

58



**UNIVERSIDAD NACIONAL
AUTONOMA DE MEXICO**

FACULTAD DE INGENIERIA

290982

**"CENTRO DE INFORMACION DE
CAJEROS AUTOMATICOS"**

**T E S I S
QUE PARA OBTENER EL TITULO DE
INGENIERO EN COMPUTACION
PRESENTA:**

VARGAS ROSAS, LUIS UBALDO



DIRECTOR DE TESIS: ING. ALBERTO TEMPLOS CARBAJAL

CIUDAD UNIVERSITARIA

ABRIL 2001



Universidad Nacional
Autónoma de México



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.

DEDICATORIA

A mi esposa Laura, por la comprensión y el apoyo que me dio al realizar este trabajo de tesis y por el amor que siempre me ha dado.

A mis hijos Luis Enrique, Eric Daniel y Carlos Eduardo, que este trabajo sea un elemento de inspiración para su vida.

A mis padres Raquel y Ubaldo, por que me educaron para la vida.

A mi hermano Héctor, por que siempre estemos juntos en las buenas y en las malas.

A mi tía Amelia, por ser la persona más cercana a mi madre.

A mi tío José Luis, por haberme guiado en la elección de la profesión que desempeño.

A mi tío Jorge, por ser una de las personas más importantes en mi formación personal.

A mis suegros Silvia y Francisco, por su apoyo incondicional.

A toda mi familia, gracias.

A mis amigos Raúl, Fernando y Moisés, por la buena época en la Universidad.

A mis amigos de BanCrecer.

TABLA DE CONTENIDO

PROLOGO	7
INTRODUCCION	9
OBJETIVO	11
CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS	13
1.1 ANTECEDENTES	13
1.2 CONCEPTOS BASICOS	14
1.2.1 PROCESAMIENTO TRANSACCIONAL Y DATA WAREHOUSE	14
1.2.2 MODELADO RELACIONAL	15
1.2.3 MODELADO DIMENSIONAL	17
1.2.4 MODELADO DIMENSIONAL Y MODELADO RELACIONAL	19
1.2.5 TABLAS DE HECHOS Y TABLAS DE DIMENSION	19
1.2.5.1 TABLAS DE HECHOS	20
1.2.5.2 DIMENSIONES Y ATRIBUTOS	20
1.3 CARACTERISTICAS DE LOS PRODUCTOS INFORMIX	22
1.3.1 INFORMIX SOFTWARE CORPORATION	22
1.3.2 INFORMIX METACUBE	22
1.3.3 IMPLEMENTACION DE DOS CAPAS	23
1.3.4 IMPLEMENTACION DE TRES CAPAS	24
1.3.5 COMPONENTES DE INFORMIX METACUBE	24
1.3.5.1 METACUBE ANALYSIS ENGINE	24
1.3.5.2 METACUBE EXPLORER	25
1.3.5.3 WEB EXPLORER	25
1.3.5.4 METACUBE FOR EXCEL	25
1.3.5.5 WAREHOUSE MANAGER	25
1.3.5.6 WAREHOUSE OPTIMIZER	25
1.3.5.7 SECURE WAREHOUSE	26
1.3.5.8 WEB PUBLISHER	26
1.3.5.9 SQL OPTIMIZER	26
1.3.5.10 SERVER INSTALLATION COMMAND FILES	26
1.3.5.11 METACUBE AGENTS AND AGENT ADMINISTRATOR	26
1.4 EL CICLO DE VIDA DIMENSIONAL	27
1.4.1 PLANEACION DEL PROYECTO	27
1.4.2 DEFINICION DE LOS REQUERIMIENTOS DEL NEGOCIO	27
1.4.3 MODELADO DIMENSIONAL	28
1.4.4 DISEÑO FISICO	28
1.4.5 DISEÑO DE LA ETAPA DE DATOS	28
1.4.6 DISEÑO DE LA ARQUITECTURA TECNICA	28
1.4.7 SELECCION E INSTALACION DEL PRODUCTO	29
1.4.8 ESPECIFICACION DE LA APLICACION DE USUARIO FINAL	29
1.4.9 DESARROLLO DE LA APLICACION DE USUARIO FINAL	29
1.4.10 IMPLEMENTACION-	29
1.4.11 MANTENIMIENTO Y CRECIMIENTO	29
1.4.12 ADMINISTRACION DEL PROYECTO	30
1.5 PROCESO DE DISEÑO DIMENSIONAL	30
1.5.1 ELEGIR UN PROCESO DE NEGOCIO A MODELAR	30
1.5.2 ELEGIR LA GRANULARIDAD DEL PROCESO DE NEGOCIO	31
1.5.3 ELEGIR DIMENSIONES POR CADA REGISTRO EN LA TABLA DE HECHOS	31
1.5.4 ELEGIR LAS METRICAS PARA LA TABLA DE HECHOS	31

TABLA DE CONTENIDO

1.6 HERRAMIENTAS DE MODELADO DE SISTEMAS	32
1.6.1 DIAGRAMAS DE FLUJO DE DATOS	32
1.6.1.1 COMPONENTES DE UN DFD	36
1.6.1.2 NIVELES DE UN DFD	38
1.6.1.3 GUIA PARA LA CONSTRUCCION DE UN DFD	39
1.6.2 DICCIONARIO DE DATOS	39
1.6.2.1 NOTACION DEL DICCIONARIO DE DATOS	39
1.6.3 ESPECIFICACIONES DE PROCESO	43
1.7 SEGURIDAD	48
1.7.1 SEGURIDAD: PUNTOS VULNERABLES	53
1.7.2 SOLUCIONES A LOS PROBLEMAS DE SEGURIDAD	58
1.7.3 SEGURIDAD EN UN AMBIENTE DATA WAREHOUSE	65
1.7.4 RECOMENDACIONES PARA LA SEGURIDAD	65
1.7.5 RESUMEN	67
CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION	69
2.1 ANALISIS DE LA SITUACION EN SU ESTADO ORIGINAL	70
2.2 ESTRATEGIA DE SOLUCION	71
2.2.1 ENTREVISTAS	71
2.3 REQUERIMIENTOS DEL USUARIO	73
2.3.1 CONCEPTOS DE NEGOCIO	74
2.3.2 REPORTES	75
2.3.3 GRAFICAS	76
2.3.4 PARAMETROS	78
2.4 IDENTIFICACION DE LAS FUENTES DE INFORMACION	78
2.5 MODELO DE DATOS LOGICO	80
2.5.1 GRANULARIDAD	80
2.5.2 DIMENSIONES	85
2.5.2.1 DIMENSIONES CAMBIANTES LENTAMENTE	85
2.5.2.2 DIMENSION TIEMPO	88
2.5.2.3 DIMENSION TIPO DE MOVIMIENTO	91
2.5.2.4 DIMENSION TIPO DE CUENTA	92
2.5.2.5 DIMENSION RAZON DE RESPUESTA	93
2.5.2.6 DIMENSION GEOGRAFIA	94
2.5.3 METRICAS	97
2.6 MODELO DE DATOS FISICO	100
2.6.1 DIMENSION	100
2.6.1.1 ELEMENTO DE DIMENSION	101
2.6.1.2 ATRIBUTO DE DIMENSION	103
2.6.2 TABLA DE HECHOS	104
2.6.2.1 METRICAS	105
CAPITULO 3. DISEÑO DE LA SOLUCION	107
3.1 DISEÑO FISICO	107
3.2 DISEÑO DE LA ETAPA DE DATOS	110
3.2.1 EXTRACCION	111
3.2.2 TRANSFORMACION	118
3.2.3 CARGA	127
3.2.3 PRESENTACION	148
3.2.3.1 DIMENSIONES	148

TABLA DE CONTENIDO

3.2.3.2 TABLA DE HECHOS _____	171
CAPITULO 4. IMPLEMENTACION DE LA SOLUCION _____	175
4.1 GRAFICAS _____	177
4.2 REPORTES _____	229
CAPITULO 5. PRUEBAS A LA SOLUCION _____	237
5.1 TECNICAS DE VERIFICACION Y VALIDACION _____	237
5.1.1 TECNICAS ESTATICAS Y DINAMICAS _____	237
5.1.2 TECNICAS FORMALES E INFORMALES _____	238
5.1.3 TECNICAS AUTOMATICAS Y MANUALES _____	238
5.1.4 PRUEBAS FUNCIONALES Y PRUEBAS ESTRUCTURALES _____	238
5.2 PROCESO DE PRUEBAS _____	239
5.2.1 PRUEBAS UNITARIAS _____	240
5.2.2 PRUEBAS DE INTEGRACION _____	241
5.2.3 PRUEBAS DE VALIDACION _____	244
5.2.4 PRUEBAS DE SISTEMA _____	244
5.3 METODOLOGIA APLICADA EN BANCRECER _____	246
5.5 RESUMEN _____	255
5.5.1 INTEGRACION DE LOS MODULOS Y PRUEBAS FINALES _____	255
5.5.2 CERTIFICACION DE LOS DATOS _____	255
CAPITULO 6. CONCLUSIONES _____	257
APENDICE A. DESCRIPCION DE LOS CAMPOS DEL TLF _____	259
APENDICE B. CODIGOS DE RESPUESTA _____	271
APENDICE C. DESCRIPCION DE LOS REPORTES ORIGINALES _____	277
CRITERIOS GENERALES _____	281
APENDICE D. MANUAL DE INSTALACION _____	287
1. INICIANDO EL PROGRAMA DE INSTALACION _____	287
1.2 PRIMEROS PASOS DE LA INSTALACION _____	288
1.3 ELIGIENDO EL SOFTWARE A INSTALAR _____	289
1.3.1 INSTALACION PARA USUARIO FINAL Y PARA ADMINISTRADOR _____	291
1.3.2 INSTALACION PARA UN DESARROLLADOR METACUBE _____	291
1.3.3 INSTALACION PERSONALIZADA _____	292
1.4 SELECCION DE LA CARPETA PARA LA APLICACION _____	294
1.5 INSTALACION DE ARCHIVOS DE LA APLICACION _____	295
2. INSTALACION DE INFORMIX-CLI _____	297
2.1 REQUERIMIENTOS DEL SISTEMA _____	299
2.1.1 SISTEMA OPERATIVO Windows 95 O Windows NT _____	299
2.1.2 DIRECCION TCP/IP _____	299
2.2 REGISTRO DEL USUARIO Y DE LA COMPAÑIA _____	300
2.3 REGISTRO DE LAS CLAVES DE ACCESO _____	301
2.4 SELECCION DE LA CARPETA Y DIRECTORIO PARA LA APLICACION _____	302
2.5 INSTALACION DE ARCHIVOS DE LA APLICACION _____	304
2.6 PASOS FINALES DE INSTALACION _____	306
3. INSTALACION DE UNA FUENTE DE DATOS CON ODBC _____	306
4. DEFINIENDO CARACTERISTICAS DE CONEXION _____	307
4.1 INICIAR METACUBE EXPLORER _____	307

TABLA DE CONTENIDO

4.2 ACCESANDO LA CAJA DE DIALOGO "Preferences"	309
4.3 ESTABLECIENDO UNA CONFIGURACION PARA CONEXION	310
APENDICE E. MANUAL DE USUARIO	313
1. ACCESO A METACUBE EXPLORER	314
2. COMO ABRIR UNA GRAFICA Y/O REPORTE DEFINIDO	317
3. CREACION DE UN NUEVO LIBRO DE TRABAJO	319
3.1. ELECCION DE UN MODELO DE DATOS	319
4. CREACION DE UNA CONSULTA	320
4.1 COMO LLENAR LOS ESPACIOS EN EL AREA DE <i>REPORT DEFINITION</i>	322
4.2 UTILIZACION DE FILTROS	323
4.2.1 FILTROS FIJOS	324
4.2.2 FILTROS PARAMETRIZADOS	325
5. EJECUCION DE UNA CONSULTA	326
5.1 EJECUCION DE CONSULTAS CON FILTROS PARAMETRIZADOS	326
5.2 EJECUCION EN LINEA DE UNA CONSULTA	328
5.3 EJECUCION DE UNA CONSULTA EN SEGUNDO PLANO	329
APENDICE F. MANUAL DE ERRORES Y SOLUCIONES	333
APENDICE G. MANUAL TECNICO	337
1. PROPOSITO DEL CLIENTE	337
2. REQUERIMIENTOS DE HARDWARE Y SOFTWARE	337
2.1 HARDWARE	337
2.2 SOFTWARE	337
3. ESQUEMA FUNCIONAL	337
3.1 QUERY MODE	337
3.2 RESULTS MODE	337
3.3 DESCRIPCION DE FUNCIONES	339
4. DESCRIPCION DE BASE DE DATOS (TABLAS E INDICES)	340
4.1 DESCRIPCION DE TABLAS DE LA BASE BDCICA	340
4.2 DESCRIPCION DE TABLAS DE MC_CICA	344
4.3 DESCRIPCION DE INDICES DE LA BASE DE DATOS "bdcica"	347
4.4 DESCRIPCION DE INDICES DE LA BASE DE DATOS "mc_cica"	349
5. SHELLS Y PROCEDIMIENTOS ALMACENADOS	351
5.1 SHELLS	351
5.2 PROCEDIMIENTOS ALMACENADOS EN LA BASE DE DATOS "bdcica"	353
5.3 PROCEDIMIENTOS ALMACENADOS EN LA BASE DE DATOS "mc_cica"	353
6. CRONS	354
7. PLAN DE CONTINGENCIAS EN CASO DE FALLAS	354
7.1 FALLA DE LA LINEA DE COMUNICACION	354
7.2 FALLAS DEL SERVIDOR	354
7.3 VIRUS	354
8. LISTA DE USUARIOS DEL SISTEMA Y DATOS GENERALES	354
9. PERMISOS UNIX Y/O INFORMIX POR USUARIO	355
10. CALENDARIO DE RESPALDOS	355
11. LISTADO DE PROGRAMAS	355

PROLOGO

El presente documento es el resultado del trabajo que he realizado en BanCrecer desde el año de 1995, dentro del área de Sistemas, especialmente en el área de Desarrollo. Aquí se describe la realización de uno de los proyectos más interesantes en los que he participado: el Centro de Información de Cajeros Automáticos.

Este documento consiste de dos grandes partes: Los capítulos del 1 al 6 y los apéndices del A al G junto con la bibliografía.

En el capítulo 1 se presentan los antecedentes del problema y los conceptos básicos para encararlo.

En los capítulos del 2 al 4 se explica como llegar a la solución del problema.

En el capítulo 5 se presentan las pruebas realizadas a la solución implementada.

En el capítulo 6 se presentan las conclusiones de este trabajo.

Los apéndices A y B presentan definiciones de datos contenidos en el archivo Transaction Log File utilizado como fuente para alimentar al modelo de datos.

El apéndice C describe los reportes existentes antes del inicio de este trabajo.

Los apéndices D al G contienen manuales que sirven como apoyo para la mejor utilización del Centro de Información de Cajeros Automáticos.

En la bibliografía se encuentran los nombres de las fuentes de información utilizadas para elaborar este trabajo.

Vertical line on the left side of the page.

Vertical line on the right side of the page.

OCBZ

INTRODUCCION

BanCrecer, como toda institución importante en México, ha realizado una labor trascendente dentro del sistema bancario desde hace varios años. A continuación se presenta una breve historia de este banco.

En 1945, en la ciudad de Monterrey, Nuevo León, se funda el Banco de Crédito y Servicio, S.A.

En 1982 se nacionaliza la banca.

El 19 de agosto de 1991, el Banco de Crédito y Servicio, S.A. conocido entonces como Bancreser, fue privatizado.

En ese año de 1991, Bancreser contaba con 5 cajeros automáticos, 70 sucursales, 2 mil 491 empleados (en promedio 35.6 empleados por sucursal), 93 mil clientes y 209 socios.

Un año más tarde, quedó formalmente constituido el Grupo Financiero BanCrecer para ofrecer a la clientela un conjunto integrado de servicios financieros. En 1993, el Grupo Financiero adquiere la mayoría de las acciones de BanOro, S.A. y al finalizar 1996, este banco queda totalmente fusionado a BanCrecer y también se incorporan las sucursales, recursos, empleados y clientes de la Caja Mexicana de Fomento S. A. P.

El nombre **BanCrecer** se adopta formalmente en 1993, año en que arrancan dos proyectos de gran importancia para el desarrollo de la institución: Uno fue la apertura y puesta en marcha de las sucursales básicas, a través de las cuales se llevarían los servicios bancarios a un número creciente de clientes y el otro fue la modernización tecnológica, a través de la implementación del sistema Altamira.

Así, en los años siguientes, se logra un crecimiento muy dinámico del número de sucursales, apoyado por una parte en tecnología de punta y, por otra, en la serie de productos MultiExpress, diseñados especialmente para la banca de menudeo.

En 1995 BanCrecer establece alianzas estratégicas con socios muy destacados. Tal es el caso de Allianz A.G. en materia de distribución de seguros, de Dresdner Bank y Allianz México para establecer y operar la Afore BanCrecer-Dresdner, y de IBM, empresa con la que se firma el contrato de Outsourcing de Red más grande de América Latina.

En 1999 el banco fue intervenido por el Instituto para la Protección al Ahorro Bancario (IPAB).

Actualmente BanCrecer cuenta con casi de 1,000 cajeros automáticos, 794 sucursales, 6 mil 516 empleados (en promedio 8.2 empleados por sucursal) y un millón 698,281 cuentas.

BanCrecer cuenta con una área de Banca Electrónica que define las políticas de negocio para los cajeros, siendo el área Corporativa de Sistemas la encargada de administrar y monitorear la red de comunicaciones, la operación de los cajeros automáticos y proporcionar mantenimiento a los programas que controlan estos dispositivos.

BanCrecer cuenta con un Centro de Cómputo desde donde se controlan:

- 27 millones de transacciones dentro de la plataforma principal.
- 8 millones de operaciones mensuales realizadas en cajeros automáticos.

INTRODUCCION

- 4 millones de llamadas que ingresan a través del sistema de atención telefónica a clientes "BanCreTel".
- La emisión mensual de 100 mil tarjetas de débito y 420 mil estados de cuenta.

Todos los servicios están monitoreados las 24 horas del día durante los 7 días de la semana.

El Centro de Cómputo BanCrecer tiene la capacidad de realizar:

- La emisión de 260 mil tarjetas de débito mensualmente.
- Impresión de más de 1 millón 800 mil estados de cuenta al mes.
- Hacer 60 millones de transacciones dentro de su plataforma principal, además de operar mensualmente 18 millones de pesos en Cajeros Automáticos.

Los clientes de BanCrecer y de los otros bancos pueden utilizar la red de cajeros automáticos de todo el país y cualquier operación realizada en los cajeros del banco se registra en el Centro de Cómputo BanCrecer.



BanCrecer



OBJETIVO

El objetivo de este trabajo de tesis es presentar los pasos necesarios para definir, analizar, diseñar e implementar un Data Warehouse o Centro de Información de cajeros automáticos, utilizando las técnicas del diseño multidimensional.

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

En la primera parte de este capítulo se presentan los antecedentes que originan la necesidad de contar con un sistema que permita consultar datos de cajeros para medir su comportamiento y tomar las decisiones necesarias en cuanto a su administración.

En la segunda parte se presentan los conceptos necesarios para realizar el análisis del problema que resolverá el CICA.

1.1 ANTECEDENTES

En esta parte se presenta el entorno en el que se encuentra el área que requiere de un sistema para administrar los cajeros automáticos: Banca Electrónica, que se encarga, apoyada por la Dirección de Sistemas, del correcto funcionamiento de los cajeros automáticos de BanCrecer.

Debido a que el área de Banca Electrónica es considerada dentro de los bancos como estratégica y de alto potencial es necesario tener fuentes de información para medir la productividad de los equipos que administra. La operación de los cajeros automáticos de BanCrecer está soportada por una computadora Tandem con sistema operativo Guardian 90, el registro de las transacciones realizadas en los cajeros es mediante el software llamado Base24 (desarrollado por la empresa estadounidense ACI) y se almacena en dos archivos conocidos como TLF "Transaction Log File": Uno para BanCrecer en México y otro para BanCrecer en Costa Rica.

Para conocer el comportamiento del uso de los cajeros y decidir su reubicación o el refuerzo de los parámetros de servicio (dotación de más efectivo, monitoreo de las caídas de línea, etc.) las áreas de Banca Electrónica y Sistemas utilizan reportes calendarizados que se generan a partir de los TLFs. La explotación de los datos almacenados en los TLF se realiza al bajar mensualmente de Tandem a cinta los archivos correspondientes y procesarlos en el mainframe IBM del banco, de donde se obtienen varios reportes cuyos programas se encuentran en lenguaje Cobol.

En 1995 se implementó el Centro de Información de Cajeros Automáticos (CICA) utilizando NewEra de Informix, que es una herramienta de programación visual para Windows. Debido a que no cumplía con la flexibilidad requerida en las consultas y por ser difícil de encontrar personal que la conociera, se decidió por parte de la Dirección de Sistemas cambiar esta versión del CICA por otra más acorde con los requerimientos cambiantes del usuario (debidos por su parte a los propios cambios del mercado), por lo que la opción fue MetaCube de Informix, cuya flexibilidad depende más del diseño de la base de datos y no tanto de la herramienta en sí.

El conocimiento de MetaCube y del diseño multidimensional se llevó a cabo mediante cursos y conferencias dentro y fuera de Informix, en los que participaron desde analistas hasta uno de los subdirectores de desarrollo. Este conocimiento teórico se ha concretado en la implementación del CICA utilizando la metodología del diseño multidimensional sobre la base de datos relacional de Informix y teniendo como interfaz de usuario o herramienta de acceso a MetaCube.

Para el sustento del análisis multidimensional expuesto como base medular de este trabajo se tomaron como referencia principal los libros de Ralph Kimball que se mencionan en la sección de Bibliografía, ya que desde el punto de vista teórico-práctico es un autor más claro y conciso que el propio Bill Inmon quien es considerado el padre del DataWarehouse.

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

1.2 CONCEPTOS BASICOS

A continuación se presentan los conceptos básicos necesarios para iniciar el análisis del problema planteado y que el CICA deberá resolver.

1.2.1 PROCESAMIENTO TRANSACCIONAL Y DATA WAREHOUSE

Antes de desarrollar las perspectivas y técnicas necesarias para diseñar un data warehouse, debemos encarar una importante controversia. El procesamiento transaccional en línea (OLTP por On-line transactional processing) es profundamente diferente del data warehousing (DW) ya que son diferentes tanto los usuarios, el contenido de los datos, las estructuras de datos, el hardware, el software, la administración, el manejo de los sistemas y los ritmos diarios de trabajo.

Tanto los sistemas OLTP como los de DW se interesan especialmente por la consistencia de los datos. Sin embargo la consistencia OLTP es microscópica. El objetivo del procesamiento transaccional es procesar un gran número de transacciones pequeñas, atómicas sin perder ninguna de ellas.

En DW la consistencia es medida globalmente. En general no nos preocupamos por una transacción individual, sino más bien que la carga actual de los datos nuevos sea un conjunto completo y consistente. En lugar de una perspectiva microscópica, tenemos una perspectiva de aseguramiento de calidad. En lugar de un cálculo técnico de la consistencia de los datos, tenemos un juicio de administración de la consistencia de los datos.

Un sistema OLTP serio procesa miles o hasta millones de transacciones por día. Cada transacción contiene una pequeña pieza de datos. Un DW serio frecuentemente procesará solo una transacción por día. Pero esta transacción contiene miles o hasta millones de registros. En lugar de llamarle a esto una transacción, le llamamos la carga de datos de producción. Lo que nos interesa es el estado consistente del sistema con el que iniciamos antes de la carga de los datos de producción y el estado consistente del sistema con que finalizamos después de una carga exitosa de esos datos. Si nos vemos forzados a detener una carga antes de que se complete, no tenemos la posibilidad de regresarla (dar "roll-back") registro por registro; en lugar de esto sobreescribiremos el sistema entero con un respaldo tomado antes del inicio de la carga.

Por un lado, los usuarios de un sistema OLTP "giran las ruedas" de la organización. Toman las ordenes, abren y cierran las cajas registradoras, registran nuevos clientes, reciben quejas, realizan reservaciones, introducen nuevos datos y corrigen datos viejos. Los usuarios de un sistema OLTP casi siempre tratan con una cuenta a la vez. Las consultas que hacen a la base de datos casi siempre traen registros simples que los habilitan para continuar con su función primaria de entrada de datos. Los usuarios de sistemas OLTP realizan las mismas tareas muchas veces. El desempeño es el rey absoluto de un sistema OLTP. Ninguna actividad "opcional" que haga lento el sistema OLTP es permitida. Las actividades de tipo DW tales como lanzar una consulta para sumarizar 100,000 registros simplemente no están permitidas en un sistema OLTP. La mayor parte de los reportes hechos en los sistemas OLTP consisten de la preparación de grandes listados de tablas completas.

Por otro lado, los usuarios de DW están "observando las ruedas" de la organización. Contabilizan las nuevas ordenes, preguntan cuando se están abriendo y cerrando las cajas registradoras, preguntan por qué se registró un nuevo cliente, preguntan que les interesa a esos clientes, preguntan cuantas reservaciones más se hicieron respecto del año pasado,

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

observan que datos son nuevos y piden que los datos erróneos sean corregidos. Los usuarios de una DW casi nunca tratan con una cuenta a la vez. En su lugar piden una página o dos de resúmenes de cuentas, usualmente requieren que cientos o miles de registros sean buscados y comprimidos dentro de una respuesta. Los usuarios de un DW cambian continuamente los tipos de preguntas que le hacen a la base de datos. Aunque la forma de hacer sus peticiones pueda ser similar de consulta a consulta, su impacto sobre la base de datos variará ampliamente: desde traer un ciento de registros de respuesta hasta traer un millón de registros de respuesta.

1.2.2 MODELADO RELACIONAL

El procesamiento transaccional se basa en el modelado relacional para funcionar. El modelado entidad-relación es una técnica de diseño lógico que busca eliminar la redundancia de los datos. Imaginemos un negocio que toma ordenes y vende productos a sus clientes. En los primeros días de la computación, mucho antes de las bases de datos relacionales, al transferir por primera vez estos datos a una computadora, probablemente capturaríamos el orden original como un gran registro con muchos campos. Tal registro podría haber sido de 1,000 bytes distribuidos a lo largo de 50 campos. Los productos en el orden probablemente estarían representados como un grupo repetitivo de campos dentro del registro maestro. Tener estos datos en la computadora habría sido muy útil, pero rápidamente aprendimos algunas lecciones básicas acerca de almacenar y manipular datos. Una de las lecciones fue que los datos en esta forma eran difíciles de mantener consistentes, debido a que cada registro dependía de sí mismo. El nombre del cliente y la dirección aparecían varias veces, ya que estos datos se repetían cada vez que se tomaba una nueva orden. Las inconsistencias en los datos eran incontrolables, ya que todas las instancias de la dirección del cliente eran independientes y actualizar la dirección del cliente era una transacción complicada.

Aún en esos primeros días, aprendimos a separar los datos redundantes dentro de tablas distintas, tales como un maestro de clientes y un maestro de productos, pero pagamos el precio. Los sistemas de software utilizados para recuperar y manipular los datos se volvieron complejos e ineficientes por que requerían de una cuidadosa atención para procesar los algoritmos que ligaban estos grupos de tablas entre sí. Necesitábamos un sistema de base de datos que fuera muy bueno para ligar tablas. Esto propició el camino para la revolución de las bases de datos relacionales donde la base de datos estaba dedicada justo a esta tarea.

Un sistema de bases de datos debe ser capaz de representar y manipular entidades (registros y campos) y sus relaciones de manera fácil y conveniente. En el enfoque relacional concebido por E.F. Codd, visualiza la base de datos lógica como una colección de tablas bidimensionales llamadas "Relaciones". Estas tablas son planas en el sentido de que no hay grupos repetidos. El enfoque relacional introduce terminología propia y tiene una tendencia a usar términos poco convencionales relacionados con las matemáticas. El enfoque relacional se fundamenta en la teoría matemática de las relaciones, por lo cual posee un buen fundamento teórico.

El modelo de datos relacional representa, como ya se mencionó, a la base de datos como una colección de tablas. Aunque las tablas son simples y de comprensión intuitiva, estas se encuentran en correspondencia directa entre el concepto de tabla y el concepto matemático de relación. Una base de datos relacional consiste de una colección de tablas, a cada una de las cuales se le asigna un nombre único.

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

Ahora ya estamos listos para definir el enfoque del modelado dimensional que se presenta en la siguiente sección.

1.2.3 MODELADO DIMENSIONAL

El modelado dimensional es un nuevo nombre para una técnica no muy nueva de hacer a las bases de datos simples y entendibles. Cuando una base de datos puede ser visualizada como un cubo de tres, cuatro, cinco o más dimensiones, uno puede imaginarse dándole vueltas y "rebanándolo" a lo largo de sus dimensiones

Pongamos un sencillo ejemplo. Imaginemos una empresa donde el presidente o director general describe lo que la compañía hace como sigue:

"Nosotros vendemos productos en varios mercados y medimos nuestros resultados a lo largo del tiempo."

Como diseñadores de data warehouse, escuchamos cuidadosamente estas palabras y agregamos nuestro especial énfasis:

"Nosotros vendemos *Productos* en varios *Mercados* y medimos nuestros resultados a lo largo del *Tiempo*."

La mayoría de la gente encuentra fácil pensar en este negocio como un cubo de datos con etiquetas en cada uno de los lados del cubo como se muestra en la figura 1.2.

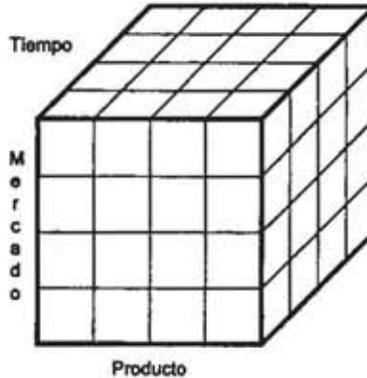


Figura 1.2 Cubo de datos

Cualquier punto dentro de este cubo es una intersección de las coordenadas definidas por los lados del cubo. Para el negocio descrito arriba etiquetamos los lados del cubo como Producto, Mercado y Tiempo. La mayoría de la gente puede imaginar que los puntos dentro del cubo son como las mediciones del negocio por la manera en que la combinación de Producto, Mercado y Tiempo son almacenados. Esto es el modelo dimensional.

Si esta perspectiva parece demasiado simple, está bien. Un modelo de datos que empieza siendo simple tiene la oportunidad de terminar simple al final del diseño, en cambio un

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

modelo de un negocio que comienza siendo complicado seguramente será complicado al final.

La atracción principal de modelar dimensionalmente un negocio es su simplicidad. En este trabajo veremos como la simplicidad es la clave que permite a los usuarios entender las bases de datos y permite al software navegar eficientemente las bases de datos.

En los siguientes párrafos mostraremos como este simple enfoque "cubista" se implementa en una base de datos relacional.

El modelado dimensional es una técnica de diseño lógico que busca presentar los datos en una estructura estándar que es intuitiva y permite el acceso de alto desempeño. Es inherentemente dimensional y se adhiere a la disciplina que utiliza el modelo relacional con algunas restricciones importantes. Cada modelo dimensional se compone de una gran tabla central conocida como **tabla de hechos** y un conjunto de tablas más pequeñas desplegadas en un patrón radial alrededor de la tabla central, conocidas como **tablas de dimensión**, como se muestra en la figura 1.3. Esta característica de tener una estructura en forma de estrella a menudo es llamada **unión estrella** o star join.

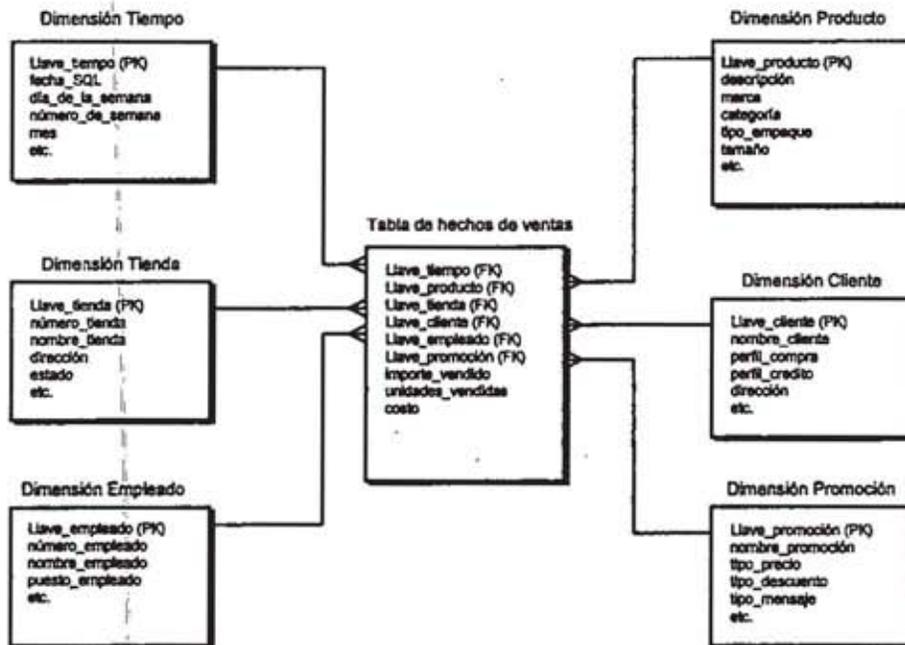


Figura 1.3 Ejemplo de Unión Estrella

A diferencia del modelo entidad-relación, el modelo dimensional es muy asimétrico. Tiene una tabla dominante en el centro de su esquema, la cual es la única que tiene múltiples uniones que la conectan a las otras tablas, estas tienen una sola unión hacia la tabla central; en otras palabras cada tabla de dimensión tiene una sola llave primaria que corresponde exactamente a uno de los componentes de la llave compuesta en la tabla de hechos. Debido a que la tabla de hechos tiene una llave primaria múltiple compuesta de dos

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

o más llaves foráneas siempre expresa una relación muchos-a-muchos, normalmente este tipo de relación no es permitido de acuerdo al proceso de modelado relacional.

En la figura 1.3 se modela un negocio que vende productos en un cierto número de mercados y mide su desempeño a lo largo del tiempo. Hemos decidido como diseñadores de la base de datos que la tabla de hechos contenga totales diarios por producto, de todos los productos vendidos. A esto se le llama **granularidad** de la tabla de hechos. En otras palabras, cada registro en la tabla de hechos representa las ventas totales de un producto específico en un mercado en un día. Cualquiera otra combinación de producto, mercado o día genera un registro diferente en la tabla de hechos.

En la siguiente sección se explica la relación entre el modelado dimensional y el modelado entidad-relación.

1.2.4 MODELADO DIMENSIONAL Y MODELADO RELACIONAL

La clave para entender la relación entre el modelado dimensional y el entidad-relación es que *un solo diagrama entidad-relación se descompone en múltiples diagramas de tabla de hechos*. Por ejemplo un diagrama entidad-relación (DER) extenso que represente cualquier posible proceso de negocio puede contener Pedidos, Embarques, Facturaciones y Devoluciones, todo en el mismo diagrama. El DER representa en un solo esquema procesos que nunca coexisten en un solo conjunto de datos en un determinado punto del tiempo.

Así que el primer paso para convertir un DER a un conjunto de diagramas de modelado dimensional es separar el DER en sus procesos de negocio discretos y modelar cada uno por separado. El segundo paso es seleccionar las relaciones muchos-a-muchos en el DER que contienen hechos numéricos, aditivos y que no sean llaves para designarlas como tablas de hechos. El tercer paso es desnormalizar todas las tablas, para quedar en tablas más sencillas con llaves simples que conecten directamente a las tablas de hechos. Estas tablas se convierten en las tablas de dimensión. En los casos en los que una tabla de dimensión se conecta a más de una tabla de hechos, se representa a esta misma tabla de dimensión en ambos esquemas y nos referimos a ellas como "conformadas" entre los dos modelos dimensionales.

El modelo maestro resultante del data warehouse para una gran compañía consiste de entre 10 y 25 esquemas unión estrella muy similares. Si el diseño se ha realizado correctamente, muchas de esas tablas de dimensión se compartirán entre las tablas de hechos.

1.2.5 TABLAS DE HECHOS Y TABLAS DE DIMENSION

La idea fundamental del modelado dimensional es que casi todos los tipos de datos de negocio pueden ser representados como una clase de cubo de datos, donde las celdas del cubo contienen valores medidos y los bordes del cubo definen las dimensiones naturales de los datos. Por supuesto que se permiten más de tres dimensiones en nuestros diseños, así que técnicamente al cubo se le debería llamar *hipercubo*, aunque los términos *cubo* y *cubo de datos* son utilizados por casi todo mundo.

Los modelos dimensionales del mundo real pueden contener entre 4 y 15 dimensiones. Los modelos de solamente 2 o 3 dimensiones son raros y comúnmente dejan pensar que se deberían agregar más dimensiones al diseño. Los modelos de 20 o más dimensiones se ven injustificados, usualmente tienen dimensiones superfluas que deberían ser combinadas.

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

Aunque la elección final de las dimensiones es una prerrogativa del diseñador, la identificación de las dimensiones está atada a la realidad del ambiente del negocio y no al gusto del diseñador.

A continuación se presentan definiciones a detalle de las tablas de hechos y las de dimensiones.

1.2.5.1 TABLAS DE HECHOS

La tabla de hechos es en donde se registran los acontecimientos del negocio. Un acontecimiento usualmente no se conoce de antemano, es una observación realizada en el lugar en donde está el mercado. Las tablas de hechos almacenan las mediciones numéricas del negocio. Cada una de esas mediciones o métricas se toman en la intersección de todas las dimensiones. Las mejores y más usuales métricas son numéricas, evaluadas continuamente y aditivas; la razón de esto es que virtualmente cualquier consulta realizada contra la tabla de hechos revisará cientos, miles o aún millones de registros que utilizará el Sistema Manejador de Bases de Datos (SMBD) para armar la respuesta. Este gran número de registros se comprimirá en unas cuantas docenas que recibirá el usuario. Sin lugar a dudas la única manera útil de comprimir estos registros para presentarlos al usuario es adicionarlos, por lo tanto si las métricas son números y son aditivos, el software fácilmente armará la respuesta para el usuario.

La sugerencia de que los hechos sean evaluados continuamente, principalmente es como una guía para el diseñador de la base de datos con el objeto de ayudarte a decidir que es un hecho y qué es un atributo dimensional. Por ejemplo, el importe vendido se evalúa constantemente por que puede tomar cualquier rango de datos a lo largo del tiempo, cada vez que se vende un producto. Si no hay actividad de productos se debe dejar el registro de esta operación fuera de la base de datos y no se debe tratar de meter con ceros representando que "nada ha pasado". Por esta razón las tablas de hechos son **esparcidas**. Es obvio en este ejemplo que el importe vendido, las unidades y el costo son cantidades aditivas, debido a que tiene sentido sumarizar el importe y otras métricas entre cada combinación de tiempo, producto, mercado y promoción.

Existen métricas **semiaditivas** y **no aditivas**, las primeras se pueden sumarizar considerando solo algunas dimensiones y las no aditivas simplemente no se pueden sumar en absoluto. Para estas últimas nos vemos forzados a utilizar conteos si deseamos sumar los registros.

A partir de las métricas existentes se pueden obtener otras métricas, como resultado de la combinación entre las existentes, al resultado se le conoce como **métricas calculadas**.

1.2.5.2 DIMENSIONES Y ATRIBUTOS

Las dimensiones se conforman de atributos. Los atributos son usualmente campos de texto y describen una característica de una cosa tangible. Los atributos más obvios son las descripciones de productos. Por ejemplo, el sabor de un producto es un atributo bien conocido del mismo producto. No medimos el atributo del sabor, lo conocemos de antemano.

Las tablas de dimensión son los elementos en donde se almacenan las descripciones textuales de las dimensiones del negocio, cada descripción ayuda a definir cada miembro de la dimensión respectiva.

En una base de datos bien diseñada cada tabla de dimensión tiene muchos **atributos** (campos); los mejores atributos son **textuales**, **discretos** y utilizados como parte de las **condiciones y encabezados de renglón**, en el conjunto de datos que recibe el usuario

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

como respuesta a su petición. Debido a que los atributos de dimensión juegan el rol de describir cada uno de los elementos de dimensión, son más útiles si son de tipo texto.

A veces, para el diseñar una base de datos, no es claro si un campo numérico extraído de la fuente de datos de producción es una métrica o un atributo. Normalmente podemos decidir preguntando si el campo numérico es una medición que varía continuamente cada vez que la registramos -esto lo hace una métrica- o si es una descripción evaluada discretamente de algo que es más o menos constante -esto lo hace un atributo de dimensión-.

Otra característica importante de las dimensiones son las **Jerarquías** que definen los atributos que las componen, estas jerarquías permiten llevar a cabo las operaciones de **drill down** y **drill up**. Los atributos son textuales, o se comportan como texto, toman valores discretos y son la fuente de condiciones en aplicaciones y de encabezados de columna en un reporte final. De hecho, uno se puede imaginar la creación de un encabezado en un reporte al "arrastrar" un atributo de dimensión hacia el reporte, haciéndolo un encabezado, como se muestra en la figura 1.4.

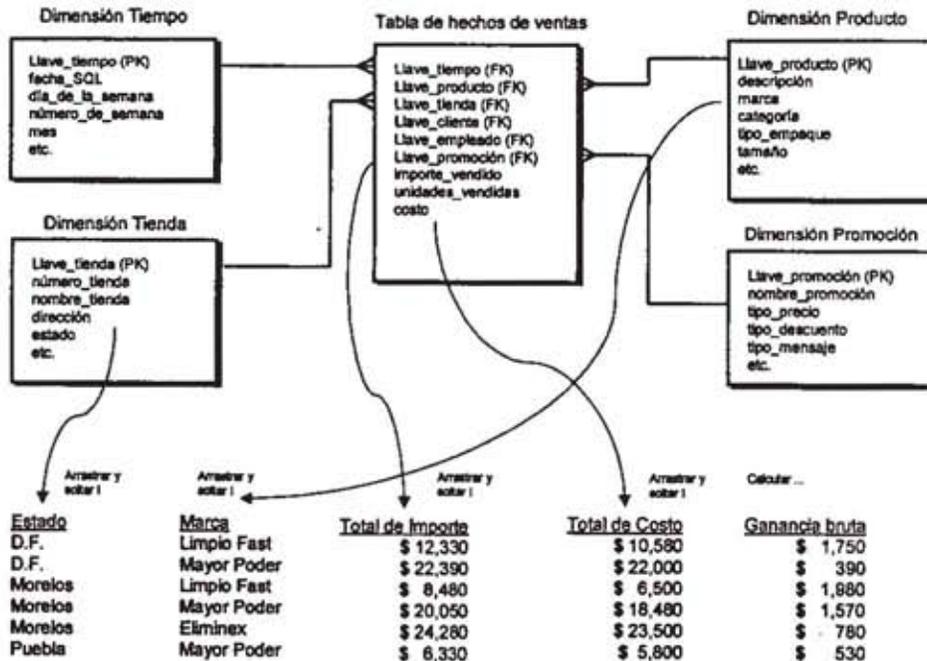


Figura 1.4 Creación de un reporte a partir de un modelo dimensional

El proceso de agregar encabezados puede estar compuesto de atributos de tantas dimensiones como el usuario lo deseé. La gran fortaleza del **Lenguaje Estructurado de Consultas (SQL-Structured Query Language)** es que esos encabezados simplemente se agregan a la lista del SELECT y a la cláusula del GROUP BY.

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

Del planteamiento del párrafo anterior vemos que la definición precisa de drilling down es agregar un columnas al encabezado del reporte, de manera similar drilling up entonces se obtiene substrayendo columnas del encabezado.

1.3 CARACTERISTICAS DE LOS PRODUCTOS INFORMIX

Por la normatividad existente dentro del área de sistemas en BanCreceer, el manejador de base de datos institucional, para aplicaciones departamentales (fuera de la computadora principal) es Informix, por lo que la implementación del Centro de Información de Cajeros Automáticos se ha llevado a cabo con la base de datos y las herramientas propias de esta compañía.

1.3.1 INFORMIX SOFTWARE CORPORATION

La compañía Informix Software Corporation fue fundada en 1980 con el propósito de ofrecer al mercado un manejador de base de datos relacional para ambiente UNIX.

En 1996 Informix compra "MetaCube" a Stanford Technology Group, para tener una herramienta de implementación de Data Warehouse.

1.3.2 INFORMIX METACUBE

Informix MetaCube fue la primera herramienta que Informix ofreció al mercado para cubrir las necesidades de sus clientes en la explotación de Data Warehouse. Utiliza la estrategia de ROLAP en donde las consultas al modelo dimensional se hacen al acceder a las tablas de una base de datos relacional, obteniendo los resultados en gráficas o reportes.

A continuación se presenta en la figura 1.5 un esquema general de la arquitectura de Informix MetaCube:

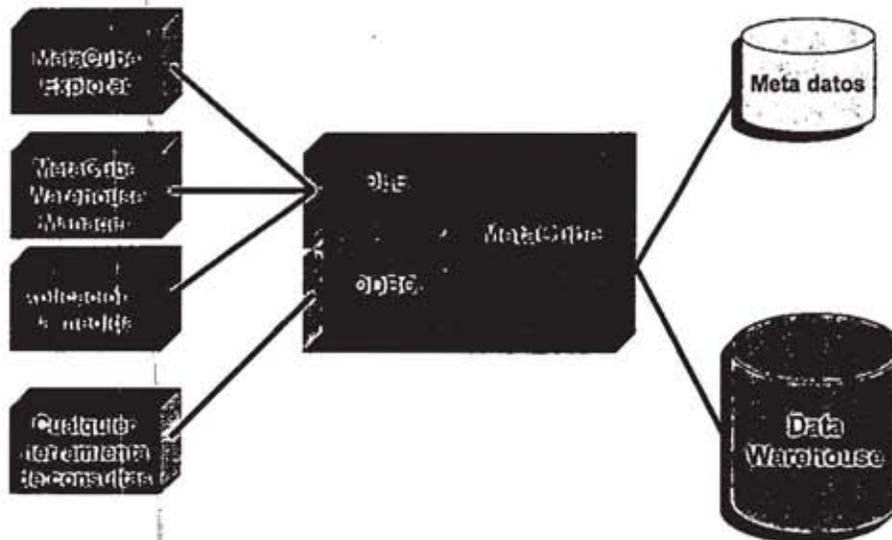


Figura 1.5 Esquema general de la arquitectura de Informix MetaCube

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

El motor de análisis de MetaCube puede implementarse utilizando una arquitectura de dos capas o una de tres capas. En una arquitectura de dos capas el motor de análisis de MetaCube y la aplicación de consultas residen en la misma máquina cliente. En una arquitectura de tres capas el motor de análisis de MetaCube reside en una máquina intermedia y las aplicaciones MetaCube residen en una máquina cliente separada.

Para decidir que tipo de arquitectura utilizar se deben seguir las siguientes consideraciones: Una arquitectura de dos capas, también llamada configuración **cliente/servidor**, se recomienda si existe alguna de las siguientes situaciones:

- No se implementará MetaCube Web Explorer.
- Se tienen máquinas clientes lo suficientemente poderosas para ejecutar el motor de análisis de MetaCube y todas las aplicaciones requeridas de MetaCube. Este hecho hace innecesario compartir recursos de máquina entre múltiples usuarios.

Una arquitectura de tres capas, también llamada configuración de capa intermedia, se recomienda si existe alguna de las siguientes situaciones:

- Se implementará MetaCube Web Explorer.
- Las máquinas clientes no son poderosas, ya que estas máquinas corriendo en una arquitectura de tres capas requieren menos recursos de cómputo y almacenamiento para el software por que accesan al motor de análisis de MetaCube localizado en la máquina de la capa intermedia.
- Se desea apalancar el poder de la máquina de la capa intermedia para mejorar el desempeño de las operaciones del motor de análisis de MetaCube, tales como pivoteo de datos y cálculos rápidos.
- Se desea centralizar la administración de los usuarios.

1.3.3 IMPLEMENTACION DE DOS CAPAS

Una arquitectura de dos capas consiste de la capa del servidor y la capa del cliente.

La capa del servidor es en donde reside la base de datos y corre en un ambiente de sistema operativo UNIX o Windows NT. La capa del cliente opcionalmente corre los Agentes de MetaCube apropiados para el sistema operativo de la máquina.

La capa del cliente es una PC con Windows a 32 bits que corre el motor de análisis de MetaCube y las aplicaciones del cliente. Las aplicaciones en la máquina cliente pueden incluir lo siguiente:

- MetaCube Explorer.
- MetaCube para Excel.
- Sistemas de presentación de información ejecutiva (EIS) hechas a la medida.
- Herramientas de consulta de terceros corriendo a través de una interfaz directa con MetaCube o a través del MetaCube SQL Optimizer. Al menos una máquina de la capa del cliente debe tener instalados los Server Installation Command Files y las herramientas de administración de MetaCube. Las herramientas de administración incluyen lo siguiente:
 - MetaCube Warehouse Manager
 - MetaCube Warehouse Optimizer
 - MetaCube Agent Administrator
 - MetaCube Secure Warehouse
 - MetaCube Web Publisher

1.3.4 IMPLEMENTACION DE TRES CAPAS

Una arquitectura de tres capas consiste de la capa del servidor, una capa intermedia y la capa del cliente.

La capa del servidor es donde reside la base de datos y corre en un ambiente de sistema operativo UNIX o Windows NT. La capa del servidor opcionalmente corre los Agentes de MetaCube apropiados para el sistema operativo de la máquina.

La capa intermedia es una máquina del calibre de un servidor con Windows NT Server o Workstation. El motor de análisis de MetaCube se instala en esta capa y corre como un paquete en el ambiente de Microsoft Transaction Server (MTS).

Los Server Installation Command Files y las herramientas de administración de MetaCube se deben instalar en la capa intermedia. Las herramientas de administración incluyen lo siguiente:

- MetaCube Warehouse Manager
- MetaCube Secure Warehouse
- MetaCube Agent Administrator
- MetaCube Warehouse Optimizer
- MetaCube Web Publisher
- MetaCube Agents for Windows NT (es opcional, si el servidor de base de datos reside en una plataforma UNIX, los agentes de MetaCube para UNIX pueden correr sobre esa plataforma).

Si se tiene más de una máquina en la capa intermedia, se debe instalar Secure Warehouse en cada una de ellas para manejar el motor de análisis de MetaCube que reside en cada una. Solo una máquina de la capa intermedia necesita tener instalados los Server Installation Command Files y el resto de las herramientas de administración.

La capa intermedia puede incluir opcionalmente un servidor Web, aunque las páginas Web para MetaCube Web Explorer pueden ser atendidas desde cualquier servidor existente en la red.

La capa cliente es una PC Windows a 32 bits que corre las aplicaciones tipo cliente. Estas aplicaciones pueden incluir lo siguiente:

- MetaCube Explorer
- MetaCube Web Explorer
- MetaCube for Excel
- Aplicaciones EIS hechas a la medida
- Herramientas de consulta de terceros corriendo a través de una interfaz directa con MetaCube o a través del MetaCube SQL Optimizer.

1.3.5 COMPONENTES DE INFORMIX METACUBE

En esta sección se describen los componentes más importantes de MetaCube, respetando los nombres originales en idioma inglés de cada uno.

1.3.5.1 METACUBE ANALYSIS ENGINE

El motor de análisis de MetaCube (MetaCube Analysis Engine) es un programa C++ a 32 bits que puede operar en ambientes operativos a 32 bits tales como Windows NT y Windows 9x. El motor de análisis de MetaCube proporciona una interfaz multidimensional, orientada a objetos hacia una base de datos relacional Informix, procesando peticiones de datos enviadas por aplicaciones tales como MetaCube Explorer y MetaCube for Excel,

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

generando SQL que envía a la base de datos por medio de un driver de ODBC (Open Database Connectivity).

Todas las aplicaciones MetaCube se conectan a la base de datos a través del motor de análisis de MetaCube. Este puede residir en la misma computadora que otras aplicaciones de MetaCube o puede residir en la capa intermedia soportando múltiples usuarios que corran aplicaciones MetaCube en otras computadoras.

1.3.5.2 METACUBE EXPLORER

El explorador de MetaCube (MetaCube Explorer) es una aplicación Windows que provee a los usuarios finales acceso a la medida a los datos almacenados en una base de datos relacional, enviando consultas a través de la interfaz de programación orientada a objetos del motor de análisis de MetaCube. Explorer ofrece una interfaz amigable y opciones para reporte y gráficas. Explorer corre en sistemas operativos a 32 bits como Windows 9x y Windows NT.

1.3.5.3 WEB EXPLORER

El explorador para Web de MetaCube proporciona la misma funcionalidad que Explorer excepto que la aplicación corre en un navegador de web.

1.3.5.4 METACUBE FOR EXCEL

MetaCube para Excel es un add-in para Microsoft Excel dirigido a usuarios que tienen familiaridad con las hojas de cálculo. No puede funcionar independientemente, sino que requiere de que esté instalado Microsoft Excel versión 7.0. Después de procesar las consultas MetaCube regresa los resultados directamente a la hoja de Excel.

1.3.5.5 WAREHOUSE MANAGER

El Warehouse Manager de MetaCube es una aplicación tipo Windows que proporciona a los administradores una interfaz gráfica para configurar que las aplicaciones MetaCube accedan a la base de datos relacional en términos multidimensionales. El motor de análisis de MetaCube almacena la descripción multidimensional en el Warehouse Manager como metadatos en un conjunto de tablas del sistema MetaCube en el servidor. Estas tablas, a menudo referenciadas como tablas de metadatos, se crean mediante scripts incluidos con los productos tanto del lado del cliente como del servidor. De igual forma como todos los productos administrativos, el Warehouse Manager funciona en sistemas operativos a 32 bits como Windows 9x y Windows NT.

1.3.5.6 WAREHOUSE OPTIMIZER

El Warehouse Optimizer es una aplicación tipo Windows que corre a 32 bits para los administradores del data warehouse. El Warehouse Optimizer analiza el modelo de datos y revisa las consultas que recibe para recomendar una estrategia de agregación en el sistema.

Los agregados almacenan datos sumarios, agrupando y sumando registros transaccionales detallados para mejorar el desempeño de las consultas que hacen uso intensivo de los recursos. Normalmente, las consultas que solicitan tal nivel de resumen de datos requieren que la base de datos realice la suma mientras procesa el resultado de la petición, pero el motor de análisis de MetaCube puede encaminar esas consultas a una tabla de agregado, obteniendo la respuesta directamente sin más procesamiento.

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

El volumen de datos crudos que puede acceder de forma directa el warehouse está determinado por la eficiencia de la estrategia de agregación. Las estrategias de agregación ineficientes pueden obstaculizar seriamente el desempeño y obstruyen a la base de datos con el traslape de las tablas de resumen que consumen una fracción substancial del espacio total disponible.

1.3.5.7 SECURE WAREHOUSE

MetaCube Secure Warehouse proporciona a los administradores de Data Warehouse una interfaz gráfica para controlar el acceso a los datos de los usuarios finales. Con Secure Warehouse, un administrador puede especificar qué usuarios tienen acceso a un Sistema de Apoyo para la toma de Decisiones (DSS – Decision Support System) que es el modelo implementado a través del Warehouse Manager con un nombre definido. El administrador de Data Warehouse también puede controlar qué información verá cada usuario en un sistema DSS y cuando puede acceder información mediante la ejecución de consultas.

Ningún usuario de ningún producto de MetaCube puede acceder un sistema DSS a menos que esté autorizado en el Secure Warehouse. Los usuarios de Web Explorer tampoco se pueden conectar a la base de datos a menos que hayan sido definidos como usuarios en el Secure Warehouse.

1.3.5.8 WEB PUBLISHER

Web Publisher puede utilizarse para crear archivos tipo HTML que contengan los resultados de una consulta. Estas páginas se pueden incorporar en un sitio web de intranet para visualizarse con un navegador. Esta herramienta administrativa corre por medio del Agent Administrator y permite la publicación en línea de reportes a personas en la organización que no utilicen aplicaciones MetaCube para consultas.

1.3.5.9 SQL OPTIMIZER

MetaCube SQL Optimizer es una librería de rutinas que permiten la optimización del SQL generado por una herramienta de consulta o una aplicación de consultas hecha a la medida.

1.3.5.10 SERVER INSTALLATION COMMAND FILES

Los server installation command files, son archivos que contienen rutinas de actualización y construcción. Las rutinas de construcción arman todas las tablas de metadatos necesarias para almacenar:

- Una descripción multidimensional de la base de datos, de la forma en que se encuentra en el Warehouse Manager.
- Las características de las carpetas, consultas y filtros, como se definieron y salvaron en aplicaciones como MetaCube Explorer y MetaCube para Excel.
- Las definiciones de las tareas enviadas para ejecución en segundo plano por los agentes de MetaCube.

1.3.5.11 METACUBE AGENTS AND AGENT ADMINISTRATOR

El administrador de agentes de MetaCube proporciona varios servicios, como programar la ejecución en segundo plano de consultas, construir tablas de datos sumariados

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

(agregados), construir tablas de ejemplos y construir tablas de elementos de dimensión, entre otros.

Los agentes de MetaCube se instalan en una plataforma NT o UNIX, usualmente en la misma plataforma que la base de datos Informix.

1.4 EL CICLO DE VIDA DIMENSIONAL

El ciclo de desarrollo de un sistema se puede ver como un torbellino, moviéndose desde la periferia durante el análisis de requerimientos, aumentando su intensidad técnica durante el diseño y alcanzando el "embudo" cuando se genera el código.

En esta sección se presenta la metodología del ciclo de vida dimensional. El diagrama general de este ciclo de vida se presenta en la figura 1.6, en él se describe la secuencia de las tareas de alto nivel requeridas para un efectivo diseño, desarrollo e implementación del data warehouse.

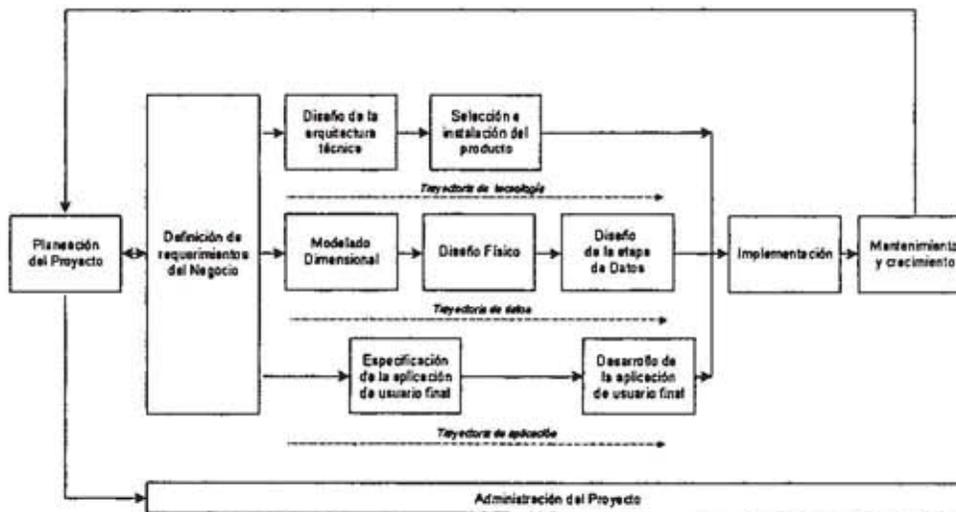


Figura 1.6 El ciclo de vida dimensional

A continuación se explica cada segmento del diagrama del ciclo de vida dimensional.

1.4.1 PLANEACION DEL PROYECTO

El ciclo de vida inicia con la planeación del proyecto. La planeación del proyecto dimensiona la definición y el alcance del proyecto de data warehouse, incluyendo la valoración de que tanta disposición existe para el proyecto y la justificación desde el punto de vista del negocio.

1.4.2 DEFINICION DE LOS REQUERIMIENTOS DEL NEGOCIO

El éxito de un data warehouse está fuertemente marcado por el entendimiento de los procesos de negocio del usuario final y sus requerimientos. Los diseñadores de data warehouse deben entender los factores clave que conducen al negocio para determinar de forma efectiva los requerimientos y traducirlos en consideraciones de diseño.

1.4.3 MODELADO DIMENSIONAL

El modelado dimensional se encuentra en la trayectoria de datos del diagrama general mostrado en la figura 1.6. La definición de los requerimientos del negocio determina los datos que se necesitan para cumplir con los requerimientos de análisis de los usuarios. Al diseñar modelos de datos que soporten estos análisis se requiere un enfoque diferente que el utilizado para el diseño de sistemas operacionales. Iniciamos construyendo una matriz que representa los procesos clave del negocio y su dimensionalidad. La matriz sirve como un plano que asegura que el data warehouse es extensible a lo largo de la organización en el tiempo.

De aquí, se realiza un análisis de datos más detallado de los sistemas operacionales que son fuentes de datos. Al acoplar este análisis de datos con un entendimiento previo de los requerimientos del negocio, podemos desarrollar un modelo dimensional que identifica la granularidad de la tabla de hechos, las dimensiones asociadas, los atributos y las rutas jerárquicas de exploración (drill) y los hechos. El diseño lógico de la base de datos se completa con las estructuras apropiadas de tablas y las relaciones entre llaves primarias y foráneas. El plan preliminar de agregación se desarrolla. Este conjunto de actividades concluye con el desarrollo del mapeo de datos origen-destino.

1.4.4 DISEÑO FISICO

El diseño físico se encuentra en la trayectoria de datos del diagrama general mostrado en la figura 1.6. Se enfoca en la definición de las estructuras físicas necesarias para soportar el diseño lógico de la base de datos. Los elementos primarios de este proceso incluyen la definición de estándares para nombres y el levantamiento del ambiente de la base de datos. Se determinan las estrategias preliminares de indexación y particionamiento.

1.4.5 DISEÑO DE LA ETAPA DE DATOS

El diseño de la etapa de datos se encuentra en la trayectoria de datos del diagrama general mostrado en la figura 1.6. El proceso de la etapa de datos tiene tres pasos mayores: extracción, transformación y carga. El proceso de extracción siempre saca a relucir controversias sobre la extracción de datos que han sido ocultadas en los sistemas operacionales. Ya que la calidad de los datos tiene un impacto significativo sobre la credibilidad del data warehouse, hay que atacar esto problemas de calidad durante la presentación de datos. Es necesario diseñar y construir dos procesos de la etapa de datos del data warehouse: Uno para la carga inicial y otro para la carga regular que es incremental.

1.4.6 DISEÑO DE LA ARQUITECTURA TECNICA

La fase de diseño de la arquitectura técnica se encuentra en la trayectoria de tecnología del diagrama del ciclo de vida.

Los ambientes de data warehouse requieren la integración de numerosas tecnologías. El diseño de la arquitectura técnica establece el almacén arquitectónico completo y la visión. Hay que considerar tres factores —los requerimientos de negocio, el ambiente técnico actual y las direcciones de estrategia técnica planeadas— para establecer el diseño de la arquitectura técnica del data warehouse.

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

1.4.7 SELECCION E INSTALACION DEL PRODUCTO

La fase de selección e instalación de producto se encuentra en la trayectoria de tecnología del diagrama del ciclo de vida.

Utilizando el diseño de la arquitectura técnica como base, es necesario evaluar y seleccionar los componentes arquitectónicos específicos tales como la plataforma del hardware, el sistema manejador de base de datos, la herramienta de acceso de datos y la herramienta para la presentación de datos. Una vez que los productos han sido evaluados (mediante un proceso de evaluación técnica) y seleccionados, se instalan y se prueban completamente para asegurar una apropiada integración punto a punto con el ambiente de data warehouse.

1.4.8 ESPECIFICACION DE LA APLICACION DE USUARIO FINAL

La fase de especificación de la aplicación de usuario final se encuentra en la trayectoria de aplicación del diagrama del ciclo de vida.

Se recomienda definir un conjunto de aplicaciones estándar de usuario final ya que no todos los usuarios necesitan un acceso a la medida al data warehouse. Las especificaciones de la aplicación describen la plantilla de los reportes, los parámetros que puede modificar el usuario y los cálculos requeridos, estas especificaciones aseguran que el equipo de desarrollo y los usuarios tienen una comprensión común de las aplicaciones que se van a entregar.

1.4.9 DESARROLLO DE LA APLICACION DE USUARIO FINAL

La fase de desarrollo de la aplicación de usuario final se encuentra en la trayectoria de aplicación del diagrama del ciclo de vida.

A continuación de la especificación, el desarrollo de la aplicación de usuario final involucra la configuración de la aplicación de metadatos y la construcción de los reportes especificados.

1.4.10 IMPLEMENTACION

La implementación representa la convergencia de la tecnología, datos y aplicaciones de usuario final accesibles desde el escritorio del usuario. Se requiere de una planeación extensiva para asegurar que las piezas de este rompecabezas se unan adecuadamente. Se debe desarrollar un proceso de capacitación que integre todos los aspectos de esta convergencia. Además, antes de que algún usuario tenga acceso al data warehouse se deben establecer estrategias de soporte y comunicación o retroalimentación.

1.4.11 MANTENIMIENTO Y CRECIMIENTO

Después de la implementación inicial del data warehouse se debe dar soporte y capacitación a los usuarios. Hay que asegurarse que los procesos están en su lugar para una efectiva operación del warehouse. Se deben realizar mediciones del desempeño a lo largo del tiempo para aplicarlas en ajustes al warehouse.

A diferencia del desarrollo tradicional de sistemas, los cambios al data warehouse se deben ver como una señal de éxito, no como una falla. Después de que se identifican las prioridades del proyecto hay que regresar al inicio del ciclo de vida apuntalando y construyendo sobre lo establecido en el ambiente del data warehouse, enfocándose en los nuevos requerimientos.

1.4.12 ADMINISTRACION DEL PROYECTO

La administración del proyecto asegura que las actividades del ciclo de vida dimensional del negocio permanezcan en su fase y en sincronía. Como se ilustra en la figura 1.6 la administración del proyecto ocurre a lo largo del ciclo de vida y se enfoca en monitorear el estado del proyecto, revisar los resultados de cada fase y preservar los límites del ámbito del proyecto. Finalmente, la administración del proyecto incluye el desarrollo de un plan comprensivo de comunicación del proyecto que dirija a los grupos del negocio y de sistemas.

En el presente trabajo se aplicará parte de la metodología del ciclo de vida presentada anteriormente ya que por el tamaño de la aplicación no es necesario considerar todos los puntos detalladamente, dos de los cuales son la especificación y el desarrollo de la aplicación de usuario final, ya que MetaCube Explorer es en sí una aplicación de usuario final a la cual hay que definirle el modelo de datos a utilizar.

En la siguiente sección se presenta el proceso de diseño dimensional que corresponde a la primera parte del Modelado Dimensional.

1.5 PROCESO DE DISEÑO DIMENSIONAL

Dentro del ciclo de vida dimensional, explicado en la sección anterior, se encuentra la fase de modelado dimensional. Este proceso define el modelo dimensional, que es el núcleo del data warehouse debido a que en él se reflejan puntualmente las actividades del negocio desde el punto de vista dimensional. A partir de este modelo se diseñan los procesos de extracción, transformación y carga que lo alimentan, así como la implementación física en la base de datos relacional de las tablas que conforman el modelo.

El proceso de diseño dimensional no debe considerarse como un ciclo lineal, ya que hay definiciones clave hechas en su etapa correspondiente, que se revisan en etapas posteriores y pueden refinarse. Dicho de otra forma, la iteración a etapas previas puede ser necesaria; la ventaja es que esa iteración a etapas no terminadas exitosamente es posible, la desventaja de este enfoque es que se vuelve más y más difícil controlar el proceso después de que ha ocurrido un cierto número de iteraciones.

A continuación se presentan los cuatro pasos a seguir para diseñar una base de datos dimensional.

1.5.1 ELEGIR UN PROCESO DE NEGOCIO A MODELAR

Un proceso de negocio es un proceso operacional mayor dentro de la organización que está soportado por alguna clase de sistema (o sistemas) existente. Ejemplos de procesos de negocios son pedidos, facturación, envíos, inventarios, contabilidad, ventas, etc. Para decidir qué procesos de negocio se van a modelar es necesario combinar la comprensión del negocio con la comprensión de qué datos están disponibles. La elección del negocio a modelar está comprendida en la fase de planeación del proyecto, debido a que en ella se define el proyecto como tal. También está comprendida en la definición de los requerimientos del negocio, ya que en esta fase se establece qué datos están disponibles. Los demás pasos explicados a continuación se encuentran en la fase de modelado dimensional del diagrama mostrado en la figura 1.6.

1.5.2 ELEGIR LA GRANULARIDAD DEL PROCESO DE NEGOCIO

La granularidad es el nivel de datos fundamental, atómico, a ser representado en la tabla de hechos. Ejemplos de granularidad son las transacciones individuales, realizadas en un momento preciso del día o del mes. Es imposible continuar con el siguiente paso sin definir la granularidad.

La granularidad es importante por que a partir de ella se determina la dimensionalidad de la tabla de hechos y esto tiene un profundo impacto en el tamaño de la base de datos. También la granularidad tiene un impacto especial sobre la frecuencia de la carga de los datos, pero sobre todo en la cantidad de datos que se deben procesar para tener una periodo de datos completamente almacenado.

En este punto del análisis se debe decidir no registrar transacciones individuales ya que el tamaño de la base de datos se puede convertir en algo no manejable y el tiempo para procesar los registros muy alto. De cualquier forma, se debe conservar cierto nivel de detalle al menor nivel posible de granularidad debido a que las consultas (queries) necesitan "hacer cortes" en la base de datos en formas muy precisas.

1.5.3 ELEGIR DIMENSIONES POR CADA REGISTRO EN LA TABLA DE HECHOS

Algunas dimensiones típicas son tiempo, producto, cliente, promoción, almacén, tipo de transacción y estado. Con cada dimensión elegida hay que describir todos los atributos dimensionales (campos) que sean discretos y de tipo texto, para llenar cada tabla.

Una definición cuidadosa de la granularidad determina la dimensionalidad inicial de la tabla de hechos. Es entonces cuando usualmente es posible agregar otras dimensiones a la granularidad básica de la tabla de hechos, donde estas dimensiones adicionales naturalmente toman solo un valor bajo cada combinación de las dimensiones primarias. Si se reconoce que una dimensión adicional viola la granularidad al ser necesarios más registros para generarla, entonces la definición de la granularidad debe revisarse para acomodar esta dimensión adicional.

1.5.4 ELEGIR LAS METRICAS PARA LA TABLA DE HECHOS

Las métricas típicas son cantidades numéricas y aditivas, como la cantidad vendida y el importe vendido.

La granularidad de la tabla de hechos permite elegir los hechos individuales y clarifica cual debe ser el ámbito de esos hechos. En el caso de tablas de hechos con transacciones individuales, usualmente existe un solo hecho, nombrado como el monto de la transacción. Los hechos o métricas siempre deben ser específicos a la granularidad de la tabla de hechos.

A veces al examinar las métricas descubriremos que es necesario realizar algunos ajustes ya sea en las definiciones de la granularidad o en el número de las dimensiones.

Por mucho la tabla de hechos es la más grande en la base de datos dimensional. Por definición las tablas de dimensión son casi siempre geoméricamente más pequeñas que la de hechos; así que toda estimación realista del espacio en disco requerido para el data warehouse puede ignorar a las tablas de dimensión; por lo que los esfuerzos por normalizar cualquiera de las tablas en una base de datos dimensional solamente para ahorrar espacio en disco son una pérdida de tiempo.

A lo largo de este documento se utilizan estos pasos detalladamente para el desarrollo de la solución al problema planteado. Hay que hacer notar que las perspectivas necesarias para implementar las decisiones que implican los puntos anteriores, dependen de la comprensión de los requerimientos del usuario final y de las fuentes de datos disponibles en los sistemas existentes.

1.6 HERRAMIENTAS DE MODELADO DE SISTEMAS

ANALISIS ESTRUCTURADO

El análisis es el proceso de definir los requerimientos para la solución de un problema. Durante el análisis las necesidades de los usuarios son examinadas y se definen las propiedades que el sistema debería tener para cumplir con esas necesidades. También se definen las restricciones del sistema y los requerimientos de desempeño. Se definen de forma precisa las funciones que se llevarán a cabo, pero no se define como trabajarán esas funciones.

El análisis es un proceso crítico en el desarrollo de sistemas y programas debido a que afecta a todos los pasos que siguen dentro del desarrollo, también es un proceso difícil por que pueden existir problemas de comunicación, cambios en los requerimientos del sistema y por que puede darse una estimación inadecuada de tiempos y costos.

Las especificaciones que siguen a un buen análisis deberían ser un modelo del sistema. Deberían ser fáciles de escribir, fáciles de entender y fáciles de cambiar.

El análisis estructurado propone resolver estas dificultades mediante un enfoque sistemático, paso a paso para realizar análisis y mediante la producción de una nueva y mejorada especificación de sistema. Para cumplir este objetivo el análisis estructurado se enfoca en una comunicación clara y concisa.

El análisis estructurado se basa en los siguientes conceptos:

- Top-down, organización jerárquica.
- Divide y conquistarás.
- Comunicación gráfica/herramientas de documentación.

El análisis estructurado utiliza el método top-down o de descomposición funcional para definir los requerimientos del sistema. La especificación del sistema producida por el proceso de análisis estructurado es un modelo top-down del sistema a ser construido. Debido a que la especificación es un modelo gráfico que es conciso y fácil de entender, el usuario se puede familiarizar con el sistema mucho antes de que se implemente. Esto proporciona una oportunidad de revisar el sistema para descubrir y corregir errores o malas interpretaciones tan pronto como sea posible durante el proceso de desarrollo.

ESPECIFICACION ESTRUCTURADA

La especificación del sistema es el conjunto de documentos que se producen con el análisis y definen la estructura del problema a ser resuelto tal como el usuario lo visualiza.

La especificación de sistema producida por el proceso del análisis estructurado es llamada especificación estructurada y consta de los siguientes elementos:

1. DIAGRAMAS DE FLUJO DE DATOS
2. DICCIONARIO DE DATOS
3. ESPECIFICACIONES DE PROCESO

1.6.1 DIAGRAMAS DE FLUJO DE DATOS

En esta sección se presenta una introducción a las técnicas de diagramación estructurada, necesarias para definir los procesos de extracción, transformación y carga del data warehouse. Estas técnicas se basan principalmente en la metodología desarrollada por E. Yourdon y T. De Marco, aunque también se mencionan las aportaciones de otros autores como C. Gane y T. Sarson.

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

Unos buenos y claros diagramas juegan una parte esencial en el diseño de sistemas complejos y en el desarrollo de programas. La forma en que seamos capaces de pensar depende del lenguaje que usamos para pensar. Cuando se utilizaban solamente números romanos, la gente común no podía multiplicar o dividir. Esa capacidad se amplió cuando los números arábigos comenzaron a usarse ampliamente. Los diagramas que dibujamos de procesos complejos son una forma de lenguaje. Con las computadoras podemos crear procesos más complejos que aquellos que pudiéramos realizar a mano. Los diagramas apropiados nos ayudan a visualizar e inventar esos procesos.

Si solamente una persona está llevando a cabo el diseño de un sistema o programa, los diagramas que utilice serán una ayuda para pensar claramente. Una pobre elección de la técnica de diagramación puede inhibir el pensamiento. Una buena elección podrá acelerar su trabajo y mejorar la calidad de los resultados.

Cuando varias personas trabajan en un sistema o programa los diagramas son una herramienta esencial de comunicación. Se necesita una técnica formal de diagramación para que los desarrolladores sean capaces de intercambiar ideas y para hacer que sus componentes separados se unan con precisión.

Los diagramas claros son una ayuda esencial para el mantenimiento. Hacen posible para un nuevo equipo entender como trabajan los programas y para diseñar los cambios. Un cambio en un lugar comúnmente afecta otras partes del programa. Si los diagramas de la estructura del programa son claros, entonces los programadores tendrán la capacidad de entender los efectos de los cambios que realizan.

Cuando se realiza alguna depuración, los diagramas claros son una herramienta muy valiosa para entender como deberían funcionar los programas y para rastrear lo que pudiera estar mal.

Si los arquitectos, agrimensores y diseñadores de partes de máquinas tienen técnicas formales de elaborar diagramas que ellos deben seguir, los analistas de sistemas y diseñadores de programas tienen la necesidad de diagramas claros debido a que estas actividades son también complejas y el trabajo de gente diferente debe entrelazarse en formas intrincadas.

Entre más grande sea el equipo de trabajo, es más grande la necesidad de precisión en los diagramas. Es difícil o imposible para los miembros de un equipo grande entender a detalle el trabajo de los otros, en lugar de esto cada miembro del equipo debería ser familiar con una visión del sistema y ver donde encaja su componente, para que en caso de cambio en su diseño no afecte el diseño de los demás.

Las técnicas de diagramación estructurada ofrecen muchas ventajas, primero, combinan las notaciones gráfica y narrativa para incrementar la comprensión. Las gráficas son especialmente útiles ya que tienden a ser menos ambiguas que la descripción narrativa; también y debido a que tienden a ser más concisas, las gráficas pueden dibujarse en menos tiempo que el que tomaría escribir un documento narrativo que contenga la misma cantidad de información. En segundo lugar las técnicas de diagramación estructurada soportan un enfoque top-down de desarrollo estructurado. Pueden describir un sistema o programa en varios niveles de detalle durante cada paso del proceso de descomposición funcional. Clarifican los pasos y el resultado del proceso de descomposición funcional al tener una forma estándar de describir la lógica de los procedimientos y las estructuras de datos.

Un diagrama de flujo de datos muestra procesos y el flujo de datos entre esos procesos. En un nivel alto es utilizado para mostrar eventos del negocio y las transacciones resultantes de esos eventos, sean en papel o en transacciones de computadora. En un nivel bajo se utiliza para mostrar programas o módulos de programa y el flujo de datos entre estas rutinas.

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

Un diagrama de flujo de datos se utiliza como el primer paso en una forma de diseño estructurado. Muestra *el flujo total de datos a través de un sistema o programa*.

Un diagrama de flujo de datos muestra como fluye un dato a través de un sistema lógico, pero no da información de control o de secuencia.

Aunque Yourdon, De Marco y otros han propuesto componentes de los diagramas de flujo de datos que hoy son muy conocidos, los autores Gane y Sarson adoptaron componentes un tanto diferentes, que en cierta forma tienen una mejor notación que aquellos popularizados por Yourdon.

En la tabla 1.1 se presenta un resumen de ambas notaciones para los diagramas de flujo de datos.

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

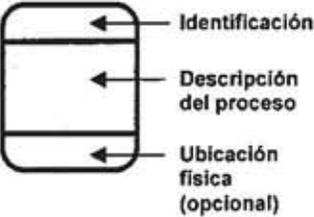
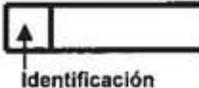
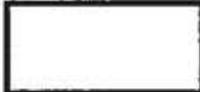
<i>Función del componente</i>	<i>Símbolo de acuerdo a Yourdon, De Marco y otros</i>	<i>Símbolo de acuerdo a Gane y Sarson</i>
Flujo de datos		
Proceso que transforma los datos		
Almacén de datos		
Fuente externa o destino de los datos		
Flujo de materiales	No tiene	
Almacén de datos mostrado varias veces en un diagrama	No tiene	
Fuente externa o destino mostrado varias veces en un diagrama	No tiene	

Tabla 1.1 Notaciones para diagramas de flujo

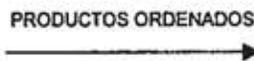
En la sección siguiente se explica cada uno de los componentes de los diagramas de flujo.

1.6.1.1 COMPONENTES DE UN DFD

Un diagrama de flujo de datos (DFD) es una representación en red de un sistema mostrando los procesos y las comunicaciones de datos entre ellos. El DFD se construye a partir de cuatro componentes básicos: el flujo de datos, el almacén de datos y el terminador.

FLUJO DE DATOS

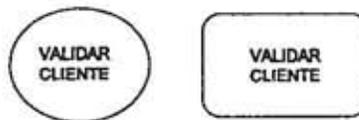
El flujo de datos traza el flujo de datos a través de un sistema o de procesos. La dirección del flujo se indica por una flecha. Los datos se identifican por nombre, con este nombre escrito a lo largo de su flecha correspondiente, por ejemplo:



El flujo de datos muestra como están conectados los procesos.

PROCESO

El proceso opera sobre datos o los transforma. Por ejemplo, puede realizar operaciones aritméticas o lógicas sobre datos para producir algún resultado. Cada proceso se representa por un círculo o un rectángulo con las esquinas redondeadas en el DFD. El nombre del proceso se escribe dentro de la figura. Se debe utilizar un verbo en infinitivo para describir la operación realizada por el proceso como se puede apreciar en el ejemplo que se muestra a continuación:



En el DFD no se muestra más información de lo que hace el proceso. Normalmente los datos fluyen hacia y desde cada proceso. Comúnmente existen múltiples flujos de datos hacia y desde un proceso como se muestra en el siguiente ejemplo:



CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS
ALMACEN DE DATOS

Un almacén de datos representa un archivo, una tabla o incluso una base de datos. En el DFD se dibuja como un par de líneas paralelas, algunas veces cerradas del lado izquierdo. El nombre del almacén de datos se escribe entre las líneas como se muestra en el siguiente ejemplo:



Cada almacén de datos se conecta al proceso por medio de datos que fluyen, la dirección de la flecha muestra si los datos están siendo leídos del almacén de datos hacia el proceso o producidos por el proceso y entonces llevados hacia el almacén de datos, esto es que si la flecha sale de un almacén de datos significa que se están leyendo datos del almacén, si la flecha entra significa que se están escribiendo datos al almacén.

En el ejemplo que se muestra a continuación, la información de error se produce por el PROCEDIMIENTO DE ERROR y escrita al ARCHIVO DE ERROR:



LIMITADOR

Un limitador muestra el origen de los datos utilizados por el sistema y al último receptor de datos producidos por el sistema. Al origen de los datos se le llama *fuentes* y al receptor de los datos se le llama *destino*. Para representar a un limitador en un DFD se utiliza un rectángulo (como se muestra enseguida) o una cuadrado doble:



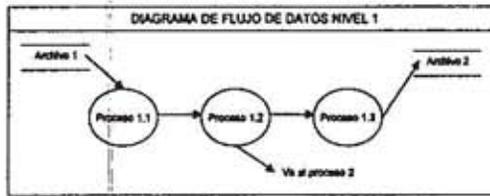
Los limitadores en realidad están fuera del DFD.

1.6.1.2 NIVELES DE UN DFD

El DFD es una herramienta para análisis top-down. Podemos utilizar DFDs para proporcionar vistas tanto de alto nivel como detalladas de un sistema o programa. Lo que se realiza dentro de una caja de proceso en un DFD puede ser mostrado en detalle en otro DFD. El DFD inicial se dice que es el DFD nivel 0, cada DFD que detalle algún proceso dentro del DFD 0 será el DFD n, donde n corresponde al número de proceso correspondiente. En el ejemplo que se muestra a continuación se visualiza esta idea.



Este diagrama muestra los procesos generales que son Proceso 1 y Proceso 2. Este diagrama es el DFD de nivel 0 por que muestra el nivel más alto de los procesos.



El DFD nivel 1 muestra los subprocesos que componen al Proceso 1, es importante notar que las entradas al Proceso 1 son los mismos datos que entran al DFD nivel 1, de igual forma las salidas del Proceso 1 son los datos de salida de este DFD.



El DFD nivel 2 muestra los subprocesos que componen al Proceso 2, los nombres de estos subprocesos contienen al número de proceso del cual vienen.

Las principales características de representar los diagramas de flujo por niveles son:

- El primer nivel es el más alto y es el 0.
- Cada nivel siguiente representa los subprocesos del proceso respectivo, o sea que el Nivel n representa los subprocesos del proceso n.
- Los nombres de los subprocesos son subíndices del proceso del cual vienen.
- Es muy importante conservar la consistencia entre diagramas de diferentes niveles, por lo que los datos que entran a un proceso de uno o varios almacenes son los mismos que deben entrar al diagrama respectivo, lo mismo aplica para los datos de salida.

Se puede continuar expandiendo la visión del sistema al ir observando en cada proceso tan lejos como se necesite, pero una regla general es que se puede seguir expandiendo los procesos o definiendo niveles de detalle de los procesos hasta que las operaciones

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

realizadas por cada proceso se puedan describir en una sola página, utilizando pseudocódigo o español estructurado por ejemplo.

A cada nivel el DFD podría contener menos de 12 procesos, preferentemente seis o siete, ya que los DFDs grandes significan que se está tratando de mostrar demasiado detalle y son difíciles de leer.

1.6.1.3 GUIA PARA LA CONSTRUCCION DE UN DFD

Existe un número de reglas adicionales que se requieren para elaborar un DFD con éxito. Algunas de estas reglas ayudarán a no elaborar DFD erróneos (por ejemplo, incompletos o lógicamente inconsistentes) o que no sean fáciles de leer. Estas reglas son:

1. Escoger nombres con significado para los procesos, flujos, almacenes y limitadores.
2. Numerar los procesos.
3. Redibujar el DFD tantas veces como sea necesario para que se facilite su lectura y presentación.
4. Evitar los DFD excesivamente complejos.
5. Asegurar que cada DFD sea internamente consistente y que también lo sea con cualquier DFD relacionado con él.

1.6.2 DICCIONARIO DE DATOS

En muchas aplicaciones de sistemas de información, el volumen de los datos es esencial. Cuando trabajan sobre un sistema varios equipos de analistas, la tarea de coordinar las definiciones de los datos se vuelve más compleja. Los individuos dependen de las definiciones establecidas por otros y de sus propias suposiciones con respecto a las especificaciones de los datos.

El diccionario de datos es un listado organizado de todos los datos pertinentes al sistema, con definiciones precisas y rigurosas para que tanto el analista tenga un entendimiento de todas las entradas, salidas, componentes de almacenes y cálculos intermedios. El diccionario de datos define los datos haciendo lo siguiente:

- Describe el significado de los flujos y almacenes que se muestran en los DFD.
- Describe la composición de paquetes de datos agregados que se mueven a lo largo de los flujos, es decir, paquetes complejos (por ejemplo el domicilio de un cliente), que pueden descomponerse en unidades elementales (como ciudad, estado y código postal).
- Describe la composición de los datos en los almacenes de datos.
- Especifica los valores y unidades relevantes de piezas elementales de información en los flujos de datos y en los almacenes de datos.
- Para el caso del CICA describe los detalles de las relaciones entre almacenes que se enfatizan en un diagrama del modelo dimensional.

1.6.2.1 NOTACION DEL DICCIONARIO DE DATOS

Existen muchos esquemas de notación para diccionarios de datos. El que se muestra a continuación es de los más comunes y utiliza varios símbolos sencillos:

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

Símbolo	Significado
=	Está compuesto de
+	Y
()	Optativo (puede o no estar presente)
{ }	Iteración
[]	Seleccionar una de varias alternativas
**	Comentario
@	Identificador (campo clave) para un almacén
	Separa opciones alternativas en una construcción

Por ejemplo si deseamos definir el Nombre de una persona con la notación presentada arriba, sería así:

Nombre = Título de cortesía + Nombre + (Segundo nombre) + Apellido

Título de cortesía = [Sr. | Srita. | Sra. | Dr. | Lic. | Prof.]

Nombre = {carácter legal}

Segundo nombre = {Carácter legal}

Apellido = {Carácter legal}

Carácter legal = [A-Z | a-z | 0-9 | ' | - |]

DEFINICIONES

La definición de un dato se introduce con el símbolo "=". En este contexto el signo "=" se lee "se define como" o "significa", por ejemplo la notación:

$$A = B + C$$

Puede leerse de las siguientes formas:

- Cuando digamos A, queremos decir una B y una C.
- A se compone de B y C.
- A se define como B y C.

Para definir por completo un dato, la definición debe incluir lo siguiente:

- El *significado* del dato dentro del contexto de la aplicación, por lo común se ofrece como comentario utilizando los asteriscos "**".
- La *composición* del dato, si se compone de partes elementales con significado.
- Los *valores* que puede tomar el dato, si es un dato elemental que no puede descomponerse más.

Así si estuviéramos construyendo un sistema médico que siga la evolución de los pacientes, podrían definirse los términos peso y estatura de la siguiente manera:

Peso = *peso del paciente al ser admitido al hospital*
 unidades: kilogramos; escala: 1-200

Estatura = *estatura del paciente al ser admitido al hospital*

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

unidades: centímetros; escala: 20-200

ELEMENTOS DE DATOS BASICOS

Las partes elementales de los datos son aquellas para las cuales ya no existe una descomposición con significado dentro del contexto del ambiente del usuario.

Cuando se han identificado los datos elementales deben introducirse al diccionario de datos. Como se indicó anteriormente, el diccionario de datos debe proporcionar una breve narrativa, encerrada entre asteriscos ** que describa el significado del término en el contexto del usuario. Desde luego, habrá términos que se definan solos, es decir, cuyo significado es universal para todos los sistemas de información o donde el analista pudiera estar de acuerdo en que no se necesita aclarar más. Por ejemplo, los siguientes pudieran considerarse términos que se auto definen en un sistema que maneja información sobre personas:

Estatura actual
Peso actual
Fecha de nacimiento
Sexo
Teléfono particular

En estos casos no se necesita un comentario narrativo; se puede usar la notación **** para indicar "sin comentarios" cuando el dato se defina por sí mismo. Sin embargo, es importante especificar los valores y unidades de medida que los datos elementales pueden tomar. Por ejemplo:

Peso actual =**
unidades: libras; escala: 1-400

Fecha de nacimiento = **
unidades: días a partir del 1° de enero de 1900; escala: 0-36500

DATOS OPCIONALES

Un dato opcional, como la frase implica, es aquel que puede estar o no presente en un dato compuesto. Existen muchos ejemplos de datos opcionales en sistemas de información:

- El nombre de un cliente pudiera o no incluir un segundo nombre.
- El domicilio de un cliente pudiera incluir o no información secundaria, como el número de departamento.

Las situaciones de este tipo deben verificarse con cuidado con el usuario y deben documentarse de forma precisa en el diccionario de datos. Por ejemplo, la notación:

Domicilio de cliente = (domicilio de envío) + (domicilio para cuenta)

Significa, literalmente, que el domicilio del cliente pudiera consistir de:

- Solo un domicilio de envío.
O bien
- Solo un domicilio para enviar cuentas.
O bien

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

- Un domicilio de envío y uno para cuentas.

O bien

- Ninguno de los dos

Esta última posibilidad es dudosa. Es mucho más probable que el usuario realmente quiera decir que el domicilio debe consistir de uno u otro o de ambos. Esto pudiera expresarse mejor de la siguiente manera:

Domicilio del cliente = [domicilio de envío | domicilio para cuentas | domicilio de envío + domicilio para cuentas]

Desde luego, la única manera de saber esto es pedirle al usuario que explique con cuidado las implicaciones de las diferentes notaciones posibles.

ITERACION

La notación de iteración se usa para indicar la ocurrencia repetida de un dato. Se lee como "ceros o más ocurrencias de". Así, la notación siguiente:

Solicitud = nombre del cliente + domicilio de envío + {artículo}

Significa que la solicitud siempre debe contener un nombre de cliente, un domicilio de envío y también cero o más ocurrencias de artículo.

En muchas situaciones reales, el usuario querrá especificar los límites inferior y superior de la iteración. Tal vez, en el ejemplo anterior, el usuario señale que no tiene sentido que un cliente haga un pedido de cero artículos; debe haber por lo menos uno. Podría también especificarse un límite superior; quizá, se permitirán cuando más 10 artículos. Puede indicarse esto de la siguiente forma:

Solicitud = nombre del cliente + domicilio de envío + 1{artículo}10

Es correcto especificar sólo el límite inferior, sólo el límite superior, ambos o ninguno. Así que se permite cualesquiera de los siguientes:

A = 1{B}

A = {B}10

A = 1{B}10

A = {B}

SELECCION

La notación de selección indica que un dato consiste de exactamente un elemento de entre un conjunto de opciones alternativas. Las opciones se encierran en corchetes "[" y "]" y se separan por una barra vertical "|". Como ejemplo típicos tenemos:

Sexo = [Femenino | Masculino]

Tipo de cliente = [Gobierno | Industria | Universidad | Otro]

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

De igual forma que otros medios de representar estructuras de programas, el español estructurado debe ser diseñado para mostrar cuatro construcciones básicas:

1. **Secuencia.** Se utiliza una secuencia simple y con sentido de arriba abajo (top-to-bottom).
2. **Condición.** Si alguna cierta condición aplica, entonces se llevará a cabo una acción dada; si no, se debe especificar una acción diferente.
3. **Caso.** Existe uno o varios casos posibles. La estructura muestra que acción se toma para cada caso posible. Un conjunto de condiciones mutuamente exclusivas es una *estructura de caso*.
4. **Repetición.** Se repite un conjunto dado de operaciones. Se muestra la condición que termina esta repetición. Las estructuras de repetición son de dos tipos:
 - **REPETIR MIENTRAS:** Las operaciones se repiten mientras aplique una condición especificada. Esta condición se prueba antes de la ejecución de las operaciones.
 - **REPETIR HASTA.** Las operaciones se repiten hasta que una condición especificada exista. Esta condición se prueba después de la ejecución de las operaciones.

Nota: Dentro del español estructurado, de ser necesario, se escribirán sentencias SQL que mejoren la especificación del proceso.

DIAGRAMAS DE ACCION

De las técnicas existentes para hacer diagramas algunas son útiles para visualizar la estructura de los programas y otras son útiles para detallar la lógica de los programas. Los diagramas de acción cumplen con estas dos características debido a que consideran que la definición de bajo nivel es una extensión natural del alto nivel.

Los componentes de los diagramas de acción son los siguientes:

- **BRAZOS**

Son los elementos básicos de construcción de los diagramas de acción. El brazo puede ser de cualquier longitud, de tal forma que tenga espacio para el texto o detalle que haga falta.

Dentro del brazo va una secuencia de operaciones. Una regla sencilla de control aplica al brazo: Uno entra en la parte de arriba, hace las cosas que tenga que hacer en una secuencia de arriba abajo y sale en la parte de abajo.

Dentro de un brazo pueden haber otros brazos. Se pueden anidar muchos brazos; la anidación muestra la estructura jerárquica de un programa. En la figura 1.7 se muestra la representación de una estructura jerárquica de bloques y su equivalente diagrama de acción.

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

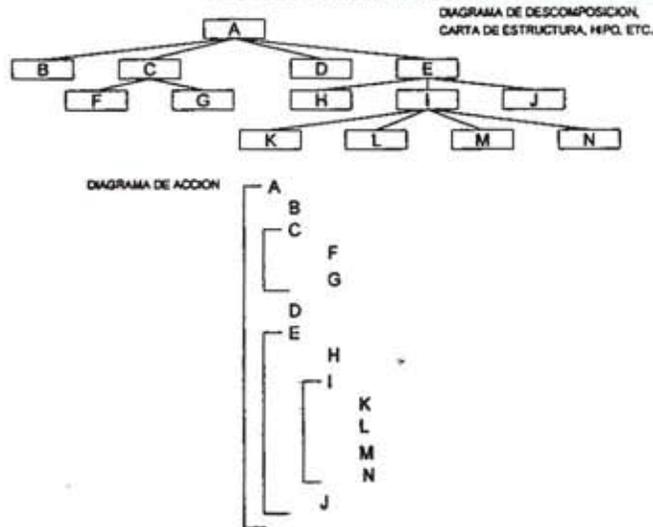
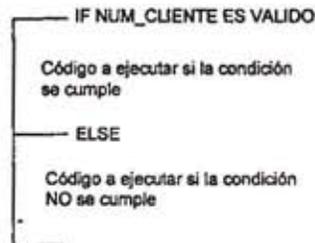


Figura 1.7 Comparación del Diagrama de Acción y uno de bloques

• CONDICIONES

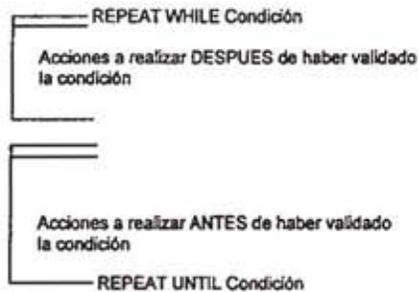
Comúnmente un módulo o subrutina se ejecuta solo si cierta condición aplica, cuando la condición no se cumple se puede utilizar la cláusula ELSE. En este caso la condición se escribe en el encabezado del brazo como se muestra en el siguiente ejemplo:



• CICLOS

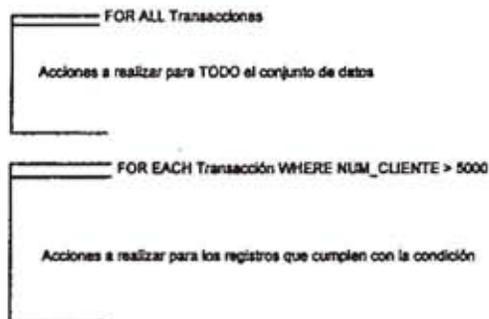
Un ciclo se representa con un brazo de repetición que tiene una línea doble en su parte alta, las acciones dentro de cada ciclo se pueden realizar mientras se cumpla cierta condición (cláusula WHILE) o hasta que se cumpla cierta condición (cláusula REPEAT); un ejemplo se muestra a continuación:

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS



• CONJUNTOS DE DATOS

Algunas veces un procedimiento necesita ser ejecutado para todos los componentes de un conjunto, por ejemplo que tenga que ser aplicado para todos los registros de una tabla. Otra opción para representar la afectación de un conjunto de datos es agregar la cláusula WHERE para calificar al FOR; ejemplos de ambos casos se muestran a continuación:



• SUBPROCEDIMIENTOS

A veces un usuario necesita agregar a un diagrama de acción un elemento que por sí mismo sea un procedimiento que contenga acciones. A este elemento se le llama subprocedimiento o subrutina y se dibuja como una caja con las esquinas redondeadas. Un subprocedimiento puede utilizarse en varios procedimientos. En el ejemplo siguiente se muestra la representación de un subprocedimiento.



• SUBPROCEDIMIENTOS NO DISEÑADOS

Algunas veces en la etapa de diseño de procedimientos, algunas secciones de un procedimiento no han sido todavía pensadas en detalle. El diseñador puede representar esto con una caja con esquinas redondeadas y signos de interrogación "?" al lado derecho.

- **PROCEDIMIENTOS COMUNES**

Algunos procedimientos aparecen más de una vez en un diagrama de acción debido a que son llamados desde más de un lugar dentro de la lógica. Estos procedimientos son llamados procedimientos comunes. Se indican dibujando una línea vertical del lado izquierdo de la caja de procedimientos, como se muestra a continuación:



- **FINALIZADORES**

Ciertas condiciones pueden causar que un procedimiento termine. Estas condiciones pueden causar la finalización del brazo en que la condición ocurre o pueden causar la finalización de varios brazos. Los finalizadores se dibujan con una flecha hacia la izquierda a través de uno o más brazos, como se muestra en los siguientes ejemplos:



1.7 SEGURIDAD

Una de las ironías del trabajo del administrador de data warehouse es la tensión entre la responsabilidad de publicar los datos y la responsabilidad de protegerlos. Por un lado, se juzga al administrador por que tan fácilmente un usuario final puede acceder todos los datos en el data warehouse. Por otro lado, se culpa al administrador si alguna información delicada llega a manos equivocadas o si los datos se pierden.

Aunque en el caso del CICA no se utiliza Internet para que el usuario consulte los datos de los cajeros, se tiene planeado para versiones posteriores migrar de la arquitectura cliente/servidor a una arquitectura basada en Web, además de que la mayoría de las aplicaciones departamentales de BanCrecer serán migradas a Java, por lo que en esta sección se presentan los conceptos generales de ligados a la seguridad en Internet.

Los administradores deben manejar activamente la seguridad del data warehouse, por lo que deben tener un buen entendimiento de los tópicos de seguridad, de tal forma que puedan contratar y supervisar expertos en seguridad que se dediquen al esfuerzo de data warehouse. Supervisar la seguridad en un ambiente extenso de data warehouse es un trabajo de tiempo completo para al menos una persona.

Por si los tópicos de seguridad no fueran suficientes, el crecimiento de la Internet ha sido como un tren expreso que ha corrido sobre las áreas de sistemas en general y sobre las instalaciones de data warehouse en particular. Al tiempo que las configuraciones de data warehouse se han estado moviendo lentamente de las configuraciones de dos capas (con un cliente robusto conversando directamente con la máquina del manejador de base de datos) a configuraciones de tres capas con un servidor de aplicaciones entre el cliente y el manejador de base de datos, ha llegado Internet con varias capas adicionales. Cada administrador de data warehouse tiene ahora la carga adicional de tratar con un servidor Web, un servidor de directorios y varias capas de firewalls y filtros empacados.

El administrador moderno de sistemas debe entender a la Internet debido a que casi cualquier componente del data warehouse se lleva a trabajar en el ambiente de Web, aún con redes que no estén conectadas con la Internet real. Así que no solo tiene la complejidad técnica del ambiente de data warehouse incrementada, también se han incrementado los riesgos del data warehouse. Los propietarios de data warehouse no pueden ignorar el impacto de la Internet en su misión y su propio trabajo, por lo que deben educarse acerca de los riesgos, las tecnologías y las nuevas perspectivas administrativas traídas por la Internet.

