre>
---	---	--	--	---

U_cliente.id_producto	WHERE U_cliente.id_producto =	M_ubicacion.id_plaza = 0	t_totsdo + p_salfin.	FOR EACH U_cliente
U_cliente.id_cliente	t_id_prod AND	NO-ERROR.	DISPLAY STREAM report	WHERE U_cliente.id_producto = t_id_prod AND
U_cliente.nombre	U_cliente.stat_alta = t_id_esta AND	t_cont = t_cont + 1.	t_cont	U_cliente.stat_alta = t_id_esta AND
U_cliente.f_alta	U_cliente.f_cancelacion >= t_f_ini AND	t_reg = t_cont.	U_cliente.id_producto	U_cliente.f_cancelacion >= t_f_ini AND
U_cliente.f_cancelacion	U_cliente.f_cancelacion <= t_f_fin	P_f_inicial = U_cliente.f_alta.	U_cliente.id_cliente	U_cliente.f_cancelacion <= t_f_fin
p_salfin			U_cliente.nombre	U_cliente.f_cancelacion <= t_f_fin
U_cliente.id_promotor	BY U_cliente.id_area	P_f_final = U_cliente.f_alta.	U_cliente.f_alta	BY U_cliente.id_promotor
M_promotor.nombre WHEN AVAILABLE M_promotor	BY U_cliente.f_cancelacion		U_cliente.f_cancelacion	BY U_cliente.id_promotor
M_ubicacion.nombre WHEN AVAILABLE M_ubicacion	BY U_cliente.id_cliente:	P_cuenta = U_cliente.id_cliente.	p_salfin	BY U_cliente.f_cancelacion
		P_prod = U_cliente.id_producto.	U_cliente.id_promotor	BY U_cliente.id_cliente:
WITH FRAME fr_imp.	VIEW STREAM report FRAME fr_header.	FIND FIRST Q_MOVIMIENTO	M_promotor.nombre WHEN AVAILABLE M_promotor	VIEW STREAM report FRAME fr_header.
DISPLAY t_reg	kk_flag = TRUE.	WHERE Q_movimiento.id_producto = U_cliente.id_producto AND	M_ubicacion.nombre WHEN AVAILABLE M_ubicacion	kk_flag = TRUE.
WITH FRAME fr_proc.	FIND FIRST M_promotor	Q_movimiento.id_cliente = U_cliente.id_cliente	WITH FRAME fr_imp.	FIND FIRST M_promotor
PAUSE 0.	WHERE M_promotor.id_promotor = U_cliente.id_promotor	NO-ERROR.	DISPLAY t_reg	WHERE M_promotor.id_promotor = U_cliente.id_promotor
DOWN STREAM report WITH FRAME fr_imp.	NO-ERROR.	NO-ERROR.	WITH FRAME fr_proc.	NO-ERROR.
END.			PAUSE 0.	
ELSE	FIND FIRST M_ubicacion	IF AVAILABLE Q_movimiento THEN	DOWN STREAM report WITH FRAME fr_imp.	FIND FIRST M_ubicacion
/**/ Canceladas ***/ IF t_are_prom THEN /**/ Canceladas Ordenado por Area ***/	WHERE M_ubicacion.id_area = U_cliente.id_area AND M_ubicacion.id_division = 0 AND	p_salfin = Q_movimiento.monto. ELSE p_salfin = 0.	END. ELSE /**/ Canceladas Ordenado por Promotor ***/	WHERE M_ubicacion.id_area = U_cliente.id_area AND M_ubicacion.id_division = 0 AND
FOR EACH U_cliente		t_totsdo =		

M_ubicacion.id_plaza = 0	t_totsdo + p_salfin.	DISPLAY STREAM report.	FRAME DEFAULT- FRAME /* Ordena por (Area/Promotor). */	&Scoped-define SELF-NAME k_f_ini.
NO-ERROR:	DISPLAY STREAM report	SKIP(2)	DO:	&ANALYZE- SUSPEND UIB- CODE-BLOCK.
t_cont = t_cont + 1.	t_cont U_cliente.id_pr oducto	"Total depositos iniciales : " t_totsdo SKIP	k_areaprom = INPUT k_areaprom. END.	_CONTROL k_f_ini C-Win ON LEAVE OF
t_reg = t_cont.	U_cliente.id_cl iente	"Total Promedio Depos. Iniciales: " t_prosdo	/* UIB-CODE- BLOCK-END */ &ANALYZE-RESUME	k_f_ini IN FRAME DEFAULT- FRAME /* Fecha Inicial */
P_f_inicial = U_cliente.f_alt a.	U_cliente.nombr e	WITH NO-LABELS NO-BOX FRAME fr_tot STREAM- IO.	&Scoped-define SELF-NAME k_estatus &ANALYZE- SUSPEND UIB- CODE-BLOCK _CONTROL	k_f_ini = INPUT k_f_ini.
P_f_final = U_cliente.f_alt a.	U_cliente.f_can celacion	t_totsdo = 0. PAGE STREAM report. END.	ON LEAVE OF k_estatus IN FRAME DEFAULT- FRAME /* Estatus (Alta/Cancelada)*/ DO:	END.
P_cuenta = U_cliente.id_cl iente.	p_salfin U_cliente.id_pr omotor	OUTPUT STREAM report CLOSE.	ON LEAVE OF k_estatus IN FRAME DEFAULT- FRAME /* Estatus (Alta/Cancelada)*/ DO:	/* UIB-CODE- BLOCK-END */ &ANALYZE-RESUME
P_prod = U_cliente.id_pr oducto.	M_promotor.nomb re WHEN AVAILABLE M_promotor	IF kk_flag THEN MESSAGE "FINALIZO EL LISTADO". ELSE MESSAGE "NO SE GENERO LISTADO". APPLY 'CLOSE' TO THIS-PROCEDURE. END.	Estados C-Win ON LEAVE OF k_estatus IN FRAME DEFAULT- FRAME /* Estatus (Alta/Cancelada)*/ DO:	&UNDEFINE SELF- NAME &ANALYZE- SUSPEND UIB- CODE-BLOCK _CUSTOM_MAIN- BLOCK C-Win
FIND FIRST Q_MOVIMIENTO	M_ubicacion.nomb re WHEN AVAILABLE M_ubicacion	MESSAGE "FINALIZO EL LISTADO". ELSE MESSAGE "NO SE GENERO LISTADO". APPLY 'CLOSE' TO THIS-PROCEDURE. END.	DO: k_estatus = INPUT k_estatus. END.	/* ***** ***** Main Block ***** ***** */
WHERE Q_movimiento.id _producto = U_cliente.id_pr oducto AND	M_ubicacion WITH FRAME fr_imp.	MESSAGE "NO SE GENERO LISTADO". APPLY 'CLOSE' TO THIS-PROCEDURE. END.	/* UIB-CODE- BLOCK-END */ &ANALYZE-RESUME	***** ***** Main Block ***** ***** */
Q_movimiento.id _cliente = U_cliente.id_cl iente	DISPLAY t_reg WITH FRAME fr_proc.	/* UIB-CODE- BLOCK-END */ &ANALYZE-RESUME	&Scoped-define SELF-NAME k_f_fin &ANALYZE- SUSPEND UIB- CODE-BLOCK _CONTROL k_f_fin C-Win ON LEAVE OF k_f_fin IN FRAME DEFAULT- FRAME /* Fecha final */ DO:	/* Set CURRENT- WINDOW: this will parent dialog-boxes and frames. */ ASSIGN CURRENT- WINDOW = {&WINDOW- NAME} THIS- PROCEDURE: CURRE NT-WINDOW = {&WINDOW-NAME}.
NO-ERROR.	PAUSE 0.	/* UIB-CODE- BLOCK-END */ &ANALYZE-RESUME	DO: k_f_fin = INPUT k_f_fin. END.	/* The CLOSE event can be used from inside or outside the
IF AVAILABLE Q_movimiento THEN	DOWN STREAM report WITH FRAME fr_imp.	&Scoped-define SELF-NAME k_areaprom &ANALYZE- SUSPEND UIB- CODE-BLOCK _CONTROL k_areaprom C- Win ON LEAVE OF k_areaprom IN	DO: k_f_fin = INPUT k_f_fin. END.	/* The CLOSE event can be used from inside or outside the
p_salfin = Q_movimiento.mo nto.	END.			
ELSE.	t_prosdo = t_totsdo / t_cont.			
p_salfin = 0.				
t_totsdo =				

<pre> procedure to */ /* terminate it. */ ON CLOSE OF THIS-PROCEDURE RUN disable_UI. /* Best default for GUI applications is... */ PAUSE 0 BEFORE- HIDE. /* Now enable the interface and wait for the exit condition. */ /* (NOTE: handle ERROR and END-KEY so cleanup code will always fire. */ MAIN-BLOCK: DO ON ERROR UNDO MAIN- BLOCK, LEAVE MAIN-BLOCK ON END-KEY UNDO MAIN- BLOCK, LEAVE MAIN-BLOCK: RUN enable_UI. IF NOT THIS- PROCEDURE:PERSI- STENT THEN WAIT-FOR CLOSE OF THIS- PROCEDURE. END. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME /* ***** Internal Procedures ***** */ </pre>	<pre> &ANALYZE- SUSPEND _UIB- CODE-BLOCK _PROCEDURE disable_UI C- Win_DEFAULT- DISABLE PROCEDURE disable_UI : /----- ----- ----- ----- ----- ----- ----- ----- ----- Purpose: DISABLE the User Interface Parameters: <none> Notes: Here we clean- up the user- interface by deleting dynamic widgets we have created and/or hide frames. This procedure is usually called when we are ready to "clean-up" after running. ----- ----- ----- ----- ----- ----- ----- ----- /* Delete the WINDOW we created */ IF SESSION:DISPLAY -TYPE = "GUI":U AND VALID- HANDLE(C-Win) THEN DELETE WIDGET C-Win. ----- ----- ----- ----- ----- ----- ----- ----- IF THIS- PROCEDURE:PERSI- STENT THEN DELETE PROCEDURE THIS- PROCEDURE. END PROCEDURE. </pre>	<pre> /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME &ANALYZE- SUSPEND _UIB- CODE-BLOCK _PROCEDURE enable_UI C-Win _DEFAULT-ENABLE PROCEDURE enable_UI : /----- ----- ----- ----- Purpose: ENABLE the User Interface Parameters: <none> Notes: Here we display/view/en- able the widgets in the user-interface. In addition, OPEN all queries associated with each FRAME and BROWSE. These statements here are based on the "Other Settings" section of the widget Property Sheets. ----- ----- ----- ----- ----- ----- ----- ----- ----- ----- DISPLAY k_estatus k_areaprom k_f_ini k_f_fin WITH FRAME DEFAULT- FRAME IN WINDOW C-Win. </pre>	<pre> ENABLE RECT-4 RECT-5 k_estatus k_areaprom k_f_ini k_f_fin BUTTON-2 BUTTON-3 WITH FRAME DEFAULT- FRAME IN WINDOW C-Win. {&OPEN- BROWSERS-IN- QUERY-DEFAULT- FRAME} VIEW C-Win. END PROCEDURE. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME /*Programa:c- lctaschecqcong.w */ &ANALYZE- SUSPEND _VERSION-NUMBER UIB_v8r12 GUI &ANALYZE-RESUME /* Connected Databases cheques PROGRESS */ &Scoped-define WINDOW-NAME C- Win &ANALYZE- SUSPEND _UIB- CODE-BLOCK _CUSTOM _DEFINITIONS C- Win /----- ----- ----- ----- ----- ----- ----- ----- ----- ----- File: Description: Input Parameters: <none> Output Parameters: <none> Author: </pre>	<pre> Created: ----- ----- ----- ----- ----- ----- /* This .W file was created with the Progress UIB. */ ----- ----- ----- ----- ----- ----- /* Create an unnamed pool to store all the widgets created by this procedure. This is a good default which assures that this procedure's triggers and internal procedures will execute in this procedure's storage, and that proper cleanup will occur on deletion of the procedure. */ CREATE WIDGET- POOL. /* ***** ***** Definitions ***** */ DEF VAR t_flag AS LOGICAL NO- UNDO. FORM HEADER "Sistema de Cheques en Dólares" AT 25 SKIP </pre>
--	--	--	--	---

		DEF VAR	Definitions	
"Impresion de Cuentas con Cheques Congelados"	Q_stoppay.f_stopp	t_id_producto	***** ***** /*	-{&FP1}id_producto
AT 21 SKIP	COLUMN-LABEL "Fec. !Stop"	LIKE U_cliente.id_producto.	&Scoped-define	{&FP2}id_producto -{&FP3}-
" Fecha del Listado al "	WITH FRAME fr_det_p DOWN	DEF VAR t_cuenta	PROCEDURE-TYPE Window	-{&FP1}id_cliente
AT 25	OVERLAY TITLE "CHEQUES CONGELADOS "	AS INTEGER.	/* Name of first Frame	-{&FP2}id_cliente -{&FP3}
TODAY	CONGELADOS " ROW	DEF VAR t_impan	and/or Browse	&Scoped-define
NO-LABELS	5 CENTERED	AS LOGICAL	and/or first Query	OPEN-QUERY-DEFAULT-FRAME
FORMAT "99/99/9999"	STREAM-IO.	FORMAT "Impresora/Pantalla"	*/	OPEN QUERY
"Hora : "	FORM	DEF VAR t_menus	&Scoped-define	DEFAULT-FRAME
AT 1	Q_stoppay.id_producto	AS CHARACTER	FRAME-NAME	FOR EACH
STRING(TIME, "HH:MM:SS")	COLUMN-LABEL "Pd"	FORMAT "X(20)"	DEFAULT-FRAME	U_cliente
"Pag."	"Pd"	EXTENT 2	/* Internal Tables (found by Frame, Query & Browse Queries)	SHARE-LOCK.
AT 90 PAGE-NUMBER FORM ">>9"	Q_stoppay.id_cliente	INITIAL	*/	&Scoped-define
SKIP	COLUMN-LABEL "Cliente"	["Consulta Por Cliente"]	&Scoped-define	TABLES-IN-QUERY-DEFAULT-FRAME U_cliente
FILL("_", 100)	U_cliente.nombre	"Consulta Por Fechas "]	INTERNAL-TABLES U_cliente	&Scoped-define
AT 01 FORMAT "X(100)"	COLUMN-LABEL "Nombre"	DEF VAR k_imprimir	/* Definitions for FRAME	List
WITH FRAME fr_header	Q_stoppay.cheque_inicial	AS LOGICAL NO-UNDO.	DEFAULT-FRAME	Definitions
100	COLUMN-LABEL "Cheq. Inicial"	DEF VAR peque	*/	*/
NO-BOX STREAM-IO PAGE-TOP NO-LABELS.	Q_stoppay.cheque_final	FORMAT "X(25)"	&Scoped-define	&Scoped-Define
FORM	COLUMN-LABEL "Cheq. Final"	INITIAL	FIELDS-IN-QUERY-DEFAULT-FRAME	ENABLED-FIELDS
Q_stoppay.id_producto	U_cliente.id_cliente	DEF VAR kk_flag	U_cliente.id_producto	U_cliente.id_producto
COLUMN-LABEL "Pd"	Q_stoppay.f_stopp	AS LOGICAL	U_cliente.id_cliente	U_cliente.id_cliente
Q_stoppay.id_cliente	COLUMN-LABEL "Fecha Stop"	INITIAL FALSE	U_cliente.id_cliente	&Scoped-define
COLUMN-LABEL "Cte"	WITH FRAME fr_det_i DOWN	NO-UNDO.	&Scoped-define	FIELD-PAIRS-
U_cliente.nombre	WIDTH 120	/* Parameters	ENABLED-FIELDS-IN-QUERY-DEFAULT-FRAME	-{&FP1}id_producto
U_cliente.nombre	STREAM-IO.	Definitions, ---	U_cliente.id_producto	-{&FP2}id_producto -{&FP3}-
U_cliente.nombre	DEF VAR t_fecha_1	*/	U_cliente.id_cliente	-{&FP1}id_cliente
"Nombre" FORMAT "X(40)"	AS DATE FORMAT "99/99/9999"	/* Local Variable	&Scoped-define	-{&FP2}id_cliente -{&FP3}
Q_stoppay.cheque_inicial	DEF VAR t_fecha_2	Definitions ---	ENABLED-TABLES-IN-QUERY-DEFAULT-FRAME	&Scoped-define
COLUMN-LABEL "Cheq. !Inicial"	AS DATE FORMAT "99/99/9999"	*/	U_cliente	ENABLED-TABLES
Q_stoppay.cheque_final	DEF VAR t_id_cliente	&ANALYZE-SUSPEND UIB-PREPROCESSOR-BLOCK	&Scoped-define	U_cliente
COLUMN-LABEL "Cheq. !Final"	LIKE U_cliente.id_cliente.	/*	FIRST-ENABLED-TABLE-IN-QUERY-DEFAULT-FRAME	&Scoped-define
		*****	U_cliente	ENABLED-OBJECTS
		*****	U_cliente	RECT-4 RECT-5
		Preprocessor	FIELD-PAIRS-IN-QUERY-DEFAULT-FRAME	RECT-13 RECT-14
				k_consulta
				t_fecha_ini
				t_fecha_final

<pre> BUTTON-2 BUTTON-3 &Scoped-Define DISPLAYED- FIELDS U_cliente.id_pr oducto U_cliente.id_cl iente &Scoped-Define DISPLAYED- OBJECTS k_consulta t_fecha_ini t_fecha_final /* Custom List Definitions */ /* List-1,List- 2,List-3,List- 4,List-5,List-6 */ /*_UIB- PREPROCESSOR- BLOCK-END */ &ANALYZE-RESUME /* ***** ***** Control Definitions ***** ***** */ /* Define the widget handle for the window */ DEFINE VAR C- Win AS WIDGET- HANDLE NO-UNDO. /* Definitions of the field level widgets */ DEFINE BUTTON BUTTON-2 AUTO- GO IMAGE-UP FILE "adeicon\run":U LABEL "Button 2" SIZE 8 BY 2.11. </pre>	<pre> DEFINE BUTTON BUTTON-3 AUTO- END-KEY IMAGE-UP FILE "adeicon\exit- au":U LABEL "Button 3" SIZE 8 BY 2.11. DEFINE VARIABLE k_consulta AS LOGICAL FORMAT "Cuenta/Fecha": U INITIAL YES LABEL "Que Listado Desea (Cuenta/Fecha)" VIEW-AS FILL-IN SIZE 10 BY 1 NO-UNDO. DEFINE VARIABLE t_fecha_final AS DATE FORMAT "99/99/9999":U LABEL "Fecha Final" VIEW-AS FILL-IN SIZE 16 BY 1 NO-UNDO. DEFINE VARIABLE t_fecha_ini AS DATE FORMAT "99/99/9999":U LABEL "Fecha Inicial" VIEW-AS FILL-IN SIZE 16 BY 1 NO-UNDO. DEFINE RECTANGLE RECT- 13 EDGE- PIXELS 2 GRAPHIC-EDGE NO-FILL SIZE 43 BY 4.47. DEFINE RECTANGLE RECT- 14 EDGE- PIXELS 2 </pre>	<pre> GRAPHIC-EDGE NO-FILL SIZE 48 BY 4.47. DEFINE RECTANGLE RECT- 4 EDGE- PIXELS 2 GRAPHIC-EDGE NO-FILL SIZE 104 BY 11.84. DEFINE RECTANGLE RECT- 5 EDGE- PIXELS 2 GRAPHIC-EDGE NO-FILL SIZE 104 BY 2.63. /* Query definitions */ &ANALYZE- SUSPEND DEFINE QUERY DEFAULT-FRAME FOR U_cliente SCROLLING. &ANALYZE-RESUME /* ***** ***** Frame Definitions ***** ***** */ DEFINE FRAME DEFAULT-FRAME k_consulta AT ROW 3.63 COL 55 COLON- ALIGNED HELP "Impresión por Cuentas y Cheques Congelados" AUTO-RETURN U_cliente.id_pr oducto AT ROW 7.58 COL 28 COLON-ALIGNED AUTO-RETURN </pre>	<pre> LABEL "Producto" VIEW- AS FILL-IN SIZE 8 BY 1 t_fecha_ini AT ROW 7.58 COL 76 COLON-ALIGNED HELP "Fecha Inicial" AUTO-RETURN U_cliente.id_cl iente AT ROW 8.9 COL 28 COLON-ALIGNED AUTO-RETURN LABEL "Cliente" VIEW- AS FILL-IN SIZE 12 BY 1 t_fecha_final AT ROW 8.9 COL 76 COLON- ALIGNED HELP "Fecha Final" AUTO-RETURN BUTTON-2 AT ROW 14.42 COL 90 HELP "Esta Seguro de Iniciar el Proceso" BUTTON-3 AT ROW 14.42 COL 100 HELP "Salir de este Proceso" "Fechas" VIEW-AS TEXT SIZE 8 BY .68 AT ROW 6.26 COL 78 BGCOLOR 9 FGCOLOR 14 "Cuenta" VIEW-AS TEXT SIZE 8 BY .68 AT ROW 6.26 COL 27. BGCOLOR 9 FGCOLOR 14 </pre>	<pre> RECT-4 AT ROW 1.79 COL 5 " Listado Cuentas y Cheques Congelados" VIEW-AS TEXT SIZE 43 BY 1.05 AT ROW 1.26 COL 36 BGCOLOR 9 FGCOLOR 14 RECT-5 AT ROW 14.16 COL 5 RECT-13 AT ROW 6.53 COL 11 RECT-14 AT ROW 6.53 COL 58 WITH 1 DOWN NO-BOX KEEP- TAB-ORDER OVERLAY SIDE- LABELS NO- UNDERLINE THREE-D AT COL 1 ROW 1 SIZE 112.4 BY 16.84. /* ***** ***** Procedure Settings ***** ***** */ &ANALYZE- SUSPEND _PROCEDURE- SETTINGS /* Settings for THIS-PROCEDURE Type: Window Allow: Basic,Browse,DB Fields,Window,Q uery Other Settings: COMPILE */ &ANALYZE-RESUME _END-PROCEDURE- SETTINGS /* ***** </pre>
--	--	--	--	--

```

*****
Create Window C
*****
***** */
&ANALYZE-
SUSPEND
_CREATE-WINDOW
IF
SESSION:DISPLAY
- TYPE = "GUI":U
THEN
CREATE WINDOW
C-Win ASSIGN
HIDDEN
= YES
TITLE
= "Sistema de
Chequera en
Dólares"
HEIGHT
= 16.9
WIDTH
= 113
MAX-
HEIGHT
= 27.95
MAX-
WIDTH
= 159.6
VIRTUAL-HEIGHT
= 27.95
VIRTUAL-WIDTH
= 159.6
RESIZE
= yes
SCROLL-BARS
= no
STATUS-AREA
= yes
BGCOLOR
= ?
FGCOLOR
= ?
KEEP-
FRAME-Z-ORDER =
yes
THREE-
D =
yes
MESSAGE-AREA
= no
SENSITIVE
= yes
ELSE {&WINDOW-
NAME} =
CURRENT-WINDOW.
/* END WINDOW
DEFINITION
*/
&ANALYZE-RESUME
/*
Runtime
Attributes and
UIB Settings
*/
&ANALYZE-
SUSPEND _RUN-
TIME-ATTRIBUTES
/* SETTINGS FOR
WINDOW C-Win
VISIBLE,,RUN-
PERSISTENT
*/
/* SETTINGS FOR
FRAME DEFAULT-
FRAME
*/
/* SETTINGS FOR
FILL-IN
U_cliente.id_cl
iente IN FRAME
DEFAULT-FRAME
EXP-LABEL
*/
ASSIGN
U_cliente.id_cl
iente:AUTO-
RESIZE IN FRAME
DEFAULT-FRAME
= TRUE:
/* SETTINGS FOR
FILL-IN
U_cliente.id_pr
oducto IN FRAME
DEFAULT-FRAME
EXP-LABEL
*/
ASSIGN
U_cliente.id_pr
oducto:AUTO-
RESIZE IN FRAME
DEFAULT-FRAME
= TRUE.
ASSIGN
k_consulta:AUTO
-RESIZE IN
FRAME DEFAULT-
FRAME
=
TRUE.
ASSIGN
t_fecha_final:A
UTO-RESIZE IN
FRAME DEFAULT-
FRAME
=
TRUE.
ASSIGN
t_fecha_ini:AUT
O-RESIZE IN
FRAME DEFAULT-
FRAME
=
TRUE.
IF
SESSION:DISPLAY
- TYPE = "GUI":U
AND VALID-
HANDLE(C-Win)
THEN C-
Win:HIDDEN =
no.
/* _RUN-TIME-
ATTRIBUTES-END
*/
&ANALYZE-RESUME
/* Setting
information for
Queries and
Browse Widgets
fields
*/
&ANALYZE-
SUSPEND_QUERY-
BLOCK FRAME
DEFAULT-FRAME
/* Query
rebuild
information for
FRAME DEFAULT-
FRAME
_TblList
=
"cheques:U_clie
nte"
_Query
is OPENED
*/
/* _FRAME
DEFAULT-FRAME
*/
&ANALYZE-RESUME
/*
Control
Triggers
*/
&Scoped-define
SELF-NAME C-Win
&ANALYZE-
SUSPEND UIB-
CODE-BLOCK
_CONTROL C-Win
C-Win
ON END-ERROR OF
C-Win /*
Sistema de
Chequera en
Dólares */
OR ENDKEY OF
{&WINDOW-NAME}
ANYWHERE DO:
/* This case
occurs when the
user presses
the "Esc" key.
In a
persistently
run window,
just ignore
this. If we
did not, the
application
would exit. */
IF THIS-
PROCEDURE:PERSI
STENT THEN
RETURN NO-
APPLY.
END.
/* _UIB-CODE-
BLOCK-END */
&ANALYZE-RESUME
&ANALYZE-
SUSPEND UIB-
CODE-BLOCK
_CONTROL C-Win
C-Win
ON WINDOW-CLOSE
OF C-Win /*
Sistema de
Chequera en
Dólares */
DO:
/* This event
will close the
window and
terminate the
procedure. */
APPLY
"CLOSE":U TO
THIS-PROCEDURE.
RETURN NO-
APPLY.
END.
/* _UIB-CODE-
BLOCK-END */
&ANALYZE-RESUME
&Scoped-define
SELF-NAME
BUTTON-2
&ANALYZE-
SUSPEND UIB-
CODE-BLOCK
_CONTROL
BUTTON-2 C-Win
ON CHOOSE OF
BUTTON-2 IN
FRAME DEFAULT-
FRAME /* Button
2 */
DO:
IF k_consulta
THEN DO:
IF NOT
AVAILABLE
U_cliente THEN
DO:
MESSAGE
"NO EXISTE
CUENTA".
RETURN
NO-APPLY.
END.
t_id_producto =
U_cliente.id_pr
oducto.
t_id_cliente =
U_cliente.id_cl
iente.
END.
ELSE DO:
IF
t_fecha_ini = ?
OR
t_fecha_final =
? THEN DO:
MESSAGE
"DEBES TECLLEAR
LOS CAMPOS".

