

44 2ej

UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO



ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES

“ARAGON”

FALLA DE ORIGEN

“ANALISIS Y DESARROLLO DE SISTEMAS DE INFORMACION PARA CONTROL DE PROYECTOS”

T E S I S

Para obtener el Título de:

INGENIERO EN COMPUTACION

P r e s e n t a :

MARIO ALBERTO ORTIZ RODRIGUEZ

Asesor: Ing. Roberto Blanco Bautista



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.

Agradezco la colaboración para el desarrollo de esta obra a las personas que con su apoyo y su experiencia hicieron posible el término de esta obra y especialmente a:

Ing. Roberto Blanco Bautista

Ing. Húgo Portilla Vázquez

Lic. Israel Juárez Ortega

Ing. Juan José Martínez Coscalla

Ing. Casidoro Domínguez Crisanto

y a la Universidad Nacional Autónoma de México por haber tenido la oportunidad de estudiar en sus aulas de sus instalaciones.

Gracias por su apoyo

Índice

Introducción	5	
1	Control de Proyectos Bajo el Enfoque de Sistemas	6
1.1	Antecedentes	7
1.1.1	Panorama Histórico del Procesamiento de Datos	7
1.1.2	Necesidad de que se Produzca más Información	8
1.1.3	Introducción al Procesamiento de Datos e Información en Nuestros Días	9
1.1.4	El Enfoque de Sistemas y el Análisis de Sistemas	13
1.1.5	La Filosofía del Enfoque de Sistemas	13
1.1.6	Análisis de Sistemas y la Solución de Problemas	15
1.1.7	Análisis de Sistemas y el Desarrollo de Sistemas de Información	16
1.1.8	El Análisis de Sistemas de Información	18
1.2	Análisis de Sistemas de Información	20
1.2.1	Los Conceptos de Datos e Información	21
1.2.2	Producción de Información a partir de Datos	25
1.2.3	Aspectos Económicos de la Información	27
1.3.	Conceptos de Análisis de Sistemas, Administración y Sistema Formal de Administración	35
1.3.1	Exposición General de los Sistemas	35
1.3.2	Exposición General de la Administración	42
1.3.3	Revisión de los Sistemas de Información	49
1.4	Análisis de los Conceptos Básicos de la Información	53
1.4.1	Estudio de los Sistemas de Información	53
1.4.2	Enfoque Jerárquico para el Diseño de Sistemas de Información	57
1.4.3	Enfoques de Sistemas para el Diseño de Información	59
2	Análisis del Sistema Mediante Modelo E-R	75
2.1.1	Modelo de Datos	76
2.1.2	Modelos Lógicos Basados en Objetos	76
2.1.3	Modelos Lógicos Basados en Registros	77
2.1.4	Modelos Físicos de Datos	78
2.2	Instancias y Esquemas	78
2.2.1	Independencia de Datos	79
2.2.2	Lenguaje de Definición de Datos	79
2.3	Metodología de Diseño de Base de Datos	80
2.3.1	Definición de Base de Datos	80
2.3.2	Diseños Tradicionales	80
2.3.3	Enfoque Entidad-Relación	81
2.3.4	Aparición de Herramientas CASE	81
2.3.5	Elementos del Modelo de Datos	81
2.3.5.1	Entidades	81
2.3.5.2	Entidad Fundamental	82
2.3.5.3	Entidad Atributiva	82
2.3.5.4	Entidad Asociativa	83
2.3.5.5	Notaciones Gráficas de Entidades	83
2.3.5.6	Atributo	83
2.3.5.7	Dominio de Atributo	84
2.3.5.8	Relación	84
2.3.5.9	Relación Uno-a-Uno	84
2.3.5.10	Relación Uno-a-Muchos	85

2.3.5.11	Relación Muchos-a-Muchos	85
2.3.5.12	Notaciones Gráficas Según Herramientas CASE	86
2.3.5.13	Redundancia	86
2.3.5.14	Redundancia Necesaria	87
2.3.5.15	Redundancia Inecesaria	87
2.3.5.16	Normalización	87
2.3.5.17	Campos Llave	87
2.3.5.18	Llave Primaria	88
2.3.5.19	Llave Foránea	88
2.3.5.20	Llave de Acceso o Llave Secundaria	88
2.3.5.21	Nivel Externo o Vista Lógica	89
2.3.5.22	Vía de Acceso	89
2.3.5.23	Entidad Lógica	89
2.3.5.24	Seguridad	90
2.3.5.25	Integridad de la Base de Datos	90
2.4	Modelo de Datos en SNAP	90
2.4.1	Concepto de Sistema en SNAP	91
2.4.2	Sub-sistema	91
2.4.3	Enfoque Entidad-Relación en SNAP	92
2.4.4	Concepto de Entidad en SNAP	92
2.4.5	Concepto de Atributo en SNAP	92
2.4.6	Concepto de Relación en SNAP	93
2.4.7	Tipos de Entidades	98
2.5	Análisis de Sistema Mediante Modelo Entidad-Relación	99
2.5.1	Definición de Entidades del Sistema	100
2.5.2	Entidades Fundamentales	100
2.5.3	Entidades Atributivas	100
2.5.4	Entidades Asociativas	100
2.5.5	Entidades Lógicas	100
2.5.6	Definición de Relación de Entidades	101
2.5.7	Definición de Atributos de Entidades	101
2.5.8	Definición de Vías de Acceso sobre las Relaciones	102
2.6	Customización del Modelo de Datos	102
2.6.1	Customización de Entidades	102
2.6.2	Definición de Vías de Acceso Alternas	103
2.6.3	Customización de Datos	104
2.6.4	Documentación del Modelo de Datos	104
2.7	Identificación de Entidades Utilizadas en el Sistema COP	104
2.7.1	Entidades Fundamentales	104
2.7.2	Entidades Atributivas	105
2.7.3	Entidades Asociativas	106
2.8	Diagrama Conceptual del Sistema COP	107
3	La Herramienta CASE	108
3.1	La Automatización del Software	109
3.1.2	Beneficios que Proporciona los CASE	110
3.1.3	Las Herramientas CASE	110
3.1.4	Ejemplos de Herramientas CASE	111
3.1.5	La Historia de los CASE	111
3.1.6	Objetivos de los CASE	114
3.2	Entorno de Desarrollo en CASE	114
3.2.1	El Banco de Trabajo CASE	115
3.2.2	Las Capacidades Gráficas	116
3.2.3	Necesidad de los Diagramas	117
3.2.3.1	Ventajas de las Técnicas de Diagramación Estructurada	118

3.2.3.2	Usos de los Diagramas Estructurados	118
3.2.3.3	Visión del Sistema Múltiple	119
3.2.4	Comprobación de Errores	121
3.3	El Soporte CASE en los Procesos de Desarrollo de Sistemas de Información	121
3.3.1	El Soporte de las Metodologías y los Procesos de Desarrollo de Sistemas	121
3.3.2	El Énfasis en las Primeras Fases del Ciclo de Vida	122
3.3.3	Los Prototipos	122
3.3.4	La Simulación	123
3.3.5	La Generación de Código	123
3.3.6	El Soporte de la Metodologías Estructuradas	124
3.3.6.1	Clasificación de las Metodologías Estructuradas	125
3.3.6.2	La Ingeniería de Sistemas	125
3.4	El Desarrollo de Sistemas Orientado al Procedimiento Frente al Orientado a los Datos	127
3.4.1	La Ingeniería de la Información	128
3.5	Soporte de los Sistemas de Tiempo Real Frente a los Sistemas de Información	129
3.5.1	Los Sistemas de Tiempo Real	130
3.6	Las Metodologías Estructuradas de Utilización más Extendida	131
3.6.1	El Análisis Estructurado	132
3.6.2	La Especificación Estructurada	133
3.6.3	El Diagrama de Flujo de Datos	133
3.6.4	El Diccionario de Datos	133
3.6.5	Las Especificaciones de Proceso	134
3.6.6	Los Pasos del Análisis Estructurado	134
3.7	El Diseño Estructurado de Yourdon	136
3.7.1	El Diagrama de Estructura	136
3.7.2	Los Pasos del Diseño Estructurado de Yourdon	136
3.8	La Metodología de Diseño de Jackson	141
3.8.1	Diagramas de Estructuras y Pasos de Jackson	142
3.8.2	El Proceso de Diseño Básico	146
3.9	Metodología de Diseño de Martín	146
3.10	La Metodología DSSD	150
3.10.1	El Diagrama de Warnier-Orr	150
3.10.2	La Formulación Orientada hacia la Salida	151
3.10.3	Los Pasos Básicos del DSSD	152
4	Desarrollo del Sistema Mediante Lenguaje SNAP	155
4.1	Pasos que se Recomiendan para el Desarrollo de Sistemas en SNAP	156
4.1.1	Modelo de Datos	156
4.1.2	Método de Desarrollo Acelerado	156
4.1.3	Utilitarios	157
4.1.4	Seguridad	157
4.2	Desarrollo del Sistema	157
4.2.1	Definición de Entidades	158
4.2.2	Customización del Modelo de Datos	160
4.2.2.1	Customización de Entidades	160
4.2.2.2	Definición de Vías Alternas de Acceso	161
4.2.2.3	Definición de Entidades lógicas	161
4.2.2.4	Modificación de los Aspectos Físicos de las Entidades	162
4.2.3	Generación de la Base de Datos	163
4.3	Documentación del Modelo de Datos	163
4.4	Desarrollo Físico del Sistema	163
4.4.1	Sistema a Trabajar	164
4.4.2	Captura y Mantenimiento de Entidades	164

4.4.2.1	Definición de Entidades	168
4.4.3	Creación de Programas Automáticos	175
4.4.3.1	Programas Automáticos	176
4.4.3.2	Especificaciones de Alto Nivel	181
4.4.3.5	Normalización	184
4.4.4	Método de Desarrollo Acelerado	182
4.5	Ejemplo Utilizando Programas Automáticos mediante un Requerimiento	188
4.5.1	Definición de Programas Automáticos	188
4.5.2	Programa de Verificación de Eliminación de Clientes	191
4.5.3	Programa de Consulta de Personal Asignado a Clientes	192
4.5.4	Consulta Simple a Clientes	199
4.5.5	Mantenimiento Simple a Clientes	203
4.5.6	Mantenimiento Alta, Baja y Cambio a Clientes de Personal Asignado	207
4.5.7	Programa de Reporte	212
4.5.7	Programa de Sesión a Clientes	225
5	Implantación y Mantenimiento	234
5.1	Capacitación e Instrucción del Personal	235
5.1.1	Métodos de Instrucción y Capacitación del Personal	236
5.2	Realización de Pruebas del Sistema	237
5.2.1	Niveles de Prueba	237
5.2.2	Clases de Pruebas	238
5.2.3	La Importancia Creciente de Pruebas	238
5.3	Conversión de Sistemas	239
5.3.1	Clases de Conversión	239
5.3.2	Enfoques de las Conversiones	240
5.3.3	Consideraciones Relativas al Banco de Datos durante la Conversión	241
5.3.4	Planeación de la Conversión	243
5.4	Secuela de la Implantación	243
	Conclusiones	245
	Bibliografía	246

INTRODUCCIÓN

El Sistema Control de Proyectos de el cual se obtuvo el nombre de esta obra pertenece a AEI Sistemas (Empresa que presta servicios de análisis, desarrollo e implantación de sistemas de información en ambiente de AS-400), se desarrolló hace ocho años. Ya que no se tuvo personal propio para que atendiera e hiciera uso de él, el sistema no logró implantarse y quedó en el abandono no obstante que se logró terminar.

Este sistema de información fue creado debido a la necesidad que tenía la empresa antes citada, de dar seguimiento a los proyectos de desarrollo de sistemas de información que tenía con diferentes clientes, en los cuales empleaba personal que a su vez tenía diferentes actividades específicas, que iban desde instaladores de equipo, programadores, analistas, líderes de proyecto etc. Las áreas que abarcaba y controlaba el sistema era tanto a nivel de proyecto como a nivel individual o personal, esto quiero decir que tanto los proyectos como las actividades del personal asignado se le daban tiempos estimados de desarrollo y terminación, a esto se le daba seguimiento con un reporte de actividades que se tenía que entregar quincenalmente. Esto permitía tener una inmediata respuesta al cliente y a la misma empresa por la falla o retraso de trabajos.

El sistema en general maneja información general de clientes, personal de la empresa, planes o proyectos.

El objetivo de esta obra es tomar este sistema y tenerlo como ejemplo en el análisis, desarrollo, implantación y mantenimiento de sistemas de información, particularmente usando una herramienta CASE ya que esta herramienta lo último que se ha desarrollado para el desarrollo de todas las etapas de desarrollo de sistemas de información. Así como hacer algunas modificaciones a programas con el propósito de actualizarlos y que éste sistema entre en funcionamiento ya que algunos de estos quedaron sin actualización a los nuevos requerimientos de funcionamiento.

En este texto se describe la secuencia de actividades que se emplean para el desarrollo de sistemas de información. En el primer capítulo se describirán los conceptos básicos que se emplean en el área de desarrollo de sistemas de información como datos, información y principalmente el enfoque de sistemas, que es una estructuración global de actividades que se conceptualizan en un diagrama y que representa todo un sistema. En el segundo capítulo se describirán los diferentes modelos de representación de los sistemas de información y en particular el modelo Entidad-Relación el cual es actualmente el más empleado y que lo emplea un sistema que desarrolla sistemas de información como en este caso el SNAP.

El SNAP es una herramienta que proporciona un ambiente de trabajo integrado para el desarrollo y mantenimiento de sistemas en el computador AS/400 de IBM. El SNAP es un producto tipo CASE que emplea técnicas modernas de modelaje de datos, utiliza herramientas de generación de programas automáticos y un poderoso lenguaje de cuarta generación, este tipo de herramienta es el tema del tercer capítulo. En el capítulo cuarto se describirá como se emplea el SNAP en el desarrollo de sistemas de información y su metodología de aplicación.

Todo sistema de información terminado pasa por un proceso de instalación y mantenimiento en donde se desarrollan diferentes actividades tanto de usuarios como de programadores y analistas, estas actividades se mencionan en el quinto y último capítulo.

Capítulo I

CONTROL DE PROYECTOS BAJO EL ENFOQUE DE SISTEMAS

Introducción

En este capítulo se presenta una explicación detallada de los conceptos de sistemas de información, en donde se irá desde un panorama histórico por el cuál se tiene noticia de las primeras necesidades que tuvo el hombre para registrar datos, hasta sus necesidades actuales de tener información según la necesite, estas necesidades originaron el desarrollo de tecnologías de análisis y desarrollo de sistemas de procesamiento de datos entre las cuales se encuentra el enfoque de sistemas, tema de este capítulo.

El enfoque de sistemas es una filosofía o una manera de concebir una estructura, que coordina de manera eficaz y óptima las actividades y operaciones dentro de un sistema. En el transcurso del desarrollo de este capítulo se examinará esta técnica junto con la explicación de temas ligados a el desarrollo de el análisis de sistemas. Entre estos puntos se encuentran: descripción de datos e información; procesamiento de datos e información; relación costo/efectividad en función con el procesamiento de datos e información; así como temas ligados con la administración como la planeación, el control y la toma de decisiones; y conceptos generales que se aplican en al área de sistemas de información como: el análisis de sistemas, análisis de entrada y salida en un sistema formal, tipos de sistemas, análisis de optimización, análisis del proceso de comunicación, modelos formal de sistema de información, tipos de procesamiento de datos, tipos de enfoques de sistemas etc.

1.1. Antecedentes

Desde los inicios de la civilización, la humanidad ha necesitado información como una ayuda en la lucha por la supervivencia, así como en los intentos para administrar las organizaciones. La creciente complejidad de la sociedad, sobre todo en la forma en que se manifiesta en las organizaciones sociales, políticas y económicas, ha aumentado en gran medida la necesidad de tener información más conveniente y oportuna. Antes de analizar a fondo las necesidades de información en las organizaciones modernas y los enfoques actuales de procesamiento de datos e información, en esta parte de capítulo se presenta una exposición general de procesamiento de datos e información. Los objetivos son:

1. Mostrar un panorama histórico del procesamiento de datos e información.
2. Introducción a la teoría y tecnología modernas del procesamiento de datos.
3. Introducción a la filosofía del enfoque de sistemas y su aplicación en el análisis de sistemas.
4. Describir la función del analista de sistemas de información en las organizaciones modernas.

1.1.1 Panorama Histórico del Procesamiento de Datos e Información

Los historiadores han encontrado que el mantenimiento de registros y el procesamiento de datos se remontan hasta 3,500 años a. C., época en que los mercaderes de Babilonia conservaban registros en tablas de arcilla. En la actualidad, los comerciantes todavía conservan registros y procesan datos, pero en vez de tablas de arcilla emplean cintas y discos magnéticos y otras formas de dispositivos de registro. En esta sección se presentan brevemente: 1) los eventos significativos en la evolución del procesamiento de datos e información y 2) las principales exigencias para que se produzca aún más información.

Eventos Significativos

Desde hace pocos años los términos "procesamiento de datos" y "procesamiento por computadora" se emplean casi como sinónimos. Es cierto que en la mayoría de las organizaciones la computadora es esencial para satisfacer gran parte de las necesidades de procesamiento de datos. Sin embargo, desde un punto de vista histórico, la computadora sólo puede considerarse como la última "revolución" dentro de la evolución de la teoría y la tecnología del procesamiento de datos. La primera y más importante "revolución" fue el desarrollo del lenguaje y de la notación matemática.

Aún cuando los historiadores no pueden estar seguros de la época en que se originaron los lenguajes hablados, se han encontrado formas de escritura que se remontan a la cultura babilónica del año 3,500 a. C. aproximadamente, y que consisten en registros impresos en tablas de arcilla (cuneiformes), sin embargo, sólo a los fenicios de 2,000 años después se les puede dar crédito por haber inventado el alfabeto. También los primeros sistemas matemáticos se remontan hasta el año 3,500 a. C. En realidad todo desarrollo del procesamiento de datos e información se basa en estos dos avances.

Utilizando el lenguaje y las matemáticas, la humanidad ha enriquecido y difundido continuamente el conocimiento y la comprensión de sí misma y de su medio ambiente. Además la

cantidad de conocimientos e información disponibles ha llegado a ser tan extensa que ninguna persona es capaz de obtenerlos todos. Debido a lo anterior, existe la tendencia a recopilar información en campos especializados tales como la física, biología, economía, psicología, contabilidad, etc. Aún cuando es cierto que cada nuevo concepto o idea que se descubra en uno de estos campos especializados, determina directamente la selección de los datos que se deben procesar y la información que se produce, es evidente que un estudio significativo de estas contribuciones sobrepasa los propósitos de este texto. El campo especializado que abarca esta sección se limita a los avances que influyen en el procesamiento de datos e información en general. Por ejemplo, desde que comenzó a emplear tablas de arcilla, la humanidad se ha valido de diferentes medios para registrar sus pensamientos y conceptos. Algunas veces se conservaron registros en piedra, pieles de animales, papiro, madera, metales, papel, y en actualidad se emplean dispositivos magnetizados.

Para poder procesar datos numéricos de manera más eficiente, la tecnología se ha desarrollado continuamente. El ábaco, que tal vez sea el más antiguo dispositivo de cálculo, ha estado en uso desde el año 3,000 a. C. En el siglo XVII se produjeron tres avances básicos en el desarrollo de la tecnología del procesamiento de datos: John Napier construyó un conjunto de barras numeradas (huesos de Napier) que simplificaban las operaciones de la multiplicación y la división; Blas Pascal proyectó y construyó la primera máquina sumadora; y por último, Gottfried Leibniz construyó una calculadora que podía sumar, restar, multiplicar y dividir. En el siglo XIX se produjeron todavía otros avances en la tecnología del procesamiento de datos. Entre éstos se encuentra el telar con tarjetas perforadas de Joseph Jacquard, la "máquina diferencial" y la "máquina analítica" de Charles Babbage, y las máquinas de tarjetas perforadas de Herman Hollerith, que se emplearon para procesar los datos del censo de 1890 en los Estados Unidos.

Si bien cada uno de estos desarrollos tecnológicos es importante en el desarrollo del procesamiento de datos e información, es necesario regresar al siglo XV para localizar la segunda "revolución", esta se debió a la invención de la imprenta. La imprenta dio a la humanidad la posibilidad de registrar, almacenar, recobrar, informar y transmitir datos e información más que cualquier otro invento anterior en un lapso de casi 500 años después.

No es hasta el siglo XX cuando se encuentra la tercera "revolución" en el procesamiento de datos e información, asociada con una serie de avances que se pueden clasificar con el nombre de "medios de comunicación masiva". La influencia de las técnicas de los medios de comunicación masiva, tales como la radio y la televisión, es evidente para cualquier persona, aún cuando esa misma persona no las comprenda por completo.

En este siglo, y antes de que finalizara la tercera "revolución", el procesamiento de datos e información comenzó a experimentar una cuarta "revolución" al parecer la computadora digital, al igual que ocurre con la influencia de los medios de comunicación masiva, la influencia de la computadora en la sociedad es muy importante. Sin embargo, al emplear la filosofía de sistemas, en la mayoría de los casos se trata de utilizar la computadora y la tecnología relacionada con ella, junto con las técnicas de los medios de comunicación masiva, para producir información más conveniente en las organizaciones actuales.

1.1.2 Necesidad que se Produzca más Información

En la sección anterior se expuso el desarrollo del procesamiento de datos e información, principalmente desde el punto de vista de los avances tecnológicos. es posible obtener una mejor intuición y comprensión si se examinan en forma breve las exigencias que existen para que se produzca cada vez más información.

Antes del siglo XVIII había dos razones principales para que se procesaran datos. En primer lugar, los hombres tenían el deseo natural de llevar la cuenta de sus propiedades y riquezas. Como se mencionó antes, los mercaderes babilónicos mantenían registros desde el año 3,500 a. C. es obvio que a medida que aumentó el intercambio y el comercio, los hombres necesitaron cada vez más medios para estar al tanto de los detalles y de la situación de los negocios. En el siglo XV Luca Pacioli desarrolló el sistema de teneduría de libros de doble entrada. Este concepto hizo posible que los eventos económicos se registraran en términos monetarios empleando una serie de cuentas relacionadas con los ingresos y los egresos. Hasta la fecha este concepto es el fundamento de los sistemas modernos de cuentas financieras.

La segunda razón para procesar datos antes del siglo XVIII la constituían los requerimientos gubernamentales. A medida que las tribus se transformaron en naciones, las autoridades de estas naciones (Egipto, Israel, Grecia, etc.) recopilaban investigaciones administrativas para que se emplearan en la recaudación de impuestos y el reclutamiento de soldados. A mediados del siglo XVIII se crearon todavía más exigencias para que se procesaran los datos de manera formal. La revolución industrial había trasladado a las fábricas las tareas básicas de la producción en el hogar y el los pequeños talleres. El desarrollo de estas grandes organizaciones manufactureras originó el desarrollo de otras industrias de servicios, tales como la investigación de mercados y la transportación. De esta manera, el gran tamaño y la complejidad de tales organizaciones hicieron imposible que el individuo administrara de manera eficaz una organización sin algún procesamiento de datos que proporcionará información adicional. Además, con la aparición del sistema de fábricas grandes y las técnicas de producción en serie la necesidad de los de producción más modernos requirió de investigaciones mayores. Estas necesidades financieras forzaron la separación entre el inversionista y la administración. Por un lado, la administración necesitaba de más información para tomar decisiones internas, y por otro, los inversionistas necesitaban información referente a la organización y al desempeño de la administración.

A medida que aparecieron las nuevas políticas de negocios aumentaba la necesidad del procesamiento de datos. Por ejemplo, la autorización de créditos hizo necesaria la actualización de las cuentas de cobros, las de pagos y las estadísticas de crédito. Los conceptos de la contabilidad financiera, que siguieron perfeccionando y difundiendo a lo largo de los años, también requirieron un mayor procesamiento de datos. A fin de obtener una mayor eficiencia en la producción, los precursores de la "administración científica" reconocieron la necesidad de un mayor procesamiento de datos e información.

La regulación mediante diversos organismos gubernamentales no sólo de las corporaciones sino de la sociedad en general ha tenido una gran influencia en los sistemas de información y procesamiento de datos en muchas organizaciones, forzando a éstas a adoptar prácticas actualizadas de procesamiento de datos. En los Estados Unidos, las dependencias de regulación gubernamental incluyen a las diferentes comisiones estatales de regulación; la Comisión de Comercio Interestatal; la Comisión Federal de Energía; la Comisión de Obligaciones e intercambio; juntas de control del medio ambiente; los departamentos de salud, educación y bienestar social; los departamentos de desarrollo habitacional y urbano, etc. de una manera u otra, todos estos grupos tienen necesidad de una gran variedad de informes provenientes de muchas organizaciones e instituciones distribuidas por todo el país, en consecuencia, la organización que proporciona los informes no sólo debe tener al día un buen sistema de procesamiento de datos, sino que también los grupos reguladores deben contar con sistemas análogos para manejar grandes cantidades de datos. De hecho, el gobierno de los Estados Unidos es con amplio margen el mayor procesador de datos en ese país.

1.1.3 Introducción al Procesamiento de Datos e Información en Nuestros Días

Como se vio en el panorama histórico del procesamiento de datos e información, las necesidades de información en las organizaciones actuales son extensas y complejas. En esta sección

se analizará la función de los modelos lógicos y matemáticos, así como el de la computadora digital, para satisfacer esas necesidades.

Modelos lógicos y matemáticos

A medida que aumenta el tamaño, la complejidad y la especialización de las organizaciones, se vuelve cada vez más difícil para los responsables de las decisiones asignar a diferentes actividades los recursos disponibles de la manera más eficaz y óptima para la organización. La planeación y la implantación de modelos lógicos y matemáticos pueden ayudar a resolver los problemas que origina la complejidad de las organizaciones modernas actuales. Por consiguiente, además de elaborar informes y métodos para el mantenimiento de registros históricos y satisfacer necesidades operativas y gubernamentales, los sistemas modernos de información deben incluir modelos lógico-matemáticos para suministrar información alternativa, predictiva, de optimización y de control. Al desarrollo de esos modelos se le suele describir como enfoque de la ciencia administrativa.

Algunos de los trabajos realizados por autores como Frederick Taylor, H. L. Gantt y Henri Fayol sentaron las bases del enfoque de la ciencia administrativa al problema de la toma de decisiones. Sin embargo, el campo de la ciencia administrativa recibió el mayor impulso durante la segunda guerra mundial y después de ella. Había una gran necesidad de asignar recursos en la segunda guerra mundial de la manera más eficaz, y para ayudar a determinar las operaciones logísticas más eficaces. Muchas de estas tentativas tuvieron éxito y no paso mucho tiempo antes de que la mayoría de tales técnicas se convirtiera en auxiliar de la administración industrial después de la guerra. De la misma manera que los estrategas militares, los gerentes de las organizaciones deben tomar decisiones que se relacionan con la vida futura de la organización, a corto y a largo plazo. Los gerentes tienen que tomar una gran cantidad de decisiones que van de lo rutinario a lo altamente complejo. Entre estas últimas se encuentran problemas como 1) promoción de ventas, 2) ubicación de plantas, 3) mezcla óptima de productos, 4) decisión de producir o comprar, 5) distribución de capital circulante, 6) programación de la producción, 7) políticas presupuestarias, 8) elaboración de pronósticos, 9) políticas de adquisición, 10) planeación financiera, 11) selección de canales de distribución, 12) selección y control de personal, 13) control de inventario, 14) control de calidad, 15) programación y control, y otros.

El objetivo central de la ciencia de la administración consiste en suministrar información para la ACCIÓN; en otras palabras, información con la cual la gerencia puede actuar. Como sugirió Wagner, las características de la ciencia administrativa son:

1. Un interés primordial en la toma de decisiones. Los principales resultados del análisis deben tener implicaciones directas y claras para la acción ejecutiva.
2. Una valoración que se basa en los criterios de eficiencia económica. Una comparación de las distintas acciones posibles se debe basar en los valores medibles que reflejen de manera inequívoca el bienestar futuro de la organización. Por ejemplo en la producción de una película comercial, estas cantidades medibles son: ingresos, flujo de efectivo y la razón de recuperación sobre la inversión creciente. Una solución recomendada de evaluar las fluctuaciones del mercado y lograr un balance óptimo entre estos factores que suelen ser conflictivos.
3. Cimentación sobre un modelo matemático formal. Los procedimientos para el manejo de datos deben ser tan explícitos que sea posible describirlos otro analista, el que a su vez derivaría los mismos resultados.
4. Dependencia de una computadora electrónica. En realidad, esta característica no es esencial sino más bien un requerimiento que demanda la complejidad del modelo matemático a la cantidad

de datos que se manejarán, o bien la magnitud de los cálculos necesarios para implantar los sistemas asociados de operación y control administrativos.

La esencia o ingrediente fundamental de la ciencia administrativa es la formulación y el manejo de modelos. Este enfoque se relaciona con los modelos lógico-matemáticos que se emplean para estudiar los aspectos cuantitativos de los problemas. El empleo de dichos modelos se basa en una actitud científica frente a los fenómenos de la administración. Esta actitud se le suele denominar método científico, que se basa en una secuencia lógica de etapas que se aplican en cualquier investigación científica. Estas etapas son:

1. Observación. El analista efectúa un examen general de la situación para definir el problema. En esta etapa se identifican todas las facetas y relaciones del problema.

2. Formulación del modelo. La elaboración de un modelo es el resultado directo de la etapa de observación. En consecuencia, expresa las interrelaciones contenidas en el problema. En el área de sistemas de información el modelo puede ser una gráfica, un diagrama de flujo, un diagrama de organización, un informe estadístico y, como ocurre a menudo, puede ser un modelo definido en términos simbólicos y matemáticos.

3. Experimentación con el modelo. Generalmente, el modelo se prueba varias veces antes de aplicarlo a la toma de decisiones. Una función primordial del modelo es su capacidad predictiva, por lo que se recopilan y manejan datos mediante el modelo para probar su eficacia. Se realizan modificaciones y mejoras, basándose en los resultados experimentales, hasta que el modelo está listo para la aplicación.

4. Aplicación. El modelo se aplica para proporcionarte a la administración la información necesaria para la toma de decisiones.

Es obvio que la aplicación de modelos lógico-matemáticos nunca puede constituir la base de todas las decisiones. En las organizaciones muchas de las necesidades de la información se pueden satisfacer estableciendo un banco de datos bien planeado y empleándolo para suministrar informes oportunos o respuestas en línea al responsable de la toma de decisiones, mediante terminales lejanas. Las decisiones significativas, que en la mayoría de los casos se basan en informes un poco más elaborados, se pueden tomar de manera eficiente si la información contenida en los informes es adecuada y oportuna.

Aún cuando el empleo de los modelos lógico-matemáticos es una componente vital para el suministro de información a la administración, por sí mismo no puede satisfacer todo el conjunto de las necesidades de información. Por lo general, el contexto de las actividades del sistema de información y procesamiento de datos abarcará y utilizará las técnicas de formación de modelos.

Computadora digital

No obstante que la computadora influye de una u otra manera, en la vida cotidiana, en la actualidad sigue siendo una de las herramientas del genio creativo de la humanidad menos comprendidas. Si a un número de personas que pasan por una calle cualquiera se les pidiera su opinión acerca de las computadoras, sería muy probable que las respuestas se parecieran a la descripción que de un elefante hicieron cinco ciegos, después de que cada uno de ellos se le dio la oportunidad de tocar una parte del cuerpo del animal. Esta situación persiste aún cuando la computadora se ha empleado

durante más de treinta años como una herramienta de procesamiento de datos en los negocios y en el gobierno.

En términos conceptuales, una computadora se compone de una unidad procesadora que efectúa algunas operaciones lógicas y aritméticas, un mecanismo para colocar datos dentro de la unidad procesadora ("entrada"), un mecanismo para almacenar los datos durante la "salida" la información que se procesó. Las posibilidades de la computadora se limitan a la ejecución de cálculos aritméticos y a la toma de decisiones, que en la actualidad se componen únicamente de dos alternativas.

La combinación de las capacidades básicas de la computadora, ejecución de cálculos aritméticos y su posibilidad de tomar decisiones permite realizar actividades más complejas, como el arreglo, sumarización, cálculos, recuperación y almacenamiento de datos que se van presentando. Los medios de entrada a una computadora y los de salida de una computadora son casi ilimitados y dependen de la computadora en cuestión. Los dispositivos usuales de entrada son cintas y discos magnéticos. Mientras que la velocidad interna de la computadora se mide en fracciones de segundo, las velocidades de entrada/salida son significativamente menores, y se relacionan en forma directa con el medio de entrada/salida que se emplee.

Una de las ideas centrales de este tema consiste en poner de manifiesto el tamaño y la complejidad crecientes de las instituciones y organizaciones actuales y el cambio tan rápido que están experimentando. La computadora, con los programas internos apropiados, constituye de manera especial una herramienta adecuada al nivel de estas complejidades y cambios. Un comentario convencional consiste en afirmar que se han desarrollado dos tipos de herramientas: el primer tipo se compone de las herramientas físicas que aumentan la potencia muscular y manual, tales como pinzas, martillos, motores, ruedas, palancas, etc. y el segundo tipo se compone de las herramientas intelectuales que amplían el poder de la inteligencia. Es obvio que la computadora se encuentra en la segunda categoría. Aunque muy poderosa, solamente es una herramienta que nunca controlará a la humanidad sino que ayudará a mantenerse al mismo nivel de las organizaciones complejas y de un medio ambiente que cambia con rapidez.

Para marchar al ritmo de los cambios rápidos, la eficiencia del responsable de las decisiones depende de un flujo heterogéneo información que sea a la vez adecuado y oportuno. Este tipo de información permitirá a los responsables de las decisiones operar en las condiciones que imponen los cambios rápidos. El suministro de este tipo de información requerirá del diseño y la implantación de sistemas de respuesta rápida. Algunos de los sistemas de respuesta rápida típicos son los sistemas de reservaciones en líneas aéreas, sistemas bancarios, sistemas de control de inventario, servicios de cotización de acciones y otros. La figura 1.0.1 es un modelo conceptual de una computadora.

La combinación de las computadoras con los medios de comunicación (Por Ejemplo, el teléfono y las microondas) ha suministrado herramientas y técnicas muy poderosas para implantar sistemas de respuesta rápida, y de esta manera le ha dado una nueva dimensión al concepto de sistemas de información. A medida en que los archivos de datos estén a alcance de la red de telecomunicaciones, el responsable de las decisiones puede recuperar la información contenida en esos archivos e interactuar con el sistema. Debido a esta interacción con el sistemas de información basada en respuestas rápidas, el espacio, el tiempo se reducen para los responsables de las decisiones, lo que les permite administrar en medio ambiente de manera más oportuna.