Una vista razonable y básica de la arquitectura que está entre el usuario y los datos se puede ver como en la figura 1.8. En esta sección discutiremos cada uno de los componentes de esta figura con el detalle suficiente para instruir al administrador de data warehouse en lo que es importante.

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

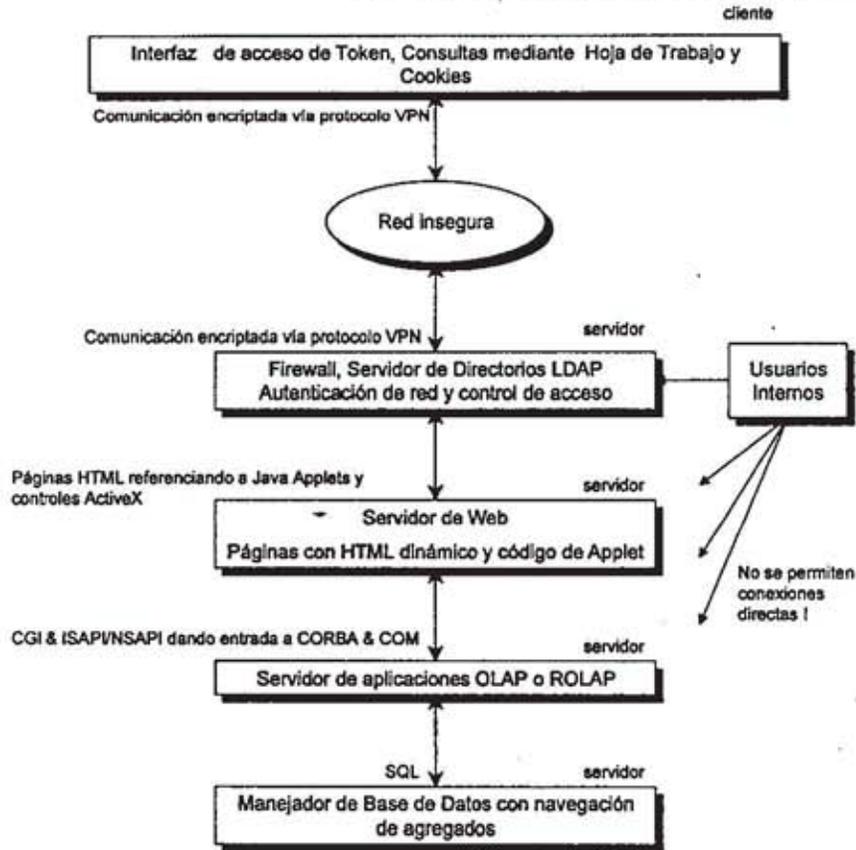


Figura 1.8 Arquitectura que está entre el usuario y los datos

Iniciamos la discusión de la seguridad con un breve repaso de la arquitectura de componente de software.

ARQUITECTURA DE COMPONENTE DE SOFTWARE

En el periodo de 1995 a 1997 se vio un número de avances significativos en la forma en que pensamos acerca de los sistemas distribuidos. Un profundo avance arquitectónico ha surgido poco a poco tanto dentro como fuera del campo de Microsoft. Este avance es conocido como *componente de software*. Hace algún tiempo la industria de software se dio cuenta que la tarea de producir grandes sistemas de software había crecido de forma demasiado compleja para continuar usando un enfoque monolítico. En el enfoque monolítico, cada nuevo y grande sistema de software se codifica desde cero y todas las interfaces entre las rutinas de programas se definen como procedimientos del proyecto. Este enfoque no tan solo falla en capitalizar el software previamente construido, sino que el tamaño del software moderno hace imposible de manejar un esfuerzo de desarrollo monolítico. Una de las versiones de Microsoft Windows 95 se reportó que estaba basada en 14 millones de líneas de código. Tal tamaño de software requiere el esfuerzo de

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

literalmente miles de desarrolladores a lo largo de un cierto número de años. La única esperanza de construir sistemas tan complicados, cumpliendo calendarios de entrega de versiones cada vez más agresivos, es basando todo en componente de software.

Hacia el final de 1997 emergieron dos competidores de sistemas para el componente de software. El sistema de Microsoft está basado en una arquitectura conocida como Modelo Objeto de Componente Distribuido (en inglés el nombre es: Distributed Component Object Model), las iniciales por el nombre en inglés son DCOM. Aunque la especificación de DCOM ya no está controlada por Microsoft, la compañía permanece como el proveedor dominante. Su conjunto de servicios y aplicaciones que usan componente de software se conoce como Plataforma Activa, en inglés el nombre es Active Platform, los componentes en sí son controles ActiveX y las aplicaciones que se invocan en clientes y servidores para lanzar estos componentes son una generalización de páginas HTML conocidas como Páginas de Servidor Activo (en inglés el nombre es: Active Server Pages). Los scripts para controles ActiveX y de Active Server Page pueden escribirse en numerosos lenguajes, incluyendo Visual Basic y C.

El enfoque competidor del componente de software está patrocinado por el campo fuera de Microsoft, que incluye a Sun Microsystems, Netscape y Oracle. Este sistema está basado en la Arquitectura de Petición al Corredor de Objeto Común (en inglés el nombre es: Common Object Request Broker Architecture), las iniciales por el nombre en inglés son CORBA. Este conjunto de servicios utiliza software componente conocido como Java Empresarial (en inglés el nombre es: Enterprise Java), los componentes en sí son JavaBeans, que pueden ser llamados desde páginas HTML y deben ser escritos en lenguaje Java.

El impacto del software hecho componente es más significativo que solamente entender como los desarrolladores profesionales de software unen sus productos. La naturaleza del software ha cambiado en formas fundamentales:

- El software se puede vender y cambiar de versiones en una manera mucho más modular. Una actualización o paquete de servicios requerido consiste de los componentes de software seleccionados.
- Muchos más distribuidores y equipos de desarrollo de software pueden agregar funcionalidad a un sistema al producir componentes de software aislados que obedezcan a interfaces estandarizadas de componente.
- Los departamentos de sistemas le comprarán sistemas grandes a grandes vendedores de software, pero tendrán muchas oportunidades para desarrollar sus propios componentes que entren en su esquema. Para agregar componentes propios el departamento de sistemas tendrá que invertir en arquitecturas modernas y robustas de componente de software como DCOM y CORBA.
- Los componentes de software tienen protocolos muy específicos para comunicarse entre ellos. Estos protocolos de comunicación permiten a los módulos de software buscarse en diferentes partes de la memoria, en diferentes ubicaciones en el disco duro local y en máquinas remotas, todo de forma transparente desde el punto de vista del usuario. Los componentes de software usualmente también se escriben para ser multitareas por lo que un solo componente puede usarse por varias aplicaciones simultáneamente.
- Los componentes de software propician a las aplicaciones distribuidas, en las cuales los variados componentes que interactúan residen en diferentes máquinas.

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS
EL IMPACTO DE INTERNET SOBRE LA ARQUITECTURA

La Internet ha propiciado y acelerado el uso de componentes de software y ha puesto una tremenda presión sobre los fabricantes para resolver un número de problemas de compatibilidad, desempeño y seguridad. Repentinamente el paradigma del cliente ha cambiado, todos los clientes se han vuelto navegadores de HTML. Algunos de los navegadores residen en una red local y algunos se conectan por medio de ella o de una línea telefónica a Internet. En muchos casos, los mismos usuarios se cambian diariamente entre las tres formas de acceso con sus navegadores y esperan el mismo software para trabajar sin importar como accedan al data warehouse. Al mismo tiempo, el original, monolítico manejador de base de datos se está fragmentando en múltiples procesos que corren en múltiples máquinas. Puede haber un navegador de agregados que intercepte las instrucciones de SQL originalmente enviados al manejador de bases de datos relacionales y dirigido al SQL a un número separado de máquinas. Puede haber un servidor de aplicaciones cuya tarea es ensamblar un reporte o análisis complejo mediante la separación de las consultas a múltiples fuentes de datos.

En la figura 1.9 se muestra como un cliente Web (un navegador basado en HTTP) interactúa con un servidor de Web para establecer una sesión con una aplicación de base de datos. Se muestra la conexión entre el cliente y el servidor, ya sea estando sobre Internet o sobre una intranet protegida para enfatizar que las dos arquitecturas son idénticas. Las capas de seguridad se presentan hasta la siguiente sección para poder discutir el flujo de control considerando el enfoque del componente de software.

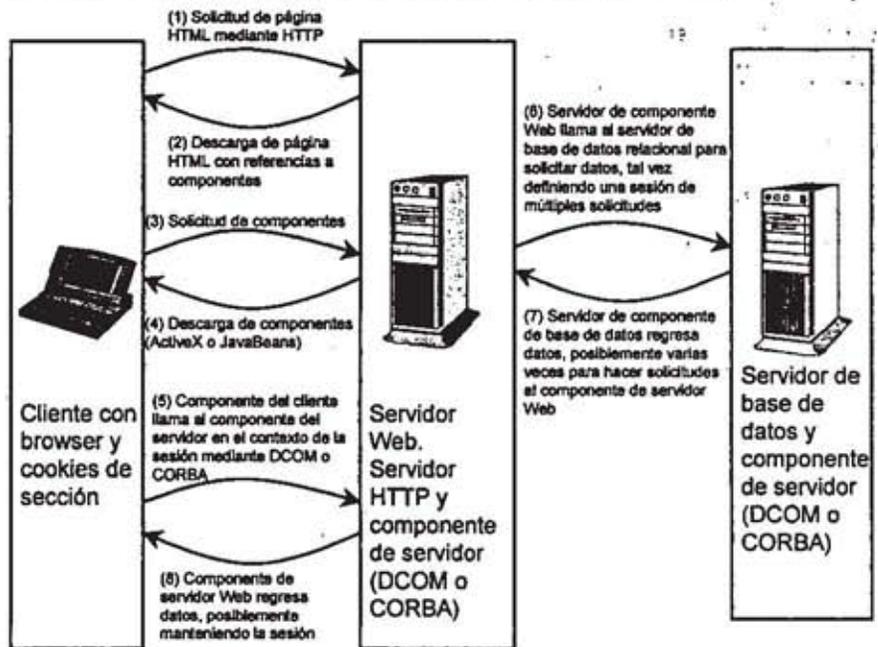


Figura 1.9 Cliente Web interactuando con un servidor de Web

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

Tomando como referencia la figura 1.9 seguimos los pasos 1 al 8 para entender como se maneja la sesión de usuario. En el paso 1, el usuario en el navegador del cliente especifica una URL basada en HTTP ubicada en un servidor Web remoto. Asumimos que el navegador del cliente y el servidor Web están conectados mediante Internet o intranet. El paso 2 provoca que una página HTML sea enviada desde el servidor Web hacia el navegador del cliente. La página HTML contiene referencias a componentes de software descargables o aplicaciones más complejas que consisten de componentes de software. En el mundo Microsoft estos componentes descargables son controles ActiveX, en el mundo fuera de Microsoft son JavaBeans, o más genéricamente, applets de Java que pueden estar compuestos de JavaBeans.

En los pasos 3 y 4, el navegador del cliente descarga los componentes de software solicitados y los carga en memoria. La nueva capacidad clave que tienen estos componentes, es que se pueden comunicar directamente con otros componentes, ya sea desde cualquier parte del navegador cliente o en el servidor Web. No necesitan lanzar una nueva petición HTTP o abrir una nueva página Web. La totalidad de los variados componentes de software se comunican directamente por medio de CORBA o DCOM. Por lo que en el paso 5, el componente de software cliente puede estar en un programa arbitrario, con la habilidad de hacer llamadas posteriores a otras máquinas, incluyendo motores de base de datos, como en el paso 6. En el paso 7, el motor de base de datos regresa datos al componente de software del servidor y en el paso 8, los datos se transfieren al navegador cliente.

Estos pasos se pueden repetir a solicitud del usuario final tantas veces como se desee en el contexto de la página HTML original. El usuario no piensa en la pantalla como una página HTML, sino como una simple interfaz gráfica que está controlando una sesión coherente. Esta habilidad de tener una sola página actuando como una interfaz de usuario coherente con muchas respuestas posibles para el usuario y tener esa sola página para completar la sesión de usuario es un gran avance en interactividad usando el modelo de navegador Web. El modelo original orientado a páginas estáticas para navegar en Web antes de CORBA o DCOM no soportaría la noción de una sesión interactiva.

El protocolo básico HTTP permanece como un protocolo sin estado. En otras palabras, el HTTP por sí mismo no proporciona soporte para una sesión de usuario. Para proporcionar una sesión coherente y continua el servidor Web debe depender de la información de sesión proporcionada por el navegador cliente cada vez que se realiza una petición entre máquinas. Cada aplicación de servidor Web requiere de su propia información para mantener una sesión coherente y así el servidor requiere que el navegador cliente ensamble y mantenga esta información de tal manera que pueda ser revelada cada vez que el cliente y servidor se comuniquen. Esta pieza de información se llama *cookie* o *galleta* en Español. Las cookies son pequeños archivos de texto que contienen información de interés para el servidor Web que son almacenados en folders especiales en la máquina cliente. Las galletas o cookies son tema de controversia ya que potencialmente pueden revelar información a un servidor Web que esté haciendo peticiones y que el usuario pudiera no sentirse conforme al divulgarla.

Estos tópicos de seguridad se discuten en la siguiente sección.

Al tiempo que el mercado desarrolla más soluciones basadas en las arquitecturas de componente de software, el equipo de data warehouse debe lidiar con la serie confusa de nuevos productos y terminología conflictiva.

Se recomienda que en cualquier nueva adquisición de software se eviten productos basados en la generación de HTML estático o de aquellos que usan la interfaz de Puente Común (en inglés: Common Interface Gateway), cuyas iniciales son: CGI, o sus derivados, ISAP y NSAPI. El CGI en particular es el enfoque de primera generación para proporcionar páginas Web interactivas y tiene severos problemas. El CGI como HTTP por sí mismo no tiene estados. Cada petición enviada desde el navegador cliente a una aplicación CGI,

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

provoca que el CGI cargue y ejecute una instancia fresca de la aplicación. Esto provoca que el desempeño se vaya degradando en un ambiente extenso de data warehouse.

1.7.1 SEGURIDAD: PUNTOS VULNERABLES

La transformación de los ambientes de cómputo en las empresas hacia ser completamente distribuidos, sistemas en red cliente/servidor, junto con la llegada de la Internet ha cambiado dramáticamente la forma de "jugar" de los dueños de data warehouse. Para la industria de este tipo es como estar sentado en una bomba de tiempo. Nuestros datos están ampliamente expuestos a riesgos. Nos enfrentamos a retos legales y profesionales significativos para entender nuestra vulnerabilidad y responder a ella efectivamente. En esta sección trataremos de enmarcar el problema de la seguridad y en la siguiente sugerimos como entender y manejar la seguridad en un ambiente de data warehouse.

El data warehouse representa la oportunidad para quienes participan en él, para aprender acerca de la empresa y sus clientes.

El tamaño total del problema de seguridad es sorprendente y subestimado. En Octubre de 1996, Ernst and Young establecieron en *Information Week* que "el 78% de los jefes de seguridad de información, directores de seguridad de información y otros ejecutivos de alto nivel reportaron que sus compañías habían perdido dinero debido a delitos de seguridad relacionados con la tecnología de información. Más del 25% reportó pérdidas mayores a veinticinco mil dólares y los infractores internos estaban presentes en el 32% de las pérdidas." En 1997 Aberdeen Group, usando mediciones ligeramente diferentes, reportó que las causas típicas de "información comprometida" incluyeron:

- 35 por ciento: error humano
- 25 por ciento: omisión humana
- 15 por ciento: empleados disgustados
- 10 por ciento: gente externa
- 7 por ciento: incendio
- 5 por ciento: inundación
- 3 por ciento: otros desastres naturales

Bajo estas definiciones podríamos asumir que estamos propensos a un ataque. Obviamente una solución draconiana es no dejar a nadie tener acceso al data warehouse, pero debemos mantener en mente que nuestra labor es publicar y diseminar la información de la empresa. Más adelante en este capítulo trataremos de resolver este conflicto, por lo pronto examinaremos los puntos vulnerables. Para cada tipo de vulnerabilidad invitamos al lector a imaginar como las actividades de una data warehouse podrían impactar específicamente en el lugar donde exista esta condición y como un administrador responsable de data warehouse podría actuar para reducir o eliminar la vulnerabilidad.

RECURSOS FISICOS

Los recursos físicos incluyen los mainframes, PCs y laptops. Los recursos físicos también incluyen a los equipos de comunicación, cableado y fibra óptica, edificios, archiveros y dispositivos off-line. Los principales puntos vulnerables de los recursos físicos se muestran en la tabla 1.2

Puntos vulnerables	Descripción básica	Como puede estar involucrado el Data Warehouse
Robo	La pérdida de un bien físico debido a que alguien simplemente se lo lleva sin permiso	Una PC puede contener resultados de información del data warehouse; una PC puede tener

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

		privilegios de acceso o claves de acceso inter construidas para el data warehouse; una PC puede contener contraseñas de encriptamiento no almacenadas en ningún otro lado; un bien puede contener metadatos necesarios para las operaciones del data warehouse. La pérdida por robo conlleva la responsabilidad que el contenido de la información ha caído en manos hostiles, con lo que se requiere el cambio de los métodos de acceso, de procedimientos y de personal.
Destrucción intencional	Un ataque deliberado sobre un bien físico. Existen una infinidad de formas por las cuales un bien físico puede ser destruido por un determinado individuo, incluyendo los golpes con un instrumento pesado, incendiarlo, vaciarle algún líquido, perforarlo con un destornillador, dejarlo caer y muchas otras.	Pérdida de datos del data warehouse; pérdida de contraseñas, pérdida de metadatos que soporten la operación, pérdida de respaldos, sin posibilidades de recuperar la información.
Incendio	Fuego accidental o deliberado, que afecta al sistema de información directamente o como parte de un incendio mayor. Esta categoría incluye daño por humo y daño por agua debido al combate contra el incendio.	La misma que la destrucción intencional, pudiendo ser más ampliamente destructivo, afectando simultáneamente a los sistemas primarios y de respaldo.
Humedad	Degradación del bien físico debido a un almacenamiento inadecuado.	Imposibilidad de recuperar versiones anteriores de los datos, la pérdida no se nota hasta que es demasiado tarde para proteger los datos.
Agua	Destrucción del bien debido a inundación, haberse mojado o haberse caído al agua.	Muchos dispositivos portátiles de alta tecnología, como PCs portátiles, pierden su funcionalidad cuando se caen al agua. Esto es menos dañino que el robo ya que la información se pierde pero no se compromete.
Suciedad	Destrucción de un bien debido a almacenamiento o manejo inadecuado. La suciedad se infiltra en sistemas de cómputo frágiles. Se hace corto circuito y las cabezas de los discos fallan.	Las mismas consideraciones que la humedad, además de la pérdida de tiempo y dinero tratando de usar y reparar sistemas dañados.
Caducidad	Destrucción del bien debido a que ya no se usa, se vuelve frágil o gradualmente se altera químicamente. Tal vez fue dejado en el sol. Tal vez fue manipulado bruscamente.	Las mismas consideraciones que la humedad.

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

Descarga eléctrica e interferencia eléctrica	Dstrucción de un bien debido a que no está protegido en contra de descargas, picos o apagones. También se incluyen los efectos de los rayos.	Pérdida del bien actual en línea del data warehouse como la máquina cliente o un servidor y tal vez el bien que está en línea posee datos que no han sido respaldados.
Magnetismo	Dstrucción del bien al exponerlo a un campo magnético, como una gran bocina o un sistema de seguridad.	Las unidades de disco y otros dispositivos magnéticos que se ven comprometidos ante un campo magnético destructivo usualmente se pierden completamente y no se pueden recuperar.
Pérdida por olvido	Un bien se descubre que no se puede utilizar debido a que "las llaves para accederlo se han perdido"	Nadie puede recordar cual es el significado de las cintas
Pérdida debida a obsolescenci a tecnológica	Un recurso se vuelve inútil debido a que ya no hay lectora de tarjetas perforadas, lectora de cinta de papel, o dispositivo de diskette de 8 pulgadas que funcionen. La obsolescencia tecnológica se puede dar en cualquier lugar dentro del sistema: en el hardware, en el software o en los procedimientos. La obsolescencia tecnológica se puede dar por que ya no haya más partes de reemplazo para arreglar un recurso.	Aunque los dispositivos de almacenamiento hayan sido cuidados apropiadamente se pueden perder si no se mantienen actuales los sistemas de hardware, los sistemas de software y los procedimientos que rodean su uso. Esta clase de pérdida es un riesgo enorme para el dueño del data warehouse, ya que los datos antiguos que se creían disponibles ya no lo sean.

Tabla 1.2 Vulnerabilidad de los recursos físicos

RECURSOS DE INFORMACION: DATOS, RECURSOS FINANCIEROS Y REPUTACION

Los recursos de información y datos incluyen casi cualquier cosa de valor que no es ni un bien físico ni es software que obtenemos de externos. Estos recursos incluyen todas las categorías de metadatos así como los datos "reales" en los documentos, hojas de cálculo, correo electrónico, sistemas gráficos y bases de datos de todos tipos. Nuestros datos e información incluyen todas las formas derivadas de información electrónica, incluyendo impresiones, fotocopias y la información que la gente lleva en sus cabezas.

Aunque en muchos aspectos nuestros recursos financieros parecen tomar la forma de información, tales recursos tienen un obvio y especial significado debido a que por lo regular son flujo y muchas veces son anónimos. Se puede pensar en los recursos financieros como en recursos de información que pueden ser removidos físicamente de nosotros, no solamente copiados.

Finalmente existe una clase especial de recurso de información que tiene un aspecto cualitativo: nuestra reputación. Nuestra reputación es lo que la gente piensa de nosotros y que tanta buena voluntad tenemos. En el más profundo sentido nuestra reputación es nuestra identidad. Nuestra reputación puede ser destruida, dañada y aún robada justo como cualquier otra forma de recurso.

Existen muchos puntos vulnerables de esta clase de recursos, a los cuales llamaremos *recursos de información* en lo consecutivo. En la 1.3 tabla que se presenta a continuación se muestran los puntos vulnerables que puede tener este tipo de información.

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

<i>Puntos vulnerables</i>	<i>Descripción básica</i>	<i>Como se puede involucrar el Data Warehouse</i>
Descubrimiento de planes confidenciales	Descubrimiento de los planes confidenciales o decisiones tentativas	El data warehouse es una ubicación conocida de planes confidenciales como presupuestos, proyecciones y otros análisis y por lo tanto será el objetivo de cualquiera que trate de encontrar esos planes.
Descubrimiento de códigos	Descubrimiento de los códigos propios o números de cuenta o contraseñas que pueden llevar a robos posteriores o acciones de quebranto.	Existen muchos lugares en el data warehouse en donde la seguridad y los códigos de acceso se introducen o almacenan. Muchas formas de metadatos se almacenan en el data warehouse y pueden contener códigos explícitos. Muchas tablas primarias de datos para el data warehouse contienen información delicada acerca de las personas como números de tarjetas o nombres de familiares que son tipos de códigos.
Descubrimiento de información mantenida confidencialmente	Descubrimiento de información mantenida confidencialmente de algún tercero, como números de tarjeta de crédito, RFC o información de negocios perteneciente a un tercero.	Esta clase de exposición de la información es especialmente mala no solo porque el dueño de la información puede ser dañado directamente, sino que quien mantiene la información puede ser probable responsable de la pérdida o tal vez de daños adicionales. También se puede perder la relación con el tercero. Por si no fuera suficiente, si uno es el empleado responsable de permitir que esta información sea sustraída muy probablemente perdería el trabajo.
Descubrimiento de información delicada	Descubrimiento de información delicada política, ética o legalmente hablando.	El data warehouse puede estar en la posición de guardar datos que realmente no deberían estar almacenados o no debería mantener por más tiempo. Si el data warehouse contiene datos política, ética o legalmente delicados, puede ser el blanco de acciones legales.

Tabla 1.3 Vulnerabilidad de la información

Existen muchas formas de que se lleve a cabo el robo en un sistema abierto, de información distribuida. En la tabla 1.4 que se presenta a continuación se describen varias formas de robo.

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

<i>Modo de robo</i>	<i>Descripción básica</i>	<i>Como el data warehouse puede bajar la guardia</i>
Transmisión inadvertida	La información se dejó accidentalmente abierta. Tal vez alguien olvidó encriptar el archivo o definir sus permisos de acceso. Tal vez la información inadvertidamente se envió por correo electrónico.	Tal vez hay un empleado descuidado o que no entiende como utilizar el sistema. Tal vez nadie ha verificado si los archivos compartidos contienen información delicada o quien exactamente a accedido a los archivos delicados. Estos lapsus son fallas en la aplicación de los procedimientos causados por usuarios o administradores que no están pensando en la seguridad.
Robo fisico como medio para robar información	El ladrón saca la computadora, la laptop o las cintas de respaldo para usarlas y ganar información. La información puede residir en el bien fisico substraído o puede incluir la validación para tener acceso a otro lugar.	El administrador de data warehouse debe involucrarse en la seguridad fisica de laptops de los empleados con acceso significativo al data warehouse.
Impostor	El ladrón pretende ser alguien más por lo que es validado por el sistema. El ladrón puede lograr esto allegándose de los registros de seguridad. Si estos registros son simplemente el nombre de texto y la contraseña de texto del usuario, entonces el impostor puede ser pasar ridículamente fácil. Las contraseñas son notorias por que no proporcionan una seguridad efectiva en el mundo real. El robo por medio de un impostor en el ambiente de redes puede no requerir un usuario y contraseña sino solamente imitar o copiar la dirección de una máquina como la IP en el protocolo TCP/IP utilizado entre las máquinas en Internet.	El administrador de data warehouse no debe ignorar los tópicos serios de verificación y control de acceso. Aún los sistemas estándar basados en contraseñas pueden ser suficientemente seguros si la administración y los empleados utilizan el sistema de contraseñas correctamente. Pero ya que la mayoría de las organizaciones dejan de lado el deseo de forzar una robusta administración y uso de contraseñas, el administrador de data warehouse debería buscar el reemplazo de contraseñas convencionales con fichas de acceso como tarjetas inteligentes o sistemas de retroalimentación biométrica.
Soborno, Robo y Extorsión	Todas estas técnicas son manipulación directa de una persona o empleado de confianza quien le otorga acceso al ladrón o le pasa información o roba alguna.	El data warehouse puede ser capaz de diagnosticar o interceptar modos inusuales de acceso en el sistema de data warehouse. Mayormente los que se necesita es un sistema de monitoreo que reporte patrones inusuales y registros de todos los accesos a ciertos datos.

Tabla 1.4 Tipos de robo

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS
HABILIDAD DE QUE EL NEGOCIO CONTINUE EN FUNCIONAMIENTO

En otros casos, el ataque significa afectar la habilidad de dar servicio o seguir con la misión del negocio, estos puntos vulnerables se muestran en la tabla 1.5 que se presenta a continuación:

Puntos vulnerables	Descripción Básica	Como se ve envuelto el Data Warehouse
Rechazo del servicio por ataque	El sistema de información está comprometido específicamente por el propósito de detenerse o por un cuello de botella en uno o más componentes. Por ejemplo, los servidores pueden ser bombardeados con miles o millones de peticiones falsas.	Los ataques externos desde la Web afectarán muy probablemente al firewall y al Web Server. Si la firewall es tirada, los usuarios externos y legítimos no podrán acceder al data warehouse. Si el Web Server es tirado, a todos los usuarios del data warehouse, tanto dentro como fuera de la firewall, se les puede rechazar el servicio. Una base de datos puede ser el blanco directo de un ataque para rechazar el servicio en la forma de muchas consultas amplias o ciertas consultas que son conocidas por causar que el manejador de la base de datos se quede colgado. Los usuarios con acceso directo de tipo SQL al manejador de base de datos pueden bloquearlo muy fácilmente aunque no sea su intención.
Falta de habilidad para reconstruir el software a un estado consistente	A través de sabotaje deliberado, pérdida accidental o simple descuido, puede ser incapaz de reconstruir a un estado consistente los sistemas.	La pérdida de un estado consistente de los datos de negocio del data warehouse tiene muchas consecuencias serias de tipo financiero y legal. La pérdida de un estado consistente de los meta datos puede significar que no se pueda recuperar un conjunto de datos o que se pierdan las reglas del negocio que se hayan estado aplicando a las transformaciones.

Tabla 1.5 Vulnerabilidad de que el negocio continúe funcionando

1.7.2 SOLUCIONES A LOS PROBLEMAS DE SEGURIDAD

En la sección anterior se presentaron tantos puntos vulnerables que una sola persona no puede pensar en todos ellos al mismo tiempo. Los problemas parecen tan complejos que es difícil saber donde empezar. En esta sección se describen las tecnologías de seguridad más desde la perspectiva de un administrador que de un técnico en la materia.

RUTEADORES Y FIREWALLS

En cada lugar en que una red se conecta con otra debemos tener un dispositivo conocido como *ruteador* que transmite paquetes entre las dos redes. El ruteador escucha a cada paquete de TCP/IP en cada red y busca a la dirección de destino. Si un paquete en una red es enviado a una computadora en la otra red, entonces el trabajo del ruteador es pasarle ese paquete. Si no hay nada más los ruteadores hacen un trabajo útil aislando el tráfico local en cada red. Si esto no pasara habría demasiado ruido innecesario en las redes

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

remotas. Los ruteadores también saben que hacer si ven un paquete con dirección a una red lejana, aunque si no saben dónde está la red lejana, saben donde se ubica un ruteador con mayor visibilidad y pasan el paquete a través, con la suposición de que eventualmente algún ruteador a algunos saltos de distancia sabrá exactamente como entregar el paquete. Los ruteadores son muy listos ya que se comunican entre ellos frecuentemente y se adaptan cuando un nuevo ruteador se inserta en la red. La "comunidad" de ruteadores es muy buena al buscar la ruta más corta hacia un destino. Esta adaptación a las rutas es una fuente significativa de problemas de seguridad. Si un empleado en una compañía conecta su PC que está en la red interna a Internet a través de una conexión telefónica, es muy posible que la comunidad de ruteadores descubra esta conexión y comience a transmitir comunicaciones desde la PC del empleado a través de Internet a otra ubicación, aún dentro de la misma compañía, si el número de saltos sobre la conexión de Internet es menor que el número de saltos a través de las conexiones seguras de la Intranet de la compañía. Esta es una de las razones por las cuales debería prohibirse la conexión por medio de módem desde dentro de un ambiente seguro de intranet.

Obviamente debido a que el ruteador busca en cualquier dirección de destino en cada paquete, también puede buscar en la dirección origen. De esta manera un ruteador puede servir como un *filtro de paquetes* como se muestra en la figura 1.10 que se presenta a continuación.

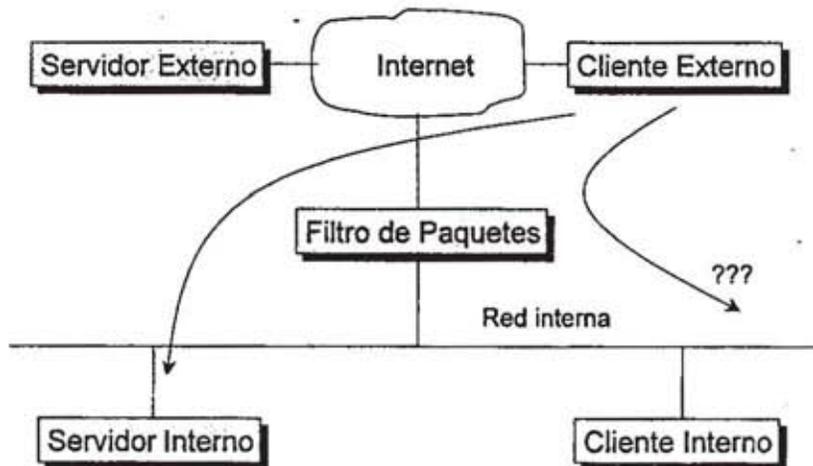


Figura 1.10 Filtro de paquetes

Un ruteador que filtra paquetes también es conocido como firewall filtrador de paquetes y puede proveer las siguientes funciones:

- Rechazar intentos de conexión desde servidores desconocidos. El firewall filtrador de paquetes solo les permite pasar a direcciones IP de origen conocidas. Aunque esto no detiene la infiltración profesional, si detiene a cualquier otro servidor no autorizado. El administrador del data warehouse puede definir un filtro de paquetes para aceptar

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

direcciones IP creadas dinámicamente que representen usuarios remotos legítimos. La utilización de direcciones dinámicas puede impedir la mayoría de los intentos de gente de afuera que quiera infiltrarse como si fueran usuarios legítimos.

- Rechazar los intentos de gente de fuera que quiere infiltrarse como gente interna. El firewall filtrador de paquetes de la figura 1.10 es la única posición capaz de reconocer a un intruso externo que trata de infiltrarse como un usuario interno, debido a que el intruso externo no puede entrar legítimamente con una dirección IP de dentro. Si el firewall dejara pasar a esta persona de fuera, entonces tendría un estado especial en la red protegida. Esta es otra forma de protección en contra de personas ajenas que tratan de infiltrarse en el data warehouse.
- Aislar todo el tráfico que venga de una sola máquina. Un firewall que filtre paquetes puede servir como una especie de guardián para un pequeño número de máquinas protegidas tales como servidores de base de datos, los cuales son accedidos solo a través de servidores de aplicaciones específicos en una red que es menos segura. Esta es una configuración clave recomendada para el data warehouse.
- Detener la contaminación y el husmeo. El firewall que filtra paquetes puede evitar que una red insegura e inestable interfiera con las intranets más seguras mediante la restricción del tráfico externo originado desde la red insegura o inestable. Adicionalmente la presencia de un firewall filtrador de paquetes impide los intentos de paquetes que quieran husmear en una red más segura desde la red pública. El administrador del data warehouse pudiera querer habilitar ciertas terminales y ciertos sistemas de bases de datos sobre redes aisladas de tal forma que los contratistas, consultores y socios industriales puedan hacer su trabajo, pero no pueden acceder, husmear o de otra manera detectar actividad de red en otra parte de la organización.
- Monitorear y auditar los patrones de acceso a páginas Web. El cuello de botella al acceso a la Web creado por el firewall permite que todo acceso a la Web sea monitoreado y registrado.
- Restringir el acceso a Web.

ENCRIPCIÓN

La encriptación es, en el más amplio sentido de la palabra, alterar un mensaje con un código secreto para que solo pueda ser leído por las personas que conocen el código. De alguna forma eso es todo lo que el administrador de data warehouse tiene que saber acerca de la encriptación. Los administradores de data warehouse necesitan entender los usos de media docena de formas de encriptar. Ningún esquema de encriptación resolverá todos los problemas de seguridad.

ENCRIPCIÓN DE LLAVE SIMÉTRICA

La forma más simple de encriptación es la *encriptación de llave simétrica*. Esta clase de encriptación también es conocida como encriptación de llave privada. Con este tipo de encriptación una sola llave secreta puede tanto cerrar como abrir los datos. La encriptación por llave simétrica es interesante principalmente en dos situaciones:

- Encriptación privada de datos. El usuario encripta datos con una llave privada que no revela a nadie más. Si la llave es buena, es esencialmente imposible para nadie más desencriptar los datos. Obviamente en esta situación si el usuario olvida la llave los datos se pierden. La encriptación privada de datos puede tener muchos usuarios en el data warehouse. Cualquier dato que sea especialmente delicado puede ser encriptado. La encriptación privada de datos funciona mejor para la tabla de hechos numérica que no haya sido restringida. Los datos encriptados de una tabla de hechos necesitarán ser desencriptados por el motor de la base de datos cuando los números lleguen para ser sumados.

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

- Encriptación compartida de datos. Si dos partes conocen la llave secreta y nadie más la conoce entonces las dos partes tienen el medio de comunicación muy seguro. Esta clase de secreto compartido es la base de muchas formas de comunicación segura. El gran problema es cómo las dos partes se deciden sobre la llave y la comunicación entre ellos sin dar a nadie más la oportunidad de interceptar la llave. La encriptación simétrica de llave es la base de muchos esquemas de comunicación seguros entre los clientes y servidores del data warehouse y entre usuarios.

La principal técnica de encriptación simétrica que es relevante para los administradores de data warehouse es el algoritmo Estándar de Encriptación de Datos (DES por Data Encryption Standard) que ha sido certificado por el gobierno de los Estados Unidos de Norteamérica y por la Organización Internacional de Estándares (ISO por International Standards Organization). Lo principal que necesitan saber los administradores de data warehouse es que la fortaleza del enfoque DES se basa en la longitud de la llave que el usuario escoge para encriptar.

Encriptación con llave pública

La encriptación con llave pública tiene el problema principal de que tan seguramente se pueden distribuir las llaves. Un conjunto diferente de esquemas de encriptación se ha desarrollado para manejar el problema de dos partes remotas definiendo una liga de comunicaciones segura, aún sobre Internet. Estos esquemas de encriptación son llamados *encriptación con llave pública*. Con este tipo de encriptación hay dos llaves, una llave encripta los datos y la otra desencripta los datos. Las llaves pueden ser utilizadas en cualquier orden. En este esquema una de las llaves se designa como la *llave privada* y la otra es la *llave pública*. Como los nombres lo indican la llave privada es mantenida en secreto por el dueño de los datos y la llave pública es transmitida libre y ampliamente a todo mundo.

Si un individuo posee una llave privada y ha publicado la correspondiente llave pública, entonces son posibles dos escenarios importantes:

- Entrega segura al individuo. Cualquier otra persona en el mundo puede encriptar un mensaje con la llave pública del individuo y entregar al individuo. Ya que solo el individuo tiene la llave privada, el mensaje no puede ser leído por nadie más excepto esa persona. El mensaje en nuestro ejemplo podría ser una descarga de datos delicados desde un servidor de data warehouse a un cliente, tal vez mediante batch. Ver la figura 1.11 que se presenta a continuación.

Encriptado Asegurado

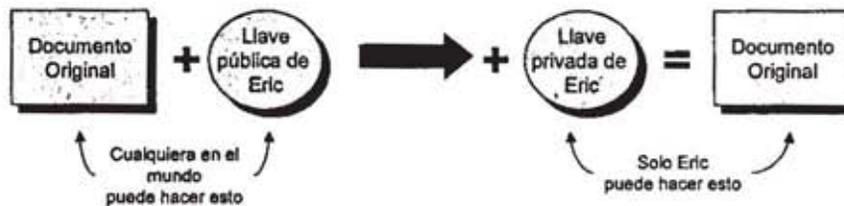


Figura 1.11 Entrega segura al individuo llamado Eric

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

- Autenticidad garantizada del individuo. Si el individuo encripta un mensaje con una llave privada, entonces todo mundo puede verificar que el mensaje es genuino, ya que la única forma de descryptar el mensaje es con la llave única y pública de la persona. Si un usuario necesita cargar números de planeación o de presupuesto al data warehouse, este esquema podría ser utilizado para garantizar que los números de verdad vinieron de parte del individuo designado. Esto se ejemplifica en la figura 1.12 que se muestra a continuación.



Figura 1.12 Autenticidad garantizada de parte del individuo llamado Eric

Si dos individuos tienen llaves públicas y privadas entonces se pueden comprometer en comunicaciones seguras:

- Comunicaciones seguras entre partes. Lalo encripta el mensaje que va de salida con la llave pública de Eric, sin embargo dentro del mensaje Lalo agrega su firma encriptada con su llave privada. Eric recibe el mensaje y descrypta exitosamente el cuerpo del mensaje con su llave privada. Eric entonces verifica la autenticidad de la firma de Lalo descryptándola con la llave pública de Lalo, lo cual se ejemplifica en la figura 1.13 que se muestra a continuación.

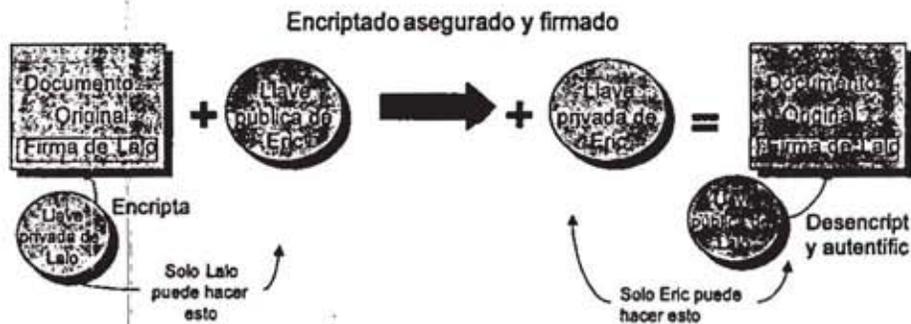


Figura 1.13 Comunicación segura entre participantes

Este esquema es la base de la mayoría de las comunicaciones sobre medios inseguros como Internet o una línea telefónica pública.

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

Autoridades de Certificación Confiable

La forma de asegurar que la llave pública de alguien es realmente la llave pública de esa persona y que no ha sido falsificada es mediante una *autoridad de certificación confiable*. Imaginemos que hay una autoridad impecable que sabemos que es ética y segura. Entendemos y confiamos en los mecanismos de la autoridad para recolectar llaves públicas. Si la autoridad alega que tiene la llave pública del Presidente de la República entonces nosotros confiamos que realmente es la llave pública del Presidente de la República y no de alguien más. Finalmente se nos entrega un CD-ROM o algún otro medio físico que contiene la llave pública y decidimos confiar en esa llave completamente.

Certificados de Software Firmado

El esquema de encriptación de llave pública puede ser utilizado para validar y confiar en software. Básicamente cualquier componente de software viene con un certificado firmado.

- Recepción de un componente de software confiable. El componente de software es acompañado por un mensaje que identifica al creador del software que está encriptado utilizando la llave privada del creador. Uno como consumidor acepta el componente de software si y solo si uno posee la llave pública del creador del software. Si uno no tiene esta llave pública, se puede solicitar a la autoridad de certificación confiable. Después de que se verifica que el componente de software es en verdad de un vendedor de software, uno puede decidir si lo quiere dejar correr en el sistema.

Software de Caja de arena

Un aspecto importante de la seguridad es la confianza que necesitamos tener en los componentes de software descargados y que recibimos de varias fuentes. La implementación del modelo de software certificado firmado no dice nada del software en sí, pero deja la carga enteramente en la confianza que le conferimos al vendedor autenticado. El modelo de desarrollo de Java, en su forma pura, se basa en un modelo muy diferente. El lenguaje Java está definido para operar con la Máquina Virtual Java, que se conoce técnicamente como *caja de arena*. Dentro de la caja de arena de Java, un programa Java obtiene su seguridad al estar drásticamente limitado en su función. Un programa de Java puro no puede leer o escribir archivos en la máquina que le rodea, ni acceder otros dispositivos de hardware, ni acceder la memoria directamente. Aunque este enfoque es interesante intelectualmente, propicia dificultades para los desarrolladores de Java ya que deben ser disciplinados y permanecer confinados.

Capa de Sockets Seguros (Secure Sockets Layer SSL)

En 1995 Netscape propuso la Capa de Sockets Seguros como un protocolo estándar en la industria para asegurar la comunicación entre un cliente y un servidor. Actualmente este estándar se usa ampliamente en Internet y en todos grandes navegadores comerciales que lo soportan. SSL utiliza una combinación de encriptación de llave pública y encriptación de llave simétrica. La encriptación de llave pública se utiliza, más típicamente, para que el usuario en la máquina cliente verifique que en verdad se ha establecido una conexión con el servidor. Opcionalmente, el servidor también puede demandar una verificación de la identidad del cliente. Este apretón de manos inicial de seguridad se usa como base para que las dos máquinas estén de acuerdo en la llave compartida de encriptación simétrica, la cual puede ser mantenida por las dos máquinas cada pocos segundos, con el objetivo de mantener el más alto nivel de confidencialidad en la seguridad.

Redes Virtuales Privadas

Una vez que un cliente y un servidor establecen una llave segura de encriptación simétrica, llamada a veces llave de sesión, el cliente puede ser confiable para el servidor como si el cliente estuviera localmente en el servidor. Esta es la base para la conexión de red virtual privada, también llamada túnel encriptado. El esquema original de Microsoft de Protocolo de Túnel Punto-a-Punto (PPTP) y el esquema de la competencia Cisco Layer 2 Forwarding (L2F) se han unido para crear un nuevo estándar llamado Protocolo de Capa 2 de Túnel (L2TP). Es muy probable que L2TP será la base para la mayoría de las implementaciones futuras de redes virtuales privadas.

Formas de Autenticación

Las contraseñas de texto plano escritas por el usuario en los teclados de computadora son una de las más resaltantes debilidades en la seguridad de la red.

Existen algunas alternativas promisorias al uso de las contraseñas tecleadas:

- **Tarjetas inteligentes.** Similares a las tarjetas para acceder a los cajeros automáticos, estas tarjetas las tiene el usuario final y se introducen o deslizan en una ranura al momento de firmarse. Del mismo modo que las tarjetas para cajeros automáticos se le puede solicitar al usuario que introduzca su número de identificación personal (NIP) para impedir que la tarjeta sea robada. Muchas tarjetas inteligentes contienen electrónica sofisticada que puede generar llaves de encriptación para la sesión cada minuto. Este enfoque proporciona protección automática en contra de ataques repetidos y después de que ha ocurrido la desencriptación de la llave de sesión. Debido a que el dispositivo para deslizar la tarjeta puede considerarse como un periférico no costoso, se podría instalar en las computadoras personales que tengan acceso al data warehouse, aunque la opción menos costosa de definir un password para las PCs podría ser más factible.
- **Barrido biométrico de huella digital.** En este caso el usuario presiona su pulgar contra un sensor al momento de firmarse. Aunque esta alternativa no proporciona la elegante generación de llaves de encriptación en cada sesión, el barrido biométrico tiene la ventaja singular de que el ser humano real esté presente con el propósito de ser autenticado. Las huellas digitales no pueden ser hurtadas, son altamente individuales y casi no hay identificaciones positivas que sean falsas. Sin embargo, alrededor del 3 por ciento de la población tiene huellas muy toscas o muy opacas que no pueden reconocidas confiablemente por los scanners biométricos.
- **Barrido biométrico de patrones de retina.** El patrón de la retina humana tiene muchas de las características únicas de las huellas digitales. Los scanners de retina pueden hacerse para trabajar correctamente en una terminal como un cajero automático, aún cuando los usuarios utilicen anteojos o lentes de contacto. Sin embargo, actualmente la tecnología para barrido retinal no es costeable para las computadoras personales.

Cookies

En la sección anterior de componente de software, describimos a las cookies como archivos de texto especiales escritos por el cliente dentro del sistema de archivos del propio cliente, que son divulgados al servidor remoto cada vez que se establece una sesión cliente/servidor. De alguna manera las cookies se requieren en el mundo del HTTP estático para proporcionar identificadores de sesión entre clientes y servidores. Aunque los cookies han obtenido una mala reputación por supuestos abusos en la seguridad y en la privacidad, como un componente de la arquitectura tienen que ser tolerados si queremos tener sesiones interactivas en Internet basadas en HTTP. La respuesta para la cuestión de los cookies no es prohibirlos, sino filtrarlos de tal manera que no revelen información que no

CAPITULO 1. ANTECEDENTES Y CONCEPTOS BASICOS

queremos que sea descubierta y discriminar a los proveedores que permiten que sus cookies contengan demasiada información acerca del ambiente del cliente.

1.7.3 SEGURIDAD EN UN AMBIENTE DATA WAREHOUSE

Ahora que hemos analizado los muchos puntos vulnerables de los sistemas de cómputo en red y que hemos realizado una breve revisión de las tecnologías que se tienen disponibles para fortalecer la seguridad, es necesario analizar cual es el rol del administrador del data warehouse.

En el presente trabajo se ha presentado la idea que el administrador del data warehouse tiene la responsabilidad central de la seguridad del data warehouse y que nadie más va a hacer el trabajo.

Un programa de seguridad efectiva es un *esfuerzo cultural continuo*, no una solución tecnológica de una sola vez. Los retos a la seguridad son tan multifacéticos y tan dinámicos como para atacarlos con una sola solución y esperar que funcione. El administrador del data warehouse debe proteger sus bienes de información con un programa de seguridad comprometido con los siguientes elementos:

- Preparación. La necesidad de información debe ser reforzada continuamente a través de un proceso de educación constante. Tal como la seguridad, las ideas principales sobre seguridad deben estar visibles en cartelones, folletos y en la pantalla.
- Apoyo ejecutivo. La administración ejecutiva debe ser educada acerca de la importancia de la seguridad y los elementos principales de la seguridad. Los ejecutivos deberían trabajar duro para crear un respecto saludable hacia las medidas de seguridad y para crear buenos ejemplos.
- Políticas. La seguridad debe ser implementada a través de un juego completo de políticas bien pensadas que sean visibles, realizables y justas. Las políticas para las computadoras y para la red deben proporcionarse junto con el manual de referencia del empleado y el equipo de recursos humanos debe estar consciente y apreciar la necesidad de seguridad.
- Vigilancia. La seguridad efectiva involucra una vigilancia continua. La seguridad debe ser revisada y renovada constantemente.
- Sospecha. Alguien en el equipo del data warehouse debe adoptar una actitud de sospecha continua. Tal persona debe revisar los registros de intentos de acceso y preguntarse por qué la gente necesita ver los datos. Aunque esta actitud de sospecha no puede dominar la administración de la seguridad, es absolutamente necesario tener disponible esta actitud para actuar como contrapeso para toda la gente que quiera sobrepasar la seguridad.
- Renovación continua. La seguridad debe ser una búsqueda dinámica y continua. Si se vuelve estática, se verá comprometida. Cada pocos meses, se debe revisar el estado de las redes locales y el estado de toda la industria, con la opción de actualizar las medidas de seguridad. Se debe esperar que la tecnología disponible tanto para el administrador de data warehouse como para el intruso cambiará, se adaptará y se volverá más poderosa.

1.7.4 RECOMENDACIONES PARA LA SEGURIDAD

En esta sección se concluye la discusión de la seguridad con un conjunto de recomendaciones muy específicas. El primer grupo de medidas son acciones tácticas que deben realizarse inmediatamente. El segundo grupo es más estratégico, aunque tal vez no exista una línea muy marcada entre los dos grupos.

Medidas Tácticas Inmediatas

- Instalar en todas partes programas que revisen virus. Es necesario instalar buscadores de virus especialmente en las PCs de usuarios finales, así como en los servidores. Hay que renovar frecuentemente los archivos con el catálogo de virus. Hay que asegurarse que las nuevas fuentes de virus (como las macros) son manejadas por los buscadores de virus. Hay que mantenerse al día acerca de las alertas de virus y los cambios en la tecnología de virus.
- Remover las unidades de disquete de todo el ambiente. Si un usuario necesita leer un disquete, debe llevar el disquete con el administrador local del sistema, quien revisará manualmente que no haya virus y que el software que traiga el disquete sea confiable y solo entonces lo montará en una unidad de disquete de un servidor para ser accedido remotamente.
- Remover los módems locales de todo el ambiente. Prohibir de las instalaciones corporativas el uso de módems que marquen telefónicamente hacia fuera desde cualquier PC de usuario, portátil o fija. Es posible permitir el filtrado de paquetes y módems con soporte de proxy en el perímetro de la red que son relativamente seguros y que pueden ser utilizados por usuarios internos para marcar por teléfono.
- Controlar todo el software instalado en las máquinas internas. Inspeccionar y calificar todos los ejecutables. Instalar un programa que en tiempo de ejecución reporte todos los ejecutables en todas las máquinas a una base de datos centralizada para análisis.
- Asignar a los usuarios contraseñas que deben memorizar y usar. Estas contraseñas deben ser robustas y bien construidas combinaciones compuestas de letras mayúsculas y minúsculas junto con números.
- Concentrar todo el acceso a Internet a través de un servidor proxy de Internet. Monitorear y controlar el acceso a sitios remotos. Proveer de guías claras por escrito para el uso apropiado de Internet.
- Instalar un firewall filtrador de paquetes para restringir el acceso desde el exterior hacia direcciones IP conocidas. Instalar un servidor bastión para interceptar todas las peticiones de servicio del exterior excepto peticiones conocidas de servicio desde direcciones IP conocidas, las cuales se consideran como confiables. Aislar al servidor bastión de la red interna con un segundo firewall filtrador de paquetes.
- Remover todos los servicios innecesarios del servidor bastión de tal forma que si es violado, habrá muy poco que el intruso pueda hacer.
- Implementar un programa de educación en seguridad. Sistemáticamente educar a los ejecutivos.
- Implementar un programa para auditar las amenazas a la seguridad, como intentos de accesos y uso inapropiado.
- Implementar un programa de seguimiento de la seguridad que revise regularmente los privilegios de seguridad de los empleados (acerca de la información que pueden ver) así como qué tan expuestos están los recursos de información (quien tiene acceso a los datos). Asegurarse que los dispositivos tanto de on-line como de respaldo estén cubiertos por este análisis.
- Asegurar físicamente todos los servidores y todos los dispositivos de respaldo. Inspeccionar y asegurar todas las facilidades de comunicación y las bóvedas de los cables. Aplicar un recorrido eléctrico a todas las redes y contabilizar todas las tomas de conexiones.

Medidas Estratégicas

Las siguientes medidas son pasos estratégicos que en muchos casos requieren de la tecnología y de un cambio en el comportamiento dentro de la organización:

- Comprometerse a un enfoque de medios de acceso para reemplazar el uso de las contraseñas escritas (por ejemplo con tarjetas inteligentes y barrido biométrico) internamente y hacia el exterior. Incluir a los proveedores y socios tecnológicos.
- Asignar una combinación de llave pública / privada a cada usuario final para usarla como la base para una autenticación segura. Este par de llaves probablemente se puede acoplar al medio de acceso especificado en el párrafo anterior.
- Comprometerse a un enfoque de túnel seguro para acceso remoto de personas confiables.
- Centralizar todo el control de acceso y de autenticación mediante un servidor de directorios basado en el protocolo LDAP. Requerirle a todos los usuarios que se filtren a través del servidor de directorios ya sean usuarios internos o externos. Administrar toda la seguridad desde este punto central. No permitir a nadie el acceso directo a base de datos o al servidor de aplicaciones.
- Exigir que todas las descargas de software estén basadas en certificados firmados. Administrar activamente la lista de los proveedores de software confiable cuyo software aceptaremos.

1.7.5 RESUMEN

El administrador del proyecto de data warehouse debe tomar un rol principal en la batalla por un ambiente seguro. El administrador del proyecto o un especialista dedicado a la seguridad deben iniciar por educar a la organización a ser receptiva acerca de los tópicos de seguridad y entonces asegurar que las medidas como las expuestas en este capítulo se implementen y mantengan. Aunque el administrador del proyecto no será, en muchos casos, quien realmente implemente esas medidas debe sentir la responsabilidad, de otra forma la seguridad terminará por perderse.

Habrà la necesidad de que técnicos especialistas implementen en la práctica cualquier cambio al ambiente, basados en los planes tácticos y estratégicos.

Existen muchos factores culturales para estimar que tanto tiempo tomará establecer un ambiente seguro. Crear la disposición puede tomar varios meses. La implementación real de estas medidas de seguridad puede también tomar un extenso periodo de tiempo. De forma más optimista, se requerirá de poco esfuerzo adicional de parte del equipo de data warehouse si en la organización ya se están tomando en cuenta estas cuestiones.

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

Para entender cabalmente cualquier problema es necesario analizarlo, o desmenuzarlo en tal proporción que sea relativamente sencillo encontrar una solución.

En este capítulo están considerados los puntos para conocer y analizar el problema, teniendo claramente definidos los requerimientos de los usuarios para proponer un modelo de datos dimensional sobre la base de dichos requerimientos.

El primer paso de todo análisis de sistemas consiste en que el analista entienda la razón del análisis que va a iniciar. Generalmente el conocimiento básico puede obtenerse a través de una entrevista preliminar con las personas que han solicitado o autorizado las actividades para resolver un cierto problema.

Las razones principales para emprender un análisis de sistemas son las siguientes:

- **Solución de Problemas**

Puede suceder que el sistema actual no esté funcionando como se esperaba o puede ser, por ejemplo que alguna área de la empresa tenga algún problema de programación, estadísticas o funcionamiento de sucursales que se debe corregir con un sistema nuevo.

- **Nuevas necesidades**

En cualquier organización pueden aparecer nuevas necesidades debido a causas externas o internas, tales como: la expedición de nuevas leyes, reestructuración de la organización, modificaciones del mercado o incluso la venta de la empresa. Independientemente de la causa que origina la nueva necesidad, el análisis de sistemas identificará las modificaciones o adiciones que deben hacerse al sistema de información con el fin de que la organización vea satisfecha dicha necesidad.

- **Cambios en la tecnología**

Otra razón para emprender un análisis de sistemas es que haya un avance en la tecnología, por ejemplo si se ha empezado a usar un equipo lector de código de barras para registrar los pedidos de los clientes, es más probable que haya necesidad de diseñar un nuevo subsistema de interfaz para aprovechar este equipo. Otro ejemplo es que la empresa tenga como uno de sus objetivos de sistemas, en primera instancia, la publicación de páginas en Internet con información general de la empresa y más adelante proporcionar servicio a sus clientes y proveedores mediante páginas elaboradas con código Java, además del análisis que implica la creación de las nuevas páginas, se requiere un cambio en la forma de atacar los problemas en el equipo de analistas y programadores "tradicionales" para manejar la tecnología orientada a objetos.

- **Mejoramiento general de los sistemas.**

Por último, el análisis puede emprenderse con la intención de encontrar el modo de hacer mejor lo que ya se está haciendo. Muchos de los sistemas actualmente en uso fueron diseñados e instalados hace muchos años. En numerosos casos han dejado de ser válidas las razones que indujeron a diseñar los sistemas en una forma determinada, debido a los cambios que se dan conforme pasa el tiempo. Así que los objetivos generales del mejoramiento de los sistemas pueden definirse como reducción de costos, proporcionar mejor servicio a los clientes y obtener información en menor tiempo y con la mayor confiabilidad.

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

En nuestro caso en particular el inicio del análisis del sistema se debe a que los reportes que recibe el área de Banca Electrónica para apoyar la administración los cajeros automáticos de BanCrecer, resuelven solo parcialmente el problema de la falta de información correcta, ya que no se reciben oportunamente y si se requiere algún nuevo reporte el tiempo de respuesta para elaborarlo (dos semanas en promedio) no es aceptable.

2.1 ANALISIS DE LA SITUACION EN SU ESTADO ORIGINAL

En esta sección se describe y analiza la situación en su estado original (previa a la implementación del sistema), junto con una propuesta para solucionar la problemática del usuario.

Debido a que el equipo IBM es el que soporta la operación diaria masiva del banco (transacciones en sucursales, contabilidad, pago de la nómina, etc.) las peticiones de Banca Electrónica y otras áreas del banco, de más reportes o la modificación de los que ya existen, son atendidas con menos prioridad que la operación diaria, la Dirección de Sistemas decidió implementar, con una interfaz "amigable", un Centro de Información de Cajeros Automáticos (CICA) para ser consultado a cualquier hora hábil desde la computadora de cada funcionario del banco interesado en el tema, por lo que en Enero de 1995 se formalizó, como parte de la estrategia tecnológica, la adquisición de diversas herramientas de Informix como el propio manejador de bases de datos para implementar los sistemas de consulta tipo Cliente-Servidor del Grupo y desahogar, en parte, las operaciones del equipo IBM.

Para conocer el comportamiento del uso de los cajeros y decidir su reubicación o el refuerzo de los parámetros de servicio (dotación de más efectivo, monitoreo de las caídas de línea, etc.) el área de Banca Electrónica utiliza reportes que recibe mensualmente de Sistemas y que en promedio suman mil hojas, con estos reportes los usuarios alimentan hojas de trabajo en Excel que sirven para obtener estadísticas.

En el Apéndice C "Descripción de los Reportes Originales" se muestra la definición y los criterios para obtener los 23 reportes originales.

Los reportes se generan a partir de los Transaction Log Files (TLF). La explotación de los datos almacenados en los TLF se realiza al bajar mensualmente de Tandem a cinta los archivos correspondientes y procesarlos en el mainframe IBM del banco de donde se obtienen varios reportes cuyos programas se encuentran en lenguaje Cobol

En la figura 2.1 se muestra un diagrama del registro y explotación de los datos de los TLFs en reportes.

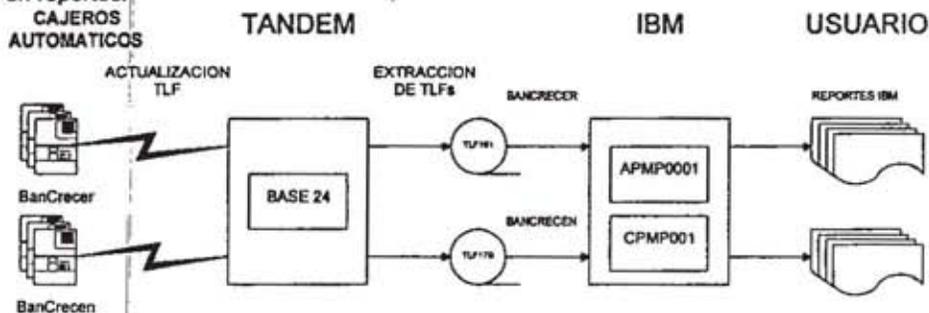


Figura 2.1 Diagrama de la situación en su estado original

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

Los datos de las transacciones realizadas en los cajeros se registran diariamente en Tandem y en IBM se procesan para ser presentados a los usuarios.

2.2 ESTRATEGIA DE SOLUCION

El nuevo sistema debe funcionar tomando los datos que se generan diariamente de las transacciones realizadas en los cajeros de BanCreceer, que se registran en Tandem y pasan a IBM. Los archivos generados en IBM se deben manipular y cargar en un modelo que se pueda presentar a los usuarios mediante la herramienta Informix MetaCube. En la figura 2.2 se presenta el diagrama de cómo se pretende que funcione el nuevo sistema.

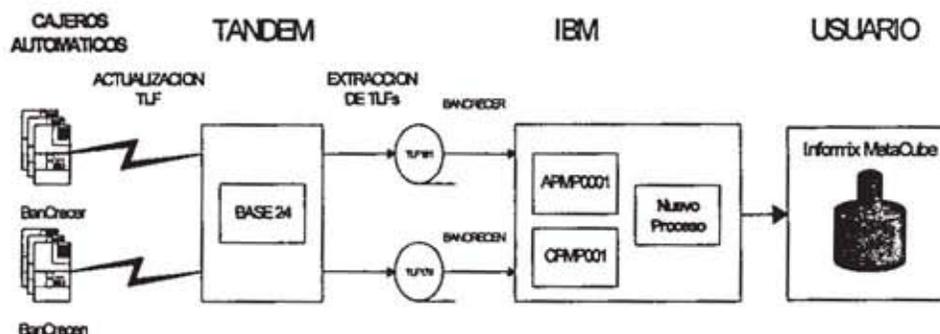


Figura 2.2 Diagrama de la situación esperada

Considerando el ciclo de vida dimensional explicado en el Capítulo 1, la estrategia de solución contempla las siguientes actividades:

1. Identificar a las áreas interesadas e involucradas en el procesamiento de datos estadísticos sobre cajeros automáticos.
2. Identificar los requerimientos de información de las áreas.
3. Identificar las fuentes de información para cumplir con los requerimientos.
4. Definir los alcances del proyecto.
5. Realizar un modelo lógico dimensional que cumpla con los requerimientos del usuario
6. Presentación de prototipo sobre la base del modelo físico dimensional.
7. Realización de ajustes al modelo
8. Definición e implementación de los procesos que alimentan al modelo físico
9. Afinación del ambiente del Data Warehouse

El análisis, desarrollo e implementación de este Centro de Información y, en lo futuro, de los que se requieran dentro del banco, será responsabilidad de la Dirección de Sistemas y de la Subdirección de Desarrollo Departamental.

En la siguiente sección se presenta la metodología para realizar las entrevistas necesarias para implementar una solución a la problemática de los usuarios.

2.2.1 ENTREVISTAS

Para obtener información de primera mano de los requerimientos de los usuarios se pueden realizar entrevistas, cuestionarios y observar directamente cómo se utiliza la información

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

actualmente. Para obtener una descripción de la problemática y las opciones para solucionarla se recurrió a la entrevista como alternativa.

En la tabla que se muestra a continuación está contenida la guía con la que se condujeron las entrevistas con las preguntas básicas aplicadas tanto a un gerente del área usuaria como a un analista del área de Sistemas que conoce los reportes actuales.

A. INTRODUCCIÓN

- Discutir los objetivos del proyecto de Data Warehouse y el estado general.
- Discutir los objetivos y el flujo de la entrevista.
- Presentar al equipo de la entrevista y sus roles.
- Confirmar el tiempo de que se dispone.
- Describir los siguientes pasos de la entrevista.

B. RESPONSABILIDADES

- Describa a su área y su relación con el resto de la compañía.
- ¿Cuáles son sus responsabilidades principales?

C. OBJETIVOS DEL NEGOCIO Y TOPICOS

- ¿Cuáles son los objetivos de su área? ¿Qué logros se ha propuesto? ¿Cuáles son los objetivos principales del negocio?
- ¿Cómo mide su grado de éxito? ¿Cómo sabe que lo está haciendo bien? ¿Qué tan seguido mide los factores clave de éxito?
- ¿Cuáles son los temas principales de negocio a los que se enfrenta hoy? ¿Qué le impide lograr sus objetivos de negocio? ¿Qué impacto tiene en la organización?
- ¿Cómo identifica los problemas / excepciones o como sabe que tiene problemas?
- Describa sus productos (u otras dimensiones claves como clientes, proveedores, etc.) ¿Cómo distingue entre productos? ¿Los productos se catalogan de una forma natural? ¿Cómo delimita una lista de miles de productos?
- ¿Qué tan seguido cambian estas categorizaciones? ¿Qué pasa con los análisis que realiza después de un cambio?

D. ANALISIS DE REQUERIMIENTOS

- ¿Qué tipo de rutinas de análisis realiza normalmente? ¿Qué datos se utilizan? ¿Cómo obtiene los datos actualmente? ¿Qué hace con la información una vez que la obtiene?
- ¿Qué análisis le gustaría realizar? ¿Existen mejoras potenciales a sus métodos o procesos actuales?
- ¿Qué tipo de análisis al vuelo realiza típicamente? ¿Quién solicita análisis a la medida? ¿Qué hacen ellos con estos análisis? ¿Tiene tiempo para contestar las demás preguntas que siguen?
- ¿Qué reportes utiliza normalmente? ¿Qué datos de los reportes son importantes? ¿Cómo utiliza la información? Si el reporte fuera dinámico, ¿Qué tendría que hacer diferente el reporte?
- ¿Qué capacidades analíticas le gustaría tener?
- ¿Existen cuellos de botella específicos para obtener la información?
- ¿Qué tanta información histórica se requiere?
- ¿Qué oportunidades existen de mejorar drásticamente sus actividades sobre la base de un mejor acceso a la información? ¿Cuál es el impacto financiero?

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

E. TERMINO DE LA ENTREVISTA

- Resumen de lo escuchado
- ¿Qué debe cumplir este proyecto para que se considere exitoso? Los criterios deben ser medibles.
- Agradecimiento a los participantes.
- Descripción de los siguientes pasos y de las oportunidades para que el usuario se involucre.

A continuación se mencionan las áreas y las personas que se entrevistaron para iniciar el análisis.

Area: Subdirección de Banca Electrónica

Persona entrevistada: Juan M. Ordaz – Subdirector

Sobre la base de la entrevista y de los reportes que actualmente recibe requiere que los que emita el nuevo sistema contemplen:

- Total de Montos de Disposiciones
- Total de Número de Disposiciones
- Total de Consultas Rechazadas
- Total de Disposiciones Rechazadas
- Total de Disposiciones Exitosas

Todos estos reportes deben emitirse por diferentes criterios:

- Por cuenta de cheques, cuenta maestra y tarjeta de crédito
- Por Cajero
- Por Plaza
- Por Región
- Por Banco

Periodicidad: Mensual.

Area: Sistemas

Persona entrevistada: Blanca L. Rosales - Analista

Esta persona explicó que del TLF es de donde se puede obtener información para el Centro de Información, facilitó documentación, manuales y proporcionó ejemplo prácticos de operaciones con cajeros automáticos para comprender el manejo de los datos.

2.3 REQUERIMIENTOS DEL USUARIO

Derivados de las entrevistas a continuación se relacionan los conceptos de negocio que se deben conocer para elaborar los reportes y las gráficas propuestos para el Centro de Información.

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

2.3.1 CONCEPTOS DE NEGOCIO -

Para comprender mejor los requerimientos del usuario se armó una tabla con los conceptos principales del negocio, que más adelante puede servir como base para un diccionario de datos.

A continuación se presentan estos conceptos básicos de negocio y su definición:

CONCEPTO	DEFINICION
Red	Conjunto de cajeros automáticos pertenecientes a todos los bancos del sistema financiero mexicano que pueden ser utilizados por cualquier tarjetahabiente.
Banco	Institución que administra las inversiones de sus clientes y otorga créditos. Para efectos del CICA un mismo banco puede tener los siguientes roles:
Banco Emisor	Es la institución que emite una tarjeta para utilizar los cajeros automáticos.
Banco Dueño	Es la institución que posee un conjunto de cajeros automáticos.
Tarjetahabientes	Son los clientes de algún banco que tienen una tarjeta de acceso a los cajeros automáticos
Tarjetahabientes propios	Son los clientes del BanCrecer que utilizan los cajeros automáticos asociados a la Red.
Tarjetahabientes de otros bancos	Son los clientes de otros bancos que utilizan los cajeros automáticos de los bancos de BanCrecer
Cajeros	Son las máquinas automáticas que permiten realizar transacciones de tarjetahabientes con los sistemas del banco.
Cajeros propios	Es el conjunto de cajeros propiedad de BanCrecer
Cajeros de otros bancos	Es el conjunto de cajeros propiedad de otros bancos.
Transacción	Toda operación que un cliente pueda realizar en los cajeros automáticos y que se identifica con un número secuencial dentro del TLF.
Transacciones exitosas	Son las operaciones (consulta, retiro, cambio de NIP, emisión del estado de cuenta, etc.) solicitadas por tarjetahabientes en cajeros automáticos y que concluyeron satisfactoriamente para el cliente.
Transacciones rechazadas	Son las operaciones solicitadas por tarjetahabientes en cajeros automáticos y que no le dieron al tarjetahabiente el resultado que esperaba.
Movimientos	Son los tipos de operaciones que el cliente puede realizar en un cajero automático. Pueden clasificarse en exitosos o rechazados.
Consultas	Son las operaciones en las que el cliente obtiene su saldo en pantalla o impreso
Retiros o Disposiciones	Son las operaciones en las que los clientes obtienen dinero del cajero.
Exitosos	Son las transacciones de retiro de efectivo que el cajero automático le otorga al cliente.
Rechazados	Son las transacciones de retiro de efectivo que el cajero automático le niega al cliente. En este caso de que hubo un rechazo es necesario definir los conceptos de reverso parcial y reverso total.
Reverso Parcial	Es cuando el cajero automático le otorga al cliente una cantidad menor a la solicitada y se debe a varias razones, entre otras que

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

	el cajero no tiene suficiente dinero o que el cliente no tiene en su saldo la cantidad solicitada.
Reverso Total	Es cuando el cajero automático no le otorga la cantidad solicitada al cliente por varias razones, entre otras que el cajero ha tenido una falla de comunicación o que el cliente no tiene en su saldo la cantidad solicitada.
Cambio de NIP	Es la operación por medio de la cual el tarjetahabiente cambia el número de identificación de su tarjeta.
Razón de respuesta	Es la explicación del resultado de una transacción, este dato es útil sobre todo para saber el motivo de los rechazos.
Operaciones de débito	Son las que se realizan sobre: <ul style="list-style-type: none"> • Cuentas de cheques • Cuenta Maestra • Tarjeta de Débito
Operaciones de crédito	Son las que se realizan sobre tarjetas de crédito.
Sesión	Un solo periodo en el que el tarjetahabiente realiza varias operaciones: introduce su tarjeta, introduce su NIP, solicita su movimiento y retira su tarjeta.

Teniendo estos conceptos definidos, a continuación se describen los reportes y gráficas requeridos por el área de Banca Electrónica.

2.3.2 REPORTES

El objetivo de los reportes es proporcionar mensualmente datos precisos sobre el comportamiento del uso de los cajeros.

No.	CONTENIDO/DESCRIPCION
1	Montos por disposiciones realizadas en cajeros de todo el banco, mediante cuenta de cheques y cuenta maestra, agrupados por banco emisor de la tarjeta.
2	Montos y frecuencia de disposiciones por cajero y banco emisor de la tarjeta.
3	Montos y frecuencia de disposiciones por región.

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

2.3.3 GRAFICAS

El objetivo de las gráficas es el de plasmar en pantalla o en papel "de un solo vistazo" los conceptos del negocio que sirven para medir el comportamiento del uso de los cajeros sin tener tantos datos numéricos como en los reportes, sobre la base de estos conceptos se deben obtener las siguientes gráficas:

No.	TIPO	CONTENIDO/DESCRIPCION
1	BARRAS	<i>Número de disposiciones exitosas.</i> Se requiere para conocer cuantos retiros fueron realizados con éxito.
2	BARRAS	<i>Número de transacciones exitosas.</i> Para todos los tipos de transacciones se conocerá cuántas fueron exitosas.
3	BARRAS	<i>Montos dispuestos por banco en cajeros propios.</i> Es un comparativo para saber en que banco se realizan más y en cuál menos retiros en cajeros del Grupo.
4	BARRAS	<i>Número de transacciones de tarjetahabientes de otros bancos en cajeros propios.</i> Es un parámetro para conocer que tanto utilizan los tarjetahabientes de otros bancos los cajeros de cada banco del Grupo.
5	BARRAS	<i>Montos dispuestos por tarjetahabientes propios y de otros bancos en cajeros propios.</i> Es una comparación del número de disposiciones realizadas por tarjetahabientes propios y de otros bancos.
6	BARRAS	<i>Montos dispuestos por tarjetahabientes propios en cajeros de otros bancos.</i> Es un parámetro para conocer, por importe, qué tanto retiran los tarjetahabientes propios en cajeros de otros bancos.
7	BARRAS	<i>Montos dispuestos por tarjetahabientes de otros bancos en cajeros propios.</i> Es un parámetro para conocer, por importe, qué tanto retiran los tarjetahabientes de otros bancos en cajeros propios.
8	BARRAS	<i>Comparativo de montos dispuestos por tarjetahabientes de otros bancos y tarjetahabientes propios en cajeros propios.</i> Esta gráfica servirá para conocer si los clientes del Banco retiran más o menos dinero que los de otros bancos.
9	BARRAS	<i>Montos dispuestos en cajeros propios mostrando la media aritmética del monto.</i> Esta gráfica servirá para conocer en que cajeros se retira por arriba o abajo de la media aritmética.
10	BARRAS	<i>Comparativo entre dos meses de montos dispuestos por cajero, promediando cada mes.</i> Esta gráfica servirá para medir el comportamiento de los retiros en los cajeros de un mes al anterior.
11	BARRAS	<i>Comparativo entre dos meses por tipo de transacción de tarjetahabientes propios, totalizando cada mes.</i> Esta gráfica servirá para conocer el comportamiento entre un mes y el anterior de las transacciones realizadas por tarjetahabientes clientes de los bancos del Grupo.

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

12	BARRAS	<i>Comparativo entre dos meses por tipo de transacción de tarjetahabientes de otros bancos, totalizando cada mes.</i> Esta gráfica servirá para conocer el comportamiento mensual de las transacciones realizadas por tarjetahabientes de otros bancos en cajeros del Grupo.
13	BARRAS	<i>Comparativo entre dos meses por consultas y retiros de crédito y débito, totalizando cada mes.</i> Esta gráfica servirá para ver la variación de los retiros realizados por cuentas de crédito y débito entre dos meses.
14	BARRAS	<i>Comparativo entre dos meses por tipo de transacción de tarjetahabientes propios y de otros bancos, totalizando cada mes.</i> Esta gráfica servirá para ver la variación del uso de los cajeros por los clientes propios y de otros bancos entre dos meses.
15	BARRAS	<i>Comparativo entre dos meses por tipo de transacción y su respuesta.</i> Servirá para conocer en que meses hubo más problemas con el acceso a los cajeros.
16	BARRAS	<i>Número de disposiciones rechazadas y su razón de respuesta.</i> Servirá para conocer cuántos retiros se rechazaron y por qué.
17	BARRAS	<i>Número de transacciones rechazadas y su razón de respuesta.</i> Servirá para conocer cuántas transacciones se rechazaron y por qué.
18	BARRAS	<i>Montos dispuestos por mes.</i> Mostrará los importes totales retirados por mes.
19	BARRAS	<i>Comparativo de montos dispuestos por mes entre tarjetahabientes propios y de otros bancos.</i> Esta gráfica mostrará la variación entre dos meses de los retiros de clientes propios y de otros bancos.
20	BARRAS	<i>Comparativo de montos dispuestos por mes entre crédito y débito.</i> Esta gráfica mostrará la variación entre dos meses de los retiros efectuados con cuentas de crédito y débito.
21	BARRAS	<i>Número de transacciones aceptadas por mes.</i> Esta gráfica servirá para medir el número de transacciones aceptadas en periodos mensuales.
22	BARRAS	<i>Comparativo de transacciones aceptadas por mes entre tarjetahabientes propios y de otros bancos.</i> Servirá para conocer a quienes se les aceptan más transacciones entre periodos mensuales.
23	BARRAS	<i>Comparativo de transacciones aceptadas por mes entre crédito y débito.</i>

Cada una de estas gráficas y reportes deberá tener la facilidad de configurarse de acuerdo a los parámetros que se explican en la siguiente sección.

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

2.3.4 PARAMETROS

<p>a) Niveles de agrupamiento (de menor a mayor):</p> <ul style="list-style-type: none"> • Cajero • Sucursal • Plaza • Región • Banco 	<p>d) Formatos de las gráficas:</p> <ul style="list-style-type: none"> • Barras • Pastel • Líneas
<p>b) Periodos de tiempo para consultar los datos:</p> <ul style="list-style-type: none"> • Mensualmente • Entre dos meses, la distancia máxima entre dos meses será de 13 meses, ya que se podrá comparar un mes contra su correspondiente en el año pasado. 	<p>e) Tipos de cuentas</p> <ul style="list-style-type: none"> • Cuentas de cheques • Cuenta Maestra • Tarjeta de Débito • Tarjetas de Crédito <p>Nota: Genéricamente los tres primeros tipos de cuentas son de <i>Débito</i> y el último es de <i>Crédito</i>.</p>
<p>c) Formatos de las gráficas:</p> <ul style="list-style-type: none"> • Barras • Pastel • Líneas 	<p>f) Transacciones:</p> <ul style="list-style-type: none"> • Consultas • Disposiciones • Cambio de NIP • Emisión del Estado de cuenta

2.4 IDENTIFICACION DE LAS FUENTES DE INFORMACION

En el Apéndice A "Descripción de los Campos del TLF" se presenta el layout del archivo Transaction Log File (TLF) de cajeros automáticos, que es la fuente de datos principal para alimentar el Centro de Información de Cajeros Automáticos.

En base al análisis de los campos del TLF, a los requerimientos de reportes y gráficas del Centro de Información y a los conceptos de negocio definidos anteriormente, a continuación se presentan los campos a extraer del TLF para el CICA.

POSICIÓN EN TLF	CLAVE DEL CAMPO	FORMATO	NOMBRE Y DESCRIPCION DEL CAMPO	POSIBLES VALORES
9-10	REC-TYP	X(2)	Tipo de registro.	Se extraerán los registros cuyo tipo sea alguno de los siguientes: 01 = Transacción financiera (operaciones realizadas por clientes). 02 = Registro con transacción administrativa. 20 = Registro con datos inválidos pero la transacción fue enviada (rechazada).
15-18	TERM-LN	X(4)	Red lógica del cajero.	Solo existe una red lógica bancaria: PRO1
19-22	TERM-FIID	X(4)	Banco propietario del cajero.	B161-BanCrecer BCCN-BanCrecen Las claves de los demás bancos comienzan con la letra 'B' excepto las operaciones realizadas en cajeros extranjeros que tienen como

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

POSICION EN TLF	CLAVE DEL CAMPO	FORMATO	NOMBRE Y DESCRIPCION DEL CAMPO	POSIBLES VALORES
				clave la palabra VISA.
23-38	TERM-ID	X(16)	Identificación del cajero.	Para BanCrecer las claves de cajeros inician con la letra 'S' y en total ocupan seis posiciones.
43-46	CRD-FIID	X(4)	Banco emisor de la tarjeta.	Es la misma descripción que para banco propietario del cajero.
47-65	CRD-PAN	X(19)	Número de tarjeta.	Por ejemplo: 4913 6500 9999 9999
79-82	AUTH-TYP	9(4)	Código de tipo de mensaje.	0200-Requerimiento para autorización 0210-Respuesta de autorización 0220-Transacción completa 0420-Transacción incompleta 5400-Ajuste a una transacción completa
111-116	TRAN-DAT	X(8)	Fecha de origen de la transacción.	Cualquier día de operación con formato AAMMDD.
117-124	TRAN-TIM	X(8)	Hora de origen de la transacción.	Cualquier hora de operación con formato AAMMDD.
125-130	POST-DAT	X(6)	Fecha de proceso de corte.	Cualquier día de operación con formato AAMMDD.
143-154	SEQ-NUM	X(12)	Número secuencial asignado a la transacción.	De 1 a 999999999999
181-182	T-CDE	X(2)	Tipo de transacción.	10-Retiro 20-Depósito (No disponible en los cajeros de BanCrecer) 30-Consulta 40-Transferencia 50-Pago con transferencia 51-Pago con sobre (No disponible en cajeros de BanCrecer) 60-Mensaje al banco 70-Emisión de estado de cuenta 81-Cambio de NIP
183-184	T-FROM	X(2)	Tipo de cuenta origen.	00-No hay cuenta origen 01-Cheques 11-Ahorro (cuenta maestra) 31-Tarjeta de crédito
187-205	FROM-ACCT	X(19)	Número de cuenta origen.	Cualquier número asignado por el banco emisor de la tarjeta.
227-234	AMT-1	BINARIO 64 S9(18) comp	Cantidad solicitada.	Cualquier cantidad de hasta 18 dígitos, que el usuario estime conveniente. Pero solamente le puede otorgar \$3,000 al día.
235-242	AMT-2	BINARIO 64 S9(18) comp	Cantidad de balance o ajuste.	Cualquier cantidad de hasta 18 dígitos.
243-250	AMT-3	BINARIO 64 S9(18) comp	Cantidad disponible o saldo.	Cualquier cantidad de hasta 18 dígitos.
257-258	RESP-BYTE-2	X(2)	Razón de respuesta.	Ver Apéndice B.
352-354	ORIG-CRNCY-CDE	9(3)	Código original de moneda.	484-Peso mexicano

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

POSICION EN TLF	CLAVE DEL CAMPO	FORMATO	NOMBRE Y DESCRIPCION DEL CAMPO	POSIBLES VALORES
355-357	AUTH-CRNCY-CDE	9(3)	Código de moneda de autorización.	484-Peso mexicano
358-365	AUTH-CONV-RATE	9(8)	Factor de conversión para el cliente.	Cantidad por la que se multiplicará el saldo del cliente para conversión a moneda nacional. La opción por defecto es 61000000, la primera posición indica el número de decimales, en este caso es igual a seis decimales después del número 1.

Estos campos se integrarán en un archivo por banco que se enviarán comprimidos diariamente mediante FTP a un servidor Unix, en donde se deben realizar los procesos de transformación y carga al modelo de datos dimensional que se explica en la siguiente sección.

2.5 MODELO DE DATOS LOGICO

El modelo de datos lógico representa la forma de analizar los procesos del negocio desde el punto de vista dimensional, sin considerar alguna base de datos o herramienta en especial; es un modelo elaborado considerando todos los aspectos del análisis y diseño dimensional.

En esta sección se analizarán los temas necesarios para definir un modelo de datos dimensional que sea parte primordial de la solución al problema de la información oportuna acerca del comportamiento de los cajeros automáticos de BanCreceer.

Recordando que el proceso de diseño dimensional explicado en el Capítulo 1 contiene los siguientes pasos:

- Elegir un proceso de negocio a modelar
- Elegir la granularidad del proceso de negocio
- Elegir las dimensiones que aplicarán para cada registro en la tabla de hechos
- Elegir las métricas que llenarán cada registro en la tabla de hechos

Debemos considerar que ya está definido el proceso de negocio a modelar que es el del comportamiento de los cajeros automáticos de BanCreceer, a continuación se explica la definición de la granularidad del proceso de negocio.

2.5.1 GRANULARIDAD

Aunque parece un paso técnicamente detallado, es crucial definir muy clara y exactamente como será cada registro de la tabla de hechos en el diseño dimensional que se propone. Sin esta definición el diseño no puede continuar y sin una definición clara, los diseñadores perderán valioso tiempo preguntando acerca de qué es una dimensión y qué es un hecho. Al definir el nivel de granularidad se estarán definiendo las llaves foráneas en la tabla de hechos, con lo que de una forma natural se definen las llaves primarias en las dimensiones, así que en este paso se comienzan a definir implícitamente las dimensiones.

Los datos de la operación de los cajeros automáticos se obtienen diariamente en dos archivos TLF: uno de las operaciones registradas por los cajeros de BanCreceer en México y otro de las operaciones registradas por los cajeros de BanCreceer en Costa Rica. El volumen máximo estimado del TLF de BanCreceer es de 460,000 registros que ocupan

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

100,000,000 bytes; para el TLF de BanCrecen el volumen máximo estimado es de 5,000 registros que ocupan 1,000,000 bytes. Los registros de los TLFs de ambos bancos contienen transacciones de tarjetahabientes (propios y de otros bancos) que utilizaron esos cajeros durante el día. Este conocimiento nos comienza a dar idea de que por el lado del tiempo la granularidad se puede definir por día; para justificar completamente esta apreciación se han realizado pruebas para evaluar la capacidad necesaria del disco para almacenar cada archivo diario del TLF, durante 13 meses, considerando que los archivos se recibirán los 365 días del año. Esto nos da una estimación del espacio requerido para almacenar esos trece meses, como se muestra en la tabla 2.1.

Cajeros	Volumen por día	Volumen por mes	Volumen para trece meses
México	100,000,000	3,000,000,000	39,000,000,000
Costa Rica	1,000,000	30,000,000	390,000,000
TOTAL (bytes)	101,000,000	3,030,000,000	39,390,000,000
TOTAL (Kb)	98,633	2,958,984	38,466,797
TOTAL (Mb)	96	2,890	37,565
TOTAL (Gb)	0.1	2.8	36.7

Tabla 2.1 Estimación de espacio requerido para almacenar el TLF

Se podría tomar el criterio de cargar los datos cada semana o cada mes y se podría perder la identificación de cada día o por otro lado realizar la carga diariamente y conservar cada día bien identificado, en el primer caso se ahorraría espacio y tiempo de proceso pero se perdería la capacidad de examinar en dado caso los datos por día, en el segundo caso la carga se realizaría conforme se genere el archivo permitiendo monitorear el proceso diariamente y corregir los problemas que pudieran ocurrir en la carga. De acuerdo a los requerimientos se debe tener información mensual, pero hemos decidido cargarla diariamente conforme se generen los TLFs y consolidarla quincenalmente, por lo que el nivel de detalle en el tiempo será: primera quincena y segunda quincena de un mes y año. Esta parte del análisis nos va dando la pauta de una dimensión obligada en todo Data Warehouse: Tiempo, que definiremos a detalle en su momento.

El otro aspecto importante a evaluar respecto del tamaño de los archivos a recibir es el tiempo de procesamiento para generarlos y para su carga inicial en el modelo, considerando pruebas en Informix de carga de archivos planos a tablas con el mismo número de campos que los mostrados en la sección 2.4 el tiempo es de 15 minutos con archivos de 30,000 registros que es el promedio de registros generados por día.

Los TLFs que se generan diariamente contienen en promedio 100 transacciones por cajero, las transacciones se realizan en sesiones por cliente, en donde una sesión puede incluir tres o más transacciones, por ejemplo: el tarjetahabiente consulta su saldo, retira cierta cantidad de efectivo y nuevamente consulta su saldo, en este caso el TLF contiene tres registros asociadas a la misma tarjeta, al mismo cajero, pero con dos tipos de movimientos: consulta y retiro, para el modelo de datos tener estos registros sin ninguna agrupación implicaría un nivel de detalle que no se justifica por los requerimientos del usuario, ya que no hay ninguna gráfica o reporte que requiera "el número de consultas de una cuenta" o la fecha y hora de alguna transacción en particular, es más, conservar este tipo de dato dentro del modelo ocuparía más espacio en disco, por lo que el nivel de detalle tal cual se generan los registros no es útil de acuerdo a los requerimientos del usuario.

Aunque la mayor parte de las gráficas se basan en mostrar los retiros no conviene agrupar los movimientos en retiros y otros movimientos, ya que hay algunas gráficas que necesitan tener los otros tipos de movimientos bien discriminados, por lo que se deben identificar los tipos de movimientos. En este punto del análisis podemos vislumbrar una dimensión más: la del tipo de movimiento.

CAPITULO 2. ANÁLISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

De acuerdo con la definición de los reportes, se deben hacer cortes por Banco, cajero o región; como se expuso en la sección 2.3.4 los niveles de agrupamiento están claramente definidos para los cajeros y considerando que el TLF contiene el detalle de cada cajero que tuvo operaciones, el nivel de detalle para este caso es el cajero. En este punto del análisis se vislumbra otra dimensión: la que contiene los cajeros y su agrupación. En este caso estos elementos que agrupan a los cajeros muestran claramente una jerarquía o niveles de agregado; el banco tiene varias regiones, una región se compone de plazas y en una plaza existen cajeros, por lo que la nombraremos como dimensión Geografía. Debido a que el TLF contiene solamente el dato de la clave del cajero será necesario contar con algún catálogo que identifique el nombre del cajero y en que plaza y región se encuentra.

Considerando que hay tres gráficas que deben presentar los datos agrupados en crédito y débito, es necesario considerar el concepto de tipo de cuenta a nivel detallado, es decir la granularidad definida en función del tipo de cuenta debe estar representada por cada tipo de cuenta en especial para no perder ningún nivel de detalle en cuanto a este rubro. De este análisis se desprende que puede existir otra dimensión: Tipo de Cuenta, ya que teniendo el nivel de detalle de cada cuenta, se pueden agrupar los tipos de cuenta de cualquier forma que los usuarios necesiten.

Para tener una imagen concreta del análisis antes expuesto referente a la granularidad, en la figura 2.3 se muestra el primer acercamiento al modelo de datos del CICA, donde se observa que el nivel de granularidad va definiendo automáticamente las dimensiones.

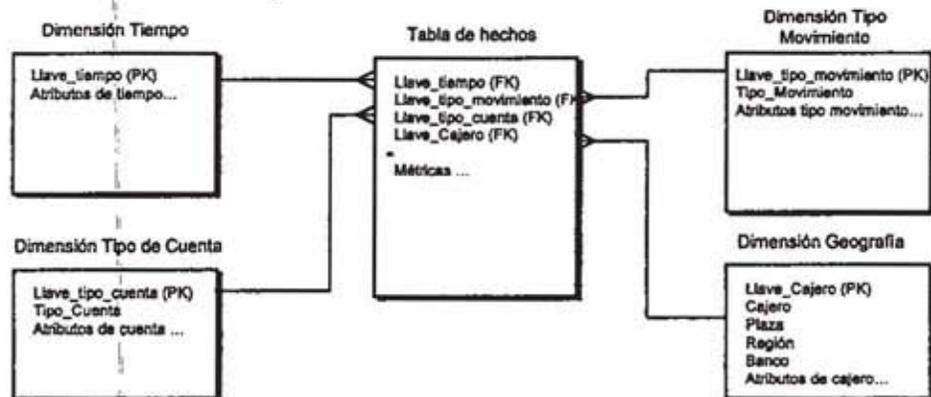


Figura 2.3 Primer acercamiento al modelo de datos

Las ideas expuestas en el Capítulo 1 respecto de la metodología del modelado dimensional y el análisis de los requerimientos de los usuarios van tomando forma en el modelo de arriba; pero, para no olvidar que este modelo estará representado en una base de datos en el mundo real, les toca ahora el turno a las dimensiones para estimar el espacio que ocuparán, para confirmar que proporcionalmente su tamaño no es representativo con el de la tabla de hechos.

Con lo que hemos expuesto del modelo y con los conocimientos de los requerimientos de los usuarios, la dimensión que más registros puede llegar a tener, sin considerar a la de Tiempo, es la de los cajeros, ya que a nivel de cajero existirían a lo más 1,000 registros, a nivel de plaza aproximadamente 100 registros, a nivel región 12 registros y a nivel banco solamente 1 registro. Si se almacenan 10 bytes para la clave de cajero y otros 10 bytes

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

para las claves de plaza, región y banco en cada registro, además 30 bytes de descripción por cada una de las claves, tendríamos en total por cada registro las cantidades que se muestran en la tabla 2.2.

Atributo	Clave (bytes)	Descripción (bytes)
Cajero	10	30
Plaza	10	30
Región	10	30
Banco	10	30
TOTAL (bytes)	40	120
TOTAL GENERAL(bytes)	160	
PARA 1,113 REGISTROS (bytes)	178,080	
PARA 1,113 REGISTROS (Kbytes)	174	
PARA 1,113 REGISTROS (Mbytes)	0.169830	
PARA 1,113 REGISTROS (Gbytes)	0.000166	

Tabla 2.2 Estimación de espacio ocupado por registro

Es necesario considerar que el espacio requerido no se incrementa en un año, por lo que el espacio estimado es válido para el mismo periodo de 13 meses considerado para la tabla de hechos, así que proporcionalmente la dimensión de los cajeros prácticamente no ocupa espacio respecto a la de hechos, incluso desde el nivel de Mbytes.

La forma en que se verificará que la granularidad del modelo cumpla con los requerimientos expuestos en la definición de reportes y gráficas es mediante la tabla 2.3 que se presenta a continuación:

Elementos de granularidad	Reportes			Gráficas																								
	1	2	3	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23		
Tipo de Movimiento	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	
Geografía	X	X	X			X	X	X	X	X	X	X	X															
Tipo de cuenta	X																									X		X
Banco Emisor	X	X	X				X	X	X	X	X			X	X	X										X		X
Tiempo (mes)													X	X	X	X	X					X	X	X	X	X	X	X
Razón de respuesta																		X	X	X								

Tabla 2.3 Verificación de la granularidad

Del análisis de esta tabla se observa que en la figura 2.3 falta como parte de la granularidad el banco emisor de la tarjeta y la razón de respuesta del movimiento solicitado.

Un mismo banco que tiene cajeros normalmente debe emitir tarjetas para sus clientes, por lo que la clave que tiene un banco en la red para identificar sus cajeros es la misma que se utiliza para identificar las tarjetas que emite. Aunque en cada registro del TLF no necesariamente el mismo banco dueño del cajero recibe solo tarjetas emitidas a sus

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

clientes, así que las combinaciones que se pueden encontrar en cada registro del TLF respecto a la relación Banco Dueño/Banco Emisor, son:

Banco Dueño	Banco Emisor
BanCrecer (B161)	BanCrecer (B161)
BanCrecer (B161)	Otros bancos

Ya que en el TLF se reportan las operaciones de los cajeros de cada banco, o sea que BanCrecer no puede ver los cajeros de otros bancos en su TLF, pero si puede ver los tarjetahabientes propios y los de otros bancos que han utilizado sus cajeros, por lo que en la dimensión Geografía se encuentra el banco dueño del cajero.

El caso de la razón de respuesta tiene que ver más con la identificación a nivel de transacción de la explicación de la respuesta registrada en el TLF, pero debido a que no almacenaremos transacciones tal y como llegan del TLF en la tabla de hechos, se deben agrupar los diferentes tipos de razones de respuesta.

En en la figura 2.4 se muestra el modelo con la dimensión Razón de Respuesta, señalando en cual dimensión está contenido el banco dueño del cajero y con la granularidad establecida.

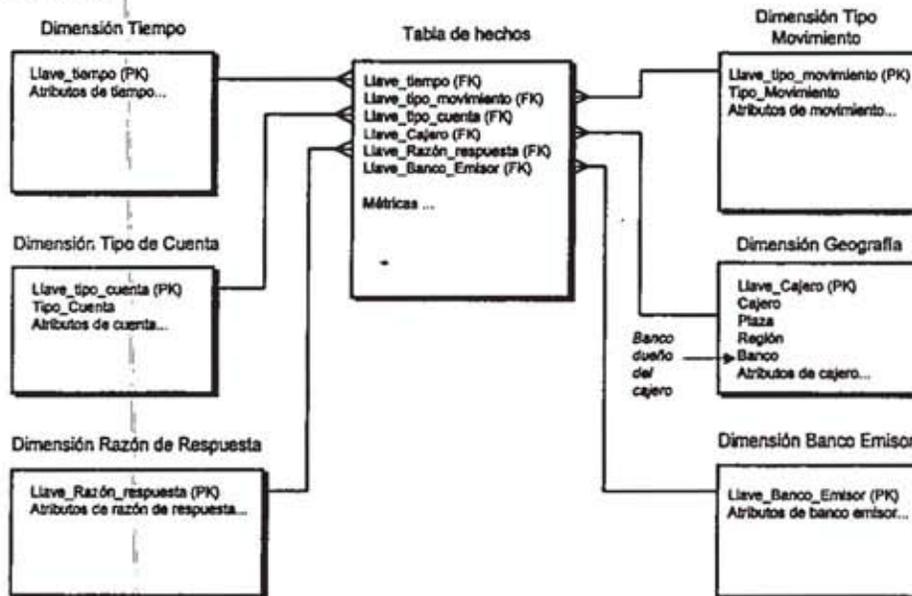


Figura 2.4 Modelo de datos con la dimensión Razón de Respuesta

2.5.2 DIMENSIONES

La función de las dimensiones se puede pensar como la "envoltura" del cubo dimensional o como las caras de un cubo para recortar y pegar que envuelven a la tabla de hechos, como se muestra en la figura 2.5

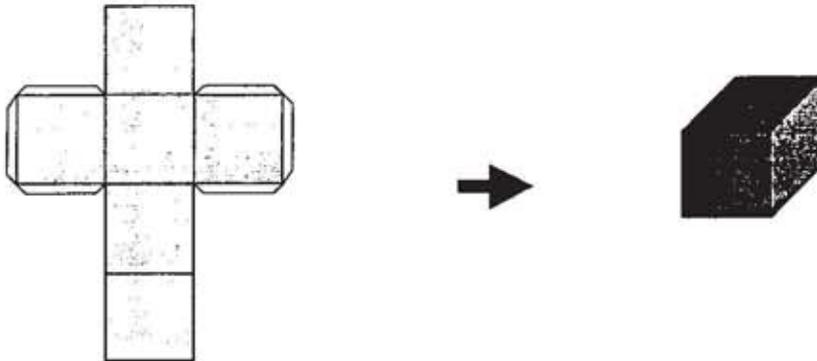


Figura 2.5 Representación de un cubo para recortar y pegar

Una definición clara de la granularidad a menudo facilita la elección de las dimensiones para la tabla de hechos. Frecuentemente la granularidad se define en términos de dimensiones primarias. La frase "Actividad diaria de tarjetahabientes propios y de otros bancos en cajeros automáticos de BanCrecer para medir la rentabilidad" especifica claramente la dimensión tiempo, la dimensión geográfica y la dimensión banco emisor de la tarjeta. Otras dimensiones pueden probarse contra la tabla de hechos para verificar si tienen sentido.

Si una dimensión no corresponde con la granularidad, se tienen dos opciones: no considerar a la dimensión o cambiar la granularidad.

Una vez que se ha escogido una dimensión, puede haber un gran número de atributos descriptivos que se pueden usar para llenar la dimensión. Estos atributos pueden venir de varias fuentes. En este punto del proceso es de ayuda hacer una lista lo más larga posible de todos los atributos descriptivos que estén disponibles para describir un elemento en la dimensión (un producto, un servicio, un cliente, una ubicación geográfica o un día). El objetivo en esta etapa del diseño es ser comprensible y claro. Los detalles de obtener los datos y su calidad se resolverán durante el diseño detallado de la fase de implementación.

2.5.2.1 DIMENSIONES CAMBIANTES LENTAMENTE

Un tema especialmente interesante durante el análisis de las dimensiones es cuando el valor de algunos de sus atributos cambia en el tiempo. Normalmente se pensaría que los datos de una dimensión no cambian, pero si se analiza, por ejemplo, el caso de los clientes, estos cambian constantemente, ya que los humanos se pueden casar, divorciar, tener más hijos, modificar su dirección, tener un mejor ingreso o incluso cambiar de nombre. Como diseñadores de bases de datos debemos decidir que hacer para manejar estos cambios. Como se ha mencionado en el Capítulo 1 referente a que una de las principales responsabilidades del Data Warehouse es representar correctamente el estado de la

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

historia inicial. Por ejemplo, en una compañía de seguros es esencial que la descripción del siniestro sea representada con la hora de la reclamación y no simplemente que sea la descripción actual de la parte asegurada.

