

62
2ej

UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE INGENIERIA



**APLICACION DE UNA METODOLOGIA ESTRUCTURADA EN EL
DESARROLLO DE UN SISTEMA ADMINISTRADOR DE PENSIONES
EN UN ENTORNO MACRO-COMPUTACIONAL**

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

P R E S E N T A N
HUMBERTO RAMOS MEJIA
CUAUHTEMOC JAVIER SAUCEDO OREGEL

DIRECTOR DE TESIS
FIS. ADAN ZEPEDA GOROSTIZA

NOVIEMBRE 1993

**TESIS CON
FALLA DE ORIGEN**



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

APLICACION DE UNA METODOLOGIA ESTRUCTURADA EN EL DESARROLLO DE UN SISTEMA ADMINISTRADOR DE PENSIONES EN UN ENTORNO MACRO-COMPUTACIONAL

INDICE

CAPITULO 1. INTRODUCCION.....	13
1. CONTENIDO.....	14
2. DEFINICIONES PRELIMINARES.....	14
2.1. SIGNIFICADO DE "SOFTWARE".....	14
2.2. INGENIERIA DE SOFTWARE.....	14
2.3. SISTEMA DE INFORMACION.....	15
2.4. SISTEMA DE INFORMACION AUTOMATIZADO.....	16
3. LA CRISIS DE SOFTWARE NACIONAL.....	17
3.1. LA CRISIS DE SOFTWARE EN MEXICO.....	17
3.2. CAUSAS DEL PROBLEMA.....	18
3.3. CONSECUENCIAS.....	19
4. INTRODUCCION DE ESTE TRABAJO.....	20
4.1. OBJETIVO DEL TRABAJO.....	20
4.2. EL PRODUCTO A AUTOMATIZAR.....	20
4.3. INTRODUCCION A PENSIONES.....	20
4.4. OBJETIVOS DEL SISTEMA.....	21
4.5. ENTORNO DE DESARROLLO.....	21
CAPITULO 2. METODOLOGIAS DE DESARROLLO DE SISTEMAS.....	23
1. MODELOS DE DESARROLLO DE SISTEMAS.....	24
2. POSICIONAMIENTO DE LAS METODOLOGIAS DE DESARROLLO DE SISTEMAS.....	28
3. METODOLOGIAS PRINCIPALES.....	31

CAPITULO 3. METODOLOGIA UTILIZADA.....	57
1. CIRCUNSTANCIAS DEL DESARROLLO	58
2. METODOLOGIA UTILIZADA	59
3. SECUENCIA DE DESARROLLO.....	60
CAPITULO 4. ANALISIS PRELIMINAR Y PLANEACION.....	65
1. ANALISIS PRELIMINAR DE LA PROBLEMÁTICA BAJO UN ENTORNO REAL.....	66
1.1. LA ETAPA DEL ANALISIS PRELIMINAR DE SISTEMAS.....	66
1.2. LA INVESTIGACION Y ANALISIS INICIAL DE LA PROBLEMÁTICA PARA EL SISTEMA SPA.....	68
2. PLANEACION DEL SISTEMA A DESARROLLAR.....	70
2.1. LA FASE DE PLANEACION DE SISTEMAS.....	70
2.2. PLANEACION DEL SISTEMA DE PENSIONES ASEGURADAS	71
CAPITULO 5. ANALISIS PROFUNDO Y ESPECIFICACION DE REQUERIMIENTOS.....	75
1. LA FASE DE ANALISIS DE REQUERIMIENTOS DE SISTEMAS.....	76
2. DESCRIPCION FORMAL DEL SISTEMA (SPA) REQUERIDO.....	78
2.1. FLUJO DE OBTENCION DE LAS ESPECIFICACIONES PARA EL SISTEMA.....	78
2.1.1. REVISION Y COMPLEMENTACION DE LAS EXPECTATIVAS DEL USUARIO	78
2.1.2. PARAMETRIZACION DEL SISTEMA (SPA).....	80
2.2. LA ESPECIFICACION FORMAL DE REQUERIMIENTOS DEL SPA.....	81
CAPITULO 6. DISEÑO.....	93
1. INTRODUCCION.....	94
2. PRODUCTOS DE LA FASE DE DISEÑO	94
3. COMPONENTES DE DISEÑO	95
3.1. DISEÑO EXTERIOR	95
3.2. DISEÑO ARQUITECTONICO	96
3.3. DISEÑO DETALLADO	95

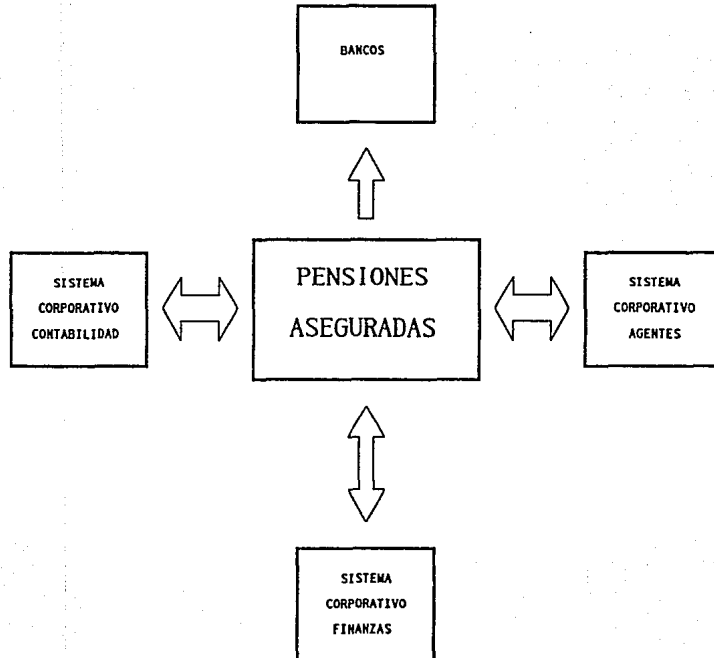
4. CONCEPTOS DE DISEÑO.....	98
4.1. REFINAMIENTO SUCESIVO	98
4.2. CONCURRENCIAS	98
4.3. MODULARIDAD	99
4.4. OCULTAMIENTO DE LA INFORMACION.....	99
4.5. ACOPLAMIENTO	100
4.6. COHESION	100
4.7. ABSTRACCION.....	100
4.8. USO DE HERRAMIENTAS.....	100
5. METODOLOGIA PARA EL DISEO DE LA BASE DE DATOS.....	101
5.1. OBTENCION DEL DICCIONARIO DE DATOS.....	101
5.2. IDENTIFICACION DE ENTIDADES Y RELACIONES.....	102
5.3. NORMALIZACION DE LAS ENTIDADES Y RELACIONES.....	103
5.4. AFINACION CON RESPECTO DE LAS FUNCIONES A SOPORTAR.	102
5.5. AFINACION CON RESPECTO A RESTRICCIONES DE LA INSTALACION.....	104
5.6. CODIFICACION DE LOS COMPONENTES DE LA BASE DE DATOS.	104
5.7. GENERACION DEL ESQUEMA FISICO DE LA BASE DE DATOS.	104
5.8. PRUEBAS DE GENERACION.	105
5.9. ADECUACIONES POSTERIORES.	105
6. SUMARIO	105
CAPITULO 7. CONSTRUCCION.....	109
1. DESCRIPCION DE LA ETAPA DE CONSTRUCCION	110
2. PROGRAMACION ESTRUCTURADA	110
3. PRINCIPIOS DE CONSTRUCCION.....	111
4. ESTANDARES DE CODIFICACION.....	112
5. AMBIENTE DE CONSTRUCCION EN LINEA	114
5.1. RELACION DE PROGRAMAS EN LINEA	116
6. AMBIENTE DE CONSTRUCCION BATCH.....	119
6.1. RELACION DE PROGRAMAS BATCH.....	120
7. SUMARIO DE HERRAMIENTAS UTILIZADAS.	122

8. PRUEBAS.....	1247
CAPITULO 8. OPERACION Y MANTENIMIENTO.....	125
1. DESCRIPCION DE LA FASE DE OPERACION Y MANTENIMIENTO.....	126
2. ADMINISTRACION DE CAMBIOS.....	127
3. COMO ACCESAR LA PARTE EN LINEA DEL SPA.....	128
4. FLUJO DE INFORMACION EN EL SISTEMA DE PENSIONES.....	129
CAPITULO 9. DESCRIPCION GENERAL DEL SPA. (OVERVIEW).....	135
1. DESCRIPCION GENERAL.....	139
1.1. MODULO DE CONTROL.....	139
1.2. MODULO DE ESTUDIOS.....	139
1.3. MODULO DE EMISION.....	140
1.4. MODULO DE COBRANZA.....	140
1.5. MODULO DE ADMINISTRACION.....	140
1.6. MODULO DE ESTADISTICA.....	140
2. PARTES DEL SISTEMA PENSIONES ASEGURADAS.....	141
3. PARTE EN LINEA.....	142
4. PARTE BATCH.....	142
5. SUBMODULOS DE LA PARTE EN LINEA.....	143
5.1. MENU PRINCIPAL.....	143
5.2. SUBMODULOS DEL MODULO DE CONTROL.....	145
. TABLA DE PROBABILIDAD DE MUERTE.....	145
. TABLA DE RECARGOS ADMINISTRATIVOS Y POR COMISIONES.....	147
. TABLA DE ESTADOS.....	148
. TABLA DE GIROS DE EMPRESA.....	149
. TABLA DE AGENTES.....	149
. TABLA DE OFICINAS.....	150
. TABLA DE FUNCIONES DEL SISTEMA.....	151
. CATALOGO GENERALIZADO DE TABLAS.....	152

. TABLA DE BANCOS.....	153
. TABLA DE IMPUESTOS.....	154
. TABLA DE ZONAS ECONOMICAS	155
. TABLA DE SUBSIDIOS	156
. TABLA DE CONCEPTOS	156
. TABLA DE FONDOS DE INVERSION	157
. TABLA DE USUARIOS DEL SISTEMA	158
5.3. SUBMODULOS DEL MODULO DE EMISION	159
. ALTA DE DATOS DEL CONTRATANTE.....	159
. ALTA DE DATOS DEL ASEGURADO	160
. ALTA DE DATOS DE LA PENSION.....	161
. ALTA DE PAGOS ESPECIALES.....	161
. ALTA DE COBERTURAS ADICIONALES	161
. ALTA DE BENEFICIARIOS	162
. CALCULO DE LA PENSION	162
5.4. SUBMODULOS DEL MODULO DE ADMINISTRACION	163
. SUBMODULO DE CONSULTAS.....	164
. SUBMODULO DE CAMBIOS	164
. COMPRA DE MONTOS ADICIONALES	165
. CAPTURA DE DEFUNCIONES	167
. CONSULTA DE POLIZAS GENERADAS POR DEFUNCION.....	167
. ACTUALIZACION DE POLIZAS GENERADAS POR DEFUNCION	168
. ENTREGA DE CHEQUES	168
. CAPTURA DE INDICES POR FONDO	168
. TIPO DE CAMBIO DEL DOLAR E INFLACION.....	168
. RETIROS DE RESERVA MATEMATICA.....	169
. FORMULA DE DIVIDENDOS	169
. RETIRO DE DIVIDENDOS	170
. CONSULTA DE LA BITACORA	170
6. SUBMODULOS DE LA PARTE BATCH	170
6.1. SUBMODULOS DEL CICLO EVENTUAL.....	170
. GENERACION DE TABLAS DE VALORES CONMUTADOS	170

6.2. SUBMODULOS DEL CICLO DIARIO	171
. ANALISIS DE LA CARTERA	172
. CALCULO DE DIVIDENDOS	172
. RECALCULO DE RENTAS	173
. CONSTITUCION DE RESERVAS	174
. PAGO DE RENTAS	174
. ACTUALIZACION DE MOVIMIENTOS	174
. GENERACION DE POLIZAS POR DEFUNCION	175
. RESCATE DE POLIZAS TERMINADAS POR DEFUNCION	175
. REPORTE A FINANZAS	176
. IMPRESION DE POLIZAS	176
. IMPRESION DE RECIBOS	176
6.3. SUBMODULOS DEL CICLO MENSUAL	176
. ANALISIS DE LA CARTERA	176
. REPORTE DE GASTOS ADMINISTRATIVOS	177
. CALCULO MENSUAL DE DIVIDENDOS	177
. REPORTE DE VALUACION DE RESERVAS	177
7. INTERFASES	177
7.1. INTERFASE A BANCOS	178
7.2. INTERFASE AL SISTEMA CORPORATIVO DE CONTABILIDAD	178
7.3. INTERFASE AL SISTEMA CORPORATIVO DE AGENTES	178
7.4. INTERFASE AL SISTEMA CORPORATIVO DE FINANZAS	179
 CAPITULO 10. CONCLUSIONES	 193
 BIBLIOGRAFIA	 195

SISTEMA DE PENSIONES ASEGURADAS INTERFASES





INTRODUCCION

1. CONTENIDO.

2. DEFINICIONES PRELIMINARES.

2.1. SIGNIFICADO DE "SOFTWARE".

2.2. INGENIERIA DE SOFTWARE.

2.3. SISTEMA DE INFORMACION.

2.4. SISTEMA DE INFORMACION AUTOMATIZADO.

3. LA CRISIS DE SOFTWARE NACIONAL.

3.1. LA CRISIS DE SOFTWARE EN MEXICO.

3.2. CAUSAS DEL PROBLEMA.

3.3. CONSECUENCIAS.

4. INTRODUCCION DE ESTE TRABAJO.

4.1. OBJETIVO DEL TRABAJO.

4.2. EL PRODUCTO A AUTOMATIZAR.

4.3. INTRODUCCION A PENSIONES.

4.4. OBJETIVOS DEL SISTEMA

4.5. ENTORNO DE DESARROLLO.

1. CONTENIDO

En la primera sección, se discuten los conceptos básicos que apoyan este trabajo, estos se refieren al significado de la palabra "Software", sus traducciones y connotaciones en nuestro idioma; así como se define a la Ingeniería de software, disciplina que se dedica a la fabricación de Software.

Dado que este trabajo, desarrolla un sistema de información automatizado. Se discute primero lo que se entiende por sistema de información, y se amplía su descripción al abordar el tema de sistema de información automatizado.

En la segunda sección, se discute la crisis del software en México, algunas de sus causas y consecuencias.

En la tercera sección, se mencionan los objetivos del trabajo, se introduce el caso práctico de estudio, se describe de forma sintética lo que es un producto de pensiones, se mencionan los objetivos del sistema que se desarrolló, y se describe el entorno de este.

2. DEFINICIONES PRELIMINARES

2.1. Significado de "Software".

El término "Software" no tiene una traducción directa al español, quizás la palabra que lo describe más fielmente es: "Lógica". Porque el Software es la entidad que contiene la lógica de un sistema de cómputo, de su funcionamiento, sus algoritmos y procedimientos.

Sin embargo, comúnmente se le traduce como "Sistema de programación", pero esta traducción no describe completamente su significado, ya que Software también se refiere a la documentación y en general a todo producto que contiene la Lógica del sistema.

Además, esta traducción reduce la disciplina a la parte puramente instrumental, es decir, de programación, y como se discutirá en este trabajo, el surgimiento de una Ingeniería de Software, deviene de reconocer que la problemática que se enfrenta, requiere de algo más que saber programar. Se requiere de una disciplina que se conforma de actividades organizativas, administrativas, analíticas, de diseño, entre otras.

2.2. Ingeniería de Software.

La ingeniería de Software es una rama de la computación, que nace de la necesidad de contar con una disciplina formal, seria y profesional, que se encargue de la fabricación de productos de software. La Ingeniería de Software se define como la aplicación de un conjunto estructurado de métodos y técnicas, para la fabricación de productos de Software, en tiempos y costos predecibles.

Al formalizar el proceso de fabricación de Software, es fundamental contar con la suficiente teoría informática, (conocimiento de Metodologías de Desarrollo, por ejemplo) procedimientos estructurados y disciplina, para desarrollar productos de software tan complejos como en la actualidad se requieren.

Actualmente, mas que nunca, es indispensable aplicar la Ingeniería de Software tanto en la construcción como en el mantenimiento de sistemas, solo así se podrá desarrollar con éxito los sistemas de información.

La aplicación de los principios, conceptos y metodologías de la ingeniería de software caracterizan al verdadero profesional en sistemas, que tan escaso se encuentra en la industria de la computación. y tan necesario es en el área macrocomputacional, donde la construcción, (podemos asegurar) de sistemas de información exige de los análisis mas elaborados y los diseños mas exactos.

La necesidad de capacitación y profesionalismo en sistemas y cómputo no es exclusivo para el desarrollo de sistemas, sino también se requiere en las áreas de "Soporte técnico", "Bases de datos", "Telecomunicaciones", en los usuarios y en general en todas las áreas que tienen que ver con el centro de cómputo.

El Software, o lógica del sistema de cómputo, tiene diversos campos de aplicaciones, entre ellos se encuentran:

- Control automático.
- Investigación y modelado (Simulación, Multimedia, etc.)
- Sistemas de información.
- Sistemas en tiempo real.

La aplicación que se presenta en este trabajo es la de Sistemas de Información.

La construcción de sistemas de Software es un "Proceso" equiparable a cualquier proceso industrial, con principios y controles similares. El sistema de Software terminado, es un "producto", como cualquier producto industrial, sujeto a principios de control de calidad, muestreo, y mercadotecnia.

Cuando se desarrolla un sistema de información por un equipo de personas, se presenta una complejidad organizativa, que no existen cuando participa un solo desarrollador.

Esta complejidad organizativa tiene que ver con los siguientes puntos:

- Partición del trabajo.
- Organización del proyecto.
- Entendimiento entre los participantes.
- Coordinación de esfuerzos.
- Medición de productos o resultados

La construcción de un sistema de información, de tiempo compartido, incrementa la complejidad del desarrollo ya que deben considerarse una serie de características de estos entornos, como los son la compartición de recursos, las concurrencias, el rendimiento y tiempo de respuesta del sistema.

2.3. Sistema de información.

El desarrollo de sistema de información, es una actividad, que en la actualidad ha tomado mucha relevancia dentro de las actividades que desarrolla toda empresa, institución y en general, toda organización social.

Ello se debe a que dichos sistemas, se han convertido en la clave competitiva y estratégica de las organizaciones. El sistema de información es el conjunto de normas y procedimientos que se establecen en un entorno dado, para organizar y administrar dicha información.

También se considera como parte de ese sistema, a la información misma, a los instrumentos que se emplean para su manipulación, y a las personas que los operan.

Las organizaciones se enfrentan a una competencia con empresas del mismo ramo. Donde el objetivo se centra en ganar el mercado, esto se logra ofreciendo productos y/o servicios de mejor calidad, con mayor atención al consumidor, con mejor distribución y comercialización.

Los sistemas de información brindan la posibilidad de incrementar y mejorar todas estas características, así como incrementar la cantidad, accesibilidad, confiabilidad y seguridad de la información, reduciendo su costo, tamaño, alargando su vida útil.

Un sistema mejora y estandariza los procesos que sufre esta información, minimiza la posibilidad de falla, deterioro y/o pérdida de la información. Al contar con un SI, se puede regenerar un resultado, con los mismos resultados a partir de los mismos datos de entrada. Esto hace a la información independiente de la persona, tiempo y herramientas que la procesan.

La labor del administrador o encargado del sistema de información es: dar el mejor uso a dicha información, garantizar su integridad, confiabilidad y seguridad. Para que dicha información se encuentre disponible con oportunidad, exactitud y legibilidad, al costo mas bajo posible.

2.4. Sistema de información automatizado

La automatización de un sistema de información, es posible auxiliándose por la computadora., y esta labor concentra gran parte del esfuerzo que se desarrolla en el mundo, con respecto a la computación, tanto Software como Hardware.

Resulta difícil, sino imposible, que una empresa pueda enfrentarse al reto de la actualidad, si es que no cuenta con un sistema de información formal, es decir, definido, escrito en uno o varios documentos, para que dicha empresa no sea tan vulnerable al cambio de personal, de instalaciones, o de estructura organizacional.

Para que una institución disponga de un sistema de información automatizado, se encuentran las siguientes opciones:

- Comprarlos, y que este se adapte a la perfección a la empresa. (Caso Ideal)
- Comprarlos y adaptarlos a la empresa.
- Hacerlos a la medida de la empresa.
- Mandarlos a Hacer a la medida de la empresa.
- Modificar o crecer alguno existente.

Cualquiera que sea el caso, el sistema tiene que ser elaborado alguna vez, por lo que esta labor no puede ser ignorada, por el contrario, debe afrontarse y conferírle su justo valor.

La opción que una organización decida para hacerse de un sistema automatizado de información, esta en función de sus necesidades, recursos y perspectivas.

Muchos empresarios optan por adaptar su organización al sistema, con las consecuentes pérdidas por realizar dicho cambio dentro de su organización.

3. LA CRISIS DE SOFTWARE NACIONAL.

3.1. La crisis del Software en México.

La Crisis del Software que se hizo patente en el mundo en los 60's, en México es en la actualidad, toda una realidad. Haciendo una revisión de los usuarios de los sistemas de software, podremos observar un común denominador: altos costos y poca satisfacción.

Primero revisemos el caso de los usuarios de baja escala, -para llamar de algún modo, a los que procesan poca información -, ellos usan paquetería de microcomputadora. El problema en este caso, es la mala utilización de la máquina y de los paquetes, debido todo ello, a la falta de capacitación formal al respecto.

El problema se complica, cuando las necesidades de manejo de información crecen. El usuario de baja escala, pasa a ser un usuario de media escala. Sigue desarrollando sus propias herramientas a través de paquetería de microcomputadora, pero ahora usa complejas hojas de cálculo, con una compleja red de archivos y directorios, bases de datos desordenadas, etc.

A la problemática del desconocimiento del equipo y del software, se añade el desconocimiento de como administrar la información. Es entonces cuando el Usuario de media escala, después de varios insabores, pérdidas de información y tiempo, decide encargar la labor, de instrumentar un sistema de información, a personal capacitado.

Para ello tiene dos opciones, encargar el desarrollo a un consultor externo a su organización, o encargarlo a un especialista interno. En cualquiera de los dos casos, el mencionado especialista resulta no serlo tanto, pues generalmente se trata de algún usuario de medio nivel que paso a la labor de especialista, sin mas capacitación que su experiencia; o por otro lado, se trata de un egresado de una escuela técnica de dudosa calidad, o un profesionista sin conocimientos de la ingeniería de software.

El usuario sufre entonces, un trauma mayor, pues después del excesivo tiempo de desarrollo, se le entrega un sistema mas complejo de lo que esperaba, muy lejano a sus requerimientos, y lleno de problemas. El usuario de medio nivel, ahora se encuentra ante un nuevo dilema: continuar soportando al sistema que se le entrego, o regresar a su antigua administración semimanual.

Pero ahora ya es demasiado tarde, la información que tiene registrada en el nuevo sistema no la puede migrar a su sistema anterior. A este usuario no le queda mas remedio, que seguir buscando en revistas especializadas en busca de algún sistema extranjero, que cubra, aunque sea en parte, sus necesidades.

El panorama en un centro de cómputo mayor, no es muy diferente, aunque si es mas grave, pues en este caso, existe mas de un usuario afectado.

En estos entornos, se han realizado mayores esfuerzos por corregir la situación, y ello se debe a que las pérdidas de información ocasionan mayores pérdidas que en los sistemas "Monousuario".

3.2. Causas del problema.

En primer lugar el problema es cognoscitivo, ¿ como realizar bien un sistema de software, si no se sabe como hacerlo ?, ni se tiene la capacidad para dominar el área de aplicación involucrada?.

Existen en el mercado, una cantidad impresionante de escuelas, que ofrecen educación en computación: Programación en distintos lenguajes, utilización de distintas paqueterías, licenciaturas con los nombres mas diversos, sin embargo, aún no existe ninguna Ingeniería de Software. A pesar de lo anterior, el problema no parece solucionarse, sino todo lo contrario. Dice un conocido refrán: "Mas peligroso que no saber, es saber a medias".

El problema entonces, no solo es la ausencia de conocimiento, sino, también la ausencia de metodologías estructuradas, que sean aplicables a un sistema real. La falta de formalidad y seriedad del mercado oferente de sistemas de información automatizada, explica la tendencia del empresario, de evitarse mas problemas y recurrir a supuestas soluciones ya creadas, sin embargo esto no resuelve su problemática organizativa, por el contrario, la complica.

Una opción muy socorrida, para intentar resolver los problemas de información, es la compra de sistemas prefabricados en el exterior. Esto fuerza una o ambas de las siguientes opciones: Adaptar la organización al sistema, adaptar el sistema a la organización. En ambos casos(aunque la primera es irreverentemente mala) el costo es muy alto, y este costo no se considera con suficiente precaución en el momento de tomar la decisión de la nueva adquisición.

Si se considera que estos mismos sistemas, causan problemas en sus países de origen, en el nuestro la problemática es mayor. Con la agravante de que los constructores no están disponibles con la oportunidad requerida, y ello fuerza a modificar - uno mismo - lo que no se entiende y dejarlo aun mas inentendible.

El desarrollo de sistemas, por tradición, en nuestro país ha venido siendo responsabilidad de personal no capacitado para tal labor, es frecuente encontrar que los niveles medios y superiores de una unidad funcional de informática, lo ocupan gente sin estudios, o con estudios que no son de computación. La construcción de los sistemas, lo llevan a cabo personas sin capacitación, egresados de escuelas técnicas de dudosa calidad, en su caso, profesionistas de la computación que tienen escasos conocimientos sobre la ingeniería de software.

Gran cantidad de sistema de software son importados, se usan en muy baja proporción, con problemas y pérdidas de información, no por que sean malos, sino porque son rígidos y no se adaptan a las necesidades de la organización.

Otro problema muy importante es el económico. El software se encuentra muy devaluado debido a la poca formalidad de la gente de la industria, y a la economía informal del software. Por su parte la mayoría de los contratos de los sistemas "hechos a la medida", tienen lagunas que son aprovechadas únicamente para sacar ventaja entre las organizaciones involucradas.

Así como es de todos conocido el mal de la duplicidad ilegal, en paquetería para la plataforma microcomputacional, tan extendido en nuestro país, que acarrea consecuencias muy perjudiciales para la gente dedicada a desarrollar sistemas en este rubro, pues se subestima y degrada el valor de la creación de software, de lo que la gente esta poco dispuesta a pagar lo justo por el sistema "hecho a la medida".

Para el entorno macrocomputacional, sucede algo análogo, las organizaciones dejan la contratación de un sistema "hecho a la medida", como ultima opción, prefieren reutilizar o duplicar un sistema existente

aunque la naturaleza y la plataforma tecnológica sea diferente, o los costos de la adaptación sean altos, se busca ese índice inicial menor, sin considerar la riqueza y experiencia que puede dejar el desarrollo de un nuevo sistema.

3.3. Consecuencias.

El software es Lógica. La importación de lógica es una debilidad crítica y estratégica nacional. La dominación de un país sobre otro, no solo es económica, sino también tecnológica y cultural. La cantidad de recursos que se emplean en tratar de hacer funcionar los Sistemas importados, es impresionante, y la cantidad de satisfacción es muy pobre.

En México no se ha entendido el lugar que juega la industria de la Lógica (software) en la sociedad, esto, muestra una falta de cultura informática grave, que trae consecuencias tangibles, que podrían ser irreversibles.

En la actualidad los activos tanto de las compañías como de las naciones, ya no se expresan por la cantidad de lingotes de oro u otro metal precioso en sus arcas. Se expresa con información. La información ha pasado a ser un activo, y se registran grandes pérdidas de este, con sistemas que pierden la información.

La asignación de recursos necesarios para las campañas electorales, así como la detección de necesidades comunitarias, las mismas elecciones, se llevan a cabo en computadoras, fallas en ellas provocan grandes pérdidas políticas. Debido a fallas en el sistema de software, se han registrado grandes pérdidas financieras, se han perdido elecciones, y hasta se han registrado muertes humanas.

La administración de los bienes tanto de una empresa como de la nación, se registran en computadoras. Y los sistemas de control de robots en la industria, de sistemas bancarios, de telecomunicaciones, de transportes, etc., son totalmente controlados por computadoras (generalmente macrocomputadoras).

El desarrollo de un sistema lógico, que dirija a un computador o a un conjunto de computadores, no es algo que debiese seguir ejecutando un artista, un ente individual, un inspirado, o una persona que no encontró ocupación sino en la computación.

Se cierra una y otra vez el círculo vicioso, lagunas de conocimiento computacional en nuestros profesionistas, sistemas no productivos, equipos de computo desaprovechados, escasez de servicios profesionales, desconfianza, organizaciones confundidas, contratación de despachos y sistemas extranjeros.. De tal forma que entre menos participen los profesionistas nacionales en los proyectos, mientras mas se les rehuya, menos madurez habrá en ellos, mas retraso, mas emigración, menos soluciones a los problemas siempre presentes

4. INTRODUCCION A ESTE TRABAJO.

4.1 Objetivo del trabajo.

Este trabajo es el resultado, de la utilización de una metodología estructurada para el desarrollo de un sistema real, en un entorno Multiusuarios, Multiprocesos, y MultiDesarrolladores, bajo una plataforma Macro-computacional (equipo Mainframe)

Este trabajo presenta, desde un punto de vista Ingenieril y práctico los "procesos" y "fases" involucradas antes y después de la construcción de un sistema de información automatizado, basado en el uso de metodologías formales de desarrollo para obtener el "producto" sistematizado y operable.

4.2. El producto a Automatizar.

Nuestro caso de estudio comienza, cuando una compañía equis de seguros, solicita al despacho consultor, en el que laboran los que presentan esta trabajo, un sistema para soportar la administración de un nuevo producto financiero.

El producto es un plan de Pensiones; producto nuevo en el mercado nacional, debido a que su venta estaba restringida a organismos gubernamentales como el IMSS. En 1990 el producto se libera para su venta a instituciones privadas. Las compañías aseguradoras toman la delantera en la preparación de planes de pensiones, para solicitar la aprobación de sus producto a la comisión nacional de compañías de seguros e instituciones financieras.

El producto de pensionamiento es de gran importancia, ya que en países donde se tiene mayor experiencia en este ramo como los Estados Unidos o Chile, existen compañías que exclusivamente se dedican a las pensiones, y son conocidas como "Pensionadoras".

Por convención, en el transcurso de esta trabajo, nos referiremos a este producto financiero como PPA (Producto de pensiones aseguradas) o como "Producto de Pensiones" de forma indistinta.

4.3. Introducción a Pensiones.

Un plan de pensiones tiene algunas complejidades técnicas. La actuaría es la disciplina que lo estudia y elabora, por ello en este trabajo se habla de varias conceptos actuariales, mas no se profundizará demasiado en ello, pues no es el objetivo de este trabajo.

A continuación se describe de forma sintética, lo que se refiere a un plan de pensiones:

Un plan de pensiones es un servicio que ofrece una institución, quien se compromete a pagar a la persona que contrata el plan, una serie de pensiones o rentas, a cambio de que esta persona constituya una reserva económica en dicha institución.

Lo anterior es con el propósito de asegurarse un ingreso en la edad de jubilación, en base de su prevención temprana, cuando se puede ahorrar una cantidad para su inversión. La institución se encarga de dar el mejor

uso a dicha reserva, para ofrecer al cliente, no solo la parte proporcional que le toque en rentas, sino un dividendo.

Por otra parte, la institución toma parte de sus ganancias para soportar sus gastos administrativos y de comercialización. En cierta forma, un plan de pensiones funciona como un sistema combinado de una caja de ahorro - para la constitución de la reserva económica -, y un sistema de nómina - al momento del pago de la rentas -, pero considera una serie de factores adicionales que complican el producto.

La determinación del monto que el cliente o asegurado, deben aportar, para obtener una específica renta, con su correspondiente periodicidad, con equis características, es normado por reglas actuariales, donde no solo interviene valores presentes de series aritméticas y geométricas, sino factores demográficos, como lo son la edad del asegurado en el momento de contratación, y su probabilidad de muerte.

Por último, Un plan de pensiones se instrumenta con tales características, que permita su flexibilidad al cambiante entorno financiero y a las diversas necesidades del mercado.

4.4 Objetivos del sistema.

El sistema que se presenta lleva como nombre "Pensiones Aseguradas", y tiene por objetivo, soportar la administración de el servicio de pensiones de forma automatizada, esto incluye:

- Registro, archivo y seguimiento a contrataciones
- Cálculos de anualidades, valuaciones y gastos de administración.
- Impresión de documentos contractuales.
- Transacciones diversas a las reservas económicas.
- Pago de rentas.
- Consultas estructuradas.
- Reportes y estadísticas.

Así mismo por convención a lo largo de este trabajo al Sistema Pensiones Aseguradas se le abreviara (SPA) y se le llamara también "Sistema de Pensiones"

4.5. Entorno de Desarrollo.

El sistema que se presenta en este trabajo, fue construido en un entorno IBM Main Frame (equipo 3090), bajo un esquema centralizado de información, este ambiente opera bajo el sistema operativo MVS (Multiple Virtual Storage: Almacenamiento Multiple Virtual).

Se empleo, como lenguaje de control de comandos, el JCL, así como múltiples herramientas de dicho entorno, tales como el IDCAMS, IEBGENER, ordenadores(sorts), compiladores, ISPF como entorno interactivo de comunicación al TSO (Time Sharing Options: Opciones de tiempo compartido), y la base de datos IDMS: (Integrated Data Management System: Sistema integrado de administración de datos)

Se emplearon tres lenguajes de programación: COBOL para los procedimientos "Batch" (Por lotes), ADS (Application Development System) para los procesos en línea, y PASCAL para el módulo en

microcomputadora. COBOL es el lenguaje de tercera generaci3n mas usado en el ambiente de los negocios, ADS es un lenguaje de cuarta generaci3n propiedad de la base de datos IDMS.

Finalmente, las metodologías estructuradas de desarrollo de sistemas utilizadas fueron la de SA/SD (Structured Analysis/Structured Design) de Yourdon/DeMarco/Constantine y la IE (Information Engineering) de James Martin



METODOLOGIAS DE DESARROLLO DE SISTEMAS

1.- MODELOS DE DESARROLLO DE SISTEMAS

2.- POSICIONAMIENTO DE LAS METODOLOGIAS DE DESARROLLO DE SISTEMAS

3.- METODOLOGIAS PRINCIPALES

1.- MODELOS DE DESARROLLO DE SISTEMAS

El proceso de conceptualización y desarrollo de un producto de software en muchas de las áreas del conocimiento, se torna cada vez mas amplio y complejo; la automatización de procesos es mas y mas indispensable; la construcción de Sistemas de Información se ha convertido en un generador de productividad de una en la mayoría de las organizaciones que deben controlar datos y procesos masivamente; por ello es de vital importancia el disponer de una " Organización Estructurada del Trabajo ", refiriendonos al Trabajo como al proceso computacional de Desarrollo de Sistemas.

Un sistema de software se comporta como un elemento tangible mas, una construcción mas, que bajo un entorno, tiene una utilización, pero también se rige por un ciclo de vida controlable y que refleja el grado de avance de la construcción.

El ciclo de vida de la construcción o desarrollo de un sistema describe la vida del sistema, desde que nace (se concibe la idea de un sistema), hasta que muere (se desecha por uno mejor), este empero no describe la Metodología para elaborarlo.

Ahora, como todo sistema se construye bajo condiciones diferentes, con recursos en bodega variados, además, los equipos desarrolladores que los construyen tienen experiencias y -escuelas- diferentes, es lógico pensar que hay un ciclo de vida para cada sistema.

Realmente, no hay una clasificación formal de los ciclos de vida, de hecho hemos detectado un desacuerdo entre los especialistas de la Ingeniería de Software sobre cuales son los diferentes ciclos de vida para el desarrollo de sistemas. Ha estos se les denomina también, Modelos -o filosofías- de desarrollo, los cuales están apegados a la naturaleza del sistema, y que mencionamos a continuación:

a) Prueba y error.

El más rudimentario de los modelos y de los más socorridos en el pasado, y actualmente todavía en muchas instalaciones.

Se trata de un modelo no estructurado de trabajo, a pesar de ello es útil en circunstancias especiales. Consiste básicamente de dos etapas (no funcionalmente definidas) para obtener el sistema, pequeño generalmente, ya que este modelo no puede plasmar un sistema con amplios alcances, donde se requiere de controles y direccionadores tan solo en la construcción.

La primer etapa considera una investigación inicial para determinar lo que se requiere. Lamentablemente en esta misma etapa, con los requerimientos mínimos, se inicia la labor de programación; con una idea escasa, no solo de que resultados debe dar el sistema, sino también de como es que se debe proceder para la obtención de dichos resultados.

La segunda etapa tratará de eliminar errores, de completar las funciones del programa, considerando adiciones de nuevas especificaciones. y de reconstruir tantas veces como sea necesario los componentes del sistema, para que este cumpla con los objetivos, si es que estos han permanecido estáticos y claros.

La deficiencia de este modelo es notable, ya que después de una serie de cambios, el código resultante se vuelve confuso, perdiendo claridad y nivel de reutilización. De esta manera los resultados esperados se van desvirtuando y labores de mantenimiento se tornan casi imposibles.

b) Cascada

Este modelo consiste de una secuencia obligada de fases por donde se encaminará el desarrollo del sistema de información, es decir, solo hay un camino para llegar a la creación de un producto, donde además, el canal de comunicación (entre las fases) es en un solo sentido, por lo que no podemos dirigirnos a una etapa sino se ha completado la previa. Tal y como se comporta una cascada donde el agua cae irremediamente a un nivel inferior. (FIG. 2.1)

Dichas fases de desarrollo son las siguientes :

- Estudio de factibilidad
- Análisis y especificación de requerimientos
- Especificación detallada
- Diseño
- Codificación y pruebas de módulos
- Integración y pruebas del sistema
- Liberación y mantenimiento

Este modelo presenta restricciones, cuando se presentan modificaciones continuas en los requerimientos, ya que no soporta actividades paralelas, ni produce resultados tangibles de manera rápida; sin embargo es ideal para proyectos largos y complejos, por la fortaleza de sus fases, y la formalidad que se puede obtener por el reconocer un inicio y cierre de fase en fase.

Este modelo introdujo la idea de un modelo estructurado de trabajo por ser el primer modelo estructurado - el primer intento -, presenta varias deficiencias (rigidez, falta de adaptabilidad, implica costos amplios), sin embargo, hasta la fecha, representa una gran evolución en la ingeniería de software.

c) Prototipo

Se basa en recurrir a un nivel pragmático (esbozo y puesta en marcha del software de manera rápida y funcional) al elaborar un prototipo inicial del sistema, se pretende dar un esqueleto base de formación al sistema. Este primer acercamiento permitirá a los desarrolladores y usuarios, determinar si es viable la alternativa de solución planteada como base del sistema.

La evaluación de este primer prototipo, podría entonces, desechar el camino elegido y reelaborar otro prototipo, de esta forma se puede continuar con un segundo, tercero o todos los prototipos necesarios para ir avanzando en la búsqueda del prototipo final que demuestre los requerimientos planteados y sea aprobado por los usuarios finales.

Cuando este modelo esta presente es peculiar, el generar un prototipo inicial simple y demostrativo, es decir sin realizar procesamiento de información; es una vista exterior donde se representan las pantallas (con la ayuda de algún software, DEMO II, en nuestro caso) navegando en ellas y procesando ficticiamente las salidas.

A esta serie de elaboraciones se le llama ciclo del prototipo (FIG. 2.2).

Como analogía podemos representarlo como una maquina mezcladora de pintura, en cuya parte superior contiene un recipiente, al cual agregamos pintura de ciertos colores seleccionados, y luego de mezclar;

preguntamos : así le gusta ?, que otros contrastes y brillos desea ?) para darnos cuenta de ... Ah! la quiere diferente, y sugerir, que le parezca con unas pequeñas manchas azules: y después de volver a mezclar, en el mismo recipiente, suceda que el cliente se decida a comprarnos 50 galones, o conformarnos con tener una muestra mas en bodega.

d) Desarrollo incremental

Este modelo, al igual que el anterior, considera la elaboración rápida y concreta de las soluciones esperadas, en base a un esqueleto, solo que en este caso no se refiere al producto entero, sino a los módulos individuales de este. Es decir se elige un módulo a la vez, el cual seguirá todas las fases del modelo, hasta su correcta operación, al aprobarse esta, se continuará con otra parte jerárquica del producto, y así sucesivamente hasta completar el producto final.

Se considera un modelo con flujo de operación "Top-Down" (Arriba-Abajo), ya que se parte de una serie de generalizaciones en la elección de los primeros módulos a desarrollar que se basan en una estructura jerárquica dejando al final el desarrollo de los submódulos inferiores.

De esta manera se genera un ciclo que agrega funciones operativas al producto final, donde las funciones menos importantes dependerán del desarrollo de las macrofunciones. La interfase a manera de menú, es la primera que se elabora, para disponer de una estructura de alto nivel que logre el control del sistema.

Difiere con el modelo de prototipo, en el sentido de que cada porción del producto en desarrollo pasa necesariamente desde el diseño, hasta la implementación y etapa de prueba tradicionales. Estas porciones corresponden a subpartes específicas del sistema final, donde en cada una de ellas se concentra en el desarrollo iterativo hasta ponerla en operación.

El liberar o dar por terminada una subparte o módulo(exceptuando aquellas etapas externas o interfases, que armaran al producto final) permite avanzar en la obtención del producto final (FIG. 2.3), y evaluar el tiempo y costo esperado.

Lo anterior hace ver que para sostenerse en este modelo se requiere de una planeación exhaustiva y cuidadosa.

e) Operacional

La idea fundamental de este modelo es dar relevancia a una descripción amplia al "Que" debe hacer el sistema, en lugar del "Como" debe hacerse. Si se considera que al automatizar el conjunto de parámetros que forman las especificaciones del sistema a desarrollar, de tal manera que puedan pasar a ser ejecutables, se simplificará el desarrollo del sistema al grado de poder omitir las fases de diseño, implementación y prueba.

Para esto se requieren de técnicas especializadas - como son los "generadores automáticos de código" y las herramientas CASE que sustituirán las fases faltantes en el dominio de este modelo, al menos, por la obtención de resultados.

Por parte del equipo desarrollador es preferible una experiencia muy vasta en el Análisis y Diseño de Sistemas, y capacidades para ampliar el dominio del estudio del problema, que resolverá el sistema, y lograr así, una lógica exacta y comprensible del conjunto de requerimientos.

Es por ello que el establecer algoritmos que concentren formalmente las especificaciones a detalle del software son necesarios para dar resultados rápidos.

En este modelo solo se consideran tres fases, la de análisis que debe ser impecable, para que se tengan resultados satisfactorios, la de especificación que debe ser totalmente formal y la de operación, cuya realización y logro de objetivos depende directamente de las técnicas especializadas utilizadas (FIG. 2.4).

Algunas desventajas de soportar este modelo para el desarrollo del sistema, son obvias, como los altos costos que se derivan de utilizar esas herramientas automáticas (y mas en nuestra realidad, donde se confunde y se espera que organice nuestra desorganización) y la escasez de equipos desarrolladores con las características solicitadas.

f) Reutilizable

El aprovechar el proceso de aprendizaje para aumentar la riqueza cognoscitiva se refleja en muchos avances, del área computacional, los compiladores son un claro ejemplo de ello, sus procesos de búsqueda y reconocimiento cada vez toman mas sentencias de código fuente, simplificando enormes esfuerzos a los programadores; y no se diga de los sistemas operativos multiusuario, que como administradores de procesos y recursos son tan eficientes que hasta dan recomendaciones en caso de contingencias.

Las bases de datos también han reutilizado funciones al proporcionar una plataforma de software que nos permite desarrollar aplicaciones complejas, sin tener que recodificar rutinas de acceso a archivo. Por su parte los "CASE", los subsistemas de Correo Electrónico, y muchos otros sistemas auxiliares, llevan, en si mismos, el concepto de reutilización.

El modelo reutilizable aplica muy bien en el entorno del proyecto que considera el desarrollo de un sistema que sustituirá a otro (sistema viejo o actual) ya sea que este o no en producción (operando con entradas y salidas reales). Este modelo se basa en ciertos procedimientos rescatables (que se ajusten tanto lógica, como físicamente a los nuevos requerimientos) que bien pueden ser primordiales o específicos (un módulo, una subrutina) para adaptarlos, agruparlos, renombrándolos, etc.; de tal manera que solo baste con reacondicionar, (por ejemplo, variables en el código existente, las interfases a submódulos, las salidas a dispositivos, valores de control, bibliotecas, etc.) si es posible, las entradas y salidas(tratando a la parte reutilizable como caja negra) afectadas, y ser parte del nuevo sistema.

Aunado a lo anterior, el modelo de reutilización también aplica en la elaboración de sistemas totalmente nuevos, no solo por ser el mas económico, sino por que se aprovecha de la alta dependencia entre los componentes que previamente fueron utilizados satisfactoriamente y que arrojaron, en la fase de pruebas, los resultados esperados.

De esta forma, los componentes individuales (separables de un sistema, módulo, submódulo, subrutina, o sentencia) son aprovechados por ser efectivos en la adaptación y modificabilidad que exige todo camino del perfeccionamiento del software, basado en soluciones practicas.

La reutilización se empleará básicamente en las fases de especificación, diseño, implementación y pruebas, ya que tanto el análisis como la operación generalmente tienden a ser circunstanciales a cada sistema y su entorno de desarrollo. (FIG. 2.5)

Para la mayoría de las etapas del ciclo de vida se podrá, entonces, colocar modificaciones y reutilizar la estructura lograda hasta el momento, por ello se sigue una estructura de abajo hacia arriba (contrariamente a la mayoría de los modelos).

g) Espiral

Se considera el mas avanzado de los modelos, ya que brinda una estructura donde pueden intercalarse, conjuntarse o fusionarse algunos de los modelos anteriormente descritos, es decir, se trata de una estructura cíclica normalizada.

Al cierre de un ciclo se pueden colocar diversos modelos, por ejemplo iniciar con un modelo de prototipo y terminar la espiral con un modelo cascada. Se dice un modelo normalizada, por que para todos los ciclos generados, sí existe un procedimiento consistente que brindará los resultados esperados, al inicio y termino de cada ciclo (flujo de transformación de un modelo en otro).

Las etapas normalizadas de cada ciclo se reducen a cuatro :

- a) Definir objetivos
- b) Evaluar y resolver riesgos
- c) Desarrollar el producto
- d) Planificar el siguiente ciclo si es que existe.

De esta forma se incrementará el alcance y la complejidad de las versiones preliminares del sistema, y se avanza con pasos firmes a la solución, minimizando el riesgo en que se incurre en todo proyecto de software.

Los modelos anteriores o representaciones del ciclo de vida del desarrollo de sistemas, no son exclusivos, es decir, puede haber mas variantes o combinación de ellos, siempre que se cumpla con ciertas reglas para que la consistencia y adaptabilidad al problema a resolver, no salga afectada; y poder asegurar un camino seguro para llevar acabo el desarrollo del sistema, que cumpla con los tiempos, objetivos y costos estimados. Sin embargo, recordar que lo importante es el resolver problemas con la mayor simpleza posible; no por que algo sea complejo se justifica el emplear métodos complejos.

2. POSICIONAMIENTO DE LAS METODOLOGIAS DE DESARROLLO DE SISTEMAS

En el punto anterior mencionamos el término "Organización Estructurada del Trabajo", nos hemos auxiliado de el como la forma óptima de realizar todo desarrollo de software, ya que debemos buscar una forma de distribución y control de la información generada a lo largo del desarrollo, que permita objetivizar el producto de software, planificar controles organizativos y verificar si hacen falta elementos evaluativos para avanzar en la obtención del sistema. Y la forma de hacerlo es a través de recursos metodologicos, es decir de utilizar Metodologias de Desarrollo de Sistemas.

Una metodología de desarrollo de sistemas es un conjunto interrelacionado de métodos, técnicas y herramientas que permiten realizar estructuradamente y guiar paso a paso cada una de las fases del ciclo de vida del sistema.

