

UNIVERSIDAD NACIONAL DE INGENIERIA
FACULTAD DE INGENIERIA INDUSTRIAL Y DE SISTEMAS



**Análisis y Diseño de los componentes de
Inteligencia de Negocios de una solución CRM**

TESIS

**Para optar por el Título Profesional de
Ingeniero de Sistemas**

Eduardo Málaga Chocano
Luis Guillermo Rodríguez Aguayo

914030A 96 - I
924040J 96 - II

LIMA – PERU
2002

*A nuestros padres,
quienes con su amor,
dedicación, apoyo y ejemplo
nos ayudan cada día a alcanzar
nuestras metas.*

INDICE

DESCRIPTORES TEMÁTICOS.....	4
1. RESUMEN EJECUTIVO.....	5
2. INTRODUCCIÓN.....	9
2.1. DEFINICIÓN Y PLANTEAMIENTO DEL PROBLEMA	11
2.2. OBJETIVO Y ALCANCE DEL ESTUDIO.....	13
2.2.1 Objetivo General	13
2.2.2 Objetivos Específicos	13
2.2.3 Alcances	14
2.3. IMPORTANCIA DEL TEMA	15
3. FUNDAMENTOS TEÓRICOS.....	16
3.1. DEFINICIÓN DE CRM	16
3.1.1 La visión de CRM	17
3.1.2 La teoría del intercambio de Valor	18
3.2. EL CICLO CRM	20
3.2.1 Identificando al Cliente	21
3.2.2 Diferenciación.....	22
3.2.3 Aprendiendo a través de la interacción del cliente.....	25
3.2.4 Personalizando la oferta	26
3.3. BENEFICIOS DE CRM	28
3.4. ARQUITECTURA TÉCNICA DE UNA SOLUCIÓN CRM.....	29
3.5. INTELIGENCIA DE NEGOCIOS	30
3.5.1 Data Warehouse	31
3.5.2 Diferencia entre un Data Warehouse y los Sistemas OLTP	34
3.5.3 Componentes de un entorno de Inteligencia de Negocios	36
3.5.3.1 Sistemas Fuente (Legacy Systems).....	36
3.5.3.2 Área de Escenificación de Datos (Data Staging Area).....	37
3.5.3.3 Data Mart	38
Análisis y Diseño de los componentes de Inteligencia de Negocios de una solución CRM	1

3.5.3.4	Metadata	42
3.5.3.5	Enterprise Data Warehouse (EDW)	43
3.5.3.6	Operational Data Store (ODS)	43
3.5.4	Procesos	45
3.5.4.1	Minería de Datos	45
3.5.5	Diseño y modelamiento	46
3.5.5.1	Modelamiento dimensional	46
3.5.5.2	Tabla de hechos (Fact Table)	48
3.5.5.3	Tabla de dimensión (Dimension Table).....	48
3.5.5.4	Esquema Estrella	51
3.5.5.5	Esquema SnowFlake.....	52
3.5.6	Extracción, Transformación y Carga	55
3.5.6.1	Extracción	55
3.5.6.1.1	Múltiples Fuentes.....	56
3.5.6.1.2	Generación de Código	56
3.5.6.1.3	Tipos de Extracción	57
3.5.6.2	Transformación	58
3.5.6.3	Carga de Datos	60
3.5.7	Acceso y Explotación de la Información	61
3.5.8	Calidad y Limpieza de Datos	64
3.5.8.1	Costo de una pobre Calidad de Datos	65
3.5.8.2	Mejorando la Calidad de Datos.....	65
3.5.9	Ciclo de vida de un proyecto de Inteligencia de Negocios	66
3.5.9.1	Planeamiento del Proyecto	68
3.5.9.2	Definición de los requerimientos del negocio	69
3.5.9.3	Modelamiento Dimensional.....	69
3.5.9.4	Diseño Físico	70
3.5.9.5	Extracción, Transformación y Carga	70
3.5.9.6	Diseño de la Arquitectura Técnica	70
3.5.9.7	Selección del Producto e Instalación	71
3.5.9.8	Especificación de la Aplicación del Usuario Final.....	71
3.5.9.9	Desarrollo de la Aplicación del Usuario Final	71
3.5.9.10	Despliegue	72
3.5.9.11	Mantenimiento y Crecimiento.....	72
3.5.9.12	Administración del Proyecto	73
4.	DESARROLLO DEL CASO DE ESTUDIO	74
4.1.	DEFINICIÓN DE LOS REQUERIMIENTOS DEL NEGOCIO	75

4.2.	MODELAMIENTO DIMENSIONAL.....	76
4.2.1	Definición de dimensiones.....	77
4.2.1.1	Dimensión Persona.....	77
4.2.1.2	Dimensión Producto.....	80
4.2.1.3	Dimensión Cliente.....	81
4.2.1.4	Dimensión Transacción.....	82
4.2.1.5	Dimensión Tiempo.....	82
4.2.2	Definición de indicadores.....	83
4.2.2.1	Indicadores de Clientes.....	83
4.2.2.2	Indicadores de Productos.....	85
4.2.2.3	Indicadores de Transacción.....	87
4.3.	DISEÑO FÍSICO.....	88
4.4.	EXTRACCIÓN, TRANSFORMACIÓN Y CARGA.....	88
4.5.	CALIDAD DE DATOS.....	90
4.6.	DISEÑO DE LA ARQUITECTURA TÉCNICA.....	91
4.7.	SELECCIÓN DEL PRODUCTO E INSTALACIÓN.....	93
4.8.	ESPECIFICACIÓN Y DESARROLLO DE LA APLICACIÓN DEL USUARIO FINAL	94
5.	CONCLUSIONES.....	115
6.	RECOMENDACIONES.....	118

BIBLIOGRAFÍA

ANEXOS

Anexo A: Descripción de las Tablas y Columnas empleadas en el Modelo Propuesto

Anexo B: Diagramas Entidad-Relación del Modelo CRM Propuesto

Anexo C: Script de creación de la base de datos en DB2 Universal Database

Anexo D: Análisis Comparativo de Manejadores de Bases de Datos y Herramientas OLAP del Mercado

Anexo E: Manual del usuario de la aplicación OLAP desarrollada

DESCRIPTORES TEMÁTICOS

1. CRM: Customer Relationship Management
2. Cliente
3. Ciclo de Vida del Cliente
4. Rentabilidad del Cliente
5. Segmentación
6. Retención, Deserción e Incorporación de Clientes
7. Marketing 1 a 1
8. OLTP : Procesamiento Transaccional en Línea
9. OLAP : Procesamiento Analítico en Línea
10. MOLAP: Multidimensional OLAP
11. ROLAP: Relational OLAP
12. Data Mining
13. Data Mart
14. Data Warehouse
15. Modelamiento Multidimensional
16. Modelo Estrella (Star Schema)
17. Modelo Copo de Nieve (Snowflake Schema)
18. Dimensiones
19. Atributos
20. Indicadores (Medidas)

1. RESUMEN EJECUTIVO

El presente trabajo de tesis tienen por objetivo explicar el proceso de análisis y diseño de los principales componentes de inteligencia de negocios (Business Intelligence, BI) empleados como base para el análisis de información de una estrategia CRM en una institución financiera. Con este fin se lleva a cabo el desarrollo de una prueba de concepto a partir de una muestra de datos tomada de un banco privado del mercado financiero peruano.

Las prácticas de inteligencia de negocios representan una alternativa tecnológica y de administración de negocios que cubren los aspectos del manejo de información para la toma de decisiones, desde su extracción de los sistemas operacionales de una empresa, depuración, transformación, carga, el diseño de estructuras de datos o modelos especiales para el almacenamiento de los datos, hasta la explotación de la información mediante herramientas comerciales de fácil uso para los usuarios de negocios. Los diferentes conceptos de inteligencia de negocios están muy ligados con el concepto de un Data Warehouse. Las estrategias de inteligencia de negocio le permiten a una empresa tener a la mano la información más relevante de su operación de una forma rápida, flexible y oportuna. Las herramientas que integran la inteligencia de negocios consolidan la información de las distintas áreas, mejorando el proceso de decisión.

Customer Relationship Management (CRM), por otro lado, es la estrategia de negocio para seleccionar y gestionar las mejores relaciones con los clientes, identificarlos, diferenciarlos, interactuar con ellos y aprender de su comportamiento para finalmente personalizar la oferta de productos y servicios con el fin de atraer, retener y desarrollar a los clientes más valiosos y

rentables. CRM requiere una filosofía de negocio centrada en el cliente y un conjunto de soluciones de tecnología de la información, tanto operacionales como analíticas, para dar soporte eficiente a los procesos de marketing, ventas y servicios, y medir el valor de la relación con los clientes. El componente analítico de una estrategia CRM, conocido como CRM Analítico, es soportado por inteligencia de negocios. Un CRM Analítico busca conocer a los clientes y la inteligencia de negocios ofrece la colección de datos sobre los clientes representado en un modelo particular, así como herramientas de análisis y explotación de la información. El presente estudio se centra en los componentes de inteligencia de negocios que forman parte de un CRM Analítico.

El enfoque para abordar este tema parte de la identificación de las preguntas de negocios orientadas al valor de los clientes que hacen los altos ejecutivos de las instituciones financieras cuando buscan calificar y cuantificar la relación que tienen con sus clientes. Usualmente, un banco bien posicionado en nuestro medio ya tiene definidos y calculados algunos indicadores claves sobre el valor de la relación con sus clientes, siendo uno de los más importantes la rentabilidad. Partiendo de la rentabilidad ya calculada para cada cliente, un banco se interesa por identificar a ese 20% de los clientes que representan el 80% de la rentabilidad total de la institución, por conocer cuáles son las variables que mejor explican la rentabilidad de un cliente, cuáles son los clientes que tienen una rentabilidad potencial que todavía no ha sido explotada, cómo calcular esa rentabilidad potencial, qué clientes nos interesa retener y cuáles no, cuál es el ratio de deserción y de incorporación anual de los clientes, cómo evoluciona la rentabilidad de los clientes durante el ciclo de vida de su relación con el banco, cómo puede el banco incrementar su rentabilidad, etc. La respuesta a estas preguntas usualmente no puede ser dada directamente desde las aplicaciones operacionales que manejan la relación con el cliente en los diferentes puntos de contacto que un banco tiene (ventanilla, cajeros automáticos, internet, etc.). El presente trabajo propone técnicas y herramientas de inteligencia de negocios para el desarrollo e implantación de un CRM Analítico que responda estas preguntas de negocios,

tomando como referencia para el caso de estudio una muestra de datos significativa de un banco importante de nuestro medio.

El desarrollo del caso de estudio propone un modelo de datos, implantado en una base de datos relacional, que al ser poblado con una muestra representativa (1% del total de la información de clientes del banco) de 13 meses consecutivos de información real de un banco, pueda ser explotado a través de aplicaciones analíticas que presenten la información en un formato gráfico y con una interfaz web que permita dar respuesta a las preguntas planteadas en el párrafo anterior con respecto a la rentabilidad de los clientes y su relación con el banco. Estas aplicaciones analíticas se han desarrollado con el uso de herramientas de inteligencia de negocios conocidas como herramientas OLAP (herramientas de procesamiento analítico en línea), que tienen la característica de presentar la información al usuario final como un grupo de indicadores y dimensiones con el fin de facilitar la formulación de consultas y ocultar su complejidad, de modo que el usuario pueda plantear sus preguntas en los mismo términos de negocios que los hace en su día a día dentro de una organización. Para el desarrollo del caso se sigue una metodología ampliamente difundida en proyectos de inteligencia de negocios, conocida como el "Ciclo de Vida Dimensional del Negocio" y que fue planteada por Ralph Kimball, uno de los precursores de los conceptos de inteligencia de negocios de mediados de la década de los 90.

El sustento y presentación del caso de estudio está precedido de un conjunto de definiciones teóricas que buscan aclarar los diferentes conceptos existentes alrededor del CRM y la inteligencia de negocios y que ayudan a entender a cabalidad los componentes de un CRM Analítico para una institución financiera, que se presenta como prueba de concepto del caso de estudio. Con este fin se plasman con claridad definiciones de CRM, la visión de CRM, la teoría del intercambio de valor, el ciclo CRM, beneficios de CRM y la arquitectura técnica de una solución CRM. Este último punto permite asociar el CRM con los conceptos de inteligencia de negocios y de tecnología de la información, para luego incluir definiciones de Data Warehouse, diferencias con los sistemas transaccionales, componentes de una solución

de inteligencia de negocios (fuentes de datos, Data Mart, Data Warehouse corporativo, metadata, ODS), minería de datos, modelamiento dimensional, tareas de extracción, transformación y carga, calidad y limpieza de datos, explotación de la información y el ciclo de vida de un proyecto de inteligencia de negocios desde el punto de vista de Ralph Kimball.

Como resultado del caso de estudio se pudo identificar que un 17% de los clientes aportaba el 80% de la rentabilidad total de la muestra analizada. Sobre este grupo de clientes el banco debería focalizarse para llevar a cabo campañas de retención y en la medida de lo posible buscar un incremento en su rentabilidad. En el otro lado, del total de clientes que daban pérdidas (rentabilidad negativa) se pudo ver que el 1% de estos clientes aportaba el 80% del total de pérdidas y que por tratarse de un grupo reducido de clientes pueden ser identificados fácilmente ya sea para buscar una mejora en su rentabilidad o para terminar la relación si una mejora es poco probable. Además al momento de realizar un análisis de distribución se identificó que el segmento conocido como "Banca Exclusiva" era el más rentable para el banco, pero que a su vez incluía muchos clientes con una rentabilidad muy baja, lo que podía sugerir una revisión del proceso que asigna los clientes a un segmento en particular para considerar algunas variables más restrictivas y evitar que clientes poco rentables vayan a grupos de alta rentabilidad. En cuanto al ratio de deserción se encontró que éste era del 20% anual, por lo que se podría estimar que el período de vida de la relación de un cliente con el banco es de unos 5 años, mientras que el ratio de incorporación está por el orden del 29%, lo que le da al banco una ratio neto de incorporación anual de 9%. La sección dedicada al caso de estudio y a las conclusiones brinda más detalle sobre los resultados encontrados.

Finalmente esperamos que el presente informe de tesis entregue un aporte importante para conocer y difundir el uso de las técnicas y las herramientas de inteligencia de negocios y su relación con una estrategia CRM, demostrando a través de un caso de estudio como contribuyen de una manera práctica y factible al análisis de información que apoya la gestión de la relación con los clientes en una institución financiera de nuestro medio.

2. INTRODUCCIÓN

Las corrientes de globalización obligan a las empresas hoy en día a esforzarse cada vez más para expandir, conservar y explotar sus segmentos de mercado. En este esfuerzo, es crítico para cada empresa conocer bien su segmento objetivo, lo que involucra un conocimiento profundo del comportamiento de los clientes y sus potencialidades. Esta regla se hace más evidente todavía en una empresa del sector financiero, donde el cliente constituye la entidad más importante para el negocio. Sin embargo, esta visión no siempre ha estado claramente presente.

En la década del 70 las estrategias de negocios se enfocaban principalmente en los temas financieros y la reducción de costos, en los años 80 el enfoque fue en la reingeniería y el rediseño de procesos, los 90 nos trajeron las ideas del planeamiento empresarial de recursos (ERP, por las siglas en inglés) y hoy en día se está viviendo una corriente que reúne un conjunto de iniciativas de marketing, ventas, organización y tecnología que tienen como objetivo común administrar el ciclo de vida del cliente y analizar el valor de la relación existente entre él y la empresa, de modo que esta relación sea la más rentable posible. Esta tendencia es CRM (Customer Relationship Management), cuyo objetivo principal es ayudar a las empresas a maximizar las ganancias de su relación con los clientes a través de vínculos individualizados y apoyado en una infraestructura tecnológica robusta con amplias capacidades analíticas. Años atrás la identificación de los clientes más rentables para un banco era muy costosa y compleja, pero hoy en día ya existen los conocimientos y los recursos tecnológicos para identificar a los clientes de acuerdo a su rentabilidad, comportamiento y características propias. Estos recursos están formados por soluciones tecnológicas que ayudan con la recolección, consulta y mantenimiento de la información de los

clientes (CRM Operacional), y por productos y enfoques orientados al análisis y explotación de la información (CRM Analítico). Un CRM Analítico busca responder las preguntas de negocios de los estrategas de una organización respecto a quiénes son los clientes más rentables, cuáles son los clientes que tienen una rentabilidad potencial que puede ser explotada, cuáles son las relaciones de poco valor que no nos interesa mantener, cómo evoluciona la rentabilidad de los clientes a través de su ciclo de vida, cuál es la tasa de deserción de nuestros clientes, etc.

La tecnología de Inteligencia de Negocios sirve como infraestructura para un CRM Analítico y ayuda en el almacenamiento y administración de los datos, el análisis de información y la explotación por parte de los usuarios finales a través de aplicaciones especializadas con interfaces gráficas de fácil navegación. La presente tesis se enfoca en los componentes de un CRM Analítico en una institución financiera de nuestro medio y propone un modelo de datos como prueba de concepto para el análisis de la relación con los clientes, apoyado en el uso de herramientas de inteligencia de negocios, con el fin de sugerir acciones orientadas a incrementar la rentabilidad de los clientes.

La segunda parte de la tesis presenta la definición y planteamiento del problema propuesto, los objetivos y el alcance del estudio y la importancia del tema desarrollado. En esta sección se sustenta el hecho de que una estrategia CRM se respalda en herramientas de inteligencia de negocios para el análisis de la información de los clientes y la toma de decisiones y define el alcance del presente estudio al ámbito del CRM Analítico.

La tercera parte cubre el fundamento teórico y presenta los conceptos a emplear durante el desarrollo de la tesis. Esta sección empieza con la presentación de los conceptos de CRM y la teoría del intercambio de valor sobre la que se sustenta, el ciclo CRM y la definición de cada uno de sus etapas, los beneficios que brinda y la arquitectura técnica que requiere una estrategia CRM. Luego se explican los componentes y conceptos de inteligencia de negocios que comúnmente se encuentran en una solución

CRM, tales como Data Warehouse, Data Mart, metada, ODS, minería de datos, modelamiento dimensional, extracción, transformación y carga de datos, explotación de datos, calidad y limpieza de datos. Finalmente, se presenta la metodología de Ralph Kimball para la implantación de proyectos de inteligencia de negocios, que será aplicada en el desarrollo del caso de estudio.

La cuarta parte está dedicada al desarrollo del caso de estudio en sí, cubriendo lo relativo a la definición de los requerimientos del negocio que se buscan responder, el modelamiento dimensional propuesto, el diseño físico requerido, los procesos de extracción, transformación y carga llevados a cabo con los datos empleados para el caso de estudio, el diseño de la arquitectura técnica, la selección de los productos de software para la prueba de concepto del caso, y las especificaciones y desarrollo de la aplicación propuesta para analizar la información de los clientes y tomar decisiones basadas en los resultados del análisis.

Finalmente, se incluyen las conclusiones y recomendaciones obtenidas como resultado del presente estudio y que buscan resumir el trabajo realizado.

Esperamos que las investigaciones, análisis, planteamientos y conclusiones expuestos en la presente tesis ayuden para fomentar un mejor entendimiento del rol que cumple el CRM Analítico dentro de una estrategia CRM y cómo los componentes tecnológicos de inteligencia de negocios interactúan entre sí para servir de infraestructura al CRM Analítico.

2.1. DEFINICIÓN Y PLANTEAMIENTO DEL PROBLEMA

El presente estudio centra su atención en el análisis y diseño de los componentes de inteligencia de negocios que forman parte de una solución CRM, tomando como ejemplo un Banco Privado, cuya cartera de clientes formados por personas naturales constituye aproximadamente el 90% del total de clientes.

Para un comerciante minorista, la rentabilidad de un cliente depende del volumen de bienes que compra, pero para un banco la rentabilidad depende también del perfil del riesgo de cada cliente y de su comportamiento. Un comerciante está satisfecho si puede vender a cualquiera, pero para un banco no es así.

Para un comerciante minorista, hay una relación simple entre el ingreso y el costo unitario. Cada producto en venta tiene un costo y un precio de venta. Para un banco, no hay una relación simple entre el ingreso y el costo: transacciones costosas se ofrecen con frecuencia de forma gratuita mientras que unos servicios con un costo marginal pueden generar importantes ingresos. A pesar de estas diferencias, los clientes ven a los bancos aún como organizaciones que venden productos y servicios. Los bancos tienen que entender cómo van a competir en este mercado y cómo van a preparar su organización para responder a los desafíos. Además necesitan herramientas de tecnología de información que les ayude a analizar su información de clientes y que les brinde el conocimiento para:

- entender claramente la rentabilidad de los clientes
- percibir cómo reforzar las relaciones para llegar a esta rentabilidad
- evitar recurrir a tácticas de marketing forzosas que atraen clientes no deseados
- asegurar que cada cliente potencialmente rentable sea retenido por un servicio individual excelente que asegurará a su vez la rentabilidad

Aunque el CRM sea una forma de hacer negocio que afecta a varias funciones, es también una buena inversión financiera. Cada cliente retenido contribuye directamente a la rentabilidad. Las campañas de marketing que producen el tipo adecuado de cliente reducen los costos de adquisición. Una respuesta rápida y oportuna a los requerimientos del servicio en el primer punto de contacto con el cliente reduce los costos (y aumenta la satisfacción del cliente).

Adicionalmente, un banco requiere que una estrategia CRM sea soportada por un conjunto de aplicaciones transaccionales que manejan información de clientes, campañas, reclamos, atenciones, etc., y que forman lo que se conoce como *CRM Operacional*. Por otro lado los usuarios de negocios, analistas y los tomadores de decisiones en general requieren de herramientas e información para realizar un análisis más detallado del comportamiento y la rentabilidad de los clientes, poniendo en aprietos a las áreas de sistemas de los bancos que no siempre cuentan con las herramientas de inteligencia de negocios para hacerlo. Estas herramientas forman parte de lo que se conoce como *CRM Analítico* y serán materia de estudio en la presente tesis.

2.2. OBJETIVO Y ALCANCE DEL ESTUDIO

2.2.1 Objetivo General

- Explicar el proceso de análisis y diseño de los principales componentes de Inteligencia de Negocios empleados como soporte en el análisis de información de una estrategia CRM en una institución financiera y proponer un modelo de datos que cubra las necesidades analíticas que dicha estrategia requiere.

2.2.2 Objetivos Específicos

- Identificar y definir los principales componentes tecnológicos empleados en la implementación de soluciones de Inteligencia de Negocios.
- Explicar la importancia de la administración de la relación con el cliente (CRM) y su impacto en el alcance de los objetivos del negocio.
- Explicar el rol, la importancia y los beneficios del uso adecuado de los conceptos de inteligencia de negocios y sus componentes tecnológicos en la implementación de una estrategia CRM.
- Explicar los beneficios del uso de tecnologías de información avanzadas y probadas en la puesta en marcha de este tipo de soluciones.

- Presentar el diseño y la prueba de concepto de una solución de inteligencia de negocios orientada a soportar la estrategia CRM en una empresa financiera de nuestro país.

2.2.3 Alcances

- Tal como se menciona en el objetivo principal, se ha escogido al sector financiero orientado a la banca personal como el escenario marco para este estudio, porque reúne una serie de características que son de gran interés: en primer lugar, el volumen de información de una empresa de esta naturaleza es muy grande y por lo mismo requiere de una especial metodología de análisis, en segundo lugar, la naturaleza de la relación que surge entre el Banco y sus Clientes ofrece un numeroso conjunto de oportunidades para el intercambio de información, y finalmente porque en el mercado peruano la difusión y experiencia práctica que se tiene sobre CRM y las Herramientas de Inteligencia de Negocios es reducida y es nuestro interés colaborar con el conocimiento de estos temas.
- El fundamento teórico explica los principales sustentos y conceptos de una solución CRM y luego describe con mayor profundidad los componentes de Inteligencia de Negocios que apoyan el análisis de información en dicha solución.
- El caso de estudio desarrollado recoge los conceptos descritos en el fundamento teórico y centra su atención en el diseño de un modelo de datos multidimensional empleado para apoyar el análisis de información de la relación entre un Banco y sus clientes, poniendo énfasis los indicadores de rentabilidad. No se incluye en el caso de estudio el detalle relacionado a los procesos de extracción, transformación, limpieza y carga de datos.
- Como resultado del caso de estudio se proponen un conjunto de acciones a tomar considerando el universo de información empleado. No se incluye en este estudio el seguimiento a estas acciones, pues la mayoría de estas recomendaciones tienen un periodo de implantación de un mediano a largo plazo.

2.3. IMPORTANCIA DEL TEMA

Tradicionalmente, las economías de escala y la masificación de los procesos de ventas y distribución de productos y servicios llevaron a las empresas a olvidar los beneficios que las economías domésticas ofrecían a sus mercados. La visión de las empresas fue entonces conseguir la eficiencia máxima de los medios y mecanismos de producción y ventas, a través de la automatización y el uso de la tecnología. Sin embargo, el notable crecimiento de la tecnología dio lugar al surgimiento de un nuevo recurso a ser explotado: la información. La inteligencia de negocios entendida como la principal herramienta en el análisis de información para la toma de decisiones estratégicas, es actualmente una práctica necesaria para lograr que las empresas mantengan y mejoren su presencia en el mercado. Su adecuada explotación ha permitido elevar notablemente la calidad de los servicios habitualmente ofrecidos, haciendo cada vez más independientes a los consumidores y creando así una nueva cultura de consumo en donde los clientes esperan y exigen recibir cada vez más y mejores productos y servicios.

Es en este contexto que surge un numeroso conjunto de enfoques orientados a diseñar nuevas y creativas maneras de aprovechar la información, consiguiendo al final mantener, incrementar y mejorar las relaciones con los clientes. CRM ha recogido esas propuestas impulsando a las empresas a ver los negocios de una manera totalmente nueva, rescatando los niveles de servicio de las economías domésticas y creando nuevas oportunidades para mejorar e incrementar la oferta y la calidad de sus productos y servicios.

3. FUNDAMENTOS TEÓRICOS

3.1. DEFINICIÓN DE CRM

En los años 70 las estrategias de negocios se enfocaron principalmente a aspectos financieros, en los años 80 las teorías se enfocaron al rediseño de procesos, posteriormente, en los años 90 surgió el planeamiento empresarial de recursos. CRM es ahora, de acuerdo a los especialistas, la siguiente tendencia entre las estrategias de negocios.

Bajo CRM se han reunido un conjunto de iniciativas de marketing, ventas, organización y tecnología, que tienen como objetivo común administrar el ciclo de vida del cliente, de manera que el valor de la relación existente entre él y la empresa sea lo más rentable posible. Para lograr maximizar las ganancias de cada relación se requiere establecer vínculos individualizados con cada tipo de cliente, a través de una infraestructura tecnológica flexible e integrada, que permita llegar con la oferta adecuada al cliente adecuado en el momento adecuado a través del canal adecuado.

Aunque el enfoque de CRM es común a los distintos tipos de negocios en los cuales es aplicado, la manera de implementarlo no necesariamente es la misma. En el sector financiero aún son pocos los casos conocidos de organizaciones que han logrado un éxito claro y sostenible a través de la aplicación de una estrategia de esta naturaleza. Sin embargo, todo indica que las primeras organizaciones que logren asimilar correctamente la visión de CRM y consigan desarrollar la infraestructura tecnológica necesaria que su implementación requiere, serán las que logren obtener los mayores beneficios de un mercado que cada vez más aprende a beneficiarse de la personalización de la oferta.

3.1.1 La visión de CRM

Las economías de escala y el enfoque al ahorro de costos aunque consiguieron incrementar las ganancias de las empresas, trajeron consigo una disminución en el nivel de personalización y la calidad de la oferta. En el sector financiero la influencia de la masificación de la producción también tuvo un gran impacto y afectó no solo los procesos de creación, venta, distribución y mantenimiento de productos y servicios sino incluso la organización laboral interna de estas empresas, centrando sus esfuerzos a optimizar indicadores de productividad y eficiencia, y no de calidad o rentabilidad a largo plazo.

CRM también habla de masificación, pero no de productos, sino del tratamiento individual a los clientes. A diferencia de la producción en masa, la masificación de los clientes consiste en reconocer que los clientes son lo más importante para el negocio, ya no los productos, y como tal todos deben ser entendidos individualmente y no como una visión abstracta de un grupo de consumidores con las mismas características y comportamiento.

Al igual que en la mayoría de negocios, en el sector financiero también se cumple la regla de Pareto con los clientes. Así, es fácil entender cuál es el riesgo de tratar a todos los clientes de la misma manera. Años atrás la identificación de los clientes más rentables era mucho más costosa y compleja. Sin embargo, ahora ya existen los conocimientos y los recursos tecnológicos para identificar a los clientes de acuerdo a su rentabilidad, y si un banco no emplea estos medios está condenado al fracaso.

Tomando en cuenta que las ganancias de una empresa solamente pueden incrementarse a través de: un aumento en la cantidad de relaciones con clientes, un aumento en la rentabilidad de las relaciones existentes, o un aumento en la duración de las relaciones con los clientes actuales; se hace evidente la necesidad de contar con estrategias y programas personalizados para cada cliente. De lo contrario se podría estar invirtiendo demasiados esfuerzos en incrementar relaciones de poco valor y a la vez descuidando o incluso perdiendo las relaciones de mayor valor.

3.1.2 La teoría del intercambio de Valor

Esta teoría es la que da sustento a CRM. Explica la forma en la que a través del análisis del valor de los clientes y el diseño de estrategias para optimizar las relaciones con ellos, las empresas pueden incrementar significativamente su rentabilidad. En Octubre de 1995, Alan Grant y Leonar Schlesinger escribieron en el Harvard Business Review las primeras definiciones que sirvieron para construir esta teoría. Ellos explicaban que el modelo que actualmente las empresas deben seguir para mejorar su rentabilidad es el de la teoría del intercambio de valor. Los precursores de cada modelo de administración o estrategia de negocios siempre fueron los que consiguieron obtener los mayores beneficios de cada aplicación, por eso es importante entender este nuevo modelo para poder obtener el mayor beneficio de su aplicación.

La teoría de intercambio de valor rescata la importancia de ciertos indicadores tales como el valor actual y el valor potencial del cliente, sobre otros más tradicionales como la participación de mercado, los índices de calidad o de satisfacción, y utiliza esta información para organizar los esfuerzos de la organización en acortar la diferencia que existe entre estos dos valores, diferencia que en la práctica siempre ha demostrado ser enorme.

Como sustento de este análisis se recurre a un ejemplo que ilustra el análisis en una tienda de abarrotes. Inicialmente, la tienda dividió sus clientes en tres: los principales (que daban el 80% del negocio), los secundarios (que dan entre el 10 y el 50% del negocio) y los no compradores. El estudio arrojó que los dos primeros constituían menos de la mitad de los clientes. En segundo lugar, se hizo un cálculo del impacto en la rentabilidad de la tienda si se conseguía pequeñas mejoras en la rentabilidad de cada cliente y se concluyó que por la estructura de costos fijos de la tienda el margen de contribución era considerable y se podía mejorar enormemente la rentabilidad. Por ejemplo convirtiendo a 200 secundarios en primarios se incrementaría en más de 20%. Vendiendo un producto en cada cliente se incrementaría el 40%. Expandiendo la base de clientes en 2% se aumentaría la rentabilidad en 45%. En tercer lugar la compañía analizó la duración de sus relaciones con los clientes. La

investigación sugirió que existía un ratio de deserción del 20% y una duración promedio de 5 años. Reduciendo el ratio de deserción al 10% se aumentaría la vida promedio a 10 años. Estos tres componentes claves de la rentabilidad representaban una excelente oportunidad de mejorar e incrementar los indicadores de rentabilidad. Y multiplicados entre sí generarían ganancias aún mayores. Aunque muchos administradores están familiarizados con estos conceptos, pocos son los que conocen el impacto que una estrategia de vinculación tendría en su negocio. Este análisis puede ser aplicado a gran variedad de negocios. Pues en todos los casos, el comportamiento de los clientes y las relaciones que establecen con la empresa están siempre lejos de ser los ideales, lo cual deja una enorme brecha potencial de mejora.

Un requisito para obtener una relación duradera con los clientes es conseguir un constante intercambio de valores para ambas partes. Sin embargo, es probable encontrar excepciones cuando una de las partes no está consciente del valor que esta recibiendo en la relación. Este valor puede ser traducido en términos monetarios en el caso de una empresa financiera como la ganancia por el cobro de los servicios que presta, pero en el caso del cliente pueden haber otros factores para determinar este valor, como la seguridad, comodidad, tranquilidad, etc. Para proveer valor a sus clientes los bancos necesitan conocer sus necesidades y así como la manera de ofrecer productos y servicios que le sean útiles. En la mayoría de casos la personalización de esta oferta es difícilmente llevada a cabo de manera individual (salvo contadas excepciones como en la banca privada).

Finalmente, es necesario precisar que de acuerdo a la teoría de intercambio de valor no siempre va a ser más rentable para una empresa el profundizar su relación con un cliente. De hecho en algunos casos incluso se podría alentar a algunos clientes a terminar su relación con la empresa. El verdadero reto, tal como se precisó anteriormente, consiste en acortar las distancias entre el actual perfil de cada cliente y su potencial comportamiento más rentable, de acuerdo a las necesidades y características individuales.

3.2. EL CICLO CRM

Entre las organizaciones que se han dedicado a realizar estudios sobre CRM, Peppers and Rogers Group ha dado una de las definiciones más aceptadas. Ellos han definido un ciclo continuo formado por etapas mediante las cuales las empresas pueden administrar su relación con los clientes a través de un constante proceso de aprendizaje. En este proceso el cliente invierte tiempo en enseñarle a la organización acerca de sus gustos y necesidades y la empresa aprovecha esta información para diferenciarse del resto del mercado estableciendo una relación más estrecha con sus clientes. Así, la competencia siempre estará a un paso atrás en la curva de aprendizaje y en consecuencia, un paso atrás en la carrera por ofrecerle productos o servicios.

El ciclo CRM consta de cuatro etapas: Identificación, Diferenciación e Interacción, y Personalización (Ver Figura 1). Los beneficios que se puedan obtener en cada fase son mínimos comparados a los que se pueden obtener en la fase final, que es la única fase visible ante los ojos del cliente.

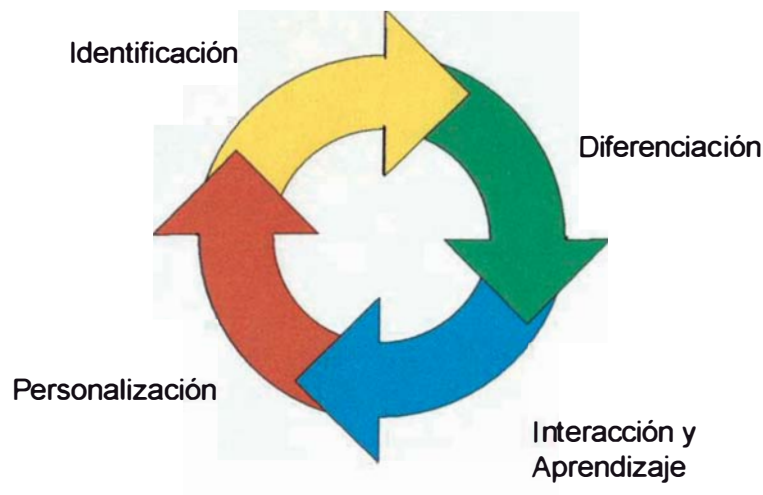


Figura 1. Ciclo CRM

3.2.1 Identificando al Cliente

El primer paso para iniciar un ciclo CRM, consiste en la identificación del mercado al cual se está atendiendo. De acuerdo a las definiciones de Peppers & Roggers en el proceso de identificación de clientes se deben considerar tres pasos: realizar un inventario de la información disponible sobre tus clientes en formatos electrónicos (en especial aquellos que puedan registrar información específica y actualizada sobre el cliente), en segundo lugar revisar la información que aunque no este guardada en formatos electrónicos pueda registrar datos importantes del cliente (como procesos no automatizados de crédito), y en tercer lugar se deben diseñar las formas de recolectar más información del cliente (uno de los ejemplos cada vez más difundido para obtener información del cliente son los programas de millas o de usuarios frecuentes, en los que se premia a los clientes que consuman más, y gracias a los cuales se puede hacer un seguimiento detallado de los sus hábitos consumo). Siempre es necesario recordar no capturar más información de la que se pueda manejar, tener demasiada información puede ser tan improductivo como tener poca.

En el caso de los bancos hay algunas consideraciones adicionales a observar: un primer problema es la diversidad de las bases de datos que registran la información del cliente y el hecho de que probablemente un cliente está registrado de distintas maneras en la misma organización, este problema exige la creación de un identificador que sea compartido y empleado alrededor de toda la organización; un segundo problema es la dualidad con la que los clientes que manejan pequeños negocios son tratados, no hay una respuesta exacta para tratar este problema. Esto depende de cada organización; otro problema es el uso de intermediarios con las grandes organizaciones, en este caso es necesario tener a todos en cuenta.

Finalmente, es necesario tener presente que la definición del cliente debe ser flexible e integral. La definición debe ser consistente en toda la empresa y no debe estar orientada a la administración en particular de un servicio o producto.

3.2.2 Diferenciación

Luego de identificar claramente a los clientes es necesario desarrollar estrategias diferenciadas para poder establecer mejores relaciones con cada uno de ellos. La segmentación de clientes ha demostrado ser una herramienta de gran utilidad para conseguir este objetivo.

El trabajo de segmentación debe ser enfocado tomando en cuenta una serie de variables y características de los clientes, lo cual no es nuevo para los bancos pues lo han venido utilizando para diferenciar a los clientes por su riesgo por ejemplo, solamente que ahora no sólo se usara para evaluar riesgos sino para incrementar la rentabilidad. Y para poder entender las necesidades del cliente se debe tomar en cuenta dos niveles de análisis: las del segmento al cual pertenece y las de cada individuo.

Existen diferentes métodos para segmentar clientes. Entre los principales se encuentran:

- Segmentación por el ciclo de vida, que emplea los eventos en la vida de cada persona como los elementos que desencadenan o influyen en sus necesidades (por ejemplo: matrimonio, un nuevo hijo, estudios universitarios, post-grado, casa, carro, etc.). El principal inconveniente para aprovechar esta situación es la frecuente necesidad de poder actuar coordinadamente junto con otras empresas proveedoras de productos para poder satisfacer estas necesidades de manera oportuna.
- Segmentación por datos demográficos, tales como edad, sexo, ingresos, etc. Estos datos son fáciles de conseguir y sirven para identificar a las personas de mayor riqueza o de mayor potencial rentable. Lamentablemente incluso entre estos segmentos pueden haber importantes diferencias que estos criterios no identifican. Además los grupos de mayor poder económico suelen diversificar más sus inversiones y no siempre son las más rentables.
- Segmentación sicográfica, que apela a definiciones sobre la actitud de las personas para poder agruparlas en distintas clases. Estas definiciones suelen ser obtenidas a través de encuestas y focus groups.

Lamentablemente, los segmentos definidos a través de esta técnica suelen ser muy difíciles de ubicar. Otros inconvenientes son que no siempre el comportamiento ideal de las personas es el que tienen en la vida real y la experiencia ha enseñado que es muy difícil establecer una relación entre estos segmentos y la rentabilidad que generan.

- Segmentación por el comportamiento, que exige identificar y definir patrones de comportamiento entre los clientes. La segmentación de este tipo recurre, por ejemplo, a variables como cantidad de productos, saldos promedios, hábitos transaccionales, etc. Este criterio es mucho más preciso que los anteriores, pero también es bastante costoso y requiere un gran esfuerzo para mantenerlo actualizado.
- Segmentación híbrida, que utiliza elementos de todas las anteriores técnicas de segmentación para crear una nueva segmentación más precisa y con una mejor correspondencia con respecto a la rentabilidad de los clientes.

Independientemente del esquema de segmentación, siempre se debe tomar en cuenta entre las variables para diferenciar a los clientes a la rentabilidad y el comportamiento (actual y potencial).

Una vez que la segmentación ya ha sido diseñada debe ser puesta a disposición de toda la organización como base para poder tratar a los clientes de manera consistente a través de todos los puntos de contacto. Las acciones de segmentación deben ser diseñadas de manera integral a través de la organización (no solamente desde el punto de vista de un canal o un producto) procurando su mayor utilidad y entendimiento en la organización (nombres claros y sencillos).

La ventaja competitiva no se da a través de la segmentación sino a través de la atención de necesidades individuales, es decir el marketing 1 a 1. Este es un proceso que se debe reforzar y potenciar lo más posible llevando, por ejemplo, una historia consistente de todos los contactos del cliente, o haciendo un estudio del comportamiento transaccional del mismo. Administrar esta información no es invadir su privacidad, por el contrario es un medio para

poder atenderlo mejor. Si bien la diferenciación individual de las necesidades es importante también lo es el saber diferenciar el valor o la rentabilidad que significa individualmente cada cliente para el Banco. Cuando se comenzó a utilizar la segmentación de la rentabilidad para entender a los clientes en el sistema financiero se terminaron varios mitos sobre el valor de los clientes. Se descubrió que los más rentables no son siempre los que tienen más productos o transaccionan más, sino que, generalmente, son los que tienen saldos altos y transaccionan poco. El cálculo del valor del cliente no debe obviar el valor futuro del cliente. Cada persona tiene un tiempo de vida como cliente y como tal tiene un valor actual y futuro.

También es probable que exista un margen de rentabilidad potencial que el banco no está aprovechando con cada cliente. Para llegar a este nivel de medición, Peppers & Roggers recomienda realizar supuestos sobre el comportamiento de los clientes y tomar como variables el tiempo de vida y la rentabilidad actual para definir a los clientes en tres grupos:

- *Los de mayor valor*, conformado por el grupo de clientes que dejan la mayor rentabilidad al banco y por lo cual merecen especial cuidado para ser retenidos (tales como reconocimientos, tasas preferenciales, trato preferencial, etc.). Antes de realizar alguna campaña es necesario entender qué es lo que hace rentables a estos clientes y no caer en el error de beneficiarlos afectando parte de lo que los hace rentables.
- *Los de mayor potencial*, conformado por aquellos clientes que, si bien no son rentables necesariamente, si tienen potencial para incrementar su margen de rentabilidad. En estos casos se debe invertir, durante un tiempo determinado, un esfuerzo adicional de marketing para atraer a estos clientes. En caso de no hacerse efectivo el esfuerzo se debe cesar el intento y tratarlos sin consideración a su potencial.
- *Los clientes bajo cero*, conformado por aquellos clientes que dejan el menor margen de rentabilidad al banco. Dependiendo el nivel de rentabilidad es necesario en algunos casos deshacerse de estos clientes a través de acciones punitivas.

Antes de realizar cualquier acción se debe analizar al cliente de manera integral y no solo por su comportamiento en una línea de productos (aún cuando la práctica demuestra que es muy raro encontrar un cliente bajo cero en un producto y muy rentable en otro). Posteriormente, el banco debe aprender a detectar a este tipo de clientes para evitar establecer relaciones no rentables con él.

3.2.3 Aprendiendo a través de la interacción del cliente

El diálogo debe ser utilizado constantemente y de manera cuidadosa para poder enriquecer la relación con los clientes. Tradicionalmente, éste se da unidireccionalmente, desaprovechando las oportunidades en las que se establecen contactos con los clientes. El diálogo debe darse bidireccionalmente, tal como si se tratase de una conversación que evoluciona con la participación de ambas partes. La creación del diálogo puede producir una muy favorable impresión en los clientes, pero debe ser empleada para mejorar la relación con los clientes a través de esa impresión.

La interacción debe darse tomando en cuenta las características del cliente y el segmento con el cual se establece la conversación. Así por ejemplo con un cliente del segmento bajo cero se debe sugerir el manejo de menos productos o de canales menos costosos mientras que con un cliente rentable la conversación debe enfocarse en temas de satisfacción. En todos los casos la conversación debe minimizar los inconvenientes del cliente para que entregue la información y debe enfocarse en mejorar o afectar la relación del cliente.

El diálogo debe comenzar por las necesidades y la opinión del cliente y no por temas directamente asociados a un producto. Las preguntas relacionadas al producto deben surgir naturalmente como conclusión de la situación o el interés del cliente. Las preguntas más útiles son las que permiten identificar fechas o acontecimientos claves en la vida de un cliente para ofrecerle productos o en general para fortalecer las relaciones con los clientes.

La información de estas interacciones debe ser administrada correctamente, una pregunta no debería realizarse dos veces. Adicionalmente, y como parte

de la información es confidencial se debe explicar claramente al cliente (cuando sea necesario) por qué se realizan las preguntas y de qué manera y por quiénes será utilizada la información. Los datos confidenciales del cliente sólo deben ser manejados por el administrador o representante de cada cliente.

3.2.4 Personalizando la oferta

El cuarto paso es actuar sobre la base de la información recolectada. Inicialmente, y con la idea de que era sumamente costosa, la personalización era aplicada solamente a los principales clientes. Sin embargo, aún cuando el costo es un factor importante, la personalización trae una serie de beneficios.

En general, la personalización de productos permite eliminar los costos por inventarios, diseñando productos a pedido del cliente. En el sector financiero, aún cuando no hay inventarios, permite reducir costos simplificando trámites e incrementando eficiencias y otro tipo de gastos. Del lado del cliente el beneficio es claro, él obtiene exactamente lo que necesita.

La personalización no significa crear un producto distinto para cada cliente. Lo que propone es identificar las variables que hacen distintas las necesidades de cada cliente y hacer flexibles a los productos ofrecidos para que el cliente pueda beneficiarse configurando su propio producto. Por ejemplo, hablando de tarjetas de crédito, la personalización se consigue manejando fechas de pago, periodos de gracia, líneas de crédito y tasas de interés distintas por cada tipo de cliente. El reto no es considerar todas las posibles combinaciones de este tipo de variables sino identificar un grupo adecuado significativo de clientes que justifique la creación de un determinado tipo de producto.

Un aspecto a considerar es la manera en que la personalización debe ser introducida y adoptada por la empresa. Los esfuerzos por implementar una estrategia CRM requieren la participación de un conjunto de áreas. Normalmente una organización bancaria presenta tres entidades involucradas con la administración del negocio: los encargados del producto, del segmento y del sector geográfico. El negocio no debe ser visto nunca como algo aislado

sino como un esfuerzo de toda la organización. Para cumplir con este objetivo se recomienda comenzar con una sola línea de productos e ir luego extendiendo la experiencia.

Asociados al concepto de personalización existen otros temas a considerar para fortalecer la relación con el cliente:

- Bundling.- que consiste en relacionar dos o más productos para aprovechar características complementarias. Prácticas comunes son por ejemplo la asociación de cuentas a plazo con cuentas corrientes (para facilitar el cobro de sobregiros).
- Packing y Delivering.- consideraran algo que personalice la entrega y el producto en sí (fotos en las tarjetas de crédito, agencias para los empleados de los clientes más importantes, entrega a la dirección que el cliente prefiera),
- Productos y Servicios Auxiliares.- (como detalles o formatos especiales para el reporte de gastos en las tarjetas de crédito, pre-aprobaciones para productos similares).
- Venta Cruzada.- Ejemplos de venta cruzada son: hipotecas y seguros de casa, prestamos y seguros vehiculares.

No hay que dejar de lado que la empresa debe ser vista por el cliente de manera unificada, por lo que la recolección de datos y entrega de productos debe evitar molestar al cliente innecesariamente.

Nuevamente la personalización es mejor iniciarla en un área en la cual se tenga control e independencia para implementarla. Nuevamente hay que tener especial cuidado al momento de definir los conceptos de rentabilidad del cliente. La rentabilidad debe ser analizada de manera global de lo contrario un cliente no rentable pero potencialmente rentable para otra línea podría no estar siendo aprovechado correctamente y para estimular el intercambio de información y colaboración entre las áreas del banco.

3.3. BENEFICIOS DE CRM

Las empresas que aplican CRM no sólo incrementan sus ratios de retención de clientes sino que consiguen un marketing mucho más efectivo, creando oportunidades para la venta cruzada y haciendo más ágiles y flexibles los procesos de creación de productos y servicios. Estas mejoras se encuentran directamente relacionadas y los beneficios que se obtienen de cada parte no serían sostenibles en el tiempo si la empresa no desarrolla CRM de una manera integral.

Los beneficios que se obtienen de CRM pueden ser agrupados en tres categorías:

- Ahorros de costos.
 - Procesos de creación de productos y servicios más efectivos
 - Disminución de costos en la administración de clientes. Una disminución del 10% durante los tres primeros años del proyecto es un logro aceptable.
- Incremento de la rentabilidad.
 - Incremento de las ventas generales, como resultado de un proceso de registro de datos más simple. Un incremento del 10% en las ventas es una meta aceptable.
 - Incremento del ratio de éxito de las campañas comerciales, al contar con un proceso selectivo basado en la segmentación de los clientes y la personalización de la oferta. Un 5% en el incremento anual en el ratio de contratos exitosos es una meta aceptable.
 - Incremento de los ratios de retención de clientes rentables.
 - Incremento de las oportunidades para la venta cruzada, al contar con mayor información sobre las necesidades y preferencias del cliente.
- Impacto estratégico
 - Mejora de los tiempos de respuesta para la atención de la demanda
 - Fortalecimiento del vínculo entre la empresa y el cliente
 - Mayor conocimiento de las preferencias del mercado

- Incremento en los ratios de satisfacción de los clientes. Un incremento del 3% durante los 3 primeros años del proyecto es un resultado aceptable.

3.4. ARQUITECTURA TÉCNICA DE UNA SOLUCIÓN CRM

El ciclo CRM, tal como ha sido mencionado anteriormente, requiere de una infraestructura tecnológica que soporte las actividades descritas en cada fase del ciclo descrito por Peppers & Roggers. Aún cuando es factible diseñar y ejecutar exitosamente acciones comerciales como las sugeridas por CRM sin contar con todos los componentes en funcionamiento, solamente cuando se tiene completa la infraestructura descrita a continuación, se puede sacar el máximo provecho de todas las interacciones producidas con el cliente.

Dependiendo de la naturaleza comercial de cada empresa estos componentes pueden variar. En el caso de las instituciones financieras podría identificarse cuatro áreas principales (Ver Figura 2).

- Sistemas transaccionales u OLTP.- es decir todos aquellos aplicativos empleados para soportar la operación básica de la entidad financiera, tales como aplicativos de cuentas corrientes, de tarjetas de crédito, de contabilidad, etc.
- Canales de Atención.- es decir aquellos aplicativos empleados para interactuar con el cliente. Estos aplicativos son la cara del Banco frente a los clientes y cualquier estrategia CRM depende fuertemente de esta capa de componentes.
- Plataforma CRM.- Dentro del ámbito descrito por CRM se pueden agrupar un conjunto de aplicativos creados para gestionar la relación con los clientes. En este rubro se encuentran los aplicativos de Rentabilidad, Ventas, Manejo de Campañas, etc.
- Inteligencia de Negocios.- Agrupa las tecnologías de análisis de información. En el caso de organizaciones de gran envergadura, estos aplicativos reposan sobre un almacén de datos que centraliza la información (Data Warehouse).