```

<pre> RETURN NO-APPLY. END. t_fecha_1 = t_fecha_ini. t_fecha_2 = t_fecha_final. END. SYSTEM-DIALOG PRINTER-SETUP UPDATE k_imprimir. IF NOT k_imprimir THEN DO: MESSAGE "IMPRESIÓN CANCELADA". OUTPUT CLOSE. RETURN NO- APPLY. END. OUTPUT TO PRINTER PAGED PAGE-SIZE 58. PUT CONTROL peque. t_flag = FALSE. IF k_consulta = TRUE THEN DO: /***** CONSULTA POR CLIENTE *****/ t_cuenta = 0. FOR EACH Q_stoppay WHERE Q_stoppay.id_pr oducto = t_id_producto AND Q_stoppay.id_cl iente = t_id_cliente. </pre>	<pre> t_cuenta = t_cuenta + 1. FIND FIRST U_cliente WHERE U_cliente.id_pr oducto = Q_stoppay.id_pr oducto AND U_cliente.id_cl iente = Q_stoppay.id_cl iente NO-ERROR. IF t_impan = TRUE THEN DO: DISPLAY Q_stoppay.id_pr oducto Q_stoppay.id_cl iente U_cliente.nombr e WHEN AVAILABLE U_cliente Q_stoppay.chequ e_inicial Q_stoppay.chequ e_final WITH FRAME fr_det_i. DOWN WITH FRAME fr_det_i. END. ELSE DO: DISPLAY Q_stoppay.id_pr oducto Q_stoppay.id_cl iente U_cliente.nombr e WHEN </pre>	<pre> AVAILABLE U_cliente t_cuenta = t_cuenta + 1. FIND FIRST U_cliente WHERE U_cliente.id_pr oducto = Q_stoppay.id_pr oducto AND WITH FRAME fr_det_p. DOWN WITH FRAME fr_det_p. END. t_flag = TRUE: END. IF t_cuenta > 0 THEN DO: DISPLAY t_cuenta LABEL "Total de Cuentas" WITH FRAME fr_totche_C OVERLAY SIDE- LABELS STREAM- IO. END. OUTPUT CLOSE. END. ELSE DO: /***** CONSULTA POR FECHAS *****/ t_cuenta = 0. FOR EACH Q_stoppay WHERE Q_stoppay.f_sto p >= t_fecha_1 AND Q_stoppay.f_sto p <= t_fecha_2. </pre>	<pre> AVAILABLE U_cliente t_cuenta = t_cuenta + 1. FIND FIRST U_cliente WHERE U_cliente.id_pr oducto = Q_stoppay.id_pr oducto AND U_cliente.id_cl iente = Q_stoppay.id_cl iente NO-ERROR. IF t_impan THEN DO: DISPLAY Q_stoppay.id_pr oducto Q_stoppay.id_cl iente U_cliente.nombr e WHEN AVAILABLE U_cliente Q_stoppay.chequ e_inicial Q_stoppay.chequ e_final WITH FRAME fr_totche_F OVERLAY SIDE- LABELS STREAM- IO. END. OUTPUT CLOSE. END. OUTPUT CLOSE. IF t_flag THEN MESSAGE "FINALIZO LISTADO". ELSE MESSAGE "NO SE GENERO LISTADO". APPLY 'CLOSE' TO THIS- PROCEDURE. END. /***** _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME </pre>	<pre> AVAILABLE U_cliente Q_stoppay.chequ e_inicial Q_stoppay.chequ e_final Q_stoppay.f_sto p WITH FRAME fr_det_p. DOWN WITH FRAME fr_det_p. END. t_flag = TRUE. END. IF t_cuenta > 0 THEN DO: DISPLAY t_cuenta LABEL "Total de Cuentas" WITH FRAME fr_totche_F OVERLAY SIDE- LABELS STREAM- IO. END. OUTPUT CLOSE. END. OUTPUT CLOSE. IF t_flag THEN MESSAGE "FINALIZO LISTADO". ELSE MESSAGE "NO SE GENERO LISTADO". APPLY 'CLOSE' TO THIS- PROCEDURE. END. /***** _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME </pre>
--	--	---	--	--


```

&Scoped-define SELF-NAME
U_cliente.id_producto
&ANALYZE-SUSPEND UIB-CODE-BLOCK
CONTROL
t_fecha_final C-Win
ON LEAVE OF t_fecha_final
IN FRAME DEFAULT-FRAME /* Fecha Final
*/
DO:
t_fecha_final = INPUT t_fecha_final.
END.
/* _UIB-CODE-BLOCK-END */
&ANALYZE-RESUME

&Scoped-define SELF-NAME
k_consulta
&ANALYZE-SUSPEND UIB-CODE-BLOCK
CONTROL
k_consulta C-Win
ON LEAVE OF k_consulta IN FRAME DEFAULT-FRAME /* Que Listado Desea (Cuenta/Fecha)
*/
DO:
k_consulta = INPUT k_consulta.
END.
/* _UIB-CODE-BLOCK-END */
&ANALYZE-RESUME

&Scoped-define SELF-NAME
t_fecha_final

```

```

&ANALYZE-SUSPEND UIB-CODE-BLOCK
CONTROL
t_fecha_final C-Win
ON LEAVE OF t_fecha_final
IN FRAME DEFAULT-FRAME /* Fecha Final
*/
DO:
t_fecha_final = INPUT t_fecha_final.
END.
/* _UIB-CODE-BLOCK-END */
&ANALYZE-RESUME

&Scoped-define SELF-NAME
t_fecha_ini
&ANALYZE-SUSPEND UIB-CODE-BLOCK
CONTROL
t_fecha_ini C-Win
ON LEAVE OF t_fecha_ini IN FRAME DEFAULT-FRAME /* Fecha Inicial */
DO:
t_fecha_ini = INPUT t_fecha_ini.
IF k_consulta THEN DO:
MESSAGE "Debe teclar los Campos de Cuenta".
RETURN NO-APPLY.
END.
/* _UIB-CODE-BLOCK-END */
&ANALYZE-RESUME

&UNDEFINE SELF-NAME
&ANALYZE-SUSPEND UIB-

```

```

CODE-BLOCK
CUSTOM_MAIN-BLOCK C-Win
/*
*****
Main Block
*****
*/
/* Set CURRENT-WINDOW: this will parent dialog-boxes and frames.
*/
ASSIGN CURRENT-WINDOW = {&WINDOW-NAME}
THIS-PROCEDURE:CURRENT-WINDOW = {&WINDOW-NAME}.
/* The CLOSE event can be used from inside or outside the procedure to
*/
/* terminate it.
*/
ON CLOSE OF THIS-PROCEDURE RUN
disable_UI.
/* Best default for GUI applications is...
*/
PAUSE 0 BEFORE-HIDE.
/* Now enable the interface and wait for the exit condition.
*/
/* (NOTE: handle ERROR and END-KEY so cleanup code will always fire.
*/
MAIN-BLOCK:

```

```

DO ON ERROR
UNDO MAIN-BLOCK, LEAVE MAIN-BLOCK
ON END-KEY
UNDO MAIN-BLOCK, LEAVE MAIN-BLOCK:
RUN
enable_UI.
IF NOT THIS-PROCEDURE:PERSISTENT THEN
WAIT-FOR CLOSE OF THIS-PROCEDURE.
END.
/* _UIB-CODE-BLOCK-END */
&ANALYZE-RESUME

Internal Procedures
*****
/*
*****
*/
&ANALYZE-SUSPEND UIB-CODE-BLOCK
PROCEDURE
disable_UI C-Win DEFAULT-PROCEDURE
disable_UI :
/*-----
-----
-----
-----
-----
-----
-----
Purpose:
DISABLE the User Interface
Parameters:
<none>
Notes:
Here we clean-up the user-interface by deleting
dynamic widgets we have created and/or hide frames. This procedure is
usually called when
we are ready to "clean-up" after running.
-----
-----
-----
-----
-----
-----
-----
/* Delete the WINDOW we created */
IF SESSION:DISPLAY-TYPE = "GUI":U AND VALID-HANDLE(C-Win) THEN DELETE WIDGET C-Win.
IF THIS-PROCEDURE:PERSISTENT THEN DELETE PROCEDURE THIS-PROCEDURE.
END PROCEDURE.
/* _UIB-CODE-BLOCK-END */
&ANALYZE-RESUME

```

<pre> OPEN all queries associated with each FRAME and BROWSE. These statements here are based on the "Other Settings" section of the widget Property Sheets. ----- ----- ----- ----- ---*/ {&OPEN-QUERY- DEFAULT-FRAME} GET FIRST DEFAULT-FRAME. DISPLAY k_consulta t_fecha_ini t_fecha_final WITH FRAME DEFAULT- FRAME IN WINDOW C-Win. IF AVAILABLE U_cliente THEN DISPLAY U_cliente.id_pr oducto U_cliente.id_cl iente WITH FRAME DEFAULT- FRAME IN WINDOW C-Win. ENABLE RECT-4 RECT-5 RECT-13 RECT-14 k_consulta U_cliente.id_pr oducto t_fecha_ini U_cliente.id_cl iente t_fecha_final BUTTON-2 BUTTON-3 WITH FRAME DEFAULT- FRAME IN WINDOW C-Win. </pre>	<pre> {&OPEN- BROWSERS-IN- QUERY-DEFAULT- FRAME} VIEW C-Win. END PROCEDURE. /*_UIB-CODE- BLOCK-END */ &ANALYZE-RESUME /*Programa:c- edoct.p */ &ANALYZE- SUSPEND _VERSION-NUMBER UIB_v8r12 GUI &ANALYZE-RESUME &Scoped-define WINDOW-NAME C- Win &ANALYZE- SUSPEND _UIB- CODE-BLOCK _CUSTOM _DEFINITIONS C- Win /*----- ----- ----- ----- ----- ----- ----- ----- File: Description: Input Parameters: <none> Output Parameters: <none> Author: Created: ----- ----- ----- ----- ----- ----- ----- ----- /* */ This .W file. was created with the Progress UIB. */ /*----- ----- ----- ----- ----- </pre>	<pre> ----- ----- -----*/ /* Create an unnamed pool to store all the widgets created by this procedure. This is a good default which assures that this procedure's triggers and internal procedures will execute in this procedure's storage, and that proper cleanup will occur on deletion of the procedure. */ CREATE WIDGET- POOL. /* ***** ***** Definitions ***** */ DEFINE NEW SHARED VARIABLE P_prod LIKE U_cliente.id_pr oducto. DEFINE NEW SHARED VARIABLE P_Cuenta LIKE U_cliente.id_cl iente. DEFINE NEW SHARED VARIABLE P_F_inicial AS DATE FORMAT "99/99/9999". DEFINE NEW SHARED VARIABLE P_F_final AS DATE FORMAT "99/99/9999". DEFINE NEW SHARED VARIABLE p_SalIni LIKE </pre>	<pre> Q_Corte.Saldo_I nicial INITIAL 0. DEFINE NEW SHARED VARIABLE P_SalFin LIKE Q_Corte.Saldo_F inal INITIAL 0. DEFINE NEW SHARED VARIABLE P_SalPro LIKE Q_Corte.Saldo_P romedio INITIAL 0. DEFINE NEW SHARED VARIABLE P_n_che_rebot LIKE Q_Corte.n_chequ es_rebotados INITIAL 0. DEFINE NEW SHARED VARIABLE P_n_che_emit LIKE Q_Corte.n_chequ es_emitidos INITIAL 0. DEFINE NEW SHARED VARIABLE P_Tot_Cheques LIKE Q_Corte.Tot_Che _Emitidos INITIAL 0.. DEFINE NEW SHARED VARIABLE P_n_Debits LIKE Q_Corte.n_Debit s INITIAL 0. DEFINE NEW SHARED VARIABLE P_Tot_Debits LIKE Q_Corte.Tot_Deb its INITIAL 0. DEFINE NEW SHARED VARIABLE P_n_Credits LIKE Q_Corte.n_Credi ts INITIAL 0. DEFINE NEW SHARED VARIABLE P_Tot_Credits LIKE </pre>	<pre> Q_Corte.Tot_Cre dits INITIAL 0. DEFINE NEW SHARED VARIABLE P_Interes LIKE Q_Corte.Interes INITIAL 0. DEFINE NEW SHARED VARIABLE P_Tasa_Int LIKE Q_Corte.Tasa_In teres INITIAL 0. DEFINE STREAM report. {vars_imp.i}. define new global shared var pc_name as char no-undo format 'x(15)'. DEF VAR k_imprimir AS LOGICAL NO- UNDO. DEF VAR peque AS CHARACTER FORMAT "X(25)" INITIAL "&_l1X_(10U_(s3 T_(s0B_(s0S_(s1 0.00v16.66h0P". DEF VAR kk_flag AS LOGICAL INITIAL FALSE NO-UNDO. /* Parameters Definitions --- */ /* Local Variable Definitions --- */ /*_UIB-CODE- BLOCK-END */ &ANALYZE-RESUME &ANALYZE- SUSPEND _UIB- PREPROCESSOR- BLOCK </pre>
--	---	--	---	--

```

/*
*****
****
Preprocessor
Definitions
***** */
&Scoped-define
PROCEDURE-TYPE
Window

/* Name of
first Frame
and/or Browse
and/or first
Query
*/
&Scoped-define
FRAME-NAME
DEFAULT-FRAME

/* Standard
List
Definitions
*/
&Scoped-Define
ENABLED-OBJECTS
RECT-5 RECT-4
t_fecha_mov
t_rep1 t_rep2
t_rep3 BUTTON-2
BUTTON-3
&Scoped-Define
DISPLAYED-
OBJECTS
t_fecha_mov
t_rep1 t_rep2
t_rep3

/* Custom List
Definitions
*/
/* List-1,List-
2,List-3,List-
4,List-5,List-6
*/

/* _UIB-
PREPROCESSOR-
BLOCK-END */
&ANALYZE-RESUME

/*
*****
****
Control
Definitions
***** */
/* Define the
widget handle
for the window
*/
DEFINE VAR C-
Win AS WIDGET-
HANDLE NO-UNDO.

/* Definitions
of the field
level widgets
*/
DEFINE BUTTON
BUTTON-2 AUTO-
GO
IMAGE-UP
FILE
"adeicon\run":U
LABEL
"Button 2"
SIZE 8 BY
2.1

DEFINE BUTTON
BUTTON-3 AUTO-
END-KEY
IMAGE-UP
FILE
"adeicon\exit-
au":U
LABEL
"Button 3"
SIZE 8 BY
2.1

DEFINE VARIABLE
t_fecha_mov AS
DATE FORMAT
"99/99/9999":U
LABEL
"Fecha.Corte"
VIEW-AS
FILL-IN
SIZE 16 BY
1 TOOLTIP
"Fecha del
Listado" NO-
UNDO.

DEFINE VARIABLE
t_rep1 AS
LOGICAL FORMAT
"Si/No":U
INITIAL NO
LABEL
"Movtos. de
Edo. Cta. Fax"
VIEW-AS
FILL-IN
SIZE 4 BY
1 NO-UNDO.

DEFINE VARIABLE
t_rep2 AS
LOGICAL FORMAT
"Si/No":U
INITIAL NO
LABEL
"Otros
Movimientos"
VIEW-AS
FILL-IN
SIZE 4 BY
1 NO-UNDO.

DEFINE VARIABLE
t_rep3 AS
LOGICAL FORMAT
"Si/No":U
INITIAL NO
LABEL
"Movtos. de
Chemical"
VIEW-AS
FILL-IN
SIZE 4 BY
1 NO-UNDO.

DEFINE
RECTANGLE RECT-
4
EDGE-
PIXELS 2
GRAPHIC-EDGE
NO-FILL
SIZE 100
BY 10.81

DEFINE
RECTANGLE RECT-
5
EDGE-
PIXELS 2
GRAPHIC-EDGE
NO-FILL
SIZE,100
BY 2.62

/*
*****
****
Frame
Definitions
***** */
DEFINE FRAME
DEFAULT-FRAME
t_fecha_mov AT
ROW 4.14 COL 44
COLON-ALIGNED
t_rep1 AT
ROW 6.24 COL 39
COLON-ALIGNED
t_rep2 AT
ROW 8.62 COL 39
COLON-ALIGNED
t_rep3 AT
ROW 10.76 COL
39 COLON-
ALIGNED AUTO-
RETURN
BUTTON-2
AT ROW 13.86
COL 87
BUTTON-3
AT ROW 13.86
COL 97
RECT-5 AT
ROW 13.62 COL 7
RECT-4 AT
ROW 2.33 COL 7
Listado de
Estados de
Cuenta Global"
VIEW-AS TEXT
SIZE
41 BY 1.05 AT
ROW 1.81 COL 34
BGCOLOR 9
FGCOLOR 14
WITH 1 DOWN
NO-BOX KEEP-
TAB-ORDER
OVERLAY
SIDE-
LABELS NO-
UNDERLINE
THREE-D
AT COL
1 ROW 1
SIZE,
112.4,BY,16.84.

/*
*****
****
Procedure
Settings
***** */
&ANALYZE-
SUSPEND
PROCEDURE-
SETTINGS
/* Settings for
THIS-PROCEDURE
Type: Window
Allow:
Basic,Browse,DB
Fields,Window,Q
uery
Other
Settings:
COMPILE
*/
&ANALYZE-RESUME
END-PROCEDURE-
SETTINGS

/*
*****
****
Create Window
***** */
&ANALYZE-
SUSPEND
CREATE-WINDOW
IF
SESSION:DISPLAY
-TYPE = "GUI":U
THEN
CREATE WINDOW
C-Win ASSIGN
HIDDEN
= YES
TITLE
= "Sistema de
Chequera en
Dólares"
HEIGHT
= 16.24
WIDTH
= 112.4
MAX-
HEIGHT
= 17.81
MAX-
WIDTH
= 112.4
VIRTUAL-HEIGHT
= 17.81
VIRTUAL-WIDTH
= 112.4
RESIZE
= yes
SCROLL-BARS
= no
STATUS-AREA
= yes
BGCOLOR
= ?
FGCOLOR
= ?

```

<pre> KEEP- FRAME-Z-ORDER = yes THREE- D = yes MESSAGE-AREA = no SENSITIVE = yes. ELSE {&WINDOW- NAME} = CURRENT-WINDOW. /* END WINDOW DEFINITION */ &ANALYZE-RESUME /* ***** Runtime Attributes and UIB Settings ***** */ &ANALYZE- SUSPEND _RUN- TIME-ATTRIBUTES /* SETTINGS FOR WINDOW C-Win VISIBLE,,RUN- PERSISTENT */ /* SETTINGS FOR FRAME DEFAULT- FRAME */ ASSIGN t_fecha_mov:AUT O-RESIZE IN FRAME DEFAULT- FRAME = TRUE. ASSIGN t_rep1:AUTO- RESIZE IN FRAME DEFAULT-FRAME = TRUE. ASSIGN t_rep2:AUTO- RESIZE IN FRAME DEFAULT-FRAME = TRUE. </pre>	<pre> ASSIGN t_rep3:AUTO- RESIZE IN FRAME DEFAULT-FRAME = TRUE. IF SESSION:DISPLAY- TYPE = "GUI":U AND VALID- HANDLE(C-Win) THEN C- Win:HIDDEN = no. /* _RUN-TIME- ATTRIBUTES-END */ &ANALYZE-RESUME Control Triggers ***** */ &Scoped-define SELF-NAME C-Win &ANALYZE- SUSPEND _UIB- CODE-BLOCK _CONTROL C-Win C-Win ON END-ERROR OF C-Win /* Sistema de Chequera en Dólares */ OR ENDKEY OF {&WINDOW-NAME} ANYWHERE DO: /* This case occurs when the user presses the "Esc" key. In a persistently run window, just ignore this. If we did not, the application would exit. */ </pre>	<pre> IF THIS- PROCEDURE:PERSI STENT THEN RETURN NO- APPLY. END. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME &ANALYZE- SUSPEND _UIB- CODE-BLOCK _CONTROL C-Win C-Win ON WINDOW-CLOSE OF C-Win /* Sistema de Chequera en Dólares */ DO: /* This event will close the window and terminate the procedure. */ APPLY "CLOSE":U TO THIS-PROCEDURE. RETURN NO- APPLY. END. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME &Scoped-define SELF-NAME BUTTON-2 &ANALYZE- SUSPEND _UIB- CODE-BLOCK _CONTROL BUTTON-2 C-Win ON CHOOSE OF BUTTON-2 IN FRAME DEFAULT- FRAME /* Button 2 */ DO: IF t_fecha_mov = ? THEN DO: MESSAGE "Debe capturar los Campos". RETURN NO- APPLY. END. </pre>	<pre> DEFINE VARIABLE l_rep1 AS LOGICAL FORMAT "Si/No". DEFINE VARIABLE l_rep2 AS LOGICAL FORMAT "Si/No". DEFINE VARIABLE l_rep3 AS LOGICAL FORMAT "Si/No". DEFINE VARIABLE f_corte AS DATE FORMAT "99/99/9999" NO-UNDO. DEFINE VARIABLE x_credit LIKE Q_Movimiento.mo nto NO-UNDO. DEFINE VARIABLE x_debit LIKE Q_Movimiento.mo nto NO-UNDO. DEFINE VARIABLE Tot_credits LIKE Q_Movimiento.mo nto NO-UNDO. DEFINE VARIABLE Tot_debits LIKE Q_Movimiento.mo nto NO-UNDO. DEFINE VARIABLE t_credit LIKE Q_Movimiento.mo nto NO-UNDO. DEFINE VARIABLE t_debit LIKE Q_Movimiento.mo nto NO-UNDO. </pre>	<pre> DEFINE VARIABLE t_trasp LIKE Q_Movimiento.mo nto NO-UNDO. DEFINE VARIABLE t_saldo_inic LIKE Q_Movimiento.mo nto NO-UNDO. DEFINE VARIABLE t_saldo_final LIKE Q_Movimiento.mo nto NO-UNDO. DEFINE VARIABLE t_credits_cte LIKE Q_Movimiento.mo nto NO-UNDO. DEFINE VARIABLE t_credits_che LIKE Q_Movimiento.mo nto NO-UNDO. DEFINE VARIABLE t_debits_cte LIKE Q_Movimiento.mo nto NO-UNDO. DEFINE VARIABLE t_debits_che LIKE Q_Movimiento.mo nto NO-UNDO. DEFINE VARIABLE n_Credits AS INTEGER FORMAT ">>>>9" NO-UNDO. DEFINE VARIABLE n_Debits AS INTEGER FORMAT ">>>>9" NO-UNDO. </pre>

<pre> DEFINE VARIABLE x_nombre AS CHARACTER FORMAT "X(25)" NO-UNDO. DEFINE VARIABLE t_titulo AS CHARACTER FORMAT "X(45)" NO-UNDO. DEFINE BUFFER BQ_saldo_chem FOR Q_saldo_chem. FORM HEADER "Institucion Financiera S.A." AT 50 SKIP. t_titulo AT 49 SKIP (1) "Fecha de emision : " AT 1 TODAY COLON 22 FORMAT "99/99/9999" STRING(TIME, "HH :MM:SS") COLON 36 "Fecha Reporte del " AT 1 f_corte AT 22 "PAG." AT 121 PAGE- NUMBER(report) COLON 126 FORMAT 999 FILL("_", 130) AT 1 FORMAT "X(130)" WITH WIDTH 132. PAGE-TOP </pre>	<pre> NO-LABELS FRAME fp_header STREAM-IO. FORM Q_Movimiento.f_ movto AT 1 COLUMN- LABEL "Date" Q_movimiento:id_ producto AT 14 COLUMN- LABEL 'Prd' Q_Movimiento.id_ cliente AT 17 COLUMN- LABEL " Acc." x_nombre AT 25 COLUMN- LABEL "Name" Q_Movimiento.id_ movto AT 51 COLUMN- LABEL "Id." Q_Tpmovto.Descr ip_Larga AT 57 COLUMN- LABEL "Description" Q_Movimiento.Re ferencia AT 83 COLUMN- LABEL "Refer" x_Debit AT 91 COLUMN- LABEL "Debits", x_Credit AT 111 COLUMN- LABEL "Credits" FRAME WITH fr_detalle WIDTH 132. DOWN STREAM-IO. FORM Q_Mov_chem.f_mo </pre>	<pre> vto AT 1 NO-LABEL Q_Mov_chem:id_p roducto AT 14 NO-LABEL x_nombre AT 25 NO- LABEL Q_Mov_chem:id_m ovto AT 51 NO-LABEL Q_Tpmovto.Descr ip_Larga AT 57 NO-LABEL Q_Mov_chem:Refe rencia AT 83 NO-LABEL x_Debit AT 91 NO- LABEL x_Credit AT 111 NO- LABEL FRAME WITH fr_detalle2 WIDTH 132 DOWN STREAM-IO. FORM SKIP (2) 'Initial Balance AT 10 P_SalIni AT 30 n_Credits AT 1 'Credits' AT 10 Tot_Credits AT 30 n_Debits AT 1 'Debits' AT 10 Tot_Debits AT 30 'Current Balance </pre>	<pre> AT 10 P_SalFin AT 30 WITH FRAME-fr_total NO-LABELS WIDTH 55 COLUMN, 40 STREAM-IO. FORM SKIP(2) 'Initial Balance ' AT 10 t_saldo_inic 'Credit ' AT 10 t_credit 'Debit ' AT 10 t_debit 'Current Balance ' AT 10 t_saldo_final WITH FRAME fr_chem400 NO-LABELS WIDTH 55 COLUMN 40 STREAM-IO. l_rep1 = t_rep1. l_rep2 = t_rep2. l_rep3 = t_rep3. SYSTEM- DIALOG PRINTER- SETUP UPDATE k_imprimir. IF NOT k_imprimir THEN DO: MESSAGE "IMPRESIÓN CANCELADA". OUTPUT STREAM report CLOSE. RETURN NO-APPLY. END. </pre>	<pre> OUTPUT STREAM report TO PRINTER PAGED PAGE-SIZE 58. PUT STREAM report CONTROL peque. f_corte = t_fecha_mov. IF l_rep1 = TRUE OR l_rep2 = TRUE OR l_rep3 = TRUE THEN DO: FIND FIRST Q_Parametro NO- LOCK NO-ERROR. /* Rep. 1 */ IF l_rep1 = TRUE THEN DO: p_f_inicial = f_corte. p_f_final = f_corte. p_cuenta = Q_Parametro.cue nta_global. /* Obtener el Saldo Inicial */ p_prod = 'Q' RUN. qfestglo.r. n_Debits = 0. n_Credits = 0. Tot_Credits = 0. Tot_Debits = 0. </pre>
--	---	--	--	---

```

/*MESSAGE "Mov.
Cta. Global"*/
t_titulo =
"REPORTE DE
ESTADO DE
CUENTA GLOBAL".
FOR
EACH
Q_Movimiento
WHERE
Q_Movimiento.f_
movto = f_corte
NO-LOCK:
FIND Q_tpmovto
WHERE
Q_tpmovto.id_mo
vto =
Q_Movimiento.id_
movto
NO-LOCK NO-
ERROR.
IF
Q_tpmovto.gener
ado = "CH" THEN
NEXT.
IF
Q_tpmovto.afec_
cta_global THEN
DO:
VIEW STREAM
report FRAME
fp_header.
FIND FIRST
U_cliente
WHERE
U_cliente.id_pr
oducto =
Q_Movimiento.id_
producto AND
U_cliente.id_cl
iente =
Q_Movimiento.id_
cliente
Q_Movimiento.f_
movto
Q_movimiento.id_
producto
Q_Movimiento.id_
cliente
P_Salfin WITH
FRAME fr_total
STREAM-IO.
END.
END. /*
l_repl */
IF l_rep2 =
TRUE THEN DO:
/*-----
----- Otros
Movimientos ---
-----*/
n_Debits
= 0.
n_Credits =
0.
Tot_Credits =
0.
Tot_Debits =
0.
PAGE
STREAM report.
MESSAGE
"Mov. Otros ".
t_titulo
= "REPORTE DE
OTROS
MOVIMIENTOS".
FOR EACH
Q_Movimiento
WHERE
Q_Movimiento.f_
movto = f_corte
NO-
LOCK:
FIND
FIRST Q_tpmovto
WHERE
Q_tpmovto.id_mo
vto =
Q_Movimiento.id_
movto
NO-LOCK NO-
ERROR.
Q_Movimiento.id_
cliente
NO-LOCK NO-
ERROR.
IF
Q_tpmovto.Natur
aleza = "C"
THEN DO:
x_Credit =
Q_Movimiento.mo
nto.
x_Debit =
Q_Movimiento.mo
nto.
n_Debits =
n_Debits + 1.
Tot_Debits =
Tot_Debits +
Q_Movimiento.mo
nto.
END.
ELSE DO:
x_Credit = 0.
x_Debit =
Q_Movimiento.mo
nto.
n_Debits =
n_Debits + 1.
Tot_Debits =
Tot_Debits +
Q_Movimiento.mo
nto.
END.
/* FOR EACH
Q_Movimiento */
IF
n_credits <> 0
OR n_debits <>
0 THEN DO:
P_SalFin =
P_Salini +
Tot_Credits -
Tot_Debits.
DISPLAY STREAM
report
P_SalIni
n_Credits
Tot_Credits
Q_Movimiento.mo
nto.
x_Debit =
0.
END.
ELSE DO:
x_Credit = 0.
x_Debit =
0.

```

```

Q_Movimiento.mo
nto.
n_Debits =
n_Debits + 1.
Tot_Debits =
Tot_Debits +
Q_Movimiento.mo
nto.
END.

IF AVAILABLE
U_cliente THEN
x_nombre =
SUBSTR(U_cliente.nombre,1,25).
ELSE
x_nombre =
FILL(' ',25).

DISPLAY STREAM
report

Q_Movimiento.f_
movto

Q_movimiento.id
_producto

Q_Movimiento.id
_cliente

x_nombre

Q_Movimiento.id
_movto

Q_tpmovto.Descr
ip_Larga

Q_Movimiento.re
ferencia

x_Debit WHEN
x_debit <> 0

x_Credit WHEN
x_Credit <> 0

WITH FRAME
fr_detalle.

DOWN STREAM
report WITH

```

```

FRAME
fr_detalle.
END.
/* fin del if
afecta_cta_glob
al*/
END. /*
fin del for
each
q_movimiento */

IF
n_credits <> 0
OR n_debits <>
0 THEN DO:
/*
P_SalFin =
P_Salini +
Tot_Credits -
Tot_Debits.
*/
DISPLAY STREAM
report
n_Credits
Tot_Credits
n_Debits
Tot_Debits
WITH FRAME
fr_totall
STREAM-IO.
END. /*
l_rep2 */

IF l_rep3 =
TRUE THEN DO:
/*
Movimientos
Chemical cta.
400 -----*/
t_debits_cte =
0.
t_credits_cte =
0.
n_Debits
= 0.
n_Credits
= 0.