Conjuntando ideas, tenemos que el uso de estas metodologías nos proporciona una Organización Estructurada del Trabajo, siendo esta la forma óptima de lograr el producto final : 'el sistema'.

Las metodologías de desarrollo de sistemas se llegan a confundir con el ciclo de vida de un sistema, con ciertas técnicas y métodos aislados y hasta con metodologías de control de proyectos.

Por ello proponemos un posicionamiento o ubicación de las metodologías de desarrollo de sistemas en la plataforma del conocimiento computacional, el cual sostenga el concepto de la Organización Estructurada del Trabajo. Esta ubicación la representamos en el diagrama (FIG. 2.6).

El producto de software (en nuestro caso el SPA) es el objeto y razón de ser de este esquema, ya que sobre su desarrollo radica la conceptualización de cada uno de sus elementos.

En primera instancia y de forma inherente, no como elemento, sino como estructura base, se encuentra el modelo de desarrollo de sistemas (ciclo de vida) que siempre existe. En seguida, se explican cada uno de los elementos de los que esta compuesto el esquema.

El primero de ellos, y el más importante (marco superior) es el de metodologías -estudio de los métodos-, que brindará el seguimiento del modelo de desarrollo para el producto y su entorno específico. Como se señaló anteriormente las metodologías están compuestas del estudio de los métodos, técnicas, y herramientas, que corresponden a los siguientes marcos internos.

Los métodos (siguiente marco en profundidad), auxilian en la obtención de alguna fase específica de la metodología elegida y como todo procedimiento lógico permite ordenar el conocimiento para llegar a los resultados esperados.

Al utilizar técnicas se proporciona una guía inmediata de la forma de llevar a cabo estructuras específicas de desarrollo, útiles para el avance cognoscitivo del ciclo de vida del desarrollo, estas se basan generalmente en las estrategias que pueden obtenerse de las estructuras de arriba-abajo y viceversa. Las técnicas se ubican en el penúltimo marco en profundidad.

Finalmente (el marco más interno) las herramientas las que muchas veces, con nombres distintos en cada metodología, obtienen el mismo resultado final, normalización de bases de datos, creación del diccionario de datos, o de la bitacora de procesos, etc. Estas herramientas pueden ser propias (desarrolladas por uno mismo como son los programas), automáticas(CASE, por ejemplo) o manuales, y hasta combinaciones de las anteriores.

Con lo que respecta a la orientación o visión que va adquiriendo o sosteniendo el desarrollo del sistema, la consideramos una capa que inyecta sentido a su adyacente (la de metodologías), se clasifica en tres capas :

a) Visión u orientación hacia los procesos.

Estudio del flujo de entradas y salidas en función de los procesos a los que esta referido el sistema. Representa al procesamiento o transformación de la información del sistema.

b) Visión u orientación hacia los datos.

Estudio del acceso a los datos y la relación entre estos, independientemente de como los datos son usados. Representa la información importante para el sistema y las relaciones entre esta información.

c) Visión u orientación hacia los eventos.

Estudio de las transformaciones que tienen cabida en el flujo de operación del sistemas. Representa el comportamiento o respuesta del sistema a eventos, el tiempo es el factor básico.

Estas visiones para el desarrollo de sistemas, además de tenerse en cuenta a lo largo del desarrollo, pasan a ser los ejes direccionadores en las fases de Análisis y Diseño de sistemas. Si hiciéramos una analogía con un plano cartesiano tendríamos, en un eje a los datos, mientras que en el otro a los procesos (FIG. 2.7).

Los procrsos, junto con el tiempo serán la estructura diagramática que sostiene a las tres técnicas o dimensiones del modelado de sistemas, las cuales están bien diferenciadas por sus funciones y la situación circunstancial en que deben aplicarse.

Cualquiera de las tres visiones se dirigen al objetivo fundamental, -la producción del sistema-, de hecho están compuestas por tres formas de modelado (la conceptual, la funcional y la física), con un flujo lógico y claramente correspondientes entre sí, como se muestra en (FIG. 2.8), cuyas transformaciones, del conceptual al funcional, y de este al físico, dan el avance necesario para la creación del sistema.

El como se subdivide en componentes el sistema, también tiene que ver con la orientación, por ejemplo para la orientación a procesos tenemos (FIG. 2.9) que el componente de más bajo nivel es el conjunto de operaciones llevadas a cabo por el sistema, las cuales exigen una descripción detallada, pero con el inconveniente de tener menor estabilidad.

Para todo sistema a desarrollar, conviene el considerar estas visiones tratando de no descartar ninguna, dejando los tres enfoques para utilizarlos en las fases adecuadas, la orientación hacia los datos, por ejemplo, aplica muy bien cuando se llega al diseño de la base de datos.

3.- METODOLOGIAS PRINCIPALES

Las metodologías que hemos identificado para mencionarlas por su importancia y mayor inferencia en el logro de objetivos del desarrollo de sistemas son las siguientes :

- SA/SD (Structured Analysis / Structured Design)
- IE (Information Engineering)
- JSD (Jackson's System Development)
- SSADM (Structured Systems Analysis and Design Methodology)
- Militar Standard

Todas estas metodologías se rigen por los conceptos del desarrollo estructurado de sistemas.

La decisión de que metodología es la correcta, y en base a que parámetros se debe elegir, resulta delicado, al grado de tener que considerarse en la fase de planeación estratégica; de no enmarcar un estudio en este rubro, podríamos caer en situaciones adversas como las siguientes ...

La más radical es tener una panacea, donde por seguir ciegamente y usar estrictamente una metodología, sin considerar el tipo de proyecto y tamaño de este, ocasiona problemas visibles que bloquearían el flujo de desarrollo del sistema.

Así mismo cuando utilizamos una metodología a lo largo del proyecto, si esta es seguida dogmáticamente porque la teoría así lo dice, aunque pragmáticamente no aplique, resultaría contraproducente al flujo óptimo de construcción general del sistema.

La determinación o elección de una metodología que cubra las necesidades del sistema a desarrollar, teóricamente se rige por tres factores genéricos; en primer término por la aplicación o problema a resolver, y en segundo por el tamaño del sistema y su complejidad; estos darán bases para distinguir la adaptabilidad de la metodología, y su congruencia con el seguimiento necesario para el desarrollo del sistema.

Cabe señalar que dicha elección depende del constructor, por ejemplo: si desarrollas como un despacho consultor, puede resultar muy complicada el negociar proponiendo unes el encargado de sistema de complicado para l un espiral, y si eres el encargado de sistema de una organización, el desarrollo incremental es conveniente.

De esta manera si se pretendiera desarrollar un sistema de control de vuelos de una aerolínea, sería imposible pensar en una metodología simplista, que no enfatizará la fase de diseño y pasará directamente a codificar.

Realmente lo que determinará finalmente la elección de una metodología es la experiencia del equipo de trabajo encargado de desarrollar el proyecto, la cual también se refleja en las posibles adaptaciones de la metodología en cuestión, al desarrollo del sistema. El concepto de la metodología ideal no es aplicable, por lo que es necesario, reorganizar etapas, conjuntarlas, cambiar su secuencia, o hasta suprimirlas, evitando así el adaptar la estructura del sistema real a un modelo teórico.

Es válido entonces decir, que existe una metodología para cada desarrollo (para cada sistema), y esta depende no solo del problema a resolver (el automatizar la administración de un producto, el control de la asignación de números telefónicos, controlar la emisión de vuelos de una línea aérea, verificar los gastos

de una compañía, administrar transacciones bancarias, controlar el pago de impuestos, etc. y de sus atributos (complejidad, magnitud, severidad, etc.), sino también del entorno donde se resolverá el problema, como es:

El área de conocimiento, recursos disponibles, plataforma computacional, la existencia o no de un sistema previo, tipo de constructor que seas; un despacho, o un interno de la compañía, o hasta interno del departamento que requiere el sistema; también factores como la disponibilidad de los usuarios y de los expertos del tópico a analizar influyen en la determinación de la metodología.

Podemos hacer también una agrupación de varias metodologías por ser parecidas, donde corresponda en primera instancia la visión u orientación en que se basan. Es decir, su uso no es rígido, estas se pueden conjuntar y sustituirse. La existencia de metodologías formales, como la SA/SD, se deben tomar en lo más posible, pero también que estas, se deben adaptar en cada caso, para resolver de mejor modo el problema.

Una metodología no hace las veces de una secuencia tiránica de pasos, sino una ayuda, una guía para resolver problemas, y en el mejor de los casos, un estándar de desarrollo. Recordar que al fijar estándares de desarrollo, no se fijan reglas inviolables, sino se deben recomendar varios estandares para dejar a libre albedrío la adopción de alguno y lo que es mas, todavía el elegido, puede sufrir adaptaciones personales.

Cabe aclarar que no nos referimos a las reglas particulares que posee cada técnica, las cuales son inamovibles; como es el identificar uno y solo un proceso de datos como el sistema, en un diagrama de contexto de la metodología SA/SD, en la fase de Análisis de Sistemas.

METODOLOGIA SA/SD (STRUCTURED ANALYSIS / STRUCTURED DESIGN)

Es una de las metodologías más conocidas, no solo por contener una riqueza metodológica, en el sentido, de utilizar más técnicas y herramientas para las fases de análisis y diseño del sistema, sino por ser precursora en las técnicas que muchas otras metodologías han imitado, tal es el caso del Diagrama de Flujo de Datos.

Cada técnica ofrece un diferente nivel de abstracción, para dar ciertas perspectivas. Esta metodología es ideal para sistemas que manipulan datos masivamente; por lo que le da una preferencia a los procesos que afectan a dichos datos. Considerando esto, esta metodología se centra bajo una orientación o visión hacia los procesos.

Los componentes básicos, en esta metodología los presentamos a continuación :

a) DFD (Diagrama de Flujo de Datos).

Describe el ambiente dentro del cual el procedimiento automático va a ser diseñado, así como la estructura lógica de la aplicación y la más importante en el análisis estructurado.

Es a través de un conjunto jerárquico de DFD's, que podemos modelar un sistema, ya que estos identifican el flujo de datos entre las funciones, los datos que son almacenados por el sistema, y las interfaces con entidades externas al sistema. Un ejemplo de este importante diagrama se muestra en (FIG. 2.10).

b) DD (Diccionario de Datos).

Lista controlada de toda definición presente en el flujo de datos entre las funciones y el almacenamiento de datos. El control se logra por medio de la centralización, evitando duplicaciones e inconsistencias.

Este componente es fundamental, pues realmente es la base de datos del proyecto de desarrollo, por ello la mayoría de las metodologías lo proponen. Ejemplificando tenemos la descripción de un DD en (FIG. 2.11)

c) SE (Idioma Estructurado).

Trozos explícitos de algún lenguaje natural que describe la transformación de datos hecha por cada elemento, como se muestra en la (FIG. 2.12)

d) SD (Diagrama Estructurado).

Descripción de la descomposición del sistema en módulos en base a los diagramas de flujo de datos obtenidos, buscando siempre que los grados de acoplamiento y cohesión entre módulos sean los correctos. Un ejemplo de un diagrama estructurado se muestra en la (FIG. 2.13)

e) TRG (Gráfica de Transformación).

Dada la extensión de los sistemas a procesos en tiempo real, estas gráficas cubrieron el análisis de sistemas de tiempo real, son muy similares a los DFD's, sin embargo, un TRG, sí muestra la secuencia de las funciones involucradas.

Por lo que toca al Análisis Estructurado, esta metodología es la que mas simplicidad da a la obtención del modelo físico del sistema a desarrollar partiendo de un sistema ya existente.

En el área del desarrollo de sistemas es común, al realizar el análisis de este, encontrarse con algún sistema existente, ya sea este en fase productiva o no, y que generalmente no contiene todo el conjunto de requerimientos para el nuevo sistema a desarrollar.

Es a través del análisis estructurado que se logra la obtención del diagrama de flujo de datos que representa físicamente al nuevo sistema a desarrollar, por medio de un conjunto de transiciones que van de los modelos físico y conceptual del sistema actual, a los correspondientes modelos conceptual y físico del nuevo sistema.

Es en el diagrama (FIG. 2.14) donde se presenta este flujo de transformación basado en los DFD que definen al sistema actual, para establecer el DFD lógico del nuevo sistema, y de ahí ir deduciendo, utilizando todos los DFD semifísicos necesarios, el modelo físico del nuevo sistema, que permitirá dar inicio al diseño estructurado (SD) como etapa posterior en la metodología.

METODOLOGIA JSD (JACKSON'S SYSTEM DEVELOPMENT)

Esta metodología aprovecha el enfoque de diseño orientado a objetos y la descomposición funcional. Intenta abarcar lo mas posible el ciclo de vida, desde el análisis de los requerimientos hasta la programación.

Sus etapas son solo tres que comprenden procesos amplios, como se muestran a continuación :

a) Modelado.

Dado un análisis del entorno real, este es representado en términos de acciones o eventos que interfieren con entidades (útiles para modelar objetos). La técnica estructurada para llegar a esto es la creación de un PSD (diagrama de estructura de procesos).

b) Red .

En esta etapa cada entidad obtenida es modelada como un proceso, y el sistema entero se convierte, entonces en una red de procesos interconectados e intercomunicados, descrita gráficamente por la SSN (red de especificación del sistema).

c) Implementación .

La red de procesos es dirigida a la implementación, por medio de una transformación secuencial. Se utiliza la técnica de inversión de procesos.

METODOLOGIA INFORMATION ENGINEERING

Esta metodología se basa en las técnicas tradicionales de desarrollo estructurado junto con los conceptos de vanguardia que ha ofrecido últimamente la ingeniería de software, como son la modelación de datos, las técnicas de obtención interactiva de especificación de requerimientos JAD (Joint Application Design), los prototipos, la aplicación de paquetes comerciales, etc.

Se rige bajo una orientación hacia el modelado de procesos, como una representación lógica de las funciones que debe ejecutar el sistema a desarrollar.

El flujo de desarrollo es de arriba hacia abajo, partiendo de un alto grado de definición de los requerimientos funcionales del sistema, se pasará a un flujo dinámico de desarrollo que definirá los requerimientos operacionales, ampliará la calidad del software, mostrará rasgos de reutilización y reducirá el tiempo de respuesta.

Así todos los requerimientos son refinados por la acción de dicho flujo, etapa por etapa, y por tratarse de la conceptualización de partir de lo general hacia lo particular ('top-down') se reducen las posibles divergencias y redundancias con los requerimientos originales dados por los usuarios. Damos a continuación las definiciones de los modelos mencionados anteriormente:

a) Modelado de los procesos a nivel conceptual.

Representación lógica de las funciones que deben ser ejecutadas por el sistema, como es el flujo de datos entre las funciones. Esencia del sistema desde una perspectiva funcional.

b) Modelado de los procesos a nivel funcional.

Surge a partir del modelado a nivel conceptual, además de mostrar las interrelaciones entre las funciones involucradas, toma en consideración algunas características físicas del sistema; entre estas la estructura organizativa del entorno del usuario.

c) Modelado de los procesos a nivel físico.

Se obtiene en base al modelado a nivel funcional y se compone principalmente de módulos que pueden ser codificados por un conjunto de instrucciones que sean ejecutadas por un sistema computacional.

Las fases que componen esta metodología son las siguientes :

— Planeación : Desarrollo de las estrategias del sistema de información y en los planes tácticos de los objetivos de la organización.

— Análisis : Se modelan y recopilan los requerimientos formales del nuevo sistema y los planes iniciales para diseñar el sistema son desarrollados.

— Diseño : Todos los componentes del nuevo sistema son diseñados y los planes para construir, integrar, probar e implementar son finalizados.

– Construcción e implementación : El nuevo sistema de aplicación es construido, integrado, probado e implementado en la organización

– Administración del proyecto : Esta última fase es la base de la metodología, por contener todas aquellas herramientas, guías y técnicas para estructurar el proyecto estimarlo (riesgos y alcances de este), valorarlo, y disponerle una administración de cambios y una calidad de software propia.

En la (FIG. 2.15) podemos representar cada una de las anteriores fases en una pirámide cuya base de construcción del sistema es la administración del proyecto. El que sea un círculo esta base indica que es dinámica, compuesta de seis procesos que operan desde la primer etapa (1) hasta la última etapa (N) de cada una de las fases piramidales que sostiene y direcciona. La (FIG. 2.16) muestra dichas fases, definidas como sigue :

– Estructuración del proyecto : Identifica y confirma los problemas que se resolverán, define el tópico los objetivos y principalmente los productos terminados del proyecto. y el diagrama del proyecto.

La estructuración se lleva acabo generalmente antes de la conclusión de una fase (es decir durante la etapa N-1), como una parte preparativa para pasar a la siguiente fase.

– Planeación del proyecto : Construcción de planes detallados del proyecto como son : los esquemas de recursos y tareas, planes para la administración de la calidad del software, administración de riesgos y entrenamiento. La planeación es típicamente completada durante la primera etapa de una fase, donde la salida de este proceso es el plan del proyecto.

– Control del proyecto : Asegura que el proyecto es procesado como esquema, para este fin el proyecto es progresivamente evaluado, las conversiones en los planes y las acciones de corrección son iniciadas si son necesarias. Este proceso toma lugar en la penúltima etapa (N-2) de cada fase del proyecto.

– Generación de reportes de estado del proyecto : A través de la información generada por el proceso de control, se dispone de una estructura de comunicación de las condiciones de estado del proyecto, la comunidad de usuarios, los tiempos del proyecto, etc. Este proceso ocurre frecuentemente a lo largo del proyecto, por lo que su uso no es exclusivo de alguna etapa.

– Administración de cambios : Como parte fundamental en todo proyecto este proceso engloba la documentación, evaluación y disposición de las solicitudes de cambios. Cada solicitud de cambios debe ser formalmente documentada, analizada por su impacto potencial en el sistema, aprobada o rechazada y esquematizada para su implantación. Este proceso se puede llevar a cabo a lo largo de todo el proyecto.

– Conclusión del proyecto : Es en la última etapa (N) de una fase del proyecto que se realiza dicha conclusión, después de que la administración del proyecto ha cumplido satisfactoriamente con los objetivos. En este proceso se da por concluida formalmente una fase, además el nivel de ejecución es evaluado y aspectos particulares son almacenados para que sirvan de referencia en futuros proyectos.

METODOLOGIA Militar Standard (MIL-STD-2167)

Esta metodología es aplicable, como todo estándar a un gran número de casos de desarrollo de software, exceptuando aquellas aplicaciones cortas, con funciones fijas e invariantes en todo el ciclo de vida del sistema.

Esta basado en un modelo de desarrollo cascada, hasta las primeras cuatro fases, es decir se tendrán :

- a) análisis de los requerimientos del software
- b) diseño preliminar
- c) diseño detallado
- d) codificación y pruebas de los módulos
- e) Integración y prueba de integración de los componentes del software
- f) configuración y prueba de los componentes del software

Las últimas dos fases las proporciona el estándar militar , dado que la liberación del software esta regida por un conjunto de configuraciones de software de computadora (CSCI) que consiste de uno o mas componentes de software, los cuales son entidades lógicas compuestas de una o más unidades (pequeñas entidades lógicas), que son implementadas dentro del código.

El principal aspecto que hace valiosa a esta metodología, es que para cada fase del ciclo de desarrollo del software(modelo), el estándar proporciona especificaciones detalladas como son las siguientes :

- Las actividades que continuaran fuera de la fase.
- Los productos o resultados que van a ser liberados.
- Las revisiones de la obtención de productos que serán monitoreadas.
- La estructura del software desarrollado y su documentación asociada.
- Los procedimientos para acceder y archivar el software y su documentación.

METODOLOGIA SSADM (Structured Systems Analysis and Design Methodology)

Esta metodología, tiene el inconveniente de ayudar solamente en la fase de análisis de sistemas, sin embargo puede aplicarse bajo los tres enfoques u orientaciones de los que hemos hablado, (orientaciones hacia procesos, datos y eventos), lo que le brinda robustez a sus propuestas de contenido para dicha fase.

Como una extensión del método de análisis estructurado propuesto por Gane & Sarson, es utilizada en gran medida por las áreas de desarrollo de sistemas del gobierno y defensa del Reino Unido. Esta metodología propone un conjunto de productos de especificaciones con sus procedimientos asociados, para realizar en análisis.

Las técnicas diagramáticas para obtener estos productos son :

a) DFD (Diagrama de Flujo de Datos) :

Con su propia nomenclatura diagramática, también se emplean los DFD's para mostrar el flujo y procesamiento de los datos en un sistema. Estos peculiares DFD de SSADM, solo manejan tres objetos: Un flujo de datos externo, que muestra el flujo de datos entre dos símbolos externos; un flujo de recursos, que señala el flujo de los recursos físicos; y un símbolo de procesos de datos primitivo que profundiza en una descripción de un proceso específico.

b) LDS (Estructura Lógica de Datos) :

Este diagrama muestra las relaciones entre las entidades de datos, lo cual indica que obtiene los mismos resultados que un diagrama de entidad relación. Este diagrama tiene el inconveniente de soportar solo relaciones binarias. También muestra una relación directa con la alimentación en el diccionario de datos, en el sentido de que no se pueden representar entidades y relaciones que no tienen entrada en dicho diccionario.

c) ELH (Historia de la Vida de Entidades) :

Con este diagrama se plasman los eventos que afectan a las entidades definidas en un sistema. Este diagrama se aplica a una simple entidad, para representar el procesamiento de los datos relacionados con dicha entidad (la cual debe tener una entrada en el DD). Es muy similar a un diagrama de estructura (STC).

FIG. (2.1)

EL MODELO CASCADA DEL DESARROLLO DE SISTEMAS

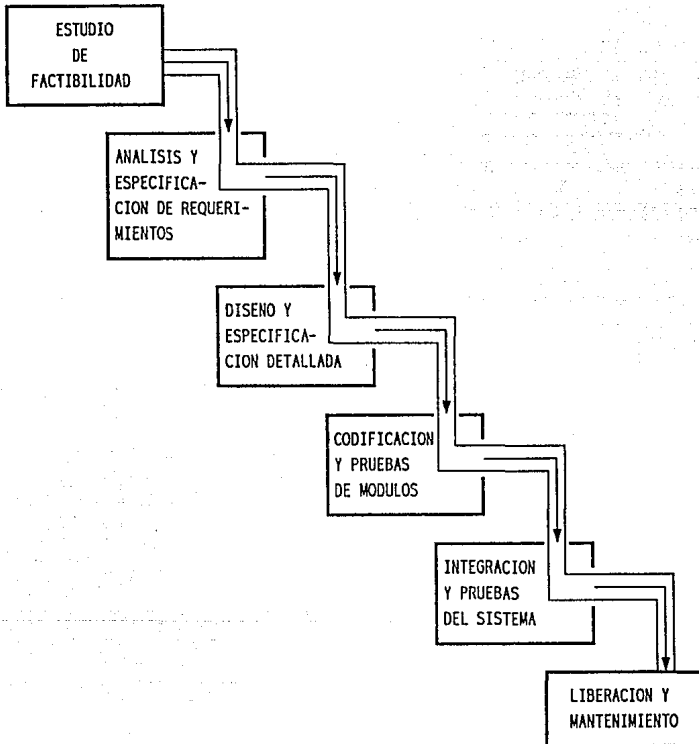


FIG. (2.2)

MODELO DE PROTOTIPO DEL DESARROLLO DE SISTEMAS

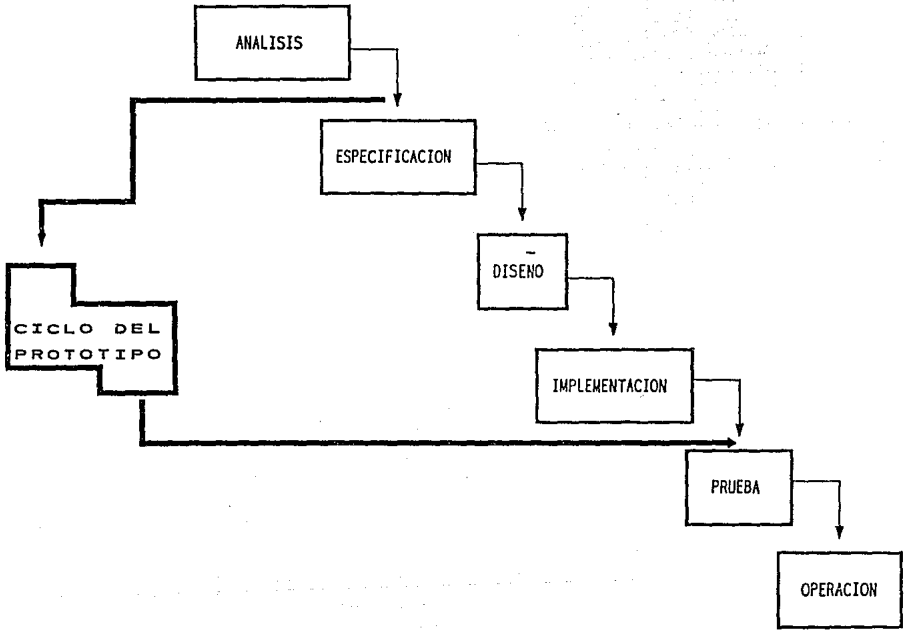


FIG. (2.3)

MODELO -INCREMENTAL- DEL DESARROLLO DE SISTEMAS

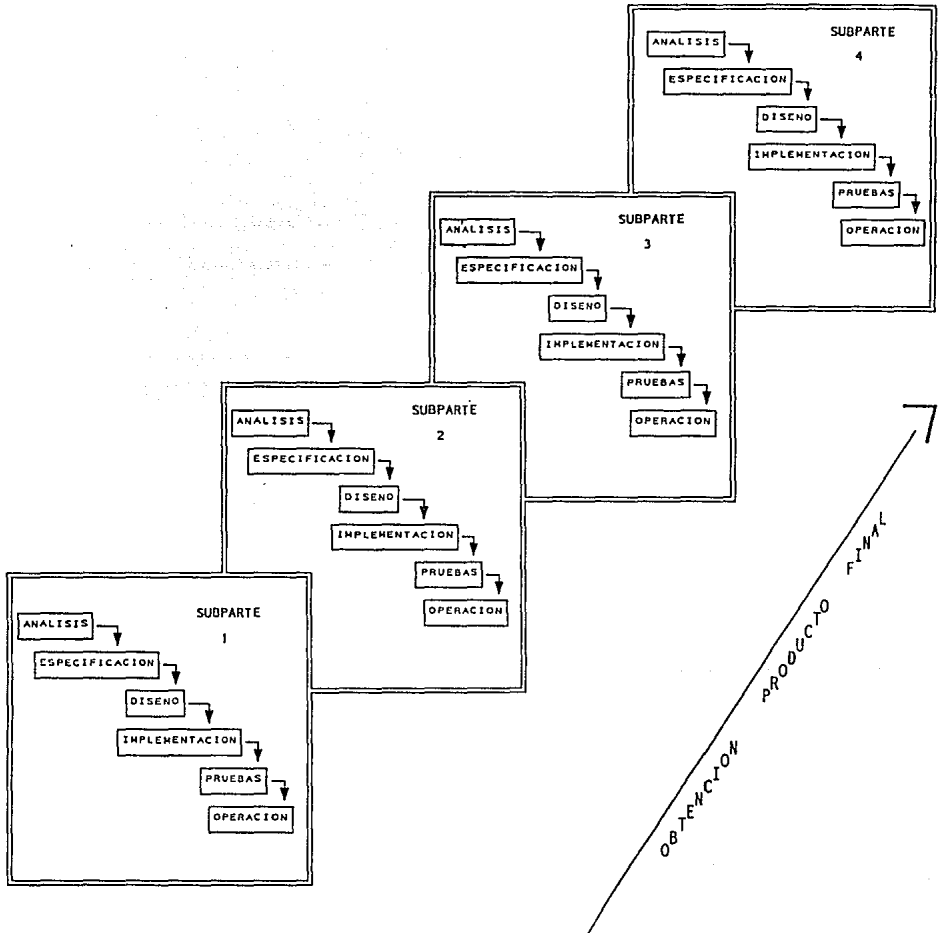


FIG. (2.4)

MODELO OPERACIONAL DEL DESARROLLO DE SISTEMAS

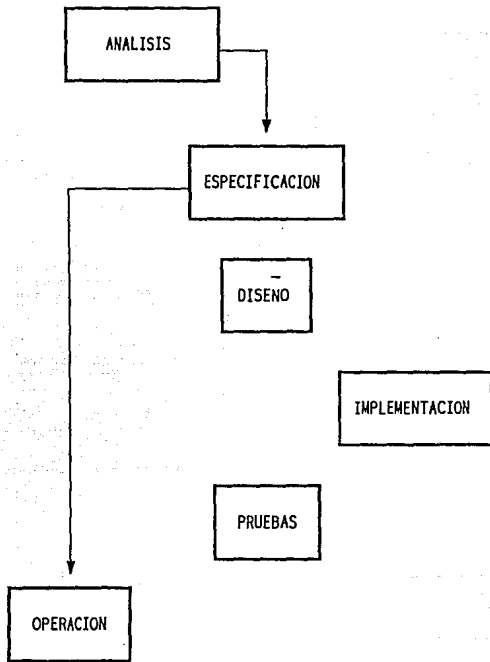


FIG. (2.5)

MODELO REUTILIZABLE EN EL DESARROLLO DE SISTEMAS

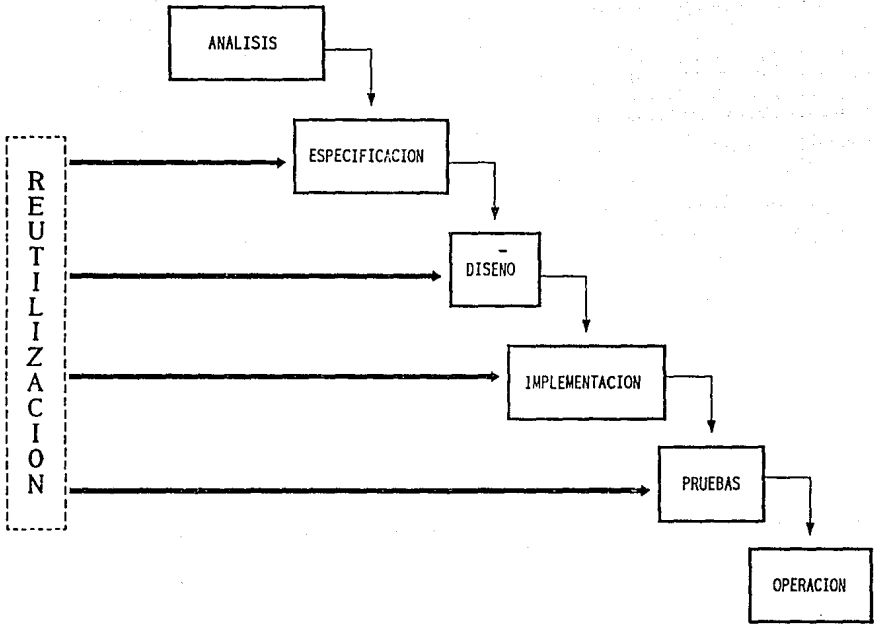


FIG. (2.6)

ORGANIZACION ESTRUCTURADA DEL TRABAJO

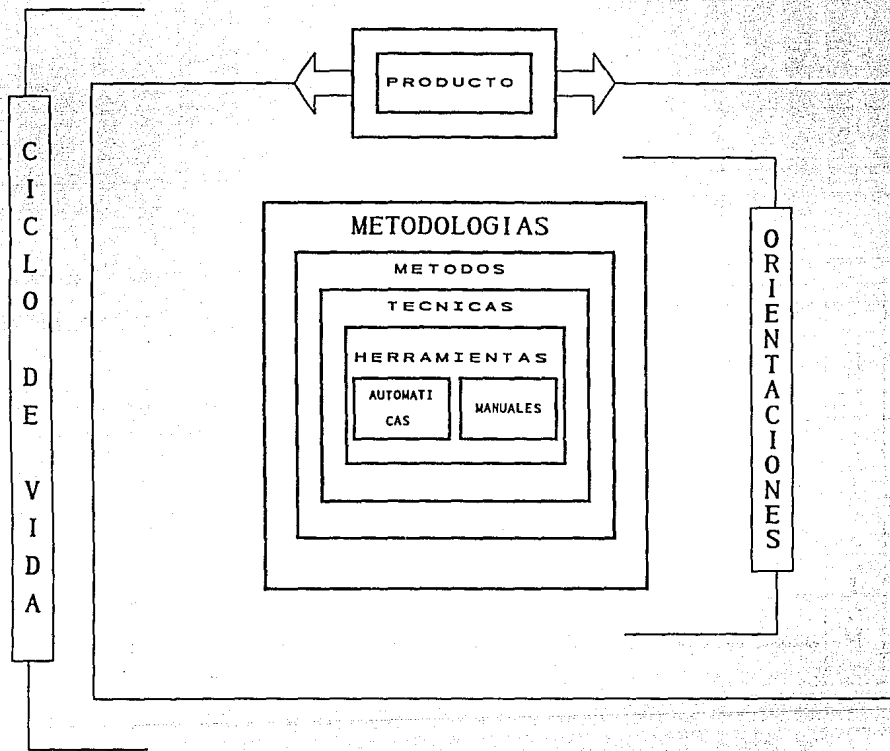


FIG. (2.7) LAS TRES DIMENSIONES DEL MODELADO DE SISTEMAS

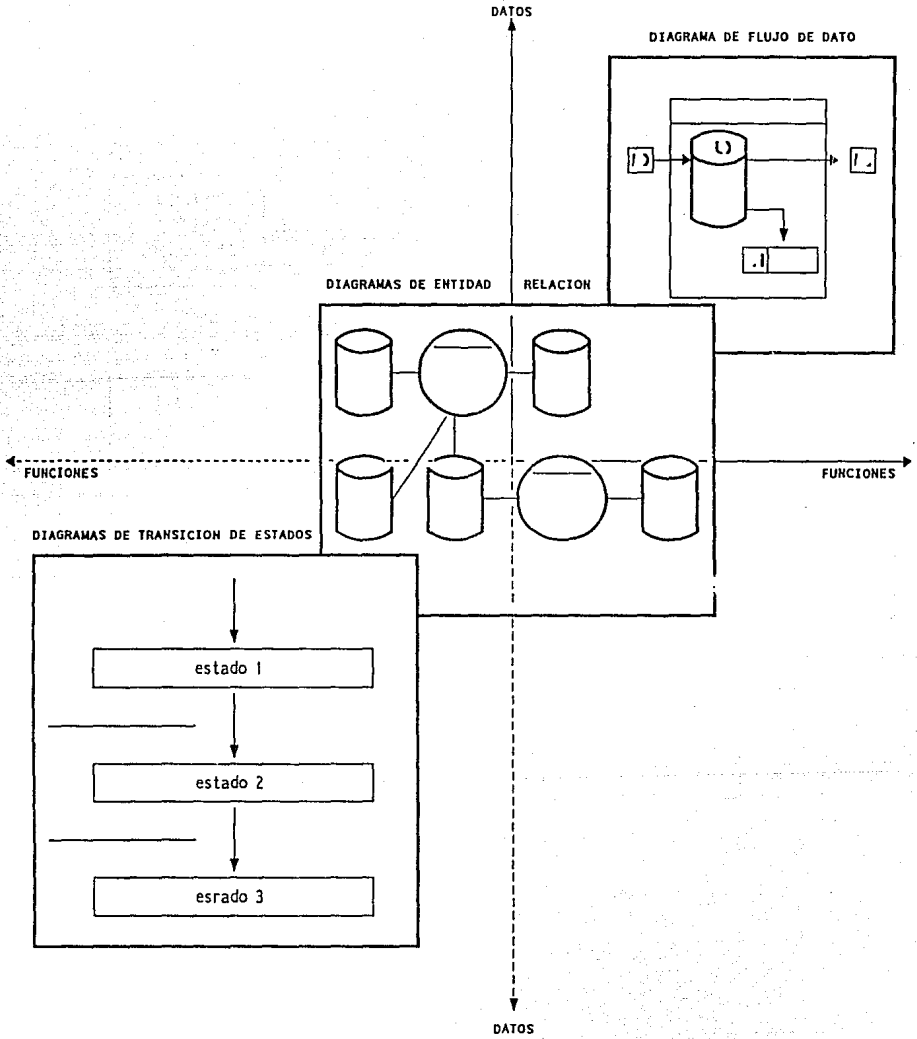


FIG. (2.8)

VISIONES U ORIENTACIONES EN EL DESARROLLO DE SISTEMAS

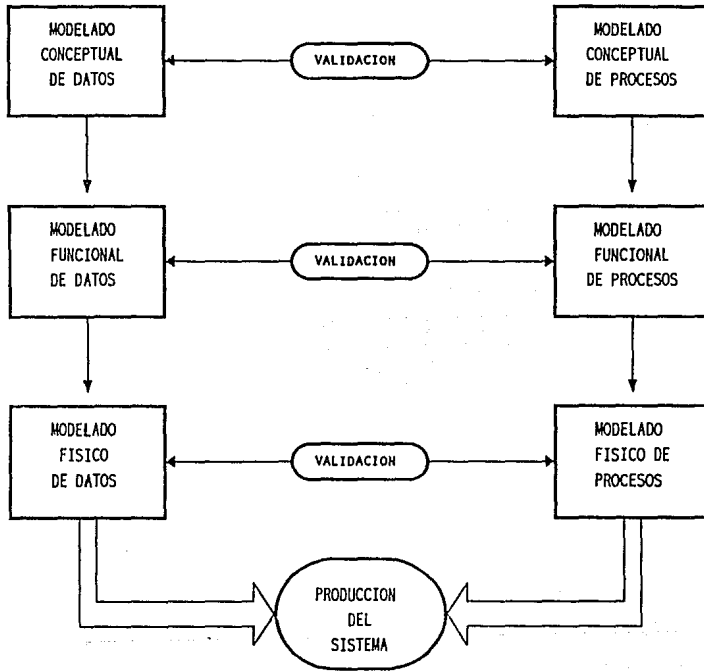


FIG. (2.9)

ESTRUCTURA JERARQUICA DE LOS COMPONENTES DE UN SISTEMA ORIENTADO A PROCESOS

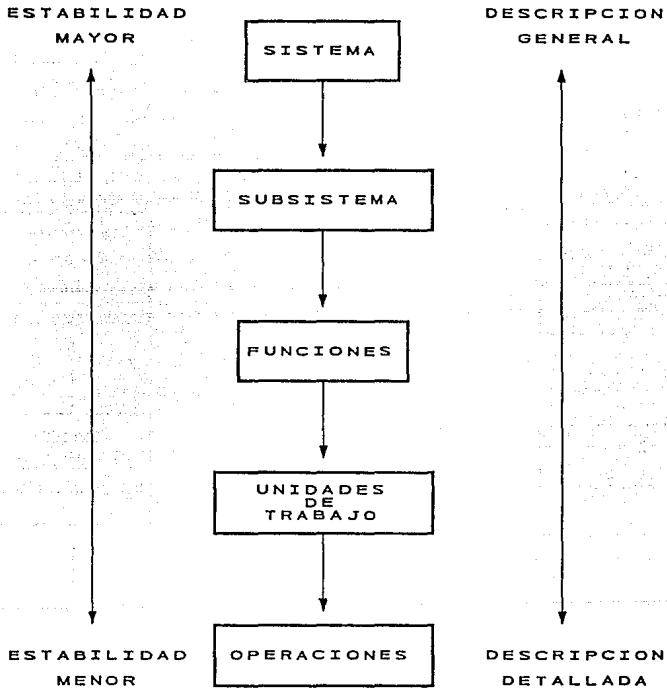
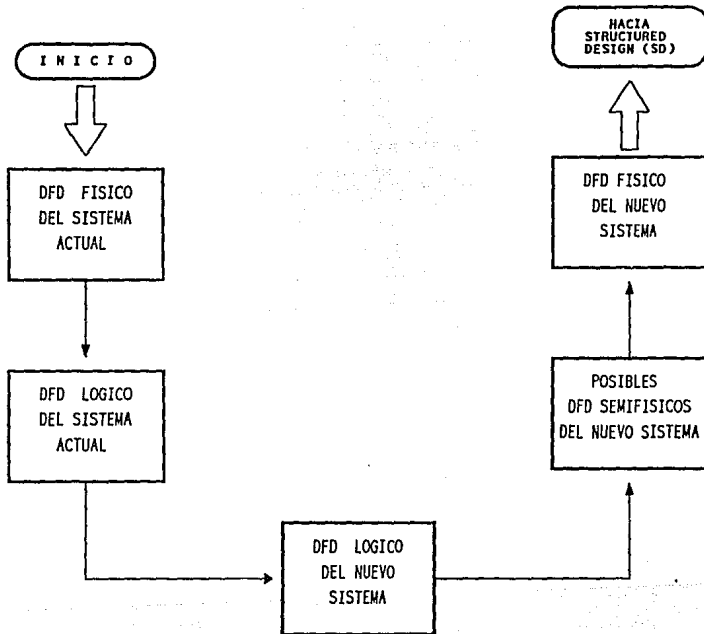


FIG. (2.14)

LOS MODELOS FISICOS Y LOGICOS EN EL ANALISIS ESTRUCTURADO



FASES DE LA METODOLOGIA INFORMATION ENGINEERING

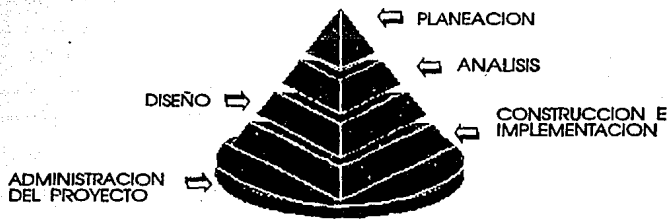


FIG. 2.15

LA FASE DE ADMINISTRACION DEL PROYECTO



ADMINISTRACION DEL PROYECTO

FIG. 2.16



METODOLOGIA UTILIZADA

- 1. CIRCUNSTANCIAS DEL DESARROLLO**
- 2. METODOLOGIA UTILIZADA**
- 3. SECUENCIA DE DESARROLLO**

1. CIRCUNSTANCIAS DEL DESARROLLO

El Sistema Pensiones Aseguradas se desarrollo bajo condiciones especiales entre ellas, quizás la más importante se refiera a que ese tipo de planes de pensiones, son nuevos en el mercado. Cuando un producto ya existe en el mercado, existe también algún sistema de administración, automático o manual, que independientemente de su buen funcionamiento, se encuentra operando y soporta una buena cantidad de casos extremos y errores comunes de operación.

Estos sistemas se configuran a base de la experiencia, y se depuran a base de prueba y error. Tratan con casos reales todo el tiempo. En nuestro caso, el producto no tenía ningún sistema de administración, aún no se encontraba definida la secuencia de operaciones diarias con las que se manejaría cada una de los contratos de pensión. Por lo que la elaboración de un sistema computarizado de información llevaba implícita la definición de la administración del producto.

Por otro lado, la definición misma del producto se encuentra en evolución es decir, se encuentra cambiando continuamente. La administración del producto, por tanto, se modifica cada vez que el producto se replantea.

Otra característica de relevancia, que normó la metodología de desarrollo es la que concierne al tipo de contrato que existe entre las personas que requieren el sistema, y las personas que lo desarrollan. Este tipo de contratos puede ser de alguna de las siguientes formas:

- El equipo de desarrollo pertenece al mismo departamento de los usuario en este caso, el contrato es más bien una orden de trabajo.
- El equipo de desarrollo del sistema, pertenece a la misma compañía a donde pertenecen los que lo necesitan. en este el contrato se realiza a través de una solicitud de los usuarios hacia la dirección de sistemas, y esta procede a evaluar su prioridad en función del resto de solicitudes de trabajo.
- El equipo de desarrollo no pertenece a la misma compañía de los usuarios, en este caso, se requiere un contrato formal, con una estimación de costos y tiempos muy precisa.

Este último fue nuestro caso.

De hecho, cada sistema tiene su problemática muy específica, si se tratase resolver todos los sistemas de igual manera, tal vez obtendríamos un buen planteamiento de alguno de ellos, pero el resto se plantearía de forma incorrecta.

Cada problema configura en sí mismo, su propia metodología de desarrollo que se configura de partes de metodologías conocidas.

Los puristas defienden la idea, que todo desarrollo de sistemas debe ser realizado en base a una metodología única, con el fin de proveer congruencia al sistema en todas sus etapas de desarrollo. Nosotros nos inclinamos por la idea, de que un ingeniero profesional debe conocer todo lo existente, para saber en que momento, cuando y en que proporción aplicar cada método o técnica, y debe estar alerta continuamente en búsqueda de mejores metodologías y herramientas de trabajo.

Un ingeniero de software es una entidad viva y participativa en la evolución de la ingeniería de software, debe seguir reglas y lineamiento que funcionen, pero nunca restringirse a estos, debe proponer siempre alternativas de una mejor solución.

Más sin embargo, esto debe ser tomado con mucha precaución, pues a una instalación de desarrollo (ó equipo) de sistemas no se la puede recomendar que se encuentre innovando continuamente, más bien esto debe ser una actividad bien controlada, ya que una instalación de desarrollo de sistemas, es una fabrica de producción de software, donde conviene mantener fija la línea de producción durante un buen tiempo. a fin de maximizar producción, minimizar errores, y estandarizar procedimientos. La idea fundamental es la siguiente:

"Hacer al sistema independiente de las personas", Ya que la empresa no depende de las personas, tampoco el sistema debe depender de determinada persona. Los casos en el que el personal de desarrollo cesa sus relaciones con la empresa, y esta pierde grandes cantidades por no saber operar ni mantener al sistema, son rasgos que caracterizan nuestra época actual, -una lucha continua entre el conocimiento y la aplicación satisfactoria de este-. Y hablan de mucha inmadurez y falta de profesionalismo en las empresas y los individuos. Lo que sí debemos tener presente es:

" Lo peor no es tener problemas (que siempre existen), sino carecer de soluciones "

2. METODOLOGIA UTILIZADA

El desarrollo bajo un esquema de cascada en tiempos restringidos no esta peleado con el desarrollo de prototipos. En nuestro caso, se desarrolló bajo ese esquema, auxiliandose de muchos métodos y técnicas de la metodologías:

- SA/SD (Structured Analysis / Structured Design) de Yourdon / DeMarco/Constantine .

- IE (Information Engineering) creada por James Martin. La metodología maestra fue la SA/SD, utilizándola a lo largo de las fases de Análisis y Diseño; mientras que la metodología IE, figuró solo como apoyo en el modelado de la base de dato.

El desarrollo bajo una filosofía de cascada es casi forzado, pues se debe tener una estimación de costos simple, para pactar un contrato con la organización antes de realizar tareas de diseño. Un factor muy importante para el caso de SPA, es que se contrato el desarrollo del sistema, fijándose de antemano restricciones de tiempo, costo, equipo de cómputo, lenguaje de programación y software a emplear.

Por lo cuál llegamos al modelo cascada como el que representaba mayor solidez a cada una de las etapas, en cuanto a tiempo y contenido (esto, que en teoría es muy idealista, como mencionaremos mas adelante, no se cumplió del todo) Cabe señalar que el modelo de cascada por que en base a este se pueden llegar a realizar prototipos, o trasladarse a un desarrollo creciente.

Se realizaron muchas iteraciones dentro de cada una de las fases del ciclo de vida, esto se debe mucho a la definición inmadura del producto y a la ausencia de algún modelo de administración de dicho producto que sirviese como base para la construcción del nuevo sistema. Sin embargo estuvieron claramente definidas las siguientes fases:

- 1.- Análisis preliminar y planeación.
- 2.- Análisis profundo y especificación de requerimientos. @PUNTUALIZA = 3.- Diseño
- 4.- Construcción y pruebas.
- 5.- Operación y mantenimiento.

La etapa de planeación se realizó intrínseca durante todo el proyecto, partiéndose de un plan inicial. Gracias a la planeación, se pudo mantener siempre al desarrollo del sistema lo más cercano posible a las condiciones iniciales de contratación, y por otro lado, el plan también sirvió para mantener informada a la

dirección de sistemas de las condiciones y al usuario sobre el estado del desarrollo. El sistema se desarrolla bajo un entorno de base de datos, esto por sí mismo introduce un filosofía de desarrollo, más tendiente a la actualización inmediata de archivos maestros, y uso de herramientas avanzadas para la construcción del sistema.