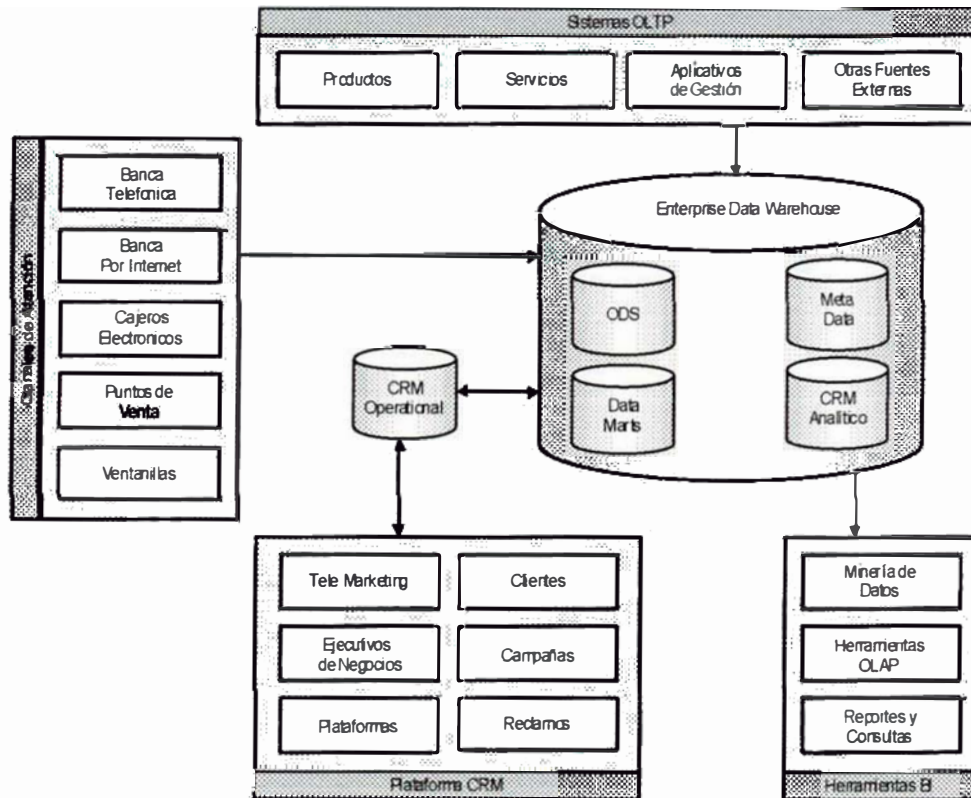


Figura 2. Arquitectura tecnológica de una solución CRM

3.5. INTELIGENCIA DE NEGOCIOS

La presente tesis gira alrededor de los componentes de inteligencia de negocios (Business Intelligence) que forman parte de un CRM Analítico y que representan los componentes tecnológicos sobre los que se basa una solución CRM. Estos componentes tecnológicos ayudarán en el almacenamiento y administración de los datos, el modelamiento de los datos, el análisis de información y la entrega a los usuarios finales a través de aplicaciones especializadas con interfaces de fácil navegación. El siguiente gráfico (Ver Figura 3) busca aclarar la participación de cada uno de los componentes en una solución de inteligencia de negocios que serán descritos en las siguientes páginas.

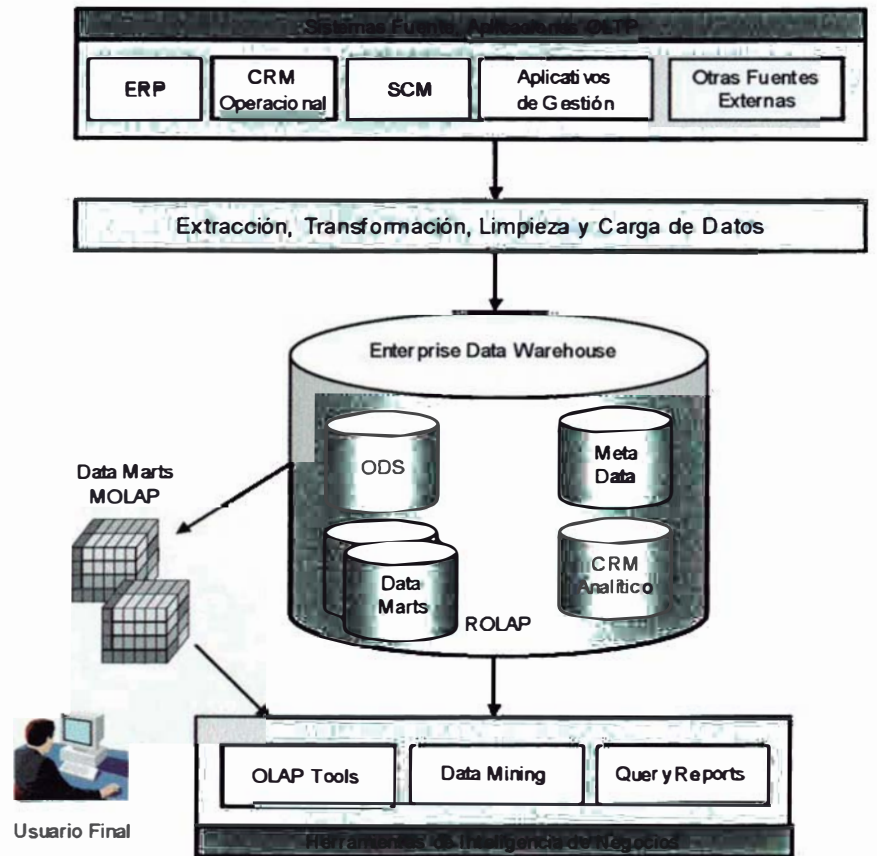


Figura 3. Componentes de Inteligencia de Negocios de una solución CRM

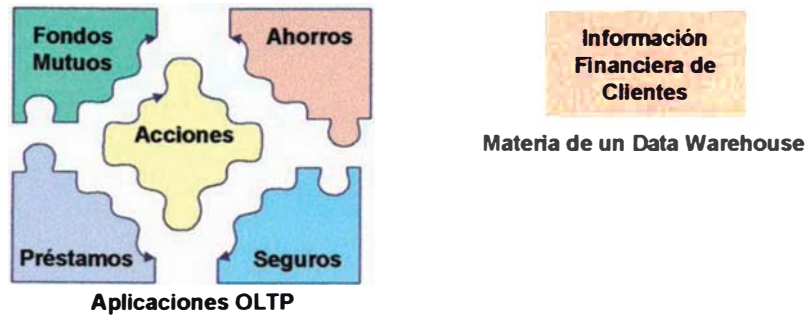
3.5.1 Data Warehouse

De acuerdo con William. H. Inmon, quien es considerado el padre del Data Warehouse: “Un Data Warehouse es un conjunto de datos integrados y orientados a una materia, que varían con el tiempo y que no son transitorios, los cuales soportan el proceso de toma de decisiones de una administración.”

Esta definición se presenta gráficamente en las siguientes láminas (Ver Figura 4) para reforzar el concepto planteado:

Orientado a una materia

Los datos son categorizados y almacenados de acuerdo a la materia del negocio a la que pertenecen, en vez de hacerlo por aplicación



Integrado

Los datos de una materia determinada son definidos y almacenados una vez



Varía con el tiempo

Los datos son almacenados como series de instantáneas, cada una representando un periodo de tiempo.



No volátil

Típicamente los datos en un data warehouse no son actualizados o eliminados.

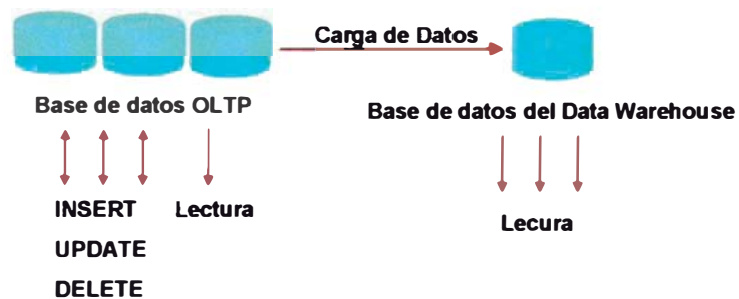


Figura 4. Características de un Data Warehouse

Para Ralph Kimball, creador del “Ciclo de Vida Dimensional del Negocio” (Business Dimensional Lifecycle™), “un Data Warehouse representa una copia de los datos transaccionales que se encuentran específicamente estructurados para consultas y análisis, y que brinda apoyo al proceso de toma de decisiones de una compañía”.

De acuerdo con algunas organizaciones, el Data Warehouse es una arquitectura. Para otras, es un depósito semánticamente consistente en datos (separados y que no interfieren con los sistemas operativos y de producción existentes) que llenan por completo los diferentes requerimientos de acceso y reporte de datos. Para algunos otros, el Data Warehouse es un proceso continuo que mezcla los datos de varias fuentes heterogéneas, incluyendo datos históricos y adquiridos para soportar la constante necesidad de consultas estructuradas y/o ad-hoc, reportes analíticos y soporte de decisiones.

Así como hay una gran divergencia para establecer una definición precisa de un Data Warehouse, hay un claro consenso que el Data Warehouse es en nuestros tiempos un ingrediente esencial en el conjunto de soluciones para el soporte de decisiones de una empresa y constituye una pieza indispensable y fundamental en la nueva orientación de las empresas hacia el cliente. Una solución CRM involucra contar con un repositorio de datos estratégicos como base fundamental para mejorar el nivel de interrelación con los clientes, dentro de la nueva filosofía de orientación al cliente, más que la orientación al producto.

Data Warehousing (entendido como el proceso para la definición, construcción, implementación y explotación de un Data Warehouse) apoya la integración de los datos dispersos en una organización y lucha contra el "caos de datos" generado por la proliferación de diferentes sistemas aislados y puntuales, que hace difícil responder con la velocidad que exige el mercado a las preguntas que se formulan los estrategias de los negocios para apoyar sus tomas de decisiones.

3.5.2 Diferencia entre un Data Warehouse y los Sistemas OLTP

El trabajo en los entornos de procesamiento transaccional en línea (OLTP, On-Line Transaction Processing), representado por las bases de datos operacionales de una empresa, difiere del trabajo llevado a cabo en un entorno de inteligencia de negocios. Los usuarios son diferentes, el contenido de los datos es diferente, las estructuras de

datos son diferentes, el hardware y el software son diferentes, la administración es diferente, el manejo de los sistemas es diferente y el ritmo diario de trabajo en un ambiente u otro es diferente. A pesar de estas diferencias tan marcadas, muchos todavía siguen empleando su pensamiento OLTP y las herramientas OLTP para diseñar entornos de inteligencia de negocios, lo cual constituye un peligro para cualquier proyecto de Data Warehousing serio. En el Cuadro 1 se muestran algunas diferencias resaltantes entre un entorno OLTP y un Data Warehouse.

Característica	OLTP	Data Warehouse
Orientación	Está orientado desde la perspectiva de la aplicación, de modo que el acceso a los datos desde la aplicación tenga la mayor eficiencia posible.	Está orientado a una materia, desde la perspectiva del usuario final.
Volumen de Información que maneja	Volumen de información de acuerdo al tipo de aplicación. La información histórica se pone fuera de línea en dispositivos de respaldo para mejorar el rendimiento de las aplicaciones.	Grandes volúmenes de información. Incluye información histórica.
Versiones del Esquema de Datos	Mantiene una única versión del esquema de base de datos, el cual puede cambiar en el tiempo dependiendo de las necesidades del negocio.	Puede llegar a contener diferentes versiones de un esquema de datos debido a que tiene que guardar información histórica y administrarla.
Nivel de Detalle de los Datos	La información se encuentra con un alto nivel de detalle, tal y como son generadas por las aplicaciones operacionales.	Almacena datos agregados o resumizados para responder con rapidez las consultas de los usuarios. También puede guardar información al nivel de detalle.
Carga de Trabajo	Procesa miles e inclusive millones de transacciones por día. Cada transacción contiene una pequeña pieza de datos.	Usualmente procesa una sola transacción que está formada por miles de registros. En vez de llamarlo una transacción se le conoce a este proceso como carga de datos.
Integración de Información	Se alimenta del ingreso de información que hacen los usuarios a través de las aplicaciones operacionales. No integra datos de otras fuentes.	Integra y organiza en un solo lugar información de diferentes fuentes, con diferentes características tecnológicas.

Tipos de Usuarios	Son los usuarios que hacen girar las ruedas que mueven el negocio y casi siempre atienden una única cuenta a la vez. Los datos que recuperan estos usuarios son registros sencillos y su labor incluye el ingreso de datos. Los usuarios OLTP realizan la misma tarea varias veces al día.	Son los usuarios que se preocupan por las ruedas que mueven el negocio. Tratan con varias cuentas a la vez y hacen consultas que involucran la lectura de miles de registros para ser presentados finalmente como resultados sumariados. Estos usuarios cambian constante el tipo de consultas que hacen.
Tiempo de Respuesta	Los tiempos de respuesta se miden en segundos. El rendimiento es el rey absoluto en un entorno OLTP. Ninguna actividad opcional es permitida que pueda restar rendimiento al sistema OLTP.	Las consultas se ejecutan en tiempos que van de minutos y pueden llegar a horas en instalaciones muy grandes, debido al volumen de Información que procesan.
Manejo de la dimensión Tiempo	Un sistema OLTP es una foto instantánea de una organización y es actualizada constantemente por las transacciones. Los indicadores de negocio cambian segundo a segundo.	Un Data Warehouse representa una serie de tiempo explícita. Los datos de los sistemas OLTP se mueven a un Data Warehouse y se almacenan como una serie de capas de datos, para representar luego los puntos en el tiempo. Los usuarios deben definir la frecuencia con la que se cargarán los datos en un Data Warehouse (diario, semanal, mensual).
Modelo de Datos	Se basa en el modelo entidad-relación, que busca eliminar la redundancia de los datos de modo que una transacción "toque" la base de datos en pocos puntos, con el fin de obtener buenos tiempos de respuesta. El modelo entidad-relación trabaja dividiendo los datos en muchas entidades discretas, cada una de las cuales representa una tabla en la base de datos OLTP, y asociándolas a través de relaciones entre ellas.	Se basa en el modelamiento dimensional, que presenta las entidades de análisis de una manera más clara para los usuarios y permite una navegación más intuitiva a través de ellas. El modelo dimensional agrupa las entidades para identificar las que contienen criterios de análisis (dimensiones) de las que contienen los indicadores del negocio (métricas). Más adelante se presentará más detalle del modelo dimensional.

Cuadro 1. Data Warehouse vs Sistemas OLTP

3.5.3 Componentes de un entorno de Inteligencia de Negocios

3.5.3.1 Sistemas Fuente (Legacy Systems)

Es un sistema operacional cuya función es capturar las transacciones del negocio. Un sistema fuente es frecuentemente llamado un sistema heredado o legado (legacy system) dentro de

un entorno mainframe. Debería pensarse en un sistema fuente como algo externo al Data Warehouse, en vista que se asume que no tenemos control sobre el contenido y el formato de los datos en el sistema legado. Las prioridades principales de un sistema fuente son alto rendimiento y disponibilidad. Las consultas contra estos sistemas fuente son específicas y basadas en cuentas, códigos o claves que forman parte del flujo normal de transacciones y son severamente restringidas en sus demandas de datos. Se asume que estos sistemas mantienen pocos datos históricos y que la administración de reportes desde estos sistemas es un trabajo muy pesado. Además, estos sistemas fuente no son consultados con la amplitud y diversidad de formas con la que es consultado un entorno de inteligencia de negocios.

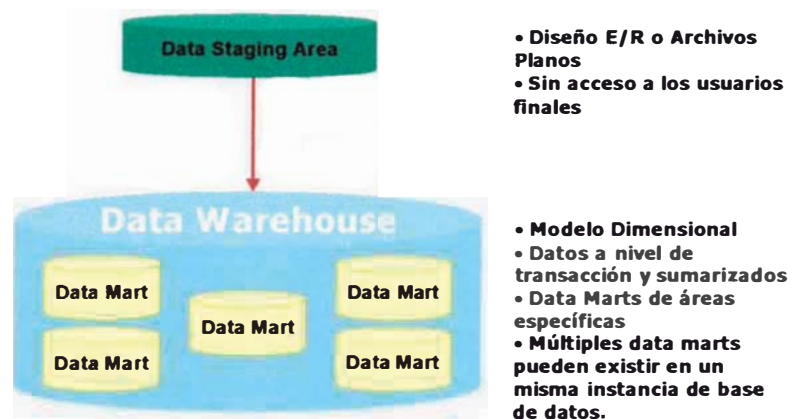
3.5.3.2 Área de Escenificación de Datos (Data Staging Area)

Es una definición presentada por Ralph Kimball y poco usada por otros autores. Constituye un área de almacenamiento donde se llevan a cabo procesos que limpian, transforman, combinan, mantienen, archivan y preparan los datos fuente para ser usados en un entorno de Data Warehousing. Un "data staging area" es todo aquello entre el sistema fuente y el servidor que servirá como repositorio del ambiente de inteligencia de negocios. Se encuentra explícitamente fuera del alcance de los usuarios finales. Aunque sería ideal si esta área de datos estuviera centralizada sobre una única pieza de hardware, es mucho más probable que se encuentre extendido sobre un número determinado de máquinas. La clave al definir las restricciones sobre esta área de datos es que no provee servicios de consulta y presentación. Esta área de datos no intenta proveer seguridad al usuario, agregados, índices o series de tiempo. Los datos antes del proceso de limpieza casi siempre se encuentran como archivos planos y después de este proceso pueden estar como archivos planos o en la tercera forma normal.

3.5.3.3 Data Mart

Es un subconjunto lógico de un Data Warehouse. Usualmente vemos a un Data Mart como la parte del Data Warehouse restringida a un proceso de negocios individual o a un grupo de procesos de negocios relacionados y orientados hacia una línea de negocios específica (LOB, siglas en inglés de “Line of Business”). El Data Mart usualmente es patrocinado y construido por una parte particular del negocio y organizado alrededor de un proceso de negocio específico, utilizando un modelamiento dimensional, con ayuda de bases de datos relacionales o multidimensionales de tecnología propietaria.

Data Warehouse y Data Mart: Enfoque de Ralph Kimball(*)



(*) The Business Dimensional Lifecycle™, 1998

Figura 5. Data Warehouse y Data Mart, Enfoque de Ralph Kimball

De acuerdo al enfoque de Ralph Kimball, un Data Warehouse es construido de la unión de todos sus Data Marts (Figura 5). Kimball enfatiza la necesidad que un Data Mart se base en definiciones coherentes con los de otros Data Marts. Así una medida del negocio coherente, como el nivel de ventas, puede ser sumariada y comparada correctamente a través de los diferentes Data Marts de una organización.

Esta definición de Kimball sugiere la posibilidad de iniciar un proyecto de inteligencia de negocios a partir de la definición e implementación de Data Marts en una organización, los cuales en conjunto formarán el Data Warehouse corporativo, siempre que se mantengan controles exigentes en los estándares y las definiciones para facilitar la integración futura en la medida que vayan apareciendo nuevos Data Marts.

Bill Inmon presenta otra perspectiva. Según Inmon, los datos fluyen desde el Data Warehouse a los diferentes departamentos para darles un uso personalizado en los sistemas de toma de decisiones. Estos datos departamentales y personalizados son llamados Data Marts. Un Data Warehouse almacena datos a un nivel muy granular, mientras que en el Data Mart se requiere cierta sumarización, conocida como agregación (Figura 6).

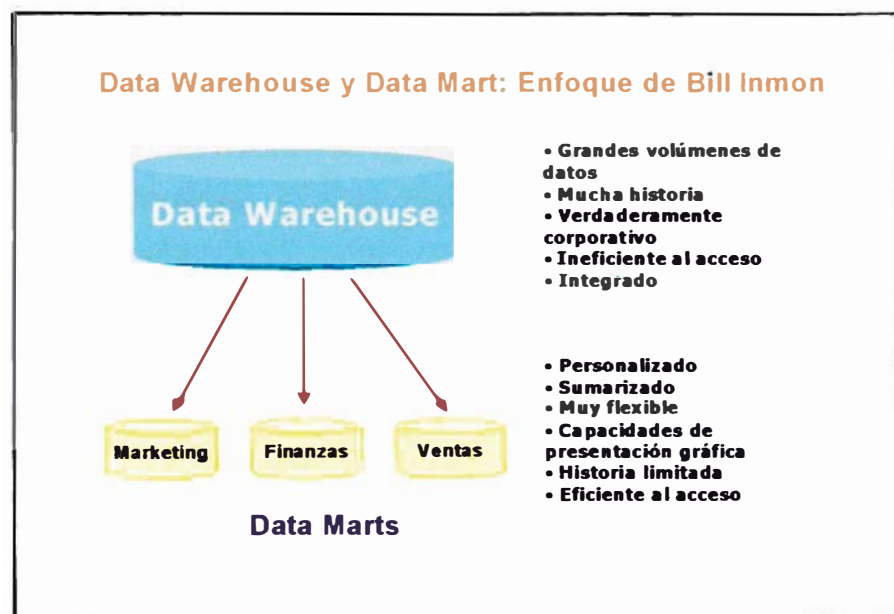


Figura 6. Data Warehouse y Data Mart, Enfoque de Bill Inmon

Inmon también diferencia un Data Mart independiente de otro dependiente, como se presenta en la Figura 7. Un *Data Mart independiente* es aquel que es alimentado directamente desde los sistemas fuente (legacy systems, sistemas operacionales relacionales, información externa, etc.). Usualmente es

patrocinado por alguna unidad del negocio (área usuaria) que busca acortar los tiempos de implementación de un sistema de soporte a la toma de decisiones. Este tipo de Data Mart es poblado usando técnicas de extracción, transformación y carga (ETL: Extraction, Transformation and Load), para llevar los datos desde los sistemas fuente hasta la base de datos donde reside el Data Mart. Por su parte, *un Data Mart dependiente* es poblado a partir de datos provenientes del Data Warehouse corporativo de una empresa, a través de la copia de los datos, en un proceso menos complejo que en el caso anterior, teniendo en cuenta que los datos que residen en un Data Warehouse ya han pasado por las etapas de extracción, limpieza, transformación y carga. Para Inmon, la construcción de un entorno de toma de decisiones debe empezar siempre por el Data Warehouse y luego por los Data Marts, porque considera que el entorno operacional nunca es una fuente legítima de datos, mientras que un Data Warehouse corporativo cumple con las condiciones necesarias para garantizar la calidad de los datos en un Data Mart y reduce las interfaces entre las aplicaciones operacionales del negocio y los Data Marts. El Data Warehouse corporativo va a contener datos granulares comunes que serán aprovechados por diferentes Data Marts en una organización cuando se implementen como Data Marts dependientes, mientras que los Data Marts independientes van a tener una parte de datos únicos a cada Data Mart y una buena parte de datos corporativos comunes repetidos en los diferentes departamentos.

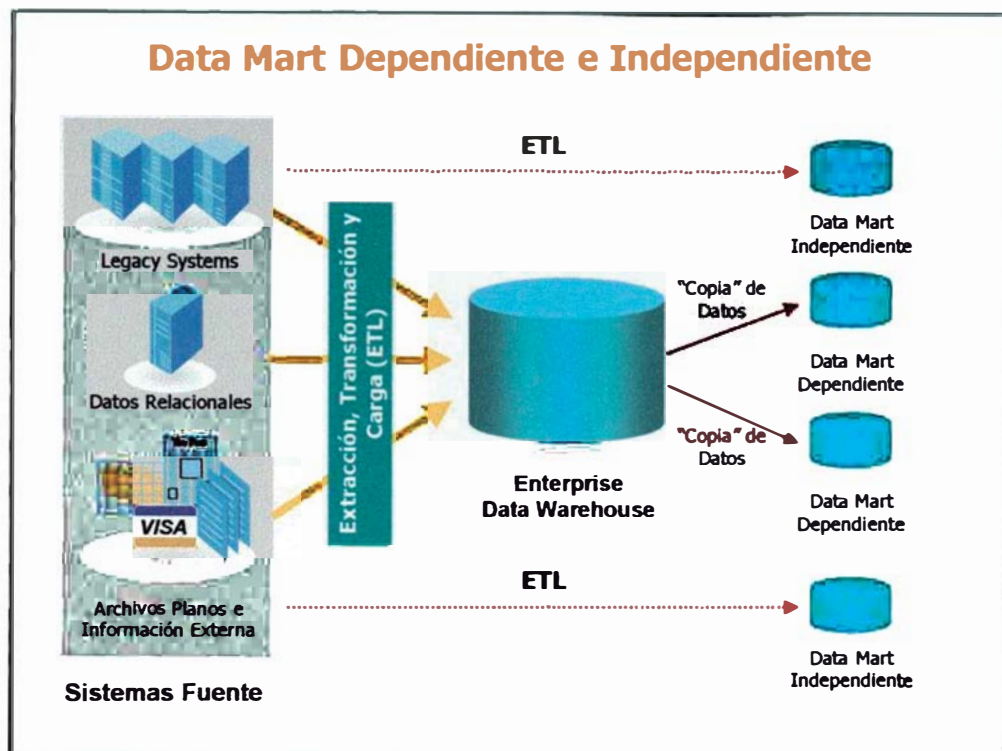


Figura 7. Data Mart Dependiente e Independiente

Lo cierto es que la teoría nos da una guía, pero la realidad de nuestras empresas y las necesidades de negocios usualmente conllevan a una mixtura de los enfoques antes presentados. En nuestro medio, muchas empresas todavía se ven atemorizadas con la idea de iniciar un mega proyecto que los lleve a definir e implementar su Data Warehouse corporativo. Las áreas de sistemas encargadas de estos proyectos usualmente piensan en el largo plazo, mientras que las áreas usuarias de alto nivel presionan por resultados inmediatos, motivados por las promesas que el proyecto de Data Warehouse esbozó cuando fue presentado en el directorio. Es por eso que algunas empresas deciden iniciar el proyecto con un Data Mart para un área específica, usualmente con la idea de contar con una prueba de concepto que deje claro las ventajas de este tipo de soluciones en los procesos de toma de decisiones. En estos casos se debería arrancar siempre con una definición de la arquitectura que contemple el Data Warehouse corporativo como una meta y los posibles Data Marts, de modo que se visualice la relación entre ellos y la estrategia a seguir para evitar la existencia de Data Marts dispersos, aislados y de difícil integración futura.

Otras empresas, usualmente de mayor envergadura, que van tras una estrategia corporativa y que son conscientes que los datos que requieren para respaldar sus tomas de decisiones en aspectos claves del negocio se encuentran dispersos en diferentes aplicaciones y plataformas, sin estandarización y de difícil acceso deciden iniciar la construcción del Data Warehouse como primer paso.

3.5.3.4 Metadata

La identificación, definición y administración de la metadata constituyen elementos vitales para todo proyecto de inteligencia de negocios. Podemos entender a la metadata como los datos acerca de los datos. En un sentido figurado podemos decir que la metadata es el DNA de un entorno de inteligencia de negocios: define qué es cada uno de los elementos y cómo trabajan todos juntos. Usualmente es difícil identificar qué es exactamente la metadata. Se podría decir que es todo, a excepción de los datos del entorno de inteligencia de negocios en sí.

Usualmente se hace necesario manejar diferentes tipos de metadata. Por ejemplo, durante las tareas de extracción, transformación y carga tendremos que administrar datos sobre el diseño físico de las tablas involucradas, las fuentes de datos, mapeos, reglas, etc.; el administrador de la base de datos estará interesado en manejar información sobre tablas del sistema, índices, definiciones de vistas, procesos de respaldo y recuperación, procedimientos almacenados y scripts SQL usados en cargas, etc.; para las aplicaciones del usuario final nos preocuparemos por administrar una nomenclatura de negocio y descripciones claras de columnas y tablas, permisos a nivel de consultas y reportes, ayudas para la navegación, perfiles de usuarios de acuerdo a roles, estadísticas de uso de los recursos, etc. En general, la metadata será usada por todos los participantes involucrados en un proyecto de inteligencia de negocios y durante todas sus fases.

3.5.3.5 Enterprise Data Warehouse (EDW)

Es el repositorio de datos corporativo, granular, integrado e histórico que soporta una amplia variedad de actividades para el soporte a las decisiones. Es el centro del universo en un proyecto de data warehousing. Un EDW es corporativo pues sirve a los requerimientos de toda la corporación. No está diseñado para satisfacer las necesidades de un departamento particular. Contiene datos muy granulares que pueden ser presentados de diferentes formas. El nivel de detalle en un EDW es muy bajo, al punto que las diferentes unidades de datos pueden ser combinadas y agregadas en muchas formas diferentes, proveyendo muchas perspectivas para el análisis. Es alimentado desde las aplicaciones (sistemas fuente) que forman el ambiente operacional, pudiendo contener un gran volumen de datos históricos que en algunos casos puede alcanzar los 10 años.

Quizá la característica más importante es que el EDW contiene datos integrados, garantizando consistencia en la definición, estructura, formato y uso de los datos. Por esta razón, el EDW constituye la columna vertebral de todas las otras formas de proceso para el apoyo a la toma de decisiones.

3.5.3.6 Operational Data Store (ODS)

Es un punto de integración para los sistemas operacionales. Esto es especialmente importante para sistemas legados o heredados que han crecido independientes uno del otro. Por ejemplo, un banco que cuenta con sistemas independientes, configurados para soportar diferentes productos: préstamos, cuentas corrientes, ahorros, etc., requiere de un ODS para integrar balances actuales e historia reciente de estas cuentas separadas bajo un único número de cliente (Figura 8).

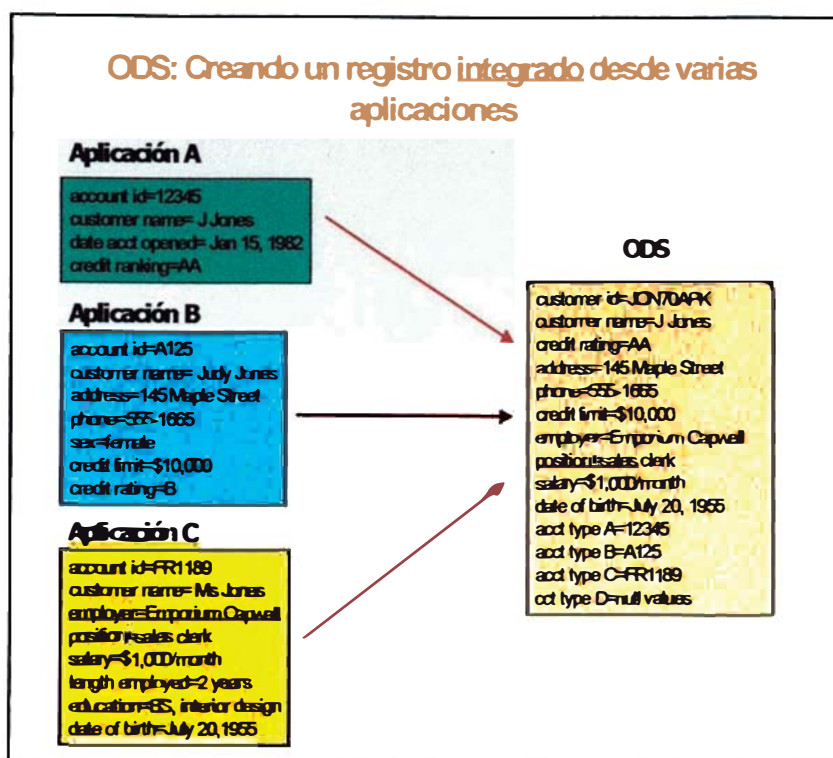


Figura 8. Operational Data Store (ODS)

Un ODS constituye un ambiente analítico e informativo que refleja en cualquier momento el estado operacional actual de su materia sujeto, inclusive cuando los datos que hacen posible ese estado operacional son administrados en diferentes aplicaciones a lo largo de la organización. Si una actualización ocurre en un sistema operacional, el ODS debe realizar esa misma actualización en tiempo real o lo más cercano al tiempo real. El ODS recibe los datos de los sistemas fuente teniendo en cuenta los procesos adecuados de limpieza, transformación y aseguramiento de la calidad y los almacena usando un modelamiento relacional. Un ODS usualmente alimenta al repositorio central de un Data Warehouse.

Cuando se descubre por primera vez el concepto de ODS existe la tentación de tratar de mezclar el ODS con el Data Warehouse. La tentación puede ser fuerte pero se recomienda que en todos los casos el ODS se construya separado del Data Warehouse. No hay condiciones donde el Data Warehouse y el ODS deban

ser combinados. Hay muchas diferencias fundamentales e importantes entre un ODS y un data warehouse. La diferencia más grande entre los dos entornos está en el contenido y la estructura de los datos encontrados en un ODS y un Data Warehouse. El ODS contiene datos actualizados o casi actualizados, mientras que el Data Warehouse contiene información histórica, así como también datos recientemente actualizados. El ODS contiene casi exclusivamente todos los datos en detalle mientras que el Data Warehouse contiene datos en detalle y resumizados. El ODS puede ser actualizado mientras que el Data Warehouse contiene registros que son instantáneas no volátiles, las cuales no se modifican y sólo son consultadas. Adicionalmente, existe una cantidad importante de datos operacionales que residen en un ODS que no serán usados nunca para un procesamiento de apoyo a la toma de decisiones y por lo tanto no serán cargados en el Data Warehouse.

3.5.4 Procesos

3.5.4.1 Minería de Datos

Las aplicaciones de minería de datos (Data Mining) constituyen aplicaciones de bases de datos que buscan patrones ocultos en grupos de datos que pueden ser usados para predecir el comportamiento futuro. Por ejemplo, los productos para minería de datos pueden ayudar a las compañías de venta al menudeo a encontrar clientes con intereses comunes, o a un banco a identificar las variables que más influyen en la rentabilidad de los clientes. El término es comúnmente mal usado para describir al software que presenta los datos en diferentes formas. Los productos de minería de datos reales no sólo cambian la presentación, sino que descubren relaciones que eran desconocidas para los datos analizados. Los parámetros de la minería de datos incluyen:

- Asociación: busca patrones donde un evento está conectado a otro evento.
- Secuencia o ruta de análisis, busca patrones donde un evento conlleva a otro evento posterior.
- Clasificación: busca nuevos patrones y puede resultar en un cambio en la forma como los datos están organizados.
- Segmentación (Clustering): encuentra y documenta visualmente grupos de hechos que no eran conocidos previamente.
- Predicción: descubre patrones en los datos que pueden llevar a predicciones razonables sobre el futuro.

3.5.5 Diseño y modelamiento

3.5.5.1 Modelamiento dimensional

Es una disciplina específica para el modelamiento de los datos que constituye una alternativa al modelamiento entidad-relación (E/R). Un modelo dimensional contiene la misma información que un modelo E/R pero empaqueta los datos en un formato simétrico cuyas principales metas de diseño son un mejor entendimiento por parte del usuario, alto rendimiento en consultas y flexibilidad al cambio. Los principales componentes de un modelo dimensional son las tablas de hechos (Fact Tables) y las tablas de dimensiones (Dimension Tables).

En una base de datos relacional, un modelo dimensional se representa por un esquema Estrella, caracterizado por una tabla de hechos central con una llave compuesta (una llave formada por más de una columna, como se ve en la Figura 9) o por un esquema SnowFlake que es una variante del esquema Estrella. En una base de datos multidimensional, el mismo modelo dimensional se implementa como un cubo (Figura 10).

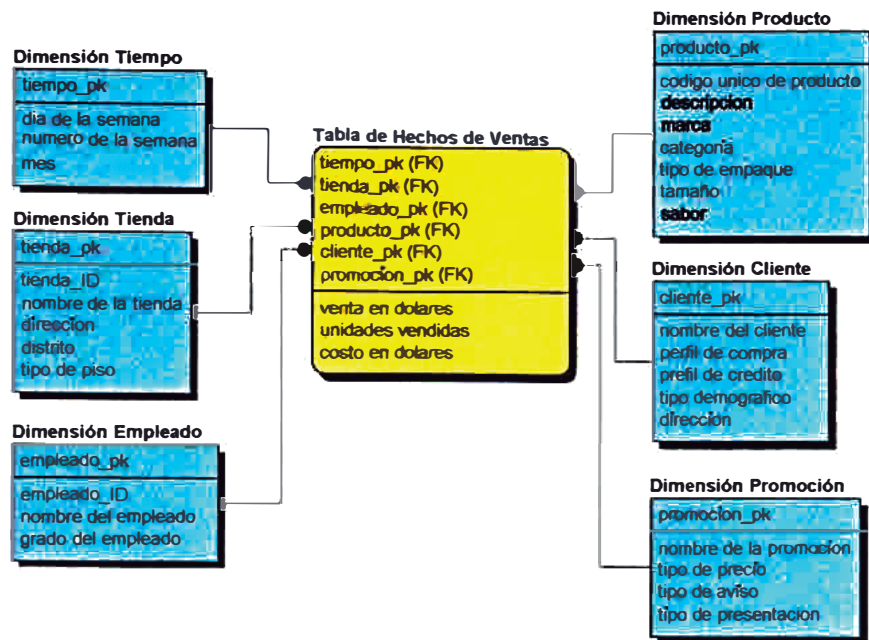


Figura 9. Modelo Dimensional en un ambiente relacional – Modelo Estrella

Los modelos dimensionales soportan la implantación de un Data Mart. La actividad general de consultar, presentar y analizar la información almacenada en los modelos dimensionales es conocida como Procesamiento Analítico en Línea (OLAP, OnLine Analytic Processing). Cuando el modelo dimensional reside en una base de datos relacional se conoce como ROLAP (Relational OLAP). Mucha de la tecnología de los proveedores OLAP del mercado es no-relacional y casi siempre está soportada por una base de datos multidimensional propietaria que maneja “cubos” de datos. Esta tecnología es conocida como MOLAP (Multidimensional OLAP). Cuando el producto OLAP es capaz de usar repositorios de datos relacionales y multidimensionales se conoce como un esquema HOLAP (Hybrid OLAP). El OLAP híbrido fue desarrollado para combinar la gran capacidad de manejo de datos de un esquema ROLAP con la capacidad superior de proceso y cálculo del MOLAP. Un esquema ROLAP puede manejar un volumen de datos muy grande pero sacrifican los tiempos de respuesta, mientras que los esquemas MOLAP brindan alto rendimiento en la resolución de consultas de

negocios al tener mucha información precalculada en los cubos, pero no pueden escalar a los volúmenes de datos que almacena una base de datos relacional corporativa.

3.5.5.2 Tabla de hechos (Fact Table)

Es una tabla primaria en un modelo dimensional que contiene los indicadores o métricas del negocio. Usualmente se usa directamente el término “hecho” (fact) para representar una métrica del negocio en un modelamiento dimensional. Los indicadores más útiles son aquellos que son numéricos y aditivos (ventas, costo del producto, transacción bancaria, etc.). La aditividad es crucial porque las aplicaciones de inteligencia de negocios casi nunca recuperan un único registro de la tabla de hechos. En vez de eso, leen y procesan cientos, miles e inclusive millones de estos registros a la vez, y la sumarización de los valores de los campos es una actividad constante y obligada. Cada tabla de hechos representa una relación muchos-a-muchos y contiene un conjunto de dos o más llaves foráneas que la unen con sus respectivas tablas de dimensiones. En el ejemplo de la Figura 9 la tabla de Hechos de Ventas tiene tres medidas: venta en dólares, unidades vendidas y costo en dólares.

3.5.5.3 Tabla de dimensión (Dimension Table)

Es el segundo tipo de tabla que encontramos en un modelo dimensional y que guarda toda la información requerida para el análisis y navegación por parte del usuario. Las tablas de dimensiones contienen muchos atributos textuales (campos) que son la base para las restricciones y agrupamientos en las consultas de una aplicación de inteligencia de negocios. Cada dimensión es definida por su llave primaria que sirve como la base para la integridad referencial con una determinada tabla de hechos con la cual está relacionada, tal como se muestra en la Figura 9. En un modelo dimensional podemos encontrar dimensiones como el Tiempo, Producto, Cliente, Transacción, Segmento, etc.

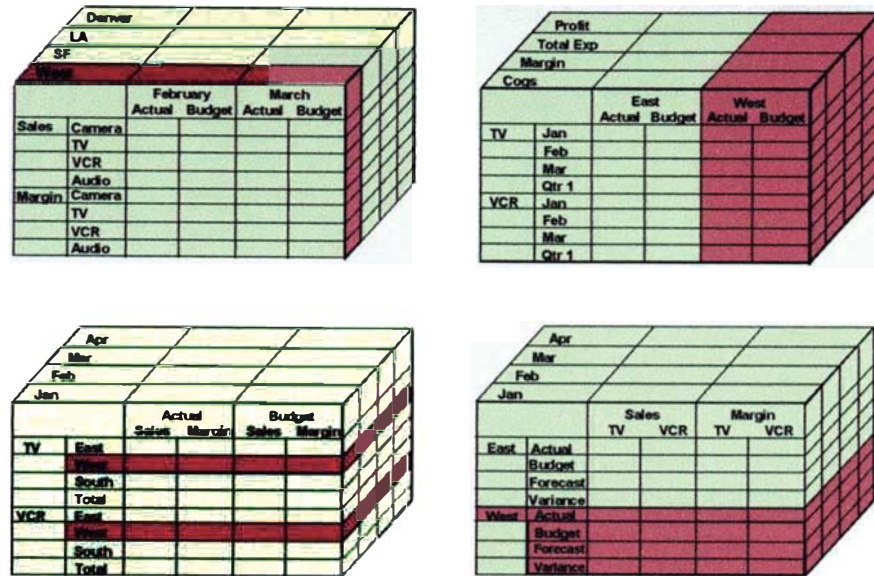


Figura 10. Modelo dimensional en una base de datos multidimensional – Cubos.

Las dimensiones ayudan a responder preguntas de negocios del tipo *¿cuál ha sido el nivel de ventas y las unidades vendidas en el primer trimestre del año por marca de productos?*. Para responder esta pregunta es necesario consultar la tabla de hechos de Ventas para obtener las medidas solicitadas (ventas y unidades vendidas), restringiendo el número de filas devueltas de acuerdo a las restricciones en los atributos de las dimensiones Producto y Tiempo. Un típico resultado de una consulta en un entorno de inteligencia de negocios se vería como el siguiente cuadro:

Marca	Venta en Dolares	Unidades Vendidas
Axon	780	263
Framis	1044	509
Widget	213	444
Zapper	95	39

Cuadro 2. Resultado de una consulta en un ambiente de inteligencia de negocios.

En una base de datos relacional la sentencia SQL que generaría el resultado anterior sería algo como:

Select p.marca,	← lista de campos a retomar
sum(f.ventas),	
Sum(f.unidades)	
From ventas_fact f,	← cláusula "from" con alias f, p y t
producto p,	
tiempo t	
Where f.producto_pk = p.producto_pk	← restricción para el join
And f.tiempo_pk = t.tiempo_pk	← restricción para el join
And t.trimestre = "1 Q 2001"	← restricción de la aplicación
Group by p.marca	← generar la suma agrupada por marca
Order by p.marca	← ordenar por marca

Las dimensiones son organizadas y agrupadas en jerarquías. Por ejemplo la dimensión Tiempo puede considerar el atributo *Días* en el primer nivel de la jerarquía, *Meses* en el segundo nivel, *Trimestres* en el tercer nivel y en el nivel más alto *Años*. Para la dimensión Geografía se podría tener una jerarquía como la mostrada en la Figura 11.

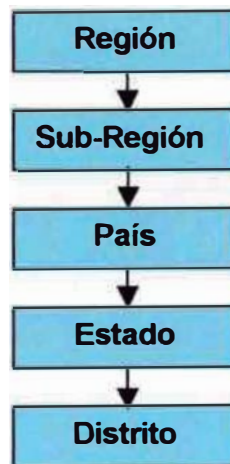


Figura 11. Ejemplo de una jerarquía en una dimensión geográfica.

Los usuarios de las aplicaciones de inteligencia de negocios navegan a través de la jerarquía de las dimensiones en una operación conocida como *"drilling down"*. Hacer *"drilling down"* sobre el resultado de una consulta quiere decir mostrar más detalle (o menos detalle en el caso del *"drilling up"*) para explicar

algo resaltante en los datos. En este contexto se puede entender que “drill down” significa agregar otro encabezado a un reporte existente, donde este encabezado corresponde a un atributo de una dimensión que pertenece a la misma jerarquía o a una jerarquía distinta. Por ejemplo, si quisiéramos consultar las ventas en dólares por tamaño de paquete para los productos de la Marca “Brawny”, obtendríamos un resultado como el del siguiente cuadro:

Marca	Tamaño Paquete	Ventas
Brawny	2-Paquetes	\$50
Brawny	3-Paquetes	\$110
Brawny	6-Paquetes	\$75

Cuadro 3. Ventas en dólares por tamaño de paquete del producto Brawny.

Luego, si adicionamos el atributo Color perteneciente a la dimensión Producto en el mismo reporte, éste se expandiría mostrándonos más detalle de las ventas para un tamaño de paquete dado y color determinado:

Marca	Tamaño Paquete	Color	Ventas
Brawny	2-Paquetes	Blanco	\$8
Brawny	2-Paquetes	Marrón	\$5
Brawny	2-Paquetes	Verde	\$37
Brawny	3-Paquetes	Blanco	\$22
Brawny	3-Paquetes	Verde	\$88
Brawny	6-Paquetes	Blanco	\$14
Brawny	6-Paquetes	Rosado	\$12
Brawny	6-Paquetes	Marrón	\$4
Brawny	6-Paquetes	Verde	\$45

Cuadro 4. Ventas en dólares por tamaño de paquete y por color del producto Brawny.

3.5.5.4 Esquema Estrella

El esquema estrella (star schema) es la representación más conocida de un modelo dimensional. En este esquema las tablas de dimensión tienen llaves primarias simples que se enlazan con la llave primaria compuesta de una tabla de hechos central. El

esquema estrella es la forma más simple de modelar en inteligencia de negocios. Es llamado un esquema estrella debido a que el diagrama entidad-relación de este esquema se asemeja a una estrella, tal como se vio en la Figura 9.

Un esquema estrella está caracterizado por una o más tablas de hechos muy grandes que contienen la información primaria de un modelo de inteligencia de negocios (indicadores) y un número de tablas de dimensiones más pequeñas (también conocidas como tablas lookup), cada una de las cuales contiene información acerca de las entradas para un atributo particular en la tabla de hechos.

Una consulta en un esquema estrella necesita la unión entre la tabla de hechos y un número de tablas de dimensiones. Cada tabla de dimensión es unida a la tabla de hechos usando la llave primaria de la dimensión y la llave foránea en la tabla de hechos. Esta unión o "join" es conocido como un "star join".

Las principales ventajas de un esquema estrella es que provee un mapeo directo e intuitivo entre las entidades del negocio analizadas por los usuarios finales y el diseño del esquema. Además provee un rendimiento usualmente optimizado para la mayoría de consultas típicas en un entorno de inteligencia de negocios.

3.5.5.5 Esquema Snowflake

La forma clásica de crear un esquema Snowflake es remover los atributos de baja cardinalidad de una tabla de dimensión y colocarlos en una tabla de dimensión secundaria que se unirá con la tabla de dimensión original a través de una llave artificial, conocida como "llave snowflake", generando una estructura de tablas que lucirán como un copo de nieve (De allí el nombre de Snowflake). Los atributos de baja cardinalidad de una dimensión

son aquellos cuyo dominio tiene un número reducido de posibles valores y que usualmente se verán repetidos en las filas de la tabla de dimensión. Uno de los objetivos de este esquema es normalizar las tablas de dimensiones para eliminar esta redundancia. En la Figura 12 se muestra un diagrama que representa como las dimensiones Producto, Cliente y Tienda han sido descompuestas para generar un esquema snowflake, creándose nuevas tablas (Tipo de empaque, Marca, Categoría, Sabor, Distrito, Tipo demográfico) que tienen llaves artificiales que las unen con las tablas de dimensiones originales a través de llaves foráneas.

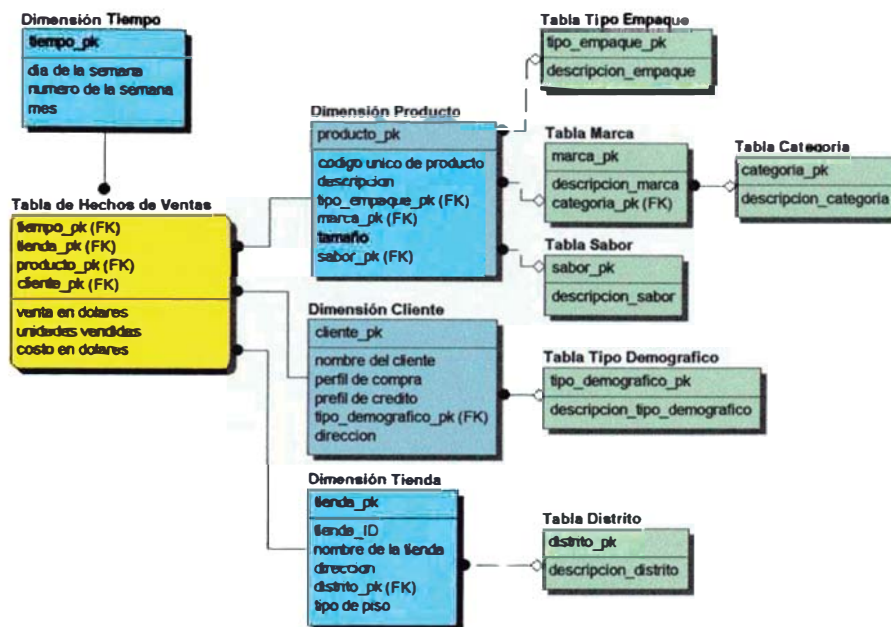


Figura 12. Esquema Snowflake (Copo de Nieve)

Algunos autores no recomiendan el uso del esquema Snowflake en un ambiente de inteligencia de negocios debido a que casi siempre hace que la presentación al usuario sea más compleja y confusa. Los diseñadores de bases de datos frecuentemente gustan de esta complejidad, pero los usuarios se ven intimidados con ella. El esquema Snowflake también hace que la mayoría de las formas de consultar los datos a través de los atributos de las tablas de dimensiones sea más lento que en el modelo estrella al tener que usar más joins para unir más tablas. Algunos

diseñadores de bases de datos utilizan este esquema con el fin de ahorrar espacio en disco al eliminar todos los campos de texto redundantes en una tabla de dimensión. Esto tiene sentido siempre que la nueva llave artificial requerida en la nueva tabla para cada entrada sea más corta que la cadena de texto que reemplaza y que la ganancia de espacio justifique la complejidad agregada al modelo. Por ejemplo, supongamos que la dimensión Producto cuenta con 250,000 filas y que reemplazamos los nombres de categoría (con un ancho de 15 bytes) con una llave artificial de 2 bytes. Con esta operación hemos ahorrado aproximadamente $250,000 \times 13$ bytes, es decir, unos 3.25 Megabytes. Posiblemente la tabla Producto en su totalidad sea mayor a 100 Megabytes y la tabla de hechos de Ventas asociada ha de tener al menos unos 10 Gigabytes. Con la operación anterior hemos ahorrado 3 Megabytes sobre una base mayor a los 10 Gigabytes, es decir un 0.03%. Este ahorro, ahora cuestionable, ha sido costado con un paso extra de administración al crear y poblar la llave artificial en la nueva tabla, así como con una penalidad severa en el rendimiento cuando se efectúen consultas que crucen los atributos de un producto.