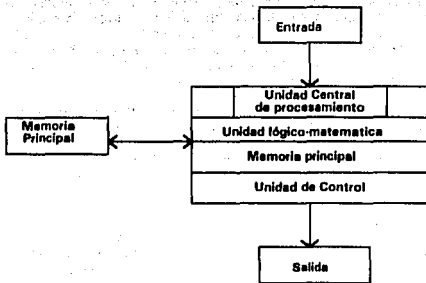


Figura 1.0.1 Modelo conceptual de una computadora digital.

1.1.4 El Enfoque de Sistemas y el Análisis de Sistemas

El enfoque de sistemas es una filosofía que se emplea ampliamente en la actualidad para dirigir la estructuración global de las actividades de procesamiento de datos, necesarias para satisfacer las necesidades de información en organizaciones modernas. En esta sección se dará una introducción a la filosofía de sistemas o enfoque de sistemas y, el análisis de sistemas. En la actualidad el análisis de sistemas se utiliza mucho como técnica general para resolver problemas, los mismo que como el método para desarrollar sistemas de información.

1.1.5 La Filosofía de Enfoque de Sistemas

El enfoque de sistemas es una filosofía, o una manera, de concebir una estructura, que coordina de manera eficaz y óptima las actividades y operaciones dentro de una organización o sistema. Un sistema se puede definir como un conjunto articulado de componentes o subsistemas ideados para alcanzar un objetivo. (El concepto de sistema se analiza detalladamente más adelante). La definición de sistema es una diferenciación lógica que se puede relacionar o no con las diferenciaciones reales que se encuentra en el mundo físico. El enfoque de sistemas se interesa por la componente individual y hace énfasis en la función que cumple dentro del sistema, más que en la función que cumple como entidad individual. El empleo del enfoque de sistemas para describir la realidad puede reportarle grandes ventajas al usuario. La eficacia de las componentes, consideradas colectivamente como un sistema,

puede ser mayor que la suma de los rendimientos de cada componente considerados por separado. Este efecto sinérgico se suele describir a firmando que " El todo es mayor que la suma de sus partes ". Una de las tendencias de las organizaciones modernas actuales consiste en la creciente especialización. Las componentes de una organización tienden a evolucionar en grupos relativamente autónomos. Cada grupo tiene sus propios objetivos y sistemas de valores, por lo que se puede perder de vista la manera en que se interrelacionan sus actividades y objetivos con los de la organización en general. La implantación de enfoque de sistemas rompe de manera radical las líneas funcionales tradicionales de la organización para lograr una optimización de la organización completa.

Por lo general, la mayoría de las organizaciones, existe un conflicto interno entre las distintas funciones. Idealmente, los problemas de cada área funcional se debe resolver de acuerdo con los objetivos de la organización global. Los requerimientos de la optimización total incluyen: 1) consideración de todas las alternativas, 2) consideración de todos los eventos, y 3) maximización de las funciones objetivas de toda organización. Para mostrar este conflicto interno entre las áreas funcionales se tomará como ejemplo el problema del control de inventario.

Por lo general el departamento de ventas desea contar con un inventario extenso de diversos productos para lograr un servicio óptimo a los clientes. Este objetivo del departamento de ventas es impracticable y, en consecuencia, antagonista con el objetivo global de la organización. Por otra parte, el departamento de producción tal vez desee producir en lotes de gran tamaño para reducir los costos de producción, pero esta reducción en los costos de producción puede originar un aumento en los inventarios de materia en proceso y requerir de mayor capital activo. El objetivo del funcionario ejecutivo de finanzas consiste en reducir al mínimo la inversión en el inventario. El administrador del almacén tiene a su vez un punto de vista diferente; su objetivo radica en establecer métodos de rutina para el manejo, recepción y envío del inventario y, de esta manera, reducir los costos de manejo. Por consiguiente, se puede observar que existen funciones distintas y conflictivas en una organización que es necesario coordinar para lograr una optimización total.

De las grandes y complejas organizaciones actuales la función de la administración consiste en coordinar e interrelacionar las actividades de las diferentes áreas funcionales y optimizar los objetivos de toda organización. En consecuencia, la filosofía del enfoque de sistemas es la manera en que la administración concibe la interrelación entre los subsistemas de la organización. El administrador eficiente debe formar un sistema integrado, partiendo de estos subsistemas individuales, a menudo conflictivos, en tal manera de todos los subsistemas trabajen juntos para alcanzar los objetivos de la organización. Como se mencionó antes, con el tamaño, complejidad y especialización crecientes, así como el rápido cambio de la mayoría de las organizaciones, este problema de reunir los subsistemas de la organización para formar un todo viable se volverá a un más difícil en el futuro.

Aunque se estudiará la manera en que el enfoque de sistemas se emplea para resolver problemas, así como para desarrollar sistemas de información, existen algunas directrices que pueden aplicarse para utilizar este concepto en general:

1. Integración. Los diferentes subsistemas del sistema se deben integrar de tal manera que se aprovechen las interrelaciones y la interdependencia entre los elementos.
2. Comunicación. Los canales de comunicación entre los subsistemas deben estar abiertos todo el tiempo.
3. Método científico. El método científico se debe aplicar utilizando las diferentes técnicas de la ciencia administrativa.
4. Orientación hacia las decisiones. A fin de hacer más eficaz la función administrativa de planeación y control, la toma de decisiones programada se desarrolla cuando es conveniente. El

objetivo radica en programar sistemas de decisiones bien definidos, tales como elaboración de horarios y sistemas logísticos, hasta el punto en que sean autorregulados. Este enfoque descargará a la administración de muchas actividades rutinarias y diversas que de otra manera estaría obligada a realizar. El tiempo que quede disponible debido a este enfoque permitirá a la administración concentrar sus esfuerzos en la toma de decisiones que todavía no está estructurada ni programada.

5. Tecnología. Siempre que sea posible, el analista deberá utilizar la tecnología moderna como auxiliar en la implantación de las técnicas que se originan a partir de las cuatro directrices anteriores; por ejemplo, la computadora mejora de manera significativa la integración, la comunicación, el método científico y la toma de decisiones programada.

1.1.6 Análisis de Sistemas y la Resolución de Problemas

El análisis de sistemas utiliza métodos cuantitativos, cuando son aplicables, además de factores cualitativos, como el juicio, la creatividad, la heurística, el sentido común y la experiencia. Cuando y dónde se debe empezar un análisis de sistemas es hasta cierto punto arbitrario. Se pueden requerir un gran número de sesiones de discusión y varios estudios y experiencias empíricos antes de efectuar el análisis de sistemas.

Dentro del contexto del enfoque de sistemas a la resolución de problemas, hay etapas básicas para la aplicación del análisis de sistemas:

1. Definición y Formulación del problema;
2. Desarrollo de soluciones alternativas;
3. Construcción de modelos que formalicen las alternativas;
4. Determinación del costo/eficiencia de las alternativas;
5. Presentación de las recomendaciones;
6. Implantación de la alternativa seleccionada.

Uno de los objetivos principales de la realización de un análisis de sistemas consiste en maximizar la eficiencia de la solución con un costo mínimo. Para lograr esta razón óptima entre costo y eficiencia es necesario obtener alternativas de cursos de acción y efectuar comparaciones. Cualquier curso de acción requiere de una asignación de recursos y produce una salida con un nivel determinado de eficiencia. Una organización puede optar por uno de los criterios excluyendo el otro, por ejemplo:

1. Criterio de la eficiencia. Para obtener un nivel específico de eficiencia en el logro de algún objetivo el analista ensaya las alternativas que alcanzarán ese nivel.
2. Criterio de costo. Para un nivel específico de recursos, el analista intenta determinar las alternativas que producirán el mayor nivel posible de eficiencia.

De manera ideal, debe haber un balance óptimo entre los dos criterios. Hay un punto en que una mayor eficiencia es insignificante cuando se comparan los costos. Por ejemplo, si se pueden proporcionar dos informes alternativos (A y B) para una función de toma de decisiones, y el informe A se puede entregar dos veces al día mientras que el informe B se puede entregar una vez al día, ¿es costoso la eficiencia adicional que representa recibir el informe A dos veces al día? Supóngase que el costo para producir el informe A es de \$100.00 por informe y que el costo para el informe B es de \$130.00. El informe A costará, por año, \$25,550 más que el informe B (suponiendo 365 días). En la toma de decisiones, la eficiencia adicional, que representa el ciclo de información del informe A, que es más rápido se debería comparar con la eficiencia adicional que se obtendría en las decisiones si los \$25,550 se asignaran a otros objetivos. Por consiguiente, en este caso se representa un fenómeno de costos de oportunidad y reasignación, implicando en la minimización óptima del costo, mientras que se maximiza la eficiencia. Lo que lograrían estos \$25,550, si no se canalizaran hacia otras alternativas, representa las oportunidades anteriores (o costo de oportunidad) de seleccionar un curso de acción determinado, en este caso sería el informe A.

1.1.7 Análisis de Sistemas y el Desarrollo de un Sistema de Información.

Por lo común la aplicación del análisis de sistemas al desarrollo de un sistema de información abarca un periodo más largo que el que se requiere para la resolución de problemas en general. Es obvio que esto no ocurre necesariamente en todos los casos. Además, en cierto sentido el desarrollo de un sistema de información se puede denominar como la solución a un problema: la necesidad de saber. Sin embargo, para describir mejor las actividades que se llevan a cabo durante el desarrollo de un sistema es posible identificar tres etapas:

1. Análisis de sistemas
2. Diseño de sistemas
3. Implantación de sistemas

Las actividades que se realizan en estas tres etapas son muy semejantes a las seis etapas que se describieron antes: Por ejemplo, el análisis de sistemas es análogo a la etapa 1; la definición y Formulación del problema. Para ser más específicos, abarca la definición y descripción de las metas, objetivos y requerimientos del sistema. La etapa de sistemas incluye actividades análogas a las etapas 2 a 5. Estas actividades se pueden re definir de la siguiente manera:

2. Desarrollo de diseños de alternativas;
3. Construcción de modelos que formalicen los diseños alternativos;
4. Presentación de recomendaciones.

Las actividades de que se compone la etapa denominada implantación de sistemas se podrán igualar a la etapa 6; implantación de la alternativa seleccionada.

Estas tres etapas del desarrollo de sistemas se representan en la figura 1.0.3 como constituyentes del ciclo vital de los sistemas. En esta figura también se muestran las principales actividades de los analistas de sistemas relacionadas con cada etapa.

El concepto de ciclo vital es una excelente manera de mostrar la naturaleza viable y dinámica del sistema de información en sí. Además, el concepto de ciclo vital constituye una exposición general de la metodología del desarrollo de sistemas.

Como sucede con toda entidad dinámica, la utilidad de la salida del sistema de información, la eficiencia de sus operaciones y la confiabilidad en la operación global del sistema pueden variar de manera considerable con el tiempo, por consiguiente, el sistema de información está expuesto a la deterioración, obsolescencia y, por último, a la sustitución. No obstante, rara vez acontece que en un momento dado, se le hagan mejoras al sistema de información por completo, en consecuencia, es evidente que aparte del sistema de información, o un subsistema de la organización, necesitará continuamente reparación o reconstrucción.

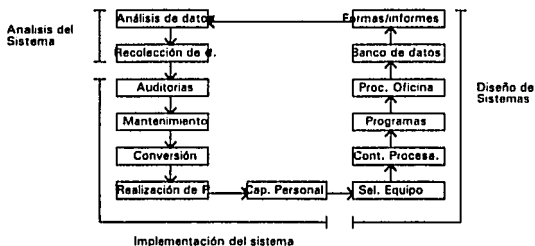


Figura 1.0.3 Ilustración del ciclo vital de un sistema de información y las principales actividades asociadas con cada etapa del ciclo.

A primera vista se podría afirmar que la duración del ciclo equivale a la razón con que el ciclo vital del sistema se repite una y otra vez. Algunas autoridades en la materia consideran la duración del ciclo en términos de años. Muchos expertos en sistemas de información calculan que un ciclo de vida normal para un sistema es de dos a diez años antes de que sea necesario un rediseño importante o una reparación total. Este cálculo acerca del ciclo vital de un sistema de información está sujeto a muchas condiciones y es necesario realizarlo por separado para cada sistema o subsistema de información. En nuestro caso, el sistema requiere que se re arranque, ya que se considera que esta bien diseñado por lo que sólo en el capítulo de desarrollo del sistema, se mencionarán como corregir o definitivamente hacer un nuevo programa que anteriormente su diseño no cumple las necesidades actuales por lo que se corrigió o cambió por completo por otro.

Algunos autores piensan que en realidad la duración del ciclo de vida de un sistema es confusa, en la medida en que hace referencia a la periodicidad o una relación de arranque-alto. En la mayoría de los casos la aplicación de las etapas en el ciclo vital es una operación continua altamente iterativa e interactiva, que por lo común se realiza a nivel de subsistemas y que en general prosigue hasta que el sistema total se modifica y, por último, se sustituye. Es rara la ocasión en que el analista de sistemas construye un sistema total partiendo de esquemas, pero si en realidad llega a suceder.

1.1.8 El Analista de Sistemas de Información

El catalizador y persona clave en el análisis, diseño e implantación de un sistema de información es el analista de sistemas. Esta posición funcional es la que debe ayudar a los usuarios de la información a definir qué tipo de información se requiere; cómo se puede transformar los datos disponibles en información; qué tecnología se puede utilizar, y cuál es la combinación óptima de dinero, personal, máquinas, materiales y métodos para implantar el sistema de información; se analizarán sus actividades dentro del contexto del ciclo vital de los sistemas.

Definición del Analista de Información

Básicamente, el analista de sistemas es una persona que sirve de interface entre los usuarios del sistema de información y los técnicos que trabajan en el sistema, tales como programadores, administradores del banco de datos y operadores de las máquinas. Esta función es lógica puesto que los sistemas de información se desarrollan para satisfacer las necesidades de una gran variedad de usuarios mediante organizaciones que (en la mayoría de los casos) disponen de una computadora como componente principal pero no única del sistema. Las funciones de programación y operación de computadoras, que por lo general se encuentran entre las operaciones del sistema de información, están estrechamente relacionadas con los aspectos técnicos de las actividades de implantación y mantenimiento.

La operación de sistemas tiene una naturaleza lógica y física. Los aspectos lógicos se refieren a la especificación del contenido y estructura del sistema de información. El aspecto físico se relaciona con las máquinas y programas (hardware y software) específicos que se van a emplear. En consecuencia, para que el analista de sistema sea eficiente debe: 1) ser creativo a la vez pragmático, 2) ser un catalizador y hasta un árbitro, y 3) pensar en la organización, así como en la tecnología.

La Función del Analista en el Ciclo Vital de los Sistemas

Durante la etapa de análisis de sistemas, el analista se ocupa de ayudar a los integrantes de la organización, la administración y demás personas encargadas de las operaciones, para identificar qué tipo de información se requiere, en qué manera se requiere. Durante esta fase, los problemas son numerosos, pero el analista debe prestar especial atención a la cuantificación de lo que a menudo se suele expresar cualitativamente o en forma vaga. Tal vez la contribución más importante que el analista puede hacer durante esta etapa consiste en auxiliar al usuario en la identificación de la información que en realidad es necesaria. Por lo general, ésta difiere de lo que los usuarios "piensan" que es necesario, puesto que sus conceptos se basan en algún tipo de noción preconcebida de lo que se puede suministrar.

Durante la etapa de diseño de sistemas, el analista tiene la oportunidad de ser creativo y original para determinar la manera en que la información se puede procesar. El aspecto principal de esta etapa radica en convertir en un plan detallado los requerimientos de información que se especifican en la etapa de investigación. Esta fase puede requerir de interacción con algunos técnicos del sistema de información. El analista debe tener en cuenta los conocimientos que proporcionen estas personas acerca de la tecnología disponible, las técnicas aplicables y la mejor manera en que se pueden utilizar los recursos.

Entre las actividades que demandan la atención del analista durante la etapa de diseño se incluyen: 1) formas de entrada y salida, 2) la estructura física y lógica del banco de datos, 3) rutinas de escritorio, 4) programas para computadora y 5) controles del procesamiento. La importancia crucial de la función del analista durante esta etapa se puso de manifiesto mediante una investigación industrial reciente, la cual reveló que menos de la mitad de los nuevos sistemas de información en proceso de diseño y desarrollo se llegan a desarrollar y a instalar completamente.

Si los conocimientos que posee un analista acerca de las técnicas y herramientas del procesamiento son obsoletos o limitados, entonces cualquier sistema que el o la analista diseñe reflejará estas fallas. Por otro lado, el analista debe evitar una planeación excesiva. En la actualidad, en muchas organizaciones gran parte de la aversión que existe por el desarrollo global de sistemas de información se puede atribuir a analistas que diseñaron (o trataron de diseñar) "supercarreteras" cuando todo lo que la organización requería, o podía financiar, era una "vereda".

Todos los esfuerzos del análisis y diseño llegan a cristalizar la etapa del ciclo vital de implantación de sistemas. El analista debe saber las fechas de arranque, capacitar y coordinar al personal que empleará el sistema, instruir al personal de computación, instalar nuevos procedimientos y formas, realizar pruebas con el sistema y mantenerse alerta para percibir descuidos y omisiones en las actividades de las etapas anteriores. Por desdicha, más de un buen sistema de información se ha descartado debido a la mala realización de la conversión y de la implantación.

El mantenimiento del sistema, aún cuando parezca una tarea poco atractiva, es muy necesaria y continúa a lo largo de toda la vida de operación del sistema. Al analista se le pedirá constantemente que explique varios aspectos del sistema al personal de operaciones y administración. Además el analista tiene que identificar los malos funcionamientos del sistema y recomendar remedios. El analista debe vigilar de manera constante el sistema buscando modos de mejorar su operación, así como para añadir nuevos elementos a medida que cambian las necesidades de los usuarios. Por último, el analista revisará de nuevo todo el sistema o subsistema a fin de recomendar a la organización un nuevo sistema o subsistema de información, con grandes mejoras.

Es evidente que es muy general la descripción anterior acerca de la función de un analista de sistemas de información es una entidad amplia y compleja. Hay muchos analistas trabajando en el diseño e implantación incluso de un sólo subsistema del sistema de información de la organización. En un momento dado, es posible que el analista ejercite actividades relacionadas con más de una etapa del ciclo vital del sistema y es necesario que estas actividades se coordinen de manera apropiada.

Para mejorar la eficiencia del sistema, el analista de sistemas se ocupará de un gran número de actividades que varían de 1) lo formal a lo informal, 2) lo cuantitativo a lo cualitativo, 3) lo estructurado a lo caótico, 4) lo específico a lo general, 5) lo tradicional a lo revolucionario. Para realizar estas actividades, el analista necesita emplear métodos y técnicas tales como diagramas de flujo, tablas de decisiones, matrices gráficas, informes descriptivos, entrevistas y técnicas de la ciencia administrativa. Estos métodos y técnicas se refuerzan entre sí al aplicarse y, cuando se les considera combinadas, constituyen la base del trabajo del analista de sistemas.

En la parte de Análisis de los sistemas de información se analizará detenidamente la metodología del desarrollo de sistemas, donde el empleo de esos métodos y técnicas, en es de importancia primordial. Sin embargo, puesto que estos auxiliares son independientes de la metodología en sí. En la figura 1.0.4 se muestra un resumen de estas técnicas y herramientas, así como la relación que existe entre éstas y las principales actividades de desarrollo de sistemas en que es posible emplearlas.

ACTIVIDADES PRINCIPALES DEL ANALISTA EN SISTEMAS

	Análisis de la Empresa		Diseño de Sistemas				Implementación de sistemas			Mantenimiento	
	Requisitos de datos	Análisis de datos	Formas y Diagramas	Diagrama de Datos	Operación del Equipo	Redacción de Especificaciones y Procedimientos del Programa	Capitación de Personal	Instalación de Software de Aplicación	Mantenimiento	Comunicación	Factor de Costo
REQUISITOS	X			X	X			X	X	X	X
DISEÑO			X	X	X	X		X	X	X	X
IMPLEMENTACIÓN						X	X	X	X	X	X
MANTENIMIENTO								X	X	X	X
OPERACIÓN								X	X	X	X
COMUNICACIÓN								X	X	X	X
FACTORES DE COSTO								X	X	X	X
ANÁLISIS DE DATOS	X	X	X	X	X	X	X	X	X	X	X
MANTENIMIENTO								X	X	X	X

Figura 1.4 Un resumen de las principales medidas y técnicas que emplea el analista en la elaboración de sistemas de información en relación con las actividades más importantes del análisis de sistemas.

La función futura del analista de sistemas

La función del analista de sistemas es cada vez más importante. En realidad justifican esta especial atención los costos del diseño, desarrollo, instalación y mantenimiento de los sistemas de información tan complejos que se requieren en la actualidad. Por lo que reviste una importancia mayor es el hecho de que no es posible dejar al azar la diferencia entre lo que costea suministrar información inadecuada o incompleta y los beneficios que ofrece la producción de información oportuna y adecuada.

1.2 Análisis de los Sistemas de Información

El notable crecimiento experimentado por las instituciones sociales, tanto en su tamaño como en su alcance, exige el desarrollo de sistemas de información igualmente amplios y complejos. Se requiere información que refleje no sólo las actividades internas de la organización, sino también la actitud de la competencia, los intereses ambientales y sociales y las tendencias políticas y económicas. Esta información debe indicar no únicamente lo que ha ocurrido o está ocurriendo, sino, lo que es más importante, lo que va a ocurrir. Es preciso conocer todo aquello que pueda contribuir a la planeación, operación y control efectivos de las actividades de la organización. La información se requiere por igual en la industria privada, en las dependencias gubernamentales, en el ejército, en las instituciones educativas, en los hospitales y en muchas otras organizaciones.

Para llevar a cabo las actividades de información de una empresa compleja, es preciso diseñar un sistema mediante el cual se pueda recopilar toda clase de datos y procesarlos de tal manera que pueda obtenerse la información requerida para tomar decisiones a todos los niveles; tanto dentro de la organización así como de las áreas que la rodean. Si queremos utilizar adecuadamente nuestros recursos, no es posible dejar que la información se produzca al azar. Para diseñar y poner en operación los sofisticados sistemas de información que exige nuestro complejo mundo actual, primero es preciso entender debidamente los conceptos básicos de datos e información. Los fines específicos del presente tema son los siguientes:

1. Dar una definición descriptiva y funcional de esos dos conceptos: datos e información.
2. Señalar, desde el punto de vista lógico y material, la forma en que los datos pueden ser procesados o manipulados para producir la información.
3. Explicar los aspectos primordiales del costo, el valor y la efectividad, en función de su relación con el procesamiento de datos e información.

1.2.1 Los Conceptos de Datos e Información

En general, los términos "datos" e "información" se usan indistintamente, aunque se refieren a dos conceptos diferentes. Desafortunadamente, está ambigüedad en el uso de la terminología a menudo da lugar a una comunicación deficiente en lo que concierne a esas dos importantes ideas: Por tanto, y con el fin de ayudar al lector a entender los conceptos mencionados, se va a definir la naturaleza de cada término de acuerdo con el uso que se le dará en esta obra.

Definición de los datos

Los datos son hechos aislados y en bruto, los cuales, situados en un contexto significativo mediante una o varias operaciones de procesamiento permiten obtener deducciones relacionadas con la evaluación e identificación de personas, eventos y objetos. La representación por medio de datos viene a sustituir a todos los factores citados. Por ejemplo, si hablamos de 12 ingenieros, de ventas al contado por valor de \$30.00 o de un edificio de oficinas, las personas, los sucesos o los objetos. La finalidad básica al recopilar y procesar los datos es producir información. El resultado del procesamiento de los datos puede aplicarse a muy diferentes fines, desde la elaboración de cheques para el pago de los sueldos, hasta la representación de un informe a la gerencia para propósitos de planeación y control. Pero si bien los datos son el elemento principal de la información, para quien los recibe, no todos los informes son igualmente importantes y oportunos. La mayoría de las empresas cuentan con una gran cantidad de datos; pero existen ciertas limitaciones cuando se trata de extraer y filtrar los hechos importantes necesarios para tomar buenas decisiones. Por ejemplo, un archivo de facturas tendrá por sí mismo muy escaso significado para el gerente de ventas; pero el procesamiento de esos datos de acuerdo con ciertos procedimientos vendrá a proporcionar una información muy significativa.

La empresa recibe una cantidad ilimitada de datos que provienen de fuentes internas y externas. Si fueran procesados sin ninguna finalidad informativa, su abundancia sería excesiva. Dicho de otro modo, sólo podrá obtenerse la cantidad apropiada de datos mediante un sistema adecuado de información. Por ejemplo los informes producidos por dichos sistemas de procesamiento de datos y colocados sobre el escritorio de los administradores ha sido comparados con un periódico que fuera impreso diariamente sin encabezados o careciendo de espacios entre palabras: una avalancha de datos pero muy pobre información. Esto implica que el volumen de datos no ha sido procesado en forma apropiada y de modo que satisfaga las necesidades del destinatario.

Hay que hacer notar desde que ningún método de procesamiento, ya sea manual o por computadora, garantiza que los datos serán procesados según las necesidades del usuario. Esas necesidades deben ser determinadas dentro de un contexto algo diferente al método de procesamiento utilizado.

Definición de la Información

La información se distingue substancialmente de los datos en que estos últimos son mensajes en bruto y no evaluados, mientras que la primera significa un aumento de conocimientos, obtenido por el receptor mediante la coordinación apropiada de los elementos con las variables de un problema. La información es la adición o el procesamiento de los datos, que puede proporcionar un conocimiento o bien un entendimiento de ciertos factores.

Si todos poseyéramos un conocimiento perfecto, no habría necesidad de información. Todo aquel que posea un conocimiento menos que perfecto necesita de la información como auxiliar para tomar decisiones. Para la mayoría de las personas, la información es el componente principal de sus decisiones, aunque por sí sola no garantiza que dichas decisiones sean las correctas. Dentro de cualquier sistema de procesamiento de datos es posible almacenar gran volumen de éstos, lo mismo que disponer de ellos con facilidad; pero la finalidad consiste en adaptar el sistema de manera que responda a las necesidades de información de los usuarios. Si se va a diseñar un sistema de información, es preciso dejar claro qué se considera como tal. Y luego, una vez que los usuarios han recibido la información, les corresponde a éstos aplicarla a sus conocimientos actuales y empezar la acción que esté indicada.

La información es un acontecimiento, o una serie de acontecimientos, que llevan un mensaje y que, al ser percibida por el receptor mediante algunos de sus sentidos, amplía sus conocimientos. Sólo el destinatario puede evaluar la significación y la utilidad de la información recibida.

Por ejemplo, los pedidos recibidos en el almacén dentro de un determinado periodo tendrán poco significado para el contador, quien los considera simples datos. Sin embargo, los envíos por cobrar enviados por el almacén dentro del mismo periodo, de mucho constituyen información para ese mismo contador. Al gerente de ventas le interesarán ambas cosas, mientras que al jefe de personal no le interesan los pedidos ni los envíos.

Idealmente, la información da a conocer el estado de las cosas; pero la percepción puede diferir notablemente entre los usuarios. El hecho de que la información contenga el mismo mensaje y sea comunicada de la misma manera y en el mismo momento no implica que será utilizada igualmente por los destinatarios. El juez que escucha a los testigos llega a una decisión de 5 contra 4 basándose en los mismos argumentos, y los analistas llegan a diferentes conclusiones con respecto a la situación financiera dentro de una empresa después de examinar los mismos estados financieros. Así la forma en que se utiliza la información depende de buena medida de la percepción, antecedentes y prejuicios del usuario.

La Función de la Información

La función primordial de la información, y por tanto de un sistema de información, consiste en aumentar el conocimiento del usuario, o de reducir su incertidumbre. La información comunicada al usuario puede ser el resultado de la aportación de datos a un modelo de decisión, y de su procesamiento. Sin embargo, tratándose de las decisiones más complejas, no puede hacer más que aumentar la probabilidad de la incertidumbre o de reducir el número de posibles elecciones.

La información proporcionada a quien debe tomar la decisión indica la probabilidad del factor de riesgo a diferentes niveles de ingresos. Por ejemplo, la probabilidad de fracasar en el nivel de ingresos D puede ser de .75. El riesgo es demasiado grande para el inversionista particular y el inversionista optará por abstenerse. En cambio, para el proyecto E la probabilidad de fracaso es de .35, o sea lo suficientemente bajo para que el inversionista acepte el riesgo. Como puede verse en este sencillo modelo, la información sólo puede señalar probabilidades y quizá una serie de alternativas a diferentes niveles de ingresos, por tanto, la función de la información consiste en reducir la gama de soluciones.

En una de 6 cajas se encuentra un billete de 1,000.00 (Ver figura 1.2.2), con certeza en la número 6). La cuestión es determinar en cuál de ellas se encuentra. Careciendo de información, las probabilidades de localizar la caja correcta son de 1/6 y las de fallar son de 5/6 (es decir, 1 - 1/6). Por tanto, en forma puramente intuitiva, se tiene una probabilidad de 1/6 de obtener \$100 (suponiendo que se da una sola oportunidad); pero, si quien va a decidir es informado que el dinero se encuentra en la caja 1 o en la 6, las probabilidades de éxito habrían aumentado a 1/2.

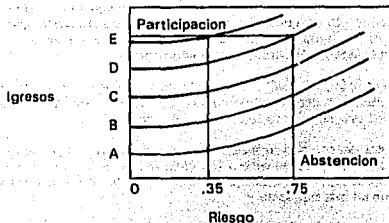


Figura 1.2. Gráfica que indica el grado de riesgo a diferentes niveles de ingreso (Para proyectos).

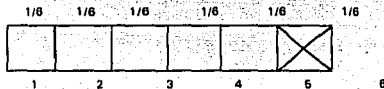


Figura 1.2.2 Hiera de cajas, una de las cuales contiene \$1,000.00

En los dos ejemplos anteriores, la función de la información consistió en dar a quien toma la decisión una base de probabilidad para su elección. No le ha indicado en forma directa lo que debe hacer, sino que ha reducido la gama de decisiones y la incertidumbre para una decisión inteligente. Otra de las funciones principales de la información es la de proporcionar un conjunto de normas, reglas de evaluación y reglas de decisión para la determinación y comunicación de advertencias y retroalimentación para fines de control. En otras palabras, suponiendo que quien toma la decisión decide invertir en un proyecto, necesita información que le ayude a controlar las operaciones que componen el proyecto.

De modo general, hay muchas partes de la información que pueden ser útiles y que influyen de un modo u otro en la respuesta del receptor ante una situación determinada. Cierta información puede surgir de la observación personal; otra puede provenir de las conversaciones sostenidas con otras personas o de las juntas del comité; otra puede originarse agentes externos tales como los periódicos u otros medios noticiosos, o en los informes del gobierno, y otra puede venir del propio sistema de información. Se insiste nuevamente que el sistema de información sólo puede proporcionar una parte de la información que necesita quien ha de tomar las decisiones y que esa información es de naturaleza formal y cuantificable. En un sentido general, el sistema de información provee al usuario, de información con respecto al estado de las cosas, comunicándole un grado mayor de predictibilidad tanto por lo que se refiere a los sucesos como por lo relativo al resultado de las actividades con la empresa. De manera que toda aquella información que pueda ser manejada, o producida, y que pueda ser cuantificable, resultará interesante.

Clasificación de la Información

No existe hasta la fecha un esquema de clasificación de la información, que sea universalmente aceptado y adaptable a todas las situaciones. Cada comunicador aporta una clasificación que se considera apropiada para alcanzar el objetivo de lograr entendimiento. En realidad, con frecuencia es difícil llegar a un acuerdo con respecto a qué constituye la buena o mala información.

Por ejemplo, si deseáramos clasificar la información de acuerdo con su origen respecto a la organización, nuestra clasificación podría consistir en decir si es externa o interna. Podríamos hacer esta misma distinción basándonos en aquellos a los que se permite usar dicha información.

La figura 1.2.3 es una tabla que ilustra las muchas formas en las que la información puede ser clasificada. No se pretende que sea completa ni definitiva; se ofrece con el fin de que el lector reconozca la pluralidad de las clasificaciones y pueda apreciar el hecho de que, a menos que la comunicación sea del todo clara, existe siempre la posibilidad de un malentendido.

Metro de Relevancia	1	2	3	4	5	6	7	8
Relación a la organización	X							
Relaciones de tiempo								
Fuente de datos			X					
Procesamiento de datos			X					
Comunicación de la información				X				
Naturaleza de la información								
Potencial funcional del usuario					X			
Flujo de información						X		
Cantidad de la información							X	
Clasificación de la información								
Interna		X						
Externa			X					
Por área				X				
Estadística				X				
En línea					X	X		
Temporales					X	X		
Períodos					X	X		
Descriptiva Antecedentes						X		
Rendimiento Presente						X		
Predictiva Futuro						X		
Símbolo ¿Qué pasa si?						X		
Constituyente de la Organización							X	
Nivel estratégico							X	
Nivel táctico							X	
Nivel operativo							X	
Horizontal								X
Vertical								X
Permanente								X
Oportuna								X
Útil								X

Figura 1.2.3 Clasificación de la información

1.2.2 Producción de Información a Partir de los Datos.

Habiendo explicado en la sección anterior los conceptos de los datos e información, corresponde ahora examinar la forma en que puede producirse la información a partir de los datos. Esta transformación se puede enfocar desde un punto de vista lógico, es decir, de las operaciones lógicas que se realizan con los datos, y desde el punto de vista material y técnico, o sea, de los métodos de más alto nivel de acuerdo con los cuales se llevan a cabo las operaciones con los datos.

Operaciones con los Datos

Básicamente, los datos son material en bruto que se necesita manejar y situar en un contexto significativo para que pueda ser útil a quien va a recibirlo. Para poner los datos en orden y dar resultados comprensibles, es preciso efectuar alguna combinación de operaciones básicas. En todo sistema de informaciones posible identificar diez operaciones básicas que pueden dar resultados comprensibles. La función que estas operaciones desempeñan dentro de un sistema de información es similar a la de las máquinas individuales en función de sus componentes. Así como todas las máquinas sencillas, también los sistemas complejos de información están compuestos de alguna combinación de operaciones sencillas con los datos. Las operaciones a que se refiere son las siguientes:

1. Captación. Esta operación se refiere al registro de datos hecho a partir de un evento o acontecimiento, en forma de notas de ventas, nóminas de personal, órdenes de compra, medidores, calibradores, etc.

2. Verificación. Se refiere a la comprobación o validación de los datos, hecha con el fin de asegurarse de que fueron obtenidos y registrados en forma correcta.

3. Clasificación. Esta operación clasifica los elementos de los datos en categorías específicas que tienen un sentido para el usuario. Por ejemplo, los informes sobre ventas pueden ser clasificados por el tipo de inventario, tamaño, cliente, vendedor, almacén de donde fue remitido o cualquier otra clasificación que las defina con claridad.