```

```

Tot_Credits =
0.
Tot_Debits =
0.
PAUSE 0.
PAGE
STREAM report:
MESSAGE
"Mov.
Chemical".
t_titulo
= "REPORTE EDO.
DE CUENTA
CHEMICAL 400".
FOR EACH
Q_Movimiento
WHERE
Q_Movimiento.f_
movto = f_corte.
NO-
LOCK:
FIND
FIRST Q_tpmovto
WHERE
Q_tpmovto.id_mo
vto =
Q_Movimiento.id
_movto
NO-LOCK NO-
ERROR.
IF
Q_tpmovto.gener
ado = "CH"
THEN DO:
VIEW STREAM
report FRAME
fp_header.
FIND FIRST
U_cliente
WHERE
U_cliente.id_pr
oducto =
Q_Movimiento.id
_producto AND
U_cliente.id_cl
iente =

```

```

Q_Movimiento.id
_cliente
IF AVAILABLE
U_cliente THEN
x_nombre =
SUBSTR(U_cliente.nombre,1,25).
ELSE
x_nombre =
FILL(' ',25).
DISPLAY STREAM
report
Q_Movimiento.f_
movto
Q_movimiento.id
_producto
Q_Movimiento.id
_cliente
x_nombre
Q_Movimiento.id
_movto
Q_tpmovto.Descr
ip_Larga
Q_Movimiento.re
ferencia
x_Debit WHEN
x_debit <> 0
x_Credit WHEN
x_Credit <> 0
WITH FRAME
fr_detalle.
DOWN STREAM
report WITH

```

```

Q_Movimiento.id
_cliente
IF AVAILABLE
U_cliente THEN
x_nombre =
SUBSTR(U_cliente.nombre,1,25).
ELSE
x_nombre =
FILL(' ',25).
DISPLAY STREAM
report
Q_Movimiento.f_
movto
Q_movimiento.id
_producto
Q_Movimiento.id
_cliente
x_nombre
Q_Movimiento.id
_movto
Q_tpmovto.Descr
ip_Larga
Q_Movimiento.re
ferencia
x_Debit WHEN
x_debit <> 0
x_Credit WHEN
x_Credit <> 0
WITH FRAME
fr_detalle.
DOWN STREAM
report WITH

```

```

/* jmcr
17/07/95 */
t_debits_che = 0.
t_credits_che = 0.
FOR EACH Q_Mov_chem
WHERE
Q_Mov_chem.f_mvto = f_corte
NO-LOCK:
FIND FIRST Q_tpmovto
WHERE
Q_tpmovto.id_mvto =
Q_Mov_chem.id_mvto
NO-LOCK NO-ERROR.
IF Q_tpmovto.generado = "CH"
THEN DO:
VIEW STREAM report FRAME fp_header.
FIND FIRST U_cliente
WHERE
U_cliente.id_producto =
Q_Mov_chem.id_producto AND
U_cliente.id_cliente =
Q_Mov_chem.id_cliente
NO-LOCK NO-ERROR.
IF Q_tpmovto.Natur

```

<pre> aleza = "C" THEN do: x_Credit = Q_Mov_chem.monto. n_Credits = Q_Movimiento.f_mvto. Tot_Credits = Q_Mov_chem.id_producto @ Q_Movimiento.id_producto Tot_Credits + Q_Movimiento.id_producto Q_Mov_chem.id_cliente @ Q_Movimiento.id_cliente x_Debit = 0. END. ELSE DO: x_Credit = 0. x_Debit = Q_Mov_chem.monto. n_Debits = x_Debit n_Debits + 1. Tot_Debits = x_Credit Tot_Debits + Q_Mov_chem.monto. t_debits_che = t_debits_che + Q_Mov_chem.monto. END. IF AVAILABLE U_cliente THEN x_nombre = SUBSTR(U_cliente.nombre,1,25). ELSE x_nombre = FILL(' ',25). </pre>	<pre> DISPLAY STREAM report Q_Mov_chem.f_mvto @ Q_Movimiento.f_mvto Q_Mov_chem.id_producto @ Q_Movimiento.id_producto Q_Mov_chem.id_cliente @ Q_Movimiento.id_cliente Q_Mov_chem.id_mvto @ Q_Movimiento.id_mvto Q_Tpmovto.Descrip_Larga Q_Mov_chem.referencia @ Q_Movimiento.referencia x_Debit WHEN x_debit <> 0 x_Credit WHEN x_Credit <> 0 WITH FRAME fr_detalle. DOWN STREAM report WITH FRAME fr_detalle. END. /* fin del if afecta_cta_global*/ END. /* fin del for-each q_mov_chem */ </pre>	<pre> /* jmcr */ IF n_credits <> 0 OR n_debits <> 0 THEN DO: DISPLAY STREAM report n_Credits Tot_Credits n_Debits Tot_Debits WITH FRAME fr_totall STREAM-IO. FIND FIRST Q_saldo_chem WHERE Q_saldo_chem.f_mvto = f_corte NO-ERROR. t_saldo_inic = Q_saldo_chem.saldo_inic. t_saldo_final = Q_saldo_chem.saldo_final. t_credit = Q_saldo_chem.creditos. t_debit = Q_saldo_chem.debitos. DISPLAY STREAM report t_saldo_inic t_credit t_debit t_saldo_final WITH FRAME fr_chem400 STREAM-IO. IF t_credit <> t_credits_cte + t_credits_che THEN END. /* l_rep3 */ kk_flag = TRUE. END. /* l_rep1 = TRUE. </pre>	<pre> t_credits_che THEN Q_saldo_chem.debitos = t_debits_cte + t_debits_che. Q_saldo_chem.saldo_final = Q_saldo_chem.saldo_inic + Q_saldo_chem.creditos - Q_saldo_chem.debitos. t_saldo_final = Q_saldo_chem.saldo_final. t_credit = Q_saldo_chem.creditos. t_debit = Q_saldo_chem.debitos. </pre>
--	---	---	--

<pre>OR l_rep2 = TRUE OR l_rep3 = TRUE */ IF kk_flag THEN MESSAGE "FINALIZO EL LISTADO". ELSE MESSAGE "NO SE GENERO LISTADO". OUTPUT STREAM report CLOSE. APPLY 'CLOSE' TO THIS-PROCEDURE. END.../*DO*/ /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME &Scoped-define SELF-NAME t_fecha_mov &ANALYZE- SUSPEND _UIB- CODE-BLOCK _CONTROL t_fecha_mov C- Win ON LEAVE OF t_fecha_mov IN FRAME DEFAULT- FRAME /* Fecha Corte */ DO: t_fecha_mov = INPUT t_fecha_mov. END. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME &Scoped-define SELF-NAME t_rep1 &ANALYZE- SUSPEND _UIB- CODE-BLOCK _CONTROL t_rep1 C-win ON LEAVE OF t_rep1 IN FRAME</pre>	<pre>DEFAULT-FRAME /* Movtos. de Edo. Cta. Fax */ DO: t_rep1 = INPUT t_rep1. END. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME &Scoped-define SELF-NAME t_rep2 &ANALYZE- SUSPEND _UIB- CODE-BLOCK. _CONTROL t_rep2 C-win ON LEAVE OF t_rep2 IN FRAME DEFAULT-FRAME /* Otros Movimientos */ DO: t_rep2 = INPUT t_rep2. END. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME &Scoped-define SELF-NAME t_rep3 &ANALYZE- SUSPEND _UIB- CODE-BLOCK _CONTROL t_rep3 C-win ON LEAVE OF t_rep3 IN FRAME DEFAULT-FRAME /* Movtos. de Chemical */ DO: t_rep3 = INPUT t_rep3. END. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME &UNDEFINE SELF- NAME</pre>	<pre>&ANALYZE- SUSPEND _UIB- CODE-BLOCK _CUSTOM MAIN- BLOCK C-win /* ***** Main Block ***** */ /* Set CURRENT- WINDOW: this will parent dialog-boxes and frames. */ ASSIGN CURRENT- WINDOW = {&WINDOW- NAME} THIS- PROCEDURE:CURRE NT-WINDOW = {&WINDOW-NAME}. /* The CLOSE event can be used from inside or outside the procedure to */ /* terminate it. */ ON CLOSE OF THIS-PROCEDURE RUN disable_UI. /* Best default for GUI applications is... */ PAUSE 0 BEFORE- HIDE. /* Now enable the interface and wait for the exit condition. */ /* (NOTE: handle ERROR and END-KEY so cleanup code</pre>	<pre>will always fire. */ MAIN-BLOCK: DO ON ERROR UNDO MAIN- BLOCK, LEAVE MAIN-BLOCK ON END-KEY. UNDO MAIN- BLOCK, LEAVE MAIN-BLOCK: RUN enable_UI. IF NOT THIS- PROCEDURE:PERSI STENT THEN WAIT-FOR CLOSE OF THIS- PROCEDURE. END. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME Internal Procedures ***** */ &ANALYZE- SUSPEND _UIB- CODE-BLOCK _PROCEDURE disable_UI C- Win DEFAULT- DISABLE PROCEDURE disable_UI : /*----- ----- ----- Purpose: DISABLE the User Interface Parameters: <none> Notes: Here we clean- up the user- interface by deleting dynamic widgets we have created and/or hide</pre>	<pre>frames. This procedure is usually called when we are ready to "clean-up" after running. ----- ----- ----- /* Delete the WINDOW we created */ IF SESSION:DISPLAY -TYPE = "GUI":U AND VALID- HANDLE(C-win) THEN DELETE WIDGET C-win. IF THIS- PROCEDURE:PERSI STENT THEN DELETE PROCEDURE THIS- PROCEDURE. END. PROCEDURE. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME &ANALYZE- SUSPEND _UIB- CODE-BLOCK _PROCEDURE enable_UI C-win _DEFAULT-ENABLE PROCEDURE enable_UI : /*----- ----- ----- Purpose: ENABLE the User Interface Parameters: <none> Notes: Here we display/view/en able the widgets in the</pre>
--	--	---	--	--

<pre> user-interface. In addition, OPEN all queries associated with each FRAME and BROWSE. These statements here are based on the "Other Settings" section of the widget Property Sheets. ----- Input Parameters: <none> ----- Output Parameters: <none> ----- Author: Created: ----- DISPLAY t_fecha_mov t_repl t_rep2 t_rep3 WITH FRAME DEFAULT- FRAME IN WINDOW C-Win. ENABLE RECT-5 RECT-4 t_fecha_mov t_repl t_rep2 t_rep3 BUTTON-2 BUTTON-3 WITH FRAME DEFAULT- FRAME IN WINDOW C-Win. {&OPEN- BROWSERS-IN- QUERY-DEFAULT- FRAME} VIEW C-Win. END PROCEDURE. /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME /*Programa:c- linggasuti.p*/ &ANALYZE- SUSPEND _VERSION-NUMBER UIB_v8r12 GUI &ANALYZE-RESUME </pre>	<pre> &Scoped-define WINDOW-NAME C- Win &ANALYZE- SUSPEND _UIB- CODE-BLOCK _CUSTOM _DEFINITIONS C- Win /*----- ----- ----- ----- ----- ----- File: Description: Input Parameters: <none> ----- Output Parameters: <none> ----- Author: Created: ----- /* Create an unnamed pool to store all the widgets created by this procedure. This is a good default which assures that this procedure's triggers and internal procedures </pre>	<pre> will execute in this procedure's storage, and that proper cleanup will occur on deletion of the procedure. */ CREATE WIDGET- POOL. /* ***** ***** Definitions ***** ***** */ DEFINE VARIABLE F_Inicial AS DATE FORMAT "99/99/9999". DEFINE VARIABLE F_Final AS DATE FORMAT "99/99/9999". DEFINE VARIABLE F_Hoy AS DATE FORMAT "99/99/9999". DEFINE VARIABLE Tot_Ingresos LIKE U_cliente.Saldo _Total INITIAL 0. DEFINE VARIABLE Tot_Gastos LIKE U_cliente.Saldo _Total INITIAL 0. DEFINE VARIABLE x_descrip AS CHARACTER FORMAT "X(30)". DEFINE VARIABLE x_monto LIKE U_cliente.Saldo _Total. DEF VAR k_imprimir AS LOGICAL NO- UNDO. DEF VAR peque AS CHARACTER FORMAT "X(25)" INITIAL "&llx_(10U_(s3 T_(s0B_(s0S_(s1 0.00v16.66h0P". </pre>	<pre> DEF VAR kk_flag AS LOGICAL INITIAL FALSE NO-UNDO. DEFINE STREAM report. FORM HEADER "Sistema de Cheques en Dólares" AT 19 SKIP "REPORTE DE INGRESOS GASTOS Y UTILIDADES" AT 21 SKIP (1) "Fecha de emision : " AT 1 TODAY COLON 22 FORMAT "99/99/9999" STRING(TIME, "HH :MM:SS") COLON 34 "Fecha Reporte del " AT 1 F_Inicial AT 22 " al " F_Final AT 39 "PAG." AT 70 PAGE- NUMBER(report). COLON 76 FORMAT 999 FILL("_",79) AT 1 FORMAT "X(79)" WITH WIDTH 80 TOP-ONLY LABELS NO- NO- FRAME fp_header. FORM SKIP (2) x_descrip AT 2 WITH NO- LABELS </pre>	<pre> NO- BOX STREAM-IO FRAME fp_letrero. FORM x_descrip AT 5 x_monto AT 40 WITH NO- LABELS NO- BOX DOWN STREAM-IO FRAME fp_detalle. FORM SKIP (1) x_descrip AT 2 x_monto AT 60 WITH NO- LABELS NO- BOX DOWN STREAM-IO FRAME fp_total. /* Parameters Definitions --- */ /* Local Variable Definitions --- */ /* _UIB-CODE- BLOCK-END */ &ANALYZE-RESUME &ANALYZE- SUSPEND _UIB- PREPROCESSOR- BLOCK /* ***** ***** Preprocessor Definitions ***** */ &Scoped-define PROCEDURE-TYPE Window </pre>
--	--	---	--	--

```

/* Name of first Frame and/or Browse and/or first Query
*/
&Scoped-define FRAME-NAME
DEFAULT-FRAME

/* Standard List Definitions
*/
&Scoped-Define ENABLED-OBJECTS
RECT-5 RECT-4
t_fecha_ini
t_fecha_final
BUTTON-2
BUTTON-3
&Scoped-Define DISPLAYED-OBJECTS
t_fecha_ini
t_fecha_final

/* Custom List Definitions
*/
/* List-1,List-2,List-3,List-4,List-5,List-6
*/

/* _UIB-PREPROCESSOR-BLOCK-END-*/
&ANALYZE-RESUME

/*
*****
*****
Control Definitions
*****
*/

/* Define the widget handle for the window
*/
DEFINE VAR C-Win AS-WIDGET-HANDLE NO-UNDO.

/* Definitions of the field level widgets
*/
DEFINE BUTTON
BUTTON-2 AUTO-GO
IMAGE-UP
FILE "adeicon\run":U
LABEL "Button 2"
SIZE 8 BY 2.1.
DEFINE BUTTON
BUTTON-3 AUTO-END-KEY
IMAGE-UP
FILE "adeicon\exit":U
LABEL "Button 3"
SIZE 8 BY 2.1.
DEFINE VARIABLE
t_fecha_ini AS DATE FORMAT "99/99/9999":U
LABEL "Fecha Final"
VIEW-AS FILL-IN
SIZE 16 BY 1 NO-UNDO.
DEFINE VARIABLE
t_fecha_ini AS DATE FORMAT "99/99/9999":U
LABEL "Fecha Inicial"
VIEW-AS FILL-IN
SIZE 16 BY 1 NO-UNDO.
DEFINE RECTANGLE
RECT-4
EDGE-PIXELS 2
GRAPHIC-EDGE NO-FILL
SIZE 101 BY 11.52.
DEFINE RECTANGLE
RECT-5
EDGE-PIXELS 2
GRAPHIC-EDGE NO-FILL
SIZE 101 BY 11.52.
SIZE 101 BY 2.62.
TAB-ORDER OVERLAY
SIDE-LABELS NO-UNDERLINE
THREE-D AT COL 1 ROW 1
SIZE 112.4 BY 16.84.
DEFINE FRAME
DEFAULT-FRAME
t_fecha_ini AT ROW 5.19 COL 39
COLON-ALIGNED HELP
"Fecha Inicial del reporte"
AUTO-RETURN
t_fecha_final AT ROW 6.81 COL 39
COLON-ALIGNED HELP
"Fecha Final del Reporte"
AUTO-RETURN
BUTTON-2 AT ROW 14.33 COL 90
HELP "Esta Seguro de Iniciar el Proceso"
BUTTON-3 AT ROW 14.33 COL 99
HELP "Salir de este Proceso"
RECT-5 AT ROW 14.14 COL 7
RECT-4 AT ROW 2.33 COL 7
LISTADO
Ingresos Gastos y Utilidad"
VIEW-AS TEXT
SIZE 38 BY 1.05 AT ROW 1.81 COL 36
BGCOLOR 9
FGCOLOR 14
WITH 1 DOWN NO-BOX KEEP-
TAB-ORDER OVERLAY
SIDE-LABELS NO-UNDERLINE
THREE-D AT COL 1 ROW 1
SIZE 112.4 BY 16.84.
/*
*****
*****
Procedure Settings
*****
*/
&ANALYZE-SUSPEND
PROCEDURE-SETTINGS
/* Settings for THIS-PROCEDURE
Type: Window
Allow: Basic, Browse, DB Fields, Window, Query
Other:
Settings:
COMPILE
*/
&ANALYZE-RESUME
END-PROCEDURE-SETTINGS
/*
*****
*****
Create Window
*****
*/
&ANALYZE-SUSPEND
CREATE-WINDOW
IF SESSION:DISPLAY
TYPE = "GUI":U
THEN
CREATE WINDOW C-Win ASSIGN
HIDDEN
TITLE = "Sistema de Chequera en Dólares"
HEIGHT = 16.43
WIDTH = 111.4
MAX-HEIGHT = 27.95
MAX-WIDTH = 159.6
VIRTUAL-HEIGHT = 27.95
VIRTUAL-WIDTH = 159.6
RESIZE = yes
SCROLL-BARS = no
STATUS-AREA = yes
BGCOLOR = ?
FGCOLOR = ?
KEEP-FRAME-Z-ORDER = yes
THREE-COMPILE = yes
MESSAGE-AREA = no
SENSITIVE = yes
ELSE {&WINDOW-NAME} = CURRENT-WINDOW
/* END WINDOW DEFINITION
*/
&ANALYZE-RESUME
/*
*****
*****
Runtime Attributes and UIB Settings
*****
*/
&ANALYZE-SUSPEND
RUN-TIME-ATTRIBUTES

```

```

/* SETTINGS FOR WINDOW C-Win
VISIBLE,,RUN-
PERSISTENT
*/
/* SETTINGS FOR
FRAME DEFAULT-
FRAME
*/
ASSIGN

t_fecha_final:A
UTO-RESIZE IN
FRAME DEFAULT-
FRAME =
TRUE.

ASSIGN

t_fecha_ini:AUT
O-RESIZE IN
FRAME DEFAULT-
FRAME =
TRUE.

IF
SESSION:DISPLAY
-TYPE = "GUI":U
AND VALID-
HANDLE(C-Win)
THEN C-
Win:HIDDEN =
no.

/* _RUN-TIME-
ATTRIBUTES-END
*/
&ANALYZE-RESUME

Chequera en
Dólares */
OR ENDKEY OF
{&WINDOW-NAME}
ANYWHERE DO:
/* This case
occurs when the
user presses
the "Esc" key.
In a
persistently
run window,
just ignore
this. If we
did not, the
application
would exit. */
IF THIS-
PROCEDURE:PERSI
STENT THEN
RETURN NO-
APPLY.
END.

/* _UIB-CODE-
BLOCK-END */
&ANALYZE-RESUME

&ANALYZE-
SUSPEND _UIB-
CODE-BLOCK
_CONTROL C-Win
C-Win
ON WINDOW-CLOSE
OF C-Win /*
Sistema de
Chequera en
Dólares */
DO:
/* This event
will close the
window and
terminate the
procedure. */
APPLY
"CLOSE":U TO
THIS-PROCEDURE.
RETURN NO-
APPLY.
END.

/* _UIB-CODE-
BLOCK-END */
&ANALYZE-RESUME

&Scoped-define
SELF-NAME C-Win
&ANALYZE-
SUSPEND _UIB-
CODE-BLOCK
_CONTROL C-Win
C-Win
ON END-ERROR OF
C-Win /*
Sistema de

```

```

_movto =
Q_tpmovto.id_mo
vto AND

Q_Movimiento.f_
movto >=
f_inicial AND

Q_Movimiento.f_
movto <=
f_final
BREAK
BY
Q_Movimiento.id_
_movto:

IF
FIRST(Q_Movimie
nto.id_movto)
THEN DO:

x_descrip =
"GASTOS".

DISPLAY STREAM
report

x_descrip WITH
FRAME
fp_letrero.

DOWN STREAM
report 1 WITH
FRAME
fp_detalle.
END.

ACCUMULATE
Q_Movimiento.Mo
nto(SUB-TOTAL
BY
Q_Movimiento.id_
_movto).

ACCUMULATE
Q_Movimiento.Mo
nto(TOTAL BY
Q_Movimiento.id_
_movto).

IF
LAST-
OF(Q_Movimie
nto.id_movto) THEN
DO:

x_descrip =
Q_tpmovto.Descr
ip_Larga.

x_monto =
ACCUM SUB-TOTAL
BY
Q_Movimiento.id_
_movto

```

```

BY
Q_Movimiento.id_
_movto

Q_Movimiento.Mo
nto.

DISPLAY STREAM
report

x_descrip
x_monto WITH
FRAME
fp_detalle.

DOWN STREAM
report 1 WITH
FRAME
fp_detalle.
END.

IF
LAST(Q_Movimien
to.id_movto)
THEN DO:

x_descrip =
"Total GASTOS".
x_Monto =
ACCUM TOTAL
Q_Movimiento.mo
nto.

tot_gastos =
x_Monto.

DISPLAY STREAM
report

x_descrip
x_monto WITH
FRAME fp_total.

x_descrip =
"UTILIDAD
BRUTA".
x_Monto =
Tot_Ingresos -
Tot_Gastos.

DOWN STREAM
report WITH
FRAME fp_total.

DISPLAY STREAM
report

x_descrip
x_monto WITH
FRAME fp_total.

```

```

PAGE STREAM-
report:

END.

END.

OUTPUT
STREAM report
CLOSE.
HIDE FRAME
fr_x NO-PAUSE..
IF kk_flag
THEN
MESSAGE
"FINALIZO EL
LISTADO".
ELSE
MESSAGE
"NO SE GENERO
LISTADO".

APPLY
'CLOSE' TO
THIS-PROCEDURE.

END.

/* _UIB-CODE-
BLOCK-END */
&ANALYZE-RESUME

&Scoped-define
SELF-NAME
t_fecha_final
&ANALYZE-
SUSPEND _UIB-
CODE-BLOCK
CONTROL
t_fecha_final
C-Win
ON LEAVE OF
t_fecha_final
IN FRAME
DEFAULT-FRAME
/* Fecha Final
*/
DO:
t_fecha_final
= INPUT
t_fecha_final.

END.

/* _UIB-CODE-
BLOCK-END */
&ANALYZE-RESUME

&Scoped-define
SELF-NAME
t_fecha_ini

```

```

&ANALYZE-
SUSPEND _UIB-
CODE-BLOCK
CONTROL
t_fecha_ini C-
Win
ON LEAVE OF
t_fecha_ini IN
FRAME DEFAULT-
FRAME /* Fecha
Inicial */
DO:
t_fecha_ini =
INPUT
t_fecha_ini.

END.

/* _UIB-CODE-
BLOCK-END */
&ANALYZE-RESUME

&UNDEFINE SELF-
NAME

&ANALYZE-
SUSPEND _UIB-
CODE-BLOCK
_CUSTOM_MAIN-
BLOCK C-Win

/*
*****
*****
Main Block
*****
***** */

/* Set CURRENT-
WINDOW: this
will parent
dialog-boxes
and frames.
*/
ASSIGN CURRENT-
WINDOW
= {&WINDOW-
NAME}
THIS-
PROCEDURE:CURRE
NT-WINDOW =
{&WINDOW-NAME}.

/* The CLOSE
event can be
used from
inside or
outside the
procedure to
*/

```

```

/* terminate
it.
*/
ON CLOSE OF
THIS-PROCEDURE
RUN
disable_UI.

/* Best default
for GUI
applications
is...
*/
PAUSE 0 BEFORE-
HIDE.

/* Now enable
the interface
and wait for
the exit
condition.
*/
/* (NOTE:
handle ERROR
and END-KEY so
cleanup code
will always
fire.
*/
MAIN-BLOCK:
DO ON ERROR
UNDO MAIN-
BLOCK, LEAVE
MAIN-BLOCK.
ON END-KEY
UNDO MAIN-
BLOCK, LEAVE
MAIN-BLOCK.
RUN
enable_UI.
IF NOT THIS-
PROCEDURE:PERSI
STENT THEN
WAIT-FOR
CLOSE OF THIS-
PROCEDURE.
END.

/* _UIB-CODE-
BLOCK-END */
&ANALYZE-RESUME

*****
*****
Internal
Procedures
*****
***** */

&ANALYZE-
SUSPEND _UIB-
CODE-BLOCK

```

APÉNDICE E

Manual de Usuario

Sistema de Cheques en Dólares

La Institución Financiera ofrece un concepto que integra diversos servicios y productos financieros y satisface las necesidades de inversión en dólares americanos en una cuenta de Cheques. Esta cuenta de Cheques permite seleccionar los productos y servicios bajo un solo contrato así como recibir la información de las operaciones en su Estado de Cuenta y es un producto tanto para personas físicas como personas morales. Los productos con que cuenta son:

- Personas Físicas
- Personas Morales

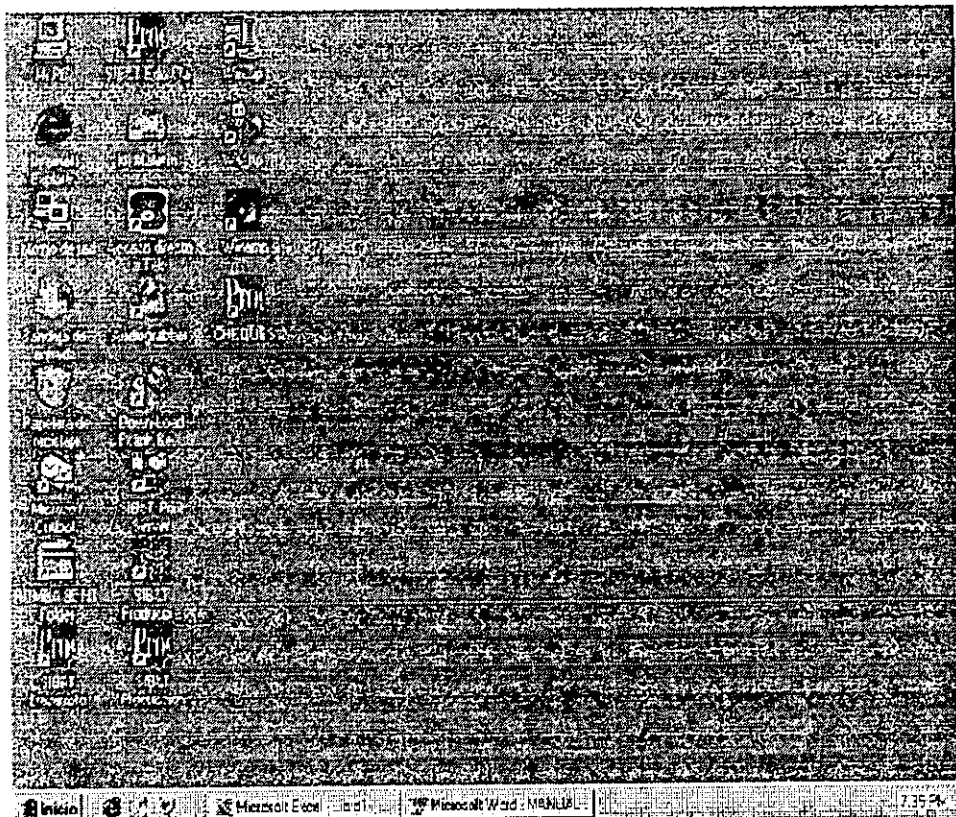


Figura E_1 Ejecución del Sistema de Cheques

- Como inicio al sistema el usuario debe ejecutar el Icono de acceso directo "CHEQUES" como se muestra en la figura E_1.
- Después de solicitará la entrada al sistema con su respectivo LOGIN, como se muestra en la figura E_2.

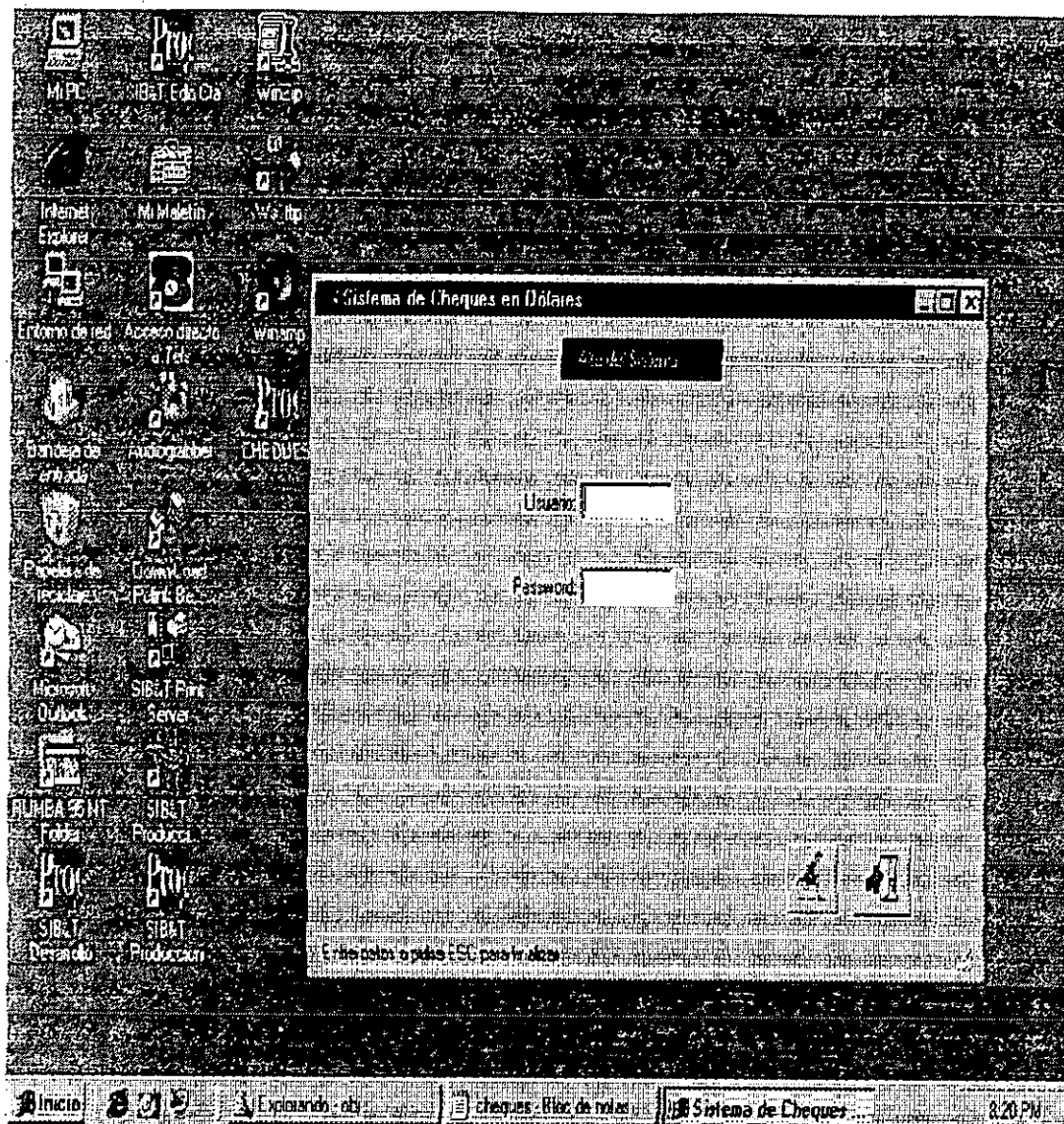


Figura E_2 Entrada al Sistema de Cheques en Dólares

Posteriormente el Sistema de Cheques en Dólares mostrará el siguiente menú de operación, el cual se muestra en la figura E_3.