A pesar de todas estas razones para evitar el uso de un esquema Snowflake, existen algunas situaciones donde es necesario construir una "subdimensión" o tabla Snowflake. Si los datos de los atributos que pensamos colocar en esta subdimensión están disponibles a un nivel de detalle diferente al de los datos primarios del modelo dimensional y son administrados, e incluso cargados, en momentos diferentes de tiempo que el resto de los datos de la dimensión original; si se logran ahorros significativos de espacio en disco y si los atributos a ser colocados en la subdimensión están altamente relacionados y son casi siempre consultados de forma conjunta, entonces se justifica crear una tabla Snowflake que albergue estos atributos y separarlos de la dimensión original. Un ejemplo común de este esquema lo

tenemos en la dimensión Cliente donde usualmente se crean subdimensiones para los atributos demográficos, los atributos de ubicación, categoría, segmento, etc.

3.5.6 Extracción, Transformación y Carga

Los procesos de extracción, transformación y carga de datos tienen lugar en lo que se conoce como el área de escenificación o preparación de datos (Data Staging Area), a la cual vamos a referirnos simplemente como el área de staging. La preocupación principal de los administradores de base de datos y administradores de sistemas en esta área está relacionada con resolver el problema específico de obtener los datos correctos desde el punto A y llevarlos al punto B, con las transformaciones adecuadas y en el mejor tiempo posible.

Los procesos en el Area de Staging empezaron como un esfuerzo de desarrollar código manualmente en lenguajes como COBOL o C. Afortunadamente la industria ha desarrollado algunas herramientas y técnicas para soportar estos procesos, que permiten automatizarlos en gran medida a través de interfaces gráficas que ocultan mucha de la complejidad al usuario final y le dan la opción de calendarizar ciertos trabajos o “jobs” para ser ejecutados de manera desatendida y con cierta frecuencia. Estas herramientas reciben el nombre de herramientas ETL por las siglas en inglés de Extract, Transformation & Load (también son conocidas como “Data Staging Tools”).

3.5.6.1 Extracción

La extracción de los datos desde los sistemas fuente es probablemente el esfuerzo más grande en un proyecto de data warehousing, especialmente si los sistemas fuentes son muy antiguos o sistemas que simplemente son un misterio para todos en la forma cómo operan y cómo almacenan los datos. De acuerdo a la experiencia de los diferentes autores, cerca del 60 por ciento de las horas de desarrollo de una solución de inteligencia de negocios son empleados en el proceso de extracción.

Frecuentemente, el reto es determinar qué datos extraer y qué tipo de filtros aplicar. Todos conocen historias acerca de campos que tienen múltiples usos, valores que no deberían existir, pagos hechos desde cuentas que no han sido creadas, etc. Desde un punto de vista de arquitectura, necesitamos entender los requerimientos del proceso de extracción de modo que podamos determinar qué tipos de servicios serán necesarios. La mayoría de procesos de extracción generan archivos de carga temporales que llegan a ser la entrada para la siguiente actividad. Así encontramos que extracciones basadas en plataformas host o mainframe son diseñadas para obtener datos y sacarlos de la fuente, llevándolos hacia una forma más simple y fácilmente accesible, como un archivo plano.

3.5.6.1.1 Múltiples Fuentes

Es difícil encontrar una solución de inteligencia de negocios que no extraiga datos desde múltiples fuentes. En la mayoría de los casos, los datos deben ser sacados de múltiples sistemas construidos con diferentes almacenamientos de datos y que están corriendo en diferentes plataformas. Inclusive los afortunados que extraen datos de un paquete integrado cliente/servidor deben todavía lidiar con múltiples plataformas y versiones.

3.5.6.1.2 Generación de Código

En algunos casos la generación de código significa crear algunos programas que se ejecuten en los sistemas fuentes para crear los archivos de extracción (como COBOL en el mainframe). Para algunas herramientas ETL la generación de código significa código interno que es ejecutado por el motor de extracción el cual en su momento lanza los comandos contra los sistemas fuente. Es importante asegurarse que las herramientas con las cuales se planea trabajar puedan hacerlo en las plataformas donde se

ejecutan los sistemas fuentes y que son capaces de generar el código de extracción necesario para obtener los datos en un formato neutral, tal como un archivo plano.

3.5.6.1.3 Tipos de Extracción

Diferentes tipos de extracción sirven diferentes propósitos al momento de construir un warehouse:

- **Cargas incrementales:** Muchos entornos de inteligencia de negocios usan cargas incrementales. Esto es típicamente basado en la fecha de transacción o algún tipo de indicador (“flag”) en el sistema fuente. La fecha de la última carga es una pequeña pieza de metadata que debe tenerse a la mano durante el proceso de carga. Es mucho más eficiente cargar incrementalmente únicamente los registros que han cambiado o han sido adicionados desde la carga anterior, sobre todo cuando el tamaño de las tablas ha alcanzado niveles donde la carga completa de todo su contenido tomaría demasiado tiempo.
- **Transacciones:** La extracción debe ser capaz de identificar diferentes tipos de transacciones debido a que los procesos de transformación y carga tendrán que tratarlos de manera diferente. Desde que la última carga se llevó a cabo, nos debe interesar ver todas las nuevas transacciones, registros modificados y registros eliminados (los registros eliminados puede indicar un estado de cancelación). En algunos casos, los sistemas fuente no tienen una marca de fecha confiable, por lo tanto no se puede saber si algo ocurrió después de la última carga de datos. Para algunos casos puede ser útil considerar el uso del archivo de log de transacciones como una ayuda para el proceso de extracción.
- **Full Refresh:** Algunas veces es necesario considerar la carga de ciertas tablas en su totalidad en vez de preocuparnos en descubrir qué es lo que ha cambiado.

Obviamente, existen límites en el tamaño para esta estrategia, pero inclusive en el caso de grandes fuentes de datos tiene sentido resincronizarlas después de un período de tiempo.

3.5.6.2 Transformación

Una vez que los datos son extraídos desde los sistemas fuentes, un conjunto de trabajos se lleva a cabo para convertirlos en algo presentable a los usuarios y de valor para el negocio. Puede ser posible, por ejemplo, que datos de servicio al cliente tengan que pasar por más de 20 pasos en el proceso de transformación antes de llegar a un estado digno de ser presentado al usuario final para su análisis y posterior toma de decisiones. La siguiente lista representa muchos de los diferentes tipos de transformaciones que podrían presentarse al momento de construir una solución de inteligencia de negocios:

- **Chequeo de integridad referencial:** La integridad referencial implica que los datos en una tabla correspondan con los datos en otra tabla. Si se tiene en la tabla de hechos el número de producto A45BZ55, es necesario tener un producto en la tabla de dimensión Producto con el mismo número o no se podría analizar las ventas de este producto. Efectivamente, si no hay un registro en la tabla de dimensión con el número del producto, un usuario puede fácilmente construir una consulta que omitirá esta venta sin siquiera darse cuenta de ello.
- **Denormalización:** Denormalizar una jerarquía de tablas separadas en una sola dimensión es un proceso estándar de transformación. Algunas de las herramientas del mercado ofrecen características de esquema estrella que automáticamente llevan a cabo esta función. En adición, cierto nivel de denormalización tiene lugar en la tabla de hechos.
- **Limpieza y Calidad de datos:** Este es un gran problema para las soluciones de inteligencia de negocios, especialmente para aquellos afectados con entidades externas como clientes,

negocios, doctores y pacientes. Este punto será cubierto en detalle más adelante.

- **Conversión de tipos de datos:** Este proceso involucra transformaciones al nivel más bajo, al convertir un tipo de dato o formato en otro. Contempla desde las conversiones de los conjuntos de caracteres EBCDIC de los mainframes IBM en caracteres ASCII, hasta las conversiones de las representaciones de fechas, números y caracteres de una base de datos a otra.
- **Cálculo, derivación, asignación:** Hay transformaciones que buscan poner en práctica las reglas de negocio que se identifican durante el proceso de requerimientos. Es importante asegurarse que la herramienta ETL que se elija tenga un juego completo de funciones disponibles, incluyendo manipulaciones de cadenas, aritmética con fecha y tiempo, sentencias condicionales y matemática básica. El poder extender estas funciones por parte del usuario es una característica muy valiosa.
- **Agregados:** La manera más directa de afectar favorablemente el rendimiento en un gran repositorio de datos es tener un apropiado conjunto de registros agregados (sumarizados) que coexistan con los registros base primarios. Un agregado es un registro de una tabla de hechos que a su vez representa una sumarización de un conjunto de registros de la misma tabla de hechos pero de un nivel más atómico. Una institución bancaria podría desarrollar su Data Warehouse registrando el monto de cada transacción de sus clientes (retiros, depósitos, compras con tarjeta de crédito, pago de una letra de un préstamo, etc.) y luego crear agregados por sucursal, por segmento, por mes, etc. para mejorar los tiempos de respuesta de los reportes más solicitados.
- **Auditoria del Contenido de los Datos:** El proceso de transformación debería llevar acabo chequeos de totales, conteos de filas y pruebas de validación como una tarea natural. Estos resultados luego deben ser comparados con los resultados generados a partir de los sistemas fuente y generar alertas en caso no coincidan.

- **Valores Nulos:** Los valores nulos pueden ser un problema porque muchos sistemas fuente no tenían una forma para representarlos. Para manipular valores nulos los programadores escogen un valor particularmente improbable (como 9/9/99 para una fecha o -1 para un número de un producto). En vista que todas las interacciones con los datos en los sistemas fuente son moderadas a través de un programa, es fácil aplicar la apropiada interpretación para un valor nulo. Ahora, si durante el proceso de creación de un Data Warehouse se ponen datos directamente desde las fuentes, estos valores improbables lucirán como valores legítimos. Los análisis darán resultados incorrectos y se tomarán decisiones equivocadas. Es necesario identificar estos valores nulos que han sido substituidos y desarrollar reglas para manipularlos en la base de datos.

3.5.6.3 Carga de Datos

Las capacidades que se necesiten durante el proceso de carga son, en gran medida, una función de la plataforma destino. Algunas de las características necesarias durante este proceso de carga son:

- **Soporte para múltiples destinos:** Los datos atómicos de un Data Mart pueden estar un servidor de base de datos y los Data Marts del proceso de negocios podrían estar en otros. Cada destino probablemente tendrá su propia sintaxis y sus propias reglas y el proceso de carga debería conocer estas diferencias y utilizarlas o evitarlas, según sea el caso.
- **Optimización de Cargas de Datos:** La mayoría de servidores de base de datos del mercado (DB2, Oracle, Informix, etc.) tienen capacidades para llevar a cabo cargas masivas (lo que se conoce como "Bulk Load") que incluyen un conjunto de características y que pueden ser puestas en un script o invocadas por la herramienta ETL a través de un API. Cada producto de base de datos tiene un juego de técnicas y trucos que permiten optimizar el rendimiento durante la carga masiva de datos y es fundamental estar conscientes de ellas y utilizarlas.

- **Soporte para el proceso de carga:** Los servicios de carga de datos también necesitan soportar requerimientos antes y después de la carga actual, tales como borrar y re-crear los índices y particionamiento físico de tablas e índices.

3.5.7 Acceso y Explotación de la Información

Las herramientas para el usuario final constituyen la cara pública de una solución de inteligencia de negocios. Es lo que los usuarios de negocios ven y con lo que trabajan día a día. De hecho, para muchos, la interfaz del usuario es la solución en sí. Ellos no saben ni se preocupan por todo el tiempo, energía y recursos detrás de ella. Lo único que desean son respuestas.

La meta principal de una solución de inteligencia de negocios debería ser poner la información lo más accesible posible para ayudar a las personas a obtener la información que necesitan. Para cumplir con este requerimiento, es necesario construir una capa entre los usuarios y la información que ocultará algo de la complejidad y los ayudará a encontrar lo que están buscando.

Los servicios a tener en cuenta cuando se evalúan herramientas para los usuarios finales usualmente son:

- **Navegación:** La navegación toma ventaja del catálogo de metadata para soportar a los usuarios en sus esfuerzos por acceder y encontrar la información que necesitan. Muestra las áreas materia del warehouse y los elementos pertenecientes a cada área.
- **Acceso y Seguridad:** Se basa en los servicios de autorización y autenticación dónde el usuario es identificado y los privilegios de acceso son asignados de acuerdo a su perfil o es rechazado su acceso si no tiene los permisos respectivos.
- **Monitoreo:** Involucra la captura de información sobre el uso de la solución de inteligencia de negocios.

- **Administración de Consultas:** Constituyen un conjunto de capacidades que manejan el intercambio entre la formulación de la consulta, su ejecución en la base de datos y el retorno del resultado hacia la computadora del usuario final. Entre las capacidades se encuentra la simplificación del contenido, la reformulación de consultas complejas, consultas a múltiples destinos, el sacar provecho de los agregados y el control de las consultas.
- **Reportes:** Los requerimientos para las herramientas de reporte estándar (reporting tools) incluyen:
 - Entorno de desarrollo de reportes
 - Servidor para la ejecución de reportes
 - Capacidades para el control de reportes por parámetros o por variables
 - Programación basada en tiempo o en eventos para la ejecución de los reportes
 - Ejecución iterativa, por ejemplo dada una lista de regiones y teniendo el mismo reporte para cada región, la herramienta genera cada reporte como un archivo para ser entregado a diferentes personas
 - Definiciones flexibles para el diseño del reporte
 - Entrega flexible del reporte:
 - A través de múltiples métodos de entrega (e-mail, web, directorio en la red, fax)
 - En la forma de diferentes tipos de resultados (archivos planos, tabla de una base de datos, hoja de cálculo)
 - Usuario con la capacidad de publicar sus reportes
 - Vínculos entre reportes, para obtener capacidades de drill-up y drill-down a través de reportes existentes, con menos o más nivel de detalle en la información.
 - Biblioteca de reportes con capacidades de navegación, que describe cada reporte, cuándo fue ejecutado, quién lo ejecutó y qué contiene.
 - Distribución masiva basada en Web

- Herramientas de administración del entorno de reportes
- **Servicios para Computadoras Personales (Desktop):** Sólo unos cuantos servicios residen hoy en una PC, pero ellos son de lejos los servicios más importantes en el mundo de inteligencia de negocios. Mucha de la calificación que le dé un usuario a su experiencia completa con una solución de inteligencia de negocios será determinada por cuán bien estas herramientas satisfacen sus necesidades. Estas herramientas tendrán que atender a diferentes tipos de usuarios desde el punto de vista tecnológico dentro de una empresa y además atenderán diferentes necesidades de información (Cuadros 5).

Necesidad de Información	Rol del Usuario	Acceso a los datos	Herramientas Comunes	Tamaño de la audiencia
Monitoreo a alto nivel-medidas claves, indicadores	Gerencias General, Directorio	"Push-Button", consola, panel de control	Interfaz tipo un sistema de información ejecutiva (EIS)	Pequeña
Seguimiento a los negocios, mercados, productos, clientes, etc.; acceso al detalle	Gerencia media, fuerza de ventas, gerencia de marketing, gerentes de negocios, representantes de servicio al cliente, etc.	Reportes Estándares conducido por parámetros	Herramientas de reporte; herramientas OLAP con capacidades de programación de reportes; entornos de consultas	Grande
Investigación-excepciones, nuevos problemas u oportunidades; desarrollo de caso de negocios	Algunos de los anteriores más Analistas de Negocios	Análisis Ad hoc	Herramientas OLAP; entornos de consultas; herramientas de análisis de alta capacidad	Mediana
Análisis Complejo-consultas compuestas, análisis estadísticos, desarrollo de modelos	Analistas de Negocios y expertos analíticos	Minería de Datos (Data Mining), análisis avanzado	Herramientas de análisis de alta capacidad; herramientas estadísticas; herramientas de minería de datos	Pequeña

Cuadro 5. Necesidades de información y Roles que las herramientas de inteligencia de negocios deben atender en una organización

3.5.8 Calidad y Limpieza de Datos

La calidad de datos depende de una cadena de eventos, muchos de los cuales están fuera del control del equipo de inteligencia de negocios. Para obtener datos de calidad el proceso de recolección de datos debe estar bien diseñado, y los recursos (usualmente personas) quienes ingresan los datos deben estar comprometidos con la calidad de la información que entregan.

Es irreal esperar que un sistema contendrá datos perfectos. Cada implementación debe definir sus propios estándares de aceptación de calidad de datos. Al definir estos estándares es útil entender las características que se esperan de los datos:

- **Preciso.** Los datos en una solución de inteligencia de negocios coinciden con los registros en los sistemas fuente. En los casos en que los valores sean diferentes, lo cual puede presentarse, se tendrá un rastro de auditoría documentado que explique las diferencias.
- **Completo.** Los datos representan el conjunto completo de datos relevantes, y los usuarios han sido notificados del alcance de los datos. Por ejemplo, un dato llamado "Ganancias Corporativas Totales" debería incluir todas las divisiones de una organización. Si no se tuvieran los datos de todas las divisiones, entonces se debería renombrar el dato, pues no representa un valor total y puede llevar a confusiones.
- **Consistente.** Los datos están libres de contradicciones. Por ejemplo, los agregados están representando fielmente a sus correspondientes datos detallados.
- **Único.** Si dos elementos representan lo mismo, deberían ser llamados de la misma manera y tener el mismo valor. Por ejemplo, I.B.M. e IBM son la misma compañía y debería uniformizarse la forma como se menciona en la solución de inteligencia de negocios.
- **A tiempo.** Los datos deben ser actualizados con una frecuencia que sea la adecuada para los usuarios de negocios.

- **Flexible.** Las definiciones de información están entendidas de tal manera que la información pueda ser analizada de varias formas.

3.5.8.1 Costo de una pobre Calidad de Datos

Cuando los datos se ingresan incorrectamente o no son considerados confiables, los usuarios de la información deben:

- Tomar la misma información de otras fuentes.
- Corregir los errores.
- Verificar la exactitud de la información.

Todas estas actividades consumen personal, tiempo y aumentan el costo. A medida que las empresas se vuelven más competitivas, necesitan asegurar la calidad de datos para apoyar la toma de decisiones y ayudar a prestar mejores servicios a sus clientes. La información de mala calidad puede llevar a una atención al cliente de bajo promedio (Por ejemplo, para una Universidad puede haber estudiantes potenciales que no reciben la correspondencia debido a que su dirección no está bien definida). La información de mala calidad también puede dar como resultado una ineficiente toma de decisiones y potencialmente una mala reputación.

3.5.8.2 Mejorando la Calidad de Datos

El primer paso para mejorar la calidad de los datos institucionales es darse cuenta que el problema existe. La temas de calidad de datos usualmente se vuelven evidentes cuando una institución esta planeando un proyecto de inteligencia de negocios. Para mejorar la calidad de datos las empresas deben de estar preparadas para:

- ✓ Observar detenidamente la práctica de su negocio relacionado a la recolección y almacenamiento de los datos.

- ✓ Establecer un custodio de los datos (dueño) por cada pieza de información que se recolecta y hacer responsable a cada custodio de la calidad de sus datos.
- ✓ Realizar un análisis profundo y objetivo de la calidad de datos existente.
- ✓ Corregir los problemas de calidad de datos existentes.
- ✓ Hacer cambios permanentes a la práctica del negocio para mejorar el futuro de los datos.
- ✓ Hacer revisión y mejoramiento de calidad de datos y procesos ya existentes.

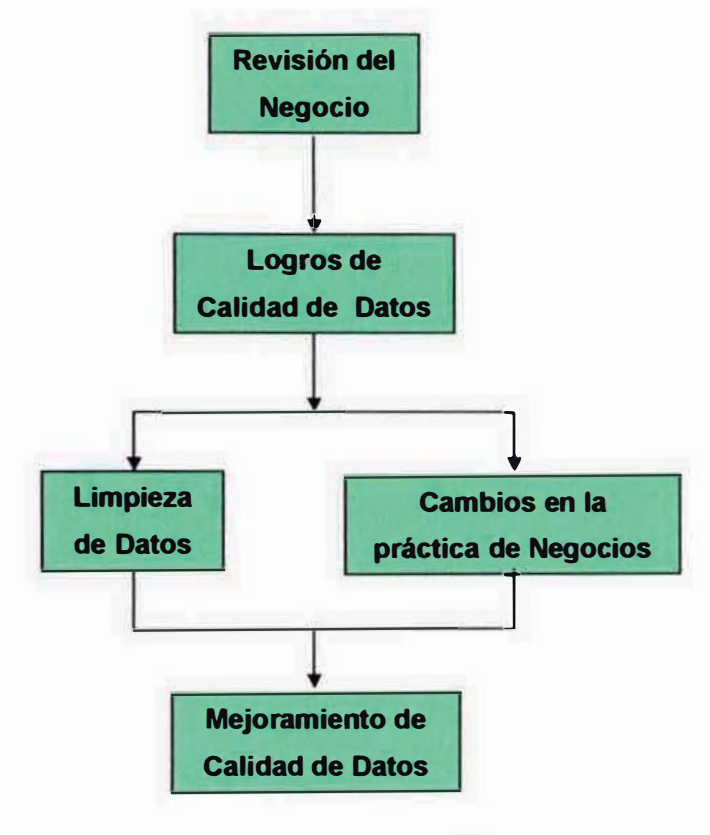


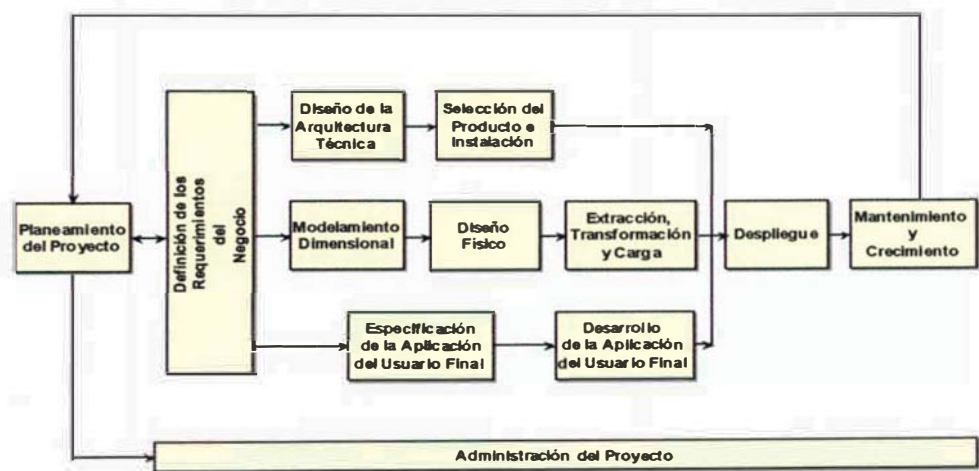
Figura 13. Ciclo de un proceso de limpieza y calidad de datos

3.5.9 Ciclo de vida de un proyecto de Inteligencia de Negocios

Una vista completa del ciclo de vida para la implementación de un proyecto de inteligencia de negocios se muestra en la Figura 14. Este diagrama presenta la secuencia de tareas de alto nivel requeridas para

un diseño, desarrollo y despliegue efectivo de un proyecto de inteligencia de negocios. Este enfoque corresponde al “Business Dimensional Lifecycle” presentado por Ralph Kimball en 1998. El diagrama de este ciclo de vida presenta el flujo general que ocurre durante la implementación de un proyecto de inteligencia de negocios. Identifica la secuencia de tareas a alto nivel y resalta las actividades que deberían llevarse a cabo concurrentemente en tres caminos: tecnología, datos y aplicación. Por ejemplo, no se debería intentar completar el diseño físico de los datos y pasar al área de proceso de los datos (data staging area) sin un entendimiento claro de los requerimientos del negocio, un diseño general de la arquitectura técnica, y la selección de arquitectura técnica específicas, incluyendo aquí la herramienta de acceso a los datos del usuario final.

El Ciclo de Vida Dimensional del Negocio (*)



(*) The Business Dimensional Lifecycle, Ralph Kimball, 1998

Figura 14. Ciclo de Vida de un proyecto de inteligencia de negocios propuesto por Ralph Kimball

Este ciclo de vida, sin embargo, no intenta reflejar los tiempos necesarios para cada etapa del proyecto. En realidad, se debería considerar que cada rectángulo mostrado en el diagrama tiene el mismo ancho, con la excepción de la administración del proyecto, que va de inicio a fin. Teniendo en cuenta la experiencia de profesionales

en la implementación de proyectos de inteligencia de negocios se puede concluir que el tiempo y los recursos necesarios para cada rectángulo representado en el diagrama no van a ser los mismos de un proyecto a otro. Cada empresa cuenta con diferentes sistemas, plataformas, procesos, volúmenes de datos y, sobre todo, necesidades diferentes. El foco de este enfoque está centrado en la secuencia y concurrencia de las actividades, y no en tiempos absolutos. Durante la etapa de planeamiento del proyecto se podrá trabajar en un estimado del tiempo para contar con una primera versión de un Data Warehouse que luego irá evolucionando y creciendo, pero siendo este tipo de proyectos relativamente nuevos en las empresas de hoy en día, y sobre todo en nuestro medio, seguro estos estimados serán superados en la mayoría de las veces. Esto no debería desanimar a las empresas a emprender un proyecto de este tipo, sino más bien llamar su atención para concentrarse más en la administración del proyecto y en el manejo de las expectativas durante todo el ciclo de vida.

3.5.9.1 Planeamiento del Proyecto

El ciclo de vida empieza con el plan del proyecto, como era de esperarse. El plan del proyecto presenta la definición y alcance del proyecto, incluyendo la evaluación de factibilidad y la justificación de negocios. Hay tareas críticas preliminares debido a la alta visibilidad y costos asociados a la mayoría de proyectos de inteligencia de negocios. Desde allí, el planeamiento del proyecto se focaliza en los requerimientos de recursos y las habilidades requeridas en el equipo, junto con la asignación de tareas, duración y secuencia de los pasos a seguir. El plan del proyecto resultante debe identificar todas las tareas asociadas en el ciclo de vida y todas las partes involucradas. El plan constituye la piedra fundamental para la administración futura de un proyecto de inteligencia de negocios. El planeamiento del proyecto es dependiente de los requerimientos del negocio, como lo denota la flecha de doble sentido entre estas dos actividades de la gráfica anterior.

3.5.9.2 Definición de los requerimientos del negocio

La probabilidad de éxito de un proyecto de inteligencia de negocios se ve altamente incrementada por un entendimiento claro de los usuarios finales del negocio y sus requerimientos. Sin este entendimiento, un Data Warehouse probablemente llegará a ser un ejercicio inútil para el equipo del proyecto. El método usado para obtener un conocimiento real de los requerimientos analíticos de los usuarios finales difiere significativamente de los más tradicionales análisis de requerimientos orientados a los datos. Los diseñadores de un Data Warehouse deben entender los factores claves que mueven el negocio para determinar efectivamente los requerimientos de negocios y traducirlos en las consideraciones del diseño. Los requerimientos del negocio establecen las bases para los tres caminos paralelos focalizados en tecnología, datos y las aplicaciones del usuario final.

3.5.9.3 Modelamiento Dimensional

La definición de los requerimientos del negocio determinan los datos necesarios para atender los requerimientos analíticos de los usuarios. El diseño de modelos de datos que soporten estos análisis requiere una estrategia diferente que la usada para el diseño de sistemas operacionales. Se debería empezar por construir una matriz que represente los procesos de negocio claves y su dimensionalidad. La matriz servirá como una guía para asegurar que el Data Warehouse es extensible a través de la organización con el transcurso del tiempo. A partir de allí se iniciará un análisis de datos detallado de los sistemas operacionales que son relevantes. Uniendo este análisis de datos con el entendimiento anticipado de los requerimientos del negocio, se desarrolla el modelo dimensional. Este modelo estará formado por el detalle de la tabla de hechos, las dimensiones asociadas, atributos y jerarquías. El diseño lógico de la base de datos es terminado con apropiadas estructuras de tablas y relaciones de llave primaria/foránea. El plan preliminar de

agregación es desarrollado también. Este conjunto de actividades concluye con el desarrollo de un mapeo de datos fuente-destino.

3.5.9.4 Diseño Físico

El diseño físico de la base de datos se focaliza en definir las estructuras físicas necesarias para soportar el diseño lógico de la base de datos. Elementos primarios de este proceso incluyen definir estándares para la nomenclatura y establecer el entorno de base de datos. Las estrategias preliminares de indexación y particionamiento son también determinadas.

3.5.9.5 Extracción, Transformación y Carga

Esta etapa es típicamente la tarea más subestimada en un proyecto de data warehousing. El proceso de extracción siempre saca a luz problemas de calidad de los datos que han sido pasados por alto en los sistema fuente operacionales. En vista que la calidad de los datos impacta significativamente en la credibilidad de una solución de inteligencia de negocios, es necesario considerar estos problemas de calidad durante el procesamiento de los datos. Como una complicación adicional, es necesario diseñar y construir dos procesos de datos: uno para poblar inicialmente el Data Warehouse y otro para las cargas regulares e incrementales.

3.5.9.6 Diseño de la Arquitectura Técnica

Los entornos de inteligencia de negocios requieren la integración de numerosas tecnologías. El diseño de la arquitectura técnica establece el marco global de la arquitectura y la visión. Es necesario considerar tres factores – los requerimientos del negocio, el entorno técnico actual, y las direcciones técnicas estratégicas planeadas – en forma simultánea para establecer el diseño de la arquitectura técnica del Data Warehouse.

3.5.9.7 Selección del Producto e Instalación

Usando el diseño de la arquitectura técnica como un marco, los componentes específicos de la arquitectura, tales como la plataforma de hardware, sistemas de administración de bases de datos, herramientas para el procesamiento de datos o herramientas de acceso a los datos, necesitan ser evaluadas y seleccionadas. Un proceso de evaluación técnica estándar es definido junto con factores de evaluación específicos para cada componente de la arquitectura. Una vez que los componentes han sido evaluados y seleccionados, son luego instalados y probados ampliamente para asegurar una apropiada integración de principio a fin con el entorno del Data Warehouse.

3.5.9.8 Especificación de la Aplicación del Usuario Final

Es recomendable definir un conjunto de aplicaciones estándares para el usuario final ya que no todos los usuarios del negocio necesitan un acceso directo al Data Warehouse. Las especificaciones de la aplicación describen el formato del reporte, los parámetros del usuario y los cálculos requeridos. Estas especificaciones aseguran que el equipo de desarrollo y los usuarios del negocio tienen un entendimiento común de las aplicaciones a ser entregadas.

3.5.9.9 Desarrollo de la Aplicación del Usuario Final

Siguiendo las especificaciones de la aplicación, el desarrollo de las aplicaciones del usuario final involucra la configuración de la metadata de la herramienta, que podemos entenderlo como la parametrización de la herramienta a utilizar, y la construcción de los reportes especificados. Óptimamente, estas aplicaciones son construidas usando herramientas avanzadas de acceso a los datos que proporcionan ganancias significativas en productividad para el equipo de desarrollo de la aplicación. Además, estas herramientas ofrecen un mecanismo poderoso para los usuarios del negocio que pueden fácilmente modificar las plantillas de

reportes y personalizarlas de acuerdo a sus necesidades inmediatas.

3.5.9.10 Despliegue

El despliegue representa la convergencia de tecnología, datos y aplicaciones de usuario final accesibles desde el escritorio del usuario del negocio. Un planeamiento extensivo es necesario para asegurar que estas piezas del rompecabezas encajen adecuadamente todas juntas. El entrenamiento al usuario de negocio integrando todos los aspectos y elementos de la solución debe ser desarrollada cuidadosamente y entregado antes la puesta a producción definitiva. Además, las estrategias de soporte al usuario y de comunicación o realimentación deben establecerse antes que cualquier usuario de negocio tenga acceso al Data Warehouse o a un Data Mart. Es crítico que el despliegue de la solución final se haga lo más orquestado posible. El despliegue debería aplazarse si todas las piezas no están lista para salir al usuario final.

3.5.9.11 Mantenimiento y Crecimiento

Una gran cantidad de trabajo es seguido al despliegue inicial de un Data Warehouse. Se necesita un foco continuo en los usuarios de negocio, proveyéndolos con un soporte constante y entrenamiento. Es necesario además focalizar la atención en la infraestructura base del proyecto, asegurando que los procesos y procedimiento están en su lugar para una operación en curso efectiva del Data Warehouse. Las métricas de aceptación y rendimiento del Data Warehouse deberían ser medidas en el tiempo y registradas para soportar el marketing interno del Data Warehouse entre las diferentes áreas del negocio, obviamente buscando que estas métricas sean las adecuadas para garantizar una aceptación entre los usuarios finales. De existir problemas, será necesario una revisión de la infraestructura, relacionada con el hardware y software que sirve de base al proyecto, que pueden concluir en tareas de afinamiento y optimización.

El otro punto importante a considerar en esta etapa es que un Data Warehouse esta sujeto a desarrollarse y crecer. A diferencia de las iniciativas tradicionales de desarrollo de sistemas, el cambio debe ser visto como un signo de éxito y no como una falla o corrección. Se debe establecer una prioridad entre los procesos para poder enfrentar las demandas de los usuarios del negocio que van a conllevar a evolución y crecimiento. Después que las prioridades del proyecto son identificadas, se retornará al inicio del ciclo de vida, apoyándose y construyendo sobre lo que ya ha sido establecido en el entorno del Data Warehouse, con un foco en los nuevos requerimientos.

3.5.9.12 Administración del Proyecto

La administración del proyecto asegura que las actividades del ciclo de vida se mantienen con el seguimiento necesario y están en sintonía en el tiempo. Las actividades de administración del proyecto ocurren a través de todo el ciclo de vida. Estas actividades se centran en el monitoreo del status del proyecto, seguimiento y resolución de los problemas encontrados y control del cambio, para preservar los límites del alcance definido al inicio. Finalmente, la administración del proyecto incluye el desarrollo de un plan de comunicación integral y de gran amplitud que pueda abarcar las organizaciones de sistemas de información y del negocio en sí. Una comunicación constante es crítica para el manejo de las expectativas, y el manejo de las expectativas es ciertamente crítico para alcanzar las metas de un Data Warehouse.

4. DESARROLLO DEL CASO DE ESTUDIO

La solución integral al problema planteado requiere, tal como fuera explicado anteriormente en el fundamento teórico, considerar las etapas del ciclo planteado por Peppers & Rogers. Tomando en cuenta que las etapas de interacción y personalización del ciclo CRM requieren de la implantación de las estrategias propuestas y la recolección de los datos que explican el efecto de las mismas para completar su análisis, en el caso de estudio nos enfocaremos en la etapa de diferenciación, soportado en un modelo de datos multidimensional desarrollado siguiendo los lineamientos de la metodología planteada por Ralph Kimball en su “Business Dimensional LifeCycle”, también descrito en el fundamento teórico. Del total de tareas que plantea la metodología de Ralph Kimball el presente estudio se concentrará en las tareas resaltadas en la Figura 15.

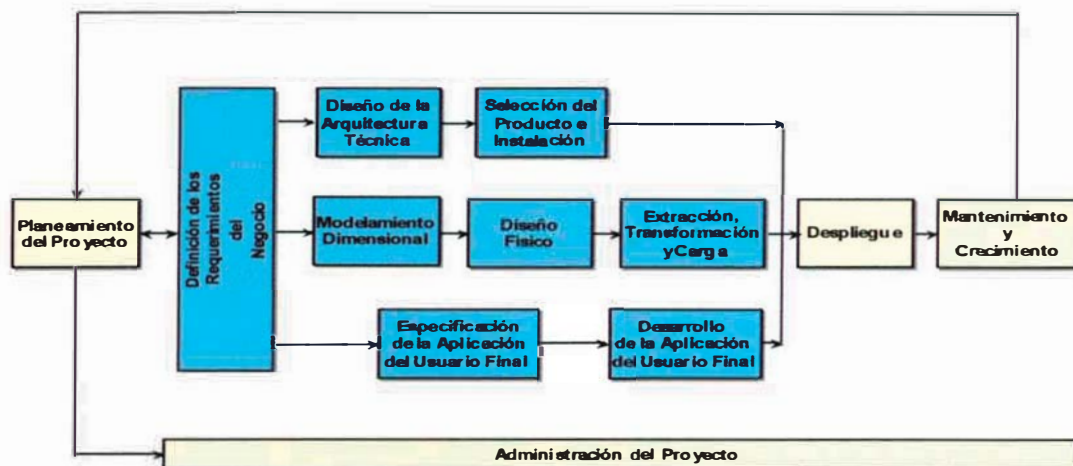


Figura 15. Etapas de un Proyecto de Inteligencia de Negocios desarrolladas en el Caso de Estudio

4.1. DEFINICIÓN DE LOS REQUERIMIENTOS DEL NEGOCIO

Las instituciones financieras, al igual que cualquier organización dedicada a la venta de servicios o productos, requieren conocer cuán rentable puede ser la relación que mantienen con sus clientes para explotarla al máximo en el caso que exista un margen de rentabilidad que aún no ha sido explotado, para desligarse de aquellos clientes que no representen beneficios económicos por su baja rentabilidad o para conocer a sus mejores clientes y asegurar que la relación existente se fortalezca en el tiempo a través de la oferta de nuevos productos y servicios que lo vinculen aún más con el Banco.

Sin embargo el conocimiento de esta rentabilidad no es suficiente para poder administrar la relación de los clientes. La rentabilidad actual es un indicador financiero que muchas instituciones conocen y calculan. CRM plantea nuevos conceptos que permiten enriquecer el análisis de esta información. Siguiendo el ejemplo con el que Peppers & Roggers ilustran su visión de CRM y que se presenta en el fundamento teórico, además de la rentabilidad actual se requiere proponer una rentabilidad potencial así como una vida esperada para que, apoyados en una segmentación adecuada, se puedan identificar grupos homogéneos de clientes para luego diseñar las estrategias comerciales que permitan incrementar la rentabilidad general de la organización.

Este estudio analiza información relacionada a la banca personal de un banco privado del medio y busca responder las preguntas clásicas del análisis CRM orientadas a un mercado financiero como el nuestro:

- ¿cuál es el segmento de clientes más rentables?
- ¿cuál es la variable que mejor explica la rentabilidad de los clientes?
- ¿cuáles son los clientes que tienen una rentabilidad potencial que nos interesa explotar?
- ¿cuál es el ratio de deserción de los clientes?
- ¿qué características tienen los clientes que perdemos?
- ¿qué porcentaje de los clientes que se van nos interesa retener basado en su rentabilidad?

Para la solución a las interrogantes planteadas se propone el diseño de un Data Mart que emplee un modelo de datos multidimensional implantado sobre una base de datos relacional que permita a los usuarios de negocios explotar información a través de herramientas OLAP con una interfaz web.

4.2. MODELAMIENTO DIMENSIONAL

El modelo de datos del Data Warehouse del banco seleccionado fue concebido para atender los requerimientos de distintas áreas de negocio y no sólo para dar soporte a un CRM Analítico, por lo que no está orientado necesariamente a responder preguntas como las planteadas en el punto previo. Por esta razón para el caso de estudio se ha diseñado un nuevo modelo que facilita el análisis de la información relacionada con la fase de Diferenciación del ciclo CRM. En los Anexo A y B se presentan todas las dimensiones y tablas de hechos que participan del modelo dimensional completo y el diagrama entidad-relación del modelo, respectivamente.

Para los propósitos del estudio nos focalizaremos en los siguientes elementos:

- Dimensiones:
 - Persona
 - Producto
 - Transacción
 - Tiempo
 - Cliente
- Indicadores:
 - Indicadores de Cliente
 - Indicadores de Producto
 - Indicadores de Transacción

Para el diseño del modelo dimensional se ha empleado la herramienta CASE ERWIN y se ha utilizado la nomenclatura del siguiente cuadro:

Tablas	
D_	Tabla Dimensional
F_	Tabla de Hechos (Facts)
X	Tabla Auxiliar
Atributos/Medidas	
Cla	Clasificación
Cli	Cliente
Cod	Código
Con	Contrato
Ctd	Cantidad
Des	Descripción
Eco	Económica(o)
Edu	Educación
Est	Estado
Fec	Fecha
Fin	Financiero
Flg	Flag
Lab	Laboral
Mod	Modificación
Mto	Monto
Nbr	Nombre
Num	Número
Per	Persona
Rie	Riesgo
Seg	Seguridad
Ult	Ultima(o)
Ven	Vencimiento
Vin	Vinculación

Cuadro 6. Nomenclatura empleada en el modelo de datos para nombras entidades y atributos.

4.2.1 Definición de dimensiones

Siguiendo las técnicas de modelamiento multidimensional aplicados a modelos de negocios se deben definir variables de análisis que respondan las preguntas básicas quién, qué, cómo, cuando y dónde. En el caso de una institución financiera estas dimensiones de análisis se traducen en datos de los clientes, los productos y las transacciones, el tiempo y el lugar. A continuación se describen las principales dimensiones consideradas en el modelo.

4.2.1.1 Dimensión Persona

La dimensión persona agrupa los atributos demográficos, económicos y financieros de las personas (ya sean jurídicas o naturales) que han

tenido algún tipo de contacto con la entidad financiera, (no solamente a los clientes actuales), de esta manera es posible consultar información histórica relacionada a personas que ya no son clientes o emplear estos datos en campañas para incrementar el universo de clientes.

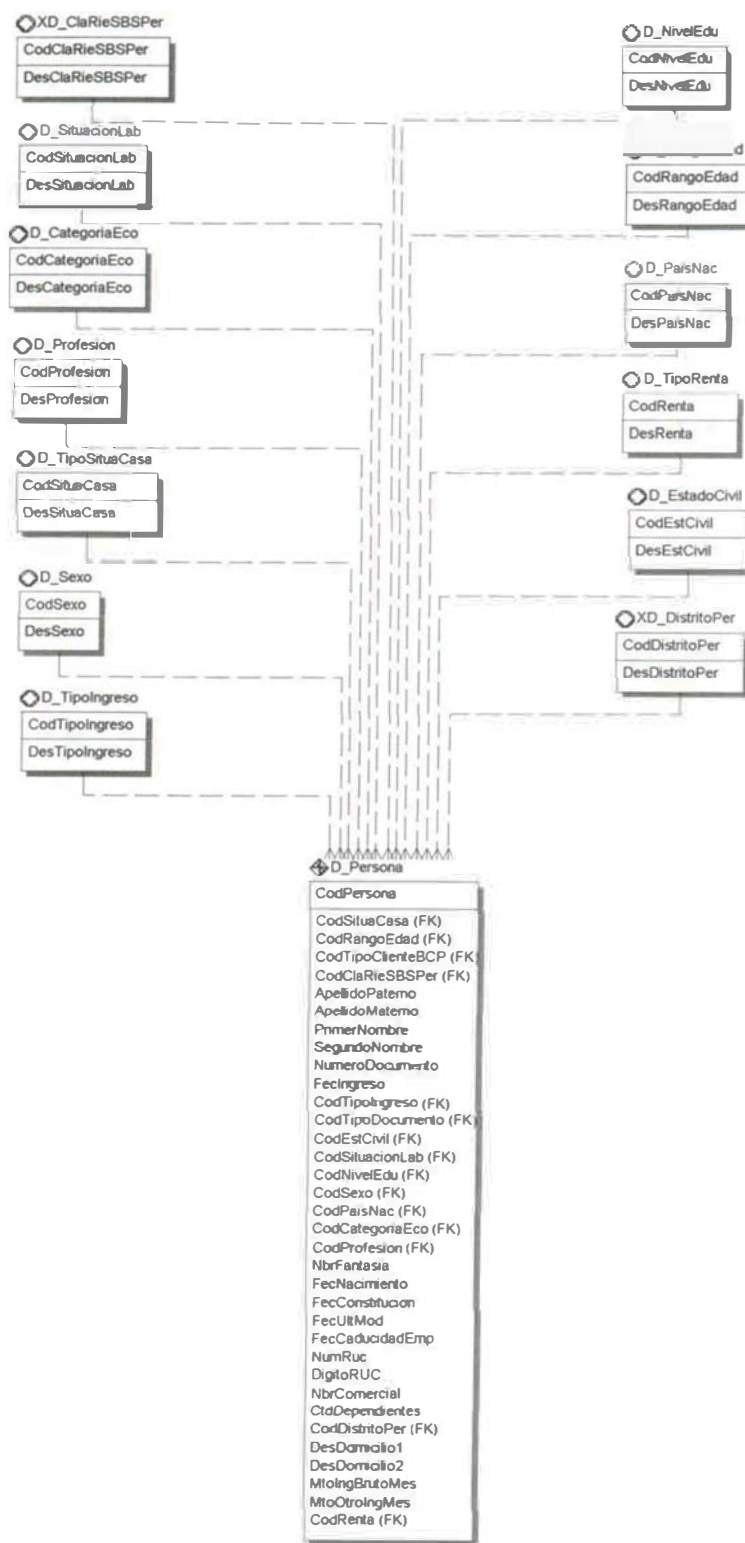


Figura 16. Dimensión Persona

4.2.1.2 Dimensión Producto

De manera tradicional la dimensión producto es independiente de la Dimensión Cliente o Persona. Sin embargo el problema con este tipo de diseño se presenta cuando se intenta reproducir la navegación descendente (drill-down) desde el cliente hacia sus productos. Para poder superar esta deficiencia se ha diseñado esta dimensión como un subconjunto de atributos de la Dimensión Persona de manera que puedan ser analizados de manera integral.

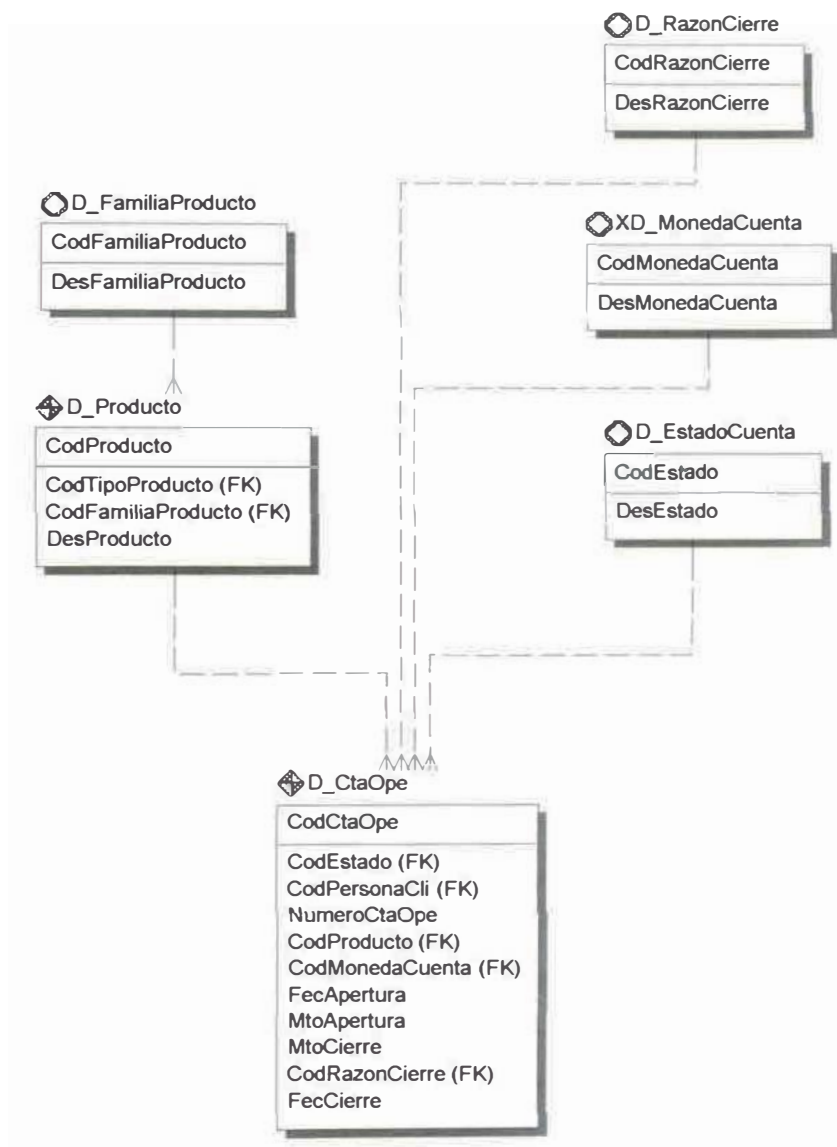


Figura 17. Dimensión Producto

4.2.1.3 Dimensión Cliente

Al igual que la dimensión Producto, el Cliente se define como un subconjunto de atributos de la dimensión Persona. En la entidad Cliente se encuentran información adicional para aquellas personas que tienen (o han tenido) algún vínculo directo con el banco.



Figura 18. Dimensión Cliente

4.2.1.4 Dimensión Transacción

Contiene un conjunto de atributos que permite analizar todas las transacciones, ya sean monetarias o no, que los clientes han realizado a través de los canales del Banco. En este modelo no se consideran como transacciones a los contactos establecidos por cliente en las plataformas del banco, tales como la presentación de reclamos, o aperturas de productos.

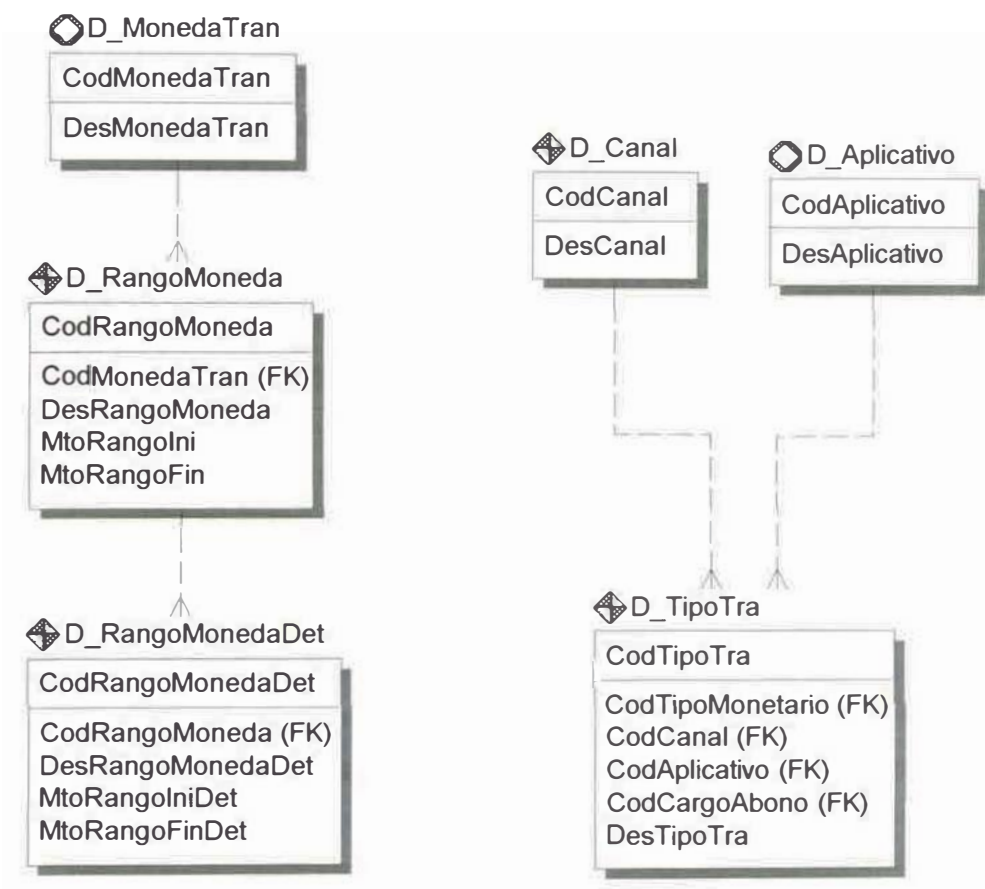


Figura 19. Dimensión Transacción

4.2.1.5 Dimensión Tiempo

Contiene atributos que permiten agrupar la información de las tablas de hechos en periodos mensuales o anuales.