La base de datos utilizada (IDMS/CA), además impone ciertas restricciones en la elaboración de programas en línea, dotándoles de una lógica clara y concisa al separa todo programa, en "Response process", que se ejecutan si y solo si determinadas teclas son presionadas.

Un "Response Process" es un segmento de código fuente, modularizado de tal manera, que solo ese segmento es ejecuta cuando el usuario presiona la tecla asociada al segmento. esta técnica simplifica mucho la lógica de los programas y les brinda de hecho, una nueva filosofía.

El lenguaje de cuarta generación de la base de datos (ADS), facilita la construcción rápida de prototipos, elaboración de pantallas y registros. En síntesis, podemos decir que se siguieron las metodologías SA/SD e "Information engineering", para las fases de Análisis y Diseño; alimentadas de conceptos de una metodología organizativa propia; bajo una filosofía de desarrollo en cascada, con metodologías específicas en la construcción de la base de datos, y en la prueba integral del sistema.

3. SECUENCIA DE DESARROLLO.

La secuencia de acontecimientos es la siguiente:

A partir que fue contratado el desarrollo del sistema, se inició con una etapa de análisis detallado - pues para la firma del contrato, ya se había realizado un análisis general y un diseño preliminar, esto con el fin de poder discutir los tiempos y los costos..

El análisis detallado, se realizó principalmente por medio de entrevistas con los usuarios del futuro sistema, se recopiló copia de todo tipo de documento concerniente al producto en cuestión, y se analizaron las entradas y salidas de información a la dirección de pensiones.

Las entrevistas estuvieron encaminadas a detectar la información que entraba y salía de la dirección de pensiones, y los procedimientos necesarios para procesar las entradas como salidas. Cuando se considero contar con la suficiente información, como para bosquejar un diagrama general de funciones, se realizó dicho diagrama, especificando sus entradas y salidas. se discutió este diagrama con el usuario.

Según se profundizaba con las entrevistas y el análisis de la información se fueron obteniendo modelos más aproximados al modelo final, iterando continuamente sobre la etapa de análisis y diseño. cuando el boceto final fue lo suficientemente detallado y se encontró aprobado por el usuario, se procedió a la elaboración de un documento, al que se le dio el nombre de "Documento de diseño", aunque en realidad era un documento de análisis y diseño general.

El documento contenía un diagrama de estructura modular del sistema, pero aún no describía a detalle como se desarrollarían dichos módulos, cuantos y cuales programas, que funciones se realizarían en línea y cuales por lotes; pero contenía una propuesta inicial sobre estándares de pantallas en línea, y alguna información sobre el diccionario de datos que daría pie a una base de datos.

Como los tiempos y costos fueron predeterminados con un análisis poco exhaustivo, cuando se desarrollo al análisis a profundidad, se detectó un aumento en la complejidad y tamaño del sistema pero este se tuvo

que dimensionar a los límites preestablecidos por el contrato. Se podría pensar en una revisión del contrato una vez finaliza la fase de análisis y diseño general, o en su defecto estimar costos y tiempos por separado para cada uno de los módulos principales del sistema, con el fin de reducir los riesgos del equipo de desarrollo en cuanto a la subestimación del trabajo.

Si se evaluase por separado cada uno de los módulos, estaríamos entrando a una filosofía de desarrollo creciente, que creemos muy conveniente en casos de incertidumbre, pues en estos modelos se trabaja con módulos de forma independiente, se evalúan costos, se desarrolla, se prueba y se libera por módulos, que son mucho más pequeños que el sistema que constituyen. Pero en nuestro caso eso no se realizó, y tuvimos que ajustar las dimensiones del sistema, a lo preestablecido en el contrato.

La participación del usuario fue intensa y continua durante toda la etapa de análisis y diseño general, sin embargo, casi no se requirió su intervención durante la etapa de desarrollo, salvo para aclarar dudas pendientes del producto, solicitar opinión del desarrollo del sistema, y dar seguimiento al calendario de desarrollo. Durante la etapa de construcción se le solicitó al usuario, elaborara una serie de pruebas predefinidas para probar el sistema, estas pruebas consisten del seguimiento de algunos casos que pudiesen presentarse en la vida real.

Los casos con los que se probara el sistema se relacionan en una lista en papel -un caso por renglón-, y la serie de eventos o acontecimientos que se probaran por caso se relacionan a manera de columnas, de tal manera que se forma una matriz de datos. A tales tipos de prueba se les conoce como "Matrices de prueba". Cabe mencionar, que el desarrollo de tales tipos de pruebas, (pruebas integrales), debe tenerse terminado antes del inicio de la etapa de construcción del sistema, pues son estos casos de prueba para los que debe ser diseñado y construido el sistema.

No aprobar estos casos de prueba, equivale a no aprobar el sistema. La matriz de prueba equivale por tanto, a la lista de especificaciones del sistema, si el sistema las cumple, el sistema se aprueba, de lo contrario se rechaza.

Mas la elaboración de tales casos de pruebas debe ser propuesto por el analista del sistema, pues no se puede esperar a que el usuario pueda plantear tales pruebas, pues no es el, el especialista en manejo de información. Por lo que consideramos que la elaboración de casos de prueba debe correr a cargo del analista y/o administrador del proyecto y no del usuario.

Tampoco el programador debe de plantear tales pruebas, pues existe el riesgo a que pruebe solo lo que el cree que deba probarse, dejando descubierto lo que se da por obvio. Sin embargo no se puede ignorar el hecho de que el especialista en el producto administrativo es el usuario y no el analista de sistemas, por lo que esta matriz debe ser validada y complementada por el usuario.

En nuestro caso, la elaboración de la matriz de pruebas corrió a cargo del usuario, cometándose el doble error de quitarle demasiado tiempo al usuario para que la elaborase, y que la matriz no estuvo lista sino, hasta el final de la etapa de construcción del sistema, cuando ya era demasiado tarde para modificarlo para que cumpliera con todas las pruebas.

Esto último significó muchos retrasos en la etapa de pruebas, pues no solo se probó el sistema y ajustaron detalles, sino que en ocasiones se tuvieron que reelaborar módulos completos. Desde el inicio del proyecto, hasta su terminación, se llevo un control de avance por medio de gráficas Gantt, mismas que se presentaban a la dirección de sistemas de la compañía y servían para mantener informados a los usuarios del avance del sistema. Dichas gráficas Gantt, por otro lado, no sirvieron mucho para controlar el proyecto, más bien, fueron de gran utilidad para informar de su avance.

Se llevó una bitácora formal de pendientes, que sirvió mucho para organizar las actividades de desarrollo, dicha lista de pendientes nació como algo informal y de uso personal, y en el transcurso de desarrollo,

evoluciono a convertirse en el plan general de desarrollo, desde donde se construían las gráficas Gantt con las que se notificaba el avance del sistema.

La lista de pendientes contenía la actividad elemental a desarrollar, la fecha de inicio y término, el responsable y observaciones. dicha lista era dinámica, es decir, se construía durante el desarrollo del sistema, ya que desde un inicio es casi imposible determinar todas las actividades elementales que se deben realizar.

El avance del sistema se evaluaba en juntas semanales que se llevaban a cabo con la dirección de sistemas, se notificaba avance, se comprometían futuros avances, se investigaba causas de problemas o retrasos. Estas juntas se iniciaron con la presencia del usuario pero en cuanto inicio la etapa de construcción del sistema, se ausentó, haciéndose presente hasta que se iniciaron las pruebas integrales del sistema.

La fase de construcción del sistema, se subdividió en cuatro etapas:

1. Construcción de la base de datos.

Etapas que consumió mucho tiempo, por ser, la base de datos, la estructura medular del sistema, y quien carga con la responsabilidad del registro y administración de grandes cantidades monetarias. se siguió una metodología para la construcción de la base de datos la cual consistió de los siguientes puntos:

Elaboración del diccionario de datos.

Determinación de las entidades, atributos y relaciones

Normalización

Bosquejos repetidos de la base de datos hasta su aprobación por la gerencia de bases de datos.

Codificación de la base de datos.

Pruebas de acceso.

2. Desarrollo del módulo de estudios en PC.

El módulo de estudios sirvió como un primer prototipo del SPA. Se eligió desarrollar este módulo primero, debido a la alta velocidad en que pueden ser elaborados los programas en una micro-computadora y la alta flexibilidad para modificarlos. De esta manera se pudo poner en contacto al usuario con el sistema y así obtuvimos una temprana retroalimentación.

3. Desarrollo de la parte "en línea" del sistema.

Esta etapa consiste en la elaboración de los programas y pantallas que se usaran en tiempo real (en línea). Primero se desarrollo un prototipo (segundo prototipo), que solo contenía las vistas de las pantallas, y un leve bosquejo de su funcionamiento. se presentó al usuario, para que diera su opinión, a partir de esta presentación el usuario planteo nuevas necesidades y requerimientos cambios en los formatos de las pantallas y hasta en el funcionamiento

Una vez aprobadas las pantallas básicas del sistema, se procedió a la formalización de los programas en línea, se documento el código fuente que no estaba documentado, se identificaron rutinas que podían ser comunes y se unificaron en rutinas generales (normalización de procesos), se instrumentó la validación de acceso a las funciones, por medio de códigos de seguridad, se añadieron validaciones de datos y registro de todo movimiento en una bitácora general del sistema.

En este momento también se inicio la construcción del programa que realiza el cálculo de las rentas, programa que requirió de mucho esfuerzo. este programa es un punto crítico del sistema, y es el corazón de toda la parte en línea, se elaboro pensando que pudiese correr en forma batch también. primero se codificó

en PL/I, pensando en optimizar el rendimiento del sistema, pero después se tradujo a cobol por restricciones de la instalación del centro de computo.

4. Desarrollo de la parte "por lotes" del sistema.

En el mismo momento de iniciar la construcción de los programas de procesos por lotes, se inició la prueba integral de la parte en línea esto no es muy recomendable, ya que se empieza a recibir observaciones para realizar cambios en la parte en línea, cuando se esta ocupado en el desarrollo de la parte batch, esto genera listas de pendientes impresionantes, que son muy riesgosas de manejar.

En nuestro caso se inicio la prueba en línea debido a restricciones de tiempo, más sin embargo, a la larga esto provoco más retrasos, que si se hubiera esperado a realizar la prueba integral hasta que se tuviese terminado todo el sistema.

El desarrollo de la parte batch, involucro a todos los reportes, impresiones de cartas y documentos, interfaces a otros sistemas, pero principalmente se ataco el pago de las rentas o pensiones, ya que este proceso es el corazón de la parte batch del sistema y corazón del sistema en general.

Tal proceso se modularizo lo más posible para repartir la carga de proceso entre varios programas y evitar así la generación de un macroprograma de lógica intrincada y de alto riesgo para el sistema de software.

El detalle del desarrollo de cada una de estas etapas se describe en el capítulo "construcción del sistema" Cabe mencionar que el desarrollo se realizó dentro de un entorno de operación ficticio, con archivos y bibliotecas de pruebas, y en el momento de proceder con la liberación, se realiza una migración de todo el sistema, a bibliotecas y archivos reales - o productivos - se inicializa el ambiente y se prueba su correcta operación, este proceso aunque parezca trivial, es muy crítico, pues el primer momento en que el sistema, empieza a afectar información real, y hasta entonces se puede medir el impacto al sistema de computo y a los demás sistemas que comparten ambiente de producción.

Una vez terminada la construcción del sistema, se procedió a probar de nuevo toda la parte en línea, pero ahora complementándose la prueba con la prueba de la parte batch, la parte en línea había sufrido algunas adecuaciones, serías en algunos casos, y podemos mencionar que la parte en línea recibió un segunda y hasta tercera re-codificación debido al alto indice de solicitudes de cambio, por ello se opto por tomar posiciones rígidas en cuanto al control de cambios al sistema.

De hecho, es recomendable llevar un sistema formal de control de cambios, y se debe considerar en que momento realizar un cambio, y en que momento regresar a tras hacia alguna etapa de diseño o de análisis. Un control de cambios garantiza cierta estabilidad al sistema, y por lo tanto, minimiza errores y posibilita entregar resultados - aunque sea erróneos - de forma más frecuente.

En la prueba integral, el usuario fue auxiliado por personal de sistemas de la compañía, quienes capturaban la información al sistema simulando casos reales y solicitaban los procesos batch al equipo de desarrollo.

Durante las pruebas integrales se depuraron errores estéticos, de lógica y en ocasiones se requirió de rediseño. sin embargo a pesar de la ardua labor de detección de errores y replanteamiento de procesos, se satisfizo por completo la matriz de prueba, dándole el usuario el visto bueno al sistema para que se procediera con su liberación.

En este momento se inicio la etapa de liberación del sistema, misma que comprendió la presentación del mismo a las áreas involucradas de la compañía, la elaboración del manual de usuario, la elaboración del manual técnico, la estandarización de los procesos por lotes y la coordinación para la puesta en marcha del sistema en producción.

1. ANALISIS PRELIMINAR DE LA PROBLEMATICA BAJO UN ENTORNO REAL.**1.1. LA ETAPA DEL ANALISIS PRELIMINAR DE SISTEMAS.****1.2. LA INVESTIGACION Y ANALISIS INICIAL DE LA PROBLEMATICA PARA EL SISTEMA SPA.****2. PLANEACION DEL SISTEMA A DESARROLLAR****2.1. LA FASE DE PLANEACION DE SISTEMAS.****2.2. PLANEACION DEL SISTEMA DE PENSIONES ASEGURADAS. (EL PLAN DEL PROYECTO).**

1. ANALISIS PRELIMINAR, REFLEJO DEL ENTORNO REAL DEL SISTEMA A DESARROLLAR.

1.1. LA ETAPA DEL ANALISIS PRELIMINAR DE SISTEMAS.

Para especificar o definir un problema, no basta con tener la información suficiente sobre lo que se necesita, es necesario contar con un conocimiento básico del área de aplicación, así como del entorno en que gira e influye en dicha aplicación. Claro que este conocimiento no se obtiene rápidamente, es todo un proceso de aprendizaje, que en esta fase (el análisis de sistemas de información) intervienen en dicho proceso, por una parte todo aquel conocimiento que dice algo del problema, obtenida por diversas fuentes (los usuarios de diversos niveles, documentos, sistemas y subsistemas existentes, y las relaciones entre estos) y por otra el analista de sistemas, que asimilará, y esquematizará los resultados obtenidos en esta etapa. ¿Cómo se podría pensar en la solución de un problema sin saber de que trata este?, ¿porqué ocurre?, ¿a quienes afecta?, y tantos cuestionamientos mas.

Al dar cabida a todas las posibles relaciones de adquisición y análisis de información, el analista de sistemas podrá sentar las bases para establecer las consideraciones a la problemática, al sistema, al proyecto y hasta de los demás integrantes del equipo de desarrollo. No olvidar que para todo desarrollo de un sistema, bajo cualquier metodología, son necesarias el siguiente personal clave:

El usuario del sistema:

Responsable(s) del sistema de información requerido que cuenta(n) con conocimiento de las especificaciones de este y de la problemática actual. Se considera como la mas confiable interfase de un producto a desarrollar.

El analista de sistemas:

Es el primer pilar que intervendrá (como todo modelo de desarrollo lo establece) por parte del equipo desarrollador y entre otras funciones, se encuentran; la determinación de la problemática, la justificación de un sistema de información en base al análisis de las necesidades del usuario, en la propuesta de una alternativa de solución, documentar formalmente toda la gama de requerimientos del sistema, y hasta puede llegar a definir un modelo conceptual del nuevo sistema. En sistemas grandes, es necesario establecer un grupo de analistas y programadores que por medio de un líder de proyecto (analista senior), agrupe todas las vistas divergentes de dicho grupo.

El diseñador de sistemas:

Realiza toda labor que establezca por medio de una metodología de diseño las diferentes facetas que involucran el nuevo software y la forma en que funcionarán y se interconectarán sus componentes. La tendencia en el diseño es la creación de un equipo de diseñadores (siempre que las extensiones del sistema lo ameriten) que a través de un coordinador logren definir estándares, procedimientos y esquemas de tiempo, y se asignen los módulos a desarrollar bajo los objetivos planteados.

El instrumentador de sistemas:

Realiza a través de una metodología la construcción del sistema requerido en base al diseño y especificaciones que marco el diseñador. En la investigación inicial, se tiene el primer contacto que tienen

el modo usuario del sistema (personal de la organización que requiere del sistema que resuelva sus necesidades) y el analista de sistemas (ubicador de problemas, manejador de estrategias, organizador de las estructuras de trabajo, controlador de dinámicas de trabajo, generador de alternativas, etc.) así mismo, en esta etapa se da el primer paso a la ubicación en el entorno real del sistema. El analista, desde este momento debe trabajar la idea de lograr una relación sólida con el usuario, pues es este el que realmente dará la descripción de la problemática, de la especificación de requerimientos, y de lo que debe resolver el sistema.

Generalmente solo un usuario experimentado (con participaciones en otros proyectos de desarrollo, con conocimientos de computo, con cultura computacional, y sobre todo con esa disposición al cambio) podrá aportar todo lo anterior.

Entre los resultados obtenidos por dicha investigación se pueden puntualizar:

- a) La problemática que se presenta
- b) Los recursos involucrados en la resolución de la problemática.
- c) Establecer el entorno del sistema.
- d) La identificación del usuario y los niveles de comunicación con este.
- e) Las condiciones de trabajo del equipo de desarrollo.
- f) Limitaciones y alcances del sistema a desarrollar.
- g) Establecer las funciones básicas que deben ser implementadas en el sistema a desarrollar.
- h) Obtención de resultados de factibilidad del proyecto a desarrollar.

La etapa de investigación inicial, no debe ser precaria, se basa en un análisis preliminar realizado metódicamente, que sentará las bases para lograr esas alternativas de solución, que más que perfectas deben ser posibles.

Ahora bien, este análisis preliminar no debe ser robusto, (que tome tiempo valioso para la presentación del plan de proyecto, crítico en todo concurso, negociación, o política de empresa), pero mucho menos trivial, pues en base a sus conclusiones, se definirá cuestiones genéricas, pero fundamentales (expectativas del usuario y del equipo de desarrollo) como son; si realmente se necesita un sistema de información, o solo se requiere dar mantenimiento a uno existente, si se puede realizar dentro de costos y tiempos aceptables, la esquematización del plan de trabajo en general, etc.

Un objetivo que se cumple para toda fase de análisis de sistemas es que esta debe establecer el QUE, y no el COMO; la fase posterior, - el diseño del producto-, se refiere al como.

El documento de análisis al que se llega como resultado preliminar de esta etapa, permite :

- a) Registrar documentos valiosos obtenidos a lo largo de esta fase de análisis
- b) Delimitar el problema, lo que posteriormente, al término del plan de desarrollo, permitirá descartar posibles alternativas de solución.
- c) Enumerar las funciones básicas que deben ser implementadas.
- d) Enunciar los resultados de factibilidad del proyecto a desarrollar.

Descriptivamente este documento de análisis preliminar debe contener lo siguiente : Como fue realizado dicho análisis; la especificación de requerimientos preliminares de usuarios, las restricciones del sistema y la documentación obtenida (que se considere clave para la realización de la metodología de desarrollo).

1.2. LA INVESTIGACION Y ANALISIS INICIAL DE LA PROBLEMATICA PARA EL SISTEMA SPA.

El producto a administrar, Producto Pensiones Aseguradas (PPA), por el sistema computacional a desarrollar nos lo presentaron como 'nuevo' en el mercado, es decir ninguna empresa disponía de un producto semejante, nunca se había presentado en el área un producto parecido, y no existían sistemas computacionales que lo administraran.

Esta peculiaridad de 'nuevo', se impondría y generalizarla, se tratará mas adelante, algunas complicaciones, ya que el producto financiero paso por, 'intervalos de autodefinition', ya que ciertas características inherentes a su descripción y funcionamiento no estaban del todo claras para los mismos usuarios. El área de conocimiento que involucraba el dominio del PPA era asombrosamente amplia, pues estaba delimitada, entre dos grandes campos el de Seguros y el de Pensiones, empero se requería de cierto conocimiento de Actuaría, Contabilidad, Administración y Evaluación financiera.

Básicamente el problema a resolver, consistía en que ante esta iniciativa de colocar un nuevo producto de pensiones en el mercado nacional se necesitaría de una fuerte infraestructura administradora que pudieran responder a todos sus procedimientos, la cual no debiese ser, por procedimientos manuales, ya que se contaba con muy poco personal para ello.

Por lo que se requería, de un sistema de información que automatizará, la administración del producto, definiendo un sistema "hecho a la medida", como la alternativa computacional factible. Entre los recursos inicialmente disponibles para la resolución de la problemática, (que darían un marco técnico inicial) tenemos :

Recursos en Hardware :

- Equipo IBM 3090 / XA modelo
- Terminales 3278
- Discos 3380
- Cintas 3480
- Controladores 3475
- Una Red de comunicaciones e interfaces...
- Micros

Recursos en Software :

Sistema operativo MVS (Multiple Virtual Storage) / SP

Base de datos IDMS

Lenguajes de programación disponibles :

- Cobol
- PL/1.
- ADS
- Pascal

Recursos Humanos :

- Personal del área de pensiones.
- Personal del área de sistemas.

No existía un sistema que administrará un producto como este, esto influyó directamente en la dinámica de trabajo a la que nos enfrentaríamos tanto el usuario, como nosotros, y mas aún por el tipo, magnitud y complejidad que presentaba el sistema. Por parte del negociador del sistema se definió que el usuario con el que íbamos a colaborar, era la dirección de pensiones de la compañía donde básicamente trabajaríamos inicialmente con tres personas, el director de pensiones y dos de sus colaboradores.

El nivel (skill) de usuario, (consideración de su experiencia) en desarrollos de productos (en equipo multiusuario) a su cargo, realmente era inexistente. Los efectos que podría acarrear esto a lo largo del sistema, no se les dio la debida importancia. En general, consideramos que este punto, el nivel de usuario, (estratégico para todo desarrollo) no estuvo bien definido, la consistencia de los usuarios afectaría grandemente (se produjo movilidad en los puestos de los usuarios, dejando ciertas lagunas de conocimiento). Por ejemplo, el nivel de comunicación, con los usuario, en un principio fue directa, posterior mente intervino un personal intermediario, (personal piloto de la dirección de sistemas de la cia. aseguradora) lo que acarreo uno de los grandes problemas que enfrentó el desarrollo del sistema, por pérdida y triangulación de la información.

Por lo que toca a la otra forma de obtención de conocimiento, la documentativa tenemos: La documentación obtenida para esta etapa.(y posteriores) que realmente fuera representativa y formalmente fue escasa, de lo que se rescata el documento (inicialmente maestro) llamado -Nota Técnica-, a partir del cual se generaría, -la carpeta de desarrollo-, crucial en todo el desarrollo del sistema. Este documento describía el producto desde una perspectiva formal, pero muy general, es decir solo daba a conocer ciertos ámbitos del producto.

Este documento no nos ofrecía una definición de requerimientos, tan solo daba una idea de lo que era este producto (a manera de presentación): definía terminología del área (Pensiones y Seguros), mostraba los componentes mas importantes del producto, mencionaba ciertas posibilidades y alcances iniciales del producto, y aunque contenía las ecuaciones actuariales que definían matemáticamente el producto, no lo definía totalmente, pues no explicaba (no en lenguaje técnico) los componentes de este, mucho menos sus procedimientos fases, lo que se esperaba de el, etc.

Respecto a la existencia de procedimientos automatizados del producto, no había más que las clásicas hojas de cálculo (con macros enormes) y una base de datos plana. Fuera de esto no había nada, explicaciones poco claras, en lenguaje natural de los componentes del sistema, sin sus relaciones, ni los ingredientes para administrarlos, etc.

Lo primero que se desarrolló para el SPA, fue el llamado módulo de Estudio, que en base a un manejador de base de datos permitió realizar las funciones de un subcomponente del producto, - el cálculo de proyecciones-, que representarían solo una parte del módulo de emisión. Así mismo este módulo desarrollado, serviría para dar cabida al primer resultado por parte del equipo de desarrollo el cual lo podemos equiparar como una peculiaridad del desarrollo a manera de prototipo.

Las condiciones de trabajo del equipo de desarrollo fue el esquema que no cambió en todo el desarrollo y se definieron desde el inicio como es, el utilizar las instalaciones de la empresa, concretamente en el área de la dirección de sistemas, donde todos los recursos necesarios serían proporcionados por la misma.

Finalmente los alcances del proyecto, eran difíciles de determinar, no se tenía una idea clara de hasta donde llegaría la automatización y mucho menos las interfases, ante todo lo que había que aprender y asimilar fue imposible, para esta etapa de arranque aislar estas dos importantes consideraciones. Todo quedaría para establecerlo después de la espera ansiosa de la aprobación del proyecto.

2. PLANEACION DEL SISTEMA A DESARROLLAR

2.1. LA FASE DE PLANEACION DE SISTEMAS.

Antes de planear algo, se debe conocer que es ese algo, es decir, se requiere de un análisis preliminar de la problemática. Por ello la planeación de sistemas ocupa la segunda posición, en el flujo lógico de todos los modelos de desarrollo. Se debe reconocer que los planes preliminares se modifican según va evolucionando el producto, por ello se debe planear pensando en el cambio. Los productos terminados de la fase de planeación, incluyen: "Definición del producto" y el "Plan del proyecto". Aunque el primero de ellos se puede realizar, en la fase de análisis preliminar del sistema e incluirlo, si se desea, en la documentación fuente para la fase planeativa. En esta cuestión de la documentación, notaremos que siempre habrá una relatividad, en gran medida por la idea de que todo buen desarrollo debe generar documentación que pueda seguir un ciclo que hemos llamado 'ciclo de modificabilidad', es decir un documento maestro (como lo es el plan de desarrollo o el de análisis formal de requerimientos, ó en nuestro caso, la carpeta de desarrollo) se actualizará, cada vez que la circunstancia lo amerite. Tal como lo muestra el esquema (emodi).

La definición de un producto debe incluir los siguientes componentes ó tópicos : (Físicamente mostrados en el documento 'Definición del producto')

Una descripción concisa del problema.

La propuesta de los objetivos del sistema y del proyecto.

Identificación de las características del usuario.

Una proposición de las funciones que el sistema realizará.

La estrategia de solución.

Las prioridades para las partes y características del producto.

Los criterios para la aceptación del sistema

Mientras que la fase planeativa debe centrarse para la obtención de los siguientes tópicos: (Físicamente mostrados en el documento 'Plan de Proyecto') :

El modelo de ciclo de vida que estará involucrado en el desarrollo.

La estructura de la organización.

Estimaciones preliminares de requerimientos de personal y recursos.

Una evaluación preliminar del costo.

Un calendario preliminar de desarrollo.

Las especificaciones para el monitoreo del proyecto y sus mecanismos de control. Un plan de proyecto óptimo debe adaptarse a las necesidades de los usuarios que desean tener como herramienta clave de trabajo, dejando en segundo término a los recursos disponibles. Lamentablemente, hasta ahora, conocemos a pocos proyectos que no estén superitados por un equipo, y costos predeterminados, dejando a la deriva esos requerimientos de los usuarios.

En general, podemos sintetizar la metodología que debe seguirse, para realizar la fase planeativa en el desarrollo de un sistema, por los siguientes puntos :

1. Desarrollar un enunciado definitivo del problema para resolver. Incluir una descripción de la situación actual, restricciones del problema y de las metas, el enunciado del problema debe realizarse empleando terminología del cliente.
2. Justificar una estrategia de solución computarizada para el problema.

-
3. Identificar las funciones por realizar, las restricciones y los subsistemas a emplear (Hardware, Software, documentación y personal).
 4. Establecer los criterios de aceptación para el sistema.

2.2. PLANEACION DEL SISTEMA DE PENSIONES ASEGURADAS. (EL PLAN DEL PROYECTO).

El plan de proyecto entregado, para el desarrollo del sistema SPA no estableció totalmente los que debiese incluir formalmente (como se mencionó en el punto anterior) descuidando cuestiones tan importantes como los criterios de aceptación del proyecto.

No se aclaró del todo la forma de monitorear y controlar el proyecto; sin embargo, se trataron los tópicos suficientes para formalizar el análisis, si se monitoreo el proyecto, se justificó la alternativa propuesta y hasta se acreditó formalmente al equipo de desarrollo. Es decir, a pesar de que el plan tendía a ser poco profundo, la negociación del proyecto concluyó que era suficiente para iniciar, por parte del equipo de desarrollo, el proyecto SPA. A continuación se muestran las consideraciones que se trataron en esta fase planeativa del sistema SPA, algunas de ellas plasmadas en el Plan de trabajo.

1) Problema a resolver:

"Esta por lanzarse al mercado un nuevo producto de pensiones, requerirá de una fuerte infraestructura administrativa, y se cuneta con muy poco personal para soportarla".

2) Situación actual:

La compañía aseguradora que lanzará el producto, cuenta en la actualidad con varios productos de seguros, cada uno con su respectivo sistema computarizado de información, y pretende manejar bajo la misma filosofía el nuevo sistema de pensiones.

Por otro lado, el producto de pensiones, presenta la característica de que es totalmente nuevo en la compañía y en el mercado nacional. Prácticamente no existe nadie que sepa sobre como manejar el negocio, y no existe ningún sistema de información que maneje pensiones en nuestro país. Sin embargo, en la compañía existen algunas personas de la reciente creada dirección de pensiones, que conocen teóricamente

sobre el negocio, y también existe un documento conocido como "Nota técnica" que describe (a nivel venta) formalmente al producto.

3) Objetivo a alcanzar:

Preparar a la compañía, para que se encuentre totalmente capacitada para manejar la administración del negocio de pensiones, en cuanto este sea puesto en marcha en el mercado.

4) Justificación de una solución computarizada para el problema:

La compañía cuenta con los suficientes recursos de cómputo y recursos asociados, como para implementar un sistema de administración computarizada, al PPA. Bajo la investigación inicial se concluye que las funciones que una administración de pensiones debiese cumplir, son totalmente computables. El tiempo y costo que representa la implementación de un sistema de pensiones, es razonable con respecto a los planes que tiene la compañía para el desarrollo de su producto de pensiones.

5) Funciones a realizar:

Dentro de las funciones a realizar en el sistema, se encuentran las siguientes: Estudios actuariales, emisión de nuevos negocios, cálculo de pensiones, Impresión de documentos actuariales, cambios y consultas, proceso de defunciones, compra de primas adicionales, retiros parciales, reportes, estadísticas, valuación de reservas, interfase a contabilidad y a sistemas varios.

6) Hardware y software disponible:

Se dispone de una computadora IBM MainFrame, bajo un sistema operativo MVS/XA, soportando a una red de terminales, que abarca toda la república. Se dispone de IDMS que es un manejador de base de datos para el desarrollo acelerado de aplicaciones.

7) Criterios de aceptación para el sistema:

Por desgracia, los criterios para la aceptación del sistema, no fueron claramente definidos, y esto nos llevo a enfrentar varios conflictos en el desarrollo posterior del sistema, sin embargo se pudieron superar todos ellos, concluyendo el proyecto de manera exitosa.

Entre los criterios de aceptación que se debieron haber fijado, se encuentran los siguientes:

- a) Metas cualitativas para el proceso de construcción: El sistema debe ser realizado por un equipo profesional siguiendo metodologías de alto nivel y suficiente documentación.
- b) Metas cuantitativas para el proceso: El sistema debe ser entregado en un plazo no mayor a 12 meses.
- c) Metas cualitativas del producto: El sistema debe ser amigable y fácil de usar.
- d) Metas cuantitativas del producto: El sistema debe permitir la emisión de nuevas pólizas en máximo un día de retraso. Los criterios de aceptación se deben especificar de tal manera que puedan verificarse. Un sistema es funcionalmente correcto si es que realiza lo que se espera que realice con las entradas y operaciones predeterminadas. Por lo que se requiere la existencia de un documento de criterios de aceptación. Dichos criterios deben ser detallados y explícitos. En el caso de SPA, la "Matriz de prueba" pudo haber servido de este documento.

El problema que enfrentamos, fue que tuvimos disponible dicho documento hasta que se concluyo con la etapa de construcción. No es correcto que una vez terminado y afinado el sistema, se establezcan una serie de casos de prueba que no estaban considerados (tal como nos sucedió en SPA), pues simplemente el sistema no fue diseñado para manejarlos, en estos casos se debe insistir en una administración de proyectos de alto nivel.

Si nuevas consideraciones aparecen una vez iniciado el desarrollo del sistema, deben ser controladas a través de un rígido control de cambio, y si la ocasión lo amerita, prorrogar todas estas nuevas consideraciones para una nueva versión del producto.

Otro punto que vale la pena mencionar, es la estimación de los costos que bien sabemos es uno de los tópicos de negociación más delicados. Existen varios métodos para la estimación del costo de software, siendo uno de los más usados el método COCOMO (COSt COnstrucción MOdel), sin embargo en el caso de SPA, el costo se encontraba preestablecido, habiéndose negociado inicialmente con parámetros de esfuerzo al estilo de este método, pero se adaptó a los recursos de la empresa y a los concursos que se llevaron a cabo para la contratación del desarrollo.

De esta manera, realmente no se estimó el costo del proyecto, sino en su lugar, se definió un presupuesto disponible, y en base a él se estimó el alcance del proyecto. Esto es un gran mal que prevalece en el mercado nacional, pues se pierde el objetivo, que es dar soluciones reales al usuario. Para contrarrestar este fenómeno, se requiere de una intensa educación sobre lo que es la Ingeniería de Software.

La estimación de costos de un producto de programación en el mercado nacional, es quizás de los problemas más impactantes en el desarrollo que ha tenido la ingeniería de software en el país, pues aunado a la forma tan informal en que se definen los presupuestos para invertir en software, el costo de un proyecto de software muchas veces tiene que competir con soluciones prefabricadas, con costos dramáticamente menores, y que en ocasiones se encuentran disponibles bajo una economía informal.

1. LA FASE DE ANALISIS DE REQUERIMIENTOS DE SISTEMAS.**2. DESCRIPCION FORMAL DEL SISTEMA (SPA) REQUERIDO.****2.1. FLUJO DE OBTENCION DE LAS ESPECIFICACIONES PARA****2.1.1. REVISION Y COMPLEMENTACION DE LAS EXPECTATIVAS DEL USUARIO****2.1.2. PARAMETRIZACION DEL SISTEMA (SPA)****2.2. LA ESPECIFICACION FORMAL DE REQUERIMIENTOS DEL SPA.**

1. LA FASE DE ANÁLISIS DE REQUERIMIENTOS DE SISTEMAS.

Como se ha señalado, lo que propone todo análisis estructurado es determinar claramente y de manera formal (documento formal de análisis) lo que el sistema debe realizar, antes de involucrarse con la fase operativa, es decir, el como debe cumplir con los objetivos planteados.

Esta fase de análisis profundo, debe seguir con esta pretención, continuando con el direccionamiento del problema, para que el sistema a desarrollar fluya de su más primitiva esencia, al esquema formal de especificación que permita :

- El análisis se acerque más a los requerimientos del usuario final por estar descrito de una manera técnica comprensible, es decir con terminología informática (propia para que el documento trascienda a niveles del personal de sistemas, diseñadores, DBA's, programadores, etc.)
- El análisis pueda disponer de mayores y más creativas alternativas de solución.

Para esta etapa es necesaria una gran interrelación entre el personal de sistemas y la comunidad de usuarios finales, para de esta forma, a la vez de mejorar los niveles de comunicación, se completen o corrijan toda la gama de especificaciones del sistema, para que el ingeniero de software formalice técnicamente tanto el entorno real del sistema, como el sistema mismo a desarrollar, en función de las necesidades del usuario y los recursos disponibles; de ahí que se le llame a esta etapa como un 'acto de balanceo'.

Todas aquellas consideraciones y requerimientos obtenidos en la fases de análisis preliminar y planeación, son transformados a un conjunto explícito de especificaciones detalladas, puestas en un lenguaje técnico, en el subproducto final de esta fase; el documento de análisis formal de requerimientos.

Los objetivos primordiales de la etapa de análisis de requerimientos son :

Transformar los requerimientos, analizados durante la etapa previa, a planeativa, en una forma precisa que este orientada a las necesidades de los ingenieros de software

Determinar el sistema de información deseado o ideal. incluyendo la especificación de objetivos.

Parametrizar el sistema ideal(deseado) en un rango realista al clarificar las necesidades de los usuarios e identificar las restricciones en los recursos(físicos y lógicos) disponibles para el desarrollo.

La transformación de requerimientos implica la traslación de los requerimientos plasmados en el plan de proyecto(vision de usuario) a una terminología informática, más precisa y reutilizable por el equipo de desarrollo. Esta transformación de requerimientos, dará una nueva perspectiva del análisis llevado hasta ahora, que podrá deducir nuevas alternativas, por ejemplo :

El determinar que no necesariamente el modelo de datos deba seguir una base de datos jerárquica, que gracias al estudio de las especificaciones con notaciones técnicas, pueda establecerse que también puede realizarse con una base de datos tipo red. O que los lenguajes a utilizar se extiendan a cualquiera que pueda realizar entradas y salidas paramétricas de archivos.

Por parte del equipo de desarrollo se necesita de una absorción y comprensión del rubro de conocimiento (pensiones, seguros) al grado de poder definir claramente el producto y sus características, sin dejarse llevar por inclinaciones personalistas que repercutan en dicha definición.

Por parte de los usuarios se necesita una disponibilidad alta, así como una completa dominación de dicho rubro, ya que ellos son realmente quienes externarán la información necesaria para obtener la descripción del sistema a desarrollar.

Por lo que toca a los resultados que debe procurarse obtener por la especificación formal de requisitos, tenemos :

- Resumen y panorama del producto. (Sin olvidar los objetivos del sistema)
- Ambiente de desarrollo, operación y mantenimiento.
- Interfases externas y flujo de datos. (DFD's)
- Requisitos funcionales.
- Requisitos de operación. (Reportes a generar por el sistema)
- Manejo de excepciones.
- Modificaciones y mejoras previsibles.
- Criterios de aceptación.
- Glosario de términos.

Para lograr todo lo anterior, es necesario tener en primera instancia, la metodología a seguir, la cual proponga una amplia gama de herramientas a utilizar, que eliminen la realización de todo ese trabajo inaprovechable que se generaría de no hacerlo así.

Una herramienta empleada para esta fase, es el diagrama de flujo de datos. (DFD).

Los diagramas de flujo de datos especifican las fuentes y destinos de los datos, sus almacenamientos, transformaciones, y los flujos entre ellos. Un almacén de datos es una estructura conceptual, en el sentido de que no se consideran detalles de la instrumentación. A diferencia de los diagramas de flujo, los datos no comprenden estructuras de decisión o detalles algorítmicos.

Las características de operación se deben establecer en términos verificables, así como los métodos para verificarlos. Esto como ya se discutió, es de suma importancia para el éxito en el desarrollo de un sistema de software.

En la medida en que estos criterios se encuentren ambiguos, en esa misma medida se desarrollara un sistema alejado de las necesidades de los usuarios, y no podran concretarse acuerdos sobre la terminación del proyecto, pues para unos este se encuentra terminado, para otros aun ni ha sido empezado.

Es una medida inteligente el reconocer que no se podrá tener un análisis total de la problemática en una sola etapa, y que para poder comprender suficientemente un problema es necesario estar involucrado en etapas de diseño y construcción.

Por ello el análisis debe considerar el desarrollo de un prototipo inicial, y dar margen a la inclusión de nuevos conceptos y eventos para una segunda etapa de análisis.

Existen notaciones formales para la especificación de requerimientos, y herramientas para emplear estas notaciones. Por ejemplo se pueden utilizar notaciones relacionales (axiomas, expresiones regulares, ecuaciones implícitas, etc.) o notaciones de estados (maquina de estados finitos, tablas de decision, redes de petri, etc.). Cabe señalar que el especificar requerimientos a través de notaciones formales tiene como objetivo facilitar las funciones a realizar en la fase de diseño, no en complicarlas.

En el sistema de pensiones, estas especificaciones debimos reconocer no fueron lo suficientemente claras y completas, y por lo mismo fue este punto quizás el principal problema que se tuvo en las fases de diseño y posteriores.

A pesar de ello, se establecieron algunos requerimientos de forma matemática, como lo fueron las fórmulas para el cálculo de las anualidades, cálculo de dividendos y valuaciones. Las cuales se mostraron en el siguiente punto (el documento de especificación de requerimientos). También fueron estos módulos los que menos modificaciones tuvieron y cubrieron las necesidades del usuario más fielmente.

El subproducto obtenido en la etapa de análisis de requerimientos es el documento llamado 'análisis formal de requerimientos'.

Entre los tópicos que debe puntualizar un documento formal de análisis de requerimientos se encuentran

- 1.- El método de realización u obtención del documento., y por ende de análisis profundo del sistema a desarrollar.
- 2.- Los requerimientos del usuario, descritos formalmente, en lenguaje técnico, comprensible por el personal de sistemas.
- 3.- Las restricciones del sistema que se detectaron en dicho análisis
- 4.- Inclusión de toda aquella documentación que se considere estratégica tanto para justificar el sistema, como para establecer procedimientos fundamentales del producto.

Finalmente, una peculiaridad importante de esta fase es que, al aplicar un mantenimiento a un sistema debe de alcanzarse hasta esta -el análisis- para así verificarse posibles cambios, que son los que de origen cambian los requerimientos de diseño, construcción u operación del sistema. Y no aplicar el ciclo incorrecto que va de la fase de mantenimiento a la de operación. Como lo muestra el siguiente esquema (EMANT)

2.1. FLUJO DE OBTENCION DE LAS ESPECIFICACIONES PARA EL SISTEMA.

2.1.1. REVISION Y COMPLEMENTACION DE LAS EXPECTATIVAS DEL USUARIO

El primer paso llevado a cabo en esta etapa de análisis de requerimientos fué el examinar y/o completar las expectativas del usuario por medio de diversos métodos como son :

a) Entrevistas

Este método es el más común y consiste simplemente en la siguiente serie de pasos :

- Seleccionar el grupo de personas (clave) a las que se les puede efectuar estas, (no por disponibilidad, sino por que aparecen, conocen o definen los componentes y procesos del producto.
- Elaborar un formato de estas donde las preguntas deben estar estructuradas.
- Realizar la entrevista, deben registrarse las respuestas y si es posible grabar la entrevista.
- Realizar una sinópsis de cada entrevista realizada.

Generalmente para un sistema ya existente, se puede entrevistar a mucho más gente. La relación de entrevistados puede obtenerse, por la correspondencia de los procesos representados en el DFD físico de ese sistema existente.

Cuando no existe un sistema previo, los entrevistados son los que definen, administran o conocen el producto, para el caso de SPA, el grupo de usuarios fue reducido y no homogéneo.

De esta manera, gracias a una conjunción lógica de ideas, se realizaron entrevistas al personal clave involucrado en el sistema como lo son la dirección de pensiones, la dirección de sistemas en sus departamentos de Base de Datos y Soporte Técnico, etc.

Sin embargo, este método no arrojó lo que se esperaba, ciertamente se tuvieron inconsistencias, como el no realizarlas de una forma metódica, el no disponer de un formato único, el no grabarlas; sin embargo, creemos que se produjeron pocos resultados, en gran medida, por las situaciones siguientes:

El frente de usuarios a entrevistar no era común, mejor dicho era inconsistente, pues alguno de ellos en cierto momento, dejarán su puesto, lo que produjo un truncamiento.

La información que tenían los usuarios del producto no era suficiente se producían lagunas en las respuestas.

El usuario recurría por información, a la dirección que se encargaba de definir actuarialmente al producto, (que por estar en definición cambiaba su descripción y componentes) luego al pasarnos la información surgían nuevas dudas, el usuario no estaba seguro de lo que había obtenido o simplemente no consultó correctamente.

Al principio no se llevaba una bitácora de los pendientes a resolver en cada nueva entrevista, tanto por nosotros, como por el usuario.

El usuario no tenía educación informática, y su grado de experiencia en el desarrollo de sistemas era muy bajo.

En general, podemos decir que la capacidad para realizar entrevistas fue mejorando, se consolidaron las buenas relaciones con el usuario, se mejoró la forma de realizarlas, se respetaba la secuencia lógica que seguían; empero sus resultados no fueron suficientes para definir formalmente el producto. Para fases posteriores en el proyecto serían estas las que sacarían algunas veces de apuros y otras provocarían fuertes contradicciones, (en lo dicho y entendido) atrasos y hasta redefiniciones.

Poco a poco se fue obteniendo la experiencia necesaria para detectar y evitar estas inconsistencias, en primera instancia se tomaron medidas organizativa, como fue el formalizar (lo dicho) y llevar una bitácora de todos los requerimientos (tanto nuevos, como ya establecidos) con los datos necesarios (entrevistado, fecha, temas tratados, puntualizaciones, etc.)

b) Análisis de procedimientos

Este método consiste en el análisis de todos aquellos procedimientos que se identifican en el diagrama de flujo de datos físico del sistema a desarrollar.

Además los procesos operativos son estudiados para identificar sus posibles entradas, conversiones y salidas. Luego estos son descritos, de lo que se puede llegar a obtener las distribuciones de volumen, y poder establecer fallas entre el documento actual de flujo y el procedimental de flujo.

Este método aplica sobre los documentos claves incluidos en el DFD del sistema y los procesos correspondientes.

Este método es conveniente cuando no tiene una idea completa o confianza en las funciones que debe realizar el sistema.

ATA TMO DE P
PLAN DE LA REVISIÓN

c) Inspección de documentos

Este método se refiere al análisis de documentos relacionados con el producto (directa e indirectamente) como pueden ser definiciones de este, análisis de mercado, reportes generados, salidas generadas

La forma de llevar a cabo la inspección se puede generalizar como sigue :

- Identificar los documentos llave en los procesos que llevará a cabo el sistema (El DFD físico)
- Verificar su existencia y si es el caso coleccionar cada uno de ellos.
- Análizar cada documento y obtener necesariamente cada uno de los datos relevantes como son : su función en el proceso, los campos que incluye, su frecuencia de uso, el tipo y tamaño,

etc.

Por estar ante un producto nuevo en el mercado, sin la presencia de un sistema computacional predefinido de este, la inspección de documentos fué el más importante. En primera instancia la información permitió obtener conocimiento del área de la aplicación (pensiones seguros) rápidamente. Luego se trataba de documentación formalmente registrada con la cual habla mayor respaldo.

A pesar de ello no se tuvieron tantos documentos como los necesarios para dar una fortaleza al análisis de un sistema tan grande como el de SPA. La nota técnica como, se mencionó anteriormente no disponía de la información significativa que describiera completamente, los componentes de este, mucho menos de criterios de administración y control del producto

La obtención del DFD lógico del SPA, se determinó tanto por la realización de entrevistas, como por la inspección de documentos llave, como son reportes, formas, cálculos, etc.

De estos documentos se tomaron los datos importantes como lo son:

- Campos de formato (tamaño y tipo)
- Frecuencia de uso
- Estructuras de datos

Lo que nos permitirá realizar una vista preliminar del diccionario de datos, que cambiaría a lo largo de todo el desarrollo y que sería uno de los pilares que sostendría el diseño y construcción del SPA.

Resulta importante confirmar que todos estos métodos logran determinar las expectativas del usuario, estableciendo lo que el usuario espera del sistema, lo que se debe satisfacer. Posteriormente el equipo desarrollador propondrá alternativas de solución o de desarrollo del sistema de información, que mas que ser perfectas, sean posibles.

2.1.2. PARAMETRIZACION DEL SISTEMA (SPA)

El siguiente paso de esta etapa, la parametrización del sistema ideal en un rango realista, se valió de todas aquellas restricciones para el desarrollo del producto. Estas básicamente se pueden dividir en restricciones técnicas :

- a) en los recursos físicos
- b) en los recursos lógicos

restricciones actuariales:

restricciones por factores externos :

a) interfases

El último paso realizado en esta etapa de análisis de requerimientos es la conjunción de los resultados para obtener la definición completa del producto a desarrollar descrita en el Documento de Análisis de Requerimientos. Tratado a continuación.