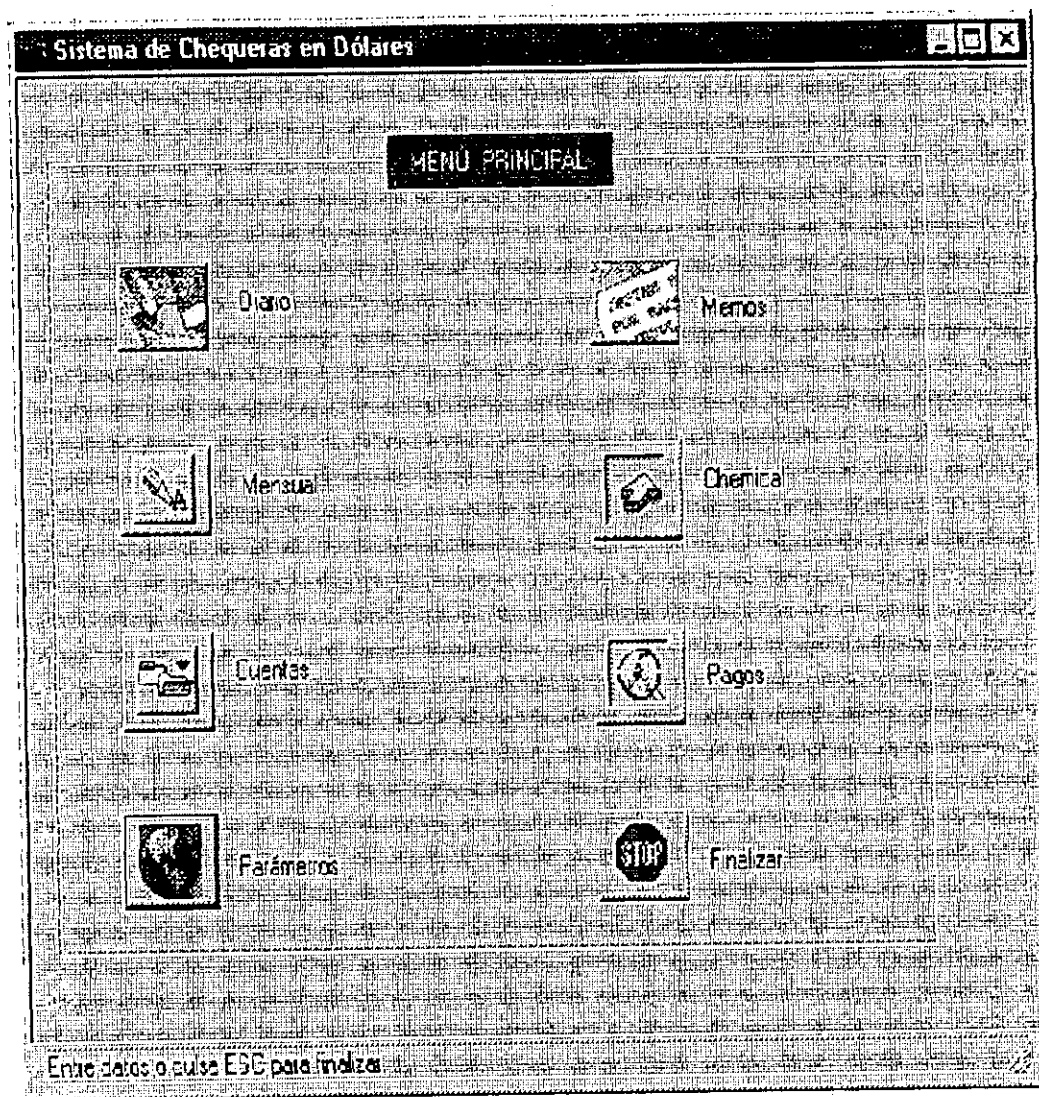


Figura E_3 Menú Principal del Sistema

Los botones del menú principal muestran cada una de las pantallas mostradas en las figuras E_4 a E_10 respectivamente.

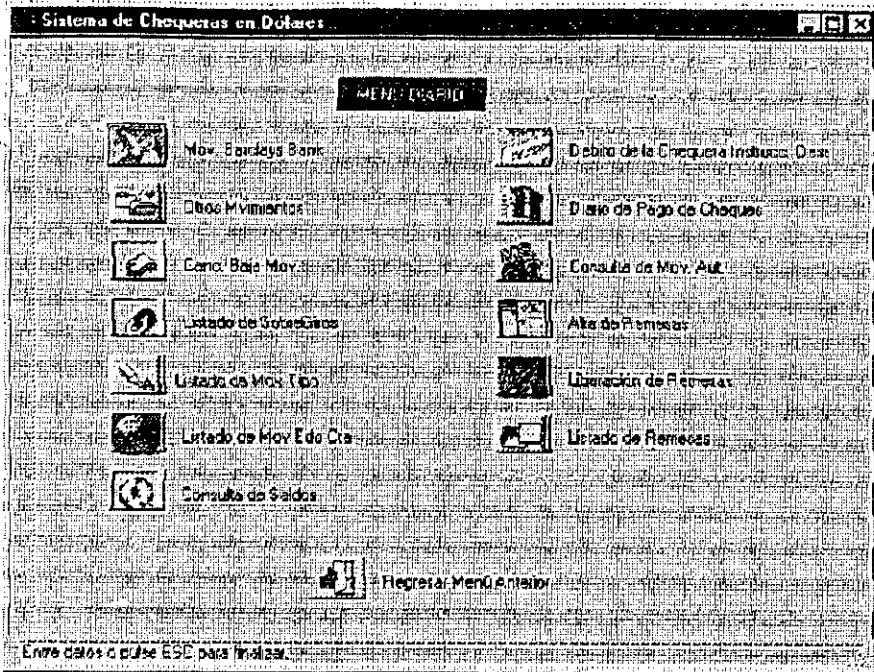


Figura E_4 Submenú de Diario

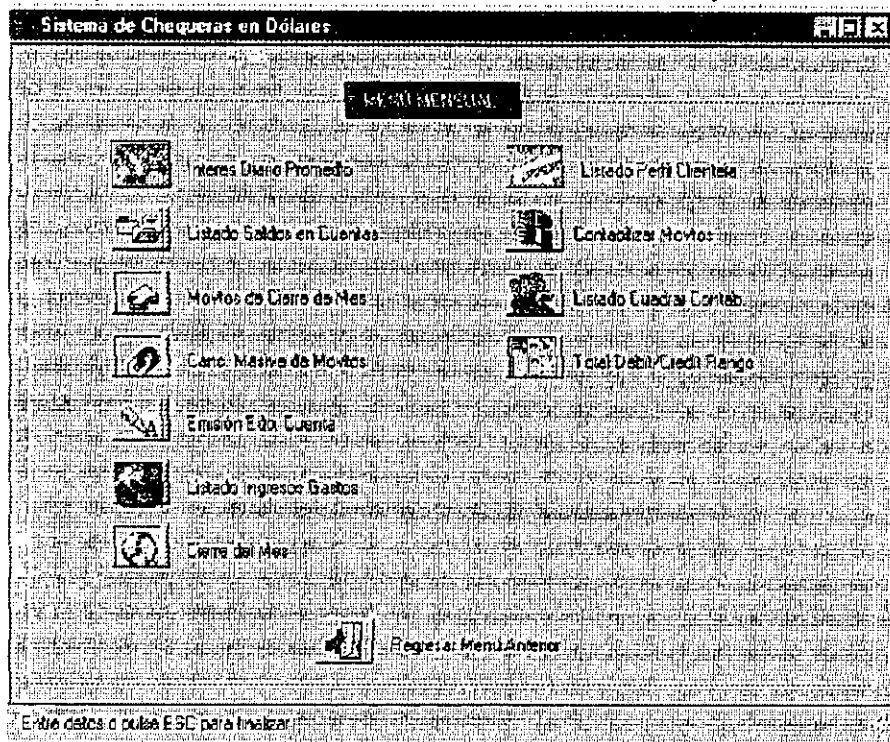


Figura E_5 Su

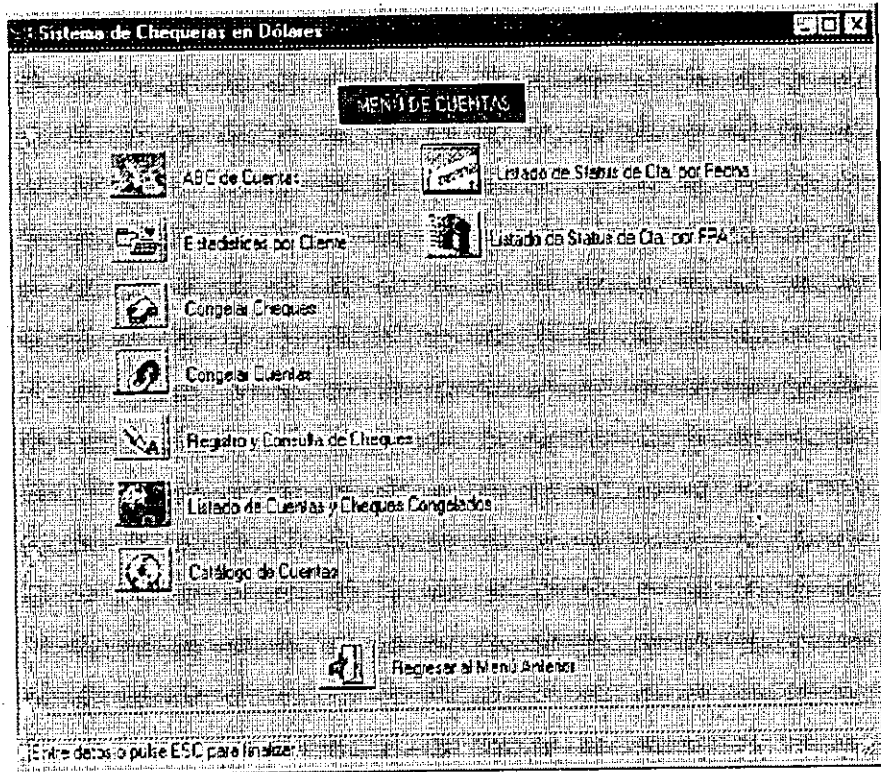


Figura E_6 Submenú de Cuentas

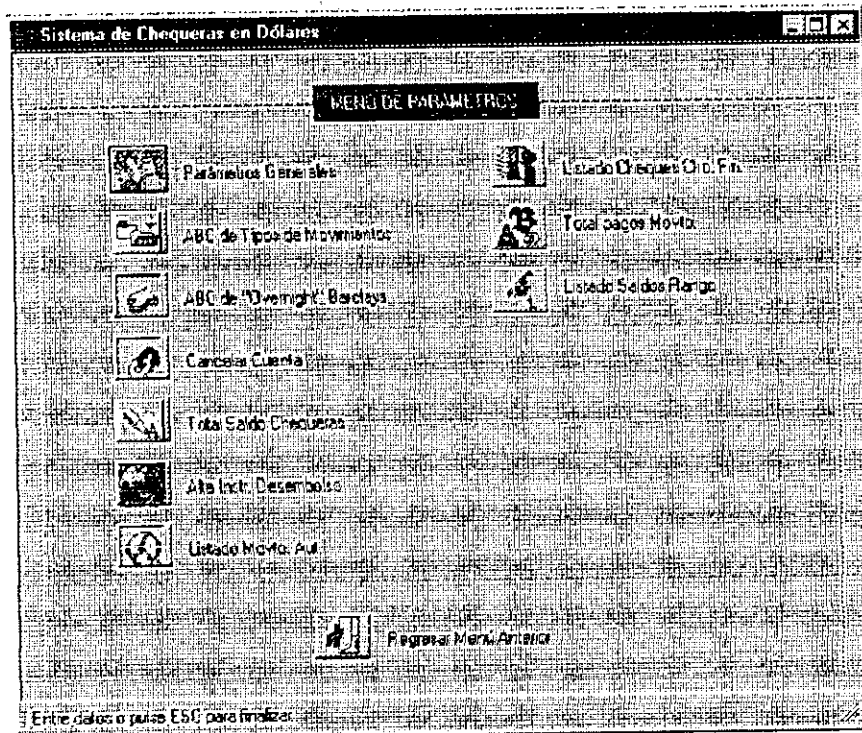


Figura E_7 Submenú de Parámetros

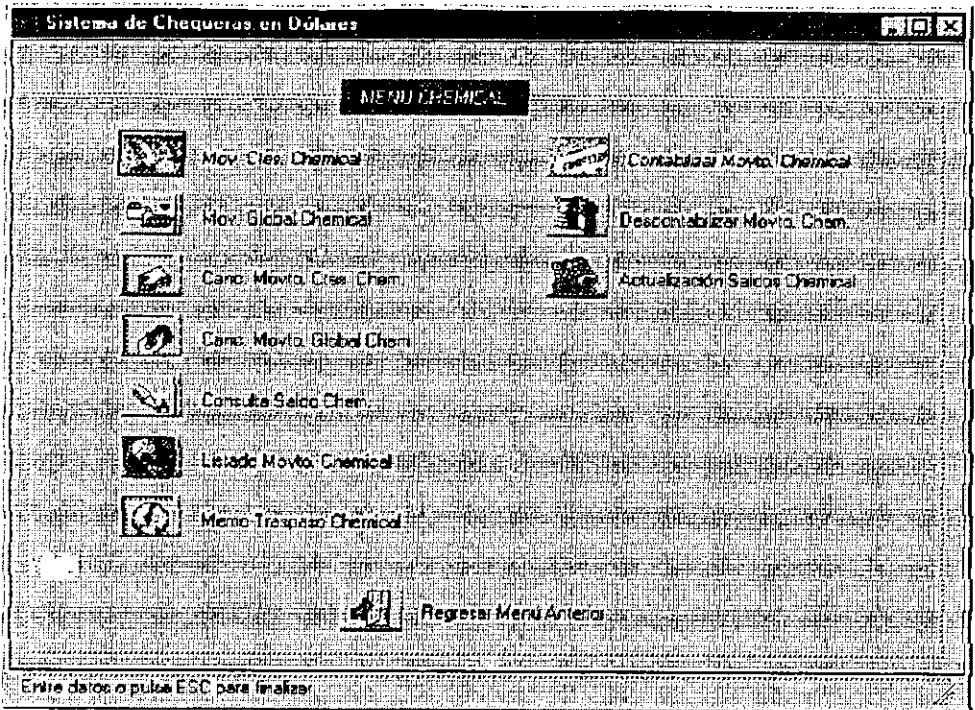


Figura E_8 Submenú de Chemical

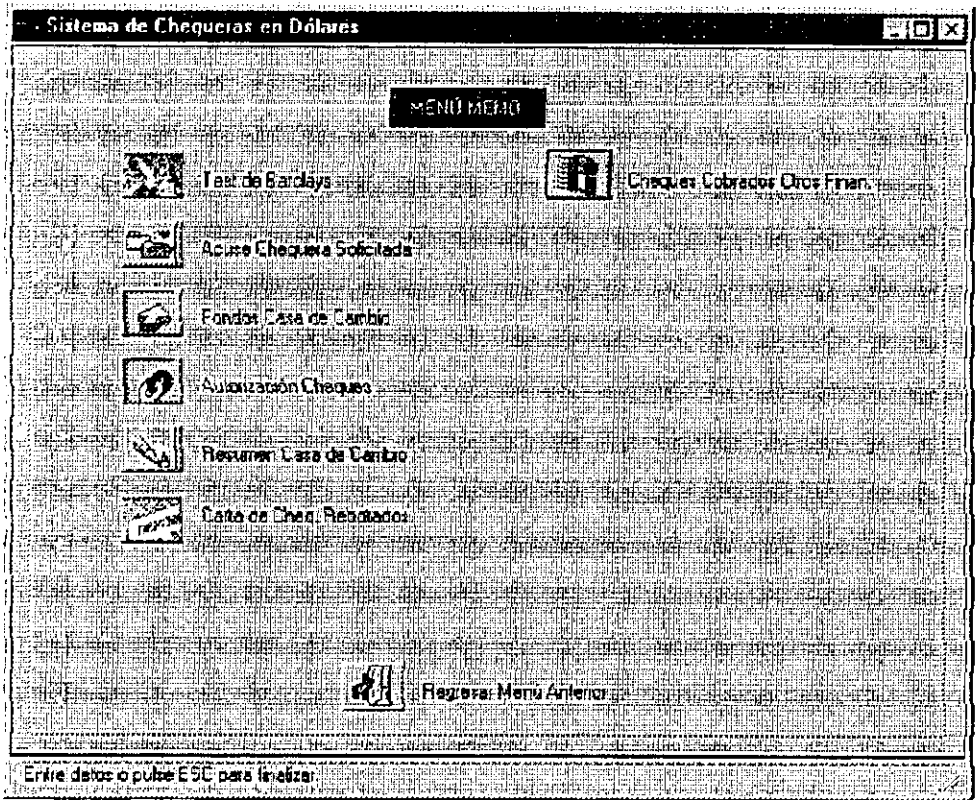


Figura E_9 Submenú de Memo

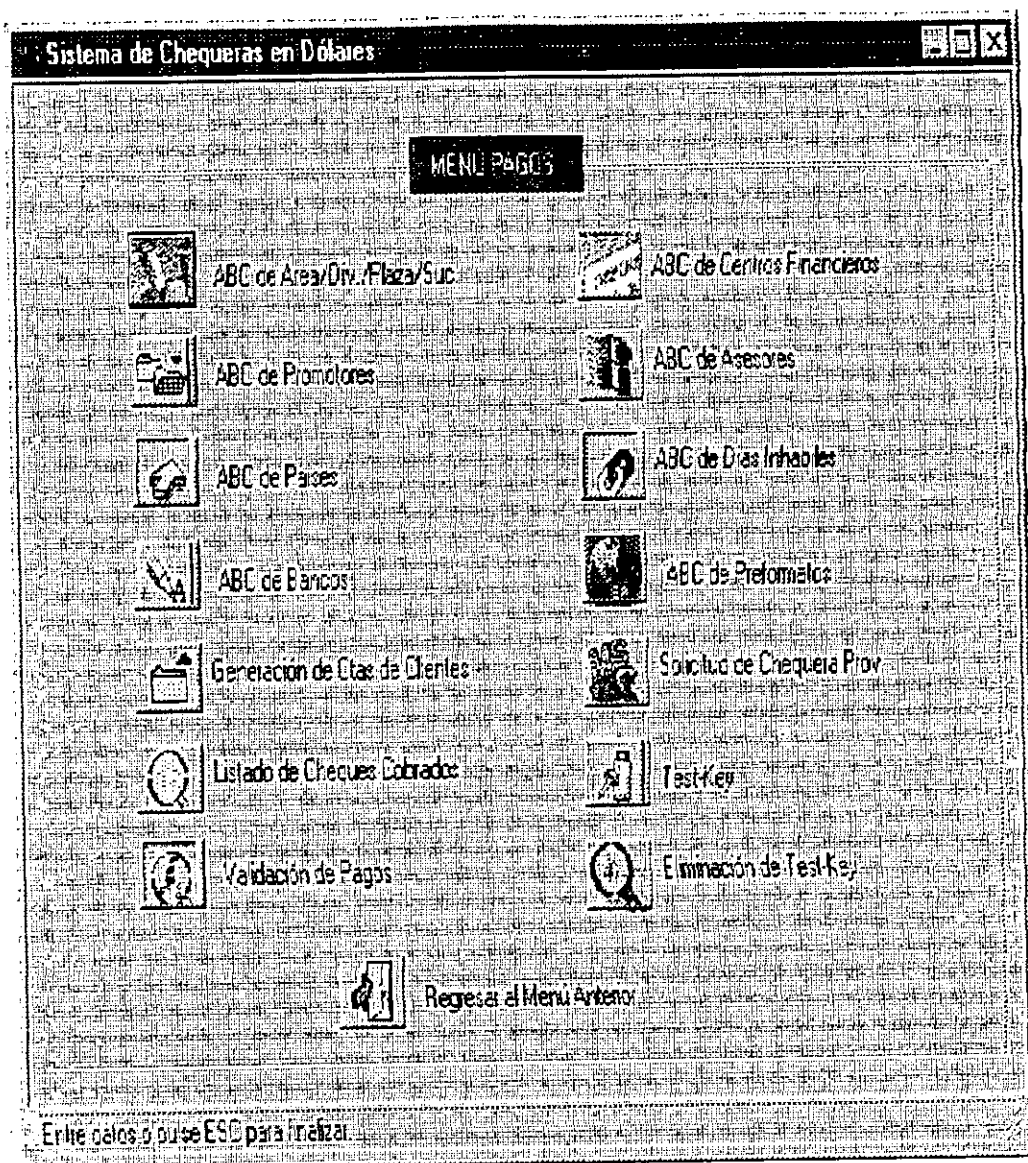


Figura E_10 Submenú de Pagos

En el Submenú de Diario se tiene la captura de movimientos de la cuenta concentradora de Barclays Bank y otros movimientos, así como alta de remesas de los cheques depositados por los clientes para su cobro. También se cuenta con saldos de cuentas y listado de los movimientos dados de alta.

En el Submenú de Mensual se tiene la generación de movimientos del cierre del mes, cierre mensual e impresión de Estado de Cuenta, así como contabilización y descontabilización de movimientos.

En el Submenú de Cuentas se tiene el alta de cuentas, consulta de movimientos estadísticos, cancelar cheques, congelar cuenta y listados de las cuentas vigentes.

En el Submenú de Parámetros se tienen los parámetros del sistema así como el catálogo de tipos de movimientos, cancelar cuentas, listado de cuentas, alta de desembolsos de cuentas y alta de "overnight" de chequera en Barclays Bank.

En el Submenú de Memos se tiene los siguientes machotes: Test Barclays, casa de cambio, autorización de cheques, carta de cheques rebotados y cheques cobrados en centros financieros.

En el Submenú de Chemical se tiene el alta de movimientos de clientes y cuenta global, así como consulta de movimientos, listado, traspaso de saldos de la cuenta concentradora y contabilización y descontabilización de movimientos.

En el Submenú de Pagos se tiene el alta de los promotores, alta de áreas de los centros regionales, alta de días inhábiles, alta de bancos, alta de centros regionales, alta de asesores, alta de preformatos, solicitud de cheques al promotor y cobro de cheques en los centros regionales.

A continuación se muestra cada opción de los submenús antes comentados y la descripción de su función.

1.0 El submenú Diario tiene las siguientes opciones.

1.1 Movimientos de Barclays Bank:

Captura de Movimientos que afectan la cuenta del cliente y la cuenta concentradora de Barclays. Este movimiento afecta únicamente aquel que sea de naturaleza "EC". En el alta del movimiento se dará una fecha valida (día hábil y que no sea fin de semana). Los campos a capturar son : Producto, cuenta, Clave del Movimiento Debit/Credit, referencia y monto, afectando el saldo del cliente directamente y teniendo validaciones en la cuenta cancelada, congelada, cheques congelados o tipos movimiento con otra naturaleza que no corresponda a movimiento de tipo Barclays. Ver la figura E_11.

The screenshot shows a window titled "Sistema de Cheques en Dólares" with a sub-header "Captura de Movimientos". Below the sub-header is a date field labeled "Fecha de los Movimientos a Capturar" with a date of 1/7/77. A table with the following columns is present: "Producto", "Cuenta", "Clave Movo", "Descripción", "Referencia", "Debit", and "Credit". Below the table are two input fields: "Cliente" and "Cta Global". At the bottom of the window, there is a footer that says "Entre datos o pulse ESC para finalizar".

Figura E_11 Captura de Movimientos

1.2 Otros Movimientos:

Captura de Otros movimientos que afecta únicamente aquel que sea de naturaleza "OM". En el alta del movimiento se dará una fecha valida (día hábil y que no sea fin de semana). Los campos a capturar son: producto, cuenta, clave del movimiento Debit/Credit afectando el saldo del cliente directamente y teniendo validaciones en la cuenta cancelada, congelada o tipo movimiento con otra naturaleza que no corresponda a movimiento de tipo Otro Movimiento (internos). Ver la figura E_12.

The screenshot shows a window titled "Sistema de Cheques en Dólares" with a sub-header "Captura de Otro Movimiento". The form contains the following fields:

- Producto: [Empty]
- Cuenta: [Empty] (with a dropdown menu showing "Cuenta con cancelación Química")
- Fecha: [/ /]
- Clave del Movto: [Empty] (with a dropdown menu showing "Débito")
- Referencia: [Empty]
- Debit: [0.00]
- Credit: [0.00]
- Saldo Disponible: [0.00]
- Saldo Total: [0.00]

At the bottom of the window, there is a button labeled "Teclee el Producto de la Cuenta" and two small icons.

Figura E_12 Captura de Otros Movimientos

1.3 Baja de Movimientos:

Baja de movimientos capturados ya sean de cualquier naturaleza, afectando el saldo del cliente y validando que no este cancelada la cuenta o congelada y la fecha sea de tipo hábil o no sea fin de semana. Los campos a capturar son: fecha y los campos mostrados son producto, cuenta, clave del Movimiento Debit/Credit, referencia y monto afectando el saldo del cliente directamente. Ver la figura E_13.

The screenshot shows a window titled "Sistema de Cheques en Dólares" with a sub-header "Baja de Movimientos". Below this is a date selection field labeled "Fecha de los Movimientos a Capturar:". The main area is a table with the following columns: "Producto", "Cliente", "Cve. Movto.", "Descripción", "Referencia", "Debit", and "Credit". The table is currently empty. Below the table are two input fields: "Cliente:" and "Cta Global:". At the bottom of the window, there is a footer that reads "Entre datos o pulse ESC para finalizar".

Figura E_13 Baja de Movimientos

1.3 Baja de Movimientos:

Baja de movimientos capturados ya sean de cualquier naturaleza, afectando el saldo del cliente y validando que no este cancelada la cuenta o congelada y la fecha sea de tipo hábil o no sea fin de semana. Los campos a capturar son: fecha y los campos mostrados son producto, cuenta, clave del Movimiento Debit/Credit, referencia y monto afectando el saldo del cliente directamente. Ver la figura E_13.

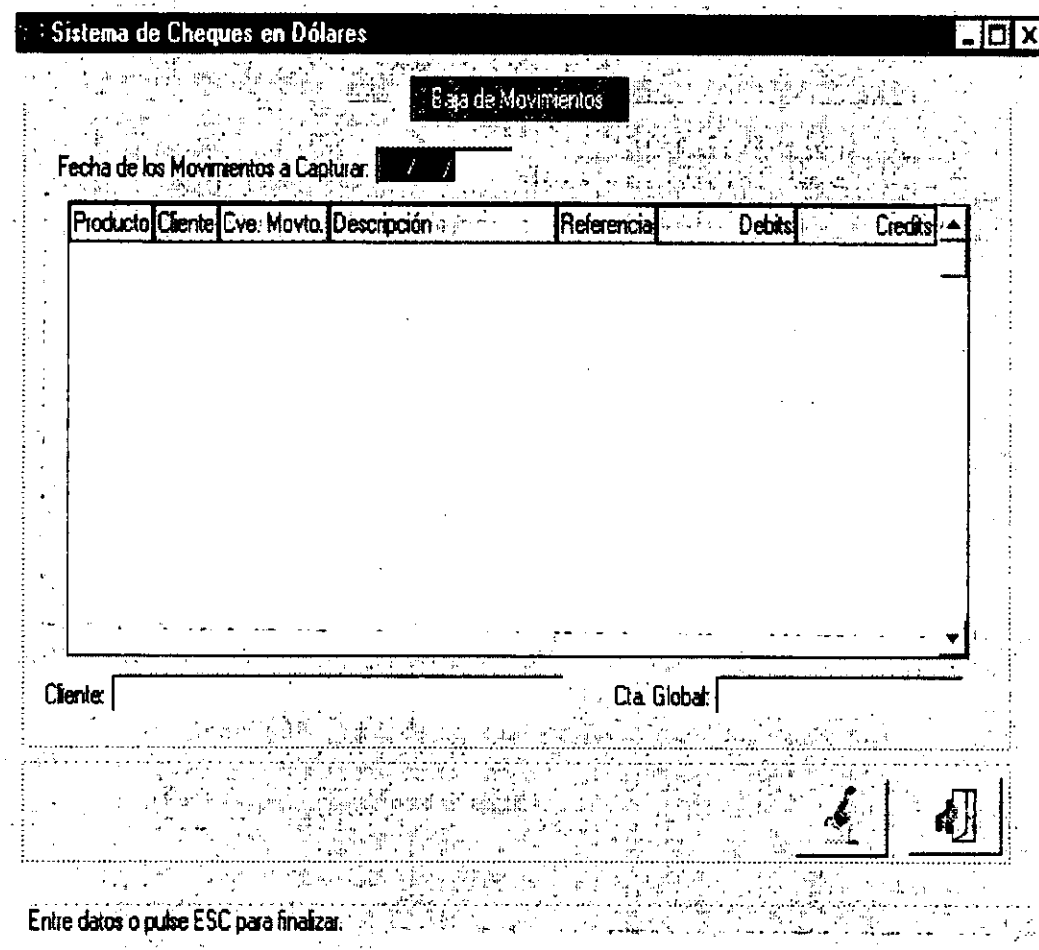


Figura E_13 Baja de Movimientos

1.5 Listado de Tipo de Movimientos:

Reporte de movimientos por tipo con fecha de inicio y final capturados. El listado tendrá producto, cuenta, fecha, nombre y saldo. Ver la figura E_15.

The screenshot shows a window titled "Sistema de Cheques en Dólares" with a sub-header "Listado Movimientos por Tipo". The form contains the following fields:

- Tipo de Movto:** A text input field containing the word "Escriba".
- Fecha Inicia:** A date input field with slashes (/ /).
- Fecha Final:** A date input field with slashes (/ /).

At the bottom of the window, there is a status bar that reads "Entre datos o pulse ESC para finalizar".

Figura E_15 Listado de Movimientos por Tipo

1.5 Listado de Tipo de Movimientos:

- Reporte de movimientos por tipo con fecha de inicio y final capturados. El listado tendrá producto, cuenta, fecha, nombre y saldo. Ver la figura E_15.