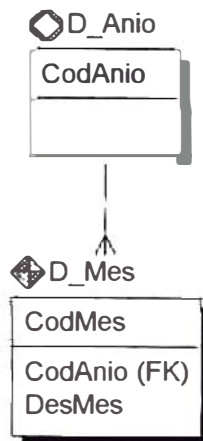


Figura 20. Dimensión Tiempo

4.2.2 Definición de indicadores

4.2.2.1 Indicadores de Clientes

A continuación se presentan las entidades que contienen los indicadores de clientes y su relación con las dimensiones del caso. Las entidades principales para este análisis son entidad de clientes (D_Clientes), la entidad de Meses (D_Mes) y la entidad que contiene la información de los indicadores (F_Cliente). Adicionalmente se han agregado algunas otras tablas para poder analizar de manera independiente los atributos actuales del cliente con sus valores históricos (sector, subsegmento, clasificación riesgo BCP, clasificación riesgo SBS).

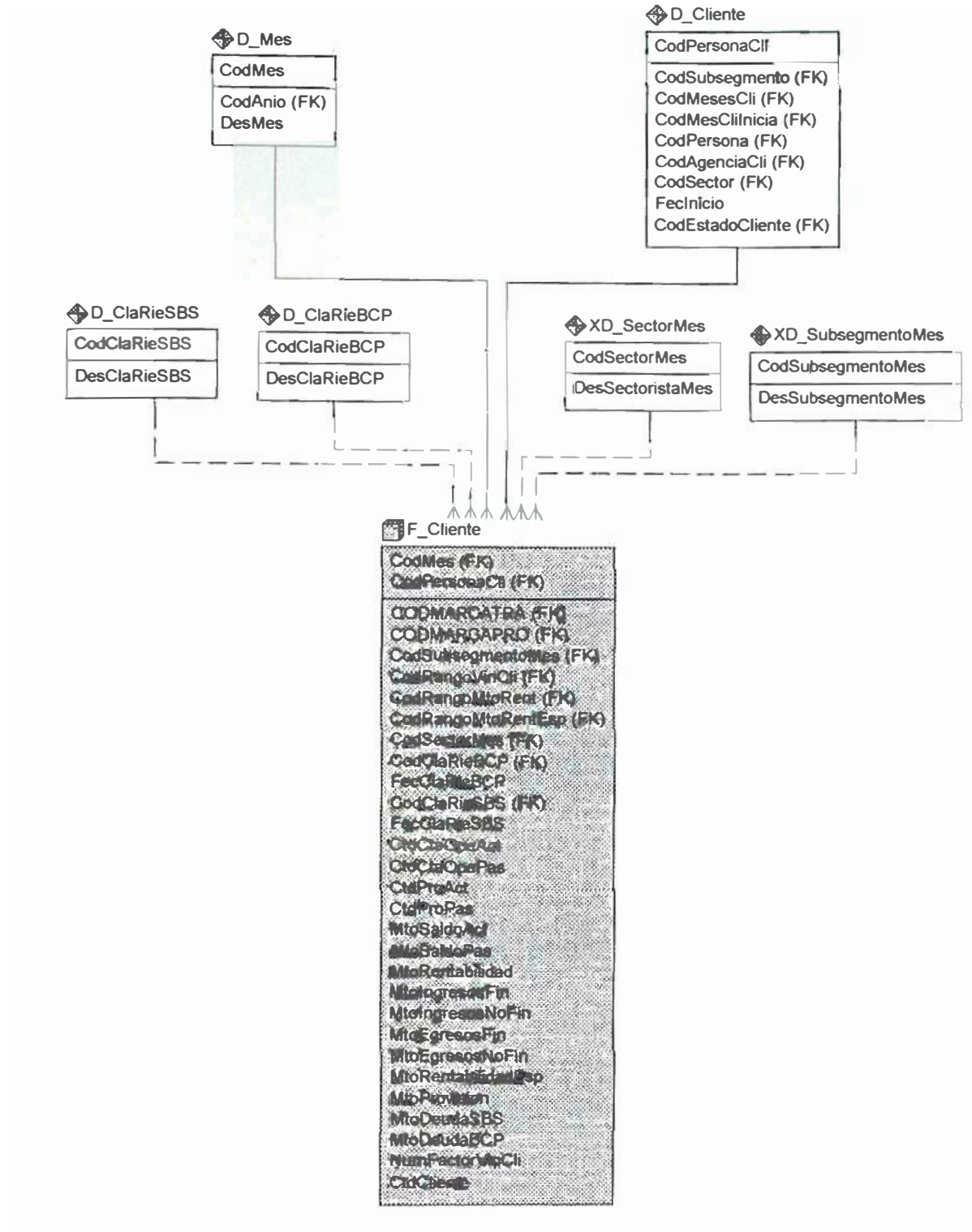


Figura 21. Indicadores de Cliente

Tradicionalmente las instituciones financieras emplean indicadores financieros o comerciales tales como:

- Rentabilidad del producto.- margen de utilidad producido por una cuenta u operación durante un periodo determinado.
- Rentabilidad del cliente.- margen de utilidad producido por los productos y servicios que un cliente posee durante un periodo determinado.
- Venta Cruzada.- cantidad de productos vendidos a un cliente.
- Saldo del cliente.- Suma de los saldos de los productos de un cliente.

Adicionalmente a estos indicadores el análisis necesario para definir una estrategia CRM requiere de otro tipo de indicadores que permitan entender mejor las relaciones existentes y potenciales con los clientes. Se han definido y calculado los siguientes indicadores:

- Rentabilidad potencial del cliente.- Margen de utilidad máximo que un cliente podría generar en el banco si se fortaleciera sus relaciones con la institución. Este indicador también es un cálculo teórico referencial y no puede ser tomado como dato exacto. El cálculo de la rentabilidad potencial se explicará en detalle más adelante.
- Tiempo de vida del cliente.- Definido como la cantidad de meses pasados desde que el cliente inició su relación con la empresa. Se define como el inicio de la relación la fecha en que fue adquirido el producto de mayor antigüedad del cliente.

4.2.2.2 Indicadores de Productos

Las entidades principales para el análisis de los indicadores de productos son la entidad de Cuentas y Operaciones (D_CtaOpe), la entidad de Mes (D_Mes) y la entidad con la información de los indicadores (F_CuentaOperación). Los indicadores de cuentas son similares a los de clientes con la diferencia que se encuentran a un mayor nivel de detalle. En la mayoría de casos los indicadores de

clientes reciben el mismo nombre pero hacen referencia a la suma de los indicadores de los productos que posee un cliente.

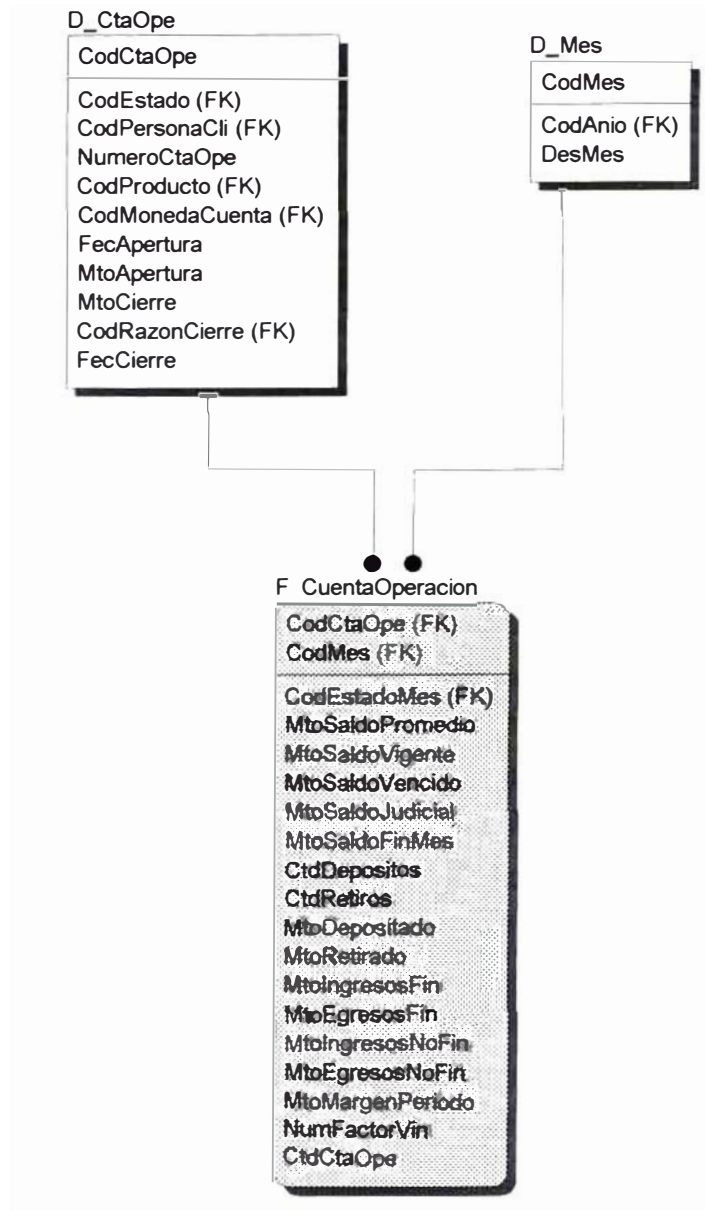


Figura 22. Indicadores de Producto

4.2.2.3 Indicadores de Transacción

Las principales entidades para el análisis de estos indicadores son: el Tipo de Transacción (D_TipoTra), que almacena la clasificación de los posibles tipos de transacciones a realizarse en los canales (cajeros automáticos, banca telefónica, banca por Internet, Ventanilla, POS, Saldomáticos, procesos batch), el Rango de Moneda (D_RangoMonedaDet) que fue creada para poder manejar el gran volumen de información que contiene la tabla de indicadores de transacciones y agrupa el monto de las transacciones de acuerdo a rangos predefinidos, la entidad de Clientes (D_Cliente) y la entidad de Meses (D_Mes).

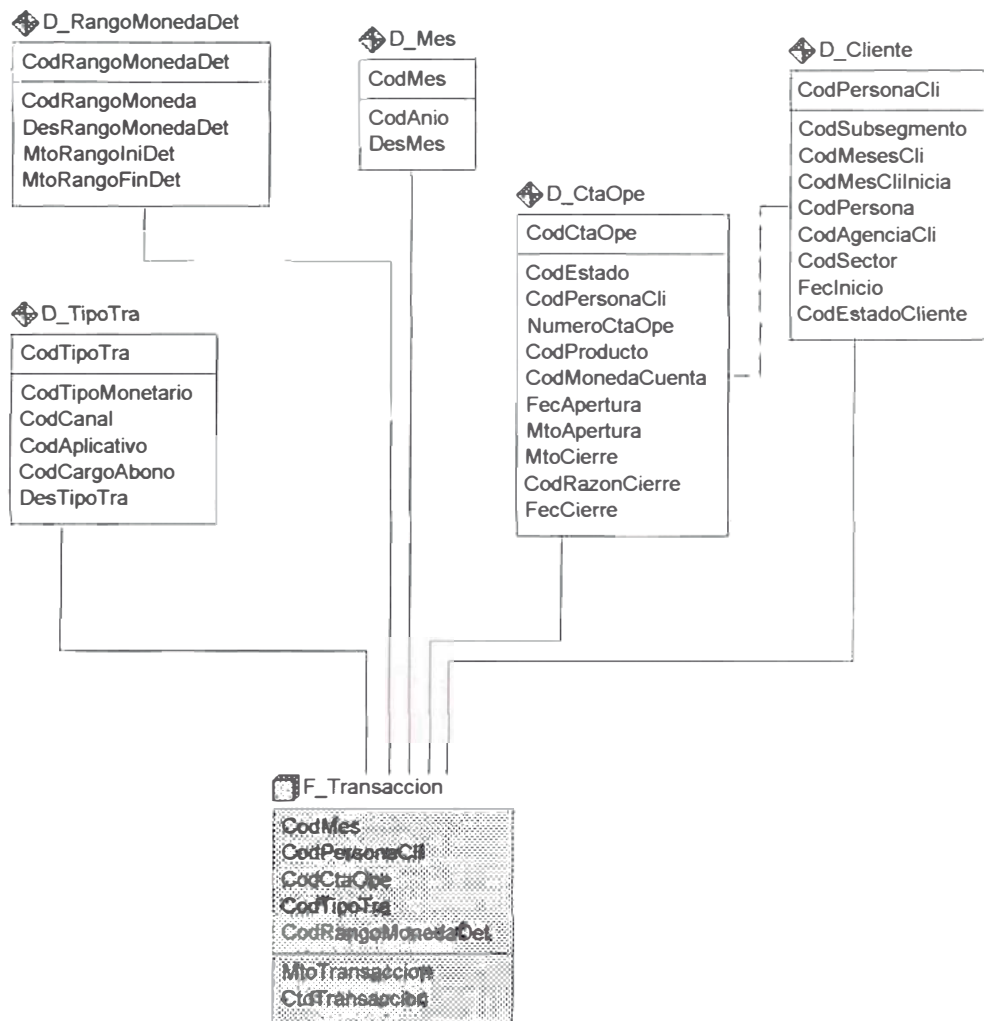


Figura 23. Indicadores de Transacción

4.3. DISEÑO FÍSICO

El banco escogido para el desarrollo del caso cuenta aproximadamente con 1'000,000 de clientes en la Banca Personal, que congrega a todas aquellas personas naturales con algún producto o servicio vigente. Para propósitos de este análisis se ha escogido 13 meses de información de una muestra aleatoria que representa el 1% del universo de clientes. Este periodo de 13 meses va desde Enero del 2001 a Enero del 2002. Para evitar distorsiones en la muestra y debido a su baja incidencia, se ha decidido eliminar todos aquellos productos típicos de la Banca Empresa y Corporativa tales como las letras, garantías, pagarés, etc. Los datos para el análisis han sido extraídos del Data Warehouse corporativo del banco seleccionado.

A partir del modelo de datos lógico multidimensional se generan las tablas de la base de datos relacional y se determinan los tipos de datos a usar. En el Anexo C se adjunta el script de creación de la base de datos generado con ayuda de la herramienta ERWIN a partir del modelo lógico y en el Anexo A se detallan los atributos de las tablas empleadas. El modelo contempla algunas entidades que no han sido cargadas pues no son consideradas en el análisis realizado, pero que se plantean en el modelo con el objetivo de complementarlo y presentar una visión más completa del diseño de un Data Mart que sirva de soporte a un CRM Analítico.

4.4. EXTRACCIÓN, TRANSFORMACIÓN Y CARGA

Para el desarrollo de este estudio se empleó como fuente de información un Banco Privado, cuyo mercado objetivo esta formado en un 90% por personas naturales. El banco seleccionado para el caso cuenta además con un conjunto de aplicativos que permiten capturar, almacenar, consultar y modificar la información relacionada a sus productos, servicios y clientes. A continuación se mencionan los aplicativos que administran la información empleada en el desarrollo del modelo de información propuesto:

Aplicativo	Descripción
Systematics	Conjunto de aplicativos destinados a la administración de los productos principales del banco tales como las cuentas corrientes, las cuentas de ahorros y los préstamos personales.
Vision Plus	Aplicativo administrador de las tarjetas de crédito VISA y AMEX.
Consynt	Aplicativo para la administración de la base de datos de clientes.
SAP	Aplicativo interno del Banco, para la administración de la solicitud y apertura de productos.
SBS	Aplicativo que recibe la información de la SBS correspondiente a la deuda de los clientes en el resto del sistema financiero.
Rentabilidad	Aplicativo encargado del calculo mensual de la rentabilidad por cliente.

Cuadro 7. Aplicativos fuentes de datos

El Banco seleccionado cuenta con un Data Warehouse como repositorio central de datos. La información que reside en este repositorio ya ha pasado por procesos de extracción, transformación, carga y limpieza de datos, por lo que podemos asumir que cuenta con un nivel de calidad aceptable.

Para el desarrollo del caso se seleccionó aleatoriamente una muestra del 1% de Clientes pertenecientes a la Banca Personal. En el caso de las tablas descriptivas y de búsqueda, tales como TipoCliente, TipoDocumento, Segmento, Canal, ActividadEconómica, Distrito, Provincia, etc., se extrajeron todas las filas de las tablas equivalentes en el modelo de datos del Data Warehouse.

La extracción de información se llevó a cabo usando sentencias SQL estándares y empleando condiciones que filtraron la información de manera uniforme para obtener una muestra lo más representativa posible. Los datos extraídos fueron almacenados en archivos de texto, usando comas como separadores de campos (archivos *.csv) y cargados en tablas intermedias con ayuda de los utilitarios de carga provistos por la base de datos destino (para el

caso de estudio se ha empleado la base de datos DB2 UDB versión 7.2). Un ejemplo de una sentencia para importar datos en DB2 a partir de un archivo plano se presenta a continuación:

```
IMPORT FROM "C:\Tesis\BI-CRM\POC\Data\lcl_codpais.csv"  
OF DEL MODIFIED BY coldel, decpt.  
COMMITCOUNT 2000  
MESSAGES "c:\tmp\messages.out"  
INSERT INTO DB2ADMIN.mm_pais;
```

Luego los datos de estas tablas intermedias se cargaron en las tablas del modelo dimensional a través de construcciones de sentencias INSERT Y SELECT.

4.5. CALIDAD DE DATOS

Durante el proceso de extracción y carga del modelo de datos propuesto y a pesar de que la información seleccionada provino de un repositorio de datos que cuenta con algunos procesos de limpieza de datos, se encontraron algunos problemas relacionados con la calidad de información. Estos problemas fueron detectados gracias a la integridad referencial empleada en el modelo, la cual permitió identificar la inexistencia de algunos registros necesarios de acuerdo a la declaración de claves foráneas y primarias.

La calidad de datos es uno de los aspectos de mayor importancia en el desarrollo de un modelo de datos. De este factor depende la validez de las conclusiones resultantes de cualquier análisis efectuado sobre el modelo. Un aspecto particularmente importante en el desarrollo de modelos multidimensionales, es la estrategia empleada para manejar las relaciones dadas entre las tablas durante las consultas a la base de datos (sentencias SQL). Tradicionalmente, los manejadores de bases de datos relacionales ofrecen ciertas opciones estándares para el tratamiento de estas relaciones:

- Inner join, es una relación conmutativa que considera solamente a aquellos registros que cumplen la igualdad definida

- Left outer join, es una relación no conmutativa que considera a todos los registros de la entidad a la izquierda de la relación y a aquellos registros de la entidad a la derecha que cumplan con la relación de igualdad definida.
- Full Outer Join, es una relación conmutativa que considera, además de los registros seleccionados por un Inner join, a aquellos registros de ambas entidades participantes en la relación, que no encontraron un par equivalente en la relación de igualdad definida.

Dependiendo la opción a emplear durante la navegación en el modelo los resultados a obtenerse podrían variar considerablemente. Así por ejemplo los resultados de un simple conteo de clientes agrupado por un atributo poblado parcialmente arrojaría resultados muy distintos efectuados si se emplea un Inner Join o un Left Outer Join.

Este comportamiento de los datos dificulta el entendimiento de los resultados y, si no es contemplado adecuadamente, puede volverse un problema serio en la credibilidad del modelo. Para evitar este problema, en el caso del modelo propuesto y durante el proceso de carga desarrollado se optó por completar todas las relaciones de datos creando registros que identifican a los datos no existentes en todas las relaciones entre entidades definidas en el modelo.

4.6. DISEÑO DE LA ARQUITECTURA TÉCNICA

Para la implantación de la prueba de concepto del caso de estudio se hará uso de un servidor de base de datos relacional donde resida el modelo multidimensional. Este modelo constituirá un Data Mart que soportará el CRM Analítico.

El Data Mart será explotado a través de una herramienta ROLAP, capaz de crear aplicaciones de inteligencia de negocios que le permitan a los usuarios analizar la información del modelo multidimensional. Esta herramienta deberá ayudar en la creación de proyectos, reportes, gráficos y en la administración

de la metadata necesaria para soportar estos proyectos. La metadata guardará información sobre las dimensiones, indicadores, jerarquías y otras definiciones que serán almacenadas también en una base de datos relacional.

La herramienta OLAP deberá ser capaz de brindar acceso a los usuarios por una interfaz web o una interfaz Windows, y soportará un esquema de tres capas, con un servidor OLAP en la capa intermedia que acceda a la base de datos y se integre con un servidor web para brindar acceso a los usuarios a través de un explorador de Internet.

Adicionalmente, para identificar qué variables explican en mayor grado la rentabilidad de los clientes se hace uso de una herramienta de minería de datos. Esta herramienta accede a la base de datos relacional para leer información de los clientes.

La arquitectura propuesta para cumplir con los objetivos del caso de estudio sería como la presentada en la Figura 24.

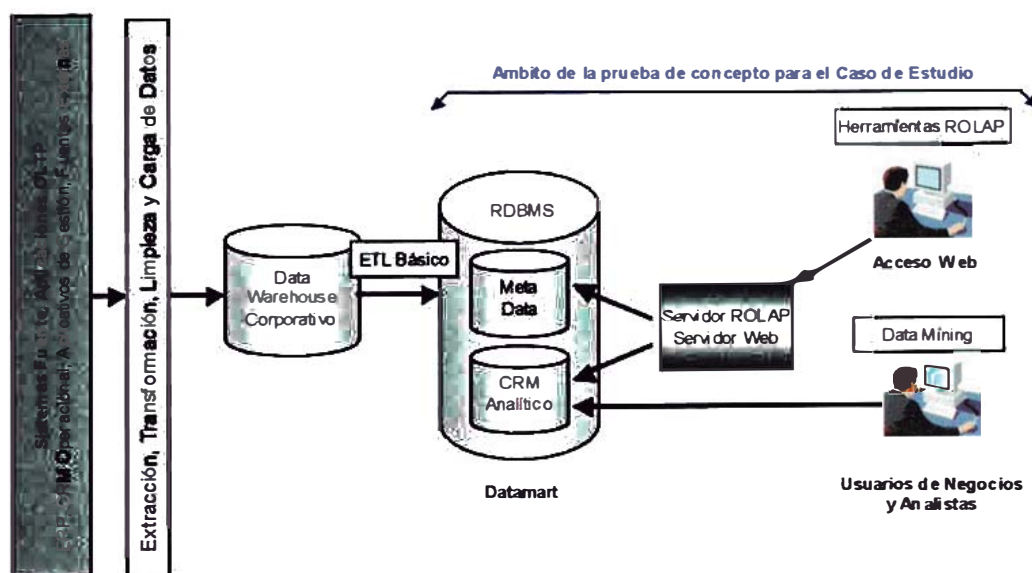


Figura 24. Arquitectura planteada para el Caso de Estudio

4.7. SELECCIÓN DEL PRODUCTO E INSTALACIÓN

Teniendo en cuenta los componentes de inteligencia de negocios definidos en la sección anterior y los requerimientos tecnológicos, la prueba de concepto del caso de estudio se ha implantado con productos reconocidos del mercado y especializados en diferentes tareas:

- Servidor de Base de Datos Relacional y herramientas SQL: DB2 UDB v. 7.2
- Herramienta ROLAP para cliente Windows: MicroStrategy Desktop 7i v. 7.2.0
- Herramienta ROLAP para cliente web: MicroStrategy Web 7i v. 7.2.0
- Servidor ROLAP: MicroStrategy Intelligent Server 7i v. 7.2.0
- Servidor Web: Microsoft Internet Information Service 5.0
- Herramienta Case: ERWIN 3.5.2
- Herramienta de Minería de Datos: KnowledgeSEEKER IV v. 4.3

En el Anexo D se presenta el posicionamiento en el mercado, las fortalezas y debilidades de los proveedores de servidores de base de datos relacionales y herramientas OLAP más reconocidos, entre los que se incluyen la base de datos DB2 UDB y la suite de productos OLAP de MicroStrategy empleadas en el desarrollo del caso de estudio.

Todos estos productos se instalaron sobre un servidor Windows 2000, con 1 GB de memoria y un procesador de 1.8 Ghz. La arquitectura final, que incluye los productos empleados en el caso de estudio, se presenta en la Figura 25.

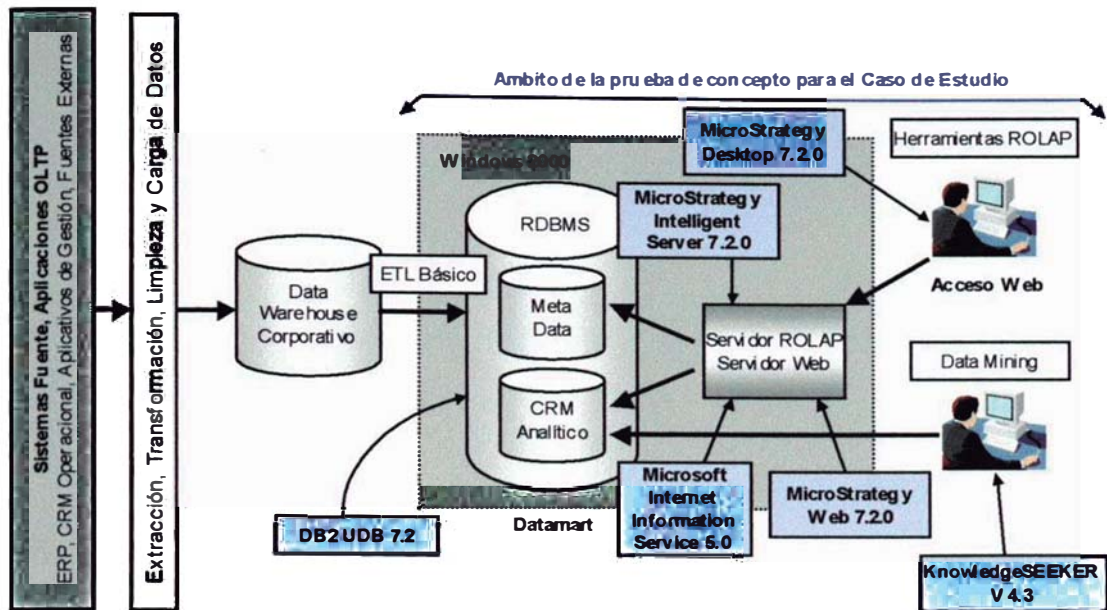


Figura 25. Productos a emplearse en la prueba de concepto del caso de estudio

4.8. ESPECIFICACIÓN Y DESARROLLO DE LA APLICACIÓN DEL USUARIO FINAL

El objetivo de la fase de diferenciación del ciclo CRM consiste en conocer al cliente a través del análisis y el entendimiento de su relación con el Banco. Sobre la base de la teoría revisada no se puede concluir en la existencia de un consenso acerca de los pasos que podrían guiar este proceso de análisis. En consecuencia, cada organización debe enfocar esta etapa de manera particular tomando en cuenta su entorno, el mercado en el que se desenvuelve, y la tecnología con la que cuenta.

De acuerdo a Peppers & Rogers, una herramienta de gran utilidad y una de las más empleadas consiste en agrupar a los clientes en grupos con características y necesidades homogéneas llamados segmentos. El banco seleccionado cuenta ya con una segmentación implementada desde hace años. Sin embargo, esta segmentación se encuentra vinculada solamente a

características socio-económicas y no a otro tipo de variables tales como el ciclo de vida, el valor presente, la brecha de rentabilidad, el nivel de lealtad, etc., variables que permitirían afinar mucho más esta agrupación de clientes y acercarla al ideal propuesto por CRM, cual es conocer al cliente íntimamente. Lamentablemente, muchos de estos parámetros no son posibles de conocer o estimar pues requieren de mecanismos de captura de información de alto costo.

Para el desarrollo de este caso se ha escogido a la rentabilidad como el indicador inicial para el análisis de la relación Banco-Cliente. Se ha seleccionado este indicador porque reúne algunas características muy convenientes para el estudio. En primer lugar se trata de un dato de relativa madurez en el Banco (se emplea desde 1997), se calcula para todos los clientes del Banco y, además porque es un indicador económico que mide cuán rentable han resultado los productos y servicios empleados por un cliente durante un periodo específico. En su cálculo se incluyen Ingresos y Costos (Financieros y no Financieros), y esta influenciado directamente por las características de cada producto, el comportamiento de pago del cliente, el nivel de utilización de los servicios y el nivel de riesgo del cliente, entre otros datos.

Dado que se cuenta con un año de información se analizará primero la distribución de la rentabilidad anual promedio de los clientes para identificar cual es el grupo más rentable y cuáles son sus características. Empleando la aplicación OLAP desarrollada en Microstrategy y detallada en el anexo E, se obtiene el siguiente reporte para analizar la distribución de la rentabilidad del banco. Todos los indicadores relativos a rentabilidad se expresan en nuevos soles (S/).

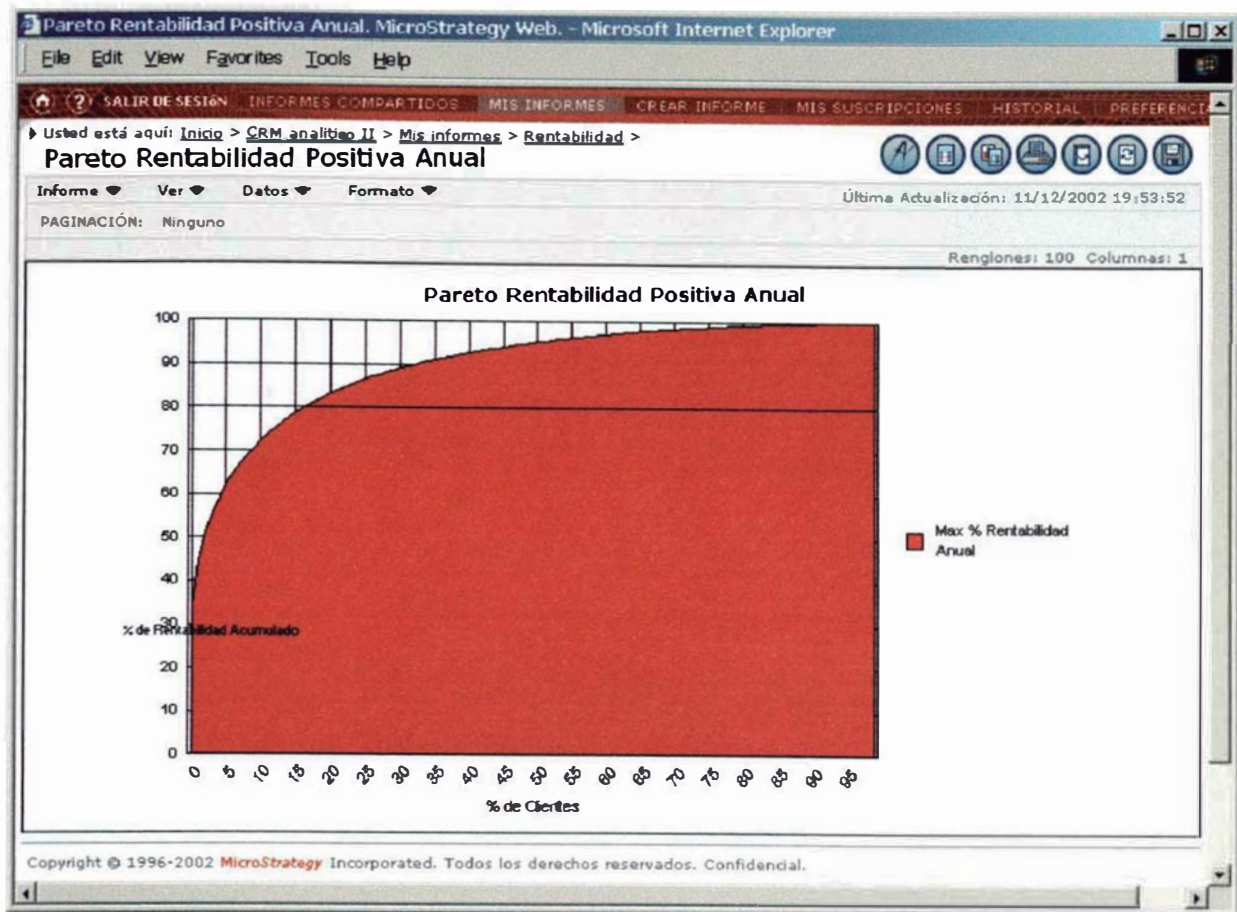


Figura 26. Pareto Rentabilidad Positiva Anual

Dado que la rentabilidad es un valor que puede ser negativo, todos los datos menores a cero han sido excluidos de este primer análisis.

Del gráfico se desprende que de aquellos clientes que aportan con rentabilidades positivas al Banco, el 17% de los clientes representan el 80% de esta la rentabilidad anual positiva. Sobre la base de esta información se definen luego los límites apropiados para agrupar a los clientes en los segmentos sugeridos por la teoría de CRM para la identificación de las brechas potenciales y el diseño de las estrategias comerciales adecuadas. Incrementando el nivel de detalle del reporte, se encuentra que este 17% de clientes registran una rentabilidad promedio anual de aproximadamente 50 soles mensuales. Este valor se tomará como punto de quiebre para la creación de los rangos de rentabilidad a utilizar más adelante.

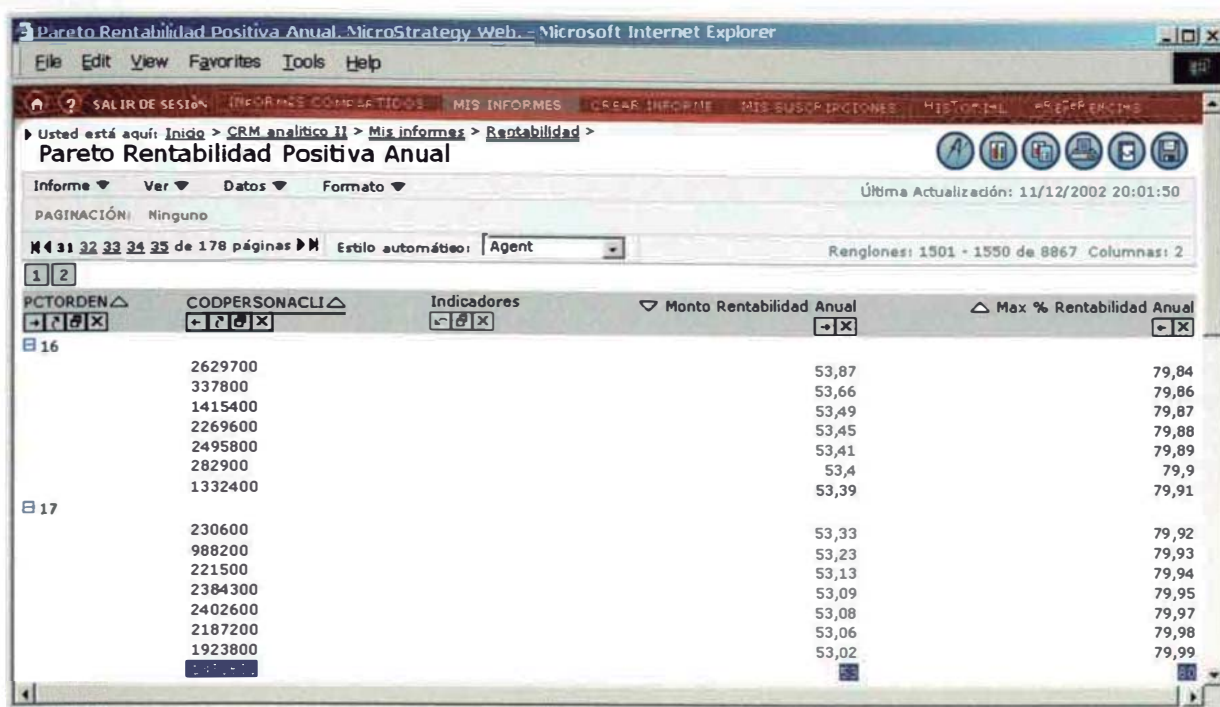


Figura 27. Detalle Pareto Rentabilidad Positiva Anual

Empleando una lógica similar se analiza a los clientes con un promedio de rentabilidad anual negativa.

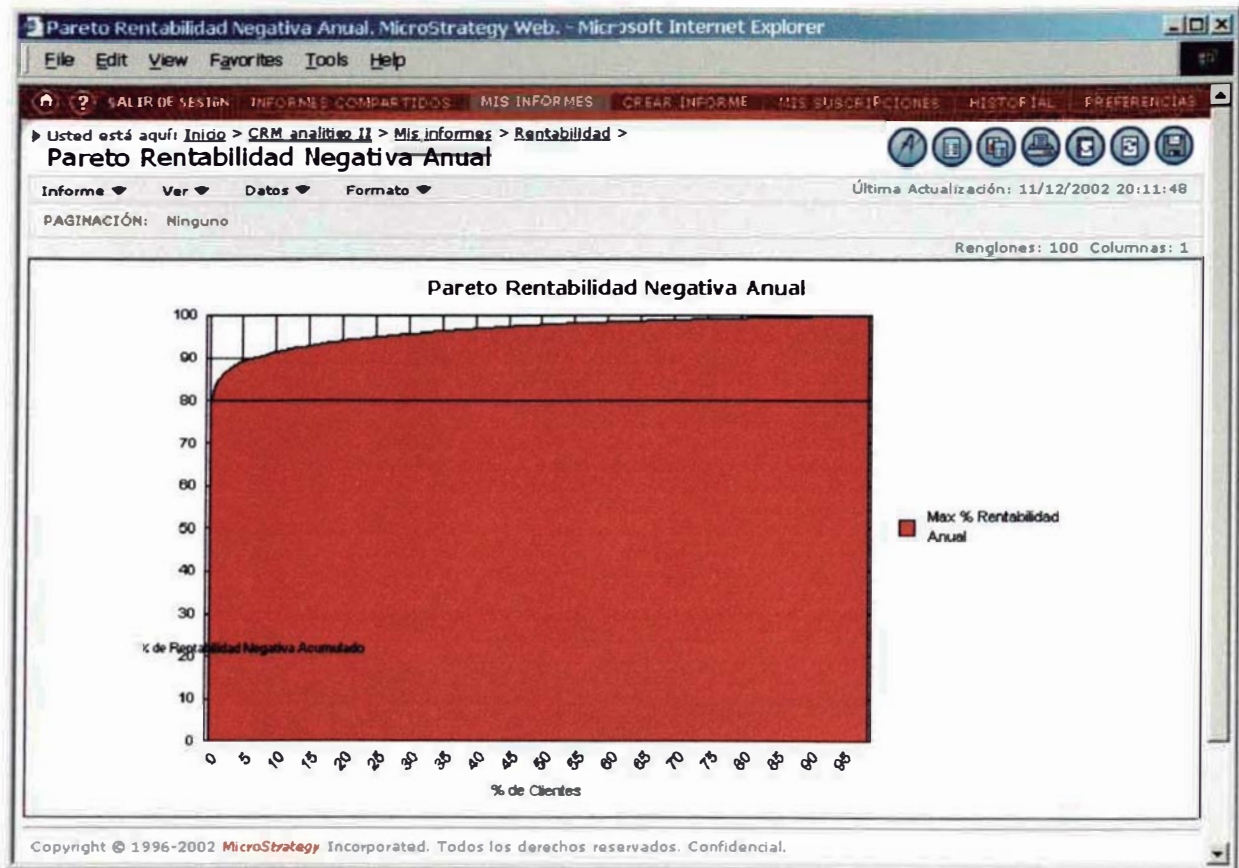


Figura 28. Pareto Rentabilidad Negativa Anual

Para estos grupos de clientes se observa claramente que la curva es mucho más acentuada que en el caso anterior y se concluye que, del grupo de clientes que registran una rentabilidad anual promedio negativa, el 1% representa el 80% de las pérdidas en rentabilidad del Banco. El punto de quiebre asociado a este porcentaje es -242 soles. Para el caso de estudio aproximó este valor a -250.

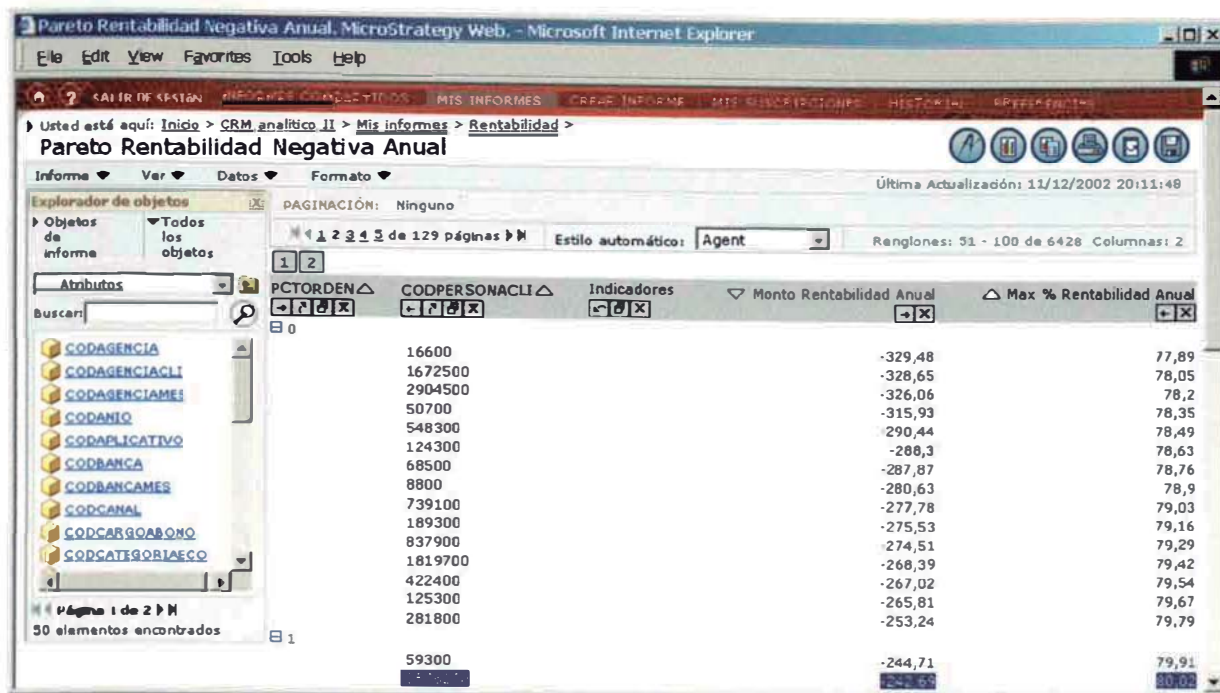


Figura 29. Detalle Pareto Rentabilidad Negativa Anual

Empleando estos valores se definieron cuatro rangos de valores que luego fueron utilizados para analizar la distribución de la rentabilidad a mayor detalle. Los rangos definidos fueron:

Rango de Rentabilidad	Limites	Descripción
MUY BAJA	< ∞, -250 >	Clientes que aportan el 80% de las rentabilidades negativas del Banco.
BAJA	[-250, 0 >	Clientes que aportan el 20% de las rentabilidades negativas del Banco.
MEDIA	[0, 50 >	Clientes que aportan el 20% de las rentabilidades negativas del Banco.
ALTA	[50, ∞ >	Clientes que aportan el 80% de las rentabilidades negativas del Banco.

Cuadro 8. Rangos de Rentabilidad Definidos para el análisis del Caso de Estudio

Estos rangos de valores fueron utilizados luego en el análisis de la distribución de la rentabilidad y en la identificación del posible valor esperado de rentabilidad por cliente.

Para analizar esta información se prepararon dos gráficos para los cuatro segmentos de mayor importancia: Consumo, Exclusiva, Negocio e Instituciones. El primer gráfico de cada segmento muestra el monto de la rentabilidad distribuido en los rangos preparados y el segundo muestra la cantidad de clientes por cada rango de rentabilidad.

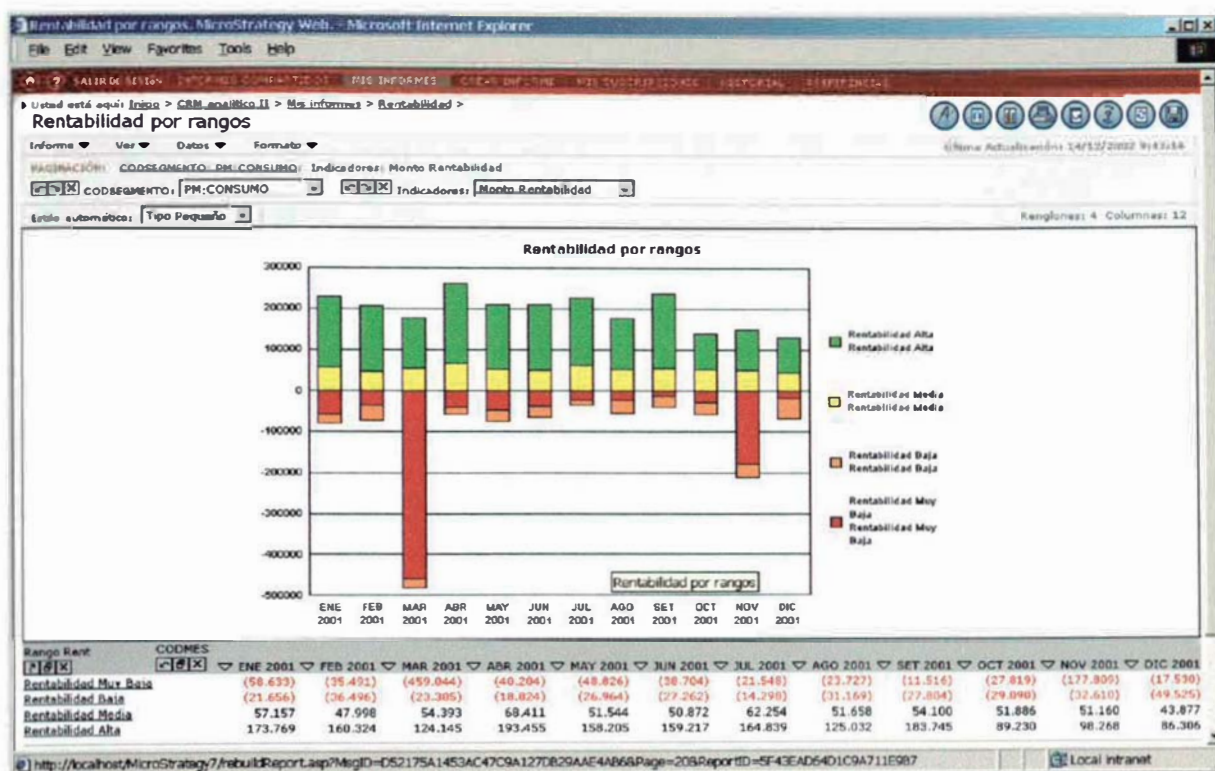


Figura 30. Segmento Consumo – Distribución de Rentabilidad Mensual

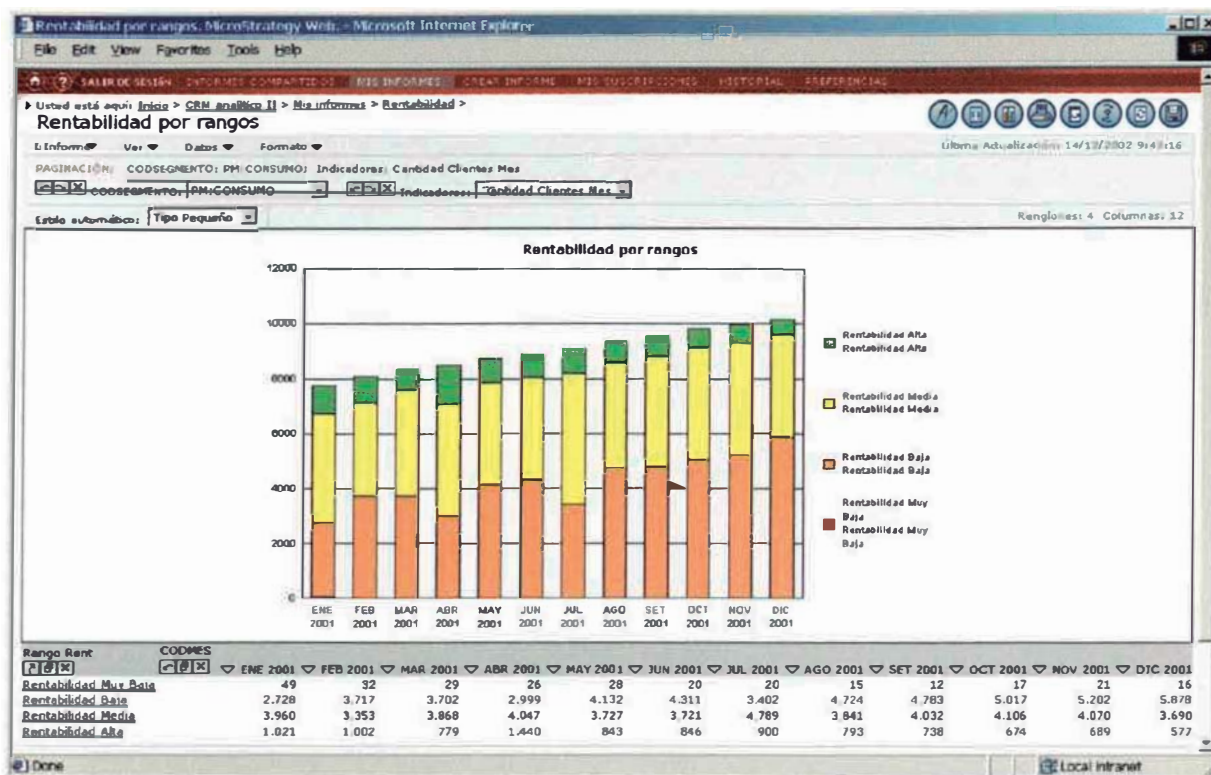


Figura 31. Segmento Consumo – Distribución de la Cantidad de Clientes

En el caso del segmento de consumo se identifica que si bien la cantidad de clientes ha ido incrementándose durante el año, la rentabilidad mensual ha tenido un comportamiento estable ligeramente decreciente a lo largo del periodo (ver figura 30 y 31).