2.2 LA ESPECIFICACION FORMAL DE REQUERIMIENTOS DEL SPA.

La especificación formal de requerimientos comprendió, una especificación narrativa, la cual se muestra a continuación; y una especificación diagramática, de la que damos algunos ejemplos de los DFD's construidos, rdiagramas: (fig* 5.4, 5.5, 5.6, 5.7)

El plan de "Pensiones aseguradas" es un seguro que se ofrece como medio de apoyo a las personas que tengan constituido un fondo destinado a su jubilación, o como consecuencia de un plan de fallecimiento o invalidez. La compañía aseguradora se responsabiliza de la administración, operación y manejo del plan de pensiones, así como de efectuar los pagos de pensión en el lugar que indique el contratante o el asegurado mismo.

El seguro de pensiones aseguradas, se adquiere mediante el capital constituido por una persona en el momento que desee contratar una pensión. Dicho capital opera como una prima única, de la que depende el monto de la pensión que se otorgue.

DEFINICIONES.

Definición de terminos utilizados:

Compañía: Compañía aseguradora quien se encargará de pagar las pensiones correspondientes al asegurado, mediante la contratación de la póliza de "Pensiones Aseguradas".

Contratantes: Persona física o moral, que contrata la póliza de pensiones aseguradas y que es responsable del pago de la prima.

Asegurado: Persona física que mediante la póliza de pensiones aseguradas, recibirá el pago de la pensión, de acuerdo a las características convenidas con la Compañía.

El contratante de la póliza puede ser el mismo asegurado, o el patrón del asegurado que contrata planes de pensión para sus empleados.

Pensión: La cantidad de dinero que periódicamente recibirá el asegurado y/o sus beneficiarios, como consecuencia de los beneficios establecidos en el seguro.

Beneficiario: Persona que conforme a la voluntad del contratante y/o asegurado, o de la ley, tenga derecho a recibir los beneficios que la póliza establece, en el supuesto que el asegurado fallezca.

Beneficiario contingente: El beneficiario designado por el asegurado para recibir una pensión a la muerte de este último. El tipo de pensión que recibirá el beneficiario contingente está determinado desde la contratación de la póliza y tendrá sus características particulares, pues será calculada especialmente para él.

REQUISITOS DE ELEGIBILIDAD.

En razón de que en este producto el beneficio mismo comienza desde la compra del seguro de pensiones aseguradas, se tiene que el requisito fundamental para la contratación de cualquier plan disponible será que el asegurado haya comenzado su estado de pensionado por vía de jubilación o invalidez, o en su caso, que el beneficiario comience su estado de pensionado como consecuencia de un plan de fallecimiento.

Por ellos será necesario que el asegurado principal y/o los beneficiario, comprueben su edad al momento de adquirir el seguro.

TIPOS DE PLANES.

Dentro del seguro de pensiones aseguradas, existe una amplia gama de modalidades o tipos de seguros, en donde todo tipo de plan, tiene un conjunto propio de beneficios.

- Pensión cierta:

La pensión se paga durante un período de tiempo determinado, y la garantía de la pensión se encuentra en la inversión ya que no depende de la sobrevivencia del asegurado.

- Pensión temporal:

La pensión se pagará durante un plazo finito de tiempo (años), siempre y cuando el asegurado permanezca con vida durante dicho plazo.

Es decir, las pensiones dejarán de pagarse a la primera ocurrencia de cualquiera de los siguientes eventos: la muerte del asegurado, o la finalización del período finito de tiempo en el que se pagarán las pensiones.

- Pensión vitalicia:

La pensión se paga siempre y cuando el asegurado se encuentre con vida, independientemente de cualquier otra variable.

- Pensión vitalicia con "n" años de garantía.

Pensión que se paga al asegurado durante un período de tiempo determinado y si sobrevive a este período, continuará recibiendo la pensión hasta el momento de su fallecimiento.

Si el asegurado fallece antes de que finalice el período de garantía, la compañía pagará la pensión correspondiente a su(s) beneficiario(s), durante dicho plazo.

- Pensión mancomunada:

Pensión que se paga al asegurado hasta que ocurra su fallecimiento, y a partir de ese momento, se les pagará al (a los) beneficiario(s) contingente(s) que el asegurado haya designado con anterioridad, un porcentaje (previamente establecido), de la pensión que el asegurado hubiese recibido antes de fallecer.

Este tipo de pensión es muy amplia, pues en ella se puede designar que el cónyuge y los hijos reciban cada uno de éstos, un porcentaje de la pensión del asegurado.

Los beneficiarios recibirán una pensión tal y como lo designe el asegurado, siendo ésta cualquiera de los planes que se encuentren vigentes en la compañía.

- Pensión diferida:

Se podrá contratar una pensión diferida para las personas que deseen prevenir su futuro, y el pago de su pensión lo reciban después de un plazo determinado, o hasta cumplir cierta edad.

Para manejar esta opción, el seguro de pensiones aseguradas ofrece las siguientes alternativas:

. Comprar una pensión que comenzará a pagarse después de un periodo de tiempo que elija el contratante. El monto de esta pensión estará determinado desde la contratación del seguro, de tal manera que si el asegurado permanece con vida podrá gozar de su pensión según el plan que haya elegido.

. Efectuar la compra de una pensión de tal manera que la prima se pague paulatinamente durante el periodo de ahorro, y la compañía comience a efectuar el pago de la pensión, después de dicho periodo, o cuando el asegurado llegue a una determinada edad, en función del capital que haya constituido.

- Combinación de las anteriores:

Puede existir la combinación que desee el asegurado, de las pensiones expuestas en los puntos anteriores, de acuerdo a sus necesidades. Es decir, podrán existir las siguientes combinaciones:

. Pensión cierta . Pensión cierta mancomunada . Pensión cierta diferida . Pensión temporal . Pensión temporal mancomunada . Pensión temporal diferida . Pensión vitalicia . Pensión vitalicia mancomunada . Pensión vitalicia diferida . Pensión vitalicia con garantía mancomunada . Pensión vitalicia con garantía diferida

A su vez, las pensiones mancomunadas, podrán contratar cualquier tipo de pensión para sus beneficiarios contingentes, así por ejemplo se podrá tener un plan de pensión con las siguientes características:

Pensión cierta a 20 años para el asegurado principal, mancomunada con dos beneficiarios contingentes, el primero con una pensión temporal a 10 años para y el segundo con una pensión vitalicia.

Otro plan podría ser armado, por ejemplo con las siguientes opciones:

Pensión vitalicia diferida a 20 años, mancomunada con un beneficiario contingente con pensión temporal de 10 años.

No todas las pensiones tienen que ser mancomunadas, también pudíese existir un plan con las siguientes características:

Pensión vitalicia con 10 años de garantía diferida a 10 años.

Ahora bien, no todos los beneficiarios tienen que ser contingentes. La diferencia básica entre un beneficiario y un beneficiario contingente, es que para un beneficiario contingente se conoce de antemano el tipo de pensión que se le asignará en caso de fallecimiento del asegurado, en cambio para un beneficiario, a la muerte del asegurado, se puede optar por retirar el total de la reserva remanente de la pensión o contratar una pensión, mas en este caso la contratación será tratada como un negocio nuevo.

Considerando que un beneficiario contingente puede tener a su vez beneficiarios (mas no contingentes), un plan de pensiones podría armarse, por ejemplo de la siguiente manera:

Pensión vitalicia para el asegurado principal, con un beneficiario no contingente, y un beneficiario contingente, cada uno con el 50% de la pensión en caso de fallecimiento del asegurado. A su vez, el beneficiario contingente designa a dos beneficiarios (no contingentes) con el 50% de la pensión cada uno.

Como se puede observar, la cantidad de planes de pensión que se pueden armar es bastante amplio y flexible. Cabe aclarar que los cálculos actuariales que se emplean para determinar el monto de la pensión dada una prima, o de la prima dada una pensión estan en función del plan y de las edades de los asegurados.

MONTOS DE LA PENSION

El monto de la pensión dependerá del capital constituido por un asegurado al momento de adquirir el seguro, y del tipo de plan contratado. Las pensiones tienen la particularidad de que se revalorizarán periódicamente si el asegurado así lo desea, de tal manera que se adecúen a sus necesidades en cada momento, y que por consiguiente no mantengan un valor nominal que se deteriore con el paso del tiempo.

PERIODICIDAD EN EL PAGO DE LAS PENSIONES

Los pagos de las pensiones podrán realizarse bajo cualquiera de las siguientes formas periódicas de pago:

. Mensuales, Bimestrales, Trimestrales, Semestrales, Anuales.

Los pagos de las pensiones se realizarán de manera anticipada o vencida a elección del contratante, en las fechas que establezca la compañía en el plan.

PAGOS EXTRAORDINARIOS

Este beneficio establece que en adición a los pagos de las pensiones que ordinariamente se realizan durante cada año de vigencia del plan, se llevarán a cabo un cierto número de pago de pensiones adicionales o extraordinarias, los cuales estarán supeditados a que el plan se encuentre vigente.

INCREMENTO DE LAS PENSIONES

Dentro del seguro de Pensiones aseguradas, podrá elegirse que la pensión contratada sea con incrementos o no, de tal manera que se tendrán las siguientes opciones:

. Pensiones sin incrementos

En este tipo de pensiones, se establece desde el inicio de la póliza el valor nominal de la pensión, y los intereses que se generen en exceso a la tasa de cálculo, se canalizarán a un fondo de dividendos financieros, del cual el asegurado tendrá disponibilidad total.

. Pensiones con incremento variable

Se establece el monto de la pensión en moneda nacional, la cual será revaluada periódicamente con base en los excedentes de interés de la tasa real de inversión respecto a la tasa técnica de cálculo. De esta

manera, en cada revaluación se comprará a edad alcanzada la pensión que corresponda al nuevo fondo, incrementando así directamente con los intereses el monto de la pensión.

. Pensiones indexadas al dólar.

Este tipo de pensiones se paga en moneda nacional y el monto se encuentra en función de la paridad peso-dólar. El valor de las pensiones crecientes de esta forma, se pagan con respecto al tipo de cambio del Dólar libre promedio de casas de cambio a la venta, vigente cinco días hábiles anteriores a la fecha de pago.

. Pensiones indexadas a la inflación.

Las pensiones indexadas al factor inflacionario, se establecen en moneda nacional y su valor se revisará mensualmente incrementándose cada vez que se acumule al menos un 5% de inflación o cada año, lo que ocurra primero.

DISTRIBUCION DE LAS PENSIONES

De acuerdo con las necesidades de cada cliente (asegurado), el pago de las pensiones se hará al contratante si así lo requiere o directamente a los asegurados (y beneficiarios en su caso) mediante diferente mecanismos.

Estos pueden ser: Pago de la pensión por medio del deposito automático a la cuenta del banco que el contratante especifique, pago a través de la emisión de cheque al asegurado, pago directo en oficinas de la compañía.

MODIFICACIONES

Todo asegurado cubierto por cualquiera de los planes de pensiones aseguradas disponibles en este producto, puede realizar una serie de modificaciones sobre sus beneficios y otras variables del plan.

Dentro de los posibles tipos de modificaciones que podrán llevarse a cabo, están los siguientes:

- . Modificaciones a los datos particulares de la póliza
- . Cambios de beneficiario
- . Modificación al tipo de plan
- . Cambios de un método de revalorización a otro.

PENSION ADICIONAL

Mediante la compra de una pensión adicional, se podrá incrementar el monto de la pensión vigente. Se emitirá de nuevo la póliza, considerando como prima inicial, la suma de la reserva en el momento de solicitar la prima adicional mas la prima adicional.

RESCATES

- Rescate parcial.

Si el asegurado desea disponer de su reserva parcialmente, lo solicitará a la compañía. En este caso el monto de la pensión se ajustará a la reserva que resultó después de otorgar el rescate parcial.

- Rescate total.

Cuando el asegurado desee poner término a su póliza y solicite el total de su reserva. Se otorgará como monto de rescate total, la reserva constituida menos una cantidad que sea equivalente a los gastos por cubrir del año póliza en que se solicite el rescate.

INVERSIONES

Las inversiones de estos planes, están ligadas al tipo de revalorización de la pensión que se haya elegido, pues el funcionamiento de éstos está dado por las tasas o instrumentos de inversión que se manejen, lo cual se refleja directamente en la tarifa.

Sin embargo, como característica general el capital de la reserva de los planes, será invertido en los instrumentos permitidos que generen las mayores tasas de inversión y seguridad al igual que la liquidez necesaria para cumplir con las obligaciones de la compañía ante los asegurados o beneficiarios.

SUPERAVIT FINANCIERO

La inversión del fondo de reservas puede generar rendimientos que sobrepasen la reserva necesaria para cumplir con las obligaciones contractuales de la compañía con el asegurado, existiendo así, un superávit financiero, mismo que se liberará al finalizar la vigencia de la póliza.

Si durante el transcurso del seguro existen instrumentos de inversión que garanticen aún más de las obligaciones con el asegurado, el excedente podrá liberarse otorgando parte como dividendos al asegurado y parte como utilidad para la compañía.

DIVIDENDOS

Los dividendos se capitalizarán mesualmente en un fondo de administración independientemente del fondo de las reservas. Dicho dividendos equivaldrá a por lo menos el 95% del diferencial del interés entre los realmente obtenidos por la inversión de la reserva y los supuestos en la tasa técnica de cálculo.

El contratante podrá retirar cualquier cantidad de este fondo de dividendos hasta su totalidad, en el momento que lo requiera, o podrá optar por alguna de las siguientes opciones:

- a) Dejarlos en depósito para su inversión.
- b) Utilizarlos para la compra de mas pensión.
- c) Cubrir las primas de otros seguros.

GASTOS DE ADMINISTRACION

Los gastos de administración correspondientes al seguro de Pensiones aseguradas, podrán cobrarse de las dos formas siguientes:

1.- Un porcentaje del fondo inicial, el cual varía hasta un 10% como máximo, dependiendo del tamaño del fondo.

2.- Un porcentaje calculado sobre el fondo inicial variando hasta un 10% como máximo y amortizado durante la vigencia de la póliza. Dicho de otra manera, se cobrará un porcentaje que en lugar de aplicarse al fondo inicial, se aplicará al monto de la pensión, mientras ésta dure.

Esto puede hacerse dada la equivalencia entre la prima única del seguro, y el monto total de las pensiones pagadas al asegurado.

ES decir:

$$\text{Prima-Unica} = \text{Renta} \times \text{Anualidad}$$

Por lo tanto:

$$(0.1) \text{ Prima-Unica} = (0.1) \text{ Renta} \times \text{Anualidad}$$

Para efectuar el cobro de dicho gastos, se tomará en cuenta las siguientes consideraciones:

a) El porcentaje que se aplique variará dependiendo directamente del rango en que se encuentre el fondo, siendo un porcentaje menor mientras más grande sea su tamaño.

b) Existirá un sistema de "Costo-servicio", es decir que se cobrará adicionalmente dependiendo del servicio que se ofrezca y de la labor administrativa que implique cada póliza.

c) Las comisiones que se otorguen al agente por la venta de estos seguros, no podrá ser mayor a los gastos de administración.

FIG. (5.1) Ciclo de Modificabilidad del Documento Maestro

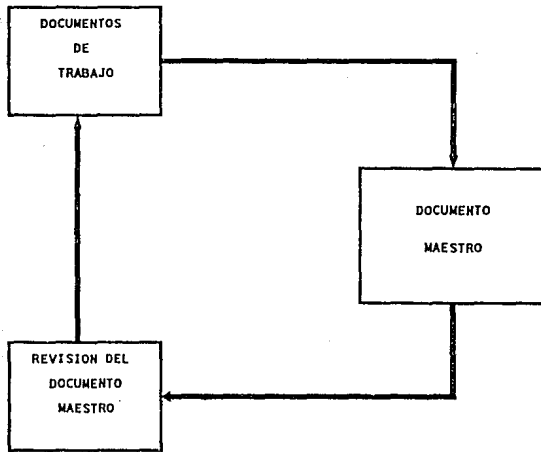


FIG. (5.2)

INVOLUCRACION DE LA FASE DE ESPECIFICACION DE REQUERIMIENTOS EN EL CICLO DE MANTENIMIENTO

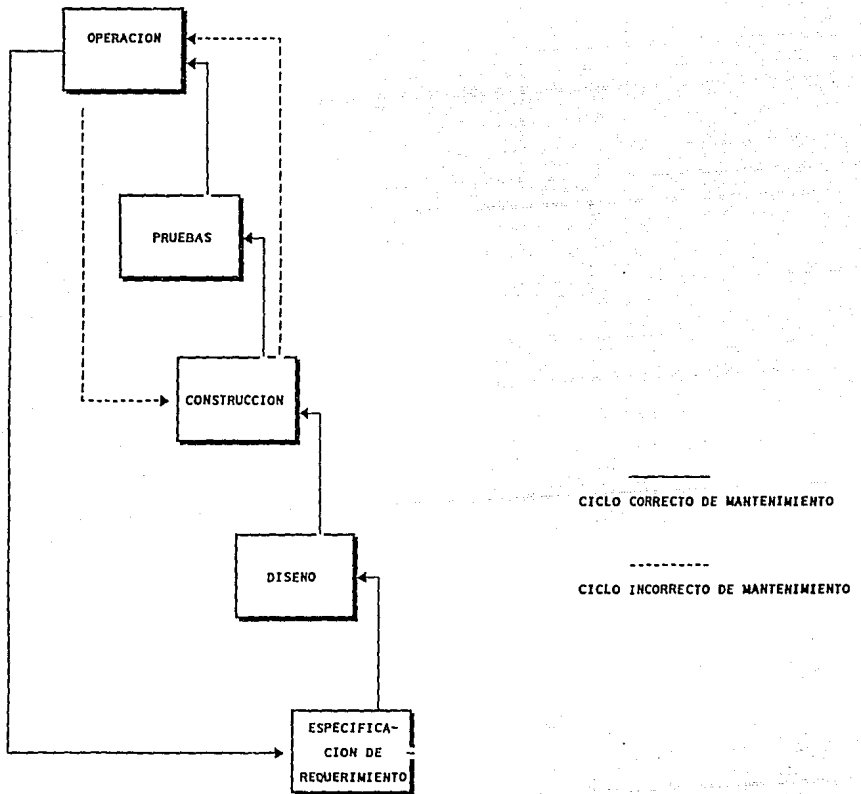


DIAGRAMA DE FLUJO DE DATOS PROCESO DE PAGO DE RENTAS

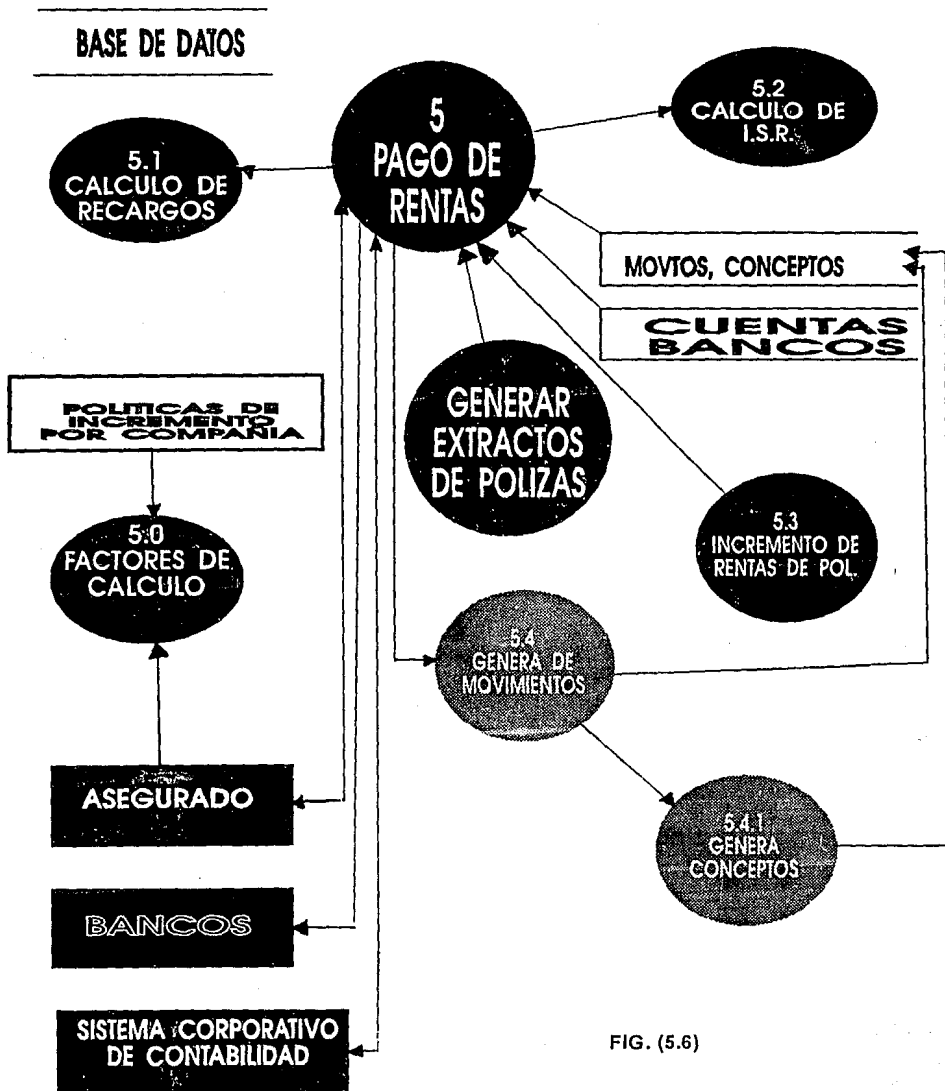


FIG. (5.6)

DIAGRAMA DE FLUJO DE DATOS PROCESO DE VALUACION DE RESERVAS

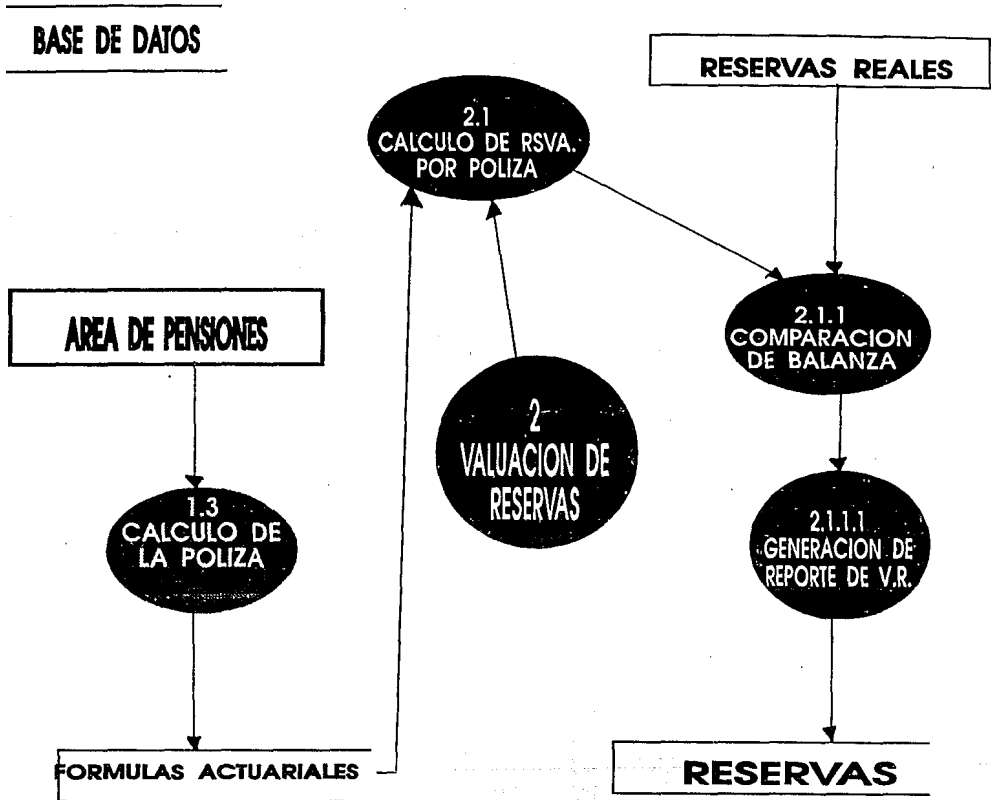


FIG. (5.5)

DIAGRAMA DE FLUJO DE DATOS CICLO MENSUAL



FIG. (5.4)

1. INTRODUCCION**2. PRODUCTOS DE LA FASE DE DISEÑO****3. COMPONENTES DE DISEÑO****3.1. DISEÑO EXTERIOR****3.2. DISEÑO ARQUITECTONICO****3.3. DISEÑO DETALLADO****4. CONCEPTOS DE DISEÑO****4.1. REFINAMIENTO SUCESIVO****4.2. CONCURRENCIAS****4.3. MODULARIDAD****4.4. OCULTAMIENTO DE LA INFORMACION****4.5. ACOPLAMIENTO****4.6. COHESION****4.7. ABSTRACCION****4.8. USO DE HERRAMIENTAS****5. METODOLOGIA PARA EL DISEÑO DE LA BASE DE DATOS****5.1. OBTENCION DEL DICCIONARIO DE DATOS****5.2. IDENTIFICACION DE ENTIDADES Y RELACIONES****5.3. NORMALIZACION DE LAS ENTIDADES Y RELACIONES****5.4. AFINACION CON RESPECTO DE LAS FUNCIONES A SOPORTAR****5.5. AFINACION CON RESPECTO A RESTRICCIONES DE LA INSTALACION****5.6. CODIFICACION DE LOS COMPONENTES DE LA BASE DE DATOS****5.7. GENERACION DEL ESQUEMA FISICO DE LA BASE DE DATOS****5.8. PRUEBAS DE GENERACION****5.9. ADECUACIONES POSTERIORES****6. SUMARIO**

1. INTRODUCCION

Las técnicas son la manifestación de los conceptos en su aplicación a situaciones particulares. Las técnicas vienen y van con las modas, los cambios tecnológicos, las condiciones económicas, etc.

Por definición, los principios fundamentales permanecen iguales a través del tiempo, proporcionando las bases para el desarrollo y evaluación de la técnicas.

Por lo anterior, podemos observar el papel estratégico que juega un diseño conceptual sobre un sistema que se desarrollará con el conjunto de técnicas disponibles.

Aunque los principios de diseño sean más estables en el tiempo, que las herramientas de desarrollo, no debe olvidarse que el diseño se basa en el análisis de una situación en el tiempo presente, es decir, el producto del diseño es dependiente del tiempo.

Para contrarrestar este efecto, se deben emplear métodos de diseño que prevean el futuro, o por lo menos que sean suficientemente flexibles para aceptar cambios drásticos al producto en el futuro.

Tanto en las artes como en la ingeniería, las consideraciones estéticas son fundamentales para el diseño; la simplicidad, elegancia y claridad de un propósito distinguen a los productos de alta calidad de los mediocres. Este valor agregado al sistema va más allá de la satisfacción de los requerimientos de desarrollo. Tiene que ver con que el usuario se sienta más que satisfecho con el producto, sino que lo sienta amable, agradable, confiable, ya que las personas dan vida a los sistema, se trata que exista simpatía por el sistema para que se le otorgue mejor vida.

Prácticamente todas las actividades que tiene que ver con el desarrollo de sistema, pueden considerarse como actividades de preparación o como de conclusión de la fase de diseño.

El diseño se lleva a cabo por medio de un refinamiento sucesivo, hasta alcanzar una clara y total descripción del producto a codificar.

2. PRODUCTOS DE LA FASE DE DISEÑO

Los productos de la fase de diseño son los siguientes: Manual preliminar del usuario, especificaciones del diseño externo y diseño estructural, especificaciones de diseño detallado y "Plan completo de pruebas"

El Manual preliminar de usuario, explica la forma de operar el sistema. El diseño externo se encarga de la vista externa del sistema: pantallas, reportes, interfases, y flujo de operación. El diseño estructural o arquitectónico trata de la arquitectura interna del sistema, funciones y estructura de datos. El diseño detallado trata sobre los algoritmos a emplear en cada función.

Después de la aceptación del diseño y del manual, se deben esperar muy pocos cambios sustantivos al proyecto, y de presentarse, deben canalizarse a través de un control de cambios formal.

El principal objetivo de la fase de diseño, es producir un diseño de calidad bajo el contexto de un producto de software de calidad.

Un control de calidad sobre la fase de diseño, no verifica que el software sea correcto o útil, sino debe concentrarse en asegurar que el diseño cumple con las especificaciones de la fase de análisis.

3. COMPONENTES DE DISEÑO

En el diseño de software, existen tres tipos distintos de actividades: el diseño externo, el diseño arquitectónico y el diseño detallado. Al diseño arquitectónico y al detallado se les conoce como diseño interno.

En la práctica, no es posible definir los requerimientos sin tener elaborado un diseño preliminar. La definición de requisitos tiene como objeto especificar los requerimientos externos, funcionales y de desempeño de un sistema, así como definir el manejo de excepciones.

La distinción entre definición de requerimientos y diseño externo no está clara, sino al contrario, en realidad es un cambio gradual de entre el "Qué" y el "Cómo".

3.1. DISEÑO EXTERIOR

El diseño exterior se concentra a resolver la interfase del software y el medio exterior. La parte más compleja de este medio exterior es el hombre, usuario del software. Debido a esta complejidad, la ruta más corta para avanzar con éxito en esta fase, es el diseño de prototipos de pantallas, reportes, etc.

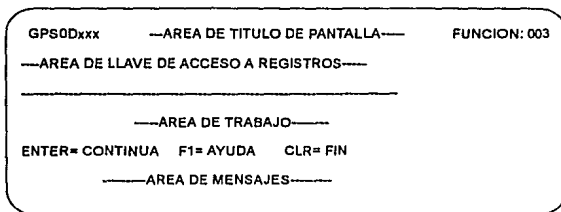
El objetivo de esta etapa es la creación de un ambiente de software donde el usuario pueda desarrollar sus actividades, con el mínimo de ansiedad, frustración, confusión, fatiga, y sobrecarga mental.

El diseño externo de software requiere de concebir, planear y especificar sus características, estas incluyen la definición de despliegue de pantallas, los formatos de los reportes, definición de las entradas y salidas de los datos, las características funcionales y la estructura general del producto.

El diseño exterior del sistema de pensiones aseguradas, se llevo a cabo casi en su totalidad en un documento conocido como "Documento de diseño", en el se especifican los formatos de todas las pantallas del sistema en línea. Estas pantallas presentan la característica de estar formateadas de manera similar, con ello se cumplen dos objetivos: Se realiza una contribución a la estandarización del ambiente de desarrollo, y se presenta una interfase común al usuario, haciendo más familiar al sistema y reduciendo la carga mental en el momento de operación.

Las pantallas presentan en la esquina superior izquierda el nombre del dialogo que las maneja, en la parte central el título de la transacción, en la parte derecha el acceso a otras funciones del sistema, presenta una área de trabajo en la parte central, una área de mensajes en la línea 23 y una área de ayuda del teclado en la línea 22.

El modelo de pantalla utilizado en el sistema de pensiones aseguradas, se muestra a continuación :



En lo que se refiere a documentos impresos, por tratarse en su mayoría de documentos contractuales, que se entregarán al asegurado, su formato y diseño corrió a cargo del departamento de formas y diseño de la institución.

En este caso, solo se realizaron los programas que realizaban las impresiones y activaban la impresora, quien finalmente tenía almacenadas las formas especiales de impresión.

3.2. DISEÑO ARQUITECTÓNICO.

El diseño arquitectónico se encarga de la vista conceptual del sistema, identificando funciones internas del proceso, descomposición funcional de alto nivel en sub-funciones, así como la definición de la estructura de datos (Base de datos), y de las relaciones entre las funciones, los datos y su almacenamiento.

Comprende la identificación de los componentes de la programación, su desacoplamiento y descomposición en módulos de procesamiento y estructuras de datos conceptuales, y la especificación de las conexiones entre componentes.

El diseño arquitectónico del sistema de pensiones aseguradas se encuentra en el diseño de la base de datos, y los diagramas de funciones (descomposición modular) que se presentan más adelante.

3.3. DISEÑO DETALLADO.

El diseño detallado no es lo mismo que la instrumentación. El primero está muy influido por el lenguaje de programación, pero no tiene que ver con aspectos sintácticos.

Durante el diseño detallado, la estructura arquitectónica es detallada en términos de la instrumentación; así el diseño es un avance que parte de conceptos abstractos hacia la representación concreta de los mismos.

El diseño detallado se lleva a cabo de forma independiente a la codificación, pues se debe resolver primero el problema a nivel conceptual para después poder concentrarse en detalles de construcción, compilación y pruebas. Con esto, También se logra tener el diseño independiente del lenguaje de programación empleado, pudiéndose por tanto, codificar en más de un lenguaje con el mínimo de esfuerzo.

Hemos alcanzado el final de la etapa de diseño detallado, cuando todas las decisiones de diseño se hayan efectuado, esto sucede cuando hemos terminado de hacernos preguntas sobre que tipo de procedimientos debemos implementar y comenzamos a preocuparnos sobre la sintaxis para expresarlo. Normalmente cada sentencia de diseño se expresa por medio de unas 5 sentencias de lenguaje de programación.

El diseño detallado se encuentra en la sección de REMARKS de cada uno de los programas. Como ejemplo se presenta a continuación el diseño detallado del programa que calcula las tablas de valores conmutados, que pertenece al ciclo eventual de la parte batch del sistema.

```
PROCESSING :
OBTAIN CALC CATALOGO PROBABILIDAD-DE-MUERTE
IF NOT-FOUND
  REGRESA
ELSE
  LEE TODA LA TABLA EN UN VECTOR DE MEMORIA
ENDIF
```

```
FOR EDAD-1 = 15 TO 100 DO
  FIND CALC CATALOGO TABLA NYD EDAD-1
  SI EXISTE BORRALA, SINO CREALA
  FOR EDAD-2 = EDAD-1 TO 100 DO
    CALCULA LX( EDAD-1, EDAD-2 )
    CALCULA DX( EDAD-1, EDAD-2 )
  STORE TABLA
  ENDFOR
ENDFOR
```

```
CALCULO NX:
FOR EDAD-1 = 15 TO 100 DO
  FOR EDAD-2 = EDAD-1 TO 100 DO
    NX = 0, INC = 0
    WHILE ( INC-EDAD-1 100 OR INC-EDAD-2 100 )
      INC-EDAD-1 = EDAD-1 + INC
      INC-EDAD-2 = EDAD-2 + INC
      POSICION-EDAD2 = INC-EDAD-2 - INC-EDAD-1 + 1
      FIND CALC CATALOGO NYD INC-EDAD-1
      OBTAIN POSICION-EDAD2 TABLA
      NX = NX + DX( EDAD-1, EDAD-2 )
      INC = INC + 1
    ENDWHILE
    FIND CALC CATALOGO NYD EDAD-1
    OBTAIN EDAD-2 TABLA
    NX( EDAD-1, EDAD-2 ) = NX
    MODIFY TABLA
  ENDFOR
ENDFOR
```

```
CALCULA LX:
IF EDAD-1 = EDAD-2
  LX = LX-TABLA-PROB-MUERTE( EDAD-1 )
ELSE
  LX = LX-TABLA-PROB-MUERTE( EDAD-1 ) *
    LX-TABLA-PROB-MUERTE( EDAD-2 ) / RADIX
ENDIF
```

```
CALCULA DX:
VX = ( 1 + I ) ** ( - ( EDAD-1 + EDAD-2 ) / 2 )
DX = VX * LX
```


4. CONCEPTOS DE DISEÑO.

4.1. REFINAMIENTO SUCESIVO

El diseño a través de un refinamiento sucesivo, se logra al posponer los detalles hasta el momento en que realmente se deban considerar, de esta manera podemos racionalizar el producto de software, sin perderlos en la complejidad de los detalles.

Un diseño de arriba a abajo se realiza cuando se describen los módulos principales y luego se descomponen en sus submódulos, este es el tipo de diseño con el que se desarrolla el sistema de pensiones aseguradas. Un modelo de abajo a arriba ocurre cuando se describen los submódulos primero, y después se construyen los módulos principales.

4.2. CONCURRENCIAS

En el pasado, solo los ingenieros dedicados a la instrumentación de sistemas operativos, bases de datos, o sistemas de tiempo real debían preocuparse de los aspectos de la concurrencia, sin embargo en la actualidad es necesario que todo ingeniero de software entienda los aspectos y el diseño concurrente.

La concurrencia se refiere al hecho que dos o más usuarios de un recurso de cómputo, lo estén usando al mismo tiempo. Esto presenta efectos colaterales que en un sistema mono-usuario no se observan.

Una concurrencia puede ser la causa de un bloqueo, es decir la contención de recursos debido a que más de dos procesos los solicitan al mismo tiempo.

Por ejemplo, un proceso A requiere el uso exclusivo de un archivo X para poder liberar la impresora Y que esta usando, por otro lado y al mismo tiempo, un proceso B requiere el uso exclusivo de la impresora Y para poder liberar el archivo X que se encuentra usando. Como se observa, ambos procesos, tanto A como B se están esperando el uno al otro, y ellos mismos son la causa de dicha espera, se ha establecido un bloqueo.

En el sistema de pensiones no se esta libre de las concurrencias, muchas de ellas sin efectos colaterales nocivos o conflictivos, pero otro, con un alto potencial de riesgo, entre los cuales figura el problema en la asignación del número de póliza.

Cada nuevo negocio que se registra en el sistema, recibe un número de póliza que lo identifica. Este número de póliza es un número secuencial. Ahora bien, se debe considerar que una vez el sistema en operación, registrará la alta de nuevos negocios de forma concurrente en todas las oficinas de la institución, o por lo menos, en más de una terminal del computador.

Si el número de póliza se asignará desde el momento de accesar la transacción de alta de nuevos negocios, sucedería lo siguiente: En la terminal A se accesa la transacción de nuevos negocios (emisión) y se le asigna el número de póliza X, Después en la terminal B también se accesa la emisión y se le asigna la póliza X+1, pero sucede que el capturista de la terminal B es más rápido que el de la terminal A, y termina la operación antes.

El número de póliza X+1 se registra en el sistema antes que el número de póliza X, pero que pasa si en la terminal A sucede algún apagón a simplemente se deciden de efectuar la operación?... El sistema registrará la siguiente secuencia de números de póliza: X-2, X-1, X+1.

Como se puede observar, el número de póliza X no existe en el sistema, si consideramos que este caso de error se presenta solo con dos terminales, el error que se generaría con 100 o más terminales (un número normal de usuarios en un sistema de "Main Frame") sería impresionante e Inaceptable.

Este problema de concurrencias, se resolvió, asignando el número de póliza hasta finalizar la función de emisión, notificandole al usuario por medio de un mensaje a la terminal, el número de póliza que se le asignó. De esta manera se evita el salto de números.

4.3. MODULARIDAD

Un módulo es una unidad independiente de software bien definida, de tal manera que no es necesario saber que es lo que hay adentro de un módulo para que lo podamos usar.

La propiedad de modularidad no es exclusiva de los procedimientos, sino también pertenece a la estructura de datos.

Los sistemas modulares incorporan colecciones de abstracciones en la que cada abstracción funcional o abstracción de dato, maneja un aspecto local del problema.

- Cada abstracción de un proceso es un subsistema claramente definido y con el potencial de ser útil para otras aplicaciones.

Un módulo debe subdividirse en un conjunto de subrutinas hasta que queda suficientemente acotado, esto es, cuando no contiene sub-funciones que sean potencialmente útiles a otros módulos. De esta manera se construye un conjunto de módulos (rutinas) que se relacionan para formar un sistema con un alto nivel de reutilización.

El mantenimiento a un módulo, ya sea por corrección de errores o adición de nuevos requerimientos, se efectúa con mínimo impacto al resto del sistema, con esto se simplifica la etapa de pruebas y la de mantenimiento y además se reduce el riesgo de alterar componentes del sistema que no requerían cambio.

4.4. OCULTAMIENTO DE LA INFORMACION

Cada módulo del sistema oculta los detalles internos de sus procesos, comunicándose los módulos solamente a través de interfaces bien definidas.

Las estructuras de mayor calidad se obtienen cuando el diseñador utiliza criterios bien definidos de modularización para dirigir las actividades del diseño.

4.5. ACOPLAMIENTO

El acoplamiento es una característica que se presenta en el diseño modular de sistemas. Se refiere a la medida en que los datos y las funciones de un módulo se encuentran interrelacionados con otros datos y funciones de otros módulos.

Entre menor sea el acoplamiento, más definido se encontrarán los módulos, y por ende se tendrá un mejor diseño.

4.6. COHESION

La cohesión también es una característica de modularidad del diseño, se refiere a la medida en que los datos y funciones se encuentren totalmente definidos y autocontenidos en su propio módulo.

Entre mayor es la cohesión, más se encuentran definidos y justificados los módulos, y por lo tanto, mejor es el diseño.

4.7. ABSTRACCION

Uno de los conceptos fundamentales de diseño es la abstracción. La abstracción es la herramienta intelectual que permite trabajar con los conceptos independientemente de las instancias particulares de estos.

Se puede realizar abstracción sobre datos y sobre funciones, estas abstracciones permiten el desarrollo de una descomposición modular del sistema.

La abstracción permite organizar y dirigir los procesos de pensamiento al posponer las consideraciones estructurales y algorítmicas hasta que las características funcionales, las estructuras de datos hayan quedado definidas.

Los sistemas modulares incorporan colecciones de abstracciones en los que cada abstracción maneja un aspecto local del problema por resolver.

Una primera abstracción es considerar al sistema como un conjunto funcional de dos elementos: SPA en línea y SPA batch.

A continuación partimos SPA en línea en seis grandes módulos (Control, Estudios, Emisión, Cobranza, Administración, Estadísticas), y a SPA batch en tres módulos (Ciclo eventual, ciclo diario y ciclo mensual). Posteriormente partimos cada uno de estos módulos hasta llegar a un nivel de abstracción casi puramente instrumental, en este momento nos encontramos en las puestas de la fase de codificación (o implementación de los programas del sistema).

4.8. USO DE HERRAMIENTAS

No obstante se requiere el uso de herramientas lo más automatizadas posibles, si es que se quiere ganar en productividad.

Las herramientas de las que nos auxiliamos en esta etapa fueron principalmente diagramas de entidad-relación, diagramas de descomposición, y generadores de programas.

No obstante no se debe olvidar que las herramientas y las técnicas no son la panacea. No hay sustituto para la creatividad, la inteligencia, y el trabajo duro aplicado dentro de un marco de trabajo sistemático.

5. METODOLOGIA PARA EL DISEÑO DE LA BASE DE DATOS

La base de datos del SPA fue lograda gracias a una modelado sucesivo, cada nuevo modelo se encontraba más lógico y completo, hasta que se obtuvo el modelo final que fue la base para la construcción física de la base de datos.

En general se pueden identificar un conjunto de pasos que se siguieron para la obtención final de la base de datos, estos son:

- Obtención del diccionario de datos.
- Identificación de entidades y relaciones.
- Normalización de las entidades y relaciones.
- Afinación con respecto de las funciones a soportar.
- Afinación con respecto a restricciones de la instalación.
- Codificación de los componentes de la base de datos.
- Generación del esquema físico de la base de datos.
- Pruebas de generación.
- Adecuaciones posteriores.

A continuación se describen cada uno de estos pasos:

5.1. OBTENCION DEL DICCIONARIO DE DATOS.

Un diccionario de datos es el antecesor de una base de datos, la diferencia consiste en que el diccionario es un simple colección de conceptos con sus respectivos significados, mientras que una base de datos es algo más que eso, pues una base de datos además esta estructurada de tal manera que permite el almacenamiento y administración de la información, (o datos) asociados a cada uno de los conceptos.

Para construir un diccionario de datos, se enlistan todos los conceptos del negocio en cuestión, en nuestro caso sobre pensiones aseguradas, estos conceptos se obtienen desde las primeras entrevistas con los usuarios, de la documentación existente del negocio (la nota técnica), de notas y memorándums internos.

Una vez construida dicha relación, se complementa con el significado y descripción de cada concepto, de esta forma, cada vez que se mencione cualquier término durante el desarrollo del concepto, se tiene la garantía de saber exactamente a lo que se esta haciendo referencia, evitando confusiones y malos entendidos.

A cada participante del proyecto se le mantiene actualizado con una copia del diccionario de datos. Este diccionario es dinámico, en el sentido que se mantiene en actualización continúa en relación a los nuevos conceptos que puedan surgir durante el proyecto.

El diccionario de datos, es además el primer paso o primera versión de la base de datos. Para que el diccionario de datos pueda dar pie a la base de datos, es necesario estructurarlo y reducirlo a su expresión mínima.

5.2. IDENTIFICACION DE ENTIDADES Y RELACIONES.

Cuando ya se conocen los elementos de información con los que se dispone y los que se requieren para la operación del negocio, se debe proceder a identificar plenamente cada entidad elemental de información, es decir, se deben identificar las "Entidades" y las "Relaciones" entre ellas. Para esto, se usan los "Diagramas entidad-relación", que es una de las técnicas principales en la fase de diseño.

Cada entidad, esta forma por un "Objeto" y por sus "atributos", o características. Las entidades identificadas en el sistema de pensiones, son las siguientes:

La póliza, que además de contener el número de identificación "Póliza", contiene las características del plan de pensiones que se contrata, tales como Períodos de temporalidad, períodos de diferimiento, tipo de plan, si es mancomunada o no, fecha de emisión, forma de recargos administrativos, etc.

El contratante, que contiene los datos generales del contratante de la póliza, su razón social, el giro de la empresa, su dirección, y su RFC.

El asegurado, que contiene los datos del asegurado de la póliza, tales como su nombre, su sexo, su edad, su domicilio.

El beneficiario, que contiene los datos del beneficiario del asegurado principal, estos datos son su nombre, su edad, el porcentaje sobre la pensión en la que son beneficiarios.

La renta, que contiene las características de la renta de determinada póliza, tales como su periodicidad, su vigencia, su monto, su tipo de incremento, y su fecha de último pago.

La reserva, que contiene datos de la reserva de determinada póliza, tales como son la reserva inicial, la reserva de dividendos, la fecha de constitución, el monto de la reserva en curso, el tipo de fondo de inversión.

El fondo de inversión, que contiene la información referente a todo un grupo de pólizas que comparten un mismo fondo de inversión, esta información se refiere al total invertido en el fondo, el tipo de fondo de inversión, el valor en curso de la participación en el fondo.

Las coberturas, que contiene información de cada una de las coberturas adicionales que se contratan con el plan de pensiones, esta información se refiere al tipo de coberturas o seguros, montos y periodicidades de las primas de aportación.

Los pagos especiales o aguinaldos, que contienen la información sobre aguinaldos especiales que se hayan contratado con el plan de pensión, estos datos son la periodicidad de pago, el monto del pago.

El agente, que contiene los datos del o de los agentes que hayan contactado determinada póliza para la institución, tales datos son la clave del agente, y su porcentaje de comisión.

El banco, que contiene información sobre el banco, tipo de cuenta y número de cuenta donde se realizarán las aportaciones de las pólizas que hayan contratado un pago automática de su pensión a cuentas bancarias.

Los conceptos, que contiene la identificación de cada uno de los conceptos susceptibles de afectar a las reservas de las pólizas.

Los movimientos, que contiene la fecha y participaciones con los que se afectan las reservas en determinada transacción.

Los asientos, que contienen el desglose de cada movimiento, y se asocia con los conceptos de la reserva.

Los recibos, que contiene la información que se imprime en cada recibo para el pago de las pensiones.

El cheque, que contiene la información para cada cheque que se entrega en las oficinas de la compañía aseguradora.

Tablas del sistema, que contiene cada una de las tablas que se administran en el módulo de tablas y políticas del sistema.

Catálogo de tablas, que contiene la identificación de cada una de las tablas el sistema.

Catálogo de operadores, que contiene el nombre, clave de acceso, departamento y nivel de acceso de cada uno de los usuarios del sistema.