4. Ordenación. Mediante esta operación los elementos de información se colocan en secuencia específica predeterminada. Por ejemplo, un registro de inventario podrá ser ordenado por clave del producto, nivel de actividad, valor monetario o cualquier otra característica que figure en el registro y que el usuario considere útil.

5. Sumarización. Esta operación combina o engloba los datos de dos maneras. Primero, los acumula en sentido matemático, como cuando se fórmula un balance general. La cantidad clasificada como activo circulante representa sin duda miles de cuentas específicas y más detalladas. Segundo, reduce los datos en el sentido lógico, como cuando el jefe de personal solicita una lista exclusivamente de los empleados asignados al departamento 13 de la empresa.

6. Cálculo. Esa operación vincula las operaciones aritméticas y lógicas de los datos. Por ejemplo, es necesario efectuar cálculos para establecer los sueldos devengados, la cobranza a los clientes, la calificación de los estudiantes, etc. En muchos casos es necesario efectuar cálculos bastante complicados para mantener los datos contenidos en ciertos modelos de administración científica, programación lineal, pronóstico, etc.

7. Almacenamiento. Mediante esta operación, los datos se guardan en algún dispositivo, como papel, microfilm o dispositivos magnéticos, donde se pueden tener disponibles y consultados cuando sea necesario.

8. Recuperación. Esta operación implica buscar y obtener acceso a datos específicos, para tomarlos del dispositivo en que se encuentran almacenados.

9. Reproducción. Esta operación copia los datos de uno a otro dispositivo o cambia su ubicación dentro del mismo. Por ejemplo, los datos almacenados en un disco magnético pueden ser reproducidos en otro disco igual o en una cinta magnética, ya sea para procesamiento posterior o por razones de seguridad.

10. Distribución/comunicación. Mediante esta operación se transfieren los datos de un lugar a otro. Se puede efectuar en muchos puntos diferentes del ciclo del procesamiento. Por ejemplo, los datos pueden ser transferidos del dispositivo al usuario. La finalidad de todo procesamiento de datos es entregar información al usuario.

La Selección de Métodos de Procesamiento.

La selección del método de procesamiento más adecuado para una aplicación u organización específicas exige que el analista de sistemas conozca a fondo tanto las necesidades de procesamiento como las posibilidades de cada uno de los métodos.

Puede decirse que las necesidades de procesamiento están determinadas por las consideraciones siguientes: 1) el volumen de los datos involucrados, 2) la complejidad de las operaciones de procesamiento, 3) las limitaciones impuestas al tiempo de procesamiento y 4) las demandas de cálculo. Para que el analista pueda elegir el método correcto de procesamiento, es fundamental que tenga presente la existencia de esos cuatro elementos. En muchas empresas, un elemento predomina hasta tal punto que los otros tres no siempre quedan cuidadosamente definidos. Por ejemplo, un importante banco procesa tal cantidad de cheques que solo el volumen justificaría la elección de una computadora. En cambio, un banco pequeño tendrá que considerar los otros aspectos de sus necesidades de procesamiento antes de elegir un método específico. Del mismo modo, una importante firma de ingenieros podría elegir la computadora basándose únicamente en sus necesidades de cálculo masivo, mientras que una empresa pequeña debe tener en cuenta el volumen, la complejidad y el tipo antes de hacer una elección.

La pregunta de cuándo una empresa elige un cierto método de procesamiento de datos con preferencia a otro, es una buena medida una decisión de carácter económico. De todas maneras, resumiendo, se puede decir que a medida que el volumen de datos se incrementa, la complejidad aumenta, las restricciones al tiempo se tornan cada vez más severas y las exigencias de cálculo se vuelven más complejas, se refiere a un nivel de automatización.

El conocimiento de las posibilidades es igualmente importante cuando se elige un método de procesamiento de datos. Aunque hay muchos aspectos a considerar, a continuación se definen quince factores básicos:

1. Inversión Inicial. Es el costo de la adquisición de los materiales y máquinas que se requieren para el procesamiento.

2. Preparación. Es el gasto que implica preparar inicialmente los datos obtenidos para el procesamiento subsecuente.

3. Conversión. Es el costo único del procesamiento inicial de los datos mediante el nuevo método.

4. Personal especializado. El nivel de conocimientos y capacitación de las personas que tendrán a su cargo el procesamiento.

5. Costo variable. El costo de una unidad de datos en relación con los cambios ocurridos en su volumen.

6. Modularidad. La posibilidad de aumentar o disminuir la capacidad de procesamiento de acuerdo con las necesidades. Por ejemplo, si una máquina puede procesar 1,000 operaciones por un día pero es necesario procesar 1,200, se necesitarán dos máquinas. Esto dará lugar a la sub-utilización de la capacidad de procesamiento.

7. Flexibilidad. Es la posibilidad de modificar el sistema de procesamiento para hacer frente a demandas nuevas y variables.

8. Versatilidad. Es la posibilidad de llevar a cabo muchos trabajos diferentes.

9. Velocidad de procesamiento. El tiempo necesario para convertir los datos en información.

10. Poder de cálculo. La posibilidad de realizar operaciones matemáticas complejas.

11. Control de procesamiento. La posibilidad de comprobar que cada una de las tareas de procesamiento se lleve a cabo conforme lo esperado.

12. Detección automática de errores. La posibilidad, por parte de los componentes del sistema, de detectar errores de procesamiento.

13. Poder de decisión. La capacidad de elegir entre alternativas para seguir con el procesamiento.

14. Alteración del sistema. El grado en que el sistema de procesamiento pierde eficacia debido a la descompostura o falta de componente o componentes. Por ejemplo, la descompostura de CPU implica la alteración total del sistema de computación, mientras que, en un sistema manual, la ausencia del empleado alterará sólo ligeramente la capacidad de procesamiento.

15. Nivel de automatización. Se explica por sí mismo.

1.2.3. Aspectos Económicos de la Información.

La información es un recurso sumamente valioso para toda organización. La mayor parte de las empresas no podría sobrevivir si careciera de información formal. En muchas de ellas se observa una tendencia cada vez mayor a ampliar la efectividad y la utilización de la información más allá de la simple vigilancia de los aspectos legales y de la solución rutinaria de problemas. Con el fin de acelerar el progreso de sus empresas, los administradores buscan activamente la solución de nuevos problemas. Para hacer frente a estas necesidades cada vez mayores se necesitan fuertes inversiones. La pregunta es: ¿justificarán la inversión de los beneficios recibidos?

La elaboración de información formal cuesta dinero. ¿Cuánto debe gastar una empresa para obtener información? Aún cuando fuera posible, no sería económico registrar y procesar cada uno de los datos disponibles. El costo del registro y procesamiento para obtener información debe compararse con el valor que dicha información tendrá para el receptor. Ya se ha hecho alusión al costo de producción de la información y el valor que está representando; pero hay una estricta división entre la estimación del valor que representa. El costo de producción de la información (es decir, el costo del sistema con sus

métodos, dispositivos y medios) es algo tangible y más o menos fácil de medir. Por otra parte, la información es conceptual por naturaleza y no tiene características tangibles salvo en representaciones simbólicas. Básicamente, es diferente y separada de los métodos materiales que la producen y carecen por sí misma de representación física. El analista, sin embargo, debe hacer frente tanto al costo de la información como a la efectividad de la misma, con el fin de optimizar las funciones del sistema. En esta sección se dará a conocer y se analizará el problema de costo/valor.

El Costo de la Información.

En algunas empresas, el procesamiento de datos para hacer frente a las operaciones legales y rutinarias y para obtener también información de alto nivel, representa entre 5 y 15% del costo total de operación de la empresa. En ciertas organizaciones financieras, el costo puede elevarse hasta 50%. Los costos de operación del sistema de información se identifican como sigue:

1. Costo del equipo. Dentro de ciertos límites, en general éste es un costo fijo o perdido, que aumentará con los altos niveles de mecanización.

2. Análisis de sistemas, diseño e implantación. Es un costo perdido que aumentará normalmente con los altos niveles de mecanización. Esta función incluye la Formulación de una metodología para los procesamientos generales de procesamiento de datos. Si se usa la computadora, será necesario incluir también la preparación de programas.

3. Costo del espacio y control de factores ambientales. Este es un costo semivariable. Como ejemplos podemos mencionar el del espacio, en los sistemas de acondicionamiento, en el de las unidades de control de energía, el de medidas de seguridad, etc. Normalmente estos costos aumentan, con los altos niveles de mecanización.

4. Costo de conversión. Es un costo perdido que incluye toda clase de cambios.

5. Costo de operación. Básicamente es un costo variable y comprende las diversas clases de personal, la instalación y mantenimiento de sistemas, los suministros, los servicios y su conservación. Estos costos se clasifican a menudo como variables y no variables. La gráfica que aparece en la figura 1.2.5 ayudará a visualizar el comportamiento del volumen de datos procesados en base al nivel de automatización.

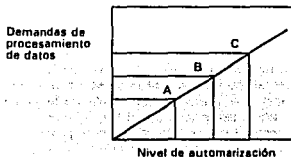


Figura 1.2.5 Relación entre las necesidades de procesamiento y nivel de automatización A: punto teórico de separación de los métodos manual y electromecánico, B punto teórico de separación entre el método electromecánico y el equipo a base de tarjetas perforadas, C punto teórico de separación entre el equipo a base de tarjetas perforadas y la computadora.

La escala vertical está expresada en dinero e indica los costos variables y no variables en relación con el volumen de unidades procesadas. El volumen aparece sobre la escala horizontal. El costo no variable, o sea la inversión en el sistema de computación, es de \$150,000.00 El costo variable por unidad de datos procesados es de \$0.15. de manera que, con el volumen de 100,000 unidades procesadas, el costo total es de \$165,000 ($\$.15 \times 100,000 + 150,000$).

Para el método electromecánico de la figura 1.2.5, el costo no variable es de \$30,000.00 y de \$1.35 por unidad procesada. Para calcular el punto de equilibrio, digamos que x es igual al número de unidades de datos procesados. Por lo tanto tendremos:

$$\begin{aligned} \$.15x + \$150,000.00 &= \$1.35x + \$30,000.00 \\ \$1.20x &= \$120,000.00 \\ x &= 100,000 \text{ unidades} \end{aligned}$$

Dicho de otro modo, con el volumen 100,000 unidades, el uso de la computadora o del método electromecánico son iguales en costo. si se considera únicamente el costo, permaneciendo todo lo demás igual, a un volumen menor de 100,000 unidades de datos resultará más económico usar el método electromecánico. Por ejemplo, a un volumen de 90,000 unidades, el costo total con el método electromecánico sería de \$151,500.00 ($\$.15 \times 90,000 + \$150,000$). Usando la computadora, el costo sería de \$163,500 ($\$.15 \times 90,000 + \$150,000$). En cambio, con un volumen de unidades mayor de 100,000, el uso de la computadora se justificaría económicamente.

Haciendo esta división general de los factores del costo, las curvas de las figuras 1.2.6 y 1.2.7 indican el comportamiento relativo del costo de los diferentes métodos de procesamiento. La conclusión general que se deriva de esas curvas de costo es que el método manual resulta más económico para volúmenes bajos de procesamiento, y que, a medida que aumenta el volumen, un mayor nivel de mecanización quedará justificado. Sin embargo, como ya se dijo, el costo aunque es muy importante, no es la única consideración.

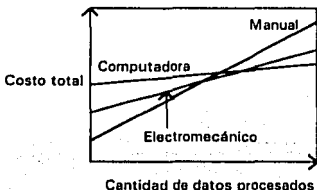


Figura 1.2.6 Costo total de diferentes métodos de procesamiento de datos.

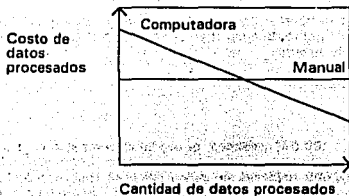


Figura 1.2.7 Costo promedio de diferentes métodos de procesamiento de datos.

Hasta aquí, nuestro análisis de costos ha supuesto que las condiciones son estáticas. Una suposición dinámica sería que el costo de los diferentes métodos no permanece constante con el paso del tiempo, debido a los adelantos técnicos, mejoras de producción masiva, mas facilidades para compartir el tiempo de la computadora, mayor utilización del equipo y niveles más altos de salarios. Todos estos factores, con excepción de los mayores salarios, han mejorado significativamente la relación entre precio y rendimiento de las computadoras. En otras palabras, resulta cada vez menos costoso procesar una unidad de datos por medio de la computadora. A la inversa, los costos del personal de oficina se han incrementado sin un aumento notable en el rendimiento y en la productividad. Por tanto, como puede verse en la figura 1.2.8, la mejora en la relación precio/rendimiento de las computadoras hace que este método sea competitivo con los métodos manuales y otros menos mecanizados a volúmenes más bajos de procesamiento. Podemos concluir entonces, que la influencia de las condiciones cambiantes ha ocasionado una disminución del costo del procesamiento por computadora. Parece que esta tendencia va a continuar. Hay que hacer notar que este análisis no comprende otras necesidades mayores de procesamiento, debido a la complejidad, las limitaciones de tiempo y la amplia capacidad de cálculo, que por lo común otros métodos menos mecanizados no pueden manejar.

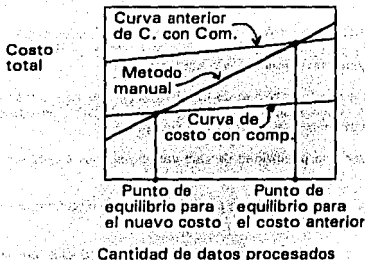


Figura 1.2.8 Efecto de una reducción del costo del cálculo (mejoramiento de la relación costo/rendimiento) en el volumen de procesamiento con el cual el uso de la computadora resulta económico comparado con métodos manuales.

El Valor de la Información

El valor de la información está basado en diez características, que se enumeran a continuación.

1. **Accesibilidad.** Esta característica se refiere a la facilidad y rapidez con que se puede obtener la información resultante. Por ejemplo, la rapidez de acceso puede medirse en términos de un minuto contra veinticuatro horas. Pero ¿Qué valor tiene esa información para el usuario? ¿Valdrá, por ejemplo, \$10.00 más por acceso con respecto al viejo método?

2. **Comprensibilidad.** Se refiere a la integridad del contenido de la información. No se refiere necesariamente al volumen sino que el resultado sea completo. Esta característica es sumamente intangible y, por tanto, difícil de cuantificar.

3. **Precisión.** Se refiere a que no haya errores en la información obtenida. Cuando se trata de un gran volumen de datos, en general se producen dos clases de errores: de transcripción y de cálculo. Muchos aspectos de esta característica pueden ser cuantificados. Por ejemplo, ¿cuál es la proporción de errores por cada mil facturas elaboradas por métodos manuales y por computadora? ¿qué valor representa el reducir el número de errores? por ejemplo, ¿Aumentarán las ventas 0.5% si el número de errores de facturación se reduce en 10%?

4. **Propiedad.** Se refiere a qué tan bien se relaciona la información con lo solicitado por el usuario. El contenido de la información debe ser apropiado para el asunto de que se trate; todo lo demás será superfluo y a la vez costoso en la elaboración. Esta característica es difícil de medir y al igual que ocurre con los demás, ojalá esté en la eficacia del diseño de los sistemas.

5. Oportunidad. Esta característica se relaciona una mejor duración del ciclo de acceso: entrada, procesamiento y entrega al usuario. Por lo común, para que la información sea oportuna, es preciso reducir la duración de este ciclo. La oportunidad puede medirse en algunos casos. Por ejemplo, ¿En qué proporción se podrá aumentar las ventas si se informa inmediatamente al cliente sobre disponibilidad de artículos de inventario?

6. Claridad. Esta característica se refiere al grado en que la información está exenta de expresiones ambiguas. La revisión de un informe puede resultar costosa. ¿Cuánto cuesta revisar este informe? (A la claridad puede asignarse un valor muy preciso en dinero).

7. Flexibilidad. Conciernen a la adaptabilidad de la información, no sólo a más de una decisión, sino a más de un responsable de la toma de decisiones. Esta característica es difícil de medir; pero puede asignarse un valor cuantificado dentro de un margen muy amplio.

8. Verificabilidad. Se refiere a la posibilidad de que varios usuarios examinen la información y lleguen a la misma conclusión.

9. Imparcialidad. Se refiere a que no exista un intento de alterar o modificar la información con el fin de llegar a una conclusión preconcebida.

10. Cuantificable. Se refiere a la naturaleza de la información producida por un sistema formal de información. Aunque a veces los rumores, conjeturas, etc. se consideran como la información, están fuera de nuestro ámbito.

Aunque muchos aspectos de estas características son difíciles, el analista debe utilizarlas cuando determina el valor de la información para compararlo con su costo. La cuestión es: ¿Qué valor tiene para el receptor un a cierta parte de la información? Refiriéndonos de nuevo a la figura 1.2.2, el receptor tenía una probabilidad de 1/6 de elegir la caja que contenía los \$1,000 si enfocaba intuitivamente la decisión. En este ejemplo, el valor esperado, con cero información, es de \$166.67. Con cierta parte de la información, las probabilidades de éxito se elaboraron a 1/2, o sea un valor esperado de \$500.00. Esa porción de información recibida, ¿Beneficia a quien toma la decisión? No lo sabemos, a menos que conozcamos el costo de esa información. Si se pagaron más de \$333.33, el costo de la información excede a su valor. Si se pagaron \$333.33, el receptor no ganará ni perderá por término medio (\$500.00 - \$166.67). ¿Y si conseguimos información suficiente por una probabilidad de 1.0, o sea el nivel de certeza? Nuevamente, si hay que pagar más de \$833.33, la información habrá costado más de lo que vale.

Costo contra Valor.

El objetivo del sistema de información es alcanzar un punto óptimo, donde el valor marginal de la información sea igual al costo marginal de su obtención. Esta relación aparece en la figura 1.2.9. Como puede verse en la ilustración, una cantidad excesiva de información, con un costo relativamente elevado, puede dar lugar a un valor marginal negativo.

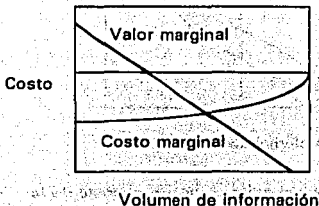


Figura 1.2.9 Relación entre el valor y el costo marginal de la información.

El nivel óptimo de procesamiento de la información aparece cuando el costo marginal de la información es igual a su valor marginal. Con respecto al nivel de resultados pueden sentarse los siguientes principios:

1. Si el valor marginal es $>$ el costo marginal: aumentese el resultado.
2. Si el valor marginal es $<$ el costo marginal: límitese el resultado.
3. Si el valor marginal = al costo marginal: el resultado es óptimo.

En términos generales, podemos ilustrar el nivel óptimo en la figura 1.2.10.

El costo total de la información está representado por LB; el valor bruto está representado por LA; y el valor neto por AB. El nivel óptimo de la información se halla en L.

Supongamos que la empresa puede obtener información mediante tres métodos diferentes (M1, M2 y M3). En relación con el costo, ¿Cuál método es más afectivo? Cuando se elige entre métodos puede producirse un efecto interesante: la relación costo/efectividad. Dicha relación se ilustra en la figura 1.2.11.

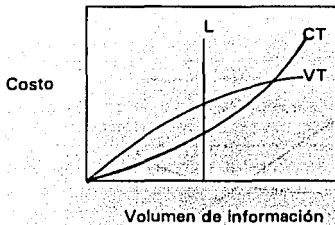


Figura 1.2.10 Relación entre el valor y el costo de la información.

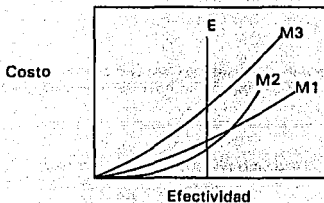


Figura 1.2.11 Relación de costo/efectividad de los diferentes tipos de información.

El análisis y diseño del sistema ha establecido un nivel designado por E. Sin embargo, el analista debe suponer una empresa dinámica cuyas necesidades cambian, lo que a su vez hace cambiar las demandas sobre sistema de información. Por lo tanto, el analista debe conocer las relaciones de costo/efectividad de los distintos métodos. Si elige a M1, puede lograrse un incremento en la demanda de información por otra parte la empresa con un aumento relativamente pequeño en el costo. El método M2 dará un pequeño incremento de efectividad acompañado de un gran incremento en el costo. En cuanto a M3, supondremos que representa un método que está más allá de las posibilidades de la empresa. Sin embargo, en muchos de los casos las empresas han tratado de adquirir sistemas que no guardan proporción con sus recursos. Evidentemente, el nivel apropiado dependerá de los métodos disponibles para obtener información y de la capacidad de la empresa para permitirse un método en particular.

1.3. Conceptos de Análisis de Sistemas, Administración y Sistema Formal de Información.

Antes de pasar a otros puntos y emprender estudios más detallados de las computadoras y de la tecnología relacionada con ellas, de los sistemas de información de los bancos de datos, y del análisis de sistemas es conveniente que establezcamos un sólido conocimiento de los conceptos de sistemas, administración y sistemas de información. Quién estudia los sistemas de información deberá entender básicamente esos conceptos, con el fin de tener un marco de referencia dentro del cual puede hacer uso de los sistemas de información lo mismo que implantarlos. Los objetivos específicos del presente tema son los siguientes:

1. Presentar un análisis conceptual detallado de los sistemas, mediante un enfoque de clasificación general y de, entrada y salida.
2. Establecer un conocimiento básico de la función administrativa, particularmente en el contexto de la relación que existe entre administradores y elementos de información que les permiten administrar eficientemente su organización.
3. Resumir los dos puntos anteriores presentando un método conceptual de todo el sistema.

1.3.1 Exposición General de los Sistemas

La palabra "sistemas" es un término ubicuo que sirve para describir un gran número de cosas, objetos, métodos o grupos. Un grupo ordenado de hechos, principios, doctrinas, creencias, y tradiciones; por ejemplo, el sistema de gobierno democrático, se describe como un sistema. El sistema respiratorio de un ser viviente es un conjunto de órganos o partes que desempeñan una o más funciones vitales. Este sistema mantiene vivo al ser, el que a su vez puede ser considerado como un sistema. Una organización compuesta por personas, dinero, métodos, materiales y máquinas, constituye un sistema comercial. Un equipo de computación, que comprende cierto número de máquinas impresoras, unidades de memoria, elementos de control, dispositivos de cintas y cartuchos y muchos otros dispositivos de procesamiento, a menudo recibe el nombre de sistema, porque representa un grupo de objetos relacionados entre sí que forman una red unificada.

Es verdaderamente necesario entender los sistemas y trabajar con ellos en forma inteligente. Por esta razón es preciso aplicar el enfoque de sistemas para conocer las características significativas de uno de ellos, las que a su vez sugirirán los cambios que pueden ser introducidos para mejorarlo. Además, la aplicación del enfoque de sistemas implica que los componentes del sistema se distribuirán e integraran de manera que su efectividad general sea óptima. Es decir, la aplicación del enfoque de sistemas permite lograr un efecto sinérgico, de manera que la acción unificada de las diferentes partes del sistema produzca un efecto mayor que la suma de dichas partes.

Clasificaciones de los Sistemas

Clasificación General de Sistemas

Un útil punto de inicio para llegar a una definición precisa es tomar una definición general que incluya todas las interpretaciones anteriormente citadas, esto es la definición del diccionario: "un sistema es un conjunto estructurado de objetos y (o) atributos junto con relaciones entre ellos".

Como puede verse, en principio, el sistema es un conjunto; esto es, contiene elementos que tienen alguna razón para que se consideren juntos, por encima de otras razones. Pero es más que un simple conjunto, también contiene las relaciones que existen entre sus elementos. De manera de que si el sistema es un paquete computacional, los elementos serán las instrucciones y las relaciones se definirán con la estructura particular del programa.

Esta definición general, aunque proporciona un modelo de referencia global en el que pueden ubicarse todas las interpretaciones, no es bastante bueno para usarse como un lenguaje de modelado.

La definición puede refinarse primero derivando una clasificación en términos de tipos de sistemas y luego desarrollando un conjunto de conceptos adecuados a cada tipo. La clasificación particular adoptada en Lancaster (Checkland, 1971) puede resumirse así:

a) **Sistemas Naturales:** Sistemas Físicos que integran el universo, en una jerarquía de sistemas subatómicos desde los sistemas de ecología hasta los sistemas galácticos.

b) **Sistemas Diseñados:** Estos pueden ser tanto físicos (herramientas, puentes, complejos automatizados) como abstractos (matemáticas, lenguaje, filosofía)

c) **Sistemas de Actividad Humana:** Por lo general describen los seres humanos que emprenden una actividad determinada, como los sistemas hombre-máquina, la actividad industrial, los sistemas políticos, etc.

d) **Sistemas Sociales y Culturales:** La mayor parte de la actividad humana existirá en un sistema social donde los elementos serán interpersonales. éste es diferente por naturaleza a las otras tres clases, que abarca la interfaz entre los sistemas de actividad natural y humana. Ejemplos de sistemas sociales pueden ser la familia, una comunidad, al igual que el conjunto de sistemas formado por seres humanos agrupados para desempeñar alguna actividad determinada, como la agrupación por una excesiva industrialización, una sociedad coral o una conferencia.

Clasificación de Sistemas por su grado de predictibilidad y complejidad

Un modo conceptual de clasificar los sistemas se basa en dos criterios distintos, que son los siguientes:

1. Nivel de predictibilidad. Este criterio está basado en un doble esquema: determinístico y probabilista.

2. Nivel de complejidad. Adoptado este criterio, es posible clasificar los sistemas en tres categorías: sencillos, complejos y sumamente complejos.

Cuando es posible pronosticar con certeza la manera en que responderá un sistema, o cómo funcionará en circunstancias diferentes, decimos que el sistema es determinista. No hay ninguna duda sobre los resultados. Dado que el estado último del sistema y la cantidad de información actualmente presente y determinada, será posible predecir, sin riesgo de equivocarse, el estado subsecuente del sistema. A la inversa, cuando no es posible pronosticar con certeza los resultados de un sistema particular, dicho sistema es probabilista es, a fin de cuentas, un sistema determinista que no logramos entender del todo. Esta observación encierra un elemento de verdad que trataremos más adelante. Como quiera que sea, todos los administradores y analistas tiene que enfrentarse a los sistemas probabilistas.

Un sistema probabilista sencillo es aquel que contiene pocos subsistemas e interrelaciones y revela un comportamiento enteramente predecible. Unos balines de acero colocados sobre una pista (como en las máquinas de juego de canicas) dan un ejemplo de un sistema determinista sencillo. Sin embargo, una vez que los balines salen de la pista, sus caminos se vuelven probabilistas. Una hilera de máquinas en la línea de producción se puede estudiar y disponer de manera que se reduzca al mínimo la distancia que recorran los materiales. Cuando se hace necesario estudiar lo que ocurre realmente cuando los materiales comienzan a fluir, el sistema se torna inmediatamente en probabilista.

Las mismas consideraciones son aplicables cuando estudiamos un sistema determinista complejo. La computadora es un sistema complejo, pero es determinista en el sentido de que hará aquello para lo cual fue programada. Un almacén automatizado es complejo, pero también determinista. Cuando el comportamiento de un sistema determinista complejo deja de ser predecible por completo, significa simplemente que ha habido alguna descompostura (una falla).

Un sistema sencillo también puede ser probabilista. Por ejemplo, el lanzar al aire una moneda es un sistema sencillo; pero es a la vez notoriamente probabilista. Un sistema de control de calidad que pronostica el número de piezas defectuosas en un conjunto, es un sistema sencillo pero al mismo tiempo probabilista.

Una empresa es un sistema probabilista complejo cuyo objetivo es la obtención de utilidades. Se toman decisiones que influyen en varios de los subsistemas que componen el sistema, y la influencia de tales decisiones, modifica hasta cierto punto las operaciones de la empresa con el fin de alcanzar cierto nivel de utilidades, todo lo cual es probabilista.

Hasta aquí hemos estudiado cuatro de las seis categorías señaladas. Un sistema tan complicado que resulte casi indescriptible será sumamente complejo. Debido a que esta quinta categoría es tan compleja, en consecuencia no podrá ser determinista. Por tanto, no existe la clasificación sistema determinista sumamente complejo.

Sin embargo, en la categoría probabilista sumamente compleja, los resultados son diferentes por ejemplo, la economía nacional es tan compleja y tan probabilista, que quizá nunca llegará a ser descrita del todo. Anteriormente se dijo que las organizaciones de nuestro tiempo se han vuelto muy complejas,

hasta el punto de que muchas de ellas entran en la categoría de probabilistas sumamente complejas. Es imperativo desarrollar técnicas de procesamiento de información que ayuden a afrontar esa complejidad.

En la figura 1.3.1 aparece una clasificación de sistemas por su predictibilidad y complejidad.

Sistema	Simple	Complejo	Extremadamente Complejo
D E T E R M I N I S T I C O	Balines de acero en una placa	Computadora	No hay Clasificación
	Línea de ensamble de producción	Bodega Automática	
P R O B A B I L I S T I C O	Lanzar una moneda al aire	Pequeña corporación	Gran corporación
	Sistema de control de calidad	Sistema de inventario	Economía nacional

Figura 1.3.1 Tabla de clasificación de sistemas.

Análisis de Causa y Efecto.

Los sistemas se pueden analizar en términos de entradas y salidas. Los elementos que entran se consideran las causas que interactúan para producir una salida; la salida constituye el efecto. Esta relación de causa y efecto se ilustra en la figura 1.3.2. El sistema actúa como una caja de funciones que cambian los elementos de entrada en resultados o elementos de salida.

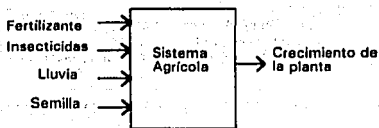


Figura 1.3.2 Sistema donde se ilustra la relación de la causa y efecto.

En un sistema agrícola, la combinación de los elementos de entrada, es decir, fertilizante, insecticidas, lluvia y semillas, hace que éstos interactúen de manera complicada para producir un efecto o elemento de salida. En este sistema, por supuesto, la salida consiste en el crecimiento de la planta.

Análisis de Efecto y Retraso

En el sistema agrícola, la entrada de causas no dará lugar a un efecto de salida simultáneo, es decir, al crecimiento inmediato de la planta. Al intervalo transcurrido después de suministrar la entrada, que durará hasta que la planta haya crecido y estén listas para la cosecha, se le llama fenómeno de efecto y retraso.

Análisis Determinista-Probabilista

El ejemplo clásico de un sistema determinista sencillo lo tenemos en un sistema de calefacción. En la mayoría de estos sistemas, la sección de poner a funcionar la calefacción o cerrarla, está bien definida y es cosa de rutina. Cuando la temperatura del local ha alcanzado el nivel indicado por la posición del termostato, está cierra la calefacción. El termostato actúa a la vez como medidor a la salida y como controlador a la entrada. Sus contactos cierran un circuito que activa la fuente productora de calor. El calor producido por este último es retroalimentado al termostato a través del aire, el cual, al calentar el elemento metálico de aquel, abre nuevamente los contactos. El aire calentado enlaza al productor de calor y al termostato en un circuito cerrado de retroalimentación, relacionando la acción de uno y de otro en una cadena sin fin de causa y efecto. En este sistema, la idea fundamental es obtener el resultado de salida por autorregulación, la cual se logra haciendo que el receptor y el medidor de la salida se también el que suministre los elementos de entrada.

Un sistema probabilista complejo tiene muchas entradas, las cuales interactúan para producir muchas salidas. El fenómeno de entrada-salida (relación causa y efecto) encaja en tres categorías: 1) cuando se sabe con certeza que, si se suministran ciertas entradas, se obtendrán ciertas salidas. 2) cuando existe incertidumbre con respecto a cuáles elementos de entrada deben suministrarse, y en qué cantidad, con el fin de obtener determinados resultados de salida, y 3) cuando algunos de los elementos de entrada y salida son totalmente desconocidos o incontrolables. Estas tres categorías existen en todo sistema probabilista complejo.

Nótese que, en el sistema comercial, la utilidad (en el caso real se obtendrán muchas salidas) no se conoce con certeza, sino que está representada con cierto nivel de probabilidad. Si los administradores del sistema supieran con certeza la combinación, número y cantidad de cada uno de los elementos indicados en el ejemplo, y si tuvieran el control sobre otros factores que no aparecen, como huelgas y desastres naturales, podría conocerse con certeza el resultado. Dicho de otro modo, la probabilidad de conocer el importe de la utilidad sería igual a uno. Pero ese no es el caso, de manera que el nivel de probabilidad será simple inferior a uno. Hay que hacer hincapié en que el resultado de la información es disminuir la incertidumbre y, al hacerlo así, puede aumentar el nivel de probabilidad aproximándolo a uno.

Una campaña política es otro ejemplo del sistema probabilístico inherente en algunos sistemas complejos.

En el sistema probabilístico complejo, la salida consiste en ganar las elecciones. Pero hay algo que es enteramente cierto: nadie sabe con certeza cuál será el resultado final. Sólo los eruditos pueden dar una probabilidad. Hasta el político y sus asistentes están indecisos con respecto a las entradas que deben suministrar, y en qué momento, para elevar las probabilidades de ganar.

Análisis de Optimización

En todo sistema se presenta la posibilidad de lograr una combinación óptima de elementos de entrada, que pueda producir los resultados deseados optimizando al mismo tiempo todo el sistema. Por ejemplo, supongamos que se tiene el proyecto de pintar un tanque para agua y nosotros tenemos la responsabilidad de elegir el número de pintores. ¿Lo harán mejor cuatro hombres que dos? ¿Será mejor ocho y no cuatro? ¿O doce en vez de ocho? Parecería que, mientras mayor sea el número de pintores elegido para el proyecto, mayor será su eficiencia; pero generalmente no es así. En la figura 1.3.6 se indica el número óptimo de pintores necesario para llevar a cabo el proyecto en el tiempo más breve.

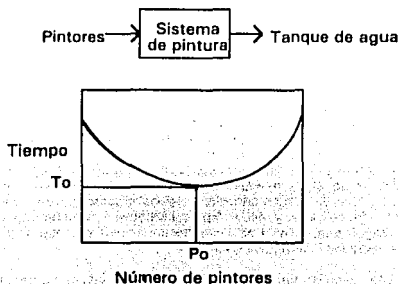


Figura 1.3.6 Optimización del sistema de un trabajo de pintura

Obsérvese que el tiempo necesario para terminar el proyecto disminuye a medida que se agregan más pintores; pero al llegar a cierto punto (cuando la inclinación de la curva es de cero), el tiempo irá en aumento si se agregan más pintores. Se ha rebasado el punto de utilidad decreciente. En consecuencia, el punto ToPo representa el número óptimo de pintores para el proyecto. En la vida real es posible observar numerosos ejemplos que señalen claramente la aplicabilidad de este análisis. En la selección y adquisición de instrumentos para el procesamiento de datos, vamos a encontrar con frecuencia con la creencia de que el hecho de tener cada vez más equipo logrará, de algún modo mágico, hacer que el sistema de información responda a las necesidades de la administración. A menudo, lo que ocurre debido a tal creencia es que los administradores se ven inundados por montones de resultados (contaminación de datos) que presentan una notoria falta de contenido.