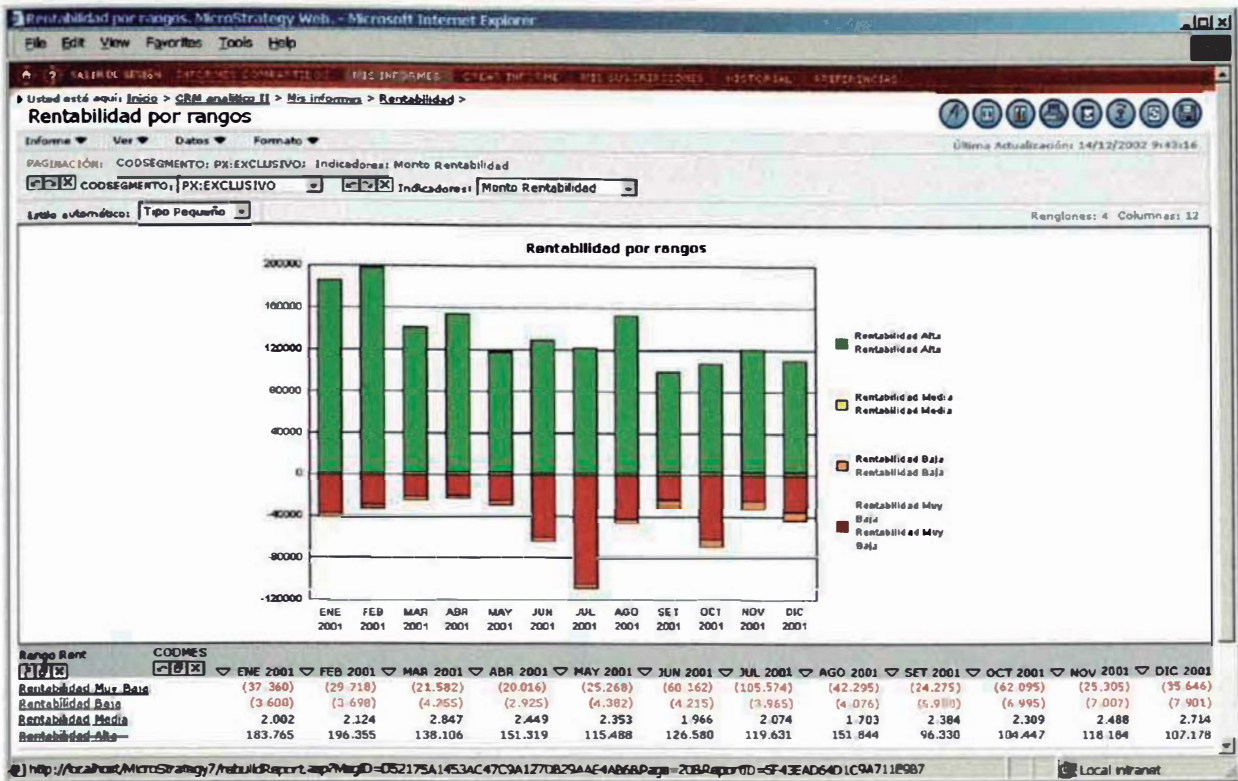


Figura 32. Segmento Exclusivo – Distribución del Monto de Rentabilidad

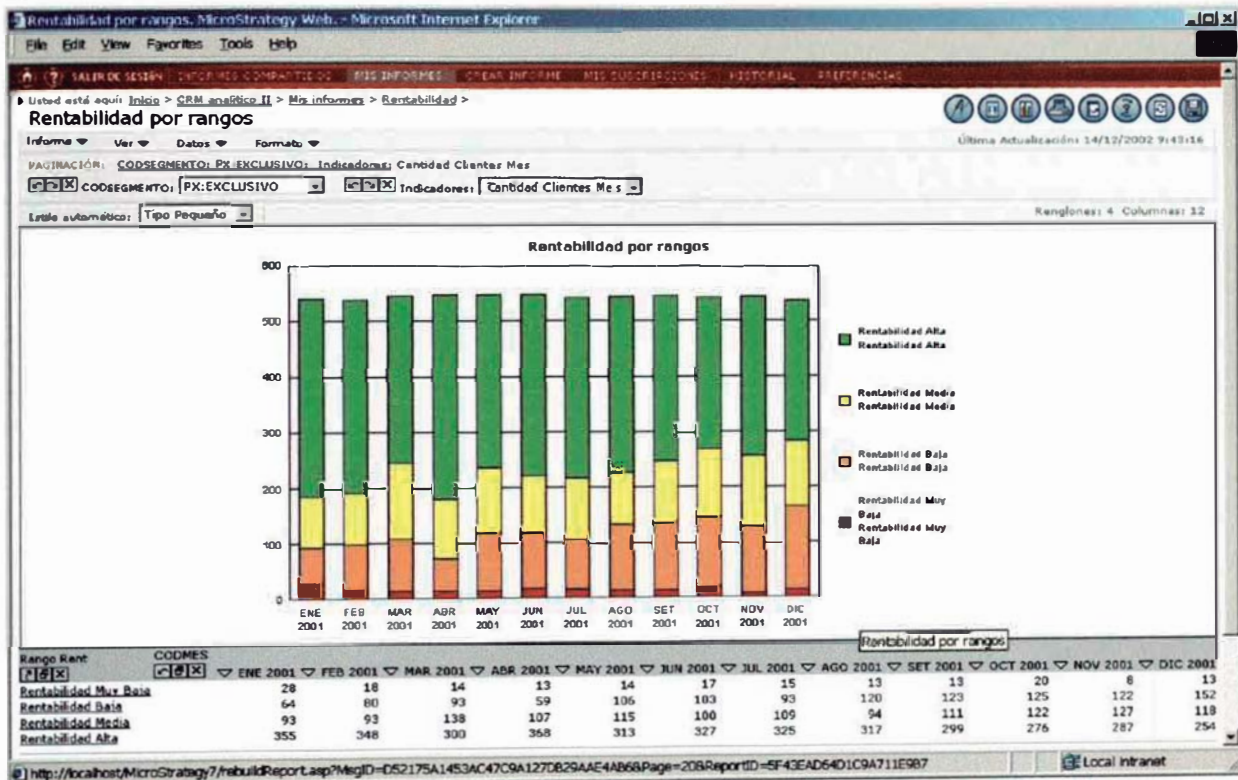


Figura 33. Segmento Exclusivo – Distribución de la Cantidad de Clientes

De los gráficos se desprende que el segmento exclusivo ha mantenido la cantidad de clientes constante durante el año. Sin embargo, la rentabilidad total del segmento si muestra una evolución decreciente (ver figura 32 y 33).



Figura 34. Segmento Negocio– Distribución del Monto de la Rentabilidad.

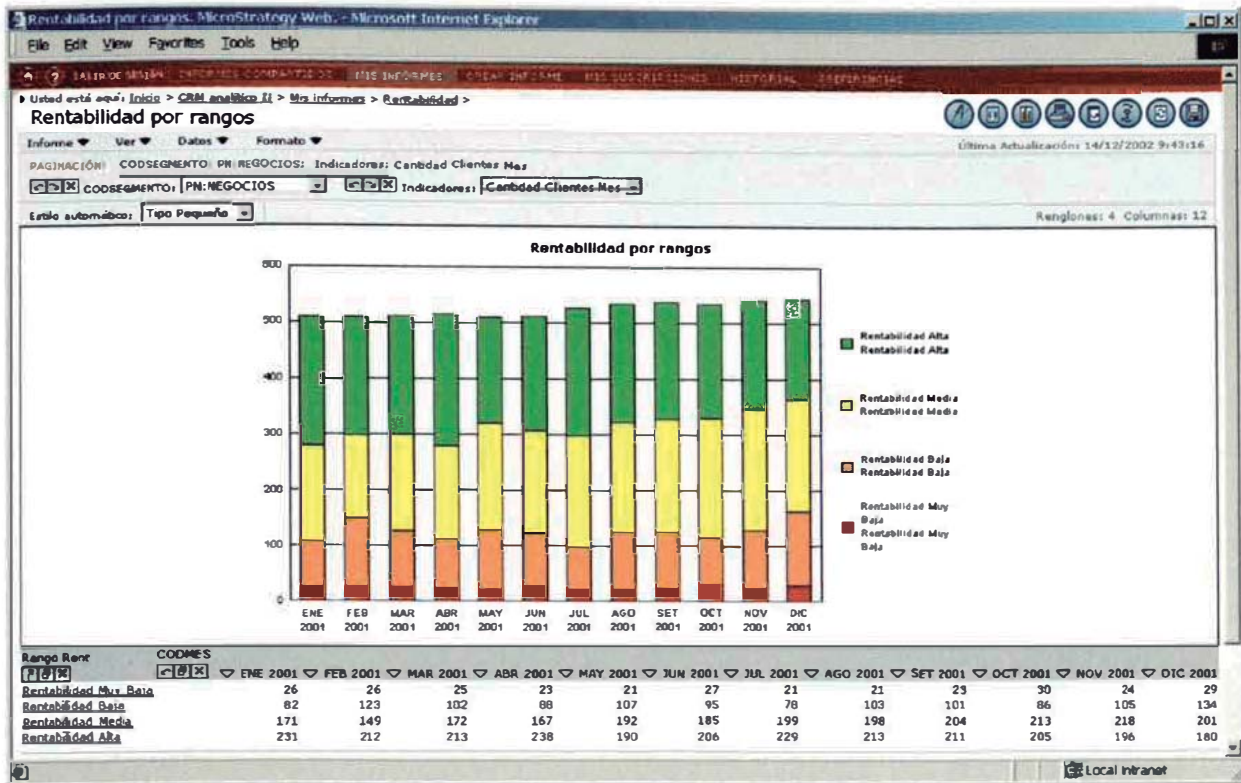


Figura 35. Segmento Negocio– Distribución de la Cantidad de Clientes

El segmento de negocios muestra un crecimiento en el monto de la rentabilidad total y también un ligero crecimiento en la cantidad de clientes. De todos los segmentos analizados, negocios es el único segmento que registra un crecimiento en la rentabilidad (ver figuras 34 y 35).

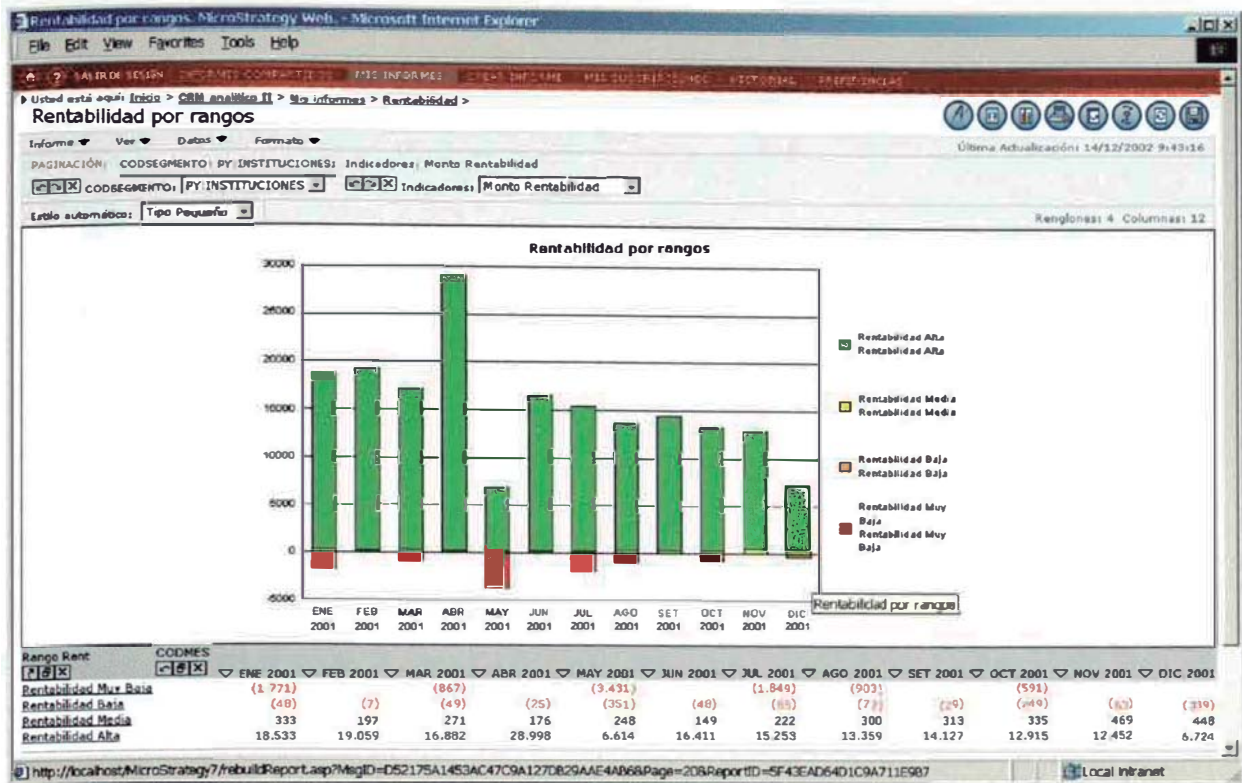


Figura 36. Segmento Instituciones – Distribución del Monto de Rentabilidad.

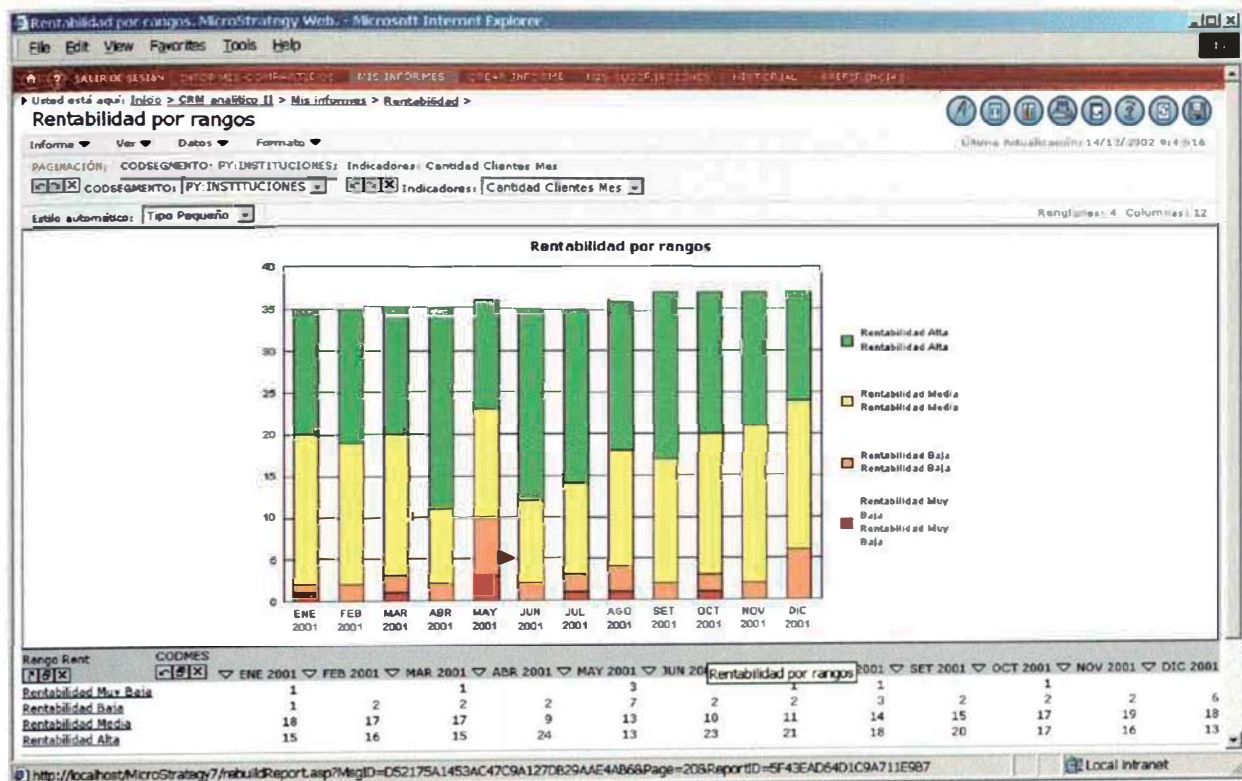


Figura 37. Segmento Instituciones – Distribución de la Cantidad de Clientes

Finalmente, se observa que el segmento institucional al igual que los segmentos de consumo y exclusivo muestra un ligero decrecimiento en el monto de la rentabilidad y la cantidad de clientes (ver figuras 36 y 37).

Con estos gráficos se identifica que los grupos de mayor rentabilidad son los segmentos Institucional y Exclusivo y que, salvo el segmento de Negocios (formado por las personas naturales con negocios propios), todos los demás registran un decrecimiento en la rentabilidad mensual. En el caso del segmento Exclusivo se observa también que este segmento alberga un grupo de clientes con una alta rentabilidad negativa que deberían ser identificados para tomar las acciones de desvinculación o de recuperación según corresponda.

El siguiente paso consistió en asignar un valor a la rentabilidad esperada para cada cliente. Este indicador es importante pues permite evaluar la posición actual de la relación del banco con el cliente e identificar posibles brechas de mejoras. Para poder determinar este indicador se empleó una herramienta de Data Mining (Knowledge Seeker IV) que permite encontrar relaciones entre variables y agrupar los datos en segmentos de comportamiento similar con respecto una variable dependiente. En este caso la variable dependiente escogida para el análisis fue la Rentabilidad Mensual, la cual fue analizada en su relación a las siguientes variables:

- Edad
- Cantidad de Meses como cliente del Banco
- Sexo
- Estado Civil
- Nivel Educación
- Segmento de Banca
- Cantidad de dependientes
- Tipo de Renta
- Ingreso Mensual
- Distrito de Residencia

Luego de analizar la información en la herramienta se obtuvo que la variable que tiene la relación más significativa con la Rentabilidad Mensual es el Segmento del cliente (el Segmento es una marca calculada por el Banco al momento de ingresar al sistema y se calcula en base a características del cliente tales como el nivel de ingresos y la edad). Este resultado confirma la validez en los cálculos empleados por el banco para la generación de este indicador.

En la figura 38 se muestra un gráfico con los resultados iniciales de este análisis. El rectángulo de la izquierda representa el universo total de clientes la distribución de sus rentabilidades mensuales. Se observa que el atributo empleado para este primer agrupamiento es el "codsegmentomes", código que identifica el segmento de cliente en cada mes. En la parte derecha se encuentran los grupos de segmentos (PG = empleados del Banco, PM = Clientes de segmento consumo, PN = Clientes del segmento negocios, PX = Clientes del segmento exclusivo, PY = Clientes del segmento institucional) y la distribución de la rentabilidad hallada en cada grupo.

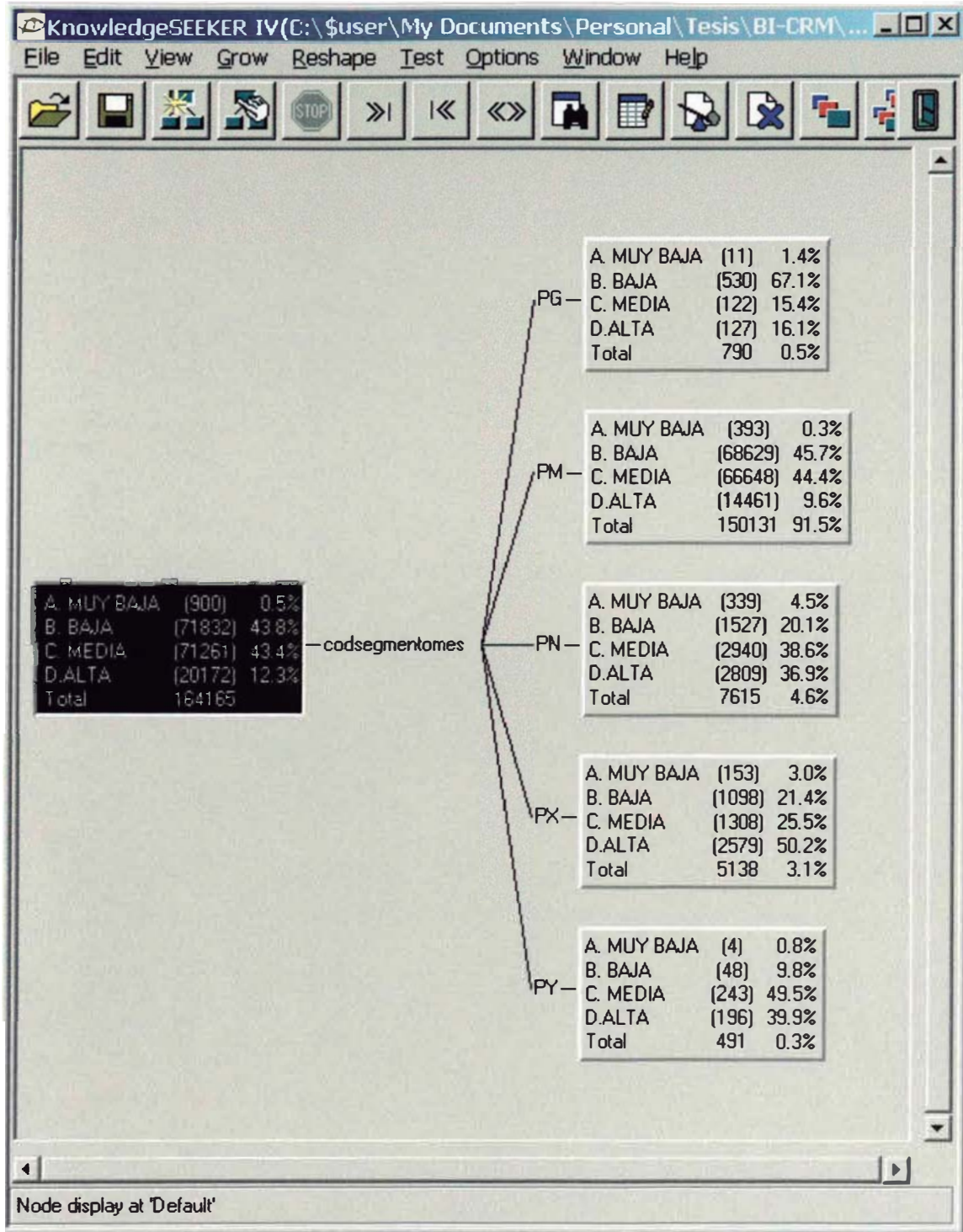


Figura 38. Variable de mayor relación con la rentabilidad: Segmento

Luego, en cada agrupación de segmentos se descubre que la siguiente variable con la relación más significativa es distinta para cada segmento. Así, dependiendo el segmento las siguientes variables a tomar en cuenta son:

- Edad, para el caso de los empleados del Banco (segmento PG) la edad es la variable de mayor relación con la rentabilidad. Este vínculo se explica en la estrecha relación que existe entre los productos que tienen los empleados en y su posición jerárquica dentro de la empresa, lo cual a su vez íntimamente relacionado con temas tales como la antigüedad laboral y la edad.
- Cantidad de Meses como cliente del Banco, de acuerdo al análisis efectuado, esta variable se identifica como la más significativa para distribuir la rentabilidad para el caso de los segmentos de clientes Institucionales, Exclusivos y Negocios (PY, PX, PN). Aunque cabe mencionar que dentro de cada segmento, los rangos hallados son distintos.
- Nivel Educación, La educación se convierte en la variable más significativa al analizar la información de los clientes de la Banca de Consumo. A medida que el nivel de educación se incrementa la rentabilidad también crece. Así, el promedio más alto se encuentra en el grupo que tiene registrado estudios de maestría.

Empleando estos criterios la herramienta de minería de datos identificó 34 grupos con un comportamiento similar respecto a la Rentabilidad Mensual. En la figura 39 se observan los 34 nodos finales y en cada uno de ellos en colores se identifica el % de clientes correspondiente a cada una de las categorías de rentabilidad definidas (Rojo = Muy Baja, Naranja = Baja, Amarillo = Media, Verde = Alta).

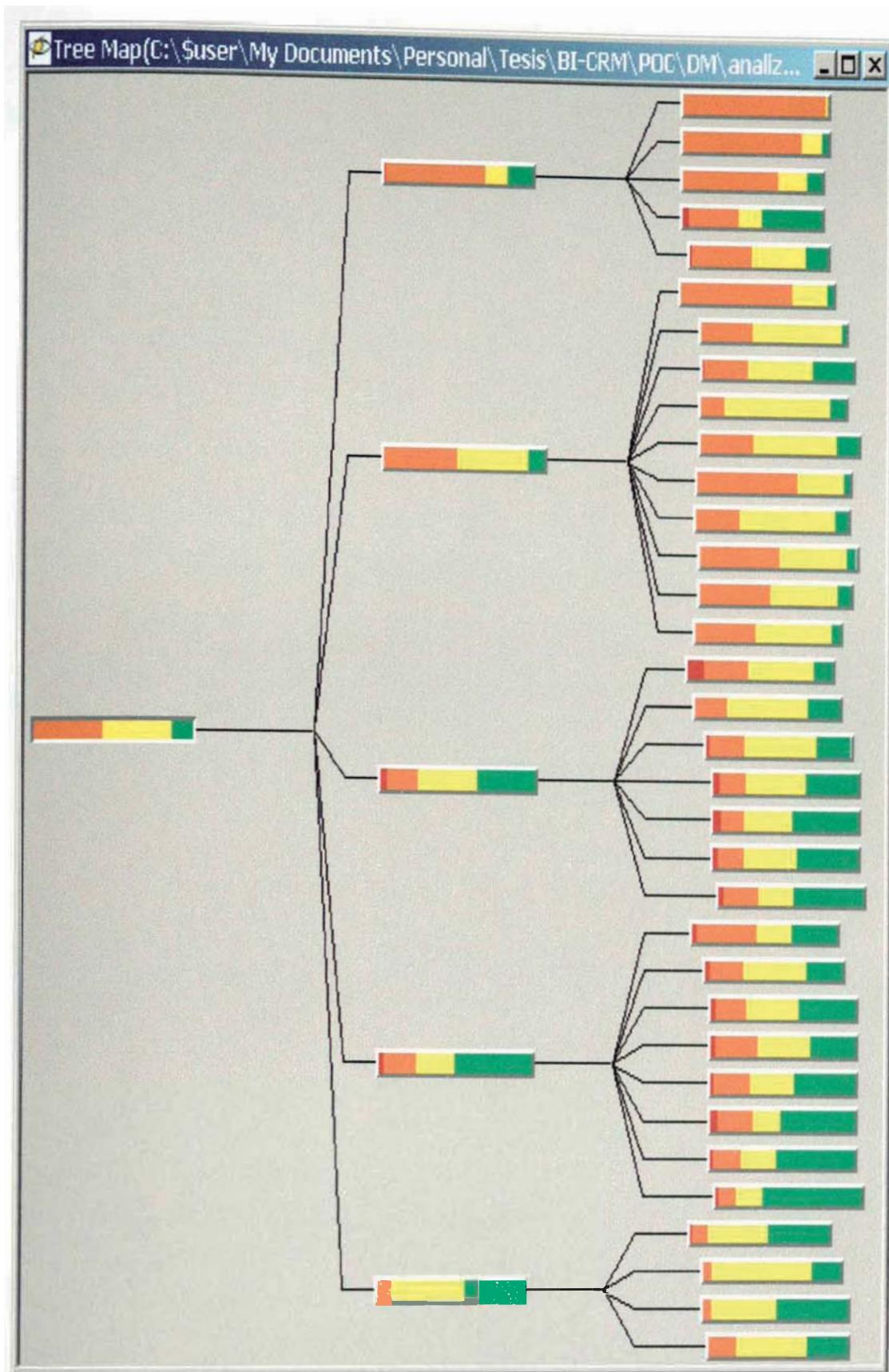


Figura 39. Análisis de Variables relacionadas con la rentabilidad

Luego de obtener estos grupos de clientes, se procedió a asignar una rentabilidad esperada tomando como base la rentabilidad promedio de los clientes con rentabilidad ALTA al interior de cada grupo. Empleando esta información se realimentó el modelo para poder analizar el comportamiento de los clientes de acuerdo a su rentabilidad esperada.

Analizando la rentabilidad por cada segmento del Banco se obtiene la distribución de la rentabilidad actual y el margen de brecha de rentabilidad que podría ser desarrollado en cada segmento de clientes.

CODSEGMENTO	CODRANGOMT	Indicadores	ENE 2001		FEB 2001		MAR 2001		ABR 2001		MAY 2001	
			Cantidad Clientes Mes	Monto Brecha Rentabilidad	Cantidad Clientes Mes	Monto Brecha Rentabilidad	Cantidad Clientes Mes	Monto Brecha Rentabilidad	Cantidad Clientes Mes	Monto Brecha Rentabilidad	Cantidad Clientes Mes	Monto Brecha Rentabilidad
			[-X]	[-X]	[-X]	[-X]	[-X]	[-X]	[-X]	[-X]	[-X]	[-X]
NO BANCO												
GRUPO												
CONSUMO												
< 00,-250>			49	(62.137)	32	(38.783)	29	(462.795)	26	(42.617)	28	(51.116)
[-250, 0>			2.728	(153.177)	3.717	(215.794)	3.702	(200.808)	2.999	(153.572)	4.132	(214.404)
[0, 50>			3.960	(144.944)	3.353	(136.791)	3.868	(161.972)	4.047	(120.265)	3.727	(125.806)
[50, 00>			1.021	98.862	1.002	85.262	779	61.401	1.440	102.741	843	95.134
NEGOCIOS												
< 00,-250>			26	(47.053)	26	(84.221)	25	(47.765)	23	(38.657)	21	(32.048)
[-250, 0>			82	(18.841)	123	(28.846)	102	(25.555)	88	(22.080)	107	(29.200)
[0, 50>			171	(27.443)	149	(25.608)	172	(29.687)	167	(31.071)	192	(38.151)
[50, 00>			231	56.361	212	37.886	213	51.336	230	61.108	190	63.869
EXCLUSIVO												
< 00,-250>			28	(41.044)	18	(33.419)	14	(24.585)	13	(22.619)	14	(30.247)
[-250, 0>			64	(11.564)	80	(16.511)	93	(17.641)	59	(14.967)	106	(29.706)
[0, 50>			93	(7.174)	93	(9.970)	138	(17.104)	107	(17.772)	115	(13.927)
[50, 00>			355	136.062	348	144.930	300	86.059	368	93.308	313	61.146
INSTITUCIONES												
< 00,-250>			1	(1.838)			1	(934)			1	(4.337)
[-250, 0>			1	(115)	2	(1.105)	2	(597)	2	(777)	2	(2.693)
[0, 50>			18	(3.729)	17	(3.036)	17	(3.707)	9	(2.252)	13	(2.709)
[50, 00>			15	15.192	16	15.235	15	12.740	24	23.936	13	3.890

Figura 40. Margen de Rentabilidad por segmento

Del reporte (figura 40) se concluye que el segmento Instituciones registra la menor brecha de rentabilidad (definida como la diferencia entre la rentabilidad actual y la rentabilidad potencial calculada anteriormente). Mientras que el segmento Consumo registra una alta brecha de rentabilidad que podría ser explotada por el banco. Considerando que este segmento alberga la mayor cantidad de clientes, las estrategias recomendadas para que el banco aproveche este margen de ganancias deberían estar enfocadas a obtener

pequeñas mejoras de rentabilidad por cada cliente. Una práctica usual de los bancos consiste en incrementar el cobro de comisiones, de mantenimiento de cuentas por ejemplo, a aquellos clientes que pertenezcan a este segmento.

En el caso de los clientes de los segmentos Exclusivo y Negocios, el margen de rentabilidad total es menor que el registrado por los segmentos de Consumo, la diferencia se presenta en la cantidad de clientes necesaria para cubrir la brecha de rentabilidad. En el segmento exclusivo, por ejemplo, para incrementar la rentabilidad mensual en S/.50,000 sería necesario incrementar las rentabilidades individuales de casi 100 clientes, que representan el 1% del universo de clientes. Es por esto que este grupo de clientes el que debería concentrar el mayor esfuerzo comercial del Banco a través de campañas de vinculación y venta cruzada.

El siguiente paso consistió en analizar la deserción de clientes. Para identificar a los clientes que se han retirado del banco se analizaron los productos que posee cada cliente agrupándolos de acuerdo a su estado de actividad. Así, se identificaron tres posibles estados:

- Productos vigentes, que agrupa a todas aquellas cuentas u operaciones que se encuentran vigentes en el sistema y con saldos distintos a cero.
- Productos vigentes pero con saldo cero, que agrupa a las cuentas u operaciones con estado vigente en el sistema pero que no registran ningún saldo en el periodo. Este grupo fue creado pues en muchos casos, el cliente no “cierra” sus cuentas cuando decide dejar de ser cliente, sino que solamente retira su dinero.
- Productos no vigentes, que agrupa a aquellas cuentas u operaciones que ya fueron cerradas, ya sea por el cliente o por el banco.

Para el análisis de la deserción se consideró como clientes activos aquellos que tuvieron al menos 1 producto en la categoría de productos vigentes en el mes. Empleando esta definición se realizó una consulta a la base de datos para graficar la evolución de los clientes que estuvieron activos en Enero 2001 (Ver figura 41).

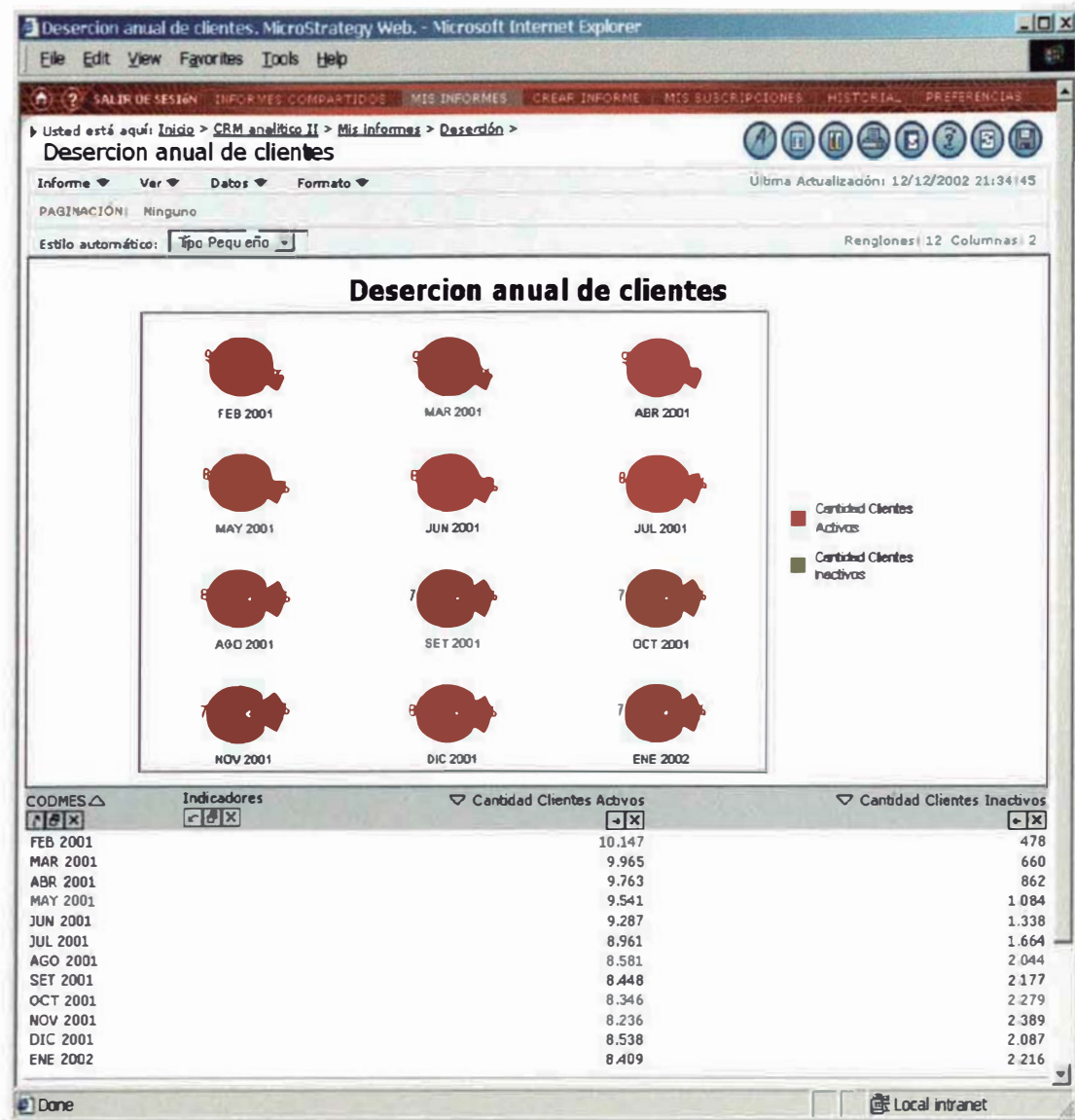


Figura 41. Deserción Anual de Clientes

Del grafico se desprende que pasado un año, del grupo de clientes que estuvieron activos en Enero 2001, el 21.9% dejó de figurar en el grupo de clientes activos en Enero del 2002. Cabe resaltar que, puesto que este indicador también referencial y depende del estado de vigencia de los productos, es posible encontrar algunas pequeñas distorsiones en la evolución de su comportamiento, en la figura por ejemplo se observa que el ratio de Diciembre fue un poco más alto que el de Noviembre, a diferencia del resto de meses en los cuales se noto un constante incremento de la deserción.

Empleando las mismas definiciones descritas para el análisis de la deserción, se analizó luego el ratio de incorporación de clientes. Nuevamente se definió con un cliente activo a aquel que tuvo durante el periodo un producto vigente con saldo distinto de cero.

Tomando como base a los clientes que estuvieron activos en Enero 2002, se procedió a analizar desde cuándo figuraron como clientes activos (Ver figura 42). Se descubrió que de estos clientes, un 29% no figuró en Enero del 2001.

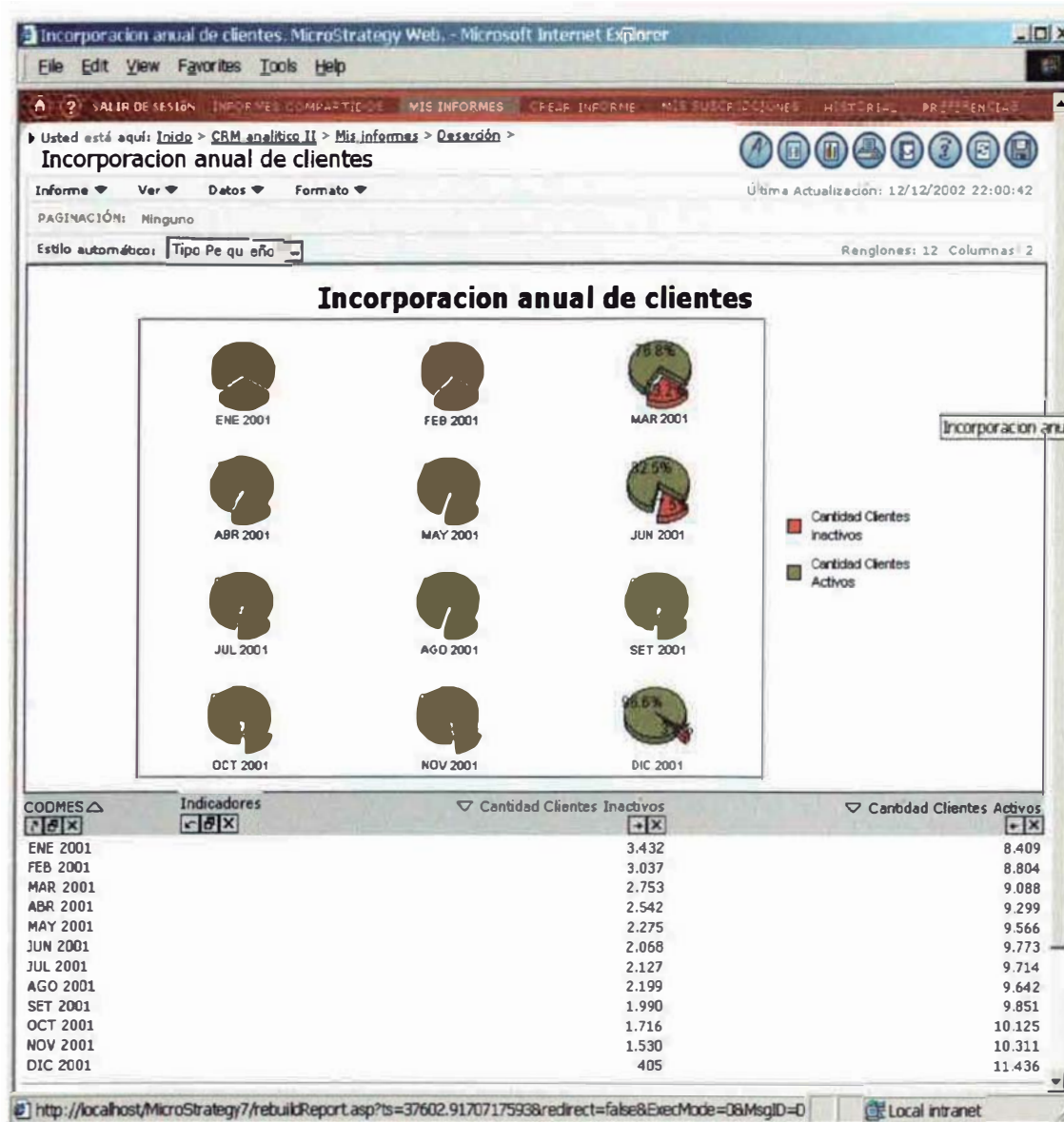


Figura 42. Incorporación Anual de Clientes

5. CONCLUSIONES

- La visión de CRM renueva un conjunto de conceptos tradicionales asociados a la administración de los clientes pero bajo la luz de un escenario distinto: las posibilidades tecnológicas para capturar y analizar información relevante a este proceso.
- En el aspecto financiero, CRM se presenta como una alternativa de gran interés pues, por sus características, las entidades financieras suelen establecer relaciones duraderas con sus clientes creando las condiciones suficientes para poder explotar esa relación, invirtiendo en aprender de ellas.
- Dentro de los componentes que forman parte de un proyecto CRM en una institución financiera, el CRM Analítico constituye el cerebro de las operaciones. Este componente reúne la información relevante a la relación del Banco con sus clientes y ofrece herramientas a los usuarios para analizar esta información y diseñar acciones comerciales que mejoren, mantengan o terminen la relación con cada cliente.
- El CRM Analítico, por sus características y objetivos, está incluido en la categoría de lo que el mercado denomina Aplicaciones de Inteligencia de Negocios. Bajo este término se consideran un conjunto de productos de software y hardware orientados a brindar a las organizaciones capacidad analítica sobre su información. Dentro de los componentes típicos de Inteligencia de Negocios que una entidad financiera requiere se encuentran: el Data Warehouse, los Data Marts departamentales, las Herramientas de Consultas de información, las herramientas de Data Mining, las herramientas de transformación y carga de datos y la

Metadata. Salvo en los casos en los que se opte por implantar una solución integral de CRM (tales como las que ofrece Siebel, Peoplesoft, Vantive, Oracle CRM, etc.) que incluyen una arquitectura pre-diseñada, desarrollar un CRM Analítico requiere necesariamente de la presencia de estos componentes.

- Se comprobó las ventajas de trabajar con tecnología ROLAP, la cual recoge las funcionalidades de la navegación OLAP pero permite reutilizar un modelo de datos implantado sobre un motor de base de datos relacional, brindando a la organización la flexibilidad de poder explotar la información de múltiples maneras y facilitando enormemente el trabajo de actualización y mantenimiento del modelo.
- En un proyecto de inteligencia de negocios la calidad de los datos empleados para poblar los modelos multidimensionales juegan un rol protagónico en la validez de los resultados obtenidos a través de las herramientas de análisis. En el caso desarrollado, fue necesario realizar diversas transformaciones y adecuaciones, tales como la eliminación de datos incorrectos, poblamiento de campos nulos, estandarización de códigos, creación de rangos de valores, etc., para asegurar la calidad de la información analizada y evitar distorsiones en los resultados finales.
- Con respecto a la información en el caso de estudio se encontró que de los clientes que producen ganancias al Banco el 17% aporta el 80% de estas ganancias al Banco. Este grupo de clientes es el que debe concentrar los mayores esfuerzos del Banco para mantener el nivel de rentabilidad que generan y en donde sea posible incrementarlo.
- De los clientes que producen pérdidas al Banco el 1% aporta el 80% de estas pérdidas al Banco. Esto representa una gran oportunidad para el Banco, pues al tratarse de una base de clientes reducida es mucho más fácil identificarlos y tomar las acciones apropiadas ya sea para mejorar su rentabilidad o para desvincularlos del Banco.

- En el análisis de la distribución de la rentabilidad se encontró que el segmento de clientes exclusivos es el más rentable, sin embargo, se observa también que en este grupo se encuentra un elevado porcentaje de clientes con una rentabilidad muy baja. Lo cual sugiere que algunos de los criterios empleados por el banco para marcar a los clientes como exclusivos no está filtrando adecuadamente a todas las personas.
- Luego de estimar una rentabilidad potencial esperada, se encontró que el Banco tiene grandes oportunidades de mejora para incrementar la rentabilidad de los clientes. Se identificó que en el caso de los clientes de la banca exclusiva existe un margen de rentabilidad de 50,000 soles mensuales que el banco podría obtener si incrementara la rentabilidad de aproximadamente 100 clientes, un número reducido que fácilmente puede ser trabajado con una campaña focalizada. En el caso del segmento Negocios existe una brecha promedio de 90,000 soles mensuales asociada a aproximadamente 300 clientes.
- Con respecto al análisis de fidelidad y duración de la relación del cliente con el banco, se encontró que el ratio de deserción anual es de 20%, por lo cual se puede estimar que el periodo de vida de la relación de un cliente con el Banco es de aproximadamente 5 años.
- Asimismo, el ratio de incorporación de clientes al Banco es de 29%, que restado al ratio de deserción nos deja un ratio neto de incorporación de clientes del 9% anual.

6. RECOMENDACIONES

- La implantación de una estrategia CRM integral, es un proceso de largo alcance y debe ser desarrollado con la planificación y el compromiso suficiente. La teoría y la práctica aconsejan comenzar con proyectos pilotos que permitan a la organización aprender y mejorar por etapas.
- Aunque la implantación de una solución CRM requiere, además del CRM Analítico, de un conjunto de componentes orientados a soportar los procesos operativos, es factible desarrollar la parte analítica de manera paralela a la operativa. Sin embargo, el riesgo de trabajar así sería no contar con la capacidad instalada suficiente para poder cumplir con las estrategias sugeridas por el componente analítico.
- Para soportar las características y funcionalidades que un CRM Analítico requiere, es necesario contar con un modelo de datos que explote la información existente de la mejor manera, sin descuidar aspectos tales como la flexibilidad y tiempo de respuesta de las consultas. Del estudio realizado se ha comprobado que, siguiendo la metodología de Ralph Kimball, los modelos multidimensionales de datos son de gran valor pues ofrecen a los analistas de información una variada gama de posibilidad de navegación sobre los datos, liberándolos de manera relativa de la necesidad de familiarizarse detalladamente con aspectos técnicos del modelo de datos y permitiéndoles concentrarse en los aspectos propios del análisis.
- El equipo técnico involucrado en un proyecto de inteligencia de negocios debería estar integrado por un especialista en los sistemas fuente, un especialista en administración de las herramientas ETL, un

administrador de la base de datos, desarrolladores de aplicaciones OLAP y en caso de ser necesario un especialista en minería de datos con conocimiento en modelos estadísticos. En general, sólidos conocimientos de SQL por parte del equipo técnico son muy útiles para la manipulación de los datos al interior de la base de datos relacional empleada.

- El uso de la tecnología ROLAP o MOLAP en un proyecto de inteligencia de negocios debe ser evaluado de acuerdo a los requerimientos y expectativas de la organización. Si la escalabilidad y el volumen de datos son un factor crítico para el proyecto, se debería elegir la opción ROLAP, pues las bases de datos relacionales escalan más que las bases de datos multidimensionales propietarias. Por otro lado, si el tiempo de respuesta de las consultas de negocios sobre el modelo multidimensional representan el aspecto más importante para el proyecto, la opción más adecuada es el uso de tecnología MOLAP, la cual gracias a la generación de cubos precalculados permite responder las consultas de negocios con mayor rapidez que una solución ROLAP. Otro punto a favor del uso de tecnología ROLAP es la flexibilidad del modelo, que puede ser explotado con diversas herramientas que soporten interfaces ODBC, OLEDB o JDBC y modificado más fácilmente que las estructuras de datos de las soluciones MOLAP.

BIBLIOGRAFÍA

1. LIBROS

- The Data Warehouse Toolkit. Ralph Kimball. John Wiley & Sons Inc. 1996.
- The Data Warehouse Lifecycle Toolkit. Ralph Kimball, Laura Reeves, Margy Ross, Warren Thornthwaite. John Wiley & Sons Inc. 1998.
- Data Warehousing, Integración de Infomación para la major toma de decisions. Harjinder S. Gill, Prakash C. Rao. Prentice Hall Hispoamericana. 1996.
- The Data Warehouse Challenge. Michael H. Brackett. John Wiley & Sons Inc. 1996.
- Data Warehouse Lifecycle – Overview, Planning, Requirements and Architecture. The Data Warehousing Institute. Laura Reeves. 2000.
- The One to One Future – Building Relationships One Customer at a Time. Don Peppers and Martha Rogers, Ph.D. Currency and Doubleday Publishing. 1997.
- Enterprise One to One – Tools to competing in the Interactive Age. Don Peppers and Martha Rogers, Ph. D. Currency and Doubleday Publishing. 1999.

2. INFORMES, ARTÍCULOS Y REVISTAS

- Creating a CRM Business Case. Naras Eechambadi, Melissa Ortiz. Quaero. 2000.
- Full Conference Proceedings. National Center for Database Marketing. 2001.

- Get a Customer-Centric View, Process Secure Transactions, Segment for Prioritization. Peoplesoft CRM for Financial Services. 2002.
- Using OLAP and Multidimensional Data for Decision Making. Revista IT Pro. IEEE. Setiembre - Octubre 2001.
- Data Warehouse and Data Mart Administration. W. H. Inmon. Pine Cone Systems, Inc. 1997.
- Metadata in the Data Warehouse. W. H. Inmon. 2000.
- What is a Data Warehouse?. W.H.Inmon, Ed Young. 2000.
- What is a Data Mart?. W.H.Inmon, Ed Young. 1997.
- The Operational Data Store. W. H. Inmon. 2000.
- Independent Data Mart vs. Dependent Data Marts. W. H. Inmon. 1999.
- Data Warehouse across a large enterprise. W.H.Inmon, R.H. Terdeman. 1999.
- Quality Data – An Improbable Dream?. Elizabeth Vannan. Revista Educause. Número 1 2001. Página 56-58.
- Data Warehousing: Cleaning and Transforming Data. Revista InfoDB. Volumen 10 Número 6. Página 11-12.
- Mercado de Base de Datos Relacionales para Inteligencia de Negocios: Diferenciación de Productos. K. Strange. Gartner Group. 6 de junio de 2001.
- Introducción a los 4 estilos de OLAP. H. Dresner, K. Strange. Gartner Group. 28 de marzo de 1996.
- Los 4 estilos de OLAP: Enfoque Relacional. H. Dresner, K. Strange. Gartner Group. 28 de marzo de 1996.
- Los 4 estilos de OLAP: Bases de Datos Multidimensionales. H. Dresner, K. Strange. Gartner Group. 28 de marzo de 1996.
- Customer Relationship Management (CRM): Overview. Rochelle Shaw, Jim Davies. Gartner Group. 11 de octubre de 2001.
- Customer Data Quality and Integration: The Foundation of Successful CRM. S.Nelson, R.Singhai, W.Janowski, N.Frey. Gartner Group. 26 de noviembre de 2001.
- MicroStrategy 7i Product Suite. Product Report. Gartner Group. 20 de septiembre de 2002.

- Business Objects 2000. Product Report. Gartner Group. 25 de octubre de 2002.
- Hyperion Solutions Essbase OLAP Server. Product Report. Gartner Group. 20 de marzo de 2001.
- Microsoft SQL Server 2000 Analysis Services. Product Report. Gartner Group. 4 de octubre de 2000.
- Oracle Express. Product Report. Gartner Group. 11 de octubre de 2000.