5.3. NORMALIZACION DE LAS ENTIDADES Y RELACIONES.

En este paso se simplifica a su mínima expresión, cada una de las entidades de datos del sistema, evitando redundancias y complejidad en el sistema. También se válida la correcta definición de las entidades y de sus relaciones.

Para llevar a cabo una normalización de bases de datos, se deben cumplir por lo menos las primeras tres reglas normales. Estas reglas, expresadas en lenguaje natural, son como siguen:

- Cada una de las características o atributos de una entidad, se relaciona uno a uno con la llave de la entidad.

- Cada atributo de una entidad, solo tiene relación con la misma entidad, y con ninguna otra.

- Cada atributo de la entidad, es candidato posible, a ser la llave de la entidad.

5.4. AFINACION CON RESPECTO DE LAS FUNCIONES A SOPORTAR.

Una vez identificado plenamente nuestro diagrama entidad-relación, nos encontramos en posibilidad de pasar a la implementación o construcción misma de la base de datos, más sin embargo, para mejorar el rendimiento del sistema, se deben considerar algunos aspectos del diseño del mismo.

Se deben validar que para cada una de las funciones del sistema, el esquema de base de datos que se propone, cumple con las necesidades de obtención y almacenamiento de datos. Además de lo anterior, todavía pueden hacerse algunos ajustes para alcanzar mayor simplicidad en la etapa de construcción del sistema.

Es por ello que el esquema final de la base de datos, difiere un poco con respecto a su bosquejo teórico, pero todo en beneficio de un sistema más simple y más modularizado.

5.5. AFINACION CON RESPECTO A RESTRICCIONES DE LA INSTALACION.

Cuando se propone un esquema de base de datos para ser construido dentro de una instalación de cómputo, todavía deben realizarse ciertas validaciones con respecto a los estándares y restricciones de la instalación.

En el caso de pensiones aseguradas, se tuvieron que eliminar algunas relaciones entre entidades que se proponían con acceso directo. Lo anterior se debe que dos entidades con acceso directo, que se relacionen por medio de un "Set", ocasionan una dispersión de datos en los archivos de la base de datos, y ello conduce a bajos tiempos de respuesta con impacto a todas las bases de datos de la instalación.

5.6. CODIFICACION DE LOS COMPONENTES DE LA BASE DE DATOS.

El diagrama entidad-relación es el plano con el que se construye físicamente una base de datos, este diagrama recibe el nombre de "Diagrama de Bachman" en un ambiente IDMS.

El Diagrama de Bachman contiene la descripción de la base de datos a nivel esquema y subesquemas, la descripción de los atributos de cada uno de los elementos, se proporciona en la definición de los registros.

Cada "Registro" se asocia a cada "Entidad" del diagrama de Bachman, en los registros se describen los "Campo" o "atributos" de las entidades, cada Campo esta formado de un nombre propio, de un formato y de una longitud de caracteres.

El formato se refiere al tipo de datos que se almacenará en el campo, este formato puede ser Alfanumérico, numérico, alfabético, etc.

El esquema es la vista total de la base de datos, donde se pueden observar todos los registros y relaciones que la construyen. Cada relación es codificada a manera de "Set" dentro de un ambiente IDMS, y nos proporcionan el método de acceso a los registros del sistema.

Un subesquema es una vista parcial a la base de datos, y nos proporciona acceso a solo a una parte de la base de datos. La determinación de que registros pertenecerán a que subesquema y cuantos subesquemas existirán en un esquema, se realiza para proporcionar seguridad, integridad y modularidad al ambiente de base de datos.

5.7. GENERACION DEL ESQUEMA FISICO DE LA BASE DE DATOS.

La base de datos se codifica con sentencias propias de la base de datos, a través de un DMCL "Device Media Control Language", mismas que se compilan y generan módulos de carga que describen a la base de datos y norman su acceso dentro del sistema.

La codificación de la base de datos y su generación (generación de los módulos de carga), corre por lo general bajo responsabilidad del departamento de base de datos.

El equipo de desarrollo, proporciona el diagrama de bachman y la codificación de los registros al departamento de base de datos, este departamento se encarga de las cuestiones físicas de la base de datos, tales como las características de los archivos y discos donde se alojará la base de datos.

5.8. PRUEBAS DE GENERACION.

Cuando el departamento de base de datos termina la generación de la base de datos, notifica al equipo de desarrollo, para que este prosiga con la validación de la generación y con la construcción del resto del sistema.

5.9. ADECUACIONES POSTERIORES.

Cuando la base de datos esta lista y terminada, se puede proseguir con la construcción del resto del sistema: código fuente, mapas y reportes.

Como el mismo sistema es susceptible a modificaciones y adecuaciones posteriores a la fase de diseño, la misma base de datos sufre de estas modificaciones.

Para cada modificación de la base de datos, se debe realizar un estudio similar al que se lleva a cabo cuando se generan por primera vez.

Una modificación a la base de datos, ocasiona que la información que estaba contenida en ella tenga que ser reestructurada para que coincida con la nueva definición.

6. SUMARIO

Una metodología para la fase de diseño, debe cumplir con las siguientes características: Notación formal y estándar, Formalizar el proceso de refinamiento sucesivo del diseño, abstracción, modularización, organización y encapsulamiento de datos.

Con estas características, se pretende lograr un producto de diseño con las siguientes características: Realista, modificable, granular, reutilizable, que pueda ser verificable con respecto al diseño, integral y duplicable.

Muchas técnicas y métodos de diseño están implícitas o explícitamente predefinidas por el tipo de aplicación, por lo que la selección de estas técnicas y métodos se realiza bajo los lineamientos que el tipo de sistema nos indican.

El diseño es una mezcla de metodología con creatividad, que no necesariamente se contradicen, más bien se coordinan para formar productos de software creativos, formales y productivos.

Una vez finalizada la etapa de diseño, se procede con la transformación del diseño detallado en un lenguaje de programación.

(A continuación se mostrarán las figuras relacionadas Fig. 7.1 y Fig. 7.2)

DIAGRAMA DE BACHMAN SISTEMA PENSIONES ASEGURADAS

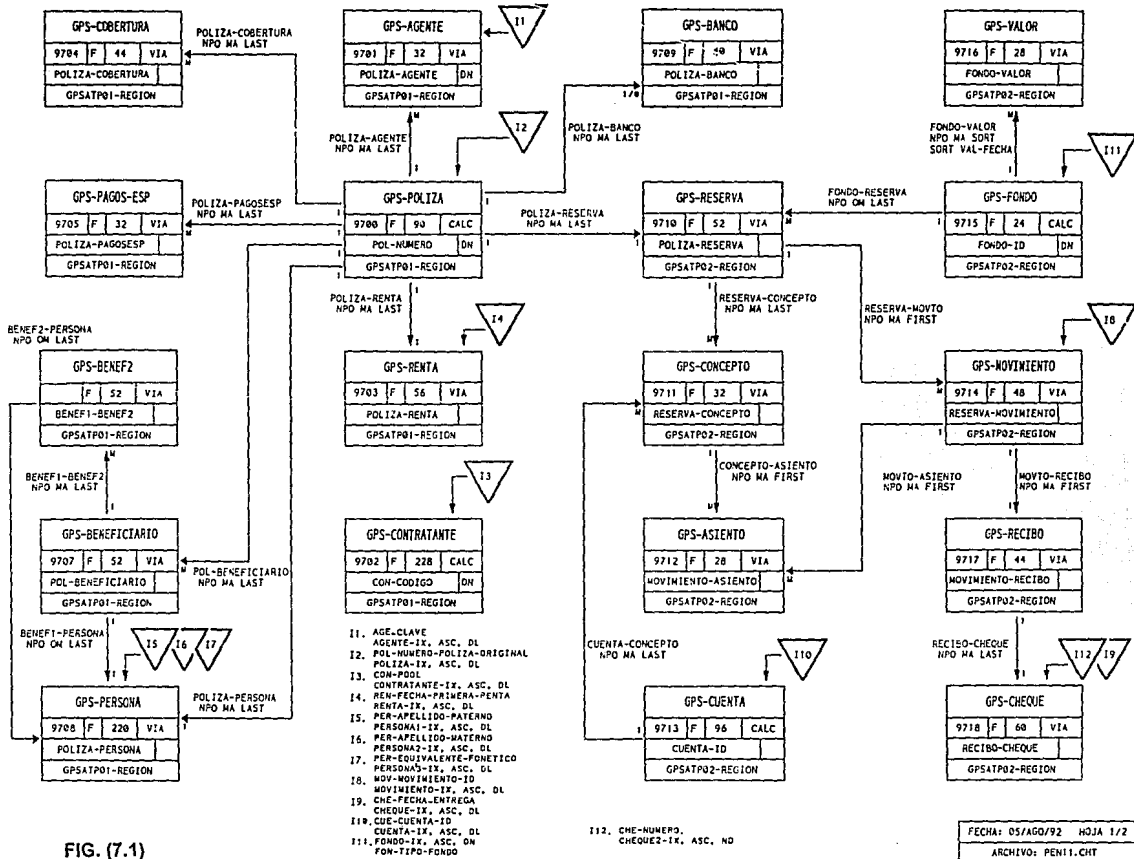


FIG. (7.1)

11. AGE-CLAVE AGENTE-IX, ASC, DL
12. POL-NUMERO-POLIZA-ORIGINAL POLIZA-IX, ASC, DL
13. CON-POOL CONTRATANTE-IX, ASC, DL
14. REN-FECHA-PRIMERA-PENTA RENTA-IX, ASC, DL
15. PER-APELLIDO-PATERNO PERSONA-IX, ASC, DL
16. PER-APELLIDO-MATERNO PERSONA-IX, ASC, DL
17. PER-EQUIVALENTE-FONETICO PERSONA-IX, ASC, DL
18. MOV-MOVIMIENTO-ID MOVIMIENTO-IX, ASC, DL
19. CHE-FECHA-ENTREGA CHEQUE-IX, ASC, DL
110. CUE-CUENTA-ID CUENTA-IX, ASC, DL
111. FONDO-IX, ASC, DN FON-TIPO-FONDO
112. CHE-NUMERO, CHEQUE-IX, ASC, ND

DIAGRAMA DE BACHMAN SISTEMA PENSIONES ASEGURADAS (SUBSISTEMA DE SOPORTE)

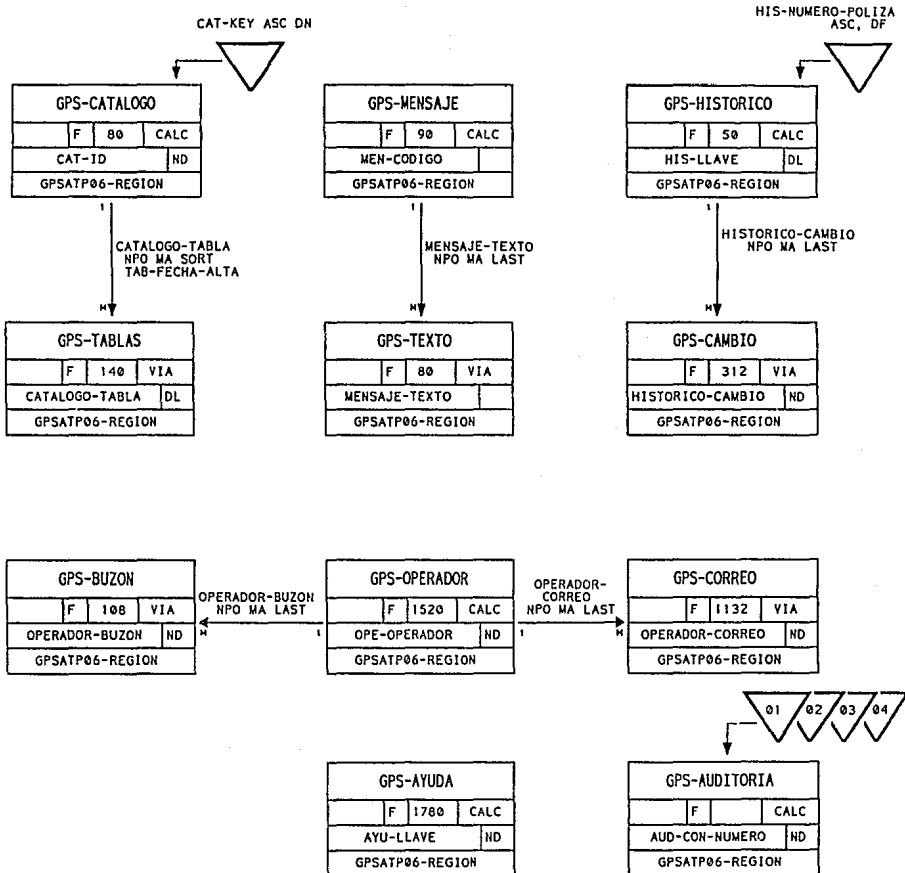


FIG. (7.2)

- 1. DESCRIPCION DE LA ETAPA DE CONSTRUCCION**
- 2. PROGRAMACION ESTRUCTURADA**
- 3. PRINCIPIOS DE CONSTRUCCION**
- 4. ESTANDARES DE CODIFICACION**
- 5. AMBIENTE DE CONSTRUCCION EN LINEA**
 - 5.1. RELACION DE PROGRAMAS EN LINEA**
- 6. AMBIENTE DE CONSTRUCCION BATCH**
 - 6.1. RELACION DE PROGRAMAS BATCH**
- 7. SUMARIO DE HERRAMIENTAS UTILIZADAS.**
- 8. PRUEBAS**

1. DESCRIPCION DE LA ETAPA DE CONSTRUCCION

En la etapa de construcción, se transforma el diseño detallado en un lenguaje de programación. El código fuente cumple dos funciones primordiales, estas son:

Produce el código ejecutable, con el auxilio de un compilador, y también es un documento formal importante por sí mismo, que expresa de forma clara y concisa la totalidad del sistema.

El objetivo de la fase de construcción es el desarrollo de código fuente sencillo y comprensible. Dada una documentación adecuada del diseño, la construcción de un producto de software debe ser un proceso sencillo, de poca tensión y muy eficiente.

El documento más importante de la etapa de construcción, es el código fuente en sí, y su soporte más valioso es su archivo particular. El archivo particular de cada uno de los elementos de programación es aquel donde se describen sus objetivos, algoritmo, autores, fechas y modificaciones. Podemos analogar este archivo a un expediente médico. Para atender a un paciente (programa), es necesario y obligado conocer previamente su expediente, si no se incurre en un riesgo tan grande como lo puede ser la pérdida de vida del paciente. Tan pronto se le haya aplicado algún medicamento o tratamiento (cambios al programa), debe registrarse en el expediente para tenerlo al día y completo.

2. PROGRAMACION ESTRUCTURADA

La programación estructurada fue el primer concepto que surgió de la llamada crisis de software, y fue el primer paso en búsqueda de formalidad y profesionalismo en la industria de software. No basta aplicar programación estructurada al desarrollo de un sistema de software, pues no es suficiente para garantizar un sistema limpio y profesional. La programación estructurada es tan solo un componente indispensable en el desarrollo metodológico de un sistema. El principio básico de la programación estructurada, es el desarrollo de programas siguiendo una forma de pensar limpia, ordenada, simple, modular y formal. El objetivo es crear programas que sean fácilmente de interpretar por diferentes personas y en diferentes momentos, de tal manera que su mantenimiento sea llevado a cabo con el mínimo de riesgos y costos.

Lo anterior se logra a través de la linealización del flujo de control, de modo que la secuencia de ejecución siga a la secuencia en que está escrito el código. La estructura dinámica de un programa a medida que se ejecuta se parece entonces a la estructura estática de un texto escrito. Esto mejora la legibilidad del código, lo cual facilita la comprensión, depuración, prueba, documentación y modificación de programas. El requisito básico de la programación estructurada es el uso de construcciones de una sola entrada y una sola salida. Para ello, se deben usar solo tres estructuras básicas de control, estas son la secuencial, la selección y la iteración. La programación estructurada no da ninguna guía para el diseño y el método de acceso de las representaciones de datos.

La programación estructurada es una técnica para linealizar el flujo de control. Se deben construir rutinas sencillas de entre 5 y 25 líneas de código. La porción ejecutable de una rutina debe instrumentar una función bien definida que pueda describirse por una frase simple que tenga un verbo activo. Se deben utilizar sangrías, espacios, paréntesis, líneas en blanco y márgenes alrededor de bloques de comentarios, para mejorar la legibilidad. Un ejemplo del código fuente del sistema de pensiones aseguradas, se presenta a continuación. Se trata de la rutina principal del módulo de pago de rentas del ciclo diario.

```

.....
PROCEDURE DIVISION.
.....
RUTINA-PRINCIPAL.
DISPLAY '*** RUTINA PRINCIPAL ***'.
PERFORM XXXX-INICIALIZA.
PERFORM XXXX-PROCESA UNTIL FIN-DE-ARCHIVO.
PERFORM XXXX-FINALIZA.

```

```

-----
* BARRE EL EXTRACTO DE POLIZAS, Y DECIDE QUE OPERACION APLICAR*
*( PAGO DE RENTAS, AGUINALDO, O AMBOS ) *
-----

```

```

XXXX-PROCESA.
  DISPLAY 'XXXX-PROCESA'.
  PERFORM XXXX-OBTAIN-POLIZA.
  PERFORM XXXX-INDEXA-LA-RENTA.
  IF VALOR-ENCONTRADO
    IF PAGA-RENTA
      PERFORM XXXX-PAGA-RENTA
    ELSE
      IF PAGA-AGUINALDO
        PERFORM XXXX-PAGA-AGUINALDO
      ELSE
        IF PAGA-AMBOS
          PERFORM XXXX-PAGA-RENTA
          PERFORM XXXX-PAGA-AGUINALDO
        ELSE
          DISPLAY '==== TIPO DE PAGO INVALIDO '
          DISPLAY 'REGISTRO: ' WK-REG-POLIZAS.
        PERFORM XXXX-IMPRIME.
        PERFORM XXXX-LEE-POLIZA.

```

3. PRINCIPIOS DE CONSTRUCCION

Durante la etapa de construcción, se debe codificar con los siguientes principios básicos:

- Programar de forma defensiva, previniendo errores del programa, del equipo de cómputo, y del usuario.

Esto se logra por medio de un estructuración simple y sencilla del programa fuente, con validación de todos los casos posibles de error, y con rutinas comunes de validación, verificando el tipo de datos e inicializarlos antes de su uso, ejecutando la rutina más común por omisión, etc.

- Programar pensando en la etapa de pruebas y de mantenimiento.

Construir rutinas simples y de pocas sentencias, con la documentación necesaria e minimizando código muerto que no se use. En general, modularizar las funciones y estructurar de forma lógica. La construcción debe estar hecha de forma tal, que permita la rápida localización de cualquier tipo de error que pudiese presentarse.

- Programar siguiendo estándares.

La codificación y en general el desarrollo de todas las etapas de la construcción de un producto de software, permite presentar una imagen homogénea y consistente de los productos, de tal manera que se facilita el entendimiento de estos para cualquier ingeniero de software que tenga que involucrarse, haciendo posible la construcción y modificación del producto a través de diferentes personas con el mínimo de impacto al software. Cuando se codifica el programa, se debe pensar que algún día alguien tendrá que revisarlo y quizás modificarlo. Si los nombres de variables y rutinas no son significativos, y no se sigue una lógica limpia de modularización, se tendrá una etapa de mantenimiento costosa y llena de sinsabores.

4. ESTANDARES DE CODIFICACION Y DISEÑO DETALLADO

Los objetivos que se persiguen con la estandarización de los programas fuente, son los siguientes:

- MAXIMIZAR EL RENDIMIENTO DE CONSTRUCCION DE PROGRAMAS
- MINIMIZAR EL ESFUERZO DE CORRECCIONES EN LA ETAPA DE PRUEBA
- MINIMIZAR EL ESFUERZO EN LA ETAPA DE MANTENIMIENTO

La creatividad siempre ocurre dentro de un marco de trabajo básico de estándares. Los artistas siguen los principios básicos de estructura y composición. Los poetas siguen el ritmo y métrica del lenguaje, y los músicos siguen una progresión de acordes fundamental.

La estandarización previene de malos estilos de programación, brinda familiaridad al sistema y lo hace amigable y entendible. Si el código es construido con un mismo estilo y filosofía, cualquier participante del proyecto podrá participar en el desarrollo o construcción de cualquier programa, minimizando el efecto de "yo no lo hice, no lo conozco".

La estandarización, hace al sistema independiente de sus constructores brindando gran flexibilidad a la administración del mismo. Al estandarizar la codificación, se obtiene un producto limpio y profesional. La estructura de los programa debe ser lo mas parecida entre si posible. esto se logra, siguiendo un estilo de descomposición modular preestablecido. La codificación del sistema de pensiones aseguradas se llevó a cabo bajo las siguientes reglas:

Reglas:

1. Dividir el área de working en las siguientes secciones, y en el siguiente orden:

- Campos del idms
- Copy a los mapas simbólicos
- Copy a los registros de archivos
- Wk-banderas
- Wk-tablas
- Wk-variables
- Wk-constantes
- Wk-mensajes-de-mapas

2. Dividir la procedure division de la siguiente manera:

- Monitor del programa
- Manejo-general-de-errores
- Carga-área-común
- Responce-key
- Retorno-a-sí-mismo

3. Se codifica siguiendo el orden del diagrama de descomposición modular (de arriba a abajo, y de izquierda a derecha)

4. El control del diálogo, se llevará a cabo por medio de pasos, utilizando el mínimo de estos, y siendo estos secuenciales a partir del primer paso.

5. En el primer paso, debe incluirse siempre el siguiente párrafo:

```
xxxx-Inicializa-variables  
perform xxxx-inicializa-mapa  
perform xxxx-inicializa-working  
perform xxxx-Inicializa-registros
```

6. Se dividen dos áreas completamente definidas:

- Area de rutina principal
- Area de subrutinas de uso general

dentro de esta, se incluyen los siguientes párrafos:

7. Todo párrafo debe llevar un encabezado que describa su función

8. Todo párrafo, debe ser menor a una pantalla completa (24 línea) incluyendo su nombre.

Si el párrafo contiene sentencias repetitivas, entonces puede ampliarse este límite de líneas.

9. Las variables y nombres de párrafo, no deben abreviarse, debe usarse la cantidad suficiente de letras, que los describa. Si no se puede, probablemente se trate de un párrafo o variable mal planteada.

10. Toda variable debe llevar prefijo, siendo este los siguientes:

Wk - para variables y contantes de la working-storage section
Sw - para switches

11. Se usará el mínimo de banderas posibles

12. No se usarán numeros ni letreros dentro de la procedure division sino, que se definirán todos estos en el área de constantes de la working-storage, algunas constantes preestablecidas, son las siguientes:

```
uno pic 9(01) value 1.  
dos pic 9(01) value 2.  
noo pic x(02) value 'no'  
sii pic x(02) value 'si'
```

13. Si una rutina ejecuta la misma función que otra de algún otro programa, esta rutina debe copiarse de ese programa.

Debe intentarse que esto suceda, por medio de la parametrización de las rutinas, y la fragmentacion de código que tienda al máximo a la reutilización.

Pero no debe existir dos versiones de la misma función, si se presenta este caso, debe condicionarse el nuevo programa, para que pueda usar la misma rutina, sin alteración. si esto no se puede, acudir con el

desarrollador de la rutina candidata a copiarse, para ver si se puede llegar a una rutina común, si esto no se puede, entonces la nueva rutina no realiza la misma función que la rutina candidata por lo que debe ser reelaborada completamente.

14. Todo párrafo lleva el siguiente prefijo:

XXXX- Mismo que será sustituido por el numero de secuencia y profundidad del párrafo, dentro del programa. esto se hará una vez se entre a la fase de pruebas integrales.

Número de secuencia (inicia con 01, con progresión de 2 en 2)

Número de profundidad (inicia con 00, va de 2 en 2)

15. Todo párrafo de manejo de errores, llevara el siguiente prefijo :

xxxx-error-

16. Todo programa deberá llevar su encabezado en el área de remarks este encabezado es el siguiente

17. Todo párrafo deberá ir separado por una sola línea en blanco

18. No se permite la sentencia goto ni la sentencia exit

19. Los programas terminarán con la sentencia goback.

20. Una línea de asteriscos se colocará una línea antes y una línea después de los nombres de las divisiones, y para el REMARKS cualquier otra línea divisoria será de guiones.

21. La numeración de datos en la working storage, será a partir del nivel 03, de tres en tres. ejemplo: 01, 03, 06, 09, etc...

5. AMBIENTE DE CONSTRUCCION EN LINEA:

El desarrollo de las funciones en línea del sistema de pensiones aseguradas, se llevo a cabo a través del lenguaje de programación ADS ("Aplication Development System"), que es un lenguaje de cuarta generación e IDMS es la base de datos propietaria. La sintaxis del ADS es similar a PASCAL en sus construcciones básicas, pero contiene una filosofía muy diferente.

Los programas ejecutables son compuestos por un conjunto de módulos fuente independientes. Cada uno de estos módulos ejecuta una función claramente determinada y se asocia a una tecla de solicitud de atención al procesador ("Responssse key"). Estos módulos de código fuente son conocidos como "Responssse process" y se ejecutan solamente cuando la "Responssse key" asociada es presionada por el usuario. Un "Mapa" es la construcción independiente de la pantalla de la transacción, en el se define la vista o imagen de la pantalla, los campos de pantalla fijo (que no pueden ser alterados) y los campos variables (para entrada y salida de datos), así como los atributos particulares de la pantalla en cuestión. Un diálogo es un módulo de carga ejecutable, y esta compuesto por un conjunto de "Responssse process" y por un "Mapa". Cada Submódulo del sistema de pensiones aseguradas de la parte en línea, se instrumento a través de un diálogo.

En otros ambientes de desarrollo en línea para Main Frame (por ejemplo CICS), a la unidad básica de construcción de sistemas, se les conoce como "Transacción", y este termino se prolonga hasta el ambiente de desarrollo IDMS, por ello el término de Diálogo y Transacción es usado indistintamente. Un módulo fuente no contiene en sus interior la definición de las variables y estructuras de datos que utiliza, por el contrario, esta definición se codifica en módulos fuente independientes conocidos como "Registros", y se ligan al

"Response process" por medio del "Diálogo". La codificación de módulos fuentes, tanto de instrucciones ("Response process"), como de datos ("Registros"), se realiza en el diccionario de la base de datos IDMS, este diccionario se accesa desde el ambiente en línea de IDMS a través de la transacción "IDD" ("Integrated Data Dictionary"),

El IDD es un sistema organizador de archivos, y su función es análoga a las bibliotecas o directorios de archivos de otros ambientes. Un ejemplo de programa fuente ADS codificado en el IDD se presenta a continuación:

```
ADD PROCESS GPSORADE
DESCRIPTION IS 'CUENTA EL NUMERO DE BENEFICIARIOS DE UNA
POLIZA
PROCESS SOURCE FOLLOWS
} * * * * *
ESTA RUTINA OBTIENE EL NUMERO DE BENEFICIARIOS DADA * } *
UNA POLIZA CAPTURADA POR EL DIALOGO ANTERIOR'
* * * * *

READY GPSATP01-REGION USAGE-MODE IS RETRIEVAL.
MOVE X-NUMERO-DE-POLIZA TO POL-NUMERO.
OBTAIN CALC GPS-POLIZA.
IF DB-REC-NOT-FOUND
    DISPLAY 'POLIZA NO ENCONTRADA'.

MOVE ZERO TO WK-CONTADOR.
MOVE ZERO TO X-NUMERO-DE-BENEFICIARIOS.
OBTAIN FIRST GPS-BENEF1 WITHIN POL-BENEF1.
WHILE NOT DB-END-OF-SET
    REPEAT.
        ADD 1 TO WK-CONTADOR.
        OBTAIN NEXT GPS-BENEF1 WITHIN POL-BENEF1.
    END.

MOVE WK-CONTADOR TO X-NUMERO-DE-BENEFICIARIOS.
DISPLAY TEXT 'CONSULTA REALIZADA CORRECTAMENTE'
MSSEND.
```

La construcción y compilación de mapas se lleva a cabo por medio de OLM ("On Line Mapping"). La construcción y compilación de diálogos se lleva a cabo por medio de ADSS ("Application Development System Generator"). La base de datos no puede ser alterada en línea, si no es por medio de programas ejecutables, sin embargo, si puede ser consultada por medio del "OLQ" ("On line Query"). IDD y OLQ son manejados por medio de sentencias o comandos de instrucciones, con sintaxis propias. ADSS y OLM son manejados por medio de paneles y menús.

El TCF ("Task Control Facility") permite conmutar de forma directa entre las diversas herramientas de desarrollo en línea. Como ya se ha mencionado, hubo la necesidad de utilizar programas ejecutables comunes para la parte batch y para la parte en línea del sistema de pensiones aseguradas. Estos programas fueron desarrollados en lenguaje de programación COBOL. Desde la parte en línea, los programas Cobol fueron ligados a los diálogos a través de una instrucción "LINK". Estos programas se invocan solo para

efectuar cálculos complicados que desde la parte en línea hubiese sido muy difícil de obtener. Los programas Cobol que fueron ligados a la parte en línea del sistema son los siguientes:

GNPA0002 Cálculo de dividendos.

GNPA0004 Generación de registros contables

GNPA0005 Cálculo de la anualidad (Para el cálculo de rentas)

La reutilización de módulos en la parte en línea fue muy positiva para el sistema en general, pues si se requería alguna modificación por ejemplo a la función de cálculo de dividendos, al alterar el programa GNPA0002 se reflejaba la modificación tanto para la parte batch como para la parte en línea, simplificando con esto la labor de depuración y cambios al sistema.

5.1. RELACION DE PROGRAMAS DE LA PARTE EN LINEA

DIALOG...: PENSION MENU PRINCIPAL

MODULO.....: EMISION

SUBMODULO....: DATOS DEL CONTRATANTE

DIALOG...: GPS0DA00 ALTA DATOS DEL CONTRATANTE

SUBMODULO....: DATOS DEL ASEGURADO

DIALOG...: GPS0DB00 ALTA DATOS DEL ASEGURADO

DIALOG...: GPS0DB10 ALTA DATOS DE LA PENSION

DIALOG...: GPS0DB20 ALTA PAGOS ESPECIALES

DIALOG...: GPS0DB30 ALTA OTRAS COBERTURAS

SUBMODULO....: DATOS DEL BENEFICIARIO

DIALOG...: GPS0DC00 ALTA DATOS DEL BENEFICIARIO

SUBMODULO....: CALCULO DE LA ANUALIDAD

DIALOG...: GPS0DCA0 CALCULO DE LA ANUALIDAD

MODULO.....: CONTROL

SUBMODULO....: CATALOGO Y TABLAS

DIALOG...: GPS0DD00 MENU DEL CATALOGO DE TABLAS

DIALOG...: GPS0DDA0 TABLA PROBABILIDAD DE MUERTE

DIALOG...: GPS0DDB0 TABLA RECARGOS POR ADMON. Y COMIS.

DIALOG...: GPS0DDC0 TABLA DE ESTADOS

DIALOG...: GPS0DDD0 TABLA DE GIROS DE EMPRESA

DIALOG...: GPS0DDE0 TABLA DE AGENTES

DIALOG...: GPS0DDF0 TABLA DE OFICINAS

DIALOG...: GPS0DDG0 FUNCIONES DEL SISTEMA

DIALOG...: GPS0DDH0 CATALOGO GENERALIZADO DE TABLAS

DIALOG...: GPS0DDI0 TABLA DE BANCOS

DIALOG...: GPS0DDJ0 TABLA DE IMPUESTOS

DIALOG...: GPS0DDK0 TABLA DE ZONAS ECONOMICAS

DIALOG...: GPS0DDL0 CONCEPTOS DE LAS RESERVAS

DIALOG...: GPS0DDM0 TABLA DE FONDOS DE INVERSION

DIALOG...: GPS0DDN0 TABLA DE SUBSIDIOS

DIALOG...: GPS0DDO0 TABLA DE COBERTURAS ADICIONALES

SUBMODULO.....: SEGURIDAD
DIALOG....: GPS0DE00 ALTA DE USUARIOS
DIALOG....: GPS0DE11 ALTA DE FUNCIONES POR USUARIO

MODULO.....: ADMINISTRACION

SUBMODULO.....: CONSULTA DATOS ACTUALES
DIALOG....: GPS0DH00 MENU DE CONSULTAS ACTUALES
DIALOG....: GPS0DH00 DATOS DEL CONTRATANTE
DIALOG....: GPS0DH80 DATOS DEL ASEGURADO
DIALOG....: GPS0DHC0 DATOS DE PAGOS ESPECIALES
DIALOG....: GPS0DHD0 DATOS DE OTRAS COBERTURAS
DIALOG....: GPS0DHE0 DATOS DE LA PENSION
DIALOG....: GPS0DHF0 DATOS DEL BENEFICIARI
DIALOG....: GPS0DHXX DATOS DE LA PENSION (CONTINUACION)
DIALOG....: GPS0DH10 DATOS DEL CONTRATANTE (CONTINUACION)
DIALOG....: GPS0DH20 DATOS DEL CONTRATANTE (CONTINUACION)
DIALOG....: GPS0DH40 DATOS DEL ASEGURADO (CONTINUACION)
DIALOG....: GPS0DH50 DATOS DEL ASEGURADO (CONTINUACION)
DIALOG....: GPS0DH60 DATOS DE PAGOS ESPECIALES (CONTINUACION)
DIALOG....: GPS0DH70 DATOS DE OTRAS COBERTURAS (CONTINUACION)
DIALOG....: GPS0DH80 DATOS DEL BENEFICIARIO (CONTINUACION)
DIALOG....: GPS0DH90 DATOS DEL BENEFICIARIO (CONTINUACION)

SUBMODULO.....: CONSULTA DATOS HISTORICOS
DIALOG....: GPS0DI00 CAPTURA DE POLIZA A CONSULTAR
DIALOG....: GPS0DIA0 DATOS DEL ASEGURADO
DIALOG....: GPS0DIB0 DATOS DEL BENEFICIARIO
DIALOG....: GPS0DI10 SELECCION DEL CAMBIO
DIALOG....: GPS0DI20 MENU DE CONSULTA A HISTORICOS

SUBMODULO.....: CAMBIO DATOS DE EMISION
DIALOG....: GPS0DJ00 MENU DE CAMBIOS
DIALOG....: GPS0DJA0 DATOS DEL CONTRATANTE
DIALOG....: GPS0DJB0 DATOS DEL ASEGURADO
DIALOG....: GPS0DJC0 DATOS DE PAGOS ESPECIALES
DIALOG....: GPS0DJD0 DATOS DE OTRAS COBERTURAS
DIALOG....: GPS0DJE0 DATOS DE LA PENSION
DIALOG....: GPS0DJF0 DATOS DEL BENEFICIARIO
DIALOG....: GPS0DJ10 DATOS DEL CONTRATANTE (CONTINUACION)
DIALOG....: GPS0DJ30 DATOS DEL ASEGURADO (CONTINUACION)
DIALOG....: GPS0DJ40 DATOS DE PAGOS ESPECIALES (CONTINUACION)
DIALOG....: GPS0DJ50 DATOS DE PAGOS ESPECIALES (CONTINUACION)
DIALOG....: GPS0DJ60 DATOS DE OTRAS COBERTURAS (CONTINUACION)
DIALOG....: GPS0DJ70 DATOS DE OTRAS COBERTURAS (CONTINUACION)
DIALOG....: GPS0DJ80 DATOS DE LA PENSION (CONTINUACION)
DIALOG....: GPS0DJ90 DATOS DEL BENEFICIARIO (CONTINUACION)

SUBMODULO.....: COMPRA DE MONTOS ADICIONALES
DIALOG....: GPS0DK00 COMPRA MONTOS ADICIONALES

SUBMODULO..... DEFUNCIONES

DIALOG.... GPSODLA0 CAPTURA DE DEFUNCIONES

DIALOG.... GPSODLB0 CONSULTA DE POLIZAS GENERADAS

DIALOG.... GPSODLC0 ACTUALIZACION DE POLIZAS GENERADAS

SUBMODULO..... CONSULTA DE CHEQUE ENTREGADOS

DIALOG.... GPSODM00 CONSULTA DE CHEQUES ENTREGADOS

SUBMODULO..... INDICES POR FONDO

DIALOG.... GPSODN00 CAPTURA DE INDICES POR FONDO

SUBMODULO..... PAGO DE RENTAS

DIALOG.... GPSODQ00 TABLA DE DOLAR E INFLACION

SUBMODULO..... RESCATES

DIALOG.... GPSODR00 CAPTURA DE RESCATES

SUBMODULO..... DIVIDENDOS

DIALOG.... GPSODSA0 FORMULA DE DIVIDENDOS

DIALOG.... GPSODSB0 RETIRO DE DIVIDENDOS

DIALOG.... GPSODSB0 CALCULO DE DIVIDENDOS

SUBMODULO..... REPORTEADOR

DIALOG.... GPSODT00 MENU DE REPORTEADOR

SUBMODULO..... CONSULTA DE BITACORA (AUDITORIA)

DIALOG.... GPSODU00 CONSULTA DE BITACORA

MODULO.....: SUBRUTINAS DE USO COMUN

DIALOG.... GPSOD300 GENERACION DE REGISTRO DE AUDITORIA

MODULO.....: INTERFASES ON-LINEDIALOG.... GPSOD200 CONSULTA LA CLAVE DEL AGENTE EN SAETA
(DICCIONARIO PRIMARIO)**MODULO.....: HERRAMIENTAS DE DESARROLLO**

DIALOG.... GPSODDEL BORRADO DE POLIZAS

DIALOG.... GPSUD035 MANTENIMIENTO AL REGISTRO TRANS-035

DIALOG.... GPSODAUD MANTENIMIENTO A REGISTROS DE AUDITORIA

DIALOG.... GPSODCS0 ALTA REGISTROS DE CHEQUES

6. AMBIENTE DE CONSTRUCCION BATCH:

El desarrollo de la parte batch del sistema, se llevo a cabo por medio de programas Cobol que se codificaron en un ambiente TSO / ISPF. TSO ("Time Sharing Options") es el sistema de tiempo compartido de los computadores IBM Main Frame bajo sistema operativo MVS.

El TSO permite crear sesiones de trabajo independientes para cada programador, de manera tal que parece que cada programador trabaja con una máquina particular independiente de las otras y relacionada solo por compartir los mismos periféricos (Discos, impresoras, cintas, etc.).

ISPF ("Interactive System Productive Functions") es un subsistema que corre bajo TSO. ISPF es un sistema de presentación o "Shell" como es conocido en otros ambientes. Presenta una serie de paneles y menús para trabajar con los elementos del sistema, y proporciona un editor sumamente potente para la edición de archivos y programas fuente. También provee con una serie de interfases para acceder la familia de productos de software que IBM a desarrollado para trabajar bajo ambiente TSO.

Uno de estos productos es el SDSF. El SDSF ("System Display and Search Facility") es un subsistema que nos permite monitorear y controlar los trabajos que se encuentran en el SPOOL. Por medio de SDSF se monitorea la ejecución y terminación de los procesos batch que son disparados desde la sesión de TSO.

Para la generación de reportes varios a la base de datos, con el fin de validar el correcto funcionamiento del sistema, se empleó el CULPRIT, que es una herramienta que permite obtener reportes desde ambiente batch a través de programas simples codificados en CULPRIT. Las compilaciones de los programas Cobol, Impresiones, y una serie de herramientas son ejecutados por medio de procesos batch de corta duración. Para poder invocar estos procesos, se deben construir procesos batch por medio de sentencias de JCL ("Job Control Language").

El JCL es un pseudolenguaje de programación por medio del cual solicitamos procesos batch al computador. Los ciclos Diario, Mensual y Eventual que componen la parte batch del sistema, también están contruidos por medio de JCL. Un "Disparador" es un segmento de código JCL que únicamente incluye la sentencia de JOB. Con este, se solicita ejecución de un procedimiento batch. El disparador del ciclo diario se muestran a continuación:

```
//TH2HRMCD JOB (00000),'CD-13',
```

```
// CLASS=Y,MSGCLASS=X,NOTIFY=TH2HRM,MSGLEVEL=(1,1)//
//.....
// * PAD0100 *
//
//
// APLICACION: PENSIONES ASEGURADAS.
//
// PROCESO: DIARIO.
//
// OBJETIVO DEL PROCESO: REALIZA EL CICLO DIARIO ( EMISION,
// ADMINISTRACION Y CONTROL ) DEL SISTEMA *
// DE PENSIONES ASEGURADAS.
//
// SE CORRE DESPUES DEL: SIN RESTRICCIONES.
//
// SE CORRE ANTES DEL: CICLO DIARIO DEL SISTEMA CORPORATIVO *
// DE CONTABILIDAD.
//
// ELABORADO POR: HUMBERTO RAMOS M. FECHA: 29/10/92 *
//
// MODIFICADO POR: FECHA: DD/MM/AA *
//
//.....
//
//JOB LIB DD DSN=DNHP.EHPA1LOA,DISP=SHR
// DD DSN=SYS2.IDMSVIDA.PROD.LOADLIB,DISP=SHR
// DD DSN=SYS2.BDDESVID.PROD.LOADSSC,DISP=SHR
// DD DSN=PNEP.ERPL3LOA,DISP=SHR GENSY SIN
// DD DSN=SYS2.PLTRANS,DISP=SHR GENSY SIN
//
//DIARIO PROC TIPO=P, TIPO DE PROCESO
// INICIO=, PASO INICIAL
// FINAL=, PASO FINAL
// RAZON=, RAZON DE REPROCESO
// RUNDATE=, FECHA DE PROCESO
// FECHAIN=, FECHA INICIO
// FECHAFIN=, FECHA FINAL
```

6.1. PROGRAMAS DE LA PARTE BATCH

PROGRAMAS DEL CICLO DIARIO:

PROGRAMA. FUNCION

```
GNPA0100 ANALISIS DE LA CARTERA DIARIA
GNPA0200 CALCULO DE DIVIDENDOS PARA POLIZAS CON MOVIMIENTOS)
```

GNPA0300 RECALCULO DE LAS RENTAS
GNPA0400 CONSTITUCION DE RESERVA
GNPA0500 PAGO DE RENTA
GNPA0600 ACTUALIZACION DE RESERVA
GNPA0700 GENERACION DE POLIZAS POR DEFUNCION
GNPA0800 RESCATE DE POLIZAS TERMINADAS POR DEFUNCION
GNPA0900 REPORTE DE FONDOS DE INVERSION A FINANZAS
GNPA1000 IMPRESION DE RECIBOS
GNPA1100 IMPRESION DE CARATULAS

PROGRAMAS DEL CICLO MENSUAL:

PROGRAMA. FUNCION

GNPA1200 ANALISIS DE LA CARTERA MENSUAL
GNPA1300 REPORTE DE GASTOS ADMINISTRATIVOS POR DIF. EN INDIC
GNPA1400 CALCULO DE DIVIDENDOS AL CORTE DE MES
GNPA1500 REPORTE DE VALUACION DE RESERVAS

PROGRAMAS DE CICLO EVENTUAL:

PROGRAMA. FUNCION

GNPA1600 GENERACION DE LAS TABLAS DE VALORES CONMUTADOS

SUBROUTINAS DE USO GENERAL:

PROGRAMA FUNCION

GNPA2100 HOJA DE STATUS DE LAS POLIZAS (PARA PRUEBAS)
GNPA0002 SUBROUTINA DE CALCULO DE DIVIDENDOS
GNPA0003 SUBROUTINA DE VALUACION DE RESERVA
GNPA0004 SUBROUTINA DE GENERACION DE REGISTROS CONTABLES
GNPA0005 SUBROUTINA DE CALCULO DE LA ANUALIDAD
GNPA0006 SUBROUTINA DE IMPRESION DE LA POLIZA
GNPA0007 SUBROUTINA DE IMPRESION DEL RECIBO
GNPA0008 SUBROUTINA DE IMPRESION DEL ESTADO DE CUENTA DE DIV.
GNPA0009 SUBROUTINA DE INGRESO DE PRIMAS A LA RES.MAT
GNPA0010 SUBROUTINA DE CINTA A BANCOS

7. SUMARIO DE HERRAMIENTAS UTILIZADAS.

En esta sección se describen las herramientas de software utilizadas para la creación del SPA, estas son:

- Lenguajes de programación.
- Herramientas de Main Frame.
- Herramientas en línea.
- Herramientas batch.
- Herramientas de PC.

Lenguajes de programación:

- PASCAL - Para el desarrollo del módulo de ESTUDIOS.
- COBOL - Desarrollo de programas batch.
- ADS - Desarrollo de programas en línea.
- JCL - Pseudolenguaje para construcción de procesos batch.
- CULPRIT - Construcción de reportes varios.

Las herramientas en línea que se utilizaron son las siguientes:

Herramientas IDMS:

- IDD - Creación y manipulación de código fuente.
- OLM - Creación y manipulación de mapas.
- OLQ - Consultas estructuradas a la base de datos.
- ADSG - Creación y manipulación de diálogos.
- TCF - Conmutación entre las diversas herramientas IDMS.

20 Herramientas no IDMS:

- TSO - Opciones de tiempo compartido (plataforma de desarrollo)
- ISPF - Editor y ayudas varias.
- SDSF - Monitor de procesos batch.

Las herramientas batch que se utilizaron son las siguientes:

Herramientas IDMS:

- IDMSDMCL - Precompilador para programas cobol que accesan IDMS.
- IDMSRPTS - Obtiene un reporte de la estructura de la base de datos.
- IDMSADSO- Obtiene listados de los Diálogos y componentes asociados.
- IDMSDDDL - Corre sesiones de IDD desde ambiente batch.
- IDMSOLQB - Corre sesiones de OLQ desde ambiente batch.
- IDMSBGEN - Crea y compila diálogos (Sesiones ADSG) desde batch.
- IDMSBMAP - Crea y compila mapas (Sesiones OLM) desde batch
- IDMSUCFB - Corre sesiones de IDMS/UCF desde batch.
- IDMSPFIX - Permite corregir ciertos errores que se generan cuando existe alguna terminación anormal en la base de datos.

20 Herramientas no IDMS:

ICKCBL00 - Compilador de programas Cobol.
 IEWL -Ligador y generador de módulos de carga (ejecutables).
 IEGBGENER -Copia e imprime archivos secuenciales.
 IEBCOPY -Respalda y recupera archivos secuenciales y particionados.
 IEFBR14 -Permite crear y borrar archivos no VSAM.
 IDCAMS -Define, y manipula archivos VSAM.
 SEARCH -Efectúa búsquedas de cadenas en bibliotecas y archivos.

HERRAMIENTAS DE PC:

El sistema de pensiones aseguradas no fue construida únicamente con instrumentos de Main Frame. Como se dijo anteriormente, el sistema de pensiones cuenta con un componente o módulo en PC, este módulo es el de ESTUDIOS para la realización de estudios actuariales como paso previo a la Emisión de la póliza. El Módulo de estudios fue desarrollado en PASCAL, y además se emplearon una serie de herramientas en PC para la elaboración de documentos, prototipos, diseño y diagramas.

A continuación se muestra la rutina para el cálculo de pensiones ciertas, elaborada en PASCAL para el módulo de estudios en PC.

```

{--- PCierta [K-]}
{
{ PENSION CIERTA
{ Recibe Vigencia
{ Condiciones generales : Anticipada/Vencida, Número de pagos por año }
{-----}}
Function PCierta(N : Integer) : Double;
Var
  TVar : Double;
Begin
  TVar := (1 - RaiseTo((1 + IP),-(M * N))) / IP;
  Case Vencida Of
    False : PCierta := (1 + IP) * TVar;
    True : PCierta := TVar;
  End;
End;
  
```

Una herramienta de mucha utilidad que se transformó en gran productividad de trabajo, fue el uso combinado de emuladores de terminales Main Frame en PC y Transmisores de Archivos. Estas herramientas sirvieron para el desarrollo de programas en PC y transmisión a Main Frame, evitando con esto, sesiones en Main Frame con la consecuencia lucha por recursos de procesador.