La respuesta es no poner todo en la tabla de hechos, no hacer cada dimensión dependiente del tiempo. Hay que explotar el hecho de que la mayoría de las dimensiones son casi *constantes* en el tiempo y que podemos preservar la estructura dimensional independiente con adiciones relativamente pequeñas para capturar la naturaleza cambiante en el tiempo. A estas dimensiones casi constantes las *llamamos dimensiones cambiantes lentamente*.

Cuando encontramos una dimensión cambiante lentamente tenemos que encarar alguna de las tres fundamentales opciones. Cada opción resulta en cierto grado de llevar el registro de los cambios en el tiempo:

- Sobrescribir los valores anteriores en el registro de dimensión y por consiguiente perder la habilidad de rastrear la historia anterior.
- Crear un registro de dimensión adicional al momento del cambio con los nuevos valores de atributo y por lo tanto hacer una segmentación de la historia muy marcada entre la descripción anterior y la actual.
- Crear campos nuevos "actuales" dentro del registro original de dimensión para registrar los nuevos valores de atributos, manteniendo también los valores de atributo originales, por lo que es posible describir la historia tanto hacia delante como hacia atrás de cuando se hace el cambio ya sea en términos de los valores originales de atributo o en términos de los valores actuales de atributo.

Específicamente nombramos a estas tres opciones de dimensiones cambiantes como Tipo Uno, Tipo Dos y Tipo Tres. Para ser más concretos, pensemos que en una dimensión de clientes tenemos un registro que describe a Mari López. Hasta Enero 15 de 1994, Mari no estaba casada, por lo que el campo estado civil en su registro de cliente tenía el valor Soltera. Mari entonces se casó en Enero 15 de 1994. Nos enfocaremos en cómo manejar el campo estado civil e ignorar los posibles cambios en otros campos que pudieron haber tenido lugar incluyendo su nueva dirección.

TIPO UNO

En este caso sobrescribimos el campo estado civil en el registro de Mari López con el valor Casada. No se necesitan más cambios en el registro de dimensión y ninguna llave se afecta en la base de datos. En este caso hemos decidido que no tiene caso mantener la descripción anterior. Podemos determinar esto al comprender las necesidades de análisis del usuario final y que tan importante es rastrear la historia anterior. La ventaja de sobrescribir, obviamente, es que es más fácil de implementar. Desde luego, el uso del Tipo Uno, evita el objetivo real, el cual es registrar de manera precisa la historia. El uso del tipo uno no está siempre mal. Si hubiéramos descubierto que nuestros datos originales acerca de Mari López tenían error y que ella estuvo casada todo el tiempo, entonces casi seguramente hubiéramos sobrescrito el campo de estado civil con el objetivo de corregirlo.

TIPO DOS

Más frecuentemente podríamos decidir que necesitamos dos descripciones correctas de Mari López. La descripción original con el estado civil = Soltera aplicaría a todos los registros de hechos hasta enero 15 de 1994 y una nueva descripción aplicaría para todos los registros de hechos correspondientes a después de enero 15 de 1994. Cumplimos esto al crear un segundo registro en la dimensión de cliente para Mari López con el estado civil = Casada. Este segundo registro de dimensión debe tener una nueva llave de cliente debido a que la llave define la unicidad en la dimensión de cliente. Esto trae como consecuencia un principio de diseño importante:

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

"El uso del tipo dos de dimensión cambiante lentamente requiere que la llave de dimensión sea generalizada. Puede ser suficiente tomar la llave propia de producción y agregar dos o tres dígitos de versión al final de la llave para simplificar el proceso de generación de llaves".

En otras palabras, el número de cliente real de Mari López, que puede ser el de su tarjeta de crédito, no puede seguir siendo usado como una llave literal para la dimensión cliente. Si vamos a tener múltiples copias de registros con la descripción de Mari López como cliente, entonces necesitamos una llave generalizada. Dos dígitos de versión al final de la llave original de cliente permitirían registrar hasta 100 cambios del estado de Mari López, mientras que tres dígitos permitirían registrar 1,000 cambios.

Hay que notar que el sistema de producción probablemente no va a crear o mantener la llave generalizada, esto es típicamente responsabilidad del equipo de Data Warehouse y debe ser implementada en el sistema de extracción de datos.

El uso del tipo dos de dimensión cambiante lentamente inevitablemente particiona la historia. Utilizamos la primera descripción (soltera) de Mari López hasta el 15 de enero de 1994 en todos los registros de la tabla de hechos que sean creados para ella, tal vez en lo concerniente a ventas. Al inicio del 15 de enero de 1994 utilizamos el segundo registro (Casada) de la tabla de dimensión para unirse con la tabla de hechos. Esto resulta en una transición excepcionalmente transparente sobre el evento del estado civil. Si preguntamos a la dimensión cliente por el nombre Mari López, entonces se obtendrían ambos registros de dimensión y se tratarían de unir a la tabla de hechos. El primer registro de Mari López se uniría solamente a los registros de ventas de antes del 15 de enero de 1994. El segundo registro de Mari López se uniría solamente con los registros de ventas a partir del 15 de enero de 1994 y después. No hay una necesidad intrínseca de poner una fecha efectiva para el cambio del registro de cliente de Mari López y no existe ninguna necesidad de restringir a la tabla de dimensión por valores de tiempo con el objeto de obtener la respuesta correcta. Comúnmente este es un punto de confusión en el diseño del Tipo dos de dimensiones cambiantes lentamente. Es por lo tanto otro principio de diseño:

"El tipo dos de dimensión cambiante lentamente automáticamente particiona la historia y una aplicación no será requerida para poner cualquier restricción de tiempo sobre fechas efectivas dentro de la dimensión".

Ya que vimos que el tipo dos de dimensión cambiante en el tiempo particiona la historia, no podremos usar el nuevo valor de un atributo cambiado sobre la historia antigua y viceversa. En el caso de Mari López si restringimos sobre Estado Civil = Casada, no veremos a Mari López antes del 15 de enero de 1994. En la mayoría de los casos esto es lo que queremos. En unos cuantos casos podríamos querer ver como habría sido la historia si hubiera sido del mismo modo todo el tiempo. Esto pasa más frecuentemente cuando se describen cambios en la fuerza de ventas. Hay un punto abrupto en el tiempo donde los nombres de todas las áreas de venta son cambiados. Sin embargo por unos cuantos meses, existe el deseo de rastrear la historia antigua en términos de nuevos nombres de áreas de venta y del mismo modo rastrear la historia nueva en términos de los anteriores nombres de áreas. Normalmente el razonamiento es que se debe ser capaz de comparar el desempeño a lo largo de la transición. Sin embargo esta clase de comparaciones no puede ser hecha con el tipo dos. Aquí es donde se necesita el Tipo tres de dimensión cambiante lentamente. Debido a la complejidad de las aplicaciones del tipo tres trataremos de evitar la implementación de esta solución. Usualmente después de unos pocos meses la gente pierde interés en los anteriores nombres de las áreas de ventas y el requerimiento desaparece.

TIPO TRES

Cuando el valor de una dimensión cambia, la mitad del tiempo se elige una solución de Tipo Uno y la otra mitad se escoge una solución de Tipo Dos.

Ocasionalmente hay una necesidad legítima de rastrear tanto los anteriores como los nuevos valores de un atributo cambiado tanto hacia delante como hacia atrás sobre el tiempo del cambio. Este es el Tipo Tres de dimensión cambiante lentamente. En este caso no creamos un nuevo registro de dimensión, en vez de esto agregamos un nuevo campo para el atributo afectado. En el caso de Mari López agregaríamos un nuevo atributo llamado Estado Civil Actual. Probablemente renombraríamos el campo anterior de estado civil para que fuera el Estado Civil Original. En este caso tiene algo de sentido agregar también un campo de Fecha Efectiva de Estado Civil Actual. Ahora cada vez que cambie el estado civil de Mari López, sobrescribiremos el campo de Estado Civil Actual y cambiaremos la Fecha Efectiva. Siempre dejaremos el campo de Estado Civil Original sin cambio. Ahora podemos rastrear toda la historia que involucra a Mari López utilizando ya sea el campo de Estado Civil Original o el campo de Estado Civil Actual. Debido a que la tabla de hechos "mira" a una sola llave por cliente, la única forma en que podemos particionar la historia basados en los cambios es con el uso del campo de Fecha Efectiva.

El tipo tres de dimensión cambiante en el tiempo está equipado para manejar solamente los valores original y actual del atributo cambiado. Los valores intermedios se pierden. Por supuesto, si existe la necesidad de particionar de manera precisa la historia, entonces se debe utilizar el Tipo Dos y todos los cambios pueden ser rastreados. Teóricamente puede ser posible mezclar los Tipos Dos y Tres pero todo esto resulta en un incremento en la complejidad de las aplicaciones, por lo que no valdría la pena el esfuerzo. Los usuarios finales no se impresionan con la complejidad, más bien la rechazan.

A continuación se presenta el análisis para definir cada dimensión del modelo.

2.5.2.2 DIMENSION TIEMPO

La dimensión tiempo se puede encontrar virtualmente en cualquier Data Warehouse, ya que prácticamente todo Data Warehouse es una serie de tiempo. Esta dimensión es una de las más interesantes de evaluar por la gran variedad de opciones que hay para su definición, ya que en ella se pueden definir campos que contengan datos como día de la semana, número de día del mes, número de semana en el año, bimestre, trimestre, semestre y otros más.

En el caso del CICA la información se requiere mensualmente, pero se ha decidido que el mínimo nivel de detalle sea por quincenas, ya que la capacidad inicial del servidor no era suficiente para terminar de procesar un día y presentarlo al siguiente.

Teniendo en mente que la información se presentará quincenalmente, el primer paso es decidir qué tipo de llave es conveniente para identificar cada quincena, había dos alternativas principales:

- Identificar cada quincena como la concatenación del año, mes y número de quincena del mes, con lo que se asegura que será única la llave de la dimensión tiempo. En la tabla que se presenta a continuación se explica como se construiría la clave considerando que la fecha inicial sea la primera quincena de enero de 1999 y la fecha final sería la segunda quincena de diciembre de 2000. Esto se muestra en la tabla 2.4.

Año	Mes	Quincena	Llave resultante
1999	01	1	1999011
1999	01	2	1999012
...

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

1999	12	1	1999121
1999	12	2	1999122
2000	01	1	2000011
...
2000	11	1	2000111
2000	11	2	2000112
2000	12	1	2000121
2000	12	2	2000122

Tabla 2.4 Primera propuesta de llave para la dimensión tiempo

- Identificar cada quincena como un número Juliano, o sea un número consecutivo de día iniciando por una fecha en particular. En la tabla que se presenta a continuación se explica como se construiría la clave considerando que la fecha inicial sea la primera quincena de enero de 1999 y la fecha final sería la segunda quincena de diciembre de 2000. Esto se muestra en la tabla 2.5.

Año	Mes	Quincena	Llave resultante
1999	01	1	1
1999	01	2	2
...
1999	12	1	23
1999	12	1	24
2000	01	1	25
...
2000	11	1	45
2000	11	2	46
2000	12	1	47
2000	12	2	48

Tabla 2.5 Segunda propuesta de llave para la dimensión tiempo

Se considera la primera opción como la mejor debido a que se puede identificar más fácilmente de qué quincena se trata.

En la tabla 2.6 se muestra como se describiría cada quincena de la dimensión tiempo.

Año	Mes	Quincena	Llave resultante	Descripción de la quincena
1999	01	1	1999011	Primera Quincena de Enero de 1999
1999	01	2	1999012	Segunda Quincena de Enero de 1999
1999	02	1	1999021	Primera Quincena de Febrero de 1999
...
2000	11	1	2000111	Primera Quincena de Noviembre de 2000
2000	11	2	2000112	Segunda Quincena de Noviembre de 2000
2000	12	1	2000121	Primera Quincena de Diciembre de 2000
2000	12	2	2000122	Segunda Quincena de Diciembre de 2000

Tabla 2.6 Descripción de cada quincena de la dimensión tiempo

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

Una vez definido el nivel de detalle de la dimensión tiempo se procede a definir los demás niveles superiores o jerarquías, como se muestra en la figura 2.6 que se presenta a continuación.

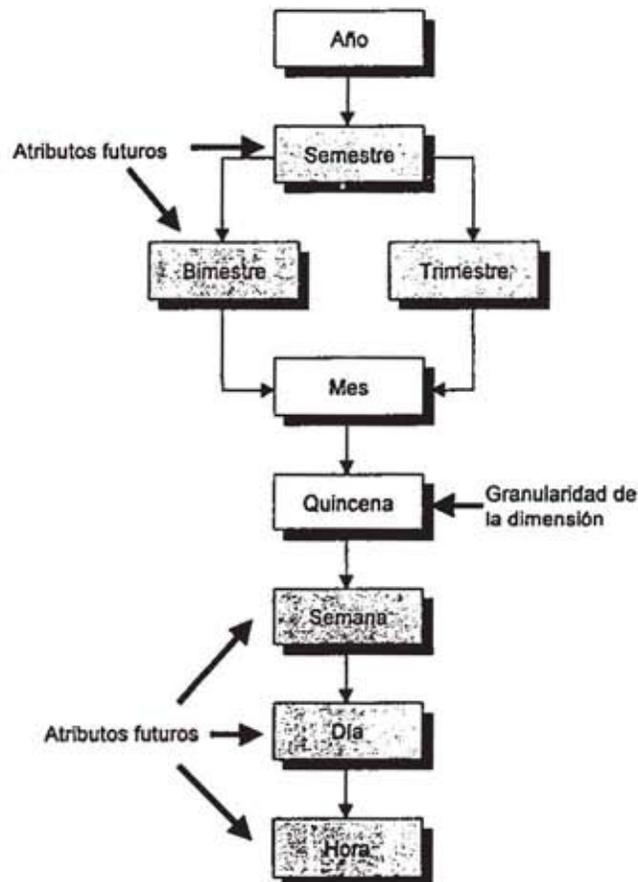


Figura 2.6 Jerarquía de la dimensión tiempo

En la figura 2.5 se observa la posible granularidad futura, además se puede definir cada atributo y su descripción como se muestra en la siguiente tabla.

DIMENSION TIEMPO					
Nombre de atributo	Descripción de atributo	Cardinalidad	Política en caso de dimensión cambiante	en de	Valores de ejemplo
Quincena	Representa el dato específico	26 (por que serían 13 meses de	No actualiza	se	1999011

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

Descripción de quincena	Representa la descripción textual de la quincena	26	No se actualiza	Primera quincena Enero 1999
Mes	Representa el mes al que corresponden las quincenas	13	No se actualiza	199901
Descripción de mes	Representa la descripción textual del mes	13	No se actualiza	Enero 1999
Año	Número de año	2 (solamente se podrá tener información de dos años en un momento dado)	No se actualiza	1999

Tabla 2.7 Definición de atributos de la dimensión Tiempo

Así que la descripción de esta dimensión y su relación con la tabla de hechos es como se muestra en la figura 2.7

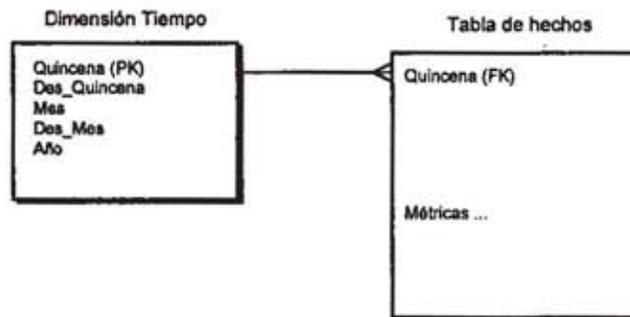


Figura 2.7 Dimensión Tiempo y su relación con la tabla de hechos

2.5.2.3 DIMENSION TIPO DE MOVIMIENTO

La dimensión tipo de movimiento se refiere a los servicios que proporcionan los cajeros automáticos de BanCrecer a sus tarjetahabientes y a los de otros bancos. Desde el punto de vista del negocio son las diferentes acciones que se pueden realizar sobre los productos que ofrece el banco a través de los cajeros automáticos.

En el caso de los cajeros automáticos de BanCrecer los tipos de movimientos que se pueden realizar son, entre otros: Retiro de efectivo, Consulta de saldo, Cambio de NIP, Compra de tiempo aire para teléfono celular. Todos estos representan el nivel de detalle más bajo de la dimensión, por lo que no existe una jerarquía para esta dimensión. En la siguiente tabla se encuentra la definición de cada atributo y su descripción.

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

DIMENSION TIPO DE MOVIMIENTO				
Nombre de atributo	Descripción del atributo	Cardinalidad	Política en caso de dimensión cambiante	Valores de ejemplo
Cve_Movimiento	Representa la clave del movimiento	10	No se actualiza	10, 11
Des_Movimiento	Representa la descripción textual del tipo de movimiento	10	No se actualiza	Retiro, Consulta, Cambio de NIP

Tabla 2.8 Definición de atributos de la dimensión Tipo de Movimiento

Así que la descripción de esta dimensión y su relación con la tabla de hechos es como se muestra en la figura 2.8

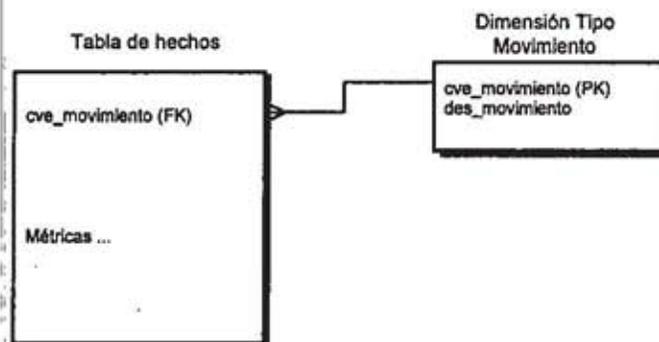


Figura 2.8 Dimensión Tipo de Movimiento y su relación con la tabla de hechos

2.5.2.4 DIMENSION TIPO DE CUENTA

El tipo de cuenta se puede ver como los productos que ofrece el banco y que implican el uso de cajeros automáticos a través de una tarjeta con banda magnética. Entre otros, los tipos de cuenta pueden ser Cheques, Ahorro, Tarjeta de Crédito. De los productos que ofrece el banco algunos que no implican el hecho de que el cliente tenga una tarjeta con banda magnética son: Inversión a plazo fijo e Inversión en el Mercado de Dinero.

Si el negocio se tratara de una tienda de autoservicio esta dimensión sería de las más grandes ya que contendría la descripción de cada uno de los miles de productos y su clasificación por diferentes criterios: perecederos, proveedor, etc.

Para esta dimensión la jerarquía está definida por la agrupación de las cuentas de crédito y las de débito, como se muestra en la figura 2.9 que se presenta a continuación.

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

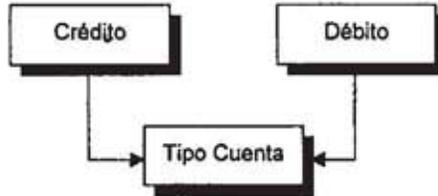


Figura 2.9 Jerarquía de dimensión Tipo de Cuenta

En la figura 2.9 no hay ningún elemento que sea para una posible granularidad futura. En tabla que se muestra a continuación se define cada atributo y su descripción.

DIMENSION TIPO DE CUENTA				
Nombre de atributo	Descripción de atributo	Cardinalidad	Política en caso de dimensión cambiante	Valores de ejemplo
Cve_Cuenta	Clave del tipo de cuenta	5	No se actualiza	01
Des_Cuenta	Representa la descripción textual del tipo de cuenta	5	No se actualiza	Cheques, Maestra, Tarjeta de Crédito

Tabla 2.9 Definición de atributos de la dimensión Tipo de Cuenta

La descripción de esta dimensión y su relación con la tabla de hechos es como se muestra en la figura 2.10

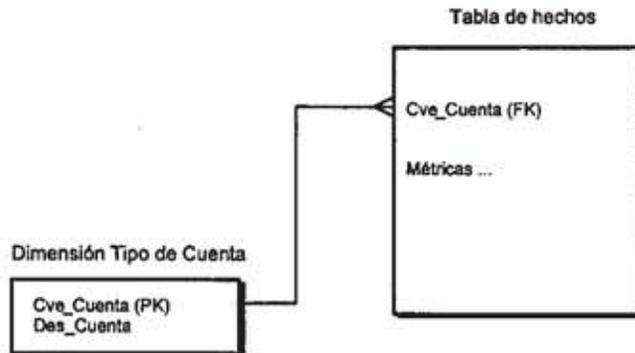


Figura 2.10 Dimensión Tipo de Cuenta y su relación con la tabla de hechos

2.5.2.5 DIMENSION RAZON DE RESPUESTA

La dimensión razón de respuesta contiene la explicación del resultado de cada operación realizada en los cajeros automáticos, originalmente es un catálogo relativamente simple ya

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

que solamente contiene una clave y la descripción de la razón de respuesta. Para los propósitos del CICA esta dimensión no tienen ninguna jerarquía definida por lo que en tabla que se muestra a continuación se define cada atributo y su descripción.

DIMENSION RAZON DE RESPUESTA				
Nombre de atributo	Descripción de atributo	Cardinalidad	Política en caso de dimensión cambiante	Valores de ejemplo
Id_Razon	Clave de la razón de respuesta	25	No se actualiza	00, 01, 50, 51, 59
Des_Razon	Representa la descripción de la razón de respuesta	25	No se actualiza	Aprobada, Fondos insuficientes, Tarjeta expirada

Tabla 2.10 Definición de atributos de la dimensión Razón de Respuesta

La descripción de esta dimensión y su relación con la tabla de hechos es como se muestra en la figura 2.11

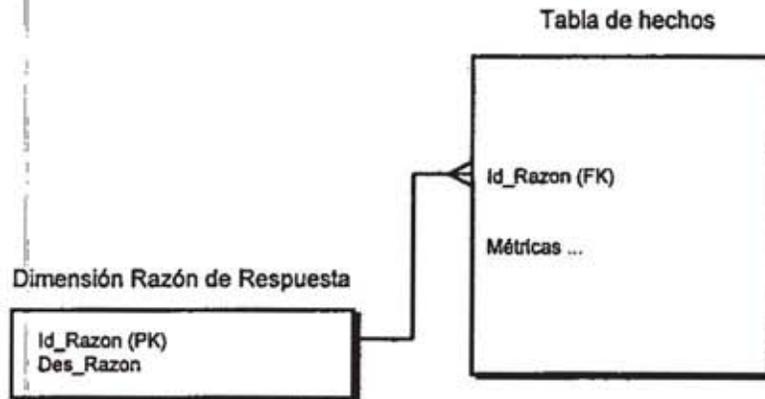


Figura 2.11 Dimensión Razón de Respuesta y su relación con la tabla de hechos

2.5.2.6 DIMENSION GEOGRAFIA

Este tipo de dimensión es junto con la dimensión tiempo la que aparece virtualmente en todo Data Warehouse, ya que describe en donde tienen lugar, en donde se ubican los hechos que se encuentran en la tabla de hechos. Para un Data Warehouse de una Compañía Aseguradora, por ejemplo, esta dimensión puede contener las ubicaciones de los siniestros que han ocurrido en un determinado lapso o la ubicación de los inmuebles asegurados en una cierta póliza; para un negocio de tiendas de autoservicio, se puede definir que esta dimensión contenga las ubicaciones de cada tienda; para el caso del CICA esta dimensión contiene las ubicaciones de los cajeros automáticos de BanCrecer. Si el objetivo del Data Warehouse fuera medir la captación del banco o la colocación de créditos que realiza la institución, habría dos elementos importantes a ubicar geográficamente: Las sucursales y los clientes.

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

En el análisis inicial del comportamiento de los cajeros se distingue claramente una jerarquía asociada a la ubicación de los cajeros: cada cajero se encuentra en una plaza (ciudad o población), cada plaza pertenece a una región y el banco está integrado por regiones. En este punto del análisis se distingue también otra jerarquía, la que define el tipo de ubicación del cajero y el sector económico al que corresponde dicha ubicación. Por ejemplo, si el cajero está ubicado en un edificio de gobierno, el sector es gobierno, si estuviera ubicado en un centro comercial, el sector sería privado.

En la figura 2.12 se muestra la jerarquía definida para la dimensión geografía.

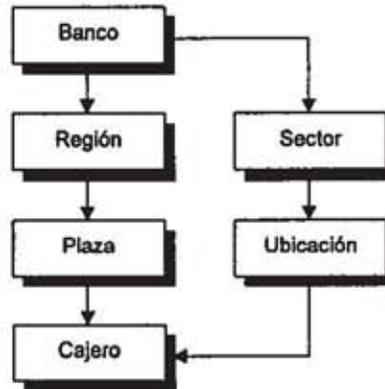


Figura 2.12 Jerarquía para la dimensión Geografía

En tabla que se muestra a continuación se define cada atributo y su descripción.

DIMENSION GEOGRAFIA				
Nombre de atributo	Descripción de atributo	Cardinalidad	Política en caso de dimensión cambiante	Valores de ejemplo
Cve_Cajero	Clave del cajero	1,000	No se actualiza	S10001
Des_Cajero	Nombre del cajero	1,000	Tipo Uno	Dorian's, Tlalpan
Cve_Plaza	Clave de la plaza	100	No se actualiza	0101, 0102, 0440, 0441, 0601, 0602
Des_Plaza	Nombre o descripción de la plaza	100	Tipo Uno	Monterrey, Zihuatanejo, León
Cve_Región	Clave de la región	12	No se actualiza	1, 2, 12
Des_Región	Nombre o descripción de la región	12	Tipo Uno	Monterrey, Metropolitana, Cancún
Cve_Ubicacion	Clave de la ubicación del	10	No se actualiza	1, 2, 15

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

	cajero			
Des_Ubicacion	Descripción de la ubicación del cajero	10	No se actualiza	Sucursal MultiExpress, Sucursal Tradicional, Empresa
Cve_Sector	Clave del sector en donde se ubica el cajero	5	No se actualiza	E, G, R
Des_Sector	Descripción del sector en donde se ubica el cajero	5	No se actualiza	Empresarial, Gobierno, Red de Sucursales
Cve_Banco	Clave del banco	15	No se actualiza	B161, B150
Des_Banco	Descripción del banco	15	No se actualiza	BanCrecer, Bancomer, Serfin

Tabla 2.11 Definición de atributos de la dimensión Geografía

Debido a que las claves de banco dueño de cajero y banco emisor de tarjeta son la misma se puede utilizar el banco dueño para crear la dimensión banco emisor. La dimensión banco emisor solamente contiene el banco.

La descripción de estas dimensiones y su relación con la tabla de hechos es como se muestra en la figura 2.13

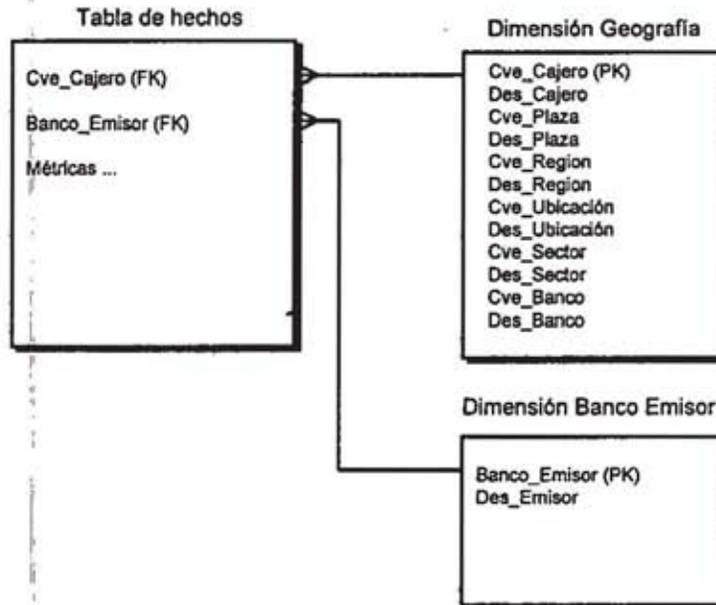


Figura 2.13 Dimensiones Geografía y Banco Emisor y su relación con la tabla de hechos

Del análisis realizado anteriormente en la figura 2.14 se muestra el modelo dimensional del CICA considerando la definición de las dimensiones realizada hasta este punto.

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

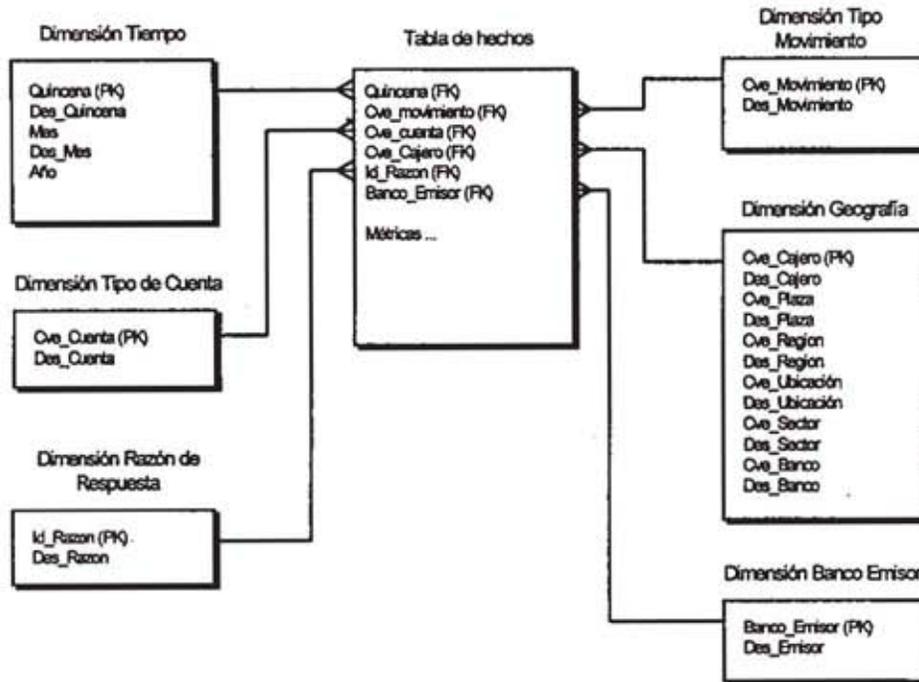


Figura 2.14 Modelo con seis dimensiones definidas parcialmente

2.5.3 METRICAS

Las métricas son los valores que sirven para medir los resultados del negocio en un periodo determinado.

Como hemos visto, el archivo TLF contiene el detalle de las transacciones que se llevan a cabo en los cajeros automáticos y este detalle proporciona la información necesaria para identificar el tiempo, ubicación, tipo de cuenta y tipo de transacción. El proceso de extracción de estos datos debe convertir los identificadores originales de cada campo de los sistemas de origen en llaves apropiadas del Data Warehouse que se ligan a las dimensiones respectivas.

El registro de hechos para una transacción individual frecuentemente contiene solo un "hecho", que es el valor de la transacción. En la mayoría de los casos etiquetamos a este hecho o métrica simplemente como Importe. Debido a que pueden existir muchas clases de transacciones en la misma tabla, no etiquetamos a este hecho con algún nombre más específico.

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

Revisando la descripción de las gráficas y reportes presentada en secciones anteriores, a continuación se presenta la tabla 2.12 que contiene las métricas que existen dentro de cada gráfica y reporte.

Métricas	Reportes			Gráficas																								
	1	2	3	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	
Importe de transacción	X	X	X			X		X	X	X	X	X	X								X	X	X					
Contador de transacción				X	X		X							X	X	X	X	X	X							X	X	X

Tabla 2.12 Métricas dentro de cada gráfica y reporte

La métrica "Importe de transacción" se refiere precisamente el monto que el cajero automático entrega al tarjetahabiente, la métrica "Contador de transacción" se refiere al número de transacciones efectuadas por cada cajero en un periodo determinado. Debido a que algunas gráficas requieren del importe o número de transacciones ACEPTADAS, y haciendo un análisis del tipo de datos que contiene el TLF, se desprende que también existen transacciones RECHAZADAS tanto en la transacción en sí (contador de transacción) como en su importe (debido a los reversos).

Así que las métricas propuestas para el modelo presentado en la figura 2.13 son:

- Importe aceptado
- Transacciones aceptadas
- Importe rechazado
- Transacciones rechazadas

Estas métricas no son derivadas de algún cálculo intermedio, sino que reflejan valores simples o básicos de los hechos que se registran en los cajeros automáticos, a partir de estos valores se pueden obtener otras métricas calculadas, como la media mensual de importes aceptados.

Para comprobar que estas métricas cumplen con todas las dimensiones en la tabla 2.13 se presenta la referencia cruzada de cada métrica con cada dimensión.

Métricas	Dimensiones					
	Tiempo	Tipo de Cuenta	Razón de respuesta	Tipo de Movimiento	Geografía	Banco Emisor
Importe aceptado	En cualquier quincena puede haber importes aceptados	Cualquier tipo de cuenta puede retirar algún importe	En el caso de un importe aceptado la razón de respuesta tiene un valor bien determinado	Para las transacciones que implican un importe el movimiento es retiro.	En cualquier cajero se pueden registrar importes aceptados	Al registrar un importe aceptado se registra el banco emisor de la tarjeta.
Transacciones aceptadas	En cualquier quincena puede haber transacciones aceptadas	Cualquier tipo de cuenta puede hacer alguna transacción exitosamente	En el caso de una transacción aceptada la razón de respuesta tiene un valor bien determinado	Para las demás transacciones el tipo de movimiento está bien determinado	En cualquier cajero se pueden registrar transacciones aceptadas	Al registrar una transacción aceptada se registra el banco emisor de la tarjeta.
Importe rechazado	En cualquier quincena	A cualquier tipo de cuenta se le puede	Para un importe rechazado la razón de	Para las transacciones que implican un	En cualquier cajero se	Al registrar un importe rechazado se

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

	puede haber importes rechazados	aplicar un reverso a su solicitud de retiro	respuesta tiene un valor bien determinado	importe el movimiento es retiro.	pueden registrar importes rechazados	registra el banco emisor de la tarjeta.
Transaccion es rechazadas	En cualquier quincena puede haber transaccion es rechazadas	A cualquier tipo de cuenta se le puede rechazar alguna transacción -	En el caso de una transacción rechazada la razón de respuesta tiene un valor bien determinado	Para las demás transacciones el tipo de movimiento está bien determinado	En cualquier cajero se pueden registrar transaccion es rechazadas	Al registrar una transacción rechazada se registra el banco emisor de la tarjeta.

Tabla 2.13 Referencia cruzada de cada métrica con cada dimensión

Considerando lo expuesto anteriormente en la figura 2.15 se presenta el primer modelo lógico dimensional completo del CICA.

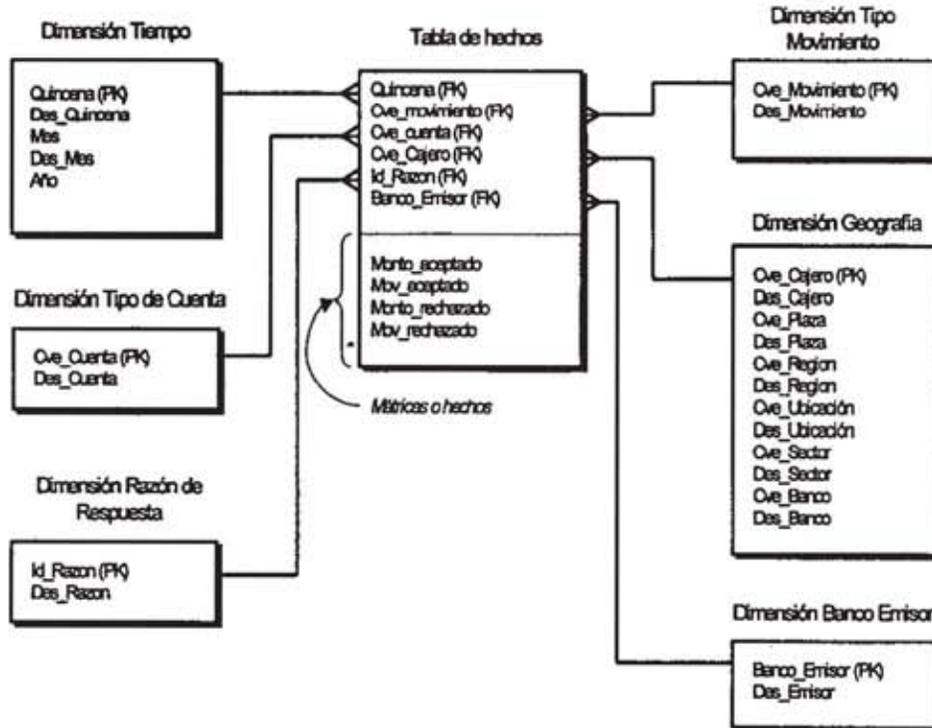


Figura 2.15 Primer modelo lógico completo

2.6 MODELO DE DATOS FISICO

El modelo de datos físico representa al modelo lógico aplicado a una base de datos y herramienta ROLAP bien determinada, en el caso del CICA la base de datos es Informix y la herramienta es Informix MetaCube.

A continuación se explican los conceptos necesarios para cargar el modelo de datos mostrado en la figura 2.15 dentro de Informix MetaCube, debido a que de esta explicación depende la creación de las tablas en Informix que contendrán el modelo en un medio físico.

Para conocer como se deben construir las tablas que contendrán físicamente al modelo se debe entrar al Warehouse Manager utilizando el acceso directo como el que se presenta a continuación:

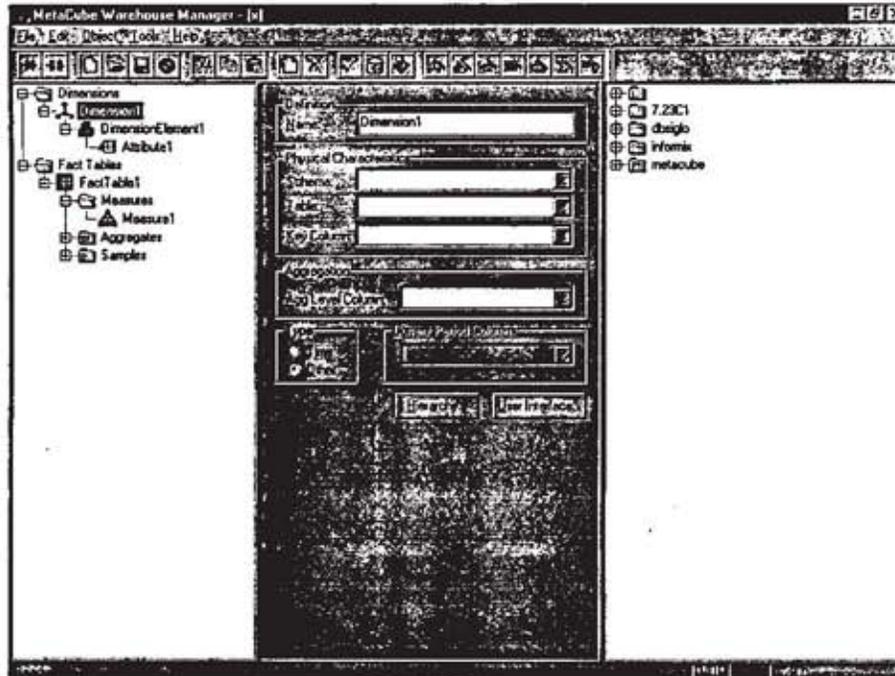


Enseguida se debe seleccionar la opción de nuevo y dar un nombre de modelo, los elementos que se deben considerar para crear el modelo en el Warehouse Manager son los siguientes:

2.6.1 DIMENSION

La pantalla que se presenta a continuación define de forma general la identificación de cada dimensión.

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION



Para crear las tablas se deben considerar los siguientes datos:

Physical Characteristics

Schema: Es el esquema físico que contiene a todas las tablas del modelo.

Table: Es el nombre de la tabla que se debe crear en la base de datos.

Key Column: Es el campo llave que une a la dimensión con la tabla de hechos.

Aggregation

Agg Level Column: En este espacio se debe identificar cual campo contendrá el número asociado a la jerarquía.

2.6.1.1 ELEMENTO DE DIMENSION

Para cada dimensión se deben definir los elementos o campos que la componen, como se presenta en la siguiente pantalla.

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION



Para definir los elementos de dimensión se deben considerar los siguientes datos:

Physical Characteristics

Key column: Especificar el campo llave del elemento de dimensión.

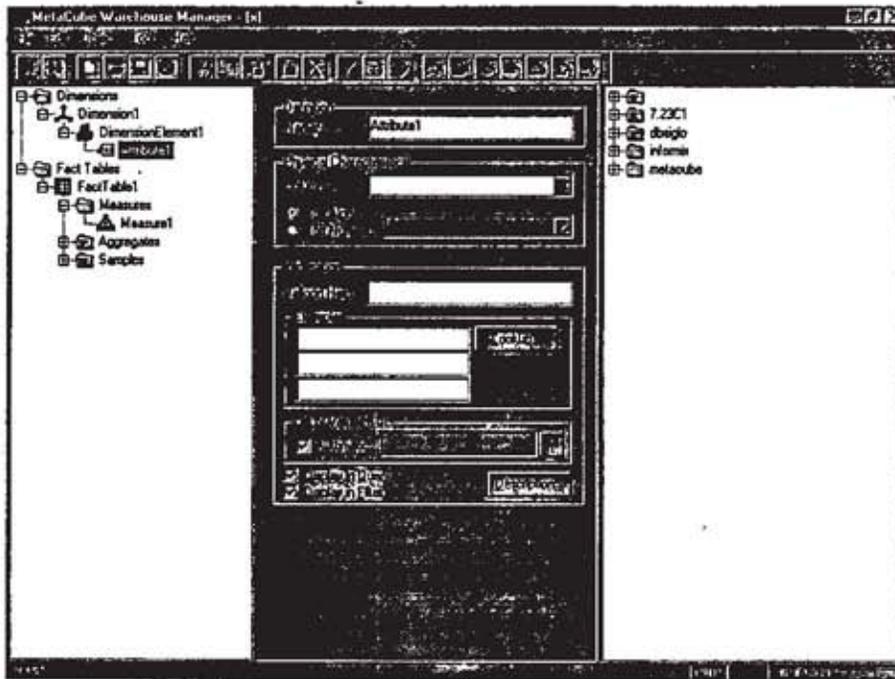
Aggregation

Aggregation level: Es el número que le corresponde al campo dentro de la jerarquía de la dimensión.

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

2.6.1.2 ATRIBUTO DE DIMENSION

Por cada elemento de dimensión se deben definir los atributos que serán visibles para MetaCube Explorer para realizar las consultas.



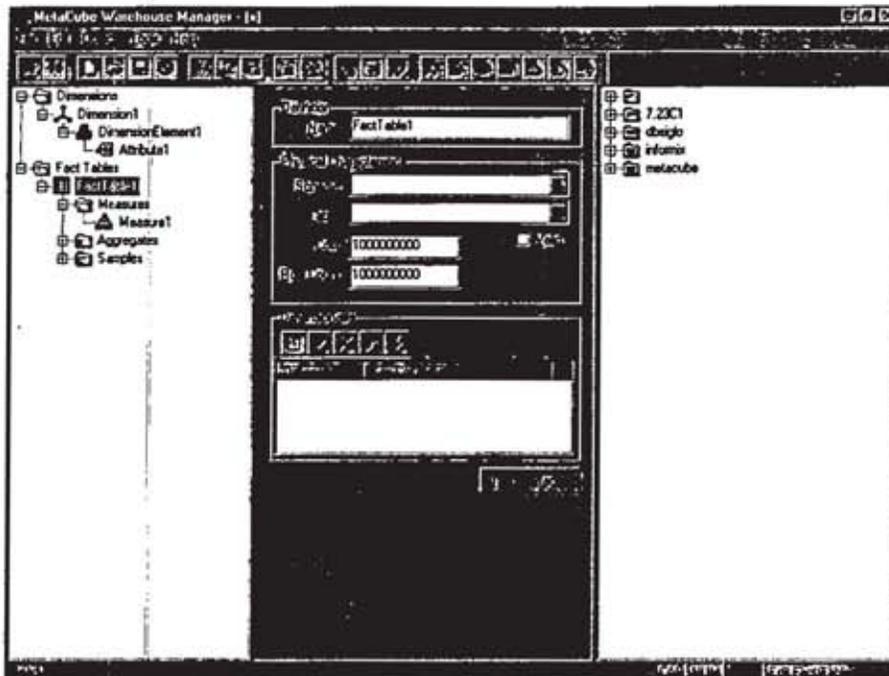
Para definir los atributos de dimensión se deben considerar los siguientes datos:

Physical Characteristics

Column: Nombre del campo correspondiente.

2.6.2 TABLA DE HECHOS

En la pantalla que se muestra a continuación se presenta la ventana para la definición de la tabla de hechos.



Para crear las tablas de hechos se deben considerar los siguientes datos:

Physical Characteristics

Schema: Es el esquema fisico que contiene a todas las tablas del modelo.

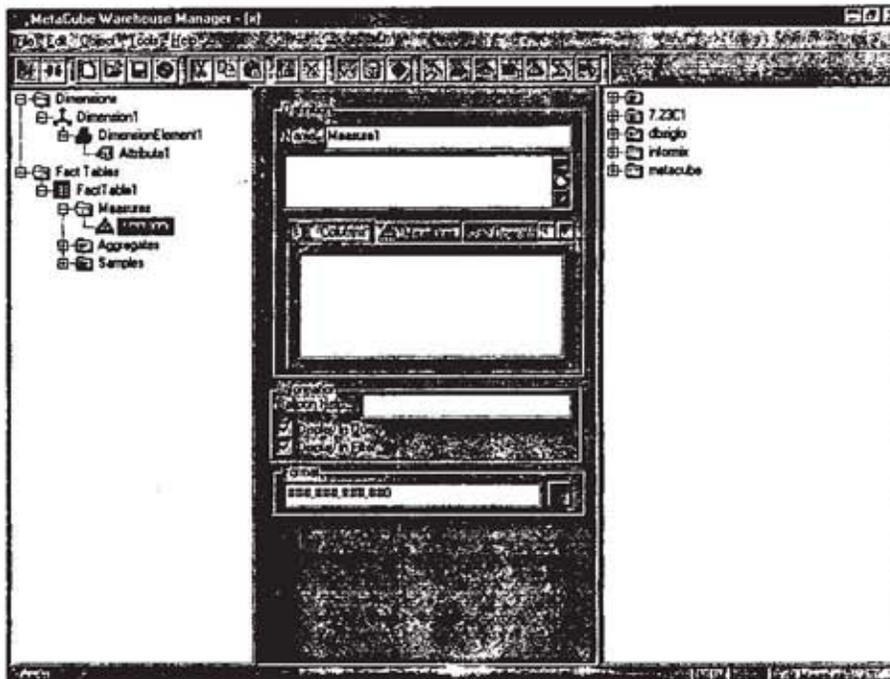
Table: Nombre de la tabla de la base de datos.

Dimensionality

En esta parte se define con cuales dimensiones se relaciona la tabla de hechos y mediante cuales campos.

2.6.2.1 METRICAS

En la pantalla que se presenta a continuación se presentan los datos requeridos para definir las métricas de la tabla de hechos.



Para definir las métricas se deben considerar los siguientes datos:

Definition

Name: Nombre de la métrica.

En el espacio que está enseguida se especifican las columnas (campos) que conforman a esta métrica.

CAPITULO 2. ANALISIS DEL PROBLEMA Y PROPUESTA DE SOLUCION

Considerando las explicaciones para implementar físicamente el modelo, en la figura 2.16 que se presenta a continuación se muestra el modelo dimensional físico del CICA.

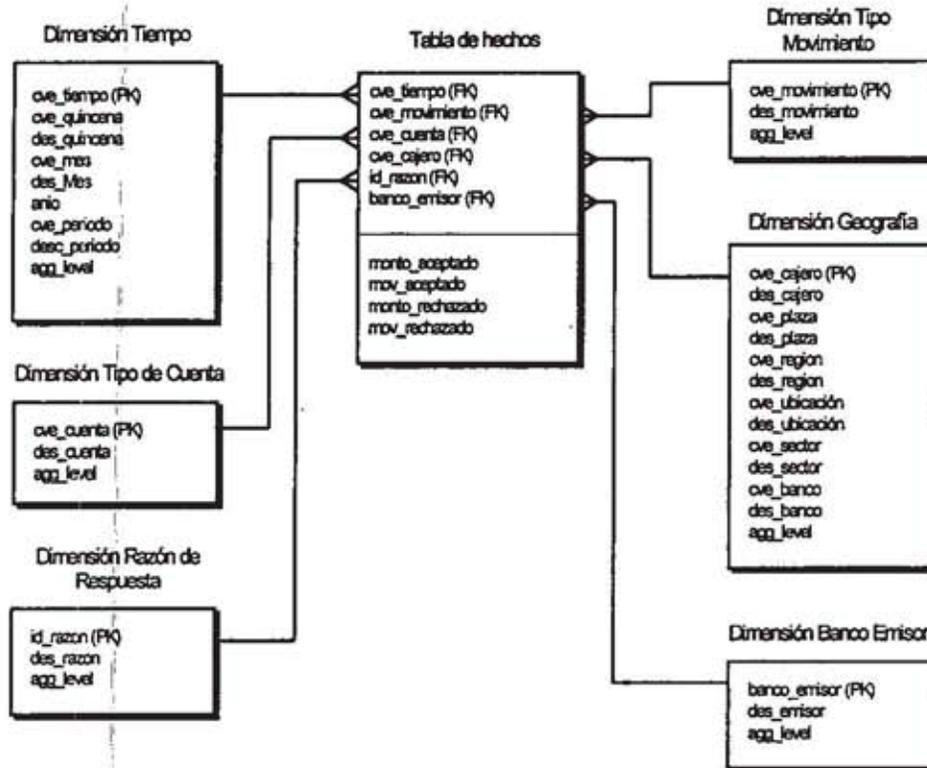


Figura 2.16 Modelo dimensional físico del CICA

En el siguiente capítulo se explicará como se crean las tablas del modelo físico en la base de datos Informix y la forma de obtener los datos de la fuente original para transformarlos y cargarlos al modelo presentado en la figura 2.16.

CAPITULO 3. DISEÑO DE LA SOLUCION

En este capítulo desarrollaremos la solución propuesta en el capítulo anterior, enfocándonos al diseño físico y al diseño de la etapa de datos.

El diseño físico comprende dos aspectos principales:

- La definición y creación de tablas de la base de datos, tanto del modelo como de las tablas auxiliares.
- La definición de los procesos de extracción, transformación y carga de los datos hacia el Data Warehouse.

Para el diseño físico utilizaremos las herramientas propias de Informix para crear las bases de datos, sus tablas y los stored procedures que sean necesarios.

La etapa de datos comprende dos aspectos principales:

- La definición de los procesos de extracción, transformación y carga. Debido a su comprobada utilidad utilizaremos técnicas tradicionales de diseño estructurado como los diagramas de flujo de datos, las especificaciones de proceso y el diccionario de datos.
- La definición de la presentación de los datos mediante Warehouse Manager.

Es importante mencionar que una vez que crearon las tablas del modelo, una parte del equipo de desarrollo comenzó a trabajar en los procesos de extracción, transformación y carga, y la otra parte del equipo inició con la definición de la presentación de los datos mediante Warehouse Manager, debido a que esta última actividad requirió menos tiempo, el equipo completo de desarrollo se dedicó a terminar con la definición de los procesos de extracción, transformación y carga.

3.1. DISEÑO FISICO

En la etapa del diseño físico nos enfocaremos principalmente a la definición para su creación, de las bases de datos y las tablas en Informix para representar físicamente el modelo de datos definido en el capítulo anterior.

A continuación se presenta el diseño físico de cada tabla del modelo dimensional mediante Informix.

TABLA DE HECHOS

Tabla de Hechos:	Contiene los datos de la tabla de hechos		
MOVIMIENTOS	Nombre de Columna	Tipo	Indice
	cve_tiempo	Integer	Si
	cve_cajero	Char(16)	Si
	banco_emisor	Char(4)	Si
	cve_cuenta	Integer	No
	cve_movimiento	Integer	Si
	ld_razon	Char(2)	No
	mov_aceptado	Integer	No
	mov_rechazado	Integer	No
	monto_aceptado	Money(16,2)	No
	monto_rechazado	Money(16,2)	No

DIMENSIONES

Dimensión: TIEMPO		Contiene los valores de tiempo aplicables a la tabla de hechos	
Nombre de Columna	Tipo	Nulos	Indice
cve_tiempo	integer	No	Si
cve_quincena	integer	No	Si
des_quincena	char(25)	No	Si
cve_mes	integer	Si	Si
des_mes	char(10)	Si	Si
anio	integer	Si	Si
cve_periodo	char(1) -	Si	Si
desc_periodo	char(15)	Si	Si
agg_level	integer	Si	Si
fecha_inicial	date	Si	Si
fecha_final	date	Si	Si

Dimensión: GEOGRAFIA		Contiene la identificación de los cajeros, plazas y regiones del banco.	
Nombre de Columna	Tipo	Nulos	Indice
cve_cajero	Char(16)	No	Si
des_cajero	Char(40)	No	Si
cve_plaza	Char(4)	No	Si
des_plaza	Char(30)	No	Si
cve_region	Integer	No	Si
des_region	Char(20)	No	Si
cve_ubicación	Integer	No	Si
des_ubicacion	Char(30)	No	Si
cve_sector	Char(1)	No	Si
des_sector	Char(30)	No	Si
cve_banco	Char(4)	No	No
des_banco	Char(30)	No	No
agg_level	Integer	No	Si

Dimensión: BANCO EMISOR		Contiene la identificación de los bancos emisores de tarjetas.	
Nombre de Columna	Tipo	Nulos	Indice
banco_emisor	Char(4)	No	Si
des_emisor	Char(30)	No	Si
agg_level	Integer	No	Si

Dimensión: CUENTA		Contiene al tipo de cuenta asociada a la tarjeta.	
Nombre de Columna	Tipo	Nulos	Indice
cve_cuenta	integer	No	Si
des_cuenta	Char(50)	No	Si
agg_level	integer	No	Si

CAPITULO 3. DISEÑO DE LA SOLUCION

Dimensión: RAZON_RESPUESTA	Contiene la explicación de la respuesta de los cajeros a cada petición de los tarjetahabientes.		
Nombre de Columna	Tipo	Nulos	Indice
id_razon	Char(2)	No	Si
des_razon	Char(30)	No	Si
agg_level	integer	No	Si

Dimensión: MOVIMIENTO	Contiene al tipo de movimiento.		
Nombre de Columna	Tipo	Nulos	Indice
id_razon	Char(2)	No	Si
des_razon	Char(30)	No	Si
agg_level	integer	No	Si

En la siguiente sección se explican los procesos para alimentar a las tablas cuya definición se ha mostrado arriba. Estos procesos implicarán la creación de tablas auxiliares intermedias que se definirán a su momento.

3.2 DISEÑO DE LA ETAPA DE DATOS

En esta sección se presenta la etapa de datos dentro del ciclo de vida dimensional, que comprende los procesos de extracción, transformación, carga y presentación de los datos de su fuente original al modelo físico del Data Warehouse. Cada proceso se presenta mediante su diagrama de flujo de datos, diccionario de datos y especificación de proceso correspondiente. Para el diagrama de flujo se tomó la metodología de Yourdon. Dentro del diccionario de datos se definieron los flujos y almacenes de datos y también algunos elementos de datos básicos, aunque todos los elementos de datos básicos se encuentran definidos en el TLF del Apéndice A "Descripción de los Campos del TLF". Para las especificaciones de proceso se utilizó el español estructurado.

Para el CICA se tomarán los archivos diarios de BanCreceer (en México) y BanCrecen (en Costa Rica) para aplicar los procesos de la etapa de datos. Los datos que se obtengan de la fuente original se tienen que transformar de un archivo plano a una o varias tablas, de ahí se cargan las tablas resultantes al modelo de datos y finalmente a través del proceso de presentación son accesibles a los usuarios mediante Informix MetaCube. En la figura 3.1 se presenta el diagrama de flujo de datos del sistema (DFD nivel 0) correspondiente a la etapa de datos.

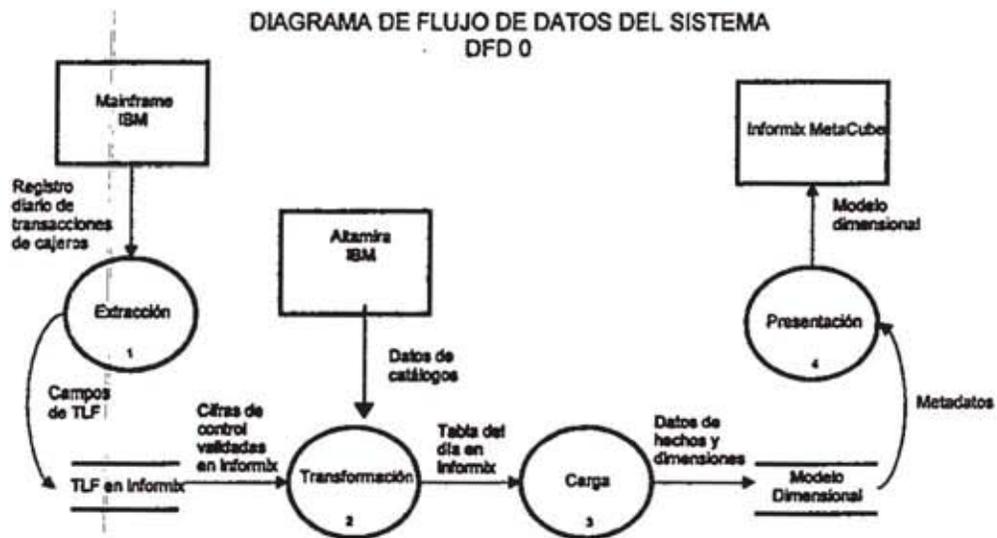


Figura 3.1 Etapa de datos

En las secciones que se presentan a continuación se presentará la especificación de cada proceso mostrado en la figura 3.1.

3.2.1 EXTRACCION

El proceso de extracción es el primer paso en la etapa de datos del Data Warehouse del CICA. Para extraer los datos es necesario identificar de donde se obtendrán y cómo se llevará a cabo este proceso.

Como se mencionó en el Capítulo 1 la fuente de información es el TLF diario y los campos a extraer son los mostrados en el Capítulo 2, en la tabla de la sección IDENTIFICACION DE LAS FUENTES DE INFORMACION.

El proceso consiste en extraer del TLF de IBM los campos identificados en la tabla del Capítulo 2, dejarlos en un archivo temporal, para ahorrar tiempo de transmisión comprimir este archivo mediante la utilidad PKZIP del Host IBM (que es compatible con el PKZIP de Unix), transmitir el archivo mediante FTP al servidor Unix del CICA y confirmar que se haya recibido correctamente en este último.

Un punto importante en la generación del archivo a extraer son las cifras de control que contienen información para validar si el archivo se generó completo y si llegó completa y correctamente a su destino, por lo que el archivo que se genere en IBM debe contener como última línea la cantidad de registros generados y la sumatoria de las columnas que serán métricas, para ser validados después del proceso de extracción.

A continuación se presenta cada diagrama de flujo y sus correspondientes especificaciones de proceso.

DIAGRAMA DE FLUJO

En la figura 3.2 se muestra el DFD 1 que corresponde a la primera parte del proceso de Extracción que consiste en el envío del archivo a Unix.

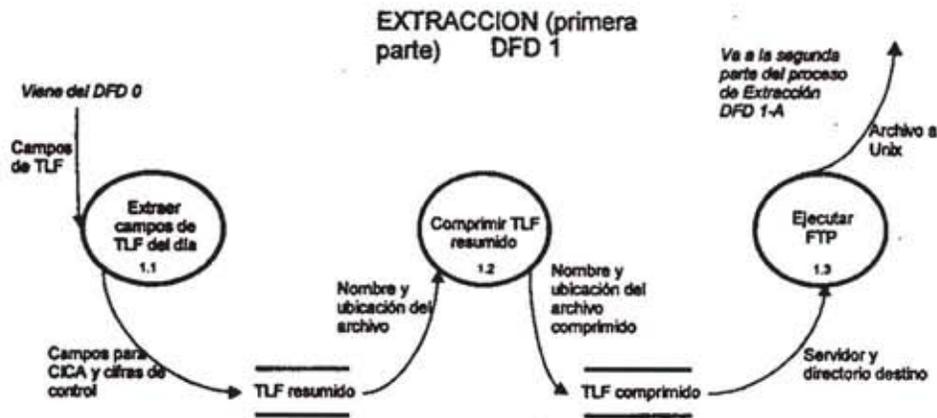


Figura 3.2 Proceso de extracción (primera parte)

En la figura 3.2 se observa que la primera parte del proceso de extracción consiste de tres subprocesos:

- 1.1 Extraer campos de TLF del día.
- 1.2 Comprimir TLF resumido.
- 1.3 Ejecutar FTP.

CAPITULO 3. DISEÑO DE LA SOLUCION

En la extracción de los campos del TLF el archivo resultante contiene los datos de un día y las cifras de control del número de registros y la suma de los campos que corresponden en el modelo a las métricas.

En el subproceso que comprime el TLF extraído se utiliza la utilería PKZIP del Host IBM para que al momento de transmitir el archivo se ocupe menos tiempo que si estuviera del tamaño normal.

En el subproceso que ejecuta el FTP se trasmite el archivo resumido y comprimido al servidor UNIX en donde se continuarán los demás procesos de Transformación, Carga y Presentación.

DICCIONARIO DE DATOS

En esta sección se presentan las definiciones de los flujos y almacenes de datos del diagrama de flujo mostrado arriba.

FLUJOS

NOMBRE DEL FLUJO	DEFINICION
Campos de TLF	Son los campos especificados en el Apéndice A "Descripción de los campos del TLF".
Campos para CICA y cifras de control	Los campos para CICA son los especificados en el Capítulo 2, Sección 2.4 "Identificación de las fuentes de información". Cifras de control = \$CONTROL + CONTEO(REGISTROS) + SUMA(AMT-1) + SUMA(AMT-2) + SUMA(AMT-3)
Nombre y ubicación del archivo.	nombre del archivo = ruta + miembro ruta = módulo al que pertenece el archivo miembro = nombre del archivo en IBM
Nombre y ubicación del archivo comprimido	nombre del archivo comprimido = ruta + nombre comprimido nombre comprimido = nombre con extensión .ZIP
Servidor y directorio destino	servidor = *nombre de equipo fuera del IBM en donde se dejará el TLF resumido y comprimido directorio destino = *ruta dentro del servidor en donde se alojará el archivo comprimido
Archivo a Unix	*Nombre del archivo enviado mediante FTP a Unix

ALMACENES DE DATOS

NOMBRE DEL ALMACEN	DESCRIPCION
TLF resumido	*Archivo que contiene los campos requeridos por el CICA, obtenidos del TLF
TLF comprimido	*TLF resumido al que se le aplica un programa de compresión de datos

ESPECIFICACIONES DE PROCESO

Las especificaciones de proceso que se describen a continuación se codificarán en el Host IBM.

1.1 Extraer campos de TLF del día

```

INICIALIZAR-PROGRAMA
ABRIR-ARCHIVO TLF-DE-ENTRADA
ABRIR-ARCHIVO TLF-DE-SALIDA
LEER REGISTRO TLF-DE-ENTRADA
MIENTRAS NO SEA FN DE ARCHIVO(TLF-DE-ENTRADA)
  INCREMENTA CONTADOR-REGISTROS-LEIDOS + 1
  
```

CAPITULO 3. DISEÑO DE LA SOLUCION

SI (TIPO-TRANSACCION = '01' OR TIPO-TRANSACCION = '02' OR TIPO-TRANSACCION = '20')

*Transacción financiera, administrativa o rechazada

ESCRIBE REC-TYP-DE-ENTRADA A REC-TYP-DE-SALIDA
ESCRIBE TERM-LN-DE-ENTRADA A TERM-LN-DE-SALIDA
ESCRIBE TERM-FIID-DE-ENTRADA A TERM-LN-DE-SALIDA
ESCRIBE TERM-ID-DE-ENTRADA A TERM-ID-DE-SALIDA
ESCRIBE CRD-FIID-DE-ENTRADA A TERM-FIID-DE-SALIDA
ESCRIBE CRD-PAN-DE-ENTRADA A CRD-PAN-DE-SALIDA
ESCRIBE TYP-DE-ENTRADA A TYP-DE-SALIDA
ESCRIBE TRAN-DAT-DE-ENTRADA A TRAN-DAT-DE-SALIDA
ESCRIBE TRAN-TIM-DE-ENTRADA A TRAN-TIM-DE-SALIDA
ESCRIBE POST-DAT-DE-ENTRADA A POST-DAT-DE-SALIDA
ESCRIBE T-CDE-DE-ENTRADA A T-CDE-DE-SALIDA
ESCRIBE T-FROM-DE-ENTRADA A T-FROM-DE-SALIDA
ESCRIBE FROM-ACCT-DE-ENTRADA A FROM-ACCT-DE-SALIDA
ESCRIBE AMT-1-DE-ENTRADA A AMT-1-DE-SALIDA
ESCRIBE AMT-2-DE-ENTRADA A AMT-2-DE-SALIDA
ESCRIBE AMT-3-DE-ENTRADA A AMT-3-DE-SALIDA
ESCRIBE RESP-BYTE-2-DE-ENTRADA A RESP-BYTE-2-DE-SALIDA
ESCRIBE SEQ-NUM-DE-ENTRADA A SEQ-NUM-DE-SALIDA
ESCRIBE ORIG-CRNCY-CDE-DE-ENTRADA A ORIG-CRNCY-CDE-DE-SALIDA
ESCRIBE AUTH-CRNCY-CDE-DE-ENTRADA A AUTH-CRNCY-CDE-DE-SALIDA
ESCRIBE AUTH-CONV-RATE-DE-ENTRADA A AUTH-CONV-RATE-DE-SALIDA
ESCRIBE USER-FLD1-DE-ENTRADA A USER-FLD1-DE-SALIDA

INCREMENTA CONTADOR-REGISTROS-ESCRITOS + 1

INCREMENTA SUMA-AMT1 + AMT-1

INCREMENTA SUMA-AMT2 + AMT-2

INCREMENTA SUMA-AMT3 + AMT-3

FIN-SI

LEER REGISTRO TLF-DE-ENTRADA

FIN-MIENTRAS

ESCRIBE TLF-DE-SALIDA('\$CONTROL',CONTADOR-REGISTROS-ESCRITOS, SUMA-AMT1, SUMA-AMT2, SUMA-AMT3)

FINALIZAR-PROGRAMA

1.2 Comprimir TLF resumido

INICIALIZAR-PROGRAMA

EJECUTA-PROGRAMA PKZIP

DEFINE-PARAMETROS-PKZIP(TLF-DE-ENTRADA,dAAMMDD.cer.cer, ARCHIVO-DE-SALIDA)

* Los parámetros son: Archivo de entrada, nombre del archivo al descomprimirse (en Unix) y nombre del archivo de salida.

* El TLF de entrada es el de salida del proceso anterior

FINALIZAR-PROGRAMA

1.3 Ejecutar FTP

INICIALIZAR-PROGRAMA

EJECUTA-PROGRAMA FTP

DEFINE-PARAMETROS-FTP(DIRECCION-IP-SERVIDOR)

* Se establece la sesión FTP

ENVIAR-USUARIO

ENVIAR-PASSWORD

ESTABLECER-SESION-MODO-BINARIO

ENVIAR-ARCHIVO(dAAMMDD.cer.zip)

TERMINAR-PROGRAMA-FTP

FINALIZAR-PROGRAMA

A continuación se presenta el diagrama de flujo de la segunda parte del proceso de extracción.

DIAGRAMA DE FLUJO

En la figura 3.3 se observa la segunda parte del diagrama de flujo del proceso de extracción.

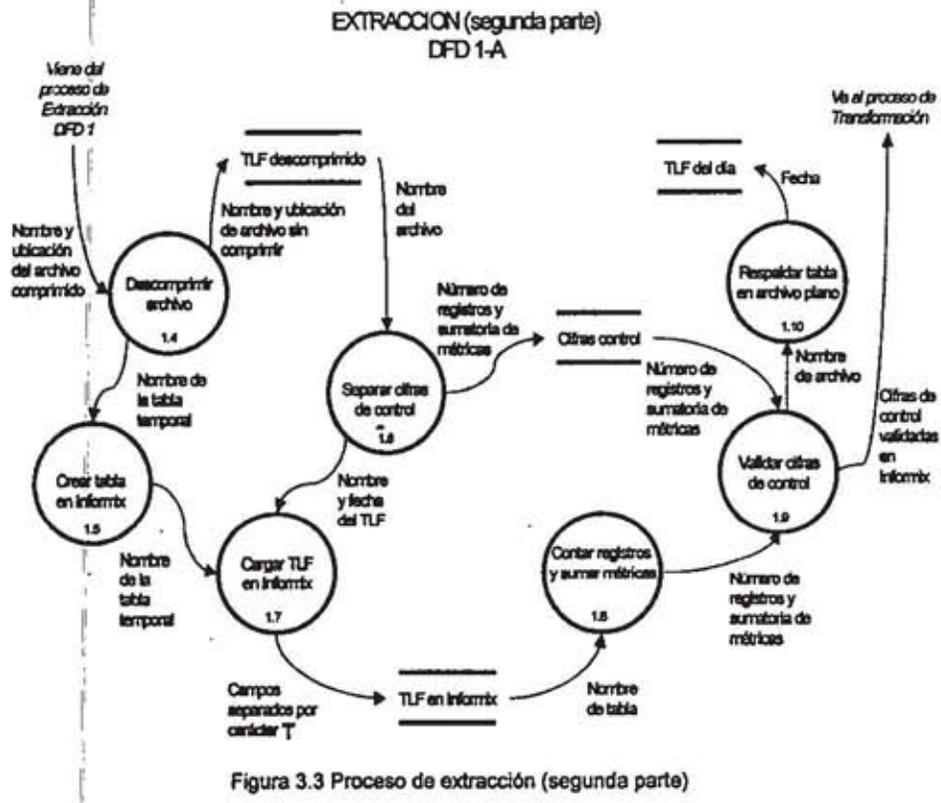


Figura 3.3 Proceso de extracción (segunda parte)

La segunda parte del proceso de extracción consiste de los siguientes subprocesos:

- 1.4 Descomprimir archivos
- 1.5 Crear tabla en Informix
- 1.6 Separar cifras de control
- 1.7 Cargar TLF en tabla Informix
- 1.8 Contar registros y sumar métricas
- 1.9 Validar cifras de control
- 1.10 Respaldar la tabla validada en archivo plano

El proceso de descomprimir el archivo es el primero que se lleva a cabo en el servidor Unix y se lleva a cabo con la utilidad PKZIP de Unix, que es totalmente compatible con el PKZIP del Host IBM.

El proceso de crear la tabla en Informix es con el propósito de tener un espacio temporal en donde alojar el archivo que se ha descomprimido.

El proceso de separar las cifras de control se lleva a cabo identificando al final del archivo la palabra \$CONTROL y las demás columnas separadas por el carácter "|".

El proceso de cargar la tabla de Informix es para tener los datos manipulables.

CAPITULO 3. DISEÑO DE LA SOLUCION

El proceso de contar registros y sumar métricas se lleva a cabo en la tabla que contiene el TLF del día.

El proceso de validar las cifras de control consiste en comparar los resultados de las cifras de control y de la tabla que contiene el TLF del día.

El proceso de respaldar la tabla validada en archivo plano se lleva a cabo cuando las cifras de control se han verificado y es con el propósito de tener una imagen del archivo correcto.
DICCIONARIO DE DATOS

En esta sección se presentan las definiciones de los flujos y almacenes de datos del diagrama de flujo mostrado arriba.

FLUJOS

NOMBRE DEL FLUJO	DEFINICION
Nombre y ubicación del archivo comprimido	Nombre del archivo comprimido = TL + año + mes + día + banco + zip año = *cifra numérica de dos últimos dígitos mes = *cifra numérica de dos dígitos día = *cifra numérica de dos dígitos banco = [.cer .cen] .zip = *extensión del archivo comprimido ubicación del archivo comprimido = *ruta que se asignará en el servidor Unix
Nombre y ubicación del archivo sin comprimir	Nombre del archivo sin comprimir = TL + año + mes + día + banco año = *cifra numérica de dos últimos dígitos mes = *cifra numérica de dos dígitos día = *cifra numérica de dos dígitos banco = [.cer .cen] ubicación del archivo sin comprimir = *ruta que se asignará en el servidor Unix
Nombre de la tabla temporal	Nombre de la tabla temporal = transaccion_tmp
Nombre y fecha del TLF	Nombre y fecha del TLF = *Identificación del TLF para validar cifras de control
Campos separados por carácter "]"	*Por defecto Informix detecta los diferentes campos mediante el separador "]"
Nombre de tabla	Nombre de la tabla = transaccion_tmp
Número de registros y sumatoria de métricas	Número de registros = Sumatoria del total de registros por día Sumatoria de métricas = Suma Importes + Suma de Ajustes + Suma de Disponible
Nombre del archivo	* Es el nombre del archivo sin comprimir
Cifras de control validadas en Informix	* Comparación de las cifras de control del archivo que llega de IBM contra el número de registros y la sumatoria de métricas
Fecha	Fecha = Año + Mes + Día *Para respaldar el archivo con cifras de control validadas

ALMACENES DE DATOS

NOMBRE DEL ALMACEN	DESCRIPCIÓN
TLF descomprimido	*Archivo regenerado a su tamaño original
TLF en Informix	*Tabla que contiene el TLF descomprimido
Cifras de control	*Datos obtenidos de la sumatoria de registros y de los tres importes del TLF descomprimido
TLF del día	*Archivo respaldado con cifras de control validadas

ESPECIFICACIONES DE PROCESO

A continuación se presentan las especificaciones de proceso de la segunda parte del proceso de extracción que se lleva a cabo en Unix.

1.4 Descomprimir archivos

```

INICIALIZAR-PROGRAMA
IDENTIFICAR-FECHA-DE-CARGA
* Descomprime archivos de México y Costa Rica
FKUNZIP tIAAMMDD.cer.zip
PKUNZIP tIAAMMDD.cer.zip
FINALIZAR-PROGRAMA
    
```

1.5 Crear tabla en Informix

```

CREAR TABLA transaccion
( transaccion_id
  transaccion_fecha
  cajero_id
  mensaje_id
  razon_id
  movimiento_id
  cuenta_acceso
  cuenta_id
  numero_cuenta
  banco_id
  moneda_id
  transaccion_corte
  transaccion_import
  transaccion_ajuste
  transaccion_disp
  tipo_cambio
  banco_tarjeta
  moneda_cliente)
    
```

1.6 Separar cifras de control

```

INICIALIZAR-PROGRAMA
BUSCAR CIFRA-CONTROL ARCHIVO-ENTRADA
ALMACENAR CIFRA-CONTROL ARCHIVO-TEMPORAL
CREAR TABLA controlt
CARGAR CIFRAS-CONTROL EN controlt
FINALIZAR-PROGRAMA
    
```

1.6 Cargar TLF en tabla Informix

```

INICIALIZAR-PROGRAMA
CARGAR tIAAMMDD.cer EN transaccion
FINALIZAR-PROGRAMA
    
```

1.7 Contar registros y sumar métricas

```

INICIALIZAR-PROGRAMA
CONTAR REGISTROS DE transaccion
SUMAR transaccion_import DE transaccion
SUMAR transaccion_ajuste DE transaccion
SUMAR transaccion_disp DE transaccion
FINALIZAR-PROGRAMA
    
```

1.8 Validar cifras de control

```

INICIALIZAR-PROGRAMA
SI (controlt.registros = CONTAR REGISTROS DE transaccion
  Y (controlt.suma1 = SUMA(transaccion_importe)
  Y (controlt.suma2 = SUMA(transaccion_ajuste)
  Y (controlt.suma3 = SUMA(transaccion_disp)
  ENTONCES controlt.status = 'S'
SINO
    
```

```
control.status = 'N'  
FIN-SI  
FINALIZAR-PROGRAMA
```

1.9 Respalda la tabla validada en archivo plano

```
INICIALIZAR-PROGRAMA  
DESCARGA transaccion EN transacAAMMDD.dat PARA dia-actual  
FINALIZAR-PROGRAMA
```

3.2.2 TRANSFORMACION

El proceso de transformación consiste de dos grandes subprocesos: obtener los datos para la tabla de hechos y obtener los datos para las dimensiones.

Para el proceso de la tabla de hechos se obtienen de manera casi inmediata los registros que servirán para la carga, para el proceso de las dimensiones se deben llevar a cabo otros subprocesos antes de pasar de la transformación a la carga.

DIAGRAMA DE FLUJO

Los subprocesos necesarios para obtener los datos para la tabla de hechos se muestran en la figura 3.4 que corresponde al DFD 2 (primera parte).

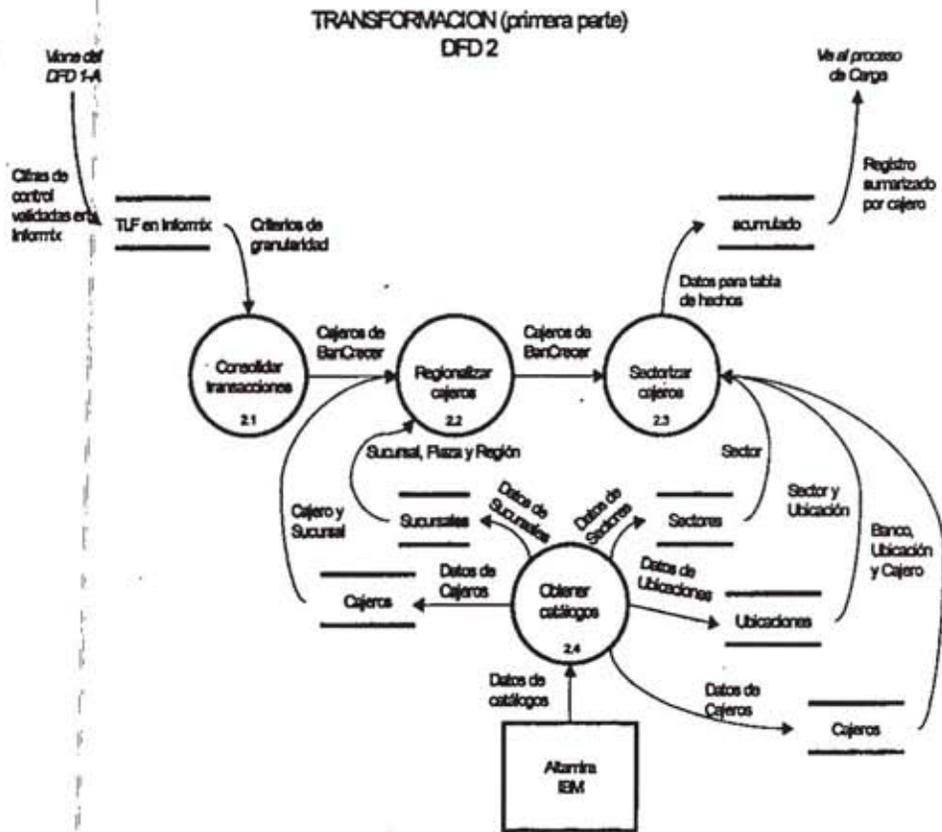


Figura 3.4 Proceso de transformación (primera parte)

El diagrama presentado anteriormente corresponde a la transformación de los datos diarios de las transacciones de los cajeros, complementados con los datos de los catálogos de Sucursales, Cajeros, Ubicaciones y Sectores que se actualizan en Alamira.

CAPITULO 3. DISEÑO DE LA SOLUCION

La primera parte del proceso de Transformación consiste de los siguientes subprocesos:

- 2.1 Consolidar transacciones
- 2.2 Regionalizar cajeros
- 2.3 Sectorizar cajeros
- 2.4 Obtener catálogos

DICCIONARIO DE DATOS

En esta sección se presentan las definiciones de los flujos y almacenes de datos del diagrama de flujo mostrado arriba.

FLUJOS

NOMBRE DEL FLUJO	DEFINICION
Cifras de control validadas en Informix	*Datos obtenidos de la sumatoria de registros y de los tres importes del TLF descomprimido
Criterios de granularidad	*Detalle mínimo de la información a nivel de cajero
Cajeros de BanCreceer	Banco Id = 0161
Cajero y Sucursal	*Datos obtenidos de la tabla de Cajeros
Sucursal, Plaza y Región	*Datos obtenidos de la tabla Sucursales
Datos para tabla de hechos	*Registros con Cajero, Sucursal, Plaza, Región, Sector y Ubicación identificados
Registro sumariado por Cajero	*Consolidación de varios registros en uno solo
Datos de Cajeros	*Información del máximo nivel de detalle
Datos de Sucursales	*Información del siguiente nivel de detalle, incluyendo Plazas y Regiones.
Datos de Sectores	*Identificación del catálogo del tipo de actividad económica a la que pertenece la empresa en donde se ubica el Cajero.
Datos de Ubicaciones	*Identificación del catálogo del tipo de edificio en donde se ubica el Cajero.
Datos de catálogos	*Información de los diferentes tipos de catálogos.

ALMACENES DE DATOS

NOMBRE DEL ALMACEN	DESCRIPCION
TLF en Informix	*Tabla que contiene el TLF descomprimido
Cajeros	*Catálogo de Cajeros de BanCreceer
Sucursales	*Catálogo de Sucursales de BanCreceer (Incluyendo Plazas y Regiones)
Sectores	*Catálogo de Sectores de los Cajeros de BanCreceer
Ubicaciones	*Catálogo de Ubicaciones de los Cajeros de BanCreceer
Acumulado	*Tabla que contiene los registros consolidados del TLF en Informix

A continuación se presentan las especificaciones de proceso.

ESPECIFICACIONES DE PROCESO

2.1 Consolidar transacciones

Este subproceso es uno de los más importantes por que en él están considerados los criterios para identificar si una transacción fue aceptada o rechazada y si el importe fue aceptado o rechazado; tales criterios son los presentados en el Apéndice C "Descripción de los Reportes Originales".

INICIALIZAR-PROGRAMA

* Variables que contendrán las métricas

Num_Aceptadas = 0

Num_Rechazadas = 0

Importe_Aceptadas = 0

Importe_Rechazadas = 0

SELECCIONA

Banco_id

Cajero_id

Cuenta_id

Banco_tarjeta

Movimiento_id

Moneda_id

Razon_id

Mensaje_id

Transaccion_corte

Transaccion_importe

Transaccion_ajuste

DE transaccion

DONDE transaccion_corte = Fecha de Procesamiento

SI (movimiento_id = '10') ENTONCES

SI (Razon_id NO-ES ('00', '01')) ENTONCES

INCREMENTA Num_Rechazadas = Num_Rechazadas + 1

INCREMENTA Importe_Rechazadas = Importe_Rechazadas + Transaccion_Importe

SINO

SI (mensaje_id = '420') ENTONCES *Reverso

SI (transaccion_ajuste = 0) ENTONCES *Reverso total

DECREMENTA Num_Aceptadas = Num_Aceptadas - 1

DECREMENTA Importe_Aceptadas = Importe_Aceptadas - Transaccion_Importe

INCREMENTA Num_Rechazadas = Num_Rechazadas + 1

INCREMENTA Importe_Rechazadas = Importe_Rechazadas + Transaccion_Importe

SINO

INCREMENTA Importe_Aceptadas = (Importe_Aceptadas - Transaccion_Importe) + Transaccion_Ajuste

INCREMENTA Importe_Rechazadas = Importe_Rechazadas + (Transaccion_Importe - Transaccion_Ajuste)

FIN-SI

SINO

SI (mensaje_id <> 9980) ENTONCES

INCREMENTA Num_Aceptadas = Num_Aceptadas + 1

INCREMENTA Importe_Aceptadas = Importe_Aceptadas + 1

FIN-SI

FIN-SI

SINO

SI (razon_id NO-ES ('00', '01')) ENTONCES

INCREMENTA Num_Rechazadas = Num_Rechazadas + 1

SINO

INCREMENTA Num_Aceptadas = Num_Aceptadas + 1

FIN-SI

2.2 Regionalizar cajeros

```

INICIALIZAR-PROGRAMA
* Variables
BUSCAR SUCURSAL, PLAZA Y REGION DE CADA CAJERO EN CATALOGOS DE SUCURSAL, PLAZA Y
REGION
SI SUCURSAL ES NULA Y PLAZA ES NULA Y REGION ES NULA
*No es cajero de BanCreer
  sSucursal = 0
  sPlaza = 0
  sRegion = 0
SINO
  sSucursal = SUCURSAL
  sPlaza = PLAZA
  sRegion = REGION
FIN-SI
    
```

2.3 Sectorizar cajeros

```

INICIALIZAR-PROGRAMA
BUSCAR UBICACIÓN, SECTOR DE CADA CAJERO
SI UBICACIÓN ES NULA Y SECTOR ES NULO
*No es cajero de BanCreer
  sUbicacion = 0
  sSector = 0
SINO
  sUbicacion = UBICACIÓN
  sSector = SECTOR
FIN-SI
    
```

```

INSERTA EN ACUMULADO_MC
CAJERO
sSucursal
sPlaza
sRegion
BANCO-DUEÑO-DEL-CAJERO
BANCO-EMISOR-DE-LA-TARJETA
CLAVE-DE-MOVIMIENTO
CLAVE-DE-CUENTA
Clave-de-moneda
Num_Aceptadas
Num_Rechazadas
Importe_Aceptadas
Importe_Rechazadas
sUbicación
sSector
Fecha_de_corte
    
```

2.4 Obtener Catálogos

```

INICIALIZAR-PROGRAMA
SI ES FINAL DEL DIA
  LEER catálogo DE CAJEROS
  ESCRIBIR catálogo en tabla Cajeros
  LEER catálogo DE SUCURSALES
  ESCRIBIR catálogo en tabla Sucursales
  LEER catálogo DE SECTORES
  ESCRIBIR catálogo en tabla Sectores
  LEER catálogo DE UBICACIONES
  ESCRIBIR catálogo en tabla Ubicaciones
FIN-SI
    
```

Los subprocesos necesarios para obtener los datos para la dimensión Geografía muestran en la figura 3.5 que corresponde al DFD 2 (segunda parte).

DIAGRAMA DE FLUJO

TRANSFORMACION (segunda parte)
DFD 2-A

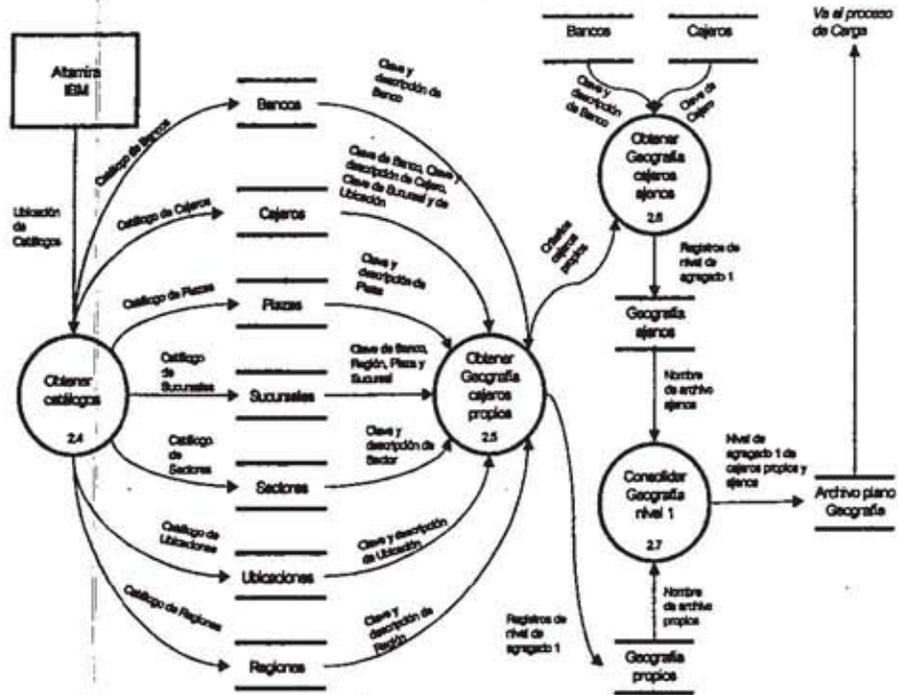


Figura 3.5 Proceso de Transformación (segunda parte)

En este subproceso de transformación se genera el nivel de agregado 1 de la dimensión Geografía; a partir del archivo plano que se genera en este subproceso, el proceso de carga creará los demás niveles de agregado para tener completa la dimensión Geografía.

Esta parte del proceso de Transformación consiste de los siguientes subprocesos:

- 2.4 Obtener catálogos
- 2.5 Obtener Geografía cajeros propios
- 2.6 Obtener Geografía cajeros ajenos
- 2.7 Consolidar Geografía nivel 1

DICCIONARIO DE DATOS

En esta sección se presentan las definiciones de los flujos y almacenes de datos del diagrama de flujo mostrado arriba.

CAPITULO 3. DISEÑO DE LA SOLUCION

FLUJOS

NOMBRE DEL FLUJO	DEFINICION
Ubicación de Catálogos	*En Altamira se ubican los catálogos que se generan diariamente, ya que pueden cambiar de un día para otro
Catálogo de Bancos	*En el caso de esta dimensión son los bancos dueños de cajeros automáticos
Catálogo de Regiones	*Regiones de BanCreceer
Catálogo de Plazas	*Plazas de BanCreceer
Catálogo de Sucursales	*Sucursales de BanCreceer
Catálogo de Sectores	*Sectores de BanCreceer
Catálogo de Ubicaciones	*Ubicaciones de BanCreceer
Catálogo de Cajeros	*Cajeros BanCreceer
Clave y descripción de Banco	*Clave y descripción del banco dueño de cajeros automáticos
Clave y descripción de Región	*Regiones de BanCreceer
Clave y descripción de Plaza	*Plazas de BanCreceer cve_plaza = region_id[3,4] + plaza_id[3,4]
Clave de Región, Plaza y Sucursal	*Regiones, Plazas y Sucursales de BanCreceer
Clave y descripción de Sector	*Sectores de BanCreceer
Clave y descripción de Ubicación	*Ubicaciones de BanCreceer
Clave de Banco, clave y descripción de Cajero, clave de Sucursal y de Ubicación	*Para diferenciar a los cajeros BanCreceer de los de otro banco se incluye la clave del banco. Para identificar a los cajeros se utiliza su clave y para el caso de los cajeros de BanCreceer su descripción. Para identificar de cada cajero de BanCreceer en cuál plaza y región se encuentran se utiliza la clave de Sucursal. Para identificar la ubicación de cada cajero de BanCreceer se utiliza la clave de la Ubicación en donde se localiza.
Nivel de agregado 1	*Datos para armar el detalle de la dimensión.

ALMACENES DE DATOS

NOMBRE DEL ALMACEN	DESCRIPCION
Bancos	*Bancos dueños de cajeros automáticos.
Regiones	*Regiones de BanCreceer
Plazas	*Plazas de BanCreceer
Sucursales	*Sucursales de BanCreceer
Sectores	*Sectores de BanCreceer
Ubicaciones	*Ubicaciones de BanCreceer
Cajeros	*Cajeros de BanCreceer
Archivo plano Geografía	*Archivo que contiene el primer nivel de agregado de la dimensión

ESPECIFICACIONES DE PROCESO

2.4 Obtener catálogos

LEER catálogo DE Bancos
 ESCRIBIR tabla DE Bancos

LEER catálogo DE Regiones
 ESCRIBIR tabla DE Regiones

LEER catálogo DE Plazas
 ESCRIBIR tabla DE Plazas

LEER catálogo de Sucursales
 ESCRIBIR tabla de Sucursales

LEER catálogo de Sectores
 ESCRIBIR tabla de Sectores

LEER catálogo de Ubicaciones
 ESCRIBIR tabla de Ubicaciones

LEER catálogo de Cajeros
 ESCRIBIR tabla de Cajeros

2.5 Obtener Geografía cajeros propios

INICIALIZAR-PROGRAMA

DESCARGAR en geografia_propios

SELECCIONA

1,cve_banco,cve_region,cve_plaza,cve_cajero,cve_ubicación,cve_sector,banco_dsc,region_dsc,plaza_dsc,cajero_dsc,ubicación_dsc,sector_dsc

DE bancos, regiones, plazas, sucursales, cajeros, ubicaciones, sectores

DONDE

cve_region de regiones = cve_region_id de plazas Y

cve_banco de regiones = cve_banco de plazas Y

cve_plaza de plazas = cve_plaza de sucursales Y

cve_region de plazas = cve_region de sucursales Y

cve_banco de plazas = cve_banco de sucursales Y

cve_sucursal de sucursales = cve_sucursal de cajeros Y

cve_ubicación de cajeros = cve_ubicación de ubicaciones Y

cve_sector de ubicaciones = cve_sector de sectores Y

cve_banco de cajeros = cve_banco_id de bancos

2.6 Obtener Geografía cajeros ajenos

INICIALIZAR-PROGRAMA

DESCARGAR en geografia_ajenos

SELECCIONA 1,cve_banco,'0','0',cve_cajero,'0','0',banco_dsc,'REGION DESCONOCIDA','PLAZA

DESCONOCIDA','CAJERO DESCONOCIDO','UBICACIÓN DESCONOCIDA','SECTOR DESCONOCIDA'

DE cajeros, bancos

DONDE cve_cajero de cajeros NO-ESTE-EN CONJUNTO-DE geografia_propios

2.7 Consolidar Geografía nivel 1

UNIR geografia_propios CON geografia_ajenos EN archivo_plano_geografia

CAPITULO 3. DISEÑO DE LA SOLUCION

Los procesos de transformación de las demás dimensiones se presentan en la figura 3.6 que se encuentra a continuación. La excepción es la dimensión tiempo que se genera y carga directamente en la base de datos sin que intervenga ningún proceso de este tipo.

DIAGRAMA DE FLUJO

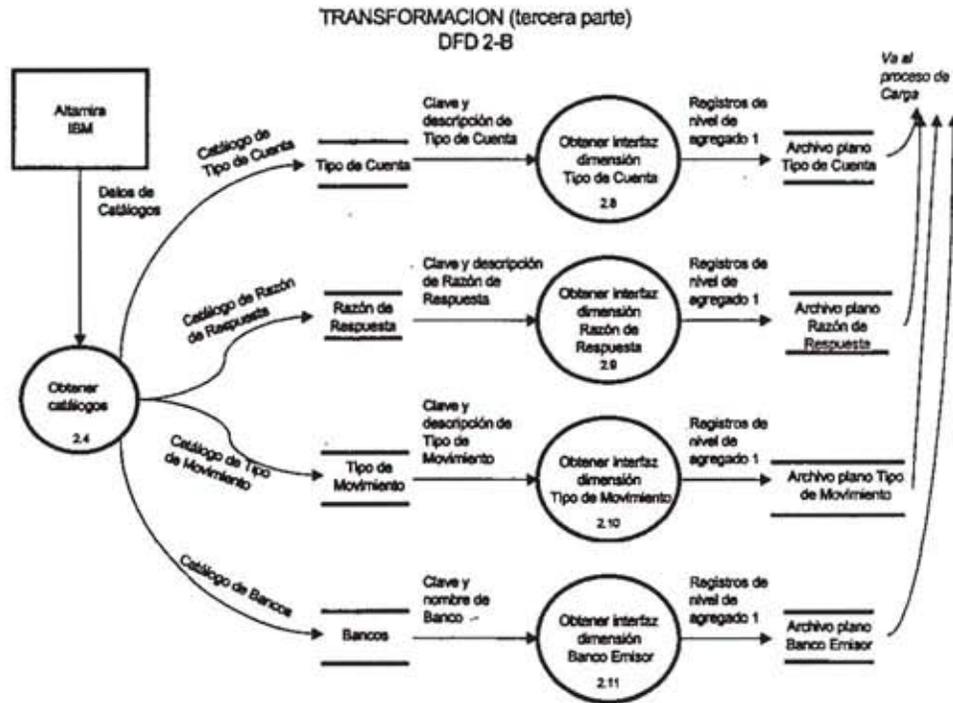


Figura 3.6 Proceso de Transformación (tercera parte)

En esta parte del proceso de Transformación se generan los archivos planos necesarios para cargar las dimensiones: Tipo de Cuenta, Razón de Respuesta, Tipo de Movimiento y Banco Emisor.

Esta parte del proceso de Transformación consiste de los siguientes subprocesos:

- 2.4 Obtener catálogos
- 2.8 Obtener interfaz dimensión Tipo de Cuenta
- 2.9 Obtener interfaz dimensión Razón de Respuesta
- 2.10 Obtener interfaz dimensión Tipo de Movimiento
- 2.11 Obtener interfaz dimensión Banco Emisor

DICCIONARIO DE DATOS

En esta sección se presentan las definiciones de los flujos y almacenes de datos del diagrama de flujo mostrado arriba.

FLUJOS

NOMBRE DEL FLUJO	DEFINICION
Datos de Catálogos	*Catálogos de Altamira
Catálogo de Tipo de Cuenta	*Tipo de cuenta que realiza la operación
Catálogo de Razón de Respuesta	*Explicación de la respuesta proporcionada por el cajero
Catálogo de Tipo de Movimiento	*Tipo de movimiento realizado
Catálogo de Bancos	*Banco dueño del cajero y/o emisor de la tarjeta que accesa al cajero
Clave y descripción de Tipo de Cuenta	*Registro desnormalizado con clave y descripción correspondiente
Clave y descripción de Razón de Respuesta	*Registro desnormalizado con clave y descripción correspondiente
Clave y descripción de Tipo de Movimiento	*Registro desnormalizado con clave y descripción correspondiente
Clave y nombre de Banco	*Registro desnormalizado con clave y descripción correspondiente
Registros con nivel de agregado 1	*Se obtienen los datos correspondientes de cada dimensión, identificándolos con el nivel de agregado 1.

ALMACENES DE DATOS

NOMBRE DEL ALMACEN	DESCRIPCION
Tipo de Cuenta	*Contiene los datos del catálogo correspondiente en un archivo plano
Razón de respuesta	*Contiene los datos del catálogo correspondiente en un archivo plano
Tipo de Movimiento	*Contiene los datos del catálogo correspondiente en un archivo plano
Bancos	*Contiene los datos del catálogo correspondiente en un archivo plano
Archivo plano Tipo de Cuenta	*Contiene los datos desnormalizados listos para cargarse a la dimensión correspondiente
Archivo plano Razón de Respuesta	*Contiene los datos desnormalizados listos para cargarse a la dimensión correspondiente
Archivo plano Tipo de Movimiento	*Contiene los datos desnormalizados listos para cargarse a la dimensión correspondiente
Archivo plano Banco Emisor	*Contiene los datos desnormalizados listos para cargarse a la dimensión correspondiente

ESPECIFICACIONES DE PROCESO

2.4 Obtener catálogos

LEER catálogo de Tipo de Cuenta
 ESCRIBIR tabla de Tipo de Cuenta

LEER catálogo de Razón de Respuesta
 ESCRIBIR tabla de Razón de Respuesta

LEER catálogo de Tipo de Movimiento
 ESCRIBIR tabla de Tipo de Movimiento

LEER catálogo de Bancos
 ESCRIBIR tabla de Bancos

2.8 Obtener interfaz dimensión Tipo de Cuenta

INICIALIZAR-PROGRAMA

DESCARGAR l, cve_cuenta, cuenta_desc de tabla Tipo_Cuenta EN Archivo_Plano_Tipo_Cuenta

2.9 Obtener interfaz dimensión Razón de Respuesta

INICIALIZAR-PROGRAMA

DESCARGAR l, cve_cuenta, cuenta_desc de tabla Razon_Respuesta EN Archivo_Plano_Razon_Respuesta

2.10 Obtener interfaz dimensión Tipo de Movimiento

INICIALIZAR-PROGRAMA

DESCARGAR l, cve_cuenta, cuenta_desc de tabla Tipo_Movimiento EN Archivo_Plano_Tipo_Movimiento

2.11 Obtener interfaz dimensión Banco Emisor

INICIALIZAR-PROGRAMA

DESCARGAR l, cve_cuenta, cuenta_desc de tabla Bancos EN Archivo_Plano_Banco_Emisor

3.2.3 CARGA

El proceso de carga se refiere a que después de que los datos han sido transformados están disponibles para alimentar a las tablas del modelo de datos.

Del mismo modo como se dividió el proceso de Transformación, el de carga consta de: Carga de las dimensiones y Carga de la tabla de hechos. Se lleva a cabo en primer lugar la carga de las dimensiones ya que en estas tablas es en donde se encuentran las llaves primarias que sirven para relacionarse con la tabla de hechos.

Dentro de la carga de las dimensiones existe un proceso especialmente diferente: el de la dimensión tiempo, ya que es en esta tabla en donde se lleva el control de la quincena actual cargada, además la tabla puede existir desde antes que se lleve a cabo la primera carga real de datos.

Para las demás tablas el proceso consiste de recibir los registros obtenidos del proceso de transformación y generar los niveles de agregado que hagan falta y que se almacenarán en cada tabla de dimensión del modelo físico.