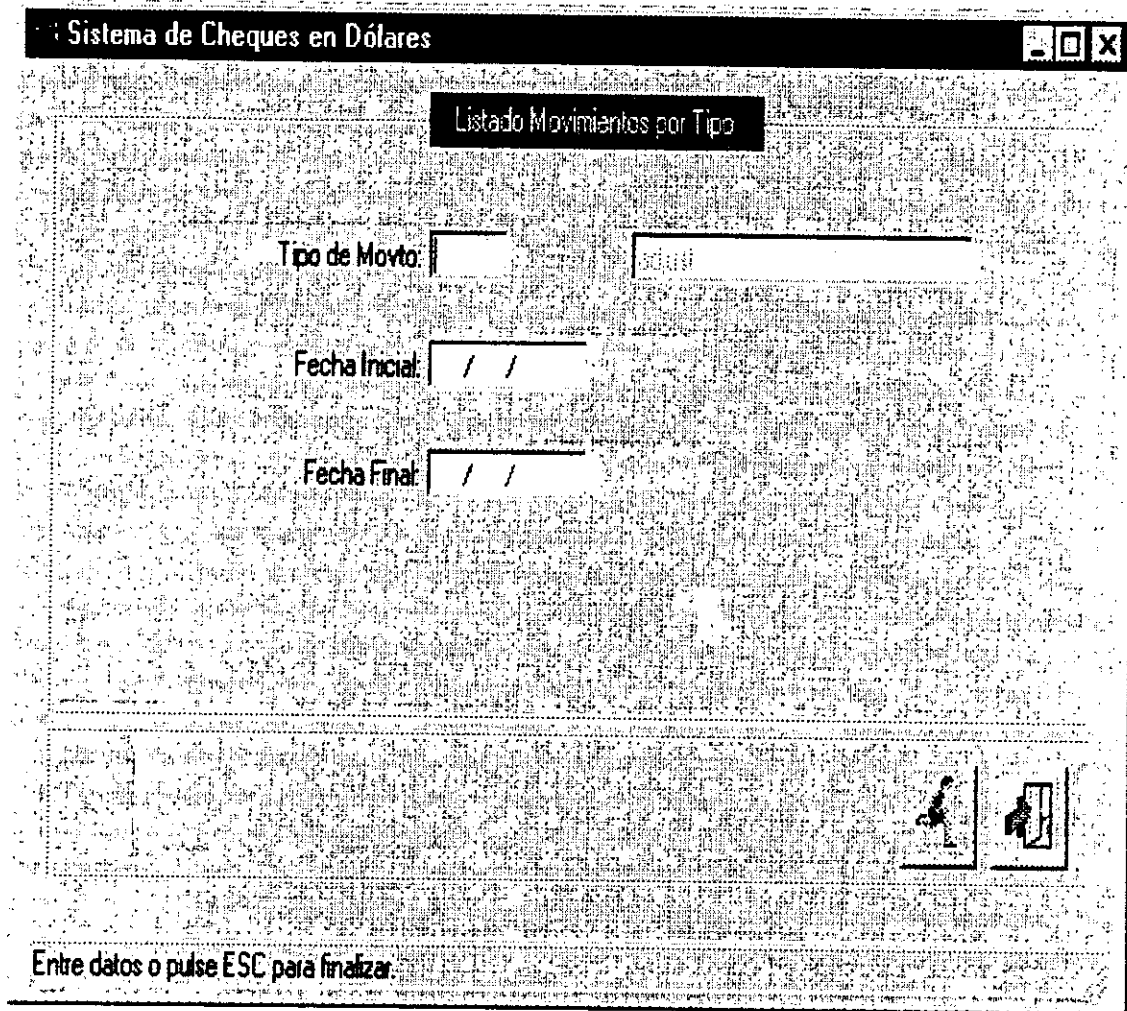


Figura E_15 Listado de Movimientos por Tipo

1.7 Consulta de Saldos:

Consulta de saldos por cuenta dando el rango de fecha inicial y final y cuales fueron los movimientos dados de alta en el sistema con esa fecha. Ver la figura E_17.

Sistema de Cheques en Dólares

Consulta de Estado de Cuenta

Producto: Cuenta: Fecha Inicial: / /
 Fecha Final: / /

Fecha Movio	Eve. Movio	Descripcion	Referencia	Debits	Credits	Id

Saldo Disponible: Saldo Total:

Teclee la Clave del Producto

Figura E_17 Consulta de Estado de Cuenta

1.7 Consulta de Saldos:

Consulta de saldos por cuenta dando el rango de fecha inicial y final y cuales fueron los movimientos dados de alta en el sistema con esa fecha. Ver la figura E_17.

Sistema de Cheques en Dólares

Consulta de Estado de Cuenta

Producto: Cuenta: Fecha Inicial: / /
Fecha Final: / /

Fecha Movto	Cve.Movto	Descripción	Referencia	Debits	Credits	Tot

Saldo Disponible: Saldo Total:

Teclee la Clave del Producto

Figura E_17 Consulta de Estado de Cuenta

1.9 Diario de pagos de Cheques:

Listado de pagos de cheques capturados con tipo de movimiento "EC" que fueron enviados a su cobro por Barclays Bank. La captura va ser la fecha de la captura. Ver la figura E_19.

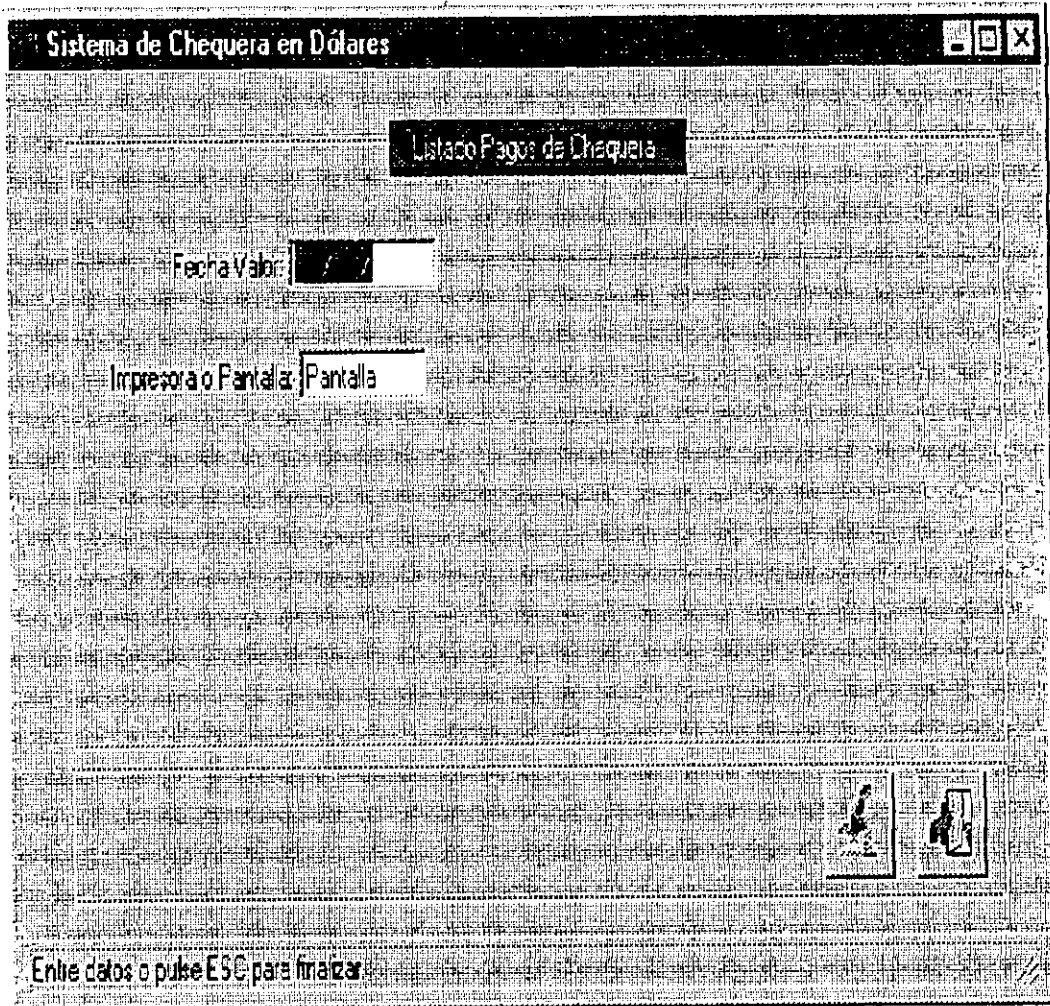


Figura E_19 Listado de Pagos de Chequera

1.9 Diario de pagos de Cheques:

Listado de pagos de cheques capturados con tipo de movimiento "EC" que fueron enviados a su cobro por Barclays Bank. La captura va ser la fecha de la captura. Ver la figura E_19.

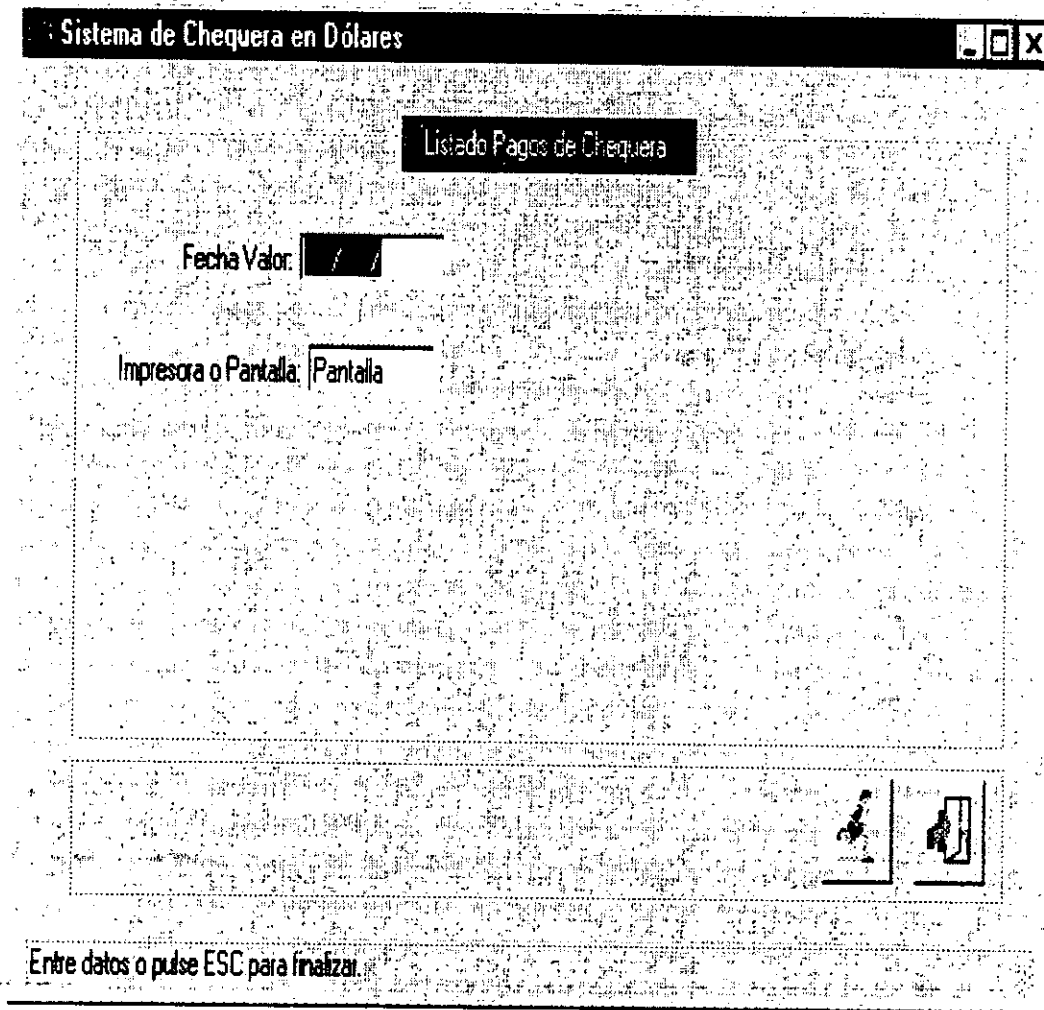


Figura E_19 Listado de Pagos de Chequera

1.11 Alta de Remesas:

Alta de cheques para su cobro y deposito por los clientes de Barclays Bank o Chase Bank. La captura será # remesa, banco, monto, referencia y número de cheque. Ver la Figura E_3.

The screenshot shows a window titled "Sistema de Cheques en Dólares" with a menu bar containing "ABC de Remesas". Below the menu bar is a table titled "Alta de Remesas" with the following columns: "No Remesa", "Id Banco", "Producto", "Cliente", "Nombre", "No Cheque", and "Monto". The table is currently empty. At the bottom of the window, there is a toolbar with several buttons, including one labeled "Add". The text "Alta de Registros REMESAS" is visible in the bottom right corner of the table area.

Figura E_3 Alta de Remesas

1.11 Alta de Remesas:

Alta de cheques para su cobro y deposito por los clientes de Barclays Bank o Chase Bank. La captura será # remesa, banco, monto, referencia y número de cheque. Ver la Figura E_3.

The screenshot displays a software window titled "Sistema de Cheques en Dólares". Inside the window, there is a sub-header "ABC de Remesas" and a table with the following columns: "No. Remesa", "Id. Banco", "Producto", "Cliente", "Nombre", "No. Cheque", and "Monto". The table is currently empty. To the right of the table, there is a button labeled "Alta de Registros REMESAS". Below the table, there is a toolbar containing an "Add" button and other icons. The text "ABC de Remesas" is also visible at the bottom left of the window.

Figura E_3 Alta de Remesas

1.12 Liberación de Remesas:

Liberación de la remesa capturada para su cobro con memo de envío y afectara los saldos de los clientes y movimiento a la cuenta. Ver la Figura E_22.

Sistema de Chequera en Dólares

Liberación de Remesas

No. Remesa: Fecha de Aplicación: 11/17/1999 Referencia: 9,999

Credit Local Barclays: DEPO Debit Local Barclays:

Credit Foraneo Barclays: DEPO Debit Foraneo Barclays: PRDE

Credit Local Chase: DECH Debit Local Chase:

Teclee el número de remesa para el envío de cheques.

Figura E_22 Liberación de la Remesa

1.13 Listado de Remesas:

Reporte de remesas pendientes por liberar o liberadas. Ver la Figura E_23.

Sistema de Chequera en Dólares

Listado Listado de Remesas

Tipo de Consulta (Fechas/Remesa): Remesas

Tipo de Mov. (Aplicados/No Aplicados): No Aplicados

Fecha Inicial: / /

Fecha Final: / /

No. Remesa: 0

La consulta por Fecha o Remesa

Figura E_23 Listado de Remesas

2.0 El submenú Mensual tiene las siguientes opciones.

2.1 Listado de Saldos en Cuentas:

Reporte con los saldos de las cuentas, disponibles y total, validando que las cuentas no esten canceladas. Ver la figura E_24.

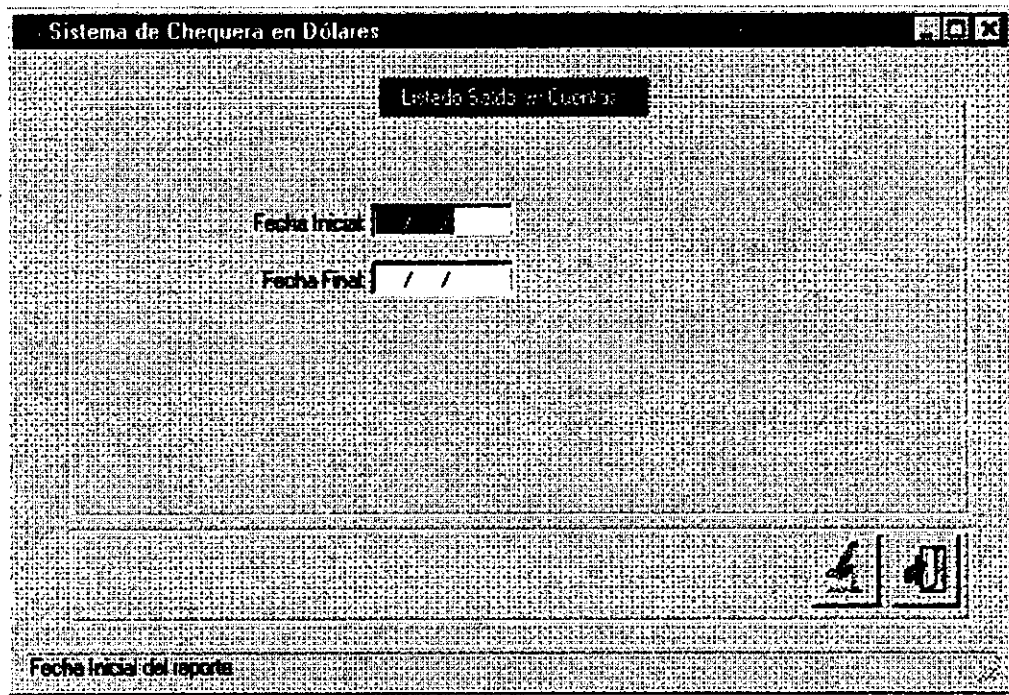


Figura E_24 Listado de Saldos de Cuentas

2.2 Movimiento de cierre de mes:

Generación de pago de intereses por saldo promedio alto, cobro de intereses por saldo promedio bajo y cobro de comisiones por cheques girados. La captura de este proceso será la tasa para personas morales y personas físicas y fecha de cierre. Ver la figura E_25.

The screenshot shows a graphical user interface window titled "Sistema de Chequera en Dólares". Inside the window, there is a sub-header "Generar Movimientos de Cierre de Mes". Below this, there are three input fields:

- Periodo: aaaa/mm/ii
- Tasa p/Persona Física: 0.00000
- Tasa p/Persona Moral: 0.00000

At the bottom of the window, there is a label "Debe girar mes: Año y Mes" and two small icons on the right side.

Figura E_25 Generar movimientos de Cierre de Mes

2.3 Cancelación masiva de movimientos:

Eliminación de movimientos generados por el cierre del mes. La captura será por tipo de movimiento y fecha. Ver la figura E_26.

The screenshot shows a window titled "Sistema de Chequera en Dólares" with a sub-dialog box titled "Cancelación Masiva de Movimientos". The dialog contains two input fields: "Id. Movimientos a Cancelar:" followed by a text box, and "Fecha de los movimientos a Cancelar:" followed by a date input field with slashes. At the bottom of the dialog, there are two buttons with icons (one with a person, one with a document) and a label "Clave del Movimiento".

Figura E_26 Cancelación de Masiva de Movimientos

2.4 Emisión Estado de Cuenta:

Impresión de los estados de cuenta de los clientes del mes operado por la cuenta. Los campos a capturar son: cuenta inicial, final, periodo y leyenda en el estado de cuenta. Ver la figura E_27.

The screenshot shows a window titled "Sistema de Cheques en Dólares" with standard Windows window controls (minimize, maximize, close). Inside the window, there is a section titled "Impresión Estado de Cuenta" containing three input fields: "Cuenta Inicial:" with a dropdown arrow, "Cuenta Final:" with a dropdown arrow, and "Periodo:" with a text input field. Below this is a section titled "Mensaje Estado de Cuenta" which contains a large, empty rectangular area for text. At the bottom right of the window, there are two small icons: one of a person walking and another of a person standing. At the bottom left, the text "Teclee el número de Cuenta" is visible.

Figura E_27 Emisión de Estado de Cuenta

2.5 Listado de ingresos, gastos y utilidad:

Reporte del cálculo de los ingresos, gastos y utilidad de las cuentas vigentes con saldo disponible y total alto. Ver la figura E_28.

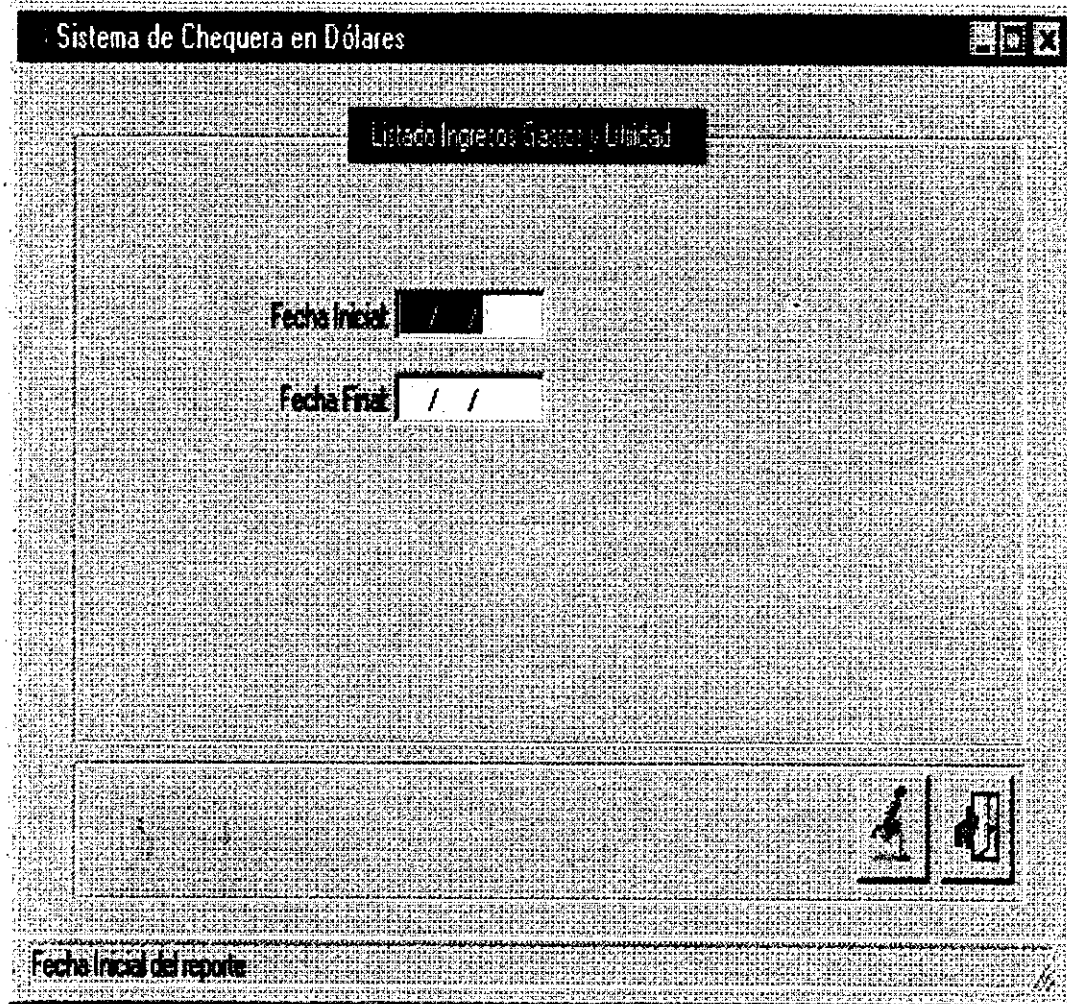


Figura E_28 Listado ingresos gastos y utilidad

2.6 Cierre del Mes:

Cálculo de los movimientos aplicados en la cuenta, generando las estadísticas como saldo final, saldo inicial, #credits, #debits, total debits, total credits, cheques emitidos, etc. Ver la figura E_29.

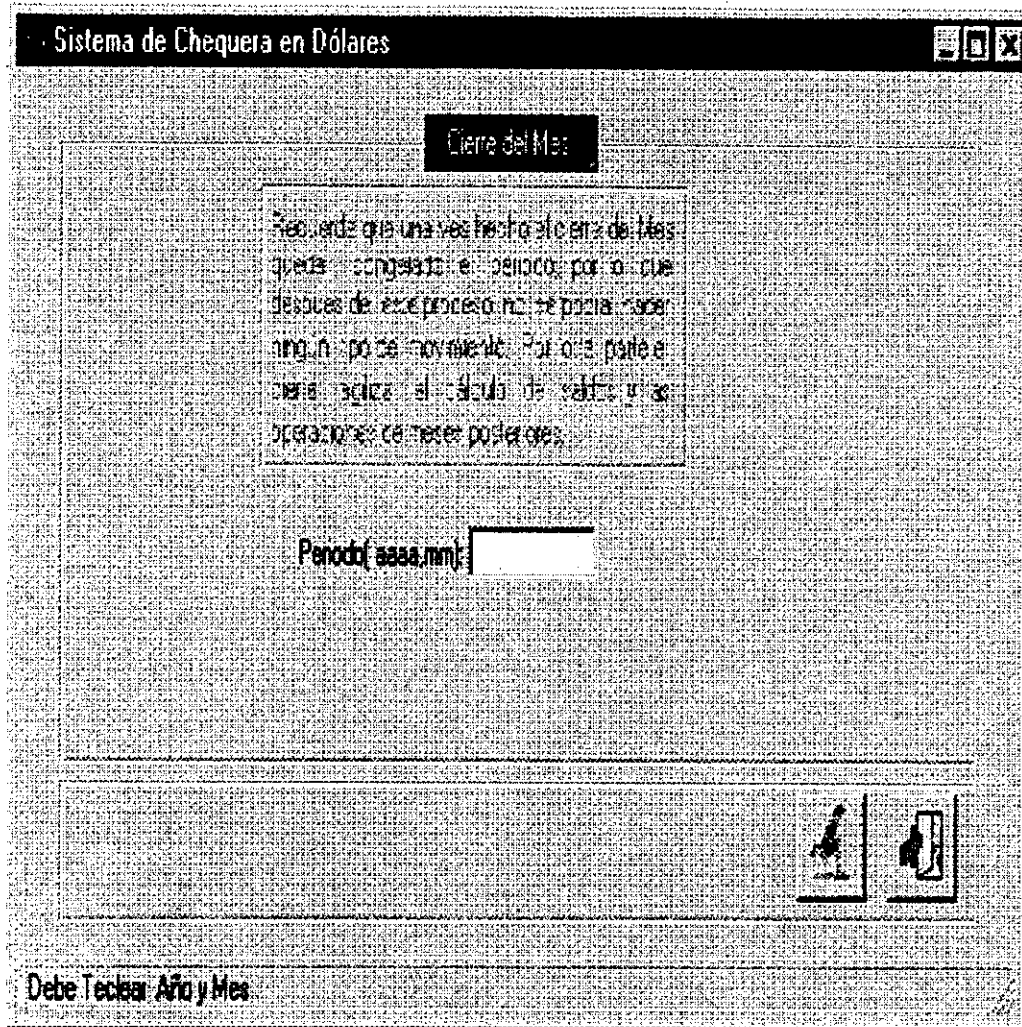


Figura E_29 Cierre del Mes

2.8 Listado perfil de la clientela:

El perfil de la cliente corresponde numero de cuentas aperturadas por centros de regionales y los montos promedios de estas. Ver la figura E_30.

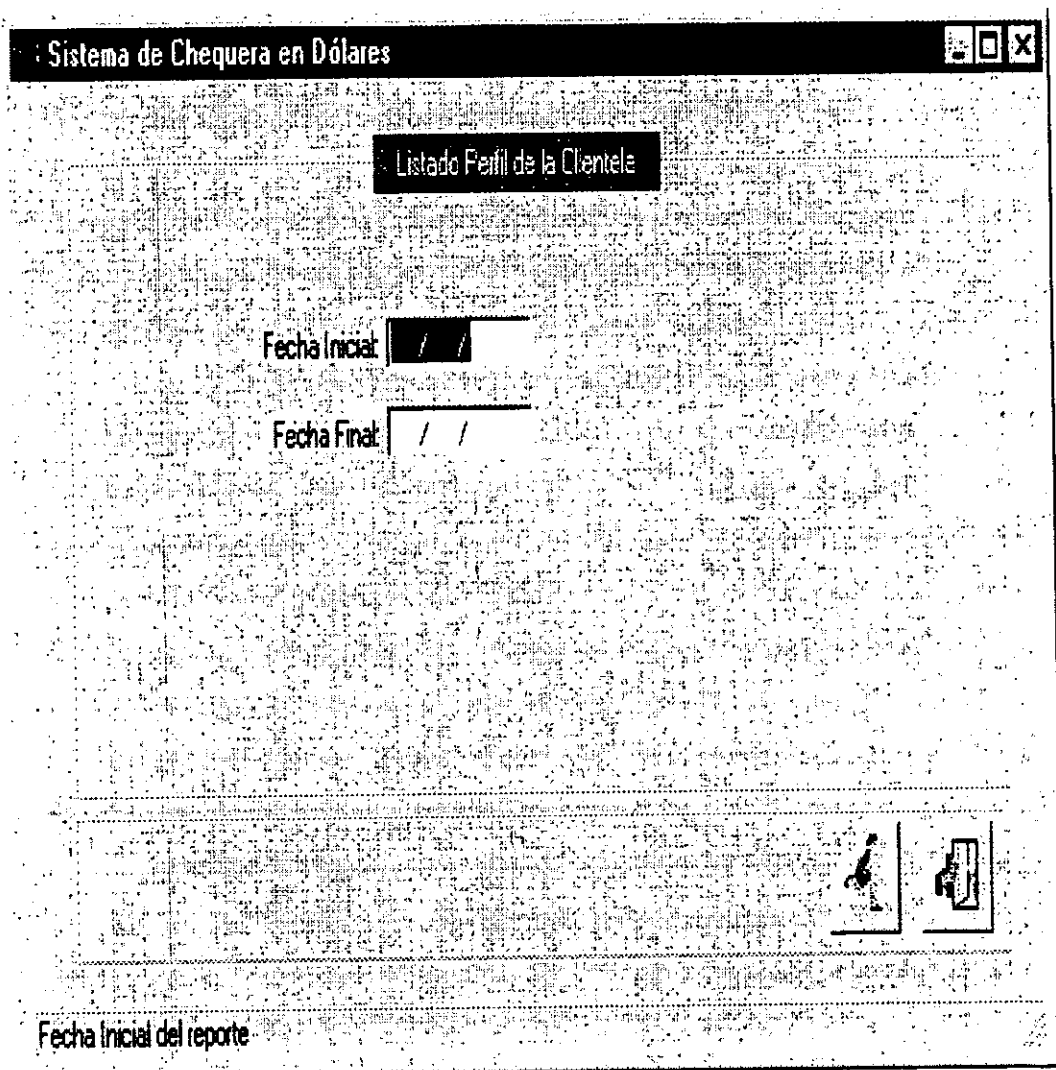


Figura E_30 Listado de la Perfil de la Clientela

2.9 Contabilizar y Descontabilizar movimientos:

Contabilización y Descontabilización de los movimientos de las cuentas, en la operación efectuada. La captura va ser por fecha. Ver la figura E_31.

Sistema de Chequera en Dólares

Contabilidad / Descontabilizar

Contabilización (Contabilidad/Descontabilizar): Descontabilizar

Contabilizar Descontabilizar

Fecha a Contabilizar: / /

Número de Poliza: 0

Fecha para Descontabilizar: / /

Teclee Contabilizar ó Descontabilizar

Figura E_31 Contabilizar y Descontabilizar movimientos

2.10 Listado cuadrar contabilidad:

Reporte usado por contabilidad como apoyo en caso que no cuadraran los movimientos contables. Figura E_32.

Sistema de Chequera en Dólares

Listado de Cuentas Contabilidad

Saldo:

Outstanding:

Movtos. en Contab.:

Total o Detallado:

Fecha Inicial:

Fecha Final:

Listar la Impresora

Figura E_32 Listado para Cuadrar Contabilidad

2.11 Total debit/credit rango:

Reporte de movimientos efectuados para los debits y credits ordenados por rango en monto. Figura E_33.