El Fenómeno de entrada-salida en los Subsistemas

En los sistemas complejos, sean deterministas o probabilistas, encontramos normalmente cierto número de subsistemas que componen el sistema total. En esos casos, la salida de un subsistema constituye la entrada del que le sigue y así sucesivamente. En la figura 1.3.7, un sistema simplificado de producción de petróleo ilustra el fenómeno de entrada-salida en los subsistemas.

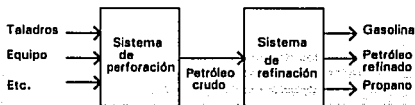


Figura 1.3.7 Secuencia de entrada-salida, mostrando dos subsistemas en un sistema de producción petrolera.

Este análisis puede relacionarse con un sistema de información donde los datos correspondientes a los resultados de las operaciones y transacciones de una empresa vienen a ser los elementos de entrada del sistema de información. La información, o sea el producto del sistema, pasa a ser a su vez el elemento de entrada para el sistema administrativo. Este último hace planes, toma decisiones y resuelve conflictos y problemas, tomando acción correctiva para el control y manipulación de los diversos elementos de entrada (dinero, máquinas, etc.) suministrando al sistema general que es la empresa. estos suministros producen así en datos para el sistema de información. El ciclo se repite una y otra vez.

1.3.2 Exposición General de Administración

¿Cuál es la función de la administración y en qué forma la desempeña? Algunas autoridades afirman que la tarea fundamental de la administración consiste en hacer frente a las condiciones cambiantes. Otros sostienen que esa tarea consiste en reconocer y asimilar los cambios tecnológicos, de manera que se produzcan valores prácticos que serán entregados a la sociedad en forma ordenada, oportuna y económica. Otros más pretenden que la administración consiste en "lograr que las cosas se hagan mediante el esfuerzo de los demás".

Un examen de lo que se ha dicho al respecto releva una multiplicidad de tareas asociadas con la función administrativa. Entre ellas figuran las siguientes: planeación, programación, dirección, organización, contratación, entrenamiento, control, supervisión, etc. Tradicionalmente, la función administrativa se ha identificado con la planeación, la organización y el control. Nosotros vamos a dividir esa función en tres tareas fundamentales: 1) planeación, 2) control y 3) toma de decisiones. Si los analistas de sistemas tienen que diseñar sistemas efectivos de información que sean útiles para el administrador, es obvio que deben entender no sólo la función administrativa, sino también sus necesidades de información.

En los sistemas, la administración puede tomar decisiones, establecer planes; pero, como ya se dijo, no tiene control sobre todas esas entradas. Por otra parte, puede encontrarse presente una buena dosis de incertidumbre con respecto a aquellos sobre los cuales ejerce control.

La Planeación.

Es el momento de señalar que todos los sistemas persiguen un objetivo. La planeación establece, con anterioridad que los administradores deben satisfacer para alcanzar las metas señaladas. La planeación es básica para el control y toma de decisiones, aunque podría aducirse que es necesario decidirse primero por un plan. Si la planeación no haría falta el control, puesto que en aquella es el fundamento de la administración y el control, así como de otras actividades, como la organización y la programación. La administración necesita información para fijar objetivos y cursos de acción. Es imposible proporcionar toda la información necesaria para abarcar todos los aspectos y contingencias de un plan; pero la que se aporta ayuda a disminuir la incertidumbre, amenaza constantemente al administrador.

Analizando más afondo la planeación encontramos cinco tareas básicas que debe llevar a cabo el que planea:

- 1) Establecer metas y objetivos.
- 2) Identificar los eventos y actividades que deben realizarse para alcanzar los objetivos.
- 3) Describir los recursos y la habilidad requeridos para realizar cada actividad.
- 4) Definir la duración de cada actividad señalada.
- 5) Determinar la secuencia, si la hay, en que deben llevarse a cabo las actividades.

De esta descripción de la planeación se deduce que es una condición permanente de la administración, si no en todas, sí en la mayoría de las organizaciones. Además, en casi todas ellas, la tarea de hacer planes la comparten en grados diversos todos los niveles de la administración.

El Control

Las cosas, en verdad, rara vez marchan de acuerdo con los planes. Las desviaciones se deben a factores que están más allá del control de los administradores; por ejemplo, errores de estimación y equivocaciones cometidas por los subordinados. En la mayor parte de los sistemas ocurre siempre que algo marcha mal, por lo cual se imponen una actividad de control. El control es un proceso que consiste en tres etapas básicas: 1) evaluar lo que produce el sistema, 2) comparar el resultado con lo planeado para encontrar las desviaciones y 3) corregir las desviaciones desfavorables poniendo el remedio que sea necesario. Subrayemos aquí que, para que el administrador pueda emprender una acción correctiva, se supone que tiene el poder de modificar o sustituir algún recurso; suposición muy importante.

Como se dijo antes, la planeación establece los objetivos, y el control garantiza el logro de esos objetivos a través de una retroalimentación de informes a la administración. En consecuencia, en esta sección, se analiza las actividades de control en los sistemas.

Elementos de retroalimentación y control. Si una persona permanece sentada por largo tiempo con las piernas cruzadas, al levantarse descubre, que le es difícil sostenerse de acuerdo con lo planeado. Lo que ocurre es que la retroalimentación de los sensores situados en la base de los pies ha desaparecido prácticamente. En consecuencia, no puede ejercer control para adoptar la posición planeada.

Al lanzar una pelota de fútbol, los zagueros confían en un sistema de retroalimentación que evalúa continuamente sus errores de posición, la rapidez y posición de sus receptores, las posiciones y velocidad de los jugadores de línea y el nivel de protección. Para controlar debidamente el lanzamiento, deben reducir el error a cero, a casi a cero.

Los elementos del sistema que han sido enumerados son la entrada, la salida y el proceso. Estos elementos representan los parámetros del sistema, una parte de los cuales se puede medir y evaluar. Para medir la efectividad general que posee el sistema para lograr sus fines, acudimos al parámetro de salida, que representa el punto central de control.

Para usar este parámetro se disponen tres elementos de control.

- 1) Establecimiento de un estándar para el resultado, es decir, el rendimiento deseado u objetivo del sistema.
- 2) Diseño e implementación de un elemento censor que recoja información relacionada con el resultado, haga una medición del rendimiento y comunique los resultados a la administración.
- 3) Empleo de uno o varios supervisores o de un mecanismo, que tomen acción correctiva si la información obtenida así lo señala. Dicha acción correctiva da lugar a la liberación de una decisión, que actúa como elementos de entrada para el sistema. En la figura 1.3.8 se da un ejemplo sencillo de control.

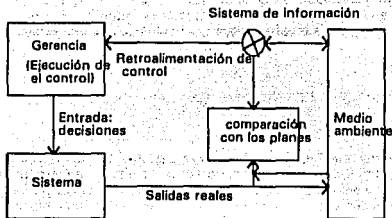


Figura 1.3.8 Elementos que componen la retroalimentación y el control.

El sistema de Información mide y evalúa los resultados del proceso. Los resultados pueden consistir en las ventas, el volumen de ventas, las utilidades, el control de calidad, las compras, el nivel de producción, el comportamiento del personal, el flujo de los recursos, la utilización de los activos, etc. Estos resultados enmanan del proceso del sistema.

Si se trata de una organización comercial (o de cualquier otra clase), el sistema de información debe entregarla a los usuarios externos, como son los accionistas o el Gobierno, pueden influir en alguna forma en las decisiones tomadas por el grupo administrativo interno. El sistema de información del exterior, acerca del nivel de la demanda, el precio de los materiales, a disponibilidad de mano de obra, el precio de la misma, la actitud de la competencia, los actos del Gobierno, la respuesta de los clientes, etc.

Para ejercer control, quien toma decisiones se basa en una salida predeterminada que representa un valor planeado o deseado. Las desviaciones con respecto a este valor le son comunicadas a quien toma decisiones por medio de información, que contiene un control de error E . Este valor es la diferencia entre la salida deseada DO y la salida real AO , que da lugar a una desviación igual a $E = DO - AO$. La corrección de esta diferencia es tarea de quien toma las decisiones. Sus medidas correctivas generan los elementos de entrada que va a activar el proceso. Si se tiene éxito, $DO - AO = 0$. Por ejemplo, si los resultados del sistema (cantidad de productos, calidad de los mismos, ventas, utilidades, etc.) no son los que debieran ser de acuerdo con los planes previamente establecidos, la información que indica las desviaciones se retroalimenta a los administradores a quienes corresponde corregirlas o modificar los planes.

El control de un sistema será efectivo: 1) si los resultados reales son debidamente medidos y comparados con los deseados, 2) si la acción necesaria es puesta en ejecución y 3) si tanto la retroalimentación de informes como la toma de decisiones son lo suficientemente rápidos como para efectuar correcciones antes de que los factores que intervienen en el proceso se vuelvan inconsistentes con las mismas.

Tipos de control. hay dos tipos de sistemas de control: 1) de circuito abierto y 2) de circuito cerrado.

1) Sistema de circuito abierto. Este sistema se establece para lograr un determinado propósito y no hay retroalimentación de información. Por ejemplo, una máquina lavadora puede ser gobernada por un control de tiempo, de manera que realice un ciclo de lavado de duración determinada. Podríamos poner la máquina en marcha, dejarla y volver más tarde, sólo para encontrarla con

que el ciclo de lavado se llevó a cabo sin agua porque la llave no fue abierta. Si el dispositivo de control hubiera sido informado de la situación mediante un sistema de retroalimentación, se habrían tomado las medidas necesarias para corregir la falla. Podemos pensar igualmente en instituciones que no operan con eficiencia ya sea por causa de un sistema deficiente de retroalimentación, o debido a que el mecanismo de control no toma en cuenta la información disponible.

A una escala mayor y más compleja, los sistemas comerciales se manejan basándose en un circuito abierto pueden llegar a situaciones críticas cuyas solución esta más allá de las posibilidades de la administración. Por ejemplo, la producción de artículos alimenticios se debe realizar con todo cuidado a fin de detectar la presencia de contaminantes. La información obtenida mediante este control se debe comunicar a los administradores a quienes corresponde corregir cualquier desviación. Sin esta retroalimentación de información, la empresa corre el riesgo de un desastre imposible de corregir. A escala más amplia, algunos historiadores han señalado que las consecuencias del ataque a Pearl Harbort se habrían podido mitigar si las autoridades hubieran recibido información necesaria.

2) Sistema de circuito cerrado. Este tipo de sistema es aquél en que el mecanismo de control recibe y utiliza la información retroalimentada. En nuestro ejemplo de la máquina de lavar, si el mecanismo de control ha sido ajustado de manera que el ciclo de lavado solamente pueda comenzar después de que una determinada cantidad de agua haya entrado al recipiente, el lavado se efectuará sólo si dicho volumen de agua se encuentra presente. El mecanismo de control es advertido por el circuito de retroalimentación.

Todas nuestra instituciones y organizaciones deberían contar con un flujo de información a través de un sistema de circuito cerrado. La razón por la cuál se necesita tal sistema se evidencia cuando consideramos que todo sistema, si ha de alcanzar sus objetivos, debe entregar periódicamente al sistema administrativo la información que le ponga al tanto del nivel de los logros. A si, podemos afirmar en general que todo sistema que se proponga a alcanzar sus metas debe utilizar el sistema de circuito cerrado para ejercer el debido control.

La Toma de Decisiones

Una de las actividades básicas de la administración consiste en conpenetrarse de las características del sistema hasta un grado que le permita juzgar el rendimiento con toda propiedad, mejorándolo dentro de las restricciones establecidas. La toma de decisiones, pues, es el proceso de elegir entre varias alternativas, que puede ser cuantitativas o cualitativas, aquella que sea la mejor para resolver un problema o arreglar un conflicto.

Elementos de la toma de decisiones. La causa por la que un administrador debe tomar una decisión es la confrontación de un problema o la presencia de una situación conflictiva. El hecho de decidir resuelve el problema o arregla el conflicto. Un proceso ordenado para llegar a una decisión contiene cuatro elementos:

- 1) Un moldeo. El modelo es una descripción cuantitativa o cualitativa del problema.
- 2) Criterios. Los criterios establecidos representan las metas u objetivos del problema de decisión (por ejemplo, dar el mejor servicio a los clientes). Cuando varios criterios entran en conflicto (

por ejemplo, mejorar el servicio y reducir el inventario), quién toma la decisión debe elegir un término medio.

3) Restricciones. Son varios los factores que se deben considerar cuando se trata de resolver el problema de decisión. La falta de recursos es una restricción.

4) Optimización. Una vez planteado claramente el problema (el modelo), el administrador determina (los criterios) y lo que puede hacerse (las restricciones). En ese punto, quien toma la decisión está listo para elegir la mejor solución, es decir, la óptima.

Clases de decisión. Los problemas de decisión y los conflictos aparecen por todas partes. Algunos son sencillos y deterministas, con pocas ramificaciones; pero otros son bastante complejos y probabilistas y sus efectos pueden ser considerables. La toma de decisiones puede ser rutinaria y bien estructurada, o compleja y mal estructurada. De manera que, en términos generales, hay dos clases de decisiones: 1) programadas y 2) no programadas.

1) Decisión programada. Esta clase de decisión implica una respuesta automática de acuerdo con las políticas previamente establecidas. Todos los problemas repetitivos y rutinarios, con parámetros bien definidos, se presentan a la decisión programada. La identificación de esta clase de decisiones y la aportación de métodos mediante los cuales puedan ser aportadas siempre que sea posible, son un reto para el analista. Para tomar esta clase de decisiones es preciso haber establecido y definido claramente una regla de decisión. Una vez establecida esta regla, todo se reduce a desarrollar un programa que permita llegar a la decisión de modo rutinario y automático.

En muchas empresas se presenta la oportunidad de adoptar la decisión programada, porque gran parte de las decisiones se toma de acuerdo con procedimientos normalizados y rutinarios de operación. El beneficio de la decisión programada consiste en que los administradores quedan libres para enfocar su atención en otras tareas más importantes.

Un ejemplo de decisión programada sería el control de inventarios, cuando la determinación del lote económico, el punto de re abastecimiento y las existencias de seguridad son manejados por el sistema de computación conforme a las necesidades. Cuando las existencias bajan hasta un nivel preestablecido, se inicia automáticamente la solicitud de un número N de artículos para re abastecer los almacenes.

2) Decisión no programada. Esta clase de decisión representa el proceso de afrontar problemas poco definidos. Generalmente dichos problemas son complejos y sólo se conoce una parte de sus parámetros, y ocurre que muchos de los parámetros conocidos presentan un alto grado de probabilidad. La toma correcta de decisiones no programadas requiere de todo el talento del administrador experimentado, más el auxilio del sistema de información. La ampliación de la planta, el desarrollo de un nuevo producto, las políticas de procesamiento y publicidad, el manejo de personal, la adquisición contra el arrendamiento, las fusiones, etc., son problemas que exigen la decisión no programada. La sección siguiente analiza a quienes toman decisiones y la función que desempeñan en esta clase de decisión.

El proceso de la toma de decisiones. Las personas que toman decisiones poseen en diversos grados la facultad de asimilar la información, lo cual determina la eficiencia de su procesamiento. Los conocimientos de una persona, unidos a su eficiencia para procesar la información, determinan su capacidad de decisión. Frente a varias alternativas, elige un objeto y trata de alcanzarlo escogiendo, con base en lo que sabe, la mejor alternativa. Si reconoce que sus conocimientos no le permiten comprender el significado de cada alternativa, busca información adicional. Si trata de actuar sin poseer información suficiente, surgirán nuevos problemas y tendrá que redoblar sus esfuerzos. La falta de información proviene de la incapacidad de las fuentes para proporcionar toda la que hace falta, o de la incapacidad de quién toma las decisiones para determinar exactamente sus necesidades. en la figura 1.3.10 se presenta una ilustración del proceso de toma de decisiones.

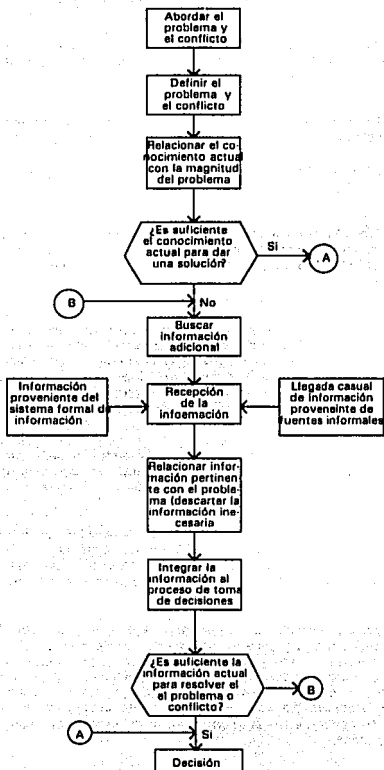


Figura 1.3.10 Flujo de información en el proceso de toma de decisiones.

Los directores de empresas, funcionarios del gobierno, administradores de colegios y muchos otros intervienen mucho tiempo en la solución de problemas y en el arreglo de conflictos. Se pretende que, en buena medida, su éxito en esas actividades estará relacionado directamente con la calidad de información que utilizan.

Nuestra idea de la toma de decisiones es que se trata de un proceso de utilización de informes, no de un proceso emocional. Por tanto, en este contexto, las dificultades que se encuentran al tomar decisiones pueden imputarse a cualquiera de los factores siguientes:

1) Información inadecuada: Es decir, información incorrecta o incompleta con respecto a las diferentes alternativas de cursos de acción o con respecto a sus implicaciones a los resultados.

2) Objetivos incorrectamente especificados; o sea, que no se ha indicado claramente cuáles resultados son los más deseables.

Niveles de la toma de decisiones. Las decisiones pueden ser desde muy superficiales y rutinarias (programadas) hasta muy complejas, o sea aquellas que producen un efecto significativo en el sistema (no programadas). Para su clasificación, situamos la toma de decisiones en tres niveles: a) estratégico, b) táctico y c) técnico.

a) Nivel estratégico. Las decisiones estratégicas se caracterizan por un alto grado de incertidumbre y están orientadas hacia el futuro. Estas decisiones establecen planes a largo plazo que influyen en toda la organización. Se fijan las metas de la empresa y se crea una serie de estrategias que pueden comprender, por ejemplo, la expansión de la planta, las fusiones, la diversificación de actividades, los desembolsos de capital o la venta del negocio. La estrategia, pues, se ocupa de los planes de largo alcance e incluye el establecimiento de objetivos y políticas, la organización, y el logro de la efectividad general.

b) Nivel táctico. Las decisiones tácticas se refieren a las actividades a corto plazo y a la distribución de los recursos para el logro de los objetivos. Esta clase de decisiones comprenden de las áreas de Formulación de presupuestos, análisis de flujo de recursos, distribución de la planta, problemas del personal, mejoramiento del producto, investigación y desarrollo.

Mientras que la toma de decisiones estratégicas implica más que una actividad de planeación, las decisiones tácticas exigen una suma casi igual a las actividades de planeación y control y su potencial para decisiones programadas es escaso, si es que existe. Para la toma de decisiones tácticas, la mayoría de las reglas de decisión está mal estructurada y no son aceptables a la rutina y a la autorregulación.

c) Nivel técnico. A este nivel de decisión, los estándares son fijos y los resultados son deterministas. La toma de decisiones técnicas es un proceso para asegurarse de que la ejecución de las tareas específicas se lleva a cabo de manera efectiva y eficiente. Esta clase de decisiones exige que se impartan órdenes específicas para controlar operaciones igualmente específicas. La función principal administrativa a la que corresponde este tipo de decisiones es la de control; la planeación figura en una escala más bien limitada. Tenemos ejemplos de esas decisiones en la aceptación o rechazo de solicitudes de crédito, en el proceso de control, en la programación, la recepción, remesas, control de inventarios y distribución de los trabajadores.

Las líneas de separación no pueden establecerse ya que en la práctica, los límites de las categorías son inciertos y tienden a traslaparse. No obstante los analistas deben estar al tanto de esta clase de decisiones y conocer la forma de diseñar el sistema de información con el fin de satisfacer las

diversas necesidades, puesto que la información generada por el sistema dependerá de esas necesidades.

1.3.3 Revisión de los Sistemas de Información

Todas las organizaciones cuentan con una u otra clase de sistema de información que, según se supone, satisface las necesidades informales y disminuye la probabilidad de que lleguen a adoptarse decisiones incorrectas. Sin embargo, muchos sistemas no pueden proporcionar información adecuada para tomar decisiones estratégicas y, hasta cierto punto, decisiones tácticas. Para la toma de decisiones estratégicas es imperativo diseñar sistemas de información que puedan captar las realidades exteriores, es decir, la actitud de los competidores, las tendencias económicas, sociales y políticas, la situación en los países extranjeros, los avances tecnológicos, etc., y de comunicar esta información a personas a quienes corresponde tomar esas decisiones.

El término "formal" se usa para describir el sistema de información que se estudia en este capítulo. Significa que nos interesa sólo la información obtenida a partir de datos objetivos y comprobables. Dicho de otro modo, nuestro sistema de información no incluye los rumores, las vías secretas, las habladurías ni las comunicaciones personales de carácter informal.

Efectividad del sistema de información para los diferentes niveles de toma de decisiones

Tomando en cuenta la diversidad de necesidades, el sistema de información debe diseñarse de manera que satisfaga a los tres niveles de toma de decisiones. En numerosas empresas, muchas decisiones estratégicas y tácticas se toman guiándose por la intuición, la heurística y la interpretación, más que por la información pertinente proporcionada por el sistema formal. La implantación de un tipo estratégico de información, procesada por el sistema, probará su utilidad para quienes deben tomar esta clase de decisiones. No obstante, un sistema formal de información tienen sus limitaciones en cuanto a su efectividad con que se puede producir información apropiada para las tres clases de toma de decisiones.

En cada uno de los esquemas se representan conceptos generales sujetos a modificación. El propósito de las ilustraciones es demostrar no sólo la efectividad potencial del sistema formal de información, sino también sus probables limitaciones. Los libros están repletos de promesas grandiosas con respecto a lo que un sistema de información puede hacer en beneficio de la administración; pero muchas de esas promesas van mucho más allá de las posibilidades actuales del arte de la información.

Al nivel estratégico de la toma de decisiones, la variedad de los problemas llega a su máximo en cualquier sistema. A menudo, esos problemas son de naturaleza única y son muy importante, y el administrador debe enfocarlos en condiciones de incertidumbre casi total. Por tanto, la creencia de que es posible desarrollar sistemas de información absolutos, que pueden solucionar cualquier problema, es sencillamente impracticable. Se requiere mucho arte, conocimientos y experiencia para tomar decisiones a este nivel; pero los sistemas de información bien diseñados pueden ayudar a reducir la variedad y la incertidumbre. En el siguiente subtema se estudiarán los métodos para lograr este objetivo.

Puesto que las decisiones tácticas se ocupan principalmente de determinar la utilización más eficiente de los recursos, los sistemas formales de información tienen a este nivel verdadera oportunidad de auxiliar a la administración de técnicas de modelado para seleccionar, simplificar, diagnosticar y

optimar. A este nivel, hay muy poca oportunidad, si es que la hay, de volver rutinario cualquier proceso de decisión. Las habilidades administrativas clásicas desempeñan todavía una importante función.

La toma de decisiones técnicas es algo bien definido, rutinario y determinista. Gran parte de la "información" destinada a este nivel proviene de las actividades administrativas normales de procesamiento de datos con que se cuenta todo sistema para hacer frente a necesidades comunes como son la Formulación de nóminas y la redacción de informes a los accionistas y al Gobierno. Por ejemplo, cuando una empresa manufacturera o vendedora recibe el pedido de un cliente, dicho pedido inicia una serie de eventos que comprende el procesamiento de datos, elaboración de informes y procesamiento físico de materiales. Por otra parte, en las empresas siempre es necesario formular nóminas para el pago de los empleados, llevar registros de ventas, elaborar informes financieros, etc. Si a este nivel se sigue un enfoque de sistemas para desarrollar el sistema de información, muchos de los datos generados pueden servir para producir información destinada a niveles más altos, cuando sea necesario. Por ejemplo, si se establece un patrón lógico de clasificación y codificación, los datos básicos sobre ventas pueden servir para generar información útil para el pronóstico de ventas, el control inventarios y el análisis de ventas (ya sea por producto, por cliente, por vendedor o por territorio). Los datos básicos se pueden asociar de manera lógica para obtener información útil para los administradores de todos los niveles.

De lo anterior no se debe llegar a la conclusión de que este nivel de procesamiento carece de importancia. Asimismo, un sistema que lleva a cabo un procesamiento de datos confiable desde el punto de vista administrativo es absolutamente esencial para manejar sin tropiezos cualquier empresa. Si los métodos de procesamiento fallan o tienen defectos a este nivel, toda la organización se enfrentará de inmediato a una crisis.

A este nivel existe un gran potencial para la toma de decisiones programadas, lo cual permite que los administradores dedique más tiempo a los otros dos niveles de decisiones creativas.

Sin duda, las necesidades de información se pueden relacionar con el proceso de toma de decisiones. Para cada actividad de decisión se debe determinar la información que requiere el administrador responsable, permitiéndole llevar a cabo sus actividades de planeación, control, organización, asignación, etc. El analista y el usuario deben considerar conjuntamente el alcance de la información, el grado del detalle, el contenido de los reportes y su frecuencia, el periodo que debe alcanzarse y los métodos de distribución y comunicación.

El usuario responsable de la toma de las decisiones que se analizan, será quien utilice principalmente la información, y debe tener bastante libertad para determinar sus necesidades y las condiciones de procesamiento y comunicación. Debido a que el sistema de información se diseña de acuerdo con las necesidades, es importante que cada usuario (administrador) se resista a la tendencia natural a solicitar más información que la que necesita y a que le sea presentada con mayor frecuencia y en forma más elaborada que lo que justifica la relación costo/efectividad.

Por ejemplo, la información en línea, que se requiere para la toma de decisiones a nivel estratégico por lo común es moderada, y acaso no se necesite. Los analistas de la empresa podrán instalar unidades audiovisuales y pantallas de despliegue visual en las oficinas de los ejecutivos, combinados con los intercomunicadores auditivos comunes. Para obtener la información que necesita, el funcionario sólo tiene que marcar un teléfono y teclear un código de acceso basado en un índice de la información disponible. dicho índice podría contener datos tales como las ventas afectadas hasta la fecha, comparadas con las planeadas, las ventas del año anterior, los ingresos a la fecha (después de impuestos) de la División A, etc.

Toda esta información es vital, pero para tomar decisiones estratégicas de consecuencia, a menudo se necesita pensar durante días, semanas y hasta meses. Las decisiones instantáneas con respecto a problemas complejos no da tiempo a que el administrador ponga a juego su propio juicio. Por otra parte, aún cuando la información en línea, por ejemplo, los viajes programados para los ejecutivos principales, los informes de la bolsa de valores, las noticias políticas, los tipos de cambio de moneda

extranjera, etc., son interesantes y necesarios, el costo de su obtención por el sistema en línea podrían exceder en mucho a la efectividad del sistema. Por lo demás, la misma información puede obtenerse en otras fuentes menos elaboradas y menos costosas.

Análisis del Proceso de Comunicación.

El concepto básico que comprende el flujo de la información y su entrega a la administración es la comunicación. Esta se efectúa cuando la información se transmite de la fuente al receptor a través de un canal definido que las une.

La comunicación efectiva es muy importante para todos los aspectos del esfuerzo humano. Existe una amplia gama de sistemas de comunicación, que van desde el contacto simple y directo entre persona y persona hasta los sistemas más complejos como son las redes de comunicación relacionadas con el sistema formal de información de la empresa, cuya finalidad es proporcionar al receptor informes objetivos, imparciales y formales.

Los objetivos generales de un sistema efectivo de información son los siguientes: 1) transmitir la información y 2) asegurarse de que se reciba la información correctamente por el usuario. La información los mensajes transmitidos al receptor a través de un canal. En el modelo también se ha incluido una fuente de ruido que interfiere, a cierto nivel, con la información que fluye de la fuente al receptor. El ruido se manifiesta a veces en forma de reportes inexactos y parciales. En un sistema grande y complejo que tiene una serie de emisores y receptores, el ruido puede producirse en los empalmes. El ruido se acumula cuando el ruido que produce un traspase se suma al que produce otro.

En el sistema de comunicación no podrá ser más efectivo que el menos efectivo de sus componentes: una cadena no podrá ser más fuerte que el más débil de sus eslabones. Si la fuente no proporciona información apropiada y oportuna, si los mensajes son distorsionados o demorados, si el receptor no puede hacer uso de los mensajes, entonces todo el sistema será degradado al más bajo denominador común. Para que el sistema sea efectivo, todas sus etapas deberán realizarse con igual efectividad.

Categorías de la comunicación. La comunicación, como la entendemos en esta sección, se divide en dos categorías: 1) comunicación personal y 2) medios de comunicación. Las características principales de la comunicación personal son las siguientes: 1) una persona específica comunicándose con otra, 2) una persona comunicándose con un grupo, o 3) un grupo comunicándose con otro grupo.

Los medios de comunicación representan formas impersonales de transmitir información al receptor. Entre ellos figuran los siguientes: libros, periódicos, televisión, radio, terminales remotas y diversos informes elaborados. Las terminales remotas están conectadas a la UPC mediante dispositivos de telecomunicación tales como teléfonos, microondas o satélites de comunicación. En algunos casos, la combinación de computadoras y equipo de comunicación puede ser un componente principal de los sistemas de información. En estos sistemas, la transmisión de datos e información se lleva a cabo de las formas siguientes: 1) tiempo compartido, 2) teleproceso, 3) transmisión de computadora a computadora y 4) consulta a distancia. La telecomunicación facilita las interacciones siguientes: hombre a máquina, para trabajo extensivo de computación hecho desde una localidad remota; dispositivos de almacenamiento a máquina, para transmisión a distancia de voluminosos archivos de datos; máquina, para sistemas vía satélite, máquina/terminal o de protección, y hombre a máquina para procesos interactivos de actualización a control remoto de archivos y rápido acceso a la información contenida en los archivos. El desarrollo de los medios de comunicación nos ha dado instrumentos y técnicas poderosas que hacen que la computadora una herramienta altamente versátil. Además de procesar gran cantidad de datos y formular reportes informativos, es también un dispositivo de interacción efectivo. Al reducir el tiempo y el espacio, la tecnología de la comunicación hace que la información se más

pertinente y concisa, puesto que puede dar respuestas específicas a preguntas concretas y reducir las demoras en la información.

Ciclo de comunicación en los sistemas de información. Un sistema de información existen muchas relaciones de comunicación. Al llevarse a cabo el desarrollo del sistema, punto que será tratará más adelante en forma detallada, la comunicación adecuada entre sus diversos componentes tiene importancia fundamental.

La falta de una comunicación efectiva constituye realmente una barrera para la investigación, el análisis, la definición y la solución de los problemas. Si el analista fuera el único encargado del trabajo de sistemas, la única limitación a la efectividad de su labor radicaría en su grado de competencia; pero, obviamente no es el único. Debe, pues establecer un ciclo completo de comunicación que comience y termine con el usuario, es decir, el receptor último de la información.

Modelo General del Sistema Formal

Aunque se estudiarán muchos aspectos del sistema formal de información, es oportuno reunir en un modelo conceptual muchos de los puntos que se han estudiado hasta aquí.

Como ya se dijo, el sistema formal de información es un sistema necesario para el manejo efectivo de un sistema total, representado por la empresa, dependencia gubernamental o institución de que se trate.

El resultado del sistema de población es, por supuesto, el número de personas. La cantidad y ubicación de esas personas, a su vez, influyen en la disposición de desperdicios, la contaminación y la conservación que son el resultado del sistema ecológico. Es necesario obtener datos de estos sistemas, procesarlos y comunicar los resultados al sistema administrativo (es decir, a los legisladores y funcionarios del Gobierno). Se tiene que en algunos de los elementos de entrada no están conectados con el sistema administrativo, por encontrarse fuera de su control. Sin embargo, adviértase que los efectos de uno de esos elementos, las enfermedades, se pueden neutralizar mediante la implantación de un programa efectivo de sanidad. Nótese además que un elemento controlable, el cuidado de la salud, ayuda a incrementar el crecimiento demográfico, mientras que otro, la educación en materia de control de la natalidad, ayuda a limitar ese crecimiento. Esta situación, que resulta paradójica, destaca las complejidades inherentes en la aplicación de métodos administrativos apropiados.

En estos dos modelos idealizados, los sistemas de información realizan tareas siguientes: 1) procesar todos los datos, rutinarios y administrativos; 2) tomar decisiones programadas cuando se presente el caso, y 3) proporcionar la información necesaria a todos los niveles del sistema administrativo para la planeación, control y toma de decisiones.

Como vemos luego, hay varias maneras de enfocar el sistema de información; por ejemplo, 1) según sus objetivos, 2) de acuerdo con los métodos específicos aplicados para satisfacer las necesidades de la administración, 3) conforme a su estructura dentro del sistema general, 4) de acuerdo con la tecnología que los hace funcional, 5) según el conjunto de datos que constituye su base y 6) atendiendo a las formas en que fluye la información como un subsistema que procesa datos y proporciona información a la administración y a otros grupos, como son los accionistas de la empresa. También debemos reconocer el hecho de que el sistema de información es la clave de la dirección efectiva de todo sistema organizativo, comercial o de otra clase.

1.4. Análisis de los Conceptos Básicos de Datos e Información

Hasta aquí nos hemos ocupado de diversos conceptos y términos que constituyen una introducción al análisis de los sistemas de información. En los temas anteriores se utilizó la expresión sistema de información, pero se hizo principalmente en relación a un área particular en desarrollo y no a la parte específica del sistema. Por ejemplo, en el tema anterior se presentó un panorama del sistema de información, el apoyo que presenta a la función administrativa y su responsabilidad de informar a grupos externos.

Los objetivos principales de esta sección son:

- 1) Definir más claramente el concepto de sistemas de información.
- 2) Examinar el flujo de la información y la disposición del sistema de la organización.
- 3) Proporcionar algunos ejemplos de sistemas de información, que estimularán la búsqueda de otras aplicaciones.