Herramientas de PC

FLOW CHART - Para desarrollo de diagramas de flujo de datos, diagramas entidad relación, cartas de estructura, etc.
 DR HALO - Para desarrollo de diagramas varios.
 ISPF/PC - Edición de documentos y programas.
 VENTURA - Editor de alta presentación.
 DEMO II - Desarrollo de prototipos.
 DBASE - Seguimiento a problemas y solicitudes de cambio al Sistema de pensiones aseguradas.
 FTTERM - File Transfer and TERMINal emulator: Transmisor de archivos de PC a Main Frame y

viceversa, y emulador de terminales.
MC3270 - Transmisor de archivos PC - Main Frame y emulador de terminales.

8. PRUEBAS

La verificación del ciclo de vida es el proceso de determinar el grado con el cual los productos de trabajo de una fase dada del ciclo de desarrollo cumplen las especificaciones de las fases previas. Durante la etapa de pruebas, debemos asegurarnos de que el producto que ha sido desarrollado, hace exactamente lo que se supone debe hacer (según los documentos de análisis) Para el sistema de pensiones aseguradas, identificamos dos fases principales de pruebas, estas son:

- Pruebas individuales o unitarias.
- Pruebas de integración.

Las pruebas individuales o unitarias, se realizaron inmediatamente después de la codificación de cada uno de los submódulos. El objetivo de estas pruebas era comprobar que los programas funcionaban bien y sin error.

Las pruebas de integración se realizaron por medio de una matriz de prueba, que como ya se mencionó, es un método para simular la operación del sistema como si se encontrara en plena etapa de operación. La matriz de prueba se construye planteando una serie de casos típicos y casos extremos que pudiesen presentarse en la vida diaria del sistema.

Estos casos se plantean en un papel cuadrículado, colocándose en cada renglón una póliza hipotética con ciertas características, y en cada columna se colocan los días de operación del sistema, de tal manera que la intersección de un renglón con una columna identifican un evento, la matriz de prueba plantea por medio de sus columnas y sus renglones una serie de casos o condiciones iniciales, y en las celdas internas de la matriz (los eventos), se plantean los resultados esperados. Para probar el sistema con la matriz de prueba, se inicializa el ambiente del sistema, con los archivos vacíos, y se capturan los casos de prueba.

En este momento se empieza a validar el correcto funcionamiento del sistema. Al procesarse los ciclos de la parte batch, se analizan los resultados y las condiciones de las pólizas. En un solo día de pruebas, se pueden simular varios días de operación del sistema.

En promedio se procesaron 4 ciclos diarios por día de prueba. Al finalizar la prueba, se adaptaron varios cambios solicitados por el usuario, de tal manera que el procesamiento de la matriz sirvió como un prototipo de operación del sistema, y gracias a ello se detectaron varias condiciones que debían ser cambiadas antes de liberar el sistema a producción.

El sistema de pensiones aseguradas fue probado y cumplió con éxito con 3 matrices diferentes de prueba, cada una con 20 pólizas, 20 ciclos diarios, 4 ciclos mensuales, y un ciclo eventual.

- 1. DESCRIPCION DE LA FASE DE OPERACION Y MANTENIMIENTO**
- 2. ADMINISTRACION DE CAMBIOS.**
- 3. COMO ACCESAR LA PARTE EN LINEA DEL SISTEMA DE PENSIONES ASEGURADAS**
- 4. FLUJO DE INFORMACION EN EL SISTEMA DE PENSIONES**

1. DESCRIPCION DE LA FASE DE OPERACION Y MANTENIMIENTO

La fase de operación y mantenimiento es la razón de ser de un sistema, de información, en esta, se cosechan los frutos de la temporada de siembra (desarrollo del sistema), es decir, es la etapa productiva del producto de software.

Sin embargo, no todo el desarrollo ha concluido, por el contrario, se inicia una nueva etapa de mantenimiento al producto de software, este mantenimiento incluye el mantenimiento correctivo a problemas no detectados en la etapa de pruebas, el mantenimiento preventivo, la adaptación a nuevo equipo de hardware (discos, impresoras), adaptación a nuevo software (sistema operativo, base de datos), adaptación a nuevas necesidades del usuario (nuevas funciones y nuevos datos), y en general a todo tipo de cambio necesario para mantener en operación al sistema de información.

El ciclo correcto de mantenimiento, comprende las fases de análisis, diseño, instrumentación, documentación y pruebas. Con el correspondiente control de cambios y planeación de proyectos. El período de vida común de un producto de programación es de uno a tres años en su desarrollo, y de cinco a quince en su uso, incluyendo el mantenimiento.

Las actividades de mantenimiento gastan más recursos que las actividades de desarrollo, el mantenimiento se distribuye a lo largo de un período mayor, por lo que el esfuerzo por unidad de tiempo es menor. Un gran porcentaje del esfuerzo se dedica a mejorar el producto. Las actividades de prueba consumen cerca del 50% del tiempo de la fase de desarrollo. Una de las razones principales en el retraso de un proyecto y sobrepasar su costo, se debe al no asignar suficiente tiempo y recursos a las actividades de prueba integrales y aceptación.

La instrumentación, normalmente considerada como la actividad predominante de la construcción de sistemas, requiere solo del 10% del tiempo total de un proyecto ordinario. La meta del análisis, diseño e instrumentación será aligerar la carga de pruebas y mantenimiento de un sistema. En la construcción de sistemas más allá del uso personal, son esenciales los procedimientos sistematizados, la documentación estándar y las revisiones formales. Por ejemplo, el sistema operativo OS/360 de IBM se desarrollo con 5000 programadores durante cinco años.

Generalmente en la etapa de mantenimiento, no se cuenta con ninguna documentación para auxiliar en las labores de mantenimiento al código fuente, por ello, se debe incluir en el programa fuente, la documentación necesaria para disminuir este efecto.

Por otro lado también se requiere contar con una documentación completa y actualizada de cada uno de los módulos del sistema. La persona que realice cualquier cambio al sistema, debe entender el diseño de este, tan bien, como lo entendía quien realizó el sistema.

Por ello, una metodología formal y común a ambas personas, facilita el trabajo de mantenimiento, y lo hace posible con el mínimo de riesgos. Quien modifica el programa no solo debe ser capaz de entender la metodología de diseño empleada en la creación del software, sino también debe ser capaz de reproducirla.

Cuando un producto de software recibe continuas modificaciones realizadas por diferentes personas con diferentes metodologías, el producto se va degradando al punto tal, que es imposible de seguir modificando.

Como es de suponer, un producto de programación se va deteriorando, con forme aumenta el numero de modificaciones que se le practican, de manera tal, que el producto llega a ser inoperable o más caro de mantener que de reelaborar. En este momento se cierra el ciclo de vida del desarrollo de sistemas, termina

la etapa productiva de un sistema y se inicia el desarrollo de un nuevo sistema que cumpla de mejor manera a las nuevas circunstancias.

2. ADMINISTRACION DE CAMBIOS

Cuando un sistema ha sido desarrollado y puesto en producción, se inicia la etapa de operación desde el punto de vista del usuario, pero desde el punto de vista del constructor se inicia la etapa de mantenimiento, en la que se deben instrumentar una metodología para controlar la vida y evolución del sistema. Cuando no existe una metodología de control de cambios al sistema, se generan sistemas gigantes, incomprensibles, y difíciles de modificar.

Una correcta metodología para la fase de mantenimiento debe incluir los siguientes puntos:

- Deben existir al menos dos ambientes totalmente separados con programas y archivos independientes, un ambiente para producción y otro ambiente para desarrollo. Toda modificación se hará primero sobre el ambiente de desarrollo, una vez probado y aceptado se procederá a liberarlo a producción.

- Las liberaciones a producción deben ser previamente calendarizadas, de tal manera, que no se libere un cambio cada vez que este se solicita, sino en su lugar, se liberen un conjunto de cambios dando lugar a una nueva versión del sistema.

- Se debe minimizar el número de liberaciones a producción, Debido a que cada liberación a producción es un riesgo, pues se cambian, añaden o borran parámetros, programas y archivos en un ambiente productivo estable que maneja datos y cantidades reales.

- Todo cambio, debe pasar por todas las fases del ciclo de vida de desarrollo, estas son: Análisis, planeación, diseño, instrumentación, y pruebas. Además de incluir la correspondiente documentación de soporte.

- Toda modificación al sistema debe ser solicitada y documentada por medio de un requerimiento formal (especificación de requerimientos). Este requerimiento debe ser numerado para instrumentar un control de cambios formal.

- Todo cambio a un programa fuente, debe incluir su documentación correspondiente inmerso en el código, donde se especifique a que se debe el cambio, quien y cuando lo realizó y sobre que solicitud de cambio se realizó (número de control).

- Se debe llevar un expediente por programa fuente, donde se plasme la historia del programa, fecha y autor, descripción, algoritmo o pseudocódigo, tanto de su creación como de los cambios subsecuentes.

- Este expediente debe estar organizado de tal manera, que permita estructurar un expediente general por módulo del sistema, y un expediente del sistema en sí.

- Cuando el número e impacto de las modificaciones sean suficientes como para haber alterado en un 30% la definición inicial del sistema se debe hacer una revisión general tanto conceptual como de diseño y si el caso lo amerita, debe pensarse en la construcción de un nuevo sistema. Debe tenerse presente que en sistemas que han sufrido muchas modificaciones, se degrada el rendimiento y se generan una gran incertidumbre sobre el correcto funcionamiento. En estos casos, puede resultar más económico un nuevo desarrollo que el seguir soportando sistemas obsoletos.

- Aún y se tenga una correcta administración de cambios, se deben considerar los casos donde los cambios sean solicitados en calidad de "Urgentes", aunque un "Urgente" casi siempre se genera por no haber actuado o previsto con anticipación y por ende es un error de administración de proyecto, un "Urgente" puede, en efecto, ser un caso impredecible y realmente urgente. En caso de "Urgentes" se debe canalizar el caso de la mejor manera para colaborar en la solución del problema, pero no se debe realizar nada sin su correspondiente documentación, solicitud de cambio, plan de cambio y diseño. Además el cambio debe ser realizado de en ambiente de desarrollo, probado y después liberado a producción.

Si se permite sean aplicados cambios sin metodología, se caerá inevitablemente en un sistema caótico, descontrolado y nada profesional. No hay pretexto para abandonar la metodología dentro de un entorno profesional de ingeniería de software. La metodología en software se elabora para hacer posible su subsistencia y evolución, si se abandona nos encontraremos de nuevo en las primeras épocas con la más incorrecta de las metodologías: programa y prueba.

3. COMO ACCESAR LA PARTE EN LINEA DEL SISTEMA DE PENSIONES

La parte en línea del sistema, se accesa desde el ambiente interactivo que proporciona IDMS. IDMS es un DC/DB, es decir:

DC.- "Data communication" (Manejador de telecomunicaciones).

DB.- "Data Base" (Manejador de base de datos)

En un ambiente IBM Main Frame, el acceso a ambientes interactivos se realiza en las terminales, por medio de VTAM. VTAM es un Software cuyas siglas significan "Virtual Telecommunications Access Method" (Método de acceso de telecomunicaciones virtual), y trabaja en un nivel inferior al sistema operativo (MVS), su función es efectuar las capas de transporte de la arquitectura SNA.

SNA es la arquitectura de computadora que emplea IBM para sus Main Frames y sus siglas significan "System Network Architecture" (Arquitectura de la red del sistema) Tanto el acceso al ambiente de desarrollo en TSO, como al ambiente interactivo de IDMS, se efectúa a través de VTAM. VTAM trabaja para el sistema operativo (MVS).

Tanto VTAM como MVS trabajan para el usuario del computador, de tal manera que el usuario no se preocupa por funciones de bajo nivel, como lo son las comunicaciones de la terminales, Impresoras, disposición de espacio en disco o en memoria, etc. VTAM se encuentra activo en todo momento, inspeccionando las solicitudes que le realicen las terminales o el sistema operativo. Para accesa el ambiente interactivo de IDMS, basta con solicitar su ingreso a VTAM, a través del comando LOGON APPLID de VTAM. En cualquier terminal del sistema de cómputo, se tecllea lo siguiente;

LOGON APPLID(IDMS)

En este momento, VTAM verifica si IDMS es una Aplicación activa del sistema, y si la terminal tiene permiso de accederla, de ser así, invoca al manejador de comunicaciones de IDMS, y este presenta la siguiente pantalla.

V93 ENTER NEXT TASK CODE: CINTA S10212 (DESARROLLO)

A lo cual contestamos con nuestra solicitud de acceso, e identificándonos con nuestra clave de usuario, (th2hrm en este caso)

V93 ENTER NEXT TASK CODE: CINTA S10212 (DESARROLLO) signon th2hrm

IDMS contesta solicitándonos nuestra contraseña

IDMS DC258002 V93 ENTER PASSWORD

Al introducir la contraseña, nos encontramos listos para trabajar con IDMS. Como IDMS se organiza por medio de diccionarios, debemos solicitar el ingreso al diccionario donde se encuentra la aplicación con la que vamos a trabajar (Pensiones aseguradas).

IDMS DC258003 V93 USER TH2HRM SIGNED ON LTERM LTVTM001 AT 14:58:07.33 92.235 V93 ENTER NEXT TASK CODE: CINTA S10212 (DESARROLLO)

dcuf dictname tstdic7

Y a continuación solicitamos sea iniciada la aplicación de Pensiones aseguradas, cuyo código de identificación es (PENSION) IDMS DC402009 V93 DICTNAME TSTDIC7 HAS BEEN SET

V93 ENTER NEXT TASK CODE: CINTA S10212 (DESARROLLO)

ads pension

En este momento se empieza a ejecutar la aplicación, y se presenta el Menú principal.

INDICE DE OPERACION

4. FLUJO DE INFORMACION EN EL SISTEMA PENSIONES ASEGURADAS

El flujo de información dentro del sistema es el siguiente:

- 1.- Puesta en operación.
- 2.- Operación diaria del sistema.
 - 2.1.- Registro de nuevos negocios.
 - 2.2.- Administración de negocios.
- 3.- Operación mensual.

I.- Puesta en operación.

Cuando el sistema es liberado a ambiente productivo, se deben realizar algunas actividades antes de que el sistema se encuentre listo para su uso diario, estas actividades son las siguientes:

- Captura de las funciones (Diálogos) del sistema.

Se requiere como primer paso dar de alta las funciones que se encontraran disponibles en el sistema, con el fin de que estas funciones puedan ser asignadas, de la forma en que convenga, a los usuarios del sistema. Esto se logra a través de una firma de usuario especial, predefinida e independiente del sistema de seguridad, que permite tener acceso al sistema, aún y no se tengan dados de alta ningún usuario en el.

- Captura de los usuarios del sistema.

En esta función se registran los usuarios que tendrán acceso al sistema, y también para que funciones tendrán acceso y para cuales no. Este paso requiere la previa existencia de la tabla de funciones.

- Inicialización de los números de póliza, de recibo y de cheque del sistema.

En este paso se inicializan con el valor de "uno" el número de póliza, recibo y cheque, a partir de este momento estos números se incrementarán de forma secuencial automáticamente. Estos números se capturan desde el catálogo generalizado de tablas, que es un componente del módulo de control.

- Captura de la tabla de Fondos de inversión.

La tabla de fondos de inversión es necesaria para poder dar de alta pólizas dentro del sistema, ya que cada póliza tiene asociado obligatoriamente un fondo de inversión. Existen cuatro fondos de inversión obligatorios según la nota técnica del producto de pensiones, estos fondos son:

DO — Para pensiones con indexación al dolar.

SN — Para pensiones sin indexación.

IN — Para pensiones indexadas a la inflación.

DI — Para el fondo de dividendos.

Pueden capturarse más fondos de inversión para pólizas indexadas a parámetros variables, los códigos que en esta tabla se definan para los fondos de inversión, esta asociado al código de fondo de inversión que se captura en la transacción de alta de datos de la póliza.

- Captura de la tabla de conceptos.

Esta tabla relaciona todos los conceptos susceptibles de afectar las reservas de las pólizas. Cuando la reserva de una póliza sufre un incremento debido por ejemplo a la compra de un prima adicional, o un decremento por ejemplo por un pago de renta, se registra dentro del sistema un "Movimiento" que se desglosa en varios asientos, cada uno de estos esta asociado a sus correspondientes "Conceptos", mismos que se capturan en este paso.

Los conceptos existentes son los siguientes:

Rentas netas

Aguinaldos netos.

Coberturas adicionales.

Recargos por gastos administrativos.

Recargos por comisiones.
Impuestos.
Retiro de reserva matemática.
Retiro de reserva de dividendos.
Primas iniciales.
Primas adicionales.
Subsidios sobre impuestos.

Los "Movimientos" no son paramétricos, es decir, no pueden ser capturados por el usuario. Los movimientos son constantes en el sistema, y tienen una relación uno a uno con cada una de las funciones de SPA.

A continuación se muestra la relación de "Movimientos" predefinidos:

Compra de prima inicial.
Compra de prima adicional.
Pago de renta.
Pago de pago especial.
Retiro de reserva matemática.
Retiro de reserva de dividendos.
Ingreso de dividendos.
Rescate total.

- Captura de la tabla de Probabilidad de muerte.

Esto se realiza desde la transacción en línea creada para tal efecto dentro del módulo de control. Solicitar al centro de cómputo, sea procesado el ciclo inicial del sistema. Este ciclo se encarga de crear las tablas actuariales para el cálculo de las pensiones, a partir de la tabla de probabilidad de muerte. El ciclo inicial puede ser procesado eventualmente en cualquier momento de la vida del sistema, si es que se quieren actualizar las tablas actuariales (Valores conmutados), por ello a este ciclo también se le conoce como "Ciclo eventual".

- Captura del resto de las tablas de soporte del sistema.

Estas tablas son: Tabla de impuestos, subsidios, zonas económicas, bancos, gjros de empresa, estados y oficinas.

2.- Operación diaria del sistema.

2.1.- Captura de nuevos negocios.

- Para poder dar de alta cualquier póliza, debe existir un contratante que la posea, por ello debe registrarse primero la información del contratante.

- A continuación se capturan los datos de la póliza, datos del asegurado, coberturas tablas y pagos especiales. El sistema realizará una validación de toda la información capturada, verificará con respecto a los valores del módulo de control (tablas de bancos, oficinas, etc.).

- Se capturan los beneficiarios de la póliza, y cuando ya toda la información este registrada, se puede proceder con el cálculo de la pensión.

- Se calcula la pensión según el monto de prima disponible, o bien, se calcula la prima necesaria según la pensión deseada.

- Se capturan los índices de los fondos de inversión, para cada uno de los índices definidos en la tabla de fondos de inversión. Esto es para poder efectuar la conversión de pesos a participaciones durante el ciclo diario.

- Se captura el tipo de cambio del dolar y el valor de la inflación, con el fin de poder efectuar pagos de rentas para las pólizas con la indexación correspondiente.

- Se procesa automáticamente y al finalizar el día, el ciclo diario, mismo que se encargará de completar las transacciones de alta de los nuevos negocios, constituirá las reservas, e imprimirá los documentos contractuales de las pólizas.

2.2.- Administración de los negocios.

- Una vez capturada la póliza y constituida su reserva, se puede proceder a realizar varias actividades de administración, entre las cuales se encuentran las siguientes:

Consulta de datos de emisión, cambios de datos de emisión, compra de montos adicionales, captura de defunciones, retiro de reserva, actualización de fórmula de dividendos, etc.

3.- Operación mensual.

Al finalizar el mes, y después de correr el ciclo diario, se procede con el procesamiento del ciclo mensual. Este ciclo realizará el cierre del mes con el cálculo de dividendos de todas las pólizas, generará varios reportes entre ellos el reporte de valuación de reservas, que es de suma importancia para la administración del negocio de pensiones.

FIG. (8.2)

DISTRIBUCION DEL ESFUERZO EN UN CICLO DE VIDA DE UN PROYECTO DE PROGRAMACION

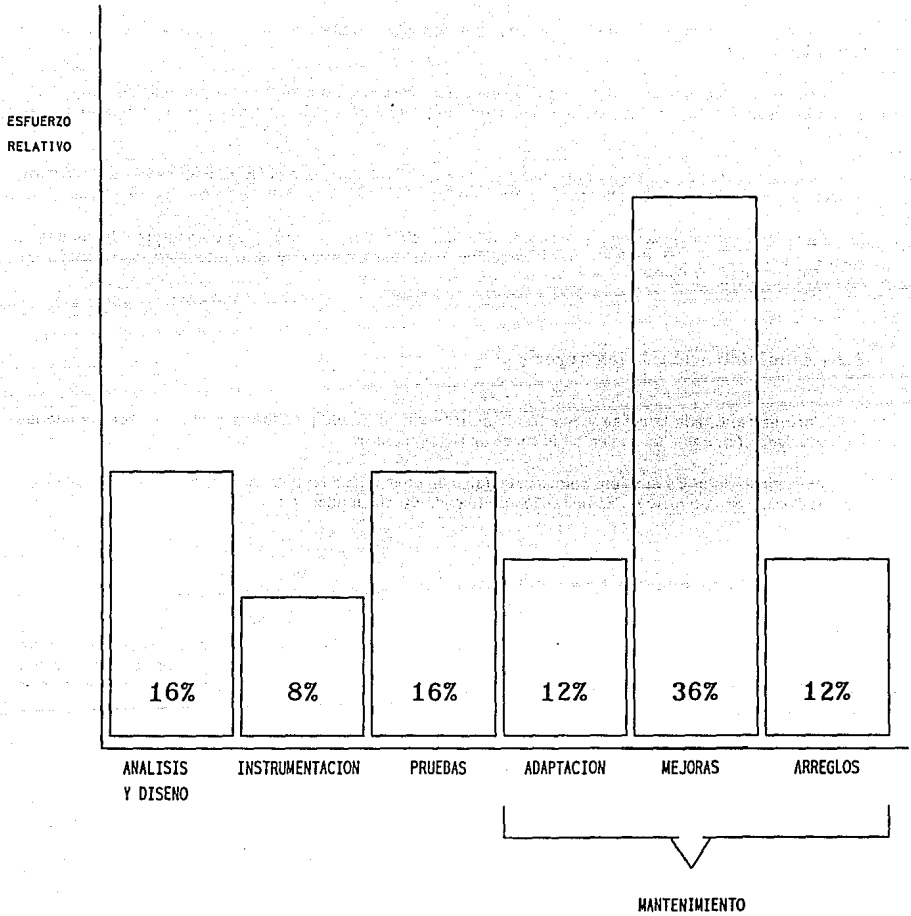
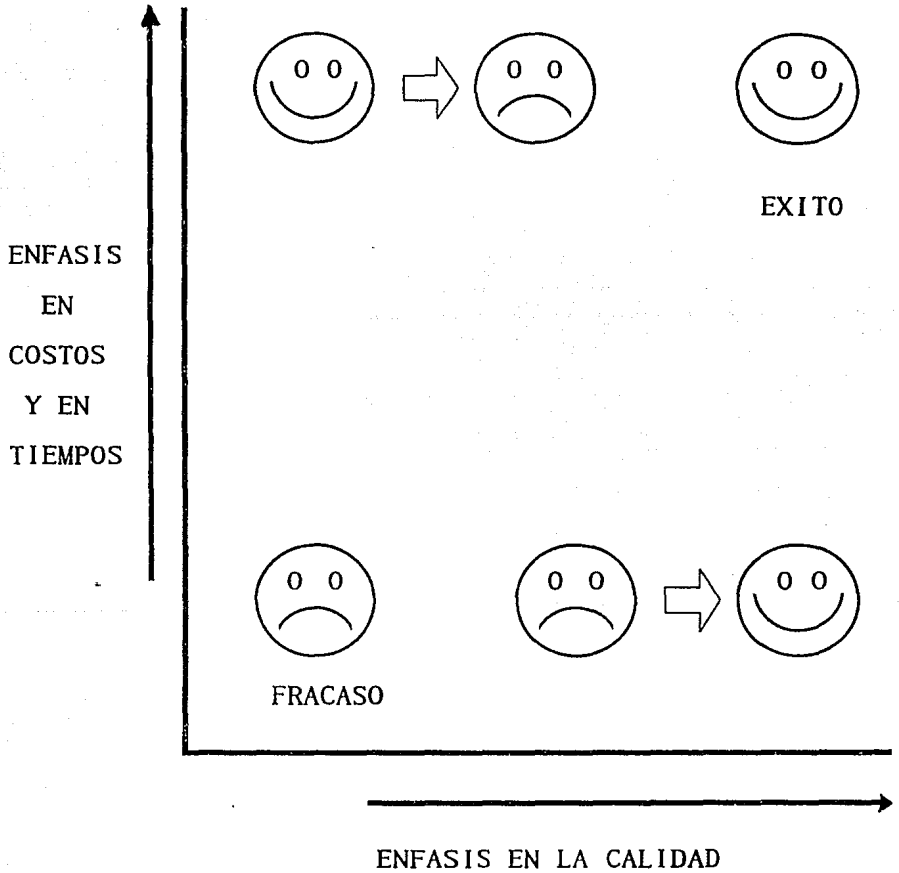


FIG. (8.1)

ESTRATEGIA DE LA DIRECCION DE PROYECTOS





OVERVIEW DEL SISTEMA DE PENSIONES

1. DESCRIPCION GENERAL

1.1. MODULO DE CONTROL

1.2. MODULO DE ESTUDIOS

1.3. MODULO DE EMISION

1.4. MODULO DE COBRANZA

1.5. MODULO DE ADMINISTRACION

1.6. MODULO DE ESTADISTICA

2. PARTES DEL SISTEMA PENSIONES ASEGURADAS

3. PARTE EN LINEA

4. PARTE BATCH

5. SUBMODULOS DE LA PARTE EN LINEA

5.1. MENU PRINCIPAL

5.2. SUBMODULOS DEL MODULO DE CONTROL

. TABLA DE PROBABILIDAD DE MUERTE

. TABLA DE RECARGOS ADMINISTRATIVOS Y POR COMISIONES

. TABLA DE ESTADOS

. TABLA DE GIROS DE EMPRESA

. TABLA DE AGENTES

. TABLA DE OFICINAS

. TABLA DE FUNCIONES DEL SISTEMA

. CATALOGO GENERALIZADO DE TABLAS

. TABLA DE BANCOS

. TABLA DE IMPUESTOS

. TABLA DE ZONAS ECONOMICAS

. TABLA DE SUBSIDIOS

. TABLA DE CONCEPTOS

. TABLA DE FONDOS DE INVERSION

. TABLA DE USUARIOS DEL SISTEMA

5.3. SUBMODULOS DEL MODULOS DE EMISION

. ALTA DE DATOS DEL CONTRATANTE

. ALTA DE DATOS DEL ASEGURADO

. ALTA DE DATOS DE LA PENSION

. ALTA DE PAGOS ESPECIALES

. ALTA DE COBERTURAS ADICIONALES

. ALTA DE BENEFICIARIOS

. CALCULO DE LA PENSION

5.4. SUBMODULOS DEL MODULO DE ADMINISTRACION

. SUBMODULO DE CONSULTAS

. SUBMODULO DE CAMBIOS

. COMPRA DE MONTOS ADICIONALES

. CAPTURA DE DEFUNCIONES

. CONSULTA DE POLIZAS GENERADAS POR DEFUCION

. ACTUALIZACION DE POLIZAS GENERADAS POR DEFUNCION

- . ENTREGA DE CHEQUES
- . CAPTURA DE INDICES POR FONDO
- . TIPO DE CAMBIO DEL DOLAR E INFLACION
- RETIROS DE RESERVA MATEMATICA
- . FORMULA DE DIVIDENDOS
- . RETIRO DE DIVIDENDOS
- . CONSULTA DE LA BITACORA

6. SUBMODULOS DE LA PARTE BATCH

6.1. SUBMODULOS DEL CICLO EVENTUAL

- . GENERACION DE TABLAS DE VALORES CONMUTADOS

6.2. SUBMODULOS DEL CICLO DIARIO

- . ANALISIS DE LA CARTERA
- . CÁLCULO DE DIVIDENDOS
- . RECALCULO DE RENTAS
- . CONSTITUCION DE RESERVAS
- . PAGO DE RENTAS
- . ACTUALIZACION DE MOVIMIENTOS
- . GENERACION DE POLIZAS POR DEFUNCION
- . RESCATE DE POLIZAS TERMINADAS POR DEFUNCION
- . REPORTE A FINANZAS
- . IMPRESION DE POLIZAS
- . IMPRESION DE RECIBOS

6.3. SUBMODULOS DEL CICLO MENSUAL

- . ANALISIS DE LA CARTERA**
- . REPORTE DE GASTOS ADMINISTRATIVOS**
- . CALCULO MENSUAL DE DIVIDENDOS**
- . REPORTE DE VALUACION DE RESERVAS**

7. INTERFASES

- 7.1. INTERFASE A BANCOS**
- 7.2. INTERFASE AL SISTEMA CORPORATIVO DE CONTABILIDAD**
- 7.3. INTERFASE AL SISTEMA CORPORATIVO DE AGENTES**
- 7.4. INTERFASE AL SISTEMA CORPORATIVO DE FINANZAS**

1. DESCRIPCION GENERAL

El sistema de pensiones aseguradas (SPA) se concibió pensando en su crecimiento, debido a que se esperaba tener un continuo aumento de facilidades en los planes de pensiones. @RELLENO1 = SPA esta compuesto de seis módulos principales, los cuales son: @PUNTUALIZA = Módulo de Control.

- Módulo de Estudios.
- Módulo de Emisión.
- Módulo de Cobranza.
- Módulo de Administración.
- Módulo de Estadísticas

En la operación, dos de ellos (Cobranza y estadísticas) no fueron construidos ya que el negocio aún no necesitaba de ellos, los otros cuatro fueron desarrollados y son tema de este trabajo.

1.1. MODULO DE CONTROL.

Se encarga de administrar al SPA en sí mismo, contiene una serie de submódulos, encargados cada uno de una función específica, como lo son: Control de acceso al sistema y seguridad, inventario de funciones y diálogos del sistema, tablas y tarifas actuariales, políticas y parámetros en general.

Este módulo permite darle cierta autonomía al sistema, ya que su operación queda en manos del usuario a través de pantallas que le dan acceso a actividades que normalmente se delegan al administrador de la base de datos.

En sistemas no paramétricos, cuando se requiere efectuar algún cambio en condiciones de operación, se tiene que afectar los programas fuente y se debe recompilar. Con un sistema paramétrico, se obtiene un mejor tiempo de respuesta a cambios y estos cambios pasan por menos manos, por lo que se minimizan también los errores.

1.2. MODULO DE ESTUDIOS.

Este módulo es el único módulo que no reside en el Main Frame. Se trata de un desarrollo en micro-computadora para realizar los estudios actuariales: cálculo de la pensión y de la prima. Se realizó empleando el lenguaje de programación PASCAL usando presentaciones de Menús descolgantes y ventanas.

El módulo no requiere control de usuarios ni mas administración que el que se le da a cualquier herramienta simple para micro-computadora, este módulo sirvió como el primer prototipo del sistema, ya que los cálculos que realiza se recodificaron para el Main frame como parte del Módulo de Emisión que se verá más adelante.

El usuario pudo tener contacto con el sistema casi de inmediato debido a la rapidez con la que se puede desarrollar en sistemas uni-usuarios. Sin embargo este módulo presenta grandes desventajas, entre las que se encuentra la obsolescencia en la que pueden incurrir las tablas y tarifas con las que se realizan los cálculos, por ello se piensa desarrollar en el futuro un módulo de interfase con el main frame para que estas tablas sean actualizables desde las tablas del sistema del Main Frame, de forma periódica.

Entre las ventajas de este módulo, se encuentra su fácil distribución, pues cada agente de seguros (quien finalmente venderá el producto financiero), podrá tener una copia de este módulo en sus computadoras personales.

1.3. MODULO DE EMISION.

Este módulo es el corazón del sistema, ya que en el se registran los nuevos negocios, así como se realizan los cálculos de pensión y se emiten las pólizas.

Este módulo esta comprendido por varios submódulos y se estima que será el de mayor índice de utilización en la primera etapa de vida en operación del sistema. Visto desde otro punto de vista, mientras más rápido y libre de error se puedan registrar nuevos negocios al sistema, más ingresos entraran a la compañía aseguradora.

1.4. MODULO DE COBRANZA

Este Módulo no se desarrolló, se pensó incluirlo pues se tiene pensado en un futuro habilitar las funciones de cobranza en pensiones, pero para su etapa inicial no se piensa tener esta cobranza.

Las pensiones se contratarán a través de una prima inicial única, y no se podrán hacer pagos parciales, por ello no se requiere la cobranza, pero en un futuro se planea establecer un sistema de ahorro en una cuenta de la compañía de tal forma que tal ahorro con sus intereses. sirvan para aportar la prima inicial.

1.5. MODULO DE ADMINISTRACION.

Este módulo comprende todas las transacciones y eventos que se aplicarán a los negocios ya constituidos, tales transacciones u movimientos son: Compra de primas adicionales (para incrementar los montos de la pensión), retiro parcial de la reserva matemática (a solicitud del asegurado), cambios y consultas a los negocios registrados en el módulo de emisión, etc.

Este módulo incrementa la complejidad del sistema, pues la combinación de las funciones de administración con las de emisión, y con las de administración misma, genera una gran cantidad de casos (pólizas) y eventos diferentes, y a todos ellos, el sistema debe ser capaz de responder de forma correcta y congruente.

1.6. MODULO DE ESTADISTICAS.

Aunque el sistema en sí incluye una serie de reportes, se planea incluir un módulo especial de estadísticas para la observación del comportamiento del mercado, quizás este módulo este compuesto por dos partes, la primera en el Main Frame, que trate de la extracción masiva de datos para estudio, y la segunda en micro-computadora para un análisis detallado de dicha información. Este Módulo esta proyectado para ser desarrollado una vez el sistema se encuentre en producción.

2. PARTES DEL SISTEMA DE PENSIONES ASEGURADAS

El sistema de pensiones aseguradas se subdivide en dos partes, estas son:

- 1.- Parte en línea (on line).
- 2.- Parte por lotes (batch).

La parte en línea se refiere a la parte que el usuario ve y con la que interactúa directamente. La parte en línea se accesa a través de una terminal (o pantalla), y con ella se dialoga o conversa de igual manera como se hace con un sistema de micro-computadora.

El acceso inicial es por medio de un menú de opciones, a través del cual se accesan las diferentes transacciones (funciones) del sistema. No se debe olvidar que este sistema se desarrollo en un entorno macro-computacional, y como todo sistema en este medio, debe diseñarse pensando en el ahorro máximo de recursos de procesador, para poder brindar conjuntamente con todo el resto de aplicaciones un adecuado tiempo de respuesta al operador (usuario), quien finalmente es quien se encuentra con la terminal frente al cliente o al agente de seguros.

Por ello todas las actividades que pudiesen ser relegadas a un procesamiento masivo nocturno se consideraron aparte, con el fin de compensar la carga de trabajo al computador y ponerlo a procesar lo complejo y tardado en la noche, hora en el que no se encuentra ningún operador usando el sistema en tiempo real (en línea).

Estas actividades incluyen la impresión de pólizas y en general de documentos contractuales, recibos de pagos de pensión, reportes, cálculos detallados de primas y pensiones, calculo de valuaciones, calculo de dividendos, etc. Algunas de estas funciones se requirieron también disponibles en el sistema en línea, pues aunque no sean solicitadas de forma frecuente, se deben realizarla en el mismo instante que se requieran sin tener que esperar al procesamiento nocturno para que se efectúen. Esto último conlleva a un sistema con un gran número de relaciones, pues algunos módulos que se diseñaron para su procesamiento nocturno por lotes (Batch), se adecuaron para que pudiesen ser reutilizados también en línea.

Estas adaptaciones brindaron gran movilidad y dinamismo al sistema, pues para efectuar un cálculo de dividendos a una póliza en especial, no solo se podrá realizar a petición especial desde una transacción en línea (dentro del módulo de administración), sino también se podrá contar con dicho cálculo de forma automática con el procesamiento batch. Dentro de las rutinas comunes al ambiente Batch como el ambiente en línea, se encuentran: Cálculo de primas, Cálculo de dividendos, Valuación de reservas.

El sistema Batch aunque no se vea, pues no presenta pantallas al usuario del sistema, y su procesamiento sea automático (ejecutado diariamente por el operador del centro de cómputo), no deja de ser de gran importancia y gran complejidad.

Cabe mencionar que la parte batch del sistema, fue la última en desarrollarse, siendo esta parte la mas compleja y pesada desde el punto de vista de consumo de recursos humanos (tiempo de programación), De tal manera que el módulo de estudios sirvió como prototipo inicial, el sistema en línea sirvió como segundo prototipo, y ya aceptados ambos se procedió con la parte batch.

3. PARTE EN LINEA

En un ambiente de base de datos IDMS, las transacciones en línea se instrumentan por medio de "Diálogos". Se desarrolló un Diálogo por cada uno de los submódulos del sistema, un Diálogo es la unidad básica funcional dentro del sistema de bases de datos IDMS, el Diálogo esta compuesto por un código ejecutable y una pantalla (o mapa), que desarrollan cierta conversación con el usuario de ahí su nombre).

Los módulos que comprenden la parte en línea del sistema de pensiones son los siguientes:

- 1.- Módulo de emisión.
- 2.- Módulo de control.
- 3.- Módulo de administración.

La descripción de cada uno de los submódulos se detalla mas adelante.

4. PARTE BATCH:

La parte batch se comprende de tres procesos independientes, cada uno de ellos compuesto por un conjunto de programas que corren secuencialmente uno tras de otro. Estos procesos son:

- 1.- Ciclo Inicial, y eventual.
- 2.- Ciclo Diario
- 3.- Ciclo Mensual. La descripción de cada uno de los submódulos se detalla mas adelante.

5. SUBMÓDULOS DE LA PARTE EN LINEA:

La parte en línea del sistema de pensiones, se accesa a través del ambiente interactivo de IDMS (tal y como se describe en el capítulo de operación).

La primera pantalla que el sistema de pensiones presenta, es el menú principal, desde donde se pueden accesar todos los submódulos de la parte en línea. El menú permite el acceso directo a todas las funciones de la parte en línea, con el propósito de eliminar los submenús, ya que la presencia de estos requeriría mayor número de ENTERS. (TECLA ENTER)

En un ambiente Main Frame el tiempo de respuesta es un parámetro básico de diseño, y se debe diseñar pensando en obtener el mejor tiempo de respuesta. Una terminal no utiliza tiempo del procesador sino hasta que se solicita por medio de una tecla de solicitud de atención ("Attention key") o un lápiz óptico.

La tecla de ENTER es la tecla que por convención en el mundo de la computación, realiza dicha solicitud de atención. Al solicitar una atención del procesador, entramos en la competencia por el uso de recursos de cómputo con el resto de los usuarios del sistema, estos usuarios varían desde otras sesiones en línea (terminales, impresoras, etc.) hasta procesos batch corriendo en el "Background". Por lo que a cada solicitud de atención del procesador, la terminal queda momentáneamente bloqueada (sin posibilidad de trabajar en ella), hasta que el procesador cumple con su solicitud.

Al tiempo de retardo entre la solicitud de atención (ENTER), y la respuesta del sistema, se le conoce como tiempo de respuesta. Cabe mencionar, que en un ambiente IBM, no solo la tecla de ENTER realiza solicitud de atención al procesador, sino también las teclas conocidas como teclas de función, estas son de PF1 a PF12 o PF24 según el tipo de la terminal, también la tecla de CLEAR y ATN realizan dicha solicitud. Por lo anterior, podemos observar como es que podemos invocar diferentes rutinas en un diálogo, según la tecla de atención que hallamos empleado.

A cada conjunto de instrucciones en un Diálogo de IDMS se le conoce como RESPONSSE PROCESS, y cada uno de ellos esta asociado a una tecla de atención, que en ambiente IDMS se conocen como RESPONSSE KEYS.

5.1. MENU PRINCIPAL

El menú del sistema de pensiones aseguradas, se implemento en una pantalla paginable, es decir, para poderla observar en su totalidad debemos utilizar varias pantallas ya que la información que se muestra no cabe en una sola. Para avanzar y retroceder páginas (pantallas) empleamos las teclas PF7 y PF8 respectivamente.

El menú del sistema de pensiones aseguradas se presenta a continuación,

92/08/22 SISTEMA DE PENSIONES ASEGURADAS TH2HRM

14:58 * MENU PRINCIPAL * PAG: 01

OPERADOR: HUMBERTO RAMOS MEJIA

1 ALTA DE CONTRATANTE	A	16 TABLA DE CONCEPTOS DE RESERVA DL
2 ALTA DE LA PENSION	B	17 TABLA FONDOS DE INVERSION DM
3 ALTA DEL BENEFICIARIO	C	18 TABLA DE SUBSIDIOS DN
4 CALCULO DE LA ANUALIDAD	CA	19 OPERADORES E
5 TABLA PROBABILIDAD DE MUERTE	DA	20 CONSULTA DE CONTRATANTE HA
6 TABLA DE RECARGOS	DB	21 CONSULTA DE ASEGURADO HB
7 TABLA DE ESTADOS	DC	22 CONSULTA DE PAGOS ESPECIALES HC
8 TABLA DE GIROS DE EMPRESA	DD	23 CONSULTA DE OTRAS COBERTURAS HD
9 TABLA DE AGENTES	DE	24 CONSULTAS DE LA PENSION HE
10 TABLA DE OFICINAS	DF	25 CONSULTA DE BENEFICIARIOS HF
11 FUNCIONES DEL SISTEMA	DG	26 CONSULTA A HISTORICO I
12 CATALOGO GENERAL DE TABLAS	DH	27 CAMBIO A CONTRATANTE JA
13 TABLA DE BANCOS	DI	28 CAMBIO A ASEGURADO JB
14 TABLA DE IMPUESTOS	DJ	29 CAMBIO A PAGOS ESPECIALES JC
15 TABLA DE ZONAS ECONOMICAS	DK	30 CAMBIO A OTRAS COBERTURAS JD

FUNCION: 000

ENTER= CONTINUA F1= AYUDA F7/F8= RET/AVANZA CLR= FIN

TECLEE LA OPCION SELECCIONADA

Al teclear PF8, avanzamos a la segunda pantalla del menú, esta segunda pantalla se muestra a continuación.

OPERADOR: HUMBERTO RAMOS MEJIA

31 CAMBIO A LA PENSION	JE
32 CAMBIO A BENEFICIARIO	JF
33 COMPRA DE MONTOS ADICIONALES	K
34 CAPTURA DE DEFUNCIONES	LA
35 CONSULTA DE POLIZAS GEN. X DEF LB	
36 ACTUALIZACION DE POLIZAS GEN. LC	
37 ENTREGA DE CHEQUES	M
38 CAPTURA DE INDICES POR FONDO	N
39 TABLA DE DOLAR E INFLACION	Q
40 RETIROS DE RESERVA MATEMATICA	R
41 FORMULA DE DIVIDENDOS	SA
42 RETIRO DE DIVIDENDOS	SB
43 CALCULO DE DIVIDENDOS	SC
45 CONSULTA DE BITACORA	U

FUNCION: 000

ENTER= CONTINUA F1= AYUDA F7/F8= RET/AVANZA CLR= FIN

TECLEE LA OPCION SELECCIONADA

En el campo FUNCION, capturamos el número de la función que deseamos accesar y a continuación ENTER.

5.2. SUBMODULOS DEL MODULO DE CONTROL

- TABLA DE PROBABILIDAD DE MUERTE.

Este submódulo tiene por objetivo el facilitar la captura, modificación y consulta de la tabla de probabilidad de muerte al usuario del sistema.

Esta tabla es la base para el desarrollo del resto de las tablas actuariales que se emplean para el cálculo de la pensión. Esta compuesta de una estimación en términos probabilísticos del Índice de muerte en cada una de las edades comprendidas entre 15 y 100 años. Antes de los 15 años no se puede contratar un seguro de pensión, después de los 100 años se considera una probabilidad del 100% de fallecimiento.

La pantalla de la tabla de probabilidad de muerte se muestra a continuación.

GPSODDAA0 TABLA PROBABILIDAD DE MUERTE FUNCION: 000

EDAD	QX	EDAD	QX	EDAD	QX	EDAD	QX	EDAD	QX	EDAD	QX
15: 0.00053	16: 0.00055	17: 0.00056	18: 0.00058	19: 0.00060	20: 0.00062						
21: 0.00064	22: 0.00067	23: 0.00070	24: 0.00073	25: 0.00076	26: 0.00080						
27: 0.00084	28: 0.00089	29: 0.00094	30: 0.00100	31: 0.00106	32: 0.00113						
33: 0.00121	34: 0.00129	35: 0.00139	36: 0.00149	37: 0.00160	38: 0.00173						
39: 0.00187	40: 0.00202	41: 0.00222	42: 0.00248	43: 0.00280	44: 0.00318						
45: 0.00362	46: 0.00411	47: 0.00465	48: 0.00524	49: 0.00588	50: 0.00655						
51: 0.00727	52: 0.00803	53: 0.00884	54: 0.00968	55: 0.01056	56: 0.01149						
57: 0.01248	58: 0.01347	59: 0.01454	60: 0.01566	61: 0.01686	62: 0.01819						
63: 0.01968	64: 0.02128	65: 0.02306	66: 0.02503	67: 0.02719	68: 0.02957						
69: 0.03220	70: 0.03509	71: 0.03827	72: 0.04177	73: 0.04562	74: 0.04985						
75: 0.05450	76: 0.05960	77: 0.06521	78: 0.07136	79: 0.07811	80: 0.08550						
81: 0.09359	82: 0.10244	83: 0.11211	84: 0.12266	85: 0.13417							
88: 0.14670	87: 0.16033	88: 0.17512	89: 0.19115	90: 0.20848							
91: 0.22719	92: 0.24733	93: 0.26698	94: 0.29211	95: 0.31683							
96: 0.34312	97: 0.37097	98: 0.40035	99: 0.43119	100: 1.00000							

F4= ACTUALIZA F1= AYUDA CLR= FIN

INTRODUZCA LOS DATOS

donde: Qx representa la probabilidad de muerte a una determinada edad "x", y comúnmente se conoce como tasa de mortalidad. La tabla que se emplea es la "Annuity table 1949", esta tabla presenta una estimación conservadora sobre la probabilidad de muerte por así convenir tanto al asegurado como a la aseguradora. Esta transacción, también se encarga del cálculo de "lx". Partiendo de la tabla de probabilidad de muerte, se tiene una relación de probabilidad de muerte (qx), a una edad "x". Si consideramos una población inicial de 1,000,000 personas (Radix), tendremos que para la edad 0 años existen 1,000,000 personas, pero a los 15 años tendremos:

$$1,000,000 \times Q(15) = 1,000,000 \times 0.00053 = 530 \text{ muertes}$$

$$1,000,000 - 530 = 1,999,470 \text{ supervivientes}$$

A la cantidad de supervivientes a la edad "x" se le conoce como "lx" se puede generalizar el cálculo de lx como:

$$lx = lx-1 - (lx-1 \cdot qx-1)$$

- TABLA DE RECARGOS ADMINISTRATIVOS Y POR COMISIONES.

Este módulo permite crear una o varias tablas distintas de recargo a ser aplicados en el momento del cobro de la prima o en el momento del pago de las rentas. Estos recargos se aplican en dos modalidades distintas según elija el asegurado, estas modalidades son: Recargos iniciales, o recargos prorrateados.

Los recargos iniciales se cobran en el momento de ingresar la prima por concepto de compra del seguro de pensiones, esto es en el submódulo de "Constitución de reservas" del ciclo diario. Los recargos prorrateados se cobran de manera diferida en cada una de las rentas de la pensión, esto es en el submódulo de "Pago de rentas" en el ciclo diario.

En el módulo de "Captura de datos de la pensión" se selecciona que modalidad de recargos se utilizará, y también qué tabla de recargos se aplicará. Pueden existir mas de una tabla de recargos, la razón para que existan mas de una tabla, es para normar, bajo condiciones diferentes de cobro a distintos tipos de clientes, y de esta manera contar con una herramienta de selección de la cartera.

La tabla asocia un porcentaje de cobro sobre la prima o la renta (según sean recargos iniciales o prorrateados respectivamente), a cada rango de prima contratada. De tal manera que las primas altas pagarán menor porciento de recargos que las primas bajas.