Dentro de la carga de la tabla de hechos se debe considerar una de las principales premisas para la permanencia de los datos: 13 meses en línea, por lo que antes de cargar un nuevo mes, se debe respaldar y borrar el mes más antiguo, por ejemplo, si se va a cargar Febrero del 2000, se debe respaldar y borrar Enero de 1999 para que solamente haya trece meses en línea.

Otros aspectos importantes de la carga de la tabla de hechos son: el costo de las consultas realizadas sobre esta tabla, que está en función directa del número de registros que tiene la tabla. Otro aspecto es la generación de tablas de agregados que contienen las combinaciones de datos de la tabla de hechos que responden a las consultas más frecuentes, sin tener que acceder a la tabla de hechos original, por lo que la respuesta a estas consultas resulta ser más rápida.

En la figura 3.7 se presenta el diagrama de flujo del proceso de carga de la dimensión Geografía.

DIAGRAMA DE FLUJO

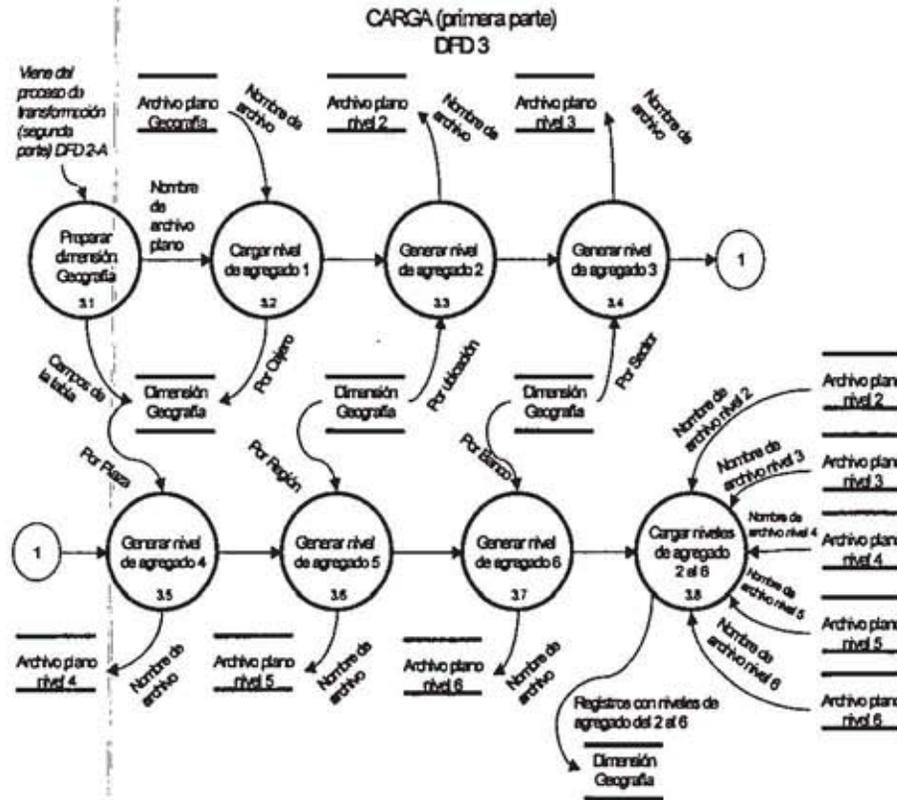


Figura 3.7 Carga de Geografía

De la figura anterior se puede ver que a partir del proceso de transformación se tienen los datos del nivel de detalle máximo o nivel de agregado 1, con esa información cargada en la tabla de dimensión de Geografía se generarán archivos temporales por cada nivel de agregado, al final se cargarán los datos de estos archivos a la tabla Geografía para tener la dimensión completamente integrada.

Los subprocesos que componen a este proceso son:

- 3.1 Preparar dimensión Geografía
- 3.2 Cargar nivel de agregado 1
- 3.3 Generar nivel de agregado 2
- 3.4 Generar nivel de agregado 3
- 3.5 Generar nivel de agregado 4
- 3.6 Generar nivel de agregado 5
- 3.7 Generar nivel de agregado 6
- 3.8 Cargar niveles de agregado del 2 al 6

DICIONARIO DE DATOS

En esta sección se presentan las definiciones de los flujos y almacenes de datos del diagrama de flujo mostrado arriba.

FLUJOS

NOMBRE DEL FLUJO	DEFINICION
Nombre de archivo plano	* Identificación del archivo que contiene datos a nivel de agregado 1
Por Cajero	* Contiene datos a nivel de Cajero: es el máximo detalle
Por Ubicación	* Datos a nivel de Ubicación
Por Sector	* Datos a nivel de Sector
Por Plaza	* Datos a nivel de Plaza
Por Región	* Datos a nivel de Región
Por Banco	* Datos a nivel de Banco
Nombre del archivo plano nivel 2	* Identificación del archivo que contiene datos agrupados por Ubicación
Nombre del archivo plano nivel 3	* Identificación del archivo que contiene datos agrupados por Sector
Nombre del archivo plano nivel 4	* Identificación del archivo que contiene datos agrupados por Plaza
Nombre del archivo plano nivel 5	* Identificación del archivo que contiene datos agrupados por Región
Nombre del archivo plano nivel 6	* Identificación del archivo que contiene datos agrupados por Banco
Registros con niveles de agregado del 2 al 8	* Identificación de los niveles de agregado

ALMACENES DE DATOS

NOMBRE DEL ALMACEN	DESCRIPCION
Archivo plano Geografía	* Archivo que contiene el máximo detalle
Archivo plano nivel 2	* Archivo que contiene los registros agrupados por Ubicación
Archivo plano nivel 3	* Archivo que contiene los registros agrupados por Sector
Archivo plano nivel 4	* Archivo que contiene los registros agrupados por Plaza
Archivo plano nivel 5	* Archivo que contiene los registros agrupados por Región
Archivo plano nivel 6	* Archivo que contiene los registros agrupados por Banco
Dimensión Geografía	* Tabla de Informix que contiene los datos de la dimensión correspondiente

ESPECIFICACIONES DE PROCESO

3.1 Preparar dimensión Geografía

INICIO DE PROGRAMA
RESPALDAR REGISTROS DE TABLA Geografía
BORRAR REGISTROS DE TABLA Geografía

3.2 Cargar nivel de agregado 1

CARGAR DE Archivo_Plano_Geografía
INSERTAR EN Geografía

3.3 Generar nivel de agregado 2

DESCARGAR EN Archivo_Plano_Nivel2
SELECCIONAR REGISTROS DISTINTOS 2,cve_banco,cve_region,cve_plaza,"cve_ubicacion,cve_sector,
des_banco,des_region,des_plaza,"des_ubicacion, des_sector
DE dimensión geografía

3.4 Generar nivel de agregado 3

DESCARGAR EN Archivo_Plano_Nivel3
SELECCIONAR REGISTROS DISTINTOS 3,cve_banco,cve_region,cve_plaza,"cve_sector,
des_banco,des_region,des_plaza,"des_sector
DE dimensión geografía

3.5 Generar nivel de agregado 4

DESCARGAR EN Archivo_Plano_Nivel4
SELECCIONAR REGISTROS DISTINTOS 4,cve_banco,cve_region,cve_plaza,"des_banco,des_region, des_plaza,"
DE dimensión geografía

3.6 Generar nivel de agregado 5

DESCARGAR EN Archivo_Plano_Nivel5
SELECCIONAR REGISTROS DISTINTOS 5,cve_banco,cve_region,"des_banco,des_region,"
DE dimensión geografía

3.7 Generar nivel de agregado 6

DESCARGAR EN Archivo_Plano_Nivel6
SELECCIONAR REGISTROS DISTINTOS 6,cve_banco,"des_banco,"
DE dimensión geografía

3.8 Cargar niveles de agregado del 2 al 6

CARGAR REGISTROS DE Archivo_Plano_Nivel2
INSERTAR EN dimensión geografía
CARGAR REGISTROS DE Archivo_Plano_Nivel2
INSERTAR EN dimensión geografía
CARGAR REGISTROS DE Archivo_Plano_Nivel3
INSERTAR EN dimensión geografía
CARGAR REGISTROS DE Archivo_Plano_Nivel4
INSERTAR EN dimensión geografía
CARGAR REGISTROS DE Archivo_Plano_Nivel5
INSERTAR EN dimensión geografía
CARGAR REGISTROS DE Archivo_Plano_Nivel6
INSERTAR EN dimensión geografía

En la figura 3.8 se muestra el diagrama de flujo de datos de los procesos de carga de las demás dimensiones (excepto la dimensión tiempo).

DIAGRAMA DE FLUJO

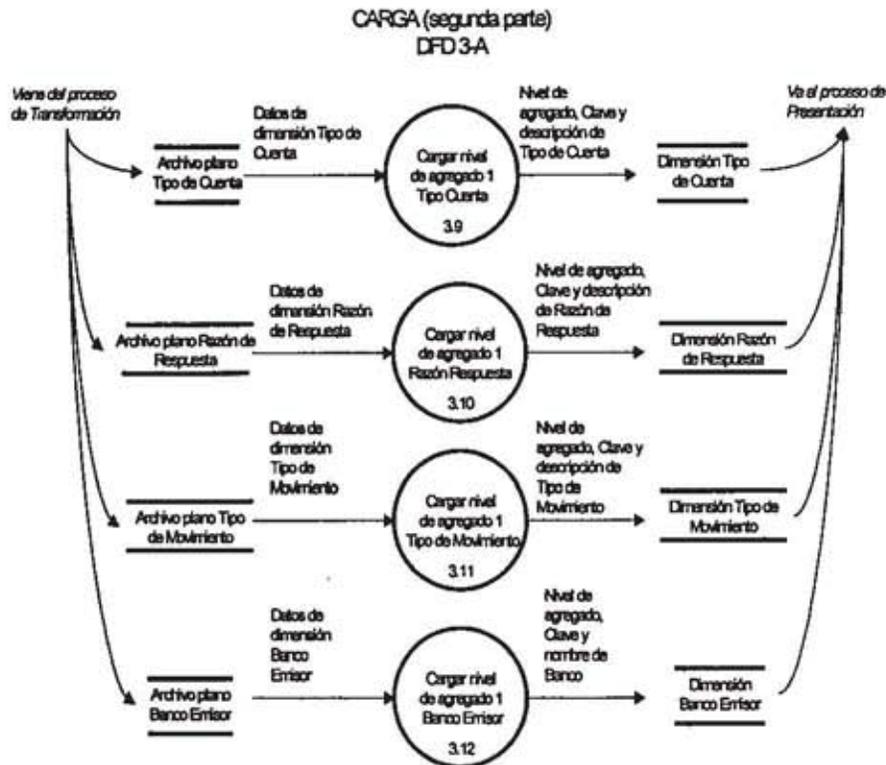


Figura 3.8 Proceso de carga de dimensiones

Las dimensiones Tipo de Cuenta, Razón de Respuesta, Tipo de Movimiento y Banco Emisor solamente tienen el nivel de agregado 1 por lo que se carga el archivo plano generado en el proceso de Transformación directamente a cada dimensión.

- Los subprocesos que componen a este proceso son:
- 3.9 Cargar nivel de agregado 1 Tipo de Cuenta
 - 3.10 Cargar nivel de agregado 1 Razón de Respuesta
 - 3.11 Cargar nivel de agregado 1 Tipo de Movimiento
 - 3.12 Cargar nivel de agregado 1 Banco Emisor

DICCIONARIO DE DATOS

En esta sección se presentan las definiciones de los flujos y almacenes de datos del diagrama mostrado arriba.

FLUJOS

NOMBRE DEL FLUJO	DEFINICION
Datos de dimensión Tipo de Cuenta	* Datos del archivo plano correspondiente al tipo de cuenta
Datos de dimensión Razón de Respuesta	* Datos del archivo plano correspondiente a la razón de respuesta
Datos de dimensión Tipo de Movimiento	* Datos del archivo plano correspondiente al tipo de movimiento
Datos de dimensión Banco Emisor	* Datos del archivo plano correspondiente al banco emisor
Nivel de agregado, clave y descripción de tipo de cuenta	* Registros que contienen los datos correspondientes al tipo de cuenta
Nivel de agregado, clave y descripción de razón de respuesta	* Registros que contienen los datos correspondientes a la razón de respuesta
Nivel de agregado, clave y descripción de tipo de movimiento	* Registros que contienen los datos correspondientes al tipo de movimiento
Nivel de agregado, clave y nombre de banco	* Registros que contienen los datos correspondientes al banco

ALMACENES DE DATOS

NOMBRE DEL ALMACEN	DESCRIPCION
Archivo plano tipo de cuenta	* Archivo obtenido del catálogo de tipos de cuenta en el proceso de transformación
Archivo plano razón de respuesta	* Archivo obtenido del catálogo de razón de respuesta en el proceso de transformación
Archivo plano tipo de movimiento	* Archivo obtenido del catálogo de tipos de movimiento en el proceso de transformación
Archivo plano banco emisor	* Archivo obtenido del catálogo de bancos en el proceso de transformación
Dimensión tipo de cuenta	* Dimensión dentro del modelo
Dimensión razón de respuesta	* Dimensión dentro del modelo
Dimensión tipo de movimiento	* Dimensión dentro del modelo
Dimensión banco emisor	* Dimensión dentro del modelo

ESPECIFICACIONES DE PROCESO

3.9 Cargar nivel de agregado 1 Tipo de Cuenta

CARGAR REGISTROS DE Archivo_Plano_Tipo_Cuenta
 INSERTAR EN dimensión tipo_cuenta

3.10 Cargar nivel de agregado 1 Razón de Respuesta

CARGAR REGISTROS DE Archivo_Plano_Razón_Respuesta
 INSERTAR EN dimensión razón_respuesta

3.11 Cargar nivel de agregado 1 Tipo de Movimiento

CARGAR REGISTROS DE Archivo_Plano_Tipo_Movimiento
 INSERTAR EN dimensión tipo_movimiento

3.12 Cargar nivel de agregado 1 Banco Emisor

CARGAR REGISTROS DE Archivo_Plano_Banco_Emisor
 INSERTAR EN dimensión banco_emisor

En la figura 3.9 se muestra el diagrama de flujo de datos de la carga de la tabla de hechos.

DIAGRAMA DE FLUJO

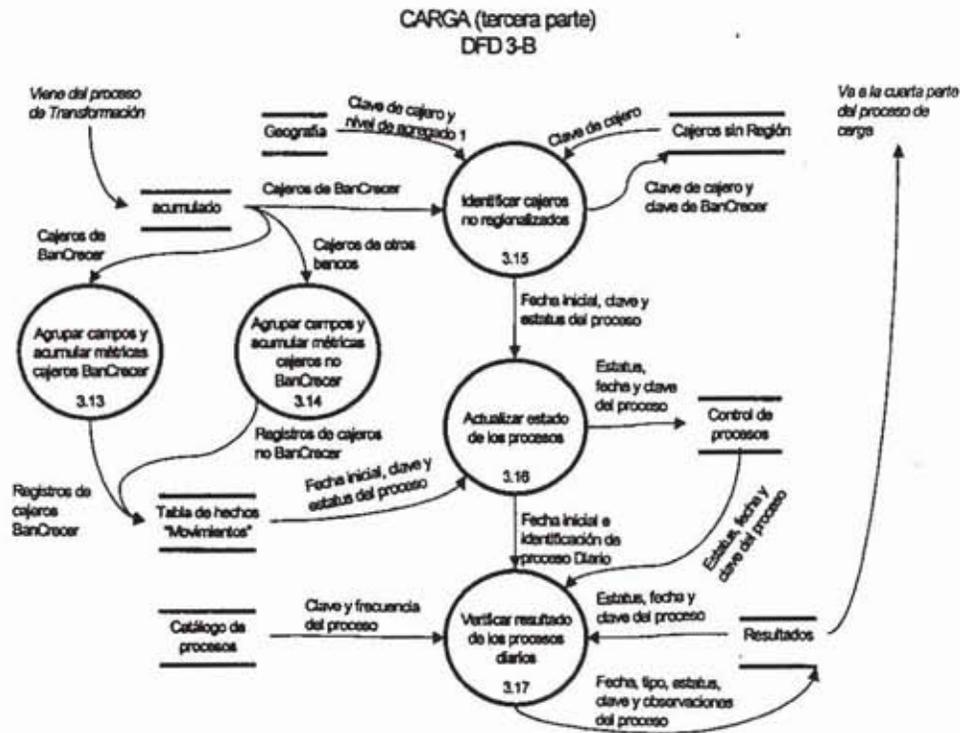


Figura 3.9 Carga de la tabla de hechos

La tabla de hechos se carga después de haber cargado las dimensiones y consiste básicamente de agrupar y sumarizar los registros obtenidos en el proceso de Transformación. Una vez cargados los registros se generan los agregados y los costos de la tabla de hechos. Cuando es fin de quincena se actualiza la dimensión tiempo identificando esa quincena como el periodo actual. En el caso de que sea fin de mes se

CAPITULO 3. DISEÑO DE LA SOLUCION

procede a respaldar y depurar los registros de la tabla de hechos del mes y año anterior, o sea el más antiguo.

Los subprocesos que componen a este proceso son:

- 3.13 Agrupar campos y acumular métricas cajeros BanCrecer
- 3.14 Agrupar campos y acumular métricas cajeros no BanCrecer
- 3.15 Identificar cajeros no regionalizados
- 3.16 Actualizar estado de los procesos
- 3.17 Verificar resultado de los procesos diarios

DICCIONARIO DE DATOS

En esta sección se presentan las definiciones de los flujos y almacenes de datos del diagrama mostrado arriba.

FLUJOS

NOMBRE DEL FLUJO	DEFINICION
Cajeros de BanCrecer	Registros de la tabla "acumulado" cuyo Banco_Id = 'B161'
Cajeros de otros bancos	Registros de la tabla "acumulado" cuyo Banco_Id <> 'B161'
Clave de cajero y clave de BanCrecer	Identificación de los cajeros cuyo Banco_Id = 'B161'
Clave de Cajero	Identificación de cada cajero
Clave de Cajero y nivel de agregado 1	Identificación de cada cajero del máximo nivel de detalle
Registros de cajeros BanCrecer	Conjunto de datos de cajeros agrupados y acumulados cuyo Banco_Id = 'B161'
Registros de cajeros no BanCrecer	Conjunto de datos de cajeros agrupados y acumulados cuyo Banco_Id <> 'B161'
Fecha inicial, clave y estatus del proceso	Datos para identificar el estado de los procesos
Fecha inicial e identificación de proceso Diario	Identificación del rango inicial de fechas y del tipo de proceso a registrar
Clave y frecuencia del proceso	Datos para identificar a los procesos del catálogo correspondiente
Fecha, tipo, estatus, clave y observaciones del proceso	Datos resultantes de la validación de los procesos diarios.

ALMACENES DE DATOS

NOMBRE DEL ALMACEN	DESCRIPCION
Acumulado	Tabla resultante del proceso de transformación
Tabla de hechos movimientos	Tabla que contiene las métricas del modelo
Geografía	Tabla que contiene los datos de la localización de los cajeros
Cajeros sin región	Tabla que contiene los datos de los cajeros de BanCrecer que no tienen identificado a que Región pertenecen
Catálogo de procesos	Tabla que contiene los tipos de procesos de back-end que se realizan en el CICA
Control de procesos	Contiene el resultado de cada subproceso que conforma a los procesos diarios, quincenales y mensuales
Resultados	Tabla que contiene el resultado final de cada proceso

ESPECIFICACIONES DE PROCESO

- 3.13 Agrupar campos y acumular métricas cajeros BanCrecer

INSERTAR EN movimientos

SELECCIONAR

acumulado.razon_id_acum,tiempo.cve_tiempo,acumulado.cajero_id_acum,acumulado.banco_id_acum,acumulado.banco_ta
rj_id_acum,acumulado.cuenta_id_acum,acumulado.movimiento_id_acum,SUMA(acumulado.aceptadas_num),SUMA(acum
ulado.rechazadas_num),SUMA(acumulado.importe_aceptadas),SUMA(acumulado.importe_rechazadas)

DE bd:cica:acumulado_mc,razon_respuesta,geografia,banco_emisor,cuenta,tiempo,movimiento

DONDE acumulado.banco_id_acum = 'B161' Y
acumulado.fecha_corte >= fecha_inicio Y
acumulado.fecha_corte <= fecha_fin Y
acumulado.razon_id_acum = razon_respuesta.id_razon Y
razon_respuesta.agg_level = 1 Y
acumulado.cajero_id_acum = geografia.cve_cajero Y
acumulado.banco_id_acum = geografia.cve_banco Y
geografia.agg_level = 1 Y
acumulado.banco_tarj_id_acum = banco_emisor.banco_emisor Y
banco_emisor.agg_level = 1 Y
acumulado.cuenta_id_acum = cuenta.cve_cuenta Y
cuenta.agg_level = 1 Y
acumulado.movimiento_id_acum = movimiento.cve_movimiento Y
movimiento.agg_level = 1 Y
fecha_inicio >= tiempo.fecha_inicial Y
fecha_fin <= tiempo.fecha_final Y
tiempo.agg_level = 1

AGRUPAR POR 1,2,3,4,5,6,7

3.14 Agrupar campos y acumular métricas cajeros no BanCrece

INSERTAR EN movimientos

SELECCIONAR

acumulado.razon_id_acum,tiempo.cve_tiempo,acumulado.cajero_id_acum,acumulado.banco_id_acum,acumulado.banco_ta
rj_id_acum,acumulado.cuenta_id_acum,acumulado.movimiento_id_acum,SUMA(acumulado.aceptadas_num),SUMA(acum
ulado.rechazadas_num),SUMA(acumulado.importe_aceptadas),SUMA(acumulado.importe_rechazadas)

DE bd:cica:acumulado_mc,razon_respuesta,geografia,banco_emisor,cuenta,tiempo,movimiento

DONDE acumulado.banco_id_acum = 'B161' Y
acumulado.fecha_corte >= fecha_inicio Y
acumulado.fecha_corte <= fecha_fin Y
acumulado.razon_id_acum = razon_respuesta.id_razon Y
razon_respuesta.agg_level = 1 Y
acumulado.cajero_id_acum = geografia.cve_cajero Y
acumulado.banco_id_acum = geografia.cve_banco Y
geografia.agg_level = 1 Y
acumulado.banco_tarj_id_acum = banco_emisor.banco_emisor Y
banco_emisor.agg_level = 1 Y
acumulado.cuenta_id_acum = cuenta.cve_cuenta Y
cuenta.agg_level = 1 Y
acumulado.movimiento_id_acum = movimiento.cve_movimiento Y
movimiento.agg_level = 1 Y
fecha_inicio >= tiempo.fecha_inicial Y
fecha_fin <= tiempo.fecha_final Y
tiempo.agg_level = 1

AGRUPAR POR 1,2,3,4,5,6,7

3.15 Identificar cajeros no regionalizados

CONTAR número de registros del catálogo de Cajeros de BanCreccr
QUE no estén en la dimensión Geografía

SI número de registros <= 0 ENTONCES
INSERTAR EN cajeros_no_region cajero_id, 'B161', Fecha Inicial
FINSI

3.16 Actualizar estado de los procesos

/* Al terminar los procesos 3.13 y 3.14 realizar lo siguiente:

CONTAR registros de control_procesos de la fecha inicial

SI número de registros <= 0 ENTONCES

INSERTAR EN control_procesos los valores clave de proceso, fecha de proceso, estatus del proceso
SINO

ACTUALIZAR los campos estatus y clave de proceso de control_procesos

DONDE fecha de proceso = fecha inicial

FINSI

3.17 Verificar resultado de los procesos diarios

SI proceso es diario ENTONCES

SI no existen datos del día en control de procesos ENTONCES

SI existen resultados del día ENTONCES

ACTUALIZAR resultados con tipo de proceso, estatus de proceso, clave de proceso y observaciones del día

SINO

INSERTAR resultados con tipo de proceso, estatus de proceso, clave de proceso y observaciones del día

FINSI

SINO

SI el estatus del día en control de procesos <> "S" ENTONCES

SI existen resultados del día ENTONCES

ACTUALIZAR resultados con tipo de proceso, "N", clave de proceso y observaciones del día

SINO

INSERTAR resultados con tipo de proceso, "N", clave de proceso y observaciones del día

FINSI

SINO

SI existen resultados del día ENTONCES

ACTUALIZAR resultados con tipo de proceso, "S", clave de proceso y observaciones del día

SINO

INSERTAR resultados con tipo de proceso, "S", clave de proceso y observaciones del día

FINSI

FINSI

En la figura 3.10 se muestra el diagrama de flujo de datos del proceso de carga de la dimensión Tiempo.

DIAGRAMA DE FLUJO

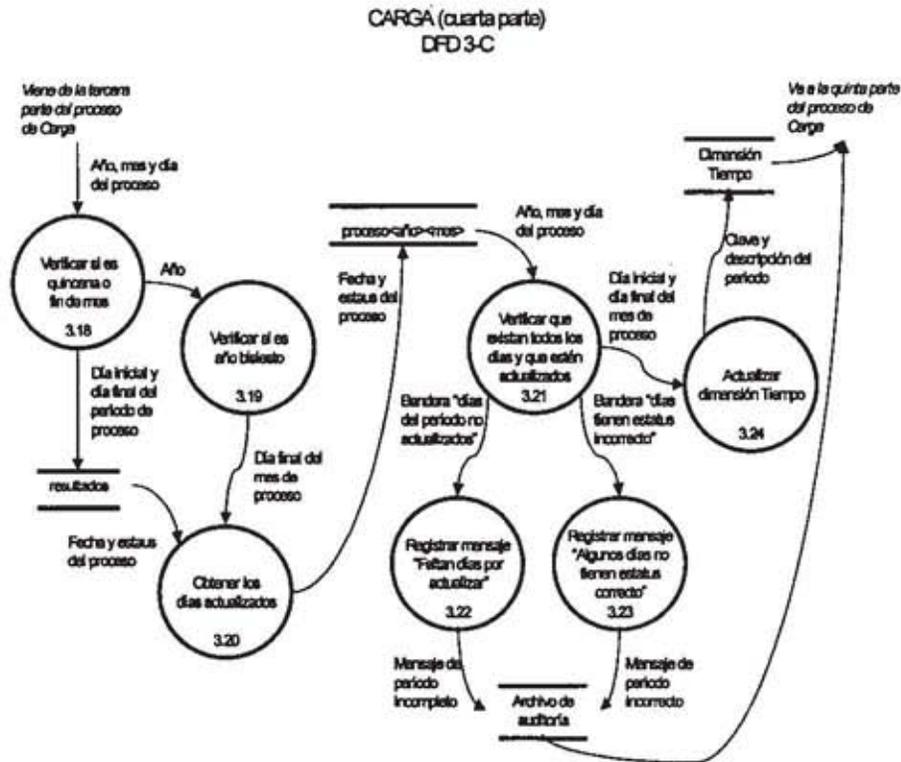


Figura 3.10 Proceso de carga de la dimensión Tiempo.

Para la dimensión tiempo el proceso de carga implica solamente una actualización de la tabla para identificar el periodo actual, o sea el más recientemente cargado.

Los subprocesos que componen a este proceso son:

- 3.18 Verificar si es quincena o fin de mes
- 3.19 Verificar si es año bisiesto
- 3.20 Obtener los días actualizados
- 3.21 Verificar que existan todos los días y que estén actualizados
- 3.22 Registrar mensaje "Faltan días por actualizar"
- 3.23 Registrar mensaje "Algunos días no tienen estatus correcto"
- 3.24 Actualizar dimensión Tiempo

DICIONARIO DE DATOS

En esta sección se presentan las definiciones de los flujos y almacenes de datos del diagrama de flujo mostrado arriba.

FLUJOS

NOMBRE DEL FLUJO	DEFINICION
Año, mes y día del proceso	Datos del proceso diario
Día inicial y día final del periodo de proceso	Identificación del día inicial y día final de la quincena o el mes
Día final del mes de proceso	Si es año bisiesto, este día final reemplaza al día final del flujo anterior
Fecha y estatus del proceso	Por cada día se tiene registrado el resultado (estatus = [S N]) de la carga
Bandera "días del periodo no actualizados"	Identificador de que algunos días del periodo no están actualizados
Bandera "días tienen estatus incorrecto"	Identificador de que algunos días tienen estatus incorrecto
Mensaje de periodo incompleto	Se escribe la leyenda "Algunos días no están cargados" en el archivo de auditoría
Mensaje de periodo incorrecto	Se escribe la leyenda "Algunos días no están correctos" en el archivo de auditoría
Clave y descripción del periodo	Clave = [Y P Blanco]. Y: Identifica el periodo actual. P: Identifica los periodos procesados. Blanco: Indica que no se ha procesado el periodo. Descripción del periodo = ["Periodo Actual" "Procesado" Blanco]

ALMACENES DE DATOS

NOMBRE ALMACEN	DEL	DESCRIPCION
resultados		Por cada día tiene registrado el resultado (estatus = [S N]) de la carga
Proceso-<ano>-<mes>		Archivo plano para validar si están cargados correcta y completamente todos los días del periodo
Archivo de auditoría		Archivo que registra el resultado de la verificación de que todos los días estén completos y correctos.
Dimensión Tiempo		Tabla que contiene los datos de la dimensión Tiempo del modelo de datos.

ESPECIFICACIONES DE PROCESO

3.18 Verificar si es quincena o fin de mes

SI día <= 15 ENTONCES

DiaInicial = 01

DiaFinal = 15

SINO

DiaInicial = 16

CASO mes EN

01) DiaFinal =31

02) DiaFinal =28

03) DiaFinal =31

04) DiaFinal =30

05) DiaFinal =31

06) DiaFinal =30

07) DiaFinal =31

08) DiaFinal =31

09) DiaFinal =30

10) DiaFinal =31

11) DiaFinal =30

12) DIAF=31,;

FINCASO

FINSI

3.19 Verificar si es año bisiesto

Residuo = módulo(año,4)

```

SI (Residuo = 0 AND Mes = 2) ENTONCES
  DiaFinal = 29
FINSI

```

3.20 Obtener los días actualizados

```

DESCARGAR EN proceso<año><mes>
  SELECCIONAR fecha_proceso, estatus_proceso DE resultados
  DONDE AÑO(fecha_proceso)=año
  Y MES(fecha_proceso)=mes
  Y DIA(fecha_proceso)>=DiaInicial

```

3.21 Verificar que existan todos los días y que estén actualizados

```

DiaAux = DiaInicial
MIENTRAS DiaAux <= DiaFinal HACER
  DiaFecha=Año/Mes/DiaAux
  SI (no existe EN archivo proceso<año><mes> datos de DiaFecha) ENTONCES
    TODOS="N"
    SALIR
  FINSI
  SI (proceso<año><mes> de DiaFecha TIENE "N") ENTONCES
    ACTUAL="N"
  FINSI
  DiaAux = DiaAux + 1
FINMIENTRAS

```

3.22 Registrar mensaje "Faltan días por actualizar"

```

SI (TODOS = "N" ) ENTONCES
  ESCRIBE EN archivo de auditoria "FALTA(N) UN(OS) DIA(S) DE ACTUALIZAR"
FINSI

```

3.23 Registrar mensaje "Algunos días no tienen estatus correcto"

```

SI (ACTUAL = "N" ) ENTONCES
  ESCRIBE EN archivo de auditoria "ALGUN(OS) DIA(S) NO TIENEN ESTATUS S"
FINSI

```

3.24 Actualizar dimensión Tiempo

```

ACTUALIZAR tiempo
  DEFINIR cve_periodo='P',
          desc_periodo='Procesado'
  DONDE cve_periodo='Y' Y desc_periodo='Periodo Actual'

```

```

ACTUALIZAR tiempo
  DEFINIR cve_periodo='Y',
          desc_periodo='Periodo Actual'
  DONDE FechaInicial >= fecha_inicial
  Y FechaFinal <= fecha_final

```

En la figura 3.11 se muestra el diagrama de flujo de datos de la quinta parte del proceso de carga.

DIAGRAMA DE FLUJO

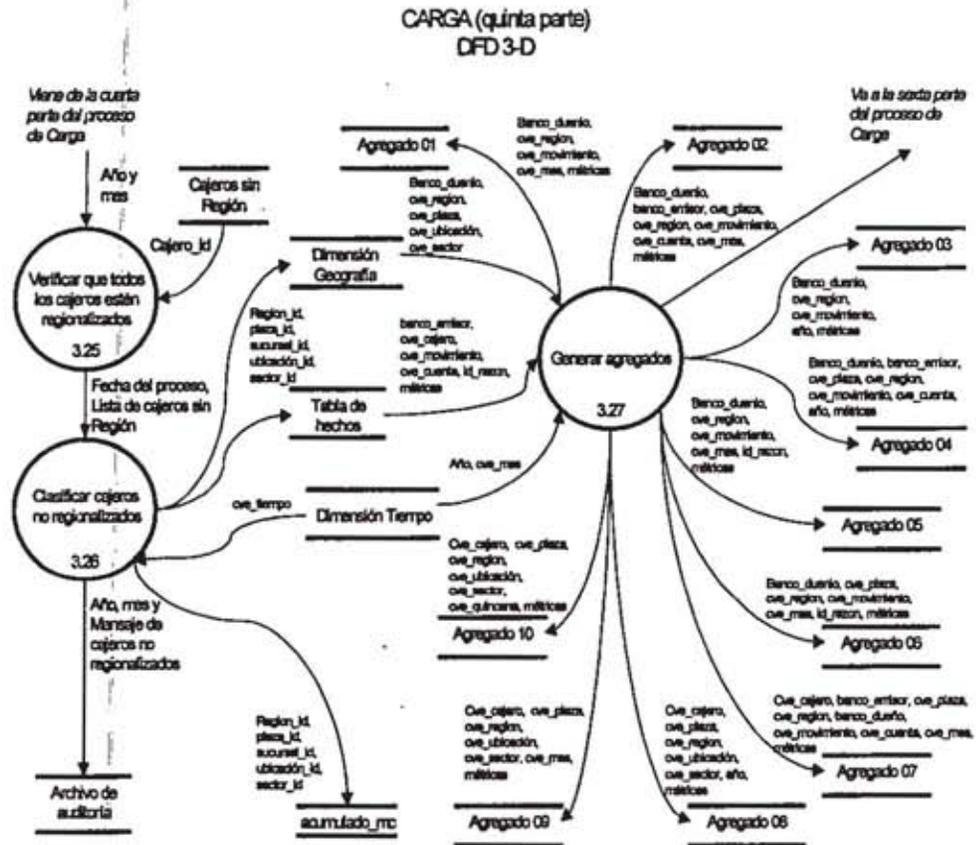


Figura 3.11 Proceso de generación de agregados

Los subprocesos que componen a este proceso son:

- 3.25 Verificar que todos los cajeros estén regionalizados
- 3.26 Clasificar cajeros no regionalizados
- 3.27 Generar agregados

FLUJOS

NOMBRE DEL FLUJO	DEFINICION
Año y mes	Datos del proceso mensual
Cajero_id	Clave del cajero obtenida de la tabla de cajeros sin región definida
Lista de cajeros sin región	Conjunto de cajeros de BanCrecer que no tienen asignada una región
Cajero_id, fecha del proceso	Datos necesarios para clasificar los cajeros no regionalizados
Mensaje de cajeros no regionalizados	Mensaje = "HAY CAJEROS QUE NO ESTAN REGIONALIZADOS"
Region_id, plaza_id, sucursal_id, ubicación_id, sector_id	Flujo de datos que se utilizará para identificar a los cajeros que no tengan región definida.
NOTA: Los demás datos que fluyen para el proceso "Generar agregados" son conjuntos formados por los campos de las tablas de hechos, de dimensión Tiempo y de la dimensión Geografía.	

ALMACENES DE DATOS

NOMBRE DEL ALMACEN	DESCRIPCION
Cajeros sin Región	Tabla en donde se registraron los cajeros de BanCrecer sin región
Archivo de auditoría	Archivo que registra los mensajes del proceso correspondiente
Acumulado_mc	Tabla de trabajo para reprocesar los cajeros de BanCrecer que no tengan Región asignada
Dimensión Geografía	Contiene los datos de las ubicaciones de los cajeros BanCrecer
Dimensión Tiempo	Contiene los periodos que se utilizarán para identificar los periodos quincenales
Tabla de hechos	Contiene las métricas
Agregado 01 al 10	Tablas con registros agrupados y sumariados de acuerdo al flujo de datos que reciben

ESPECIFICACIONES DE PROCESO

3.25 Verificar que todos los cajeros estén regionalizados

SI (es última quincena del mes) ENTONCES
 CONTAR número de cajeros de cajeros_no_region
 DONDE status=0
 Y AÑO(fecha_corte)= Año del proceso
 Y MES(fecha_corte)= Mes del proceso

3.26 Clasificar cajeros no regionalizados

SI (número de cajeros > 0) ENTONCES
 ESCRIBIR EN archivo de auditoría "EXISTEN CAJEROS NO REGIONALIZADOS DEL AÑO año del proceso Y MES mes del proceso"

FINSI

MIENTRAS (NO SEA FIN DE lista de cajeros sin Región)

OBTENER siguiente cajero_id en lista

ACTUALIZAR acumulado_mc

DEFINIR region_id_acum="9999",
 plaza_id_acum="9999",
 sucursal_id_acum="9999",
 ubicacion_id_acum=0,
 sector_id_acum="0"

DONDE cajero_id_acum = cajero_id en lista

Y fecha_corte >= Fecha del proceso

ACTUALIZAR geografía

DEFINIR agg_level = 1
 Banco_id = 'B161'
 Region_id = 9999
 Plaza_id = 9999
 Ubicación_id = 0
 Region_id = 0

```

Banco_dsc = 'BANCRECER'
Region_dsc = 'POR REGIONALIZAR'
Plaza_dsc = 'POR REGIONALIZAR'
Nom_cajero = 'POR REGIONALIZAR'
Ubicación_dsc = 'UBICACION DESCONOCIDA'
Sector_dsc = 'SECTOR DESCONOCIDO'
DONDE eve_cajero = cajero_id en lista
INSERTAR EN movimientos
SELECCIONAR razon_id_acum,
cve_tiempo,
cajero_id_acum,
banco_id_acum,
banco_tarj_id_acum,
cuenta_id_acum,
movimiento_id_acum,
SUMA(aceptadas_num),
SUMA(rechazadas_num),
SUMA(importe_aceptadas),
SUMA(importe_rechazadas)
DE acumulado_mc,
tiempo
DONDE banco_id_acum = 'B161'
Y cajero_id_acum = cajero_id en lista
Y fecha_corte >= Fecha del proceso
Y agg_level = 1
FINMIENTRAS
    
```

3.27 Generar agregados

<pre> INSERTAR EN Agregado 01 SELECCIONAR cve_region, banco_duenio, cve_movimiento, cve_mes, SUMA(mov_aceptado), SUMA(monto_aceptado), SUMA(monto_rechazado), SUMA(mov_rechazado), DE geografia, movimientos, tiempo DONDE cve_cajero DE geografia = cve_cajero DE movimientos Y nivel de agregado DE geografia = 1 Y cve_tiempo DE tiempo = cve_tiempo DE movimientos Y nivel de agregado DE tiempo = 1 Y desc_periodo DE tiempo = 'Periodo Actual' </pre>	<pre> INSERTAR EN Agregado 02 SELECCIONAR banco_duenio, banco_emisor, cve_plaza, cve_region, cve_movimiento, cve_cuenta, cve_mes, SUMA(mov_aceptado), SUMA(monto_aceptado), SUMA(monto_rechazado), SUMA(mov_rechazado), DE movimientos, geografia, tiempo DONDE Cve_cajero DE geografia = cve_cajero DE movimientos Y Nivel de agregado DE geografia = 1 Y cve_tiempo DE tiempo = cve_tiempo DE movimientos Y dbsiglo.tiempo.agg_level = 1 AND nivel de agregado DE tiempo = 1 Y desc_periodo DE tiempo = 'Periodo Actual' </pre>
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

CAPITULO 3. DISEÑO DE LA SOLUCION

<p>INSERTAR EN Agregado 03 SELECCIONAR cve_region, banco_duenio, cve_movimiento, año, SUMA(mov_aceptado), SUMA(monto_aceptado), SUMA(monto_rechazado), SUMA(mov_rechazado), DE geografia , movimientos , tiempo DONDE cve_cajero DE geografia = Cve_cajero DE movimientos Y nivel de agregado DE geografia = 1 Y cve_tiempo DE tiempo = cve_tiempo DE movimientos Y nivel de agregado DE tiempo = 1 Y desc_periodo DE tiempo = 'Periodo Actual'</p>	<p>INSERTAR EN Agregado 04 SELECCIONAR banco_emisor DE movimientos, cve_plaza DE geografia, cve_region DE geografia, banco_duenio DE geografia, cve_movimiento DE movimientos, cve_cuenta DE movimientos, año DE tiempo, SUMA(mov_aceptado), SUMA(monto_aceptado), SUMA(monto_rechazado), SUMA(mov_rechazado), DE movimientos , geografia , tiempo DONDE cve_cajero DE geografia = Cve_cajero DE movimientos Y nivel de agregado DE geografia = 1 Y cve_tiempo DE tiempo = cve_tiempo DE movimientos Y nivel de agregado DE tiempo = 1 Y desc_periodo DE tiempo = 'Periodo Actual'</p>
<p>INSERTAR EN Agregado 05 SELECCIONAR cve_region, banco_duenio, cve_movimiento, cve_mes, id_razon, SUMA(mov_aceptado), SUMA(monto_aceptado), SUMA(monto_rechazado), SUMA(mov_rechazado) DE geografia , movimientos , tiempo DONDE cve_cajero DE geografia = cve_cajero DE movimientos Y nivel de agregado DE geografia = 1 Y cve_tiempo DE tiempo = cve_tiempo DE movimientos Y nivel de agregado DE tiempo = 1 Y desc_periodo DE tiempo = 'Periodo Actual'</p>	<p>INSERTAR EN Agregado 06 SELECCIONAR cve_plaza, cve_region, banco_duenio, cve_movimiento, cve_mes, id_razon, SUMA(mov_aceptado), SUMA(monto_aceptado), SUMA(monto_rechazado), SUMA(mov_rechazado) DE geografia , movimientos , tiempo DONDE cve_cajero DE geografia = cve_cajero DE movimientos Y nivel de agregado DE geografia = 1 Y cve_tiempo DE tiempo = cve_tiempo DE movimientos Y nivel de agregado DE tiempo = 1 Y desc_periodo DE tiempo = 'Periodo Actual'</p>
<p>INSERTAR EN Agregado 07 SELECCIONAR cve_cajero, banco_emisor, cve_plaza, cve_region, banco_duenio, cve_movimiento, cve_cuenta, cve_mes, SUMA(mov_aceptado), SUMA(monto_aceptado), SUMA(monto_rechazado),</p>	<p>INSERTAR EN Agregado 08 SELECCIONAR cve_cajero, cve_plaza, cve_region, cve_ubicacion, cve_sector, año, SUMAR(mov_aceptado), SUMAR(monto_aceptado) DE geografia , movimientos ,</p>

CAPITULO 3. DISEÑO DE LA SOLUCION

<p>SUMA(mov_rechazado) DE geografia, movimientos, tiempo DONDE cve_cajero DE geografia = cve_cajero DE movimientos Y nivel de agregado DE geografia = 1 Y cve_tiempo DE tiempo = cve_tiempo DE movimientos Y nivel de agregado DE tiempo = 1 Y desc_periodo DE tiempo = 'Periodo Actual'</p>	<p>tiempo DONDE cve_cajero DE geografia = cve_cajero DE movimientos Y nivel de agregado DE geografia = 1 Y cve_tiempo DE tiempo = cve_tiempo DE movimientos Y nivel de agregado DE tiempo = 1 Y desc_periodo DE tiempo = 'Periodo Actual'</p>
<p>INSERTAR EN Agregado 09 SELECCIONAR cve_cajero, cve_plaza, cve_region, cve_ubicacion, cve_sector, cve_mes, SUMAR(mov_aceptado), SUMAR(monto_aceptado) DE geografia, movimientos, tiempo DONDE cve_cajero DE geografia = cve_cajero DE movimientos Y nivel de agregado DE geografia = 1 Y cve_tiempo DE tiempo = cve_tiempo DE movimientos Y nivel de agregado DE tiempo = 1 Y desc_periodo DE tiempo = 'Periodo Actual'</p>	<p>INSERTAR EN Agregado 10 SELECCIONAR cve_cajero, cve_plaza, cve_region, cve_ubicacion, cve_sector, cve_quincena, SUMAR(mov_aceptado), SUMAR(monto_aceptado) DE geografia, movimientos, tiempo DONDE cve_cajero DE geografia = cve_cajero DE movimientos Y nivel de agregado DE geografia = 1 Y cve_tiempo DE tiempo = cve_tiempo DE movimientos Y nivel de agregado DE tiempo = 1 Y desc_periodo DE tiempo = 'Periodo Actual'</p>

En la figura 3.12 se muestra el diagrama de flujo de datos de la última parte del proceso de carga.

DIAGRAMA DE FLUJO

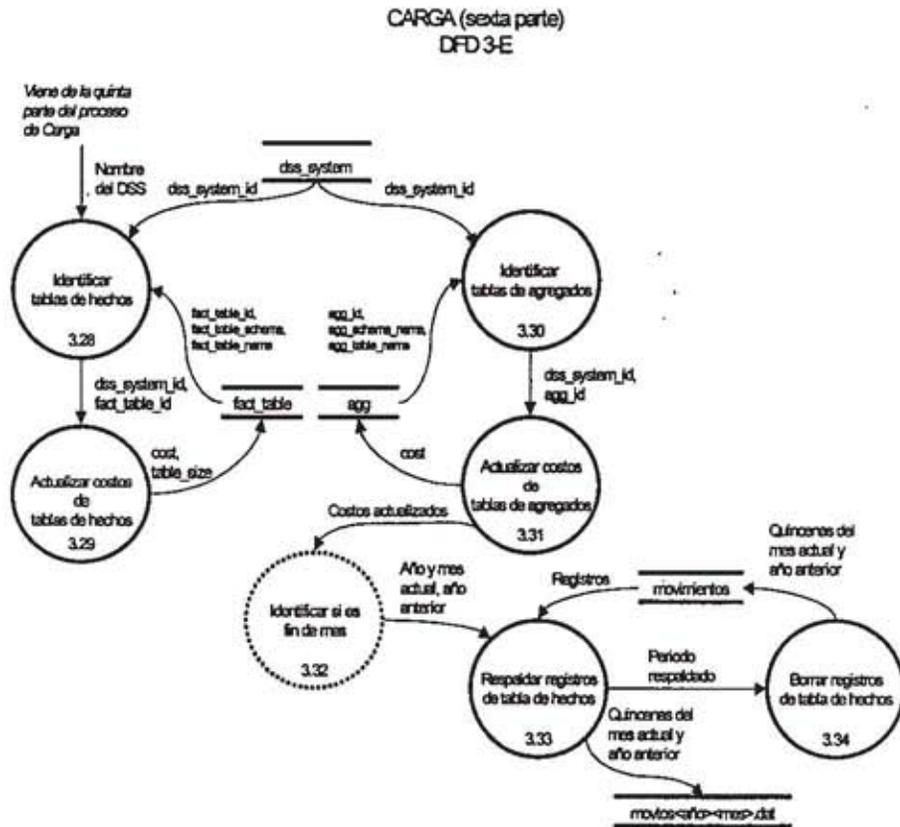


Figura 3.12 Generación de costos y depuración de tabla de hechos

Los subprocesos que componen a este proceso son:

- 3.28 Identificar tablas de hechos
- 3.29 Actualizar costos tablas de hechos
- 3.30 Identificar tablas de agregados
- 3.31 Actualizar costos tablas de agregados
- 3.32 Identificar si es fin de mes
- 3.33 Respalda registros de tabla de hechos
- 3.34 Borrar registros de tabla de hechos

FLUJOS

NOMBRE DEL FLUJO	DEFINICION
Nombre del DSS	Identificación del nombre del modelo del CICA dentro de las tablas de MetaCube
dss_system_id	Identificación de la clave del modelo del CICA dentro de las tablas de MetaCube
fact_table_id, fact_table_schema, fact_table_name	Clave, esquema y nombre de cada tabla de hechos dentro del modelo del CICA
Cost, table_size	cost = Cantidad de registros por cada tabla de hechos
fact_table_id, fact_table_schema, fact_table_name	fact_table_id = Identificador de tabla de hechos fact_table_schema = Esquema de la tabla de hechos fact_table_name = Nombre de la tabla de hechos
agg_id, agg_schema_name, agg_table_name	agg_id = Identificador de tabla de agregados agg_schema_name = Esquema de la tabla de agregados agg_table_name = Nombre de la tabla de agregados
Costos actualizados	Se debe conocer si los costos ya están actualizados para proceder con el respaldo
Año y mes actual, año anterior	Estos datos sirven para realizar el respaldo del mes actual pero del año anterior
Registros	Son los datos de la tabla movimientos que se respaldan
Quincenas del mes actual y año anterior	Se respaldan los datos del año anterior considerando el número de mes actual y se depuran los registros considerando estos criterios
Periodo respaldado	Sirve para saber si el respaldo se realizó correctamente para proceder a depurar

ALMACENES DE DATOS

NOMBRE DEL ALMACEN	DESCRIPCION
dss_system	Tabla de sistema de MetaCube en donde se registran los datos del modelo especificado mediante Warehouse Manager. dss = decision support system
fact_table	Tabla de sistema de MetaCube en donde se registran los datos de las tablas de hechos que pueden existir en diferentes dss
agg	Tabla de sistema de MetaCube en donde se registran los datos de las tablas de agregados que pueden existir en diferentes dss
movtos<año><mes>.dat	Archivo plano que contiene el respaldo de los registros del año anterior de la tabla de hechos
movimientos	Nombre de la tabla de hechos del modelo del CICA, cuyos registros se respaldan y depuran para mantener 13 meses en línea

ESPECIFICACIONES DE PROCESO

3.28 Identificar tablas de hechos

INICIALIZAR PROCESO

```
ASIGNAR dssid=( SELECCIONAR dss_system_id
DE metacube.dss_system
DONDE dss_system_desc = nombre del dss para CICA);
```

INICIALIZAR arreglo_nombres_tablas

```
MIENTRAS (no sea fin de tabla) SELECCIONAR fact_table_id,fact_table_schema,fact_table_name
DE metacube.fact_table
DONDE dss_system_id=dssid
ASIGNAR idtabla = fact_table_id
ASIGNAR nombretabla = fact_table_schema.fact_table_name
INSERTAR nombretabla, idtabla EN arreglo_nombres_tablas
FINMIENTRAS
FIN PROCESO
```

3.29 Actualizar costos tablas de hechos

MIENTRAS (no sea fin de arreglo)

```
LEER nombretabla, idtabla DE arreglo_nombres_tablas
```

```

ACTUALIZAR metacube.fact_table
DEFINIR cost = (SELECCIONAR CONTAR(*) DE tabla)
      table_size = (SELECCIONAR CONTAR(*) DE tabla)
DONDE dss_system_id = dssid;
Y fact_table_id = idtabla
FINMIENTRAS
FIN PROCESO

```

3.30 Identificar tablas de agregados

```

INICIALIZAR arreglo_nombres_tablas
MIENTRAS (no sea fin de tabla) SELECCIONAR agg_id,agg_schema_name,agg_table_name
      DE metacube.agg
      DONDE dss_system_id=dssid
      ASIGNAR idtabla = agg_id
      ASIGNAR nombretabla = agg_table_name
      INSERTAR nombretabla, idtabla EN arreglo_nombres_tablas
FINMIENTRAS
FIN PROCESO

```

3.31 Actualizar costos tablas de agregados

```

MIENTRAS (no sea fin de arreglo)
      LEER nombretabla, idtabla DE arreglo_nombres_tablas
      ACTUALIZAR metacube.agg
      DEFINIR cost = (SELECCIONAR CONTAR(*) DE tabla)
DONDE dss_system_id = dssid;
Y fact_table_id = idtabla
FINMIENTRAS
FIN PROCESO

```

3.32 Identificar si es fin de mes

```

SI (dia inicial del periodo de proceso = 16) ENTONCES
      FIN_MES = Verdadero
SINO
      FIN_MES = Falso
FINSI
FIN PROCESO

```

3.33 Respalda registros de tabla de hechos

```

INICIALIZAR PROCESO
año_respaldo = año_proceso - 1
quincena_inicial = año_respaldo+mes_proceso+1
quincena_final = año_respaldo+mes_proceso+2
DESCARGAR EN movtos<año_respaldo><mes_proceso>.dat
      SELECCIONAR * DE movimientos
      DONDE cve_tiempo >= quincena_inicial
      Y cve_tiempo <= quincena_final
FIN PROCESO

```

3.34 Borrar registros de tabla de hechos

```

SI (EXISTE(movtos<año_respaldo><mes_proceso>.dat)) ENTONCES
      BORRAR DE movimientos
      DONDE cve_tiempo >= quincena_inicial
      Y cve_tiempo <= quincena_final
FINSI
FIN PROCESO

```

3.2.3 PRESENTACION

El proceso de presentación consiste de dos subprocesos: el primero es la definición de las dimensiones y la tabla de hechos dentro del MetaCube Warehouse Manager, el segundo es la presentación que hace en pantalla MetaCube Explorer de las dimensiones y la tabla de hechos al usuario final para realizar sus consultas.

La definición dentro del Warehouse Manager se realiza la primera vez, cuando el modelo dimensional se tiene ya establecido físicamente, con las tablas que se mencionan al principio del este capítulo y se redefinirá en caso de que se necesite darle mantenimiento al modelo.

La presentación al usuario final siempre será la misma hasta que no cambien la tabla de hechos o las dimensiones.

A continuación se presenta la forma en que se define el modelo dentro del Warehouse Manager.

En la figura 3.13 se muestra el esquema general que aparece en pantalla con la primera definición de Dimensiones y Hechos.

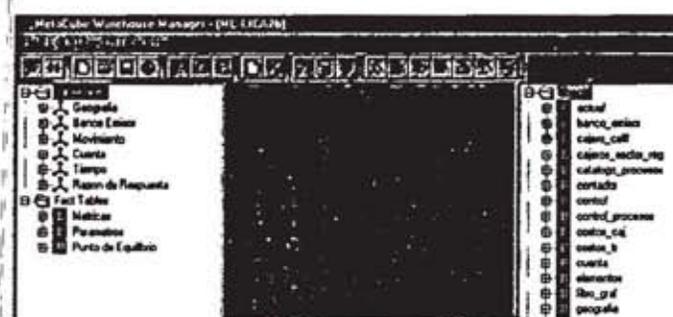


Figura 3.13 Dimensiones y tablas de hechos

En la imagen se muestran del lado izquierdo los nombres de las dimensiones y la sección para definir una o varias tablas de hechos. Del lado derecho se muestra el esquema físico con las tablas correspondientes al modelo dimensional.

3.2.3.1 DIMENSIONES

En esta sección se explica como definir mediante Warehouse Manager cada una de las dimensiones del modelo.

Como se verá en las secciones que se presentan enseguida, la pantalla de definición de una dimensión se compone de tres secciones principales: la sección de la izquierda es la representación lógica de la dimensión y sus elementos, la sección derecha es la representación física de la tabla de la dimensión y sus campos. La sección central es la que hace la relación entre el modelo lógico con las tablas de la base de datos.

En la sección del centro se encuentran: el nombre de la dimensión, las características físicas, la columna que representa la agregación, el tipo de dimensión (de tiempo u otro), la jerarquía dentro de la dimensión y la interfaz de usuario que es el icono que representará a la dimensión cuando aparezca en MetaCube Explorer.

Las dimensiones que se explican a continuación son:

- Geografía
- Banco Emisor
- Movimiento
- Cuenta
- Tiempo
- Razón de respuesta

DIMENSION GEOGRAFIA

En la figura 3.14 se muestra la definición de la dimensión Geografía.

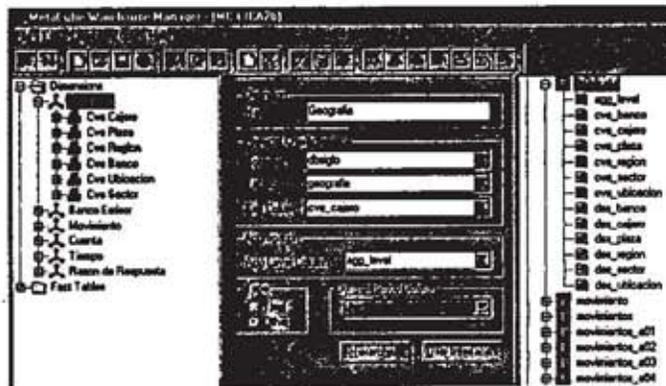


Figura 3.14 Definición de la dimensión Geografía

En la sección izquierda de la figura 3.14 se definen los elementos de la dimensión Geografía que son:

- Cajero
- Plaza
- Región
- Banco
- Ubicación
- Sector

Cada uno de estos elementos tiene un número único que define su lugar dentro de la jerarquía de la dimensión.

En la sección central de la figura 3.14 se muestra la definición de la dimensión, sus características físicas, la columna que define el nivel de agregación, el tipo de dimensión, la jerarquía y la interfaz de usuario.

En la sección derecha de la figura 3.14 se muestra el esquema físico de la tabla que define la dimensión.

Para esta dimensión la jerarquía se define mediante el botón "Hierarchy" como se muestra en la figura 3.14.a

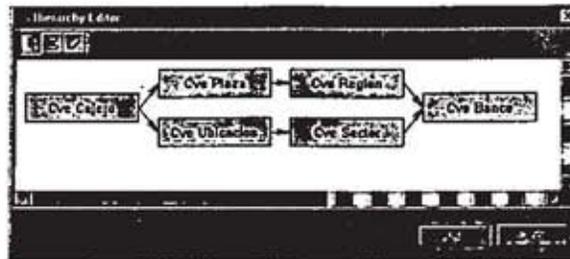


Figura 3.14.a Jerarquía de la dimensión Geografía

La interfaz de usuario se define mediante el botón "User Interface" y es como se muestra en la figura 3.14.b.



Figura 3.14.b Interfaz de usuario de la dimensión Geografía

A continuación se presenta la definición de los elementos de la dimensión Geografía.

CAJERO

En la figura 3.15 se muestra la definición del elemento Cajero de la dimensión Geografía.

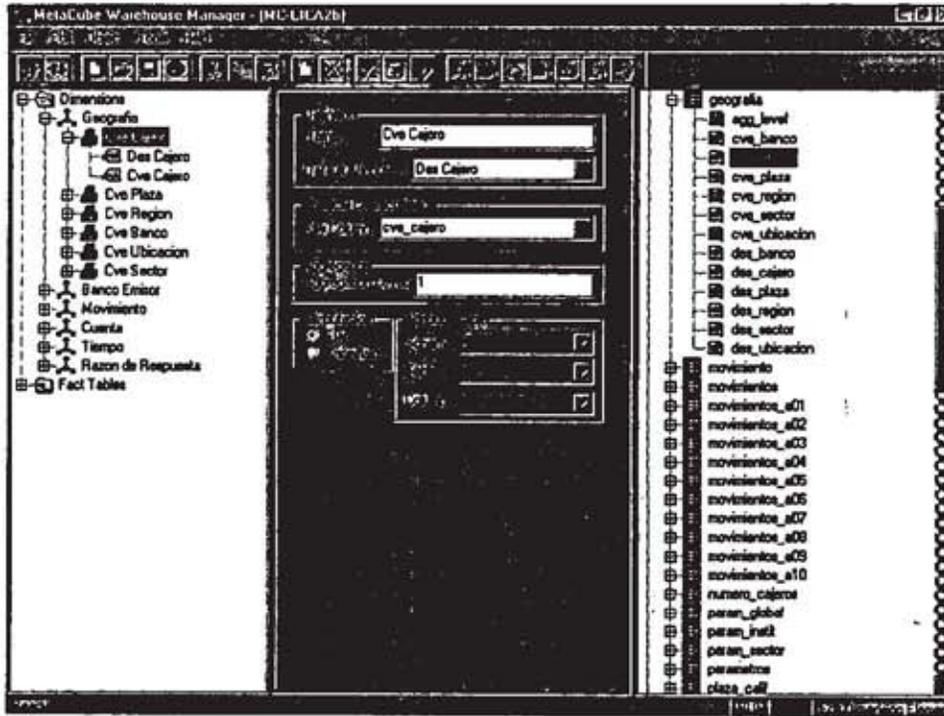


Figura 3.15 Definición del elemento Cajero de la dimensión Geografía

La definición del elemento de dimensión incluye: su nombre y atributo por defecto, la columna llave, su nivel de agregado dentro de la jerarquía de la dimensión y si tiene un esquema de estrella o de copo de nieve.

Los atributos del elemento de dimensión son los que se muestran en las figuras 3.15.a y 3.15.b que se presentan a continuación.

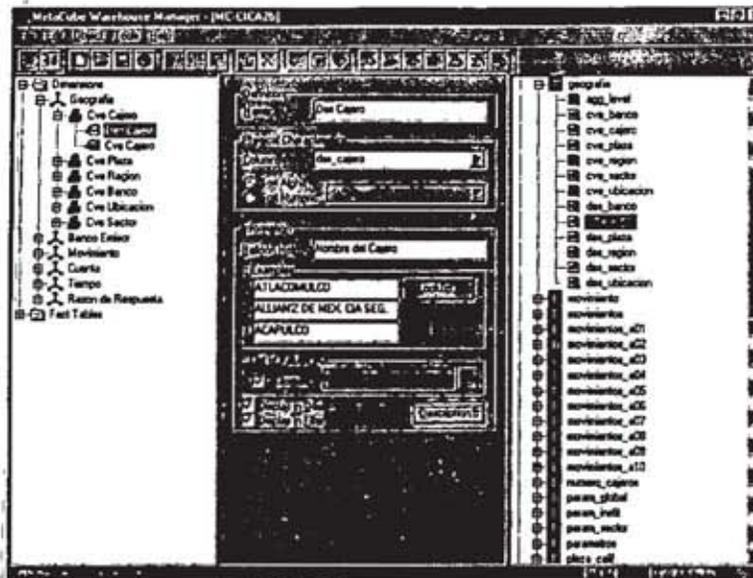


Figura 3.15.a Atributo para la descripción del Cajero

En la figura 3.15.a se muestra que el campo Descripción de Cajero estará disponible al momento de hacer consultas en MetaCube Explorer como lo indica el campo "Display in Query" que está seleccionado.

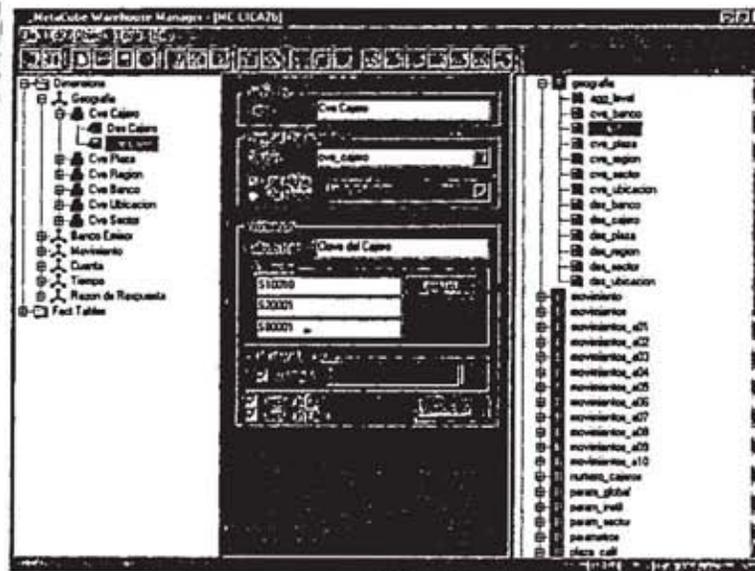


Figura 3.15.b Atributo para la clave de Cajero

En la figura 3.15.b se muestra que el campo Clave de Cajero estará disponible al momento de hacer consultas en MetaCube Explorer como lo indica el campo "Display in Query" que está seleccionado.
PLAZA

En la figura 3.16 se muestra la definición del elemento Plaza de la dimensión Geografía.

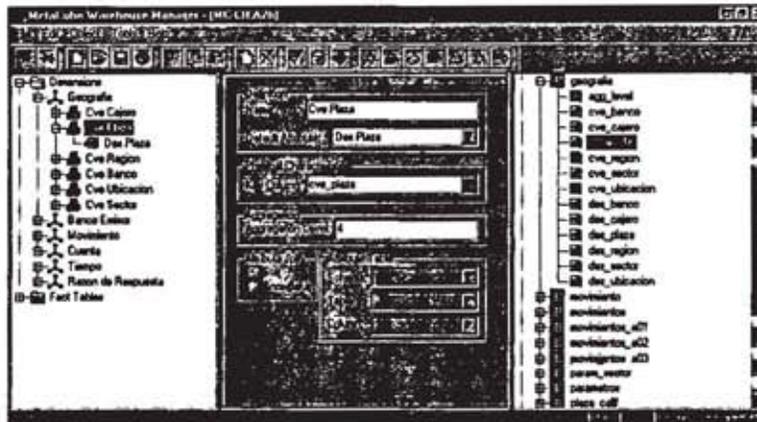


Figura 3.16 Definición del elemento Plaza de la dimensión Geografía

El atributo del elemento de dimensión es el que se muestra en la figura 3.16.a que se presenta a continuación.

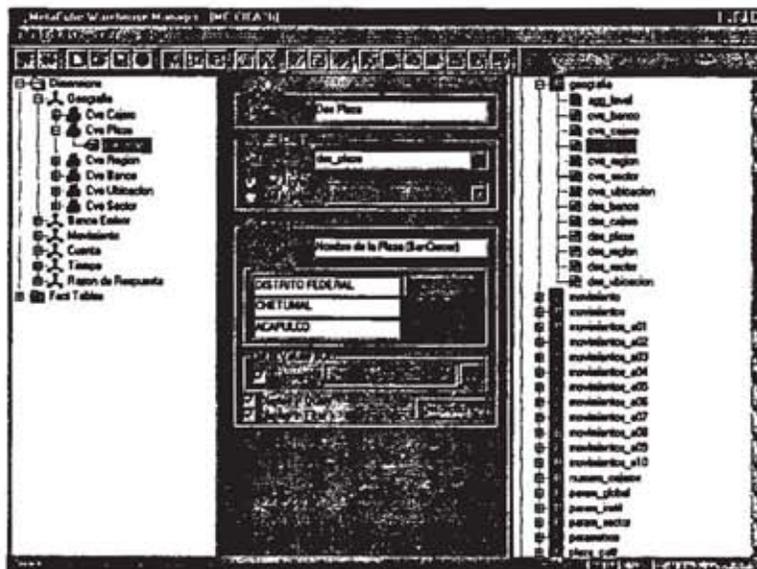


Figura 3.16.a Atributo para la descripción de la Plaza

REGION

En la figura 3.17 se muestra la definición del elemento Región de la dimensión Geografía.

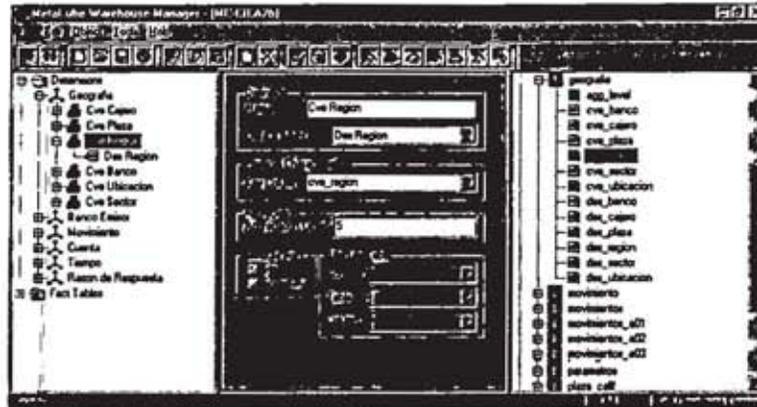


Figura 3.17 Definición del elemento Región de la dimensión Geografía

El atributo del elemento de dimensión es el que se muestra en la figura 3.17.a que se presenta a continuación.

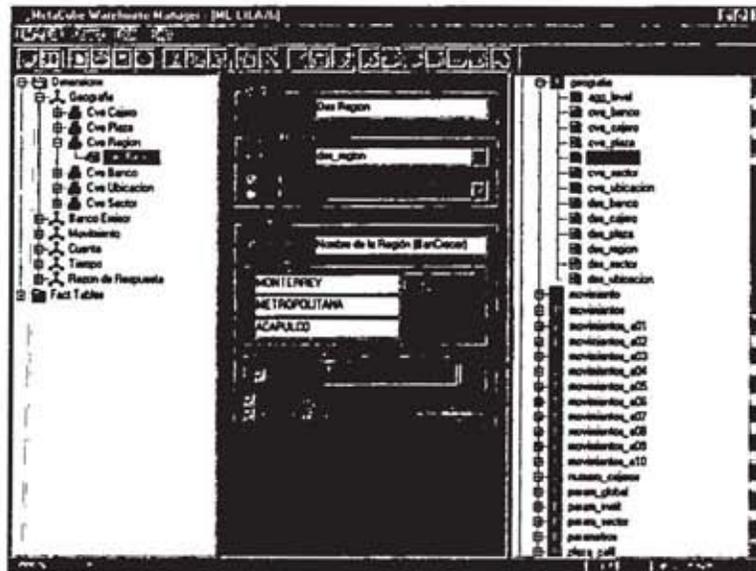


Figura 3.17.a Atributo para la descripción de la Región

BANCO DUEÑO

En la figura 3.18 se muestra la definición del elemento Banco Dueño (Cve Banco) de la dimensión Geografía.

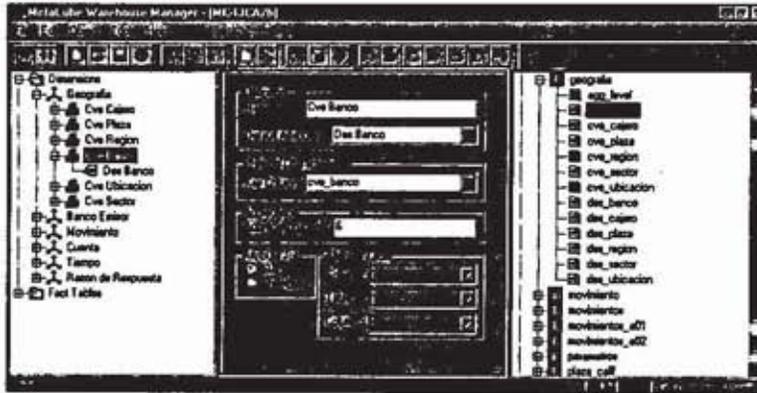


Figura 3.18 Definición del elemento Banco Dueño de la dimensión Geografía

El atributo del elemento de dimensión es el que se muestra en la figura 3.18.a que se presenta a continuación.

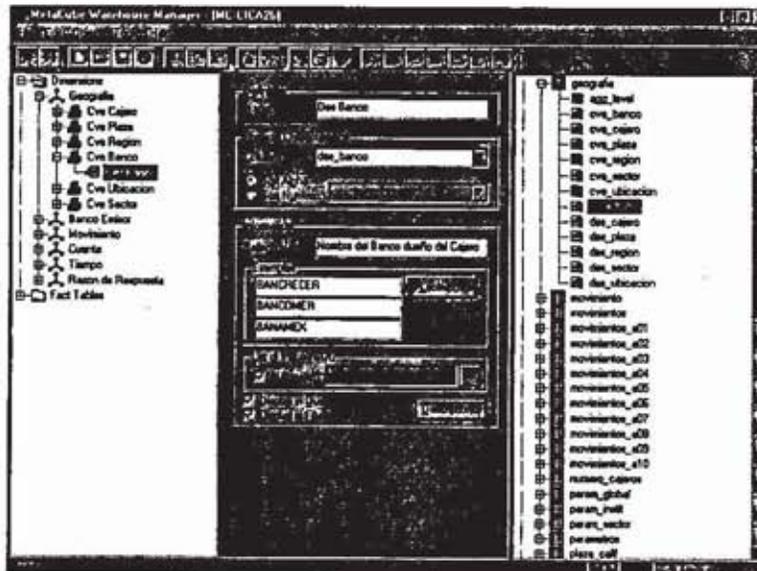


Figura 3.18.a Atributo para la descripción del Banco Dueño

UBICACION

En la figura 3.19 se muestra la definición del elemento Ubicación de la dimensión Geografía.

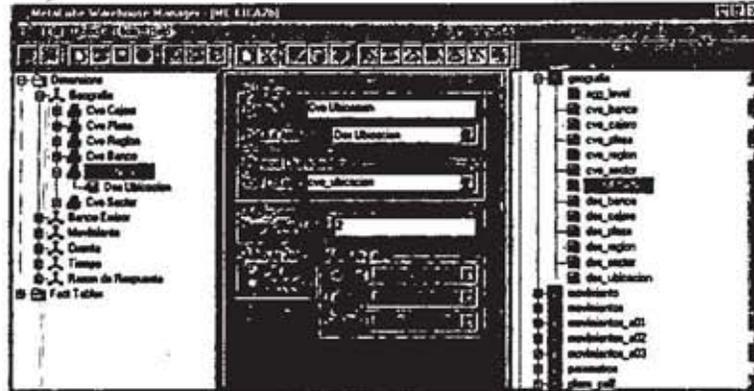


Figura 3.19 Definición del elemento Ubicación de la dimensión Geografía

El atributo del elemento de dimensión es el que se muestra en la figura 3.19.a que se presenta a continuación.

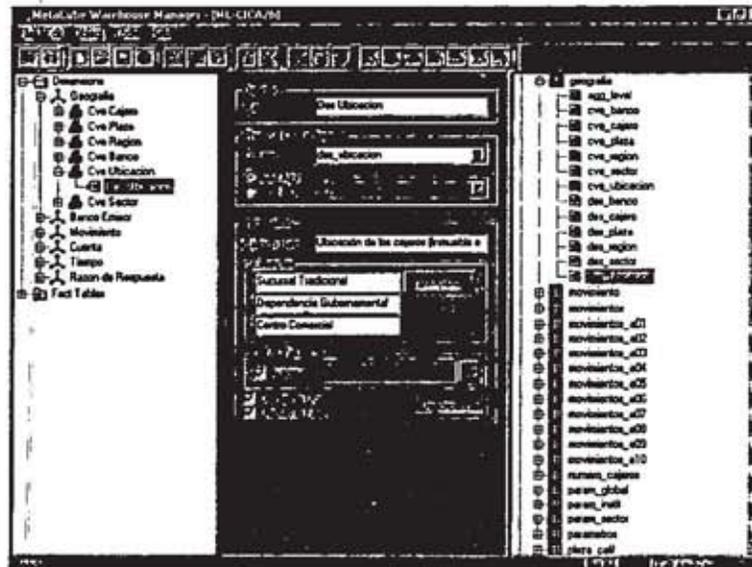


Figura 3.19.a Atributo para la descripción de la Ubicación

SECTOR

En la figura 3.20 se muestra la definición del elemento Sector de la dimensión Geografía.

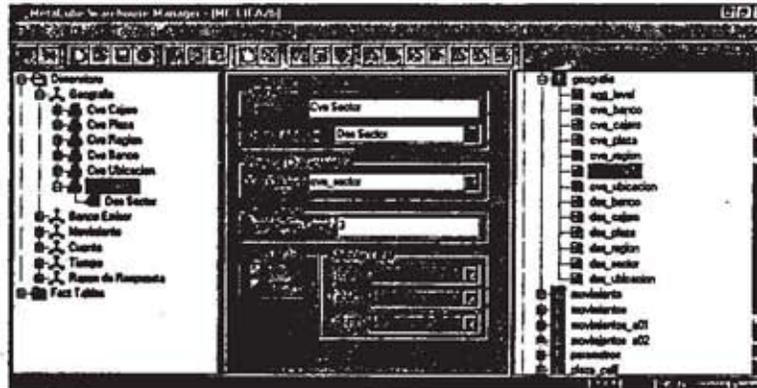


Figura 3.20 Definición del elemento Sector de la dimensión Geografía

El atributo del elemento de dimensión es el que se muestra en la figura 3.20.a que se presenta a continuación.

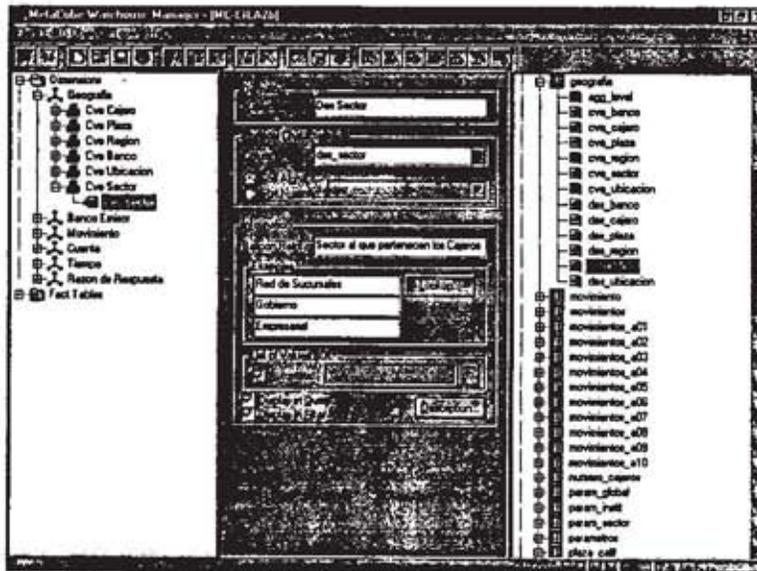


Figura 3.20.a Atributo para la descripción del Sector

DIMENSION BANCO EMISOR

En la figura 3.21 se muestra la definición de la dimensión Banco Emisor.

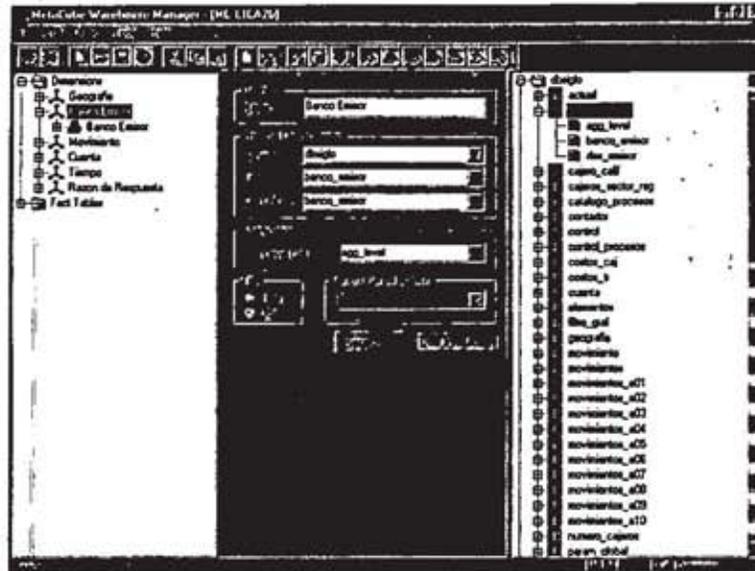


Figura 3.21 Definición de la dimensión Banco Emisor

En la sección izquierda de la figura 3.21 se define el elemento de la dimensión Banco Emisor, el cual tiene un número que define la jerarquía de la dimensión.

En la sección central de la figura 3.21 se muestra la definición de la dimensión, sus características físicas, la columna que define el nivel de agregación, el tipo de dimensión, la jerarquía y la interfaz de usuario.

En la sección derecha de la figura 3.21 se muestra el esquema físico de la tabla que define la dimensión.

Para esta dimensión no se ha definido ninguna jerarquía. En el caso de esta dimensión, la interfaz de usuario es como se muestra en la figura 3.21.a.

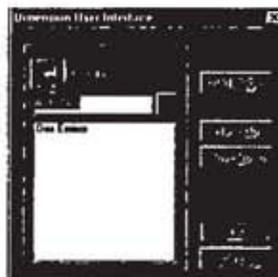


Figura 3.21.a Interfaz de usuario para Banco Emisor

A continuación se presenta el elemento de la dimensión Banco Emisor

BANCO EMISOR

En la figura 3.22 se muestra la definición del elemento Banco Emisor de la dimensión Banco Emisor.

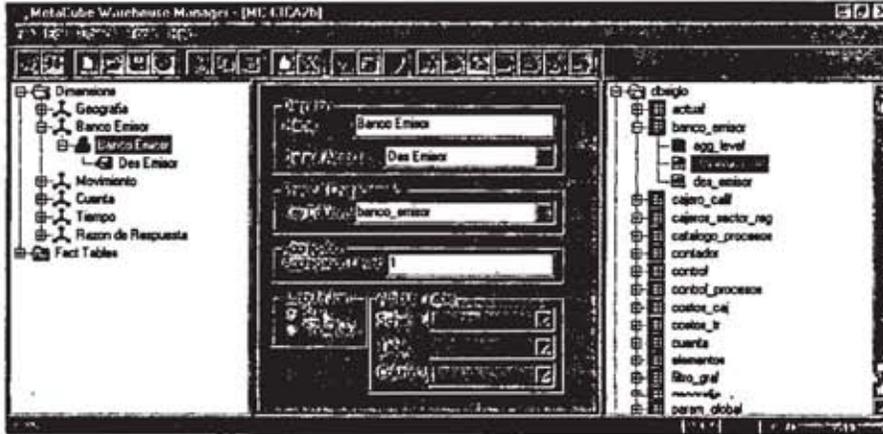


Figura 3.22 Definición del elemento Banco Emisor de la dimensión Banco Emisor

El atributo del elemento de dimensión es el que se muestra en la figura 3.22.a que se presenta a continuación.

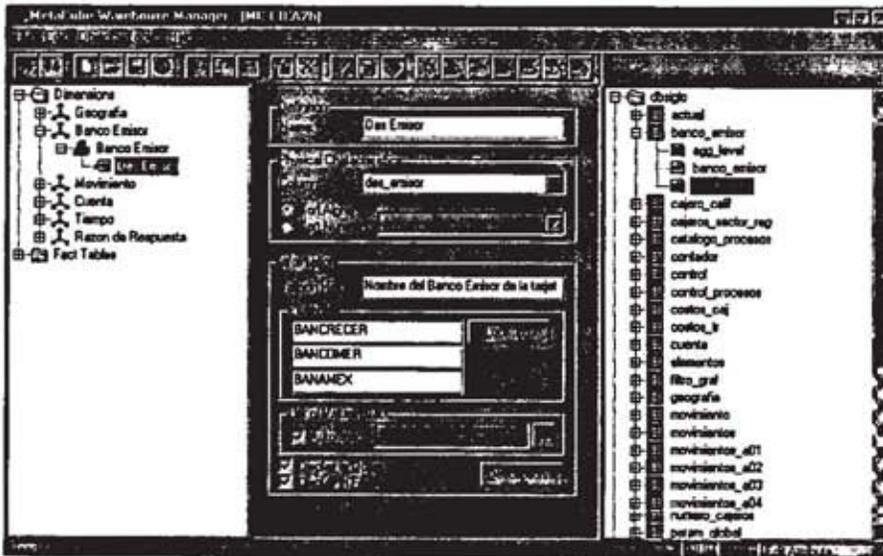


Figura 3.22.a Atributo para la descripción del Banco Emisor

DIMENSION MOVIMIENTO

En la figura 3.23 se muestra la definición de la dimensión Movimiento.



Figura 3.23 Definición de la dimensión Movimiento

En la sección izquierda de la figura 3.23 se define el elemento de la dimensión Movimiento, el cual tiene un número que define la jerarquía de la dimensión.

En la sección central de la figura 3.23 se muestra la definición de la dimensión, sus características físicas, la columna que define el nivel de agregación, el tipo de dimensión, la jerarquía y la interfaz de usuario.

En la sección derecha de la figura 3.23 se muestra el esquema físico de la tabla que define la dimensión.

Para esta dimensión no se ha definido ninguna jerarquía. En el caso de esta dimensión, la interfaz de usuario es como se muestra en la figura 3.23.a.



Figura 3.23.a Interfaz de usuario para Movimiento

A continuación se presenta el elemento de la dimensión Movimiento.

CLAVE MOVIMIENTO

En la figura 3.24 se muestra la definición del elemento Clave Movimiento de la dimensión Movimiento.

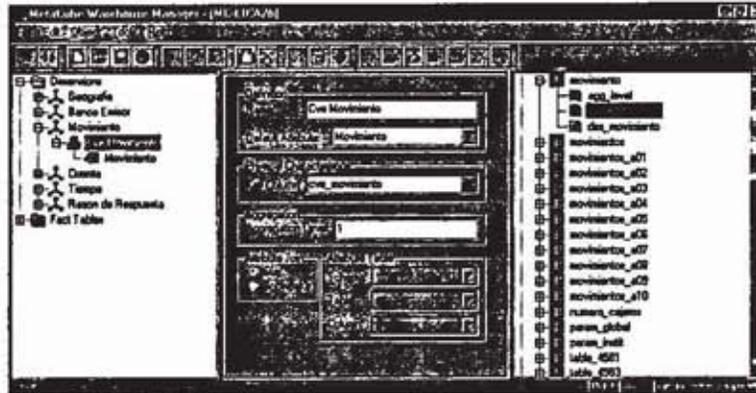


Figura 3.24 Definición del elemento de dimensión Clave de Movimiento

El atributo del elemento de dimensión es el que se muestra en la figura 3.24.a que se presenta a continuación.

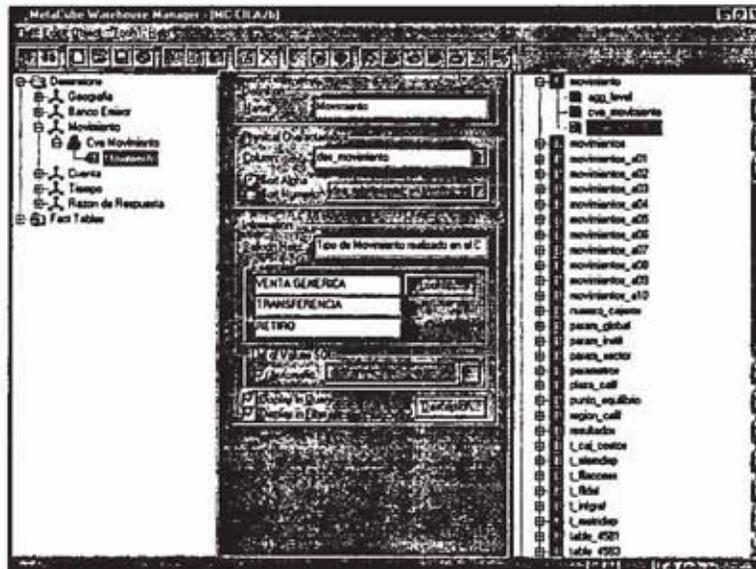


Figura 3.24.a Atributo de dimensión Clave de Movimiento

DIMENSION TIEMPO

En la figura 3.27 se muestra la definición de la dimensión Tiempo.

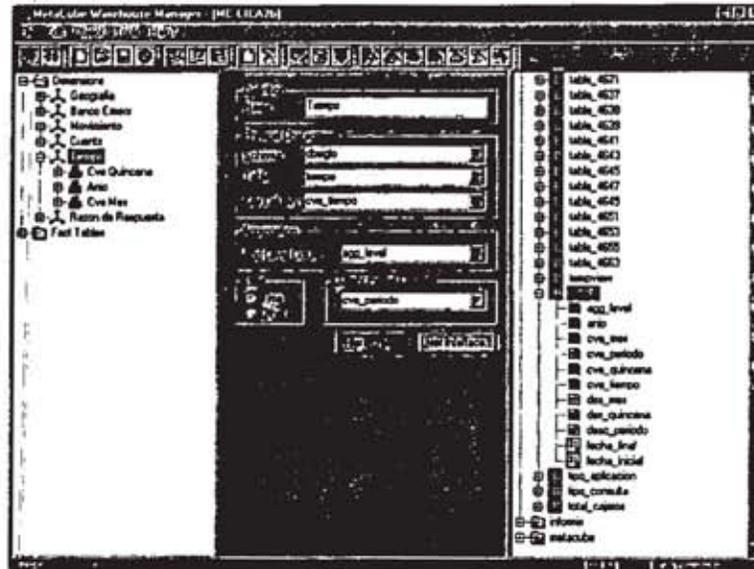


Figura 3.27 Definición de la dimensión Tiempo

En la sección izquierda de la figura 3.27 se definen los elementos de la dimensión Tiempo que son:

- Quincena
- Año
- Mes

Cada uno de estos elementos tiene un número único que define su lugar dentro de la jerarquía de la dimensión.

En la sección central de la figura 3.27 se muestra la definición de la dimensión, sus características físicas, la columna que define el nivel de agregación, el tipo de dimensión, la jerarquía y la interfaz de usuario. En la sección derecha se muestra el esquema físico de la tabla que define la dimensión. Para esta dimensión la jerarquía se define mediante el botón "Hierarchy" como se muestra en la figura 3.27.a



Figura 3.27.a Jerarquía de la dimensión Tiempo

La interfaz de usuario se define mediante el botón "User Interface" y es como se muestra en la figura 3.27.b.



Figura 3.27.b Interfaz de usuario para la dimensión Tiempo

A continuación se presenta la definición de los elementos de la dimensión Tiempo.

QUINCENA

En la figura 3.28 se muestra la definición del elemento Quincena de la dimensión Tiempo.

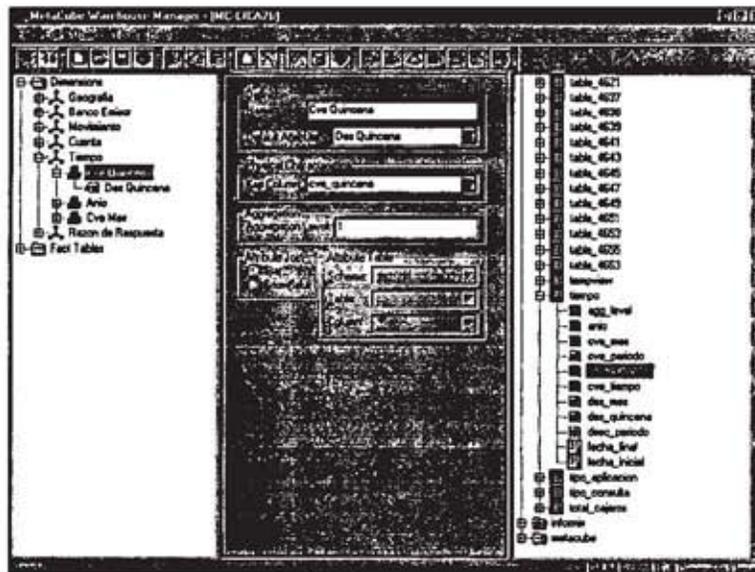


Figura 3.28 Elemento Quincena de la dimensión Tiempo

El atributo del elemento de dimensión es el que se muestra en la figura 3.28.a que se presenta a continuación.

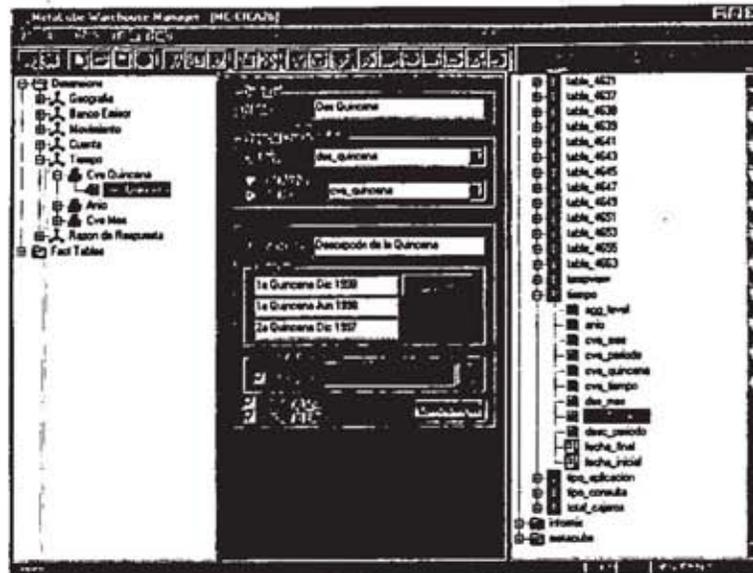


Figura 3.28.a Atributo del elemento de dimensión Quincena

AÑO

En la figura 3.29 se muestra la definición del elemento Año de la dimensión Tiempo.

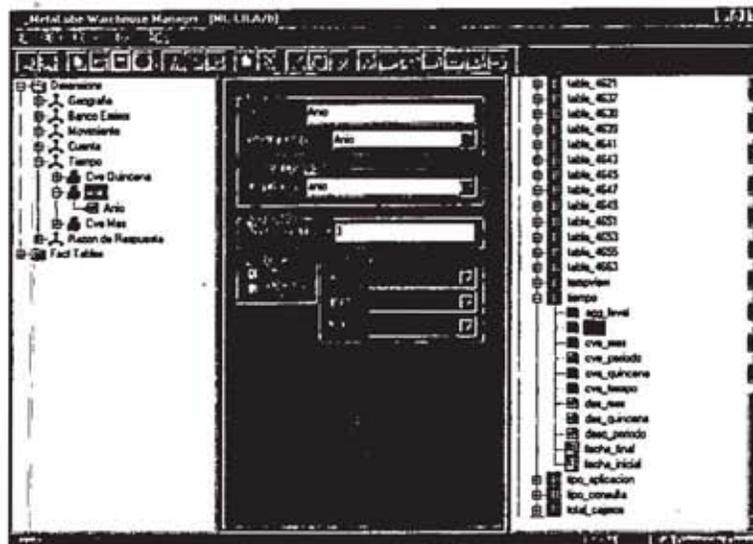


Figura 3.29 Elemento Año de la dimensión Tiempo

El atributo del elemento de dimensión es el que se muestra en la figura 3.29.a que se presenta a continuación.

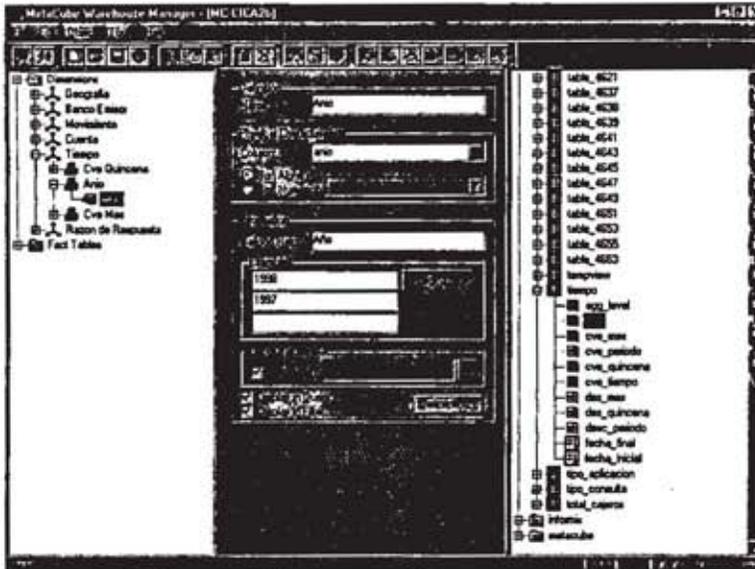


Figura 3.29.a Atributo del elemento de dimensión Año

MES

En la figura 3.30 se muestra la definición del elemento Mes de la dimensión Tiempo.

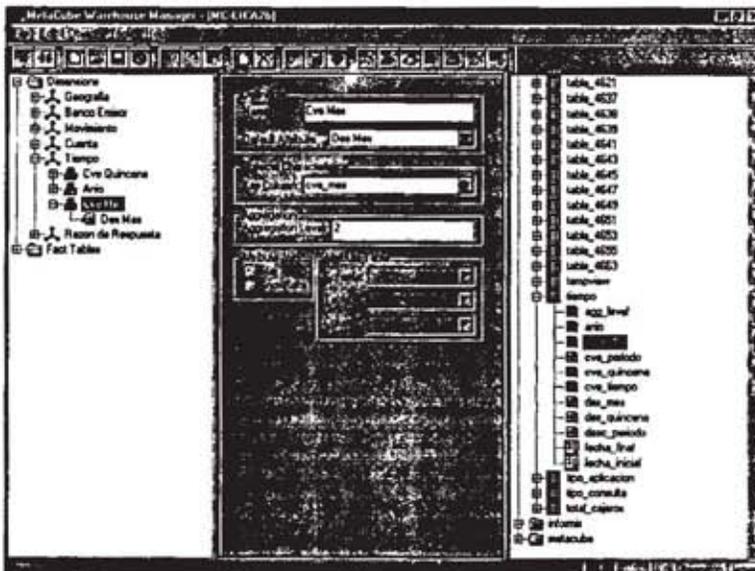


Figura 3.30 Elemento Mes de la dimensión Tiempo

El atributo del elemento de dimensión es el que se muestra en la figura 3.30.a que se presenta a continuación.

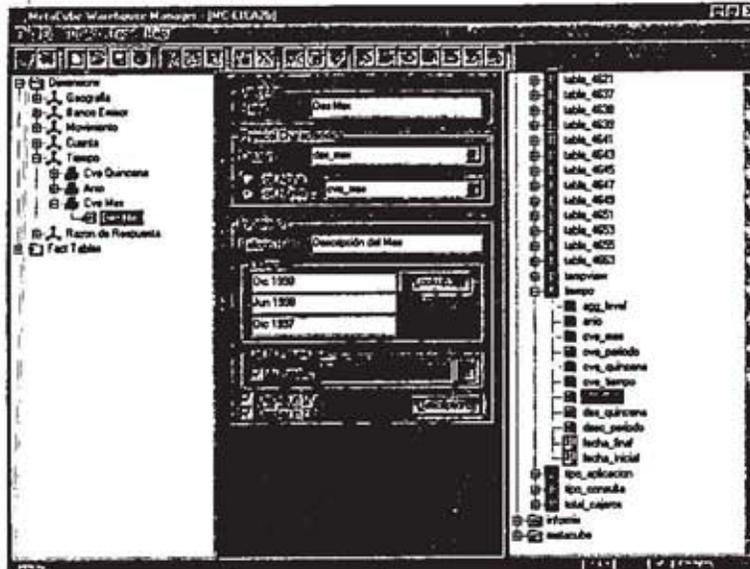


Figura 3.30.a Atributo del elemento de dimensión Mes

DIMENSION RAZON DE RESPUESTA

En la figura 3.31 se muestra la definición de la dimensión Razón de Respuesta.

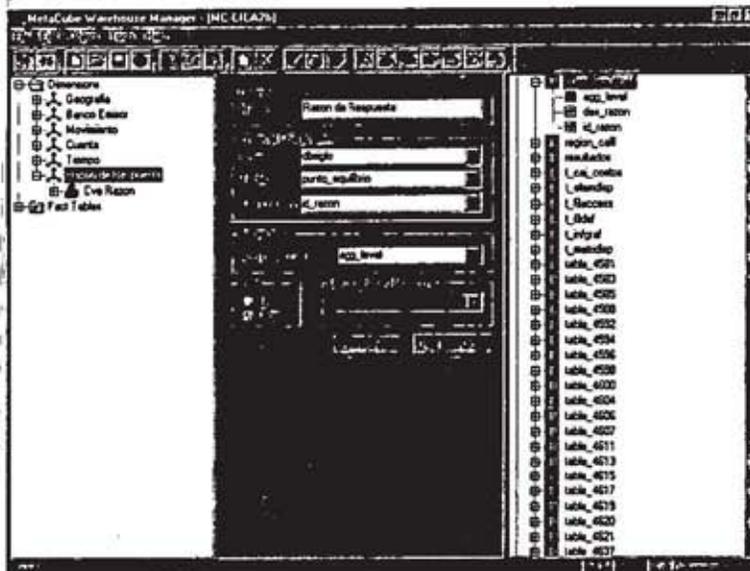


Figura 3.31 Definición de la dimensión Razón de Respuesta

En la sección izquierda de la figura 3.31 se define el elemento de la dimensión Razón de Respuesta que es:

- Clave Razón

En la sección central de la figura 3.31 se muestra la definición de la dimensión, sus características físicas, la columna que define el nivel de agregación, el tipo de dimensión, la jerarquía y la interfaz de usuario.

En la sección derecha de la figura 3.31 se muestra el esquema físico de la tabla que define la dimensión.

Para esta dimensión no se definió ninguna jerarquía.

La interfaz de usuario se define mediante el botón "User Interface" y es como se muestra en la figura 3.31.a.



Figura 3.31.a Interfaz de usuario para la dimensión Razón de Respuesta

A continuación se presenta la definición del elemento de la dimensión Razón de Respuesta.

CLAVE RAZON

En la figura 3.32 se muestra la definición del elemento Clave Razón de la dimensión Razón de Respuesta.