1.4.1 Estudio General de los Sistemas de Información

Las expresiones "sistemas de información a la gerencia", "sistema de información a la gerencia por computadora, y otros, son denominaciones de las obras que se refieren al tema. En los últimos años estas expresiones (y muchas otras) han sido examinadas con atención, pero aún existe gran confusión en cuanto a su verdadero significado y la función que desempeña en las organizaciones. Esta confusión afecta no sólo al personal administrativo, sino también a muchos expertos que, según se supone, implantan los sistemas de información. Se han presupuesto numerosas definiciones con el fin de dar sentido al concepto de sistema de información; pero ninguna ha tenido aceptación general.

Desde que se introdujo el enfoque de sistemas de información a la gerencia, a principios de la década de los setenta, ha provocado muchas reacciones, a favor y en contra. El sistema de información a la gerencia se ha utilizado para implicar toda clase de aplicaciones, desde el procesamiento de datos básicos para pago de sueldos y cobranzas, hasta los sistemas de contabilidad, la elaboración de catálogos para bibliotecas, los sistemas de control de procesos, etc.

O bien el concepto original no fue entendido claramente, o bien se distorsionó a propósito con el fin de aplicar la nueva terminología a sistemas que en realidad no están a la altura del concepto.

Desde el principio, nosotros hemos utilizado la expresión "sistema de información" en vez de sistema de información a la gerencia", "sistema de información por computadora" cualquier otro de significado más restringido.

Las razones para usar "sistema de información" son las siguientes:

- 1) Tiene una connotación más amplia, puesto que implica un cúmulo de información destinada no sólo a la dirección del sistema organizativo, sino también a los grupos externos.
- 2) Es enteramente posible que muchas organizaciones, cuyo volumen de operaciones, complejidad, relaciones de tiempo y necesidades de cálculo sean moderadas, establezcan un buen

sistema de información sin recurrir a la computadora y a la tecnología asociada. Dicho de otro modo, el sistema de información no tiene que basarse necesariamente en el uso de la computadora.

3) Se piensa que los autores de un libro no deben tomar partido, sino llevar a cabo una presentación comprensiva examinando todas las facetas del tema que se trate. Por ejemplo, si el sistema de información a la gerencia denota normalmente un enfoque integrado de la disposición del sistema de información; pero, como demostraremos más adelante, hay otros enfoques. Asimismo, que se quiere que el lector se percate de que los conceptos y técnicas anunciados anteriormente son igualmente aplicables a la creación de sistemas de información para instituciones no lucrativas y organismos gubernamentales (escuelas, ayuntamientos, etc.) como para empresas comerciales.

Definición de los sistemas de información

Todas las organizaciones cuentan con alguna clase de sistema de información, aunque no se trate más que de un archivador y de un pequeño número de cuentas en el libro mayor. Sin embargo, para contar con un sistema funcional de información que satisfaga diversas necesidades, todos los datos medibles deben ser organizados de manera que sea fácil registrarlos, almacenarlos, procesarlos, recuperarlos y comunicarlos según los quieran los usuarios.

Para nuestros fines, un sistema de información se puede definir de este modo: es un conjunto sistemático y formal de componentes, capaz de realizar operaciones de procesamiento de datos con los siguientes propósitos: a) llevar las necesidades de procesamiento de datos correspondientes a los aspectos legales y otros, de las transacciones, b) proporcionar información a los administradores, en apoyo de las actividades de planeación, control y toma de decisiones, y c) producir una gran variedad de informes, según se requiere, para los grupos externos.

Un sistema de información se parece mucho a un sistema de producción que toma las materias primas y las convierte en un producto, que será utilizado por el consumidor o que constituirá a su vez la materia prima para otra fase de conversión. En forma similar, el sistema de información convierte la materia prima (datos) ya sea en un informe utilizable o en un elemento de entrada para el siguiente ciclo de procesamiento.

El sistema de información debe contener, si no todos, por lo menos algunos de los elementos siguientes, que preslan eficacia a los componentes señalados anteriormente:

- 1) Dispositivos de entrada y preparación de datos
- 2) Dispositivos de almacenamiento de datos
- 3) Equipo y medio de telecomunicación
- 4) Equipo de procesamiento de datos
- 5) Dispositivos terminales
- 6) procedimientos, programas, métodos y documentación

7) Modelos de manejo de datos, por ejemplo, modelos de contabilidad y presupuestación, técnicas estándar de costeo, modelo de costo volumen-utilidad, programación lineal, modelos de inventario, técnicas estadísticas, etc.

8) Salas para la toma de decisiones, con pizarrones y gráficas

9) Máquinas duplicadoras

10) Obviamente, analistas de sistemas de información para establecer y utilizar los elementos anteriores.

Objetivos del Sistema de Información

En su forma más elemental, el sistema de información cumple la función de almacén de datos correspondientes a las transacciones y lleva a cabo las operaciones rutinarias de procesamiento concernientes a: 1) registro de pedidos, 2) facturación, 3) cuentas por cobrar, 4) compras, 5) cuentas por pagar, 6) nóminas, 7) informe básico de inventario y 8) libro mayor general. a este nivel fundamental no encontraremos una marcada labor de análisis de sistemas para establecer las necesidades de información de los diversos administradores, sobre todo a los niveles táctico y estratégico. En consecuencia, a menudo dichos administradores no podrán obtener informes vitales para el funcionamiento efectivo de sus áreas.

La información de más alto nivel debe obtenerse como un producto secundario del procesamiento básico de datos, desarrollando modelos y métodos que presenten información adecuada a cada nivel administrativo tomando en cuenta el alcance y naturaleza de la información y el grado que interactúa cada administrador. En otras palabras, la información producida por el sistema debe presentar las diez características que se estudió en el subtema 2 y que referimos aquí como referencia: 1) accesibilidad, 2) Comprensibilidad, 3) precisión, 4) propiedad, 5) oportunidad, 6) claridad, 7) flexibilidad, 8) verificabilidad, 9) imparcialidad y 10) cuantificabilidad. Por tanto, el sistema de información es algo más que un sistema básico de contabilidad y procesamiento de datos. No se debe considerar como un fin en sí mismo, sino pensar que se basa en: 1) el flujo de datos y las operaciones de procesamiento, 2) la determinación de las necesidades de información 3) el flujo de la información y 4) las interacciones administrativas que tienen lugar en toda la organización y que el propio sistema contribuye a apoyar.

En las diferentes organizaciones, el concepto del sistema de información supone que existe una relación necesaria entre el sistema formal de información, la función administrativa, la estructura de la empresa y los usuarios de la información.

Información Formal contra Información Informal

Si todos los administradores de una empresa fueran capaces de guardar en la memoria la información que necesitan, no se necesitarían los sistemas formales de información. Bastaría con un sistema básico de procesamiento de datos para atender al pago de sueldos, la cobranza y otras actividades rutinarias. Pero los administradores de las grandes empresas de nuestros días no pueden guardar en la memoria toda la información que necesitan. Dadas la variedad y complejidad de asuntos a examinar y de problemas a resolver día con día, es imprescindible que reciban una gran cantidad de información formal a través de reportes de rendimiento, estados financieros, análisis de ventas, evaluaciones, alternativas y respuestas emanadas de un sistema diseñado formalmente para este fin.

Puesto que la memoria del hombre es limitada, el sistema de información viene a ser una extensión formal de la misma.

Sin embargo, no toda la información que recibe un administrador proviene del sistema de información. Por ejemplo, puede obtener informes muy valiosos en fuentes ajenas del sistema, llamando por teléfono a un amigo o colega, o conversando con un grupo de personas. Esta clase de información proviene de fuentes no estructuradas específicamente para dar informes al administrador. En consecuencia, todo informe proviene de fuentes ajenas al sistema de información se considera informal. Aquí no se tratará de establecer que clase de información tiene mayor significación. Desde el punto de vista cotidiano, el sistema proporciona toda la información formal necesaria para el manejo de la empresa. No obstante, los datos informales pueden producir el mayor efecto. Es posible que tal información lleve a cambiar el rumbo de la organización.

En primer lugar, es preciso entender que un sistema de computación, en sí mismo, no es un sistema de información. Sin embargo, puede ser un instrumento básico que incremente la efectividad del sistema y proporcione los medios para llevar a cabo ciertas actividades que en otra forma serían imposibles. Es necesario señalar que no todos los sistemas formales de información necesitan de la computadora y su tecnología, que la computadora no necesariamente mejora un sistema y que no todos los reportes que produce una computadora se pueden calificar como información. El hecho de computerizar el sistema actual, o un proceso anterior, no necesariamente acrecentará su efectividad; más bien sucede lo contrario: la instalación de una computadora en un sistema de información erróneamente diseñado, normalmente no hará sino incrementar los errores y fallas existentes en forma excepcional. Lo que debe hacerse primero es llevar a cabo un análisis formal y apropiado del trabajo de sistemas, incorporando la computadora y otros medios semejantes sólo en el caso de que con ello se mejore la efectividad del sistema. Dicho de otro modo, debe pensarse primero en el sistema y después en los dispositivos y equipo necesarios para ponerlo en marcha.

Muchos administradores están desconectados con sus sistemas de información y critican duramente al sistema de computación por la falta de resultados. Son muchas las causas de tal situación; pero pueden mencionarse tres principales: 1) esperanzas exageradas, que no pueden ser satisfechas, 2) falta de un análisis apropiado de sistemas y 3) un síndrome computaritis que se manifiesta en los ejecutivos que adquieren computadoras y otros artefactos electrónicos como una panacea para los problemas administrativos.

Es muy frecuente la tendencia a adquirir costosos conjuntos de luces intermitentes y otros dispositivos, cuando todo lo que se necesita es lograr que la información llegue a los administradores en el momento en que la necesitan. En muchos sistemas, esto puede conseguirse por métodos menos automatizados de procesamiento de datos.

Por tanto, las computadoras y la tecnología con ella asociada pueden ser instrumentos muy complicados para el analista de sistemas. Utilizados adecuadamente y analizados de acuerdo con la relación costo/efectividad, en muchos casos pueden acrecentar en gran medida la efectividad de los sistemas de información, sobre todo cuando el volumen, la complejidad y las demandas de oportunidad y cálculo son elevados. No hay duda de que, con el advenimiento de la computadora, los medios de almacenamiento, las terminales y la telecomunicación, el analista de sistemas tiene la oportunidad de formalizar y sistematizar, hasta un nivel no logrado anteriormente, las actividades de procesamiento de datos e información de las empresas. Por ejemplo, las telecomunicaciones que se han facilitado debido a la multiprogramación, los DAD, y la tecnología de la comunicación de datos, han reducido casi a cero las dimensiones de espacio y tiempo, permitiendo, por tanto, que un funcionario autorizado, ubicado en algún lugar remoto, obtenga información inmediata. En el procesamiento de cantidades masivas de datos han aumentado la precisión y rapidez, debido al mayor refinamiento de un gran número de dispositivos de entrada. La organización y estructuración de grandes cantidades de datos en los bancos de datos, facilita mucho los archivos legibles por computadora. Tomando en cuenta el valor de estas posibilidades, las computadoras y su tecnología se vuelven relativamente menos costosas cada año.

Como ya se dijo anteriormente, hay dos métodos para procesar datos: 1) manual y 2) por computadora, pero la computadora y su tecnología predomina y predominará notablemente en muchos sistemas actuales de información, además de que proporciona un método más sistemático para ilustrar muchos conceptos específicos.

1.4.2 Enfoque Jerárquico para el Diseño de Sistemas de Información

La manera en que la información fluye por la organización depende básicamente de dos cosas: 1) la filosofía de la administración y 2) el enfoque dado al diseño de los sistemas de información. La filosofía de la administración concierne al grado de centralización o descentralización de la autoridad y la responsabilidad. Nuestro análisis no se relaciona específicamente con esta comparación; sólo se trata de una forma subcotidiana. En cuanto a lo segundo, es decir el enfoque dado al diseño y a la disposición del sistema de información, hay básicamente dos enfoques generales: 1) el enfoque jerárquico y 2) el enfoque de sistemas. El enfoque jerárquico se subdivide en dos clases: 1) con procesamiento de datos y 2) con procesamiento des centralizado. Ambos se estudiarán en esta sección. En la sección siguiente se estudiará el enfoque de sistemas de la estructura del sistema de información.

Ya sea que hagamos referencia a los métodos centralizados o des centralizados de procesamiento de datos, las características siguientes pertenecen al enfoque jerárquico:

1. Las actividades de procesamiento de datos no guardan relación con un análisis de las necesidades de información. Es decir, las instalaciones para el procesamiento han sido dispuestas para procesar los datos, haciendo poco caso de las necesidades de los usuarios específicos.
2. Los bancos de datos separados se dividen entre las líneas funcional, departamental y divisional.
3. En el procesamiento y flujo de la información se observan las relaciones entre subordinado y superior. Es decir, la información se solicita de arriba hacia abajo a través de las líneas funcional, departamental o divisional, y el reporte sube por esas líneas.

Por tradición un gran número de organizaciones aplican el enfoque jerárquico cuando el procesamiento de datos, y la correspondiente recopilación de informes, son manejados por cada área funcional de acuerdo con sus objetos particulares. El nivel básico de cada área funcional recoge los datos correspondientes a sus operaciones, los hace procesar y pasa un informe al siguiente nivel estratégico. Además, la empresa que utiliza este enfoque cuenta con un departamento de contabilidad independiente de las otras áreas funcionales, o basado en el área de finanzas. Este departamento procesa especialmente datos financieros para toda organización y lleva su catálogo de cuentas. Pero además de procesar las cuentas por pagar, cuentas por cobrar, etc., su función en la organización es la de administración y contraloría. Tradicionalmente, estas funciones proporcionan escasa información funcional a los funcionarios de la empresa. En otras palabras, el departamento de contabilidad puede visualizarse como un sistema de información financiera que lleva a cabo un catálogo de cuentas, realiza tareas rutinarias de procesamiento de datos y, primordialmente, presenta estados financieros a los usuarios internos y externos.

Como ya se dijo anteriormente, hay dos métodos para procesar datos: 1) manual y 2) por computadora, pero la computadora y su tecnología predomina y predominará notablemente en muchos sistemas actuales de información, además de que proporciona un método más sistemático para ilustrar muchos conceptos específicos.

1.4.2 Enfoque Jerárquico para el Diseño de Sistemas de Información

La manera en que la información fluye por la organización depende básicamente de dos cosas: 1) la filosofía de la administración y 2) el enfoque dado al diseño de los sistemas de información. La filosofía de la administración concierne al grado de centralización o descentralización de la autoridad y la responsabilidad. Nuestro análisis no se relaciona específicamente con esta comparación; sólo se trata de una forma subcotidiana. En cuanto a lo segundo, es decir el enfoque dado al diseño y a la disposición del sistema de información, hay básicamente dos enfoques generales: 1) el enfoque jerárquico y 2) el enfoque de sistemas. El enfoque jerárquico se subdivide en dos clases: 1) con procesamiento de datos y 2) con procesamiento des centralizado. Ambos se estudiarán en esta sección. En la sección siguiente se estudiará el enfoque de sistemas de la estructura del sistema de información.

Ya sea que hagamos referencia a los métodos centralizados o des centralizados de procesamiento de datos, las características siguientes pertenecen al enfoque jerárquico:

1. Las actividades de procesamiento de datos no guardan relación con un análisis de las necesidades de información. Es decir, las instalaciones para el procesamiento han sido dispuestas para procesar los datos, haciendo poco caso de las necesidades de los usuarios específicos.
2. Los bancos de datos separados se dividen entre las líneas funcional, departamental y divisional.
3. En el procesamiento y flujo de la información se observan las relaciones entre subordinado y superior. Es decir, la información se solicita de arriba hacia abajo a través de las líneas funcional, departamental o divisional, y el reporte sube por esas líneas.

Por tradición un gran número de organizaciones aplican el enfoque jerárquico cuando el procesamiento de datos, y la correspondiente recopilación de informes, son manejados por cada área funcional de acuerdo con sus objetos particulares. El nivel básico de cada área funcional recoge los datos correspondientes a sus operaciones, los hace procesar y pasa un informe al siguiente nivel estratégico. Además, la empresa que utiliza este enfoque cuenta con un departamento de contabilidad independiente de las otras áreas funcionales, o basado en el área de finanzas. Este departamento procesa especialmente datos financieros para toda organización y lleva su catálogo de cuentas. Pero además de procesar las cuentas por pagar, cuentas por cobrar, etc., su función en la organización es la de administración y contraloría. Tradicionalmente, estas funciones proporcionan escasa información funcional a los funcionarios de la empresa. En otras palabras, el departamento de contabilidad puede visualizarse como un sistema de información financiera que lleva a cabo un catálogo de cuentas, realiza tareas rutinarias de procesamiento de datos y, primordialmente, presenta estados financieros a los usuarios internos y externos.

Procesamiento de Datos Centralizado.

Con un procesamiento de datos centralizado, la mayor parte de las operaciones de procesamiento se efectúa en una sección especial. Por lo común, se trata de un Departamento de Procesamiento de datos. Sin embargo, el procesamiento puede efectuarse también 1) en una oficina de servicios, compañía privada ajena a la empresa, que ofrece diferentes servicios de procesamiento, 2) en instalaciones de tiempo compartido, cuyos servicios pueden comprarse o rentarse a alguna compañía privada, y 3) en un contrato de administración de recursos, en virtud del cual una empresa privada toma a su cargo la operación de procesamiento de datos. Sin embargo, las empresas de hecho no están limitadas a enfocarse de una sola manera en el diseño de su sistema de información. Por ejemplo, una empresa cuyo procesamiento de datos está descentralizado podría tener igual facilidad para tener diferentes áreas funcionales, usando recursos diferentes para llevar sus necesidades individuales de procesamiento. Por ejemplo, los departamentos de ingeniería e investigación y desarrollo podrían obtener recursos compartidos, mientras que el de finanzas usaría una computadora propia además de comprar tiempo adicional a una oficina de servicios. Al mismo tiempo, el personal de comercialización podría estar usando la computadora del departamento de finanzas, mientras que el grupo de fabricación haría uso de un sistema computarizado operado por una empresa administradora, y así sucesivamente.

Procesamiento de Datos Descentralizado

La estructura del enfoque jerárquico para el procesamiento descentralizado de datos implica también como ocurre el enfoque jerárquico que centraliza el procesamiento, la existencia de varias áreas funcionales más o menos autónomas, que constituyen suborganizaciones dentro de la organización. El flujo de la información sigue siendo vertical dentro de cada área funcional. Sin embargo, hay una gran diferencia, y es que cada área tiene control sobre sus propias actividades de procesamiento, mientras que, con el procesamiento centralizado, el departamento de procesamiento electrónico de datos está separado de las otras áreas.

Las actividades de procesamiento de datos se dispersan entre las áreas tradicionales de la organización. Las funciones contables se efectúan en el departamento de finanzas o en el de controlaría; el control de inventarios y el procesamiento de datos de fabricación los maneja algún grupo del área de producción, etc. Las aplicaciones de procesamiento de datos vienen a ser una u otra de las áreas funcionales de la empresa.

Resumen General del Enfoque Jerárquico

Independientemente del método empleado, el flujo general de la información es el mismo. La diferencia radica en que el procesamiento centralizado está controlado por una autoridad central, mientras que el procesamiento descentralizado lo controla exclusivamente el administrador del área a la cual sirve. A sí mismo, aunque los ejemplos anteriores parecen indicar lo contrario, el procesamiento de datos centralizado puede dispersar el personal y el equipo, sobre todo en las empresas que cuentan con varias divisiones con el fin de tener esos medios cerca de donde está el trabajo; pero el procesamiento seguirá bajo control central.

1.4.3 Enfoque de Sistemas para el Diseño de Sistema de Información.

El objetivo básico del enfoque de sistemas para el diseño de sistemas de información consiste en tener disponible un gran banco de información comprensible que fluya con toda oportunidad hacia aquellos administradores que, al recibirla, puedan tomar decisiones efectivas. La finalidad de este enfoque es optimizar la organización en conjunto, superando así la sub-optimización. Como ya se dijo, la persona clave a quien corresponde aplicar el enfoque de sistemas es el analista, identifica la información y las necesidades de procesamiento de datos, y formula métodos para satisfacer esas necesidades.

Aunque el enfoque de sistemas reconoce estrictamente los niveles estratégico, táctico y técnico de decisión, así como las actividades clásicas de planeación y control, considera a la organización como un conjunto interrelacionado y coordinado de subsistemas que, con sus conexiones, se parecen más a una red que a una jerarquía. El enfoque de sistemas, además, separa en forma muy apreciable el sistema de información. Es decir, el sistema administrativa realiza las actividades de planeación, control y toma de decisiones, mientras que el sistema de información le proporciona información en la cantidad y calidad adecuadas.

Básicamente, siguiendo el enfoque de sistemas, es posible desarrollar dos clases de sistemas de información: 1) el sistema integrado y 2) el sistema distribuido. De diversos modos, el sistema integrado es similar al método de procesamiento centralizado. Mientras que el sistema distribuido se asemeja al método descentralizado. En esta sección se señalarán las diferencias significativas.

Antes de estudiar específicamente las dos clases de sistemas de información, en la figura 1.4.8 ilustraremos la relación que existe entre el sistema administrativo y el sistema de información.

Adviértase que la información fluye directamente hacia el usuario que la requiere y está autorizado para utilizarla. Al mismo tiempo, hay poco o ningún re procesamiento de datos entre el superior y el subordinado. Mediante este enfoque, es mucho más fácil implantar un flujo de información tanto vertical como lateral, que trasciende las líneas funcionales tradicionales.

El Sistema Integrado

Como se dijo en la sección anterior, con el procesamiento de datos centralizado el énfasis recae estrictamente en las actividades de procesamiento. En ese caso, generalmente existe un departamento de Procesamiento de Datos, que incluye una colección de máquinas manejadas por técnicos y donde se toman decisiones, basadas en factores que giran alrededor de las condiciones internas, tomando apenas en cuenta los factores que influyen en el conjunto de la organización. Es decir, por lo común el departamento de procesamiento de datos se encuentra aislado de la corriente general de las operaciones.

Hay que tener en cuenta que los diferentes subsistemas utilizan los datos, en grados diferentes y que, en consecuencia, el sistema debe estar diseñado con las relaciones apropiadas entre los subsistemas, con el fin de beneficiar a toda la red independientemente.

Por su parte, el enfoque de sistemas integrados de información implica canalizar todos los datos de la organización hacia un banco común para realizar el procesamiento y las funciones de información de toda la empresa en forma conjunta. Este enfoque está muy de moda en nuestros días, pero, parece muy importante que todos los objetivos se pueden alcanzar en futuro previsible. Por ahora, la tarea de recopilar todos los datos y atender todas las necesidades de información de una empresa es técnica y económicamente impracticable. No obstante, aunque la implantación de un sistema totalmente integrado

parece irrealizable por el momento, muchas empresas están tratando de lograr una mayor integración. Este avance se basa en dos aspectos: 1) la falta de sofisticación con el enfoque jerárquico, debido a la fragmentación y la falta de coordinación de la función informativa, y 2) la posibilidad potencial de lograr una mayor integración gracias a los avances tecnológicos observados sobre todo en el área de las computadoras y las telecomunicaciones.

El sistema de organización integrado promete muchos beneficios, pero también puede presentar algunos problemas para la administración que no esté lista para comprometerse en la medida necesaria para instalar tal sistema. Puesto que el concepto de sistema integrado es un tanto monolítico, un compromiso que no sea total puede dar lugar al caos. La implantación de esta clase de sistema exige: un empeño total por parte de la administración (a todos los niveles), basado en un plan maestro de gran alcance, la contratación de personal altamente especializado, la adquisición de equipo completo y el desembolso del capital suficiente.

Además de exigir la inversión a largo plazo, de tiempo y recursos, el sistema integrado requiere también el cambio de los métodos tradicionales de manejo de datos e información. Puesto que el sistema integrado reúne todas las funciones de recopilación de datos, procesamiento, producción de información y comunicación, debe efectuarse una consolidación de los datos financieros y de operación. Como está consolidación comprende a la función contable, desaparece la necesidad de un departamento separado de contabilidad, como tal. Esto sucede porque corresponderá al sistema de información llevar a cabo todas las actividades, además de proporcionar otros informes operativos.

Otras áreas que reciben la influencia del enfoque de sistemas integrados incluyen al personal destinado a ciertas operaciones específicas, como son la producción y la comercialización. La tarea principal de estas áreas consiste en proporcionar información a los diferentes niveles administrativos mediante un análisis de datos basado en la aplicación de la investigación de operaciones y los modelos aplicados a la administración. Puesto que la finalidad del sistema integrado de información es servir a los usuarios de toda la organización, resulta natural que todas las funciones de estas áreas quedarán también incorporadas al sistema, lo cual dará lugar a la utilización óptima y coordinada de esas funciones.

En las subsecciones siguientes se analizará el enfoque de sistemas integrados con el fin de entender mejor su naturaleza y determinar sus ventajas y desventajas.

1. Un componente clave -El banco de datos. Las características básicas de un sistema integrado son: 1) una rápida respuesta a las preguntas planteadas a través de las terminales a control remoto. 3) actualización instantánea y simultánea de archivos y 4) procesamiento centralizado de datos en lotes, además del procesamiento en línea. Aunque las terminales de control remoto y el procesamiento en línea se presentan como características del sistema integrado, estos sistemas se pueden diseñar sin esas características, según las necesidades de acceso. Dicho de otro modo, es posible, por ejemplo, manejar todas las actividades en base a lotes. Sin embargo, adviértase que todas esas características implican la utilización de banco común de datos, que es el componente clave de todo sistema integrado de información. Los elementos de entrada tiene su origen en los eventos, y las transacciones se registran, ya sea directamente en el banco de datos o en documentos fuente que se guardan para su entrada anterior al sistema.

La selección de los dispositivos de registro que se van a utilizar para el banco de datos depende de las alternativas disponibles y del propósito de los diferentes archivos. Al examinar el uso de los archivos, el analista debe evaluar la utilidad principal de cada uno. Es necesario responder a las preguntas siguientes:

a) ¿Existe redundancia en el sistema actual de archivo? ¿Puede reducirse esa redundancia? Es decir, ¿Pueden consolidarse los archivos?

b) ¿En qué se utiliza el archivo? ¿Es una etapa intermedia, o sirve para resultados definitivos? ¿Tiene que ser clasificada o fusionado para un procesamiento posterior?

c) ¿Con qué frecuencia se cambia o actualiza el archivo?

d) ¿Qué criterio se sigue para la conservación de archivos?

e) ¿En qué forma se usa el archivo? ¿Podrá el procesamiento por lotes satisfacer las demandas impuestas al sistema, o se plantea en forma de preguntas la mayoría de las demandas. ¿Con qué frecuencia se plantean consultas al archivo?

f) ¿Es muy necesario el archivo? ¿Se agregan o se suprimen registros de manera constante? ¿Cuál es su tasa de crecimiento?

Cuando se requiere un procesamiento periódico y por lotes, la cinta magnética constituye un dispositivo aceptable de almacenamiento. En cambio, si la demanda proviene de la línea, se requiere un dispositivo de acceso directo (DAD). Aunque algunos teóricos insisten en que el banco común de datos debe conservarse en un archivo de fácil acceso, se piensan que tal disposición es técnica y económicamente impracticable por ahora. Un archivo general eliminaría la duplicación y la redundancia; pero la longitud de los registros sería tremenda. Desde un punto de vista práctico, el banco de datos está formado realmente de un cierto número de archivos, incluyendo el de clientes y ventas, el de inventario, el de contabilidad y el de personal, coordinados de manera que disminuyan la redundancia en el registro, mantengan la eficiencia del procesamiento y proporcionen información al usuario en el momento oportuno. Corresponde al analista diseñar el banco de datos de manera que satisfaga las necesidades de los usuarios, ya que de otro modo seguirán llevando su propio sistema y malograrán el propósito del banco de datos. En la figura 1.4.9 se ilustran el sistema integrado y su banco común de datos.

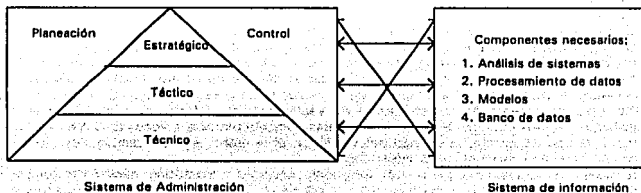


Figura 1.4.8 Flujo de la información con el enloque de sistemas.

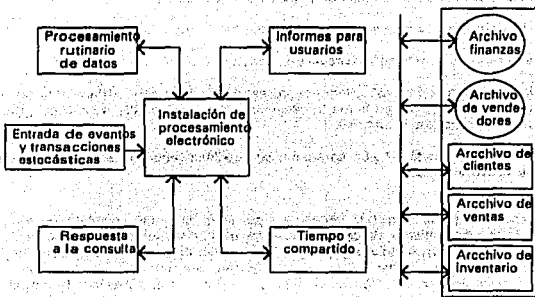


Figura 1.4.9 Un sistema integrado de información, con banco de datos.

2. Un ejemplo típico. Como se dijo en el párrafo anterior, el banco de datos no significa necesariamente un solo archivo, sino cierto número de archivos relacionados entre sí que contengan datos para diferentes aplicaciones. Con respecto a la consolidación de archivos para reducir la duplicación y la redundancia, el analista de sistemas busca los puntos comunes de las partidas y operaciones. La información con respecto a los clientes de un ejemplo de un área en que a menudo el analista puede establecer un banco común de datos. La información sobre los clientes de una empresa tiene importancia por lo menos para cuatro departamentos funcionales diferentes: 1) registro y cobro de pedidos (R/C.P), 2) contabilidad de ventas (C.V.), 3) cuentas por cobrar y créditos (C.C./C.), 4) publicidad y promoción (P./P.). El departamento de registro y cobra de pedidos necesita la información correspondiente al cliente para formular la documentación del embarque y la factura. Esta información incluye el número de cliente, nombre y dirección para cobranza, nombre y dirección de envío, instrucciones especiales de manejo, número de vendedor, etc. El departamento de contabilización de ventas requiere información similar con el fin de elaborar diversos informes de análisis de ventas. también requiere a veces la clave especial que indica el tipo de cliente, la división del vendedor, el territorio, etc. En cambio, las cuentas por cobrar necesitan únicamente la información precisa para formular los estados de cuenta y los recordatorios. El cuerpo departamento funcional, publicidad y promoción, necesita el nombre y el domicilio, así como alguna clave especial que indique envíos de correspondencia seleccionada.

En una empresa típica, cuando se aprueba a un nuevo cliente, se formula algún tipo de autorización. Copia de dicha autorización se remiten a cada uno de los departamentos correspondientes, para generar la entrada al archivo maestro de clientes. Siguiendo el enfoque del banco común de datos, una sola unidad dentro del sistema de información asume la responsabilidad de proporcionar información precisa y oportuna sobre el cliente a un archivo único, donde puede ser tomada para diferentes usos según convenga y haya sido autorizado. En algunas áreas, como cuentas por cobrar y centralización de ventas, donde se tiene también información de actividades, esta información puede llevarse en archivos creados para facilitar el procesamiento.

En la figura 1.4.10 pueden verse las diferencias en la amplitud de los archivos. En la mitad superior se muestra el enfoque tradicional; en la parte inferior aparece en el de banco común de datos. Obsérvese que, en este último, se requieren elementos comunes a todos los archivos, con el fin de relacionar el archivo de información estática con el archivo de información dinámica. No obstante, la redundancia es mínima. En la parte el capítulo II se estudiará más a fondo la estructura del Banco común de datos. Por ahora, nuestro propósito consiste únicamente en mostrar la manera en que puede formarse un banco común de datos.

3. La red de comunicaciones. Los usuarios, repartidos por toda la organización, interactúan directamente con el sistema integrado. A menudo, esta interacción se efectúa a través de los dispositivos de telecomunicación; pero a veces podría hacerse simplemente solicitando un reporte y recogiéndolo días después. Para establecer una red de comunicaciones entre seis usuarios en el sistema integrado, se necesitarían: N (seis) canales, como se indica en la figura 1.4.11.

Campo de datos	Tamaño	Aplicaciones que emplean campos			
		E./F.C.	C.V.	C.C./C.	P./P.
(1) No. de cuenta del cliente	10	X	X	X	X
(2) Nombre del cliente	30	X	X	X	X
(3) Dirección del cliente	60	X		X	X
(4) Nombre del destinatario del envío	30	X	X		
(5) Dirección del destinatario del envío	60	X			
(6) Instrucciones de manejo especial	45	X			
(7) Clave2 de la división de ventas	5	X	X	X	X
(8) Clave de territorio	3	X	X		X
(9) Clave de vendedor	2	X	X	X	X
(10) Número de DUN	8	X	X		
(11) Comisión	6	X	X		
(12) Límite de crédito	6	X	X		
(13) Clase de transacción	2	X	X		
(14) Clave de industria	8	X		X	X
(15) Partidas abiertas	140			X	
(16) Ventas mensuales	72		X		
Total de caracteres de información por cliente (al total de todas las aplicaciones es de 806 caracteres por cliente).		251	182	245	128
(1) No. de cuenta del cliente	10	X	X	X	X
(2) Nombre del cliente	30	X			
(3) Dirección del cliente	60	X			
(4) Nombre del destinatario del envío	30	X			
(5) Dirección del destinatario del envío	60	X			
(6) Instrucciones de manejo especial	45	X			
(7) Clave2 de la división de ventas	5	X			
(8) Clave de territorio	3	X			
(9) Clave de vendedor	2	X			
(10) Número de DUN	8	X			
(11) Comisión	6	X			
(12) Límite de crédito	6	X			
(13) Clase de transacción	2	X			
(14) Clave de industria	8	X			
(15) Partidas abiertas	140			X	
(16) Ventas mensuales	72		X		
Total de caracteres de información por cliente (al total de todas las aplicaciones es de 507 caracteres por cliente).		275	82	150	0

** Tomando como base 35 caracteres por partida abierta; No se necesita un archivo por separado para publicidad y promoción.

Figura 1.4.10 Necesidades comparadas del archivo tradicional y el banco de datos.

4. El jefe de sistemas de información. El jefe de Sistemas de Información tiene tareas diferentes a las que realiza quien en general se denomina Jefe de la Computadora o Jefe de Procesamiento de Datos. Sus responsabilidades son mucho más amplias, puesto que debe conocer a fondo todas las operaciones que se efectúan en la empresa. Este funcionario debe ser capaz de unificar los diferentes componentes de la organización y sub-optimizarlos mediante un flujo de información un flujo de información dirigido imparcialmente hacia todos los usuarios. Siendo un organizados más que un técnico, tendrá que ocuparse de resolver los problemas y de optimizar los objetivos generales de la empresa. Sus actividades deben ser neutrales e independientes de otros ejecutivos; no debe tener preferencias por ningún departamento y debe responder únicamente ante la dirección, los dueños o los consejeros de la organización.