GPS0DB0 TABLA DE RECARGOS POR ADMINISTRACION Y COMISIONES FUNCION: 000

ACCION: (A=ALTA /B=BAJA /D=DESPLEGADO) CODIGO: 01

PAGINA: 0001

FECHA	PRIMA	FORMA	PERIODICIDAD	FORMA	TIPO	PORCENTAJE	BAJA
VIGENCIA	TOPE	PAGO	CARGO	GRUPO	X.ADM	X.COM	(B)
99 01 01	000,009,999,999		BIMESTRAL	INICIAL		00.4000.40	
99 01 01	000,999,999,999		BIMESTRAL	INICIAL		00.4000.40	
99 01 01	000,009,999,999		TRIMESTRAL	INICIAL		00.3000.30	
99 01 01	000,999,999,999		TRIMESTRAL	INICIAL		00.3000.30	
99 01 01	000,009,999,999		SEMESTRAL	INICIAL		00.2000.20	
99 01 01	000,999,999,999		SEMESTRAL	INICIAL		00.2000.20	
99 01 01	000,009,999,999		ANUAL	PRORRAT		00.2000.10	
99 01 01	000,999,999,999		ANUAL	PRORRAT		00.2000.10	
99 01 01	000,000,999,999		ANUAL	INICIAL		00.1000.10	
99 01 01	000,999,999,999		ANUAL	INICIAL		00.1000.10	

F1= AYUDA F2= NUEVO.RENGLON F4= ACTUALIZA F7/F8= AVANZA/RET CLR= FIN

PULSE [F2] PARA NUEVOS RENGLONES, [F4] PARA ACTUALIZAR

Cada tabla de recargos tiene asociado un CODIGO como identificación, las tablas se pueden desplegar, borrar, o dar de alta, según lo que se seleccione en el campo de ACCION. En estas tablas vemos que para una póliza con recargos iniciales, pago de renta anual y con una prima inicial de 999,000,000, le corresponde un recargo por gastos de administración del 0.1% y otro 0.1% como recargo para el pago de comisiones al agente de seguros.

- TABLA DE ESTADOS.

En esta tablas se asocian los códigos de estado (dos dígitos numéricos) a su descripción larga (el nombre del estado de la república). Esta tabla se crea con el propósito de simplificar y evitar errores de captura, ya que en lugar de capturar el nombre del estado, se captura solo su código. El salario mínimo diario por estado, se emplea para el cálculo de impuestos. La captura simplificada, por medio de códigos, es de gran utilidad para garantizar una entrada de datos correcta, pues se minimizan errores "de dedo" y ortográficos, y por otro lado, se agiliza la captura. La desventaja es que exige que el usuario conozca todos estos códigos. Para usuarios expertos que trabajan todo el día con el sistema, esto último no representa gran reto. En las pantallas de captura se instrumenta una función asociada a una tecla (F4) que mostrará la lista de códigos posibles para el campo en el que se encuentre posicionado el cursor.

GPS0DDCO TABLA DE ESTADOS FUNCION: 000

PAGINA: 0001

20	CODIGO DESCRIPCION	SMGD BAJA (B)
20	01 AGUASCALIENTES	450,000
20	02 BAJA CALIFORNIA SUR	500,000
20	03 BAJA CALIFORNIA NORTE	500,000
20	04 CAMPECHE	400,000
20	05 CHIHUAHUA	550,000
20	06 COLIMA	450,000
20	07 DURANGO	500,000
20	08 GUANAJUATO	450,000
20	09 GUERRERO	450,000
20	10 MORELOS	450,000
20	11 OAXACA	450,000
20	12 PUEBLA	450,000
20	13 QUERETARO	450,000
20	14 QUINTANA ROO	400,000
20	15 SAN LUIS POTOSI	400,000
20	16 TABASCO	400,000

F1= AYUDA F2= NUEVO.RENGLON F4= ACTUALIZA F7/F8= AVANZARET CLR= FIN
 PULSE [F2] PARA NUEVOS RENGLONES, [F4] PARA ACTUALIZAR

- TABLA DE GIROS DE EMPRESA.

En esta tabla se asocian giros de empresa con un código de captura. Como las empresas pueden ser compradoras de seguros de pensiones para sus empleados, es importante conocer el giro de estas empresas, por fines estadísticos.

GPS0DD70 TABLA DE GIROS DE EMPRESA FUNCION: 000

PAGINA: 0001

CODIGO DESCRIPCION

1001 EDUCACION
1002 GOBIERNO
1003 TIENDA MENUDEO
1004 CASA DE BOLSA
1005 SERVICIOS PUBLICOS
1006 SERVICIOS INDEPENDIENTES

F1= AYUDA F2= NUEVO,REGLON F4= ACTUALIZA F7/F8= AVANZA/RET CLR= FIN

PULSE [F2] PARA NUEVOS REGLONES, [F4] PARA ACTUALIZAR

- TABLA DE AGENTES.

En esta tabla (relación o lista), se registran todos los agentes de seguros que tienen permiso para la venta del seguro de pensiones, con el fin de validar en el momento de captura de nuevos negocios, que el agente tenga licencia para la venta del seguro. Esta tabla contiene la clave del agente y su nombre. Esta tabla, fue sustituida mas adelante, por una interfase que se desarrollo entre el sistema de pensiones y el sistema corporativo de ventas y agentes.

GPS0DD70 TABLA DE AGENTES FUNCION: 000

PAGINA: 0001

CLAVE	NOMBRE
20921	EDGAR GARCES
34587	CUAUHTEMOC SAUCEDO
44444	HUMBERTO RAMOS
68989	DILIANA MONTIEL
98909	VICENTE ORTEGA
11111	RODOLFO RAMIREZ
34458	HIRAM TERRAZAS
98989	OSCAR IVAN LEPE
68789	FERNANDO VAZQUEZ

F1= AYUDA F2= NUEVO.REGLON F4= ACTUALIZA F7/F8= AVANZA/RET CLR= FIN
 PULSE [F2] PARA NUEVOS RENGLONES, [F4] PARA ACTUALIZAR

- TABLA DE OFICINAS.

En esta tabla se asocian las oficinas de la aseguradora con sus respectivos códigos. Para cada póliza se captura la oficina donde se contrato el plan de pensiones.

GPS0DD70 TABLA DE OFICINAS FUNCION: 000

PAGINA: 0001

OFICINA	NOMBRE
01	DEL VALLE
02	COYOACAN
03	INSURGENTES SUR
04	CUAUHTEMOC
05	DOCTORES
06	MIGUEL ANGEL DE QUEVEDO
07	PILARES
08	TLALPAN

F1= AYUDA F2= NUEVO.REGLON F4= ACTUALIZA F7/F8= AVANZA/RET CLR= FIN
 PULSE [F2] PARA NUEVOS RENGLONES, [F4] PARA ACTUALIZAR

- TABLA DE FUNCIONES DEL SISTEMA.

En esta tabla se lleva un relación de las funciones en línea que contiene el sistema, el nombre del Diálogo y la descripción de la función. Esta tabla sirve como entrada para el módulo de "Usuarios del sistema" ya que a cada usuario, se le asignan solo cierta cantidad de funciones del sistema. La tabla también sirve para la construcción dinámica del menú de acceso, ya que solo las funciones asignadas al usuario en sesión son presentadas en el menú.

GPS0DDG0 CATALOGO DE FUNCIONES FUNCION: 000

PAGINA: 0001

NIVEL DE ACCESO: 0=SIN.ACCESO 1=CONSULTA 2=CAMBIO 3=ALTA 4=BAJA

-[FUNCION]-[DIALOGO]-[D E S C R I P C I O N]-[NIVEL-MANTENIMIENTO]-

A	GPS0DA00	ALTA DE CONTRATANTE	X
B	GPS0DB00	ALTA DE LA PENSION	X
C	GPS0DC00	ALTA DEL BENEFICIARIO	X
CA	GPS0DCA0	CALCULO DE LA ANUALIDAD	X
DA	GPS0DDA0	TABLA PROBABILIDAD DE MUERTE	X
DB	GPS0ddb0	TABLA DE RECARGOS	X
DC	GPS0DDC0	TABLA DE ESTADOS	X
DD	GPS0DDD0	TABLA DE GIROS DE EMPRESA	X
DE	GPS0DDE0	TABLA DE AGENTES	X
DF	GPS0DDF0	TABLA DE OFICINAS	X
DG	GPS0DDG0	FUNCIONES DEL SISTEMA	X
DH	GPS0DDH0	CATALOGO GENERAL DE TABLAS	X
DI	GPS0DDI0	TABLA DE BANCOS	X
DJ	GPS0DDJ0	TABLA DE IMPUESTOS	X
DK	GPS0DDK0	TABLA DE ZONAS ECONOMICAS	X
DL	GPS0DDL0	TABLA DE CONCEPTOS DE RESERVA	X

F1= AYUDA F2= NUEVO.RENGLON F4= ACTUALIZA F7/FB= AVANZA/RET CLR= FIN

- CATALOGO GENERALIZADO DE TABLAS.

Este catalogo nos permite acceder a todo tipo de tabla del sistema, pero nos presenta la información sin formato. Esta función se crea para dar mayor facilidad en el manejo de tablas al administrador del sistema, quien es una persona con suficientes conocimientos técnicos como para trabajar con esta función. Todas las tablas del sistema, se almacenan en el registro GPS-TABLA que es un registro multiformato, siendo un formato para cada tipo de tabla.

Para acceder al registro GPS-TABLA de forma directa, se emplea un registro padre cuyo nombre es GPS-CATALOGO. Existe un registro GPS-CATALOGO para cada tipo diferente de tabla. Hay algunos tipos de tablas que tienen mas de una tabla, por ejemplo la tabla de recargos TIENDE A CONFUNDIR , MEJOR DECIR EXISTEN TABLAS MAESTRAS, POR QUE TAL VEZ SE CONTRADIGA CON QUE CADA TABLA DE LAS TABLAS TIENEN MAS DE UN REGISTRO GPS-CATALOGO y las tablas de valores conmutados. Un conjunto de tablas del mismo tipo, son poseídas por un mismo registro padre GPS-CATALOGO.

La transacción del "Catálogo generalizado", usa directamente la construcción lógica del sistema de tablas, por ello, se debe indicar primero con que tipo de tabla se va a trabajar, o lo que es lo mismo, cuál es la clave del registro GPS-CATALOGO a utilizar.

La OCURRENCIA es el número de la tabla deseada, por lo general es la número uno, pues casi todas las tablas solo presentan una ocurrencia, pero otras tablas presentan mas de una ocurrencia, por ejemplo la tabla de valores conmutados presenta 86 ocurrencias.

A través del catálogo generalizado se pueden consultar y modificar todas las tablas del módulo de control. De hecho este módulo es un tanto redundante.

Su justificación consiste en que a través de esta transacción se tiene acceso a todas las tablas del sistema, y no solo a las del módulo de control; entre las tablas que se pueden acceder y que no pertenecen al módulo de control se encuentran las tablas de valores conmutados, y las de número de póliza. En la pantalla que se muestra a continuación, se observa la tabla del "Número de póliza", esta tabla es especial, pues solo presenta una ocurrencia (un renglón).

Para ser mas precisos, esta no es una tabla en sí, mas bien, es solo una forma de emplear la potencia del sistema LA POTENCIA DEL AREA DE TABLAS O DEL MODULO DE ... de tablas para el almacenamiento de datos aislados, en este caso es el último número de póliza capturado en el sistema.

Cada vez que se captura un nuevo negocio, y se solicita su cálculo (Cálculo de la renta o Prima), el módulo de emisión consulta la tabla del número de póliza para determinar que número de póliza sigue, de esta forma la asignación del número de póliza, se realiza de forma automática y secuencial.

De forma similar a la tabla de número de póliza, existen las tablas de "Número de recibo", y de "Número de cheque" para la numeración de los recibos y cheques emitidos por el sistema por el concepto de pago de rentas.

GPS0DDH0 CATALOGO GENERALIZADO DE TABLAS FUNCION: 012

ACCION: (A=ALTA /B=BAJA /D=DESPLEGADO) PAGINA: 0001

CLAVE CATALOGO: NP USUARIO....: TH2HRM FECHA: 920821 @PANTALLA = OCURRENCIA....: 01
DESCRIPCION:

TABLA USUARIO DATOS (B)

NP01 TH2HRM 000000012

F1= AYUDA F2= NUEVO.RENGLON F4= ACTUALIZA F7/F8= AVANZA/RET CLR= FIN
PULSE [F2] PARA GENERAR TABLAS, [F4] PARA ACTUALIZAR

- TABLA DE BANCOS.

Catálogo de bancos susceptibles de efectuar pagos de pensión, gracias al acuerdo entre la aseguradora y dichos bancos. La facilidad de pagar pensiones por medio de cuentas en los bancos, nos presenta la necesidad de contar con una interfase hacia los sistemas computarizados de dichos bancos. La interfase es parte de los módulos del sistema que se procesan de manera batch.

GPS0DD70 TABLA DE BANCOS FUNCION: 000

PAGINA: 0001

BANCO CUENTA DESCRIPCION

SF 000000001 BANCA-SERFIN
BA 020200002 BANCOMER
BX 030030300 BANAMEX
CO 000000200 COMERMEX
BO 909090909 BANCO OBRERO
BJ 909090909 BANCO DEL EJERCITO

F1= AYUDA F2= NUEVO.RENGLON F4= ACTUALIZA F7/F8=AVA

NZA/RET CLR= FIN

PULSE [F2] PARA NUEVOS RENGLONES, [F4] PARA ACTUALIZAR

- TABLA DE IMPUESTOS.

Presenta el por ciento de impuestos que se aplicarán sobre el pago de pensiones en función al monto de la pensión. El cálculo de impuestos se realiza en el submódulo de "pago de rentas" del ciclo diario. Para realizar dicho cálculo se emplean la tabla de impuestos, la tabla de zonas económicas, la tabla de subsidios y el monto de la pensión.

Cabe hacer notar, que el cálculo de impuestos debe ser realizado cada vez que se paga una renta, ya que las rentas mismas cambian debido a los siguientes movimientos: Reindexación de la renta, Retiros parciales de la reserva, reinversión de dividendos en la reserva, compras de prima adicional.

El algoritmo para la realización del cálculo de impuestos se presenta a continuación:

1. Determinar la Pensión mensual equivalente.

Muchas pensiones podrán tener un pago de pensión mensual, por lo que la pensión mensual equivalente es igual a la pensión nominal, pero otras pensiones mensuales equivalentes tendrán que calcularse, pues podrán tener un pago bimestral, trimestral, semestral o anual.

2. Calcular la Parte Exenta de impuestos.

$$\text{Parte} = (\text{Número de SMGD deducibles}) \times (\text{Monto del SMGD})$$

Donde: SMGD = Salario mínimo general diario.

3. Calcular la parte gravable de impuestos.

$$\text{Parte gravable} = (\text{Pensión mensual equivalente}) - (\text{Parte Exenta})$$

4. Calcular la parte excedente.

$$\text{Excedente} = (\text{Parte gravable}) - (\text{Limite inferior de impuesto})$$

5. Calcular el impuesto marginal.

$$\text{Impuesto marginal} = (\text{Excedente}) \times (\text{tasa de impuesto})$$

6. Calcular el impuesto de tarifa.

$$\text{Impuesto tarifa} = (\text{Impuesto marginal}) + (\text{cuota fija de imp.})$$

7. Calcular el acreditamiento de impuesto.

$$\text{Acreditamiento} = (\% \text{ acreditamiento}) \times (\text{SMGM})$$

Donde: SMGM = Salario Mínimo General mensual.

8. Calcular el ISPT (\$QUE ES ?);

$$\text{ISPT} = (\text{Impuesto de tarifa}) - \text{Acreditamiento}$$

9. Calcular el subsidio.

$$\text{Subsidio} = ((\text{Impuesto marginal}) \times (\text{tasa de subsidio})) + (\text{Cuota fija de subsidio})$$

10. Calcular el total a pagar por la pensión mensual equivalente

$$\text{Total a pagar} = (\text{Pensión mensual equivalente}) - \text{ISPT} + \text{Subsidio}$$

11. Convertir la pensión mensual equivalente a la periodicidad inicial de la pensión.

LIMITE-INFERIOR	LIMITE-SUPERIOR	CUOTA-FIJA	TASA BAJA (B)
000,000,000.00	000,000,009.00	0,000,000,100	10
000,000,009.00	000,000,099.00	0,000,000,100	10
000,000,099.00	000,000,999.00	0,000,000,200	10
000,000,999.00	000,009,999.00	0,000,000,200	10
000,009,999.00	000,099,999.00	0,000,000,300	10
000,099,999.00	009,999,999.00	0,000,000,300	10
000,999,999.00	099,999,999.00	0,000,000,400	10
009,999,999.00	999,999,999.00	0,000,000,400	10

F1= AYUDA F2= NUEVO.RENGLON F4= ACTUALIZA F7/F8= AVANZA/RET CLR= FIN
PULSE [F2] PARA NUEVOS RENGLONES, [F4] PARA ACTUALIZAR

- TABLA DE ZONAS ECONOMICAS.

Presenta una relación de zonas económicas con su respectivo salario mínimo general mensual (SMGM), pues este dato se emplea para el cálculo de impuestos.

ZONA-ECONOMICA	SMGM	SMGD	NUMERO	SMGD	PORCENTAJE BAJA (B)
	DEDUCIBLES	ACREDITACION			
1	100,000,000	1,000,000	10	10	
2	100,100,000	1,100,000	10	10	
3	100,100,000	1,100,000	10	10	
4	100,200,000	1,200,000	10	10	
5	100,200,000	1,200,000	10	10	
6	100,300,000	1,300,000	10	10	

F1= AYUDA F2= NUEVO.RENGLON F4= ACTUALIZA F7/F8= AVANZA/RET CLR= FIN
PULSE [F2] PARA NUEVOS RENGLONES, [F4] PARA ACTUALIZAR

- TABLA DE SUBSIDIOS.

Presenta el porcentaje de subsidio que se aplicará a la pensión en función del monto de la pensión.

GPS0DDNO TABLA DE SUBSIDIOS (80-A) FUNCION: 000
PAGINA: 0001

LIMITE-INFERIOR	LIMITE-SUPERIOR	CUOTA-FIJA	SUBSIDIO/IMPUESTO (B)
000,000,000.00	000,000,009.00	100	10.00
000,000,009.00	000,000,099.00	100	10.00
000,000,099.00	000,000,999.00	100	10.00
000,000,999.00	000,009,999.00	100	10.00
000,009,999.00	000,099,999.00	100	10.00
000,099,999.00	009,999,999.00	100	10.00
000,999,999.00	099,999,999.00	100	10.00
009,999,999.00	999,999,999.00	100	10.00

F1= AYUDA F2= NUEVO.RENGLON F4= ACTUALIZA F7/F8= AVANZA/RET CLR= FIN
PULSE [F2] PARA NUEVOS RENGLONES, [F4] PARA ACTUALIZAR

- TABLA DE CONCEPTOS.

Esta es una de las funciones mas poderosas del sistema, ya que permite la captura dinámica de conceptos a ser considerados en todo movimiento económico sobre las reservas de las pólizas. Esta tabla no se almacena en el registro de GPS-TABLA, sino en el GPS-CONCEPTO, este registro es uno de los pilares de la base de datos y conjuntamente con el registro GPS-MOVIMIENTO permite la ubicación dinámica y flexible de nuevos conceptos que afectan a las reservas. Cada MOVIMIENTO a la reserva se registra de manera automática, desglosándose en CONCEPTOS que después se presentarán en los recibos y en la interfase hacia el sistema contable corporativo de la aseguradora.

GPS0DDL0 TABLA DE CONCEPTOS FUNCION: 000
PAGINA: 0001

ID	DESCRIPCION	BAJA (B)
101	RENTAS	
102	AGUINALDOS	
103	COBERTURAS ADICIONALES	104 GASTOS ADMINISTRATIVOS
105	COMISIONES	
106	IMPUESTOS	
107	RETIROS DE RESERVA MATEMATICA	
108	RETIROS DE RESERVA DE DIVIDENDOS	
109	PRIMAS INICIALES	
201	PRIMAS ADICIONALES	

F1= AYUDA F2= NUEVO.RENGLON F4= ACTUALIZA F7/F8= AVANZA/RET CLR= FIN
PULSE [F2] PARA NUEVOS RENGLONES, [F4] PARA ACTUALIZAR

- TABLA DE FONDOS DE INVERSION.

Esta función es muy importante, pues permite la captura dinámica de fondos de inversión dentro del sistema de pensiones. Cada póliza se constituye a través de una reserva económica, misma que se invierte en los fondos de inversión. La inversión de todas las reservas en fondos de diversos inversion, permite participar con mayor índice de utilidad en el mercado de valores. del anterior párrafo confuso.

Para cada póliza se le asocia un fondo de inversión, mas para el asegurado esto no se percibe, ya que el asegurado recibe su pensión en adición con los dividendos que se obtengan del uso productivo de su reserva. En la pantalla de captura de la pensión, se pide sea capturado el código del fondo de inversión en el que se invertirá la reserva de dicha póliza.

La tabla de fondos de inversión permite la administración del registro GPS-FONDO, y este a su vez permite acceder de manera directa las reservas de las pólizas con el mismo fondo de inversión. El registro GPS-FONDO es la clave para la generación del "Reporte a finanzas" del ciclo diario, que como se verá mas adelante, es una función muy importante para el sistema de pensiones.

Dicha tabla se presenta a continuación :

GPS0DDM0 TABLA DE FONDOS DE INVERSION FUNCION: 000		
PAGINA: 0001		
ID	DESCRIPCION	DIFEREN.
DO	FONDO DE INVERSION PARA POLIZAS INDEXADAS AL DOLAR	100
IN	FONDO DE INVERSION PARA POLIZAS INDEXADAS A LA INFLACION	100
SN	FONDO DE INVERSION PARA POLIZAS SIN INDEXACION	100
VA	FONDO DE INVERSION PARA POLIZAS INDEXADAS VARIABLEMENTE	100

F1= AYUDA F2= NUEVO.RENGLON F4= ACTUALIZA F7/F8= AVANZA/RET CLR= FIN
 PULSE [F2] PARA NUEVOS RENGLONES, [F4] PARA ACTUALIZAR

- TABLA DE USUARIOS DEL SISTEMA.

En esta tabla se capturan las claves de acceso, nombres y referencias de los usuarios autorizados a usar el sistema, así como las funciones a las que tendrá permiso de acceso.

GPS0DE10 CATALOGO DE OPERADORES FUNCION: 000
 MOVIMIENTO : m (A = ALTA, M = MANTENIMIENTO) CVE OPERADOR:th2hrm

OPERADOR	NOMBRE	OFICINA
----------	--------	---------

INDIQUE LA "ACCION" DESEADA

En la siguiente pantalla, se dan de alta las funciones por usuarios, con su respectivo nivel de mantenimiento permitido a cada pantalla.

GPS0DE11 CATALOGO DE OPERADORES FUNCION:
 MOVIMIENTO : MENTENIMIENTO PAG: 0001
 CVE OPERADOR: TH2HRM NOMBRE: HUMBERTO RAMOS MEJIA
 VIGENCIA DEL: 000000 AL 000000 PASSWORD:
 OFICINA : 00 UBICACION :

NIVEL DE ACCESO ----				
FUNCION	DESCRIPCION	CONSULTAS	ALTAS	BAJAS CAMBIOS
A	ALTA DE CONTRATANTE	S	S	S S
B	ALTA DE LA PENSION	S	S	S S
C	ALTA DEL BENEFICIARIO	S	S	S S
CA	CALCULO DE LA ANUALIDAD	S	S	S S
DA	TABLA PROBABILIDAD DE MUERTE	S	S	S S S
DB	TABLA DE RECARGOS	S	S	S S
DC	TABLA DE ESTADOS	S	S	S S
DD	TABLA DE GIROS DE EMPRESA	S	S	S S S
DE	TABLA DE AGENTES	S	S	S S
DF	TABLA DE OFICINAS	S	S	S S
DG	FUNCIONES DEL SISTEMA	S	S	S S S

PF4 =DEFAULT PF7 =RETROCEDE PF8 =AVANZA

- ALTA DE DATOS DEL ASEGURADO.

En esta pantalla se dan de alta los datos generales del asegurado: nombre, domicilio, edad, sexo, referencias. Algunos de estos datos se emplean para realizar los cálculos actuariales de su pensión, muchos otros se emplean para destinar la correspondencia que se emita al asegurado.

GPS0DB00 INFORMACION DEL ASEGURADO- DATOS PERSONALES FUNCION: 002

POLIZA.....: CODIGO DE CONTRATANTE:

APELLIDO PA.:
APELLIDO MA.: NOMBRES.....:

DOMICILIO...: COL...:
POBLACION...: ESTADO: C.P.:
TELEFONO.....:

SEXO.....:
ESTADO CIVIL.....:
FECHA DE NACIMIENTO.:
R.F.C.....:
TIPO DE DOCUMENTO...: NUMERO DE REFERENCIA..:

ENTER= CONTINUA F1= AYUDA CLR= FIN
DIGITE DATOS DE ASEGURADO DATOS PERSONALES

- ALTA DE DATOS DE LA PENSION.

En esta pantalla se captura la información que tiene que ver con la pensión: tipo de pensión, si es mancomunada o no, si es diferida o no, monto de la prima disponible o de la renta deseada, fondo de inversión, etc. Esta pantalla se presenta automáticamente después de la captura de datos del asegurado.

20 GPS0DB10 INFORMACION DEL ASEGURADO- DATOS DE PENSION FUNCION: 002

STATUS(D/C):

POLIZA.....: 0000000000 APELLIDO A

PENSION: (P,G,M,D)....: [][][][]

P.VIGENCIA.....: 00

P.GARANTIA.....: 00

P.DIFERIMIENTO.....: 00

PERIODICIDAD DEL PAGO:

PAGO DIVIDENDOS(E/P/D):

TIPO DE INCREMENTO....:

TABLA DE DESCUENTOS...:

INICIAL/PRORRATEADA...:

AGENTE 1.....:

AGENTE 2.....:

AGENTE 3.....:

RENTA POR PERIODO....:

PRIMA TARIFA.....:

ANTIC/VENCIDO.....:

MONEDA.....: P

FONDO DE INVERSION...:

PAGA IMPUESTO (S/N)..:

COMISION DEL TOTAL 1.: 0.00

COMISION DEL TOTAL 2.: 0.00

COMISION DEL TOTAL 3.: 0.00

FORMA DE PAGO:

BANCO.....:

CUENTA:

DIA DE PAGO.....: 00

OFICINA OFF.....:

SUCURSAL.....:

TIPO DE CUENTA.....:

ENTER= CONTINUA F1= AYUDA F2= PAGOS-ESP F5= COBERTURAS CLR= FIN

DIGITE DATOS DE ASEGURADO DATOS DE PENSION

- ALTA DE PAGOS ESPECIALES.

En esta pantalla se capturan los pagos especiales o aguinaldos del plan de pensión. Estos aguinaldos existen para otorgar una renta mayor en determinados días del año, con el fin de que operen como el aguinaldo que normalmente estaba acostumbrado recibir el asegurado.

- ALTA DE COBERTURAS ADICIONALES.

En esta pantalla se capturan las coberturas adicionales que selecciona el asegurado, estas coberturas se pagan automáticamente con parte del monto de la pensión, y pueden ser seguros por gastos médicos, seguros de vida, etc.

- ALTA DE BENEFICIARIOS.

Se capturan los datos generales, y edades de los beneficiarios. Los beneficiarios pueden ser mancomunados o no mancomunados, pueden ser beneficiarios del asegurado principal o beneficiarios de otros beneficiarios.

GPS0DC00 INFORMACION DE BENEFICIARIOS FUNCION: 003
 POLIZA.....:

TIPO DE BENEFICIARIO: CONTINGENTE DE: 00
 APELLIDO PA.: APELLIDO MA.:
 NOMBRES.....:

DOMICILIO...: COL...:
 POBLACION...: ESTADO: C.P.:
 TELEFONO.....:

PORCENTAJE DE PENSION.: 0.00
 FECHA DE NACIMIENTO...: R.F.C.....: 000000
 TIPO DE DOCUMENTO...: NUMERO REFERENCIA:
 SEXO.....: PENSION (P,G,M,D): [] [] [] []
 PERIODO DE VIGENCIA...: 00 PERIODO GARANTIA.: 00

ENTER= CONTINUA F1= AYUDA F2= PAGOS-ESP F5= COBERTURAS CLR=FIN
 DIGITE DATOS DE BENEFICIARIO

- CALCULO DE LA PENSION.

Este es el módulo central de la parte en línea del sistema, ya que se encarga de realizar los cálculos actuariales para determinar el monto de la pensión en función de un prima dada, o ya bien, del cálculo de la prima necesaria para una pensión deseada. Al invocar esta función se inicia la vida productiva de la póliza en el sistema, ya que las pólizas calculadas en el transcurso del día, serán procesadas de forma batch para la constitución de sus reservas y la emisión de sus documentos contractuales. Este módulo invoca la ejecución de la rutina generalizada de cálculo de pensiones, que se instrumentó en un programa Cobol que también se accesa desde módulos del sistema de la parte batch. @RELLENO1 = El programa de cálculo generalizado de pensiones, es de los módulos más complejos del sistema pues es capaz de realizar los siguientes cálculos:

- Cálculo de pensiones simples.
- Cálculo de pensiones mancomunadas.
- Cálculo de pensiones mancomunadas dobles.
- Cálculo de pensiones con pagos especiales.
- Cálculo de pensiones con recargos iniciales.
- Cálculos combinados con todos los anteriores.

Submódulo de cambios.

- Cambios a datos del contratante.
- Cambios a datos del asegurado.
- Cambios a datos del beneficiario.
- Cambios a datos de pagos especiales.
- Cambios a datos de coberturas adicionales.
- Cambios a datos de la pensión.
- Compra de montos adicionales.
- Captura de defunciones.
- Consulta de pólizas generadas por defunción.
- Actualización de pólizas generadas por defunción.
- Entrega de cheques.
- Captura de índices por fondo.
- Tipo de cambio del dolar e inflación.
- Retiros de reserva matemática.
- Fórmula de dividendos.
- Retiro de dividendos.
- Consulta de bitácora.

A continuación se describen cada uno de ellos:

- SUBMODULO DE CONSULTAS.

Este módulo comprende una serie de transacciones para efectuar consultas a la información registrada en el módulo de emisión.

Las consultas y los cambios a los datos de emisión, se presentan por separado, para reducir riesgos de alterar la información cuando solo se deseaba consultar. Otra causa por la que se separa estos submódulos, es liberar al usuario de manejar intrincadas transacciones de Altas, Cambios y Consultas en una sola pantalla. Las pantallas de consultas, cambios y emisión, son en esencia iguales, solo cambian algunos aspectos secundarios, por esto mismo las transacciones de emisión, consultas y cambios presentan un gran cantidad de subrutinas comunes.

- SUBMODULO DE CAMBIOS.

En este módulo se realizan todos los cambios que se deban efectuar sobre la información registrada en emisión. Antes de permitir realizar cualquier cambio, el sistema solicita de forma explícita se le indiquen cuales datos son los que se desean alterar, con el propósito de impedir la modificación de los datos que no se mencionen y así evitar modificaciones involuntarias de datos. Cuando se alteran datos que están relacionados con el cálculo de la pensión, se marca la póliza con un estatus que indica que se debe recalcular la pensión, de tal forma que el ciclo diario recalcula automáticamente la póliza y genera la documentación necesaria para entregarse al asegurado.

Las transacciones de cambios, se dividen en dos pantallas, la primera es para indicar cuales serán los campos a alterar, y la segunda es para alterar los campos. La segunda pantalla presenta la misma imagen que las pantallas de emisión. La razón por la cual se presenta primero una pantalla para indicar cuales campos se alterarán, es para evitar modificaciones involuntarias a campos no deseados.

GPS0DJ80 MODIFICACION A INFORMACION DE ASEGURADOS FUNCION: 028

POLIZA.....: 1 SELECCIONE CON 'X' LOS CAMPOS A MODIFICAR

20 X APELLIDO PATERNO
20 APELLIDO MATERNO
20 NOMBRES
20 DOMICILIO
20 COLONIA
POBLACION
ESTADO
CODIGO POSTAL
TELEFONO
SEXO
ESTADO CIVIL
FECHA DE NACIMIENTO
R.F.C.
TIPO DE DOCUMENTO
NUMERO DE REFERENCIA

ENTER= CONTINUA F1= AYUDA CLR= FIN

GPS0DJFO MODIFICACION A INFORMACION DE BENEFICIARIOS FUNCION: 032

POLIZA.. 1 CONTINGENTE DE: 0 CONSECUTIVO: 01

x APELLIDO PATERNO PORCENTAJE DE LA PENSION
APELLIDO MATERNO FECHA DE NACIMIENTO
NOMBRES R.F.C.
20 TIPO DE DOCUMENTO NUMERO DE REFERENCIA
DOMICILIO SEXO
COLONIA TIPO DE PENSION
POBLACION VIGENCIA
ESTADO PERIODO DE GARANTIA
CODIGO POSTAL FORMA DE PAGO
TELEFONO TIPO DE BENEFICIARIO

MARQUE CON 'X' LOS CAMPOS A MODIFICAR

ENTER= CONTINUA F1= AYUDA CLR= FIN

MODIFICACION DE BENEFICIARIOS

GPS0MJAO MODIFICACION A INFORMACION DE CONTRATANTES FUNCION: 027

CODIGO.....: off SELECCIONE CON 'X' LOS CAMPOS A MODIFICAR

x RAZON SOCIAL

POOL
 DOMICILIO
 COLONIA
 POBLACION
 ESTADO
 CODIGO POSTAL
 TELEFONO
 REPRESENTANTE
 EXTENSION
 GIRO
 ZONA ECONOMICA
 ZONA GEOGRAFICA
 TIPO DE GRUPO

ENTER= CONTINUA F1= AYUDA CLR= FIN
 MODIFICACION DE CONTRATANTES

GPS0DJEO MODIFICACION A INFORMACION DE DATOS DE PENSION FUNCION: 031
 POLIZA.....: 1 SELECCIONE CON 'X' LOS CAMPOS A MODIFICAR

X TIPO DE PENSION TABLA DE DESCUENTOS
 VIGENCIA DIA DE PAGO
 PERIODO DE GARANTIA INICIAL/PRORRATEADO
 DIFERIMIENTO AGENTES
 FECHA DE EMISION COMISIONES DEL TOTAL
 IMPUESTO FORMA DE PAGO
 PERIODICIDAD DEL PAGO BANCO
 ANTICIPADO/VENCIDO SUCURSAL
 TIPO DE INCREMENTO CUENTA
 MONEDA TIPO DE CUENTA

ENTER= CONTINUA F1= AYUDA CLR= FIN

- COMPRA DE MONTOS ADICIONALES.

Esta transacción permite ingresar montos adicionales de prima a la prima inicial, para con esto, incrementar el monto de la pensión. Esta transacción genera un movimiento a la reserva y por lo mismo es procesada en el ciclo diario, para efectuar el incremento a la reserva en términos de participaciones.

GPS0MK00 COMPRA DE MONTOS ADICIONALES DE RENTA FUNCION: 033

POLIZA.....:

SE GENERARA UN MOVIMIENTO DE "PRIMA ADICIONAL", Y SE REIMPRIMIRA LA POLIZA
 NO SE ASIGNA NUEVO NUMERO DE POLIZA

FECHA DE APLICACION....: 920822

PRIMA ADICIONAL.....\$

CONFIRMACION.....\$

CODIGO DE RETORNO

ENTER= CONTINUAR

F1= AYUDA

CLR= FIN

TECLEE EL MONTO ADICIONAL

- CAPTURA DE DEFUNCIONES.

En caso de que el asegurado fallezca, se debe terminar el pago de las pensiones a este, para continuar efectuándolo a sus beneficiarios o en su defecto abonarles a estos el total de la reserva que aún existiese. En esta transacción se registra tal fallecimiento, dando origen al proceso de defunciones que realiza la parte "Batch" del sistema.

GPS0DLA0

CAPTURA DE DEFUNCIONES

FUNCION: 034

POLIZA.....: 0000000013

FECHA DE DEFUNCION: 920622

APELLIDO PA.: PRUEBA UNO

APELLIDO MA.: A

NOMBRES.....: A

DOMICILIO...: A

COL...: A

POBLACION...: A

ESTADO: 01 C.P.: 1 TELEFONO...: 1

SEXO.....: M ESTADO CIVIL.....: S

FECHA DE NACIMIENTO.: 700101

R.F.C.....: AAAA 700101

ENTER= CONTINUA

F1= AYUDA

F2= ACTUALIZA STATUS CLR= FIN

TECLEE LA FECHA Y F2 PARA REGISTRAR LA DEFUNCION.

- CONSULTA DE POLIZAS GENERADAS POR DEFUNCION.

En este módulo se pueden consultar cuantas y cuales fueron las pólizas que el sistema generó de manera automática en la parte "batch", debido a un proceso de defunción. Las pólizas que se generan automáticamente debido a una defunción, son las de los beneficiario contingentes de la póliza original.

- ACTUALIZACION DE POLIZAS GENERADAS POR DEFUNCION.

Para que una póliza generada por defunción se encuentre completa, es necesario registrar cierta información adicional que no se tenía presente en el momento de registrar la información de los beneficiarios (ahora asegurados), por ello se requiere actualizar la información para que las pólizas se encuentren totalmente definidas.

- ENTREGA DE CHEQUES.

En este módulo se registra la entrega y devolución de cheques que se realiza en las oficinas de la aseguradora. No todas las pólizas tienen su pago asociado a cheques emitidos en las oficinas de la institución. Existen varias opciones para efectuar los pagos por concepto de pago de pensión, dichas opciones se capturan en las pantallas de emisión. Estas opciones pueden ser pago de pensión por cuenta bancaria del asegurado, por cheque emitido al contratante o por cheque emitido al asegurado.

```

GPS0DM00      *** ENTREGA DE CHEQUES ***  FUNCION:
                +---(-FECHA-DE-ENTREGA )---+
ACCION: POLIZA: 0000000000 CHEQUE: 0000000000 INICIO: 000000 FIN:000000
(E = CHEQUES ENTREGADOS, N = CHEQUES NO ENTREGADOS, R = REGISTRO DE ENTREGA )
-----
NUM.CHEQUE EMISION ENTREGA  MONTO  NUM.POLIZA  DESTINATARIO

```

```

-----
ENTER= CONTINUA F1= AYUDA F2= CONFIRMA.ENTREGA F7/F8= RET/AVANZA CLR= FIN
CAPTURE LA ACCION A REALIZAR, [ ENTER ] PARA CONTINUAR

```

- CAPTURA DE INDICES POR FONDO.

En esta transacción se registra el valor de las participaciones por cada fondo de inversión del sistema, esta información permite realizar las conversiones pesos- participaciones y participaciones-pesos que requiere el sistema para interactuar con el mundo exterior. Cabe mencionar que las pantallas y documentos impresos se expresan en pesos, mientras que la información interna al sistema se expresa en participaciones.

- TIPO DE CAMBIO DE DOLAR E INFLACION.

En esta transacción se registra el tipo de cambio del dolar con respecto al peso, y el valor inflacionario por día con respecto al inicio del año. Esta información permite realizar la reindexación de las pólizas (en el ciclo diario), tanto las que se indexan según el tipo de cambio del dolar, como las que se indexan según el Índice inflacionario.

- RETIROS DE RESERVA MATEMATICA.

En esta transacción se realizan retiros parciales y totales sobre la reserva de la póliza. Los retiros generan un movimiento a la reserva, por lo que las pólizas con tales retiros, son procesadas en el ciclo diario, para efectuar la conversión del retiro a participaciones.

GPS0DR00 RESCATES FUNCION: 040

POLIZA.....: 10 MONTO DISPONIBLE

SE GENERARA UN MOVIMIENTO DE "RESCATE SOBRE LA RESERVA"
NO SE GENERA NUEVA POLIZA, SE REIMPRIME LA CARATULA DE EMISION

FECHA DEL RESCATE.: 92 08 22

MONTO DEL RESCATE.:

CONFIRMACION.....:

ESTAN CORRECTOS SUS DATOS S/N

ENTER= CONTINUA F1= AYUDA CLR= FIN

TECLEE EL NUMERO DE POLIZA

- FORMULA DE DIVIDENDOS.

En esta transacción se registra el porcentaje sobre dividendos que se cobrará por gastos de administración. Dicho porcentaje se fija en un 5% por omisión, pero puede ser modificado para cualquier póliza, según se vea el comportamiento del mercado.

GPS0JSA0 FORMULA DE DIVIDENDOS FUNCION: 041

POLIZA.....: 0000000001

FORMULA DE DIVIDENDOS: 95.00

ESTAN CORRECTOS LOS DATOS?

ENTER= CONTINUA F1= AYUDA F4= ACTUALIZA CLR= FIN

PRESIONE [F4] PARA ACTUALIZAR

- RETIRO DE DIVIDENDOS.

En esta transacción se realizan retiros sobre el fondo de dividendos. Las pólizas con un retiro de dividendos, son procesadas en el ciclo diario para efectuar la conversión de pesos a participaciones.

GPS0D700 RETIRO DE DIVIDENDOS FUNCION: 040

POLIZA.....: 10 MONTO DISPONIBLE

SE GENERARA UN MOVIMIENTO DE "RETIRO DE DIVIDENDOS"
NO SE GENERA NUEVA POLIZA, SE REIMPRIME LA CARATULA DE EMISION

FECHA DEL RETIRO...: 92 08 22

MONTO DEL RETIRO...

CONFIRMACION.....:

ESTAN CORRECTOS SUS DATOS S/N
ENTER= CONTINUA F1= AYUDA CLR= FIN
TECLEE EL NUMERO DE POLIZA

- CONSULTA DE BITACORA.

Todo movimiento en el sistema: Alta de registros, cambios, retiros de reserva, etc..., generan un registro auxiliar conocido como registro de "auditoría" o de "bitácora", en dicho registro se presenta el usuario, lugar y fecha del movimiento realizado en el sistema. Esta información es base para llevar un registro estadístico de actividades y un control sobre la información en el sistema. La información de estos registros de bitácora puede ser consultada por esta transacción. Se puede consultar por tipo de operación efectuada sobre el sistema (función), usuario, fecha o por póliza.

6. SUBMODULOS DE LA PARTE BATCH:**6.1. SUBMODULOS DEL CICLO EVENTUAL****- GENERACION DE TABLAS DE VALORES CONMUTADOS.**

Este proceso, se encarga de construir las tablas actuariales que se emplean para el cálculo de la pensión. El cálculo de una pensión se realiza en línea a través de un submódulo del módulo de Emisión, sin embargo, para poder determinar el monto de una renta (o pensión) dado una determinada prima, es necesario recurrir a una serie de fórmulas actuariales que emplean como parámetros principales el tipo de plan de pensiones deseado, la edad del asegurado y de sus beneficiarios.

Con estos datos se consultan unas tablas, conocidas como tablas de valores conmutados para finalmente determinar el monto de la pensión. Las tablas de valores conmutados son derivadas de la tabla de probabilidad de muerte.

Para pensiones mancomunadas existen varias tablas de valores conmutados. Debido a que los valores de las tablas de valores conmutados son función directa de la tabla de probabilidad de muerte, y la tabla de probabilidad de muerte puede variar en base a la experiencia reportada por el mercado asegurado, es necesario contar con un mecanismo para recalcular las tablas cada vez que se requiera.

El recalcado de las tablas de valores conmutados no es un proceso que pueda ejecutarse cada vez que se requiera realizar un cálculo de una pensión, simplemente no es un proceso viable en un computador de tiempo compartido como lo es un "Main Frame". Tampoco es necesario recalcular todas las tablas de valores conmutados para cada cálculo de pensión en particular. Es por ello que se instrumentó un proceso "Batch" especial que será procesado cada vez que sea alterada la tabla de probabilidad de muerte, y que almacenará los valores de las tablas actuariales en registros de la base de datos.

Cada vez que se requiera un cálculo de una pensión, se consultará a la base de datos en consulta de los valores conmutados que se requiera. Se considera en este caso, que el tiempo consumido y complejidad de proceso en hacer un acceso a disco, es menor al tiempo y de menor complejidad si es que se realizará un recalcado de la tabla de valores conmutados cada vez que se solicitará un cálculo de pensión. Los valores conmutados, son valores asociados a una edad "x", estos son D_x y N_x , los cuales se obtienen a partir de la siguiente fórmula:

$$D_x = l_x * (1 + i)^{-x}$$

$$N(x+t) = \text{SUM}(D_{x+t})$$

Donde:

l_x = Supervivientes a la edad "x",

(Este dato se calcula en la tabla de probabilidad de muerte del módulo de control).

i = Tasa de Interés técnico.

x = Edad del asegurado.

SUM = Sumatoria desde $t = 0$ hasta $t = 100$

Véase Fig. 9.12

6.2. SUBMÓDULOS DEL CICLO DIARIO

El ciclo diario, como su nombre lo indica, es un proceso que se realiza diariamente por las noche de manera "Batch", su función es completar los procesamientos de todas las transacciones que se desarrollaron en el transcurso del día.

El ciclo diario esta subdividido en los siguientes submódulos:

- Análisis de la cartera.
- Cálculo de dividendos.
- Recálculo de rentas.
- Constitución de reservas.
- Pago de rentas.
- Actualización de movimientos.
- Generación de pólizas por defunción.
- Rescate de pólizas terminadas por defunción.
- Reporte a finanzas.
- Impresión de pólizas.
- Impresión de recibos.

Fig. 9.09.

DESCRIPCION DE LOS SUBMODULOS DEL CICLO DIARIO:

- ANALISIS DE LA CARTERA.

Este proceso tiene por objetivo detectar todas las pólizas que deben ser procesadas en los siguientes submódulos del ciclo diario. Efectuando un único barrido general al archivo maestro de pólizas. De forma tal que se generan una serie de archivos temporales (tienen vida solo durante dure la ejecución del ciclo diario), que contienen una relación de pólizas y sus respectivos procesos que se les debe aplicar, una relación para cada diferente proceso.

- CALCULO DE DIVIDENDOS

Este módulo tiene por objeto el cálculo de dividendos de todas las pólizas que han sufrido algún movimiento en el transcurso del día. La compañía aseguradora comparte los dividendos generados por la correcta administración de los fondos de inversión. El cálculo de estos dividendos se realiza en base al valor de la participación y de fórmulas actuariales. Como el calcular dividendos a todas las pólizas todos los días generaría una sobrecarga de trabajo al computador, y por otro lado este dato no se requiere sino hasta que se realiza una disminución o aumento a la reserva de la póliza, se optó por realizar este cálculo solo para las pólizas que registraron algún movimiento a sus reservas. En el momento que una póliza registra un movimiento a su reserva, entonces se calculan los dividendos generados desde el último cálculo de dividendos que tuvo a la fecha, de esta forma, no se deja ningún período sin considerar para el cálculo de dividendos, el cálculo se efectúa solo cuando se requiera y no se sobrecarga al computador.

Los dividendos generados pueden ser pagados de tres formas diferentes al asegurado, según este halla elegido en el momento de contratación, el tipo de pago de dividendos se captura en las pantallas de emisión. El cálculo de dividendos también puede ser solicitado de manera explícita desde una transacción en línea del módulo de administración, esto nos hace ver la necesidad de crear un módulo de cálculo de dividendos común a la parte "Batch" y a la parte en línea. Este módulo común se implemento por medio de un programa

que puede ser invocada desde programas "batch", y también desde programas en línea. La complejidad para instrumentar un programa con tales características, nos llevó a innovar algunas estrategias de comunicación de programas Batch" con el ambiente en línea de la base de datos. El módulo de cálculo de la pensión también fue instrumentado de esta forma, ya que también existe la necesidad de efectuar dicho cálculo desde "batch" y desde el sistema en línea.

- RECALCULO DE LAS RENTAS.