3. DIRECCIONES INTERNET

CRM y Marketing 1 a 1:

www.1to1.com, Peppers and Rogers Group

www.crmcommunity.com

www.quaero.com

Inteligencia de Negocios:

www.dw-institute.com

www.billinmon.com

www.rkimball.com

www.olapcouncil.org

www.olapreport.com

www.microstrategy.com

www.ibm.com/software/data/bi

www.dmreview.com

www.intelligententerprise.com/info_centers/bi

Analistas Independientes de Tecnología y Tendencias del Mercado:

www.gartner.com

www.idc.com

ANEXO A

DESCRIPCIÓN DE LAS TABLAS Y COLUMNAS EMPLEADAS EN EL MODELO PROPUESTO

DESCRIPCIÓN DE LAS TABLAS Y COLUMNAS EMPLEADAS EN EL MODELO PROPUESTO

Tabla	Descripción	Columnas	Tipo de Datos	Desc. Columna	PK?	FK?
D_Agencia	Agencias y Sucursales del Banco en donde los clientes pueden realizar sus transacciones.	CodAgencia	INTEGER		Si	
		CodRegion	SMALLINT			Si
		CodDistritoAge	CHAR(6)			Si
		DesAgencia	VARCHAR(50)			
D_Anio	Año de la información	CodAnio	SMALLINT		Si	
D_Aplicativo	Aplicativo desde el cual se obtuvo la información transaccional.	CodAplicativo	CHAR(1)		Si	
		DesAplicativo	VARCHAR(20)			
D_Banca	Banca actual del cliente.	CodBanca	CHAR(1)		Si	
		DesBanca	VARCHAR(30)			
D_Canal	Canal por donde se efectuó la transacción.	CodCanal	CHAR(1)		Si	
		DesCanal	VARCHAR(25)			
D_CategoriaEco	Categoria económica de la persona.	CodCategoriaEco	VARCHAR(4)	Código de Categoria Socio Económica.	Si	
		DesCategoriaEco	VARCHAR(50)			
D_ClaRieBCP	Código de Clasificación de Riesgo otorgado por el Banco a la persona.	CodClaRieBCP	CHAR(1)		Si	
		DesClaRieBCP	VARCHAR(40)			
D_ClaRieSBS	Código de Clasificación de Riesgo otorgado por la SBS a la persona.	CodClaRieSBS	CHAR(1)		Si	
		DesClaRieSBS	VARCHAR(20)			
D_Cliente	Clientes del Banco. Son clientes aquellos que tienen un producto o servicio vigente con el Banco.	CodPersonaCli	INTEGER		Si	
		CodSubsegmento	CHAR(3)			Si
		CodMesesCli	SMALLINT			Si
		CodMesCliInicia	INTEGER			Si
		CodPersona	INTEGER			Si
		CodAgenciaCli	INTEGER			Si
		CodSector	VARCHAR(6)			Si
		FeInicio	DATE	Fecha de Inicio de la relación con el cliente		
D_CtaOpe	Cuentas u operaciones.	MtoApertura	DECIMAL(16,2)			
		FecApertura	DATE			
		CodMonedaCuenta	SMALLINT			Si
		MtoCierre	DECIMAL(16,2)			
		CodRazonCierre	SMALLINT			Si
		FecCierre	DATE			
		CodEstado	SMALLINT			Si
		CodCtaOpe	BIGINT			Si
		CodProducto	SMALLINT			Si
		CodPersonaCli	INTEGER			Si
D_DepartamentoAge	Departamento en donde se encuentra ubicada la agencia.	CodDepartamentoAge	CHAR(6)		Si	
		DesDepartamentoAge	VARCHAR(25)			
D_DistritoAge	Distrito en donde se	CodDistritoAge	CHAR(6)		Si	

Tabla	Descripción	Columnas	Tipo de dato	Desc. Columna	PK?	FK?
	encuentra ubicada la agencia.	CodProvinciaAge	CHAR(6)			Si
		DesDistritoAge	VARCHAR(30)			
D_EstadoCivil	Estado civil de la persona.	CodEstCivil	VARCHAR(3)	Indicador de Estado Civil. CAS : Casado (a) CBS : Casado con separación de bienes CON : Conviviente SEP : Separado (a) o Divorciado (a) SOL : Soltero (a) VIU : Viudo (a)	Si	
		DesEstCivil	VARCHAR(40)	Descripción del Indicador de Estado Civil.		
D_EstadoCliente	Estado de la relación del cliente con el Banco.	CodEstadoCliente	SMALLINT		Si	
		DesEstadoCliente	VARCHAR(20)			
D_EstadoCuenta	Estado de la cuenta u operación.	CodEstado	SMALLINT		Si	
		DesEstado	VARCHAR(20)			
D_FamiliaProducto	Máxima agrupación de tipos de productos.	CodFamiliaProducto	CHAR(3)		Si	
		DesFamiliaProducto	VARCHAR(50)			
D_GerenteBanca	Gerente de la Banca a la cual pertenece el cliente.	CodGerenteBanca	CHAR(6)		Si	
		DesGerenteBanca	VARCHAR(25)			
D_GerenteZona	Gerente de la Zona a la cual pertenece el cliente.	CodGerenteZona	CHAR(6)		Si	
		CodGerenteBanca	CHAR(6)			Si
		DesGerenteZona	VARCHAR(50)			
D_JefeGrupo	Jefe del grupo de funcionarios al que pertenece el cliente.	CodJefeGrupo	CHAR(6)		Si	
		CodGerenteZona	CHAR(6)			Si
		DesJefeGrupo	VARCHAR(50)			
D_Mes	Mes de la información.	CodMes	INTEGER		Si	
		CodAnio	SMALLINT			Si
		DesMes	VARCHAR(20)			
D_MesCliInicia	Mes en que se inicio la relación con el cliente.	CodMesCliInicia	INTEGER		Si	
		DesMesCliInicia	VARCHAR(20)			
D_MesesCli	Cantidad de meses de antigüedad de la relación con el cliente.	CodMesesCli	SMALLINT		Si	
D_Moneda	Moneda de la cuenta u operación.	CodMoneda	SMALLINT		Si	
		DesMoneda	VARCHAR(50)			
D_MonedaTran	Moneda en que se efectuó la transacción.	CodMonedaTran	CHAR(4)		Si	
		DesMonedaTran	CHAR(15)			

Tabla	Descripción	Columnas	Tipo de Dato	Desc. Columna	PK?	FK?
D_NivelEdu	Nivel de educación de la persona.	CodNivelEdu	VARCHAR(3)	Indicador Nivel Educativo. ANA : Analfabeto BAC : Bachiller DOC : Doctorado FAR : Fuerzas Armadas y Policiales LIC : Licenciado MAE : Maestría NDI : No dio información solicitada PRI : Primaria SEC : Secundaria SUP : Superior TEC : Técnico TEI : Técnico TIT : Titulado	Si	
		DesNivelEdu	VARCHAR(45)	Descripción del Indicador Nivel Educativo.		
D_PaisNac	País de nacimiento de la persona.	CodPaisNac	VARCHAR(3)		Si	
		DesPaisNac	VARCHAR(40)			
D_Persona	Personas que tienen o han tenido alguna relación con el Banco.	CodPersona	INTEGER		Si	
		CodSituaCasa	CHAR(5)			Si
		CodRangoEdad	SMALLINT			Si
		CodTipoClienteBCP	CHAR(2)			Si
		CodClaRieSBSPer	CHAR(1)			Si
		ApellidoPaterno	VARCHAR(25)			
		ApellidoMaterno	VARCHAR(25)			
		PrimerNombre	VARCHAR(25)			
		SegundoNombre	VARCHAR(25)			
		NumeroDocumento	CHAR(13)	Numero del Documento del cliente		
		FeclIngreso	DATE	Fecha de Ingreso de la persona a la base de datos		
		CodTipoIngreso	CHAR(1)			Si
		CodTipoDocumento	CHAR(1)			Si
		CodEstCivil	VARCHAR(3)	Indicador de Estado Civil. CAS : Casado (a) CBS : Casado con separación de bienes CON : Conviviente SEP : Separado (a) o Divorciado (a) SOL : Soltero (a) VIU : Viudo (a)		Si

Tabla	Descripción	Columnas	Tipo de Dato	Desc. Columna	PK?	FK?
		CodSituacionLab	VARCHAR(3)	Indicador de Situación Laboral. DEP : Dependiente EST : Estudiante IND : Independiente JUB : Jubilado NDI : No dio información solicitada SUC : Su Casa		Si
		CodNivelEdu	VARCHAR(3)	Indicador Nivel Educativo. ANA : Analfabeto BAC : Bachiller DOC : Doctorado FAR : Fuerzas Armadas y Policiales LIC : Licenciado MAE : Maestría NDI : No dio información solicitada PRI : Primaria SEC : Secundaria SUP : Superior TEC : Técnico TEI : Técnico TIT : Titulado		Si
		CodSexo	VARCHAR(1)	Indicador de Sexo. F : Femenino M : Masculino		Si
		CodPaisNac	VARCHAR(3)			Si
		CodCategoriaEco	VARCHAR(4)	Código de Categoría Socio Económica.		Si
		CodProfesion	CHAR(3)	Código de la Profesión		Si
		NbrFantasia	VARCHAR(40)	Nombre Fantasia. Nombre con que se conoce a la empresa.		
		FecNacimiento	DATE	Fecha de Nacimiento de la persona		
		FecConstitucion	DATE	Fecha Constitución Empresa.		
		FecUltMod	DATE	Fecha Última Modificación Escrita.		
		FecCaducidadEmp	DATE	Fecha Caducidad Empresa		
		NumRuc	CHAR(10)	Número de RUC		
		DigitoRUC	CHAR(1)	Código Verificador del RUC.		
		NbrComercial	VARCHAR(60)	Nombre Comercial.		
		CtdDependientes	INTEGER	Cantidad de personas dependientes.		
		CodDistritoPer	CHAR(6)	Distrito en el que vive la persona.		Si

Tabla	Descripción	Columnas	Tipo de Dato	Desc. Columna	PK?	FK?
		DesDomicilio1	VARCHAR(100)	Domicilio de la Persona.		
		DesDomicilio2		Segundo Domicilio de la persona.		
		MtoIngBrutoMes	DECIMAL(16,2)	Ingreso Bruto Mensual.		
		MtoOtroingMes	DECIMAL(16,2)	Otros Ingresos mensuales.		
		CodRenta	SMALLINT			Si
D_Producto	Tipo de productos ofrecidos por el Banco.	CodProducto	SMALLINT		Si	
		CodTipoProducto	CHAR(1)			Si
		CodFamiliaProducto	CHAR(3)			Si
		DesProducto	VARCHAR(40)			
D_Profesion	Profesión de la persona.	CodProfesion	CHAR(3)	Código de la Profesión	Si	
		DesProfesion	VARCHAR(40)			
D_ProvinciaAge	Provincia en donde se encuentra la agencia.	CodProvinciaAge	CHAR(6)		Si	
		CodDepartamentoAge	CHAR(6)			Si
		DesProvinciaAge	VARCHAR(25)			
D_RangoEdad	Rango de edad de la persona.	CodRangoEdad	SMALLINT		Si	
		DesRangoEdad	VARCHAR(30)			
D_RangoMoneda	Rango del monto transaccionado.	CodRangoMoneda	CHAR(8)		Si	
		CodMonedaTran	CHAR(4)			Si
		DesRangoMoneda	VARCHAR(30)			
		MtoRangoIni	DECIMAL(16,2)			
		MtoRangoFin	DECIMAL(16,2)			
D_RangoMonedaDet	Rango detallado del monto transaccionado.	CodRangoMonedaDet	CHAR(8)		Si	
		CodRangoMoneda	CHAR(8)			Si
		DesRangoMonedaDet	VARCHAR(30)			
		MtoRangoIniDet	DECIMAL(16,2)			
		MtoRangoFinDet	DECIMAL(16,2)			
D_RangoMtoRent	Rango de Rentabilidad.	CodRangoMtoRent	INTEGER		Si	
		DesRangoMtoRent	VARCHAR(20)			
		MtoRentMin	DECIMAL(16,2)			
		MtoRentMax	DECIMAL(16,2)			
D_RangoVinCli	Rango de Vinculación del cliente.	CodRangoVinCli	SMALLINT		Si	
		DesRangoVinCli	VARCHAR(20)			
		NumVinCliMin	INTEGER			
		NumVinCliMax	INTEGER			
D_RazonCierre	Motivo por el que cerro la cuenta.	CodRazonCierre	SMALLINT		Si	
		DesRazonCierre	VARCHAR(20)			
D_Region	Regio geográfica.	CodRegion	SMALLINT		Si	
		DesRegion	VARCHAR(25)			
D_Sector	Sector del cliente. Un sector es una agrupación de clientes asignada a un funcionario de negocios.	CodSector	VARCHAR(6)		Si	
		CodJefeGrupo	CHAR(6)			Si
		DesSectorista	VARCHAR(50)			
D_Segmento	Segmento actual del cliente	CodSegmento	CHAR(2)		Si	
		CodBanca	CHAR(1)			Si
		DesSegmento	VARCHAR(35)			
D_Sexo	Genero de la persona.	CodSexo	VARCHAR(1)	Indicador de Sexo. F : Femenino M : Masculino	Si	

Tabla	Descripción	Columnas	Tipo de Datos	Desc. Columna	PK?	FK?
		DesSexo	VARCHAR(15)	Descripción del Indicador de Sexo.		
D_SituacionLab	Situación laboral de la persona.	CodSituacionLab	VARCHAR(3)	Indicador de Situación Laboral. DEP : Dependiente EST : Estudiante IND : Independiente JUB : Jubilado NDI : No dio información solicitada SUC : Su Casa	Si	
		DesSituacionLab	VARCHAR(35)	Descripción del Indicador de Situación Laboral.		
D_Subsegmento	Sub-segmento del cliente.	CodSubsegmento	CHAR(3)		Si	
		CodSegmento	CHAR(2)			Si
		DesSubsegmento	VARCHAR(60)			
D_TipoCargoAbono	Indica si la transacción fue un cargo o un abono.	CodCargoAbono	CHAR(1)		Si	
		DesCargoAbono	VARCHAR(15)			
D_TipoClienteBCP	Identifica la manera en que la persona se volvió cliente del Banco.	CodTipoClienteBCP	CHAR(2)		Si	
		DesTipoClienteBCP	VARCHAR(25)			
D_TipoDocumento	Identifica el tipo de documento de la persona.	CodTipoDocumento	CHAR(1)		Si	
		DesTipoDocumento	VARCHAR(50)			
D_TipoIngreso	Identifica el tipo de ingreso de la persona a la base de datos.	CodTipoIngreso	CHAR(1)		Si	
		DesTipoIngreso	VARCHAR(20)			
D_TipoMonetario	Identifica si la transacción fue monetario o de consulta.	CodTipoMonetario	CHAR(1)		Si	
		DesTipoMonetario	VARCHAR(15)			
D_TipoProducto	Identifica si el producto es un pasivo, activo o servicio.	CodTipoProducto	CHAR(1)		Si	
		DesTipoProducto	VARCHAR(30)			
D_TipoRenta	Renta: Primera: Rentas producidas por el arrendamiento, subarrendamiento y cesión de bienes. Segunda: Rentas de otros capitales. Tercera: Rentas del comercio, la industria y otras consideradas por la ley. Cuarta: Rentas del trabajo independiente. Quinta: Rentas del trabajo dependiente.	CodRenta	SMALLINT		Si	
		DesRenta	VARCHAR(20)			
D_TipoSituaCasa	Identifica el tipo de propiedad de la casa.	CodSituaCasa	CHAR(5)		Si	
		DesSituaCasa	VARCHAR(20)			
D_TipoTra	Identifica el tipo de la transacción.	CodTipoTra	CHAR(7)		Si	
		CodTipoMonetario	CHAR(1)			Si
		CodCanal	CHAR(1)			Si

Tabla	Descripción	Columnas	Tipo de Dato	Desc. Columna	PK?	FK?
		CodAplicativo	CHAR(1)			Si
		CodCargoAbono	CHAR(1)			Si
		DesTipoTra	VARCHAR(50)			
F_Cliente	Información histórica mensual sobre los principales indicadores del cliente.	CodMes	INTEGER		Si	Si
		CodPersonaCli	INTEGER		Si	Si
		CodSubsegmentoMes	CHAR(3)			Si
		CodRangoVinCli	SMALLINT			Si
		CodRangoMtoRent	INTEGER			Si
		CodRangoMtoRentEsp	INTEGER			Si
		CodSectorMes	VARCHAR(6)			Si
		CodClaRieBCP	CHAR(1)			Si
		FecClaRieBCP	DATE	Fecha de la clasificación BCP.		
		CodClaRieSBS	CHAR(1)			Si
		FecClaRieSBS	DATE	Fecha de la Clasificación SBS.		
		CtdCtaOpeAct	SMALLINT	Cantidad de Cuentas u Operaciones Activas.		
		CtdCtaOpePas	SMALLINT	Cantidad de cuentas u operaciones Pasivas.		
		CtdProAct	SMALLINT	Cantidad de Productos Activos,		
		CtdProPas	SMALLINT	Cantidad de productos pasivos.		
		MtoSaldoAct	DECIMAL(16,2)	Suma de los saldos de los productos Activos.		
		MtoSaldoPas	DECIMAL(16,2)	Suma de Saldos de los productos Pasivos.		
		MtoRentabilidad	DECIMAL(16,2)	Rentabilidad mensual del cliente.		
		MtoIngresosFin	DECIMAL(16,2)	Ingresos Financieros mensuales del Cliente.		
		MtoIngresosNoFin	DECIMAL(16,2)	Ingresos no Financieros mensuales del Cliente.		
		MtoEgresosFin	DECIMAL(16,2)	Egresos Financieros mensuales del Cliente.		
		MtoEgresosNoFin	DECIMAL(16,2)	Egresos no Financieros mensuales del Cliente.		
		MtoRentabilidadEsp	DECIMAL(16,2)	Rentabilidad esperada o potencial del cliente.		
		MtoProvision	DECIMAL(16,2)	Provisión para el cliente.		
		MtoDeudaSBS	DECIMAL(16,2)	Deuda mensual registrada en la SBS.		
		MtoDeudaBCP	DECIMAL(16,2)	Deuda total en el BCP.		
		NumFactorVinCli	DECIMAL(16,2)	Factor de vinculación mensual del cliente.		
		CtdCliente	INTEGER	Campo auxiliar para realizar conteos de clientes.		

Tabla	Descripción	Columnas	Tipo de dato	Desc. Columna	PK?	FK?
F_CuentaOperacion	Información histórica sobre los indicadores de cuentas y operaciones.	CodCtaOpe	BIGINT		Si	Si
		CodMes	INTEGER		Si	Si
		CodEstadoMes	SMALLINT			Si
		MtoSaldoPromedio	DECIMAL(16,2)	Saldo Promedio Mensual de la cuenta.		
		MtoSaldoVigente	DECIMAL(16,2)	Saldo en estado vigente del mes.		
		MtoSaldoVencido	DECIMAL(16,2)	Saldo en estado vencido del mes.		
		MtoSaldoJudicial	DECIMAL(16,2)	Saldo en estado Judicial del mes.		
		MtoSaldoFinMes	DECIMAL(16,2)	Saldo final del mes.		
		CtdDepositos	SMALLINT	Cantidad de depósitos asociados a la cuenta durante el mes.		
		CtdRetiros	SMALLINT	Cantidad de retiros asociados a la cuenta durante el mes.		
		MtoDepositado	DECIMAL(16,2)	Monto total depositado por la cuenta durante el mes.		
		MtoRetirado	DECIMAL(16,2)	Monto total retirado por la cuenta durante el mes.		
		MtoIngresosFin	DECIMAL(16,2)	Ingresos financieros de la cuenta durante el mes.		
		MtoEgresosFin	DECIMAL(16,2)	Costos financieros de la cuenta durante el mes.		
		MtoIngresosNoFin	DECIMAL(16,2)	Ingresos no financieros de la cuenta durante el mes.		
		MtoEgresosNoFin	DECIMAL(16,2)	Egresos no financieros de la cuenta durante el mes.		
		MtoMargenPeriodo	DECIMAL(16,2)	Margen de la cuenta en el mes		
		NumFactorVin	DECIMAL(16,2)	Factor de vinculación de la cuenta		
		CtdCtaOpe	BIGINT	Campo auxiliar para realizar conteos de cuentas.		
F_Persona	Información histórica sobre los indicadores de personas.	CodMes	INTEGER		Si	Si
		CodPersona	INTEGER		Si	Si
		CodClaRieSBSPerMes	CHAR(1)			Si
		MtoDeudaSBS	DECIMAL(16,2)	Monto total de la deuda registrada en la SBS durante el mes.		
		CtdPersona	INTEGER	Campo auxiliar para realizar conteos de personas.		
F_RetencionCliente	Información sobre los indicadores de retención del cliente.	FlgClientelnicia	SMALLINT			
		FlgClienteActivo	SMALLINT			
		FlgClienteTermina	SMALLINT			
		CodPersonaCli	INTEGER		Si	Si

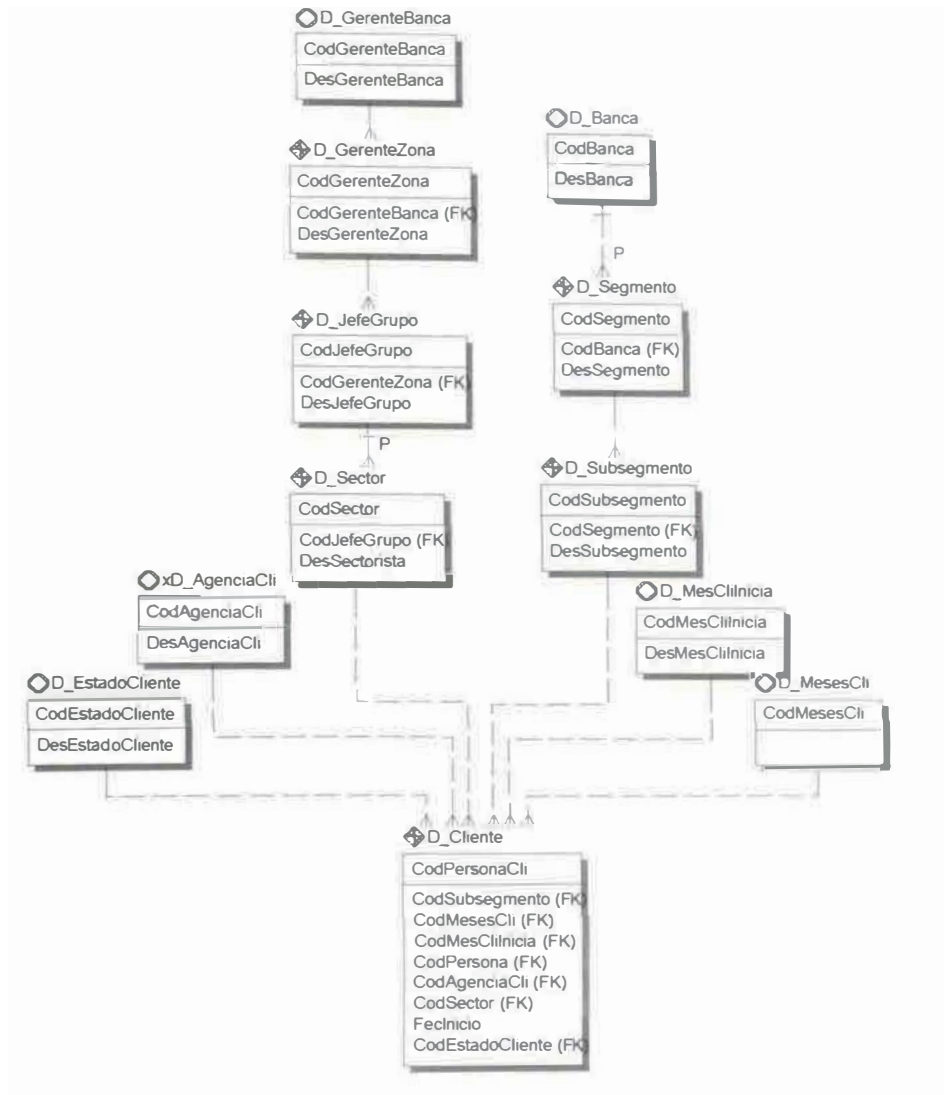
Tabla	Descripción	Columnas	Tipo de Dato	Desc. Columna	PK?	FK?
		CodMes	INTEGER		Si	Si
F_Transaccion	Información sobre los indicadores de transacciones.	CodMes	INTEGER		Si	Si
		CodPersonaCli	INTEGER		Si	Si
		CodCtaOpe	BIGINT		Si	Si
		CodTipoTra	CHAR(7)		Si	Si
		CodRangoMonedaDet	CHAR(8)		Si	Si
		MtoTransaccion	DECIMAL(16,2)	Monto Transaccionado		
		CtdTransaccion	BIGINT	Cantidad de transacciones		
xD_AgenciaCli	Agencia asociada al cliente.	CodAgenciaCli	INTEGER		Si	
		DesAgenciaCli	VARCHAR(20)			
XD_ClaRieSBSPer		CodClaRieSBSPer	CHAR(1)		Si	
		DesClaRieSBSPer	VARCHAR(20)			
XD_ClaRieSBSPerMes		CodClaRieSBSPerMes	CHAR(1)		Si	
		DesClaRieSBSPerMes	VARCHAR(20)			
XD_DistritoPer		CodDistritoPer	CHAR(6)		Si	
		DesDistritoPer	VARCHAR(30)			
XD_EstadoCuentaMes	Estado mensual de la cuenta.	CodEstadoMes	SMALLINT		Si	
		DesEstadoMes	VARCHAR(20)			
XD_MonedaCuenta		CodMonedaCuenta	SMALLINT		Si	
		DesMonedaCuenta	VARCHAR(50)			
XD_RangoMtoRentEsp		CodRangoMtoRentEsp	INTEGER		Si	
		DesRangoMtoRentEsp	VARCHAR(20)			
XD_SectorMes		CodSectorMes	VARCHAR(6)		Si	
		DesSectoristaMes	VARCHAR(50)			
XD_SubsegmentoMes		CodSubsegmentoMes	CHAR(3)		Si	
		DesSubsegmentoMes	DECIMAL(16,2)			

ANEXO B

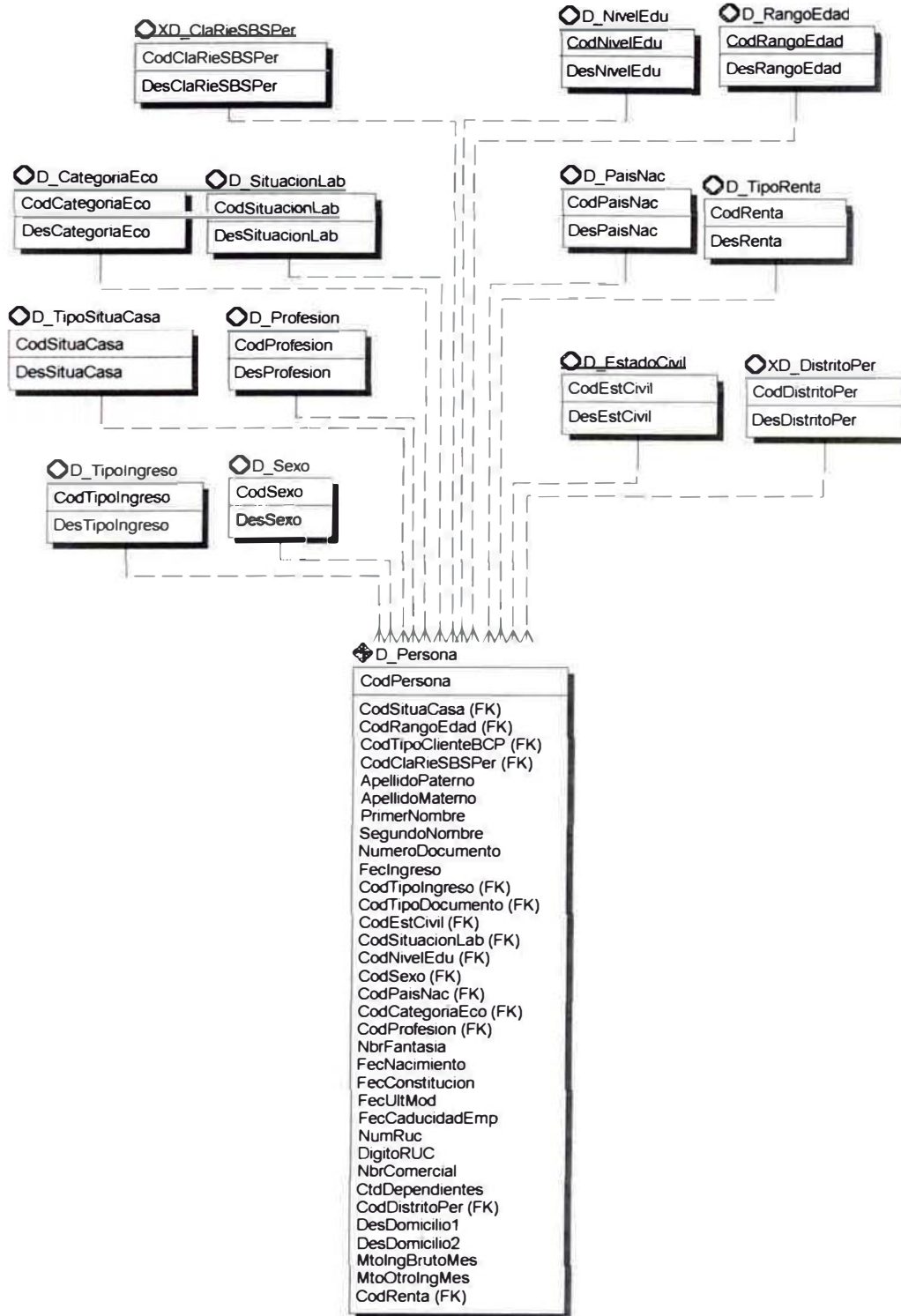
DIAGRAMAS ENTIDAD-RELACIÓN DEL MODELO CRM PROPUESTO