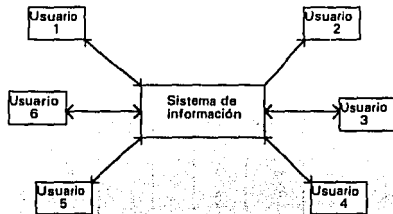


Figura 1.4.11 Red de comunicaciones entre seis subsistemas, en una empresa que aplica el enfoque de sistemas.

El jefe de sistemas de información debe tener un conocimiento profundo de todas las áreas funcionales. Por ejemplo, en una empresa manufacturera, debe entender las funciones de fabricación, ingeniería, comercialización, logística y finanzas. Sin ese conocimiento, le será difícil atender las necesidades de cada usuario y ver la forma en que se relacionan las diferentes funciones.

Este funcionario también debe ser hábil en el manejo de tareas de grupo y proyectos, ya que una buena parte del trabajo de sistemas se lleva a cabo sobre esa base. Debe estar familiarizado con las técnicas de planeación, programación y control así como con la relación costo/efectividad. También es necesario tener un buen conocimiento de la tecnología de la computación, las técnicas de procesamiento de la información y los principios de la administración y la toma de decisiones.

La responsabilidad del Jefe de Sistemas de Información consiste en dirigir el sistema que lleva a cabo el procesamiento de datos y las actividades para todos los departamentos funcionales principales de la empresa. El sistema es un generador de información; todas las demás áreas funcionales, así como los subsistemas, son los usuarios de esa información.

Específicamente, entre los deberes del jefe de Sistemas de Información figuran los siguientes:

- a) La planeación y control de todas las actividades incluidas en el sistema de información.
- b) La evaluación y selección del equipo de procesamiento y medios auxiliares.
- c) La revisión del sistema y el desarrollo de diseños alternativos, cuando sean necesarios.

d) La selección de personal. Incluyendo: a) contadores, b) analistas, c) programadores, d) analistas de investigación de mercados, e) jefes de los equipos de procesamiento de datos, f) analistas de presupuestos, g) investigadores de operaciones, y así sucesivamente.

e) La creación de un programa de entrenamiento de los empleados que ingresan al sistema de información. A sí mismo, el establecimiento de los criterios de rendimiento de todo el personal.

f) La elaboración de un sistema de prioridades para el procesamiento.

g) El establecimiento de los requisitos de edición de los elementos de entrada y seguridad de los datos.

h) La definición y documentación de los objetivos de todo sistema de información y la obtención de su aprobación por parte de la gerencia.

i) El desarrollo de un programa educativo y de entrenamiento para los usuarios del sistema.

5. Posición del sistema integrado dentro de las organizaciones. La posición del sistema integrado de información está basada en cuatro criterios:

1) Por ejemplo, el sistema de información para las reservaciones de una línea aérea exige casi exclusivamente el procesamiento en línea, con mucha autonomía por parte de los usuarios.

2) El nivel de automatización. Para el uso intenso de equipo complejo, su manejo requiere de personas con alto nivel de preparación. Con frecuencia, estas personas están mucho mejor capacitadas que los administradores a quienes proporcionan información.

3) El nivel de integración. Un sistema totalmente integrado denota por naturaleza un sistema autónomo, independiente del control de la administración funcional.

4) Nivel de asignación de recursos. Si la dirección considera que el sistema de información es una función importante dentro de la organización, el Jefe de Sistemas de Información y su personal ocuparán un alto nivel. Siendo así, el Jefe de sistemas de información comparecerá únicamente ante el presidente o principal ejecutivo de la empresa.

Considerando tales funciones, la figura 1.4.12 presenta esta estructura organizativa, señalando la posición adecuada del sistema.

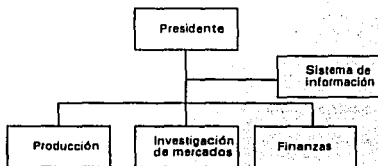


Figura 1.4.12 Ubicación apropiada del sistema integrado de información.

A este nivel, el sistema de información puede realizar sus tareas sin restricciones y proporcionar informes imparciales, objetivos de detalle, a todos los usuarios. El grupo que elabore en el sistema debe estar formado por especialistas de información, cuyo deber y responsabilidad consiste en proveer la información apropiada a todos los usuarios, esta función la deben desempeñar en la forma que crean conveniente, a través de un trabajo de investigación y consulta con los usuarios, sin ser estorbados ni limitados por ningún grupo (o personas) dentro o fuera de la organización. Puesto que son especialistas, quienes forman el personal de información tienen la obligación de investigar las necesidades respectivas y recomendar métodos cuando y donde se presente la oportunidad. Todo servicio a las diferentes unidades debe presentarse de manera imparcial, independiente y profesional.

6. Ventajas probables del enfoque de sistemas integrados de información. Las ventajas que pueden derivarse de la implantación de un sistema integrado de información son las siguientes:

a) Reducción de la redundancia y la duplicación en los archivos y en el trabajo de programación, así como una mayor estandarización.

b) Mayor seguridad, control y protección al banco de datos contra el acceso de personas no autorizadas.

c) Reducción en la intervención del personal en las operaciones de entrada, procesamiento y salida, disminuyendo así la posibilidad de error.

d) Permite la actualización instantánea y simultánea de los archivos (los que se encuentren en el DAD), mediante la cual se podrá obtener información del momento e identificar situaciones que exijan atención inmediata y acción correctiva.

e) Permite a más de un usuario recuperar, actualizar o suprimir datos del banco común (siempre que los archivos se encuentren en el DAD).

f) Libera a la administración de las actividades rutinarias de procesamiento de datos y toma de decisiones.

g) Puesto que el sistema integrado satisface una gran variedad de necesidades de información y sirve a múltiples aplicaciones de procesamiento en toda la organización, existe un potencial de economía en gran escala. Con un volumen apropiado de procesamiento, la tecnología de computación ofrece un reducido costo de operación. Esta reducción se debe a que el costo unitario por el elemento procesado, es más bajo tanto en equipo como en personal. En el sistema integrado se logra una mejor utilización de equipo y personal especializado y, por tanto, la implantación y aplicación más formal y rápida de nuevas técnicas.

h) Si la empresa tiene un gran sistema integrado, con equipo moderno mayores serán sus oportunidades de encontrar empleados calificados y profesionales. También es posible establecer y contar con mejores programas de entrenamiento.

i) Un mejor uso de la capacidad de procesamiento. Con el enfoque jerárquico, una división podría tener capacidad excesiva, mientras que en otra habría una sobrecarga.

j) Un aumento general del rendimiento, debido a la disponibilidad de información más oportuna, apropiada y exacta. Por ejemplo, una mejor investigación de créditos y un control más efectivo de inventarios hacen que el costo sea más bajo. Del mismo modo, propicia la reducción de pedidos atrasados y del mejor servicio a los clientes puede derivarse un incremento en el ingreso.

k) Liberación del sistema administrativo, separando la información de la administración. El sistema administrativo puede obtener una mayor amplitud de información y trabajar a diferentes

niveles de detalle sin necesidad de que entre a diversas actividades de procesamiento. La administración tiene acceso a diferentes niveles de información. Esto es un factor positivo que mantiene alertas y eficientes a los administradores de centros de utilidades, porque la información con respecto al centro (o división) proviene del sistema de información y no del propio centro de utilidades.

l) Una reducción en la parcialidad de la información, que sería inherente a otros sistemas, ya que las actividades de decisión y operación está separadas de las de evaluación y comportamiento. Los datos se recopilan y la información se produce desde una posición (y un punto de vista) desligada e independiente de quienes toman las decisiones de operación. La separación del sistema de información deja a la administración en libertad de administrar, usando como pauta y guía los informes reportados. El sistema de información conserva la objetividad y realiza el procesamiento de manera impersonal, lo cual da lugar a la propagación de informes pertinentes e imparciales, sin hacer desmerecer a toda la organización.

m) La posibilidad de que áreas pequeñas o remotas tengan acceso a un sistema centralizado de computación a sus archivos. Una sola área podría no estar en condiciones de tener un equipo propio.

n) Una mayor posibilidad de implantar y poner en marcha planes maestros para el sistema, acordes con los planes a largo plazo de la organización en conjunto.

o) La posibilidad de hacer una mejor evaluación general de los proyectos en cuanto a su factibilidad desde el punto de vista técnico, económico, operativo y de programación.

7. Desventajas probables del enfoque de sistemas integrados de Información. Las desventajas probables son las siguientes:

a) Para lograr una efectividad máxima, el personal del sistema de información, sobre todo los analistas de sistemas, deben tener el nivel necesario de autoridad y responsabilidad para ejecutar sus tareas en forma adecuada. De otro modo, el sistema estará condenado al fracaso.

b) Sin la cooperación de todos los niveles administrativos, el sistema no logrará sus objetivos.

c) Falta de personal calificado para diseñar, implantar y mantener un sistema altamente integrado y que utiliza equipo complicado.

d) Varios usuarios, ahora sin el control de las actividades de procesamiento, podrían tender a rebelarse contra el procesamiento integrado.

e) Existe la posibilidad de que un sistema integrado no responda a las necesidades de los usuarios.

f) Los paros pueden ser catastróficos en los sistemas integrados. Por ejemplo, si la CPU se descompone, el sistema se alterará totalmente a menos que el sistema de información tenga un respaldo. Sin embargo, esto resulta costoso y desde luego, redundante. No es posible exagerar esta desventaja, porque, estando todas las actividades de procesamiento integradas en un sistema, una falla de cierta duración podría significar la ruina de la empresa, especialmente si se trata de instituciones financieras como bancos, firmas de corredores de bolsa y compañías de seguros, donde el procesamiento de datos constituye la savia de la organización.

g) El costo es muy elevado.

h) Debido a las interdependencias, es difícil introducir modificaciones.

I) El intento de implantación no tendrá éxito a menos que la administración esté dispuesta a comprometer a largo plazo su tiempo y sus recursos.

J) La implantación de un sistema totalmente integrado es un riesgo desde los puntos de vista técnico y financiero, incluso para las grandes empresas.

El Sistema Distribuido.

Hasta aquí, parecería que el sistema integrado tiene mucho que ofrecer como enfoque viable del desarrollo de un sistema de información. Al mismo tiempo, muchas autoridades piensan que este enfoque es una equivocación y no puede satisfacer de manera efectiva las muy diversas necesidades de la organización.

La experiencia ha demostrado que el enfoque masivo del sistema "universal" conduce rápidamente a la elevación substancial en el costo indirecto y administrativo de la programación, por analogía, la computadora no es una máquina universal. "Un camión, una bicicleta u un auto para carreras tienen cada uno un motor diferente, diseñado para dar un alto rendimiento dentro de su uso específico. Un sistema de información computarizado que pretenda servir a todos y para todo, desde el comienzo está condenado al fracaso. El equilibrio del costo hay que buscarlo entre el gasto que implica el control de cierto número de sistemas especializados dispersos, que individualmente puede ser óptimo pero globalmente resulta muy elevado, y la posesión de un sistema universal".

El principio básico del enfoque de sistemas integrados es que lleva a cabo una concentración total de datos de la organización en un banco común, realizándose todo el procesamiento desde un complejo central. Quienes apoyan el sistema integrado piensan que es lógico que las necesidades de información de todos los usuarios pueden ser satisfechas equitativamente, sin importar si se trata de finanzas, comercialización o producción. Sin embargo, el Profesor Jhon Dearden sostiene que se observa una notable falta de éxito en la implantación de sistemas integrados y que ello se debe a cuatro sofismas:

Sofisma # 1: La información administrativa es lo suficientemente homogénea como para constituir un área de especialización que un experto pueda manejar.

Sofisma # 2: Si los diferentes sistemas de información de la empresa se establecen por separado, el sistema resultante de información administrativa adolecerá necesariamente de falta de coordinación, por lo tanto será ineficiente y poco satisfactorio.

Sofisma # 3: El enfoque de sistemas es una bendición para administradores de empresas.

Sofisma # 4: Es práctico centralizar el control de todo el sistema de información administrativa de la empresa.

El profesor Dearden agrega que las empresas deberían tratar de desarrollar y coordinar sistemas individuales que satisfagan las necesidades de información de las diversas áreas de la organización. Piensa que este enfoque es necesario en vista de que un sistema integrado abarca un conjunto tan enorme de actividades, que nadie es capaz de poseer la habilidad suficiente para integrar en forma efectiva un complejo de esta clase.

Así pues, la alternativa para el enfoque de los sistemas distribuidos. Mientras que el sistema integrado es monolítico por naturaleza, el sistema distribuido es modular. El sistema integrado hace uso de la instalación central de procesamiento de datos con un banco común; en cambio, el sistema

distribuido utiliza un grupo o combinación de sistemas de información, dispersos de tal manera que se forma un sistema de subsistemas, o un sistema de islas de información.

El objetivo básico del sistema distribuido consiste en establecer subsistemas relativamente independientes que, sin embargo, están vinculados por las relaciones de comunicación. Este sistema de sistemas constituye una red de subsistemas, ubicados en las áreas en que se necesitan y que son adecuados a las mismas. En esa red se presentarán tres condiciones básicas: 1) algunos de los subsistemas tendrán que interactuar con otros subsistemas, 2) algunos tendrán que compartir con otros los archivos y hasta los recursos de procesamiento de datos, y 3) algunos subsistemas tendrán muy poca interacción con los demás y, para todo intento y propósito, estarán bastante separados y serán auto suficientes.

Como se hizo con el enfoque de sistemas integrados, en las subsecciones que siguen se analizará el enfoque de sistemas distribuidos, con el fin de ofrecer una idea más completa del mismo y determinar sus ventajas y desventajas.

1. El banco de datos. El establecimiento de un banco común de datos para toda la empresa exige una buena suma de recursos, habilidad y tiempo. Por otra parte, las diferentes áreas funcionales pueden requerir un tratamiento diferente y no todos los datos se prestarán a la integración. Además, en virtud de la diversidad inherente a ciertas organizaciones, sólo una parte de los datos será aplicable a todas las áreas. En tal caso, estos elementos comunes pueden ser consolidados y almacenados en algún punto central, para que otras áreas tengan fácil acceso los mismos, cuando y si es que llegan a necesitarlos.

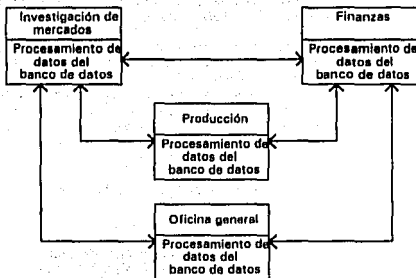
Anticipándose a lo que ocurrirá en los próximos diez años, el primer pronóstico es la desaparición de la actual tendencia al banco de datos como imagen predominante del usuario modelo en el desarrollo de los sistemas de información a la gerencia. La atención se volverá hacia la información que depende del usuario.

Parece que, para muchas empresas, el enfoque racional de la estructuración del banco de datos implica pensar en una red de bancos. Uno de ellos (por ejemplo, ubicado en las oficinas principales) contendría un resumen general de datos, funcionando como almacén y como cámara de compensación, con capacidad para interactuar con algunos, si no con todos, además bancos del sistema distribuido.

... En realidad, muy pocos archivos deben ser centralizados. El sistema de reservaciones de las empresas aéreas parece justificar la centralización; pero aun en este caso, el archivo centralizado único no es necesariamente el mejor enfoque. Los archivos concentrados se requieren sólo a lo largo de rutas específicas; muchas partes del sistema tal vez nunca buscarán tener acceso a los mismos. Lo mismo ocurre con todos los inventarios distribuidos en todo el país que comprenden muchos almacenes y depósitos. Será raro que un comerciante de California solicite un artículo a un almacén de Florida.

Los grandes archivos pueden convertirse en muchos archivos pequeños, situado cada uno en "áreas de necesidad". Si en la localidad no es posible encontrar lo que se busca, una persona con un teléfono será la solución más efectiva y económica del problema. Si se desea mayor "sofisticación" a un costo más alto, varias máquinas pequeñas podrán comunicarse entre sí mediante simples preguntas: "¿Tienen en existencia la parte X?" El control no se pierde distribuyendo la información; más bien mejora y el costo es menor.

2. Un ejemplo típico. El sistema distribuido dispersa tanto los bancos de datos como las facilidades de procesamiento, situándolos cerca de los usuarios. En la figura 1.4.13 puede verse una típica empresa manufacturera que utiliza el sistema distribuido.



Ejemplo de un sistema distribuido.

Figura 1.4.13 Ejemplo de un sistema distribuido

En este ejemplo, hay una multiplicidad de archivos de datos y medios de procesamiento. Por ejemplo, vamos a suponer que la empresa cuenta con la siguiente red general de dispositivos de procesamiento:

a) El departamento de comercialización tiene una minicomputadora, dos discos para el almacenamiento de datos locales, un TRC y un teletipo. Este subsistema puede procesar la mayor parte de sus datos, particularmente los que sirven para propósitos informativos, como análisis de ventas, tendencias en el mercado, relaciones de la competencia y comparaciones de rendimiento. Gran parte de los datos que requiere y procesa la proporciona el banco situado en las oficinas generales. El departamento de comercialización necesita muy pocos datos del área de finanzas, pero requiere muchos de la producción, sobre todo con respecto a inventarios y a programas de artículos terminados.

b) El departamento de producción cuenta con una "terminal inteligente" (terminal que por lo común incorpora una minicomputadora autónoma para procesamiento), en vista de que, aunque no necesita hacer una gran cantidad de procesamiento local, sí tiene que transmitir bastante información a las áreas de comercialización y finanzas, lo mismo que las oficinas generales. El departamento de producción utiliza su subsistema, además de usarlo para transmisión de informes para toda la red, para programar la producción, para analizar el flujo y para controlar las variaciones.

c) El departamento de finanzas cuenta con un sistema de computación más o menos grande, con varios receptores de cinta y discos. Este departamento efectúa una gran cantidad de análisis, como comportamiento de las utilidades, variaciones con respecto al presupuesto, el flujo de efectivo, tendencias en la industria, conciliación mensual, administración de activos, planeación de efectivo, Formulación de estados financieros, etc. Recibe de las otras una gran cantidad de datos.

d) A las oficinas generales corresponde realizar el procesamiento normal de rutina y proporcionar, de vez en cuando, cualquier resumen de información requerido por otras áreas. Este departamento tiene un sistema computador de usos generales, que cuenta con impresoras, varios

dispositivos de cinta. Aquí se elaboran nóminas, facturas, reportes de compras etc. También se prepara información para transmitirla a todas las demás áreas.

3. La red de comunicaciones. A diferencia de los usuarios que interactúan exclusivamente con un sistema integrado, en un sistema distribuido los usuarios interactúan uno con el otro y todos entre sí. En la mayoría de los casos, este sistema requiere cierto equipo de comunicación. Obviamente, si el sistema está formado por computadoras y todo el equipo relacionado, requerirá una red de comunicaciones más o menos complejo. En todo caso, ya sea en camión se remitan por correo o electrónicamente, la red del sistema distribuido necesitará $N(N - 1)/2$ canales de comunicación. Dicha red aparece en la figura 1.4.14. El sistema distribuido utiliza un número de canales de comunicación (personas o medios) mayor que el sistema integrado. En la figura 1.4.14 se ha puesto que son seis subsistemas que deben comunicarse entre sí con el fin de coordinar sus operaciones. Se necesitarán quince canales para manejar el envío de información: $N(N - 1)/2$, o sea $6(6 - 1)/2 = 15$.

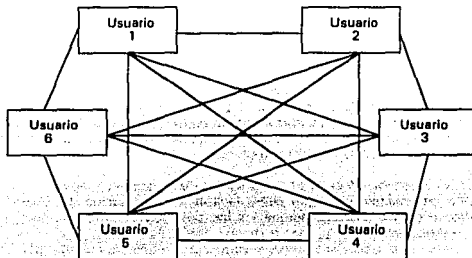


Figura 1.4.14 Red de comunicaciones entre seis subsistemas, en una empresa que aplica el enfoque de sistemas distribuidos.

4. El jefe de sistemas de información. En un sistema distribuido, el jefe de Sistemas de Información desempeña básicamente las mismas funciones que un sistema integrado; pero tendrá más hacia la coordinación. En algunas empresas, el jefe se reportará con los gerentes de departamento, de área funcional o de división, en vez de hacerlo ante el director general en otras, comparecerá ante la dirección y se reportará también con los gerentes departamentales. Algunos pensarán que no hace falta un Jefe de Sistemas de Información. Nosotros creemos que no existe esta fuerza coordinadora, ejercida a través de los analistas de sistemas, ocasionaría que el sistema distribuido volviera a ser un sistema jerárquico con procesamiento des centralizado.

5. Posición del sistema distribuido en las organizaciones. El sistema distribuido es una red de subsistemas que han sido adecuados para satisfacer las demandas de información de los departamentos o grupos usuarios. además, cada subsistema está situado físicamente dentro del área a la cual sirve.

6. Ventajas probables del enfoque de sistemas distribuidos de información. Las ventajas que se pueden derivar de la implantación de un sistema distribuido son las siguientes:

a) Su relación costo/efectividad. El desarrollo, logrado en los años recientes, de minicomputadoras electrónicas de gran capacidad, así como las telecomunicaciones, han acrecentado en gran medida la relación costo/efectividad del enfoque de sistemas distribuidos.

b) El empleo del sistema distribuido puede reducir los costos generales, relevando a la instalación central de una parte de la carga de procesamiento y disminuyendo la cantidad de datos que se debe transmitir. Sin embargo, la transmisión de datos no se reducirá si existe demasiada interacción entre los usuarios. Mientras mayor sea ésta, más probabilidades habrá de que este aspecto se convierta en una desventaja.

c) Este sistema se puede modificar con mayor facilidad cuando se trata de satisfacer las demandas del usuario.

d) Algunos intentos de conversión al sistema integrado han sido abandonados en vista del apoyo indeciso de la administración. Parece que hay más propensión a apoyar el concepto distribuido.

e) La seguridad, el respaldo, la recuperación y el control pueden manejarse con facilidad.

f) La mayoría de los sistemas distribuidos requiere una programación y una tecnología relativamente sencillos. En cambio, para una tecnología compleja y un manejo complicado del banco de datos.

g) Las demandas debidas al volumen, la complejidad, la oportunidad y las necesidades de cálculo de la empresa se pueden satisfacer con más precisión. La capacidad del sistema se desperdicia menos puesto que la distribución es más uniforme en toda organización. Debidamente equilibrada, la red no alcanzará nunca el punto de utilidades decrecientes, mientras que un sistema integrado hay una gran probabilidad de que tal cosa ocurra.

h) Resulta más sencillo, y tal vez menos costoso, procesar los datos donde se les utiliza o donde se producen, consolidando un resumen en algún punto central. La distribución del equipo y el procesamiento resulta mucho menos costosa, porque los instrumentos son más económicos, el manejo de los datos es menos complejo, y cuesta menos desarrollar un programa sencillo y distribuir un millar de copias, que crear una versión compleja y multiprogramada del mismo material, en un solo tanto, para un complejo central.

i) La descompostura de un subsistema no degradará significativamente a toda la red.

j) Es posible agregar nuevos subsistemas sin afectar a los demás.

7. Desventajas probables del enfoque de sistemas de información distribuidos, Las desventajas posibles son:

a) Hay una reducción considerable en las respuestas que se pueden proporcionar a los usuarios en línea, por lo que se refiere al acceso a todas las partes del sistema.

b) La extracción de datos congruentes, de diferentes archivos, puede ser difícil.

c) Pueden producirse inconsistencias en el sistema, dando lugar a desigualdades en la edición, en el formato y en los dispositivos de procesamiento de datos generales.

d) La coordinación de actividades puede resultar más difícil, sobre todo en casos en que los subsistemas llegan a quedar débilmente acoplados y se tornan independientes.

e) A menudo, el sistema distribuido requiere, en total, más personal calificado; por ejemplo, programadores.

f) Debido a la existencia de varios bancos de datos, habrá mayor duplicación. Por ejemplo, y ésta es una solución típica, el nombre de un cliente puede aparecer en el departamento de ventas, en el de finanzas y en las oficinas generales.

g) El sistema distribuido exige un mayor número de canales de comunicación.

Evaluación adicional de los sistemas integrado y distribuido.

Hay muchos modos de pensar en cuanto a la estructuración de los sistemas de información. Van desde el sistema totalmente integrado hasta los sub-conjuntos informativos totalmente aislados e independientes, pasando por todas las combinaciones y escalas intermedias. Para fines de análisis se ha estudiado cuatro enfoques distintos. No se apoya el enfoque jerárquico, en sus dos modalidades de procesamiento centralizado y des centralizado, por que: 1) el énfasis recae en las en las actividades de procesamiento rutinarias; 2) carece de orientación hacia las necesidades de información, y 3) el uso del enfoque de sistemas está muy limitado. Estos inconvenientes nos llevan a la pregunta ¿Cuál sistema es mejor, el integrado o el distribuido?

En realidad, el sistema distribuido es un sistema de información con cierto grado de integración. O, viéndolo de otro modo, el sistema integrado es un sistema de información con cierto grado de distribución. Ninguno es absoluto. Por otra parte, hay muchas funciones que se pueden integrar en diversos grados, y también pueden integrarse las combinaciones de funciones. El resultado es una lista interminable de variedades de integración. Para simplificar, a continuación se presenta los tipos generales de integración:

a) Integración de los datos en banco de datos. Esta integración significa que, en lugar de estar contenidos en diferentes archivos no relacionados, los datos se almacenan en forma planeada para facilitar su recuperación. La forma exacta de almacenar los datos depende de los resultados que se buscan. El almacenamiento puede hacerse con el fin de reducir la redundancia, facilitar el almacenamiento, simplificar el acceso, reducir costos, etc.

b) La integración en bancos de datos hace que la información disponible para los usuarios esté igualmente integrada. Es decir, el usuario puede obtener en una unidad información necesaria con respecto a una tarea o evento, sin que tenga que buscar una parte en un lugar y el resto en otros. Los datos dispuestos de manera tal que toda la información que el usuario necesita en relación con un tema determinado le sea presentada en una unidad, independientemente de su origen.

c) Integración de las funciones de procesamiento de datos. Con este enfoque, las aplicaciones no son ya cuestión de programas individuales computarizados. Las funciones realizadas por estos programas las realiza ahora un grupo de módulos orientados funcionalmente. Por ejemplo, la obtención de datos es manejada por un procesador de elementos de entrada común, independientemente de su origen.

d) Integración del flujo de datos. En la mayoría de las empresas hay un flujo natural de información. Por ejemplo, en una empresa manufacturera, el punto de partida puede ser el departamento de ingeniería. Luego, la corriente de información con respecto al producto pasará por las secciones de planeación, fabricación y prueba. La información financiera tendrá un flujo natural diferente. Cuando hablamos de integrar el flujo de los datos, nos referimos a la división del sistema de información de la empresa en módulos, en forma planeada, de manera que los resultados producidos por un módulo de procesamiento se puedan usar directamente como elementos de

entrada en otros módulos, además, todos los datos que se requieren corriente abajo en el procesamiento se obtienen de la fuente.

e) Integración de datos en bancos de datos e integración de funciones de procesamiento. Este sistema es una combinación de los puntos 1 y 2.

f) Integración de datos en bancos de datos e integración del flujo.

g) Integración de funciones de procesamiento e integración de procesamiento de flujo de datos.

h) Integración de los datos en bancos de datos, integración de las funciones de procesamiento e integración de flujo.

i) Integración de resultados. Esta alternativa simula un banco de datos integrado desde el punto de vista del usuario, dándole la misma capacidad de recuperación que tendría como un banco integrado. Este método puede implantarse utilizando redes integradoras, copiando los archivos o estableciendo diversos programas de recuperación.

El problema consiste en deducir si conviene inclinarse por un sistema más distribuido, con un mínimo de integración, o hacia un sistema altamente integrado. Una evaluación justa de los dos enfoques (y de las combinaciones intermedias) podría basarse en las ventajas y desventajas ya señaladas. Sin embargo, el analista de sistemas debe determinar primero las circunstancias del caso, el tipo de estructura de la empresa y los objetivos de la administración, para evaluar luego las ventajas y desventajas.

A parte de la comparación de las ventajas y desventajas que se presentan los sistemas integrado y distribuido, las dos consideraciones predominantes son las siguientes: 1) la forma en que la administración desea manejar la empresa, y 2) el nivel de diversificación. Por ejemplo, si una unidad integrada, el sistema de información deberá estar tan integrado como sea posible. En cambio, si desea manejarla a base de unidades funcionales separadas, se da preferencia al sistema distribuido.

El nivel de diversificación inherente a ciertas organizaciones también tiene a ciertas organizaciones también tiene estrecha relación con la manera de estructurar el sistema de información. Por ejemplo, en una gran empresa manufacturera, como las compañías aerospaciales, existe una gran diversificación entre las distintas áreas funcionales. Esa situación existe no porque dichas áreas funcionales sean realmente independientes desde el punto de vista de los sistemas, sino porque los vínculos y relaciones son tan complejos y vagos que un sistema de información integrado con capacidad para apoyarlas a todas sobrepasaría las posibilidades de la actual tecnología de sistemas. Un sistema diseñado para este tipo de organización tendría que ser distribuido, con varios sub-niveles de integración.

Las empresas aéreas comerciales, por su parte, llevan sus operaciones en forma bastante unificada. La administración general, la programación y el mantenimiento giran alrededor del sistema de reservaciones. Por lo tanto, a juzgar por las interdependencias y puntos comunes que se observan en estas áreas, parece que el sistema altamente integrado resultaría apropiado.

Capítulo II

ANÁLISIS DEL SISTEMA MEDIANTE MODELO E-R

Introducción

El análisis de sistemas consiste en separar las partes que constituyen un sistema, con el fin de estudiarlas y evaluarlas para ver si hay un mejor método que satisfaga las necesidades de la administración. En este caso el sistema y el análisis ya está desarrollado, pero aún así en este capítulo se estudiará y explicará lo que es el análisis para entender el sistema desarrollado. La metodología usada es de el modelo entidad-relación, pero hay otras las cuales se dará una pequeña introducción de su funcionamiento en el desarrollo de sistemas y se describirá su estructura particular. Principalmente se estudiará el modelo entidad-relación que es por el cual se desarrollo el sistema y representado conceptualmente, además de que es ampliamente usado en la actualidad para desarrollar sistemas de información.

El modelo entidad-relación es representativo de los modelos lógicos basados en objetos. Hay otros con diferentes características de aplicación, según el tipo de relación de archivos, su semántica y sus limitaciones.

El SNAP lo utiliza el como metodología de desarrollo de generación de sistemas de información por lo cual se dará una amplia información respecto a este modelo.

Para entender este método de desarrollo su estudiarán brevemente los modelos de datos y sus herramientas que son utilizadas para su apoyo como los conceptos de las entidades atributos, relaciones entre entidades etc.

Por último se mostrará el diseño que muestra las relaciones existentes entre las diferentes entidades y una explicación general de cada una de estas en sistema COP (Control de Proyectos).

2.1 Modelo de datos

El modelo de datos representa el primer paso dentro de la metodología de desarrollo de sistemas de información por lo cual el SNAP contiene una sección especial para la inserción de los modelos ya realizados. Dentro de este ambiente de trabajo se agrupan todas las herramientas necesarias para definir, documentar y generar la representación conceptual de la empresa u organización. El modelo ya está realizado por lo que solamente se estudiará las entidades o archivos y sus relaciones comunes a través de el diagrama conceptual ya que el sistema no tuvo modificaciones en su estructura, sino correcciones a algunos programas.

2.1.1 Modelos de Datos

Para describir la estructura de una base de datos es necesario definir el modelo de datos. La cual agrupa herramientas conceptuales para describir los datos, sus relaciones, su semántica y sus limitantes. Se han propuesto varios modelos de datos, los cuales pueden dividirse en tres grupos: los modelos lógicos basados en objetos y registros, y los modelos físicos de datos.

2.1.2 Modelos Lógicos Basados en Objetos

Los modelos lógicos basados en objetos se utilizan para describir los datos en los niveles conceptual y de visión. Se caracterizan por el hecho de que permiten una estructuración bastante flexible y hacen posible especificar claramente las limitaciones de los datos. Existen aproximadamente 30 modelos distintos de este tipo, y es probable que aparezcan más. Algunos de los más conocidos son:

- 1) El modelo entidad-relación
- 2) El modelo binario
- 3) El modelo semántico de datos
- 4) El modelo infológico

Se ha escogido el modelo entidad-relación como representativo de la clase de modelos lógicos basados en objetos porque ha tenido bastante aceptación para representar el modelo de la base de datos de sistemas de información y como ya se dijo anteriormente es utilizado por el SNAP.

El modelo de datos entidad-relación se basa en una percepción de un mundo real que consiste en un conjunto de objetos básicos llamados entidades, y las relaciones entre estos objetos. Una entidad es un objeto que existe y puede distinguirse de otros. La distinción se logra asociando a cada entidad un conjunto de atributos que describen al objeto. Por ejemplo, los atributos número y saldo describe una cuenta específica en un banco. Una relación es una asociación entre varias entidades. Por ejemplo, una relación Cliente asocia a un cliente con cada una de las cuentas que tiene. el conjunto de todas las entidades y relaciones, respectivamente.

Además de entidades y relaciones, el modelo E-R representa ciertas limitantes que debe cumplir el contenido de la base de datos. Una de estas limitantes importantes es la cardinalidad de mapeo, que expresa el número de entidades con las que puede asociarse otra entidad por medio de un conjunto de relaciones.

La estructura lógica general de una base de datos puede expresarse gráficamente por medio de un diagrama E-R que consta de los siguientes componentes:

- 1) Rectángulos, que representan conjuntos de entidades
- 2) Elipse, que representan atributos.
- 3) Rombos que representan relaciones entre conjuntos de entidades.
- 4) Líneas, que conectan los atributos a los conjuntos de entidades y los conjuntos de entidades a las relaciones

Cada componente se etiqueta con el nombre correspondiente, mas adelante se dan ejemplos de notaciones empleadas para representar este tipo de modelo.

2.1.3 Modelos Lógicos Basados en Registros.

Los modelos lógicos basados en registros se utilizan para describir los datos en los niveles conceptual de visión. A diferencia de los modelos de datos basados en objetos, estos modelos sirven para especificar tanto la estructura lógica general de la base de datos como una descripción en un nivel más alto de la implantación. Sin embargo, no permiten especificar en forma clara las limitaciones de los datos que han tenido la más amplia aceptación:

Modelo relacional. Los datos y las relaciones entre los datos se representan por medio de una serie de tablas, cada una de las cuales tiene varias columnas con nombres únicos. Para dar un ejemplo, piénsese en una base de datos que incluye a los clientes y sus cuentas. Una muestra de una base de datos relacional se presenta en la siguiente figura:

Nombre	Calle	Ciudad	Número
Héctor	Roble	México	900
Lucía	Rosal	México	56
Julio	Collma	Toluca	74
Irene	Juarez	Toluca	12

Número	Saldo
900	845
56	456
12	45
74	563

Ejemplo de una base de datos relacional

El modelo jerárquico es similar al modelo de red en cuanto a que los datos y las relaciones entre los datos se representan por medio de registros y ligas, respectivamente. El modelo jerárquico difiere del de red en que los registros están organizados como conjunto de árboles en vez de gráficas arbitrarias.