Cada vez que se efectúa algún cambio a las pólizas, pueden resultar alterados algunos de los datos en los que está basado el cálculo de la anualidad. Si esto sucede, las pólizas deben recalcularse para ajustar el monto de la pensión. La póliza debe ser recalculada si es que se agregan beneficiarios contingentes, cambias las edades de estos o del asegurado principal, se realiza una compra de prima adicional o algún retiro de la reserva matemática. Este módulo, realiza el cálculo de la anualidad ($\text{Prima} = A \cdot \text{Renta}$) de manera automática, de esta forma ayuda al usuario para que este no tenga que solicitar el recalculation por su cuenta.

- CONSTITUCION DE RESERVAS.

Este submódulo, se encarga de constituir las reservas para cada póliza que haya sido ingresada al sistema durante el transcurso del día. Constituir una reserva, se refiere al hecho de ingresar la aportación efectuada por el asegurado (Prima) a un fondo común, este fondo común es administrado por la compañía aseguradora para obtener máximos rendimientos y seguridad económica. Si las aportaciones que realizan los asegurados, se ingresaran a cuentas bancarias, la compañía aseguradora no podría garantizar tasas de interés superiores a las bancarias, y en este caso, sería mejor inversión depositar las primas directamente al banco sin tener que comprar un plan de pensiones. Cuando una prima se ingresa a dicho fondo común, se realiza una conversión de pesos a participación, de tal forma que el sistema de pensiones administra únicamente participaciones, y solamente en el momento de entrar o salir dinero del sistema, se convierte a pesos. El valor de una participación en pesos, es capturado en el módulo de Administración de la parte en línea, y su valor se determina en la dirección de finanzas de la compañía aseguradora.

Esto nos hace ver la necesidad de contar con una interfase a la dirección de finanzas, esta interfase se realiza de forma semi-manual, ya que a dicha dirección se hace llegar un reporte diario del comportamiento de la cartera de pensiones, y en dicha dirección se encargan de administrar el submódulo de valores de la participación dentro del sistema de pensiones. La aportación realizada por cada asegurado no se pierde al conjuntarse con otras aportaciones de otros asegurados, al contrario, su administración se efectúa por separado, solo los totales son reportados a un fondo común. La aportación realizada por el asegurado (Prima), al momento de ingresarse al sistema, se convierte a participaciones, este proceso se conoce como constitución de la reserva, de tal forma que la prima inicial es igual a la reserva inicial de la póliza, la primera expresada en pesos y la segunda en participaciones. La razón de efectuar la constitución de la reserva hasta el ciclo nocturno, es que el tipo de cambio de la participación se captura hasta el final del día, después de haber observado el comportamiento del mercado durante el día. Entonces la conversión a participaciones debe realizarse tomando en cuenta el último valor ingresado de la participación, esto obliga a diferir la constitución de las reservas hasta finalizar el día.

- PAGO DE RENTAS.

Este módulo puede considerarse el corazón de la parte "Batch", pues es el encargado de efectuar el pago de las rentas a las asegurados que se encuentran en su etapa de jubilación. Por otro lado, el cálculo de la pensión puede considerarse el corazón del sistema de la parte en línea del sistema. Todo aumento o disminución a la reserva constituida de una póliza, es registrada en el sistema por medio de un registro especial denominado "movimiento", cada movimiento a su vez se desglosa en varios "Asientos", uno para cada "concepto" del "movimiento".

El pago de rentas genera el movimiento de pago de rentas, con un desglose por asientos relacionados con los siguientes conceptos: Renta, impuestos, subsidio, aguinaldos, coberturas adicionales, comisiones y gastos administrativos. El módulo de pago de rentas realiza también el cálculo de recargos por gastos de administración para pólizas con recargos prorrateados, también realiza el cálculo de impuestos sobre la renta y de subsidios.

Este módulo también incrementa las rentas de las pólizas. En el momento de contratación el asegurado decide que tipo de indexación desea ya sea incremento de su renta según incrementa la inflación, o según el tipo de cambio del dolar, o según algún otro parámetro de incremento.

El plan de pensiones proporciona un incremento periódico del monto de la renta, para evitar que las rentas se devalúen con el pasar del tiempo. El cálculo de la renta y de la prima toma en cuenta la tasa de interés supuesta para poder proporcionar dicha indexación.

Dentro del módulo de administración de la parte en línea del sistema, se proporciona una transacción para capturar el tipo de cambio del dolar y el valor de la inflación. Este registro permite realizar el incremento de la renta cada vez que se alcancen las políticas de incremento.

Las políticas de incremento también son registradas en el módulo del sistema en línea. Dentro del módulo de administración existe una transacción para registrar el valor de los fondos de inversión, debido a que el incremento en las rentas también puede ser en función de los fondos de inversión, dicha transacción también sirve para controlar el incremento de las rentas.

Este módulo también prepara los recibos de pago, los cheques y la información que será enviada a los bancos para el ingreso de primas directamente a cuentas de los asegurados.

En general la parte línea del sistema puede ser vista como la preparación para poder efectuar el pago de las rentas. Dentro de la parte en línea se registran los nuevos negocios cuyas rentas serán pagadas en el futuro, se calculan las rentas que se pagarán y se registran las políticas de pago de renta.

- ACTUALIZACION DE MOVIMIENTOS.

Debido a que el sistema administra únicamente participaciones, las operaciones de compra de prima adicional, retiro de reserva matemática y retiro de reserva de dividendos que se realizan en pesos, deben ser convertidas por el sistema a participaciones.

Este módulo se encarga de realizar la traducción de los movimientos realizados a las reservas de las pólizas desde el módulo de administración a participaciones. Por otro lado, también se encarga de preparar los recibos, cheque e información para los bancos para efectuar pagos en efectivo.

Dentro del módulo de pago de rentas y de actualización de movimientos, se tienen implícitas dos interfase, una es hacia los bancos que tengan convenio con la compañía aseguradora para realizar pago de rentas directamente a cuentas de los asegurados, y otra es hacia el sistema corporativo de contabilidad de la compañía aseguradora.

La interfase hacia los bancos se realiza por medio de una cinta que se genera en los procesos batch y que se entrega por correo a los bancos para que en ellos dicha información sea capturada a sus respectivos centros de cómputo.

La información sobre que bancos tienen convenio con la compañía aseguradora se encuentra en la tabla de bancos del módulo de tablas del sistema en línea. La interfase hacia el sistema corporativo de contabilidad, se instrumenta a través de un submódulo que es invocado desde diversas partes del sistema tanto en línea como en batch, dicho submódulo es invocado cada vez que existe un movimiento a las reservas y en general su función es traducir los movimientos y sus respectivos asientos a los asientos contables requeridos por la institución.

- GENERACION DE POLIZAS POR DEFUNCION.

El proceso de defunciones es en si todo un sistema que reside inmerso dentro del sistema de pensiones. Cuando fallece un asegurado que tenía vigente una póliza de pensiones, el resto de su reserva es abonado a los beneficiarios que se tuviesen asignados, ya sea por medio de un pago único (retiro total de reserva), o a través de la constitución automática de reservas individuales a cada uno de los beneficiarios.

La constitución automática de reservas lleva implícito la generación automática de pólizas de pensión a cada uno de los beneficiarios. Este proceso es el que realiza este módulo. Para que una póliza entre en el proceso de defunciones, se registra la defunción del asegurado principal en una transacción del módulo de administración.

El ciclo diario detecta que dicha póliza entra en un proceso de defunción, y procede a realizar un retiro total de la reserva del asegurado fallecido, para constituir con esta, las reservas de sus beneficiarios.

- RESCATE DE POLIZAS TERMINADAS POR DEFUNCION.

No siempre se podrá tener todos los datos de todos los beneficiarios que tuvieron pólizas generadas de forma automática debido a una defunción, entonces no se podrá liberar las pólizas al mismo tiempo. Cuando esto sucede, tampoco se puede realizar el rescate de la reserva de la póliza en defunción, sino hasta que se disponga de la totalidad de los datos para poder liberar las pólizas generadas.

Este módulo detecta el momento en que todas las pólizas generadas por una defunción, son completadas y liberadas. En este momento se realiza un rescate total sobre la reserva de la póliza original, con sus respectivos asientos contables.

- REPORTE A FINANZAS.

El reporte a finanzas es un reporte especial, que resume los movimientos del día por cada uno de los fondos de inversión, y los reporta a la dirección de finanzas de la institución. Este reporte sirve para evaluar la actividad del mercado de pensiones y es una herramienta determinante para dirigir las decisiones de inversión en el mercado de dinero, y por lo tanto, para fijar el valor de la participación dentro del sistema de pensiones.

- IMPRESIÓN DE RECIBOS.

Este submódulo imprime todos los recibos generados por concepto de pago de rentas y por retiros ya sea parciales o totales de las reservas de la póliza.

- IMPRESION DE POLIZAS.

Este submódulo tiene por objeto, imprimir todos los documentos contractuales generados por el ingreso de nuevos negocios o compras de prima adicional que se registraron durante el día.

6.3. SUBMODULOS DEL CICLO MENSUAL

El ciclo mensual es procesado el último día del mes, después de haber procesado el ciclo diario de ese mismo día. Su procesamiento es mensual debido a que realiza funciones orientadas a la administración del sistema y no tanto de operación diaria. Los submódulos de este ciclo son los siguientes:

- Análisis mensual de la cartera.
- Reporte de Gastos Administrativos.
- Cálculo mensual de dividendos.
- Reporte de valuación de reservas.

DESCRIPCION DE LOS SUBMODULOS DEL CICLO MENSUAL:

- ANALISIS MENSUAL DE LA CARTERA.

Este proceso, como el análisis diario de la cartera, se encarga de detectar todas las pólizas que sufrirán algún tipo de procesamiento subsiguiente en el resto del ciclo. Esto con el fin de optimizar el tiempo de barrido a los archivos maestros del sistema.

- REPORTE DE GASTOS ADMINISTRATIVOS.

Este reporte tiene por objeto, el calcular el monto por concepto de gastos administrativos que se han generado durante el mes en el sistema.

- CALCULO MENSUAL DE DIVIDENDOS.

Al final del mes, se realiza un cierre de la cartera del sistema de pensiones, esto obliga a realizar un calculo de dividendos de todas las pólizas aun y no hayan presentado movimientos durante el mes.

- REPORTE DE VALUACION DE RESERVAS.

Este reporte es de suma importancia, pues hace una evaluación general de que tan bien se encuentran administradas las reservas de las pólizas dentro del sistema. El reporte calcula por medio de una serie de fórmulas actuariales, la cantidad en la reserva que debiese existir para cada una de las pólizas, y luego compara esta cantidad con respecto a la cantidad que en realidad existe según el tipo de cambio de la participación. La diferencia es conocida como déficit o superávit del negocio, y es el objetivo final del reporte, (y del negocio también).

Este módulo presenta la mayor complejidad técnica del sistema, debido a que las fórmulas utilizadas para el cálculo de las reservas "teóricas", son muy complejas e incluyen llamadas a la rutina general del cálculo de las pensiones. Esta complejidad técnica se resolvió modularizando el problema, se crearon una serie de submódulos cada uno encargado de realizar parte del cálculo. De esta forma el programa general, solo es un monitor de actividades y quien finalmente reporta los resultados en forma de un reporte formateado y listo para entregar al usuario.

7. INTERFASES

Como todo sistema, el de pensiones aseguradas no es un sistema aislado, se relaciona con su medio externo a través de interfases. Las interfases del sistema de pensiones, se encuentran tanto en la parte batch como en la parte en línea, dichas interfases son las siguientes:

- Interfase a bancos.
- Interfase al Sistema corporativo de contabilidad.
- Interfase al Sistema corporativo de agentes.
- Interfase al Sistema corporativo de finanzas.

A continuación se describe cada una de las interfases.

7.1. INTERFASE A BANCOS.

Esta interfase se lleva a cabo por medio de dos pasos, el primero se desarrolla en la parte en línea y el segundo en la parte batch. En la parte en línea se capturan y se mantiene actualizada la tabla de bancos, los bancos son las instituciones financieras con las que la compañía aseguradora mantiene convenios para el depósito de las rentas de los asegurados. En la parte Batch, se genera una cinta con la información necesaria para que los abonos a las cuentas de los asegurados sean realizados directamente a los sistemas de Información de los bancos.

Dicha cinta es generada en el centro de cómputo de la compañía aseguradora, y es trasladada a los centros de cómputo de los bancos, para que a su vez los sistemas de información de los bancos, carguen la información a las cuentas de los asegurados. La cinta a bancos es generada por los submódulos de Pago de rentas, Actualización de movimientos, y Generación de dividendos de la parte batch del sistema de pensiones aseguradas.

7.2. INTERFASE AL SISTEMA CORPORATIVO DE CONTABILIDAD.

Todas las operaciones de entrada y salidas de efectivo del sistema de pensiones, deben ser registradas al sistema corporativo de contabilidad de la compañía aseguradora, para que esta a su vez reporte dichos movimientos a sus accionistas, al gobierno y a sus socios comerciales. La interfase al sistema de contabilidad, es la más robusta, requirió el desarrollo de un submódulo para dicha interfase, dicho submódulo contiene una parte en batch y otra en línea.

En la parte en línea, se capturan y mantienen actualizados los parámetros a ser usados en la parte batch, estos parámetros son las cuentas contables en las que se registrarán cada tipo de movimiento económico en el sistema de pensiones aseguradas.

Los movimientos económicos en el sistema de pensiones aseguradas, se realizan siempre sobre las reservas económicas de los asegurados, y se almacenan en el registro GPS-MOVIMIENTO de la base de datos.

En la parte batch, se convierte la información de los registros GPS-MOVIMIENTO y GPS-ASIENTO al formato de los registros que maneja el sistema corporativo de contabilidad. Esta conversión se lleva a cabo a través de un programa que es invocado cada vez que se registra un movimiento a las reservas económicas. Dicho programa, también es invocado desde la parte en línea del sistema, debido a que también desde la parte en línea se puede solicitar un cálculo de dividendos, mismo que genera movimientos a las reservas económicas.

7.3. INTERFASE AL SISTEMA CORPORATIVO DE AGENTES.

Esta interfase se realiza desde la parte en línea del sistema, y su función es validar que los agentes de seguros que se registren en las transacciones de emisión, tengan licencia para la venta del seguro de pensiones. Inicialmente se había diseñado una tabla en el módulo de control, en la que se registrarían los agentes con licencia para la venta del seguro de pensiones. Mas adelante se pensó en una interfase directa al sistema corporativo de agentes, ya que en este se registran dichas licencias.

La interfase al sistema corporativo de agentes se realizó por medio de un programa que invocan las transacciones de alta de datos de la pensión y Cambios a datos de la pensión. El programa simplemente valida en la base de datos del sistema corporativo de agentes, que el agente que se esta registrando en pensiones tenga la licencia.

7.4. INTERFASE AL SISTEMA CORPORATIVO DE FINANZAS.

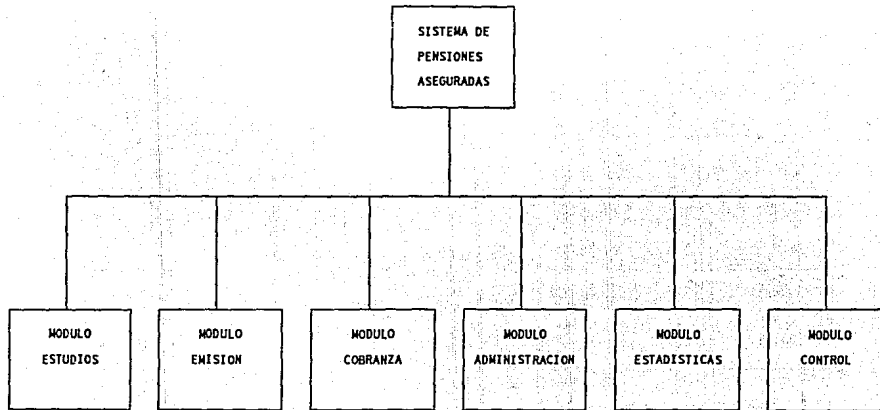
Desde la transacción de "Captura de índices por fondo" del módulo de administración se registran los valores de las participaciones de los diferentes fondos de inversión del sistema de pensiones.

El valor de las participaciones se captura desde la dirección de finanzas de la compañía aseguradora, esta dirección determina el valor de la participación en función de los resultados obtenidos en los instrumentos de inversión.

La dirección de finanzas determina cuantas participaciones y en que forma se invertirá en los instrumentos de inversión, en función de la información proporcionada por el "Reporte a finanzas" que se genera en el ciclo diario del sistema de pensiones. Dicho reporte es un sumario del comportamiento de las reservas durante el día, y sumariza las entradas y salidas de efectivo de los diferentes fondos de inversión del sistema de pensiones aseguradas.

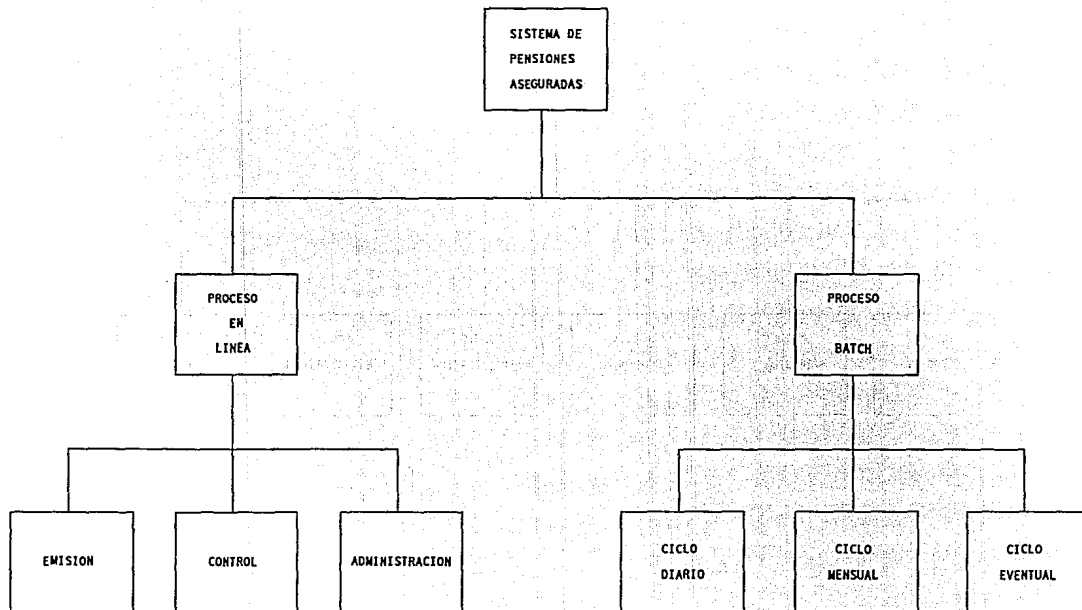
SISTEMA DE PENSIONES ASEGURADAS

MODULOS DEL SISTEMA



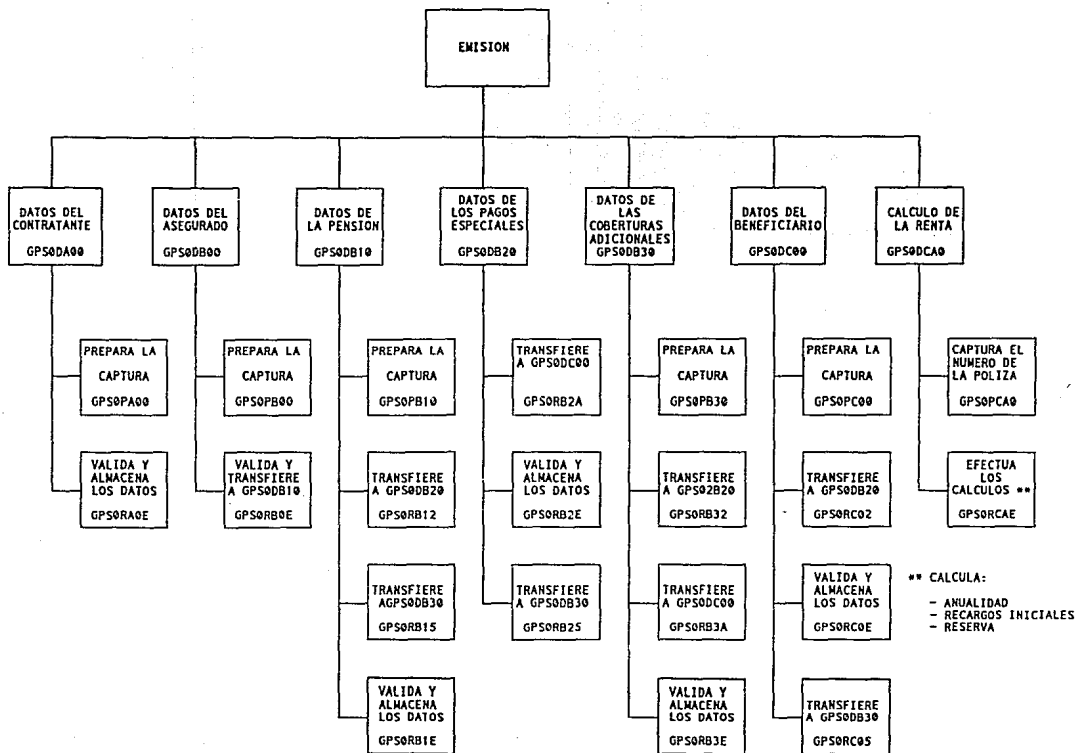
SISTEMA DE PENSIONES ASEGURADAS

DIAGRAMA DE FUNCIONES



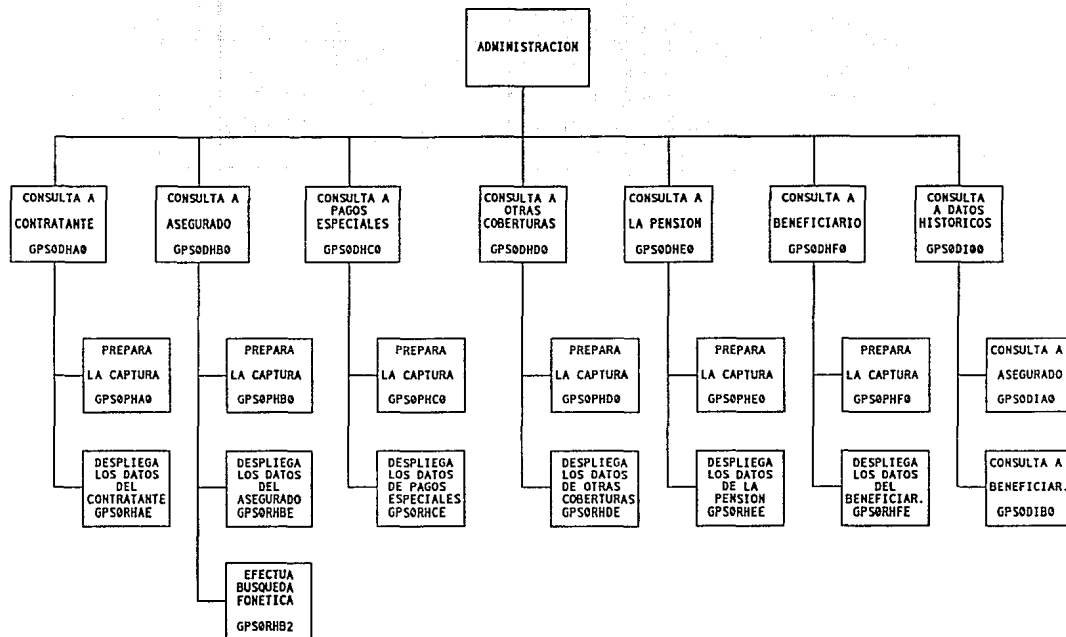
SISTEMA DE PENSIONES ASEGURADAS

DIAGRAMA DE FUNCIONES



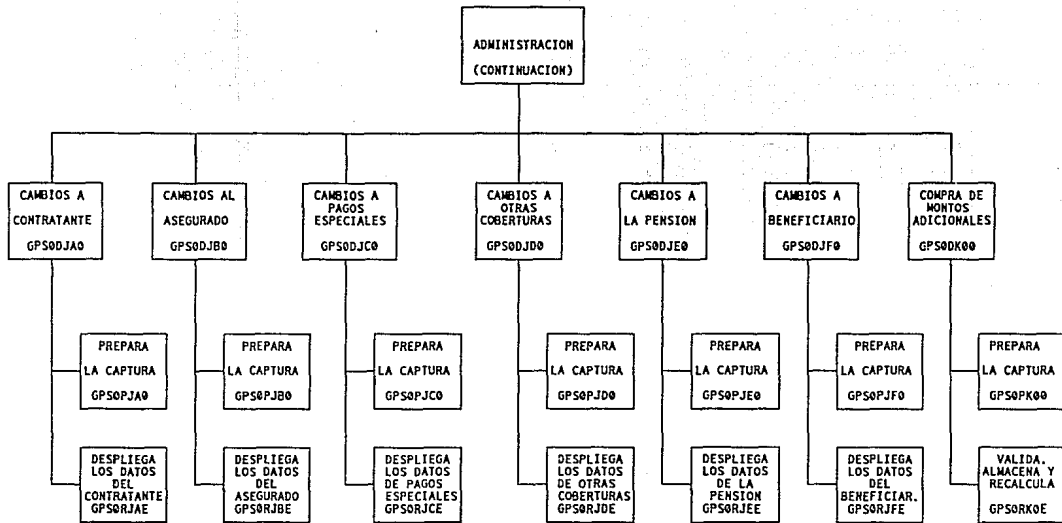
SISTEMA DE PENSIONES ASEGURADAS

DIAGRAMA DE FUNCIONES



SISTEMA DE PENSIONES ASEGURADAS

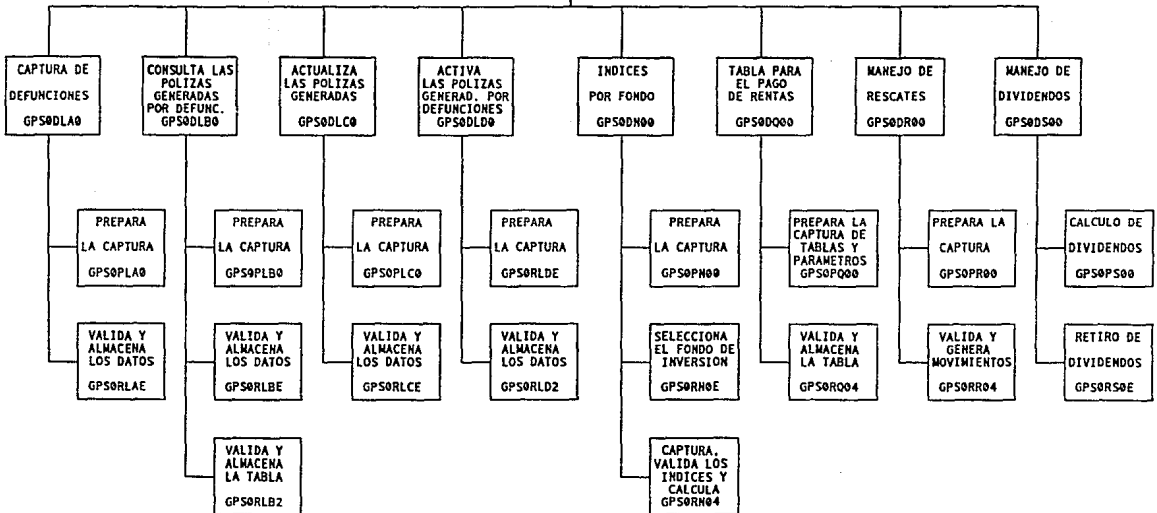
DIAGRAMA DE FUNCIONES



SISTEMA DE PENSIONES ASEGURADAS

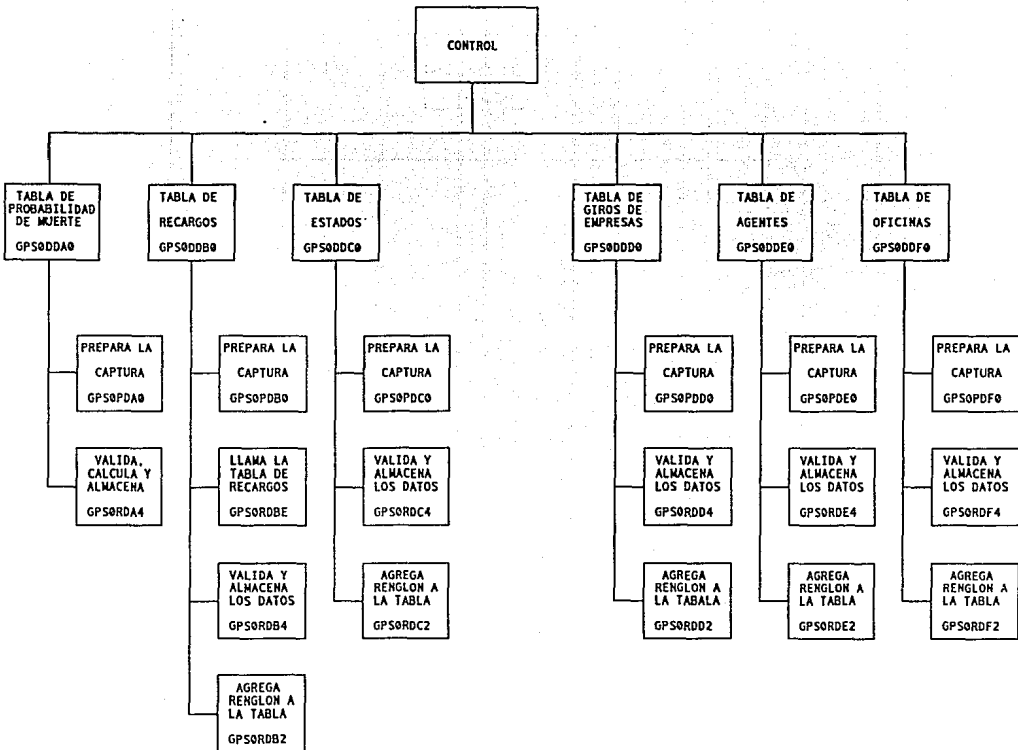
DIAGRAMA DE FUNCIONES

ADMINISTRACION
(CONTINUACION)



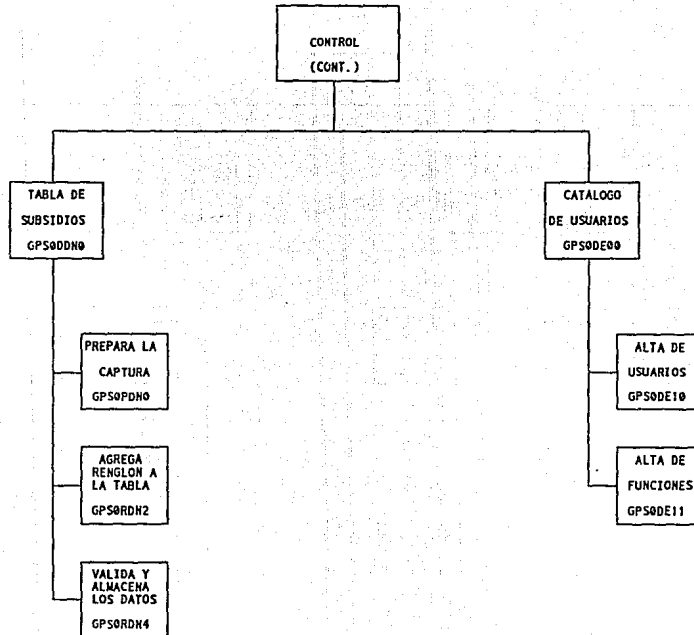
SISTEMA DE PENSIONES ASEGURADAS

DIAGRAMA DE FUNCIONES



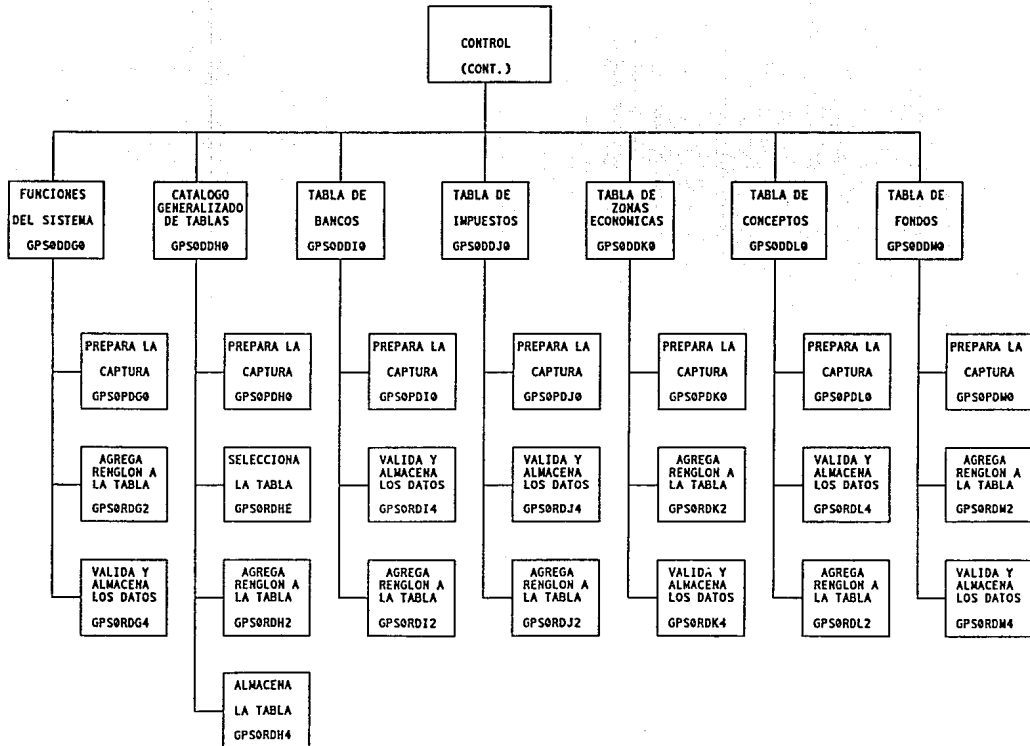
SISTEMA DE PENSIONES ASEGURADAS

DIAGRAMA DE FUNCIONES



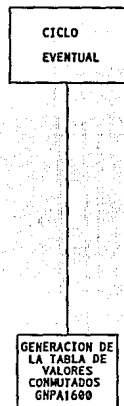
SISTEMA DE PENSIONES ASEGURADAS

DIAGRAMA DE FUNCIONES



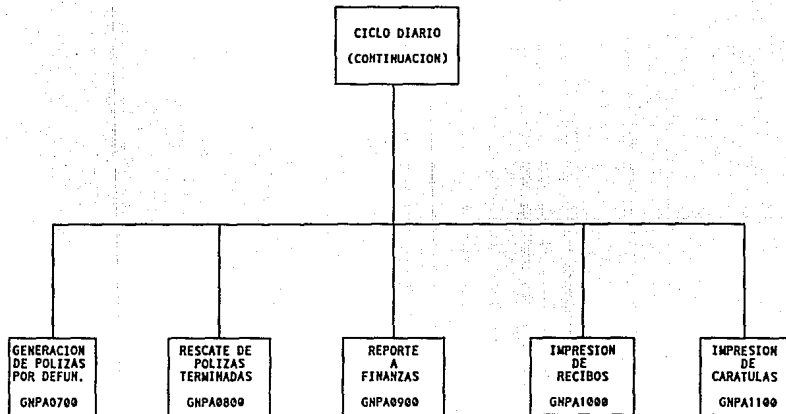
SISTEMA DE PENSIONES ASEGURADAS

DIAGRAMA DE FUNCIONES



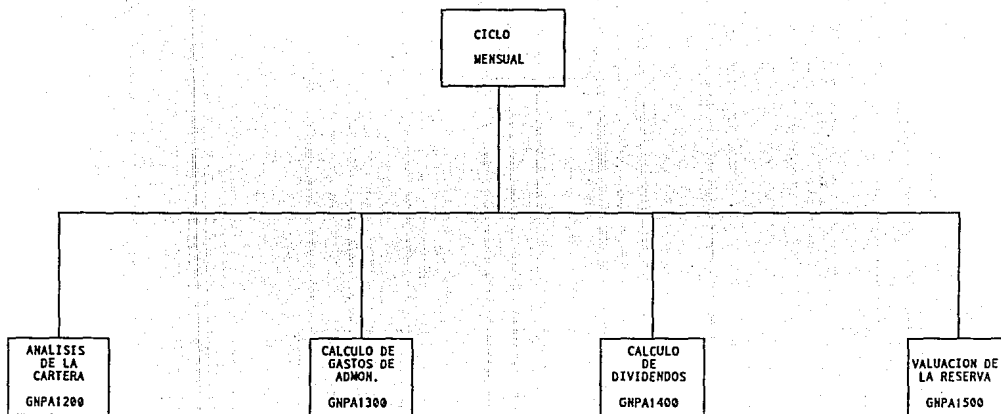
SISTEMA DE PENSIONES ASEGURADAS

DIAGRAMA DE FUNCIONES



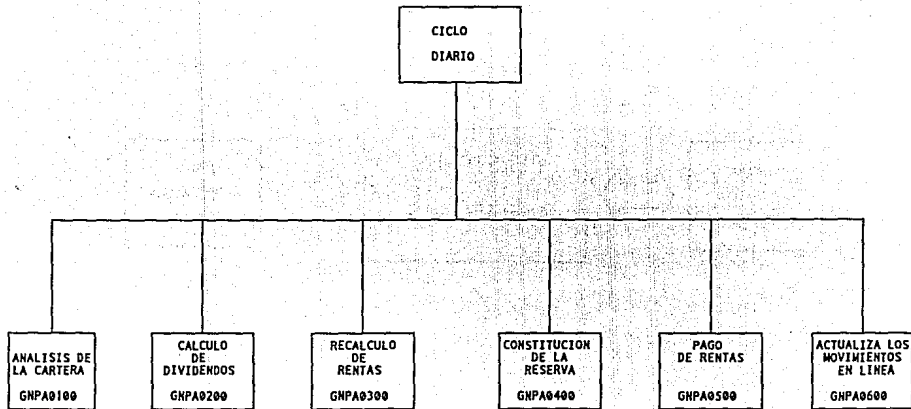
SISTEMA DE PENSIONES ASEGURADAS

DIAGRAMA DE FUNCIONES

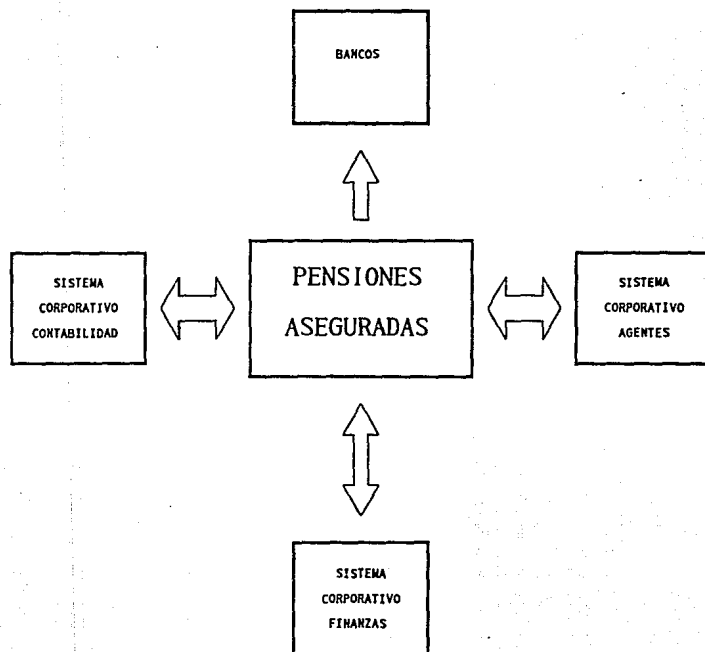


SISTEMA DE PENSIONES ASEGURADAS

DIAGRAMA DE FUNCIONES



SISTEMA DE PENSIONES ASEGURADAS INTERFASES



Los temas siempre presentes en la ingeniería de software son la calidad del software, la productividad del sistema y del equipo de desarrollo; así como la mayor satisfacción de los requerimientos del sistema, y en el trabajo de los ingenieros de software.

El objetivo fundamental de esta ingeniería es proporcionar métodos, herramientas y técnicas para facilitar que la gente competente lleve a cabo un trabajo organizado e inteligente. Los gurús y los magos del software huyen de las técnicas sistemáticas; por su parte dichas técnicas nunca convertirán la incompetencia en pericia.

Los problemas principales de esta ingeniería son el control de la complejidad; la falta de restricciones físicas y en los límites de la producción del software; así como la tendencia de mezclar aspectos que pertenecen a diferentes niveles de abstracción.

En todo desarrollo de sistemas, siempre será necesario primero analizar el problema, planear una estrategia de solución, diseñar una solución, para luego, implantar y validar la solución, preparar los documentos de apoyo, y modificar y actualizar el sistema conforme evolucionen el ambiente y las necesidades del usuario. Por lo tanto, el paradigma básico de solución de problemas en la ingeniería de software parece ser sólido.

Las Metodologías de Desarrollo de Sistemas son el sustento vital para todo desarrollador de sistemas, sobre todo en la plataforma macrocomputacional, no solo por la complejidad, sino, porque exige una integración lógica del conocimiento -no es posible realizar Análisis de procesos o Diseño de Estructuras de manera aislada o sin control-. Además en esta plataforma existe una gama de relaciones mutuamente dependientes y factores implicados (como es la interrelación departamental del área de sistemas, las políticas de control de proyectos, el flujo centralizado de la información, los procesos involucrados con la liberación a producción de un sistema, etc.) generadas por el uso productivo de un equipo Main-Frame (en una organización con políticas bien definidas).

De esta forma las metodologías de desarrollo son indispensables e insustituibles para avanzar en el ciclo de vida de todo sistema, (por mas pequeño que este sea; una fase de mantenimiento, por ejemplo) de forma más firme ya que se apoyan de conocimiento científico - modelos formales con gran capacidad de adaptación a las circunstancias reales -.

El uso de Metodologías no es rígido, (estas se pueden conjuntar, sustituir, y completar para adaptarse al problema y a su entorno, claro, como en todo conocimiento formal hay leyes inviolables. Al utilizar dichas metodologías se obtendrá un producto ("el sistema") con mayores probabilidades de lograr atributos que demuestran la mejor calidad (adaptabilidad, eficiencia, modificabilidad, etc.) de este, y entonces, aseguren a la organización con factores de aceptación final : elevación de índices productivos, reducción de costos, ampliación de la cartera de clientes y hasta reorganización de ciertas áreas para eficientar procesos.

Una Metodología de Desarrollo de Sistemas es un conjunto interrelacionado de métodos, técnicas y herramientas que tiene como objeto el estructurar y guiar , a todas las fases del ciclo de vida del desarrollo. Hemos propuesto un esquema general llamado 'Organización Estructurada del Trabajo', donde se puede ubicar fácilmente a las Metodologías, como la base para el desarrollo de productos de Software, ambientado bajo cualquier modelo o ciclo de vida.

En el desarrollo del sistema SPA, se utilizaron dos metodologías estructuradas (SA/SD e IE), alimentadas con conceptos de una metodología organizativa propia. Estas se aplicaron evitando concurrencias (sin aplicar técnicas al mismo tiempo en una misma fase), sino que se sustituyeron y adaptaron en ciertas fases, completándose no solo entre sí, sino entre nuestros métodos organizativos (bitacora, lista de pendientes, procesos documentativos, etc.). Fue gracias a dicha aplicación que se pudo reestructurar, corregir y reubicar el sistema, para cumplir con los objetivos esperados.

Para todo desarrollo de un sistema, es fundamental la fase de Análisis de Sistemas, manteniendo siempre presente el realizar satisfactoriamente el documento maestro, "Especificación Formal de Requerimientos" que comprende los criterios de aceptación; llave para la aprobación del sistema, por los usuarios.

La Ingeniería de Software es una nueva disciplina profesional, que rápidamente esta evolucionando y en donde se requiere una continua e intensiva capacitación, y sobretodo un enfoque profesional en la solución de problemas.

En la medida que se reconozca el nivel profesional y el formalismo requerido en la ingeniería de software, se podrá responder a la altura de la problemática de información actual, de lo contrario, la problemática continuará y se agravará.

El trabajo en equipo es la única manera en que grandes proyectos pueden ser realizados. La organización es la clave para trabajar eficientemente en equipo. El estructurar las funciones a realizar por cada uno de los integrantes resulta fundamental para lograr el producto final, "el sistema".

Mientras un individuo no desarrolle una metodología consistente de trabajo (organización propia del trabajo), nunca será en realidad un profesional. Un profesional se distingue porqué hace uso de un conjunto de conocimientos bien comprendidos y justificados.

La capacidad para trabajar con grandes sistemas de cómputo y para desarrollar sistemas que serán usados por gran cantidad de gente durante varios años, no se adquiere de la práctica o de la vida diaria, si bien estos elementos son esenciales, no son los puntos claves para adquirir tal capacidad. Tal capacidad se adquiere por medio de una formación profesional que comprenda tanto aspectos técnicos como humanísticos.

Hemos sido testigos de la degradación del papel profesional en la industria del software. Los sistemas son desarrollados por personal no capacitado y sin formación profesional. Lo anterior se traduce en los servicios de cómputo tan malos que en la actualidad se ofrecen.

Es responsabilidad del profesional del software el recuperar su papel en la industria, formalizar procedimientos, capacitar al personal y orientar a los usuarios de los sistemas en general. Todo ello redundará en una mejora gradual de los servicios de cómputo y por lo mismo, en la mejora de los servicios que ofrecen todas las empresas que se auxilian del computador como herramienta de trabajo productiva.

BIBLIOGRAFIA

ANDREW P. SAGE/ JAMES D.PALMER
"SOFTWARE SYSTEMS ENGINEERING"
ED. WILEY, 1990

CARLO GHEZZI/MEHDI JAZAYERI/DINO MANDRIOLI
"FUNDAMENTALS OF SOFTWARE ENGINEERING"
ED. PRENTICE HALL,1991

EASY CASE
"METHODOLOGY GUIDE"
ED. EVERGREEN CASE TOOLS INC, 1993

ERNEST & YOUNG
"FACILITING CHANGE : THE KEY TO SUCCESS WITH INFORMATION TECHNOLOGY"
CENTER FOR INFORMATION TECHNOLOGY AND STRATEGY, 1992

FAIRLEY
"INGENIERIA DE SOFTWARE"
ED. MACGRAW HILL, 1988

GREGORY W. JONES
"SOFTWARE ENGINEERING"
ED. JOHN WILEY&SONS, 1990

HARVEY M. DEITEL
"AN INTRODUCTION TO OPERATING SYSTEM"
ED. ADDISON WESLEY, 1990

JAMES MARTIN
INFORMATION ENGINEERIN(1)
ED. PRENTICE HALL, 1989

MERLE P. MARTIN

"ANALYSIS AND DESIGN OF BUSINESS INFORMATION SYSTEMS"

ED. MACMILLAN PUBLISHING, 1991

MICHAEL J. POWERS

"STRUCTURED SYSTEMS DEVELOPMENT"

ED. YOURDON PRESS, 1991

PRASAD, N.S

"IBM MAINFRAME, ARCHITECTURE AND DESIGN"

ED. J. RANODI, MACGRAW-HILL, 1989

ROGER FOURNIER

**"PRACTICAL GUIDE TO STRUCTURED SYSTEM DEVELOPMENT AND
MAINTENANCE"**

ED. YOURDON PRESS, 1991

STERN & STERN

STRUCTURED COBOL PROGRAMMING

ED. WILEY, 1991

WHITEN/BENTLEY/BARLOW

"SYSTEMS ANALYSIS & DESIGN METHOD"

ED. IRWIN , 1989

YOURDON/CONSTANTINE

"STRUCTURED DESIGN"

ED. PRENTICE HALL, 1979

FUNDACION MEXICANA PARA LA CALIDAD TOTAL, A.C.

"MONOGRAFIAS (4,5,6)", 1992

NOTA TECNICA DEL PRODUCTO PENSIONES ASEGURADAS

DIRECCION DE PENSIONES, 1992