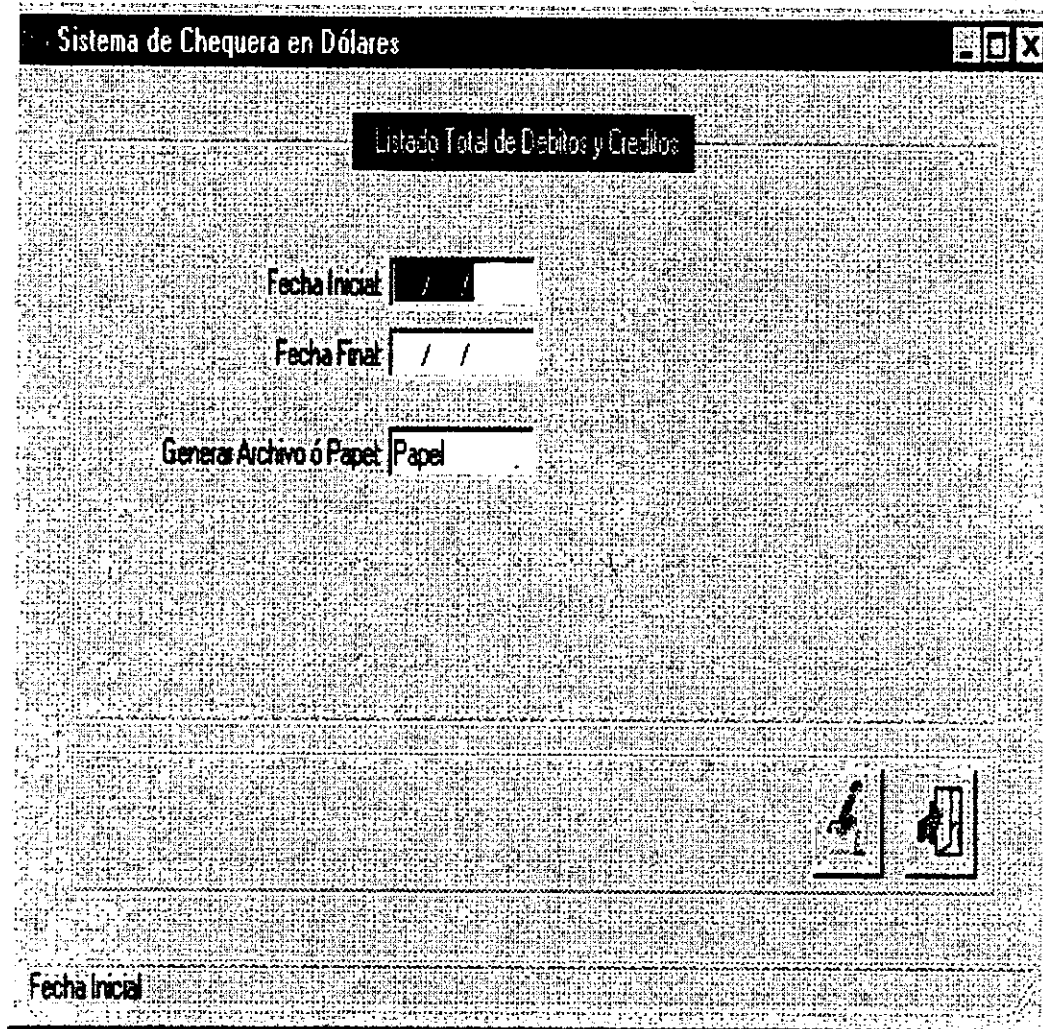


Figura E_33 Listado Total de Créditos y Debitos

3.0 El submenú de Cuentas tiene las siguientes opciones.

3.1 ABC de Cuentas:

Mantenimiento de cuentas, apertura de cuenta, así como cancelación de la cuenta. La captura son los datos generales de la cuenta. Ver la figura E_34.

Sistema de Cheques en Dólares

Altas, Bajas y Cambio al Catálogo de Clientes

Seleccione una Pestaña:

Sumario	Saldos y IDs	Otros
---------	--------------	-------

Producto: No. de Cliente: 22809 Nombre: Cuenta concentradora Chemical

Atención: _____ RFC: _____

Dirección(1): _____ C.P.: _____

Dirección(2): _____ Estado: 19

Dirección(3): _____ País: _____

Dirección(4): _____

Fecha de Alta: 12/10/1999 Fecha de Cancelación: _____

Teléfono 1: _____ Teléfono 3: _____

Teléfono 2: _____ Teléfono 4: _____

¿Cliente de Tarjeta? No

¿Cliente de Crédito? No

¿Cliente de Inversión? No

¿Cuenta Especial? SI

Save Reset Add Copy Delete

Navigation icons: Home, Back, Forward, Stop

Figura E_34 Mantenimiento a Cuentas

3.2 Estadísticas por cliente:

Consultar los cierres mensuales de las cuentas como saldos iniciales, Saldos Finales, # Credits, # Debits, etc. Ver la figura E_35.

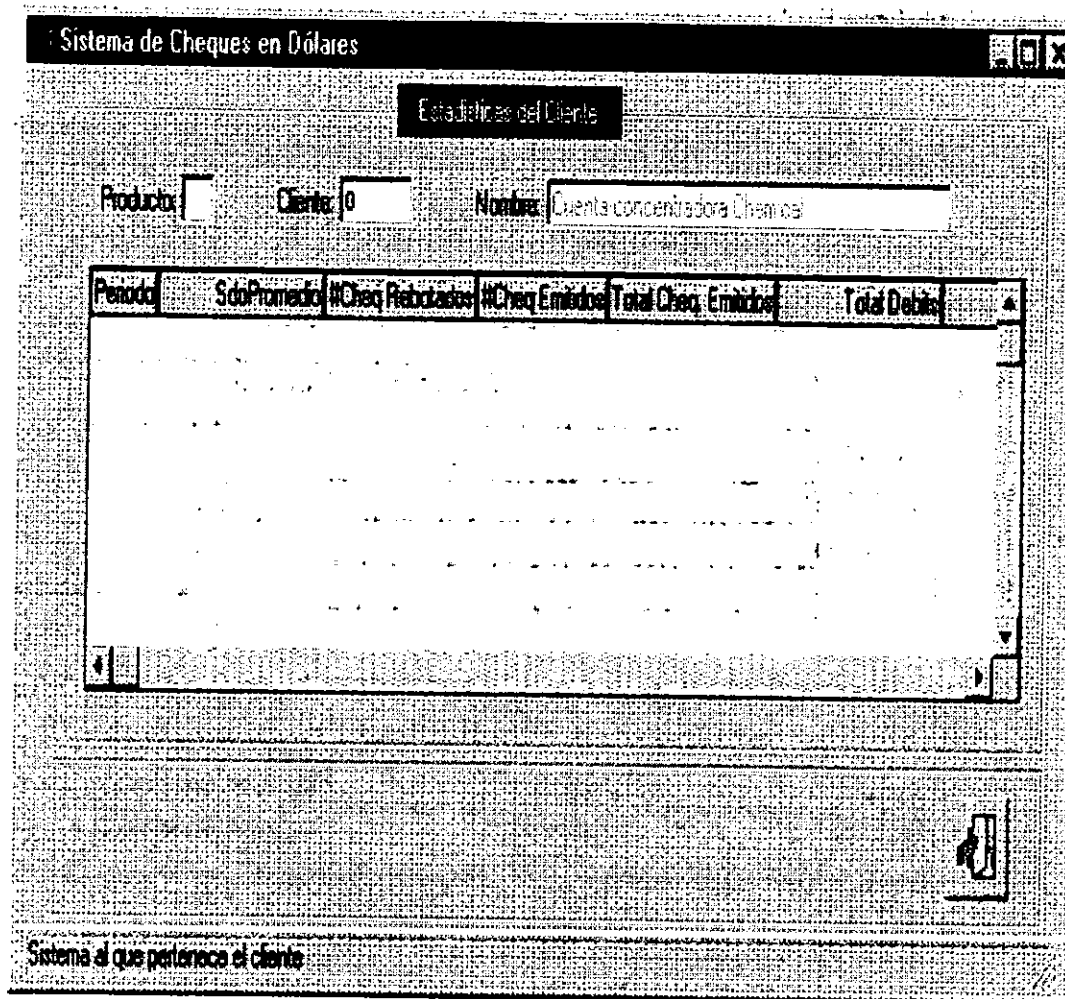


Figura E_35 Estadísticas del Cliente

3.3 Congelar Cheques:

Congelar cheques extraviados que fueron dados de alta para el cliente y que reportó como robados o extraviados. Ver la figura E_36.

The screenshot shows a graphical user interface for a check system. The main window is titled "Sistema de Chequera en Doláres" and contains a sub-window titled "Congelar Cheques (Stop Payment)". The sub-window has several input fields: "Producto" (with a dropdown arrow), "Cuenta" (with a dropdown arrow and the value "0"), "Nombre" (a text field), "Giro" (a text field), "Direccion" (a text field), "Fecha Stop Payment" (a date field with slashes), "Cheque Inicial" (with a dropdown arrow and the value "0"), and "Cheque Final" (with a dropdown arrow and the value "0"). At the bottom right of the sub-window, there are two small icons: one of a person and one of a check. Below the sub-window, there is a label "Sistema al que pertenece el cliente".

Figura E_36 Stop Payment

3.4 Congelar Cuentas:

- 3.4.1 Congelar cuentas de cheques por falta de documentación en la apertura, mal uso de la cuenta o falta de actividad de la cuenta. Ver la figura E_37.

The screenshot shows a graphical user interface for a banking system. The main window is titled "Sistema de Chequera en Dolares" and contains a sub-window titled "Congelar Cuenta de Cheques". The sub-window has several input fields: "Producto" with a dropdown menu, "Cuenta" with a text box containing "0", "Nombre" with a text box containing "Cuenta de cheques", "Saldo" with a text box, and "Congelada:" with a dropdown menu set to "No". At the bottom of the sub-window, there are two icons: a left-pointing arrow and a right-pointing arrow. Below the sub-window, there is a label "Sistema al que pertenece el cliente".

Figura E_37 Congelar Cuentas

3.5 Listado de Cuentas y Cheques congelados:

Reporte de cuentas y cheques congelados que fueron registrados como congelados. Ver la figura E_39.

Sistema de Chequera en Doláres

Listado Cuentas y Cheques Congelados

Que Listado Desea (Consulta/Fecha):

Cuenta **Fecha:**

Producto: Fecha Inicial:

Clientes: Fecha Final:

Impresión por Cuentas y Cheques Congelados

Figura E_39 Listado de Cuentas y Cheques congelados

3.6 Listado de Catálogo de cuentas:

Reporte de cuentas dadas de alta, ordenadas por nombre, #cuenta, area/ división/plaza/sucursal o promotor. Ver la figura E_40.

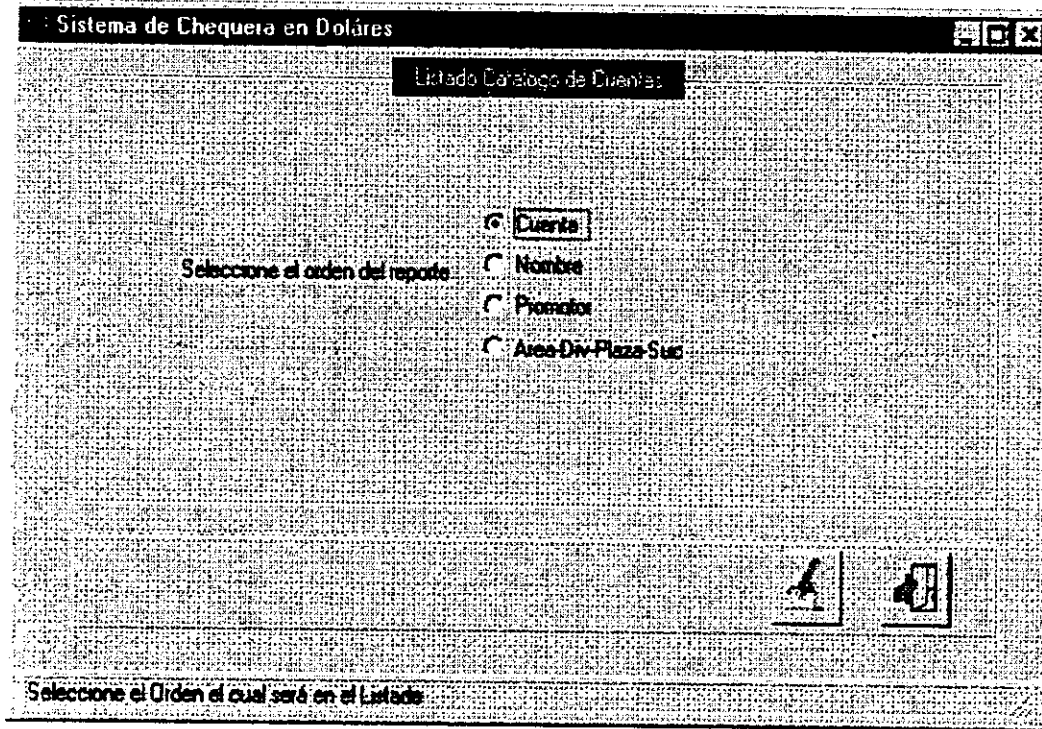


Figura E_40 Listado de Catálogo de Cuentas

3.7 Listado de Estatus de Cta. por fecha:

Reporte de estatus de cuenta ordenado por fecha. Ver la figura E_41.

The screenshot shows a window titled "Sistema de Chequera en Dólares" with a standard Windows-style title bar. The main content area is titled "Listado Cuentas Oídas de día Cancelada". Below this title, there are several input fields: "Estatus (Alta/Cancelada):" with the value "Cancelada" entered; "Fecha inicial:" followed by a date input field containing "/ /"; and "Fecha final:" followed by another date input field containing "/ /". In the bottom right corner of the main area, there are two small icons, one of which appears to be a printer. At the very bottom of the window, a text prompt reads "Entre datos o pulse ESC para finalizar."

Figura E_41 Listado de Estatus de Cta. por fecha

3.8 Listado de Estatus de Cta. por FPA:

Reporte de estatus de cuenta ordenado por fecha y asesor. Ver la figura E_42.

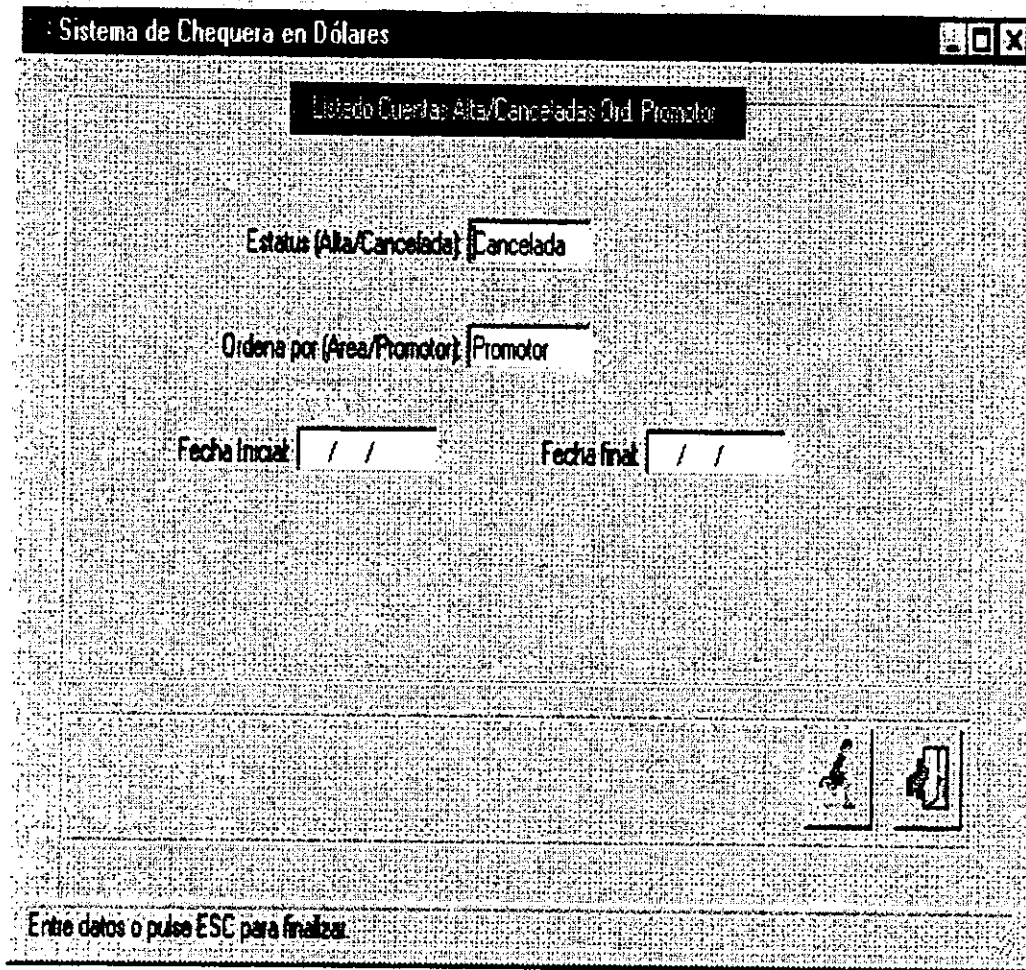


Figura E_42 Listado de Estatus de Cta. por FPA

4.0 El submenú Parámetros tiene las siguientes opciones.

4.1 Parámetros del Sistema:

Alta de claves de movimientos por default, así como nombre de la empresa y límites de saldos de personas físicas y morales. Ver la figura E_43.

Sistema de Cheques en Dólares

Banco:	STUDIOS FINANCIEROS S.A.	Dirección:	LOMAS DE SANTA, FE.	Teléfono-Ciudad:	010045827
			MEXICO, D.F.	Cuenta Global:	99999
			MEXICO	F. Sistema:	03/07/1991
				Cta. Aut. Tarjetas:	89096
Mínimo Persona Física:	2,500.00	Mínimo Persona Moral:	2,500.00		
Sol. Mín. / Con Talonario F:	20,000.00	Sol. Mín. / Con Talonario M:	20,000.00		
Clave de Cheques:	CHEK	CCCA	OPOF	CPFC	CHFC
				Límite Cobrar:	9,000.00
Cve Cargo Cheq:	ICHC	Cve. Rel Cheq:	RCHC	Cve Pago Int:	INPA
				Cve Cargo Promedio:	LOBC
Pag Inv Debt:	TRPY	Pag Inv Cae:	TTPY	Alt Cred Debt:	TRLO
				Alt Cred Cae:	NELO
Alt Inv Debt:	TTIN	Alt Inv Cae:	TRIN	Pag Cred Debt:	LOPY
				Pag Cheq Debt:	CPFC
Pag Aut TarCae:	FTDE	Pag Cred Cae:	TLPY	Pag Aut Tarde:	FUTR
				Otro Pago:	OTAP

Save Reset Add Copy Delete

Proporcione el nombre del banco.

Figura E_43 Parámetros del Sistema

4.2 ABC de tipos de movimientos:

Mantenimiento claves movimientos credits, debits para la afectación de las cuentas. Ver la figura E_44.

Sistema de Cheques en Dólares

ABC de Tipos de Movimientos

Clave Movto:

Descripción Larga:

Descripción Corta:

Afecta Saldo Total Cliente: Naturaleza (Credit/Debit):

Afecta Saldo Disponible: Es Ingreso?:

Afecta la Cuenta Global: Es Gasto?:

Generado por: Dias p/afectar Saldo:

(EC) Captura de Estado de Cuenta
(DM) Captura de Otros Movtos
(AU) Generado Autom.
(CH) Captura de Chemical Bank

Monto Fijo:

Save Reset Add Copy Delete

Navigation icons: Home, Back, Forward, End

Figura E_44 Mantenimiento a Tipos de Movimientos

4.3 ABC de Overnight con Barclays:

Mantenimiento a las inversiones de tipo "overnight" en Barclays Bank, es una opción de inversión en activos con los recursos de los pasivos. Ver la figura E_45.

The screenshot shows a window titled "Sistema de Cheques en Dólares" with a sub-header "ABC de 'Overnight' de Chequeras". The interface contains the following fields and controls:

- id_producto:** []
- id_cliente:** [99999]
- f_movta:** [05/27/1994]
- ON** (Section Header)
- Plazo:** [4]
- Tasa:** [3.7500]
- Fecha Venc.:** [05/31/1994]
- Monto:** [1,400,000.00]
- Chequera** (Section Header)
- Plazo:** [0]
- Tasa:** [.0000]
- Fecha Venc.:** [05/02/1950]
- Monto:** [0.00]

At the bottom of the window, there is a toolbar with buttons labeled "Save", "Reset", "Add", "Copy", "Delete", and a printer icon. Below the toolbar are navigation arrows: a left arrow, a double left arrow, a double right arrow, and a right arrow.

Figura E_45 ABC de Overnight con Barclays

4.4 Cancelar Cuenta:

Cancelación de cuentas y cambio de estatus, ya sea por inactividad o cancelación por el usuario. Ver la figura E_46.

The screenshot shows a window titled "Sistema de Chequera en Dólares" with a sub-header "Cancelar Cuenta de Cheques". The form contains the following fields:

- Producto:
- Cuenta:
- Nombre:
- Saldo:
- Estatus de Cuenta (Alta/Cancelada):

At the bottom of the window, there is a label "Sistema al que pertenece el cliente" and two small icons.

Figura E_46 Cancelar Cuenta de Cheques

4.5 Total saldo Cheques:

Reporte que suma los saldos disponibles de Cuentas Fisicas, Morales y cuantas cuentas están vigentes por tipo de cuenta. Ver la figura E_47.

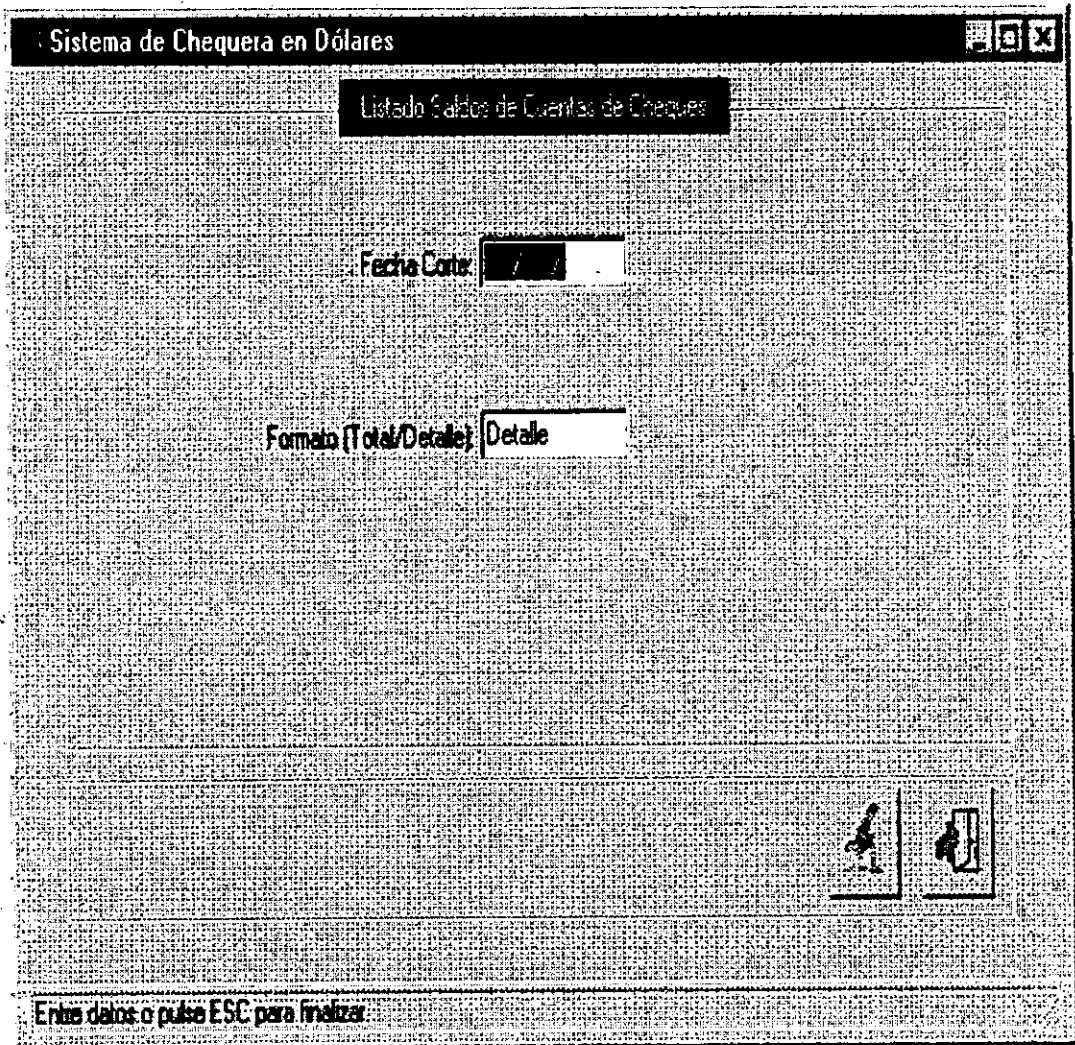


Figura E_47 Listado Saldos de Cuentas de Cheque

4.6 Alta de Instrucciones Desembolso:

Alta de instrucciones de transferencia por cuenta, que permitira enviar fondos a cualquier Banco mundial. Ver la figura E_48.

Sistema de Cheques en Dólares

Instrucciones de Desembolso

Producto: Cuenta: Usar Preformato: No

Your Ref	Ref Ref	ci_party_type	ci_party_id	ci_party_name	ci_party_addr

Tache la Cuenta del Cliente

Figura E_48 Alta de Instrucciones de desembolso

4.7 Listado de Movimientos Automáticos:

Reporte de movimientos automáticos generados por el sistema internamente. Ver la figura E_49.

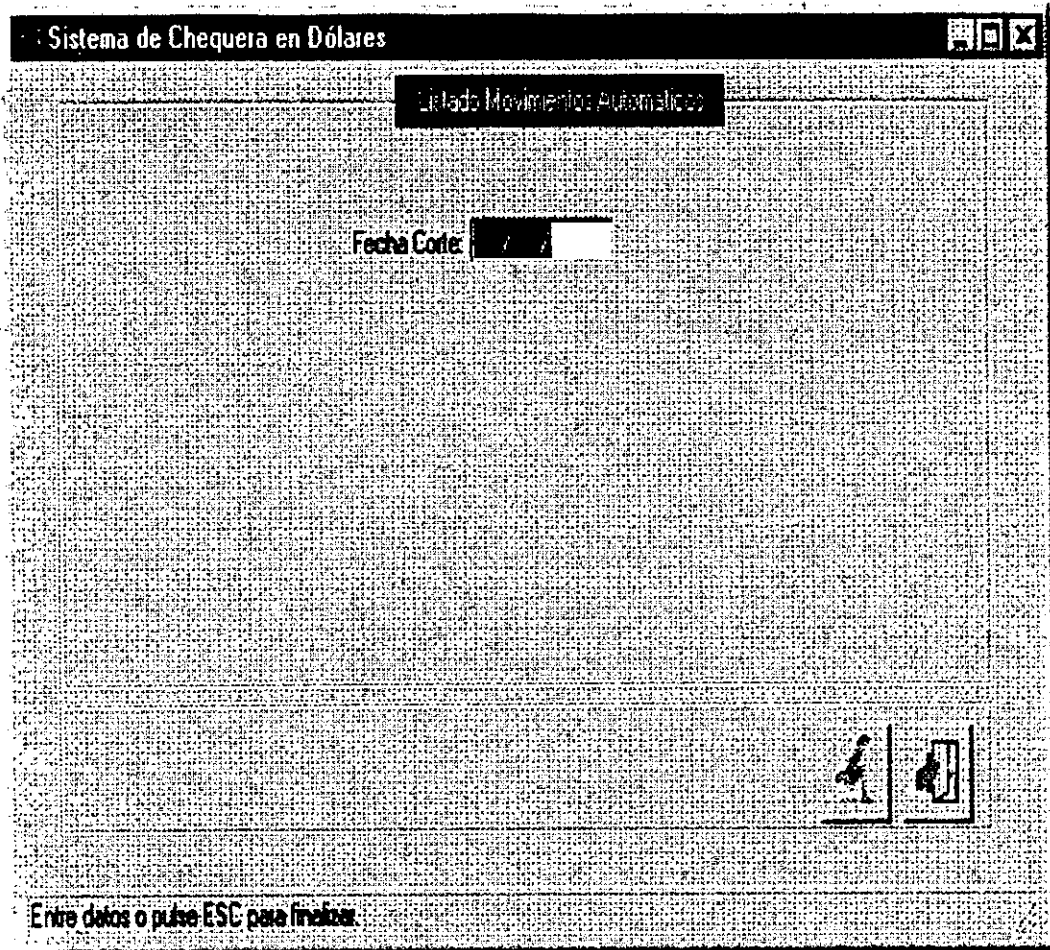


Figura E_49 Listado de Movimientos Automáticos

4.8 Listado Cheques Centros Financieros:

Reporte de Cheques cobrados en los centros regionales, generando el monto, como se cobro y que cuentas cobraron en el centro regional. Ver la figura E_50.