2.1.4 Modelos físicos de los datos

Los modelos físicos de los datos sirven para describir los datos en el nivel más bajo. A diferencia de los modelos lógicos de los datos, son muy pocos los modelos físicos utilizados. Algunos de los más conocidos son:

- 1) El modelo unificador
- 2) La memoria de cuadros

Los modelos físicos de los datos capturan aspectos de la implantación de los sistemas de la base de datos.

2.2 Instancias y esquemas

Las bases de datos cambian con el tiempo al insertarse información en la base de datos y eliminarse de ella. El conjunto de información almacenado en la base de datos. El diseño general de la base de datos se llama esquema de la base de datos, los esquemas se alteran muy rara vez o nunca.

2.2.1 Independencia de los datos

La capacidad de modificar una definición de esquema en un nivel sin afectar la definición de esquema en el nivel inmediato superior se denomina independencia de los datos. Existen dos niveles de tal independencia:

Independencia física, que es la capacidad de modificar el esquema físico sin obligar a que se vuelvan a escribir los programas de aplicaciones. En algunas ocasiones son necesarias modificaciones en el nivel físico para mejorar el rendimiento.

Independencia lógica, que es la capacidad de modificar el esquema conceptual sin obligar a que se vuelvan a escribir los programas de aplicaciones. Las modificaciones a nivel conceptual son necesarias siempre que se altera la estructura lógica de la base de datos (por ejemplo, el agregar cuentas de mercado de valores a un sistema bancario).

La independencia lógica de los datos es más difícil de lograr que la independencia física, ya que los programas de aplicaciones dependen en alto grado de la estructura lógica de los datos que se tienen acceso.

2.2.2 Lenguaje de Definición de Datos

Un esquema de base de datos se especifica por medio de una serie de definiciones que se expresan en un lenguaje especial llamado lenguaje de definición de datos (en inglés: DDL, data definition language). El resultado de la compilación de las proposiciones DDL es un conjunto de tablas que se almacenan en un archivo especial llamado diccionario (o directorio) de datos.

Un directorio de datos es un archivo que contiene mandatos, es decir, "datos acerca de los datos". Este archivo se consulta antes de leer o modificar los datos reales en el sistema de base de datos.

La estructura de almacenamiento y los métodos de acceso empleados por el sistema de base de datos especifican por medio de un conjunto de definiciones de un tipo especial DDL llamado lenguaje y definición de los datos. El resultado de la compilación de estas definiciones es una serie de instrucciones que especifican los detalles de implantación de los esquemas de base de datos que normalmente no pueden ver los usuarios.

El SNAP, posee toda una metodología fundamentada en una serie de filosofías actuales de desarrollo estructurado de sistemas, cuya finalidad primordial es obtener una alta productividad tanto en la fase de desarrollo y de en la de mantenimiento. Bajo esta visión integrada de desarrollo, es necesario tener en cuenta que cada definición que se incluya en el repositorio central del SNAP va a ser re utilizada en cualesquiera de las siguientes etapas de desarrollo, y a la vez, van a convertirse en la base sobre la que se apoya toda la metodología de desarrollo del SNAP.

**ESTA TESIS NO DEBE
SALIR DE LA BIBLIOTECA**

2.3 Metodología de Diseño de Base de Datos

El modelaje de la información nace como una necesidad derivada del proceso de diseño de la base de datos.

2.3.1 Definición de la base de datos.

El proceso de diseño de una base de datos se compone de tres niveles de definición y de un nivel físico donde se apoyan los restantes. A continuación se explicará cada uno de estos niveles.

1) El nivel físico. El nivel físico es donde se establecen todas las características físicas de los datos que van a ser almacenados en los dispositivos físicos de almacenamiento. Se trabaja a nivel de bytes, direcciones físicas, cilindros, capacidades, algoritmos de acceso generalmente implantados en el hardware o firmware (circuitería o micro código), etc. En la mayoría de los sistemas, este nivel es tarea del sistema operativo.

2) El nivel interno. Este nivel es donde se define la estructura de almacenamiento de los datos y la estrategia de acceso a éstos. Se trabaja a un nivel de campo, registro, archivo, índice, estructura de datos, direcciones lógicas, algoritmos de acceso generalmente implantados en software, etc. Esta definición se conoce como esquema interno.

3) El nivel conceptual. Es donde se definen las características y relaciones lógicas entre los datos que se desean almacenar. Se trabaja a nivel de campo, entidad, relación, vía de acceso, etc. Esta definición se conoce como esquema conceptual.

4) El nivel externo. Es donde se definen las características de los datos almacenados tal y como las ve el usuario exterior; por eso se le conoce como vista lógica o vista externa. Se trabaja a nivel de información. En muchos sistemas administradores de base de datos no se hace diferencia entre los dos últimos niveles, identificando el nivel externo con el nivel conceptual. En este caso cada vista es una forma diferente de manejar la misma información. Por otro lado, si se tiene que definir por aparte este nivel, entonces cada vista será una parte determinada o sub-conjunto de la totalidad del esquema conceptual, y cada definición se conocerá como sub-esquema.

2.3.2 Diseños Tradicionales

Tradicionalmente, la definición de la base de datos dependía directamente del enfoque o modelo de la estructura de almacenamiento (nivel interno) de la base de datos. De este modo, al diseñar la definición conceptual se tenía que tomar en cuenta su arquitectura interna. Las arquitecturas más utilizadas son: relacional, de redes y jerárquica como se menciono anteriormente.

Generalmente, estos sistemas proveen un Lenguaje de Definición de Datos como se mencionó anteriormente, con el fin de poder definir cada uno de los niveles descritos anteriormente. El administrador de la base de datos se da a la tarea de definir cada uno de los niveles anteriormente descritos, primero el nivel conceptual, luego el nivel interno, y si es necesario, terminar con el nivel exterior.

ESTRUCTURA DE LA BASE DE DATOS

Además, algunos lenguajes de definición de datos permiten que a nivel exterior el usuario maneje la base de datos de manera diferente a como realmente está organizada físicamente, por ejemplo, existen sistemas de redes que permiten definir un nivel exterior relacional, pero su arquitectura interna sigue siendo de redes.

2.3.3. Enfoque Entidad-Relación

Con la aparición de las teorías de bases de datos semánticas y las herramientas de modelamiento de información se ha conseguido independizar la definición conceptual de los datos y la arquitectura física de los mismos.

Dentro de este tipo de herramientas de modelamiento de información una técnica muy popular es el modelo Entidad-Relación, que fue desarrollado por Peter Chen a partir de 1976 y que se ha ido perfeccionando a través de los años. Este modelo representa toda una filosofía de desarrollo y se basa en la percepción de un mundo compuesto por objetos (entidades) con características propias (atributos) y de asociaciones entre esos objetos (relaciones).

2.3.4 Aparición de la Herramienta CASE

Este enfoque Entidad-Relación para el modelamiento de la información tuvo un enorme enriquecimiento con la aparición de las herramientas CASE, sobre todo con las gráficas.

Las herramientas gráficas facilitaron las tareas de análisis y diseño de la base de datos, mientras que los modelos de definición, los generadores de aplicaciones y prototipos, y los lenguajes de cuarta generación aceleraron las labores de construcción de los sistemas. En ambos casos, la herramienta CASE controla la definición y manipulación del modelo, y permite total documentación del sistema.

2.3.5 Elementos del Modelo de Datos

Como se apuntó anteriormente, el modelo de datos se compone de tres elementos básicos, las entidades, los atributos y las relaciones entre entidades. En seguida se presentan las definiciones y características de estos elementos.

2.3.5.1 Entidades

Una entidad es cualquier objeto del mundo real que tiene vida propia o que puede distinguirse de otros objetos del mismo tipo, y del cual se desea guardar información. Puede ser una representación de algo concreto como un auto o una persona, o de algo abstracto como un concepto o una fecha.

El tipo de entidad es el conjunto de características de todas las entidades del mismo tipo, es decir, es la definición de la entidad (o la conceptualización de la entidad) dentro del modelo de datos. De esta manera, el conjunto de características de todos los automóviles se define por el tipo de entidad "Automóvil", el conjunto de características de todas las personas va a estar definido por el tipo de entidad "Persona".

Nota: Para efectos prácticos, en este capítulo se va a identificar el concepto de entidad con el tipo de entidad.

Una instancia o ocurrencia de la entidad son los datos o valores específicos para una entidad. Por ejemplo, para el tipo de entidad "Persona" una ocurrencia será todos los datos de una persona específica, su nombre, "Jorge", su fecha de nacimiento "1 de Enero de 1990", su edad "2 años", etc.

Generalmente en un sistema administrador de base de datos relacional como del AS/400, una entidad se convierte físicamente en un archivo, y una ocurrencia representa un registro dentro de ese archivo.

Dentro del proceso de modelamiento de información, se encuentran tres tipos diferentes de entidades: 1) entidades fundamentales, 2) entidades atributivas y 3) entidades asociativas.

2.3.5.2 Entidad Fundamental

Una entidad fundamental es un objeto independiente con vida propia dentro del sistema, es decir, que no depende de ninguna otra entidad, sin embargo, sí puede relacionarse con cualquier entidad (simple que no sea una relación de dependencia), dentro de un sistema de facturación, las siguientes pueden ser entidades fundamentales: "Artículo", "Cliente", "Bodega", "Factura", etc.

2.3.5.3 Entidad Atributiva

Una entidad atributiva se debe crear cuando existe un sub-grupo de datos opcionales o de un sub-grupo de datos repetitivos que dependen completamente de la existencia de una ocurrencia de otra entidad o entidad principal.

Un sub-grupo de datos opcionales son aquellos atributos que no son necesarios en varias o muchas ocurrencias de la entidad principal permitiendo valores nulos, o pueden definirse en una nueva entidad atributiva conocida como sub-entidad atributiva o sub-entidad dependiente (o simplemente sub-entidad). Por ejemplo, en un sistema de personal, una sub-entidad atributiva puede ser "Datos de maternidad", la cual contiene información propia de maternidad (número de hijos, partos etc.) que sólo debe almacenarse para aquellas empleadas que son madres. Estos atributos pudieron ser definidos de la entidad "Empleado" permitiendo si generalmente van a tener o no valores en la entidad. Si generalmente van a contener valores nulos (no van a contener valores) entonces se deben incluir en una sub-entidad dependiente.

Un sub-grupo de datos repetitivos son aquellos atributos que pueden asumir muchos valores para una misma ocurrencia de la entidad, estos atributos deben definirse en una nueva entidad atributiva conocida como entidad dependiente o entidad detalle. Por ejemplo, una entidad dependiente es el caso de "Línea de factura" ya que una misma factura tiene varias o múltiples líneas de factura (atributos repetitivos), "Línea de factura" es el componente o el detalle de "Factura".

2.3.5.4 Entidad Asociativa

Una entidad asociativa es básicamente una relación de la cual se desea almacenar información. Una relación se debe convertir a una entidad cuando se desea definir atributos propios de la relación (atributos que no pertenecen a ninguna de las entidades a relacionar) y principalmente cuando se definen relaciones de muchos-amuchos. En nuestro ejemplo, si un mismo artículo se puede almacenar en varias bodegas a la vez y una bodega puede almacenar varios artículos, entonces se debe definir una nueva entidad asociativa "Artículo en bodega" para poder relacionar las entidades "Artículo" y "Bodega", además, puede definir un atributo propio de esa relación, como es la localización de ese artículo en esa bodega.

2.3.5.5 Notaciones gráficas de entidades

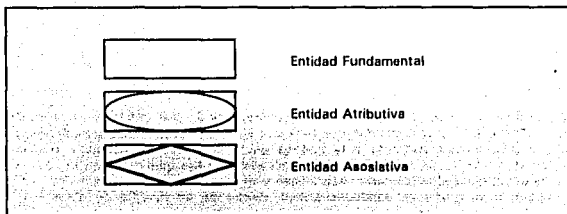


Figura 2.0.1 Notación gráfica para entidades

Muchas herramientas CASE gráficas han adoptado diferentes notaciones. Por ejemplo, las entidades se representan por medio de rectángulos (a veces lo hace a través de óvalos), la anterior figura representa la notación que se utilizará en este capítulo y que corresponde a la notación utilizada por el VAW (Visible Analyst Workbench).

2.3.5.6 Atributo

Los atributos son las propiedades que caracterizan cada entidad, y representa la información que se desea almacenar con respecto a la entidad a la cual pertenecen. Por ejemplo, en una entidad "Artículo" algunos posibles atributos son: "Nombre del artículo", "Código del Artículo", etc.

Los atributos no sólo permiten mantener la información importante respecto a cada entidad, sino que también permiten identificar una ocurrencia específica en forma unívoca dentro de la entidad.

Volviendo al ejemplo, el "Código del artículo" permite distinguir un artículo de otro artículo ya que es único para cada uno.

Generalmente cada atributo se va a convertir en un campo del archivo de la entidad a la que pertenece.

2.3.5.7 Dominio del atributo

El dominio del atributo determina las características de los valores válidos que puede almacenar, establece el tipo o formato físico del dato, es decir, si permite valores numéricos o alfanuméricos, también establece un rango de valores o un conjunto de valores permitidos, y puede establecer una verificación especial.

Por ejemplo, el "Nombre del artículo" puede definirse el siguiente dominio: Dato alfanumérico de 30 caracteres que permita tanto mayúsculas como minúsculas. O bien, Precio Unitario" como: Dato numérico de 5 enteros con 2 decimales, mayor o igual que 100,000.

2.3.5.8 Relación

Una relación establece las reglas entre las ocurrencias de dos entidades. Como ocurre en el mundo real los objetos (entidades) cuentan con toda una serie de asociaciones (relaciones) entre ellos, por ejemplo, un artículo está asociado a la bodega en la cual está almacenado, una factura está relacionada con el cliente respectivo, etc. Estas relaciones son las que permiten la existencia de un sistema real, es decir, que todas las partes interactúen y afecten al todo. Por lo anterior, nunca debería existir una entidad o un grupo de entidades aislado dentro de un modelo de datos, ya que al no interactuar con otras partes no pertenece directamente al sistema total.

La cardinalidad de la relación determina el tipo de asociación entre las ocurrencias de dos entidades. Esta permite que existan relaciones de uno-a-uno, de uno-a-muchos (o muchos-a-uno) y de muchos-a-muchos.

Generalmente, en un sistema relacional las asociaciones entre entidades se convierten en campos de relación llaves foráneas.

2.3.5.9 Relación uno-a-uno

Una relación de uno-a-uno se define cuando una ocurrencia de una entidad se relaciona tan sólo con una ocurrencia de la otra entidad, y viceversa.

Estas relaciones son muy útiles cuando se tiene una serie de atributos opcionales dentro de una entidad, es decir, cuando varias o muchas ocurrencias de la entidad no necesitan de esos datos por no conocerse o porque son inaplicables a ellas. En este caso, esos atributos se pueden eliminar de la entidad original, creando una nueva sub-entidad dependiente que se le relacione de uno-a-uno. Este es el

ejemplo de "Datos de Maternidad" y "Empleado" explicados cuando se definió sub-entidad atributiva. Las subentidades atributivas siempre se relacionan de uno-a-uno con la entidad principal.

2.3.5.10 Relación uno-a-muchos

Una relación de uno-a-muchos se define cuando una ocurrencia de una entidad se relaciona con muchas ocurrencias de la otra entidad, pero una ocurrencia de ésta tan sólo con una ocurrencia de la primera.

Estas relaciones se utilizan para asociar entidades dependientes con la entidad principal, como por ejemplo "Línea de Factura".

También se utiliza simplemente para asociar dos entidades independientes con el fin de ampliar la información de una de las entidades, por ejemplo, "Factura" con "Cliente", donde una factura sólo pertenece a un cliente pero un cliente puede tener varias facturas, siendo ambas independientes entre sí. Esta relación permite que la "Factura" pueda utilizar la información del "Cliente" y evitar así la redundancia.

Nota: Las relaciones de "uno-a-muchos" y de "muchos-a-uno" en una entidad son idénticas, sólo difieren en la dirección en que se vean. Sin embargo, en este capítulo se seguirá hablando de relaciones de "muchos-a-uno" debido a que la relación se debe definir en la entidad "muchos" hacia la entidad "uno".

2.3.5.11 Relación muchos-a-muchos

Una relación de muchos-a-muchos se define cuando una ocurrencia se relaciona con muchas ocurrencias de la entidad, y viceversa. Sin embargo, este tipo de relación no se puede definir directamente, sino a través de una nueva entidad de relación, la cual se puede relacionar de muchos-a-muchos con las diferentes entidades a relacionar, como se explicó con "Artículo" y "Bodega" en la definición de entidades asociativas.

2.3.5.12 Notaciones Gráficas Según Herramientas CASE

Quando se indico con notaciones utilizadas en entidades, las diferentes herramientas CASE gráficas han adoptado diferentes notaciones. Por lo general, las relaciones se presentan por líneas (a veces por rombos).

Nota: En este capítulo se seguirá utilizando la segunda notación (dobles flechas).

Como se ve en la figura 2.1.2, en la de dobles flechas la cardinalidad se determina por el número de flechas de la siguiente manera: la flecha simple significa la entidad "uno" y la de doble flecha indica la entidad "muchos".

La relación debe definirse con la entidad menos importante, o sea, en la entidad subordinada. En relaciones muchos-a-uno se debe de definir en la entidad "muchos" y en las relaciones uno-a-uno en la entidad dependiente.

2.3.5.13 Redundancia

La redundancia es una repetición de la misma información en dos o más lugares diferentes. Esta repetición puede causar errores de inconsistencias si se actualiza esa información en un lugar y no en otro. Por esta razón, es necesario que toda redundancia sea controlada, es decir, que permanezca consistente en todo momento.

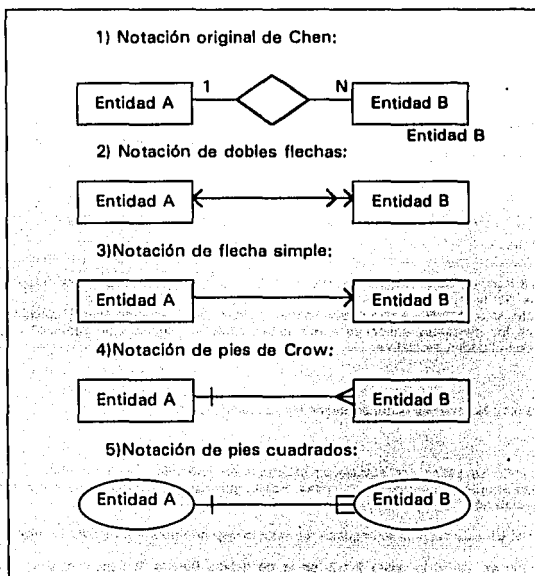


Figura 2.1.2 Notaciones gráficas para Relaciones de "uno" a " muchos"

2.3.514 Redundancia Necesaria

Existe redundancia necesaria cuando se debe repetir un dato en diferentes entidades para que el modelo funcione correctamente, éste es el caso de las llaves foráneas y las llaves primarias de un sistema relacional, por ejemplo, cuando se incluye el Código de Artículo" en la línea de factura. Sin embargo, esta redundancia debe ser controlada para mantener la integridad de la base de datos.

2.3.5.15 Redundancia Innecesaria

La redundancia innecesaria aparece cuando se define en forma incorrecta el modelo de datos, y se encuentran atributos diferentes que en realidad deben almacenar la misma información, por ejemplo, cuando se define el nombre del cliente tanto en la entidad de "Cliente" como en la entidad "Factura". Este tipo de redundancia debe evitarse o ser controlada por el programador.

2.3.5.16 Normalización

La normalización es una técnica matemática desarrollada a partir del año 70 por el Dr. E.F. Código, con el objeto de asegurar la consistencia interna de la base de datos en los sistemas de arquitectura relacional por medio de la eliminación (o máxima reducción) de toda la redundancia no necesaria.

En resumen, una entidad está normalizada si, y sólo si, no posee atributos repetitivos, y cada campo no-llave es mutuamente independiente pero depende directamente de una sola llave primaria.

Dentro del modelo entidad-relación la normalización se consigue fácilmente, sobre todo porque los atributos que se agregan a una entidad, por definición, son propios de ella misma y no de otras entidades, eliminando así en gran parte la redundancia innecesaria; por esta razón, no se deben definir atributos en una entidad que no pertenezcan directamente a ella, y para relacionar dos entidades se hace a través de la definición de una relación y no de un atributo común. La eliminación de atributos repetitivos se obtiene creando una nueva entidad dependiente, con el atributo a repetir, que se relacione de muchos-a-uno con la entidad original.

2.3.5.17 Campos llave

Una llave es un campo o una combinación de campos cuya información tiene una función especial dentro del archivo que pertenece.

2.3.5.18 Llave Primaria

Una llave primaria es un campo o una combinación de campos que identifican en forma única cada ocurrencia de la entidad, y por tanto, no habrá más de una ocurrencia con los mismos valores, es decir, no tendrá duplicados, ni tampoco podrá tener un valor nulo.

Sin embargo, es posible que dentro de una entidad haya diferentes campos o conjuntos de campos que cumplan con la definición anterior, en este caso, esos campos se conocen como llaves candidatas (candidatas a convertirse en llave primaria), pero sólo uno de ellos se convertirá en llave primaria, aquel más conveniente por su menor tamaño o porque me asegura que no existirán dos llaves repetidas, etc.

Por ejemplo, dentro de la entidad "Cliente" puede existir un "Número de asegurado", el cual es único para cada cliente, sin embargo, es improbable que la empresa pueda recolectar todos los números de asegurado para sus clientes, del mismo modo, existe la posibilidad de que más de un cliente posea el mismo nombre y aún cuando nos aseguremos que si lo sea este será muy grande y por tanto incómodo, por lo que es más factible asignarle a cada cliente un código que lo identifique. "Código de cliente" será la llave primaria de la entidad "Cliente".

En el caso de "Línea de Factura" la llave primaria se establece por el "Número de Factura" relacionada junto con un "Número de Línea" que la identifique.

En un modelo entidad-relación la llave primaria se genera a partir de las relaciones de dependencia junto con los atributos de identificación.

2.3.5.19 Llave Foránea

La llave foránea se define por el campo o la combinación de campos que permiten la relación entre entidades en un sistema relacional, es decir, son los campos comunes en ambas entidades que hacen posible la relación. La llave foránea equivale a la llave primaria de la entidad relacionada, pero eso no significa que necesariamente sea llave primaria de la entidad en la que se define la relación. Las llaves foráneas producen datos redundantes pero necesarios en la base de datos, es decir, representan la redundancia necesaria.

Importante: En un modelo entidad-relación no se definen llaves foráneas, debido a que éstas se generan a partir de las relaciones que sí se deben definir, esto lo hace el modelo automáticamente importando o sincronizando los campos de la llave primaria de la entidad relacionada dentro de la entidad donde se define la relación, el modelo asegura el control de este tipo de redundancia. En el caso de "Factura", cuando se define la relación a "Cliente" automáticamente el modelo importa a la entidad "Factura" la llave primaria de la entidad "Cliente", es decir, que le agrega el campo "Código de Cliente".

2.3.5.20 Llave de Acceso o Llave de Secuencia

Una llave de acceso es un campo o combinación de campos que permiten localizar una ocurrencia dentro de la entidad a través de la información que contienen. No necesariamente una llave

de acceso es única, es decir, puede permitir duplicados; al contrario de las llaves primarias, que son llaves de acceso pero sin duplicados.

Una llave es un campo o una combinación de campos que determinan el orden en que se organizan las ocurrencias dentro de la entidad, con base en la información que contienen. No necesariamente el orden establecido debe ser ascendente, también puede ser descendente o en algún orden en especial, como por ejemplo, primero en salir, o último en entrar primero en salir, etc.

Generalmente, en un sistema relacional, la llave de acceso y la llave de secuencia son iguales, y se definen dentro de un índice.

2.3.5.21 Nivel Externo o Vista Lógica

Como se definió al principio de este capítulo, el nivel o vista externa es la forma cómo el usuario o programador puede ver y manejar la información de la base de datos.

En un modelo de entidad-relación, donde las definiciones hacen al nivel conceptual, una vista externa es una forma particular como se puede manejar la información de una entidad. Una vista se compone de la vía de acceso a la entidad, de los atributos y relaciones que la componen, y de la seguridad que permite o no accederla por unos u otros usuarios.

2.3.5.22 Vía de Acceso

La vía de acceso establece el orden a través de un campo o combinación de campos conocido como llave de secuencia. A través de esta llave también se puede acceder en forma directa un registro con un valor específico, y puede validarse automáticamente la unicidad de cada ocurrencia. También puede establecerse un conjunto de registros a incluirse o excluirse con base de un valor de uno o varios campos. La vía de acceso principal equivale a ordenar todos los registros a través de la llave primaria.

Por ejemplo, una vista para "Cliente" será ordenado por "Código de cliente" sin permitir duplicados, y otra será ordenado por "Nombre de cliente" en forma descendente omitiendo a aquellos que no tengan tope de crédito.

2.3.5.23 Entidad Lógica

Una vista externa también puede tener parte de los atributos o relaciones de la entidad o compartir atributos de las entidades relacionadas. En este caso, el usuario ve a la entidad con una estructura diferente a la definición original (cómo si fuera otra entidad), por lo que se le conoce como entidad lógica. Por ejemplo, puede existir una vista que permita modificar el nombre del cliente pero excluye el tope de crédito para no poderlo modificar.

2.3.5.24 Seguridad

La seguridad de la vista de la entidad establece cuales usuarios pueden consultar la entidad y cuáles la pueden actualizar.

2.3.5.25 Integridad de la Base de Datos

La integridad de la base de datos tiene como objetivo que la información contenida en ella sea correcta en todo momento. Para ello debe respetar las siguientes reglas o restricciones de integridad:

Regla de redundancia

Se debe eliminar la redundancia innecesaria y controlar la redundancia necesaria.

Regla de Integridad Referencial (relaciones)

Toda llave foránea (campos de relación) debe contener en todo momento un valor existente como llave primaria en la entidad relacionada (control de la redundancia necesaria), o bien, puede contener un valor nulo. Por esta razón, un registro no puede eliminarse mientras exista alguna relación con él.

Regla de Integridad de Dominio

Se debe controlar otros tipos de restricciones que tienen que ver con los valores permitidos para cada atributo, en otras palabras, son condiciones especiales de verificación o validación sobre los valores de los campos. El dominio de valores permitidos para un atributo se le conoce como dominio del atributo.

2.4. Modelo de datos en SNAP

El modelo de datos se trabaja a través de una estructura lógica y normalizada de listas de opciones, lo cual le permite definirlo rápida y ordenadamente; además la normalización permite que se aprenda en poco tiempo el SNAP, cual se eleva la productividad considerablemente.

En SNAP se puede establecer, dentro de un ambiente integrado, toda una estructura de datos lógica, customizada con una gran variedad de aspectos físicos y de documentación que redundan en la productividad. Todas las definiciones que se realizan en forma conceptual son luego traducidas por SNAP automáticamente a definiciones físicas nativas del AS-400, sin requerir ningún esfuerzo de programación o digitación adicional.

En el capítulo IV se explica como usar estas ventajas que son proporcionadas por el SNAP.

2.4.1 Concepto de Sistema en SNAP

Tradicionalmente los sistemas se han definido como aplicaciones cuya información esta aislada a la de los demás sistemas, y muchas veces, aunque necesiten compartir datos, este aislamiento provoca que exista redundancia incontrolable. En SNAP el concepto de sistema es mas amplio y ajustado a la realidad.

Un sistema SNAP es el conjunto independiente de definiciones, datos y programas que se interrelacionan para desarrollar una o varias aplicaciones. Bajo este concepto, un sistema representa la integración de toda la información de una organización o empresa, de manera que una aplicación puede compartir cualquier definición, dato o programa de otra aplicaron, eliminando así cualquier nivel de redundancia Innecesaria.

Recomendación: Se recomienda desarrollar todas las aplicaciones para una empresa u organización en un solo sistema SNAP, y organizar cada aplicación dentro de un subsistema diferente.

2.4.2 Subsistema

Además dentro de cada sistema SNAP se definen los sub-sistemas que interactuan con un sub-conjunto específico de una sección o un departamento, lo cual es análogo al concepto tradicional de sistema.

Por ejemplo, en una organización podemos tener el sistema total que contiene todas las entidades que representan la información del subsistema de control de activos, subsistema de facturación, etc.

Como es lógico los sub-sistemas comparten cierta información, por ejemplo, el subsistema de contabilidad necesita información del subsistema de control de activos y del subsistema de facturación. En SNAP lo anterior no ofrece ningún obstáculo para mantener el sistema de información integrado, ya que la información es compartida por todos los sub-sistemas de un mismo sistema.

En SNAP tanto las entidades como los programas están asociados a un subsistema específico para permitir un trabajo mas ordenado y organizado. Lo anterior permite que se trabaje únicamente, si así lo desea, con las entidades relacionadas con un subsistema para agilizar en una aplicación específica, o bien, se puede trabajar con todas las entidades del sistema.

Recomendación: Se recomienda definir las entidades y programas comunes para todas las aplicaciones

del sistema en el subsistema "BASE", de manera que no esté asociado a ninguna aplicación específica. este es el caso del programa base para los programas automáticos.

Como se habrá podido notar para trabajar en SNAP, el primer paso es definir el sistema y el subsistema específico sobre el que se trabajará. Estos pasos son muy sencillos y se explican detalladamente en el desarrollo del sistema.

2.4.3 Enfoque Entidad-Relación en SNAP

El modelo de datos en SNAP se basa en el enfoque entidad-relación definido en la sección anterior. De esta manera, cuando se define el modelo se realiza a través de entidades, atributos y relaciones. El SNAP, con base en las relaciones definidas, se encarga de sincronizar todas las llaves foráneas necesarias y de controlar la integridad de la base de datos.

2.4.4 Concepto de Entidad en SNAP

Una entidad en SNAP es la definición de un objeto del mundo real del cual se desea almacenar información. Para ello, la entidad se compone de atributos y relaciones con otras entidades.

2.4.5 Concepto de atributos en SNAP

Los atributos son las características propias que se desean almacenar en una entidad.

Atributos de identificación

Los atributos de identificación, conocidos con la abreviatura "AID", son aquellos atributos dentro de la entidad que sirven para identificar, localizar y ordenar cada una de las ocurrencias, y por tanto, forman parte de su llave primaria (la llave primaria se forma por los campos de la dependencia junto con los atributos de identificación).

Atributos Generales

Los atributos generales, conocidos con la abreviatura "ATR", son aquellos atributos dentro de la entidad que sirven para describirla, es decir, para almacenar la restante información.

Atributos Lógicos

Los atributos lógicos, conocidos con la abreviatura "LOG" y que sólo pertenecen a entidades lógicas, son aquellos atributos derivados de otros, y sirven para extraer parte o concatenar otros atributos de la entidad.

Dominio del Atributo

El dominio en SNAP, se conoce como tipo de dato, el cual se define en los parámetros de instalación, y sirve para definir el formato físico de los atributos, longitud, decimales, códigos de edición, aceptación de minúsculas o negativos, y rutinas de validación.

2.4.8 Concepto de Relación en SNAP

Las relaciones son establecidas en SNAP por medio de dos tipos de asociaciones: dependencias y referencias entre ellas. Por medio de las referencias y dependencias se pueden establecer todas las diversas relaciones que pueden existir entre dos o más entidades. En la siguiente sección se explicará en detalle estas relaciones.

Tipos de Relación

En SNAP las relaciones establecen las asociaciones entre dos entidades, y deben ser definidas en la entidad "muchos" en las relaciones de uno-a-muchos, o en la entidad dependiente en relaciones de uno-a-uno.

En SNAP automáticamente importa todos los campos llave correspondiente a la entidad relacionada en la entidad donde se define la relación, incluyéndolos como campos de esta entidad, y los transforma o no en campos llaves en esta entidad según sea el tipo de relación.

Relación de Dependencia

Una entidad es dependiente de otra si, por su naturaleza forma parte o es un componente de ésta otra, es decir, su existencia está condicionada directamente a la existencia de la entidad de la que depende.

Esta condición significa que para poder obtener una ocurrencia de la entidad dependiente, es necesario primero identificar la ocurrencia de la entidad principal, o en otras palabras, los campos de la dependencia forman parte de la llave primaria de la entidad dependiente, por esto sirve para identificar,

localizar y ordenar cada una de las ocurrencias de esta entidad (la llave primaria se forma por los campos de la dependencia junto con los atributos de identificación).

Además, una entidad puede depender de una o más entidades.

En SNAP una dependencia se representa por la abreviatura "DEP", definiéndola en la dependiente e indicando la entidad principal. Y gráficamente se representa con una línea continua entre dos entidades, empezando por la entidad dependiente hacia la principal.

La relación de dependencia permite establecer las siguientes relaciones:

- 1) relación de uno-a-uno desde sub-entidades dependientes.
- 2) Relación de uno-a-muchos desde las entidades dependientes.
- 3) Relación de muchos-a-muchos.

A continuación se explicará cómo se definen estas relaciones en SNAP, y se mostrará gráficamente los ejemplos en la notación establecida.

Relaciones de Uno-a-uno Desde Sub-entidades Dependientes

Una relación de uno-a-uno se establece definiendo una dependencia en la sub-entidad dependiente con la entidad principal, sin colocar ninguna otra dependencia ni atributo de identificación. Ver figura 2.1.3.

Por ejemplo, si en el sistema de personal se establece que es importante que exista un archivo de datos de maternidad, gran cantidad de empleados no van a tener información de este tipo, por cuanto muchos empleados van a ser hombres, y muchas mujeres no son todavía madres. Entonces se puede determinar que la información de maternidad es una sub-entidad del empleado, que se relaciona de uno-a-uno. Esto porque un dato de maternidad sólo puede existir si existe el empleado, y a su vez un empleado puede no ser madre (no tener información de maternidad por no ser aplicable a ese empleado) o relacionarse con un sólo registro de maternidad. De esta manera, se debe definir una dependencia en la entidad "Datos de maternidad" hacia "Empleado".

Relaciones de Muchos-a-uno Desde Entidades Dependientes

Una relación de muchos-a-uno se establece definiendo una dependencia en la entidad dependiente (entidad "muchos") con la entidad principal (entidad "uno") junto con un atributo de identificación, sin colocar ninguna otra dependencia. Ver figura 2.1.4

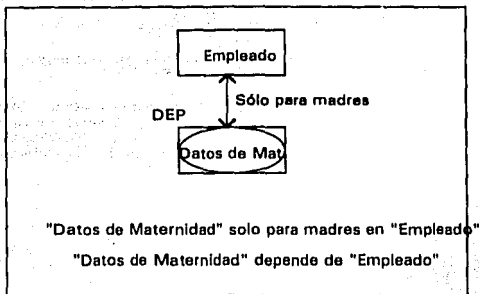


Figura 2.1.3 Relación de uno-a-uno de subentidades dependientes.