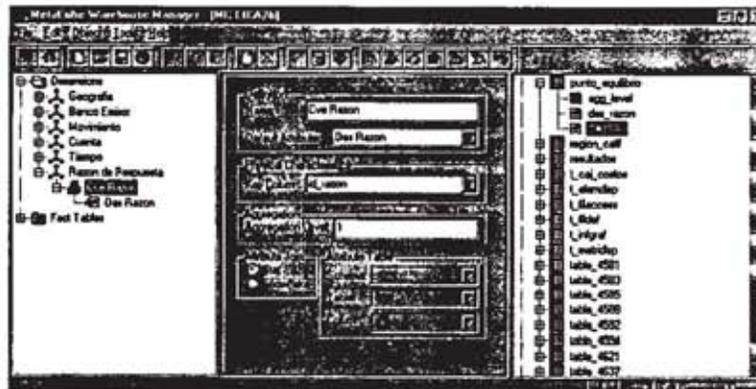


Figura 3.32 Elemento Clave Razón de la dimensión Razón de Respuesta

El atributo del elemento de dimensión es el que se muestra en la figura 3.32.a que se presenta a continuación.

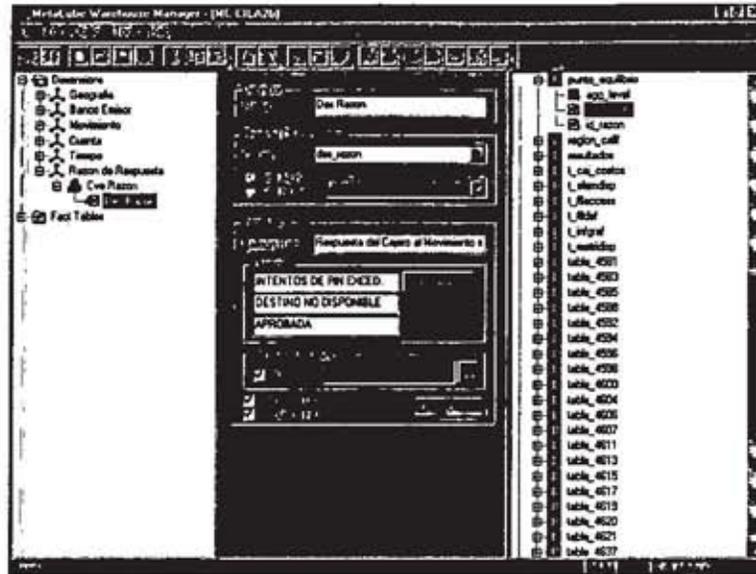


Figura 3.32.a Atributo del elemento de dimensión Clave Razón

Del lado derecho de la figura 3.33 se observa el esquema físico al cual pertenece la tabla de hechos.

METRICAS

En esta sección se presentan las pantallas para definir las métricas de la tabla de hechos.

En la figura que se presenta a continuación se observa la definición de las métricas.



Figura 3.34 Definición de las métricas de la tabla de hechos

Como se mencionó en la definición del modelo las métricas de la tabla de hechos son:

- Movimientos Aceptados
- Monto Aceptado
- Monto Rechazado
- Movimiento Rechazado

En la sección central de la figura 3.34 se presentan las expresiones mediante las cuales se puede definir las métricas:

- Suma (SUM)
- Promedio (AVG)
- Conteo (COUNT)
- Conteo de distinto (COUNT(DISTINCT))
- Máximo (MAX)
- Mínimo (MIN)

Las columnas que se pueden elegir como métricas se presentan en la siguiente figura:



Figura 3.34.a Columnas elegibles para ser métricas

Dentro de las columnas que no se pueden elegir como métricas se encuentran las llaves que definen la dimensionalidad, las cuales son:

- Banco Emisor (banco_emisor)
- Banco Dueño (cve_banco)
- Cajero (cve_cajero)
- Cuenta (cve_cuenta)
- Movimiento (cve_movimiento)
- Tiempo (cve_tiempo)
- Razón de Respuesta (id_razon)

La definición propiamente de las métricas dentro del modelo se realiza en la parte central de la figura 3.34 como se muestra en la siguiente figura.

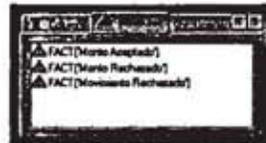


Figura 3.34.b Definición de las métricas dentro de la tabla de hechos

AGREGADOS

Los agregados son tablas que contienen resultados previamente sumariados de las consultas más comunes y que MetaCube utiliza para que esas consultas tengan una respuesta más rápida.

Para definirlos dentro del Warehouse Manager es necesario decidir qué métricas se utilizarán y mediante cuales llaves se relacionará con las dimensiones, esto último lo realiza el propio Warehouse Manager durante el proceso de construcción del agregado.

Los agregados que definieron como se muestra en las siguientes figuras.

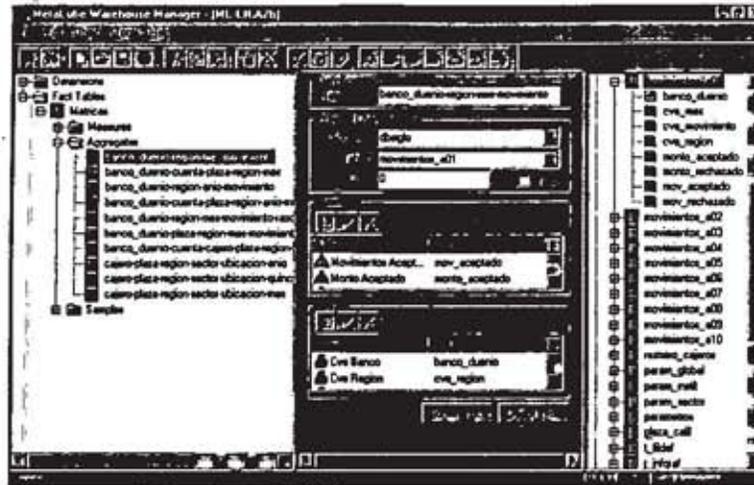


Figura 3.35 Definición de Agregado 01

Para cada agregado Warehouse Manager genera una sentencia SQL, mediante el botón "Create Info", que involucra a todas las columnas que integrarán a la tabla de agregado con las condiciones sobre la tabla de hechos original y las dimensiones, como se muestra en la siguiente figura.



Figura 3.35.a Sentencia SQL generada para crear el Agregado 01

NOTA: Esta sentencia SQL se debe copiar en donde se realizarán los procesos de carga para que al final de la carga de datos de la tabla de hechos original se ejecute y se genere o actualice la tabla del agregado.

Para definir los demás agregados se procede de forma similar a como se indicó para el agregado 1.

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

En este capítulo se presentan las especificaciones de cómo se construyeron las gráficas y reportes con MetaCube Explorer. La definición de cada gráfica y reporte se dividió en "Query Mode" y "Results Mode" que corresponden a la modalidad de diseño y de presentación de resultados respectivamente. En total se construyeron 27 gráficas, con variantes en las gráficas 15, 21 y 24 (para diferenciar los conceptos de crédito y débito) y 3 reportes. A continuación se presenta una relación de las gráficas y reportes construidos.

TIPO	NOMBRE DE ARCHIVO	DESCRIPCION
Gráficas	grafica01.mcw	Número de disposiciones exitosas.
	grafica02.mcw	Número de transacciones exitosas.
	grafica03.mcw	Montos dispuestos por banco.
	grafica04.mcw	Transacciones de otros tarjetahabientes en nuestros cajeros.
	grafica05.mcw	Montos dispuestos por todos los tarjetahabientes.
	grafica06.mcw	Montos dispuestos de nuestros tarjetahabientes en otros cajeros.
	grafica07.mcw	Montos dispuestos de otros tarjetahabientes en nuestros cajeros.
	grafica08.mcw	Comparativo de montos dispuestos de nuestros tarjetahabientes y otros tarjetahabientes.
	grafica09.mcw	Montos dispuestos en cajeros propios mostrando el promedio.
	grafica10.mcw	Comparativo de montos dispuestos.
	grafica11.mcw	Comparativo por tipo de transacción todos los tarjetahabientes.
	grafica12.mcw	Comparativo por tipo de transacción de Nuestros tarjetahabientes.
	grafica13.mcw	Comparativo por tipo de transacción y respuesta.
	grafica14.mcw	Comparativo de tipo de transacción de nuestros tarjetahabientes y otros tarjetahabientes.
	grafica15.mcw	Comparativo por tipo de transacción Crédito.
	grafica15a.mcw	Comparativo por tipo de transacción Débito.
	grafica16.mcw	Comparativo por tipo de transacción otros tarjetahabientes.
	grafica17.mcw	Número de disposiciones rechazadas por Razón de Respuesta.
	grafica18.mcw	Número de transacciones rechazadas por Razón de Respuesta.
	grafica19.mcw	Montos dispuestos por mes.
	grafica20.mcw	Montos dispuestos por mes nuestros tarjetahabientes y otros tarjetahabientes.
	grafica21.mcw	Montos dispuestos por mes Crédito.
	grafica21a.mcw	Montos dispuestos por mes Débito.
	grafica22.mcw	Transacciones aceptadas por mes.
	grafica23.mcw	Transacciones aceptadas por mes nuestros tarjetahabientes y otros tarjetahabientes.
	grafica24.mcw	Transacciones aceptadas por mes Crédito.
	grafica24a.mcw	Transacciones aceptadas por mes Débito.

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

TIPO	NOMBRE DE ARCHIVO	DESCRIPCION
Reportes	reporte01.mcw	Retiros de todos los tarjetahabientes en cajeros propios.
	reporte02.mcw	Montos dispuestos de todos los tarjetahabientes.
	reporte03.mcw	Montos dispuestos de todos los tarjetahabientes en cajeros de BanCrecer.

La extensión de los archivos (mcw) se debe a que cada gráfica y reporte se define en un libro de trabajo (workbook) de MetaCube. Para conocer más acerca de la creación de libros de trabajo se recomienda consultar a partir de la página 2-13 del manual "Explorer User's Guide, version 3.1".

Para construir algunas gráficas y reportes, que requieren agrupaciones especiales, se utilizó el concepto de "Cubeta" o Bucket, el cual se encuentra en MetaCube Explorer como una característica que permite personalizar reportes, al definir grupos de atributos para ser utilizados como base de la sumarización de datos.

Para conocer más acerca de la definición de buckets se recomienda consultar a partir de la página 3-6 del manual "Explorer User's Guide, version 3.1".

También se definieron filtros que pueden ser utilizados por cualquier usuario y que permiten delimitar las consultas de una forma transparente.

Para conocer más acerca de la definición de filtros se recomienda consultar a partir de la página 4-3 del manual "Explorer User's Guide, version 3.1".

4.1 GRAFICAS

A continuación se presenta la especificación de cada una de las gráficas en "Query Mode" (modo de diseño de la consulta) y en "Results Mode" (resultado de la consulta y formateo de la salida).

Gráfica 01-"Número de Disposiciones Exitosas"

Query Mode

Por Banco						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Duenio	Des Mes	Movimientos Aceptados	Banco Dueño Movimiento	Des Duenio=BANCRECER Movimiento = RETIRO	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un sólo mes Sin filtro
Por Región						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Region	Des Plaza	Movimientos Aceptados	Banco Dueño Movimiento	Des Duenio=BANCRECER Movimiento = RETIRO	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un sólo mes Una Región
Por Plaza						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Plaza	Des Cajero	Movimientos Aceptados	Banco Dueño Movimiento	Des Duenio=BANCRECER Movimiento = RETIRO	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un sólo mes Una Plaza

Gráfica 02 - "Número de Transacciones Exitosas"

Esta gráfica es similar a la Gráfica 01, con las siguientes diferencias:

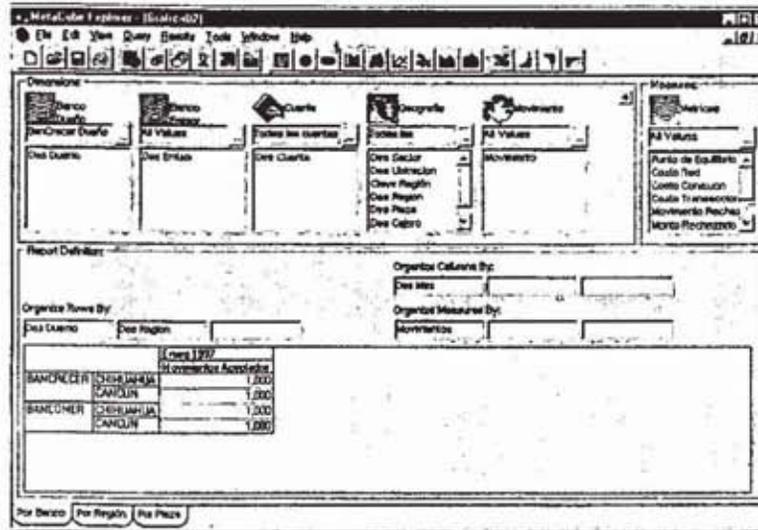
Query Mode

1. En la sección de Filtros Fijos, se eliminan la dimensión Movimiento y la descripción Movimiento = RETIRO

Results Mode

1. El título cambia a "NUMERO DE TRANSACCIONES EXITOSAS".
2. En el eje Y el texto cambia a "Número de Transacciones".

A continuación se presentan las imágenes de la definición de esta gráfica.



Query Mode



Results Mode

Gráfica 03 - *Montos Dispuestos por Banco*

Query Mode

Por Banco						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Duenio	Des Mes	Monto Aceptado	Banco Dueño Banco Emisor Movimiento	Des Duenio=BANCRECER Des emisor <> BANCRECER Movimiento = RETIRO	Cuenta Tiempo Geografía	Uno o todos los tipos de Cuentas Un sólo mes Sin filtro
Por Región						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Region	Des Mes	Monto Aceptado	Banco Dueño Banco Emisor Movimiento	Des Duenio=BANCRECER Des emisor <> BANCRECER Movimiento = RETIRO	Cuenta Tiempo Geografía	Uno o todos los tipos de Cuentas Un sólo mes Una Región
Por Plaza						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Plaza	Des Mes	Monto Aceptado	Banco Dueño Banco Emisor Movimiento	Des Duenio=BANCRECER Des emisor <> BANCRECER Movimiento = RETIRO	Cuenta Tiempo Geografía	Uno o todos los tipos de Cuentas Un sólo mes Una Plaza
Por Cajero						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Cajero	Des Mes	Monto Aceptado	Banco Dueño Banco Emisor Movimiento	Des Duenio=BANCRECER Des emisor <> BANCRECER Movimiento = RETIRO	Cuenta Tiempo Geografía	Uno o todos los tipos de Cuentas Un sólo mes Un Cajero por Nombre (D)

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Gráfica 04 - "TRANSACCIONES DE OTROS TARJETAHABIENTES EN NUESTROS CAJEROS"

Query Mode

Por Banco						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Duenio	Des Mes	Movimientos Aceptados	Banco Dueño	Des Duenio=BANCRECER	Cuenta	Uno o todos los tipos de Cuentas
Des Emisor			Banco Emisor	Des emisor <> BANCRECER	Tiempo	Un sólo mes
					Geografia	Sin filtro
Por Región						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Region	Des Mes	Movimientos Aceptados	Banco Dueño	Des Duenio=BANCRECER	Cuenta	Uno o todos los tipos de Cuentas
Des Emisor			Banco Emisor	Des emisor <> BANCRECER	Tiempo	Un sólo mes
					Geografia	Una Región
Por Plaza						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Plaza	Des Mes	Movimientos Aceptados	Banco Dueño	Des Duenio=BANCRECER	Cuenta	Uno o todos los tipos de Cuentas
Des Emisor			Banco Emisor	Des emisor <> BANCRECER	Tiempo	Un sólo mes
					Geografia	Una Plaza
Por Cajero						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Cajero	Des Mes	Movimientos Aceptados	Banco Dueño	Des Duenio=BANCRECER	Cuenta	Uno o todos los tipos de Cuentas
Des Emisor			Banco Emisor	Des emisor <> BANCRECER	Tiempo	Un sólo mes
					Geografia	Un Cajero por Nombre

Results Mode

Titulo	Eje X	Eje Y	Legend
Format Title	Format Axis Title	Format Axis Title	
Texto: TRANSACCIONES DE OTROS TARJETAHABIENTES EN NUESTROS CAJEROS	No visible	Text: Número de Transacciones	Des Mes
Font: Arial, Negrita, 10		Orientation: Up	Font: Arial, Normal, 8
Location: Custom, Top 0.28m	Format Axis Label	Format Axis Label	Location: Visible, Custom, Top 1 cm
	Text: Automatic	Text: Automatic	
	Font: Arial, Negrita, 8	Font: Arial, Normal, 8	

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Format Code: Format Code: Number #,##0

Dimension

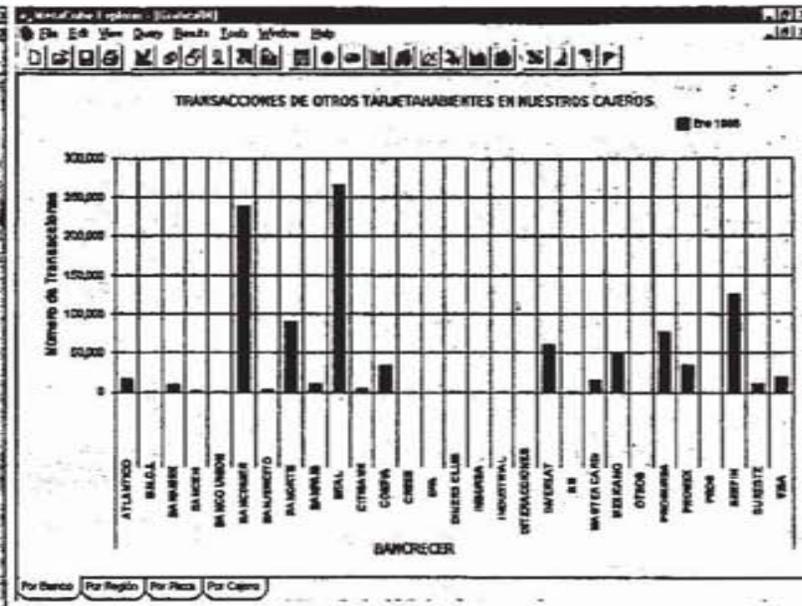
Report Definition

Organize Rows By

Des Dueno	Des Emisor	Cuenta
SANCORER	SANCORER	1,200

Por Banco Por Región Por Plaza Por Cajero

Query Mode



Results Mode

Gráfica 05 - "MONTOS DISPUESTOS POR TODOS LOS TARJETAHABIENTES"

Query Mode

Por Banco						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Duenio	Des Mes	Monto Aceptado	Banco Dueño	Des Duenio=BANCRECER	Cuenta	Uno o todos los tipos de Cuentas
Des Region			Movimiento	Movimiento = RETIRO	Tiempo	Un sólo mes
					Geografía	Sin filtro
Por Región						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Region	Des Mes	Monto Aceptado	Banco Dueño	Des Duenio=BANCRECER	Cuenta	Uno o todos los tipos de Cuentas
Des Plaza			Movimiento	Movimiento = RETIRO	Tiempo	Un sólo mes
					Geografía	Una Región
Por Plaza						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Plaza	Des Mes	Monto Aceptado	Banco Dueño	Des Duenio=BANCRECER	Cuenta	Uno o todos los tipos de Cuentas
Des Cajero			Movimiento	Movimiento = RETIRO	Tiempo	Un sólo mes
					Geografía	Una Plaza

Results Mode

Título	Eje X	Eje Y	Legend
Format Title	Format Axis Title	Format Axis Title	
Texto: MONTOS DISPUESTOS POR TODOS LOS TARJETAHABIENTES	No visible	Text: Montos Dispuestos	Des Mes
Font: Arial, Negrita, 10		Orientation: Up	Font: Arial, Normal, 8
Location: Custom, Top 0.28m		Font: Arial, Negrita, 10	Location: Visible, Top Right
	Format Axis Label	Format Axis Label	
	Text: Automatic	Text: Automatic	
	Font: Arial, Negrita, 8	Font: Arial, Normal, 8	
	Format Code:	Format Code: Currency	
		\$\$,##0.[Red](\$#,##0)	

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

MetaCube Explorer - [Database]

File Edit View Query Results Tools Window Help

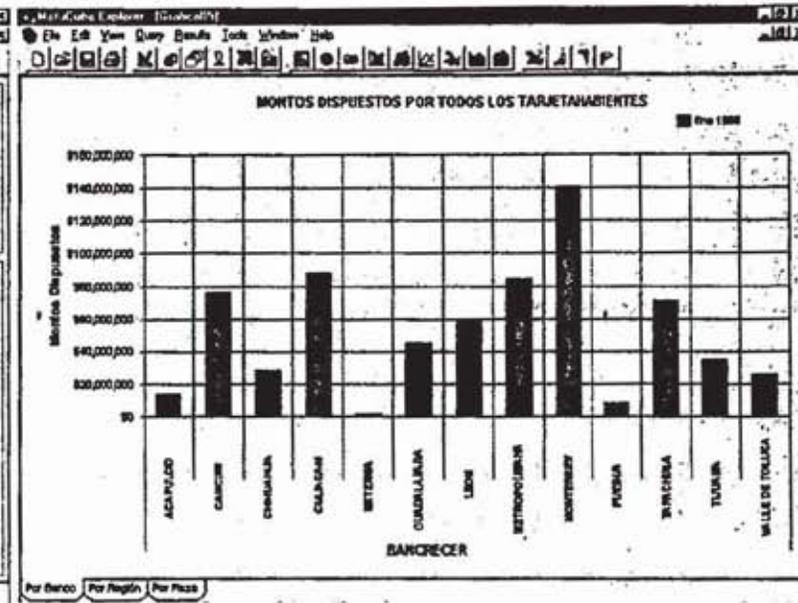
Dimensiones: Banco, Cuenta, Origen, Movimiento, Mensajes

Report Definition: Organize Rows By: Por Banco, Por Región

From 1997		Montos Aceptados
BANCRECER	CHIHUAHUA	1,000.00
BANCRECER	CANLUN	1,000.00
BANCRECER	CHIHUAHUA	1,000.00
BANCRECER	CANLUN	1,000.00

Per Banco Per Región Per País

Query Mode



Results Mode

Gráfica 06 - *MONTOS DISPUESTOS DE NUESTROS TARJETAHABIENTES EN OTROS CAJEROS*

Query Mode

Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Duenio	Des Mes	Monto Aceptado	Banco Dueño Banco Emisor Movimiento	Des Duenio<>BANCRECER Des Emisor = BANCRECER Movimiento = RETIRO	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un sólo mes Sin filtro

Results Mode

Título	Eje X	Eje Y	Legend
Format Title	Format Axis Title	Format Axis Title	
Texto: MONTOS DISPUESTOS DE NUESTROS TARJETAHABIENTES EN OTROS CAJEROS	No visible	Text: Montos	Des Mes
Font: Arial, Negrita, 10		Orientation: Up	Font: Arial, Normal, 8
Location: Custom, Top 0.28 cm		Font: Arial, Negrita, 10	Location: Visible, Top Right
	Format Axis Label	Format Axis Label	
	Text: Automatic	Text: Automatic	
	Font: Arial, Negrita, 8	Font: Arial, Normal, 8	
	Format Code:	Format Code: Currency	
		\$\$,##0;[Red](\$#,##0)	

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

MetaCube Explorer (Gráfico06)

File Edit View Query Results Tools Window Help

Dimensiones:

- Banco: Des Banco, Des Banca
- Cuenta: Des Cuenta, Des Cuenta
- Geografía: Des Sector, Des Ubicación, Des Región, Des Plaza, Des Cajero
- Movimiento: Des Movimiento
- Métrica: Des Métrica

Report Definition:

Organize Rows By: Des Cuenta

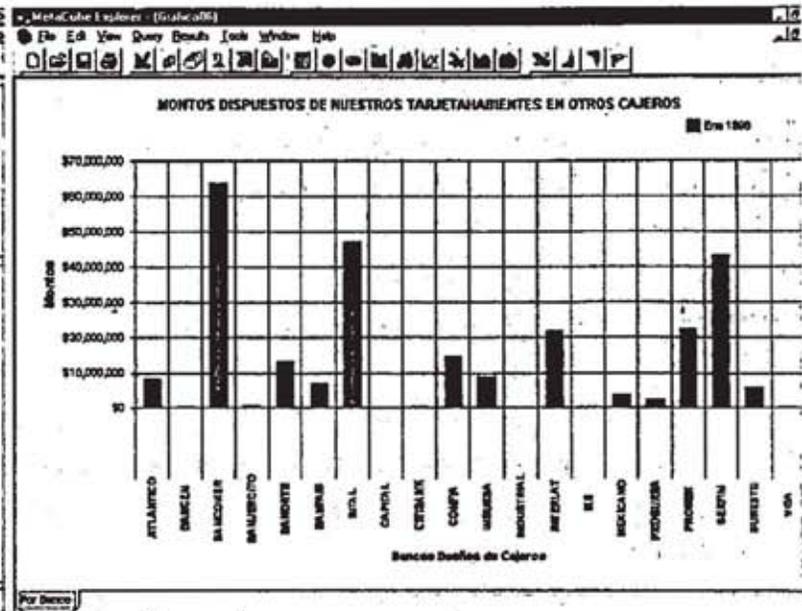
Organize Columns By: Des Mes

Organize Measures By: Monto Acreditado

Year 1997	Monto Acreditado
BANCO CCI	1,000.00
BANCO MER	1,000.00

Por Banca

Query Mode



Results Mode

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Gráfica 07 - *MONTOS DISPUESTOS DE OTROS TARJETAHABIENTES EN NUESTROS CAJEROS*

Query Mode

Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Emisor	Des Mes Des Duenio	Monto Aceptado	Banco Dueño Movimiento Banco Emisor	Des Duenio=BANCRECER Movimiento=RETIRO Des emisor<> BANCRECER	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un sólo mes Sin filtro
Por Región						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Emisor	Des Mes Des Region	Monto Aceptado	Banco Dueño Movimiento Banco Emisor	Des Duenio=BANCRECER Movimiento=RETIRO Des emisor<> BANCRECER	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un sólo mes Una Región
Por Plaza						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Emisor	Des Mes Des Plaza	Monto Aceptado	Banco Dueño Movimiento Banco Emisor	Des Duenio=BANCRECER Movimiento=RETIRO Des emisor<> BANCRECER	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un sólo mes Una Plaza
Por Cajero						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Emisor	Des Mes Des Cajero	Monto Aceptado	Banco Dueño Movimiento Banco Emisor	Des Duenio=BANCRECER Movimiento=RETIRO Des emisor<> BANCRECER	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un sólo mes Un Cajero por Nombre

Gráfica 08 - *COMPARATIVO DE MONTOS DISPUESTOS DE NUESTROS THB Y OTROS THB*

Query Mode

Por Banco		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Elementos		Dimensión	Descripción	Dimensión	Descripción
Des Duenio	Des Emisor ** BUCKET(Attributes.Banco Emisor.Des Emisor, [Nuestros THB, LIST("BANCRECER"), [Otros THB, OTHER])	Monto Aceptado	Movimiento	Movimiento=RETIRO	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Sin filtro

Results Mode

Título	Eje X	Eje Y	Legend
Format Title	Format Axis Title	Format Axis Title	
Texto: COMPARATIVO DE MONTOS DISPUESTOS DE NUESTROS THB Y OTROS THB	No visible	No visible	Des Emisor
Font: Arial, Negrita, 10			Font: Arial, Normal, 8
Location: Visible, Top			Location: Visible, Top Right
	Format Axis Label	Format Axis Label	
	Text: Automatic	Text: Automatic	
	Font: Arial, Negrita, 8	Font: Arial, Normal, 8	
	Format Code:	Format Code: Currency \$#,##0,[Red](\$#,##0)	

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Gráfica 09 - "MONTOS DISPUESTOS EN CAJEROS PROPIOS MOSTRANDO EL PROMEDIO"

Query Mode

Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Duenio	Des Mes	Monto Aceptado	Banco Dueño	Des Duenio = BANCRECER	Cuenta	Uno o todos los tipos de Cuentas
Des Region			Movimiento	Movimiento = RETIRO	Tiempo	Un sólo mes
					Geografia *	Sin filtro
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Region	Des Mes	Monto Aceptado	Banco Dueño	Des Duenio=BANCRECER	Cuenta	Uno o todos los tipos de Cuentas
Des Plaza			Movimiento	Movimiento=RETIRO	Tiempo	Un sólo mes
					Geografia *	Una Región
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Plaza	Des Mes	Monto Aceptado	Banco Dueño	Des Duenio=BANCRECER	Cuenta	Uno o todos los tipos de Cuentas
Des Cajero			Movimiento	Movimiento=RETIRO	Tiempo	Un sólo mes
					Geografia *	Una Plaza

Results Mode

Titulo	Eje X	Eje Y	Legend
Format Title	Format Axis Title	Format Axis Title	
Texto: MONTOS DISPUESTOS EN CAJEROS PROPIOS MOSTRANDO EL PROMEDIO	No visible	Text: Montos	Des Mes
Font: Arial, Negrita, 10		Orientation: Up	Font: Arial, Normal, 8
Location: Custom, Top 0.28 cm		Font: Arial, Negrita, 10	Location: Visible, Top Right
	Format Axis Label	Format Axis Label	
	Text: Automatic	Text: Automatic	
	Font: Arial, Negrita, 8	Font: Arial, Normal, 8	
		Format Code: Currency \$#,##0;	
		[Red] (\$#,##0)	

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

MetaCube Explorer (Estado: off)

File Edit View Query Breaks Tools Window Help

Observar

Inicio Datos Datos Geografía Movimiento Medicas

Inicio Datos: Banco, Cuenta, Fecha, Valor, Cuenta, Fecha, Valor, Movimiento

Medicas: Punto de Equilibrio, Costo Red, Costo Conexión, Costo Transacción, Monto Recaudado, Monto Aceptado

Repeat Definition

Organize Rows By: Des, Cuenta, en Region

Organize Columns By: Des Mes

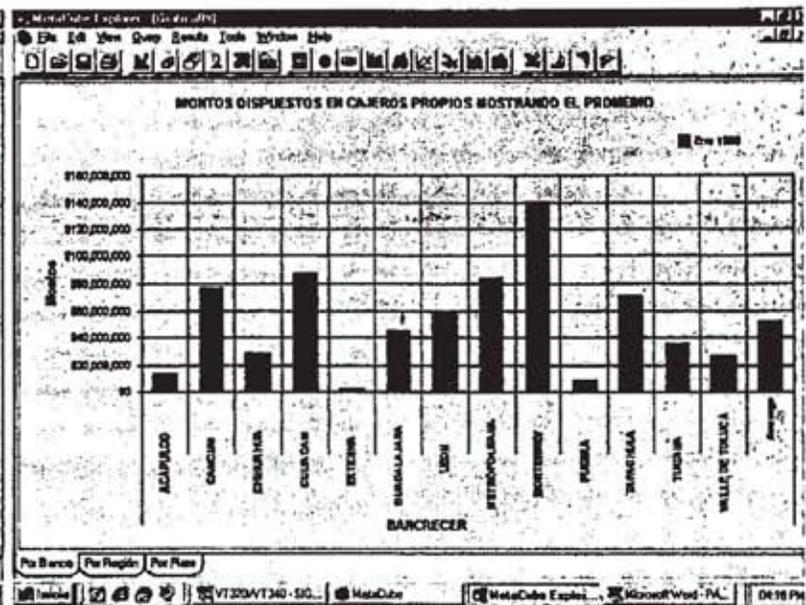
Organize Measures By: Monto Aceptado

Bank	Region	Month	Monto Aceptado
BANCRECER	QUERETARO	Nov 1998	1,000.00
BANCRECER	QUERETARO	Nov 1998	1,000.00
BANCRECER	QUERETARO	Nov 1998	1,000.00
BANCRECER	QUERETARO	Nov 1998	1,000.00

Por Banco Por Región Por Mes

Inicio MetaCube Explorer Microsoft Word - PA... 04:15 PM

Query Mode



Results Mode

Gráfica 10 - *COMPARATIVO DE MONTOS DISPUESTOS*

Query Mode

Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Duenio Des Region	Des Mes	Monto Aceptado	Banco Dueño Movimiento	Des Duenio = BANCRECER Movimiento = RETIRO	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Sin filtro
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Region Des Plaza	Des Mes	Monto Aceptado	Banco Dueño Movimiento	Des Duenio=BANCRECER Movimiento=RETIRO	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Una Región
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Plaza Des Cajero	Des Mes	Monto Aceptado	Banco Dueño Movimiento	Des Duenio=BANCRECER Movimiento=RETIRO	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Una Plaza

Results Mode

Título	Eje X	Eje Y	Legend
Format Title	Format Axis Title	Format Axis Title	
Texto: COMPARATIVO DE MONTOS DISPUESTOS	No visible	Text: Montos	Des Mes
Font: Arial, Negrita, 10		Orientation: Up	Font: Arial, Normal, 8
Location: Custom, Top 0.28 cm		Font: Arial, Negrita, 10	Location: Visible, Top Right
	Format Axis Label	Format Axis Label	
	Text: Automatic	Text: Automatic	
	Font: Arial, Negrita, 8	Font: Arial, Normal, 8	
		Format Code: Currency	
		\$\$,##0.[Red](\$#,##0)	

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

MetaCube Explorer - [Database]

File Edit View Query Results Tools Window Help

Dimensiones

Banco: Banco Cuenta Cliente Movimiento

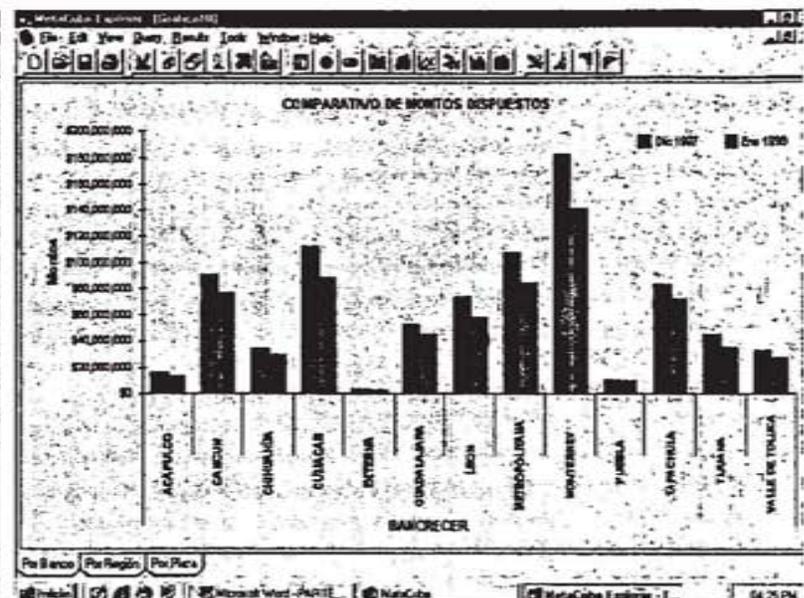
Filtros Definidos

Organizar Filtros By: Por Banco Por Plaza

Banco	Plaza	Monto Aceptado
BANCOMER	OAXACA	1,000,000
BANCOMER	QUERETARO	1,000,000

MetaCube Explorer - L. Microsoft Word - PPT1E... 04:45 PM

Query Mode



Results Mode

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Gráfica 11 - "COMPARATIVO POR TIPO DE TRANSACCION TODOS LOS TARJETAHABIENTES"

Query Mode

Por Banco						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Duenio	Des Mes	Movimientos Aceptados	Banco Dueño	Des Duenio=BANCRECER	Cuenta Tiempo Geografía	Uno o todos los tipos de Cuentas Un mes o bimestre Sin filtro
Movimiento						
Por Región						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Region	Des Mes	Movimientos Aceptados	Banco Dueño	Des Duenio=BANCRECER	Cuenta Tiempo Geografía	Uno o todos los tipos de Cuentas Un mes o bimestre Una Región
Movimiento						
Por Plaza						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Plaza	Des Mes	Movimientos Aceptados	Banco Dueño	Des Duenio=BANCRECER	Cuenta Tiempo Geografía	Uno o todos los tipos de Cuentas Un mes o bimestre Una Plaza
Movimiento						
Por Cajero						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Cajero	Des Mes	Movimientos Aceptados	Banco Dueño	Des Duenio=BANCRECER	Cuenta Tiempo Geografía	Uno o todos los tipos de Cuentas Un mes o bimestre Un Cajero por Nombre
Movimiento						

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Results Mode

Titulo	Eje X	Eje Y	Legend
Format Title	Format Axis Title	Format Axis Title	
Texto: COMPARATIVO POR TIPO DE TRANSACCION TODOS LOS THB	No visible	Text: Número Transacciones	Des Mes
Font: Arial, Negrita, 10		Orientation: Up	Font: Arial, Normal, 8
Location: Custom, Top 0.28 cm		Font: Arial, Negrita, 10	Location: Visible, Top Right
	Format Axis Label	Format Axis Label	
	Text:	Text: Automatic	
	Font: Arial, Negrita, 8	Font: Arial, Normal, 8	
		Format Code: Number #,##0	

MetaCube Explorer - (Parte.01)

Organice Reporte Por:

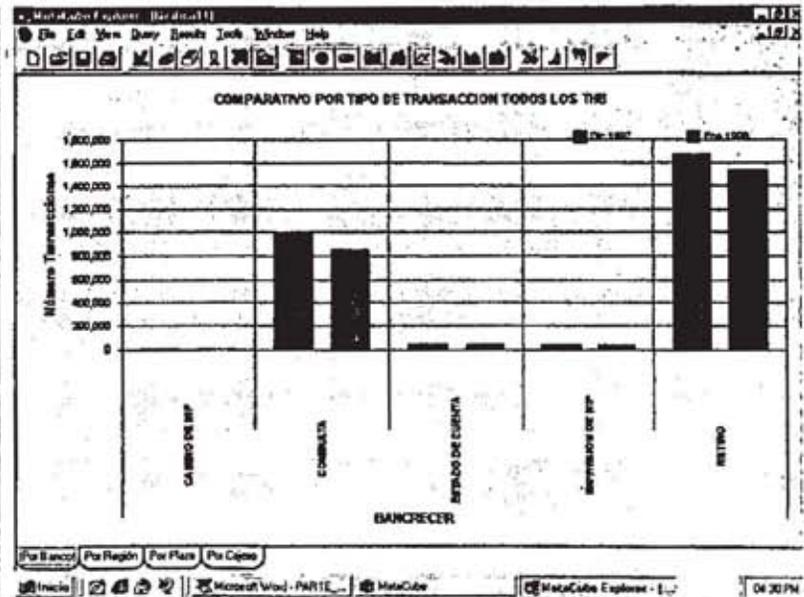
Organice Columna By:

Organice Fila By:

		Cantidad
SANCREER	DEPOSITO	1.000
SANCREER	CONSULTA	1.000
SANCRON	DEPOSITO	1.000
SANCRON	CONSULTA	1.000

05:55 PM

Query Mode



Results Mode

Gráfica 12 - "COMPARATIVO POR TIPO DE TRANSACCION DE NUESTROS TARJETAHABIENTES"

Esta gráfica es similar a la Gráfica 11, con las siguientes diferencias:

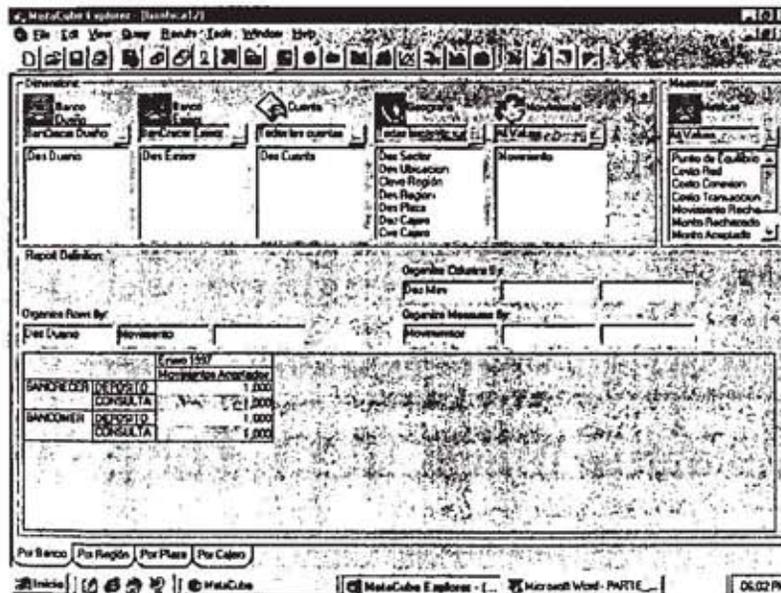
Query Mode

1. En la sección de Filtros Fijos, se agrega la dimensión Banco Emisor y la descripción Des Emisor = BANCRECER

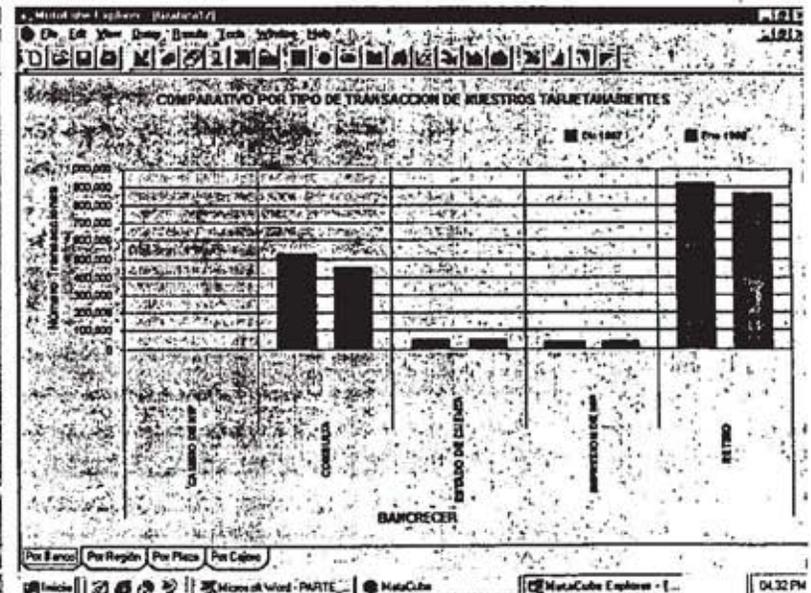
Results Mode

1. El título cambia a "COMPARATIVO POR TIPO DE TRANSACCION NUESTROS TARJETAHABIENTES".

A continuación se presentan las imágenes de la definición de esta gráfica.



Query Mode



Results Mode

Gráfica 13 - "COMPARATIVO POR TIPO DE TRANSACCION Y POR RESPUESTA"

Query Mode

Por Banco						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Duenio, Movimiento, Des Mes	Ninguno	Movimiento Rechazado Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Sin filtro
Por Región						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Region, Movimiento, Des Mes	Ninguno	Movimiento Rechazado Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Una Región
Por Plaza						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Plaza, Movimiento, Des Mes	Ninguno	Movimiento Rechazado Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Una Plaza
Por Cajero						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Cajero, Movimiento, Des Mes	Ninguno	Movimiento Rechazado Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Un Cajero por Nombre

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Results Mode

Titulo	Eje X	Eje Y	Legend
Format Title	Format Axis Title	Format Axis Title	
Texto: COMPARATIVO POR TIPO DE TRANSACCION Y POR RESPUESTA	No visible	Text: Número Movimientos	Métricas ("Movimiento Rechazado", "Movimientos Aceptados")
Font: Arial, Negrita, 10		Orientation: Up	Font: Arial, Normal, 8
Location: Custom, Top 0.22 cm		Font: Arial, Negrita, 10	Location: Visible, Right
	Format Axis Label	Format Axis Label	
	No visible	Text: Automatic	
		Font: Arial, Normal, 8	
		Format Code: Number #,##0	

MetaCube Explorer (Default)

File Edit View Query Results Tools Window Help

Dimensiones: Banco, Cuenta, Geografía, Movimiento, Métricas

Report Definition:

Organize Rows By: Des Cuentas, Movimiento, Des Mes

Organize Columns By: Movimiento Rechazado, Movimiento Aceptados

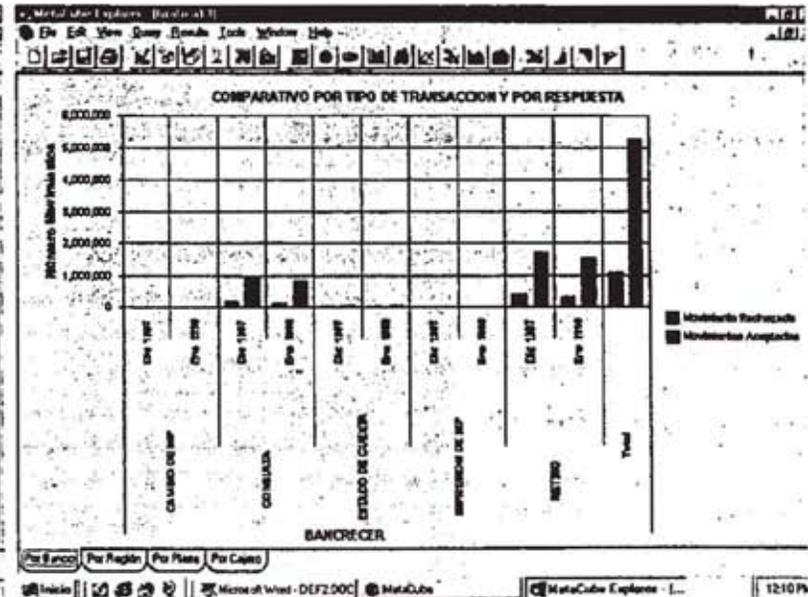
		Movimiento Rechazado	Movimiento Aceptados
BANCRECER DEPOSITO	Enero 1997	1,000	1,000
	Octubre 1998	1,000	1,000
CONSULTA	Enero 1997	1,000	1,000
	Octubre 1998	1,000	1,000
SANCORNER DEPOSITO	Enero 1997	1,000	1,000
	Octubre 1998	1,000	1,000
CONSULTA	Enero 1997	1,000	1,000
	Octubre 1998	1,000	1,000

Per Banco Per Región Per Plaza Per Cajas

MetaCube Explorer - L... Microsoft Word - WHITE...

06:20 PM

Query Mode



Results Mode

Gráfica 14 - "COMPARATIVO DE TIPO DE TRANSACCION DE NUESTROS TARJETAHABIENTES VS OTROS TARJETAHABIENTES"

Query Mode

Por Banco						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Duenio *** Movimiento Des Mes	Des Emisor ** BUCKET(Attributes.Banco Emisor ,Des Emisor,[Nuestros Tarjetahabientes, LIST("BANCRECER"), (Otros Tarjetahabientes, OTHER)])	Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Sin filtro
Por Región						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Region *** Movimiento Des Mes	Des Emisor ** BUCKET(Attributes.Banco Emisor ,Des Emisor,[Nuestros Tarjetahabientes, LIST("BANCRECER"), (Otros Tarjetahabientes, OTHER)])	Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Una Región
Por Plaza						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Plaza *** Movimiento Des Mes	Des Emisor ** BUCKET(Attributes.Banco Emisor ,Des Emisor,[Nuestros Tarjetahabientes, LIST("BANCRECER"), (Otros Tarjetahabientes, OTHER)])	Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Una Plaza

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Por Cajero						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Cajero *** Movimiento Des Mes	Des Emisor ** BUCKET(Attributes.Banco Emisor .Des Emisor,[Nuestros Tarjetahabientes, LIST("BANCRECER ")]), [Otros Tarjetahabientes, OTHER])	Movimientos Aceptados	Banco Dueño	Des Duenio BANCRECER	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Un Cajero por Nombre

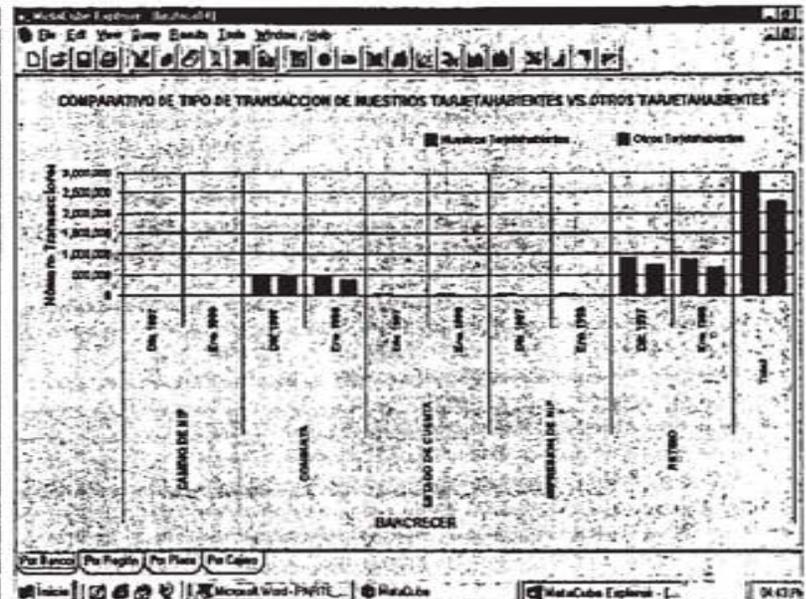
Results Mode

Título	Eje X	Eje Y	Legend
Format Title	Format Axis Title	Format Axis Title	
Texto: COMPARATIVO DE TIPO DE TRANSACCION DE NUESTROS TARJETAHABIENTES VS OTROS TARJETAHABIENTES	No visible	Text: Número Transacciones	Des Emisor
Font: Arial, Negrita, 10		Orientation: Up	Font: Arial, Normal, 8
Location: Custom, Top 0.28 cm		Font: Arial, Negrita, 10	Location: Visible, Top Right
	Format Axis Label	Format Axis Label	
	No visible	Text: Automatic	
		Font: Arial, Normal, 8	
		Format Code: Number #,##0	

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Organismo	Tipo de Operación	Fecha	Monto
BANCRECER	DEPOSITO	01/01/1999	1,000
	CONSULTA	01/01/1999	1,000
SANCORNER	DEPOSITO	01/01/1999	1,000
	CONSULTA	01/01/1999	1,000

Query Mode



Results Mode

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Gráfica 15 - "COMPARATIVO POR TIPO DE TRANSACCION DE CREDITO"

Query Mode

Por Banco						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Duenio	Des Cuenta	Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Tiempo	Un mes o bimestre
Movimiento			Cuenta	Des Cuenta = TARJETA DE CREDITO	Geografia	Sin filtro
Des Mes						
Por Región						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Region	Des Cuenta	Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Tiempo	Un mes o bimestre
Movimiento			Cuenta	Des Cuenta = TARJETA DE CREDITO	Geografia	Una Región
Des Mes						
Por Plaza						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Plaza	Des Cuenta	Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Tiempo	Un mes o bimestre
Movimiento			Cuenta	Des Cuenta = TARJETA DE CREDITO	Geografia	Una Plaza
Des Mes						
Por Cajero						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Cajero	Des Cuenta	Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Tiempo	Un mes o bimestre
Movimiento			Cuenta	Des Cuenta = TARJETA DE CREDITO	Geografia	Un Cajero (nombre)
Des Mes						

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

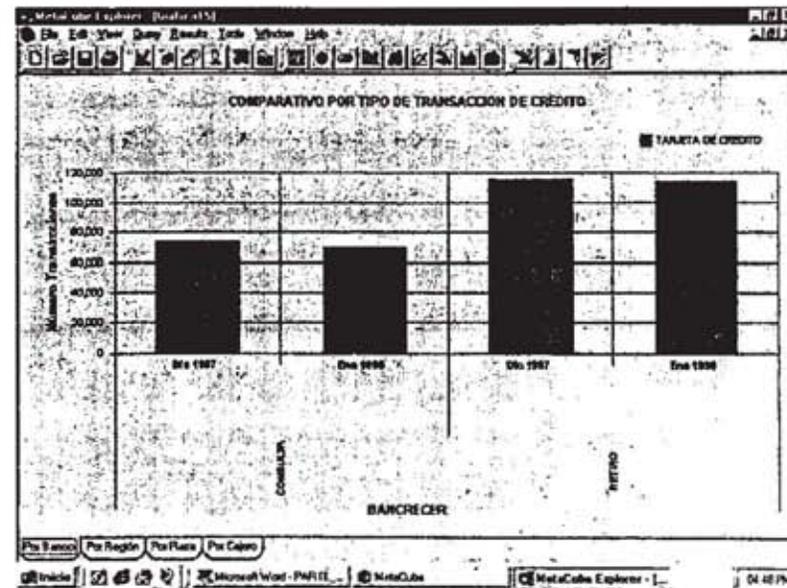
Results Mode

Titulo	Eje X	Eje Y	Legend
Format Title	Format Axis Title	Format Axis Title	
Texto: COMPARATIVO POR TIPO DE TRANSACCION DE CREDITO	No visible	Text: Número Transacciones	Des Cuenta
Font: Arial, Negrita, 10		Orientation: Up	Font: Arial, Normal, 8
Location: Custom, Top 0.32 cm		Font: Arial, Negrita, 10	Location: Visible, Top Right
	Format Axis Label	Format Axis Label	
	No visible	Text: Automatic	
		Font: Arial, Normal, 8	
		Format Code: Number #,##0	

The screenshot shows the 'Query Mode' interface. At the top, there are menu options like 'File', 'Edit', 'View', 'Query', 'Results', 'Tools', 'Window', and 'Help'. Below the menu is a toolbar with various icons. The main area is divided into several sections:

- Dimensiones:** Includes 'Banco' (with a dropdown menu), 'Cuenta' (with a dropdown menu), and 'Movimiento' (with a dropdown menu).
- Report Definition:** Contains fields for 'Organiz. Columna By:' (Des Cuenta) and 'Organiz. Filas By:' (Des Dueño).
- Data Table:** A table with columns for 'BANCIEROS', 'DEPOSITO', 'Enero 1997', 'Diciembre 1998', and 'Movimiento Acreditado'. The data shows values of 1,000 for each combination of bank and month.

Query Mode



Results Mode

Gráfica 15a - "COMPARATIVO POR TIPO DE TRANSACCION DE DEBITO"

Esta gráfica es similar a la Gráfica 15, con las siguientes diferencias:

Query Mode

1. En Filtros Fijos cambia Des Cuenta = CUENTA DE CHEQUES, CUENTA MAESTRA.

Results Mode

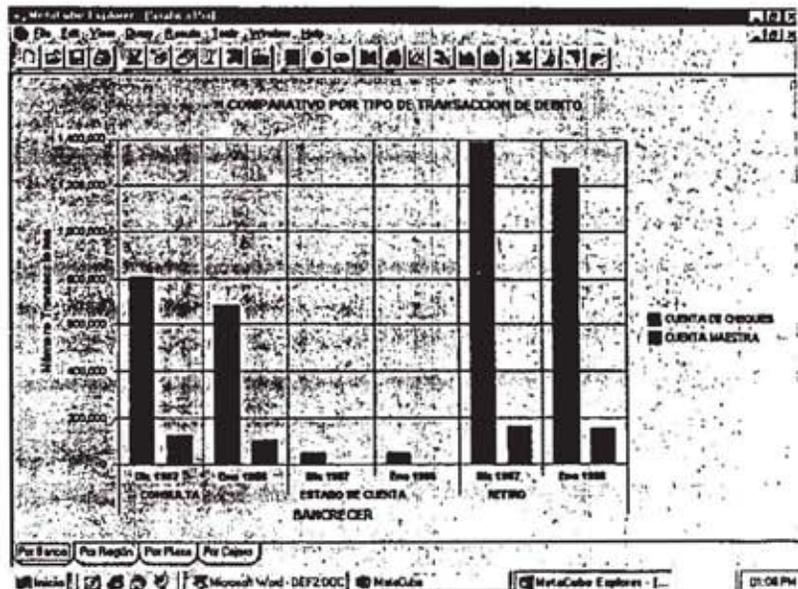
1. El título cambia a "COMPARATIVO POR TIPO DE TRANSACCION DE DEBITO".

A continuación se presentan las imágenes de la definición de esta gráfica.

The screenshot shows the MetaCube Explorer interface in Query Mode. The 'Filtros Fijos' (Fixed Filters) section is configured with 'Des Cuenta' set to 'CUENTA DE CHEQUES' and 'CUENTA MAESTRA'. Below the filters, a table displays account movements for 'SANQUENA' and 'CORIGUENA' from January 1997 to December 1998.

Entidad	Operación	Fecha	Monto
SANQUENA	DEPOSITO	Enero 1997	1,000
		Diciembre 1998	1,000
CORIGUENA	DEPOSITO	Enero 1997	1,000
		Diciembre 1998	1,000
SANQUENA	DEPOSITO	Enero 1997	1,000
		Diciembre 1998	1,000
CORIGUENA	DEPOSITO	Enero 1997	1,000
		Diciembre 1998	1,000

Query Mode



Results Mode

Gráfica 16 - *COMPARATIVO POR TIPO DE TRANSACCION OTROS TARJETAHABIENTES*

Query Mode

Por Banco						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Duenio Movimiento	Des Mes	Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER Des Emisor <> BANCRECER	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Sin filtro
Por Región						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Region Movimiento	Des Mes	Movimientos Aceptados	Banco Dueño Banco Emisor	Des Duenio = BANCRECER Des Emisor <> BANCRECER	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Una Región
Por Plaza						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Plaza Movimiento	Des Mes	Movimientos Aceptados	Banco Dueño Banco Emisor	Des Duenio = BANCRECER Des Emisor <> BANCRECER	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Una Plaza
Por Cajero						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Cajero Movimiento	Des Mes	Movimientos Aceptados	Banco Dueño Banco Emisor	Des Duenio = BANCRECER Des Emisor <> BANCRECER	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Un Cajero por Nombre

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Results Mode

Titulo	Eje X	Eje Y	Legend
Format Title	Format Axis Title	Format Axis Title	
Texto: COMPARATIVO POR TIPO DE TRANSACCION OTROS TARJETAHABIENTES	No visible	Text: Número Transacciones	Des Mes
Font: Arial, Negrita, 10		Orientation: Up	Font: Arial, Normal, 8
Location: Custom, Top 0.32 cm		Font: Arial, Negrita, 10	Location: Visible, Custom, Top 1.07 cm
	Format Axis Label	Format Axis Label	
	Text:	Text: Automatic	
	Font: Arial, Normal, 8	Font: Arial, Normal, 8	
	Format Code:	Format Code: Number #,##0	

MetaCube Explorer - [Parte 11]

File Edit View Query Results Tools Window Help

Dimensiones:

- Cuentas: Des Cuenta, Des Ene, Des Cuenta, Des Cuenta
- Movimientos: Des Sector, Des Liberación, Des Región, Des País, Des Cajas, Des Cajas
- Movimientos: Des Sector, Des Liberación, Des Región, Des País, Des Cajas, Des Cajas

Report Definition:

Operate From By:

Des Cuenta, Movimiento

Operate Measure By:

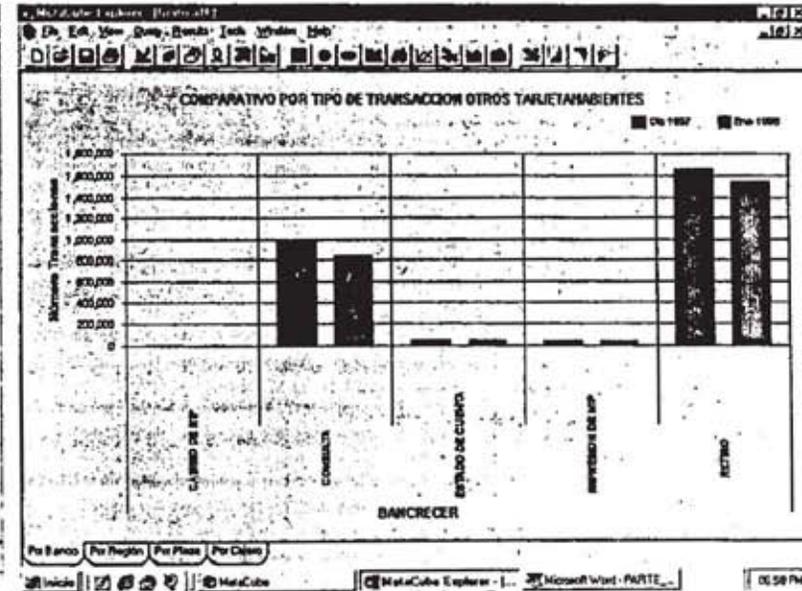
Des Cuenta, Movimiento

Cuenta	Movimiento	Monto
BANCOER DEPOSITO	Movimiento Acreditado	1,000
BANCOER CONSIGLIA		1,000
BANCOER DEPOSITO		1,000
BANCOER CONSIGLIA		1,000

Por Banco Por Región Por Plaza Por Cajas

Inicio MetaCube Explorer - [Parte 11] Microsoft Word - PARTE... 06:50 PM

Query Mode



Results Mode

Gráfica 17 - *NUMERO DE DISPOSICIONES RECHAZADAS POR RAZON DE RESPUESTA*

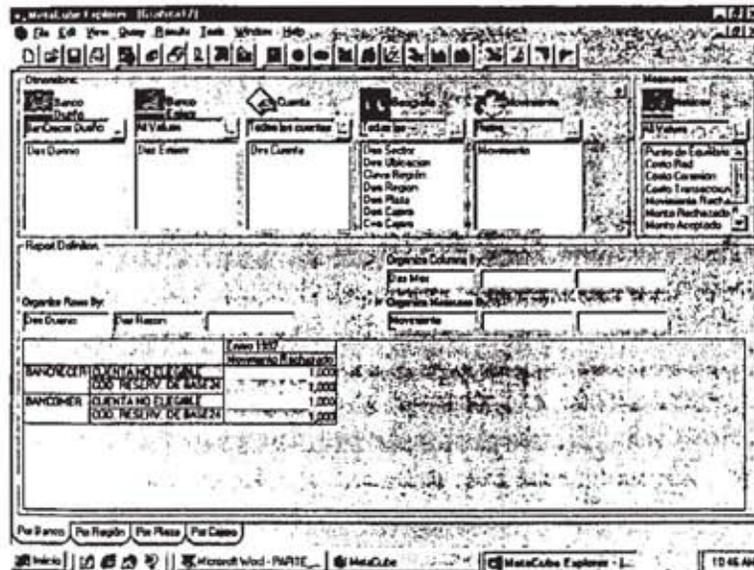
Query Mode

Por Banco						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Duenio, Des Razon	Des Mes	Movimiento Rechazado	Banco Dueño Movimiento	Des Duenio = BANCRECER Movimiento=RETIRO	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un sólo mes Sin filtro
Por Región						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Region, Des Razon	Des Mes	Movimiento Rechazado	Banco Dueño Movimiento	Des Duenio = BANCRECER Movimiento=RETIRO	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un sólo mes Una Región
Por Plaza						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Plaza Des Razon	Des Mes	Movimiento Rechazado	Banco Dueño Movimiento	Des Duenio = BANCRECER Movimiento=RETIRO	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un sólo mes Una Plaza
Por Cajero						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Cajero Des Razon	Des Mes	Movimiento Rechazado	Banco Dueño Movimiento	Des Duenio = BANCRECER Movimiento=RETIRO	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un sólo mes Un Cajero por Nombre

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Results Mode

Titulo	Eje X	Eje Y	Legend
Format Title	Format Axis Title	Format Axis Title	
Texto: NUMERO DE DISPOSICIONES RECHAZADAS POR RAZON DE RESPUESTA	No visible	Text: Número Disposiciones	Des Mes
Font: Arial, Negrita, 10		Orientation: Up	Font: Arial, Normal, 8
Location: Visible, Custom, Top 0.32 cm		Font: Arial, Negrita, 10	Location: Visible, Custom, Top 1.12 cm
	Format Axis Label	Format Axis Label	
	Text: Automatic	Text: Automatic	
	Font: Arial, Negrita, 8	Font: Arial, Normal, 8	
	Format Code:	Format Code: Number #,##0	



Query Mode



Results Mode

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

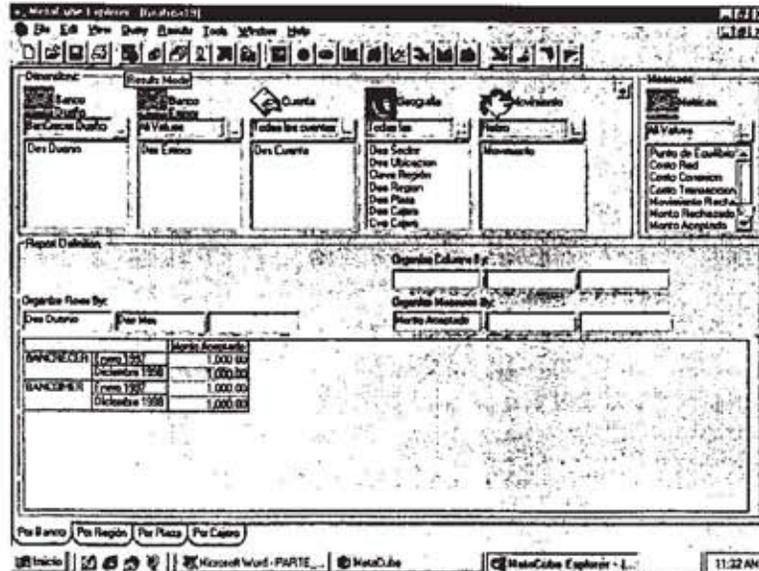
Gráfica 19 – *MONTOS DISPUESTOS POR MES*

Query Mode

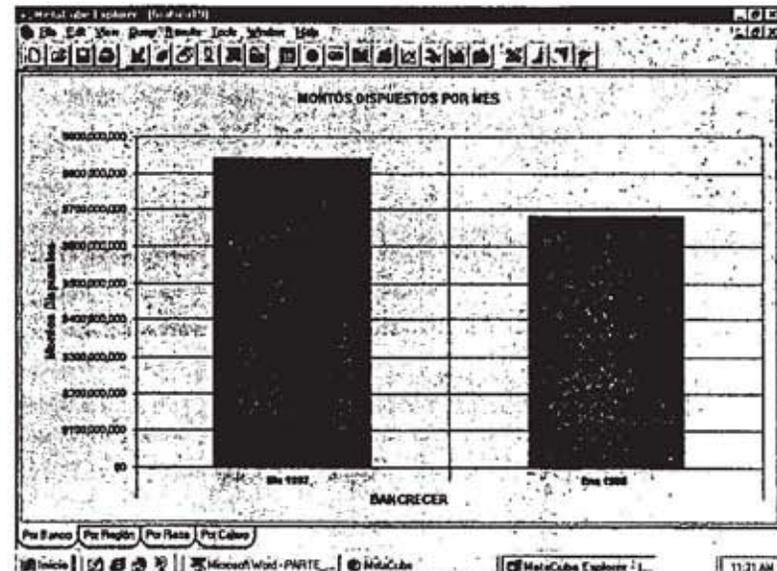
Por Banco						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Duenio	Ninguno	Monto Aceptado	Banco Dueño Movimiento	Des Duenio = BANCRECER Movimiento = RETIRO	Cuenta Tiempo Geografía	Uno o todos los tipos de Cuentas Un mes o bimestre Sin filtro
Des Mes						
Por Región						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Region	Ninguno	Monto Aceptado	Banco Dueño Movimiento	Des Duenio = BANCRECER Movimiento = RETIRO	Cuenta Tiempo Geografía	Uno o todos los tipos de Cuentas Un mes o bimestre Una Región
Des Mes						
Por Plaza						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Plaza	Ninguno	Monto Aceptado	Banco Dueño Movimiento	Des Duenio = BANCRECER Movimiento = RETIRO	Cuenta Tiempo Geografía	Uno o todos los tipos de Cuentas Un mes o bimestre Una Plaza
Des Mes						
Por Cajero						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Cajero	Ninguno	Monto Aceptado	Banco Dueño Movimiento	Des Duenio = BANCRECER Movimiento = RETIRO	Cuenta Tiempo Geografía	Uno o todos los tipos de Cuentas Un mes o bimestre Un Cajero por Nombre
Des Mes						

Results Mode

Titulo	Eje X	Eje Y	Legend
Format Title	Format Axis Title	Format Axis Title	
Texto: MONTOS DISPUESTOS POR MES	No visible	Text: Montos Dispuestos	No visible
Font: Arial, Negrita, 10		Orientation: Up	
Location: Visible, Custom, Top 0.32 cm		Font: Arial, Negrita, 10	
	Format Axis Label	Format Axis Label	
	Text: Automatic	Text: Automatic	
	Font: Arial, Negrita, 8	Font: Arial, Normal, 8	
	Format Code:	Format Code: Currency	
		\$\$,##0;(Red)(\$\$,##0)	



Query Mode



Results Mode

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Gráfica 20 - "MONTOS DISPUESTOS POR MES NUESTROS TARJETAHABIENTES Y OTROS TARJETAHABIENTES"

Query Mode

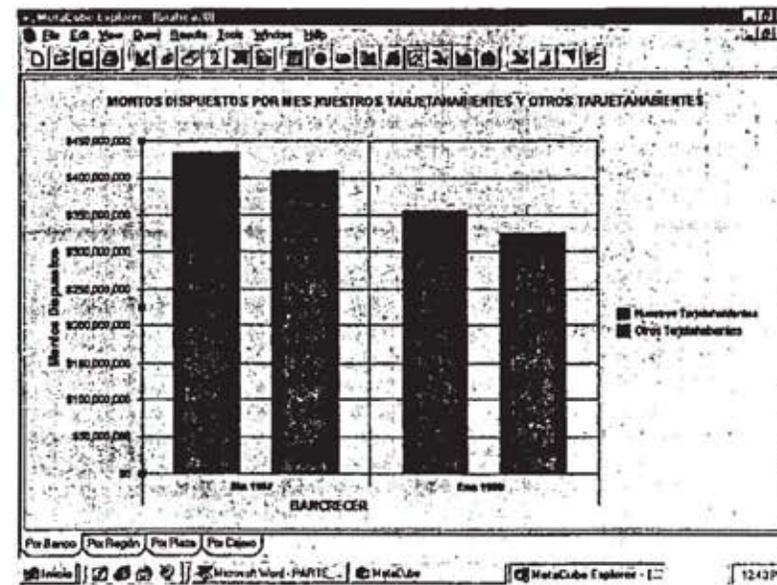
Por Banco						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Duenio Des Mes	Des Emisor ** BUCKET(Attributes.Banco Emisor.Des Emisor,[Nuestros Tarjetahabientes, LIST("BANCRECER ")]). [Otros Tarjetahabientes, OTHER])	Monto Aceptado	Banco Dueño Movimiento	Des Duenio = BANCRECER Movimiento = RETIRO	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Sin filtro
Por Región						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Region Des Mes	Des Emisor ** BUCKET(Attributes.Banco Emisor.Des Emisor,[Nuestros Tarjetahabientes, LIST("BANCRECER ")]). [Otros Tarjetahabientes, OTHER])	Monto Aceptado	Banco Dueño Movimiento	Des Duenio = BANCRECER Movimiento = RETIRO	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Una Región
Por Plaza						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Plaza Des Mes	Des Emisor ** BUCKET(Attributes.Banco Emisor.Des Emisor,[Nuestros Tarjetahabientes, LIST("BANCRECER ")]). [Otros Tarjetahabientes, OTHER])	Monto Aceptado	Banco Dueño Movimiento	Des Duenio = BANCRECER Movimiento = RETIRO	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Una Plaza
Por Cajero						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Cajero Des Mes	Des Emisor ** BUCKET(Attributes.Banco Emisor.Des Emisor,[Nuestros Tarjetahabientes, LIST("BANCRECER ")]). [Otros Tarjetahabientes, OTHER])	Monto Aceptado	Banco Dueño Movimiento	Des Duenio = BANCRECER Movimiento = RETIRO	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Un Cajero por Nombre (D)

Results Mode

Título	Eje X	Eje Y	Legend
Format Title	Format Axis Title	Format Axis Title	
Texto: MONTOS DISPUESTOS POR MES NUESTROS TARJETAHABIENTES Y OTROS TARJETAHABIENTES	No visible	Text: Montos Dispuestos	Des Emisor
Font: Arial, Negrita, 10		Orientation: Up	Font: Arial, Normal, 8
Location: Visible, Custom, Top 0.32 cm		Font: Arial, Negrita, 10	
	Format Axis Label	Format Axis Label	
	Text: Automatic	Text: Automatic	Location: Visible, Custom, Top 6.33 cm
	Font: Arial, Negrita, 8	Font: Arial, Normal, 8	
	Format Code:	Format Code: Currency \$#,##0;[Red](\$#,##0)	

Query Mode

Organize Rows By	Organize Columns By																														
<table border="1"> <tr> <td></td> <td>Des Emisor</td> <td>Des Mes</td> </tr> <tr> <td>BANCO CEE</td> <td>Ene 1999</td> <td>1,000.00</td> </tr> <tr> <td></td> <td>Dic 1999</td> <td>1,000.00</td> </tr> <tr> <td>BANCO CEE</td> <td>Ene 1999</td> <td>1,000.00</td> </tr> <tr> <td></td> <td>Dic 1999</td> <td>1,000.00</td> </tr> </table>		Des Emisor	Des Mes	BANCO CEE	Ene 1999	1,000.00		Dic 1999	1,000.00	BANCO CEE	Ene 1999	1,000.00		Dic 1999	1,000.00	<table border="1"> <tr> <td></td> <td>Des Emisor</td> <td>Des Mes</td> </tr> <tr> <td>BANCO CEE</td> <td>Ene 1999</td> <td>1,000.00</td> </tr> <tr> <td></td> <td>Dic 1999</td> <td>1,000.00</td> </tr> <tr> <td>BANCO CEE</td> <td>Ene 1999</td> <td>1,000.00</td> </tr> <tr> <td></td> <td>Dic 1999</td> <td>1,000.00</td> </tr> </table>		Des Emisor	Des Mes	BANCO CEE	Ene 1999	1,000.00		Dic 1999	1,000.00	BANCO CEE	Ene 1999	1,000.00		Dic 1999	1,000.00
	Des Emisor	Des Mes																													
BANCO CEE	Ene 1999	1,000.00																													
	Dic 1999	1,000.00																													
BANCO CEE	Ene 1999	1,000.00																													
	Dic 1999	1,000.00																													
	Des Emisor	Des Mes																													
BANCO CEE	Ene 1999	1,000.00																													
	Dic 1999	1,000.00																													
BANCO CEE	Ene 1999	1,000.00																													
	Dic 1999	1,000.00																													



CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Gráfica 21 - "MONTOS DISPUESTOS POR MES CREDITO"

Query Mode

Por Banco						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Duenio	Ninguno	Monto Aceptado	Banco Dueño Cuenta Movimiento	Des Duenio = BANCRECER Des Cuenta = TARJETA DE CREDITO Movimiento = RETIRO	Tiempo Geografia	Un mes o bimestre Sin filtro
Des Mes						
Por Región						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Region	Ninguno	Monto Aceptado	Banco Dueño Cuenta Movimiento	Des Duenio = BANCRECER Des Cuenta = TARJETA DE CREDITO Movimiento = RETIRO	Tiempo Geografia	Un mes o bimestre Una Región
Des Mes						
Por Plaza						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Plaza	Ninguno	Monto Aceptado	Banco Dueño Cuenta Movimiento	Des Duenio = BANCRECER Des Cuenta = TARJETA DE CREDITO Movimiento = RETIRO	Tiempo Geografia	Un mes o bimestre Una Plaza
Des Mes						
Por Cajero						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Cajero	Ninguno	Monto Aceptado	Banco Dueño Cuenta Movimiento	Des Duenio = BANCRECER Des Cuenta = TARJETA DE CREDITO Movimiento = RETIRO	Tiempo Geografia	Un mes o bimestre Un Cajero por Nombre (D)
Des Mes						

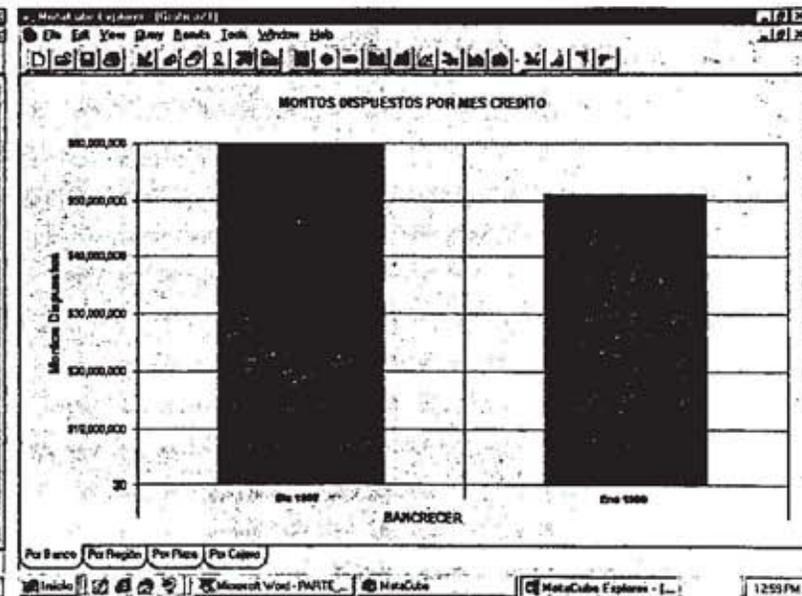
CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Results Mode

Título	Eje X	Eje Y	Legend
Format Title	Format Axis Title	Format Axis Title	
Texto: MONTOS DISPUESTOS POR MES CREDITO	No visible	Text: Montos Dispuestos	No visible
Font: Arial, Negrita, 10		Orientation: Up	
Location: Visible, Custom, Top 0.32 cm		Font: Arial, Negrita, 10	
	Format Axis Label	Format Axis Label	
	Text: Automatic	Text: Automatic	
	Font: Arial, Negrita, 8	Font: Arial, Normal, 8	
	Format Code:	Format Code: Currency	
		\$#,##0;(Red)(\$#,##0)	

Query Mode

Query Mode



Results Mode

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Gráfica 21a - *MONTOS DISPUESTOS POR MES DEBITO*

Query Mode

Por Banco						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Duenio	Des Cuenta **	Monto Aceptado	Banco Dueño	Des Duenio = BANCRECER Des Cuenta = CUENTA DE CHEQUES, CUENTA MAESTRA Movimiento = RETIRO	Tiempo	Un mes o bimestre
Des Mes	BUCKET(Attributes.Cuenta.Des Cuenta,[Cuenta de Cheques LIST("CUENTA DE CHEQUE ")],[Cuenta Maestra LIST("CUENTA MAESTRA ")]))		Cuenta		Geografía	Sin filtro
			Movimiento			
Por Región						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Region	Des Cuenta **	Monto Aceptado	Banco Dueño	Des Duenio = BANCRECER Des Cuenta = CUENTA DE CHEQUES, CUENTA MAESTRA Movimiento = RETIRO	Tiempo	Un mes o bimestre
Des Mes	BUCKET(Attributes.Cuenta.Des Cuenta,[Cuenta de Cheques LIST("CUENTA DE CHEQUE ")],[Cuenta Maestra LIST("CUENTA MAESTRA ")]))		Cuenta		Geografía	Una Región
			Movimiento			
Por Plaza						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Plaza	Des Cuenta **	Monto Aceptado	Banco Dueño	Des Duenio = BANCRECER Des Cuenta = CUENTA DE CHEQUES, CUENTA MAESTRA Movimiento = RETIRO	Tiempo	Un mes o bimestre
Des Mes	BUCKET(Attributes.Cuenta.Des Cuenta,[Cuenta de Cheques LIST("CUENTA DE CHEQUE ")],[Cuenta Maestra LIST("CUENTA MAESTRA ")]))		Cuenta		Geografía	Una Plaza
			Movimiento			

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Por Cajero						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Cajero Des Mes	Des Cuenta ** BUCKET(Attributes.Cuenta.Des Cuenta,[Cuenta de Cheques LIST("CUENTA DE CHEQUE "),[Cuenta Maestra LIST("CUENTA MAESTRA ")])	Monto Aceptado	Banco Dueño Cuenta Movimiento	Des Duenio = BANCRECER Des Cuenta = CUENTA DE CHEQUES, CUENTA MAESTRA Movimiento = RETIRO	Tiempo Geografía	Un mes o bimestre Un Cajero por Nombre (D)

Results Mode

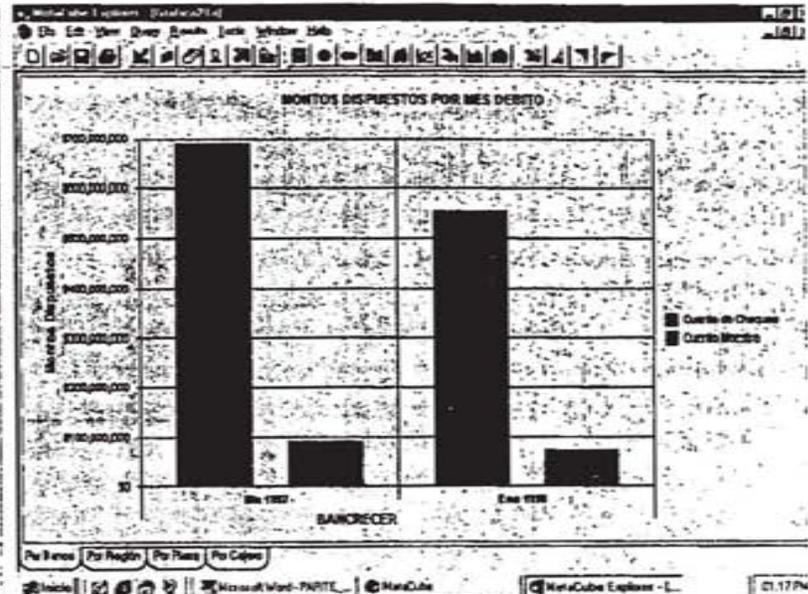
Título	Eje X	Eje Y	Legend
Format Title	Format Axis Title	Format Axis Title	
Texto: MONTOS DISPUESTOS POR MES DEBITO	No visible	Text: Montos Dispuestos	Des Cuenta
Font: Arial, Negrita, 10		Orientation: Up	Font: Arial, Normal, 8
Location: Visible, Custom, Top 0.32 cm		Font: Arial, Negrita, 10	Location: Visible, Custom, Top 6.33 cm
	Format Axis Label	Format Axis Label	
	Text: Automatic	Text: Automatic	
	Font: Arial, Negrita, 8	Font: Arial, Normal, 8	
	Format Code:	Format Code: Currency	
		\$\$,##0:[Red](\$\$,##0)	

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

The screenshot shows the MetaCube Explorer interface in Query Mode. It features several filter sections: 'Banco' (with 'Banca Credito' selected), 'Cuenta' (with 'Cuentas de Debito' selected), 'Geografico' (with 'Mexico' selected), and 'Moneda' (with 'Pesos' selected). Below these are 'Report Definition' and 'Deposits Filtered By' sections. A data table is displayed with the following content:

Depositor	Deposited	Moneda	Monto
BANCRECER	1997	1,000.00	
BANCRECER	Diciembre 1997	1,000.00	
BANCRECER	Enero 1997	1,000.00	
BANCRECER	Diciembre 1997	1,000.00	

Query Mode



Results Mode

Gráfica 22 - *TRANSACCIONES ACEPTADAS POR MES*

Query Mode

Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Duenio		Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Cuenta Tiempo Geografía	Uno o todos los tipos de Cuentas Un mes o bimestre Sin filtro
Des Mes						
Por Región						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Region		Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Cuenta Tiempo Geografía	Uno o todos los tipos de Cuentas Un mes o bimestre Una Región
Des Mes						
Por Plaza						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Plaza		Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Cuenta Tiempo Geografía	Uno o todos los tipos de Cuentas Un mes o bimestre Una Plaza
Des Mes						
Por Cajero						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Cajero		Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Cuenta Tiempo Geografía	Uno o todos los tipos de Cuentas Un mes o bimestre Un Cajero por Nombre (D)
Des Mes						

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Results Mode

Título	Eje X	Eje Y	Legend
Format Title	Format Axis Title	Format Axis Title	
Texto: TRANSACCIONES ACEPTADAS POR MES	No visible	Text: Número Transacciones	No visible
Font: Arial, Negrita, 10		Orientation: Up	
Location: Visible, Custom, Top 0.32 cm		Font: Arial, Negrita, 10	
	Format Axis Label	Format Axis Label	
	Text: Automatic	Text: Automatic	
	Font: Arial, Negrita, 8	Font: Arial, Normal, 8	
	Format Code:	Format Code: Number #,##0	

MetaCube Explorer [Bancrec/2]

Dimensiones:

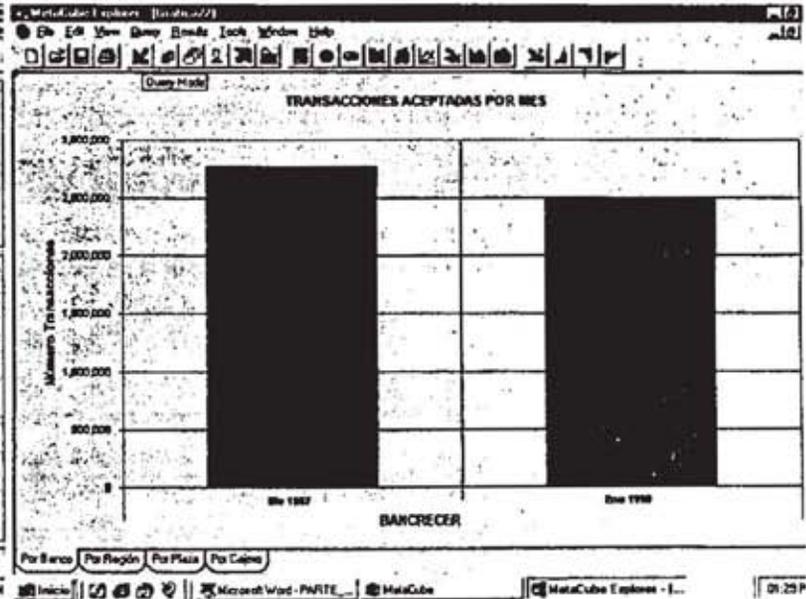
- Bank: Bancrec
- Plaza: Plaza
- Cuenta: Cuenta
- Geografía: Geografía
- Movimiento: Movimiento
- Medidas: Medidas

Report Definition:

Organiza Rows By:

Bank	Plaza	Monto Aceptado
BANCRECER	Plaza 1997	1,000
BANCRECER	Diciembre 1998	1,000
BANCRECER	Plaza 1997	1,000
BANCRECER	Diciembre 1998	1,000

Query Mode



Results Mode

Gráfica 23 - *TRANSACCIONES ACEPTADAS POR MES NUESTROS TARJETAHABIENTES Y OTROS TARJETAHABIENTES*

Query Mode

Por Banco						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Duenio Des Mes	Des Emisor ** BUCKET(Attributes.Banco Emisor.Des Emisor, [Nuestros Tarjetahabientes, LIST("BANCRECER ")), [Otros Tarjetahabientes, OTHER])	Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Sin filtro
Por Región						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Region Des Mes	Des Emisor ** BUCKET(Attributes.Banco Emisor.Des Emisor, [Nuestros Tarjetahabientes, LIST("BANCRECER ")), [Otros Tarjetahabientes, OTHER])	Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Una Región
Por Plaza						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Plaza Des Mes	Des Emisor ** BUCKET(Attributes.Banco Emisor.Des Emisor, [Nuestros Tarjetahabientes, LIST("BANCRECER ")), [Otros Tarjetahabientes, OTHER])	Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Una Plaza

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Por Cajero						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Cajero Des Mes	Des Emisor ** BUCKET(Attributes.Banco Emisor.Des Emisor, [Nuestros Tarjetahabientes, LIST("BANCRECER ")], [Otros Tarjetahabientes, OTHER])	Movimientos Aceptados	Banco Dueño	Des Duenio BANCRECER	= Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Un Cajero por Nombre (D)

Results Mode

Título	Eje X	Eje Y	Legend
Format Title	Format Axis Title	Format Axis Title	
Texto: TRANSACCIONES ACEPTADAS POR MES NUESTROS TARJETAHABIENTES Y OTROS TARJETAHABIENTES	No visible	Text: Número Transacciones	Des Emisor
Font: Arial, Negrita, 10		Orientation: Up	Font: Arial, Normal, 8
Location: Visible, Top		Font: Arial, Negrita, 10	Location: Visible, Custom, Top 6.40 cm
	Format Axis Label	Format Axis Label	
	Text: Automatic	Text: Automatic	
	Font: Arial, Negrita, 8	Font: Arial, Normal, 8	
	Format Code:	Format Code: Number #,##0	

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

MetaCube Explorer [Gráfico2]

File Edit View Query Results Tools Window Help

Dimensiones: Results Mode

Banko Cuenta Geografía Movimiento Medida

Banko Cuenta Geografía Movimiento Medida

Report Definition

Organize Rows By:

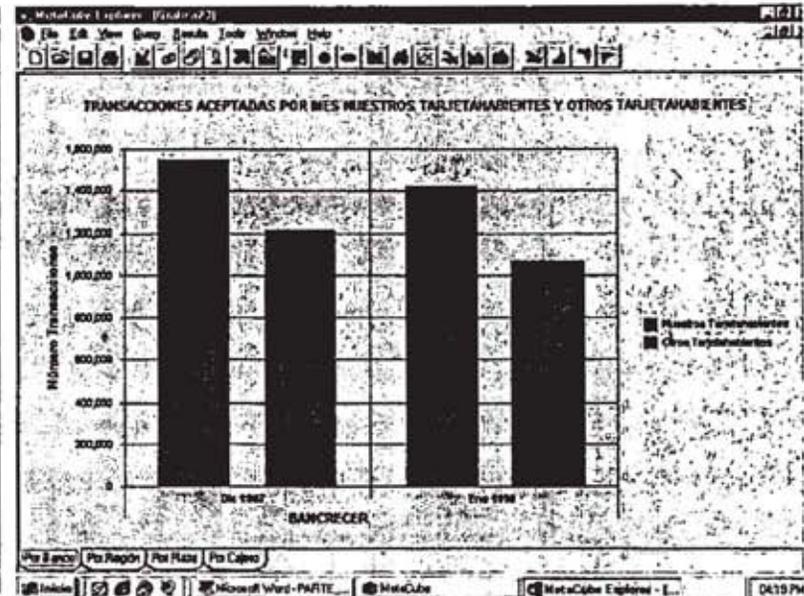
Organize Columns By:

Banko	Periodo	Movimiento Aceptado
BANCRECER	enero 1998	1.000
BANCOMER	enero 1998	1.000
BANCOMER	abril 1998	1.000

Por Banco Por Región Por Plaza Por Cajero

Inicio MetaCube Explorer - 04:13 PM

Query Mode



Results Mode

Gráfica 24 - "TRANSACCIONES ACEPTADAS POR MES CREDITO"

Query Mode

Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Duenio	Ninguno	Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Tiempo	Un mes o bimestre
Des Mes			Cuenta	Des Cuenta = TARJETA DE CREDITO	Geografía	Sin filtro
Por Región						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Region	Ninguno	Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Tiempo	Un mes o bimestre
Des Mes			Cuenta	Des Cuenta = TARJETA DE CREDITO	Geografía	Una Región
Por Plaza						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Plaza	Ninguno	Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Tiempo	Un mes o bimestre
Des Mes			Cuenta	Des Cuenta = TARJETA DE CREDITO	Geografía	Una Plaza
Por Cajero						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Cajero	Ninguno	Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Tiempo	Un mes o bimestre
Des Mes			Cuenta	Des Cuenta = TARJETA DE CREDITO	Geografía	Un Cajero por Nombre

Results Mode

Titulo	Eje X	Eje Y	Legend
Format Title	Format Axis Title	Format Axis Title	
Texto: TRANSACCIONES ACEPTADAS POR MES CREDITO	No visible	Text: Número Transacciones	No visible
Font: Arial, Negrita, 10		Orientation: Up	
Location: Visible, Custom, Top 0.32 cm		Font: Arial, Negrita, 10	
	Format Axis Label	Format Axis Label	
	Text: Automatic	Text: Automatic	
	Font: Arial, Negrita, 8	Font: Arial, Normal, 8	
	Format Code:	Format Code: Number #.##0	

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Microsoft Excel - [Moneda]

File Edit View Query Results Tools Help

Dimensiones:

Banco: Banco, Cuenta: Cuenta, Grupo de: Grupo de, Movimiento: Movimiento, Mensaje: Mensaje

Report Definition:

Organize Rows By: Des. Cuenta, Des. Mes

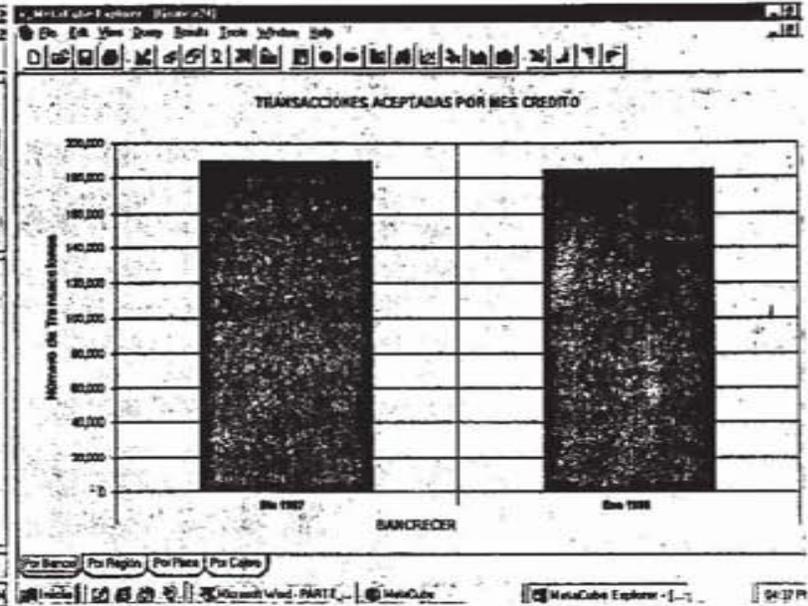
Organize Columns By: Des. Cuenta, Des. Mes

BANQUE	Mes	Movimientos Aprobados
BANQUE	Septiembre 1998	1,000
BANQUE	Diciembre 1998	1,000
BANQUE	Enero 1999	1,000
BANQUE	Diciembre 1999	1,000

Por Banco | Por Región | Por Plaza | Por Cajero

Inicio | Microsoft Word - PARTE... | MetaCube | Microsoft Excel - [Moneda] | 04:36 PM

Query Mode



Results Mode

Gráfica 24a - "TRANSACCIONES ACEPTADAS POR MES DEBITO"

Esta gráfica es similar a la Gráfica 24, con las siguientes diferencias:

Query Mode

1. En Filtros Fijos se cambia Des Cuenta = "CUENTA DE CHEQUES", "CUENTA MAESTRA"

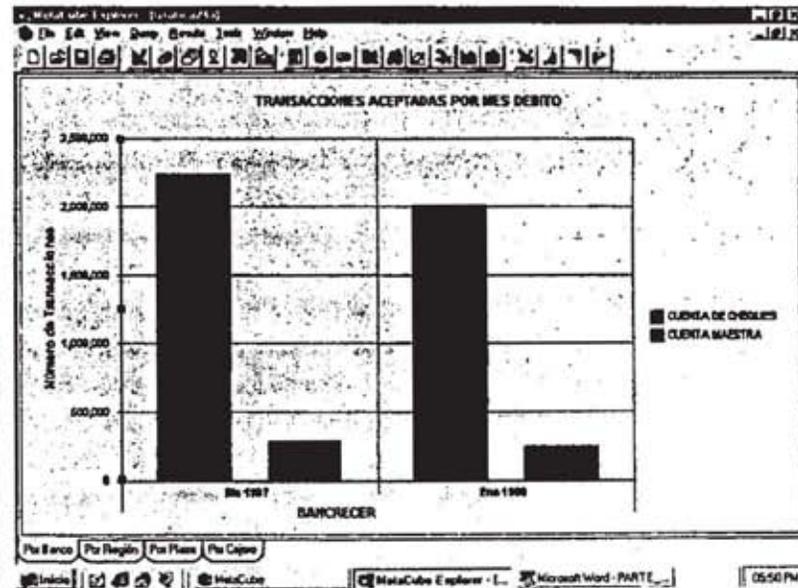
Results Mode

1. El título cambia a "TRANSACCIONES ACEPTADAS POR MES DEBITO".

A continuación se presentan las imágenes de la definición de esta gráfica.



Query Mode



Results Mode

4.2 REPORTES

Para los reportes se presenta solamente el "Query Mode" ya que el formateo de "Results Mode" lo realiza automáticamente MetaCube Explorer.

A continuación se presenta la especificación de cada uno de los reportes.

Reporte 01 - "Retiros todos los Tarjetahabientes en cajeros propios"

Query Mode

Por Banco						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Duenio Des Emisor	Des Cuenta * BUCKET(Attributes.Cuenta.Des Cuenta, [Cuenta de Cheques, LIST("CUENTA DE CHEQUES ")], [Cuenta Maestra, LIST("CUENTA MAESTRA ")]))	Monto Aceptado	Banco Dueño	Des Duenio = BANCRECER	Cuenta Tiempo Geografía	Uno o todos los tipos de Cuentas Un mes solamente Sin filtro
Por Región						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Region Des Emisor	Des Cuenta * BUCKET(Attributes.Cuenta.Des Cuenta, [Cuenta de Cheques, LIST("CUENTA DE CHEQUES ")], [Cuenta Maestra, LIST("CUENTA MAESTRA ")]))	Monto Aceptado	Banco Dueño	Des Duenio = BANCRECER	Cuenta Tiempo Geografía	Uno o todos los tipos de Cuentas Un mes o bimestre Una Región
Por Plaza						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Des Plaza Des Emisor	Des Cuenta * BUCKET(Attributes.Cuenta.Des Cuenta, [Cuenta de Cheques, LIST("CUENTA DE CHEQUES "), [Cuenta Maestra, LIST("CUENTA MAESTRA ")])	Monto Aceptado	Banco Dueño	Des Duenio = BANCRECER	Cuenta Tiempo Geografia	Uno o todos los tipos de Cuentas Un mes o bimestre Una Plaza
-------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------	-------------	---------------------------	-------------------------------	-----------------------------------------------------------------------

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Reporte 02 - 'Montos Dispuestos de todos los Tarjetahabientes'

Query Mode

Por Región						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Region		Movimientos Aceptados	Movimiento	Movimiento = RETIRO	Cuenta	Uno o todos los tipos de Cuentas
Cve Cajero		Monto Aceptado			Tiempo	Un mes solamente
Des Cajero					Geografia	Sin filtro
Des Emisor						
Por Plaza						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Plaza		Movimientos Aceptados	Movimiento	Movimiento = RETIRO	Cuenta	Uno o todos los tipos de Cuentas
Cve Cajero		Monto Aceptado			Tiempo	Un mes solamente
Des Cajero					Geografia	Una Plaza
Des Emisor						
Por Cajero						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Cve Cajero		Movimientos Aceptados	Movimiento	Movimiento = RETIRO	Cuenta	Uno o todos los tipos de Cuentas
Des Cajero		Monto Aceptado			Tiempo	Un mes solamente
Des Emisor					Geografia	Un Cajero por Nombre (D)

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

MediCable Express - [Reporte] [F12]

File Edit View Query Search Tools Window Help

Dimensiones: Banco, Cuenta, Cuente, Geografica, Movimiento, Medico

Report Definition:

Organiza Rows By: Des Region, Des Centro, Des Centro

Organiza Columns By: Des Centro, Des Centro

ORGANIZACION	DESCRIPCION	MOVIMIENTO	MONEDAS ACREDITADAS	MONTOS ACREDITADOS
CARRIZOPAL	CARRIZOPAL ALTERNATIVA	BANDRETER	1,000	\$1,000.00
		BANDRETER	1,000	\$1,000.00
		BANDRETER	1,000	\$1,000.00
CARRIZOPAL	CARRIZOPAL ALTERNATIVA	BANDRETER	1,000	\$1,000.00
		BANDRETER	1,000	\$1,000.00
		BANDRETER	1,000	\$1,000.00
CARRIZOPAL	CARRIZOPAL ALTERNATIVA	BANDRETER	1,000	\$1,000.00
		BANDRETER	1,000	\$1,000.00
		BANDRETER	1,000	\$1,000.00

Montos Disponibles de todos los Trabajadores-Por Region
Montos Disponibles de todos los Trabajadores-Por Centro

Query Mode

MediCable Express - [Reporte] [F12]

File Edit View Query Search Tools Window Help

Region	Centro	Centro	Centro	Movimientos Acreditados	Montos Acreditados
MONTENEGRO	BASICA 31	ATLANTICO	4	4	\$4.00
		BARRANQUILLA	4	4	\$4.00
		BANDRETER	150	150	\$150.00
		BANDRETER	1,000	1,000	\$1,000.00
		BANDRETER	200	200	\$200.00
		BANDRETER	100	100	\$100.00
		BANDRETER	100	100	\$100.00
		BANDRETER	100	100	\$100.00
		BANDRETER	100	100	\$100.00
		BANDRETER	100	100	\$100.00
		BANDRETER	100	100	\$100.00
		BANDRETER	100	100	\$100.00
		BANDRETER	100	100	\$100.00
		BANDRETER	100	100	\$100.00
		BANDRETER	100	100	\$100.00
		BANDRETER	100	100	\$100.00
		BANDRETER	100	100	\$100.00
		BANDRETER	100	100	\$100.00
		BANDRETER	100	100	\$100.00
		MONTENEGRO	HOSPITAL CIVIL	ATLANTICO	100
BANDRETER	100			100	\$100.00

Montos Disponibles de todos los Trabajadores-Por Region
Montos Disponibles de todos los Trabajadores-Por Centro

Results Mode

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Reporte 03 - "Montos dispuestos de todos los tarjetahabientes en cajeros de BanCreceer"

Query Mode

Por Banco						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Duenio	Ninguno	Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Cuenta	Uno o todos los tipos de Cuentas
Des Region		Monto Aceptado	Movimiento	Movimiento = RETIRO	Tiempo Geografia	Un mes solamente Sin filtro
Por Región						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Region	Ninguno	Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Cuenta	Uno o todos los tipos de Cuentas
Des Plaza		Monto Aceptado	Movimiento	Movimiento = RETIRO	Tiempo Geografia	Un mes o bimestre Una Región
Por Plaza						
Elementos		Métricas	Filtros Fijos		Filtros Variables	
Por Renglón	Por Columna		Dimensión	Descripción	Dimensión	Descripción
Des Plaza	Ninguno	Movimientos Aceptados	Banco Dueño	Des Duenio = BANCRECER	Cuenta	Uno o todos los tipos de Cuentas
Des Cajero		Monto Aceptado	Movimiento	Movimiento = RETIRO	Tiempo Geografia	Un mes o bimestre Una Plaza

CAPITULO 4. IMPLEMENTACION DE LA SOLUCION

Query Mode

	Movimiento Asignación	Monto Asignado
MONTELEONE	25,500	113,800,000.00
CANCUN	100,000	217,000,000.00
OBTALAMPA	58,700	210,500,000.00
OBTALAMPA	100,000	200,000,000.00
EXTERNA	4,700	2,000,000.00
OBTALAMPA	20,500	84,000,000.00
LEON	120,700	200,000,000.00
MONTESQUITANA	100,000	200,000,000.00
MONTELEONE	75,700	100,000,000.00
PUEBLA	10,000	10,000,000.00
TAMPICO	100,000	100,000,000.00
TULAMA	77,000	100,000,000.00
VALLE DE TOLUCA	50,000	100,000,000.00
TOTAL	1,541,100	2,000,000,000.00

Results Mode

CAPITULO 5. PRUEBAS A LA SOLUCION

La actividad de pruebas se puede considerar como una parte de las labores propias del aseguramiento de calidad, ya que las pruebas deben demostrar que el sistema que soluciona el problema funciona como lo espera el usuario final o que simplemente no funciona. La calidad del sistema depende de la correcta definición inicial de los requerimientos hasta su implementación en un sistema real. En estas actividades están involucradas las áreas de Diseño, Desarrollo e Implantación.