DIAGRAMAS ENTIDAD-RELACIÓN DEL MODELO CRM PROPUESTO

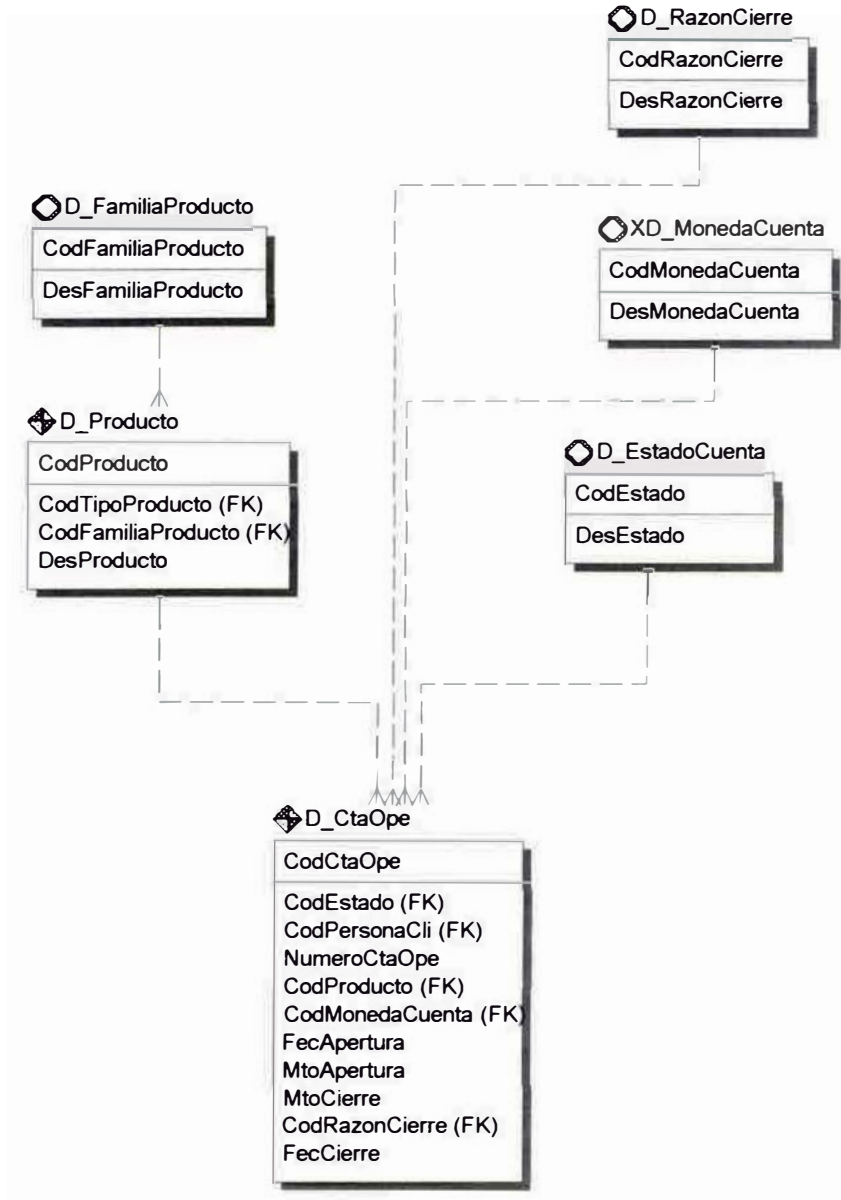
Dimensión Cliente



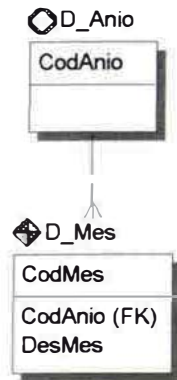
Dimensión Persona



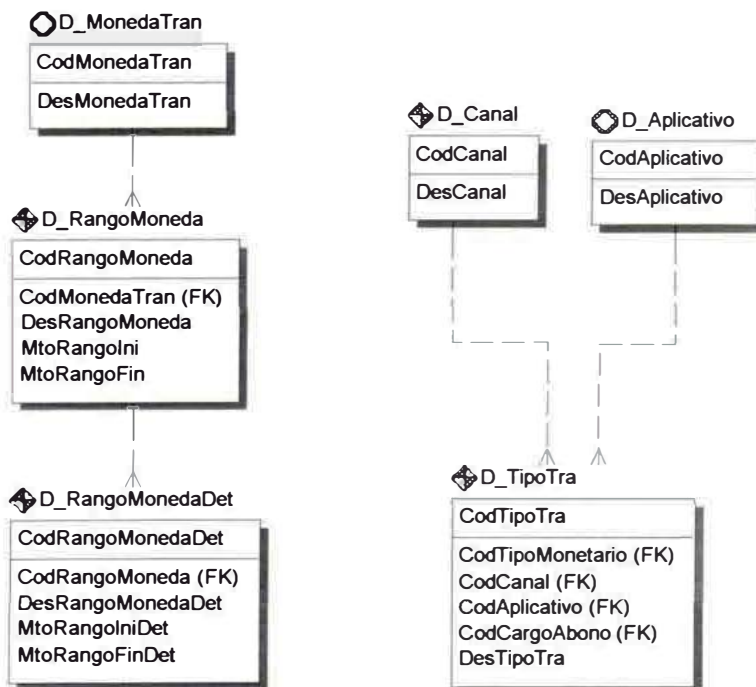
Dimensión Producto



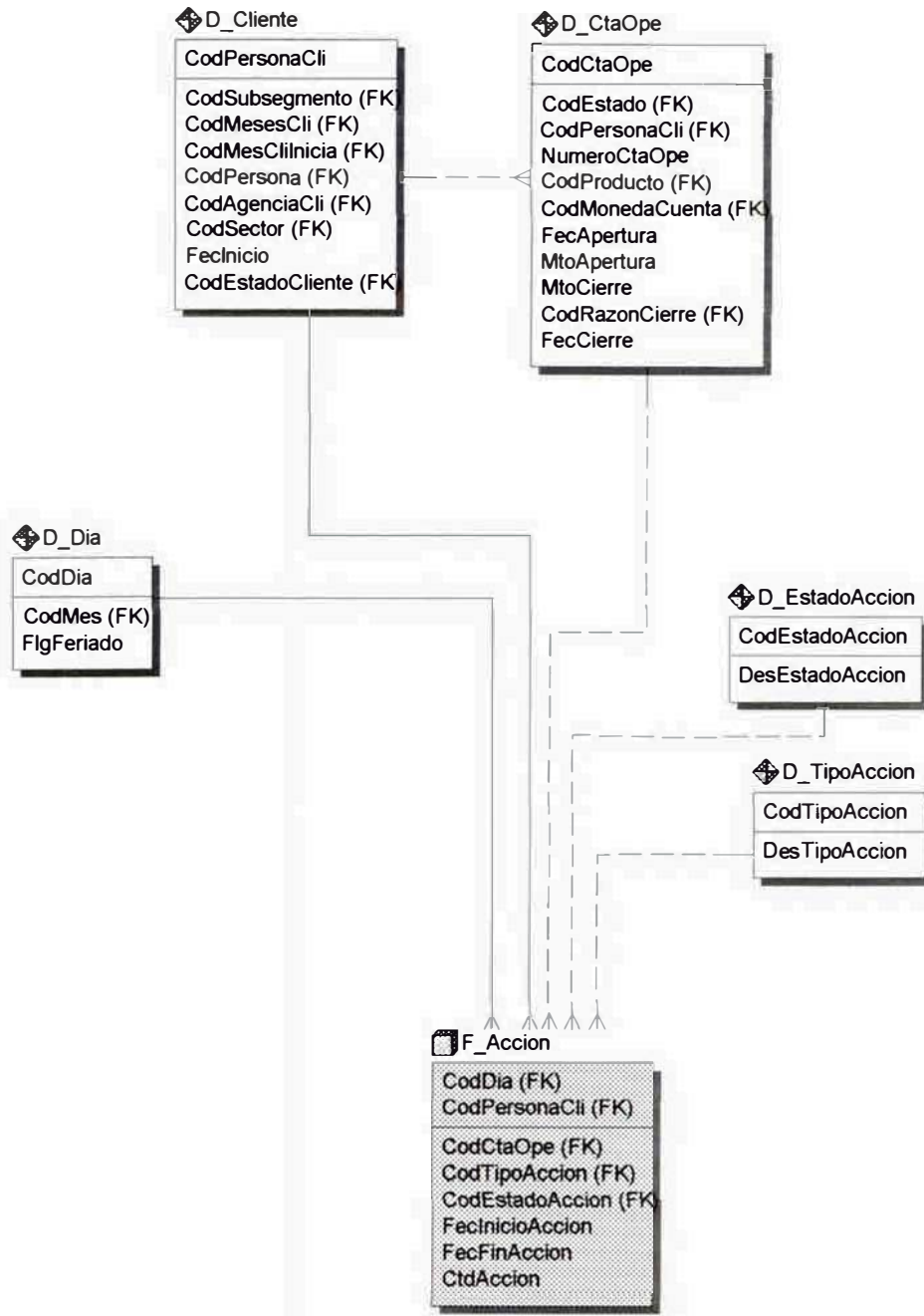
Dimensión Tiempo



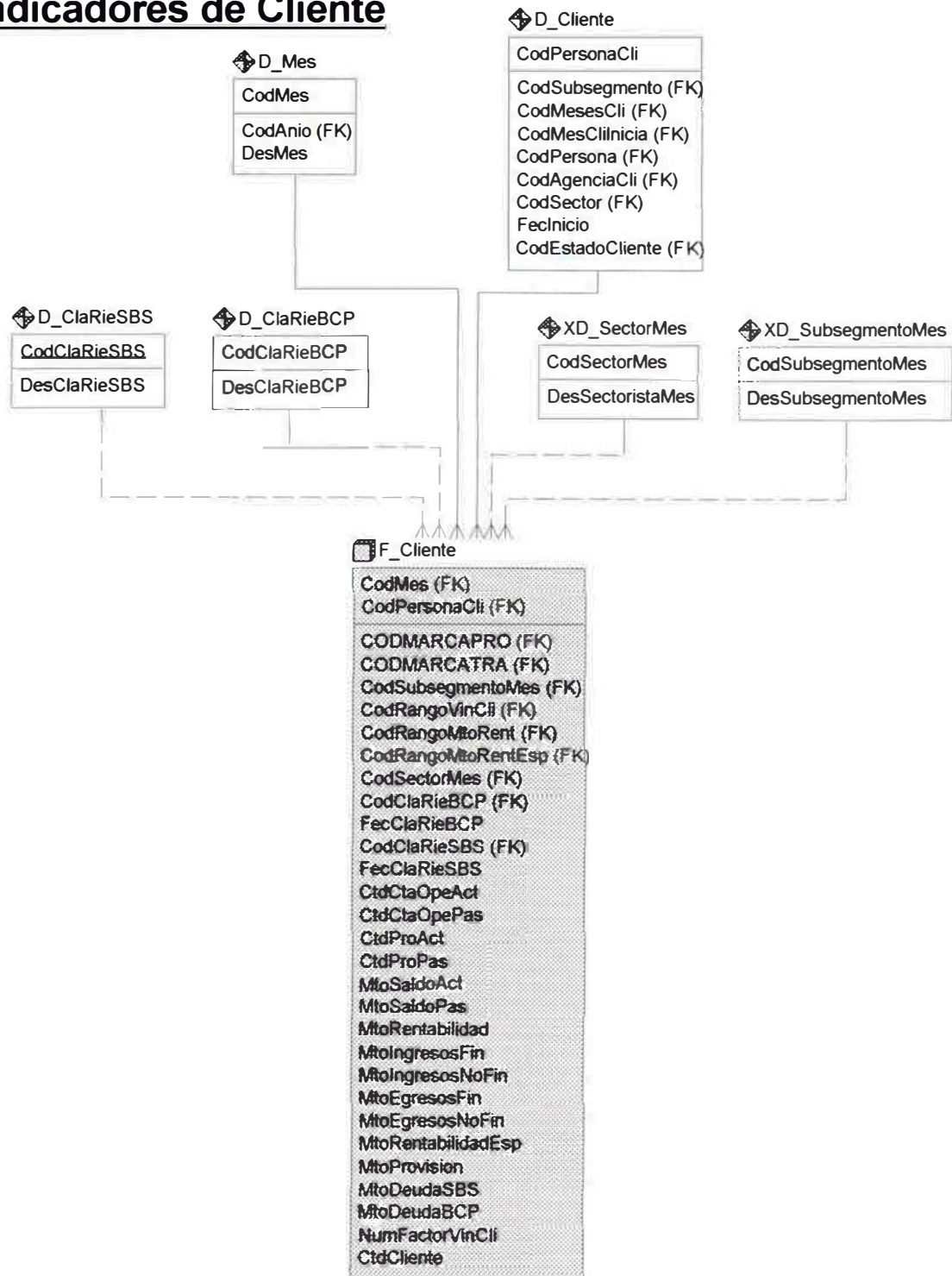
Dimensión Transaccion



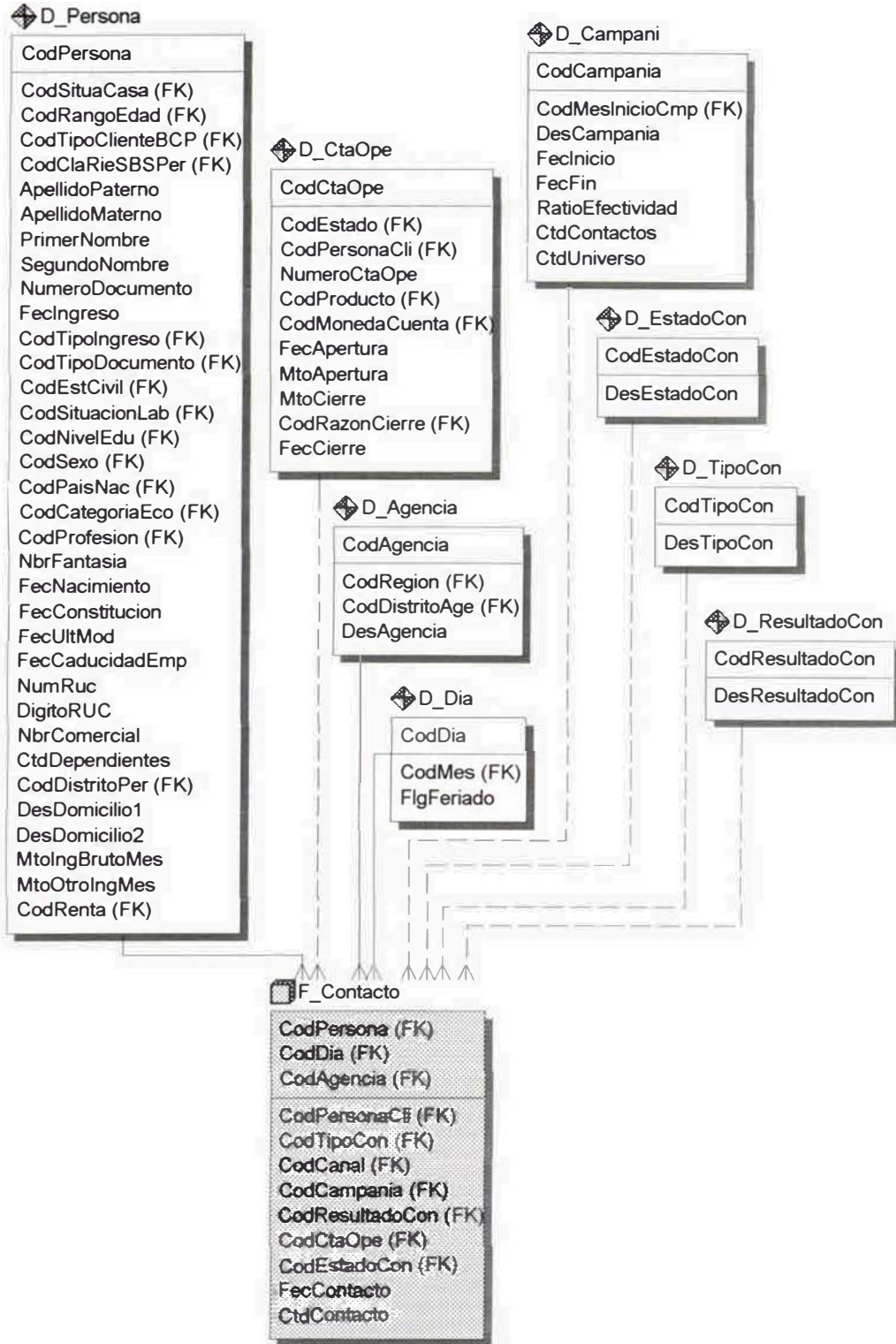
Indicadores de Acciones Comerciales



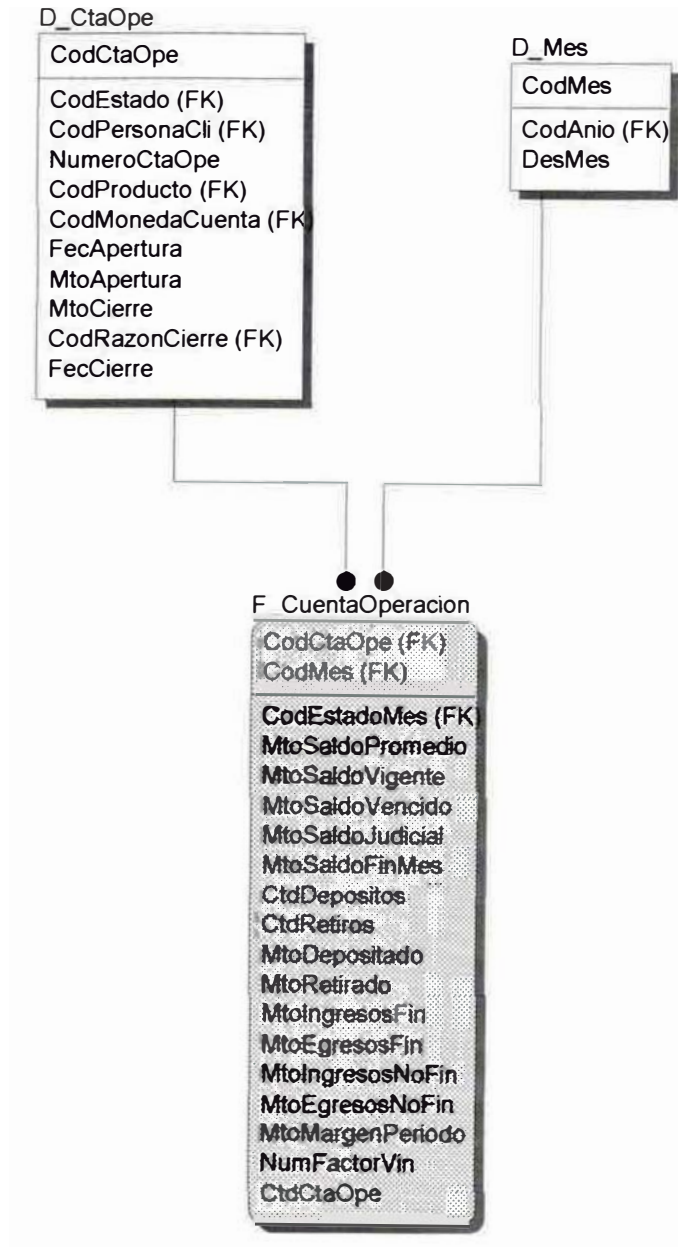
Indicadores de Cliente



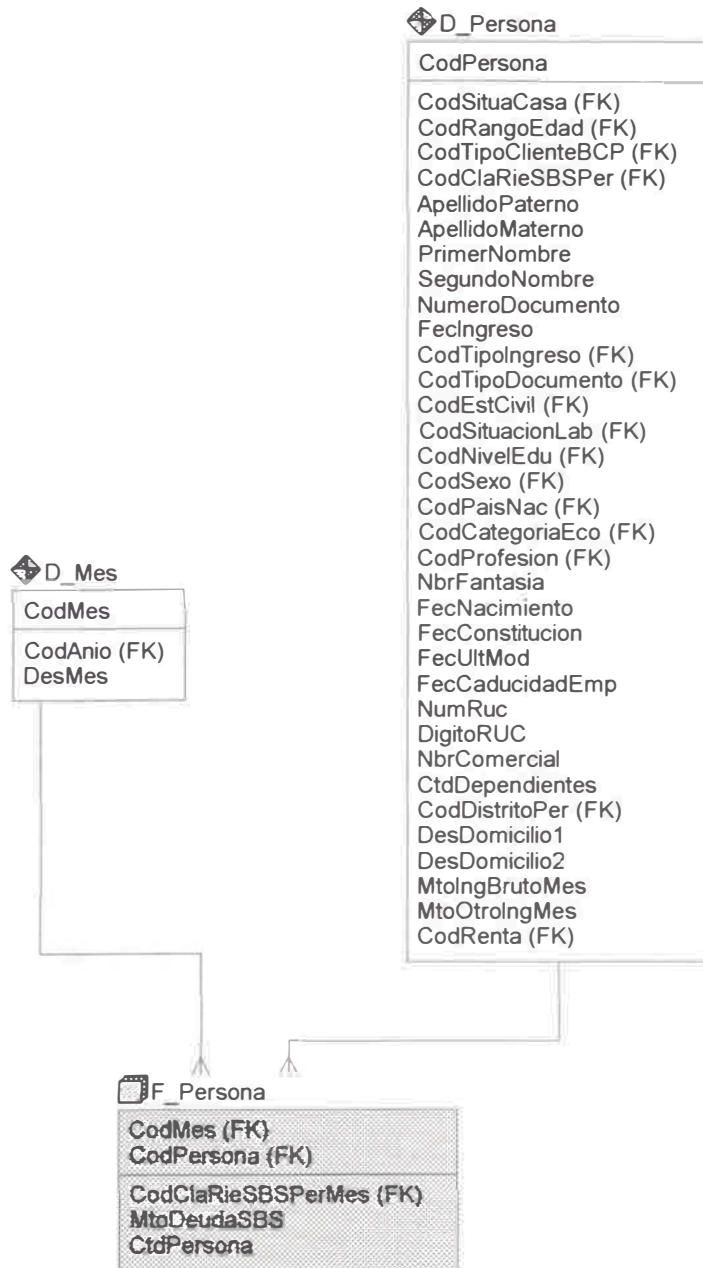
Indicadores de Contacto



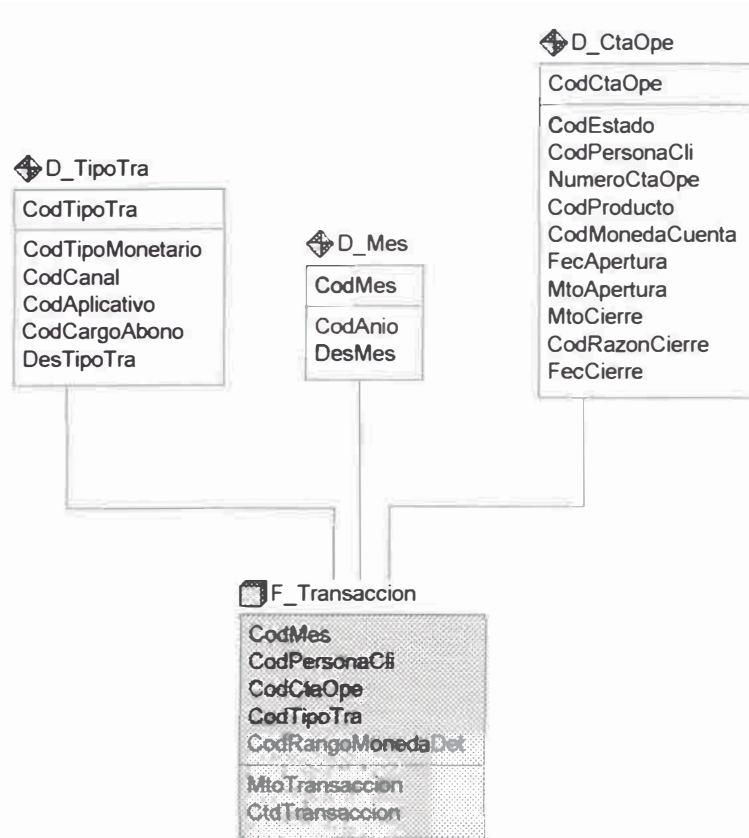
Indicadores de Productos



Indicadores de Persona



Indicadores Transaccionales



ANEXO C

SCRIPT DE CREACIÓN DE LA BASE DE DATOS EN DB2 UDB

V. 7.2

SCRIPT DE CREACIÓN DE LA BASE DE DATOS EN DB2 UNIVERSAL DATABASE

Definición de Tablas:

```
CREATE TABLE D_MARCATRA (  
  CODMARCATRA    VARCHAR(15) NOT NULL,  
  PRIMARY KEY (CODMARCATRA)  
);
```

```
CREATE TABLE D_MARCAPRO (  
  CODMARCAPRO    CHAR(6) NOT NULL,  
  DESMARCAPRO    VARCHAR(18),  
  PRIMARY KEY (CODMARCAPRO)  
);
```

```
CREATE TABLE D_GerenteBanca (  
  CodGerenteBanca  CHAR(6) NOT NULL,  
  DesGerenteBanca  VARCHAR(25),  
  PRIMARY KEY (CodGerenteBanca)  
);
```

```
CREATE TABLE D_GerenteZona (  
  CodGerenteZona   CHAR(6) NOT NULL,  
  CodGerenteBanca  CHAR(6),  
  DesGerenteZona   VARCHAR(50) NOT NULL,  
  PRIMARY KEY (CodGerenteZona)  
);
```

```
CREATE TABLE D_JefeGrupo (  
  CodJefeGrupo     CHAR(6) NOT NULL,  
  CodGerenteZona   CHAR(6),  
  DesJefeGrupo     VARCHAR(50) NOT NULL,  
  PRIMARY KEY (CodJefeGrupo)  
);
```

```
CREATE TABLE D_Sector (  
  CodSector        VARCHAR(6) NOT NULL,  
  CodJefeGrupo     CHAR(6) NOT NULL,  
  DesSectorista    VARCHAR(50) NOT NULL,  
  PRIMARY KEY (CodSector)  
);
```

```
CREATE TABLE D_TipoSituaCasa (  
  CodSituaCasa     CHAR(5) NOT NULL,  
  DesSituaCasa     VARCHAR(20),  
  PRIMARY KEY (CodSituaCasa)  
);
```

```
CREATE TABLE D_TipoRenta (  
  CodRenta         SMALLINT NOT NULL,  
  DesRenta         VARCHAR(20),  
  PRIMARY KEY (CodRenta)  
);
```

```
CREATE TABLE D_MesesCli (  
  CodMesesCli     SMALLINT NOT NULL,  
  PRIMARY KEY (CodMesesCli)  
);
```

```
CREATE TABLE D_RangoMtoRent (  
    CodRangoMtoRent INTEGER NOT NULL,  
    DesRangoMtoRent VARCHAR(20),  
    MtoRentMin DECIMAL(16,2),  
    MtoRentMax DECIMAL(16,2),  
    PRIMARY KEY (CodRangoMtoRent)  
);  
  
CREATE TABLE XD_RangoMtoRentEsp (  
    CodRangoMtoRentEsp INTEGER NOT NULL,  
    DesRangoMtoRentEsp VARCHAR(20),  
    PRIMARY KEY (CodRangoMtoRentEsp)  
);  
  
CREATE TABLE D_Moneda (  
    CodMoneda SMALLINT NOT NULL,  
    DesMoneda VARCHAR(50),  
    PRIMARY KEY (CodMoneda)  
);  
  
CREATE TABLE XD_MonedaCuenta (  
    CodMonedaCuenta SMALLINT NOT NULL,  
    DesMonedaCuenta VARCHAR(50),  
    PRIMARY KEY (CodMonedaCuenta)  
);  
  
CREATE TABLE D_RangoEdad (  
    CodRangoEdad SMALLINT NOT NULL,  
    DesRangoEdad VARCHAR(30),  
    PRIMARY KEY (CodRangoEdad)  
);  
  
CREATE TABLE D_ClaRieSBS (  
    CodClaRieSBS CHAR(1) NOT NULL,  
    DesClaRieSBS VARCHAR(20),  
    PRIMARY KEY (CodClaRieSBS)  
);  
  
CREATE TABLE D_Banca (  
    CodBanca CHAR(1) NOT NULL,  
    DesBanca VARCHAR(30) NOT NULL,  
    PRIMARY KEY (CodBanca)  
);  
  
CREATE TABLE D_Segmento (  
    CodSegmento CHAR(2) NOT NULL,  
    CodBanca CHAR(1) NOT NULL,  
    DesSegmento VARCHAR(35),  
    PRIMARY KEY (CodSegmento)  
);  
  
CREATE TABLE D_DepartamentoAge (  
    CodDepartamentoAge CHAR(6) NOT NULL,  
    DesDepartamentoAge VARCHAR(25),  
    PRIMARY KEY (CodDepartamentoAge)  
);  
  
CREATE TABLE D_ProvinciaAge (  
    CodProvinciaAge CHAR(6) NOT NULL,  
    CodDepartamentoAge CHAR(6) NOT NULL,  
    DesProvinciaAge VARCHAR(25),  
    PRIMARY KEY (CodProvinciaAge)  
);
```

```
CREATE TABLE D_DistritoAge (  
    CodDistritoAge CHAR(6) NOT NULL,  
    CodProvinciaAge CHAR(6) NOT NULL,  
    DesDistritoAge VARCHAR(30),  
    PRIMARY KEY (CodDistritoAge)  
);  
  
CREATE TABLE D_Region (  
    CodRegion SMALLINT NOT NULL,  
    DesRegion VARCHAR(25),  
    PRIMARY KEY (CodRegion)  
);  
  
CREATE TABLE D_Agencia (  
    CodAgencia INTEGER NOT NULL,  
    CodRegion SMALLINT,  
    CodDistritoAge CHAR(6),  
    DesAgencia VARCHAR(50),  
    PRIMARY KEY (CodAgencia)  
);  
  
CREATE TABLE D_ClaRieBCP (  
    CodClaRieBCP CHAR(1) NOT NULL,  
    DesClaRieBCP VARCHAR(40),  
    PRIMARY KEY (CodClaRieBCP)  
);  
CREATE TABLE D_Profesion (  
    CodProfesion CHAR(3) NOT NULL,  
    DesProfesion VARCHAR(40),  
    PRIMARY KEY (CodProfesion)  
);  
  
CREATE TABLE D_CategoriaEco (  
    CodCategoriaEco VARCHAR(4) NOT NULL,  
    DesCategoriaEco VARCHAR(50),  
    PRIMARY KEY (CodCategoriaEco)  
);  
  
CREATE TABLE D_PaisNac (  
    CodPaisNac VARCHAR(3) NOT NULL,  
    DesPaisNac VARCHAR(40),  
    PRIMARY KEY (CodPaisNac)  
);  
  
CREATE TABLE D_Sexo (  
    CodSexo VARCHAR(1) NOT NULL,  
    DesSexo VARCHAR(15),  
    PRIMARY KEY (CodSexo)  
);  
  
CREATE TABLE D_EstadoCivil (  
    CodEstCivil VARCHAR(3) NOT NULL,  
    DesEstCivil VARCHAR(40),  
    PRIMARY KEY (CodEstCivil)  
);  
  
CREATE TABLE D_NivelEdu (  
    CodNivelEdu VARCHAR(3) NOT NULL,  
    DesNivelEdu VARCHAR(45),  
    PRIMARY KEY (CodNivelEdu)  
);
```

```
CREATE TABLE D_SituacionLab (
  CodSituacionLab VARCHAR(3) NOT NULL
  DesSituacionLab VARCHAR(35),
  PRIMARY KEY (CodSituacionLab)
);
```

```
CREATE TABLE D_TipoMonetario (
  CodTipoMonetario CHAR(1) NOT NULL,
  DesTipoMonetario VARCHAR(15),
  PRIMARY KEY (CodTipoMonetario)
);
```

```
CREATE TABLE D_Canal (
  CodCanal CHAR(1) NOT NULL,
  DesCanal VARCHAR(25),
  PRIMARY KEY (CodCanal)
);
```

```
CREATE TABLE D_Aplicativo (
  CodAplicativo CHAR(1) NOT NULL,
  DesAplicativo VARCHAR(20),
  PRIMARY KEY (CodAplicativo)
);
```

```
CREATE TABLE D_TipoCargoAbono (
  CodCargoAbono CHAR(1) NOT NULL,
  DesCargoAbono VARCHAR(15),
  PRIMARY KEY (CodCargoAbono)
);
```

```
CREATE TABLE D_TipoTra (
  CodTipoTra CHAR(7) NOT NULL,
  CodTipoMonetario CHAR(1),
  CodCanal CHAR(1),
  CodAplicativo CHAR(1),
  CodCargoAbono CHAR(1),
  DesTipoTra VARCHAR(50),
  PRIMARY KEY (CodTipoTra)
);
```

```
CREATE TABLE D_FamiliaProducto (
  CodFamiliaProducto CHAR(3) NOT NULL,
  DesFamiliaProducto VARCHAR(50),
  PRIMARY KEY (CodFamiliaProducto)
);
```

```
CREATE TABLE XD_SubsegmentoMes (
  CodSubsegmentoMes CHAR(3) NOT NULL,
  DesSubsegmentoMes DECIMAL(16,2),
  PRIMARY KEY (CodSubsegmentoMes)
);
```

```
CREATE TABLE D_RangoVinCli (
  CodRangoVinCli SMALLINT NOT NULL,
  DesRangoVinCli VARCHAR(20),
  NumVinCliMin INTEGER,
  NumVinCliMax INTEGER,
  PRIMARY KEY (CodRangoVinCli)
);
```

```
CREATE TABLE XD_SectorMes (
  CodSectorMes VARCHAR(6) NOT NULL,
  DesSectoristaMes VARCHAR(50) NOT NULL,
  PRIMARY KEY (CodSectorMes)
);
```

```

);
CREATE TABLE D_Anio (
    CodAnio      SMALLINT NOT NULL
    PRIMARY KEY (CodAnio)
);

CREATE TABLE D_Mes (
    CodMes      INTEGER NOT NULL
    CodAnio     SMALLINT,
    DesMes      VARCHAR(20),
    PRIMARY KEY (CodMes)
);

CREATE TABLE D_Subsegmento (
    CodSubsegmento CHAR(3) NOT NULL,
    CodSegmento   CHAR(2),
    DesSubsegmento VARCHAR(60),
    PRIMARY KEY (CodSubsegmento)
);

CREATE TABLE D_MesCilinia (
    CodMesCilinia INTEGER NOT NULL
    DesMesCilinia VARCHAR(20),
    PRIMARY KEY (CodMesCilinia)
);

CREATE TABLE D_TipoClienteBCP (
    CodTipoClienteBCP CHAR(2) NOT NULL,
    DesTipoClienteBCP VARCHAR(25),
    PRIMARY KEY (CodTipoClienteBCP)
);

CREATE TABLE XD_ClarieSBSPer (
    CodClarieSBSPer CHAR(1) NOT NULL,
    DesClarieSBSPer VARCHAR(20),
    PRIMARY KEY (CodClarieSBSPer)
);

CREATE TABLE XD_DistritoPer (
    CodDistritoPer CHAR(6) NOT NULL,
    DesDistritoPer VARCHAR(30),
    PRIMARY KEY (CodDistritoPer)
);

CREATE TABLE D_TipoIngreso (
    CodTipoIngreso CHAR(1) NOT NULL,
    DesTipoIngreso VARCHAR(20),
    PRIMARY KEY (CodTipoIngreso)
);

CREATE TABLE D_TipoDocumento (
    CodTipoDocumento CHAR(1) NOT NULL,
    DesTipoDocumento VARCHAR(50),
    PRIMARY KEY (CodTipoDocumento)
);

CREATE TABLE D_Persona (
    CodPersona      INTEGER NOT NULL,
    CodSituaCasa    CHAR(5),
    CodRangoEdad    SMALLINT,
    CodTipoClienteBCP CHAR(2),
    CodClarieSBSPer CHAR(1),

```

```

ApellidoPaterno    VARCHAR(25),
ApellidoMaterno    VARCHAR(25),
PrimerNombre      VARCHAR(25),
SegundoNombre     VARCHAR(25),
NumeroDocumento   CHAR(13),
FecIngreso        DATE,
CodTipoIngreso    CHAR(1),
CodTipoDocumento  CHAR(1),
CodEstCivil       VARCHAR(3),
CodSituacionLab   VARCHAR(3),
CodNivelEdu       VARCHAR(3),
CodSexo           VARCHAR(1),
CodPaisNac        VARCHAR(3),
CodCategoriaEco   VARCHAR(4),
CodProfesion      CHAR(3),
NbrFantasia       VARCHAR(40),
FecNacimiento     DATE,
FecConstitucion   DATE,
FecUltMod         DATE,
FecCaducidadEmp   DATE,
NumRuc            CHAR(10),
DigitoRUC         CHAR(1),
NbrComercial      VARCHAR(60),
CtdDependientes   INTEGER,
CodDistritoPer    CHAR(6),
DesDomicilio1     VARCHAR(100),
DesDomicilio2     VARCHAR(100),
MtoIngBrutoMes    DECIMAL(16,2),
MtoOtroIngMes     DECIMAL(16,2),
CodRenta          SMALLINT,
PRIMARY KEY (CodPersona)
),

CREATE TABLE xD_AgenciaCli (
  CodAgenciaCli    INTEGER NOT NULL,
  DesAgenciaCli    VARCHAR(20),
  PRIMARY KEY (CodAgenciaCli)
),

CREATE TABLE D_EstadoCliente (
  CodEstadoCliente SMALLINT NOT NULL,
  DesEstadoCliente VARCHAR(20),
  PRIMARY KEY (CodEstadoCliente)
),

CREATE TABLE D_Cliente (
  CodPersonaCli    INTEGER NOT NULL,
  CodSubsegmento   CHAR(3),
  CodMesesCli      SMALLINT,
  CodMesCiiInicia  INTEGER,
  CodPersona       INTEGER,
  CodAgenciaCli    INTEGER,
  CodSector        VARCHAR(6),
  FecInicio        DATE,
  CodEstadoCliente SMALLINT,
  PRIMARY KEY (CodPersonaCli)
),

CREATE TABLE F_Cliente (
  CodMes           INTEGER NOT NULL,
  CodPersonaCli    INTEGER NOT NULL,
  CODMARCAPRO     CHAR(6) NOT NULL,
  CODMARCATRA     VARCHAR(15) NOT NULL,
  CodSubsegmentoMes CHAR(3),
  CodRangoVinCli   SMALLINT,
  CodRangoMtoRent  INTEGER,
  CodRangoMtoRentEsp INTEGER,
  CodSectorMes     VARCHAR(6),
  CodClaRieBCP    CHAR(1),
  FecClaRieBCP    DATE,

```



```

CodClaRieSBS      CHAR(1),
FecClaRieSBS      DATE,
CtdCtaOpeAct      SMALLINT,
CtdCtaOpePas      SMALLINT,
CtdProAct         SMALLINT,
CtdProPas         SMALLINT,
MtoSaldoAct       DECIMAL(16,2),
MtoSaldoPas       DECIMAL(16,2),
MtoRentabilidad   DECIMAL(16,2),
MtoIngresosFin    DECIMAL(16,2),
MtoIngresosNoFin  DECIMAL(16,2),
MtoEgresosFin     DECIMAL(16,2),
MtoEgresosNoFin   DECIMAL(16,2),
MtoRentabilidadEsp DECIMAL(16,2),
MtoProvision      DECIMAL(16,2),
MtoDeudaSBS       DECIMAL(16,2),
MtoDeudaBCP       DECIMAL(16,2),
NumFactorVinCli   DECIMAL(16,2),
CtdCliente        INTEGER,
PRIMARY KEY (CodMes, CodPersonaCli)
),

CREATE TABLE XD_EstadoCuentaMes (
  CodEstadoMes     SMALLINT NOT NULL,
  DesEstadoMes     VARCHAR(20),
  PRIMARY KEY (CodEstadoMes)
),

CREATE TABLE D_EstadoCuenta (
  CodEstado        SMALLINT NOT NULL,
  DesEstado        VARCHAR(20),
  PRIMARY KEY (CodEstado)
),

CREATE TABLE D_TipoProducto (
  CodTipoProducto  CHAR(1) NOT NULL,
  DesTipoProducto  VARCHAR(30),
  PRIMARY KEY (CodTipoProducto)
),

CREATE TABLE D_Producto (
  CodProducto      SMALLINT NOT NULL,
  CodTipoProducto  CHAR(1),
  CodFamiliaProducto CHAR(3),
  DesProducto      VARCHAR(40),
  PRIMARY KEY (CodProducto)
),

CREATE TABLE D_RazonCierre (
  CodRazonCierre   SMALLINT NOT NULL,
  DesRazonCierre   VARCHAR(20),
  PRIMARY KEY (CodRazonCierre)
),

CREATE TABLE D_CtaOpe (
  CodCtaOpe        BIGINT NOT NULL,
  CodEstado        SMALLINT,
  CodPersonaCli    INTEGER,
  NumeroCtaOpe     CHAR(25),
  CodProducto      SMALLINT,
  CodMonedaCuenta  SMALLINT,
  FecApertura      DATE,
  MtoApertura      NUMERIC(16,2),
  MtoCierre        NUMERIC(16,2),
  CodRazonCierre   SMALLINT,

```

```

    FecCierre      DATE,
    PRIMARY KEY (CodCtaOpe)
);

```

```

CREATE TABLE F_CuentaOperacion (
    CodCtaOpe      BIGINT NOT NULL,
    CodMes         INTEGER NOT NULL,
    CodEstadoMes   SMALLINT,
    MtoSaldoPromedio  DECIMAL(16,2),
    MtoSaldoVigente  DECIMAL(16,2),
    MtoSaldoVencido  DECIMAL(16,2),
    MtoSaldoJudicial DECIMAL(16,2),
    MtoSaldoFinMes   DECIMAL(16,2),
    CtdDepositos    SMALLINT,
    CtdRetiros      SMALLINT,
    MtoDepositado   DECIMAL(16,2),
    MtoRetirado     DECIMAL(16,2),
    MtoIngresosFin  DECIMAL(16,2),
    MtoEgresosFin   DECIMAL(16,2),
    MtoIngresosNoFin DECIMAL(16,2),
    MtoEgresosNoFin DECIMAL(16,2),
    MtoMargenPeriodo DECIMAL(16,2),
    NumFactorVin    DECIMAL(16,2),
    CtdCtaOpe      BIGINT,
    PRIMARY KEY (CodCtaOpe, CodMes)
);

```

```

CREATE TABLE D_MonedaTran (
    CodMonedaTran  CHAR(4) NOT NULL,
    DesMonedaTran  CHAR(15),
    PRIMARY KEY (CodMonedaTran)
);

```

```

CREATE TABLE D_RangoMoneda (
    CodRangoMoneda CHAR(8) NOT NULL,
    CodMonedaTran  CHAR(4),
    DesRangoMoneda VARCHAR(30),
    MtoRangoIni    DECIMAL(16,2),
    MtoRangoFin    DECIMAL(16,2),
    PRIMARY KEY (CodRangoMoneda)
);

```

```

CREATE TABLE D_RangoMonedaDet (
    CodRangoMonedaDet CHAR(8) NOT NULL,
    CodRangoMoneda    CHAR(8),
    DesRangoMonedaDet VARCHAR(30),
    MtoRangoIniDet    DECIMAL(16,2),
    MtoRangoFinDet    DECIMAL(16,2),
    PRIMARY KEY (CodRangoMonedaDet)
);

```

```

CREATE TABLE F_Transaccion (
    CodMes         INTEGER NOT NULL,
    CodPersonaCli  INTEGER NOT NULL,
    CodCtaOpe      BIGINT NOT NULL,
    CodTipoTra     CHAR(7) NOT NULL,
    CodRangoMonedaDet CHAR(8) NOT NULL,
    MtoTransaccion DECIMAL(16,2),
    CtdTransaccion BIGINT,
    PRIMARY KEY (CodMes, CodPersonaCli, CodCtaOpe, CodTipoTra,
                CodRangoMonedaDet)
);

```

```

CREATE TABLE XD_ClarieSBSPerMes (
    CodClarieSBSPerMes CHAR(1) NOT NULL,
    DesClarieSBSPerMes VARCHAR(20),

```

```
PRIMARY KEY (CodClaRieSBSPerMes)
);
```

```
CREATE TABLE F_Persona (
    CodMes          INTEGER NOT NULL,
    CodPersona      INTEGER NOT NULL,
    CodClaRieSBSPerMes CHAR(1),
    MtoDeudaSBS     DECIMAL(16,2),
    CtdPersona      INTEGER,
    PRIMARY KEY (CodMes, CodPersona)
);
```

```
CREATE TABLE F_RetencionCliente (
    CodMes          INTEGER NOT NULL,
    CodPersonaCli   INTEGER NOT NULL,
    FlgClientelnicia SMALLINT,
    FlgClienteActivo SMALLINT,
    FlgClienteTermina SMALLINT,
    PRIMARY KEY (CodMes, CodPersonaCli)
);
```

```
CREATE TABLE F_CLIENTEPARETO (
    CodPersonaCli   INTEGER NOT NULL,
    MTORENTABILIDADPRO DECIMAL(16,2),
    MTORENTABILIDADACU DECIMAL(16,2),
    PRIMARY KEY (CodPersonaCli)
);
```

Definición de Vistas:

```
CREATE VIEW D_SectorMes (CodSectorMes, CodJefeGrupoMes, DesSectoristaMes) AS
SELECT a.CodSector, a.CodJefeGrupo, a.DesSectorista
FROM D_Sector a;
```

```
CREATE VIEW D_JefeGrupoMes (CodJefeGrupoMes, CodGerenteZonaMes, DesJefeGrupoMes) AS
SELECT a.CodJefeGrupo, a.CodGerenteZona, a.DesJefeGrupo
FROM D_JefeGrupo a;
```

```
CREATE VIEW D_GerenteZonaMes (CodGerenteZonaMes, CodGerenteBancaMes, DesGerenteZonaMes) AS
SELECT a.CodGerenteZona, a.CodGerenteBanca, a.DesGerenteZona
FROM D_GerenteZona a;
```

```
CREATE VIEW D_GerenteBancaMes (CodGerenteBancaMes, DesGerenteBancaMes) AS
SELECT a.CodGerenteBanca, a.DesGerenteBanca
FROM D_GerenteBanca a;
```

```
CREATE VIEW D_RangoMtoRentEsp (CodRangoMtoRentEsp, DesRangoMtoRentEsp) AS
SELECT a.CodRangoMtoRent, a.DesRangoMtoRent
FROM D_RangoMtoRent a;
```

```
CREATE VIEW D_MonedaCuenta (CodMonedaCuenta, DesMonedaCuenta) AS
SELECT a.CodMoneda, a.DesMoneda
FROM D_Moneda a;
```

```
CREATE VIEW D_ClaRieSBSPerMes (CodClaRieSBSPerMes, DesClaRieSBSPerMes) AS
SELECT a.CodClaRieSBS, a.DesClaRieSBS
FROM D_ClaRieSBS a;
```

```
CREATE VIEW D_ClaRieSBSPer (CodClaRieSBSPer, DesClaRieSBSPer) AS
SELECT a.CodClaRieSBS, a.DesClaRieSBS
FROM D_ClaRieSBS a;
```

```
CREATE VIEW D_SegmentoMes (CodSegmentoMes, DesSegmentoMes, CodBancaMes) AS
SELECT a.CodSegmento, a.DesSegmento, a.CodBanca
FROM D_Segmento a;
```

```
CREATE VIEW D_BancaMes (CodBancaMes, DesBancaMes) AS
SELECT a.CodBanca, a.DesBanca
```

```

FROM D_Banca a;

CREATE VIEW D_DistritoPer (CodDistritoPer, DesDistritoPer, CodProvinciaPer) AS
SELECT a.CodDistritoAge, a.DesDistritoAge, a.CodProvinciaAge
FROM D_DistritoAge a;

CREATE VIEW D_ProvinciaPer (CodProvinciaPer, CodDepartamentoPer, DesProvinciaPer) AS
SELECT a.CodProvinciaAge, a.CodDepartamentoAge, a.DesProvinciaAge
FROM D_ProvinciaAge a;

CREATE VIEW D_DepartamentoPer (CodDepartamentoPer, DesDepartamentoPer) AS
SELECT a.CodDepartamentoAge, a.DesDepartamentoAge
FROM D_DepartamentoAge a;

CREATE VIEW D_AgenciaMes (CodAgenciaMes, DesAgenciaMes) AS
SELECT a.CodAgencia, a.DesAgencia
FROM D_Agencia a;

CREATE VIEW D_AgenciaCli (CodAgenciaCli, DesAgenciaCli) AS
SELECT a.CodAgencia, a.DesAgencia
FROM D_Agencia a;

CREATE VIEW D_EstadoCuentaMes (CodEstadoMes, DesEstadoMes) AS
SELECT a.CodEstado, a.DesEstado
FROM D_EstadoCuenta a;

CREATE VIEW D_SubSegmentoMes (CodSubsegmentoMes, CodSegmentoMes, DesSubsegmentoMes) AS
SELECT a.CodSubsegmento, a.CodSegmento, a.DesSubsegmento
FROM D_Subsegmento a;

```

Definición de Llaves Foráneas:

```

ALTER TABLE D_GerenteZona
ADD FOREIGN KEY (CodGerenteBanca)
REFERENCES D_GerenteBanca;

ALTER TABLE D_JefeGrupo
ADD FOREIGN KEY (CodGerenteZona)
REFERENCES D_GerenteZona;

ALTER TABLE D_Sector
ADD FOREIGN KEY (CodJefeGrupo)
REFERENCES D_JefeGrupo;

ALTER TABLE D_Segmento
ADD FOREIGN KEY (CodBanca)
REFERENCES D_Banca;

ALTER TABLE D_ProvinciaAge
ADD FOREIGN KEY (CodDepartamentoAge)
REFERENCES D_DepartamentoAge;

ALTER TABLE D_DistritoAge
ADD FOREIGN KEY (CodProvinciaAge)
REFERENCES D_ProvinciaAge;

ALTER TABLE D_Agencia
ADD FOREIGN KEY (CodRegion)
REFERENCES D_Region;

ALTER TABLE D_Agencia
ADD FOREIGN KEY (CodDistritoAge)
REFERENCES D_DistritoAge;

ALTER TABLE D_TipoTra
ADD FOREIGN KEY (CodTipoMonetario)
REFERENCES D_TipoMonetario;

ALTER TABLE D_TipoTra
ADD FOREIGN KEY (CodCanal)
REFERENCES D_Canal;

```

```
ALTER TABLE D_TipoTra
  ADD FOREIGN KEY (CodAplicativo)
    REFERENCES D_Aplicativo;

ALTER TABLE D_TipoTra
  ADD FOREIGN KEY (CodCargoAbono)
    REFERENCES D_TipoCargoAbono;

ALTER TABLE D_Mes
  ADD FOREIGN KEY (CodAnio)
    REFERENCES D_Anio;

ALTER TABLE D_Subsegmento
  ADD FOREIGN KEY (CodSegmento)
    REFERENCES D_Segmento;

ALTER TABLE D_Persona
  ADD FOREIGN KEY (CodSituaCasa)
    REFERENCES D_TipoSituaCasa;

ALTER TABLE D_Persona
  ADD FOREIGN KEY (CodRenta)
    REFERENCES D_TipoRenta;

ALTER TABLE D_Persona
  ADD FOREIGN KEY (CodRangoEdad)
    REFERENCES D_RangoEdad;

ALTER TABLE D_Persona
  ADD FOREIGN KEY (CodTipoClienteBCP)
    REFERENCES D_TipoClienteBCP;

ALTER TABLE D_Persona
  ADD FOREIGN KEY (CodClaRieSBSPer)
    REFERENCES XD_ClaRieSBSPer;

ALTER TABLE D_Persona
  ADD FOREIGN KEY (CodDistritoPer)
    REFERENCES XD_DistritoPer;

ALTER TABLE D_Persona
  ADD FOREIGN KEY (CodTipoIngreso)
    REFERENCES D_TipoIngreso;

ALTER TABLE D_Persona
  ADD FOREIGN KEY (CodTipoDocumento)
    REFERENCES D_TipoDocumento;

ALTER TABLE D_Persona
  ADD FOREIGN KEY (CodEstCivil)
    REFERENCES D_EstadoCivil;

ALTER TABLE D_Persona
  ADD FOREIGN KEY (CodSituacionLab)
    REFERENCES D_SituacionLab;

ALTER TABLE D_Persona
  ADD FOREIGN KEY (CodNivelEdu)
    REFERENCES D_NivelEdu;

ALTER TABLE D_Persona
  ADD FOREIGN KEY (CodSexo)
    REFERENCES D_Sexo;

ALTER TABLE D_Persona
  ADD FOREIGN KEY (CodPaisNac)
    REFERENCES D_PaisNac;

ALTER TABLE D_Persona
  ADD FOREIGN KEY (CodCategoriaEco)
    REFERENCES D_CategoriaEco;
```

```

ALTER TABLE D_Persona
  ADD FOREIGN KEY (CodProfesion)
    REFERENCES D_Profesion;

ALTER TABLE D_Cliente
  ADD FOREIGN KEY (CodSubsegmento)
    REFERENCES D_Subsegmento;

ALTER TABLE D_Cliente
  ADD FOREIGN KEY (CodMesesCli)
    REFERENCES D_MesesCli;

ALTER TABLE D_Cliente
  ADD FOREIGN KEY (CodMesCliInicia)
    REFERENCES D_MesCliInicia;

ALTER TABLE D_Cliente
  ADD FOREIGN KEY (CodPersona)
    REFERENCES D_Persona;

ALTER TABLE D_Cliente
  ADD FOREIGN KEY (CodAgenciaCli)
    REFERENCES xD_AgenciaCli;

ALTER TABLE D_Cliente
  ADD FOREIGN KEY (CodSector)
    REFERENCES D_Sector;

ALTER TABLE D_Cliente
  ADD FOREIGN KEY (CodEstadoCliente)
    REFERENCES D_EstadoCliente;

ALTER TABLE F_Cliente
  ADD FOREIGN KEY (CODMARCAPRO)
    REFERENCES D_MARCAPRO;

ALTER TABLE F_Cliente
  ADD FOREIGN KEY (CODMARCATRA)
    REFERENCES D_MARCATRA;

ALTER TABLE F_Cliente
  ADD FOREIGN KEY (CodSubsegmentoMes)
    REFERENCES XD_SubsegmentoMes;

ALTER TABLE F_Cliente
  ADD FOREIGN KEY (CodRangoVinCli)
    REFERENCES D_RangoVinCli;

ALTER TABLE F_Cliente
  ADD FOREIGN KEY (CodRangoMtoRent)
    REFERENCES D_RangoMtoRent;

ALTER TABLE F_Cliente
  ADD FOREIGN KEY (CodRangoMtoRentEsp)
    REFERENCES XD_RangoMtoRentEsp;

ALTER TABLE F_Cliente
  ADD FOREIGN KEY (CodSectorMes)
    REFERENCES XD_SectorMes;

ALTER TABLE F_Cliente
  ADD FOREIGN KEY (CodMes)
    REFERENCES D_Mes;

ALTER TABLE F_Cliente
  ADD FOREIGN KEY (CodClaRieBCP)
    REFERENCES D_ClaRieBCP;

ALTER TABLE F_Cliente
  ADD FOREIGN KEY (CodClaRieSBS)
    REFERENCES D_ClaRieSBS;

```

```
ALTER TABLE F_Cliente
  ADD FOREIGN KEY (CodPersonaCli)
    REFERENCES D_Cliente;

ALTER TABLE D_Producto
  ADD FOREIGN KEY (CodTipoProducto)
    REFERENCES D_TipoProducto;

ALTER TABLE D_Producto
  ADD FOREIGN KEY (CodFamiliaProducto)
    REFERENCES D_FamiliaProducto;

ALTER TABLE D_CtaOpe
  ADD FOREIGN KEY (CodMonedaCuenta)
    REFERENCES XD_MonedaCuenta;

ALTER TABLE D_CtaOpe
  ADD FOREIGN KEY (CodEstado)
    REFERENCES D_EstadoCuenta;

ALTER TABLE D_CtaOpe
  ADD FOREIGN KEY (CodPersonaCli)
    REFERENCES D_Cliente;

ALTER TABLE D_CtaOpe
  ADD FOREIGN KEY (CodProducto)
    REFERENCES D_Producto;

ALTER TABLE D_CtaOpe
  ADD FOREIGN KEY (CodRazonCierre)
    REFERENCES D_RazonCierre;

ALTER TABLE F_CuentaOperacion
  ADD FOREIGN KEY (CodEstadoMes)
    REFERENCES XD_EstadoCuentaMes;

ALTER TABLE F_CuentaOperacion
  ADD FOREIGN KEY (CodMes)
    REFERENCES D_Mes;

ALTER TABLE F_CuentaOperacion
  ADD FOREIGN KEY (CodCtaOpe)
    REFERENCES D_CtaOpe;

ALTER TABLE D_RangoMoneda
  ADD FOREIGN KEY (CodMonedaTran)
    REFERENCES D_MonedaTran;

ALTER TABLE D_RangoMonedaDet
  ADD FOREIGN KEY (CodRangoMoneda)
    REFERENCES D_RangoMoneda;

ALTER TABLE F_Transaccion
  ADD FOREIGN KEY (CodCtaOpe)
    REFERENCES D_CtaOpe;

ALTER TABLE F_Transaccion
  ADD FOREIGN KEY (CodRangoMonedaDet)
    REFERENCES D_RangoMonedaDet;

ALTER TABLE F_Transaccion
  ADD FOREIGN KEY (CodTipoTra)
    REFERENCES D_TipoTra;

ALTER TABLE F_Transaccion
  ADD FOREIGN KEY (CodMes)
    REFERENCES D_Mes;

ALTER TABLE F_Transaccion
  ADD FOREIGN KEY (CodPersonaCli)
    REFERENCES D_Cliente;
```

```
ALTER TABLE F_Persona
  ADD FOREIGN KEY (CodPersona)
    REFERENCES D_Persona;

ALTER TABLE F_Persona
  ADD FOREIGN KEY (CodClaRieSBSPerMes)
    REFERENCES XD_ClaRieSBSPerMes;

ALTER TABLE F_Persona
  ADD FOREIGN KEY (CodMes)
    REFERENCES D_Mes;

ALTER TABLE F_RetencionCliente
  ADD FOREIGN KEY (CodPersonaCli)
    REFERENCES D_Cliente;

ALTER TABLE F_RetencionCliente
  ADD FOREIGN KEY (CodMes)
    REFERENCES D_Mes;

ALTER TABLE F_CLIENTEPARETO
  ADD FOREIGN KEY (CodPersonaCli)
    REFERENCES D_Cliente;
```


ANEXO D

ANÁLISIS COMPARATIVO DE MANEJADORES DE BASE DE DATOS Y HERRAMIENTAS OLAP DEL MERCADO

ANÁLISIS COMPARATIVO DE MANEJADORES DE BASE DE DATOS Y HERRAMIENTAS OLAP DEL MERCADO

1. Manejadores de Base de Datos

En el mundo del Data Warehouse y el soporte a las soluciones de Inteligencia de Negocios, no todos los sistemas manejadores de bases de datos son creados igual. El éxito con los productos de bases de datos en aplicaciones OLTP no necesariamente se ha de traducir en éxito con una aplicación para toma de decisiones.

El mercado de sistemas gestores de base de datos para este último entorno continúa consolidado alrededor de IBM, NCR y Oracle, con Microsoft en la parte baja de la pelea. La mayoría de empresas continúan viendo en UNIX a la plataforma natural para un data warehouse que pueda escalar en el tiempo, siempre que no exista una fuerte presencia de los equipos mainframe en una organización, cosa común en un banco mediano/grande. En estas pocas situaciones, usualmente DB2 para OS/390 (sistema operativo de los equipos mainframe de IBM) es preferido como la plataforma natural. Sobre Windows NT/2000 el foco ha permanecido en datamarts pequeños a medianos con poca complejidad. Poca atención le han dado los usuarios a la escalabilidad con las soluciones de Cluster en entornos Windows y existen pocas implementaciones de data warehouses medianos a grandes con esta alternativa.

Informix y Sybase han capturado poca atención en el mercado de data warehousing y sus respectivos productos de bases de datos han estado luchando para figurar en este entorno. Se podría decir que Informix ha tenido relativamente un mayor éxito al ganar cuentas importantes y muy grandes en diferentes latitudes, que le dieron alguna visibilidad y lo acercaron a los líderes del mercado, además de obtener un bien ganado reconocimiento por su innovación tecnológica y alto rendimiento, principalmente en entornos UNIX. En julio del 2001, IBM anunció la adquisición de la línea de productos Informix, incluyendo los productos de base de datos posicionados especialmente para entornos de data warehouse, comprometiéndose a darles continuidad en el tiempo con nuevas versiones y mejor tecnología. Sybase, con una estrategia confusa que ha involucrado diferentes productos de base de datos (Sybase IQ y Sybase Adaptive Server) y algunas adquisiciones en el camino, nunca presentó de forma clara su propuesta de valor sobre los principales jugadores en el mercado de data warehouse.

Para propósitos de este estudio, vamos a analizar con más detalle algunas características de los productos de IBM, Oracle, Microsoft y NCR.

1.1. IBM DB2 UDB

IBM está teniendo éxito con DB2 UDB (Universal Database) sobre equipos IBM RS/6000, el cual está ganando reconocimiento como un producto viable para implementaciones de inteligencia de negocios complejos. Desde 1997, empezando con la liberación del DB2 UDB v.5.1, IBM ha mantenido la presión competitiva en el mercado de bases de datos para inteligencia de negocios con una sucesión de nuevas versiones que incluyen v.5.2, v.6.1, v.7.1 y v.7.2 en los años siguientes. Estas versiones fortalecieron la arquitectura de DB2 UDB, entregando un conjunto de características y capacidades orientadas hacia el soporte de un data warehouse y el procesamiento de consultas complejas. DB2 UDB es calificado como un líder tecnológico por la fortaleza de su tecnología de optimización de consultas y el grado como DB2 puede escalar en arquitecturas SMP (Symmetric Multiprocessing) y DMPP (Distributed Memory Parallel Processors), tanto en Unix como en Windows NT/2000.

La liberación rápida de nuevas versiones de DB2 le ha permitido a IBM alcanzar e inclusive sobrepasar a otros proveedores en temas de características, pero a costa de la madurez del producto. Debido a que la mayoría de empresas no son capaces de actualizarse al mismo ritmo, estas versiones se encuentran poco probadas por la ausencia de una base instalada considerable y el tiempo que toma para proveer retroalimentación desde ambientes de producción real hacia los grupos de investigación y desarrollo de IBM. IBM ha trabajado mucho en reforzar su mensaje de marketing para el mercado de data warehouse corporativo, con buen resultado que lo ha llevado a mejorar su participación. Este incremento en el mercado aplica principalmente a los productos DB2 sobre plataformas RS/6000 (equipos fabricados por IBM), mientras que IBM continua luchando y ganando terreno en la batalla por ganar credibilidad sobre plataformas no-IBM (por ejemplo Sun Solaris).

1.2. Oracle 8i/9i

La principal fortaleza de Oracle es su excelente posicionamiento en el mercado de bases de datos y la vasta cantidad de socios de negocios, tanto de software como de hardware, con los que trabaja. Oracle 8i introdujo un número de

mejoras para datamarts en un modelo estrella y para la administración de datos (por ejemplo, las nuevas capacidades de particionamiento) que demostraron una maduración lenta pero continuada. La arquitectura shared-everything (todo-compartido) de Oracle brinda fortalezas en arquitecturas SMP de alto nivel (con más de 12 procesadores). No obstante, Oracle 8i todavía tiene retos en el manejo de grandes volúmenes de datos y en la complejidad de los modelos de datos (optimización de consultas). La mayoría de empresas usan técnicas radicales de denormalización (modelo estrella), tales como un uso intensivo de tablas agregadas, vistas materializadas y amplia indexación para asistir el servidor de base de datos de Oracle en la tarea de optimizar las consultas y reducir los retos en la administración de la carga de trabajo. Oracle 9i provee alguna evolución a las capacidades de Oracle, pero no es revolucionario y busca ponerse a la par de las capacidades de DB2 UDB o Teradata en este tema.

1.3. Microsoft SQL Server 2000

Microsoft ha gastado mucho en investigación y desarrollo para SQL Server 2000, focalizado en soportar aplicaciones OLAP. Sin embargo, las aplicaciones OLAP son sólo un ejemplo de muchos tipos de aplicaciones que son desarrolladas en un entorno de inteligencia de negocios y en implementaciones de data warehouse. Esto encaja bastante bien con la estrategia de Microsoft para ganar en el mercado intermedio con implementaciones rápidas de datamarts soportados por productos OLAP. Sin embargo, para implementaciones neutrales a la aplicación (modelos complejos de datos) y una base de datos grande (aproximadamente 200GB o más de datos) y con un amplio conjunto de aplicaciones a soportar, SQL Server tiene retos significativos. El reciente énfasis de Microsoft en las soluciones Cluster para ayudar con las implementaciones de data warehouse requiere un esfuerzo manual significativo y un monitoreo constante por parte de los administradores de base de datos. SQL Server no soporta particionamiento de tablas y no soportaría la propagación de tablas a través de múltiples nodos en Windows NT/2000. Para desplegar los datos en varios nodos, el administrador de base de datos necesitaría configurar tablas separadas como particiones y usar vistas para dar la apariencia de una sola tabla a los usuarios. Con un foco en el tema OLAP, el optimizador de consultas de SQL Server no está probado en el mundo real para la implementación de un modelo de datos complejo. Finalmente, el nivel de madurez de SQL Server se queda atrás del de otros productos de base de datos en el tema de soportar data warehouses. Aunque existen muchas implementaciones de datamarts con SQL

Server, éstas no llegan a probar en un ambiente real las exigencias a las que son expuestos los servidores de base de datos en implementaciones de data warehouse.

1.4. NCR Teradata

En 1997 hubo mucha confusión alrededor de la estrategia de data warehouse de NCR y NCR estuvo solícito para vender un producto de terceros (Oracle, Informix) en vez de Teradata, si eso era lo que el usuario solicitaba. La solución vino en la forma de instruir a la fuerza de ventas para que venda Teradata a toda costa y en incrementar la inversión en tecnología de Teradata. Claramente, la fortaleza de Teradata ha sido su probada capacidad en las implementaciones de data warehouses de gran escala y complejos (en las consultas y en el modelo de datos). Además, su rendimiento en consultas intensivas y complejas brinda un soporte a la concurrencia de consultas que es significativamente superior al de los demás competidores del mercado. Un empresa norteamericana que usa Teradata para administrar 1.3 Terabytes de datos detallados indica que soporta un volumen de 150,000 consultas por día contra el servidor de base de datos. Sin embargo, Teradata requiere un nivel de inversión que lo deja sólo al alcance de grandes corporaciones.

Podríamos resumir algunas de las fortalezas y debilidades de los productos descritos en el siguiente cuadro:

DBMS	Fortalezas	Debilidades
DB2 UDB EE/EEE (EE=Enterprise Edition, EEE=Extended Enterprise Edition)	<ul style="list-style-type: none"> • Buen desempeño sobre sistemas SMP y MPP. • Robusto optimizador de consultas. • Lider en cluster sobre Windows NT/2000. • Soporte para grandes datamarts y data warehouses. • Rico en características y capacidades. 	<ul style="list-style-type: none"> • Adolece de credibilidad sobre plataformas no-IBM. • No ha sido probado con implementaciones grandes y complejas, y con alta concurrencia de consultas. • Rápida liberación de versiones al mercado conlleva a una inmadurez del producto. • Administración.
Oracle 8i/9i	<ul style="list-style-type: none"> • Buen desempeño sobre sistemas SMP y NUMA (Non Uniform Memory Access). • Liderazgo en el mercado (market share) y en el posicionamiento en la mente de los compradores (mind 	<ul style="list-style-type: none"> • Falta de efectividad sobre plataformas DMPP. • Rendimiento afectado por alto volumen de consultas concurrentes. • Optimización de consultas para

DBMS	Fortalezas	Debilidades
	share). <ul style="list-style-type: none"> • Amplio soporte para herramientas y soluciones de inteligencia de negocios de terceros. • Soporte para grandes datamarts y data warehouses. 	modelos de datos complejos. <ul style="list-style-type: none"> • Particionamiento de Datos. • Administración. • No ha sido probado con implementaciones grandes y complejas.
Microsoft SQL Server	<ul style="list-style-type: none"> • Fuerte integración con Windows NT/2000. • Fácil de Administrar. • Presencia importante en el nivel bajo e intermedio del mercado de data warehouse. • Fuerte inversión en recursos de investigación y desarrollo. 	<ul style="list-style-type: none"> • No ofrece un buen soporte para entornos en cluster con Windows NT/2000. • Escalabilidad limitada en entornos multiusuario y en el manejo de grandes volúmenes de datos. • Limitaciones en la optimización de consultas para modelos de datos complejos. • Carece de particionamiento de datos.
NCR Teradata	<ul style="list-style-type: none"> • Alto rendimiento para grandes data warehouses y data marts. • Administración, reducido número de administradores de bases de datos es requerido. • Optimización de consultas con soporte para modelos de datos complejos. • Soporte para un gran número de consultas concurrentes. 	<ul style="list-style-type: none"> • Disponibilidad sobre entornos Windows y MPP. • Capacidad de ejecución de NCR como compañía. • Marketing y posicionamiento para un mercado más amplio. • Costo inicial de la solución más alto que el resto de proveedores. • Reducido soporte para aplicaciones y herramientas de inteligencia de negocios.

2. Herramientas de Consulta, Reporteo y Análisis en Línea (OLAP)

2.1. Hyperion Essbase OLAP Server

Hyperion Essbase es un motor de procesamiento analítico en línea (OLAP) optimizado para planeamiento de negocios, análisis y aplicaciones de reportes gerenciales tales como análisis de rentabilidad, presupuesto, pronósticos, planeamiento, consolidaciones financieras, y análisis de ventas. Es la base de la familia de productos Hyperion Essbase. Basado en una arquitectura multi-capas (multi-tier), Hyperion Essbase soporta accesos de lectura y escritura multiusuario; capacidad para el procesamiento de grandes volúmenes de datos; cálculos analíticos robustos y complejos; navegación flexible a través de los

datos; y tiempos de respuesta rápidos y consistentes. Este servidor puede escalar de un entorno mono-usuario hacia soluciones empresariales OLAP atendiendo a miles de usuarios con aplicaciones OLAP que manejen cientos de gigabytes de información. Además de su bases de datos multidimensional propia (MOLAP), Hyperion Essbase soporta base de datos OLAP relacionales (ROLAP) incluyendo IBM DB2, Oracle, Informix y Sybase. Esto permite que se acceda, como parte del análisis, a bases de datos relacionales muy grandes que almacenan el detalle de los datos, donde el precio a pagar está medido en el rendimiento. La razón principal para usar el almacenamiento multidimensional nativo de Hyperion Essbase en vez de una solución ROLAP es por que se espera que la base de datos multidimensional optimizada entregue mejor rendimiento cuando el número de usuarios concurrentes se incremente. Es posible combinar datos de ambos tipos de almacenamientos en una misma aplicación.

Essbase OLAP Server corre sobre plataformas Windows (Windows 95/98/NT/2000/XP), Sun Solaris, HP-UX, IBM AIX, AS/400, OS/390 y Linux. Sus principales competidores en el mercado son Microsoft SQL Server OLAP Services y Oracle Express (ahora conocido como Oracle OLAP Service). A la fecha tiene una mayor participación en el mercado, muy cercana a la suma de los dos competidores mencionados. Además una ventaja importante frente a Microsoft es su capacidad de ejecutarse sobre una variedad de plataformas adicionales a Windows. IBM ha realizado un acuerdo con Hyperion a través del cual comercializa una solución OLAP denominada DB2 OLAP Server basada en un 100% en el servidor OLAP de Hyperion y que se posiciona como la extensión OLAP de su servidor de base de datos relacional DB2 Universal Database (DB2 UDB). Las nuevas versiones de Hyperion Essbase llevan consigo nuevas versiones de DB2 OLAP Server.

Fortalezas:

- **Funcionalidad para aplicaciones OLAP:** Históricamente, Hyperion se ha focalizado en entregar suites de herramientas para aplicaciones financieras. Hyperion Essbase tiene más de 200 funciones analíticas y soporte para lectura/escritura con muchas aplicaciones analíticas. Tiene un lenguaje de script robusto para cálculos de dimensiones, procedurales y estadísticos, y este lenguaje de cálculos puede ser extendido a través de la creación de macros y funciones definidas por el usuario.