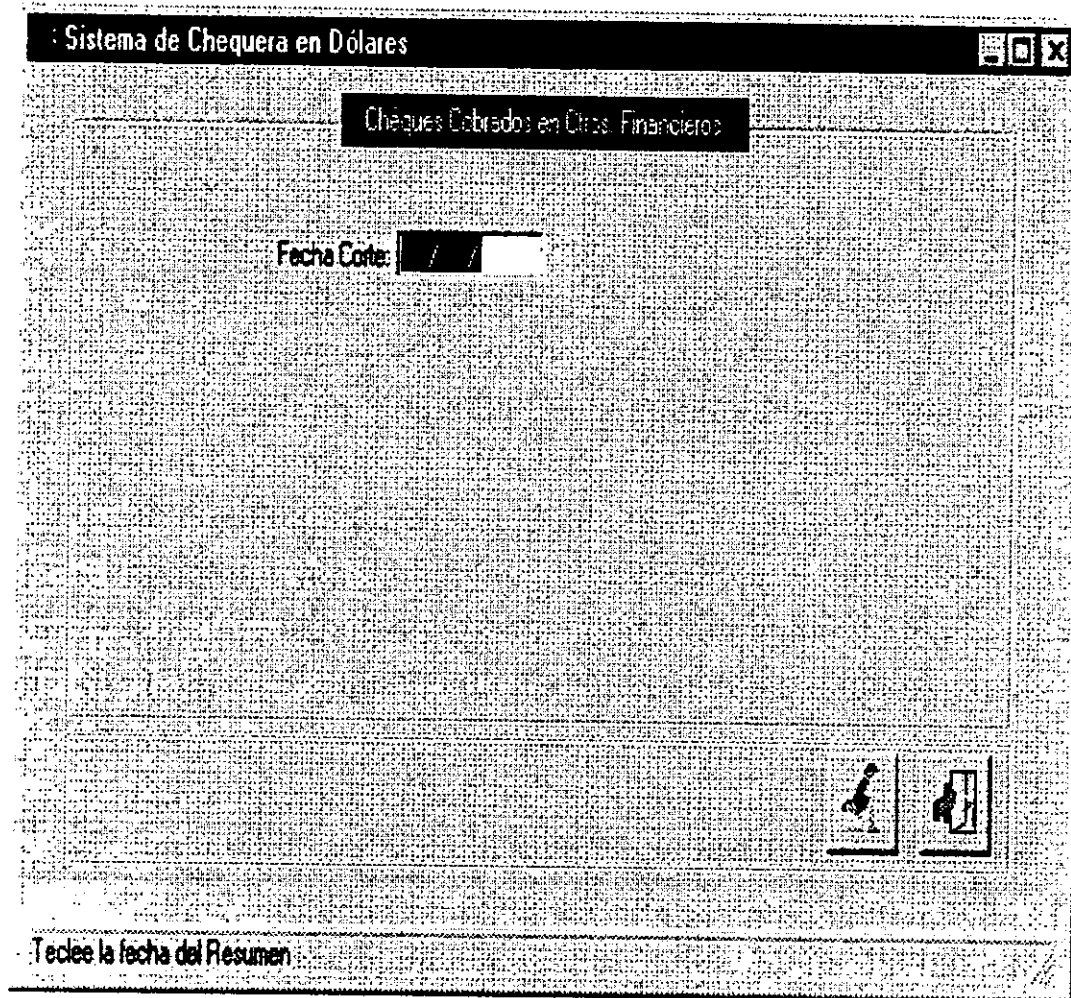


Figura E_50 Listado Cheques Centros Financieros

4.9 Total pagos movimientos:

Reporte de los pagos efectaodos por las cuentas con los preformatos usados en la liberación del pago. Ver la figura E_51.

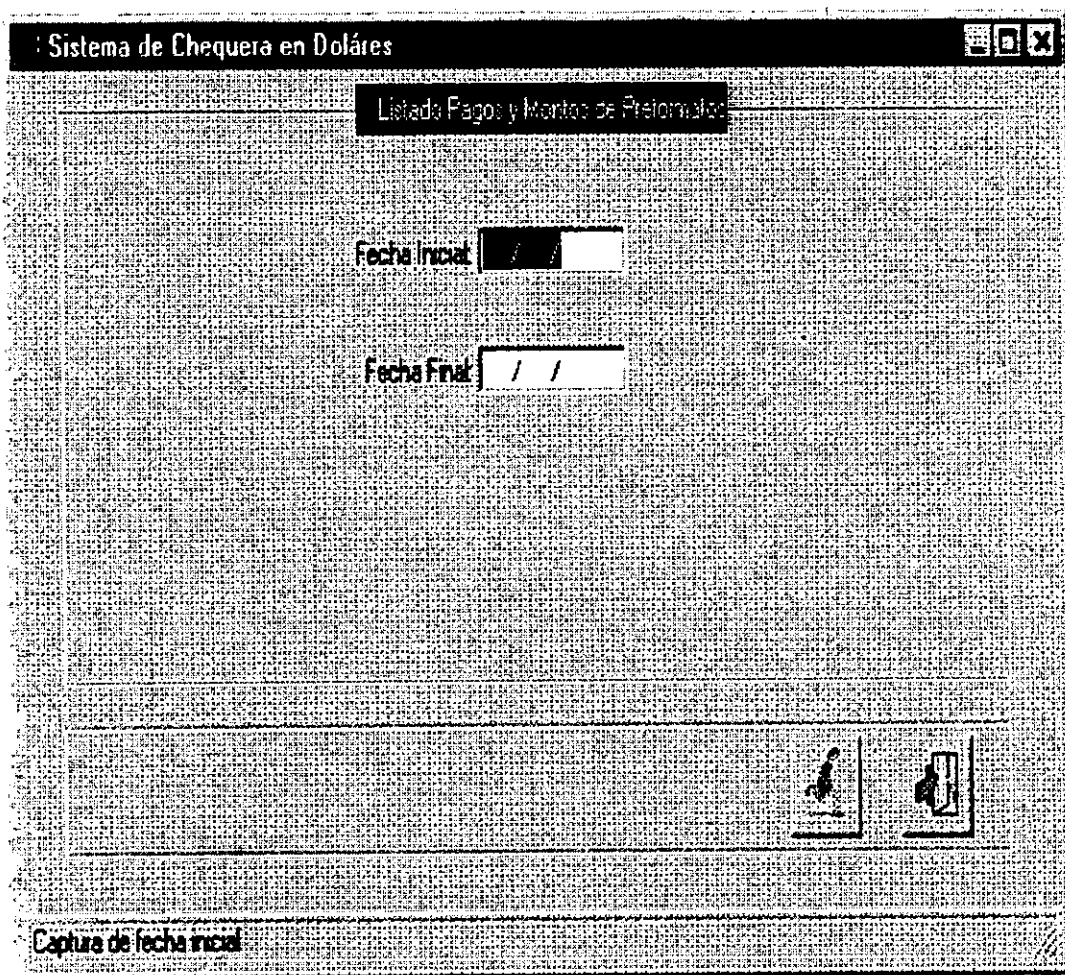


Figura E_51 Listado Total pagos movimientos

4.10 Listado de saldos rangos:

- Listado de los movimientos por rango de las cuentas, son aquellos que son ordenados por el monto efectuado en los movimientos de la cuenta; es generado por un rango de monto y fecha de corte. Ver la figura E_52.

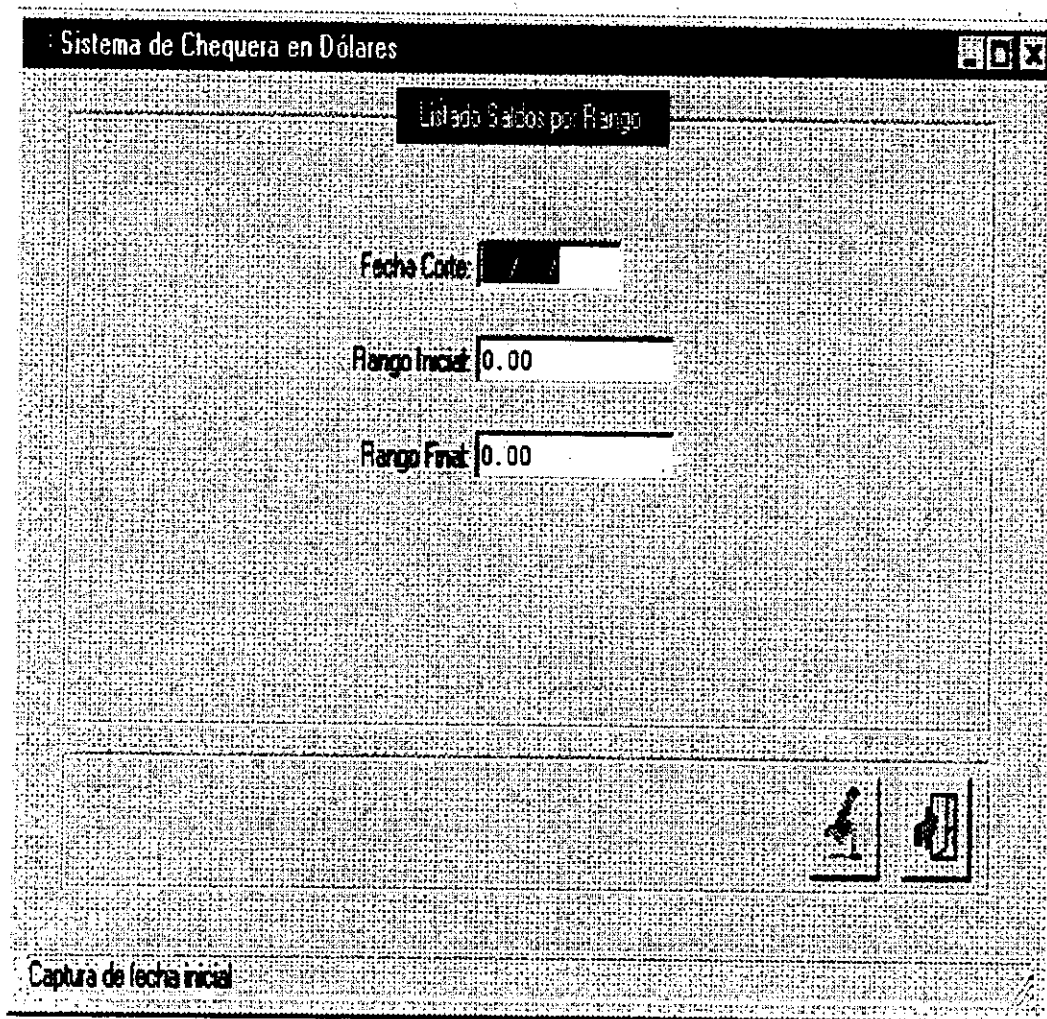


Figura E_52 Listado de Saldos por Rango

5.0 El submenú Memos tiene las siguientes opciones.

5.1 Test de Barclays:

Memo de traspaso de fondos de Barclays Bank a Chase Bank, capturando el monto, firmas y Fecha de corte. Ver la figura E_53.

The screenshot shows a window titled "Sistema de Chequera en Dólares" with a sub-window titled "Memo Test Barclays". The form contains the following fields and elements:

- Fecha Valor:** A date selection field.
- Cantidad:** A numeric field containing the value "0.00".
- Cuenta:** A text input field.
- Firma 1:** A large text input field for the first signature.
- Firma 2:** A large text input field for the second signature.
- Bottom right:** Two icons, one resembling a printer and the other a window.
- Bottom left:** The text "Captura de fecha inicial" is visible.

Figura E_53 Memo Test Barclays

5.2 Acuse de Chequera Solicitada:

Memo de Chequera solicitada y solicitud de chequera definitiva para el área de administración y documentación, capturando Fecha del memo, cuenta del cliente y chequera definitiva. Ver Figura E_54.

The screenshot shows a window titled "Sistema de Chequera en Dolares" with a sub-header "Memo Acuse Chequera". The form contains the following fields and elements:

- Fecha Memo:** A date input field.
- Tipo de Chequera (Definitiva/Provisional):** A dropdown menu with "Provisional" selected.
- Producto:** A dropdown menu with "Q" selected.
- Cuenta:** A dropdown menu with "0" selected.
- Cuenta correspondiente (Unidad):** A text input field.
- Navigation icons:** Two icons at the bottom right, one with a left arrow and one with a right arrow.
- Label:** "Captura de fecha inicial" at the bottom left.

Figura E_54 Memo Acuse Chequeras

5.3 Fondos Casa de Cambio:

Memo de casa de cambio donde se notifica que se cargará a la cuenta de cheques de un cliente y se efectuará una compra de dólares americanos. Se captura el monto, fecha, cantidad y Cuenta a aplicar. Ver la figura E_55.

The screenshot shows a window titled "Sistema de Chequera en Doláres" with a sub-header "Memo Casa de Cambio". The form contains the following fields:

- Fecha Valr: [/ /]
- Cantidad: 0.00
- Cliente: []
- Cantidad Letra: []

At the bottom right, there are two icons: a printer and a refresh/cancel icon. At the bottom left, the status bar reads "Captura de fecha inicial".

Figura E_55 Memo Casa de Cambio

5.4 Autorización Cheques:

Memo autorización de pago de un documento en casa de cambio en la cuenta de cheques de un cliente. Se captura documento, importe y cuenta. Ver la figura E_56.

The screenshot shows a graphical user interface for a check system. The window title is "Sistema de Chequera en Dólares". Inside, there is a sub-header "Autorización Casa de Cambio". The form includes the following fields and controls:

- Documento:** A text input field.
- Cliente:** A dropdown menu with the value "Q".
- Cuenta:** A text input field with the value "0".
- Importe:** A text input field with the value "0.00".
- Cuenta concentradora:** A dropdown menu with the value "Cheavol".
- Teclee el Documento:** A button at the bottom left.
- Icons:** Two small icons at the bottom right, one showing a person and the other showing a document.

Figura E_56 Memo de Autorización de Cheques

5.5 Resumen casa de cambio:

Memo resumen con los cobros de los movimientos efectuados por los diferentes clientes en la casa de cambio. Se capturará la fecha del memo. Ver Figura E_57.

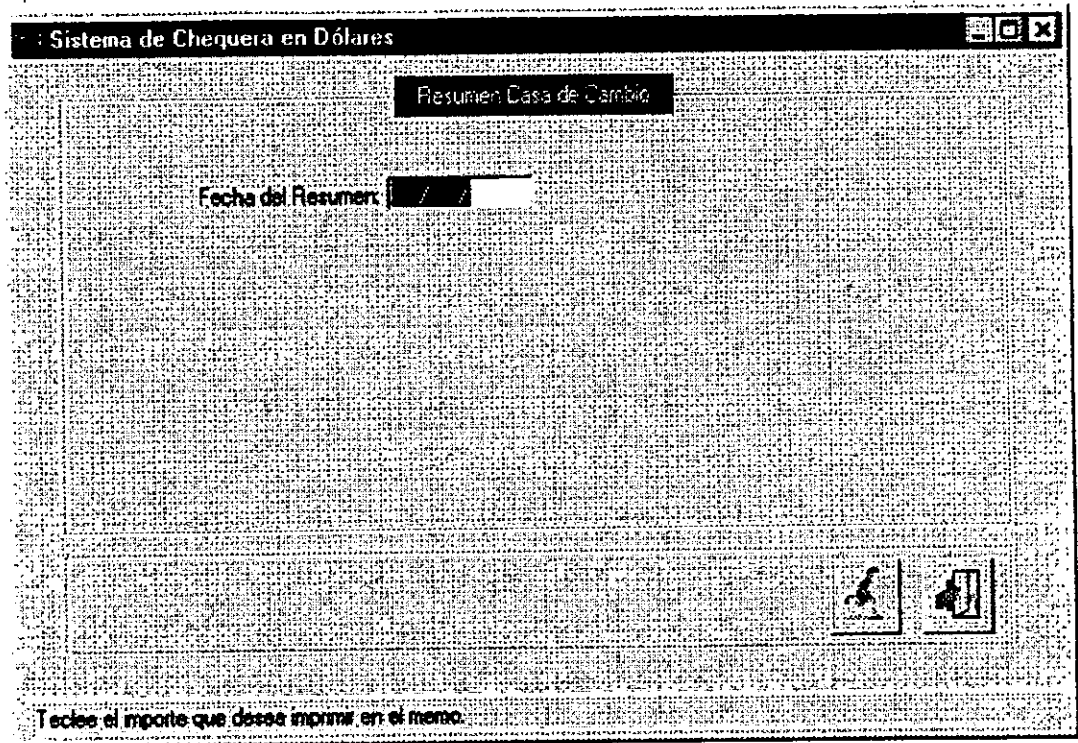


Figura E_57 Resumen Casa de Cambio

5.6 Carta de cheques rebotados:

Memo de cheques rebotados que son enviados a los clientes porque su cheque fue devuelto por Chase Bank o Barclays Bank. Se capturará fecha, monto, cuenta y número de cheque. Ver la figura E_58.

The screenshot shows a window titled "Sistema de Chequera en Dólares" with a sub-header "Carta de Cheques Rebotados". The form contains the following fields and values:

- Fecha Corte: [/ /]
- Producto: [Q]
- Cuenta: [0]
- Cuenta concentradora: [Cuenta concentradora Ethenical]
- Cheque: []
- Monto: [0.00]

At the bottom right, there are two icons: a magnifying glass and a printer. At the bottom left, there is a footer instruction: "Teclee la fecha del Resumen".

Figura E_58 Carta de Cheques Rebotados

5.7 Cheques cobrados Ctros. Financieros:

Reporte de los cheques cobrados en los centros financieros ordenado por monto. Se capturará únicamente fecha de corte. Ver la figura E_59.

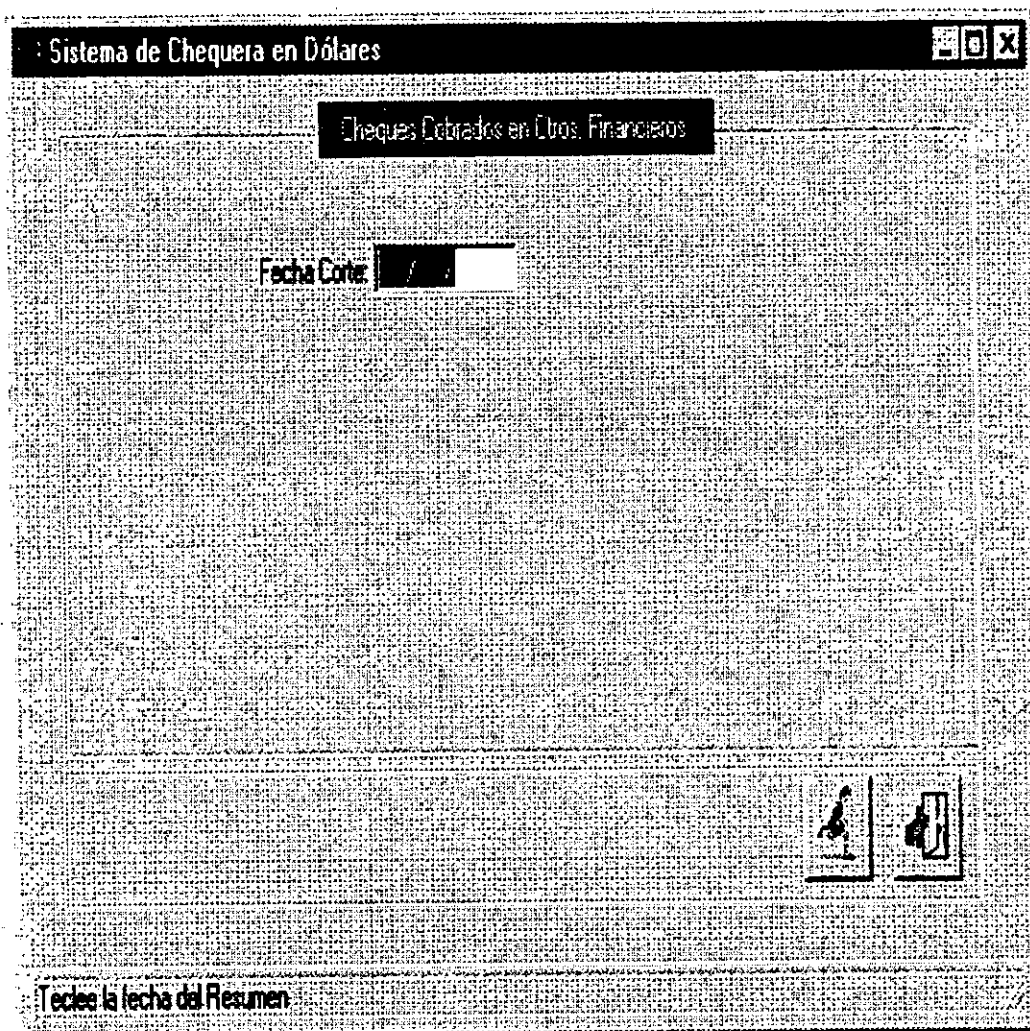


Figura E_59 Cheques cobrados Ctros. Financieros

6.0 El submenú Chemical tiene las siguientes opciones.

6.1 Movimientos Clientes Chemical:

Alta de Movimientos con naturaleza de tipo "CH" Chemical que afectan a las cuentas de los clientes y el saldo de la cuenta concentradora. En el alta del movimiento se dará una fecha valida (día hábil y que no sea fin de semana). Los campos a capturar son: producto, cuenta, clave del movimiento Debit/Credit, referencia y monto afectando el saldo del cliente directamente y teniendo validaciones en la cuenta cancelada, congelada, cheque congelados o tipos de movimiento de otra naturaleza que no corresponda a un movimiento de tipo Chase. Ver la figura E_60.

The screenshot shows a software window titled "Sistema de Cheques en Dólares" with a sub-window titled "Captura de Movimientos Chemical Clientes".

At the top, there is a label "Fecha de los Movimientos a Capturar:" followed by a date input field.

Below this is a table with the following columns: "Producto", "Cliente", "Cve. Movim.", "Descripción", "Referencia", "Debits", and "Credits". The table is currently empty.

Below the table, there are two input fields: "Cliente:" and "Saldo Disponible:".

At the bottom of the window, there is a button labeled "Teclee la Fecha Capturar" and two small icons.

Figura E_60 Captura de Movimientos de Chemical Clientes

6.2 Movimientos Global Chemical:

Alta de movimientos con naturaleza de tipo "CHP" Chemical que afecta a la cuenta concentradora directamente. En el alta del movimiento se dará una fecha valida (día hábil y que no sea fin de semana). Los campos a capturar son: Producto, cuenta, Clave del Movimiento Debit/Credit, referencia y monto afectando el saldo del cliente directamente y teniendo validaciones en la cuenta cancelada, congelada, cheque congelados o tipos movimiento con otra naturaleza que no corresponda a movimiento de tipo Chase. Ver la figura E_6.

The screenshot shows a window titled "Sistema de Cheques en Dólares" with a sub-header "Captura de Movimientos Chemical Global". Below the sub-header is a date field labeled "Fecha de los Movimientos a Capturar". The main area contains a table with the following columns: "Producto", "Cve. Cuenta", "Descripción", "Referencia", "Debits", and "Cred". Below the table are two input fields: "Cliente" and "Saldo Final". At the bottom right, there are two icons and a button labeled "Salir del p". At the bottom left, there is a label "Teclea la Fecha Capturar".

Figura E_61 Captura de Movimientos de Chemical Global

6.3 Cancelación movimientos Ctes. Chemical:

Eliminación de movimientos que afectaron a clientes del Chemical Bank; dependiendo de la naturaleza, se decrementará el saldo de quien lo generó. El campo a capturar es la fecha del movimiento. Ver la figura E_62.

The screenshot shows a window titled "Sistema de Cheques en Dólares" with a sub-header "Eaja de Movimientos". Below the header is a date input field labeled "Fecha de los Movimientos a Capturar:". The main area contains a table with the following columns: "Producto", "Cliente", "Cve. Movto.", "Descripción", "Referencia", "Debits", and "Credits". The table is currently empty. Below the table are two input fields: "Cliente:" and "Cia. Global:". At the bottom right, there is a button labeled "Salir del progr". A footer message reads "Entre datos o pulse ESC para finalizar".

Figura E_62 Cancelación movimientos Ctes. Chemical

6.4 Cancelación movimientos Global Chemical:

Eliminación de movimientos que afectan únicamente la cuenta concentradora de Chemical. El campo a capturar es la fecha de movimiento. Ver la figura E_63.

The screenshot shows a window titled "Sistema de Cheques en Dólares". Inside, there is a sub-window titled "Baja Movimientos Global Global". Below this title is a date field labeled "Fecha de los Movimientos a Capturar". The main area contains a table with the following columns: "Producto", "Cta. Cuenta", "Descripción", "Referencia", "Detals", and "Cred.". The table is currently empty. Below the table, there are several input fields: "Cuenta:", "Saldo Final:", and "Registros dados de al:". At the bottom of the window, there is a "Teclas de Fecha Captura" field and some navigation icons.

Figura E_63 Cancelación movimientos Global Chemical

6.5 Consulta Saldo Chemical:

Consulta de movimientos de la cuenta concentradora; se consultará por el campo producto, cuenta y fecha (periodo).
Ver la figura E_64.

Sistema de Cheques en Dólares

Consulta de Estado de Cuenta Química

Producto: Cuenta: Fecha Inicial: / /
Fecha Final: / /

Fecha Movto	Cve. Movto	Descripción	Referencia	Creda	Débito	Tr.

Saldo Disponible: 0.00 Saldo Total: 0.00

Teclas la Clave del Producto:

Figura E_64 Consulta Saldo Chemical

6.6 Listado Movimientos Chemical:

Reporte de los movimientos que fueron dados de alta en la operación del día. Ver la figura E_65.

Sistema de Chequera en Dólares

Listado de Estado de Cuenta Global

Fecha Corte: / /

Movtos. de Edo. Da. Fax: No

Otros Movimientos: No

Movtos. de Chemical: No

Entre datos o pulse ESC para finalizar

Figura E_65 Listado de Estado de Cuenta Global

6.7 Memo Traspaso Chemical:

Traspaso del saldo de la cuenta concentradora para invertir los recursos fuera de la cuenta, usando los preformatos previamente dados de alta a la cuenta concentradora. Ver la figura E_66.

Sistema de Cheques en Dólares

Libros de Pago de la Cuenta Chemical

Fecha Movtos: Saldo Final: Monto a Pagar:

Saldo Actual: Cuenta a Debitar: (9876543-1234)

Cuenta:

Your Ref	Ref Ref	#Pago De	Instruc Pago	CPartyType	CPartyId	CPartyName
----------	---------	----------	--------------	------------	----------	------------

Teclee la fecha del alta de los Movtos.

Figura E_66 Memo Traspaso Chemical

6.8 Contabilización Chemical:

Contabilización de los movimientos que se encuentran registrados en la cuenta concentradora. El campo a capturar es la fecha. Ver la figura E_67.

The screenshot shows a window titled "Sistema de Chequera en Dólares" with a sub-header "Contabilizar Mov. Chemical". Below this, there are two input fields: "Fecha Cortar" and "No. de Póliza". The "No. de Póliza" field contains the value "000,000". At the bottom of the window, there is a status bar that reads "Entre datos o pulse ESC para finalizar".

Figura E_67 Contabilización de Chemical

6.9 Descontabilización Chemical:

Regreso de la contabilización de los movimientos en la cuenta concentradora. El campo a capturar es la fecha. Ver la figura E_68.

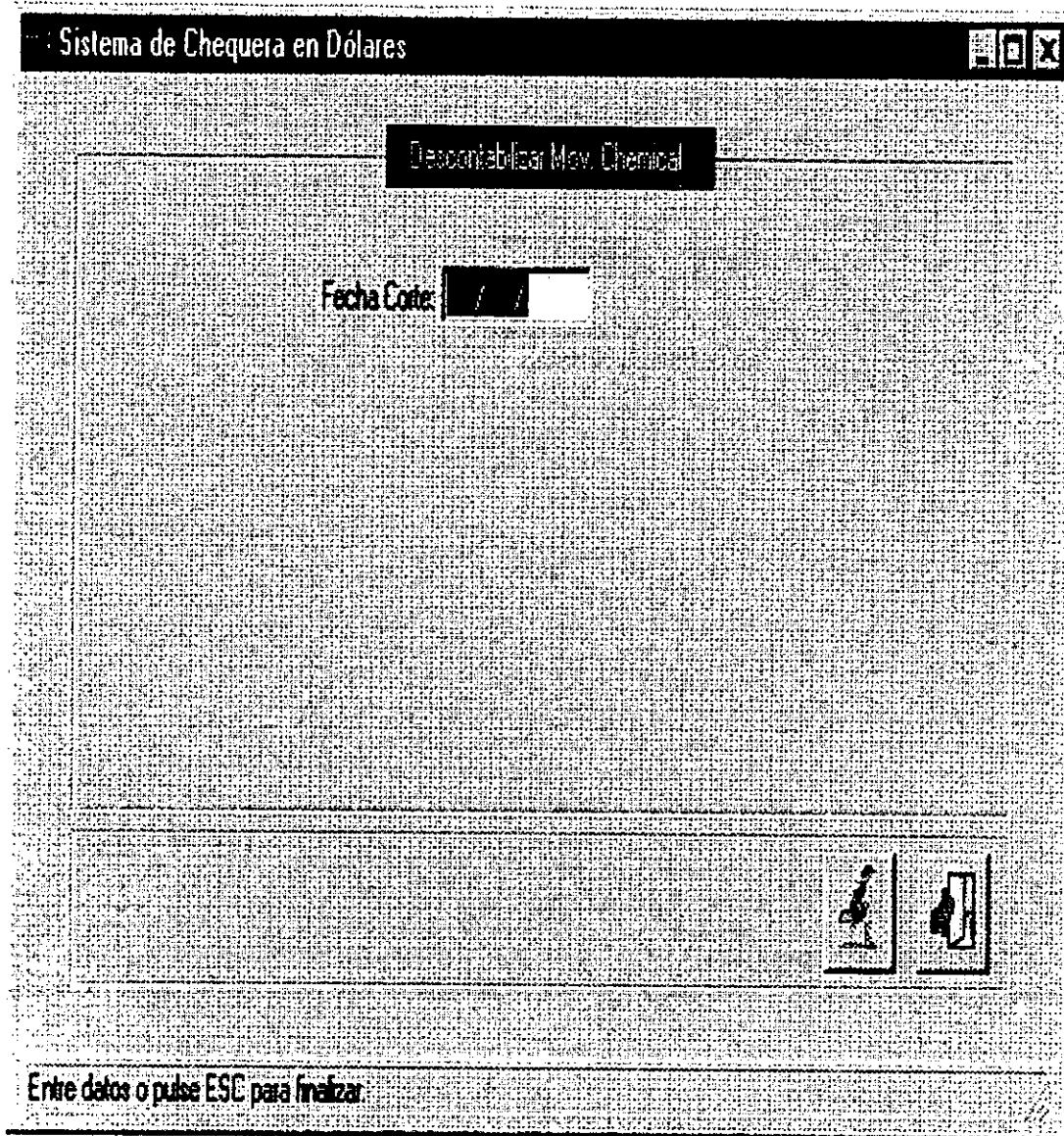


Figura E_68 Descontabilización Chemical

6.10 Autorización Saldo Chemical:

Proceso de verificación de saldos iniciales y finales de la cuenta concentradora. El campo a capturar es la fecha de actualización. Ver la figura E_69.