Pero, ¿Por qué son necesarias las pruebas?

Las pruebas a los sistemas son necesarias por cuatro razones principales:

- Detectar y resolver errores lo antes posible.
- Verificar que la aplicación cumple con los requerimientos.
- Capacitar y entregarle el sistema a los usuarios.
- Asegurar el funcionamiento de las operaciones.

Para que las pruebas se realicen adecuadamente se debe seguir una metodología. En las siguientes secciones se presentan tanto la metodología seguida como el proceso de las pruebas para validar que el CICA cumple con las expectativas de los usuarios finales.

5.1 TECNICAS DE VERIFICACION Y VALIDACION

Las técnicas de verificación y validación van desde métodos informales, como el chequeo de escritorio, hasta métodos muy formales como las pruebas de corrección. Estas incluyen el análisis dinámico, en el cual el código del programa es ejecutado y el análisis estático en el cual los algoritmos y las estructuras de los programas son examinados. También incluyen una variedad de técnicas manuales y automatizadas. Finalmente, incluyen métodos para estudiar qué funciones se realizan y como se implementan esas funciones.

Las técnicas de verificación y validación se dividen en cuatro categorías:

1. Técnicas estáticas y dinámicas
2. Técnicas formales e informales
3. Técnicas automatizadas y manuales
4. Pruebas funcionales y pruebas estructurales

5.1.1 TECNICAS ESTATICAS Y DINAMICAS

Cuando pensamos en verificación de software, usualmente pensamos en técnicas dinámicas, como la pruebas a los programas que involucran la ejecución del programa. Pero las técnicas estáticas son igualmente importantes por dos razones. Primero, las técnicas dinámicas por si mismas no detectan todos los tipos de errores de software. Por ejemplo, errores de definiciones de datos, variables inicializadas incorrectamente y variables no referenciadas comúnmente no se detectan por las técnicas dinámicas. Las técnicas dinámicas, tampoco pueden ser usadas para verificar los requerimientos o la especificación de diseño si no están escritos de una manera formal y computable.

Las técnicas estáticas usualmente son parte de las técnicas dinámicas. El uso de técnicas dinámicas normalmente sigue tres pasos:

1. Aplicación de técnicas estáticas, como el análisis de flujo de control, análisis del flujo de datos y análisis de complejidad, para determinar que pruebas de software se insertarán en el código de los programas.
2. Ejecución de los programas con las pruebas que generen información de análisis.
3. Análisis de los resultados de la ejecución de los programas

Las técnicas estáticas son importantes para determinar los casos de prueba apropiados y para generar los datos de prueba necesarios para las pruebas dinámicas. También son importantes para demostrar lo correcto del software en etapas tempranas del desarrollo ya que pueden ser usadas para analizar las especificaciones de los requerimientos y del diseño así como el código de los programas.

5.1.2 TECNICAS FORMALES E INFORMALES

La verificación tradicional del software ha sido tratada informalmente. Los métodos como el chequeo de escritorio y las pruebas de programa guiadas principalmente por el instinto del programador han sido las más frecuentemente usadas. Estos métodos informales se han concentrado en la verificación del código de los programas.

Con la llegada del diseño estructurado, los métodos formales, en el sentido de procedimientos disciplinados, se volvieron más ampliamente aceptados y aplicados. Estas técnicas incluyen revisiones e inspecciones estructuradas. Además de que se aplican al código de los programas, se aplican a las especificaciones de requerimientos y de diseño. Sin embargo los métodos formales, en el sentido de pruebas matemáticas, no se introdujeron como parte del diseño estructurado. Los métodos de verificación matemática no han sido ampliamente aceptados como técnicas prácticas de verificación. Esto en parte se debe a que la mayoría de las técnicas formales son técnicas manuales tediosas de aplicar, no solo a todos sino a los programas pequeños.

5.1.3 TECNICAS AUTOMATICAS Y MANUALES

En el compilador existen algunas herramientas automatizadas. Otras existen en preprocesadores y postprocesadores separados. Se utilizan para generar datos de prueba, para comparar resultados esperados y reales de pruebas, para generar y verificar aserciones y para verificar el cumplimiento de estándares. Las herramientas automatizadas son más útiles para verificar la consistencia y la corrección. No son de mucha ayuda para verificar la integridad (verificar que el software realiza todas las funciones requeridas). Los errores de integridad comúnmente son invisibles para los métodos automáticos debido a que solamente analizan lo que sale de la especificación o del código, no lo que se ha dejado fuera. Para la verificación de integridad los métodos manuales normalmente son más exitosos para detectar errores de omisión.

5.1.4 PRUEBAS FUNCIONALES Y PRUEBAS ESTRUCTURALES

Las siguientes son reglas que pueden servir como objetivos de las pruebas:

1. Las pruebas son un proceso que consiste de ejecutar un programa con la intención de encontrar algún error.
2. Un buen caso de prueba es aquel que tiene una alta probabilidad de encontrar un error todavía no descubierto.
3. Una prueba exitosa es aquella que descubre un error aún no descubierto.

Las Pruebas Funcionales o "pruebas de caja negra" tienen que ver con las funciones que el software realiza pero no con como se implementan esas funciones. Las pruebas

funcionales se pueden aplicar a la especificación de requerimientos, a la especificación de diseño o al código del programa. Se usa para la validación del software.

Casos de prueba y datos de prueba para pruebas funcionales se derivan de los requerimientos externos sin tener que ver a los algoritmos de diseño o a la estructura interna de los programas.

Por otro lado, las pruebas estructurales o "pruebas de caja blanca" tienen que ver con la exactitud de la implementación de los programas. Se aplica a la especificación formal de diseño (como el español estructurado) o al código de programas. Los casos de prueba y datos de prueba se derivan de analizar los algoritmos de los programas, la estructura de control, las estructuras de datos y los cálculos aritméticos.

Tanto las pruebas funcionales como las pruebas estructurales se consideran componentes esenciales de los métodos de verificación estructurada. Las pruebas estructuradas son más útiles para encontrar errores de consistencia y de exactitud.

5.2 PROCESO DE PRUEBAS

El proceso de pruebas dentro del enfoque estructurado consiste de cuatro etapas:

1. PRUEBAS UNITARIAS
2. PRUEBAS DE INTEGRACION
3. PRUEBAS DE VALIDACION
4. PRUEBAS DE SISTEMA

Idealmente las etapas del proceso de pruebas deberían realizarse secuencialmente. Por ejemplo, las pruebas de integración deberían terminarse antes de comenzar las pruebas de validación. Dentro de cada etapa, sin embargo, los pasos de las pruebas pueden proceder en paralelo. Más de un módulo se puede probar al mismo tiempo de forma unitaria; en el caso de las pruebas de integración, más de un subsistema o módulo se puede probar al mismo tiempo.

En la tabla que se presenta a continuación están las pruebas funcionales y estructurales en su contexto con las etapas del proceso de pruebas.

Técnica	Descripción	Etapa de pruebas			
		Unitarias	Integración	Validación	Sistema
Pruebas Estructurales (pruebas de caja blanca)	Tienen que ver con la forma en que están implementadas las funciones	•	•		
Pruebas Funcionales (pruebas de caja negra)	Tienen que ver con qué funciones están implementadas pero no como se implementaron			•	•

Tabla 5.1 Técnicas de pruebas y etapas correspondientes

En las secciones que siguen se presenta cada etapa de pruebas. Al principio de este trabajo se hizo notar que el proceso de desarrollo de un sistema se puede ver como un torbellino, moviéndose desde la periferia durante el análisis de requerimientos, la intensidad técnica se acelera durante el diseño y el "embudo" se alcanza cuando se genera el código. Ahora regresamos al torbellino, moviéndose hacia fuera con cada paso que se completa. De la forma en que vemos al torbellino, el concepto de un sistema es todo lo que existe;

cuando salimos del torbellino, se ha desarrollado un software operacional y está listo para la integración con un sistema más grande.

5.2.1 PRUEBAS UNITARIAS

Las pruebas unitarias son pruebas estructurales realizadas al nivel de un módulo de programa. Cada módulo, función o subrutina usualmente se prueba unitariamente por el programador como parte de la etapa de codificación.

Las pruebas unitarias se enfocan al esfuerzo de verificar las unidades más pequeñas del desarrollo de software: el módulo. Las pruebas unitarias siempre son de tipo caja blanca y esta etapa puede llevarse a cabo en paralelo para múltiples módulos.

CONSIDERACIONES PARA LAS PRUEBAS UNITARIAS

Se evalúan cinco características primarias de un módulo durante las pruebas unitarias:

- La interfaz del módulo
- Estructura de datos local
- Rutas de ejecución
- Rutas de manejo de errores
- Condiciones límite que afectan todo lo anterior

Se requieren pruebas del flujo de datos por toda la interfaz del módulo antes de que se inicie cualquier otra prueba. Si los datos no entran y salen apropiadamente, todas las demás pruebas son discutibles.

Cuando un módulo realiza entradas y salidas externas, se deben realizar pruebas adicionales a la interfaz.

La estructura de datos local para un módulo es una fuente común de errores. Los casos de prueba deben estar diseñados para descubrir errores en las siguientes categorías:

1. Declaración incorrecta o inconsistente.
2. Iniciación errónea o valores por defecto.
3. Nombres de variable incorrectos (mal escritos o truncados).
4. Tipos de datos inconsistentes.
5. Flujo bajo, sobre flujo o excepciones de direccionamiento.

Entre los errores potenciales que deben probarse cuando se evalúa el manejo de errores están:

1. La descripción del error no se entiende
2. Los errores que aparecen no corresponden al error encontrado.
3. La condición de error causa una intervención del sistema antes que el manejo de errores.
4. La descripción del error no proporciona suficiente información para ayudar en caso de que se presente el error.

La prueba de límite es la última (y probablemente más importante) tarea en la etapa de las pruebas unitarias. El software frecuentemente falla en los límites, esto es, los errores frecuentemente ocurren cuando el n -ésimo elemento de un arreglo de " n " dimensiones es procesado, cuando la n -ésima repetición de un ciclo es encontrado. Los casos de prueba que ejercitan la estructura de datos, el flujo de control y los valores de los datos por debajo, dentro y por encima de los máximos y mínimos es muy probable que descubran errores.

5.2.2 PRUEBAS DE INTEGRACION

Mientras que las pruebas unitarias se realizan a nivel de módulo, las pruebas de integración se realizan en el ámbito de subsistema, donde un subsistema es una jerarquía de módulos. Ya que los módulos han sido probados de forma unitaria antes de las pruebas de integración, pueden ser tratados como cajas negras, permitiendo que las pruebas de integración se concentren en las interfaces entre módulos. Los objetivos de las pruebas de integración son: verificar que cada módulo se desempeña correctamente dentro de la estructura de control y que sus interfaces son correctas.

Las pruebas de integración son una técnica sistemática para ensamblar software mientras que al mismo tiempo se realizan pruebas para descubrir errores asociados con las interfaces.

Las pruebas de integración se realizan combinando módulos por pasos. En cada paso se agrega un módulo a la estructura del programa y las pruebas se concentran en ejercitar este nuevo módulo. Algunas veces se incluyen pruebas de regresión como un segundo componente de las pruebas de integración. En las pruebas de regresión se ejercitan los otros módulos en el programa para determinar si han sido afectados negativamente por el nuevo módulo.

Cuando se ha demostrado que un módulo funciona apropiadamente dentro de la estructura de un programa, se agrega otro módulo y las pruebas continúan. Este proceso se repite hasta que todos los módulos se han integrado y probado.

Existen dos enfoques principales para las pruebas de integración: Integración ascendente e Integración descendente.

INTEGRACION ASCENDENTE

En la integración ascendente, los módulos de más bajo nivel se integran y prueban primero. Esto significa que las pruebas de integración proceden hacia arriba en la estructura jerárquica de control del programa como se muestra en la figura que se presenta a continuación:

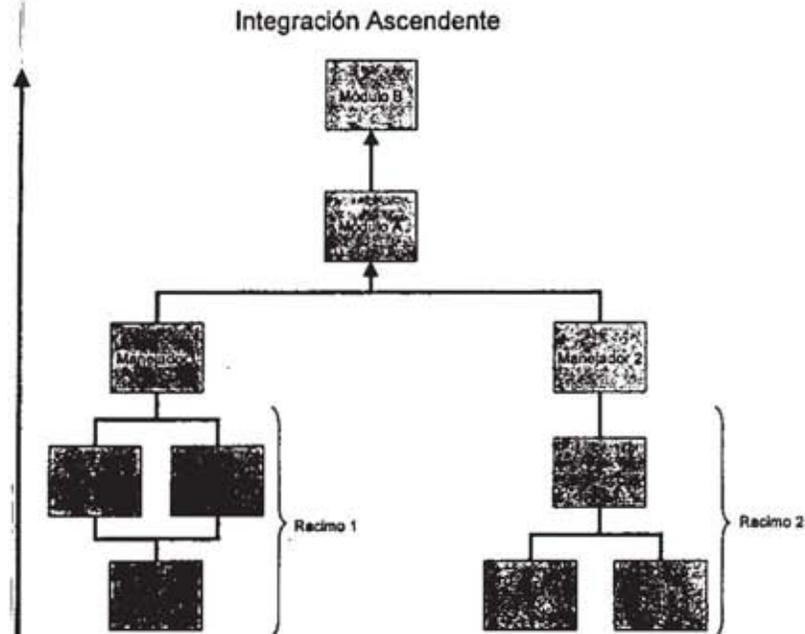


Figura 5.1 Integración ascendente de pruebas

La estrategia de integración ascendente debe implementarse con los siguientes pasos:

1. Los módulos de bajo nivel se combinan en racimos que realizan una subfunción específica.
2. Un manejador (un programa de control para pruebas) se escribe para coordinar las entradas y salidas del caso de prueba.
3. Se prueba el racimo.
4. Se eliminan los manejadores y los racimos se combinan, moviéndose hacia arriba en la estructura del software.

Problemas con la integración ascendente.

Existen varios problemas con la integración de este tipo:

- Los programas manejadores pueden requerir un gran esfuerzo para desarrollarlos y pueden ser una fuente adicional de errores.
- Las funciones de programa de alto nivel se prueban al final y menos. Los errores críticos frecuentemente ocurren en funciones de alto nivel y la integración abajo-arriba puede tardar en encontrarlos.
- Los errores de interfaces pueden no aparecer hasta tarde en la prueba de integración. Entre más tarde se descubran los errores más difícil y costoso será corregirlos.

INTEGRACION DESCENDENTE

En la integración descendente, los módulos de alto nivel se integran y prueban primero. Esto significa que las pruebas de integración bajan por la estructura de control jerárquica que se muestra en la siguiente figura.

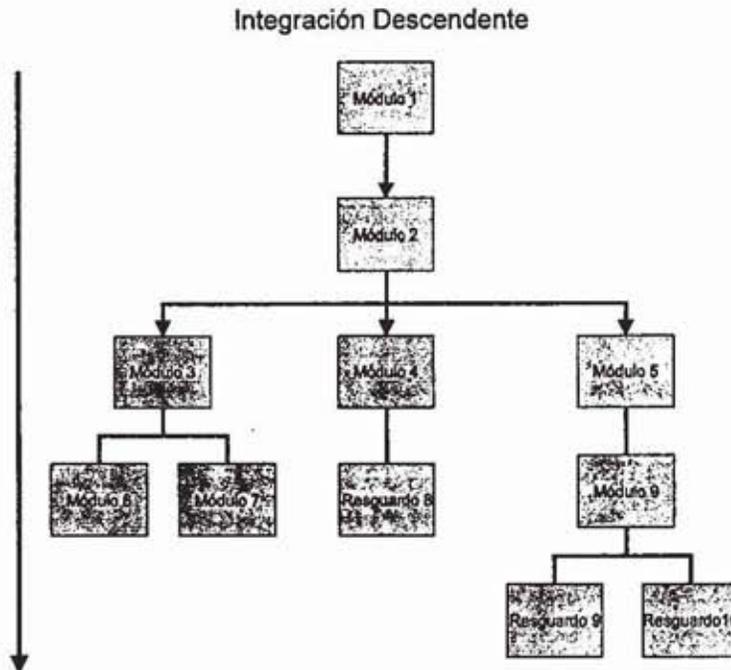


Figura 5.2 Integración descendente de pruebas

La integración descendente requiere la creación de módulos de prueba llamados resguardos. Cuando se prueba un módulo, los módulos que este llama se representan por los resguardos, los cuales usualmente solo regresan el control al módulo que los llamó. Al continuar la prueba hacia abajo en la estructura del programa, cada resguardo se reemplaza por el código del módulo real.

La integración descendente ofrece varias ventajas importantes sobre la integración ascendente:

- La cantidad de datos de prueba necesarios crece al ir avanzando las pruebas de integración, permitiendo que se puedan compartir los datos de prueba en lugar de requerir datos de prueba separados para cada módulo al irse integrando.
- Las funciones de alto nivel se prueban primero y más.
- Las pruebas de integración pueden comenzar temprano ya que los módulos de alto nivel pueden codificarse primero y pueden probarse con resguardos.
- Los resguardos son más fáciles de construir que los manejadores.

Problemas con la integración descendente.

Aunque usualmente se prefiere sobre la integración puramente ascendente, puede haber problemas con la integración descendente:

- Algunas veces los resguardos pueden ser difíciles de construir como el módulo real. Esto puede consumir tiempo y ser propenso a errores.
- Las funciones críticas de bajo nivel no se prueban sino hasta tarde en el proceso de desarrollo. Problemas con las funciones críticas de bajo nivel pueden forzar un rediseño tardío de varias funciones de alto nivel.

COMBINACION DE INTEGRACION DESCENDENTE / ASCENDENTE

En pequeños programas, no hace mucha diferencia que tipo de integración se utilice. En este caso, las pruebas unitarias y de integración se juntan en un solo paso. Pero en proyectos largos, se prefiere más formalidad en la separación de los pasos de prueba. Aunque se prefiere una combinación de ambos tipos de integración, que capitaliza las ventajas y minimiza los problemas. Esta combinación es básicamente el tipo descendente que incorpora un enfoque ascendente desviándose de una dirección puramente descendente para considerar primero a las más críticas y/o complejas partes. Las siguientes son algunas consideraciones para las pruebas de integración:

1. Utilizar un enfoque descendente para probar la estructura de control del programa y verificar primero la exactitud de las funciones de alto nivel.
2. Usar el enfoque ascendente para probar los módulos que tienen interfaces con el ambiente del sistema operativo en complejas o nuevas formas y probar módulos que realicen la lógica más crítica o compleja en el programa.
3. Incluir pruebas de regresión como un componente de las pruebas de integración.

5.2.3 PRUEBAS DE VALIDACION

Al término de las pruebas de integración, el software está completamente ensamblado como un paquete; los errores de interfaces se han descubierto y corregido y debe iniciarse una serie de pruebas finales al software: las pruebas de validación. La validación se puede definir de muchas maneras, pero una definición simple es que la validación tiene éxito cuando el software funciona de la manera como lo esperaba el usuario.

El objetivo de la validación es descubrir cualquier implementación incorrecta o que no cumpla con la especificación de requerimientos.

Las estrategias típicas para las pruebas de validación incluyen:

- Los casos de prueba deben ejecutar las partes más importantes y más usadas del programa.
- Los casos de prueba deben representar uso normal o esperado del programa.
- Los casos de prueba deben seleccionarse para exponer los errores bajo condiciones extremas o críticas de procesamiento.

5.2.4 PRUEBAS DE SISTEMA

El software es parte de un sistema de cómputo. Al final del proceso el software se incorpora con otros elementos del sistema (como nuevo hardware e información) y se realizan una serie de pruebas de integración del sistema y validación. Estas pruebas ya no son responsabilidad del equipo que hizo el desarrollo. Si embargo los pasos realizados durante el diseño del software y las pruebas pueden mejorar las probabilidades de una exitosa integración del software en un sistema grande.

CAPITULO 5. PRUEBAS A LA SOLUCION

Un problema clásico de pruebas del sistema es "apuntar con el dedo." Esto ocurre cuando se descubre un error y cada elemento del equipo de desarrollo culpa al otro del problema. En lugar de seguir con esta discusión inútil el ingeniero de software debe anticipar todos los problemas potenciales de interfaces y

1. Diseñar formas para manejo de errores que verifiquen toda la información que venga de otros elementos del sistema.
2. Conducir una serie de pruebas que simulen "datos malos" u otros errores potenciales en la interfaz del software.
3. Registrar los resultados de las pruebas para usarlos como "evidencia" en caso de que ocurra el problema de "apuntar con el dedo"
4. Participar en la planeación y diseño de las pruebas del sistema para asegurar que el software se prueba adecuadamente.

El paso final en las pruebas de sistema comúnmente es llamado la prueba de aceptación. La prueba de aceptación que es conducida por el usuario final en lugar que el desarrollador del sistema, puede ir desde una prueba informal, hasta una serie planeada y sistemáticamente ejecutada de pruebas. De hecho, las pruebas de aceptación pueden llevarse a cabo sobre un periodo de semanas o meses, por lo que los errores acumulados que pueden degradar el sistema en el tiempo sean descubiertos.

DISEÑO DE CASOS DE PRUEBA

El objetivo principal del diseño de casos de prueba es definir una combinación de datos de prueba que tengan la más alta probabilidad de descubrir un error o una clase de errores. Es materialmente imposible probar todas las opciones y combinaciones de un programa. Aún las pruebas para un pequeño subconjunto de las combinaciones de un programa, pueden resultar en una cantidad enorme de datos de prueba. Entonces el diseñador de casos de prueba debe aplicar técnicas para descubrir el mayor número posible de errores con el mínimo número razonable de pruebas.

En las siguientes secciones se presentan alguna técnicas de diseño de casos de prueba.

Cobertura Lógica

El término cobertura lógica se usa para definir a un conjunto de procedimientos de prueba que resultan en grados progresivamente más completos de pruebas de rutas de programas. Se pueden proponer los siguientes grados de cobertura:

1. Cada instrucción es ejecutada al menos una vez.
2. Cada decisión es evaluada para conocer las consecuencias de los resultados verdaderos o falsos.
3. Se prueban rutas más complejas mediante la permutación de diferentes condiciones.

El diseño de los casos de prueba depende de una clara definición de todas las variables de decisiones y combinaciones de datos que satisfagan cada grado de cobertura lógica. Los casos de prueba iniciales deben asegurar que todas las instrucciones han sido ejecutadas. Para lograr altos niveles de cobertura lógica se deben crear tablas de decisión de variables y sus relaciones con los datos de entrada. Los casos de prueba se diseñan para ejecutar cada alternativa de decisión individualmente o para probar rutas específicas que resulten de una combinación de decisiones.

Particionamiento de equivalencia

Un caso de prueba ideal descubre una clase de errores (por ejemplo errores con una estructura de datos local) que de otra forma podrían requerir la ejecución de muchos casos

antes de que se pueda observar el error general. El particionamiento de equivalencia se esfuerza por definir un caso de prueba que descubra clases de errores, por lo tanto reduce el número total de casos de prueba que deben desarrollarse.

El diseño de casos de prueba para el particionamiento de equivalencia se basa en la evaluación de clases de equivalencia para condiciones de entrada. Por ejemplo, una clase de equivalencia válida para un registro de un número telefónico pudiera ser como sigue:

Código de área – Un blanco o un número de tres dígitos

Prefijo – Número de tres dígitos que no comiencen con 0 o 1.

Sufijo – Número de cuatro dígitos.

Extensión – Número de hasta cuatro dígitos.

Una clase de equivalencia inválida para el registro de arriba pudiera ser un código de área de dos dígitos o un código de área < 100 (una prueba especial pudiera definirse para códigos de área 800 y 900).

Se deben seleccionar casos de prueba de tal forma que el mayor número de atributos de una clase de equivalencia se ejercite una vez. El particionamiento de equivalencia es una técnica de caja negra. Solo se utiliza información de entrada para desarrollar un caso de prueba.

Análisis de valores límite

El análisis de los valores límite lleva a una selección de casos de prueba que ejercitan valores límite. Muchos errores de software ocurren justo por debajo, en o por encima de los valores límite de los índices, estructuras de datos y valores escalares. Los casos de prueba que ejercitan este dominio tienen una alta probabilidad de descubrir errores.

5.3 METODOLOGIA APLICADA EN BANCRECER

Tradicionalmente se piensa que las pruebas a los sistemas se deben realizar al final de la etapa de desarrollo, pero una de las principales desventajas de este enfoque es que la gente que realizará las pruebas se integra hasta un punto muy avanzado del proyecto, con lo cual se tiene poco tiempo para entender lo que debe hacer el sistema, por lo tanto no hay un conocimiento a detalle de lo que realiza el sistema y por consiguiente de cómo probarlo.

La metodología utilizada en el área de Sistemas de BanCrecer para realizar las pruebas a los sistemas es una adaptación del Modelo-V de Andersen Consulting. A continuación se presenta un resumen de esta metodología.

Modelo-V

El Modelo-V, como su nombre lo indica, representa la forma de realizar las pruebas de un sistema siguiendo la trayectoria de una letra V: descendiendo de la izquierda y ascendiendo hacia la derecha. En este modelo se encuentran en el nivel más alto, a ambos extremos de la V, dos etapas, una que representa la definición del problema a nivel más general y otra que representa la prueba que certifica que el problema se ha resuelto y que esto redundará en beneficios para la empresa. Los niveles inferiores representan etapas más específicas como se muestra en la figura que se presenta a continuación:

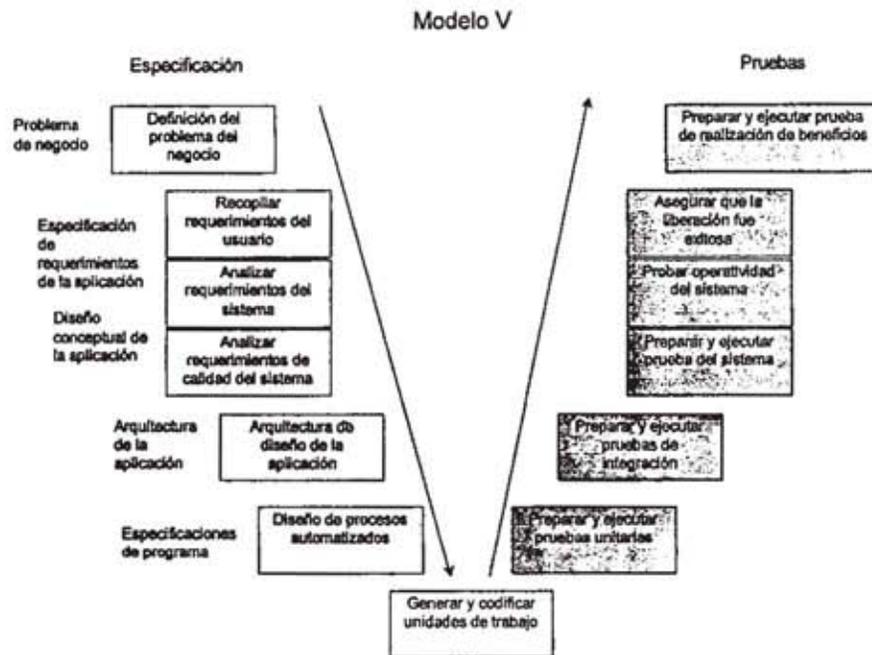


Figura 5.3 Modelo-V de pruebas

A medida que se desciende en la rama izquierda del Modelo-V se representan etapas más específicas en el desarrollo de la aplicación, al ascender por la rama derecha se presentan etapas más generales.

Dentro de cada etapa se elaboran documentos específicos conteniendo los detalles de lo realizado y que sirven como puntos de partida para las siguientes etapas. A estos documentos se les conoce como entregables, por que son el resultado concreto del trabajo realizado en cada etapa y que serán proporcionados a las demás áreas para apoyar su trabajo en las etapas que les corresponden.

Las acciones que se toman de la parte de Pruebas del modelo hacia la parte de Especificación son las siguientes:

- Pruebas. Se valida que la especificación se implementa y conjunta apropiadamente
- Verificación. Es parte de los criterios de salida de una etapa, para asegurar que los procedimientos y estándares se han considerado.
- Validación. Es parte de la revisión del proyecto para asegurar que los entregables entre etapas continúan satisfaciendo el problema del negocio.
- Criterios de entrada y salida. Se utilizan para mejorar la calidad de los entregables. Los criterios son estándares predefinidos que los entregables deben cumplir antes de salir de una etapa y entrar a otra.

En la figura que se presenta a continuación se ejemplifica cada una de las acciones sobre el Modelo-V:

Modelo V

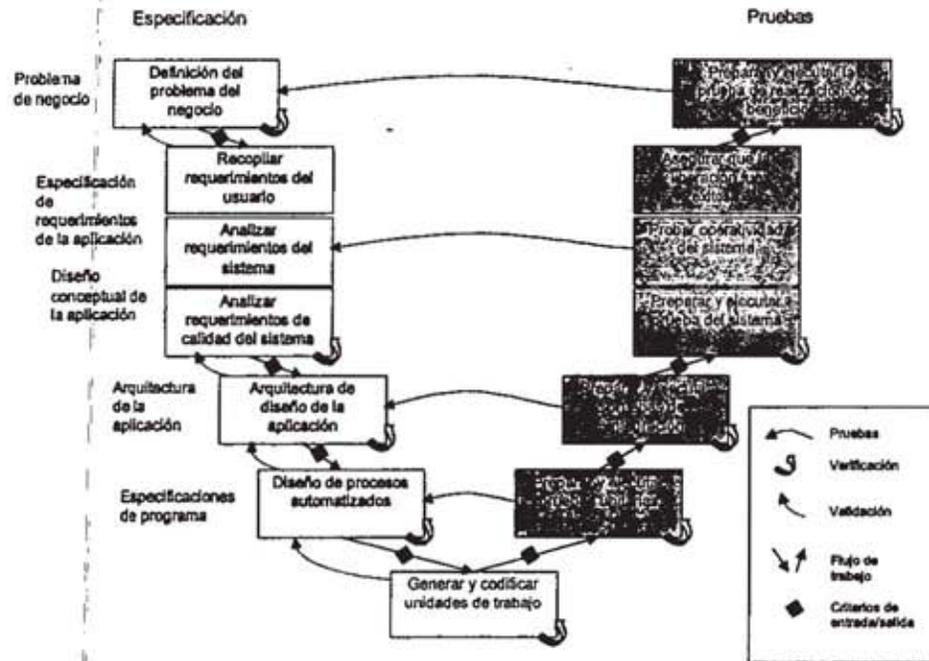


Figura 5.4 Modelo-V de pruebas completo

Una de las ventajas principales de este modelo es que se pueden encontrar y arreglar los problemas en la etapa en que se originaron.

En el área de Sistemas de BanCreceer el esquema que se aplica para las pruebas es el que se muestra en la siguiente figura:

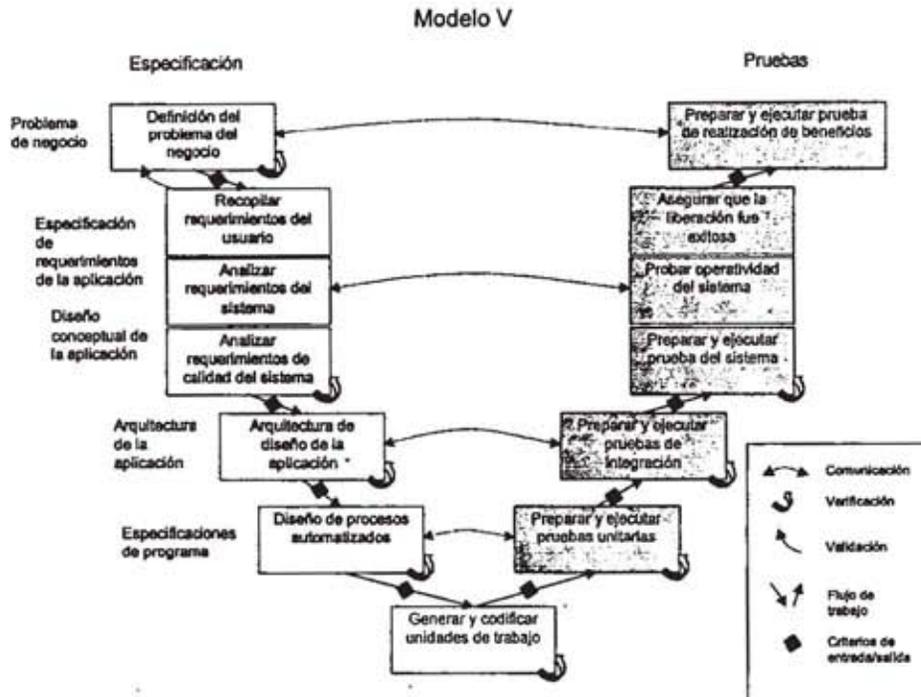


Figura 5.5 Modelo-V de pruebas de BanCreceer

Las actividades que se realizan entre cada etapa del Modelo-V de pruebas de BanCreceer son las siguientes:

- **Comunicación.** El área de Implantación que realizará las pruebas comienza a conocer desde la definición del problema hasta el diseño de los procesos automatizados, pero sin realizar pruebas a estas etapas. El objetivo de la comunicación es que al iniciar las etapas de pruebas exista un conocimiento bien cimentado del sistema por parte de Implantación.
- **Verificación.** Las áreas de Diseño, Desarrollo e Implantación son responsables de que cada documento que salga de una etapa a otra cumpla con los estándares y los procedimientos establecidos.
- **Validación.** La validación se realiza solamente entre la etapa de recopilación de requerimientos del usuario y la definición del problema del negocio, debido a que el área de Diseño se encarga por su parte de que cada etapa cumpla con la definición del problema.
- **Criterios de entrada / salida.** Cada área que recibe un documento revisa que lo reciba de otra área de acuerdo a los estándares.

5.4 REALIZACION DE LAS PRUEBAS

Para la realización de las pruebas se utilizó la metodología del Modelo-V, en especial se elaboró el documento "Propuesta de Prueba de Producto" que se presenta a continuación.

Propuesta de Prueba de Producto

[1] Producto: Centro de Información de Cajeros Automáticos (CICA).			
[2] Liberación: 1.0			
[3] Plataforma: Windows 95/Unix/Informix			
[4] Configuración: De acuerdo al plan de pruebas de Implantación			
[5] Preparado por: LVR/RRC	Fecha: 1999/06/01	[7] Versión: 1.0	Página: 1
[6] Aprobado por: MMC/RVA	Fecha: 1999/06/15	[8] Estatus: Aprobado	
[9] Introducción			
<p>El propósito de este documento es describir la propuesta de la prueba de producto para el Centro de Información de Cajeros Automáticos (CICA). La propuesta de prueba de producto contiene las siguientes secciones:</p> <ul style="list-style-type: none"> • Objetivos y alcance de las pruebas. • Riesgos. • Enfoque de pruebas de regresión. • Requerimientos del ambiente de pruebas. • Mediciones • Criterios de entrada / salida • Recursos y plan de trabajo para las pruebas. 			
[10] Objetivos y alcance de las pruebas			
<p>El objetivo de las pruebas de producto es asegurar que el CICA cumple con los requerimientos definidos en el capítulo 2, en la sección "Requerimientos del Usuario", estas pruebas se enfocarán a dos grandes rubros:</p> <ul style="list-style-type: none"> • Pruebas de funcionalidad del producto MetaCube Explorer. • Pruebas de consistencia de los datos reportados por MetaCube Explorer. <p>Las pruebas del producto aseguran que el nuevo sistema funcionará aceptablemente en todas sus facetas, las pruebas de consistencia de datos garantizarán su veracidad y confiabilidad con lo que se mejorará la capacidad de los usuarios para la toma de decisiones en cuanto a la ubicación de los cajeros o el reforzamiento de su operación.</p> <p>Se utilizará el enfoque de tres niveles y tres pasos para las pruebas de este producto.</p> <p>Nivel 1. Es una prueba horizontal de alto nivel de todas las áreas funcionales de los procesos de Extracción, Transformación y Carga descritos en el Capítulo 3. Su propósito es evaluar el flujo de datos por todo el sistema. Esta evaluación asegura que el ambiente y el código son estables antes del inicio del resto de las pruebas.</p> <p>Nivel 2. Es una serie de pruebas verticales detalladas de las áreas funcionales. Estas son pruebas independientes que permiten la evaluación concurrente y por lo tanto reducen el tiempo total de pruebas. Las pruebas de nivel 2 deben iniciarse una vez que las pruebas de nivel 1 han determinado que el ambiente y el código son estables.</p> <p>Nivel 3. Es una serie de pruebas de procesamiento de excepciones, esto es evaluar la habilidad del sistema para manejar caídas del servidor, entre otras excepciones.</p> <p>Cada ciclo dentro de cada nivel de pruebas se ejecutará tres veces (o sea tres pasos).</p>			

El objetivo del paso 1 es recorrer la prueba tan rápido como sea posible, encontrando tantos defectos como sea posible e implementando arreglos donde se necesiten.

El objetivo del paso 2 es hacer pruebas de regresión a los defectos corregidos en el paso 1 y determinar si los arreglos del paso 1 causaron aún más defectos.

El objetivo del paso 3 es hacer pruebas de regresión a los defectos corregidos en el paso 2; este paso debe correr sin problemas.

La técnica de tres niveles y tres pasos es una estrategia estructurada la cual:

Hace más fácil predecir cuando estarán completas las pruebas. Esta estrategia se enfoca en encontrar tantos problemas como sea posible lo más temprano posible, por lo que habrá tiempo para arreglarlos y probarlos nuevamente. Pronto dentro del proceso, se obtendrá un sentido de qué esperar por todo lo que resta de las pruebas; esto en su momento permite predecir cuando las pruebas estarán completas.

Reduce el tiempo que debe transcurrir para las pruebas. Esta estrategia recomienda una mezcla de ciclos independientes y dependientes. Un pequeño número de ciclos de prueba deben ser dependientes para asegurar que los datos pueden pasar a través de todo el sistema correctamente. El remanente de los ciclos de prueba deben ser independientes de tal forma que un solo defecto crítico no detenga toda la prueba del producto. Esto reducirá el tiempo total de pruebas.

Asegura pruebas completas y de calidad. El componente de tres pasos de esta estrategia se basa en que las pruebas de regresión aseguran que los defectos están completamente corregidos y que las correcciones no han descompuesto nada más.

En las pruebas de este producto se aplicará la estrategia de tres niveles y tres pasos como se indica a continuación:

Nivel 1: Horizontal de Alto Nivel

Ciclo 1 – Procesos en Unix e Informix

Ciclo 2 – Presentación de datos en MetaCube

Nivel 2: Vertical Detallada

Ciclo 1 – Procesos en Unix e Informix

Subciclo A – Carga de datos

Subciclo B – Análisis de impacto

Subciclo C – Estimación de cargas de trabajo

Subciclo D – Implementación de correcciones

Subciclo E – Migración de actualizaciones

Subciclo F – Mantenimiento de versiones actualizadas

Ciclo 2 – Presentación de datos en MetaCube

Subciclo A – Seguridad en la carga de datos y en el acceso a MetaCube

Subciclo B – Soporte de mediciones

Subciclo C – Soporte de Reportes/Vistas/Consultas

Subciclo D – Soporte de seguimiento de auditorías

Subciclo E – Mantenimiento a parámetros del sistema

Nivel 3: Procesamiento de Excepciones

Ciclo 1 – Fallas del servidor

Ciclo 2 – Pruebas de stress

Ciclo 3 – Pruebas de desempeño

Idealmente cada nivel de pruebas ocurriría en una región lógica separada, para que los niveles pudieran traslaparse y que el tiempo de las pruebas fuera menor; pero en el caso de los recursos asignados al CICA solo se tiene un servidor de pruebas con una sola región lógica.

Las condiciones de prueba para los requerimientos de calidad –incluyendo la flexibilidad, desempeño, confiabilidad y disponibilidad del sistema- se definirán durante el desarrollo de la especificación de requerimientos de la aplicación.

[11] Riesgos

Riesgo del plan. El tiempo disponible para la ejecución de las pruebas de producto del CICA es muy corto. Esta pequeña ventana de tiempo no permite mucho espacio para retrasos en la ejecución. Cualquier retraso en la construcción de programas moverá a las tareas subsecuentes, incluyendo las pruebas del producto.

Plan de Mitigación. La preparación de las pruebas ocurrirá durante la construcción y se completará hacia el final de las pruebas de integración para asegurar que la ejecución de las pruebas de producto pueden iniciar inmediatamente después de que se haya realizado la ejecución de las pruebas de integración.

Plan de contingencias. Si el tiempo estimado para las pruebas no es suficiente para terminar sin retrasos, se debe evaluar el tiempo extra, recursos extras o intercambio de recursos.

[12] Regresión en las pruebas

La estrategia de tres pasos para las pruebas de producto facilitarán que se pueda regresar a probar los defectos encontrados en la prueba. Además todo el modelo de pruebas del producto estará documentado y será repetible en cualquier paso que se requiera.

Por cada código corregido se realizará nuevamente una prueba completa de componente. Si la corrección afecta una prueba de integración, esta prueba se realizará nuevamente. Cualesquiera nuevas condiciones creadas como resultado de implementar un arreglo serán actualizadas al conjunto existente de condiciones de prueba.

[13] Requerimientos para el ambiente de pruebas

Configuración Técnica. El ambiente de pruebas del producto se asemejará al ambiente de producción lo más cercanamente posible; todo el hardware y software empacado usado en la creación del ambiente de pruebas del producto se copiará del ambiente de producción. Esto incluye la configuración de hardware y software de las estaciones de trabajo, definiciones de LAN y definiciones de base de datos.

El ambiente de pruebas estará lógicamente separado del ambiente de desarrollo. Si los recursos están disponibles la separación puede ser física, al contar con un servidor de desarrollo y otro para las pruebas.

En el área de Sistemas de BanCrecer se tiene un esquema de ambientes para el desarrollo, para las pruebas y para la puesta en producción de los sistemas. Para los sistemas cliente/servidor normalmente las áreas de Desarrollo cuentan con un servidor para crear las bases de datos y los programas necesarios para entregar una primera versión de un sistema al área de Implantación, que a su vez cuenta con un servidor de pruebas que contiene una copia exacta de la versión de los programas y las bases de datos que se encuentra en Desarrollo. La fase de pruebas e integración del sistema la realiza el área de Implantación mediante el esquema de pruebas unitarias y pruebas de integración.

En la figura 5.6 se muestra una representación de los ambientes de Desarrollo y Pruebas.

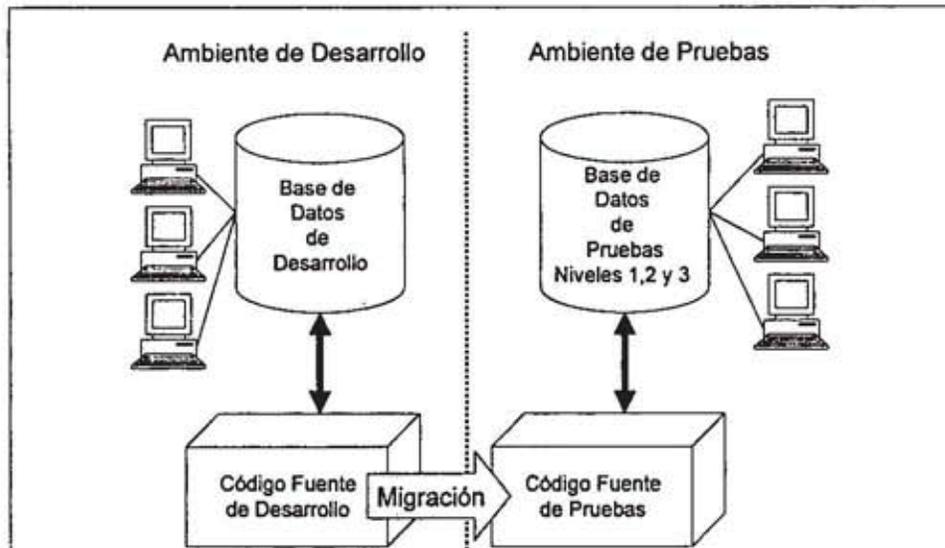


Figura 5.6 Ambientes de Desarrollo y de Pruebas

Interfaces Externas. La fase de prueba del sistema evaluará la internase entre el CICA y el host IBM, ya que de este último se reciben los archivos diarios del TLF para la carga de las tablas.

Administración de los datos de prueba. Habrá dos conjuntos de datos de prueba, uno para validar las cargas de IBM al servidor Unix e Informix, y otro para validar que los datos presentados por MetaCube sean correctos para un mes de información de los cajeros. Para el primer caso se evaluará la carga de día 1 y 15 del mes, considerando que los días anteriores, de un mes atrás, ya están cargados. Para el segundo caso se evaluarán los datos de un mes cargados y procesados en el modelo dimensional del CICA. Los dos conjuntos de datos representan 30,000 registros por día aproximadamente. Después de realizar las pruebas exitosamente se realizará un respaldo de los datos por parte de quien ejecute las pruebas. El respaldo dará evidencia de la ejecución exitosa de las pruebas.

Ambiente de código fuente. Se realizará un inventario de los programas requeridos para las pruebas del sistema para compararlos con los de producción una vez que este cambio se realice, para asegurar que todos los programas se encuentran y no ninguno ajeno a como debe ser el sistema.

Modificación de los parámetros del sistema. Habrá un analista experimentado responsable de hacer los cambios necesarios a las tablas del sistema y a los datos del sistema. Se documentarán todos los cambios necesarios.

Seguridad. Cada persona del área de Implantación tendrá acceso a un usuario genérico (PRUEBAS01, PRUEBAS02, etc.) que será establecido por el área de soporte técnico, para que las pruebas no se detengan si se enferma alguna de las personas, sale de vacaciones, etc.

Si los resultados de las pruebas son satisfactorios el área de Implantación otorga su visto bueno para que la versión del sistema pase a Producción. El pase a Producción implica armar un paquete que incluye la versión de la base de datos y de los programas que se probaron para que sean copiados al servidor de producción en donde antes se realiza una copia de respaldo de la versión a reemplazar.

En la figura 5.7 se muestra un diagrama simplificado del proceso de desarrollo-pruebas-producción.

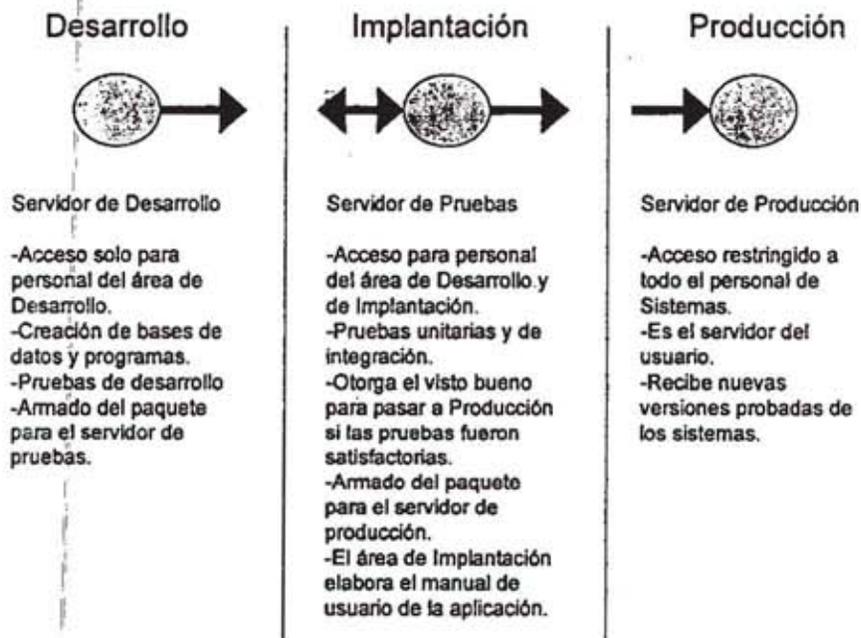


Figura 5.7 Diagrama Desarrollo-Implantación-Producción

Las pruebas de unidad tienen como objetivo descubrir errores en los módulos individuales del sistema. Estos módulos se prueban aislados unos de otros en un ambiente artificial formado por los programas conductores y los datos necesarios para ejecutar los módulos. Las pruebas de unidad deben ser tan exhaustivas como sea posible para garantizar que sea probado cada caso representativo empleado en cada módulo. Dichas pruebas son más fáciles, si las estructuras se componen de módulos pequeños y débilmente acoplados. En el caso de las pruebas a los subsistemas, su propósito principal es verificar la operación de las interfaces entre los módulos en el subsistema. Se deben probar tanto las interfaces de control como las de datos.

Las pruebas del sistema se relacionan con las interfaces, la lógica de decisión, el flujo de control, los procedimientos de recuperación, la eficiencia global, la capacidad y las características del tiempo del sistema integrado en su totalidad.

El objetivo a lograr, siguiendo esta secuencia de pruebas, es dar seguimiento, orden y por tanto estabilidad al proceso de desarrollo del software, permitiendo obtener un producto de

CAPITULO 5. PRUEBAS A LA SOLUCION

mejor calidad. Así mismo, para lograr lo anterior, nos auxiliamos de las denominadas pruebas de demostración que tienen como regla los siguientes tres puntos:

1. Objetivo práctico del sistema.
2. Metodología empleada en el sistema.
3. Comprobación de la metodología.

Las pruebas ascendentes suelen revelar que es más fácil crear casos de prueba y observar las entradas de esta, sin embargo, tiene el inconveniente de que si existen errores de diseño en los módulos de alto nivel se detectan hasta una etapa avanzada de la prueba del sistema.

5.5 RESUMEN

Se realizaron pruebas junto con personal del área de Implantación de acuerdo al documento "Propuesta de Prueba del Producto", presentado en la sección anterior, en dos grandes rubros:

- Pruebas de funcionalidad del producto MetaCube Explorer.
- Pruebas de consistencia de los datos reportados por MetaCube Explorer.

Para las pruebas de funcionalidad se realizaron las siguientes actividades:

5.5.1 INTEGRACION DE LOS MODULOS Y PRUEBAS FINALES

Una vez probados los módulos de Extracción, Transformación y Carga, se anexaron uno a uno en el sistema final. Cada vez que un nuevo módulo se incorporaba se ejecutaba la aplicación para verificar su correcto funcionamiento conviviendo con los otros módulos. Si se detectaba una falla en el funcionamiento del sistema se depuraba el código y solo cuando este funcionaba sin problemas se procedía a agregar otro nuevo.

Al conjuntar los módulos se obtuvo la integración de todos los elementos del sistema. Para cumplir el objetivo de que el sistema fuera amigable para el usuario se le explicó a las personas que lo iban a utilizar la forma en que MetaCube presenta los datos y la manera de navegar en la aplicación, para esto se tuvo como apoyo al personal del área de Implantación para capacitar a los usuarios, utilizando el Manual de Usuario que se presenta en el Apéndice E.

Para las pruebas de consistencia de los datos reportados se realizaron las siguientes actividades:

5.5.2 CERTIFICACION DE LOS DATOS

Una vez que contamos con la aplicación final se implementaron pruebas para verificar que los datos reportados a nivel banco, región, plaza y cajero coincidieran con los datos reportados directamente del sistema Tandem, con las cifras validadas por el área de Implantación se dio por concluido el proceso de integración ascendente. Para ello se realizaron tablas comparativas como la que se muestra a continuación:

CAPITULO 5. PRUEBAS A LA SOLUCION

BanCrecer Sistemas							
Pruebas a los datos del CICA Realizadas por el Area: Implantación				Fecha: 1999/04/15		Resultado	
CAJERO	CONSULTAS	CAMBIO DE NIP	ESTADO DE CUENTA	Número	Importe		
S10101	1,344	25	13	4,400	\$22,550,000	Ok	
S20229	313	0	0	1,152	\$10,950,000	Ok	
S30001	1,210	0	0	2,475	\$17,890,400	Ok	
S40005	704	2	5	845	\$5,000,450	Ok	
...							
...							

Con los resultados obtenidos, en cuanto al funcionamiento de los módulos de Extracción, Transformación y Carga y en cuanto a la certificación de los datos, el área de Implantación dió su visto bueno para liberar el sistema CICA.

CAPITULO 6. CONCLUSIONES

El concepto de Data Warehouse tiene mucho que ver con la manera de entender y mejorar los procesos de cualquier área de negocio de la empresa, es una forma muy útil de obtener información a partir de los datos transaccionales, para empresas en donde se tiene una madurez aceptable en cuanto a los sistemas tradicionales utilizados, así como en la mentalidad de los ejecutivos que tomarán decisiones basados en sistemas de este tipo. Si alguna empresa está interesada en considerar la idea del Data Warehouse deberá tener resuelta la operatividad de sus sistemas transaccionales, con una estabilidad garantizada, pero sobre todo deberá existir el entusiasmo y compromiso suficiente de los equipos involucrados (directivos, usuarios, personal de sistemas, auditores y proveedores) para lidiar con una forma de trabajo diferente.

En el caso del CICA la experiencia de haber implementado el sistema en BanCrecer bajo una arquitectura cliente/servidor proporcionó las bases, al personal de sistemas participante, para el posterior desarrollo de sistemas de Data Warehouse. La tendencia natural en estas fechas será implementar nuevas versiones del sistema utilizando Internet, con todo lo que ello implica: desde capacitación en Java (o algún otro lenguaje) y las herramientas integradoras, hasta la revisión de los esquemas de seguridad en las redes de BanCrecer.

En lo particular ha sido para mí una experiencia muy gratificante trabajar en el desarrollo del CICA, ya que tuve la oportunidad de revisar y aplicar conceptos especialmente interesantes, como la metodología de análisis y diseño de sistemas, la utilización de bases de datos y la documentación que expone ideas y aclara conceptos, entre otros. También lo que he aplicado al elaborar este trabajo han sido la disciplina, la constancia y mucho entusiasmo, ya que concluyo un tema pendiente en el ámbito académico y profesional, pero más que nada concluyo un tema pendiente de mi vida.

APENDICE A. DESCRIPCION DE LOS CAMPOS DEL TLF

APENDICE A. DESCRIPCION DE LOS CAMPOS DEL TLF

Las siguientes páginas describen los campos incluidos en cada registro del Archivo de Registro de Transacciones o Transaction Log File (TLF). Esta información se obtuvo del "Transaction Processing Manual" de ACI.

POSICION	CLAVE Y FORMATO DEL CAMPO	NOMBRE Y DESCRIPCION DEL CAMPO
1-474	TLF	TLF El archivo de registro de transacciones o Transaction Log File (TLF) tiene una longitud de 474 caracteres. Cada registro está dividido en los siguientes grandes bloques: HEAD - Encabezado (posiciones 1-76). AUTH - Transacciones financieras (posiciones 77-474).
1-76	HEAD	Encabezado. El TLF contiene datos de cada transacción financiera (aprobada o rechazada), realizada por cada tarjetahabiente y procesada por Base24 durante un día. El TLF es utilizado por los programas reportadores de Base24 y puede también ser extralido diariamente para proveer datos detallados de las transacciones. Los registros se escriben secuencialmente en el TLF y un nuevo TLF es creado por un proceso iniciador al momento de hacer un corte lógico en la red. Una vez que se ha creado un nuevo TLF, el anterior se cierra y está disponible para ser utilizado en reportes y en procesos de extracción. Los siguientes campos contienen información del encabezado para todos los registros del TLF.
1-8	DAT-TIM Binario 64 S9(18) comp	Fecha y hora de grabación en el TLF. Contiene la fecha y hora en que se grabó el registro dentro del TLF. El formato es: AAMMDDhhmmsscc, donde AA = Año MM = Mes DD = Día hh = Hora mm = Minutos ss = Segundos cc = Centésimas de segundo
9-10	REC-TYP X(2)	Tipo de registro. Identifica el tipo de transacción. Los posibles valores son: 00 = Registro con la posición del Extract. 01 = Transacción financiera (operaciones realizadas por clientes). 02 = Registro con transacción administrativa. 04 = Registro con transacción administrativa. 20 = Registro con datos inválidos pero la transacción fue enviada (rechazada). 21 = Registro con datos inválidos y la transacción no fue enviada. 22 = Para uso futuro.

APENDICE A. DESCRIPCION DE LOS CAMPOS DEL TLF

POSICION	CLAVE Y FORMATO DEL CAMPO	NOMBRE Y DESCRIPCION DEL CAMPO
11-14	AUTH-PPD X(4)	Nombre del proceso de autorización. Nombre del proceso de autorización que introdujo el registro en el TLF.
15-38	TERM	Los valores en los siguientes campos identifican al cajero de donde se generó la transacción.
15-18	LN X(4)	Red lógica del cajero. Clave de la red lógica donde se ubica el cajero. Actualmente solo hay una red lógica bancaria: PRO1.
19-22	FIID X(4)	Banco propietario del cajero. Clave del banco propietario del cajero en donde se realizó la transacción. Todas las claves comienzan con una "B" y enseguida contienen tres dígitos, por ejemplo: B161 es BanCrecer y BCCN es BanCrecen.
23-38	TERM-ID X(16)	Identificación del cajero. Clave que identifica al cajero en donde se realizó la transacción.
39-68	CRD	Los valores en los siguientes campos identifican al emisor de la tarjeta y al tarjetahabiente asociados con la transacción.
39-42	LN X(4)	Red lógica asociada al emisor de la tarjeta. Red lógica en donde se ubica el cliente por su tarjeta. Actualmente es la misma red que la del cajero.
43-46	FIID X(4)	Banco emisor de la tarjeta. Clave del banco que emitió la tarjeta. La estructura es igual que la del banco propietario del cajero.
47-65	PAN	Número de tarjeta. Este campo contiene el número de cuenta primario o Primary Account Number (PAN) de la tarjeta.
47-65	NUM X(19)	Número de cuenta de acceso. Cuenta de acceso del cliente al sistema de cajeros automáticos.
66-68	MBR-NUM 9(3)	Número de miembro o familiar. Cuando en este campo aparezca un número mayor a ceros indicará cuál número de familiar hizo la transacción. El valor por defecto es = 000 (El titular realizó la transacción).
69-72	BRCH-ID X(4)	Sucursal del cajero. Sucursal asociada al cajero en donde se realizó la transacción. A este dato no se le da mantenimiento en el TLF por lo que no es utilizable para el Centro de Información.
73-76	REGN-ID X(4)	Región del cajero. Región en la que se ubica el cajero en donde se realizó la transacción. A este dato no se le da mantenimiento en el TLF por lo que no es utilizable para el Centro de Información.
77-474	AUTH	Los siguientes campos están incluidos en los registros de transacciones financieras que entran al TLF. Los registros de transacciones financieras están identificados por los valores 01, 20 o 21 dentro del campo REC-TYP (posiciones 9-10) en el encabezado.

APENDICE A. DESCRIPCION DE LOS CAMPOS DEL TLF

POSICION	CLAVE Y FORMATO DEL CAMPO	NOMBRE Y DESCRIPCION DEL CAMPO
77-78	TYP-CDE X(2)	Uso de sobre o de cheque. Indica si en la transacción se utilizó un sobre o un cheque Nota: Actualmente en los cajeros de BanCreceer no está habilitado el uso de sobre o cheques.
79-82	TYP 9(4)	Código de tipo de mensaje. Indica el tipo de mensaje que se recibió en la transacción: <u>Transacciones financieras</u> 0200 = Requerimiento para autorización 0210 = Respuesta de autorización. 0220 = Transacción completa. 0221 = Transacción completa duplicada. 0230 = Reconocimiento de transacción completa. <u>Impresión de Estado de Cuenta</u> 0205 = Requerimiento. 0215 = Autorización. <u>Reversos</u> 0420 = Reverso ISO. 0421 = Repetición de reverso ISO. 0430 = Reconocimiento de reverso ISO. <u>Administración de Red</u> 0800 = Logon, Logoff, Prueba de eco. 0810 = Respuesta. <u>SAF</u> 0220 = Aviso. 0221 = Repetición de aviso. 0230 = Respuesta. <u>Misceláneos</u> 5400 = Ajuste a una transacción completa. 9906 = Selección de múltiples cuentas. 9980 = Se quedó dinero en el dispensador de efectivo.
83-84	RTE-STAT 9(2)	Mensaje de estado. Es el código que determina el mensaje de estado a nivel del sistema. Los posibles valores son: 00 = Todo bien. 01 = Error de proceso. 02 = Falla en el dispositivo de seguridad. 11 = Destino no disponible. 12 = Línea calda.
85	ORIGINATOR X(1)	Origen del mensaje (requerimiento). Identifica donde se originó la transacción. Los posibles valores son: 1 = Dispositivo controlado por Base24. 2 = Controlador de cajeros. 3 = Proceso de autorización. 4 = Interfaz con el host. 5 = Host. 6 = Interfaz de intercambio. 7 = Intercambio.

APENDICE A. DESCRIPCION DE LOS CAMPOS DEL TLF

POSICION	CLAVE Y FORMATO DEL CAMPO	NOMBRE Y DESCRIPCION DEL CAMPO
85	RESPONDER X(1)	Origen de la respuesta al mensaje. Identifica donde se originó el mensaje de respuesta (0210). Los posibles valores son: 0 = Valor por defecto, indica que aún no hay respuesta. 1 = Dispositivo controlado por Base24. 2 = Controlador de cajeros. 3 = Proceso de autorización. 4 = Interfaz con el host. 5 = Host. 6 = Interfaz de intercambio. 7 = Intercambio.
87-94	ENTRY-TIM BINARIO 64 S9(18) comp	Fecha y hora de entrada a Base24. Fecha y hora cuando se notificó a Base24 de la transacción. El formato es AAMMDDhhmmsscc, donde: AA = Año. MM = Mes. DD = Día. hh = Hora. mm = Minutos. ss = Segundos. cc = Centésimas de segundo.
95-102	EXIT-TIM BINARIO 64 S9(18) comp	Fecha y hora de salida. Fecha y hora cuando se envió la transacción al host, en el caso de que la transacción requiera autorización del host. El formato es AAMMDDhhmmsscc, donde: AA = Año. MM = Mes. DD = Día. HH = Hora. mm = Minutos. ss = Segundos. cc = Centésimas de segundo.
103-110	RE-ENTRY-TIM BINARIO 64 S9(18) comp	Fecha y hora de recepción del host. Fecha y hora cuando se recibió la transacción del host, cuando el host autorizó y envió la transacción a Base24. El formato es AAMMDDhhmmsscc, donde: AA = Año. MM = Mes. DD = Día. hh = Hora. mm = Minutos. ss = Segundos. cc = Centésimas de segundo.
111-116	TRAN-DAT X(6)	Fecha de origen de la transacción. Fecha en la que se originó la transacción (en el cajero o en el host). El formato es AAMMDD, donde: AA = Año. MM = Mes. DD = Día.

APENDICE A. DESCRIPCION DE LOS CAMPOS DEL TLF

POSICION	CLAVE Y FORMATO DEL CAMPO	NOMBRE Y DESCRIPCION DEL CAMPO
117-124	TRAN-TIM X(8)	Hora de origen de la transacción. Hora en la que se originó la transacción (en el cajero o en el host). El formato es hhmmsscc, donde: hh = Hora. mm = Minutos. ss = Segundos. cc = Centésimas de segundo.
125-130	POST-DAT X(6)	Fecha de proceso de corte. Fecha en que la transacción entrará a proceso de corte y compensación de la red bancaria. El formato es AAMMDD, donde: AA = Año. MM = Mes. DD = Día.
131-136	ACQ-ICHG-SETL-DAT X(6)	Fecha de corte con intercambio de origen. Fecha en que la transacción entrará a proceso de corte y compensación entre redes bancarias, con la red que originó la transacción y Base24. (Como solo hay una red este campo tiene ceros). El formato es AAMMDD, donde: AA = Año. MM = Mes. DD = Día.
137-142	ISS-ICHG-SETL-DAT X(6)	Fecha de corte con intercambio de quien autorizó. Fecha en que la transacción entrará a proceso de corte y compensación entre redes bancarias, con Base24 y la red que recibió la transacción. El formato es AAMMDD, donde: AA = Año. MM = Mes. DD = Día.
143-154	SEQ-NUM X(12)	Número secuencial asignado a la transacción. Número secuencial asignado a la transacción por el cajero automático, cuando el cajero no lo asigna el encargado de asignarlo es el controlador de cajeros.
155-156	TERM-TYP 9(2)	Tipo de cajero. El tipo de cajero puede ser: 01 = Diebold 910. 02 = IBM 3624 versión 8 - cartucho dual. 03 = Docutel 2300. 04 = Burroughs RT650/750. 06 = NCR 1770/1780. 09 = Diebold 911 dispositivo externo. 10 = Diebold 911 servicio en lobby. 11 = Docutel 5100. 14 = IBM 3624 versión 7 - cartucho sencillo. 15 = IBM 3624 versión 7 - cartucho dual. 16 = IBM 3624 versión 6 - cartucho sencillo. 17 = IBM 3624 versión 6 - cartucho dual. 18 = IBM 3624 versión 8 - cartucho sencillo. 20 = Diebold 912. 22 = NCR 5070/5080.

APENDICE A. DESCRIPCION DE LOS CAMPOS DEL TLF

POSICION	CLAVE Y FORMATO DEL CAMPO	NOMBRE Y DESCRIPCION DEL CAMPO
		26 = IBM 4730 de consola sencilla. 27 = IBM 4730 de consola dual. 28 = Omron ATM. 29 = Omron RCD. 30 = Diebold 1000 32 = Mirror 40 = Diebold 906 45 = IBM 3614 versión 5 - cartucho sencillo. 46 = IBM 3614 versión 5 - cartucho dual. 48 = IBM 3624 versión 8 - cartucho sencillo D01. 49 = IBM 3624 versión 8 - cartucho dual D02.
157-158	TIM-OFST BINARIO 16 S9(4) comp	Diferencia en huso horario. La diferencia en horas (sumando o restando minutos) entre el lugar donde se ubica el cajero y el lugar en donde se ubica el procesador Tandem.
159-169	ACQ-INST-ID-NUM 9(11)	Número de identificación del propietario del cajero. Identifica al banco propietario del cajero automático, utilizado para el ruteo de mensajes entre el host y Base24.
170-180	RCV-INST-ID-NUM 9(11)	Número de identificación del banco emisor. Identifica al banco emisor de la tarjeta, utilizado para el ruteo de mensajes entre el host y Base24.
181-186	TRAN-CDE	Los valores en los siguientes campos definen el código de la transacción.
181-182	T-CDE X(2)	Tipo de transacción. Los siguientes códigos están soportados por Base24, pero no todos los códigos están soportados por todos los cajeros automáticos. Los posibles valores son: 03 = Garantía de cheque. 04 = Verificación de cheque. 10 = Retiro. 11 = Chequeo de efectivo. 20 = Depósito. 24 = Depósito con devolución de cambio. 30 = Consulta. 40 = Transferencia. 50 = Pago con transferencia. 51 = Pago ensobretado. 60 = Mensaje al banco. 61 = Acceso solamente. 62 = Revisión de tarjeta. 70 = Impresión de estado de cuenta. 81 = Cambio de NIP.
183-184	T-FROM X(2)	Tipo de cuenta origen. Indica el tipo de cuenta origen de la operación: 00 = No hay cuenta origen. 01 = Cheques. 11 = Ahorro y Cuenta Maestra. 31 = Tarjeta de crédito.

APENDICE A. DESCRIPCION DE LOS CAMPOS DEL TLF

POSICION	CLAVE Y FORMATO DEL CAMPO	NOMBRE Y DESCRIPCION DEL CAMPO
185-186	T-TO X(2)	Tipo de cuenta destino. Indica el tipo de cuenta destino de la operación: 00 = No hay cuenta destino. 01 = Cheques. 11 = Ahorro y Cuenta Maestra. 31 = Tarjeta de crédito.
187-205	FROM-ACCT X(19)	Número de cuenta origen. Número de la cuenta a partir de la cual se hizo la transacción. Si este número no se necesita o se desconoce contiene ceros.
206	USER-FLD1 X(1)	Este campo no se utiliza.
207-225	ACCT-NUM X(19)	Número de cuenta destino. Número de la cuenta a la cual se está afectando con una operación. Si este número no se necesita o se desconoce contiene ceros.
226	MULT-ACCT 9(1)	Identificador de selección de cuenta. Indica si la transacción es por una cuenta primaria, por cuentas múltiples o por disposición inmediata. Los valores posibles son: 0 = Transacción de cuenta primaria. El cajero donde se originó la transacción no soporta selección de cuentas múltiples. 1 = Transacción de cuentas múltiples. El cajero donde se originó la transacción soporta selección de cuentas múltiples. 2 = Transacción de disposición rápida.
227-234	AMT-1 BINARIO 64 S9(18) comp	Cantidad solicitada. Importe solicitado en la transacción. En el caso de mensajes de respuesta forzada (0220) o reversos (0420) este campo contiene el monto originalmente solicitado. Para mensajes de ajuste (5400) este campo indica el monto original completado o entregado.
235-242	AMT-2 BINARIO 64 S9(18) comp	Cantidad de balance o ajuste. Para la mayoría de los mensajes de reverso (0420) este campo contiene la cantidad que realmente se le proporciona al cliente. Para la mayoría de los mensajes de respuesta (0210) este campo contiene una cantidad de ajuste. Para cuentas de crédito este es el ajuste de crédito. Para la mayoría de los mensajes de petición (0200) este campo contiene ceros. Para mensajes de ajuste (5400) este campo indica el monto completado actualizado.
243-250	AMT-3 BINARIO 64 S9(18) comp	Cantidad disponible o saldo. Este es el saldo disponible para cuentas de débito y es el crédito disponible para cuentas de crédito. Para la mayoría de los mensajes de respuesta (0210) este campo contiene un importe de ajuste. Para un mensaje de reverso (0420) que sea resultado de una transacción de depósito con devolución de efectivo,

APENDICE A. DESCRIPCION DE LOS CAMPOS DEL TLF

POSICION	CLAVE Y FORMATO DEL CAMPO	NOMBRE Y DESCRIPCION DEL CAMPO
		este campo contiene el importe del depósito si esa parte de la transacción no se completó exitosamente y contiene el monto completado de la parte de devolución de la transacción si el depósito se completó, pero la devolución de efectivo no se completó exitosamente. Este es el único tipo de reverso que utiliza este campo.
251-254	DEP-BAL-CR BINARIO 32 S9(9) comp	Crédito en firme por depósito. Para una transacción de depósito segmentado, este campo contiene el importe del crédito otorgado a la primera cuenta involucrada en la transacción.
255	DEP-TYP 9(1)	Tipo de sobre para depósito o pago. Actualmente en los cajeros de BanCrecer no está habilitado el uso de sobre.
256-258	RESP-CDE	Los valores en los siguientes campos definen el código de respuesta asignado por el autorizador de transacciones.
256	RESP-BYTE-1 X(1)	Indicador que determina si la tarjeta fue regresada o retenida. Los valores posibles son: 0 = La tarjeta fue regresada. 1 = La tarjeta fue retenida.
257-258	RESP-BYTE-2 X(2)	Razón de respuesta. Es un código que identifica la razón por la cual una transacción fue aprobada o rechazada. Cuando es aceptada el código es menor que diez. Nota: Ver el Apéndice B para conocer una descripción de los códigos de respuesta más representativos.
259-283	TERM-NAME-LOC X(25)	Dirección del cajero. Calle y número donde se ubica el cajero.
284-305	TERM-OWNER-NAME X(22)	Nombre del banco propietario del cajero. Idem
306-318	TERM-CITY X(13)	Ciudad donde se ubica el cajero. Idem
319-321	TERM-ST-X X(3)	Estado donde se ubica el cajero. Idem
322-323	TERM-CNTRY-X X(2)	País donde se ubica el cajero. Idem
324-351	ORIG	El concepto de datos originales aplica solamente a reversos (0420) y ajustes (5400). Si la transacción es un reverso o un ajuste estos campos contienen información de la transacción original. Los campos que se presentan a continuación componen al ORIG.
324-335	OSEQ-NUM X(12)	Número de la transacción original. Número secuencial de la transacción original.
336-339	OTRAN-DAT X(4)	Fecha de la transacción original. El formato es AAMMDD, donde: AA = Año. MM = Mes. DD = Día.
340-347	OTRAN-TIM	Hora de la transacción original.

APENDICE A. DESCRIPCION DE LOS CAMPOS DEL TLF

POSICION	CLAVE Y FORMATO DEL CAMPO	NOMBRE Y DESCRIPCION DEL CAMPO
	X(8)	El formato es hhmmsscc, donde: hh = Hora. mm = Minutos. ss = Segundos. cc = Centésimas de segundo.
348-351	B24-POST-DAT X(4)	Fecha de proceso de corte de la transacción original. Fecha en que la transacción original entrará al proceso de corte.
352-354	ORIG-CRNCY-CDE 9(3)	Código original de moneda. Clave del tipo de moneda de la transacción original.
355-384	MULT-CRNCY	Los siguientes campos contienen la información referente al tipo de cambio de la moneda del banco emisor de la tarjeta y del banco propietario del cajero.
355-357	AUTH-CRNCY-CDE 9(3)	Código de moneda de autorización. Código que identifica el tipo de moneda utilizado en la respuesta de autorización.
358-365	AUTH-CONV-RATE 9(8)	Factor de conversión para el cliente. Tipo de cambio de la institución que autoriza. El valor en este campo se utiliza para calcular el saldo final. El primer dígito contiene el corrimiento del punto decimal desde la posición más a la derecha. El valor por defecto es 61000000 (o sea que la cantidad se multiplica por uno).
366-368	SETL-CRNCY-CDE 9(3)	Fecha y hora de conversión de moneda. Tipo de moneda utilizado en las operaciones y balances del banco propietario del cajero donde se realizó la transacción.
369-376	SETL-CONV-RATE 9(8)	Factor de conversión para el cajero. Tipo de cambio de la entidad que hace la afectación. El valor en este campo se utiliza para calcular el saldo final. El primer dígito contiene el corrimiento del punto decimal desde la posición más a la derecha. El valor por defecto es 61000000 (o sea que la cantidad se multiplica por uno).
377-384	CONV-DAT-TIM Binario 64 S9(18) comp	Fecha y hora de conversión de moneda. Contiene la fecha y hora cuando se realizó la conversión de moneda. El formato es AAMMDDhhmmsscc, donde: AA = Año. MM = Mes. DD = Día. hh = Hora. mm = Minutos. ss = Segundos. cc = Centésimas de segundo.
385-386	RVSL-RSN 9(2)	Razón de reverso o ajuste. Es un indicador que identifica la razón para que el tipo de mensaje sea reverso (0420) o ajuste (5400). Los posibles valores para mensajes de reverso son: 01 = Tiempo excedido. 02 = Orden rechazada. 03 = Destino no disponible.

APENDICE A. DESCRIPCION DE LOS CAMPOS DEL TLF

POSICION	CLAVE Y FORMATO DEL CAMPO	NOMBRE Y DESCRIPCION DEL CAMPO
		<p>08 = Transacción cancelada por el cliente. 10 = Error de equipo. 20 = Transacción dudosa. Los posibles valores para mensajes de ajuste son: 12 = Monto original incorrecto. 13 = Falla del cajero. 14 = Reverso dudoso. 15 = Reverso con importe incorrecto. 16 = Transacción duplicada. 17 = Error de reconciliación. 18 = Efectivo o anticipo agregado. 21 = Falla del Código de Autenticación de Mensajes 22 = Error del MAC en sincronización de llaves. 23 = Error de respuesta de mensaje. 24 = MAC inválido.</p>
387-402	PIN-OFST X(16)	<p>Corrimiento del NIP. Este dato se calcula por el proceso de autorización cada vez que el tarjetahabiente cambia su NIP.</p>
403	SHRG-GRP X(1)	<p>Grupo de compartición. Clave del grupo del cual se forma parte en la compartición de recursos.</p>
404	DEST-ORDER X(1)	<p>Destino que autorizó. Indicador que determina que destino realizó la autorización final. Los posibles valores son: A = Destino alterno. P = Destino primario.</p>
405-410	AUTH-ID-RESP X(6)	<p>Número secuencial asignado por el Host. Contiene el número secuencial asignado por el host.</p>
411-419	REFR	<p>Los siguientes campos indican el impacto al archivo de balance (PBF).</p>
411-411	IMP-IND X	<p>Indicador de impacto al balance. Clave para saber si la transacción afecta o no al balance. Los posibles valores son: 0 = No afecta al balance. 1 = Afecta al balance.</p>
412-413	AVAIL-IMP X occurs 2 times	<p>Impacto al disponible. Los valores posibles son: 0 = No afecta. 1 = Afecta positivamente (suma el saldo). 2 = Afecta negativamente (resta el saldo).</p>
414-415	LEDG-IMP X occurs 2 times.	<p>Impacto al saldo real. Los valores posibles son: 0 = No afecta. 1 = Afecta positivamente (suma el saldo). 2 = Afecta negativamente (resta el saldo).</p>
416-417	HLD-AMT-IMP X occurs 2 times.	<p>Impacto al crédito por depósito. Los valores posibles son: 0 = No afecta. 1 = Afecta positivamente (suma el saldo). 2 = Afecta negativamente (resta el saldo).</p>
418-419	USER-FLD3	<p>Filler.</p>

APENDICE A. DESCRIPCION DE LOS CAMPOS DEL TLF

POSICION	CLAVE Y FORMATO DEL CAMPO	NOMBRE Y DESCRIPCION DEL CAMPO
	X occurs 2 times.	Campo llenado con espacios.
420-420	DEP-SETL-IMP-FLG X	Compensación con depósitos. Clave que indica si los depósitos entran a la compensación. Los posibles valores son: 0 = No entran a la compensación. 3 = Si entran a la compensación.
421-421	ADJ-SETL-IMP-FLG X	Compensación con ajustes. Clave que indica si los ajustes entran a la compensación. 0 = No entran a la compensación. 3 = Si entran a la compensación.
422-425	REFR-IND	Centinela fin de proceso Refresh.
422-422	PBF1 X	Centinela de fin de cheques.
423-423	PBF2 X	Centinela de fin de ahorros.
424-424	PBF3 X	Centinela de fin de tarjeta de crédito.
425	PBF4 X	CPBF en lugar de PBF.
426-441	USER-FLD4 X(16)	Filler. Campo llenado con espacios.
442-452	FRWD-INST-ID-NUM X(11)	Número de ruteo (filler).
453-463	CRD-ACCPT-ID-NUM X(11)	Número de identificación del Host origen.
464-474	CRD-ISS-ID-NUM X(11)	Número de identificación del Host que autorizó.

APENDICE B. CODIGOS DE RESPUESTA

APENDICE B. CODIGOS DE RESPUESTA

Los siguientes son los códigos de respuesta más representativos, que identifican al campo RESP-BYTE-2 que se encuentra en la posición 257-258 del archivo TLF.

CODIGO	DESCRIPCION
00	<p>APROBADA CON BALANCES La transacción fue aprobada y se regresaron balances contables con la respuesta de la transacción. El proceso de Autorización de BASE24-atm regresa un código 00 cuando aprueba las siguientes transacciones bajo el procesamiento CAF/PBF:</p> <ul style="list-style-type: none"> • Disposición de crédito • Disposición de cuenta de cheques • Disposición de cuenta maestra • Depósitos • Consultas
01	<p>APROBADO-SIN BALANCES POR DESPLEGAR La transacción fue aprobada y no se regresaron balances contables con la respuesta de la transacción. El proceso de Autorización de BASE24-atm regresa un código 01 cuando:</p> <p>a) Aprueba transacciones bajo procesamiento solo-NEG, NEG/UAF y solo-CAF. b) Aprueba las siguientes transacciones bajo procesamiento CAF/PBF:</p> <ul style="list-style-type: none"> • Depósito con devolución de efectivo • Depósito fraccionado • Pago desde cuenta de cheques a cuenta de crédito • Pago desde cuenta maestra a cuenta de crédito • Transferencia • Verificación de cheque • Garantía de cheque • Verificación de efectivo • Pago en sobre • Mensaje a la institución financiera <p>c) Aprueba una transacción de solo acceso</p>

APENDICE B. CODIGOS DE RESPUESTA

CODIGO	DESCRIPCION
50	<p>USO NO AUTORIZADO El proceso de Autorización de BASE24-atm regresa un código 50 bajo las siguientes circunstancias:</p> <ul style="list-style-type: none"> a) El código de captura en el NEG no es cero. b) El código de razón en el NEG es 01 (extraviada), 02 (robada), 05 (negada) u 11 (cerrada). c) El estado de la tarjeta en el CAF es 2 (extraviada), 3 (robada), 6 (uso en BASE24-pos) o 9 (cerrada). d) El estado de la tarjeta en el CAF es 4 (restringida) y la transacción no es alguna de las siguientes: <ul style="list-style-type: none"> • Depósito • Depósito con devolución de efectivo • Consulta • Pago con sobre • Mensaje a la institución financiera • Transacción de solo acceso e) El estado de la tarjeta en el CAF es 0 (expedida pero no activa) y la institución o el prefijo de la tarjeta no está habilitado para permitir a los tarjetahabientes seleccionar sus propios NIPs. f) El estado de la tarjeta en el CAF es 0 (expedida pero no activa) y la institución o el prefijo de la tarjeta está habilitado para permitir a los tarjetahabientes seleccionar sus propios NIPs, pero ya hay un corrimiento de NIP en el CAF.
51	<p>TARJETA EXPIRADA La transacción fue negada por que la tarjeta utilizada ha expirado.</p>
52	<p>TARJETA INVALIDA La transacción fue negada por que bajo proceso solo-CAF o CAF/PBF, un registro de CAF no pudo ser accesado por el número de tarjeta y número de miembro involucrado en la transacción.</p>
53	<p>NIP INVALIDO La transacción fue negada debido a alguna de las siguientes condiciones:</p> <ul style="list-style-type: none"> a) El NIP de tarjetahabiente fue introducido incorrectamente o no coincidió con el NIP asociado a la tarjeta utilizada en la transacción. b) Los dos NIPs introducidos en una transacción de cambio de NIP no fueron los mismos.

APENDICE B. CODIGOS DE RESPUESTA

CODIGO	DESCRIPCION
56	<p>CUENTA NO ELEGIBLE La transacción fue negada debido a restricciones o dificultades con la cuenta de aplicación especificada en el registro CAF del tarjetahabiente. El proceso de Autorización de BASE24-atm regresa un código 56 bajo las siguientes circunstancias:</p> <ul style="list-style-type: none"> a) El registro CAF del tarjetahabiente no contiene una cuenta apropiada para la transacción. b) El registro CAF del tarjetahabiente contiene números de cuenta duplicados (dos números de cuenta idénticos del mismo tipo). c) La cuenta de aplicación seleccionada para la transacción tiene un estatus en el CAF de 0 (sin relación), 9 (cerrada), 2 (restringida) o 4 (restringida primaria). d) Una disposición rápida de efectivo requiere que el proceso de autorización proporcione el número de cuenta a utilizar, pero no hay cuenta primaria disponible en el CAF. e) Bajo procesamiento solo CAF o CAF/PBF, el (los) número(s) de cuenta proporcionados para la transacción no están en el CAF. f) Bajo procesamiento solo-CAF o CAF/PBF, se intentó una transacción de verificación de efectivo, pero no hay cuentas listadas en el CAF.
57	<p>TRANSACCION NO REALIZABLE La transacción fue negada por que el propietario del cajero o emisor de la tarjeta no permiten la transacción. El proceso de Autorización de BASE24-atm regresa un código 57 bajo las siguientes circunstancias:</p> <ul style="list-style-type: none"> a) El propietario del cajero no permite la transacción requerida basado en las definiciones de "transacciones definidas" de TDF o ICF. b) El emisor de la tarjeta no permite la transacción requerida basado en las definiciones de "transacciones permitidas" del IDF. c) Se solicitó una transacción de revisión de tarjeta y el nivel de autorización fue 1 o el método de autorización fue NEG o NEG/UAF. d) Se solicitó una transacción de revisión de tarjeta y el proceso de Autorización no pudo obtener información de la cuenta del CAF bajo el método de autorización CAF o CAF/PBF.
58	<p>FONDOS INSUFICIENTES SIN MONTO 3 Se negó la transacción debido a que se requirieron fondos de una cuenta en la cual los disponibles fueron menores o iguales a cero. El proceso de Autorización de BASE24-atm regresa ceros en el campo RQST.AMT-3 del STM.</p>
59	<p>FONDOS INSUFICIENTES RESPECTO AL MONTO 3 Se negó la transacción debido a que se requirieron fondos de una cuenta en la cual los disponibles fueron mayores a cero, pero menores que el monto requerido. El proceso de Autorización de BASE24-atm regresa un código de respuesta 59 solamente bajo procesamiento CAF/PBF. En este caso, el proceso de Autorización localiza el monto de los fondos disponibles en la cuenta en el campo RQST.AMT-3 del STM.</p>
60	<p>LIMITE DE USO EXCEDIDO La transacción fue negada ya que el tarjetahabiente ya había alcanzado el número máximo de disposiciones permitidas para el periodo actual.</p>
61	<p>EL LIMITE DE DISPOSICIONES SERIA EXCEDIDO La transacción fue negada por que podría haber causado que se excediera el límite de disposición del tarjetahabiente para el periodo actual de acumulación de uso.</p>

APENDICE B. CODIGOS DE RESPUESTA

CODIGO	DESCRIPCION
62	<p>INTENTOS DE NIP EXCEDIDOS La transacción fue negada por que el tarjetahabiente ya ha alcanzado el número máximo de intentos de NIP permitidos para el periodo actual.</p>
63	<p>LIMITE DE DISPOSICION YA ALCANZADO La transacción fue negada por que el tarjetahabiente ya ha alcanzado el límite de disposiciones para el periodo actual.</p>
64	<p>DISPOSICION INVALIDA DE EFECTIVO DE TARJETA DE CREDITO La transacción fue negada por que no cumplía con los criterios del emisor de la tarjeta para transacciones de tarjeta de crédito. El proceso de Autorización de BASE24-atm regresa un código 64 bajo las siguientes circunstancias: a) El monto especificado para una disposición de crédito o transferencia de cuenta de crédito estuvo debajo del mínimo especificado en el CPF. b) La diferencia entre el monto de la disposición y el mínimo especificado no fue divisible entre el incremento estándar de la cuenta de crédito especificado en el CPF.</p>
67	<p>MONTO INVALIDO DE REGRESO DE EFECTIVO La transacción fue negada por que el monto de regreso solicitado en un depósito con transacción de regreso de efectivo excedió el monto del depósito.</p>
69	<p>NO HAY COMPARTICION ENTRE EL EMISOR DE LA TARJETA Y EL PROPIETARIO DEL CAJERO La transacción fue negada debido a que el propietario del cajero no tiene relación con el emisor de la tarjeta. El proceso de Autorización de BASE24-atm regresa un código de respuesta 69 cuando el emisor de la tarjeta no es el propietario del cajero y ninguno de los grupos de compartición de cajeros en el TDF o ICF concuerda con los grupos de compartición del emisor en el IDF.</p>
72	<p>DESTINO NO DISPONIBLE La transacción fue negada debido a que el destino al que el proceso de Autorización ruteó la transacción no estaba disponible. El proceso de Autorización de BASE24-atm regresa un código de respuesta 72 bajo las siguientes circunstancias: a) Cuando una petición de transacción es enviada por el proceso de Autorización pero no puede ser depositada en su destino. b) La petición de transacción no puede ser depositada en el host de back-end y BASE24 no tiene definido esperar al host. c) Una consulta no puede obtenerse a través de un host y no puede ser aprobada por BASE24 ya que el método de autorización no es CAF/PBF. d) Una transacción de estado de cuenta no puede obtenerse a través de un host. e) Una transacción de solo acceso requirió proceso por un host pero el host no estaba disponible. f) El nivel de autorización está offline pero la transacción solicitada no está soportada offline.</p>
73	<p>PROBLEMA DE RUTEO La transacción fue negada debido a un problema de configuración. El proceso de Autorización de BASE24-atm regresa un código de respuesta 73 bajo las siguientes circunstancias: a) El proceso de Autorización es el destino alterno pero no reconoce el prefijo. b) El proceso de Autorización no reconoce al prefijo asociado con la transacción y no puede determinar otro destino al cual enviar la transacción.</p>

APENDICE B. CODIGOS DE RESPUESTA

CODIGO	DESCRIPCION
90	<p>TRANSACCION APROBADA-VENTANA INTERNA El proceso de Autorización de BASE24-atm regresa un código de respuesta 90 cuando el cajero permite transacciones administrativas dentro de la ventana de balance especificada en el IDF y la transacción administrativa fue introducida dentro de la ventana de balance de cajero.</p>
91	<p>TRANSACCION APROBADA-VENTANA EXTERNA El proceso de Autorización de BASE24-atm regresa un código de respuesta 91 cuando el cajero permite transacciones administrativas dentro de la ventana de balance especificada en el IDF y la transacción administrativa fue introducida fuera de la ventana de balance de cajero.</p>
92	<p>TRANSACCION APROBADA-BALANCE A CUALQUIER HORA El proceso de Autorización de BASE24-atm regresa un código de respuesta 92 cuando el cajero permite transacciones administrativas a cualquier hora.</p>

APENDICE C. DESCRIPCION DE LOS REPORTES ORIGINALES

Derivado del análisis de la situación original a continuación se presenta cada uno de los reportes originales con su descripción.

Nota: Los nombres de los reportes "MAPn" son las iniciales del nombre del gerente de Banca Electrónica que los solicitó originalmente (Mario Alberto Pedraza).

MAP1 - REGISTRO GLOBAL MENSUAL POR NUMERO DE DISPOSICIONES POR REGION

SORTEADO POR: REGION DESGLOSE DEL MAP2 Y MAP3
CAJERO
BANCO

- Reporte mensual por número de disposiciones por región de los cajeros BanCreceer.
- Deberá contener la cantidad de disposiciones de los tarjetahabientes (ajenos o propios) en los cajeros BanCreceer.
- El reporte deberá ser sorteado por región y por cajeros BanCreceer, realizando una separación por bancos dependiendo del banco dueño de cada tarjeta.

MAP2 - REGISTRO GLOBAL MENSUAL POR NUMERO DE DISPOSICIONES POR BANCO

SORTEADO POR: BANCO RESUMEN DEL MAP1 Y MAP3

- Reporte mensual por número de disposiciones **por Banco**.
- Deberá contener la cantidad de disposiciones de los tarjetahabientes (ajenos o propios) en los cajeros BanCreceer.
- El reporte deberá separar las transacciones por banco, dependiendo del banco dueño de cada tarjeta.
- El campo de **Fecha** deberá contener la fecha del TLF: **POST-DATE**.
- El campo de **Banco** deberá contener el nombre fijo BanCreceer y el de **Clave B161**, ya que este reporte es solo para cajeros BanCreceer.

MAP3 - REGISTRO GLOBAL MENSUAL POR NUMERO DE DISPOSICIONES POR CAJERO

SORTEADO POR: CAJERO BANCO RESUMEN DEL MAP1

- Reporte mensual por número de disposiciones **por Cajero**.
- Deberá contener la cantidad de disposiciones de los tarjetahabientes (ajenos o propios) en los cajeros BanCreceer.
- El reporte deberá ser sorteado por cajeros BanCreceer y separar las transacciones por banco dependiendo del banco dueño de cada tarjeta.
- El campo de **Fecha** deberá contener la fecha del TLF: **POST-DATE**.
- El campo de **Banco** deberá contener el nombre fijo BanCreceer y el de **Clave B161**, ya que este reporte es solo para cajeros BanCreceer.

MAP4 - REGISTRO GLOBAL DIARIO DE TRANSACCIONES EXITOSAS POR REGION

SORTEADO POR: REGION DESGLOSE DEL MAP5
BANCO

- Reporte diario de transacciones exitosas por región de los cajeros BanCreceer.

APENDICE C. DESCRIPCION DE LOS REPORTES ORIGINALES

- Deberá contener la cantidad de transacciones exitosas (Consultas, Retiros, Cambio de NIP e Impresión de Estado de Cuenta), de los tarjetahabientes (ajenos o propios) en cajeros BanCreceer.
- El reporte deberá ser sorteado por región, realizando una separación por bancos de dependiendo del banco dueño de cada tarjeta.

MAP5 - REGISTRO GLOBAL DIARIO DE TRANSACCIONES EXITOSAS POR BANCO

SORTEADO POR: BANCO RESUMEN DEL MAP4 Y MAP6

- Reporte diario de transacciones exitosas **por Banco en cajeros BanCreceer**
- Deberá contener la cantidad de transacciones exitosas (Consultas, Retiros, Cambio de NIP e Impresión de Estado de Cuenta), de los tarjetahabientes (ajenos o propios) en cajeros BanCreceer.
- El reporte deberá realizar una separación por banco, dependiendo del banco dueño de cada tarjeta.

MAP6 - REGISTRO GLOBAL DIARIO DE TRANSACCIONES EXITOSAS POR CAJERO

SORTEADO POR: CAJERO DESGLOSE DEL MAP5
BANCO

- Reporte diario de transacciones exitosas **por Cajeros BanCreceer**
- Deberá contener la cantidad de transacciones exitosas (Consultas, Retiros, Cambio de NIP e Impresión de Estado de Cuenta), de los tarjetahabientes (ajenos o propios) en cajeros BanCreceer.
- El reporte deberá ser sorteado por cajeros BanCreceer y separar las transacciones por banco dependiendo del banco dueño de cada tarjeta.
- El campo de **Fecha** deberá contener la fecha del TLF: **POST-DATE**.
- El campo de **Banco** deberá contener el nombre fijo BanCreceer, ya que este reporte es solo para cajeros BanCreceer.

MAP7-REGISTRO GLOBAL DIARIO DE DISPOSICIONES RECHAZADAS POR CAJERO

SORTEADO POR: CAJERO
BANCO

- Reporte diario de disposiciones rechazadas **por cajero BanCreceer**.
- Deberá contener la cantidad de disposiciones rechazadas de los tarjetahabientes (ajenos o propios) en cajeros BanCreceer.
- El reporte deberá ser sorteado por cajeros BanCreceer y separar las transacciones por banco dependiendo del banco dueño de cada tarjeta.

MAP8 - REGISTRO GLOBAL MENSUAL DE DISPOSICIONES RECHAZADAS POR REGION

SORTEADO POR: REGION DESGLOSE DEL MAP9
BANCO

- Reporte mensual de disposiciones rechazadas **por región de los cajeros BanCreceer**.
- Deberá contener la cantidad de disposiciones rechazadas de los tarjetahabientes (ajenos o propios) en cajeros BanCreceer.
- El reporte deberá ser sorteado por región, realizando una separación por bancos dependiendo del banco dueño de cada tarjeta.

APENDICE C. DESCRIPCION DE LOS REPORTES ORIGINALES

MAP9 - REGISTRO GLOBAL MENSUAL DE DISPOSICIONES RECHAZADAS POR BANCO

SORTEADO POR: BANCO RESUMEN DEL MAP8

- Reporte mensual de disposiciones rechazadas **por banco en cajeros BanCrecer.**
- Deberá contener la cantidad de disposiciones rechazadas de los tarjetahabientes (ajenos o propios) en cajeros BanCrecer.
- El reporte deberá separar las transacciones por banco, dependiendo del banco dueño de cada tarjeta.

MAP10 - REGISTRO GLOBAL MENSUAL DE CONSULTAS RECHAZADAS POR BANCO

SORTEADO POR: BANCO RESUMEN DEL MAP11 Y MAP12

- Reporte mensual de consultas rechazadas **por banco en cajeros BanCrecer.**
- Deberá contener la cantidad de consultas rechazadas de los tarjetahabientes (ajenos o propios) en cajeros BanCrecer.
- El reporte deberá separar las transacciones por banco, dependiendo del banco dueño de cada tarjeta.

MAP11- REGISTRO GLOBAL MENSUAL DE CONSULTAS RECHAZADAS POR REGION

SORTEADO POR: REGION
BANCO DESGLOSE DEL MAP10

- Reporte mensual de consultas rechazadas **por región de los cajeros BanCrecer**
- Deberá contener la cantidad de consultas rechazadas de los tarjetahabientes (ajenos o propios) en cajeros BanCrecer.
- El reporte deberá ser sorteado por región, realizando una separación por bancos dependiendo del banco dueño de cada tarjeta.

MAP12- REGISTRO GLOBAL MENSUAL DE CONSULTAS RECHAZADAS POR CAJERO

SORTEADO POR: CAJERO
BANCO DESGLOSE DEL MAP10

- Reporte mensual de consultas rechazadas **por cajero BanCrecer.**
- Deberá contener la cantidad de consultas rechazadas de los tarjetahabientes (ajenos o propios) en cajeros BanCrecer.
- El reporte deberá ser sorteado por cajeros BanCrecer y separar las transacciones por banco dependiendo del banco dueño de cada tarjeta.

MAP13- REGISTRO GLOBAL DIARIO DE CONSULTAS RECHAZADAS POR CAJERO

SORTEADO POR: CAJERO
BANCO DESGLOSE DEL MAP15

- Reporte diario de consultas rechazadas **por cajero BanCrecer.**
- Deberá contener la cantidad de consultas rechazadas de los tarjetahabientes (ajenos o propios) en cajeros BanCrecer.
- El reporte deberá ser sorteado por cajeros BanCrecer y separar las transacciones por banco dependiendo del banco dueño de cada tarjeta.

MAP14- REGISTRO GLOBAL DIARIO DE CONSULTAS RECHAZADAS POR REGION

SORTEADO POR: REGION
BANCO DESGLOSE DEL MAP15

- Reporte diario de consultas rechazadas **por región de los cajeros BanCrecer.**

APENDICE C. DESCRIPCION DE LOS REPORTES ORIGINALES

- Deberá contener la cantidad de consultas rechazadas de los tarjetahabientes (ajenos o propios) en cajeros BanCrecer.
- El reporte deberá ser sorteado por región, realizando una separación por bancos dependiendo del banco dueño de cada tarjeta.

MAP15- REGISTRO GLOBAL DIARIO DE CONSULTAS RECHAZADAS POR BANCO
SORTEADO POR: BANCO RESUMEN DEL MAP13 Y MAP14

- Reporte diario de consultas rechazadas **por banco en cajeros BanCrecer.**
- Deberá contener la cantidad de consultas rechazadas de los tarjetahabientes (ajenos o propios) en cajeros BanCrecer.
- El reporte deberá separar las transacciones por banco, dependiendo del banco dueño de cada tarjeta.

MAP16- REGISTRO GLOBAL DIARIO POR MONTO DE DISPOSICIONES POR REGION
SORTEADO POR: REGION DESGLOSE DEL MAP18
BANCO

- Reporte diario por monto de disposiciones **por región de los cajeros BanCrecer**
- Deberá contener la suma de los montos de las disposiciones de los tarjetahabientes (ajenos o propios) en los cajeros BanCrecer.
- El reporte deberá ser sorteado por región y por cajeros BanCrecer, realizando una separación por bancos dependiendo del banco dueño de cada tarjeta.