- Escalabilidad y rendimiento: Hyperion Essbase ofrece escalabilidad al proveer soporte para aplicaciones distribuidas a través de redes LAN y WAN, vinculando o navegando estas particiones como se desee. Las aplicaciones particionadas ofrecen la capacidad para paralelizar modelos, los cuales pueden ser cargados, consultados o calculados en paralelo sobre equipos con múltiples procesadores, y también para navegar entre aplicaciones con dimensiones y reglas de negocio diferentes. Las mejoras en rendimiento y escalabilidad en las versiones más recientes de Hyperion Essbase incluyen un incremento de la capacidad dimensional, mejoras en la concurrencia de las aplicaciones y de los usuarios, mejoras en las capacidades multi-cubo, entrada/salida optimizada y bases de datos residentes en memoria.
- Rendimiento en Consultas: El rendimiento de las consultas OLAP multidimensionales (MOLAP) en Essbase excede el de las soluciones OLAP relacionales (ROLAP). Los tiempos de respuesta rápidos vienen de los datos pre-calculados, agregados y técnicas optimizadas de indexamiento que maneja Hyperion Essbase. Por otro lado, las soluciones ROLAP usan SQL para acceder a las bases de datos relacionales, el cual es inherentemente lento, requiriendo joins complejos y lectura de grandes grupos de datos.
- Plataforma abierta: Hyperion Essbase soporta el despliegue de aplicaciones en Unix, Windows NT/2000/XP, S/390 y AS/400. El soporte para almacenamiento de datos en bases de datos relacionales incluye DB2, Informix, Oracle y Sybase. Mientras Hyperion Essbase ha tenido éxito en parte por focalizarse en las necesidades de los clientes y en desarrollar tecnología para un servidor OLAP que satisfaga las necesidades de procesamiento, ha apalancado Excel y Lotus 123 como la interfaz para el cliente. Hyperion tiene un extenso soporte con más de 50 herramientas front-end de terceros, tales como Crystal Reports, Business Objects, Brio o Cognos.
- Soporte para bases de datos OLAP Híbrido (HOLAP): En adición a su propia base de datos multidimensional, Hyperion Essbase soporta bases de datos OLAP relacional, incluyendo IBM DB2, Oracle, Informix y Sybase. Esta característica soporta el requerimiento de escalar para soportar bases de datos muy grandes, particularmente aquellas donde los datos en detalle van a ser accedidos con una finalidad OLAP. Esta estabilidad viene con un precio que es el hecho que el rendimiento se ve afectado.

Limitaciones:

- Depende de herramientas de terceros: Con el fin de complementar la familia de productos Hyperion Essbase, Hyperion se basa en herramientas de terceros para proporcionar una solución de inteligencia de negocios total. Si bien esto puede ser una fortaleza, al momento de integrar diferentes productos de diferentes proveedores no es claro que todas las interfaces están bien unidas o que todas las funcionalidades nativas se aprovechen al máximo. Hyperion hoy en día ofrece Hyperion Reports y Hyperion Analyzer como herramientas para los usuarios finales, pero la gran mayoría de clientes continúa usando interfaces de terceros.
- Dependencia de ODBC: Hyperion Essbase se basa en ODBC, en vez de los drivers nativos de bases de datos, para cargar datos desde las bases de datos relacionales. El acceso por ODBC puede tener implicancias en el rendimiento, el cual puede o no ser significativo, dependiendo de la frecuencia de la carga de datos y de los requerimientos del usuario en cuanto al tiempo de respuesta de las operaciones sobre datos relacionales.
- Proceso de carga costoso en tiempo: Hyperion Essbase, como cualquier servidor OLAP, requiere un proceso de carga de datos que puede ir desde minutos a muchas horas para ser terminado, dependiendo de la naturaleza de los datos y el número de agregaciones o pre-cálculos.

2.2. MicroStrategy 7i

MicroStrategy apoya el trabajo de las personas a cargo del análisis de información, brindando capacidades de inteligencia de negocios contra los datos almacenados en bases de datos relacionales a través de opciones avanzadas de consulta y reporte, análisis OLAP (ROLAP) y entrega proactiva de información. MicroStrategy proporciona conectores optimizados para casi todas las bases de datos relacionales más conocidas del mercado, como DB2, Oracle, Informix, Sybase, Teradata y SQLServer. Los comandos SQL que son generados automáticamente son diferentes y optimizados para cada una de las bases de datos. Incluye soporte para fuentes de datos heterogéneas a través de IBM DataJoiner, con lo cual se puede acceder y unir la información contenido en múltiples tablas de diferentes bases de datos. Los principales competidores de MicroStrategy son Brio, Business Objects y Cognos que proporcionan una funcionalidad y arquitectura distribuida, basada en componentes, muy similar a la que entrega MicroStrategy.

Fortalezas:

- **Arquitectura y Escalabilidad:** Con una arquitectura de tres capas, MicroStrategy es una suite de componentes integrados que distribuyen el procesamiento a través de múltiples servidores. MicroStrategy todavía lleva a cabo mucho del procesamiento en el servidor de base de datos, explotando la tecnología de la base de datos. Sin embargo, en los casos donde es óptimo ejecutar cálculos en una capa intermedia, el motor analítico de MicroStrategy lo habilita para manejar sus propias manipulaciones de datos. El resultado final es un sistema que toma ventaja de los mejores aspectos del procesamiento de una capa intermedia y del procesamiento de una base de datos relacional.
- **Clustering, Balance de carga y Prueba de fallos:** Dos niveles de clustering y prueba de fallos son posibles con MicroStrategy, con el fin de brindar escalabilidad, rendimiento y alta disponibilidad. El primero involucra usar el hardware o software disponible en el mercado para configurar un entorno de cluster con el Web Server o el Intelligence Server (motor analítico) de MicroStrategy. La segunda opción es emplear las capacidades propias de balanceo de carga y prueba de fallos del Web Server de MicroStrategy, el cual soporta entornos de cluster de múltiples nodos. Luego el Web Server puede distribuir los requerimientos de los usuarios a diferentes Intelligence Servers para balancear la carga de trabajo del servidor.
- **Interfaz HTML de MicroStrategy Web:** La interfaz HTML de MicroStrategy simplifica el despliegue de las aplicaciones de inteligencia de negocios al no requerir código adicional en el explorador de Internet y brinda soporte a todos los exploradores de Internet a través de cualquier "firewall" para cualquier usuario. Los productos Web ofrecen a los usuarios un gran porcentaje (del orden del 80 a 90 por ciento) de las capacidades de reporte disponibles en los productos "desktop". El uso de DHTML (Dynamic HTML) para navegar por los datos proporcionan una interfaz muy funcional y fácil de usar para los usuarios finales. MicroStrategy facilita que los usuarios ejecuten reportes contra los datos actuales, publiquen y se suscriban a los reportes y que los personalicen.
- **Motor analítico y funcionalidad:** MicroStrategy contiene más de 150 funciones matemáticas, estadísticas, OLAP y financieras, y puede llevar cabo una función cuando el servidor de base de datos relacional no cuenta con una función requerida.

- Facilidades para la administración centralizada: Desde una interfaz gráfica se puede controlar proyectos y acceso a los reportes para los usuarios, construir y realizar pruebas en un ambiente de desarrollo, llevar a cabo análisis de impacto, administrar la seguridad, manejar asistentes para la creación de reportes, etc.
- Paridad Web y Windows: MicroStrategy Web es tan bueno o mejor que los productos de la competencia en cuanto a la compatibilidad de su contraparte para Desktop (Windows). Aunque todavía hay un número de funciones presente en Desktop que los usuarios Web no pueden usar, los usuarios Web de MicroStrategy tienen la capacidad de diseñar reportes, una característica ausente en muchos productos de la competencia.

Limitaciones:

- Soporte para Repositorios Multidimensionales: MicroStrategy soporta sólo bases de datos relacionales como repositorios de datos y, a diferencia de sus competidores, no provee acceso a servidores OLAP de terceros. MicroStrategy no tiene un “cubo” propio local, lo cual significa que no puede ser usado fuera de línea para análisis.
- Rendimiento ROLAP: El rendimiento de las consultas sobre bases de datos relacionales típicamente es menor que aquel sobre servidores multidimensionales como Hyperion Essbase.
- Soporte Unix: Los productos de MicroStrategy no corren sobre Unix, un factor negativo para la escalabilidad de las aplicaciones, aunque el soporte para Unix está planeado.
- Acceso a Base de Datos Heterogéneas: MicroStrategy no tiene una capacidad nativa para acceder datos desde fuente heterogéneas. MicroStrategy soporta IBM Relational Connect para proporcionar esta capacidad.
- Balanceo de Carga Simplista: El proceso para distribuir requerimientos es simplista, basado estrictamente en cuántos procesos están activos sobre cada Intelligent Server. Por consiguiente, esta distribución no considera cuán ocupado está cada servidor.
- Migración de aplicaciones personalizadas: Aplicaciones personalizadas basadas en el API de MicroStrategy 6 no son soportadas en MicroStrategy 7.

2.3. Business Objects 2000

Business Objects provee una amplia suite de herramientas de inteligencia de negocios para consulta, reporte y procesamiento analítico en línea (OLAP) para una variedad de entornos (Web, Windows, dispositivos móviles) que habilita a los usuarios para acceder, analizar y compartir información dentro y fuera de una empresa.

Fortalezas:

- **Funcionalidad Web y Cliente/Servidor comparables:** Los productos web de Business Objects (WebIntelligence, 3tW Business Objects) tienen mucha de la funcionalidad de reporte y análisis del producto completo cliente/servidor.
- **Acceso a servidores OLAP:** WebIntelligence 2.7 ha mejorado el acceso y análisis de datos en Microsoft SQL Server 2000 Analysis Services e IBM DB2 OLAP Server (Hyperion Essbase), lo cual permite que los usuarios exploten las funcionalidades de estos servidores.
- **Interfaz de usuario OLAP y "Drill-Through" relacional:** WebIntelligence 2.7 tiene una interfaz OLAP mejorada basada en XML que permite transferir consultas entre diferentes fuentes e incorpora capacidades de "drill-through", que permiten ir directamente desde un reporte OLAP hacia un reporte relacional sin tener que utilizar otras herramientas.
- **Facilidad de Uso:** Desde una única interfaz (InfoView) los usuarios pueden inicializar todas sus funciones sin tener que ir y venir de una herramienta a otra. InfoView puede llamar automáticamente a BusinessObject o a WebIntelligence.
- **Interfaz de Usuario Funcional y Personalizable:** La interfaz puede ser personalizada por y para cada usuario, modificando la página inicial de acceso que puede contener cualquier documento personal o corporativo almacenado en el repositorio de BusinessObjects.

Debilidades:

- **Diferencias en el software cliente para Web y Windows:** Mientras que la funcionalidad de WebIntelligence y BusinessObjects es bastante equivalente, cada producto está optimizado para su entorno objetivo por lo que la implantación de una función en entornos diferentes con frecuencia resulta en diferentes modos de operación.

- Documentos en BusinessObjects y WebIntelligence: Debido a que los documentos en WebIntelligence son codificados y almacenados de manera diferente a como lo hace BusinessObjects, los documentos de uno no pueden ser manipulados por el otro.
- Reporte Corporativo: Las capacidades de reporte de BusinessObjects 2000 son algo limitadas y no ofrecen la misma facilidades como los productos de reporte corporativo que producen grandes volúmenes de reportes impresos.

2.4. Microsoft SQL Server 2000 Analysis Services

Los servicios de análisis de Microsoft SQL Server constituyen un servidor intermedio para análisis OLAP y minería de datos. Los servicios de análisis y el servicio PivotTable proporcionan la capacidad para diseñar, crear y administrar cubos y modelos de minería de datos desde un data warehouse y permiten acceso cliente a los datos OLAP y de minería. El servidor de análisis administra los datos y el servicio PivotTable trabaja con el servidor para proporcionar acceso cliente a los datos. Microsoft SQL Server 2000 Analysis Services corre sólo sobre plataformas Windows NT/2000. A pesar de ser cerrada en este sentido, es abierta en el sentido que puede acceder a otras bases de datos relacionales a través de OLE DB y permite que otras herramientas de usuario final puedan leer datos desde su repositorio multidimensional a través del estándar OLE DB for OLAP. Si bien los servicios de análisis son parte del producto SQL Server 2000, éstos funcionan de manera independiente al servidor de base de datos. El ponerlos juntos es una decisión de marketing, pero puede ser conveniente para los usuarios de SQL Server en caso requieran desarrollar aplicaciones de inteligencia de negocios.

Fortalezas:

- Rendimiento y Escalabilidad: Incluye características de escalabilidad y mejora del rendimiento tales como compresión de datos en los cubos, agregación de datos al momento y particionamiento. Extendiendo un cubo en varios servidores al dividirlo en particiones permite que se pueda resolver las consultas en paralelo y distribuir la carga en el procesamiento de datos.
- Facilidad de uso para los administradores: Analysis Services incluye asistentes y editores que simplifican el uso del sistema por los administradores de base de datos y otros profesionales de sistemas.
- Funcionalidad y Programación: Analysis Services provee componentes que pueden ser invocados desde aplicaciones personalizadas y facilitar el acceso

a sus objetos, como bases de datos, dimensiones,. Métricas, cubos, roles, modelos de minería de datos, etc.

- Cubos Partidos, Virtuales y Enlazados: Soporta el partir los cubos buscando mejorar el rendimiento en las consultas paralelas. Cubos múltiples pueden ser unidos en cubos virtuales sin necesidad de construir un nuevo cubo y mantenimiento aún el diseño óptimo de cada cubo. Un cubo puede ser almacenado en un servidor y referenciado como un vínculo en otro servidor, evitando la opción costosa de almacenar y mantener copias sobre diferentes servidores.
- Modelo de Seguridad: Cuenta con una seguridad robusta tanto para los administradores como para los usuarios finales. La seguridad de los administradores es controlada a través del grupo "OLAP Administrators" en Windows NT o Windows 2000. La seguridad de los usuarios finales especifica cuál usuario puede acceder a los datos y los tipos de operación que pueden llevar a cabo, incluyendo si tiene acceso de lectura o de lectura/escritura.
- Opciones de Almacenamiento Flexible (ROLAP, MOLAP y HOLAP): El producto de Microsoft es uno de los pocos de este tipo que ofrece los tres tipos de almacenamiento OLAP, brindando mayor flexibilidad para elegir el almacenamiento más adecuado.
- Minería de Datos: Dos algoritmos de minería de datos se incluyen con el Análisis Services: árbol de decisión y "clustering". Están diseñados para una minería de datos ligera. Para un trabajo de minería "serio" seguro los usuarios tendrían que recurrir a un producto especializado de Minería de Datos, como Knowledge Seeker o IBM Intelligent Miner.

Limitaciones:

- Soporte Unix: Microsoft SQL Server Análisis Services y la base de datos relacional no corren sobre plataformas Unix.
- Soporte para otras bases de datos: Microsoft proporciona drivers OLEDB para Oracle 8i/9i y para IBM DB2. En otros casos, los clientes deben conectarse a la bases de datos a través de OLEDB para ODBC a menos que obtengan un driver OLEDB nativo de un tercero. No hay soporte para acceder otros servidores OLAP, como Hyperion Essbase u Oracle Express.
- Minería de Datos: Si bien es una característica interesante el contar con capacidades de minería incluidas, ésta característica sólo proporciona dos modelos básicos para una minería "ligera". La interfaz de usuario no viene incluida, por lo que los usuarios necesitarán usar productos de terceros.

2.4. Oracle Express

Los productos Oracle Express soportan procesos de negocios analíticos, tales como modelamiento, predicción, análisis estadístico, análisis de ventas y distribución y análisis financiero. Express Server es un componente clave de las herramientas de inteligencia de negocios de Oracle, que incluyen Oracle Express Objects, una herramienta gráfica orientada a objetos para el desarrollo de aplicaciones de inteligencia de negocios personalizadas; Oracle Discoverer para ejecutar consultas ad hoc; y Oracle Reports para labores de generación de reportes. Express Server opera tanto en plataformas cliente/servidor como web. Es una herramienta híbrido OLAP (HOLAP) capaz de soportar tanto bases de datos relacionales y su propia base de datos multidimensional. Express Server es también la tecnología base que soporta las aplicaciones empaquetadas de inteligencia de negocios de Oracle: Oracle Financial Analyzer, Oracle Sales Analyzer y Oracle Demand Planning.

Fortalezas:

- Capacidades de Análisis y Predicción: Express Server tiene más de 300 funciones de cálculo, incluyendo varias funciones de correlación, predicción, tendencia y estadísticas para soportar numerosos tipos de análisis avanzado.
- Solución OLAP completa: Oracle Express Server y los productos de inteligencia de negocios de Oracle se presentan como una solución completa.
- Integración con las herramientas de inteligencia de negocios de Oracle: Integración con Oracle Reports para reporteo corporativo y Oracle Discoverer para consultas ad hoc y análisis.
- Soporta almacenamiento relacionales y multidimensionales: Express Server puede almacenar y administrar los datos en su propia base de datos multidimensional y también puede acceder y analizar datos guardados en bases de datos relacionales. Esta arquitectura híbrida OLAP proporciona una infraestructura escalable y robusta y aprovecha estos dos tipos de almacenamiento y análisis.
- Escalabilidad sobre plataformas diversas: Express Server corre sobre Windows NT, Sun Solaris, IBM AIX y otros sistemas. Además soporta la arquitectura de 64 bits en equipos Unix.
- Desarrollo de aplicaciones OLAP orientadas a objetos: Oracle Express Objects integra tecnología orientada a objetos con tecnología OLAP para

permitir que los desarrolladores creen aplicaciones multidimensionales, cliente/servidor y web.

- Soporte Web e Intranet: Un usuario puede exportar la información de Oracle Express hacia la web con las capacidades de consulta y análisis conocidas en la versión cliente/servidor de Oracle Express.

Limitaciones:

- Mezcla de tecnología nueva con tecnología antigua: La línea de productos Oracle Express aún retiene algo de la tecnología antigua que lo vio nacer.
- Integración con la base de datos Oracle: Oracle ha trabajado en integrar Express Server más de cerca con su servidor de base de datos relacionales y en las últimas versiones de su base de datos relacional se ha promocionado esta integración.
- Escritura en base de datos no-Oracle: Express permite al acceso a bases de datos de otros proveedores para leer datos, pero no es posible escribir sobre estas bases de datos.
- Migración de aplicaciones cliente/servidor hacia la web.

ANEXO E

MANUAL DE USUARIO DE LA APLICACIÓN OLAP DESARROLLADA

MANUAL DE USUARIO DE LA APLICACIÓN OLAP DESARROLLADA

El aplicativo desarrollado aprovecha la interfaz estándar de Microstrategy Web, la cual se inicia con la pagina por defecto del servidor que muestra todos los proyectos definidos (ver figura 1).

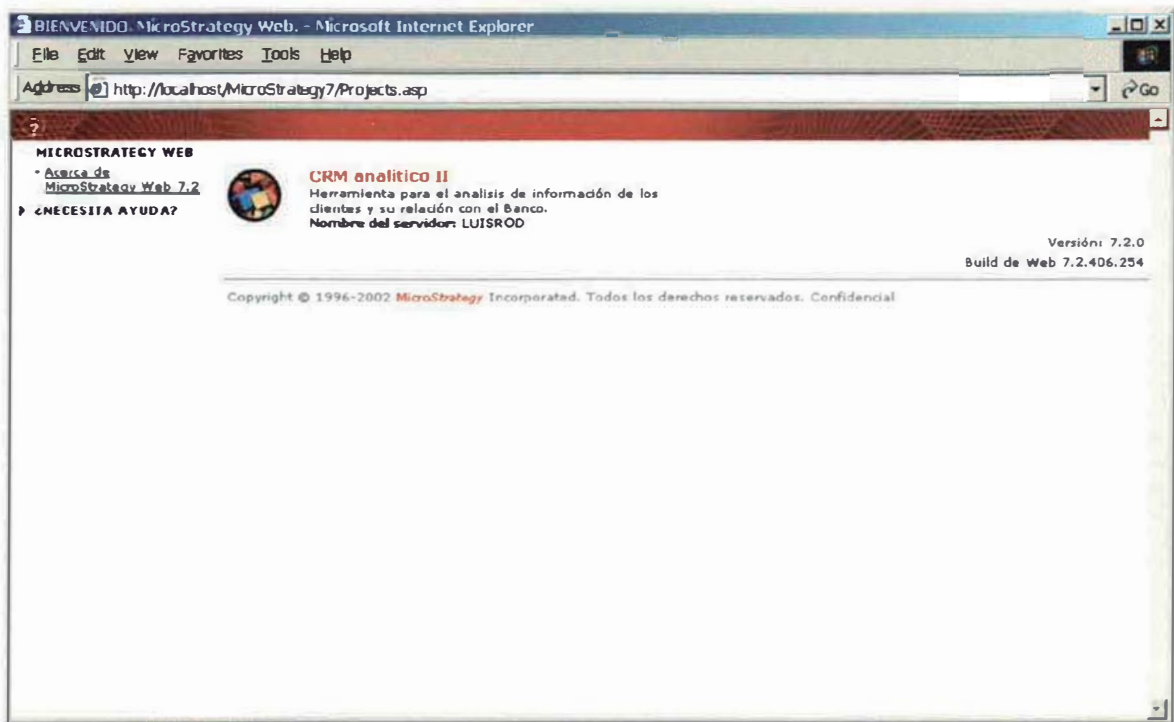


Figura 1. Pantalla Inicial

Luego de escoger el proyecto se presenta la pantalla de ingreso al sistema (ver figura 2).



Figura 2. Acceso al sistema

Microstrategy, al igual que la mayoría de herramientas del mercado, permite asociar características específicas, de seguridad por ejemplo, a un rol o a usuario en particular. En el caso del proyecto desarrollado no se han definido roles especiales y todos los usuarios tienen el mismo nivel de acceso.

Luego de ingresar el usuario y la contraseña se observa la pantalla con las opciones estándar para un usuario de consulta (ver figura 3).

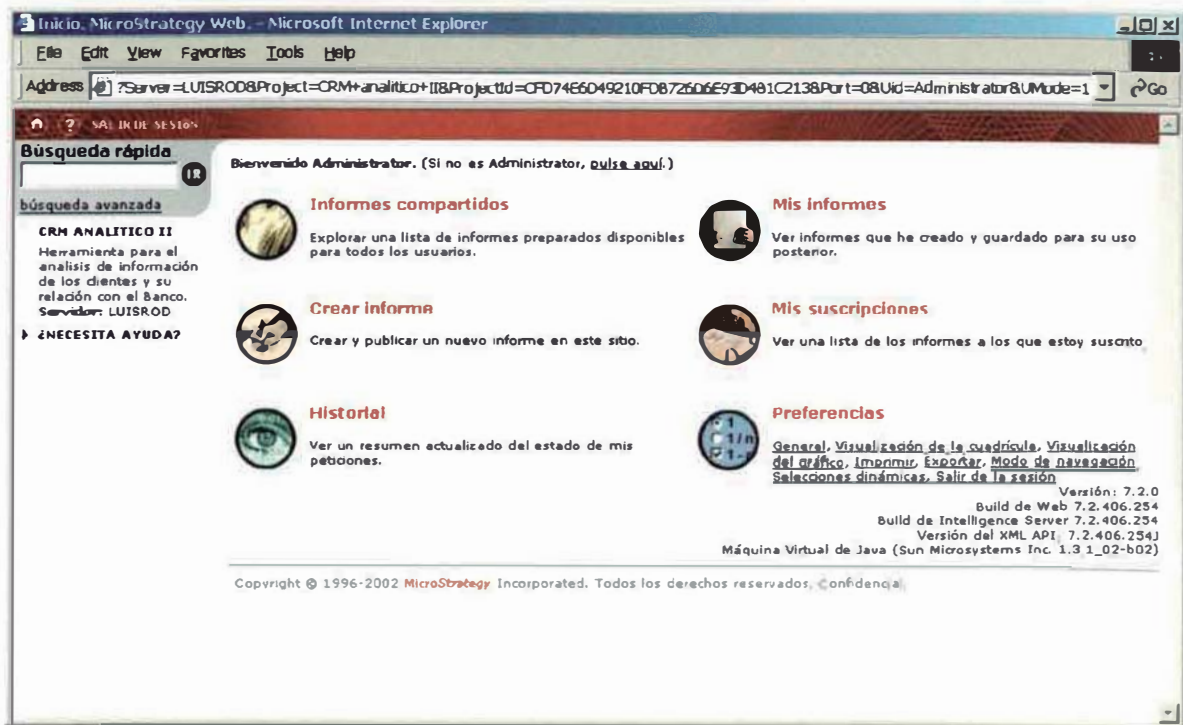


Figura 3. Menú principal

A continuación se describen las opciones de la pantalla inicial.

- Informes compartidos.- Carpeta en donde se almacenan los Informes definidos para todos los usuarios del proyecto.
- Mis informes.- Carpeta en donde quedan guardados las consultas definidas por el usuario de la sesión.
- Crear informe.- Opción que permite crear informes nuevos basándose en atributos, jerarquías e indicadores predefinidos.
- Mis suscripciones.- Carpeta que permite establecer enlaces a informes para que se actualicen con determinada periodicidad.
- Historial.- Carpeta que almacena enlaces a los informes enviados anteriormente.
- Preferencias.- Opción para la configuración de los parámetros y preferencias del usuario.

Haciendo sobre la opción de Mis informes se ingresa a las carpetas empleadas para el usuario del proyecto (ver figura 4).

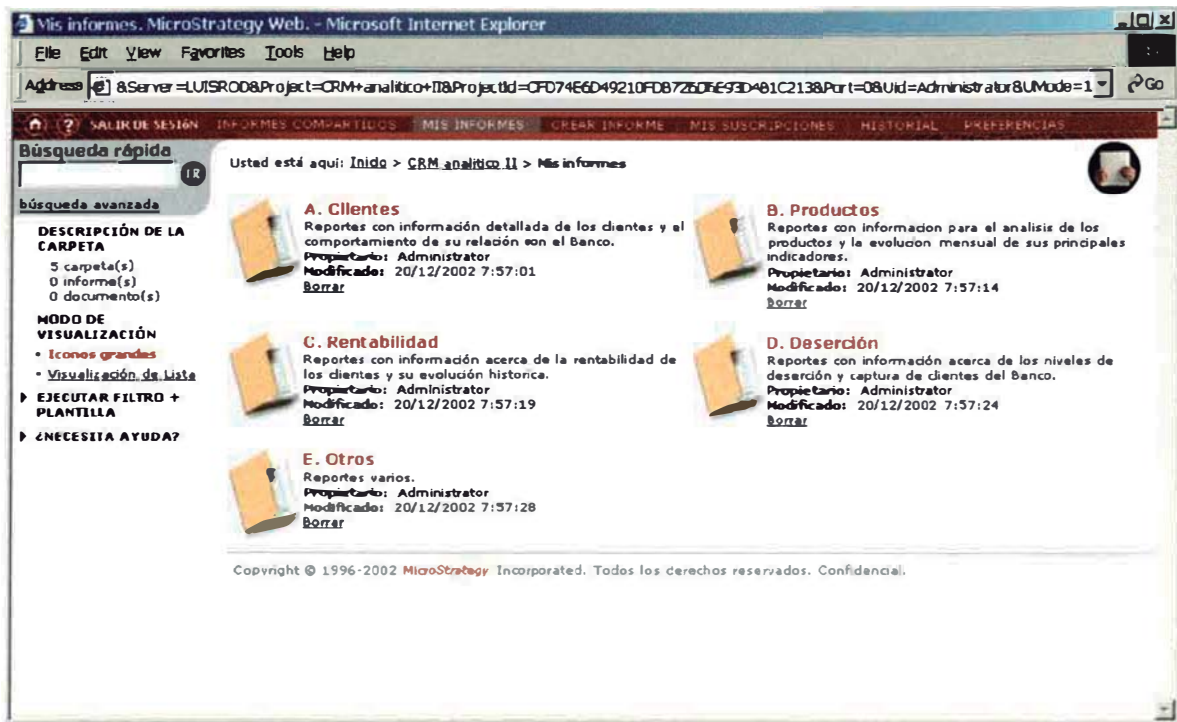


Figura 4. Carpetas del proyecto

Al escoger una carpeta se despliegan los informes disponibles. Por ejemplo en el caso de la carpeta de Rentabilidad se obtiene el siguiente listado (ver figura 5):

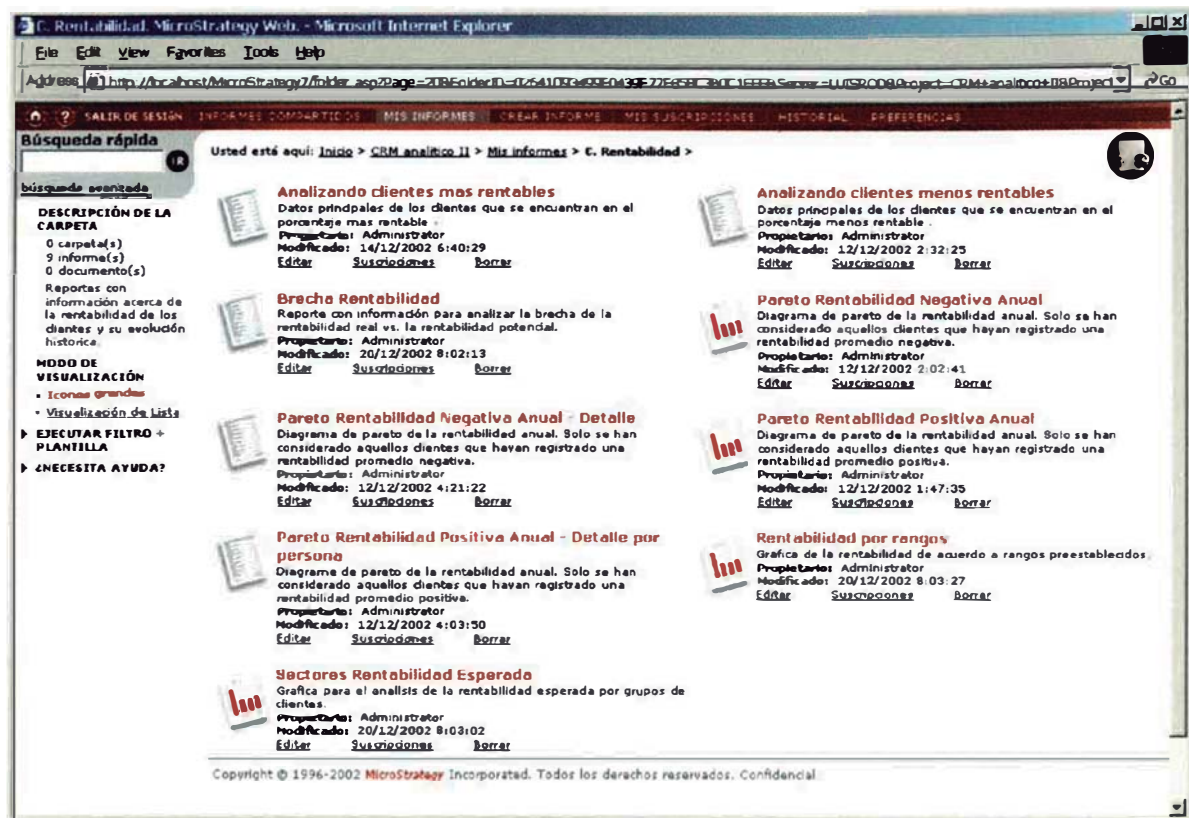


Figura 5. Informes de Rentabilidad

Tal como se observa, y al igual que en el caso de las carpetas, debajo de cada informe se observa una descripción del contenido del informe. Para seleccionar un informe basta con hacer clic sobre el nombre del informe y la consulta será enviada al motor de consultas de Microstrategy.

Al enviar una consulta al motor y mientras ésta se resuelve se presenta una pantalla como la siguiente, desde la cual se puede escoger la opción de enviar la consulta al historial o bien esperar hasta el final de la ejecución (ver figura 6).



Figura 6. Informe en ejecución

Esta pantalla le permite al usuario esperar por su respuesta o regresar a su análisis enviando nuevas consultas. Cuando la consulta esta resuelta se presentan los resultados en una nueva pantalla como la que se muestra después (ver figura 7):

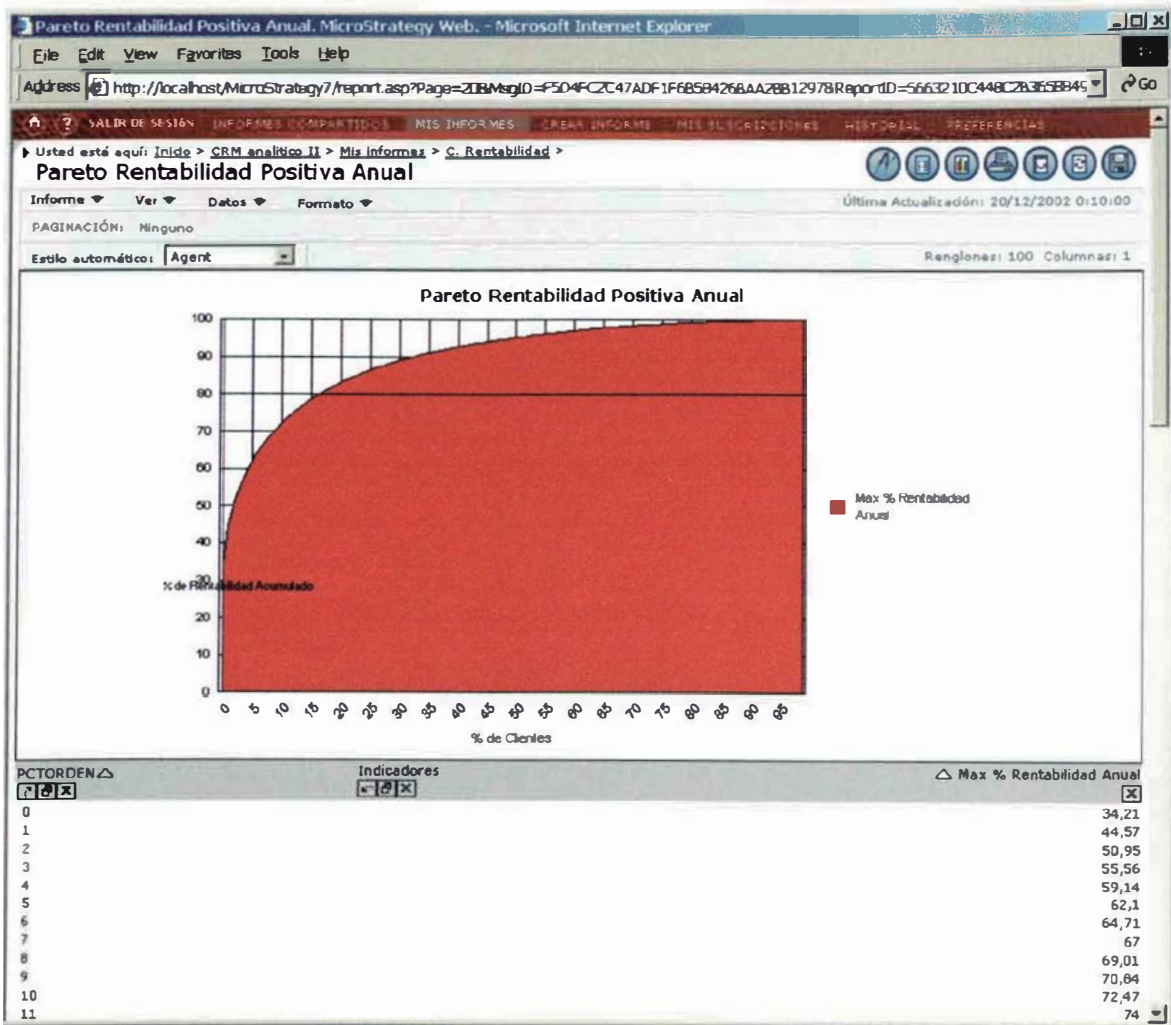


Figura 7. Ejemplo de un informe

A diferencia del informe anterior, existen otros que antes de ser enviados presentan una segunda pantalla pidiendo datos adicionales. Estos datos agregan mayor flexibilidad a cada consulta, permitiendo al usuario hacer mas especifica la consulta antes de enviarla a la base de datos. En el caso del informe "Rentabilidad por Rangos", por ejemplo, se solicita elegir los meses que se mostrarán en el informe final (ver figura 8).

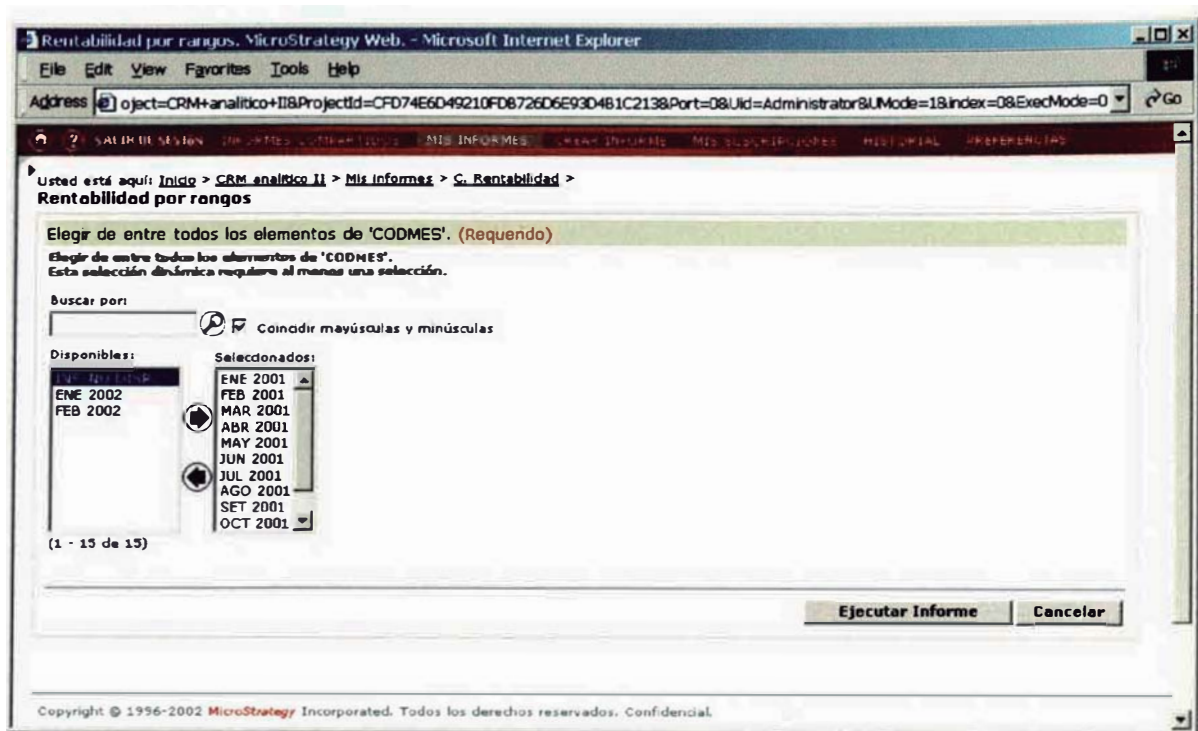


Figura 8. Solicitud de datos adicionales

Luego de seleccionar los meses se obtiene el resultado final (ver figura 9).

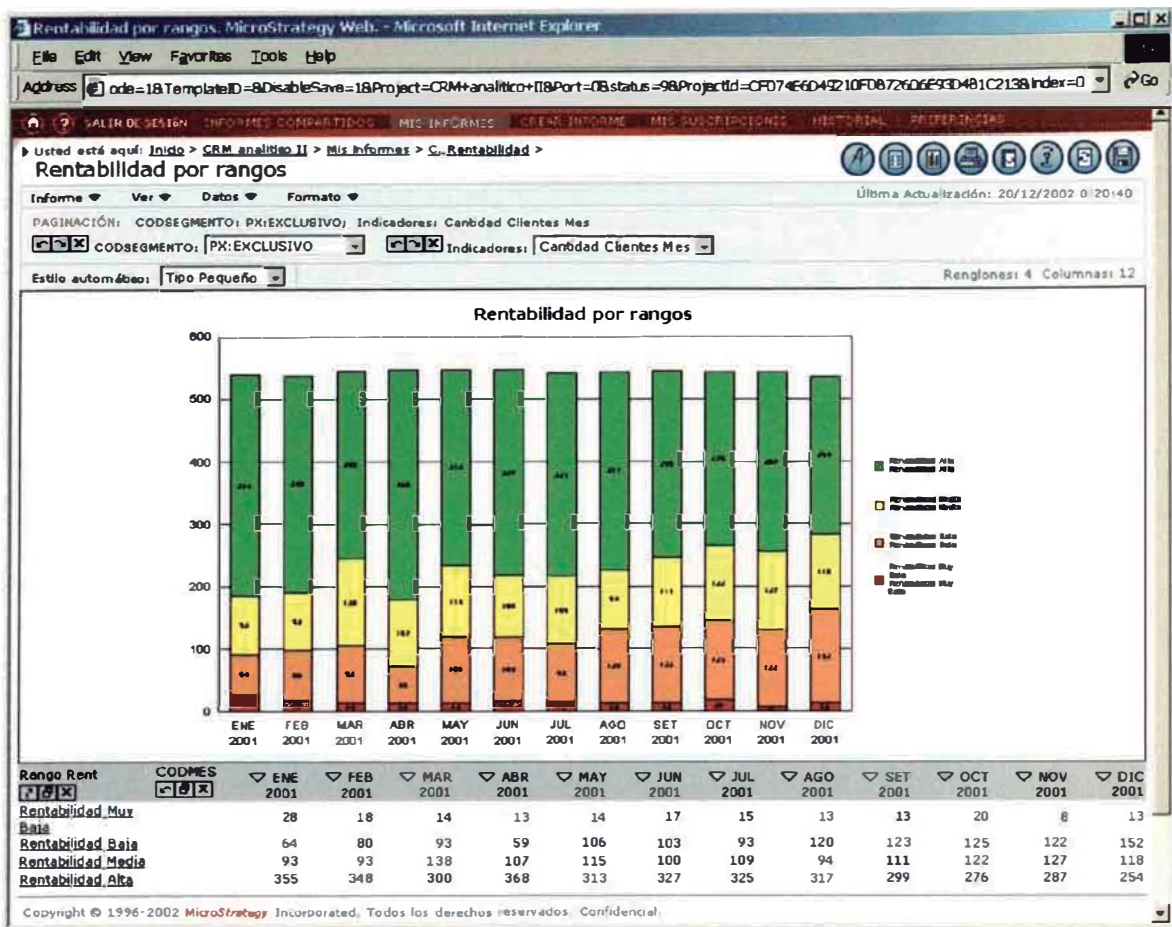


Figura 9. Solicitud de datos adicionales

Adicionalmente, están las funciones de navegación dentro del mismo informe. En la cabecera de cada consulta se encuentra el área de paginación, la cual contiene el listado de atributos posibles a considerar en el informe. Estos atributos pueden ser seleccionados en línea para modificar la selección actual (ver figura 10).

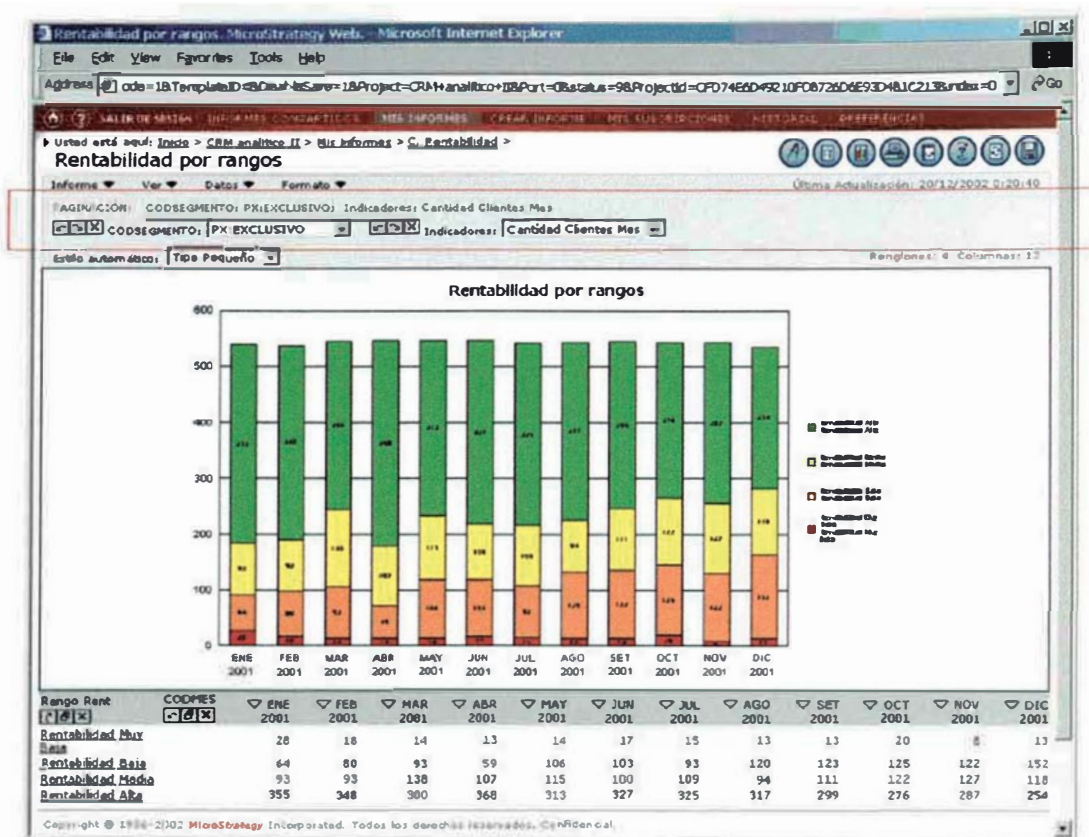


Figura 10. Paginación

De otro lado está el manejo de filas y columnas, las cuales pueden ser intercambiadas entre sí, enviadas a la paginación o eliminadas del informe, empleando el botón respectivo asociado a cada acción (flecha, recuadros, aspa) ubicado junto al nombre del atributo (ver figura 11).

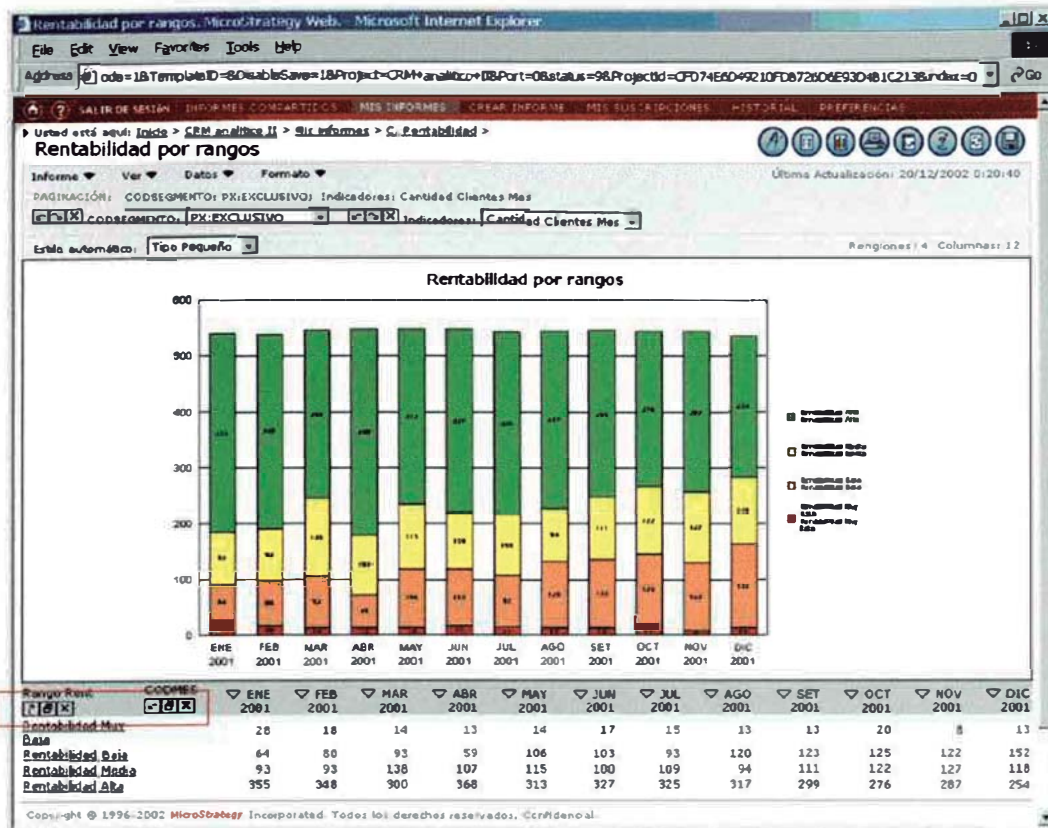


Figura 11. Manejo de filas y columnas

Finalmente, están las opciones de manejo del informe, que están ubicadas en la parte superior de la pantalla y permiten: editar el informe, ver los datos en modo reporte, ver los datos en modo gráfico, enviar a impresión, exportar a Microsoft Excel, refrescar los datos, volver a consultar los parámetros y grabar el informe (ver figura 12).

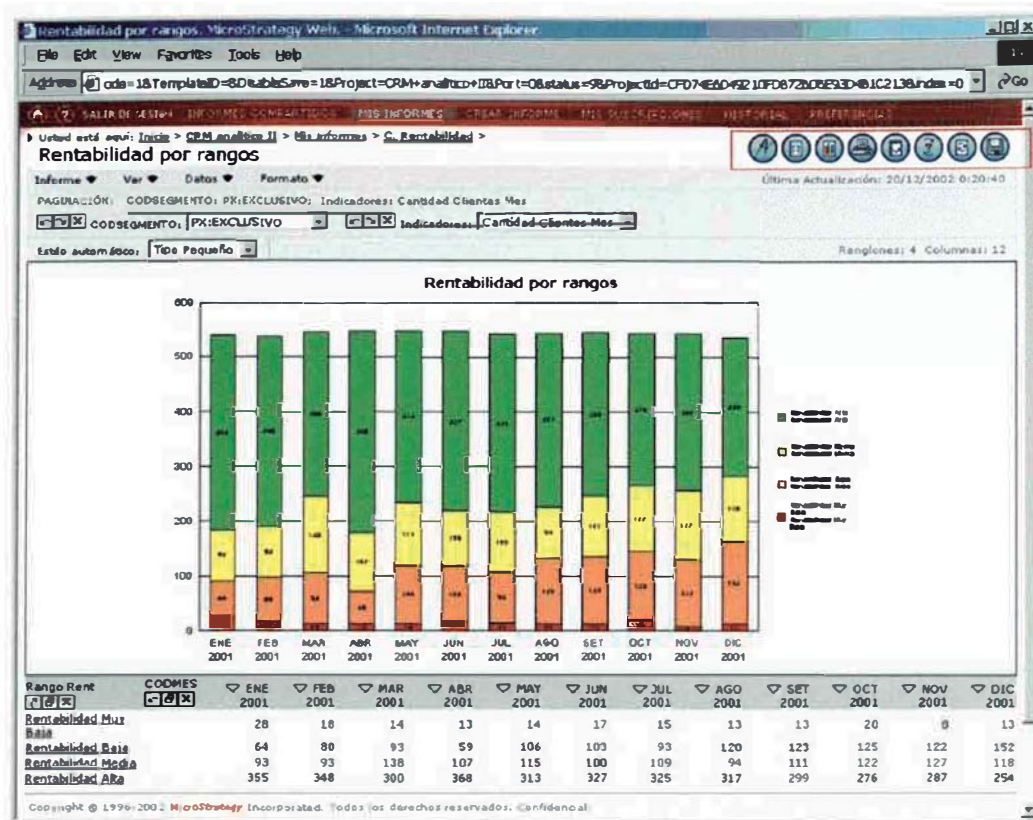


Figura 12. Menú del informe