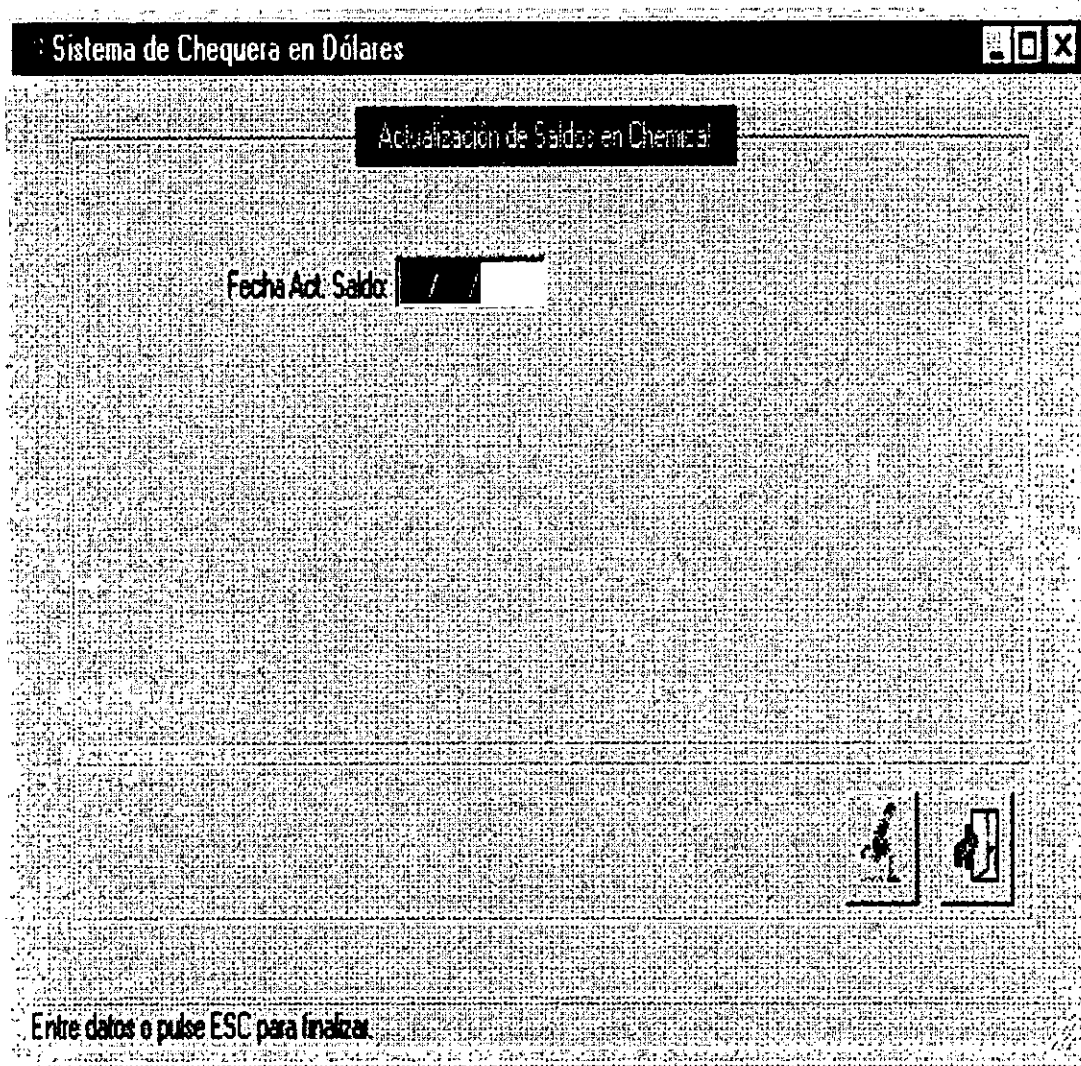


Figura E_69 Actualización de Saldos del Chemical

7.0 El Submenú de Pagos tiene las siguientes opciones.

7.1 ABC Area/División/Plaza/Sucursal:

Mantenimiento a las ubicaciones que corresponden a los clientes para las estadísticas de apertura de cuentas. La captura serán las claves de área, división, plaza y sucursal. Ver la figura E_70.

Sistema de Cheques en Dólares

ABC de Sucursales

Ubicación: Zona

Área 1 División 1 Plaza 1

Sucursal: Nombre: Suc.27 Mariano Escobedo

Dirección:

Calle: Mariano Escobedo 555 Teléfono 1: (915) 254-86-70

Colonia: Anzures Teléfono 2: (915) 254-84-31

Ciudad: Mexico D.F. C.P.: 40000 Teléfono 3: () - -

Estado: 9 País: ME Fax: () - -

Save Reset Add Copy Delete Print

Figura E_70 ABC Area/División/Plaza/Sucursal

7.2 ABC Promotores:

Mantenimiento a los promotores que están dados de alta en los centros regionales. La captura es clave de promotor, nombre y ubicación. Ver la figura E_71.

The screenshot shows a software window titled "Sistema de Cheques en Dólares". Inside the window, there is a sub-window titled "ABC de Promotores". The form contains the following fields and values:

- Cve. Prom.: 1
- Nombre: ASESOR SIN NOMBRE
- Cve. Area: 38
- Cve. Division: 1
- Cve. Plaza: 1
- Cve. Sucursal: 1
- Origen: S

At the bottom of the form, there is a control bar with the following buttons: Save, Reset, Add, Copy, Delete, and a small icon. Below the control bar, there are three navigation buttons: a left arrow, a right arrow, and a double right arrow.

Figura E_71 ABC Promotores

7.3 ABC Países:

Mantenimiento de Países, con los días inhábiles que corresponden a la operación del sistema! Ver la figura E_72.

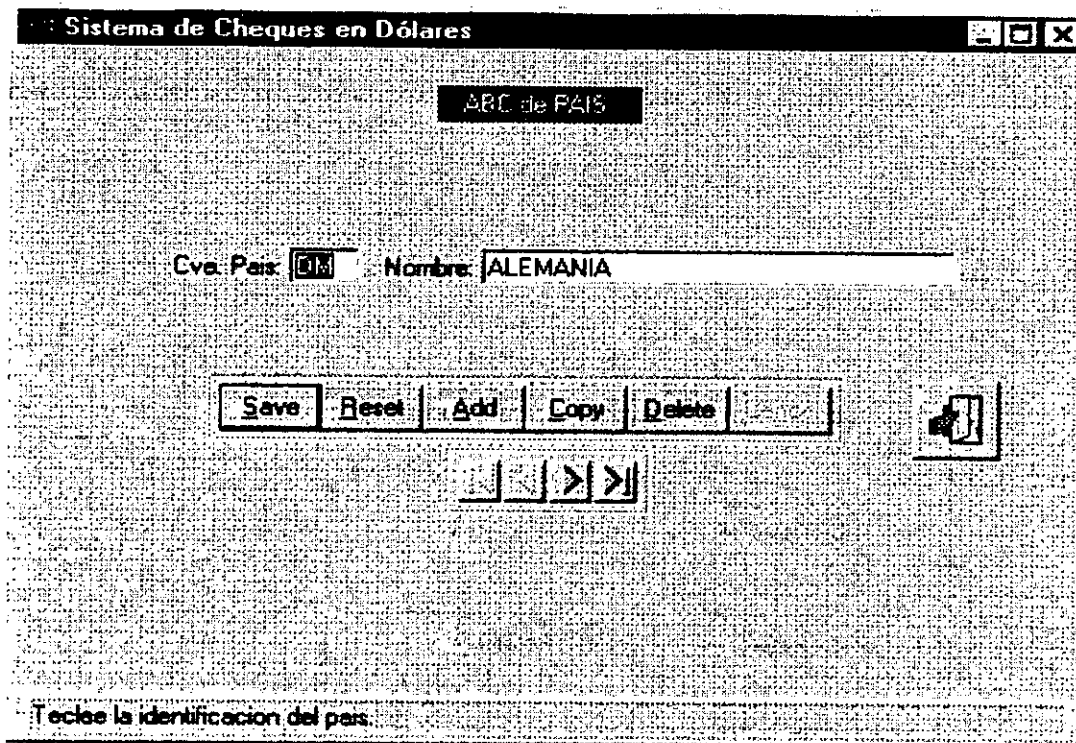


Figura E_72 ABC Países

7.4 ABC Bancos:

Mantenimiento a Bancos corresponde al alta de los preformatos de la salida de dinero que corresponde a los pagos. La captura son datos generales de los bancos "Swift, Aba, Account, etc.". Ver la figura E_73.

Sistema de Cheques en Dólares

ABC de Bancos

Nombre del Banco: Cuenta:

Dirección: ABA:

CHIPS LID: Swift:
 CHIPS Part: St:

Navigation arrows: [Left] [Right] [Home] [End]

Figura E_73 ABC Bancos

7.5 Generación de Ctas. de Clientes:

Generación de cuentas para la apertura de cuentas con el algoritmo de dígito verificador. La captura es cuenta inicial y cuantas cuentas solicitará. Ver la figura E_74.

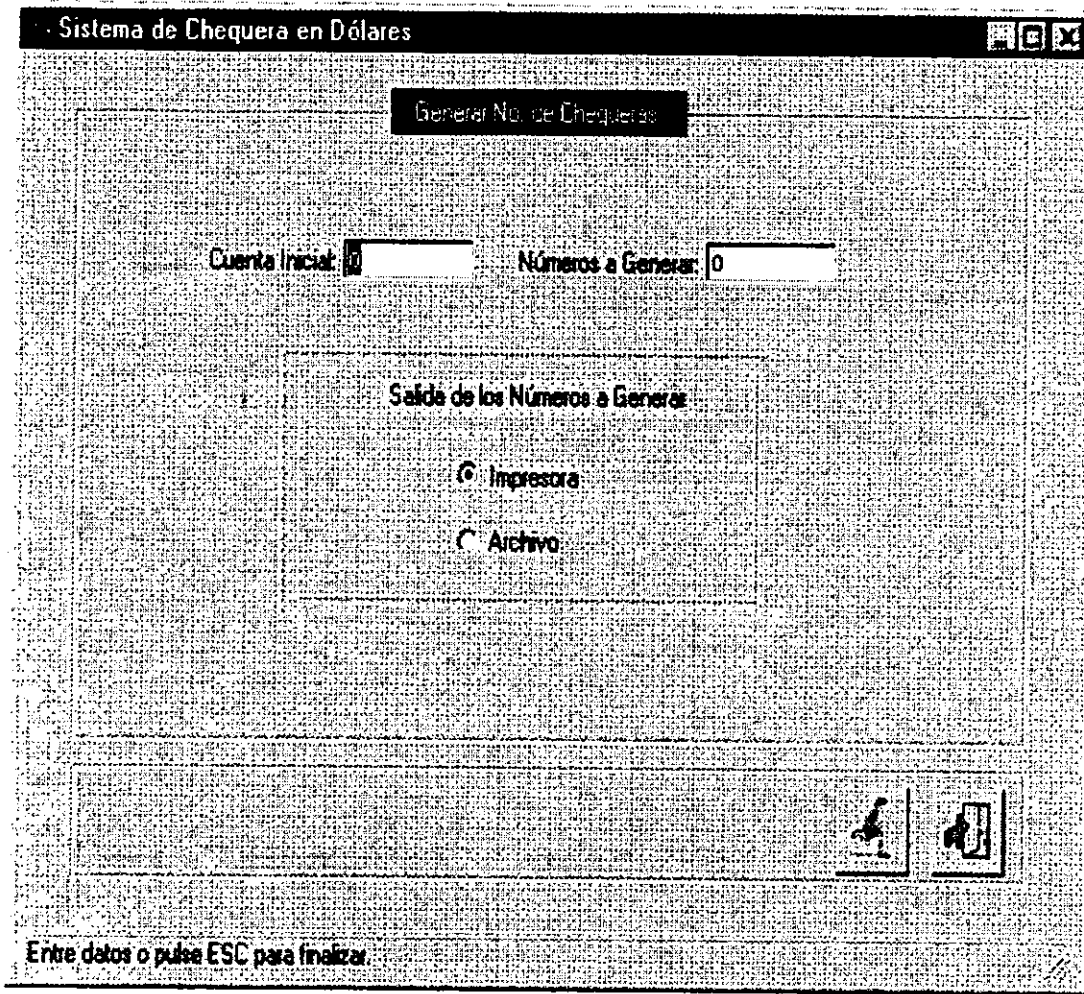


Figura E_74 Generación de Ctas. de Clientes

7.6 Listado de Cheques cobrados:

Reporte de cheques cobrados en los centros regionales para la verificación de los mismos. Se captura un rango de fechas. Ver la figura E_75

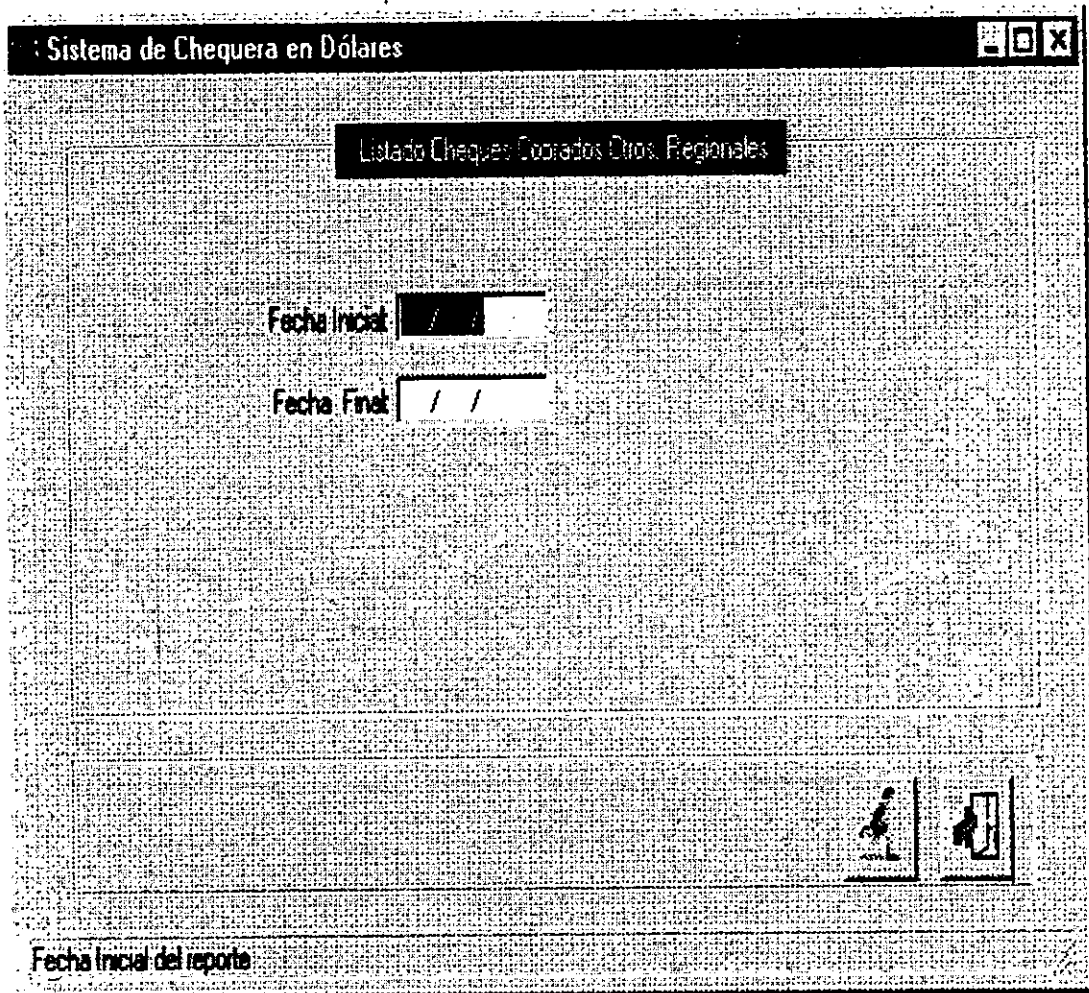


Figura E_75 Listado de Cheques cobrados

7.7 Validación de pagos:

Validación de las salidas de dinero de la cuenta concentradora de Chemical Bank hacia otros bancos. La captura son todos los pagos que se generarán para su impresión "T2". Ver la figura E_76.

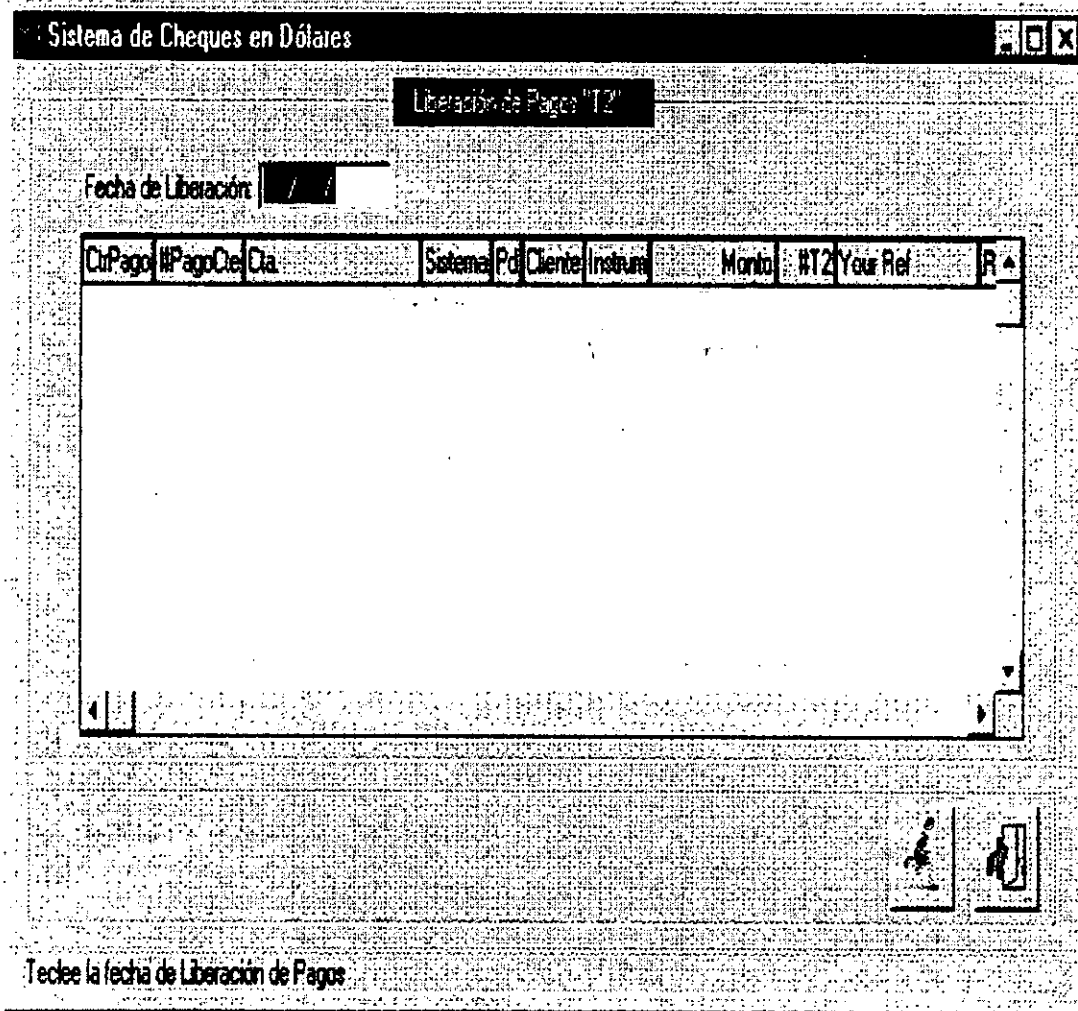


Figura E_76 Validación de pagos

7.8 ABC de Centros Financieros:

Mantenimiento a los centros financieros para su cobro de cheques. La captura es alta de centro regional. Ver la figura E_77.

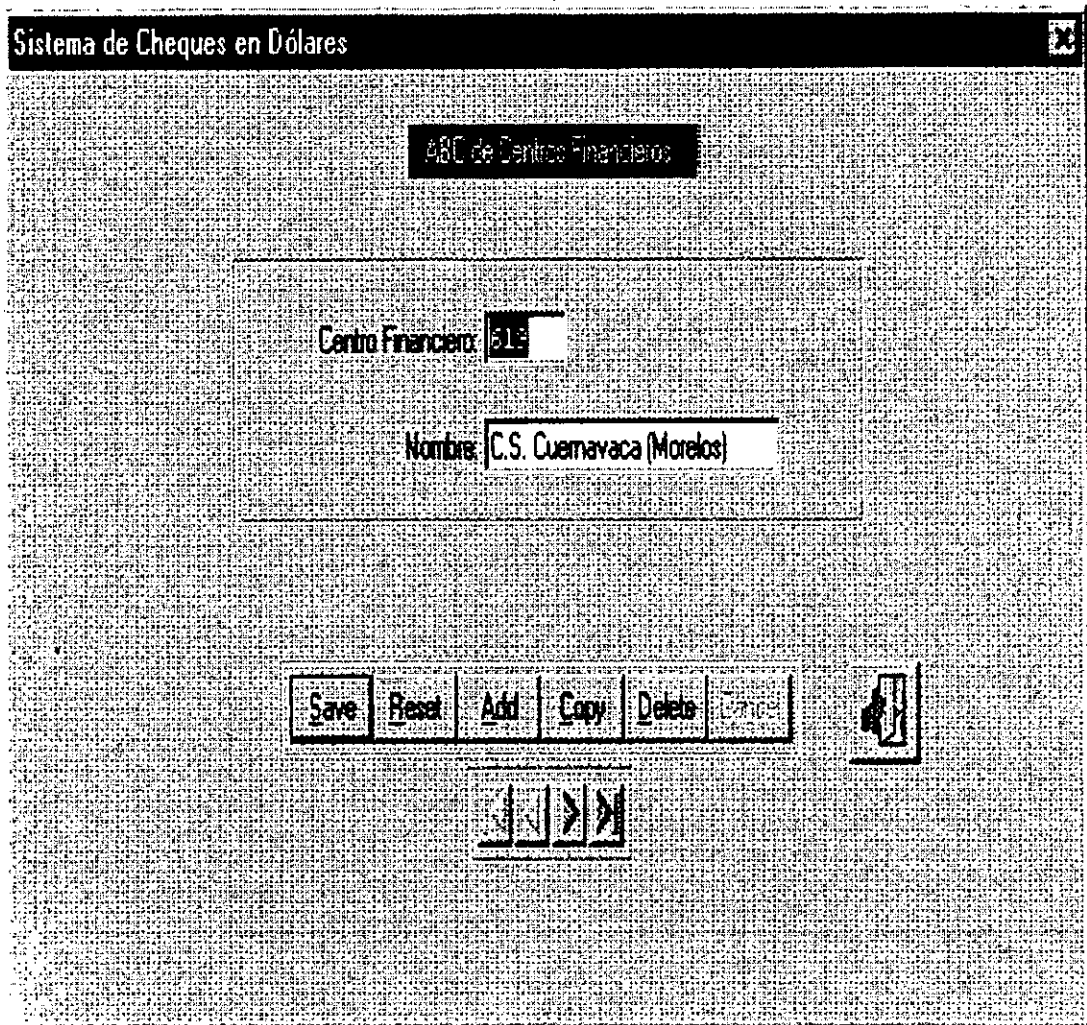


Figura E_77 ABC de Centros Financieros

7.9 ABC de Asesores:

Mantenimiento de los asesores que solamente autorizarán el cobro de cheques en los centros regionales. La captura de asesores corresponde al centro regional en que se encuentra el asesor. Ver la figura E_78.

Sistema de Cheques en Dólares

ABC de Asesores

Id del Asesor: 018 Nombre: Raul Guayante Rivera

Stat: Alta: Cancelada Centro Financiero: 19354

Nombre del Centro Financiero: C.S. Nogales (Sonora)

Save Reset Add Copy Delete Print

Navigation arrows: [Left] [Right] [Up] [Down]

Figura E_78 ABC de Asesores

7.10ABC de Días Inhábiles:

Mantenimiento de los días inhábiles que no permitirán la operación de días festivos en México o Estados Unidos. Ver la figura E_78.

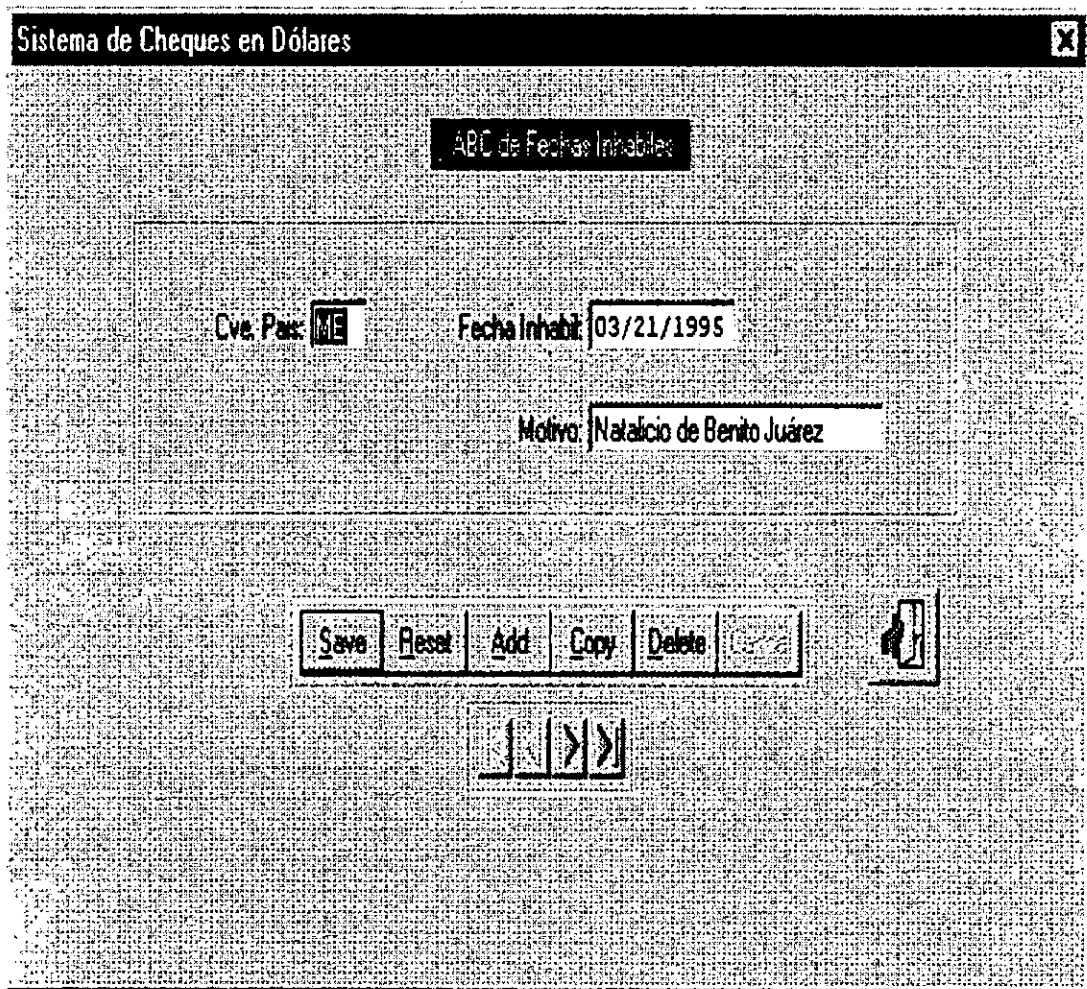


Figura E_78 ABC de Días Inhábiles

7.11 ABC de Preformatos:

Mantenimiento de los preformatos previamente definidos para ser usados por el cliente. Ver la figura E_79.

Sistema de Cheques en Dólares

Alas, Bases y Cambios de Preformatos

Seleccione una Pestaña

Parte 1	Parte 2	Parte 3
---------	---------	---------

No. de Preformato:

Nombre: ABA DIVISAS / FIRST CITY HOUSTON

Acc Party Id:

Cr Party Id:

Acc Party Name: ABA DIVISAS CASA DE CAMBIO

Cr Party Name: FIRST CITY TEXAS HOUSTON

Acc Party Add (1): HOUSTON, TX

Cr Party Add (1): PO BOX 2557

Acc Party Add (2):

Cr Party Add (2): HOUSTON TX 77252-2557

Acc Party Add (3):

Cr Party Add (3):

Acc Party Type:

Cr Party Type:

Save Reset Add Copy Delete

Navigation icons: Home, Back, Forward, Stop

Figura E_79 ABC de Preformatos

7.12 Solicitud de Cheques a Proveedor:

Solicitud de Cheques al proveedor para la entrega al cliente y para tener en "stock" (existencia). Ver la figura E_80.

Sistema de Chequera en Dólares

Solicitud de Chequeras al Proveedor

Producto:

Cuenta Inicial: Nombre:

Cuenta Final: Nombre:

Chequeras por Cuenta: Cheques por Chequera:

Cheq. (Normal/Urgente): Fecha de Generación:

Cheques con Nombre (S/N):

Entre datos o pulse ESC para finalizar

Figura E_80 Solicitud de Cheques a Proveedor

7.13 Test-Key:

Cobro de Cheques en los centros regionales afectando directamente a la cuenta del cliente. Ver la figura E_81.

Sistema de Chequera en Dólares

Pagos de Cheques en Centros Financieros

Centro Financiero: 0082 Nomb. Cto. Finan.: E. S. Cuernavaca (México)

Asesor: 31567 Nomb. Asesor: Lima Martínez González

Producto: 0 Cuenta: 0 Nombre: Venta concentrada a Chemical

Monto: 0.00 No. Cheque: 0 Fecha Morto: / /

Destino: EFEC Test: 0

Teclee la clave del Cto. Regional

Figura E_81 Test-Key

7.14 Eliminación de Test-Key:

Eliminación del movimiento de cobro de cheque en los Centros Regionales. Ver la figura E_82.

Sistema de Chequera en Dólares



Cancelación de pago de Cheque en Ctro. Financieros

Centro Financiero: Nomb. Ctro. Finan.:

Asesor: Nomb. Asesor:

Producto: Cuenta: Nombre:

Monto: No. Cheque: Fecha Movto.:

Tecee la clave del Ctro. Regional

Figura E_82 Eliminación de Test-Key