MAP17- REGISTRO GLOBAL DIARIO POR MONTO DE DISPOSICIONES POR CAJERO
SORTEADO POR: CAJERO DESGLOSE DEL MAP18
BANCO

- Reporte diario por monto de disposiciones **por Cajero.**
- Deberá contener la suma de los montos de las disposiciones de los tarjetahabientes (ajenos o propios) en los cajeros BanCrecer.
- El reporte deberá ser sorteado por cajeros BanCrecer y separar las transacciones por banco dependiendo del banco dueño de cada tarjeta.

MAP18- REGISTRO GLOBAL DIARIO POR MONTO DE DISPOSICIONES POR BANCO
SORTEADO POR: BANCO RESUMEN DEL MAP16 Y MAP17

- Reporte diario por monto de disposiciones **por Banco.**
- Deberá contener la cantidad de disposiciones de los tarjetahabientes (ajenos o propios) en los cajeros BanCrecer.
- El reporte deberá separar las transacciones por banco, dependiendo del banco dueño de cada tarjeta.

MAP19- REGISTRO GLOBAL DIARIO POR NUMERO DE DISPOSICIONES EN CAJERO
SORTEADO POR: CAJERO DESGLOSE DEL MAP21
BANCO

- Reporte diario por número de disposiciones aprobadas **por Cajero.**
- Deberá contener la cantidad de disposiciones aprobadas de los tarjetahabientes (ajenos o propios) en los cajeros BanCrecer.
- El reporte deberá ser sorteado por cajeros BanCrecer y separar las transacciones por banco dependiendo del banco dueño de cada tarjeta.

APENDICE C. DESCRIPCION DE LOS REPORTES ORIGINALES

MAP20- REGISTRO GLOBAL DIARIO POR NUMERO DE DISPOSICIONES POR REGION
SORTEADO POR: REGION DESGLOSE DEL MAP21 Y
MAP19

CAJERO
BANCO

- Reporte diario por número de disposiciones **por región de los cajeros BanCrece**
- Deberá contener la cantidad de disposiciones de los tarjetahabientes (ajenos o propios) en los cajeros BanCrece.
- El reporte deberá ser sorteado por región y por cajeros BanCrece, realizando una separación por bancos dependiendo del banco dueño de cada tarjeta.

MAP21- REGISTRO GLOBAL DIARIO POR NUMERO DE DISPOSICIONES POR BANCO
SORTEADO POR: BANCO RESUMEN DEL MAP19 Y MAP20

- Reporte diario por número de disposiciones **por Banco**.
- Deberá contener la cantidad de disposiciones de los tarjetahabientes (ajenos o propios) en los cajeros BanCrece.
- El reporte deberá separar las transacciones por banco, dependiendo del banco dueño de cada tarjeta.

MAP22- REGISTRO GLOBAL MENSUAL POR MONTO DE DISPOSICIONES POR
CAJERO

SORTEADO POR: CAJERO CHECAR ENTRE EL MAP23
BANCO

- Reporte mensual por monto de disposiciones **por Cajero**.
- Deberá contener la suma de los montos de las disposiciones aprobadas y reversos parciales de los tarjetahabientes (ajenos o propios) en los cajeros BanCrece.
- El reporte deberá ser sorteado por cajeros BanCrece y separar las transacciones por banco dependiendo del banco dueño de cada tarjeta.

MAP23- REGISTRO GLOBAL MENSUAL POR MONTO DE DISPOSICIONES POR
REGION

SORTEADO POR: REGION CHECAR ENTRE EL MAP22
BANCO

- Reporte mensual por monto de disposiciones **por región de los cajeros BanCrece**
- Deberá contener la suma de los montos de las disposiciones aprobadas y reversos parciales, de los tarjetahabientes (ajenos o propios) en los cajeros BanCrece.
- El reporte deberá ser sorteado por región y por cajeros BanCrece, realizando una separación por bancos dependiendo del banco dueño de cada tarjeta.

A continuación se presentan los criterios que aplican a todos los reportes.

CRITERIOS GENERALES

1. El campo de **Fecha** deberá contener la fecha del TLF: **POST-DATE**.
2. El campo de **Banco** deberá contener el nombre fijo BanCrece, ya que este reporte es solo para cajeros BanCrece.

APENDICE C. DESCRIPCION DE LOS REPORTES ORIGINALES
PARA IDENTIFICAR LA REGION

1. Debajo del titulo el reporte deberá llevar el nombre de la región. Se identificará con el campo: **REGN-ID** Se deberá tomar este campo con la información previamente obtenida de una tabla de DB2 y esta información servirá como llave para obtener la descripción de la misma, a través de otra tabla de DB2.
2. El campo de **Fecha** deberá contener la fecha del **TLF POST-DATE**.
3. El campo de **Banco** deberá contener el nombre fijo **BanCrecer** y el de **Clave B161**, ya que este reporte es solo para cajeros **BanCrecer**.

PARA DETECTAR NUESTROS CAJEROS

1. Nuestros cajeros se identificarán con el campo: **TERM-FIID = B161**
2. En "**NOMBRE DE CAJERO**" debe contener lo que traiga el campo **TERM-ID**

PARA DETECTAR LAS DISPOSICIONES

1. DISPOSICION DE CUENTA DE CHEQUES.

Será la suma de todas las disposiciones aprobadas y reversos parciales de cheques de tarjetahabientes (propios o ajenos), en los cajeros **BanCrecer** de la región. Los valores por campo son los siguientes:

```
T-CDE      = 10
T-FROM     = 01
USER-FLD1  = 0
RESP-BYTE  < 10
REC-TYP    NOT=21
TYP        NOT=9980
```

2. DISPOSICION DE CUENTA MAESTRA.

Será la suma todas las disposiciones aprobadas y reversos parciales con cuenta maestra de tarjetahabiente (propio o ajeno) que hayan realizado alguna transacción en cajeros **BanCrecer** por Región. Los valores por campo son los siguientes:

```
T-CDE      = 10
T-FROM     = 11
USER-FLD1  = 0
RESP-BYTE  < 10
REC-TYP    NOT=21
TYP        NOT=9980
```

3. DISPOSICION DE TARJETA DE CREDITO.

Será la suma de todas las disposiciones aprobadas y reversos parciales de crédito de tarjetahabientes (propios o ajenos), en los cajeros **BanCrecer** de la región. Los valores por campo son los siguientes:

```
T-CDE      = 10
T-FROM     = 31
USER-FLD1  = 0
RESP-BYTE  < 10
REC-TYP    NOT=21
TYP        NOT=9980
```

**APENDICE C. DESCRIPCION DE LOS REPORTES ORIGINALES
PARA LOS REVERSOS PARCIALES**

Se deben de tomar en cuenta los reversos parciales como disposiciones, considerando lo siguiente:

Cada vez que se encuentre un TYP 0420 se tendrá que validar el AMT-2 de la transacción 0420 y este AMT-2 debe contener una cantidad diferente de ceros, si es así se trata de un reverso parcial y se deberá contabilizar como una transacción aprobada y se deberá eliminar el 210 de esta misma transacción.

Para hacer casar entre el TYP = 0210 y el = 0420 se deberá hacer con los campos de SEQ-NUM y el PAN.

PARA IDENTIFICAR AL BANCO EMISOR DE LAS TARJETAS

Para identificar a los tarjetahabientes que utilizaron los cajeros BanCrecer se utiliza el campo CRD-FIID y esta información servirá como llave para obtener la descripción de la misma, a través de una tabla de DB2 dentro del host IBM.

1. Para "NUMERO" se deberá tomar en cuenta el campo: CRD-FIID
2. Para "NOMBRE" se deberá validar la tabla arriba mencionada por medio del campo CRD-FIID y poner el nombre de Banco que le corresponda.
3. TOTAL GLOBAL
Es la suma de todas las disposiciones aprobadas y reversos parciales por columna (Cheques, Maestra, Crédito y Total).

NOTA: Este mismo reporte deberá realizarse para BanCrecen. Solo se considerarán los siguientes cambios:

Banco	CRD-FIID	TERM-FIID
BanCrecer	B161	B161
BanCrecen	B170	B170

PARA CASOS ESPECIALES

Caso 1: Existen casos en que las transacciones de un reverso (0420) tienen diferente código de respuesta que su transacción original (0210, 0215, 0220) en estos casos se tendrá que validar lo siguiente:

TYP=(0210,0215,0220) con RES-BYTE=>09 y
TYP=0420 con RES-BYTE=<09
Esta transacción se ignorará

Caso 2: Existe la posibilidad de que por la hora del corte de los cajeros, en el caso de una transacción de reverso (0420) quede grabada incompleta, esto es que el TYP (0210,0215, o 0220) se grave en un día diferente al del TYP 0420. El TYP (0210, 0215 o 0200) no se podrá detectar y se considerará como una transacción normal, pero el 0420 quedará sin su TYP (0210, 0215 o 0220), en este caso no se tomará en cuenta este registro (0420).

NOTA: Todas las validaciones del campo TYP deberán considerar 0210, 0215, 0220 y 0420

APENDICE C. DESCRIPCION DE LOS REPORTES ORIGINALES
PARA LAS TRANSACCIONES

1- DISPOSICIONES.

Será la suma de todas las disposiciones aprobadas y reversos parciales de Cheques, Cuentas maestras y Crédito de tarjetahabientes (propios o ajenos), en cajeros BanCrecer. Los campos y sus valores son los siguientes:

T-CDE = 10 - Retiro.
USER-FLD1 = 0
RESP-BYTE-2 <10
T-FROM = 01 - Cheques
T-FROM = 11 - Maestra
T-FROM = 31 - Crédito
REC-TYP NOT=21
TYP NOT=9980

2- CONSULTAS

Los campos y sus valores son los siguientes:

T-CDE = 30 - Consultas
USER-FLD1 = 0
RESP-BYTE-2 <10
T-FROM = 01 - Cheques
T-FROM = 11 - Maestra
T-FROM = 31 - Crédito
REC-TYP NOT=21
TYP NOT=9980

3 - ESTADO DE CUENTA

Los campos y sus valores son los siguientes:

T-CDE = 70 - Estado de cuenta
RESP-BYTE-2 <10
USER-FLD1 = 0
T-FROM = 01 - Cheques
T-FROM = 11 - Maestra
T-FROM = 31 - Crédito
REC-TYP NOT=21
TYP NOT=9980

4 - CAMBIOS DE NIP

Para cambio de NIP se incluirá una columna especial, ya que en este caso no se registra a que Cuenta se le realizó el cambio. Los campos y sus valores son los siguientes:

T-CDE = 81
RESP-BYTE-2 <10
REC-TYP NOT=21
TYP NOT=9980

APENDICE C. DESCRIPCION DE LOS REPORTES ORIGINALES
PARA DETECTAR LAS DISPOSICIONES RECHAZADAS

1 - DISPOSICIONES RECHAZADAS.

Será la suma de todas las disposiciones rechazadas y reversos totales de Cheques, Cuentas maestras y Crédito de tarjetahabientes (propios o ajenos), en cajeros BanCrecer.

```
T-CDE      = 10 - Retiro.  
USER-FLD1 = 0  
RESP-BYTE-2 >9  0  REC-TIP=21  0  TYP=9980  
T-FROM     = 01 - Cheques  
T-FROM     = 11 - Maestra  
T-FROM     = 31 - Crédito
```

PARA DETECTAR LOS REVERSOS TOTALES "RECHAZOS".

Se deben de tomar en cuenta los reversos totales como rechazos de las disposiciones, considerando lo siguiente:

Cada vez que se encuentre un TYP 0420, se tendrá que validar el AMT-2 de la transacción 0420 y este AMT-2 debe contener ceros, si es así se trata de un reverso total y se considerará como un rechazo. Se tendrá que eliminar el 210 de esta misma transacción.

APENDICE D. MANUAL DE INSTALACION

Antes de ejecutar la instalación de MetaCube Explorer, es necesario cerrar todas las aplicaciones, incluyendo antivirus, Microsoft Office y la barra rápida de Microsoft Office, entre otras. Si durante la instalación aparece un error que indique que el programa instalador no pudo sobrescribir el archivo msvcr20.dll, es que hay un conflicto entre el programa instalador y la barra rápida de Microsoft Office. Si la barra rápida de Microsoft Office se carga al iniciar la PC, hay que removerla temporalmente de la carpeta Inicio (Startup). Iniciar la PC nuevamente y ejecutar otra vez el programa de instalación.

1. INICIANDO EL PROGRAMA DE INSTALACION

Para comenzar el programa de instalación de MetaCube, inserte el CD-ROM en el drive de su PC. Use la opción **Ejecutar...** (**Run...** para Windows en inglés) del sistema operativo Windows, tal y como se muestra en la figura siguiente:

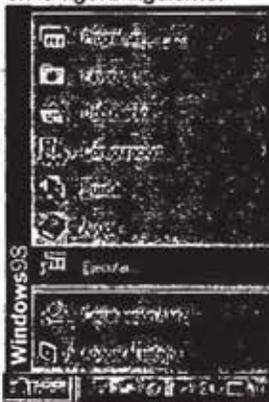


Figura D.1 Selección de "Ejecutar" para iniciar la instalación

Dé clic en la opción **Ejecutar...** y teclee el drive designado para el CD-ROM, seguido por el comando "setup"; por ejemplo:

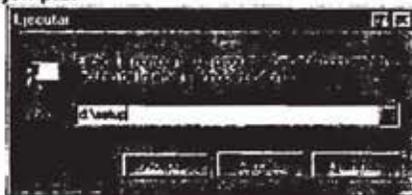


Figura D.2 Nombre del programa de instalación

Presione el botón **Aceptar** (**Install**, para Windows en Inglés). Instalar MetaCube de esta manera ayuda al sistema operativo a desinstalar el producto de ser necesario, identificando y borrando todos los archivos cargados por el programa de instalación dentro del directorio MetaCube y otros directorios del disco duro, removiendo entradas de registro y borrando archivos de inicialización cargados por el programa de instalación. MetaCube también

incluye su propia utilidad de desinstalación. Al empezar la instalación, aparece la siguiente pantalla:



Figura D.3 Preparación del ambiente para la instalación

1.2 PRIMEROS PASOS DE LA INSTALACION

La primera pantalla del proceso de instalación es la presentación de la aplicación, da algunas recomendaciones y algunas precauciones que hay que tomar antes de proceder a la instalación. Una vez que haya leído esta primer pantalla, puede hacer clic sobre el botón Next > para proseguir a la siguiente pantalla.

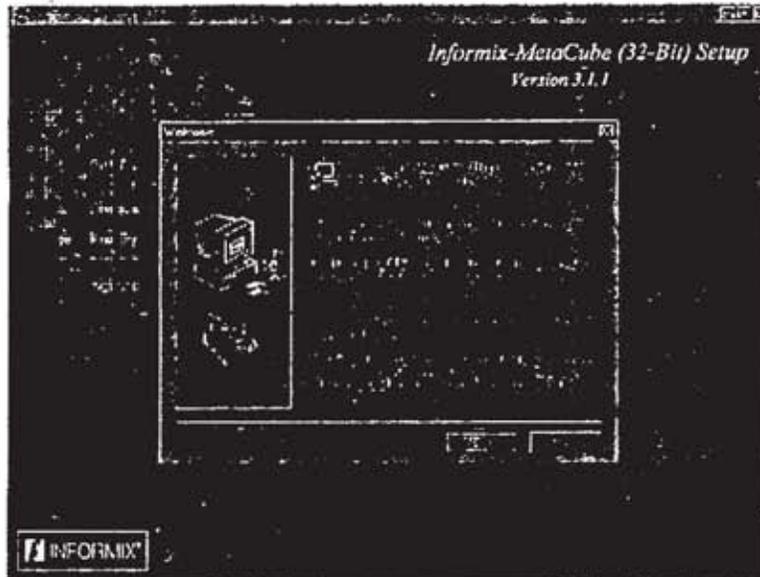


Figura D.4 Mensaje de bienvenida a la instalación

La función de los botones en esta pantalla, es la siguiente:

- Next: continúa con la instalación de la aplicación.
- Cancel: terminará el proceso antes de haber instalado la aplicación.

Por omisión, el programa de instalación instala el software MetaCube dentro del directorio MetaCube. Si el programa de instalación detecta una previa instalación de MetaCube, puede instalar la nueva versión en el mismo directorio. Si el directorio MetaCube no existe, el programa de instalación lo creará, como se muestra en la siguiente figura.

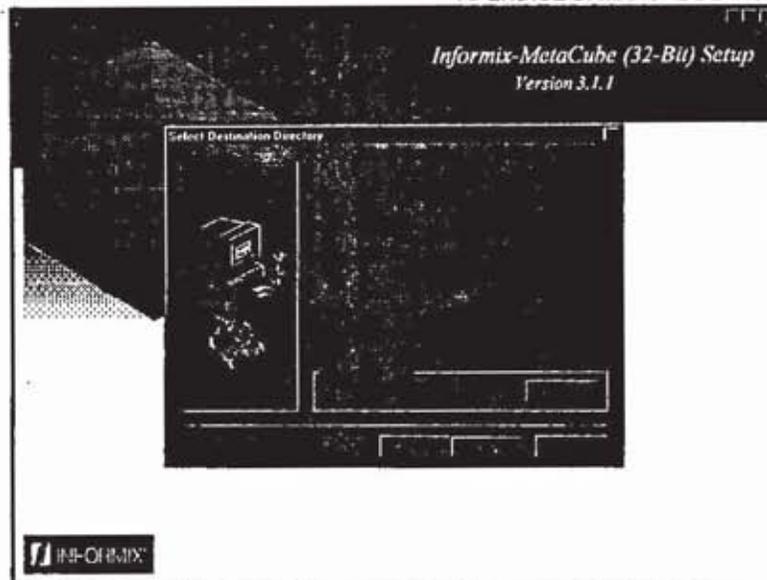


Figura D.5 Selección del directorio destino

La función de los botones en esta pantalla, es la siguiente:

- **Browse:** presenta una caja de diálogo llamada "Choose Directory" en la cual podrá seleccionar otro directorio, donde desea que la aplicación sea instalada.
- **Back:** regresa a la pantalla anterior.
- **Next:** continúa con la instalación de la aplicación.
- **Cancel:** terminará el proceso antes de haber instalado la aplicación.

1.3 ELIGIENDO EL SOFTWARE A INSTALAR

Una vez que ha seleccionado el directorio para los archivos de programas de MetaCube, el programa de instalación preguntará para que se le indique que archivos deberán ser instalados.

El programa instalador ofrece opciones de instalación que corresponden a tres tipos de perfiles de usuarios:

- Usuario final
- Administrador
- Desarrollador MetaCube.

Hay también una opción **Custom** (personalizada), la cual permite especificar exactamente las aplicaciones y archivos de soporte a instalar.

La pantalla que se despliega al entrar en este punto, es la siguiente:

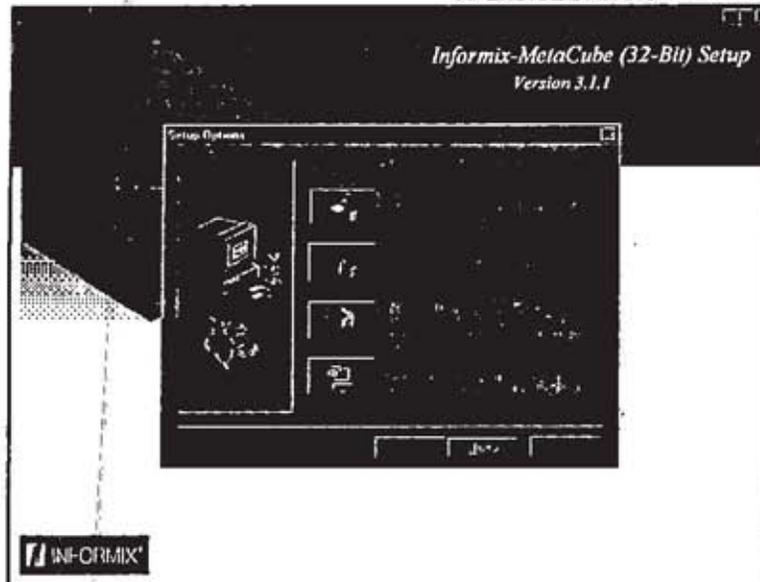


Figura D.6 Elección del software a instalar

La explicación de los iconos del perfil de instalación, es la siguiente:



Instalación para un usuario final.
Recomendada para usuarios de MetaCube Explorer, Excel y otras aplicaciones.



Instalación para un administrador.
Recomendado para el administrador del DataWarehouse y/o administradores de MetaCube.



Instalación para un desarrollador de MetaCube.
Recomendada para desarrolladores de aplicaciones que usan la API de MetaCube para desarrollar aplicaciones o funciones personalizadas.



Custom (personalizada).
Recomendada para usuarios avanzados y administradores de sistemas solamente. Se puede personalizar todas las opciones disponibles del Setup.

La función de los botones en esta pantalla, es la siguiente:

- Back: regresa a la pantalla anterior (Figura D.5).
- Next: continúa con la instalación de la aplicación (este botón permanece desactivado hasta que un tipo de instalación es elegido).
- Cancel: terminará el proceso antes de haber instalado la aplicación.

La tabla siguiente muestra las opciones estándar de instalación para usuarios, desarrolladores y administradores, asumiendo que todos los componentes del software de cliente MetaCube están disponibles.

APENDICE D. MANUAL DE INSTALACION

Componente	Usuario	Desarrollador	Administrador
MetaCube Analysis engine	✓	✓	✓
MetaCube Explorer	✓	✓	✓
MetaCube for Excel	✓	✓	✓
Web Publisher	✓	✓	✓
Warehouse Manager			✓
Agent Administrator			✓
Warehouse Optimizer			✓
Extension Wizard		✓	
Demonstration Database	✓	✓	✓
Server Installation Command Files			✓
Informix-CLI	✓	✓	✓

Tabla D.1 Componentes a instalar de acuerdo al perfil del usuario

Nota: Aunque el componente *Extension Wizard* se ejecuta sobre todas las plataformas de 32 bits, se recomienda usar Windows NT versión 3.51 o posterior para cualquier desarrollo de Visual C++.

1.3.1 INSTALACION PARA USUARIO FINAL Y PARA ADMINISTRADOR

Si alguna de estas dos opciones fue la escogida, no tendrá que especificar ninguna otra cosa. Pase al punto 1.4.

1.3.2 INSTALACION PARA UN DESARROLLADOR METACUBE

Si este fue el tipo de instalación elegido, se necesitará tener instalado el producto **Microsoft Visual C ++**. Se deberá especificar el directorio donde se debe instalar un componente de MetaCube llamado "MetaCube Extension Wizard". Para este propósito aparece la siguiente pantalla:

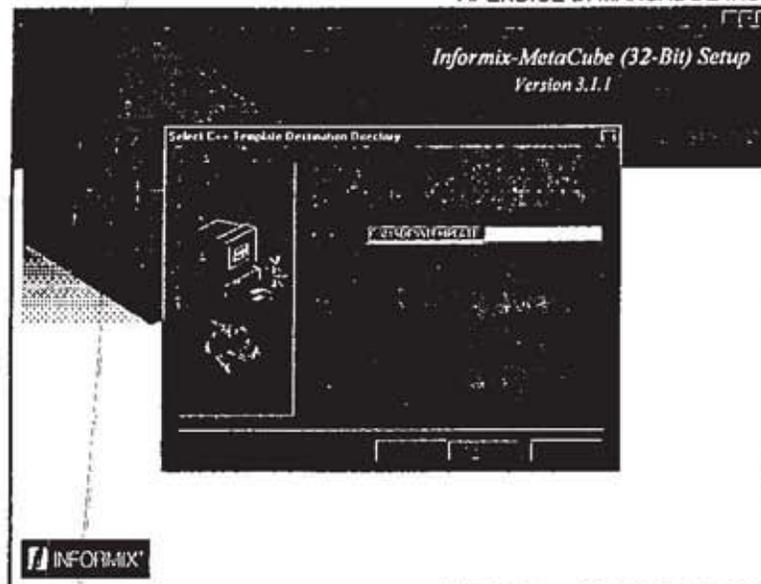


Figura D.7 Selección de un directorio para desarrollo con C++

La función de los botones en esta pantalla, es la siguiente:

- Back: regresa a la pantalla anterior (Figura D.6).
- Next: continúa con la instalación de la aplicación.
- Cancel: terminará el proceso antes de haber instalado la aplicación.

1.3.3 INSTALACION PERSONALIZADA

La opción personalizada (Custom) permite especificar exactamente las aplicaciones y archivos de soporte a instalar. Para seleccionar cualquier componente disponible, de clic en la caja a la izquierda de ese componente, dejando una marca. Para borrar la selección de un componente, de clic nuevamente en la caja y con esto se removerá la marca, como se muestra en la siguiente figura.

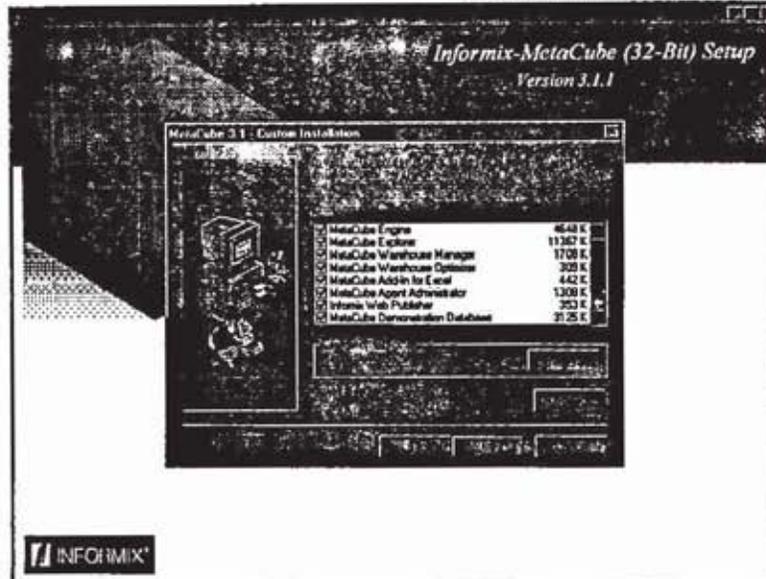


Figura D.8 Instalación personalizada

La función de los botones en esta pantalla, es la siguiente:

- **Browse:** presenta una caja de diálogo llamada "Choose Directory" en la cual podrá seleccionar otro directorio, donde desea que la aplicación sea instalada
- **Disk Space:** presenta una caja de diálogo llamada "Available Disk Space" donde se puede especificar el drive, donde se desea que la aplicación sea instalada.
- **Back:** regresa a la pantalla anterior (Figura D.7).
- **Next:** continúa con la instalación de la aplicación.
- **Cancel:** terminará el proceso antes de haber instalado la aplicación.

Nota: si entre los componentes escogidos esta "MetaCube Extension Wizard", deberá seguir las instrucciones del punto 1.3.2.

1.5 INSTALACION DE ARCHIVOS DE LA APLICACION

Después de haber definido todas las opciones requeridas por el programa de instalación, el siguiente paso es la instalación de los archivos de la aplicación. La pantalla previa a la instalación de los archivos nos muestra los parámetros definidos, se pueden checar y si todo está correcto, continuar con el proceso.

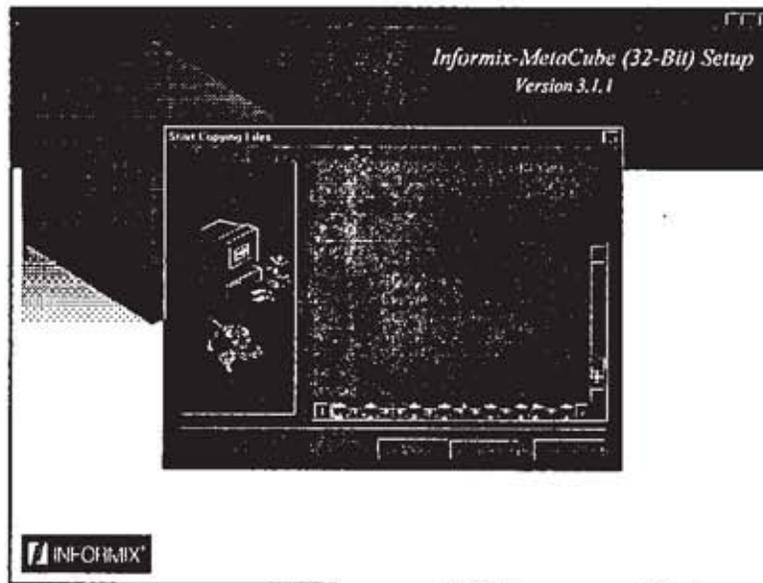


Figura D.10 Selecciones de usuario realizadas

La función de los botones en esta pantalla, es la siguiente:

- Back: regresa a la pantalla anterior (Figura D.9).
- Next: continúa con la instalación de la aplicación.
- Cancel: terminará el proceso antes de haber instalado la aplicación.

Al estar copiando los archivos de la aplicación, mostrará las siguientes pantallas:

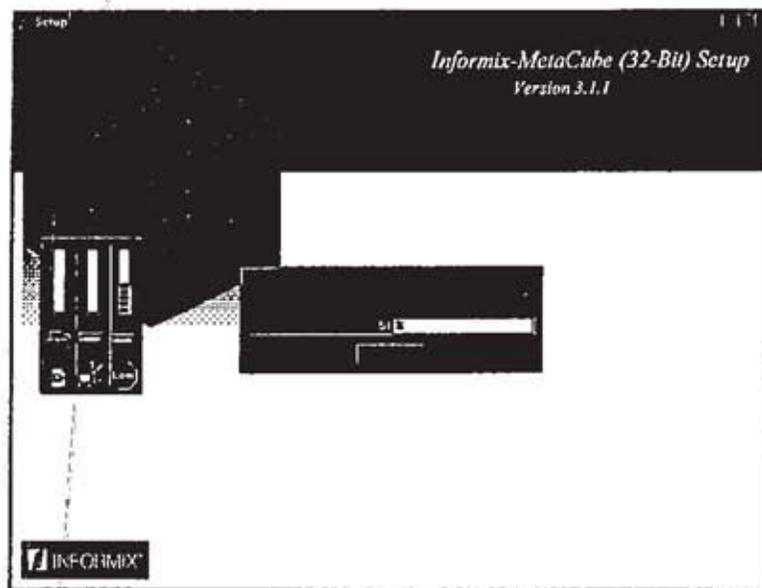


Figura D.11 Copia de los archivos

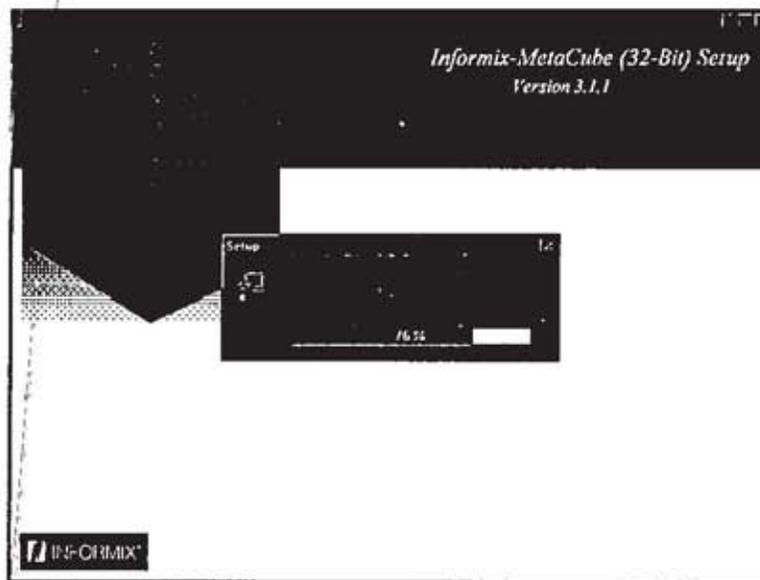


Figura D.12 Preparación del ambiente de instalación

Al finalizar la copia de los archivos, mostrará la siguiente pantalla:

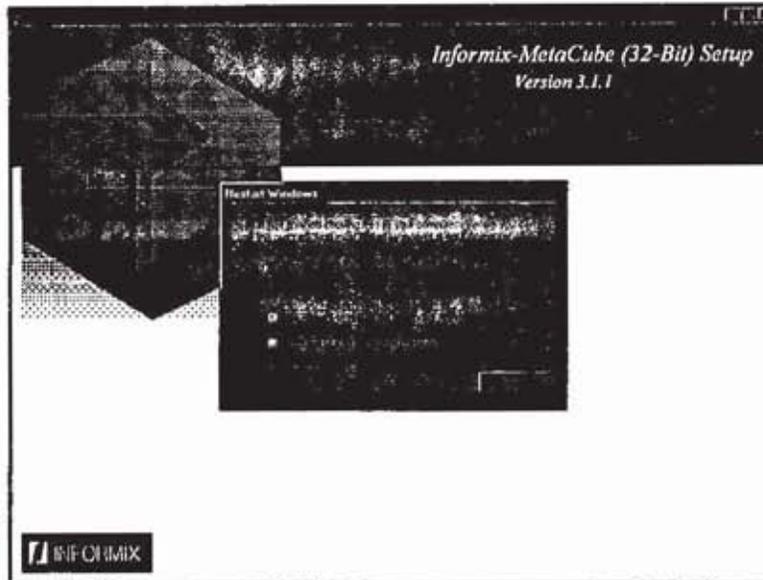


Figura D.13 Ventana de selección para reiniciar Windows

Para poder continuar con la instalación de INFORMIX-CLI, elegir la opción "No, I will restart my computer later" y dar clic en el botón "OK".

2. INSTALACION DE INFORMIX-CLI

Para que las aplicaciones cliente se conecten a un servidor de base de datos de INFORMIX, el software INFORMIX-CLI debe ser instalado en la PC. Durante la instalación de MetaCube, puede elegir instalar el software de INFORMIX-CLI como parte de la instalación de MetaCube, o lo puede instalar por separado posteriormente. Si elige instalar INFORMIX-CLI durante la instalación de MetaCube, el instalador de MetaCube empezará una segunda instalación que carga el software de INFORMIX-CLI.

El paquete de conectividad INFORMIX-CLI consiste en software de conexión para el servidor de bases de datos de INFORMIX y un administrador de ODBC. En adición, la documentación en línea de INFORMIX-CLI es instalada. Está puede ser encontrada en el subdirectorio, llamado *release*, en el directorio donde INFORMIX-CLI es instalado (por defecto, este directorio es nombrado como *Informix*).

La pantalla inicial de la instalación de INFORMIX-CLI es la siguiente:

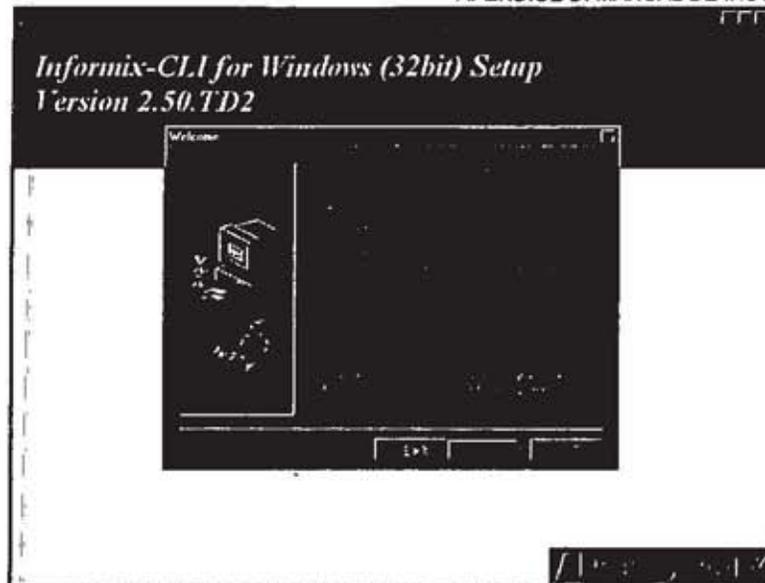


Figura D.14 Pantalla de bienvenida a la instalación de Informix-CLI

La función de los botones en esta pantalla, es la siguiente:

- Back: regresa a la pantalla anterior (permanece inactivo).
- Next: continúa con la instalación de la aplicación.
- Cancel: terminará el proceso antes de haber instalado la aplicación.

2.1 REQUERIMIENTOS DEL SISTEMA

Al continuar con la instalación, se despliega una pantalla como la que se muestra a continuación:

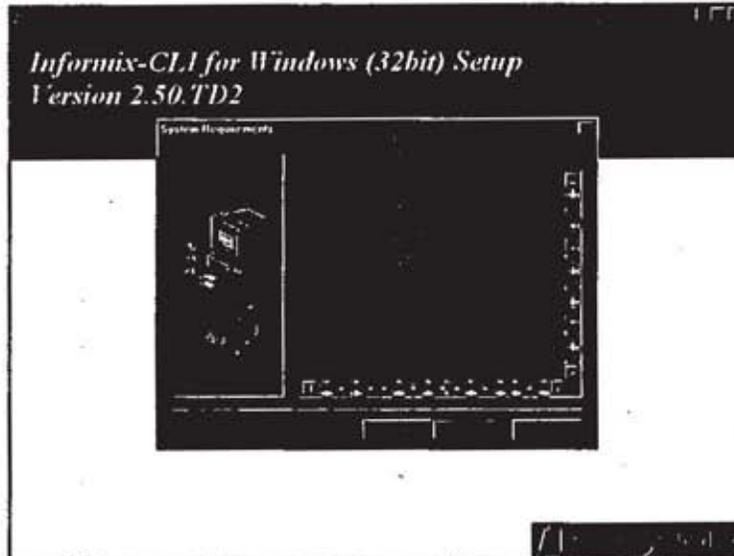


Figura D.15 Pantalla de aviso de requerimientos del sistema

2.1.1 SISTEMA OPERATIVO Windows 95 O Windows NT

Se debe contar con esta versión del sistema operativo para utilizar MetaCube.

2.1.2 DIRECCION TCP/IP

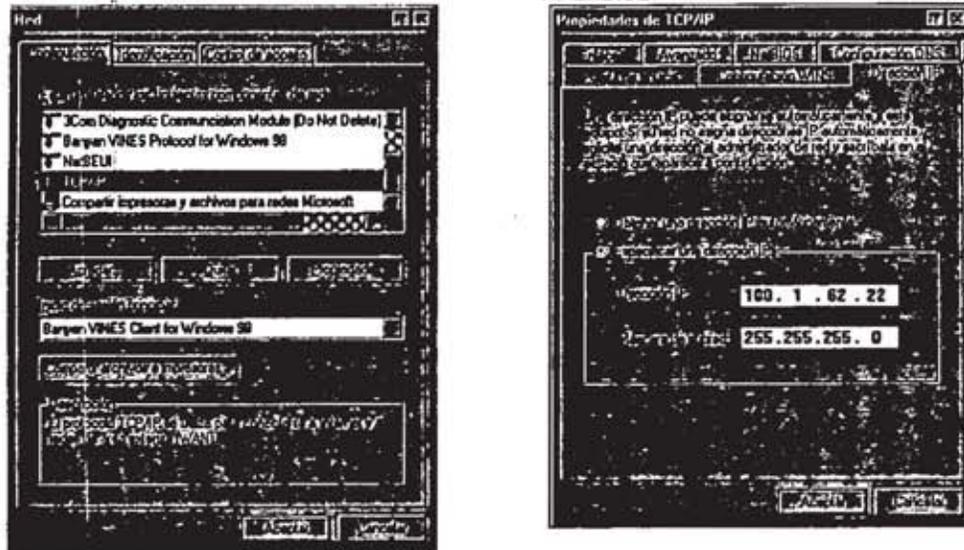
Para definir una dirección TCP/IP independiente por cada máquina cliente se debe entrar al Panel de Control, seleccionar el icono Red, como el que se muestra a la derecha.



Enseguida aparecerá una pantalla como la que se muestra enseguida, seleccionar la opción TCP/IP.

En la pantalla "Propiedades de TCP/IP" como la que se muestra enseguida, proporcionar la Dirección IP asignada a la PC cliente.

APENDICE D. MANUAL DE INSTALACION



2.2 REGISTRO DEL USUARIO Y DE LA COMPAÑIA

La pantalla siguiente es para hacer el registro del nombre del usuario y el nombre de la compañía.

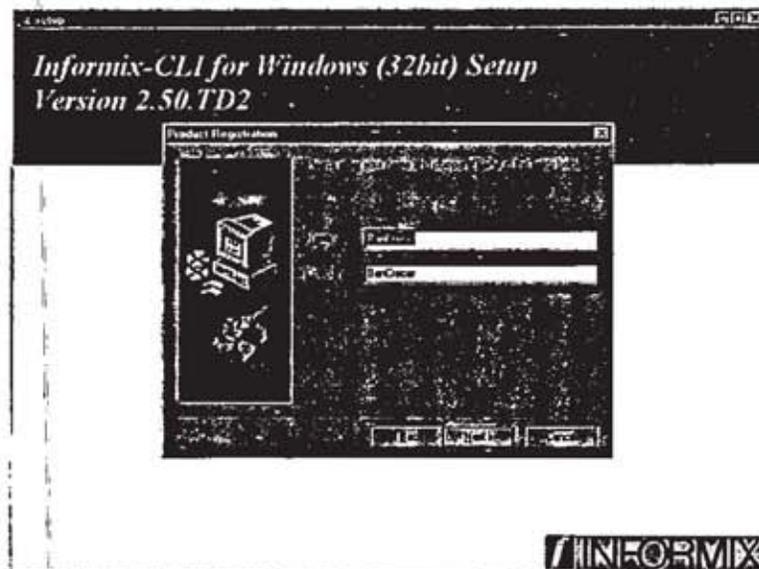


Figura D.16 Registro del nombre del usuario

La siguiente pantalla es para confirmar la información proporcionada del nombre del usuario y de la compañía.

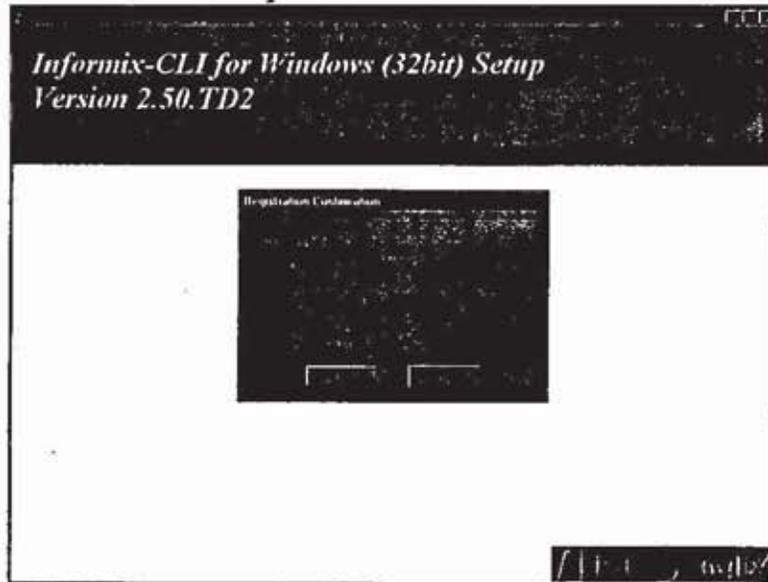


Figura D.17 Confirmación del registro

2.3 REGISTRO DE LAS CLAVES DE ACCESO

Durante la instalación del software de INFORMIX-CLI, se preguntará un número de serie y un número de llave.

El programa de instalación no continuará hasta que haya capturado estos datos.

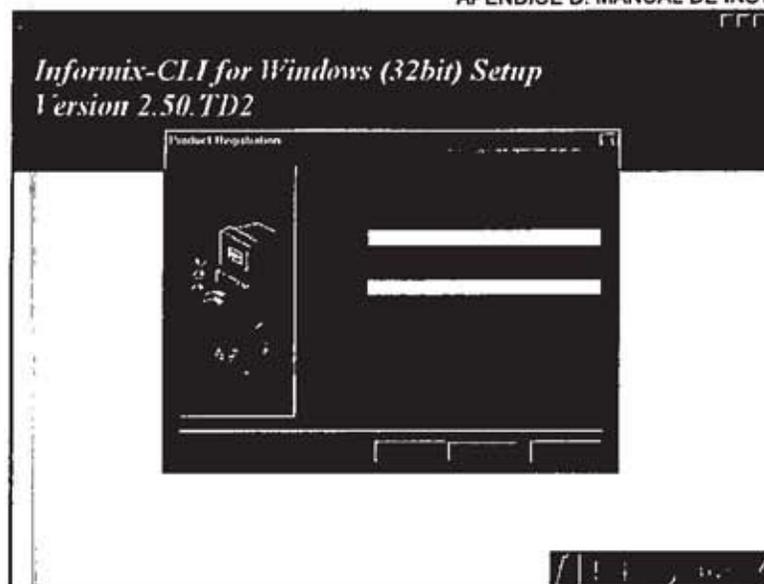


Figura D.18 Pantalla de claves para instalación

La función de los botones en esta pantalla, es la siguiente:

- Back: regresa a la pantalla anterior.
- Next: continúa con la instalación de la aplicación.
- Cancel: terminará el proceso antes de haber instalado la aplicación.

2.4 SELECCION DE LA CARPETA Y DIRECTORIO PARA LA APLICACION

Después de haber introducido los datos anteriores, se tiene que escoger la carpeta de Windows 9x, donde se dejarán todos los componentes de la aplicación. Por defecto, el programa de instalación crea la carpeta "Informix -CLI 32". En este punto de la instalación, aparece la siguiente pantalla:

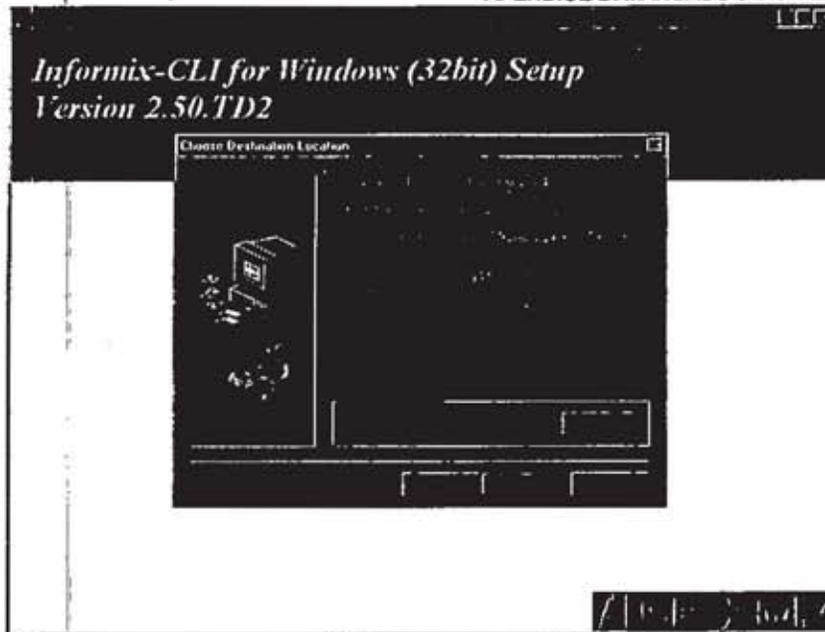


Figura D.20 Pantalla de directorio destino

La función de los botones en esta pantalla, es la siguiente:

- Browse: presenta una caja de diálogo llamada "Choose Directory" en la cual podrá seleccionar otro directorio, donde desea que la aplicación sea instalada.
- Back: regresa a la pantalla anterior .
- Next: continúa con la instalación de la aplicación.
- Cancel: terminará el proceso antes de haber instalado la aplicación.

2.5 INSTALACION DE ARCHIVOS DE LA APLICACION

Al estar copiando los archivos de la aplicación, mostraré las siguientes pantallas:

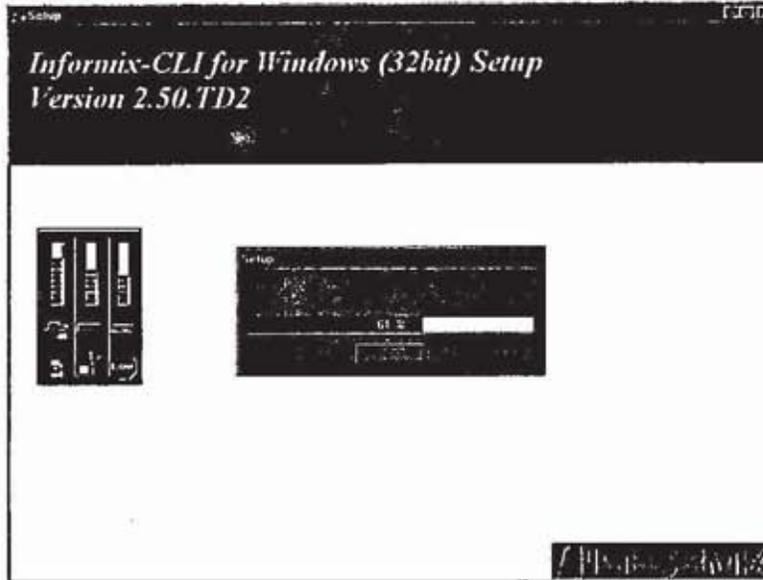


Figura D.21 Pantalla de progreso de la instalación

Al finalizar la copia de los archivos, mostrará la siguiente pantalla:

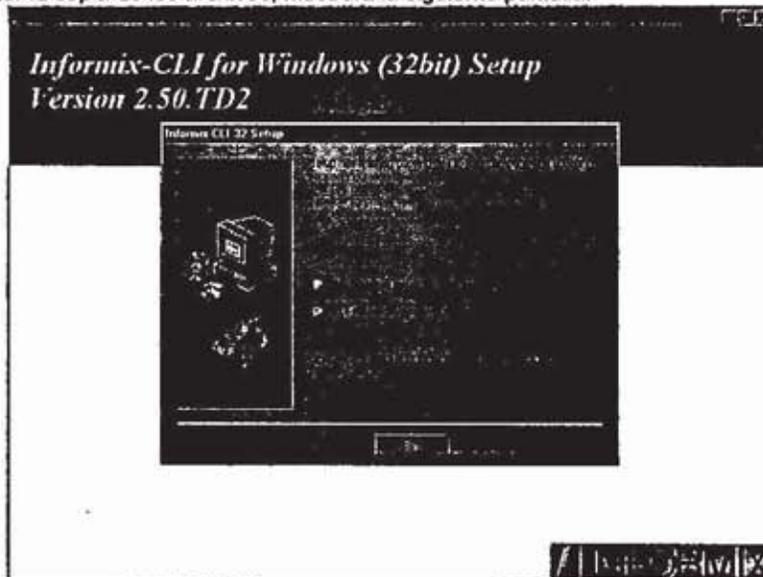


Figura D.22 Pantalla de selección de opciones al terminar la instalación

2.6 PASOS FINALES DE INSTALACION

El programa de instalación permite confirmar las opciones elegidas de instalación. Puede regresar atrás y hacer cambios, si es necesario.

Si el programa de instalación no pudo cargar todos los archivos durante la instalación, preguntará si desea reiniciar la PC. Reiniciar la PC permite al programa de instalación completarla, cargando todos los archivos MetaCube.

3. INSTALACION DE UNA FUENTE DE DATOS CON ODBC

Abrir el panel de control y en Data Sources elegir Add, como se muestra en la siguiente figura:



Figura D.23 Selección del tipo de fuente de datos

Enseguida introducir los datos como los que se muestran a continuación:

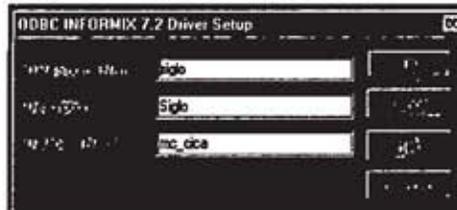


Figura D.24 Definición de características del driver de conectividad

Después seleccionar el botón Advanced y aparecerá una pantalla en la que se deben introducir los datos como los que se muestran a continuación:

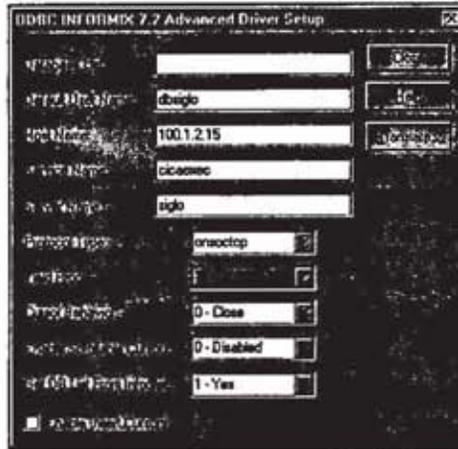


Figura D.25 Pantalla de definición de características de conectividad

Para terminar se debe dar Close y Ok en la ventana anterior. Dentro del directorio WINDOWS debe existir un archivo SERVICES (sin extensión) con la siguiente línea:
 cicaexec 1529/tcp.

4. DEFINIENDO CARACTERISTICAS DE CONEXION

Una vez terminada la instalación del software de MetaCube, se deben definir las características de conexión para MetaCube Explorer para que se conecte a la base de datos Informix que almacena el DataWarehouse. Esto se hace de la siguiente manera:

4.1 INICIAR METACUBE EXPLORER

Iniciar la aplicación MetaCube Explorer como se muestra en la siguiente figura:

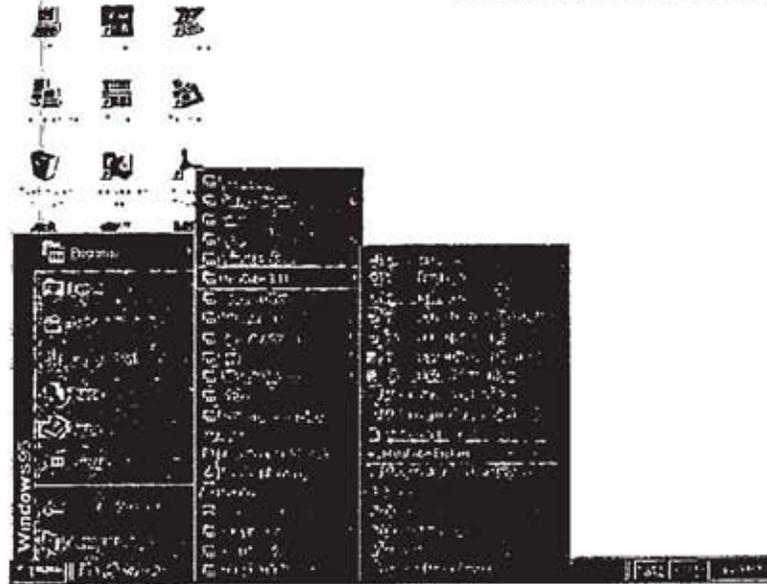


Figura D.26 Selección de la aplicación MetaCube

Dé clic en el icono de **MetaCube Explorer** y la aplicación empezará a cargarse. Al recién entrar a la aplicación se despliega la siguiente pantalla:



Figura D.27 Pantalla de solicitud de claves de acceso

APENDICE D. MANUAL DE INSTALACION

En esta pantalla deberá capturar su usuario y la contraseña, para poder acceder el servidor.

4.2 ACCESANDO LA CAJA DE DIALOGO "Preferences"

El lugar donde se definen las características de conexión es en la caja de diálogo "Preferences". Para acceder a dicha caja debe hacer lo siguiente:

- Dar clic en el botón cancelar de la caja de diálogo MetaCube Login.
- En la barra de menú, dar clic en la opción Tools, se despliega una caja de menú y dentro de esta, dar clic en la opción Options.... Esto se muestra en la siguiente figura:

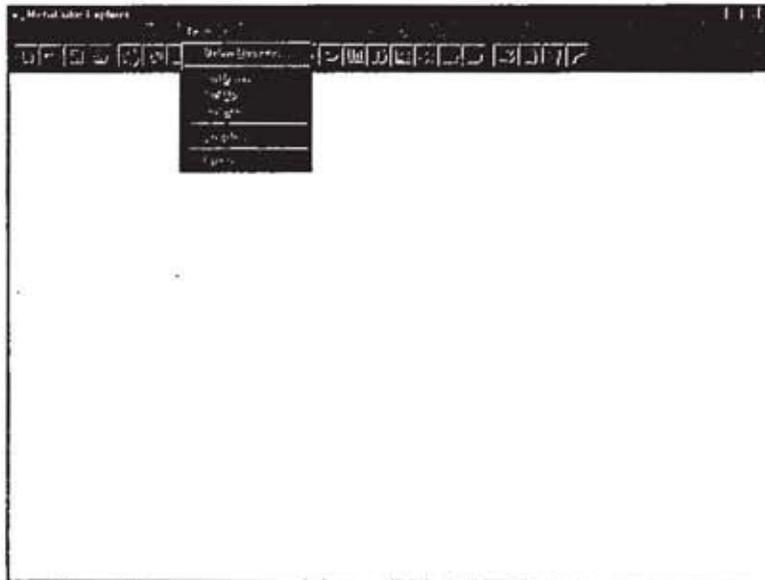


Figura D.28 Selección de "Options"

- Después de hacer los pasos del punto b), aparece la caja de diálogo "Preferences":



Figura D.29 Selección del modelo de datos

4.3 ESTABLECIENDO UNA CONFIGURACION PARA CONEXION

En la caja de diálogo "Preferencias" aparecen varios "tabs" para trabajar. El "tab Configuración" es donde se definen las características de conexión. Las funciones de los botones en este "tab" son:

- Botón Add:** Se usa para añadir una nueva configuración de conexión.
- Botón Edit:** Se usa para editar una configuración de conexión ya existente. Los nombres de las configuraciones existentes son listados en la caja de texto de la izquierda.
- Botón Delete:** Se usa para borrar una configuración existente.

Si en la caja de texto del lado izquierdo, no aparece ninguna configuración existente, debe dar clic en el botón **Add...** Si aparece la configuración **siglo**, selecciónela y de clic en el botón **Edit..**

Aparecerá la siguiente caja de diálogo:

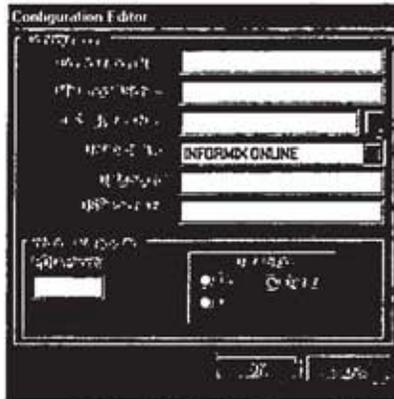


Figura D.30 Configuración para la conexión

En esta caja de diálogo, se pide información en los siguientes campos:

- Configuration Name:** cualquier nombre, probablemente desee usar el mismo nombre que el usado en el nombre del "Host Connection". Para la aplicación de MetaCube Explorer se debe especificar "siglo".
- Metamodel Schema:** el esquema/propietario de las tablas metadata de MetaCube. Aquí se debe capturar "MetaCube."
- DSS System Name:** nombre del sistema DSS para acceder a la base de datos. Capturar "MC-CICA".
- Database Type:** se especifica el tipo de la base de datos. Debe capturar "INFORMIX ONLINE".
- Data Source:** este dato debe ser igual al especificado en el "data source" del ODBC. Se puede consultar el administrador del ODBC para consultar este dato. Se debe capturar "siglo".
- DSS Cache File:** el nombre del archivo local de caché. El nombre por defecto para este archivo es "metacube.dss".

APENDICE D. MANUAL DE INSTALACION

PDQ Priority: para habilitar esta característica de Informix, capture un número entre -1 y 100. El número que se debe capturar es -1.

Data Skip: Hay 3 diferentes opciones que se pueden elegir:

On: habilita la característica "Data Skip" de Informix, permitiendo a las consultas terminar aún cuando los datos no estén disponibles.

Off: deshabilita la característica "Data Skip" de Informix.

Default: utiliza la configuración por defecto para "Data Skip" tal y como está configurado en el servidor de la base de datos. Esta es la opción que deberá elegir.

Al final de capturar todos los datos solicitados, la caja de diálogo debe quedar así:



Figura D.31 Editor de configuración

Después debe dar clic en el botón OK. Aparecerá la caja de diálogo "Preferences" y también debe dar clic en el botón OK de esa caja.

Finalmente, salga de la herramienta y vuelva a entrar para que la nueva configuración tome efecto.

APENDICE E. MANUAL DE USUARIO

La aplicación CICA, que está desarrollada con la herramienta INFORMIX-MetaCube Explorer, contempla 27 gráficas y 3 reportes preconstruídos como se explicó en el capítulo 4 "Implementación de la Solución".

MetaCube Explorer es una herramienta gráfica para acceso de datos, que facilita la rápida recuperación y análisis de datos de negocios, almacenados en una gran base de datos multi-dimensional. La interfaz gráfica de MetaCube Explorer despliega múltiples vistas (gráficas de barras, de pastel, de líneas, en dos y en tres dimensiones) de la información extraída por cualquier consulta (query).

El tipo de usuario al que está enfocado esta aplicación es a un analista de negocios, un usuario que tenga los conocimientos suficientes para saber que representan los datos y como interpretar la información que se obtenga, aparte de un manejo aceptable de la computadora y amplios conocimientos de diversas aplicaciones y sistemas operativos (EXCEL, Word, PowerPoint, WINDOWS95, entre otros).

El CICA mediante MetaCube Explorer le facilita al usuario hacer lo siguiente:

- Obtener resultados de consultas complejas
- Capacidad para obtención de información más detallada ó, si se desea, consolidar la información
- Exportar datos a otras aplicaciones (como EXCEL)
- Imprimir reportes y gráficas

Esta aplicación tiene definidas 27 gráficas y 3 reportes, que se han construido basándose en las especificaciones del departamento de Banca Electrónica para sus reportes estadísticos. Esto no es limitante para que se puedan crear nuevas gráficas y reportes. La persona que desee crear nuevas gráficas o reportes con ayuda de MetaCube Explorer debe tener un conocimiento del negocio (cajeros automáticos) aceptable. Esto es porque se debe saber como manejar los datos que se tienen, para elaborar una consulta congruente y obtener resultados útiles.

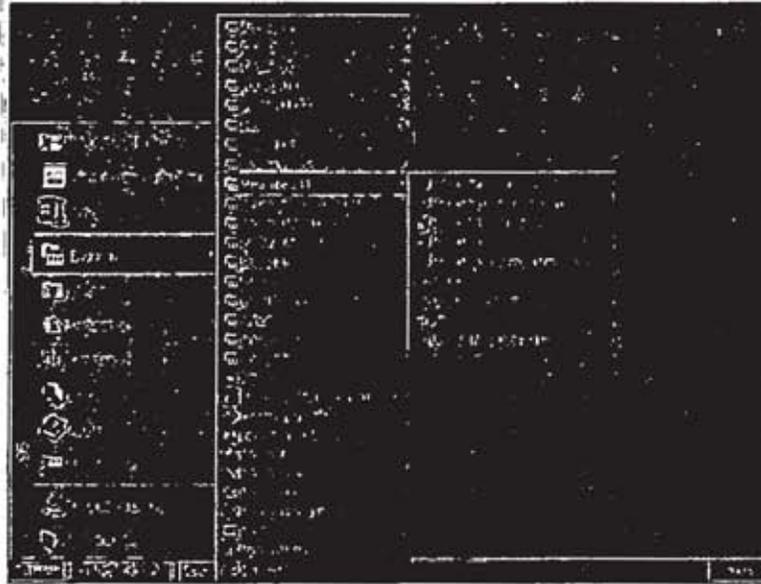
Se recomienda que después de la instalación de la aplicación, las gráficas y reportes definidos se graben en un directorio que las identifique, por ejemplo: C:\CICA.

Por ser una aplicación comprada por BanCrecer a una compañía extranjera de desarrollo de software (del tipo de MICROSOFT), casi todo el ambiente de la aplicación está en el idioma inglés (menú principal y sus opciones, ayudas, mensajes del sistema, etc.). Por ser el proveedor este tipo de compañía, no es posible realizar modificaciones a las características originales de la aplicación para tener una versión en idioma español.

Nota: si se desea conocer el manejo total y completo de todas y cada una de las opciones del esquema funcional de MetaCube Explorer, es recomendable acudir al manual "Explorer User's Guide".

1. ACCESO A METACUBE EXPLORER

Para acceder a MetaCube Explorer del Menú Inicio de Windows 9x se debe elegir Inicio, Programas, Metacube 3.11, Metacube Explorer o el icono de acceso directo como se muestra a continuación:

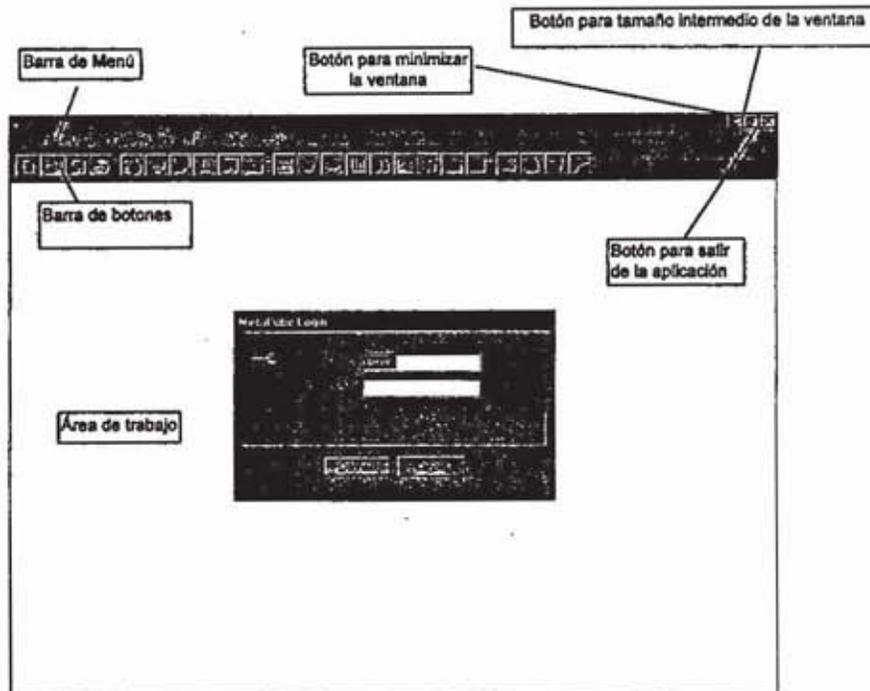


Forma de acceder a MetaCube Explorer.



Icono de acceso directo.

Al acceder a la aplicación METACUBE, aparece la siguiente pantalla:

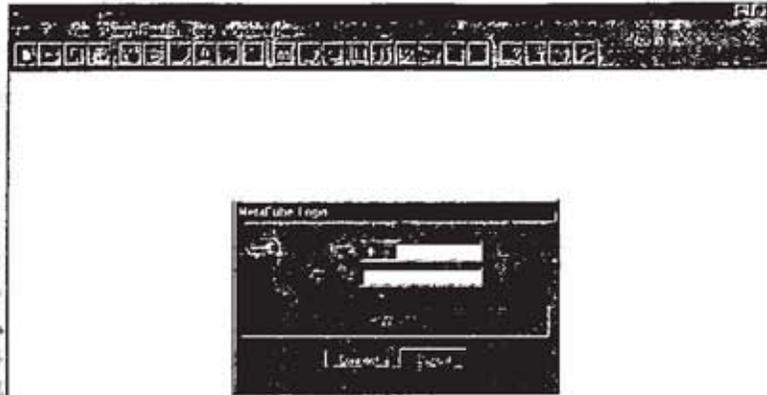


En esta pantalla se encuentra la caja de diálogo "MetaCube Login", donde aparecen los campos **User** y **Password**. Para poder hacer conexión con el servidor y empezar a trabajar con la aplicación, debe hacer lo siguiente:

- A) En el campo **User**, aparece por defecto la palabra: *dbsiglo*, para pasar al siguiente campo utilice la tecla del tabulador o mueva el apuntador del "mouse" y dé un clic en el espacio del campo **Password**.
- B) En el campo **Password** debe teclear: *siglo123* (al teclear en esta parte, sólo aparecerán asteriscos).
- C) Presionar <Enter> y espere a que se haga la conexión.

Nota: Si no desea hacer la conexión al modelo de datos por defecto, se debe usar el botón **Cancel** y mediante el menú **Tools** elegir otro modelo. El nombre del modelo para esta aplicación es *siglo*.

La pantalla que aparecerá al hacer la conexión con el servidor es la siguiente:



Cabe hacer mención que no todas las opciones de la barra del menú están disponibles al entrar a la aplicación. Por ejemplo, las opciones **E**dit, **Q**uery, **R**esults y **W**indow de la barra de menú, no están activadas recién inicia la aplicación.

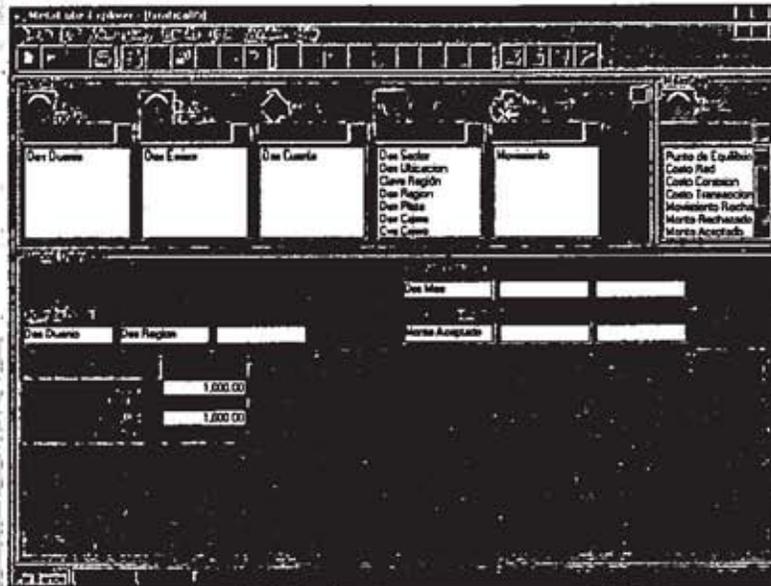
Si se elige el botón **Cancelar** en la caja de diálogo "MetaCube Login" no saldrá de la aplicación, pero no se hará conexión con el servidor donde se encuentran las bases de datos y, por consiguiente, no podrá abrir ninguna gráfica ni reporte. Si intenta abrir cualquiera de ellos, aparecerá el siguiente mensaje:



Este mensaje indica que no se puede abrir el archivo porque se está fuera de línea.

Nota: Para realizar la conexión se utiliza el menú File y la opción Connect mediante la que aparecerá una pantalla como la de la página anterior. A continuación se presenta un ejemplo de la pantalla de conexión:

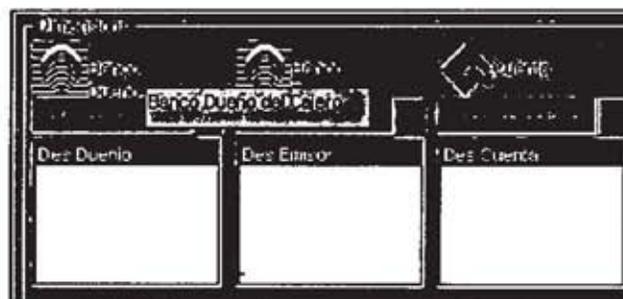
D) En la pantalla anterior, se seleccionó el workbook llamado "Grafica05" (se encuentra sombreada). Para poder trabajar con esta, debemos dar clic en el botón . La pantalla que aparece después de hacer esto es la siguiente:



Para conocer la manera de ejecutar esta consulta, puede pasar al punto 4. CREACION DE UNA CONSULTA (QUERY).

Si elige el botón  en la pantalla de la página anterior se cierra la caja de diálogo "Open Workbook" y regresa a la pantalla inicial.

Para obtener una descripción del significado de cada elemento de dimensión y de las métricas se debe utilizar el botón derecho del "mouse" al estar el apuntador en alguno de ellos, por ejemplo, para saber a que se refiere "Des Duenio" el resultado de presionar el botón derecho en el área correspondiente es un recuadro amarillo como en la siguiente pantalla:



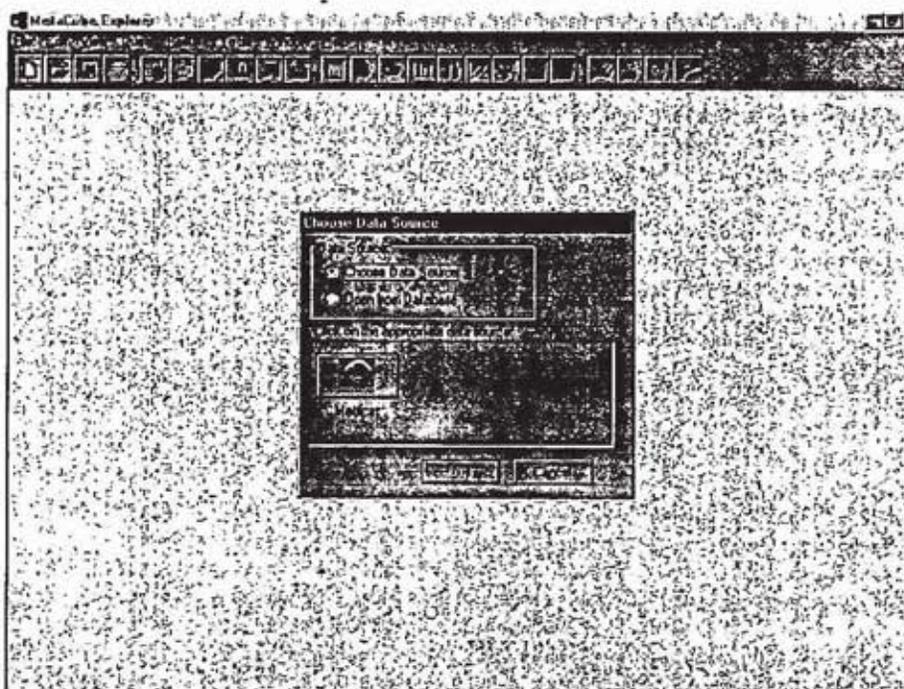
3. CREACION DE UN NUEVO LIBRO DE TRABAJO

El usuario que desee crear un nuevo libro de trabajo (*workbook*) debe tener un conocimiento del negocio (cajeros automáticos) aceptable. Esto es porque se debe saber como manejar los datos que se tienen, para elaborar una consulta congruente y obtener resultados útiles.

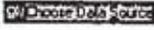
Para crear una nueva área de trabajo (*workbook*) se tienen dos opciones:

- A) Del menú principal, elegir la opción **File** y luego la opción **New...** ó
- B) De la barra estándar, elegir el icono 

Al hacer cualquiera de estos dos pasos, aparecerá la siguiente pantalla:

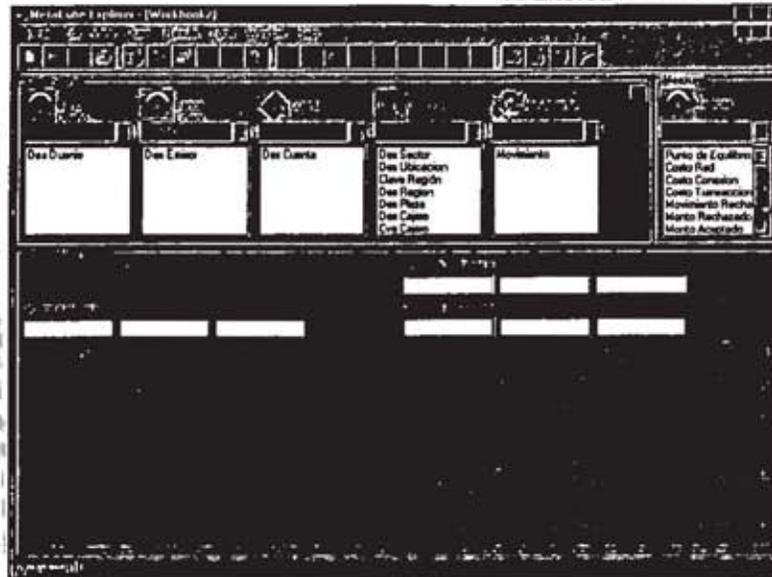


3.1. ELECCION DE UN MODELO DE DATOS

Para poder trabajar en una nueva área de trabajo, en la pantalla anterior, aparece la caja de diálogo "Choose Data Source", donde aparece elegida por defecto la opción **Choose Data Source** . Deberá dejar esa selección y dar *click* en el botón **Metrics**



Después de haber hecho clic en este botón, aparecerá la siguiente pantalla:

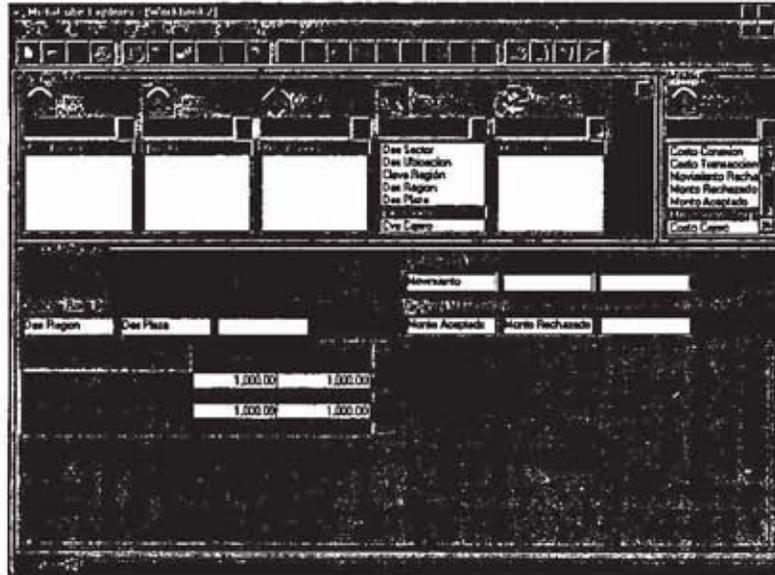


Esta pantalla es la nueva área de trabajo. Para saber cómo utilizarla para hacer una consulta pase al punto 4. CREACION DE UNA CONSULTA (QUERY).

4. CREACION DE UNA CONSULTA

El usuario que desee crear una nueva consulta debe tener un conocimiento del negocio (cajeros automáticos) aceptable. Esto es porque se debe saber como manejar los datos que se tienen, para elaborar una consulta congruente y obtener resultados útiles.

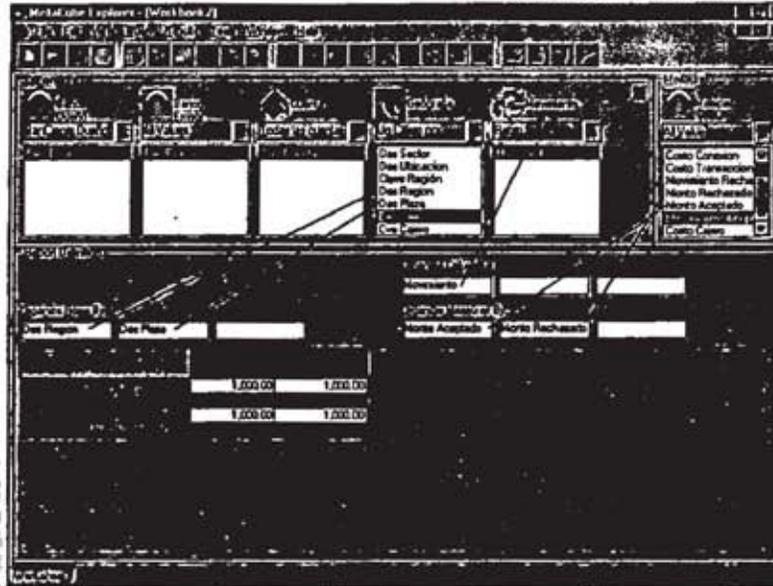
Para conocer la manera de formar una consulta (query), veamos un ejemplo, considerando la siguiente pantalla:



El área de trabajo en pantalla consta de tres divisiones: **Dimensions**, **Measures** y **Report Definition**.

- A) En el área de **Dimensions** se encuentran las "dimensiones" (tablas o archivos, junto con sus campos) del modelo de datos.
- B) En el área de **Measures** se encuentran los campos sobre los cuales se pueden realizar cálculos (sumatorias, promedios, porcentajes, etc.).
- C) En el área de **Report Definition** es donde se definen los "renglones" (rows), "columnas" (columns) y las "métricas" (measures) de la consulta que queremos realizar.

En la pantalla que se presenta enseguida observamos el origen de los datos del área de **Report Definition**.



Es conveniente notar lo siguiente:

- A) Los apartados **Organize Rows By** y **Organize Columns By** , toman sus datos sólo del área de **Dimensions**.
- B) El apartado **Organize Measures By** , toma sus datos solamente del área **Measures**.
- C) En la parte inferior del área de trabajo, se puede ver un ejemplo de los resultados que se obtendrán, después de ejecutar la consulta.

4.1 COMO LLENAR LOS ESPACIOS EN EL AREA DE *REPORT DEFINITION*

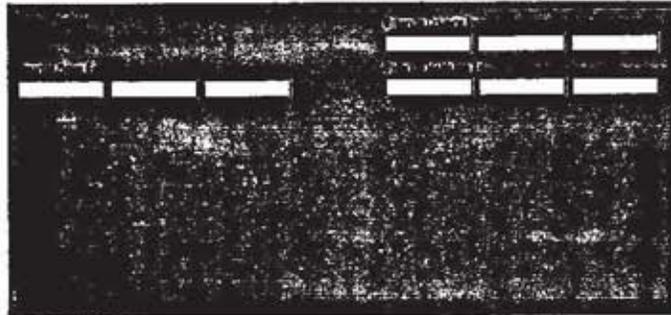
En la sección **Report Definition** en cualquiera de los tres apartados que están ahí (**Rows**, **Columns** y **Measures**) los datos los tiene que colocar el usuario. Por ejemplo: Para colocar "Des Region" en el apartado **Organize Rows By** , tiene que posicionarse en la dimensión "Geografía", luego poner el señalador del "mouse" sobre el campo "Des Region", manteniendo el botón derecho del "mouse" presionado sobre ese campo, aparece el icono



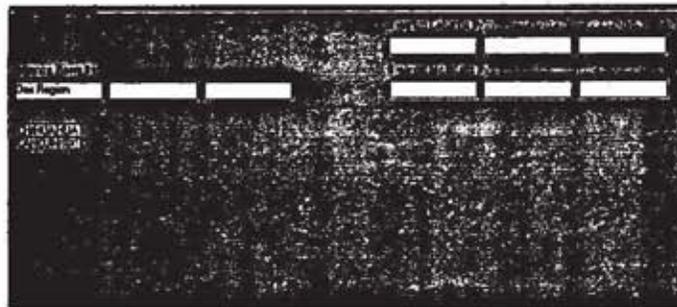
Hay que arrastrar el "mouse" (siempre con el botón derecho presionado) hasta donde se encuentra el rectángulo en blanco del apartado (**Rows**, **Columns** o **Measures**) donde

queremos poner ese campo. Al llegar al espacio elegido, aparecerá el icono  . En ese momento, debe dejar de presionar el botón derecho del "mouse" y aparecerá el campo elegido en el espacio indicado.

Esta pantalla presenta la sección **Report Definition** antes de llenar los espacios de los apartados.



La pantalla siguiente presenta la sección **Report Definition** después de poner el campo "Des Region" en el apartado **Organize Rows By**.



4.2 UTILIZACION DE FILTROS

Después de haber elegido los campos que queremos que aparezcan en nuestra consulta, el siguiente paso es establecer criterios o condiciones para dicha consulta. Estos criterios se conocen como **filtros**.



Arriba tenemos el área **Dimensions** completa. Para poder ver la totalidad de las "dimensiones" debe dar clic en el icono  y así aparecerán las dimensiones restantes (la dimensión "Tiempo" y "Razón de Respuesta", en este caso). Si quiere ver de nuevo las otras dimensiones debe dar clic en el icono .

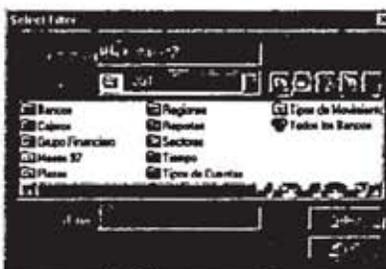
4.2.1 FILTROS FIJOS

En la dimensión "Banco Dueño" existe un **Filtro Fijo**, denominado así por una condición que se establece a criterio del usuario y la cual deberá cumplirse siempre durante cualquier ejecución de la consulta, para obtener una información determinada.

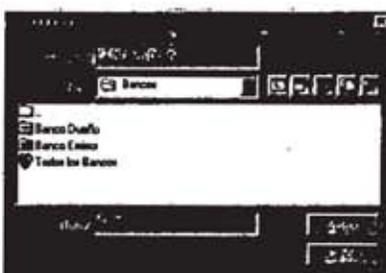
El **filtro fijo**, en este ejemplo, es **Banco Dueño = BanCreceer Dueño**. Con esto establecemos que toda la información que se extraiga, durante la ejecución de la consulta, sea donde se cumpla la condición de que el banco dueño del cajero automático sea BanCreceer. Esta condición permanecerá fija cuantas veces ejecutemos la consulta.

La manera de establecer este filtro es la siguiente:

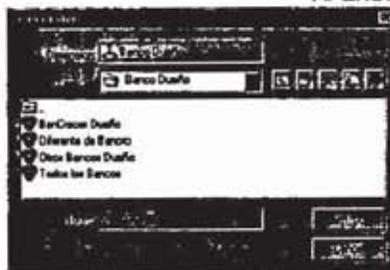
- a) Al dar clic en el botón  de la dimensión "Banco Dueño" aparecerá la pantalla que se muestra enseguida. De esta ventana ("Select Filter"), se debe elegir la carpeta  y dar doble clic. Después de hacer esto, aparecerá la pantalla que se muestra enseguida.



- b) De la ventana que se presenta a continuación ("Select Filter"), elegir la carpeta  y dar doble clic. Después de hacer esto, aparecerá la pantalla que se muestra enseguida.



- c) Este es el paso final y lo único que tiene que hacer es elegir el icono  BanCreceer Dueño y después dar clic en el botón . Al finalizar este paso, aparecerá el filtro fijo que ha establecido en la dimensión "Banco Dueño".



D) Para establecer un **filtro fijo** en cualquiera de las otras dimensiones del área de **Dimensiones** tiene que seguir los mismo pasos aquí descritos, sólo asegurándose de elegir la carpeta adecuada para cada dimensión.

4.2.2 FILTROS PARAMETRIZADOS

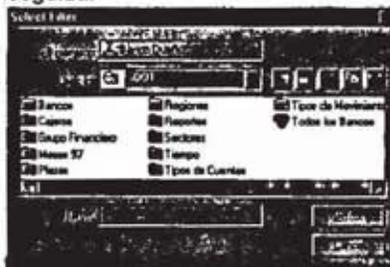
En la dimensión "Geografía" existe un **Filtro Parametrizado**, llamado así porque en el momento de "lanzar" a ejecución una consulta, nos pide un dato que sirve como condición que la consulta debe cumplir para extraer información.

El **filtro parametrizado**, en este ejemplo, es Geografía = Una Región por Nombre.

La manera de establecer un **filtro parametrizado** es igual a la vista para establecer un **filtro fijo**, sólo hay que considerar la dimensión con la que estamos trabajando y las carpetas que le correspondan a estas, para poder hacer una correcta parametrización.

A continuación vamos a ver todos los pasos que se requieren para establecer un **filtro parametrizado** a la dimensión "Geografía".

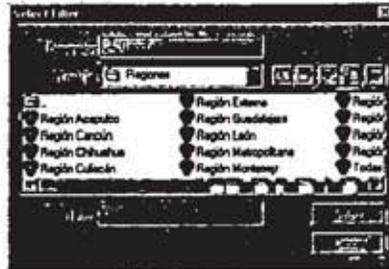
A) Dar clic en el botón  de la dimensión "Geografía". Al hacer esto, aparecerá la pantalla que se muestra enseguida.



B) De esta ventana ("Select Filter"), elegir la carpeta  **Regiones** y dar doble clic . Después de hacer esto, aparecerá la pantalla que se muestra enseguida.

C) Para elegir el **filtro parametrizado** debemos de recorrer la pantalla hacia la derecha, para hacer esto presione el icono  de la barra de desplazamiento y continúe presionándolo hasta que llegue al final de los datos. Ahí podrá ver y deberá seleccionar

la definición de filtro  Una Región por Nombre y después dar clic en el botón . Al finalizar este paso, aparecerá el filtro parametrizado que ha establecido en la dimensión "Geografía".



D) Para conocer la manera en que trabaja un filtro parametrizado, puede pasar a la sección 5.1. EJECUCION DE CONSULTAS CON FILTROS PARAMETRIZADOS.

5. EJECUCION DE UNA CONSULTA

Una vez que ha terminado de crear la consulta, necesita ejecutarla para obtener la información deseada. Hay dos maneras de ejecutar una consulta: una forma de ejecución en línea y otra por Query Back.

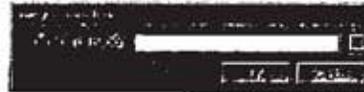
- A) Ejecución en línea. Esta forma de ejecución es en la cual "lanzamos" la consulta y forzosamente, tenemos que esperar a que termine la consulta y no despliegue los resultados, para poder hacer otra cosa en la PC.
- B) Ejecución en segundo plano (Query Back) sin tener que esperar a que termine la consulta, para poder hacer otra cosa en la PC.

Para cualquiera de los dos tipos de ejecución que se desee "lanzar", se deberá de dar clic en el icono  de la barra estándar y posteriormente definir el tipo de ejecución que queremos, tal y como se ve en el punto 5.2. EJECUCION EN LINEA DE UNA CONSULTA

5.1 EJECUCION DE CONSULTAS CON FILTROS PARAMETRIZADOS

Al ejecutar una consulta que en su definición contenga uno o más filtros parametrizados, se tendrá que proceder de la siguiente manera:

- A) Al ejecutar la consulta, de cualquiera de las dos formas descritas en la parte de arriba, aparecerá la caja de diálogo "Query Parameters". Esta caja nos indica el dato que se tiene que introducir (Nombre de Región, en este caso)
- B) Si lo desea, puede posicionarse en el área



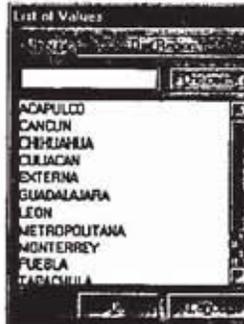
destinada para captura y puede teclear el dato que corresponda. Para que lo capturado sea aceptado presione el botón .

C) Otra manera de dar el dato que solicita el filtro, es hacer clic sobre el botón  de la caja de diálogo "Query Parameters". Después de hacer esto, aparecerá la ventana que se muestra abajo.

D) Deberá hacer clic sobre el botón  en la ventana que aparece a la derecha y aparecerá la misma caja de diálogo, pero ya con todos los datos referentes al atributo (en este ejemplo, la Descripción de todas las Regiones), tal y como se puede ver en la siguiente pantalla.

E) De la lista de nombres que aparecen en la caja de diálogo, puede escoger la que desee o puede usar los botones   (si es que la lista abarca más del área visible de la ventana) para desplazarse hacia arriba o abajo, hasta localizar la región deseada. Tendrá que seleccionar el dato y hacer doble clic o seleccionarlo y presionar el botón .

Nota: Para algunos atributos como el nombre de los cajeros, por ser una lista muy larga, no es posible para la herramienta desplegar todos los valores existentes por lo que se debe utilizar la letra inicial del nombre buscado, oprimir  y aparecerán los nombres cuya inicial sea la proporcionada, a la derecha se muestra un ejemplo de esto.



APENDICE E. MANUAL DE USUARIO

F) Después de haber realizado el paso E), aparecerá en la caja de diálogo "Query Parameters" la región escogida, tal y como se muestra en la figura de la derecha. Para confirmar que todo está correcto, presione el botón .



G) Para cualquier filtro parametrizado que se haga sobre cualquier dimensión, se procederá de la misma forma.

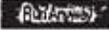
H) También puede capturar los datos de manera escrita. Tiene que empezar poniendo comilla simple ('), después escribir el nombre de la regional con mayúsculas (ACAPULCO, CANCUN, CHIHUAHUA, etc.) y cerrar con comilla simple (') nuevamente. Si la captura es el nombre del mes y año, se debe teclear comilla simple ('), la primer letra del nombre del mes en mayúscula seguida de las dos siguientes letras en minúsculas, luego un espacio en blanco, el año con 4 dígitos , un espacio en blanco y para finalizar una comilla simple ('). El dato debe quedar como se muestra en el cuadro de diálogo "Query Parameters" de la derecha. Para confirmar que todo está correcto, presione el botón .



5.2 EJECUCION EN LINEA DE UNA CONSULTA

Como se vio al inicio del punto 5, para "lanzar" la ejecución de cualquier consulta, se deberá de dar clic en el icono  de la barra estándar y posteriormente definir el tipo de ejecución que queremos, tal y como lo veremos a continuación.

Al ejecutar la consulta, aparece la pantalla del lado derecho.

Para ejecutar la consulta en línea, debe presionar el botón  y esperar a que se presenten los resultados.



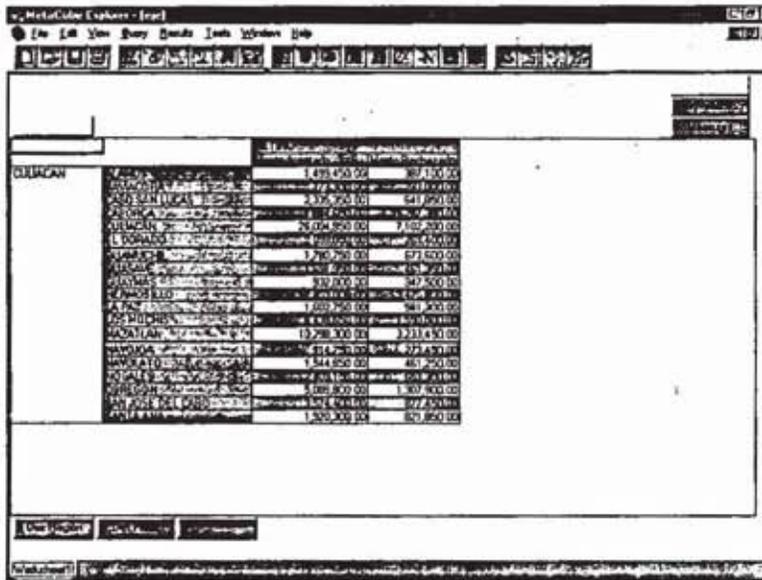
Al momento de estar ejecutándose la consulta, se mostrará la ventana del lado derecho.



La pantalla que se despliega, una vez que se obtienen los resultados, es la siguiente.

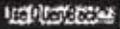
En caso de que existan más columnas de las que se pueden ver en pantalla, con un clic dentro del área de la tabla se activa la barra de desplazamiento para visualizar esas otras columnas.

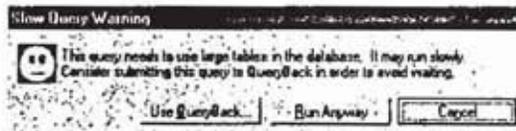
Para el caso de las gráficas no aplica el concepto de barra de desplazamiento por definición de la herramienta.



5.3 EJECUCION DE UNA CONSULTA EN SEGUNDO PLANO

Como se vio al inicio del punto 5, para "lanzar" la ejecución de cualquier consulta, se deberá de dar clic en el icono  de la barra estándar y posteriormente definir el tipo de ejecución que queremos, tal y como lo veremos a continuación.

- A) Al ejecutar la consulta, aparece la pantalla del lado derecho.
- B) Para ejecutar la consulta en el servidor sin dejar "atrapada" la PC cliente, debe presionar el botón . Después



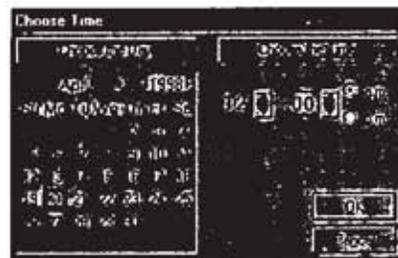
de hacer esto, aparecerá la siguiente ventana.

C) Deberá de dar un nombre a la consulta en el campo "Job Name", ejemplos de este nombre pueden ser: *tarea1*, *job-1*, *work01*, etc., el propósito de este nombre es para facilitar el monitoreo del estatus de la consulta.



D) La caja de diálogo "When to Run" es para especificar el momento de ejecución de la consulta. Para correr inmediatamente la consulta la opción por defecto es **As soon as possible**.

También se puede seleccionar la opción **Specific day/time**. Esta opción nos permite programar la consulta para que sea ejecutada el día y la hora que nosotros especifiquemos. Para poder acceder la caja de diálogo "Choose Time" debemos dar clic en el botón **Choose Time**. La pantalla que se despliega al elegir este botón es la siguiente.



E) Para modificar el mes o el año, tiene que dar clic en los iconos **←** **→** de las columnas de mes y año.

F) Para escoger la hora, tiene que dar clic en el icono **↓** hasta que aparezca la hora deseada. También existe la opción para indicar si es la mañana (a.m.) o en la tarde (p.m.):

Después de escoger el horario de ejecución, para confirmar que todo está correcto, presione el botón **OK** y regresará a la pantalla anterior (caja de diálogo "Submission Info").

G) En la caja de diálogo "Submission Info", existen otras dos opciones que se pueden especificar: **Priority** y **Frecuency**. Se pueden elegir diferentes niveles de prioridad (**Priority**) usando las flechas ubicadas en esa opción. La prioridad 3 es por defecto, 1 es la prioridad más baja y 5 es la más alta.

Se puede usar la opción **Frecuency** para ejecutar una (once) o varias veces (daily, weekly, monthly o annually) la consulta.

APENDICE E. MANUAL DE USUARIO

Después de especificar todas las opciones, para confirmar que todo está correcto, presione el botón **OK** y aparecerá el mensaje de la derecha (en donde tiene que teclear <Enter> o dar clic en el botón **Aceptar**).

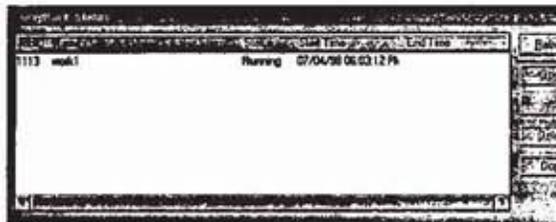
Esto significa que la consulta (query) entró al proceso de ejecución (no necesariamente está ejecutándose).



H) Para monitorear el estatus de la consulta, se debe de elegir la opción **View** del menú principal, dentro de este menú elegir la opción **QueryBack Status** y dentro de este menú, la opción **Jobs for this Worksheet...**

Al dar clic en esta última opción, aparece la pantalla de la derecha.

Se puede usar el botón **Refresh** para que se vaya actualizando el estatus del proceso.



Nota: Pueden existir procesos (jobs) con estatus **Pending** o **Error** por lo que se debe oprimir el botón **Info** para conocer los detalles de dicho estatus.

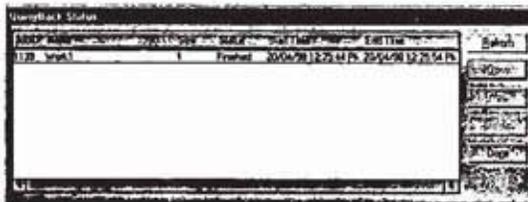
I) Al finalizar la ejecución de la consulta, se desplegará la siguiente pantalla:



J) Para ver el resultado de la consulta, se debe de elegir la opción **View** del menú principal, dentro de este menú elegir la opción **QueryBack Status** y dentro de este menú, la opción **Jobs for this Worksheet...**

Al dar clic en esta última opción, aparece la pantalla de la derecha.

Debemos dar clic en la línea de estatus, para que los botones se activen. Al hacer esto la caja de diálogo "QueryBack Status" se mostrará como sigue:



APENDICE E. MANUAL DE USUARIO

En esta parte debemos dar clic en el botón  y esperar a que se muestren en la pantalla los resultados de la consulta. La pantalla es la que se muestra abajo.



En caso de que existan más columnas de las que se pueden ver en pantalla, con un clic dentro del área de la tabla, se activa la barra de desplazamiento para visualizar esas otras columnas.

Para el caso de las gráficas no aplica el concepto de barra de desplazamiento por definición de la herramienta.

Municipio	REVENIDO	REVENIDO
AVANZO	1,459,450.00	307,100.00
AVANZORA	2,723,000.00	464,000.00
CASO SANTIPALM	2,325,250.00	641,050.00
CELESTACION	2,014,500.00	406,240.00
CELICAN	25,034,550.00	7,102,200.00
EL DORADO	3,023,650.00	6,324,400.00
GUAMUCHIL	1,780,750.00	673,600.00
GUAYMAS	1,828,200.00	3,307,240.00
GUAYMAS	937,000.00	347,500.00
HERMOSILLO	1,127,640,000.00	1,148,594.00
LA PAZ	1,602,250.00	500,300.00
LOS MOCHIS	2,238,350.00	1,000,000.00
MAZATLAN	10,258,300.00	3,230,450.00
NAVJOYA	1,016,250.00	222,420.00
NAVOLATO	1,944,950.00	481,250.00
NOBALES	2,601,150.00	6,440,000.00
OREGUN	5,029,800.00	1,307,300.00
SAN JOSE DEL CABO	3,474,000.00	2,774,250.00
SANLUIS	1,920,300.00	801,850.00

1



Tesis y Encuadernaciones

Martínez

Tesis, Libros, Folletos

AMATISTA No. 88 COL. ESTRELLA

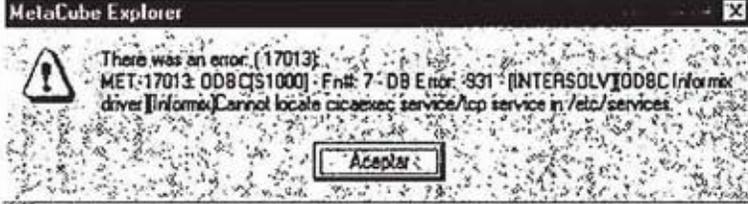
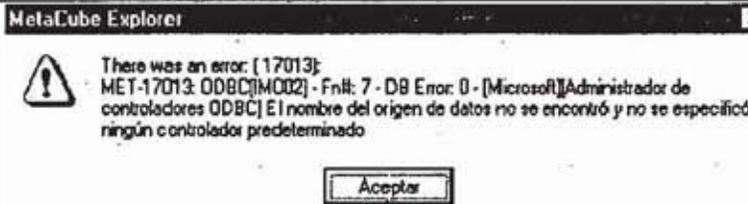
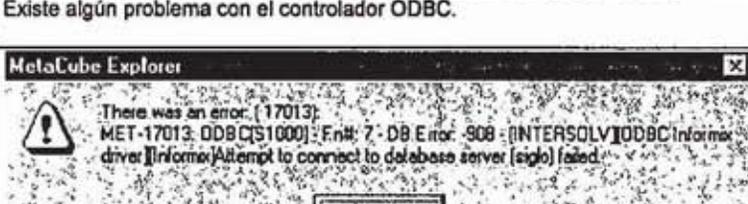
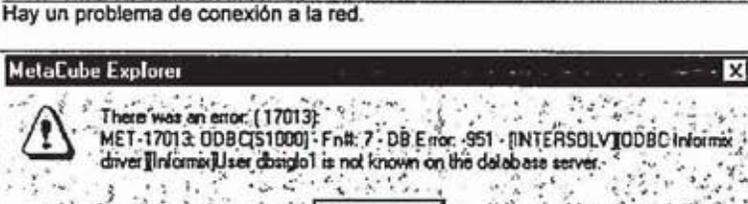
MEXICO, D.F. C.P. 07810

TEL. 55-77-61-42

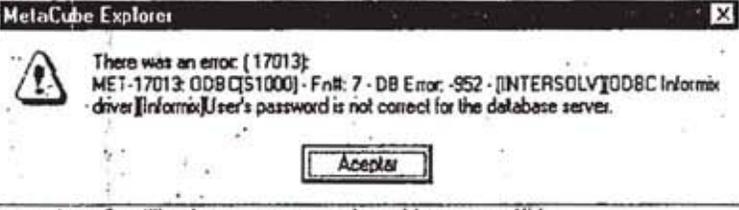
APENDICE F. MANUAL DE ERRORES Y SOLUCIONES

Este documento muestra las situaciones en las que se presentan los mensajes de errores básicos al utilizar MetaCube Explorer de acuerdo al manual "Explorer User's Guide" de Informix.

Es importante mencionar que su publicación está restringida a los usuarios autorizados ya que contiene información que sirve para acceder a los datos del modelo del CICA.

SITUACION: AL INICIAR METACUBE EXPLORER	
ERROR	DESCRIPCION / SOLUCION
 <p>No se pudo localizar el servicio "cicaexec" en el archivo "SERVICES".</p>	<p>Definir en el archivo SERVICES que se encuentra en C:\WINDOWS el servicio "cicaexec" de la siguiente manera: cicaexec 1529/tcp</p>
 <p>Existe algún problema con el controlador ODBC.</p>	<p>Instalar o verificar con ayuda de la aplicación <u>ODBC Administrador 32</u>, la configuración del DSN del usuario "siglo".</p>
 <p>Hay un problema de conexión a la red.</p>	<p>Puede haber varios motivos:</p> <ul style="list-style-type: none"> - El cable de la red de la PC está desconectado ó - El servidor esta "caldo" ó - No hay servicio de red.
 <p>El usuario usado para acceder el servidor no es válido.</p>	<p>El usuario correcto es "dbsiglo" para poder acceder al servidor "siglo". Si está seguro de haber tecleado correctamente el dato, debe verificar que el usuario no haya sido cambiado, en ese caso, consulte al administrador de la base de datos.</p>

APENDICE F. MANUAL DE ERRORES Y SOLUCIONES

SITUACION: AL INICIAR METACUBE EXPLORER	
ERROR	DESCRIPCION / SOLUCION
 <p>There was an error (17013): MET-17013: ODB [S1000] - Fmt: 7 - DB Error: -952 - [INTERSOLV]ODBC Informix driver [informix]User's password is not correct for the database server.</p> <p>Aceptar</p>	<p>La contraseña correcta es "siglo123" para el usuario "dbsiglo". Si está seguro de haber tecleado correctamente el dato, debe verificar que la contraseña no haya sido cambiada, en ese caso, consulte al administrador de la base de datos.</p>
<p>ERROR: "Table or view does not exist"</p> <p>DESCRIPCION / SOLUCION: Puede aparecer este error por una de las siguientes razones: a) MetaCube Explorer puede estar buscando tablas en los esquemas equivocados. Puede seleccionar nombres de esquemas en la caja de diálogo "Preferences". b) Su cuenta de usuario puede no tener permisos para acceder las tablas, en ese caso debe pedir al administrador de la base de datos que le establezca los permisos apropiados. c) La tabla puede no existir. Debe consultar al administrador de la base de datos.</p>	

SITUACION: AL EJECUTAR UNA CONSULTA	
ERROR	DESCRIPCION / SOLUCION
 <p>There was an error (17010): MET-17010: ODBC [S1000] - Fmt: 7 - DB Error: -952 - [INTERSOLV]ODBC Informix driver [informix]No data for the requested query.</p> <p>Aceptar</p>	<p>Se debe analizar la estructura de la consulta realizada para encontrar algún problema que cause el que no se pueda obtener datos. Es muy probable que en la fecha especificada para la consulta, no exista información.</p>
 <p>There was an error (17011): MET-17011: ODBC [S1000] - Fmt: 7 - DB Error: -952 - [INTERSOLV]ODBC Informix driver [informix]No attributes specified for the query.</p> <p>Aceptar</p>	<p>Se debe especificar los atributos (rangiones, columnas y métricas) necesarios para poder realizar una consulta.</p>

APENDICE F. MANUAL DE ERRORES Y SOLUCIONES

SITUACIÓN: AL EJECUTAR UNA CONSULTA	
ERROR	DESCRIPCIÓN / SOLUCIÓN
 <p>No hay autorización para usar un filtro o el filtro no existe.</p>	<p>Debe revisar con el administrador del DataWarehouse, si el usuario con el que está trabajando, tiene autorización de usar los filtros deseados, o una explicación de los filtros existentes.</p>
 <p>No se encontró impresora en la cual imprimir.</p>	<p>Para corregir este problema, existen varias cosas que se pueden hacer:</p> <ul style="list-style-type: none"> - Si no tenemos impresora definida, agregar una impresora y establecerla como predeterminada. - Verificar que la impresora esté encendida y/o en línea.
SITUACIÓN: AL EJECUTAR UNA CONSULTA MEDIANTE "QUERYBACK"	
<p>ERROR: You are not authorized to use background query functionality. Please contact the system administrator.</p>	
<p>DESCRIPCIÓN / SOLUCIÓN: No está registrado como un usuario de "Queryback", así que no puede enviar consultas por "Queryback". Debe consultar al administrador del DataWarehouse para obtener privilegios de "Queryback".</p>	
<p>ERROR: Parameter Not Found</p>	
<p>DESCRIPCIÓN / SOLUCIÓN: La función "Queryback" usa parámetros para calcular periodos cambiantes de tiempo. Por ejemplo, si un filtro es "semana actual", se usa un parámetro para calcular la "semana actual" cuando la consulta está ejecutándose. Este error generalmente significa que algunos o todos los parámetros que se necesitan no están configurados. Consulte a su administrador de DataWarehouse para crear los parámetros necesarios.</p>	

APENDICE G. MANUAL TECNICO

En este apéndice se muestra el manual técnico del CICA.

1. PROPOSITO DEL CLIENTE

El propósito de las PCs cliente es que los usuarios puedan conectarse, a través de MetaCube, al modelo de datos del CICA (que utiliza las bases de datos bdcica y mc_cica), para realizar las consultas de información que necesita.

Para el uso de esta aplicación es necesario que cada usuario tenga una computadora personal (PC) para poder instalarla y utilizarla.

2. REQUERIMIENTOS DE HARDWARE Y SOFTWARE

2.1 HARDWARE

La aplicación requiere la siguiente configuración de hardware:

- PC con procesador 486 o superior.
- Un drive CD-ROM
- Mouse soportado por Microsoft Windows.
- De 9 a 16 MB de espacio en disco, dependiendo de los componentes de MetaCube a instalar.
- 12 MB en RAM como mínimo.

2.2 SOFTWARE

A continuación se presenta una tabla que describe el software que se instalará en cada máquina cliente:

Nombre	Versión	Directorio	Espacio en disco
Windows	85 o superior	C:\WINDOWS	Depende de los componentes a instalar
Informix-MetaCube	3.11 (32 bits)	C:\METACUBE	Depende de los componentes a instalar
Informix-CLI	2.5	C:\INFORMIX	5 MB

3. ESQUEMA FUNCIONAL

Una de las características de MetaCube Explorer es que después de haber seleccionado una fuente de datos, tiene dos módulos de trabajo, que son: Query Mode y Results Mode. A continuación se describe cada uno.

3.1 QUERY MODE

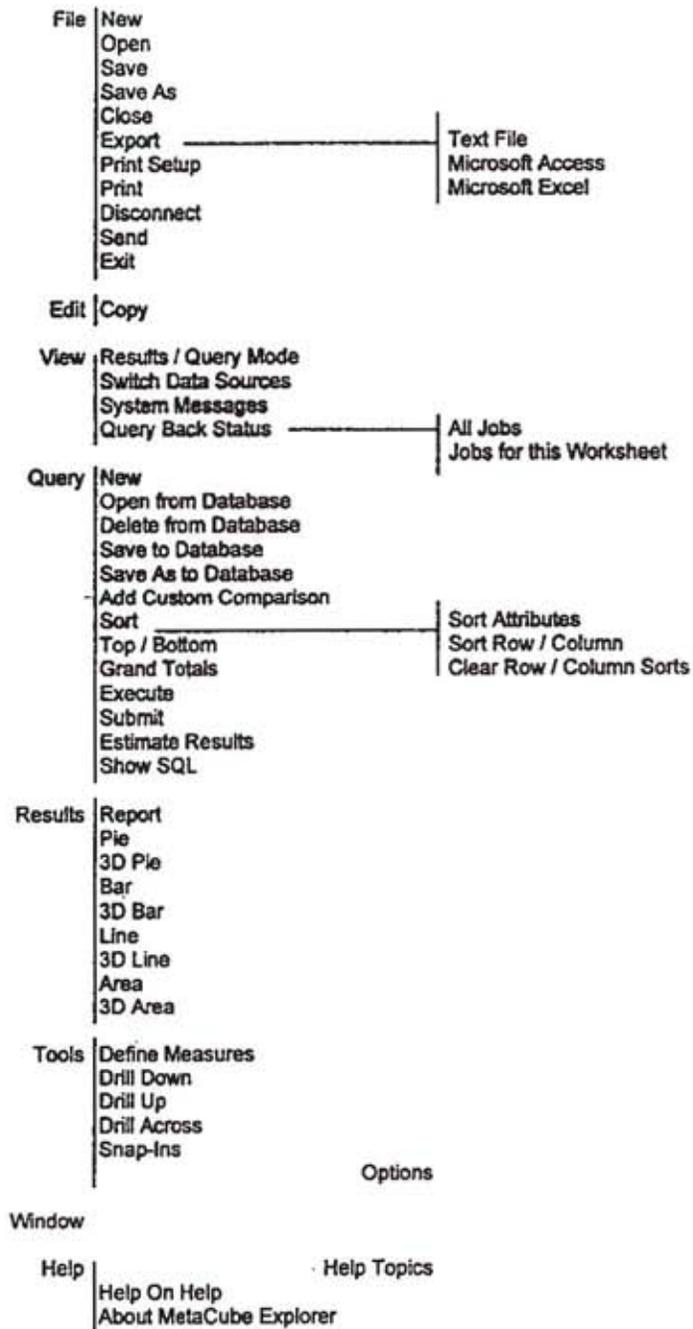
En modo de consultas o Query Mode se pueden armar las consultas que ejecutará el motor de MetaCube. En este modo la interfaz de MetaCube provee las funciones necesarias para formular, ejecutar y –en su caso– almacenar las consultas. Cuando se inicia MetaCube, la hoja de trabajo que aparece ("Worksheet") se encuentra en Query Mode.

3.2 RESULTS MODE

En el modo de resultados o Results Mode se presentan los resultados de las consultas realizadas en Query Mode. Cuando se ejecuta la consulta, MetaCube cambia la hoja de trabajo a Query Mode a Results Mode y despliega el resultado en un reporte o una gráfica. Por defecto MetaCube despliega los resultados como un reporte. El modo de resultados permite:

- Agregar formato adicional para despliegue del reporte o gráfica.
- Obtener datos adicionales que no se solicitaron en la consulta original.

A continuación se presenta un diagrama de las funciones que realiza MetaCube Explorer en modo de consulta y modo de resultados.



3.3 DESCRIPCION DE FUNCIONES

A continuación se explican las funciones presentadas en la sección anterior.

File	New	Crea un nuevo archivo de trabajo.
	Open	Abre un archivo de trabajo.
	Save	Guarda el archivo de trabajo actual.
	Save As	Guarda el archivo de trabajo actual con otro nombre.
	Export	Exporta los datos en un reporte, con tres opciones: <ul style="list-style-type: none"> • Como un archivo de texto. • A Microsoft Access. • A Microsoft Excel.
	Print Setup	Establece las opciones de configuración para la impresión.
Edit	Print	Imprime el reporte o la gráfica actual.
	Connect / Disconnect	Conecta o desconecta la base de datos que contiene el DSS.
	Send	Envía por correo un archivo de trabajo.
	Copy	Copia la selección actual de texto al portapapeles.
View	Results / Query Mode	Cambia entre los modos de consulta y resultados.
	Switch Data Sources	Cambia a otro DSS desde el actual.
	System Messages	Revisa los mensajes enviados por el administrador desde el Warehouse Manager.
	Query Back Status	Despliega el resultado de las consultas enviadas mediante Query Back.
Query	Show Report Info	Despliega detalles del reporte en modo de resultados.
	New	Crea una hoja de trabajo para realizar una nueva consulta.
	Open from Database	Abre, para ejecutarla, una consulta almacenada en la base de datos.
	Delete from Database	Borra una consulta almacenada en la base de datos.
	Save to Database	Almacena una consulta en la base de datos.
	Save As to Database	Almacena una consulta en la base de datos con un nuevo nombre.
	Add Custom Comparison	Crea una consulta para comparar atributos de una o entre dimensiones.
	Sort	Establece criterios de ordenamiento para el despliegue de resultados.
	Top / Bottom N	Selecciona los "N" primeros valores del resultado de una consulta para presentarlos.
	Grand Totals	Incluye cálculos para totalizar por renglones y columnas.
	Execute	Ejecuta una consulta en forma directa, o sea en línea.
Submit	Ejecuta una consulta que no "atrapa" a la máquina cliente, ya que la envía por "Query Back".	
Results	Estimate Results	Ejecuta consultas sobre tablas pequeñas para obtener resultados más rápidamente.
	Show SQL	Muestra en pantalla la orden SQL MetaCube Explorer construye al armar una consulta.
	Report	Crea un reporte para desplegar los resultados de una consulta.
Results	Pie	Crea una gráfica de pastel para desplegar los resultados de una consulta.
	3D Pie	Crea una gráfica de pastel tridimensional para desplegar los resultados de una consulta.
	Bar	Crea una gráfica de barras para desplegar los resultados de una consulta.
	3D Bar	Crea una gráfica de barras tridimensional para desplegar los resultados de una consulta.

APENDICE G. MANUAL TECNICO

	Line	Crea una gráfica lineal para desplegar los resultados de una consulta.
	3D Line	Crea una gráfica lineal tridimensional para desplegar los resultados de una consulta.
	Area	Crea una gráfica de área para desplegar los resultados de una consulta.
	3D Area	Crea una gráfica de área tridimensional para desplegar los resultados de una consulta.
Tools	Define Measures	Define una métrica calculada en función a las ya existentes.
	Drill Down	Ejecuta una nueva consulta que contiene mayor detalle de los datos.
	Drill Up	Ejecuta una nueva consulta que contiene menor detalle de los datos.
	Drill Across	Ejecuta una nueva consulta con atributos adicionales de otras dimensiones.
	Snap Ins	Añade, quita o temporalmente deshabilita la construcción propia de funciones.
	Options	Configura a MetaCube Explorer para conectarse al DSS.
Window Help		Lista los archivos de trabajo abiertos para abrir alguno.
	Help Topics	Proporciona ayuda en línea acerca de MetaCube Explorer
	Help On Help	Capacita al usuario sobre como usar la ayuda en línea.
	About MetaCube Explorer	Despliega información acerca de la versión de MetaCube Explorer.

4. DESCRIPCION DE BASE DE DATOS (TABLAS E INDICES)

A continuación se presentan las tablas e índices más representativos del CICA.

4.1 DESCRIPCION DE TABLAS DE LA BASE BDCICA

Nombre de Campo	Descripción	Tipo	Nulos
razon_id_acum	Identificar la razón de respuesta del cajero.	char(2)	SI
banco_id_acum	Clave de banco dueño del cajero.	char(4)	SI
region_id_acum	Clave de región.	char(4)	SI
plaza_id_acum	Clave de plaza.	char(4)	SI
sucursal_id_acum	Clave de sucursal.	char(4)	SI
cajero_id_acum	Clave de cajero.	char(16)	SI
ubicacion_id_acum	Clave de ubicación.	smallint	SI
sector_id_acum	Clave de sector.	char(1)	SI
fecha_corte	Fecha del corte.	date	SI
banco_tarj_id_acum	Clave del banco emisor de la tarjeta.	char(4)	SI
movimiento_id_acum	Clave del tipo de movimiento efectuado en el cajero.	decimal(2)	SI
cuenta_id_acum	Clave de la cuenta de la tarjeta.	char(2)	SI
moneda_id_acum	Clave de tipo de moneda.	decimal(3)	SI
aceptadas_num	Número de transacciones aceptadas.	decimal(18)	SI
importe_aceptadas	Importe de las transacciones afectadas.	decimal(18,2)	SI
rechazadas_num	Número de transacciones rechazadas.	decimal(18)	SI
importe_rechazadas	Importe de las transacciones rechazadas.	decimal(18,2)	SI
costo_ca_acum	Importe de costo de cajero.	money(12,4)	SI
costotr_acum	Importe de costo otros.	money(12,4)	SI
costcnx_acum	Importe de costo de conexión.	money(12,4)	SI
costred_acum	Importe de costo de red.	money(12,4)	SI
punto_equl_acum	Clave de punto de equilibrio.	integer	SI

Tabla: banco			
Objetivo: Tener un catálogo de bancos que permita identificar a quien pertenecen las transacciones.			
Nombre de Campo	Descripción	Tipo	Nulos
Banco_id	Identificador del banco	Char (4) PK	NO
Grupo_id	Identificador del grupo	Char (4)	NO
Banco_dsc	Descripción del banco	Char (30)	NO

Tabla: cajero			
Objetivo: Tener un catálogo de cajeros que permita identificar a que cajeros pertenecen los movimientos.			
Nombre de Campo	Descripción	Tipo	Nulos
Cajero_id	Identificador del cajero.	Char (16) PK	NO
Sucursal_id	Identificador de la sucursal.	Char (4)	SI
Cajero_dsc	Descripción del cajero.	Char (30)	NO
Ubicacion_id	Identificador de la ubicación.	Smallint	SI
Banco_id	Identificador del banco.	Char(4)	SI
Status	Bandera de modificación.	Integer	SI

Tabla: cajeros_no_region			
Objetivo: Contiene aquellas claves de cajeros que presentan movimientos pero no que están regionalizados.			
Nombre de Campo	Descripción	Tipo	Nulos
Cajero_id	Identificador del cajero	Char (16)	SI
Banco_id	Identificador del banco	Char (4)	SI
Fecha_corte	Fecha del movimiento	Date	SI
Status	Bandera de modificación.	Integer	SI

Tabla: fechas			
Objetivo: Llevar el control de los días procesados.			
Nombre de Campo	Descripción	Tipo	Nulos
Dias_proces	Fecha de los días que si han sido cargados por el sistema.	Integer	NO

Tabla: moneda			
Objetivo: Contiene la clave y la descripción de las monedas utilizadas en la aplicación.			
Nombre de Campo	Descripción	Tipo	Nulos
Moneda_id	Clave de la moneda.	decimal(3)	NO
Moneda_dsc	Descripción de la moneda.	char(30)	NO

Tabla: plaza			
Objetivo: Tener un catálogo de plazas que permita asignar plazas para la regionalización establecida.			
Nombre de Campo	Descripción	Tipo	Nulos
Plaza_id	Clave de la plaza.	Char (4)	NO
Region_id	Clave de región.	Char (4)	NO
Plaza_dsc	Descripción de la plaza.	Char (30)	NO
Banco_id	Clave del banco.	Char (4)	NO

Tabla: razon_respuesta			
Objetivo: Contiene las razones de respuestas de las transacciones realizadas en los cajeros automáticos.			
Nombre de Campo	Descripción	Tipo	Nulos
razon_id	Clave de la razón de respuesta.	Char(2)	NO
razon_dsc	Descripción de la respuesta.	Char(30)	NO

APENDICE G. MANUAL TECNICO

Tabla: region			
Objetivo: Tener un catálogo que almacene las regiones asignadas por el banco.			
Nombre de Campo	Descripción	Tipo	Nulos
Region_id	Clave de la región.	Char (4)	NO
Banco_id	Clave del banco.	Char (4)	NO
Region_dsc	Descripción de la región.	Char (30)	NO

Tabla: sector			
Objetivo: Almacenar las claves de los sectores en donde pueden estar asignados los cajeros.			
Nombre de Campo	Descripción	Tipo	Nulos
Sector_id	Clave del sector.	Char (1)	NO
Sector_dsc	Descripción del sector.	Char (25)	NO
Costo_ca	Costo cajero automático.	Integer	NO
Punto_equi	Punto de equilibrio.	Integer	NO

Tabla: sucursal			
Objetivo: Crear un enlace con la región y la plaza de acuerdo a la asignación del cajero.			
Nombre de Campo	Descripción	Tipo	Nulos
Sucursal_id	Clave de la sucursal.	Char (4)	NO
Plaza_id	Clave de la plaza.	Char (4)	NO
Region_id	Clave de la región.	Char (4)	SI
Banco_id	Clave de la banco.	Char (4)	SI

Tabla: tipo cuenta			
Objetivo: Identificar los tipos de cuenta.			
Nombre de Campo	Descripción	Tipo	Nulos
Cuenta_id	Clave de cuenta o identificador.	Char (2)	NO
Cuenta_dsc	Descripción de la cuenta (00 - no hay cuenta, 01 - cuenta de cheques, 11 - cuenta maestra, 31 - tarjeta de crédito).	Char (30)	NO

Tabla: tipo movimiento			
Objetivo: Contiene los movimientos que se permiten realizar en los cajeros automáticos.			
Nombre de Campo	Descripción	Tipo	Nulos
Movimiento_id	Clave del movimiento.	Decimal (2)	NO
Movimiento_dsc	Descripción del movimiento.	Char (30)	NO
Comision	Comisiones a cobrar.	Decimal (5,2)	SI
Porcentaje	Porcentaje de comisión.	Char (1)	SI

Tabla: tipo respuesta			
Objetivo: Contiene los tipos de respuestas de las transacciones realizadas en los cajeros automáticos.			
Nombre de Campo	Descripción	Tipo	Nulos
Mensaje_id	Clave del mensaje.	Decimal(4)	NO
Mensaje_dsc	Descripción del mensaje.	Char(30)	NO

APENDICE G. MANUAL TECNICO

Tabla: transaccion			
Objetivo: Almacenar los movimientos diarios por cajero.			
Nombre de Campo	Descripción	Tipo	Nulos
Transaccion_id	Identificación de la transacción.	Char (12)	NO
Transaccion_fecha	Fecha de la transacción.	datetime year to second.	NO
Cajero_id	Clave del cajero.	Char (16)	NO
Mensaje_id	Clave del mensaje que envía el cajero automático	Decimal (4)	NO
Razon_id	Clave de la razón de respuesta.	Char (2)	SI
Movimiento_id	Clave del movimiento realizado.	Decimal (2)	SI
Cuenta_acceso	Cuenta de acceso de la tarjeta que realiza la transacción.	Char (19)	SI
Cuenta_id	Clave de la cuenta.	Char (2)	SI
Numero_cuenta	Número de cuenta de la tarjeta que accesa.	Char (19)	SI
Banco_id	Clave del banco.	Char (4)	SI
Moneda_id	Clave de la moneda.	Decimal (3)	SI
Transaccion_corte	Fecha de corte de la transacción.	Datetime year to second	SI
Transaccion_import	Importe.	Float	SI
Transaccion_ajuste	Ajuste del importe.	Float	SI
Transaccion_disp	Importe disponible de la transacción.	Float	SI
Tipo_cambio	Tipo de cambio.	Decimal (8)	SI
Banco_tarjeta	Banco emisor de la tarjeta.	Char (4)	NO
Moneda_cliente	Tipo de moneda en que se dispone.	Decimal (3)	NO

Tabla: ubicacion			
Objetivo: Tener un catálogo con las ubicaciones de los cajeros.			
Nombre de Campo	Descripción	Tipo	Nulos
Ubicacion_id	Clave de la ubicación.	Smallint PK	NO
Ubicacion_dsc	Descripción de la ubicación.	Varchar (30,0)	NO
Sector_id	Clave del sector.	Char (1)	NO

4.2 DESCRIPCION DE TABLAS DE MC_CICA

Tabla: banco_emisor			
Objetivo: Almacenar la DIMENSION de bancos desde el punto de vista de que emiten tarjetas			
Nombre de Campo	Descripción	Tipo	Nulos
agg_level	Contiene el nivel de agregado	integer	SI
banco_emisor	Clave del banco.	char(4)	NO
des_emisor	Descripción del banco.	char(30)	NO

Tabla: catalogo_procesos			
Objetivo: Identificar cada proceso dentro de la extracción, transformación y carga.			
Nombre de Campo	Descripción	Tipo	Nulos
clave_proceso	Clave del proceso.	char(3)	NO
descrip_proceso	Descripción del proceso.	char(75)	NO
nombre_proceso	Nombre del proceso.	char(20)	NO
frecuencia_proceso	Cada cuando se realiza el proceso (Diario, Quincenal o Mensual)	char(1)	NO

Tabla: control			
Objetivo: Identificar el estado del resultado de la carga de un periodo (normalmente un día)			
Nombre de Campo	Descripción	Tipo	Nulos
fecha_inicial	Fecha inicial del rango.	date	NO
fecha_final	Fecha final del rango.	date	NO
status_proceso	Estado del proceso.	char(1)	SI

Tabla: control_procesos			
Objetivo: Identificar el estado de cada proceso en un periodo determinado (normalmente un día)			
Nombre de Campo	Descripción	Tipo	Nulos
clave_proceso	Clave del proceso (para relacional con la tabla catalogo_procesos)	char(3)	NO
fecha_proceso	Fecha de realización del proceso.	date	NO
estatus_proceso	Estado del proceso.	char(1)	NO

Tabla: cuenta			
Objetivo: Almacenar los datos de la DIMENSION que contiene los datos de los tipos de cuenta.			
Nombre de Campo	Descripción	Tipo	Nulos
agg_level	Nivel de agregado.	integer	SI
cve_cuenta	Clave del tipo de cuenta.	integer	NO
des_cuenta	Descripción del tipo de cuenta.	char(50)	NO

APENDICE G. MANUAL TECNICO

Tabla: geografía			
Objetivo: Almacenar los datos de la DIMENSION que contiene la localización de los cajeros automáticos de BanCrecer y de los otros bancos.			
Nombre de Campo	Descripción	Tipo	Nulos
agg_level	Indica el nivel de agregamiento de la información	integer	SI
cve_banco	Clave del banco.	char(4)	SI
cve_region	Clave de la región.	integer	SI
cve_plaza	Clave de la plaza.	char(4)	SI
cve_cajero	Clave del cajero.	char(16)	SI
cve_ubicacion	Clave de la ubicación.	integer	SI
cve_sector	Clave del sector.	char(1)	SI
des_banco	Descripción del banco.	char(30)	SI
des_region	Descripción de la región.	char(20)	SI
des_plaza	Descripción de la plaza.	char(30)	SI
des_cajero	Descripción del cajero.	char(40)	SI
des_ubicacion	Descripción de la ubicación.	char(30)	SI
des_sector	Descripción del sector.	char(30)	SI

Tabla: Movimiento			
Objetivo: Almacenar los datos de la DIMENSION correspondiente. Contiene la descripción de los tipos de movimientos que se pueden realizar en los cajeros automáticos BanCrecer.			
Nombre de Campo	Descripción	Tipo	Nulos
agg_level	Indica el nivel de agregamiento de la información.	integer	SI
cve_movimiento	Clave del tipo de movimiento.	integer	NO
des_movimiento	Descripción del tipo de movimiento.	char(30)	NO

Tabla: Movimientos			
Objetivo: Almacenar los datos de la TABLA DE HECHOS del CICA. Contiene la información de las métricas de las operaciones realizadas en los cajeros automáticos.			
Nombre de Campo	Descripción	Tipo	Nulos
id_razon	Clave de la razón de respuesta.	char(2)	SI
cve_tiempo	Clave del tiempo.	integer	NO
cve_cajero	Clave del cajero.	char(16)	NO
cve_banco	Clave del banco dueño del cajero.	char(4)	NO
banco_emisor	Clave del banco emisor de la tarjeta.	char(4)	NO
cve_cuenta	Clave del tipo de cuenta de la tarjeta.	integer	NO
cve_movimiento	Clave del tipo de movimiento.	integer	NO
mov_aceptado	Número de movimientos aceptados.	integer	SI
mov_rechazado	Número de movimientos rechazados.	integer	SI
monto_aceptado	Importe de los movimientos aceptados.	money(16,2)	SI
monto_rechazado	Importe de los movimientos rechazados.	money(16,2)	SI

Tabla: razon respuesta			
Objetivo: Contiene los datos de la DIMENSION correspondiente. Contiene las razones de respuestas de las transacciones realizadas en los cajeros automáticos.			
Nombre de Campo	Descripción	Tipo	Nulos
agg_level	Indica el nivel de agregamiento de la información.	integer	SI
id_razon	Clave de la razón de respuesta.	char(2)	SI
des_razon	Descripción de la razón de respuesta.	char(30)	SI

Tabla: resultados			
Objetivo: Contiene los estatus de los días procesados durante la carga de información para CICA.			
Nombre de Campo	Descripción	Tipo	Nulos
fecha_proceso	Fecha del día de proceso.	date	NO
tipo_proceso	Tipo del proceso realizado.	char(1)	SI
estatus_proceso	Estatus del día de proceso.	char(1)	SI
clave_proceso	Clave del último proceso realizado.	char(3)	SI
observaciones	Descripción del estatus del día.	char(75)	SI

Tabla: tiempo			
Objetivo: Contiene los datos de la DIMENSION correspondiente. Sirve para identificar las quincenas procesadas.			
Nombre de Campo	Descripción	Tipo	Nulos
Cve tiempo	Clave de tiempo.	Integer	SI
Agg_level	Máximo nivel de detalle por tiempo (quincena).	Integer	SI
Cve_periodo	Clave del periodo (I = Periodo actual, P = Procesado).	Char (1)	SI
Desc_periodo	Descripción del periodo.	Char (15)	SI
Cve_quincena	Clave de la quincena procesada.	Integer	SI
Cve_mes	Clave del mes.	Integer	SI
Año	Año.	Integer	SI
Des_quincena	Descripción de la quincena.	Char (25)	SI
Des_mes	Descripción del mes.	Char (10)	SI
Fecha_inicial	Fecha inicial de la quincena.	Date	SI
Fecha_final	Fecha final de la quincena.	Date	SI

4.3 DESCRIPCION DE INDICES DE LA BASE DE DATOS "bdcica"

Tabla: acumulado_mc		Base de Datos: bdcica	
Nombre de Indice	Nombre de Campo	Tipo	Cluster
acumulado_mc_x0 1	razon_id_acum cajero_id_acum banco_id_acum banco_tarj_id_acum cuenta_id_acum movimiento_id_acu m costo_ca_acum costotr_acum costcnx_acum costred_acum punto_equi_acum	dupls	NO
acumulado_mc_x0 2	fecha_corte	dupls	NO
acumulado_mc_x0 3	razon_id_acum	dupls	NO
acumulado_mc_x0 4	cajero_id_acum plaza_id_acum region_id_acum	dupls	NO
acumulado_mc_x0 5	banco_id_acum	dupls	NO
acumulado_mc_x0 6	banco_tarj_id_acum	dupls	NO
acumulado_mc_x0 7	cuenta_id_acum	dupls	NO
acumulado_mc_x0 8	movimiento_id_acu m	dupls	NO
acumulado_mc_x0 9	fecha_corte banco_id_acum cajero_id_acum	dupls	NO

Tabla: banco		Base de Datos: bdcica	
Nombre de Indice	Nombre de Campo	Tipo	Cluster
xpkbanco	banco_id	unique	NO
xif57banco	grupo_id	dupls	NO
xif58banco	banco_dsc	dupls	NO

Tabla: cajero		Base de Datos: bdcica	
Nombre de Indice	Nombre de Campo	Tipo	Cluster
xpkcajero	cajero_id	unique	NO
xif45cajero	sucursal_id	dupls	NO
xif50cajero	cajero_dsc	dupls	NO
ix434_5	banco_id	dupls	NO

Tabla: moneda		Base de Datos: bdcica	
Nombre de Indice	Nombre de Campo	Tipo	Cluster
xpkmoneda	moneda_id	unique	NO

APENDICE G. MANUAL TECNICO

Tabla:	plaza	Base de Datos:	bdclca
Nombre de Índice	Nombre de Campo	Tipo	Cluster
xpkplaza	plaza_id region_id banco_id	unique	NO
xif55plaza	region_id banco_id	dupls	NO

Tabla:	razon_respuesta	Base de Datos:	bdclca
Nombre de Índice	Nombre de Campo	Tipo	Cluster
xpkrazon_respuesta	razon_id	unique	NO

Tabla:	region	Base de Datos:	bdclca
Nombre de Índice	Nombre de Campo	Tipo	Cluster
xpkregion	region_id banco_id	unique	NO
xif54region	banco_id	dupls	NO
xif55region	region_dsc	dupls	NO

Tabla:	sector	Base de Datos:	bdclca
Nombre de Índice	Nombre de Campo	Tipo	Cluster
105_15	sector_id	unique	NO

Tabla:	sucursal	Base de Datos:	bdclca
Nombre de Índice	Nombre de Campo	Tipo	Cluster
xpkbanco_regpla	sucursal_id	unique	SI
xif47sucursal	plaza_id region_id banco_id	dupls	NO
idx_suc	banco_id sucursal_id plaza_id region_id	dupls	NO

Tabla:	tipo_cuenta	Base de Datos:	bdclca
Nombre de Índice	Nombre de Campo	Tipo	Cluster
xpktipo_cuenta	cuenta_id	unique	NO

Tabla:	tipo_movimiento	Base de Datos:	bdclca
Nombre de Índice	Nombre de Campo	Tipo	Cluster
xpktipo_movimiento	movimiento_id	unique	NO

Tabla:	tipo_respuesta	Base de Datos:	bdclca
Nombre de Índice	Nombre de Campo	Tipo	Cluster
xpktipo_respuesta	mensaje_id	unique	NO

APENDICE G. MANUAL TECNICO

Tabla:		transaccion	Base de Datos: bdcica	
Nombre de Índice	Nombre de Campo	Tipo	Cluster	
xtransaccion	banco_id	dupls	NO	
	cajero_id			
	cuenta_id			
	banco_tarjeta			
	movimiento_id			
	moneda_id			
	razon_id			
	mensaje_id			
xtransaccion_02	transaccion_id	dupls	NO	
	transaccion_fecha			
	cajero_id			
xtransaccion_03	mensaje_id			
	transaccion_corte	dupls	NO	

Tabla:		ubicacion	Base de Datos: bdcica	
Nombre de Índice	Nombre de Campo	Tipo	Cluster	
ix2255_1	ubicacion_id	unique	NO	

4.4 DESCRIPCION DE INDICES DE LA BASE DE DATOS "mc_cica"

Tabla:		banco_emisor	Base de Datos: mc_cica	
Nombre de Índice	Nombre de Campo	Tipo	Cluster	
xpkbanco_emisor	banco_emisor	unique	SI	
x_agglevel_bem	agg_level	dupls	NO	
x_demisor	des_emisor	dupls	NO	

Tabla:		cuenta	Base de Datos: mc_cica	
Nombre de Índice	Nombre de Campo	Tipo	Cluster	
xpkcuenta	cve_cuenta	unique	SI	

Tabla:		geografia	Base de Datos: mc_cica	
Nombre de Índice	Nombre de Campo	Tipo	Cluster	
x_geog	cve_cajero	dupls	NO	
x_geog_dc	des_cajero	dupls	NO	
x_geog_r	cve_region	dupls	NO	
x_geog_p	cve_plaza	dupls	NO	
xgeografia_1	des_region	dupls	NO	
xgeografia_2	des_plaza	dupls	NO	
x_geog_sector	cve_sector	dupls	NO	
x_geog_ubicacion	cve_ubicacion	dupls	NO	
x_geog_agglevel	agg_level	dupls	NO	
x_geog_dubicacion	des_ubicacion	dupls	NO	
x_geog_dsector	des_sector	dupls	NO	
x_geog_cajplareg	cve_cajero	dupls	NO	
	cve_plaza			
	cve_region			
x_geog_parametros	cve_cajero	dupls	NO	
	cve_banco			
x_geog_parametros1	des_cajero	dupls	NO	
	cve_plaza			
	des_plaza			

APENDICE G. MANUAL TECNICO

	cve_region des_region		
--	--------------------------	--	--

Tabla: movimiento		Base de Datos: mc_cica	
Nombre de Índice	Nombre de Campo	Tipo	Cluster
xpkmovimiento	cve_movimiento	unique	SI
x_des_movimiento	des_movimiento	dupls	NO

Tabla: movimientos		Base de Datos: mc_cica	
Nombre de Índice	Nombre de Campo	Tipo	Cluster
x_movimiento1	cve_movimiento cve_cajero banco_emisor cve_banco cve_tiempo	dupls	NO
x_movimiento2	cve_cajero cve_banco cve_tiempo	dupls	NO
x_movimiento3	cve_cajero cve_movimiento cve_tiempo	dupls	NO
x_movimiento4	cve_cajero	dupls	NO
x_movimiento5	cve_movimiento	dupls	NO
x_movimiento6	cve_tiempo	dupls	NO

Tabla: razon respuesta		Base de Datos: mc_cica	
Nombre de Índice	Nombre de Campo	Tipo	Cluster
x_razon	id_razon	dupls	NO
x_agglevel_peq	agg_level	dupls	NO
x_drazon	des_razon	dupls	NO

Tabla: tiempo		Base de Datos: mc_cica	
Nombre de Índice	Nombre de Campo	Tipo	Cluster
x_dmes	des_mes	dupls	NO
x_finicial	fecha_inicial	dupls	NO
x_ffinal	fecha_final	dupls	NO
x_tiempo	cve_tiempo	dupls	NO
x_tiempo_mes	cve_mes	dupls	NO
x_tiempo_anio	anio	dupls	NO
x_tiempo_agg	agg_level	dupls	NO
x_tiempo_dquin	des_quincena	dupls	NO
x_periodo	cve_periodo	dupls	NO
x_dperiodo	desc_periodo	dupls	NO
x_quincena	cve_quincena	dupls	NO

5. SHELLS Y PROCEDIMIENTOS ALMACENADOS

5.1 SHELLS

Existen una gran variedad de shells que realizan diferentes funciones: crear tablas e índices, cargar registros en las tablas y actualizar tablas. Los shells se ubican en:

- Servidor: siglo
- Dirección IP: 100.1.2.15
- Directorio: /usr2/dbsiglo/cica/shell
- Usuario: dbsiglo
- Contraseña: siglo123

A continuación se dará una breve descripción de cada uno de los shells.

Nombre de shell	Parámetros	Propósito
actua_agre.sh	ANIO=\$1:Año del proceso. MES=\$2:Mes del proceso. DIA=\$3:Día del proceso.	Ejecutar los procesos quincenales y mensuales. Actualiza los agregados, costos y archivos para la fase ejecutiva y operativa de CICA (solamente se ejecuta cuando todos los días del 1 al 15 ó del 16 al último del mes están actualizados y con cifras de control correctas).
actua_tie.sh	ANIO=\$1:Año del proceso. MES=\$2:Mes del proceso. DIA=\$3:Día del proceso.	Actualiza en la tabla "tiempo" de la base de datos "mc_cica", los siguientes campos: <i>cve_periodo</i> y <i>desc_periodo</i> con "P" y "Periodo Actual" respectivamente, para el periodo anterior al cierre. <i>cve_periodo</i> y <i>desc_periodo</i> con "Y" y con "Periodo Actual" respectivamente, para el periodo de actualización Nota: solamente se ejecuta cuando todos los días del 1 al 15 ó del 16 al último del mes están actualizados y con las cifras de control diarias correctas.
ayer	FECHA: La fecha del proceso (formato AAMMDD).	Genera la fecha del día anterior.
borra.sh	Ninguno	Borra los archivos previamente respaldados en una cinta.
califica.sh	Fecha del proceso	Ejecuta el procedimiento "califica" en "mc_cica" con parámetros determinados.
carga.sh	anio=\$1:Año del proceso. mes=0\$2:Mes del proceso. dia=\$3:Día del proceso. max=\$4:Día del proceso	Carga los archivos TLF recibidos en la tabla "transacción" en "bdcica" y genera las cifras del control del proceso. Ejecuta el shell "control.sh" y los procedimientos "actua_cli" y "sp_carga".
carga_fact.sh	Ninguno	Inserta en la tabla "movimientos" (en "mc_cica"), los registros de la tabla "acumulado_mc" (en "bdcica") y una clave de tiempo de la tabla "tiempo" (en "mc_cica").
carga_geografia.sh	Ninguno	Crea la tabla "geografia" (en "mc_cica") a partir del archivo "geografia.unf" y crea los niveles de agregamiento para la información.
control.sh	Ninguno	Actualiza la tabla "control_tmp" (en "bdcica") con las cifras de control recibidas en los archivos TLF.

APENDICE G. MANUAL TECNICO

Nombre de shell	Parámetros	Propósito
crea_transac.sh	Ninguno	Crea la tabla "transacción" (en "bdcca") junto con sus índices.
imovimientos_a01.sh	Ninguno	Genera agregado "movimientos_a01" a partir de la tabla "movimientos".
imovimientos_a02.sh	Ninguno	Genera agregado "movimientos_a02" a partir de la tabla "movimientos".
imovimientos_a03.sh	Ninguno	Genera agregado "movimientos_a03" a partir de la tabla "movimientos".
imovimientos_a04.sh	Ninguno	Genera agregado "movimientos_a04" a partir de la tabla "movimientos".
imovimientos_a05.sh	Ninguno	Genera agregado "movimientos_a05" a partir de la tabla "movimientos".
imovimientos_a06.sh	Ninguno	Genera agregado "movimientos_a06" a partir de la tabla "movimientos".
imovimientos_a07.sh	Ninguno	Genera agregado "movimientos_a07" a partir de la tabla "movimientos".
imovimientos_a08.sh	Ninguno	Genera agregado "movimientos_a08" a partir de la tabla "movimientos".
imovimientos_a09.sh	Ninguno	Genera agregado "movimientos_a09" a partir de la tabla "movimientos".
imovimientos_a10.sh	Ninguno	Genera agregado "movimientos_a10" a partir de la tabla "movimientos".
imovimientos_all.sh	Ninguno	Crea todos los agregados desde "movimientos_a01" hasta "movimientos_a10" a partir de la tabla "movimientos".
limpia.sh	Ninguno	Limpiar de los archivos TLF, los caracteres de control que llegan y en su lugar coloca un "pipe" ().
load_tra.sh	Ninguno	Carga en la tabla "transaccion_tmp" (en "bdcca") a partir del archivo TLF.
regional.sh	\$ANIO = Año \$MES = Mes	Por cada cajero que encuentre en la tabla cajeros_no_region actualizará las siguientes tablas: bdcca:acumulado_mc mc_cica:geografia mc_cica:movimientos identificando la region_id, plaza_id como '9999' y su region_dsc y plaza_dsc como 'POR REGIONALIZAR'. Para la ubicación_id y sector_id las claves se actualizan como 0 y '0' respectivamente, del mismo modo la ubicación_dsc y sector_dsc se actualizan como 'UBICACIÓN DESCONOCIDA' y 'SECTOR DESCONOCIDO' respectivamente.
siglo-cica.sh	Ninguno	Es el proceso principal encargado de ejecutar la carga y cierre diario, quincenal y/o mensual. Llama a todos los demás shells y procedimientos que intervienen en este proceso.
veri_control.sh	\$AP = Año \$ME = Mes \$DP = Día 23 = Número de proceso "D" = Identifica que el proceso es diario	Verifica los procesos anteriores en la tabla "control_procesos" (en "mc_cica"). Esto es para que si el proceso anterior (o cualquier otro) está incorrecto se detenga la ejecución de todo el proceso.
veri_fechas.sh	Año, Mes y Día de proceso	Verifica fechas dentro de los archivos TLF

5.2 PROCEDIMIENTOS ALMACENADOS EN LA BASE DE DATOS "bdcica"

Existen "stored procedures" tanto en la base de datos "bdcica" como en "mc_cica". A continuación se especificarán y se dará una breve descripción de cada uno de ellos. Para ver el listado correspondiente al más representativo ver la sección LISTADO DE PROCEDIMIENTO ALMACENADO.

Nombre	Parámetros	Propósito
actua_ctl	cExt char(5): Extensión del archivo que se está procesando ("cer" para BanCrecer y "cen" para BanCrecen).	Actualizar tabla "controlt" de "bdcica" con las cifras de control recibidas en el archivo TLF.
actualiza_geo	cCve_cajero.....char(6): clave del cajero que se requiere dar de alta.	Inserta un cajero nuevo en la tabla "geografia" de "mc_cica" y crea los niveles de agregados requeridos.
cajeros	dFechaIni.....char(10), dFechaFin.....char(10): Indican el periodo de tiempo para el que se está ejecutando el proceso. El formato es AAAA/MM/DD.	Busca en la tabla "geografia" de "mc_cica", aquellos cajeros que no están regionalizados y los registra en la tabla "cajeros_no_region" de "bdcica".
sp_carga	iAño.....char(4): Año del proceso. iMes.....char(2): Mes del proceso. iDia.....char(2): Día del proceso. cExt.....char(3): Extensión del banco que se está procesando.	Carga el archivo TLF recibido a la tabla "transaccion_tmp" de "bdcica" y de esta última a la tabla "transaccion" de "bdcica".
sp_movim	dFechaInicio char(10), dFechaCierre char(10): Indican el periodo de tiempo para el que se está ejecutando el proceso. El formato es AAAA/MM/DD.	Carga los registros de la tabla "transaccion" de "bdcica" a la tabla "acumulado_mc" de "bdcica".

5.3 PROCEDIMIENTOS ALMACENADOS EN LA BASE DE DATOS "mc_cica"

Nombre	Parámetros	Propósito
actua_pro	cClave_Proceso.....char(3), cFecha_Proceso.....char(10), cEstatus_Proceso...char(1)	Inserta o actualiza en la tabla "control_procesos" de "mc_cica", el estatus de terminación de los procesos.
actualiza_movtos	cCve_cajero...char (6):Clave del cajero que se requiere actualizar. cFecha char(10):Fecha a partir de la cual se actualizarán los registros del cajero. El formato es AAAA/MM/DD.	Inserta en la tabla "movimientos" de "mc_cica", a partir de la tabla "acumulado_mc" de "bdcica", los registros de los cajeros actualizados.
costos_p	dss....varchar(240): Nombre del modelo de datos.	Actualiza los costos para las tablas de hechos y de agregados.
genera_parametros	Ninguno	Actualiza la tabla "parametros" de "mc_cica" para el mes de proceso.
graba_resultados	cProceso_tmp....char(1):Tipo de proceso. cClave_tmp....char(3):Clave del proceso. cFecha_tmp....char(10):Fecha del proceso (formato AAAA/MM/DD).	Graba en la tabla "resultados" de "mc_cica", el estatus del día de proceso.
total_cajeros	iCve_mes.....integer:Clave del mes de proceso (formato AAAAMM).	Inserta en la tabla "numero_cajeros", los cajeros productivos del mes especificado junto con los datos del sector, de la región y de la plaza.
veri_proceso	cFecha.....char(10):Fecha del proceso (formato AAAA/MM/DD) cProceso char(1):Tipo de proceso.	Verifica si todos los procesos diarios, quincenales y/o mensuales están ejecutados.

6. CRONS

El cron que existe para CICA se ejecuta en el servidor **siglo** (IP 100.1.2.15.) con usuario **dbsiglo** y password **siglo123**. Está programado para ejecutarse automáticamente todos los días: a las 04:30 a.m. ejecuta el shell **limpia.sh** y a las 04:40 a.m. ejecuta el shell **siglo-cica.sh**. El fin de este cron es hacer la carga y el cierre diario y mensual.

El cron se ejecuta en el servidor **siglo** (IP 100.1.2.15.) con usuario **dbsiglo** y password **siglo123**. Está se encuentra en el mismo servidor en la ruta **/usr2/dbsiglo/cica/shell**. El fin de este cron es hacer la carga y el cierre diario y mensual.

Mínuto	Hora	Año	Mes	Día	Shell	Servidor
30	04	*	*	*	/usr2/dbsiglo/cica/shell/limpia.sh	SIGLO 100.1.2.15
40	04	*	*	*	/usr2/dbsiglo/cica/shell/siglo-cica.sh	SIGLO 100.1.2.15

7. PLAN DE CONTINGENCIAS EN CASO DE FALLAS

El plan de contingencias contiene los puntos básicos para recuperar en caso de falla.

7.1 FALLA DE LA LINEA DE COMUNICACION

Debido a que no se cuenta con un servidor de respaldo, en caso de que no exista conexión con el servidor "siglo" (IP 100.1.2.15) será necesario esperar a que se restablezca la comunicación para continuar trabajando.

7.2 FALLAS DEL SERVIDOR

En caso de que exista algún problema de hardware o de software con el servidor "siglo" es necesario reportarlo a la regional puesto que no existe un servidor de respaldo, por lo tanto el usuario no podrá trabajar con la aplicación hasta que se repare el servidor.

7.3 VIRUS

Debido a que las máquinas en las que se instala la aplicación "CICA" con MetaCube Explorer deben tener sistema operativo Windows 95 y que la paquetería antivirus "SACN" no ha sido liberada en BanCrecer, es necesario reportar esta situación a Soporte a En caso de contingencia se debe recuperar el respaldo más reciente tanto del sistema operativo como de las Bases de Datos (bdcica y mc_cica) y en caso de ser necesario los respaldos parciales. Para más información ver el detalle del plan de respaldos.

8. LISTA DE USUARIOS DEL SISTEMA Y DATOS GENERALES

Medios de Pago y Banca Electrónica

Usuario: Ramón Velarde
 Puesto: Director Banca Electrónica
 Ubicación: Reforma 116 piso 9
 Tel. Directo: 53-29-63-63
 R.D.I.: 8-510-2230 / 2391

Usuario: María de Jesús González Ramírez (Responsable del sistema)
 Puesto: Subgerente Información y Estadística

Ubicación: Reforma 116 piso 9
 Tel. Directo: 53-29-63-63
 R.D.I.: 8-510-2230 / 2391

Tarjeta de Crédito

Usuario: Sergio Pérez
 Puesto: Analista de reportes y estadísticas
 Ubicación: Reforma 126 piso 2
 Tel. Directo:
 R.D.I.: 8-515-2410

9. PERMISOS UNIX Y/O INFORMIX POR USUARIO

user dbsiglo

DBSCHEMA Schema Utility INFORMIX-SQL Versión 7.22.UC2.
 Copyright (C) Informix Software, Inc., 1984-1996.
 Software Serial Number INF#J010494.
 grant dba to "dbsiglo";

10. CALENDARIO DE RESPALDOS

Tipo de Respaldo	Frecuencia	Cintas		Horario de Respaldo.	Observaciones
		Retención	Número		
Parcial de TLF	Cada semana.	Mensual	4 de 2GB.	Sábado a las 20:00 hrs.	Ruta de los archivos TLF : /usr2/dbsiglo/cica/respaldos_tlf
Base de Datos	Para Nivel 0. Cada Semana.	Cada dos semanas	2 de 4GB.	Domingo a las 10:00 hrs.	
	Para Nivel 1. Diario.	Cada dos semanas	12 de 2GB.	A las 22:00 hrs.	
Total	Cada mes.	Cada dos meses	2 de 4GB.	Primer domingo de cada mes a las 10:00 hrs.	

11. LISTADO DE PROGRAMAS

Por cuestiones de espacio, se presenta solamente el listado de un shell y de un procedimiento almacenado.

11.1 LISTADO DE SHELL

A continuación se presenta uno de los shells más representativos del CICA.

- siglo-cica.sh

```

#-----#
# PROGRAMA: siglo-cica.sh #
# FECHA...: 11/marzo/1999 #
# AUTOR...: CESAR SALAS #
# OBJETIVO: ACTUALIZACION DE APLICACION #
    
```

```

#-----#
# SIGLO-CICA #
#-----#
# MODIFICA: CESAR SALAS #
# FECHA...: 13/septiembre/1999 #
# OBJETIVO: SE AGREGO PROGRAMA #
#          veri_fechas.sh PARA CHECAR #
#          FECHAS DE ARCHIVOS TLF #
#          (SOLO PARA FIN DE MES) #
#-----#
# MODIFICA: CESAR SALAS #
# FECHA...: 15/mayo/2000 #
# OBJETIVO: SE AGREGO PROGRAMA #
#          crea_transac2.sh PARA DEPURAR #
#          TABLAS transaccion (bdcica) Y #
#          transaccion_tmp A DIARIO. #
#-----#
# MODIFICA: Luis Vargas #
# FECHA...: 4/enero/2000 #
# OBJETIVO: Procesar los archivos TLF descomprimidos que se llaman #
#          tlaamdd.cer y tlaamdd.cen respectivamente y no #
#          tlaamdd.cer.cer y tlaamdd.cen.cen #
#          como estaba hasta el 31/12/2000 #
#-----#
# DECLARACION DE FUNCIONES #
#-----#
# GRABA LOS DIAS NO PROCESADOS #
#-----#
diasno()
{
  ANIO_F=`echo $FECHAINI|cut -c2-5`
  MES_F=`echo $FECHAINI|cut -c7-8`
  DIA_F=`echo $FECHAINI|cut -c10-11`
  echo $ANIO_F$MES_F$DIA_F >> $HOME/diasno

  echo "LOAD FROM $HOME/diasno
  INSERT INTO diasno;

  UNLOAD TO $HOME1/diasno.unl
  SELECT DISTINCT dias_no_proces
  FROM diasno;

  DELETE FROM diasno;

  LOAD FROM $HOME1/diasno.unl
  INSERT INTO diasno;" | dbaccess bdcica

  rm $HOME1/diasno.unl
}
#-----#
# GRABA EN TABLAS fechas DE bdcica #
#-----#
fechas()
{
  ANIO_F=`echo $FECHAINI|cut -c2-5`
  MES_F=`echo $FECHAINI|cut -c7-8`
  DIA_F=`echo $FECHAINI|cut -c10-11`
  echo $ANIO_F$MES_F$DIA_F >> $HOME/fechas

  echo "LOAD FROM $HOME/fechas
  INSERT INTO fechas;

  UNLOAD TO $HOME1/fechas.unl
  SELECT DISTINCT dias_proces
  FROM fechas;
}

```

APENDICE G. MANUAL TECNICO

```

DELETE FROM fechas;
LOAD FROM $HOME1/fechas.unl
INSERT INTO fechas;" | dbaccess bdcica

rm $HOME1/fechas.unl
)
-----
INFORMIXDIR=/usr/informix
export INFORMIXDIR
PATH=$INFORMIXDIR/bin:$PATH
export PATH
#
ONCONFIG=onconfig.sig
INFORMIXSERVER=siglo
PROYECTO=/usr2/dbsiglo/cica
DATOS=/datos
DBDATE=Y4MD/
DBTEMP=/tmp
export ONCONFIG INFORMIXSERVER PROYECTO DATOS DBDATE DBTEMP
#
HOME=/usr2/dbsiglo/cica/shell
HOME1=/usr2/dbsiglo/cica/datos
HOME2=/usr2/dbsiglo/cica/scripts
export HOME HOME1 HOME2
#
FECHA="hora" `date`
echo $FECHA > $HOME/fecha
touch $HOME/ftpget.log
#
# DETERMINA TIPO DE PROCESO A EJECUTAR
#
-----
if [ "$1" = "" ]; then
#-----
# PROCESO AUTOMATICO (SIN PARAMETROS)
#-----
DIA=`date +%w`
dia=`echo $DIA+1|bc`
$HOME/ayer:tee $HOME/fecha
AP=`cut -c1-2 $HOME/fecha`
MP=`cut -c3-4 $HOME/fecha`
DP=`cut -c5-6 $HOME/fecha`
ME=$MP
DI=$DP
FECHA=$AP$MP$DP
echo "-----" > $HOME/ftpget.log
echo PROCESO DE CARGA Y CIERRE --AUTOMATICO-- >> $HOME/ftpget.log
else
#-----
# PROCESO MANUAL (CON PARAMETROS)
#-----
AP=$1
MP=$2
DP=$3
ME=$MP
DI=$DP
FECHA=$AP$MP$DP
echo "-----" > $HOME/ftpget.log
echo PROCESO DE CARGA Y CIERRE --MANUAL-- >> $HOME/ftpget.log
fi;
#-----
# PROCESO DE CARGA Y CIERRE SIGLO-CICA
#-----
#
#-----
# DETERMINA FECHAS DE AYER
# Y HOY PARA LOS PROCESOS

```

APENDICE G. MANUAL TECNICO

```

#-----
FECHA_AYER=`$HOME/ayer $FECHA`
FECHA_AYER=`date +%C`$FECHA_AYER
FECHA_AYE2=`echo $FECHA_AYER|cut -c1-4`/"`echo $FECHA_AYER|cut -c5-6`
FECHA_AYE2=`echo $FECHA_AYE2`/"`echo $FECHA_AYER|cut -c7-8`
FECHA_AYE2=""$FECHA_AYE2""
#
FECHAINI=`date +%C`$AP/$ME/$DP
FECHAFIN=`date +%C`$AP/$ME/$DP
FECHAINI=""$FECHAINI""
FECHAFIN=""$FECHAFIN""
export FECHAINI FECHAFIN FECHA
#
echo PARA APLICACION: Siglo-CICA >>$HOME/ftpget.log
echo FECHA: $FECHAFIN >>$HOME/ftpget.log
date >>$HOME/ftpget.log
#-----
# VERIFICA ESTATUS DE PROCESO DEL DIA ANTERIOR
# EN TABLA "resultados" DE "mc_cica"
#-----
echo "-----">>$HOME/ftpget.log
echo VERIFICA ESTATUS DE PROCESO EN TABLA -resultados- DEL DIA ANTERIOR $FECHA_AYE2
>> $HOME/ftpget.log
date >> $HOME/ftpget.log
echo "UNLOAD TO $PROYECTOSDATOS/tempo$FECHA.unl
SELECT fecha_proceso, estatus_proceso FROM resultados
WHERE fecha_proceso=$FECHA_AYE2
ORDER BY fecha_proceso" | dbaccess mc_cica
#
if [ -s $PROYECTOSDATOS/tempo$FECHA.unl ]
then
  if grep "s" $PROYECTOSDATOS/tempo$FECHA.unl
  then
    echo "-----">>$HOME/ftpget.log
    echo EL DIA $FECHA_AYE2 TIENE ESTATUS CORRECTO EN TABLA -resultados- >>
$HOME/ftpget.log
    echo EL PROCESO CONTINUARA NORMALMENTE >> $HOME/ftpget.log
  else
    echo "-----">>$HOME/ftpget.log
    echo EL DIA $FECHA_AYE2 TIENE ESTATUS INCORRECTO EN TABLA -resultados- >>
$HOME/ftpget.log
    echo NO SE PUEDE CONTINUAR CON EL PROCESO HASTA CORREGIR EL PROBLEMA >>
$HOME/ftpget.log
    echo PARA MAS INFORMACION VER TABLA -control_procesos- EN -mc_cica- >>
$HOME/ftpget.log
    echo EL DIA $FECHAINI LO GUARDARA A LA TABLA diasno -dias no procesados- DE
LA BASE DE DATOS -bdcica- >> $HOME/ftpget.log
    #-----
    diasno
    #-----
    rm $PROYECTOSDATOS/tempo$FECHA.unl
    exit
  fi
else
  echo "-----">>$HOME/ftpget.log
  echo EL DIA $FECHA_AYE2 NO SE ENCONTRO EN TABLA -resultados- >> $HOME/ftpget.log
  echo EL PROCESO CONTINUARA NORMALMENTE >> $HOME/ftpget.log
fi
#-----
# VERIFICA ESTATUS DE PROCESO DEL DIA A PROCESAR
# EN TABLA "resultados" DE "mc_cica"
#-----
echo "-----">>$HOME/ftpget.log
echo VERIFICA ESTATUS DE PROCESO EN TABLA -resultados- DIA A PROCESAR $FECHAINI >>
$HOME/ftpget.log
date >> $HOME/ftpget.log

```

APENDICE G. MANUAL TECNICO

```

echo "UNLOAD TO $PROYECTO$DATOS/tempo$FECHA.unl
SELECT fecha_proceso, estatus_proceso FROM resultados
WHERE fecha_proceso=$FECHAINI
ORDER BY fecha_proceso" | dbaccess mc_cica
#
if [ -s $PROYECTO$DATOS/tempo$FECHA.unl ]
then
  if grep "S" $PROYECTO$DATOS/tempo$FECHA.unl
  then
    echo "-----">>$HOME/ftpget.log
    echo EL DIA $FECHAINI TIENE ESTATUS CORRECTO EN TABLA -resultados- >>
$HOME/ftpget.log
    echo NO SE PUEDE PROCESAR DE NUEVO >> $HOME/ftpget.log
    rm $PROYECTO$DATOS/tempo$FECHA.unl
    exit
  else
    echo "-----">>$HOME/ftpget.log
    echo EL DIA $FECHAINI TIENE ESTATUS INCORRECTO EN TABLA -resultados- >>
$HOME/ftpget.log
    echo NO SE PUEDE CONTINUAR CON EL PROCESO HASTA CORREGIR EL PROBLEMA >>
$HOME/ftpget.log
    echo PARA MAS INFORMACION VER TABLA -control_procesos- EN -mc_cica- >>
$HOME/ftpget.log
    echo EL DIA $FECHAINI SE GUARDARA EN LA TABLA diasno -dias no procesados- DE
LA BASE DE DATOS -bdcica- >> $HOME/ftpget.log
    #-----
    diasno
    #-----
    rm $PROYECTO$DATOS/tempo$FECHA.unl
    exit
  fi
else
  echo "-----">>$HOME/ftpget.log
  echo EL DIA $FECHAINI NO SE ENCONTRO EN TABLA -resultados- >>$HOME/ftpget.log
  echo EL PROCESO CONTINUARA NORMALMENTE >> $HOME/ftpget.log
fi
rm $PROYECTO$DATOS/tempo$FECHA.unl
#-----
echo "-----">>$HOME/ftpget.log
echo VALIDA QUE EXISTAN ARCHIVOS t1$FECHA.cer.zip Y t1$FECHA.cen.zip
>>$HOME/ftpget.log
date >> $HOME/ftpget.log
test -f $HOME1/t1$FECHA.cer.zip
if [ "$?" = 0 ]; then
  test -f $HOME1/t1$FECHA.cen.zip
  if [ "$?" = 0 ]; then
    #-----
    # Modificacion: 2001/01/04
    # Descripcion: Los archivos descomprimidos tienen extension .cer y .cen
    # Autor: Luis Vargas
    #-----
    echo "execute procedure actua_pro('D01',$FECHAFIN,'S') | dbaccess mc_cica
    #-----
    echo "-----">>$HOME/ftpget.log
    echo DESEMPACA ARCHIVOS t1$FECHA.cer.zip Y t1$FECHA.cen.zip
    >>$HOME/ftpget.log
    date >> $HOME/ftpget.log
    cd $HOME1
    pkunzip -do $HOME1/t1$FECHA.cer.zip
    pkunzip -do $HOME1/t1$FECHA.cen.zip
    #
    echo LIMPIA CARACTERES DE CONTROL A ARCHIVOS t1$FECHA.cer.cer Y
t1$FECHA.cen.cen >>$HOME/ftpget.log
    date >> $HOME/ftpget.log
    chmod 777 $HOME1/t1$FECHA*
    sed "s/\^@\//g" $HOME1/t1$FECHA.cer.cer > $HOME1/tlcer

```

APENDICE G. MANUAL TECNICO

```

sed "s/^\@//g" $HOME/tl$FECHA.cen.cen > $HOME/tlcen
cp -f $HOME/tlcer $HOME/tl$FECHA.cer
cp -f $HOME/tlcen $HOME/tl$FECHA.cen
#
echo BORRA ARCHIVOS tlcer Y tlicen>>$HOME/ftpget.log
date >> $HOME/ftpget.log
rm $HOME/tlcer
rm $HOME/tlcen
rm $HOME/tl$FECHA.cer.cer
rm $HOME/tl$FECHA.cen.cen
#-----
# VERIFICA FECHAS DENTRO DE ARCHIVOS
#-----
echo "-----">>$HOME/ftpget.log
echo VERIFICA FECHAS DENTRO DE ARCHIVOS TLF >>$HOME/ftpget.log
echo CON PROGRAMA veri_fechas.sh >>$HOME/ftpget.log
date >> $HOME/ftpget.log
cd $HOME2
veri_fechas.sh `date +%C`$AP $ME $DP
#
cd $HOME1
echo MUEVE ARCHIVOS tl$FECHA.cer.zip Y tl$FECHA.cen.zip>>$HOME/ftpget.log
date >> $HOME/ftpget.log
mv $HOME/tl$FECHA.cer.zip ../respaldos-tlf
mv $HOME/tl$FECHA.cen.zip ../respaldos-tlf
#
case $DP in
01) DP=1;;
02) DP=2;;
03) DP=3;;
04) DP=4;;
05) DP=5;;
06) DP=6;;
07) DP=7;;
08) DP=8;;
09) DP=9;;
esac
case $MP in
01) MP=1;;
02) MP=2;;
03) MP=3;;
04) MP=4;;
05) MP=5;;
06) MP=6;;
07) MP=7;;
08) MP=8;;
09) MP=9;;
esac
#-----
# BORRA Y CREA TABLA transaccion Y SUS INDICES
#-----
echo "-----">>$HOME/ftpget.log
echo DROPEA TABLA transaccion CON PROGRAMA crea_transac.sh >>$HOME/ftpget.log
date >> $HOME/ftpget.log
cd $HOME2
crea_transac.sh
#-----
# BORRA REGISTRO
# EN TABLA controlt
#-----
echo "DELETE FROM controlt
WHERE control_corte=($FECHAINI)" | dbaccess bdcica
#-----
# CORRE SHELL: carga.sh
#-----
echo "-----">>$HOME/ftpget.log

```

APENDICE G. MANUAL TECNICO

```

echo CARGA REGISTROS DE TLF A TABLA transaccion CON PROGRAMA carga.sh
>>$HOME/ftpget.log
date >> $HOME/ftpget.log
AP=`date +%C`$AP
cd $HOME2
carga.sh $AP $MP $DP $DP >> $HOME/ftpget.log
#-----
# VERIFICA SI SE CARGO
# EL ARCHIVO DE COSTA RICA
#-----
echo "-----">>$HOME/ftpget.log
echo VERIFICA SI SE CARGO TLF DE COSTA RICA >>$HOME/ftpget.log
date >> $HOME/ftpget.log
echo "UNLOAD TO $PROYECTO$DATOS/estatus$FECHA.unl
SELECT bancrecen FROM controlt
WHERE control_corte=($FECHAINI)" | dbaccess bdcica
if grep "S" $PROYECTO$DATOS/estatus$FECHA.unl
then
echo SI SE CARGO ARCHIVO DE COSTA RICA. DIA: $FECHAINI>>$HOME/ftpget.log
else
echo NO SE CARGO ARCHIVO DE COSTA RICA. DIA: $FECHAINI>>$HOME/ftpget.log
echo PARA MAS INFORMACION VER TABLA -control_procesos- EN -mc_cica- >>
$HOME/ftpget.log
echo EL PROCESO SE CANCELARA. >>$HOME/ftpget.log
echo "execute procedure actua_pro('D09',$FECHAINI,'N')" | dbaccess mc_cica
#-----
diasno
#-----
rm $PROYECTO$DATOS/estatus$FECHA.unl
exit
fi
rm $PROYECTO$DATOS/estatus$FECHA.unl
#-----
# VERIFICA SI SE CARGO
# EL ARCHIVO DE MEXICO
#-----
echo "-----">>$HOME/ftpget.log
echo VERIFICA SI SE CARGO TLF DE MEXICO >>$HOME/ftpget.log
date >> $HOME/ftpget.log
echo "UNLOAD TO $PROYECTO$DATOS/estatus$FECHA.unl
SELECT bancrecer FROM controlt
WHERE control_corte=($FECHAINI)" | dbaccess bdcica
if grep "S" $PROYECTO$DATOS/estatus$FECHA.unl
then
echo SI SE CARGO ARCHIVO DE MEXICO. DIA: $FECHAINI>>$HOME/ftpget.log
else
echo NO SE CARGO ARCHIVO DE MEXICO. DIA: $FECHAINI>>$HOME/ftpget.log
echo VERIFIQUE TABLA controlt EN bdcica >>$HOME/ftpget.log
echo PARA MAS INFORMACION VER TABLA -control_procesos- EN -mc_cica- >>
$HOME/ftpget.log
echo EL PROCESO SE CANCELARA. >>$HOME/ftpget.log
echo "execute procedure actua_pro('D18',$FECHAINI,'N')" | dbaccess mc_cica
#-----
diasno
#-----
rm $PROYECTO$DATOS/estatus$FECHA.unl
exit
fi
rm $PROYECTO$DATOS/estatus$FECHA.unl
#-----
# VERIFICA CIFRAS DE CONTROL
#-----
echo "-----">>$HOME/ftpget.log
echo VERIFICA CIFRAS DE CONTROL DE TABLA controlt EN bdcica>>$HOME/ftpget.log
date >> $HOME/ftpget.log
echo "UNLOAD TO $PROYECTO$DATOS/cifras$FECHA.unl

```

APENDICE G. MANUAL TECNICO

```

SELECT status_procesot FROM control
WHERE control_corte=($FECHAINI)" | dbaccess bdcica
if grep "S" $PROYECTOSDATOS/cifras$FECHA.unl
then
echo LAS CIFRAS DE CONTROL SI CHECAN. DIA: $FECHAINI>>$HOME/ftpget.log
echo "execute procedure actua_pro('D21',$FECHAINI,'S')" | dbaccess mc_cica
else
echo LAS CIFRAS DE CONTROL NO CHECAN. DIA: $FECHAINI>>$HOME/ftpget.log
echo VERIFIQUE TABLA controlt EN bdcica >>$HOME/ftpget.log
echo EL PROCESO SE CANCELARA. >>$HOME/ftpget.log
echo "execute procedure actua_pro('D21',$FECHAINI,'N')" | dbaccess mc_cica
#-----
# diasno
#-----
rm $PROYECTOSDATOS/cifras$FECHA.unl
exit
fi
rm $PROYECTOSDATOS/cifras$FECHA.unl
#-----
# GENERA ARCHIVO PLANO DE TABLA transaccion
#-----
echo "-----">>$HOME/ftpget.log
echo GENERA ARCHIVO PLANO transac$FECHA.dat >>$HOME/ftpget.log
date >>$HOME/ftpget.log
echo "UNLOAD TO $HOME1/transac$FECHA.dat
SELECT * FROM transaccion" | dbaccess bdcica
if [ -s $HOME1/transac$FECHA.dat ]
then
echo "execute procedure actua_pro('D22',$FECHAINI,'S')" | dbaccess mc_cica
pkzip $HOME1/transac$FECHA.dat.zip $HOME1/transac$FECHA.dat
if [ -s $HOME1/transac$FECHA.dat.zip ]
then
rm $HOME1/transac$FECHA.dat
mv $HOME1/transac$FECHA.dat.zip ../respaldos-tlf
fi
else
echo "execute procedure actua_pro('D22',$FECHAINI,'N')" | dbaccess mc_cica
#-----
# diasno
#-----
fi
#-----
# EJECUTA STORE PROCEDURE sp_movim EN bdcica
#-----
echo "-----">>$HOME/ftpget.log
echo CARGA REGISTROS DE transaccion A acumulado_mc EN
bdcica>>$HOME/ftpget.log
echo EJECUTA PROCESO sp_movim.sql>>$HOME/ftpget.log
echo FECHA INICIAL: $FECHAINI >>$HOME/ftpget.log
echo FECHA FINAL...: $FECHAINI >>$HOME/ftpget.log
date >>$HOME/ftpget.log
echo "execute procedure sp_movim($FECHAINI,$FECHAFIN)" | dbaccess bdcica
#-----
# ACTUALIZA TABLA geografia EN mc_cica
#-----
echo "-----">>$HOME/ftpget.log
echo ACTUALIZA TABLA geografia EN mc_cica
date >>$HOME/ftpget.log
if [ ! -s $PROYECTOSDATOS/geografia.unl ]
then
echo "NO SE ENCONTRO ARCHIVO: $PROYECTOSDATOS/geografia.unl" >>
$HOME/ftpget.log
echo "PROCESO DE CARGA DE TABLA geografia SE CANCELA">>$HOME/ftpget.log
else
echo EJECUTA PROGRAMA carga_geografia.sh>> $HOME/ftpget.log
cd $HOME2

```

APENDICE G. MANUAL TECNICO

```

carga_geografia.sh
fi
#-----
# VERIFICA PROCESOS ANTERIORES EN TABLA control_procesos
# PARA DECIDIR SI CONTINUA O DETIENE EL PROCESO
#-----
echo "-----">>$HOME/ftpget.log
echo VERIFICA ESTATUS DE PROCESOS ANTERIORES >> $HOME/ftpget.log
echo EN TABLA control_procesos DE mc_cica>> $HOME/ftpget.log
echo CON PROGRAMA veri_control.sh >> $HOME/ftpget.log
date >> $HOME/ftpget.log
cd $HOME2
veri_control.sh $AP $ME $DP 23 "D"
#
if [ -s $PROYECTO$DATOS/est_ctl$AP$ME$DP.unl ]
then
  if grep "NO" $PROYECTO$DATOS/est_ctl$AP$ME$DP.unl
  then
    echo "NO TODOS LOS PROCESOS ESTAN EJECUTADOS CORRECTAMENTE." >>
$HOME/ftpget.log
    echo "PROCESO CANCELADO." >> $HOME/ftpget.log
    rm $PROYECTO$DATOS/est_ctl$AP$ME$DP.unl
    #-----
    diasno
    #-----
    exit
  fi
else
  echo "ARCHIVO $PROYECTO$DATOS/est_ctl$AP$ME$DP.unl NO ENCONTRADO. PROCESO
CANCELADO." >> $HOME/ftpget.log
  #-----
  diasno
  #-----
  exit
fi
rm $PROYECTO$DATOS/est_ctl$AP$ME$DP.unl
#-----
# CARGA REGISTROS EN TABLA movimientos DE mc_cica
#-----
echo "-----">>$HOME/ftpget.log
echo CARGA REGISTROS DE acumulado_mc A movimientos EN
mc_cica>>$HOME/ftpget.log
echo EJECUTA PROCESO carga_fact.sh>>$HOME/ftpget.log
echo FECHA INICIAL: $FECHAINI >>$HOME/ftpget.log
echo FECHA FINAL.: $FECHAINI >>$HOME/ftpget.log
date >>$HOME/ftpget.log
cd $HOME2
carga_fact.sh
#-----
# GRABA EN TABLA "cajeros_no_region" (bdcica) CAJEROS
# BANCRECER QUE NO EXISTEN EN geografia (mc_cica)
#-----
echo "-----">>$HOME/ftpget.log
echo GRABA EN TABLA cajeros_no_region EN bdcica >> $HOME/ftpget.log
echo LOS CAJEROS NO REGIONALIZADOS >> $HOME/ftpget.log
echo PROGRAMA cajeros.sql >> $HOME/ftpget.log
date >> $HOME/ftpget.log
echo "execute procedure cajeros($FECHAINI,$FECHAFIN)" | dbaccess bdcica
#-----
fechas
#-----
rm $HOME1/*err
#-----
# BORRA Y CREA TABLAS transaccion Y transaccion_tmp
#-----
echo "-----">>$HOME/ftpget.log

```

APENDICE G. MANUAL TECNICO

```

echo DROPEA TABLAS transaccion Y transaccion_tmp CON PROGRAMA
crea_transac2.sh >>$HOME/ftpget.log
date >> $HOME/ftpget.log
cd $HOME2
crea_transac2.sh
#-----
# GRABA EN TABLA "resultados", EL RESULTADO DE LA
# ACTUALIZACION DIARIA, QUINCENAL Y/O MENSUAL.
#-----
echo "-----">>$HOME/ftpget.log
echo GRABA EN TABLA resultados EN mc_cica >>$HOME/ftpget.log
echo EL RESULTADOS DE LOS PROCESOS DIARIOS>>$HOME/ftpget.log
echo PROGRAMA veri_proceso.sql>>$HOME/ftpget.log
date >> $HOME/ftpget.log
echo "execute procedure veri_proceso($FECHAINI,'D')" | dbaccess mc_cica
#-----
# ACTUALIZA TABLA tiempo EN mc_cica
#-----
echo "-----">>$HOME/ftpget.log
echo ACTUALIZA TABLA tiempo EN mc_cica >>$HOME/ftpget.log
echo PROGRAMA actua_tie.sh>>$HOME/ftpget.log
date >> $HOME/ftpget.log
cd $HOME2
actua_tie.sh $AP $ME $DP
#-----
# LLAMA ACTUALIZACION DE AGREGADOS, COSTOS Y
# ARCHIVO PARA FASE EJECUTIVA Y OPERATIVA
#-----
echo "-----">>$HOME/ftpget.log
echo LLAMA ACTUALIZACION DE AGREGADOS, COSTOS Y >>$HOME/ftpget.log
echo ARCHIVOS PARA FASE EJECUTIVA Y OPERATIVA >>$HOME/ftpget.log
echo PROGRAMA actua_agre.sh>>$HOME/ftpget.log
date >> $HOME/ftpget.log
cd $HOME2
actua_agre.sh $AP $ME $DP
#-----
else
echo "-----">>$HOME/ftpget.log
echo NO EXISTE ARCHIVO $HOME1/t1$FECHA.cen.zip >> $HOME/ftpget.log
date >> $HOME/ftpget.log
echo "execute procedure actua_pro('D01',$FECHAINI,'N')" |dbaccess mc_cica
#-----
diasno
#-----
exit
fi
else
echo "-----">>$HOME/ftpget.log
echo NO EXISTE ARCHIVO $HOME1/t1$FECHA.cer.zip >> $HOME/ftpget.log
date >> $HOME/ftpget.log
echo "execute procedure actua_pro('D01',$FECHAINI,'N')" |dbaccess mc_cica
#-----
diasno
#-----
exit
fi
#-----
# VERIFICA SI LA FECHA ESTA #
# EN TABLA -diasno- PARA #
# BORRAR ESE REGISTRO #
#-----
ANIO_F=`echo $FECHAINI|cut -c2-5`
MESD_F=`echo $FECHAINI|cut -c7-8`
DIA_F=`echo $FECHAINI|cut -c10-11`
echo "DELETE FROM diasno
WHERE dias_no_proces = $ANIO_F$MESD_F$DIA_F;

```

```

UNLOAD TO $HOME/diasno
SELECT * FROM diasno;"|dbaccess bdcica
-----#
# CREA COPIA DE ftpget.log #
-----#
cName=ftpget$FECHA.log
cFile=$PROYECTO$DATOS/$cName
cp -f $HOME/ftpget.log $cFile
mv $cFile ../logs

```

11.2 LISTADO DE PROCEDIMIENTO ALMACENADO

A continuación se presenta uno de los procedimientos almacenados más representativos del CICA.

• sp_movim.sql

```

DROP PROCEDURE sp_movim;
CREATE PROCEDURE sp_movim(dFechaInicio char(10), dFechaCierre char(10))

-- Procedimiento cargar de Transaccion a acumulado_mc
-- Autor: Luis Vargas
-- Fecha: 14/07/1998
-- Descripción: Utiliza el criterio de los 7 campos para agrupar registros,
--              mientras sea el mismo bloque de 7 campos va acumulando tanto
--              aceptadas como rechazadas.
--              Gracias al análisis realizado por Mario Millan con una tabla
--              de verdad se valida primero si el tipo de movimiento es retiro, a
--              continuación si es aceptada o rechazada y al final si es reverso
--              total o parcial.
--
-- Nota: Debido a que las operaciones que implican reversos vienen en parejas,
--       el primer elemento parece una transaccion aceptada (por lo que se agrega al
--       numero e importe de aceptadas) pero el segundo elemento es el que indica
--       (por el mensaje=420) que es un reverso (total cuando ajuste=0) por lo que
--       se debe descontar 1 y el importe del reverso a los acumuladores de numero e
--       importe de aceptadas respectivamente y agregarse al acumulador de numero e
--       importe de rechazadas.
--
-- Modificación: 23 de julio de 1998
-- Autor.....: Cesar Salas A.
-- Descripción.: Se agrego el registro de control de procesos
--
-- Modificación: 17 de Octubre de 1998
-- Autor: Luis Vargas
-- Descripción: Debido a que hay cajeros de Visa que tienen la misma clave que
--              los de Bancrecer, cuando se detecta alguno se le agrega a la
--              clave de cajero la del banco diferente a Bancrecer.
--
-- Modificación: 22 de Octubre de 1998
-- Autor: Cesar U. Salas Araujo
-- Descripción: Se cambio la longitud de las variables "sCajero" y "sCajeroTmp":
--              tenían char de 8 y se cambiaron a char de 16.
--
-- Modificación: 22 de Octubre de 1998
-- Autor: Cesar U. Salas Araujo
-- Descripción: Se aplica la función TRIM a la variable "sCajeroTmp"
--              al formar la clave de cajero compuesta (cajero + banco)
--              para quitar los espacios a la clave del cajero.
--
-- Definición de variables
DEFINE sCajero      char(16);
DEFINE sCajeroTmp  char(16);

```

APENDICE G. MANUAL TECNICO

```

DEFINE sSucursal      char(4);
DEFINE sPlaza         char(4);
DEFINE sRegion        char(4);
DEFINE sBanco         char(4);
DEFINE sBancoTmp      char(4);
DEFINE sRazon         char(2);
DEFINE sRazonTmp      char(2);
DEFINE sCuenta        char(2);
DEFINE sCuentaTmp     char(2);
DEFINE sBancoTarjeta  char(4);
DEFINE sBancoTarjetaTmp char(4);
DEFINE sTransaccion   char(12);
DEFINE nMovimiento    numeric(2);
DEFINE nMovimientoTmp numeric(2);
DEFINE nMoneda        numeric(3);
DEFINE nMonedaTmp     numeric(3);
DEFINE nMensaje       numeric(4);
DEFINE fImporte       float;
DEFINE fAjuste        float;
DEFINE nAceptadas     integer;
DEFINE nRechazadas    integer;
DEFINE nReverso       integer;
DEFINE nIndefinido    integer;
DEFINE nccRegistros   integer;
DEFINE nccAceptadas   integer;
DEFINE nccRechazadas  integer;
DEFINE nccReverso     integer;
DEFINE nccIndefinido  integer;
DEFINE fAceptadas     float;
DEFINE fRechazadas    float;
DEFINE fcciAceptadas  float;
DEFINE fcciRechazadas float;
DEFINE siMovtos       smallint;
DEFINE siCuentas      smallint;
DEFINE siMovto        smallint;
DEFINE siCuenta       smallint;
DEFINE siPrimeraVez   smallint;
DEFINE dCorte         date;
DEFINE dFechaI       date;
DEFINE dFechaF       date;
DEFINE siUbicacion    smallint;
DEFINE sSector        char(1);
DEFINE cStatus        char(1);
DEFINE mCosto_ca      money(12,4);
DEFINE mCostotr       money(12,4);
DEFINE mCostcnx       money(12,4);
DEFINE mCostred       money(12,4);
DEFINE iPunto_equi    integer;
DEFINE dCorteTmp      date;
DEFINE NoScer        char(1);
-- Definicion de constantes
DEFINE NULL_INTEGER  numeric(1);
DEFINE RECHAZO_VAL   numeric(3);
DEFINE RECHAZO_VAL2  numeric(4);

-- Asignacion de constantes
LET NULL_INTEGER = 0;
LET RECHAZO_VAL = 420;
LET RECHAZO_VAL2 = 9980;

-- Inicializacion de Variables
LET nAceptadas = 0;
LET nRechazadas = 0;
LET nReverso = 0;
LET nIndefinido = 0;
LET fAceptadas = 0;

```

APENDICE G. MANUAL TECNICO

```

LET fRechazadas      = 0;
LET siPrimeraVez    = 0;
LET sBancoTmp       = "";
LET sCajeroTmp      = "";
LET sCuentaTmp      = "";
LET sBancoTarjetaTmp = "";
LET nMovimientoTmp  = "";
LET nMonedaTmp      = "";
LET sRazon          = " ";
LET sRazonTmp       = " ";
LET cStatus         = "";
LET mCosto_ca       = 0;
LET mCosto_tr       = 0;
LET mCosto_cn       = 0;
LET mCosto_red      = 0;
LET iPunto_equi     = 0;
LET nccRegistros    = 0;
LET nccAceptadas    = 0;
LET nccRechazadas   = 0;
LET nccReverso      = 0;
LET nccIndefinido   = 0;
LET fcciAceptadas   = 0;
LET fcciRechazadas  = 0;
LET NoBcer          = "";

FOREACH Cajeros_Cursor FOR
-- El cursor agrupa por los 7 campos dentro del sig. select:
SELECT tr.banco_id, tr.cajero_id, tr.cuenta_id, tr.banco_tarjeta,
tr.movimiento_id, tr.moneda_id, tr.razon_id, tr.mensaje_id, tr.transaccion_corte,
tr.transaccion_import, tr.transaccion_ajuste
INTO   sBanco, sCajero, sCuenta, sBancoTarjeta,
nMovimiento, nMoneda, sRazon, nmensaje, dCorte, fImporte, fAjuste
FROM   transaccion tr
WHERE  transaccion_corte >= date(dFechaInicio) AND
transaccion_corte <= date(dFechaCierre)
ORDER BY 1,2,3,4,5,6,7

-- Cuenta Registros Leidos
LET nccRegistros = nccRegistros + 1;

-- Constantes para los cajeros que no esten regionalizados
LET sSucursal = "0000";
LET sPlaza    = "0000";
LET sRegion   = "0000";

-- Solo se hace una vez
IF ( siPrimeraVez = 0 ) THEN
LET sBancoTmp      = sBanco;
LET sCajeroTmp     = sCajero;
LET sCuentaTmp     = sCuenta;
LET sBancoTarjetaTmp = sBancoTarjeta;
LET nMovimientoTmp = nMovimiento;
LET nMonedaTmp     = nMoneda;
LET sRazonTmp      = sRazon;
LET dCorteTmp      = dCorte;

LET siPrimeraVez=1;
END IF

-- Bloque para acumular por banco_id (banco dueño del cajero), cajero_id,
-- cuenta_id (tipo de cuenta), banco_tarjeta_id (banco emisor de tarjeta),
-- movimiento_id (tipo de movimiento), moneda_id y razon_id.
--
-- Si es la misma llave va acumulando aceptadas y rech.
IF (sBancoTmp      = sBanco      AND

```

APENDICE G. MANUAL TECNICO

```

sCajeroTmp      = sCajero      AND
sCuentaTmp     = sCuenta      AND
sBancoTarjetaTmp = sBancoTarjeta AND
nMovimientoTmp = nMovimiento  AND
nMonedaTmp     = nMoneda      AND
sRazonTmp      = sRazon)
THEN
--
-- Inicia bloque para validar el tipo de transaccion
-- y si fue aceptada o no.
--
-- Retiros
IF nMovimiento = '10' THEN
-- Si no se acepto el retiro acumula rechazadas
IF sRazon not in ("00","01") THEN
LET nRechazadas = nRechazadas + 1;
LET fRechazadas = fRechazadas + fImporte;
ELSE
-- Si se acepto el retiro debe verificar los reversos
IF nMensaje = RECHAZO_VAL THEN -- Si es 420 (Reverso)
IF fAjuste = NULL_INTEGER THEN --Reverso Total
LET nAceptadas = nAceptadas - 1;
LET fAceptadas = fAceptadas - fImporte;
LET nRechazadas = nRechazadas + 1;
LET fRechazadas = fRechazadas + fImporte;
ELSE
-- Reverso Parcial
LET fAceptadas = (fAceptadas - fImporte) + fAjuste;
LET fRechazadas = fRechazadas + (fImporte - fAjuste);
END IF
ELSE
-- No hay reverso
IF nMensaje <> RECHAZO_VAL2 THEN
LET nAceptadas = nAceptadas + 1;
LET fAceptadas = fAceptadas + fImporte;
END IF
END IF
END IF
ELSE
-- Diferente a retiros
-- Si no se acepto la transaccion acumula rechazadas
IF sRazon not in ("00","01") THEN
LET nRechazadas = nRechazadas + 1;
ELSE
-- Para los que no son retiros acumula las aceptadas
LET nAceptadas = nAceptadas + 1;
END IF
END IF
--
-- Fin del bloque para validar tipo de transaccion
-- y si fue aceptada o no.
--
ELSE
--
-- Si es otra llave inserta los datos que acumulo (aceptados y rech.)
--
-- Para cajeros de BanCreceer tiene que regionalizarlos
-- e identificar su ubicacion y sector
SELECT su.sucursal_id, su.plaza_id, su.region_id
INTO sSucursal, sPlaza, sRegion
FROM cajero ca, sucursal su
WHERE ca.sucursal_id = su.sucursal_id
AND ca.cajero_id = sCajeroTmp;
LET NoBcer = 'F';
IF sSucursal IS NULL THEN
LET NoBcer = 'T';

```

APENDICE G. MANUAL TECNICO

```

    LET sSucursal = '00000';
END IF
IF sPlaza IS NULL THEN
    LET NoBcer = 'T';
    LET sPlaza = '00000';
END IF
IF sRegion IS NULL THEN
    LET NoBcer = 'T';
    LET sRegion = '00000';
END IF
-- Si el banco dueño del cajero no es Bancrecer pero el cajero
-- si se encontro en el catalogo de cajero (caso VISA)
-- se le agrega a la clave del cajero la clave del banco
IF sBancoTmp <> 'B161' AND NoBcer = 'F' THEN
    LET sCajeroTmp = TRIM(sCajeroTmp) || sBancoTmp;
END IF
-- identifica la ubicacion y el sector para el cajero de acuerdo
-- al banco que tenga registrado en el catalogo de cajero
-- (para cajeros que no sean Bancrecer no debe existir ninguno
-- de los datos)
SELECT ca.Ubicacion_id,u.Sector_id,s.costo_ca,s.punto_equi
INTO siUbicacion, sSector, mCosto_ca,iPunto_equi
FROM cajero ca, Ubicacion u, Sector s
WHERE ca.Ubicacion_id = u.Ubicacion_id
AND ca.cajero_id = sCajeroTmp
AND u.Sector_id = s.Sector_id
AND ca.banco_id = sBancoTmp;
IF siUbicacion IS NULL THEN
    LET siUbicacion = '0';
END IF
IF sSector IS NULL THEN
    LET sSector = ' ';
    LET mCosto_ca = 0;
    LET iPunto_equi = 0;
END IF
INSERT INTO acumulado_mc(razon_id_acum,
                        cajero_id_acum,
                        sucursal_id_acum,
                        plaza_id_acum,
                        region_id_acum,
                        banco_id_acum,
                        banco_tarj_id_acum,
                        movimiento_id_acum,
                        cuenta_id_acum,
                        moneda_id_acum,
                        aceptadas_num,
                        rechazadas_num,
                        importe_aceptadas,
                        importe_rechazadas,
                        Ubicacion_id_acum,
                        Sector_id_acum,
                        Costo_ca_acum,
                        Costo_tr_acum,
                        Costo_cn_x_acum,
                        Costo_red_acum,
                        Punto_equi_acum,
                        fecha_Corte)
VALUES (sRazonTmp,
        sCajeroTmp,
        sSucursal,
        sPlaza,
        sRegion,
        sBancoTmp,
        sBancoTarjetaTmp,

```

APENDICE G. MANUAL TECNICO

```

nMovimientoTmp,
sCuentaTmp,
nMonedaTmp,
nAceptadas,
nRechazadas,
fAceptadas,
fRechazadas,
siUbicacion,
sSector,
mCosto_ca,
mCostotr,
mCostcnx,
mCostred,
iPunto_equi,
dCorteTmp);
--Limpia las variables
LET nAceptadas = 0;
LET nRechazadas = 0;
LET fAceptadas = 0;
LET fRechazadas = 0;

-- Actualiza las variables temporales a donde apunta el cursor
LET sBancoTmp = sBanco;
LET sCajeroTmp = sCajero;
LET sCuentaTmp = sCuenta;
LET sBancoTarjetaTmp = sBancoTarjeta;
LET nMovimientoTmp = nMovimiento;
LET nMonedaTmp = nMoneda;
LET sRazonTmp = sRazon;
LET dCorteTmp = dCorte;

rech. -- Antes de leer el siguiente registro debe almacenar las aceptadas y
-- de donde esta el cursor
-- Inicia bloque para validar el tipo de transaccion
-- y si fue aceptada o no.
--
-- Retiros
IF nMovimiento = '10' THEN
-- Si no se acepto el retiro acumula rechazadas
IF sRazon not in ("00","01") THEN
LET nRechazadas = nRechazadas + 1;
LET fRechazadas = fRechazadas + fImporte;
ELSE
-- Si se acepto el retiro debe verificar los reversos
IF nMensaje = RECHAZO_VAL THEN -- Si es 420 (Reverso)
IF fAjuste = NULL_INTEGER THEN --Reverso Total
LET nAceptadas = nAceptadas - 1;
LET fAceptadas = fAceptadas - fImporte;
LET nRechazadas = nRechazadas + 1;
LET fRechazadas = fRechazadas + fImporte;
ELSE
-- Reverso Parcial
LET fAceptadas = (fAceptadas - fImporte) + fAjuste;
LET fRechazadas = fRechazadas + (fImporte - fAjuste);
END IF
ELSE
-- No hay reverso
IF nMensaje <> RECHAZO_VAL2 THEN
LET nAceptadas = nAceptadas + 1;
LET fAceptadas = fAceptadas + fImporte;
END IF
END IF
ELSE
-- Diferente a retiros

```

APENDICE G. MANUAL TECNICO

```

-- Si no se acepto la transaccion acumula rechazadas
IF sRazon not in ("00","01") THEN
    LET nRechazadas = nRechazadas + 1;
ELSE
    -- Para los que no son retiros acumula las aceptadas
    LET nAceptadas = nAceptadas + 1;
END IF
END IF
--
-- Fin del bloque para validar tipo de transaccion
-- y si fue aceptada o no.
END IF

END FOREACH;

--
-- Debido a que los datos del ultimo bloque no se insertan se agrega el sig.
codigo
--
SELECT su.sucursal_id, su.plaza_id, su.region_id
INTO sSucursal, sPlaza, sRegion
FROM cajero ca, sucursal su
WHERE ca.sucursal_id = su.sucursal_id
AND ca.cajero_id = sCajeroTmp;
LET NoBcer = 'F';
IF sSucursal IS NULL THEN
    LET NoBcer = 'T';
    LET sSucursal = '00000';
END IF
IF sPlaza IS NULL THEN
    LET NoBcer = 'T';
    LET sPlaza = '00000';
END IF
IF sRegion IS NULL THEN
    LET NoBcer = 'T';
    LET sRegion = '00000';
END IF
-- Si el banco duenio del cajero no es Bancrecer pero el cajero
-- el cajero si se encontro en el catalogo de cajero (caso VISA)
-- se le agrega a la clave del cajero la clave del banco
IF sBancoTmp <> 'B161' AND NoBcer = 'F' THEN
    LET sCajeroTmp = TRIM(sCajeroTmp) || sBancoTmp;
END IF
-- identifica la ubicacion y el sector para el cajero de acuerdo
-- al banco que tenga registrado en el catalogo de cajero
-- (para cajeros que no sean Bancrecer no debe existir ninguno
-- de los datos)
SELECT ca.Ubicacion_id, u.Sector_id, s.costo_ca, s.punto_equi
INTO siUbicacion, sSector, mCosto_ca, iPunto_equi
FROM cajero ca, Ubicacion u, Sector s
WHERE ca.Ubicacion_id = u.Ubicacion_id
AND ca.cajero_id = sCajeroTmp
AND u.Sector_id = s.Sector_id
AND ca.banco_id = sBancoTmp;
IF siUbicacion IS NULL THEN
    LET siUbicacion = '0';
END IF
IF sSector IS NULL THEN
    LET sSector = ' ';
    LET mCosto_ca = 0;
    LET iPunto_equi = 0;
END IF

-- Aqui se inserta el ultimo bloque de la tabla
INSERT INTO acumulado_mc(razon_id_acum,
                        cajero_id_acum,
```

APENDICE G. MANUAL TECNICO

```

sucursal_id_acum,
plaza_id_acum,
region_id_acum,
banco_id_acum,
banco_tarj_id_acum,
movimiento_id_acum,
cuenta_id_acum,
moneda_id_acum,
aceptadas_num,
rechazadas_num,
importe_aceptadas,
importe_rechazadas,
Ubicacion_id_acum,
Sector_id_acum,
Costo_ca_acum,
Costotr_acum,
Costcnx_acum,
Costred_acum,
Punto_equi_acum,
fecha_Corte)

VALUES (sRazonTmp,
sCajeroTmp,
sSucursal,
sPlaza,
sRegion,
sBancoTmp,
sBancoTarjetaTmp,
nMovimientoTmp,
sCuentaTmp,
nMonedaTmp,
nAceptadas,
nRechazadas,
fAceptadas,
fRechazadas,
siUbicacion,
sSector,
mCosto_ca,
mcostotr,
mcostcnx,
mcostred,
iPunto_equi,
dCorte);
-- Acumula las Variables para Cifras de Control
LET nccAceptadas = nccAceptadas + nAceptadas;
LET nccRechazadas = nccRechazadas + nRechazadas;
LET nccReverso = nccReverso + nReverso;
LET nccIndefinido = nccIndefinido + nIndefinido;
LET fcciAceptadas = fcciAceptadas + fAceptadas;
LET fcciRechazadas = fcciRechazadas + fRechazadas;

--
EXECUTE PROCEDURE mc_cica:actua_pro('D23',dfechaInicio,'S');
--
END PROCEDURE;

```

BIBLIOGRAFIA

"The Data Warehouse Lifecycle Toolkit"

Ralph Kimball
Wiley
Estados Unidos
1998

"The Data Warehouse Toolkit"

Ralph Kimball
Wiley
Estados Unidos
1996

"INFORMIX MetaCube – Explorer User's Guide Version 3.1"

Informix Software, Inc.
Estados Unidos
Mayo 1997

"INFORMIX MetaCube – Warehouse Manager's Guide Version 3.1"

Informix Software, Inc.
Estados Unidos
Mayo 1997

"MetaCube v2.1 Technical Training"

Informix Software, Inc.
Estados Unidos
1997

"An Introduction to Data Warehousing"

Informix Software, Inc.
Estados Unidos
1997

"Structured Techniques the basis for CASE"

James Martin
Prentice Hall
Estados Unidos
1988

"Análisis Estructurado Moderno"

Edward Yourdon
Prentice Hall
México
1988

"Software Engineering: A Practitioner's approach"

Roger S. Pressman
Mac Graw Hill
Estados Unidos
1984