Siguiendo con el ejemplo de facturas, diferentes líneas de facturas dependen de una misma factura, puesto que una línea sólo puede existir dentro de una factura, y esta se compone de muchas líneas. De este modo, se debe definir una dependencia en la entidad "Línea de Factura" hacia "Factura" junto con el atributo de identificación "Número de Línea".

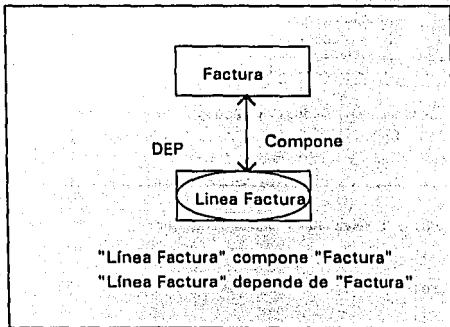


Figura 2.1.4 Relación muchos-a-uno de entidades depend.

Relaciones de muchos-a-muchos

Una relación de muchos-a-muchos se establece definiendo en una entidad asociativa dos o más dependencias con las entidades que se desean relacionar. Ver Figura 2.1.5.

En el caso de que se establezca que un mismo artículo a vender a vender pueda ser almacenado en diferentes bodegas, y que una bodega pueda almacenar diferentes artículos, se determina entonces una relación de muchos-a-muchos entre artículos y bodegas. Para definir esta relación se debe crear una nueva entidad "Artículos en Bodegas" y definir en ella dos dependencias, una hacia "Artículo" y otra hacia "Bodega".

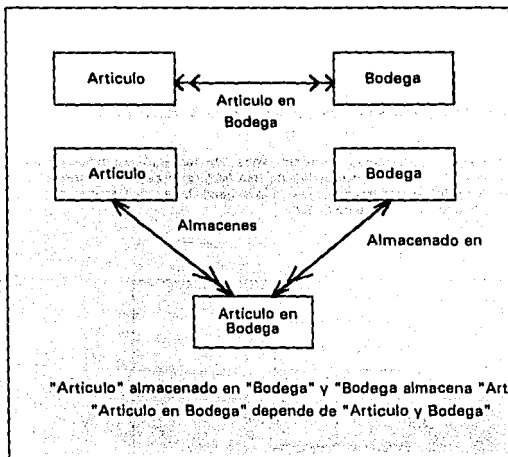


Figura 2.1.5 Relaciones muchos-a-muchos

Relación de Referencia

Una entidad se refiere a otra ocurrencia cuando se desea ampliar su información con la de otra entidad, siendo ambas independientes entre sí, y mantener a la vez la integridad de la base de datos.

En SNAP una relación se representa por la abreviatura "REF" definiéndola en la entidad "muchos" e indicando la entidad "uno".

Al igual que con las dependencias, una entidad puede hacer referencia a una o más entidades.

La relación de referencia nace por la necesidad de reducir la redundancia innecesaria, de manera que los atributos no-llave dependan directamente de una sola llave primaria. Por lo tanto, la referencia sólo permite establecer la siguiente relación:

Relaciones de muchos-a-uno desde entidades independientes

Una relación de muchos-a-uno entre entidades independientes se establece definiendo una referencia en la entidad "muchos" con la entidad "uno". De esta manera toda la información contenida en la entidad "uno" sólo va a ser almacenada una vez en una de sus ocurrencias, mientras que todas las ocurrencias de la entidad "muchos" sólo repetirá la llave foránea y no así los demás atributos.

Para terminar con el ejemplo de facturación, una factura y el cliente que compra con esa factura son entidades independientes, es decir, ni la factura ni el cliente se componen ni de sus existencias se dependen, sin embargo, la factura pertenece a un cliente, y el cliente puede comprar con muchas facturas, estableciendo una relación de muchos-a-uno. Para ello, defina una referencia en la entidad "Factura" hacia la entidad "Cliente". Ver figura 2.1.6.

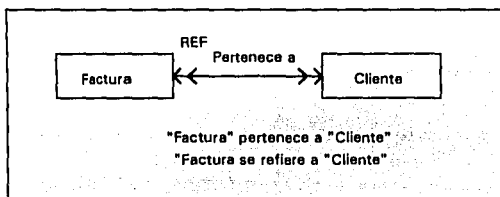


Figura 2.1.6 relación muchos-a-uno de entidades independientes.

Existe la posibilidad de que una entidad tenga que relacionarse más de una vez con otra entidad. Debido a que SNAP, como protección contra la redundancia, no permite que se repita el nombre de una componente dentro de una misma entidad (dos atributos iguales o dos relaciones iguales), es necesario crear una entidad lógica, la cual es una vista externa de la entidad física (vista que puede o no definirse igual que la entidad física).

Por ejemplo, una empresa tiene como política que para dar crédito a un cliente éste debe tener dos personas que respondan económicamente sobre sus obligaciones (fiadores), entonces se puede definir una entidad "Fiador", a la que la entidad "Clientes con Crédito" se deba referir dos veces (Fiador 1 Fiador 2), para ello se debe definir en la entidad "Fiador" una entidad lógica "Otro fiador" sin ninguna

característica adicional (entidad lógica igual a la entidad física), y así poder definir en la entidad "Clientes con crédito" dos referencias, hacia "Fiador" y otra hacia "Otro Fiador".

El siguiente es otro ejemplo, cuando una empresa se dedica a la producción de artículos, éstos pueden ser simples (materia prima, artículos sin componentes) o compuestos (artículos producidos a partir de otros productos simples o compuestos). Bajo este esquema se puede definir una entidad "componentes de artículos" la cual relaciona de muchos-a-muchos los artículos compuestos con sus componentes (que en realidad es otro artículo). Para lograr esto se debe crear una entidad lógica en "Artículo" que se llame "artículo compuesto" y que muestre sólo aquellos artículos cuyo atributo "Tipo" sea "Tipo compuesto" y muestre sólo aquellos artículos cuyo atributo "Tipo" sea "Tipo compuesto" (vista lógica filtrada por el tipo de artículo), de esta manera puedo definir dos dependencias en "Componentes de artículos uno hacia "Artículo Compuesto" y otra hacia "Artículo".

Relaciones cíclicas o auto-relaciones

También existe la posibilidad de que una relación pueda relacionarse con sí misma, en este caso se debe crear también una entidad lógica para que funcione en forma similar al apartado anterior. sin embargo, estas auto-relaciones sólo sirven con referencias, puesto que No es posible que una entidad dependa de sí misma puesto que la primera ocurrencia de la entidad no tendrá ninguna ocurrencia a quién depender, y esto no se permite por la regla de integridad de la entidad (no pueden haber llaves primarias con valores nulos).

Volviendo al ejemplo anterior de las políticas de crédito, la empresa puede exigir que un cliente con crédito deba adquirir una recomendación de otro cliente con crédito de la empresa, des decir, que la entidad necesita referirse a sí misma. Para ello, se define una entidad lógica en "Cliente con crédito" una referencia a "Recomendación".

2.4.7 Tipos de entidades en SNAP

En SNAP el tipo de entidad se determina por las relaciones de dependencia y los atributos de identificación que se hayan definido.

Entidades Fundamentales

Una entidad fundamental es aquella entidad que no posee dependencias a ninguna otra entidad.

En SNAP existe el concepto de entidad primaria que es aquella entidad fundamental que tampoco se refiere a ninguna otra entidad, o sea, es la entidad que no contiene relaciones.

Entidades atributivas

Una sub-entidad dependiente es aquella que posee una sola dependencia sin ningún atributo de identificación.

Una entidad dependiente es aquella que posee una sola dependencia hacia otra entidad junto con un atributo de identificación.

Entidades Asociativas

Una entidad asociativa es aquella que posee más de una dependencia hacia otra entidad, con o sin atributos de identificación.

2.5 Análisis del Sistema Mediante Modelo Entidad-Relación

El modelo de datos se puede definir en forma lógica siguiendo una metodología formal, que permite aprovechar al máximo las características del desarrollo acelerado del SNAP.

Los pasos que componen la metodología siguen la filosofía de trabajo Top Down, lo que permitirá desarrollar las labores en forma consistente con la solución de todo el problema, esto es, solucionar el todo por medio de la solución de las partes que la componen. El enfoque Top Down es altamente utilizado, en el área computacional e Ingeniería en general, por sus características idóneas para la solución de problemas en forma sencilla, eficaz y eficiente.

Antes de empezar a diseñar el modelo de datos es necesario tener una visión clara del sistema que se está desarrollando, es decir, debe de existir una etapa de análisis previa para tener un total conocimiento de los requerimientos de información y de las reglas lógicas que gobiernan los estados válidos entre los datos y los procesos

Es recomendable realizar primero un diseño preliminar utilizando alguna herramienta gráfica de diagramación entidad-relación con las notaciones descritas anteriores, ya sea a través de otra herramienta CASE, o bien, con papel y lápiz.

Una vez listo para la definición del modelo de datos en SNAP, se procede a seguir los siguientes pasos bajo la Metodología de Desarrollo en SNAP:

- 1) Definir las entidades**
- 2) Definir las relaciones entre entidades**
- 3) Definir los atributos entre entidades**
- 4) Definir las vías de acceso sobre las relaciones**
- 5) Customizar el modelo de datos**
- 6) Generar la base de datos**

2.5 1 Definición de entidades de el Sistema

El primer paso es definir las entidades que conforman el sistema. Aquí se pretende establecer en forma general los componentes del modelo de datos, en otras palabras, establecer simplemente cuáles van a ser los objetos sobre los que se desea mantener información, y no cómo se va a mantener esa información. Razón por la cual no se necesita definir la estructura de cada entidad, más bien se debe obtener una enumeración o lista de las entidades del sistema.

2.5 2 Entidades Fundamentales

Las entidades más fáciles de reconocer dentro de un sistema son las entidades fundamentales, puesto que no se derivan de ninguna otra.

2.5.3 Entidades Atributivas

Dentro del análisis del sistema ya se ha establecido cuáles datos son parte de una entidad y cuáles son opcionales, de la misma manera, se conoce cuáles datos son atómicos y cuáles respectivos. En estos casos, se debe definir nuevas entidades (atributivas) para almacenar los datos opcionales y los respectivos.

También existe la posibilidad de que una entidad que se haya definido en el sistema sea en realidad un componente de otra entidad (la entidad es más bien una atributiva), de modo que se debe definir la entidad principal a la cual depende.

2.5.4 Entidades Asociativas

Si existe alguna relación muchos a muchos conocida se debe crear una nueva entidad para que se pueda realizar esta relación.

2.5.5 Entidades Lógicas

Se debe crear además una entidad lógica por cada auto-referencia o se va a establecer más de una relación entre entidades.

2.5.8 Definición de Relaciones entre Entidades

Una vez definidas las entidades se procede a definir las relaciones entre ellas. Las relaciones sirven para establecer las reglas que permiten mantener la integridad referencial de la base de datos.

Dependencias

Se deben definir relaciones de dependencia en las entidades atributivas y entidades asociativas. en las entidades atributivas se debe definir sólo una dependencia hacia la entidad principal. Y en las entidades asociativas se debe definir más de una dependencia, una por cada entidad a relacionar. Es importante recordar que los campos de la dependencia van a formar parte de la llave primaria en la entidad donde se define.

Referencias

Las relaciones de referencia se pueden definir en cualquier tipo de entidad (el número de dependencias es el que establece el tipo de entidad), y se define cuando la entidad es independiente con respecto a la entidad con la que se relaciona. es importante recordar que los campos de la referencia pasan a formar parte de la entidad donde se define como campos no-llave.

2.5.7 Definición de Atributos de Entidades

Una vez definida la estructura lógica que habrá de tener el Modelo de Datos, se procede a definir la composición de las diversas entidades definidas en el primer paso.

Atributos de Identificación

Los atributos de identificación o "AID" son los que van a formar parte de la llave primaria de la entidad. Si la entidad tiene definida una dependencia, entonces, el "AID" va a diferenciar cada una de las ocurrencias que dependen de la misma ocurrencia en la entidad principal (los campos de la dependencia simple están antes de los atributos de identificación en la llave primaria).

Atributos Generales

Los atributos generales "ATR" son las características de cada entidad. Sólo se deben de incluir sólo aquellos atributos que pertenecen propiamente a la entidad.

2.5.8 Definición de las Vías de Acceso sobre las Relaciones

Para que el SNAP puede mantener la integridad referencial de la base de datos (controlar que exista o no relaciones sobre una ocurrencia en una entidad) necesita una vía de acceso por cada relación definida en las entidades.

Vías de Acceso sobre Referencias

Se debe definir una vía de acceso por cada referencia de la entidad.

Vías de sobre dependencias (dependencias no iniciales)

Cuando una entidad tiene una dependencia, la vía de acceso principal sirve para verificar la integridad de datos, por lo que no se necesita definir otra vía alterna.

Pero cuando existe más de una dependencia, la vía de acceso principal sólo sirve para verificar la integridad de la primera dependencia, pero las restantes (las dependencias no iniciales) necesitan una vía alterna para su verificación.

Por tanto, se debe definir una vía de acceso por cada dependencia no inicial de una entidad.

2.6 Customización del Modelo de Datos

Una vez completada la estructura conceptual de la base de datos, se puede proceder a customizar el modelo de datos, es decir, a realizar los ajustes necesarios para completar su definición.

Estos ajustes tienen que ver con especificaciones de características físicas y lógicas a nivel de entidades y de atributos

2.6.1 Customización de Entidades

Una entidad se puede ajustar tanto a nivel externo como a nivel físico. a nivel externo se pueden definir vías alternas de acceso o entidades lógicas, y a nivel físico se pueden definir ciertas características para la generación del archivo físico.

2.6.2 Definición de vías Alternas de Acceso

Una vía de acceso permite leer secuencialmente la entidad ordenada por llave, o acceder directamente un registro si se conoce la llave o bien buscar el inicio o final de un grupo de registros que contengan parte de la llave (la parte inicial de la llave).

En muchas ocasiones las características de las aplicaciones de un sistema requieren que una entidad pueda ser accesada u ordenada por muchos atributos diferentes a la llave primaria. La llave primaria de una entidad representa la vía de acceso. Sin embargo, el desarrollador puede definir vías alternas de acceso para poder manejar la información de la entidad de manera diferente, por ejemplo, para ordenar los registros por algún campo en particular, para definir una llave candidata (duplicados no permitidos en otros campos que no sean la llave primaria), para buscar información por algún campo que no sea la llave principal, etc.

Cuando se definen las vías de acceso se toman en cuenta:

- 1) Integridad referencial (explicado anteriormente)
- 2) Diferentes ordenamientos en los programas CON LIS (ver capítulo IV)
- 3) Diferentes estructuras de reportes en los programas LIS (ver capítulo IV)
- 4) Filtros de información de la entidad
- 5) Otros procesos u ordenamientos sobre la información.

Definición de Entidades Lógicas

Las entidades lógicas son un "espejo" de la entidad física a la que pertenece, en el sentido de que aunque SNAP las considera entidades independientes, físicamente sus datos son parte de los datos de la entidad física. Por esta razón, la llave principal de la entidad lógica es igual a la llave principal de la entidad física, es decir, puede tener muchos atributos que la física o atributos derivados de los físicos (atributos lógicos). Sin embargo, en la vía de acceso principal de la entidad lógica se puede definir filtros, de manera que no puede manejar todos los datos de la entidad física.

Cuando se definen entidades lógicas se debe tomar en cuenta:

- 1) Múltiples relaciones a una misma entidad o auto-referencias (explicado en la definición de entidades)
- 2) Diferentes formas de ver o manejar una entidad (Vista externa).

2.6.3 Customización de Datos

Al igual que con la customización de una entidad, los atributos se pueden ajustar tanto a nivel externo como a nivel físico. a nivel externo se pueden definir una ampliación del dominio determinado por el tipo de dato, y a nivel físico se pueden definir ciertas características para la generación del campo físico.

Definición de Lista de valores para el Atributo

Una lista de valores es un conjunto de códigos con sus correspondientes descripciones que tienen como finalidad establecer el conjunto de valores permitidos por el campo, y que el usuario final pueda escoger por pantalla uno de esos valores.

2.6.4 Documentación del Modelo de Datos

Además, en cada uno de estos pasos deben considerarse la documentación, las cual es una buena técnica de desarrollo, cada vez que se defina algún componente (entidad, vía de acceso, etc.) documentarlo.

2.7 Identificación de Entidades Utilizadas en el sistema COP

A continuación se explica brevemente cada una de las entidades y las relaciones existentes entre estas en el Sistema de Control de Proyectos.

2.7.1 Entidades Fundamentales:

Actividades:

Es la entidad que contiene la clave y descripción de los tipos de actividades que son asignados al personal de la compañía.

Clientes:

Entidad que contiene los datos generales de los clientes de la compañía (AEI Sistemas), los cuales son proporcionados por los mismos clientes.

Conocimientos:

Entidad que contiene la clave y descripción de los conocimientos del personal de la empresa.

Histórico de Planes:

Entidad que contiene los movimientos efectuados y consta por lo tanto del total de atributos de la entidad planes en forma duplicada, para registrar el estado del registro antes y después del movimiento. La alimentación de esos campos se realiza pasando el contenido de los atributos de la entidad Planes antes del movimiento a realizar a los atributos de la entidad Histórico de Planes (En una parte reconocida como de "Antes"), y los atributos contenidos en la parte de "antes" pasan a una reconocida como "Después".

Personal:

Entidad que contiene los datos generales del personal que labora en la empresa, los cuales son proporcionados por el mismo personal.

Puestos:

Entidad que contiene la clave y descripción de los puestos asignados al personal de la empresa.

2.7.2 Entidades Atributivas

Detalle de Reporte de Actividades:

Entidad que contiene consecutivos de un mismo reporte de actividades.

Comentarios:

Entidad que contiene los comentarios de los productos de la entidad Productos.

Planes:

Esta entidad tiene el carácter de principal ya que es la base del sistema "Control de Proyectos. Aquí se describen los componentes fundamentales para el control de un Plan y/o Proyecto y se encuentra de acuerdo a un nivel de desglosamiento (1°, 2°, 3°, 4°, 5°) deseado. Además tiene un comentario asociado.

Productos:

Entidad que contiene los datos generales de los productos, lo cual permite llevar el control de los mismos a un nivel detallado de los planes y/o proyectos en práctica, ejemplo, Análisis, programación básica, Procedimientos, etc. esta entidad tiene dependencia de la entidad Planes y esta a su vez depende de Clientes, por lo tanto cuando se da mantenimiento a la entidad Productos debe existir el Plan y el Cliente.

Reporte de Actividades:

Entidad que contiene todas las actividades diarias realizadas por el personal que labora en la empresa.

A esta entidad se le da mantenimiento a través de un reporte de actividades que es un documento que el mismo personal llena y que se debe de entregar quincenalmente, de lo contrario esto puede ameritar la suspensión de pago por parte de la empresa de ese mismo periodo que se este entregando dicho reporte de actividades.

Tareas:

Entidad que contiene la descripción de todas las tareas que han de efectuarse para cada Producto por lo tanto depende de Planes.

Comentario de Planes:

Entidad que contiene el consecutivo de Comentarios de Planes con el respectivo comentario.

2.7.3 Entidades Asociativas

Personal/Conocimiento:

Entidad que contiene la relación existente entre la entidad Personal y la entidad Puestos con información adicional de el nivel de conocimientos de el personal.

Personal/Clientes:

Entidad que contiene la relación existente entre la entidad Personal y la entidad Clientes con información adicional de fecha de asignación, porcentaje de asignación y responsable de proyecto (S y/o N).

En la siguiente página se muestra el modelo conceptual de el sistema COP.

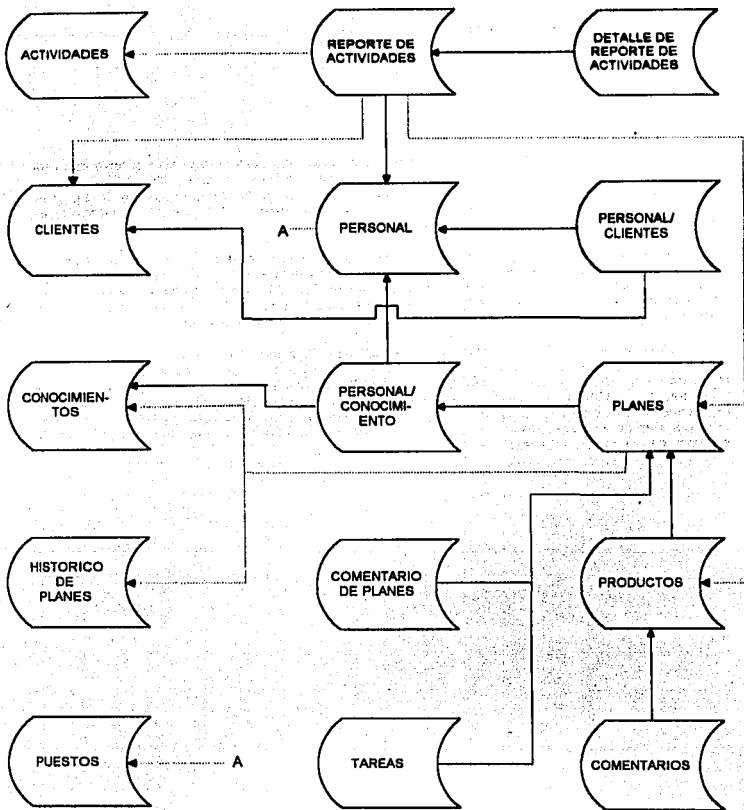


Diagrama de Relación de Entidades del Sistema COP

Capítulo III

HERRAMIENTA CASE

Introducción

Un cambio en las herramientas de software está sucediendo hoy en día. las herramientas para automatizar varias de las tareas del ciclo de vida se crean para utilizarse en los micros. Muchas de tales herramientas se centran en el análisis de software y en las tareas de diseño. Con ayuda de estas herramientas, los profesionales del desarrollo pueden crear sistemas de software interactivamente en computadores personales y estaciones de trabajo.

Tanto los departamentos que manejan sistemas de información como los de desarrollo de sistemas de tiempo real, están recibiendo estos cambios. Hay entusiasmo sobre esta nueva tecnología de software, que puede resolver los problemas que desde hace 25 años aquejaba a la productividad. Esta nueva tecnología emplea lenguajes de cuarta generación y la emplean las herramientas CASE (Computer Aided Software Engineering) Herramienta de ingeniería asistida por computador.

Definida de una forma simple, la herramienta de ingeniería asistida por computador es la automatización del desarrollo de sistemas de información, comúnmente referida como tecnología CASE. Hay más expectación y esperanza asociada a los CASE que cualquiera de las tecnologías de software anteriores. Los CASE están consiguiendo que los profesionales del desarrollo olviden las tecnologías de la tercera generación pues los CASE reemplazó a los lenguajes de la tercera generación y su base principal son los lenguajes de cuarta generación como la más poderosa tecnología de desarrollo de sistemas de manejo de información.

3.1.1 La Automatización del Software

Los CASE proponen una nueva formulación del concepto de ciclo de vida de los sistemas de información, basada en la automatización. La idea básica que subyace en los CASE es de proporcionar un conjunto de herramientas bien integradas y que ahorran trabajo, enlazando y automatizando todas las facetas del desarrollo de vida de los sistemas de información (ver figura 3.1.1).

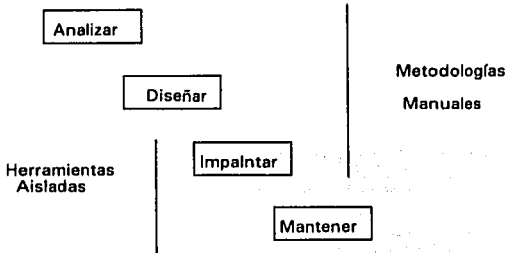


Figura 3.1.1. Las tecnologías tradicionales de software son de dos tipos: manuales y con herramientas aisladas. La tecnología CASE es una combinación de herramientas y metodologías totalmente integrada, con especial énfasis en la automatización de vida de los sistemas.

Las tecnologías tradicionales de sistemas son de dos tipos: herramientas y metodologías. Estas categorías incluyen herramientas de tercera, de cuarta y (más recientemente) de quinta generación. La mayoría de estas herramientas son autónomas, basadas en un mainframe (ordenador principal) y dirigidas a la implantación del ciclo de los sistemas de manejo de información.

En la categoría de metodologías de construcción de sistemas, se incluyen las metodologías de desarrollo manual, como el análisis estructurado, el diseño estructurado y la programación estructurada. Estas tecnologías definen un disciplinado proceso para el desarrollo de los sistemas paso a paso.

La tecnología CASE es una combinación de herramientas de software y de metodologías. Más aún, los CASE es diferente de las primitivas tecnologías del software porque se centra en el problema de la productividad del sistema y no solamente en la implantación de soluciones. Extendiéndose a todas las facetas del ciclo de vida del software. Los CASE es la tecnología del desarrollo de sistemas de información más completa hoy en día. Los CASE aborda los problemas de la productividad del sistema en cuestión, durante todo el ciclo de vida, automatizando muchas de las tareas del análisis y del diseño como también las tareas de la implantación y el mantenimiento de los programas.

Como las metodologías estructuradas manuales son demasiado tediosas y de un trabajo muy extensivo, en la práctica raramente se sigue a un nivel más detallado. Los CASE hace prácticas las metodologías estructuradas manuales al automatizar el dibujo de diagramas estructurados y la generación de la documentación del sistema.

3.1.2 Beneficios que Proporcionan los CASE

El cuadro 3.1.1 muestra las ventajas que los CASE ofrece a los profesionales del desarrollo de sistemas de información. Esta exposición sugiere que estos lenguajes no son enteramente una nueva tecnología nueva, sino que está constituida sobre técnicas y herramientas comprobadas en la práctica. En este sentido, estas herramientas pueden definirse como un conjunto de conceptos estructurados y metodologías con, un nuevo empaque, la novedad es la automatización.

- Permite las técnicas estructuradas
- Impone las ingenierías de software y de la información
- Aumenta la calidad del software mediante comprobación automática
- Favorece la realización de prototipos
- Simplifica el mantenimiento del programa
- Acelera el proceso de desarrollo
- Libera al profesional de desarrollo de la principal parte creativa en el desarrollo de software
- Anima al desarrollo evolucionado y gradual
- Posibilita la re utilización de los componentes de software

Cuadro 3.1.1 Ventajas de la tecnología CASE

3.1.3 Las Herramientas CASE

La columna vertebral de la tecnología CASE es una nueva generación de herramientas de software, algunas de estas herramientas están basadas en micros, tienen potentes entornos gráficos que hacen más amigable el interfaz del usuario y se incorporan a bases de trabajo donde pueden ser fácilmente llamadas, alimentadas y utilizadas por otros. Las herramientas CASE se enfocan hacia la productividad individual del profesional del desarrollo del software. Con el uso de las estaciones, de las redes de área local y de herramientas basadas en computadores personales, el profesional de desarrollo de software puede trabajar por primera vez desde un entorno de desarrollo de software fiable y dedicado.

Ya que las herramientas CASE se utilizan principalmente por los profesionales de desarrollo de sistemas de información y no por los usuarios finales, la tecnología CASE no debe considerarse solamente como una mejora de las herramientas de tercera generación. Además, las herramientas CASE difieren de las herramientas de tercera generación en otros dos aspectos. Primero, muchas herramientas CASE están diseñadas para estaciones de trabajo, mientras que muchas herramientas de tercera generación están diseñadas para ordenadores principales (mainframes). Segundo, los CASE es una tecnología de software de propósito general para utilizarse en el desarrollo de todo tipo de sistemas (grandes y pequeños, comerciales y científicos, en línea y de tiempo real), mientras que las herramientas de tercera generación se utilizan principalmente para desarrollar aplicaciones de tamaño pequeño o mediano.

En general, las herramientas CASE se diferencian de las primitivas herramientas de software en algunas cosas muy importantes. A saber, las herramientas fueron diseñadas para:

- 1.- Soportar un entorno personal dedicado

- 2.- Utiliza gráficos para especificar y documentar los sistemas
- 3.- Juntar todas las fases del ciclo de vida del software
- 4.- Capturar y juntar en el ordenador toda la información sobre el entorno del software de un sistema, desde los requerimientos iniciales hasta actividades para un mantenimiento constante.
- 5.- Utilizar la inteligencia artificial para realizar automáticamente muchas de las rutinarias tareas del desarrollo y mantenimiento del software.

3.1.4 Ejemplos de Herramientas CASE

Aún más, las herramientas CASE realizan muchos más tipos de tareas que las herramientas de tercera generación. Soportan la automatización de una gran variedad de tareas de desarrollo y mantenimiento del software, incluyendo la gestión del proyecto. Ejemplos de herramientas CASE incluyen:

- Herramientas de diagramación para el dibujo de diagramas estructurados y la creación de especificaciones de sistemas gráficos.
- Pantallas de informes gráficos para la creación de las especificaciones del sistema y de formulación de prototipos simples
- Diccionarios, sistemas de gestión de bases de datos y utilidades para almacenamiento información y consulta de la información técnica y de proyectos del sistema de gestión
- Herramientas de validación de especificaciones para la detección automática de las que son incompletas, incorrectas sintácticamente o inconsistentes.
- Generadores de código para generar automáticamente código ejecutable a partir de las especificaciones obtenidas gráficamente.
- Generadores de documentación para la documentación técnica y del usuario requeridas por las técnicas estructuradas.

Aunque las herramientas son una parte muy importante de CASE, la tecnología CASE consta de mucho más. Los CASE es una re definición del entorno completo de software.

Una forma clave en que los CASE cambia el entorno de desarrollo del software es que muchas herramientas CASE operan en entornos de estación de trabajo. Esto cambia el desarrollo de software en un proceso altamente interactivo en el cual la detección de los errores comienza en los iniciales de la definición de los requerimientos del sistema.

3.1.5 La Historia de los CASE

Como se muestra en la figura 3.1.4, la historia de la tecnología CASE comienza a principios de los años ochenta con la introducción de la documentación asistida por computador y de las herramientas de diagramación. Estas herramientas representan parte de las primeras herramientas del desarrollo de software basadas en computadoras personales y de los primeros intentos de automatizar el análisis y el diseño de las tareas. Estas herramientas se idearon simplemente para el dibujo en puestos individuales. Se emplearon para crear diagramas de flujo estructurados (de flujo de datos, organigramas y de entidad-

relación). Su propósito era producir automáticamente la documentación estructurada requerida por las distintas metodologías de desarrollo estructurado. Las diferentes herramientas CASE soportan diferentes metodologías de desarrollo estructurado, como el desarrollo estructurado del análisis y de diseño de Yourdon, el diseño estructurado de Jackson o la Ingeniería de la Información de Martín.

Principios de los años Ochenta	Mediados de los años Ochenta	Últimos años de los Ochenta	Principios de los Años Noventa
Documentación asistida por computador. Diagramación asistida por computador. Herramientas de análisis y de diseño.	Comprobación automática del análisis del diseño. Depósito automático de la información del sistema.	Generación automática del código a partir de las especificaciones diseño. Engarce del diseño automático y la programación automática.	Conductor de metodología inteligente. Interfaz amigable con el usuario. La re utilización como metodología de desarrollo.

Figura 3.1.4 Los CASE es una tecnología con una buena sintonía de pocos años.

Las primitivas herramientas CASE se dirigieron principalmente a la automatización de la documentación y de la comunicación como una mejora de la productividad del software. A mediados de los años ochenta, las herramientas CASE se mejoraron para proporcionar dos funciones muy importantes:

1. Comprobación automática de diagramas estructurados.
2. Almacenamiento de diagramas estructurados en librerías o enciclopedias.

Los diagramas se revisan para asegurar que están completos y correctos antes de utilizarlos como especificaciones de diseño de la implantación del programa. La comprobación se basa en las reglas de la metodología estructurada soportada. Además, se registran las referencias cruzadas de la información y se eliminan las redundancias.

Después de la validación de diagramas, estos se almacenan en diccionarios o bases de datos, llamados depósitos CASE. Una vez almacenados los diagramas se pueden actualizar fácilmente compartiéndolos todo el equipo del proyecto y más tarde utilizados en otros proyectos de desarrollo de software.

En este estado, el foco de la CASE está en el diseño automático, porque el análisis y el diseño se consideran como fases iniciales del ciclo de vida del software. Realizando un trabajo mejor en las fases iniciales del ciclo de vida redundará favorablemente en la calidad y costo del sistema. Con la asistencia automática de las herramientas CASE, el tiempo de diseño se reduce substancialmente, a la vez que se producen especificaciones de diseño de alta calidad y sin errores.

El paso siguiente en la evolución de la tecnología CASE fue la unión de diseño automático con la programación automática. Mientras que el diseño automático se relaciona con la asistencia por

computador a las tareas de análisis y diseño, programación automática significa generadores automáticos de código. La unión implica que el 80 al 90 por ciento del sistema del software puede generarse a partir del diseño de diagramas estructurados. (Ver figura 3.1.5).

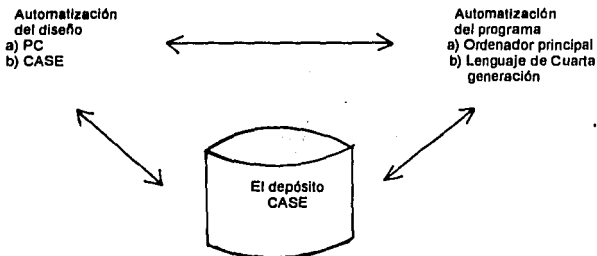


Figura 3.1.5 El engarce de la automatización del diseño con automatización del programa significa que puede generarse automáticamente todos los sistemas de software desde las especificaciones gráficas a alto nivel del sistema. Una parte muy importante de esta unión se debe al depósito CASE.

El diseño automático lo soportan las herramientas CASE basadas en ordenadores personales desarrolladas durante la primera mitad de los años ochenta. La programación automática la proporcionan los generadores de aplicaciones y de código de la cuarta generación, la mayoría de los cuales están basados en mainframes. Para unir el diseño automático y la programación automática se requiere un puente entre dos entornos hardware (ordenadores personales mainframes) y dos tecnologías de software (CASE y cuarta generación). Una parte muy importante de este puente será añadir un depósito al entorno CASE.

El depósito CASE es un mecanismo para almacenar y organizar toda la información necesaria para la gestión del proyecto, y parte, para la generación automática de código. Incluye información sobre el problema que se va a resolver, sobre el dominio del problema, sobre los procesos del software que están siendo utilizados, sobre modelos de datos de procesos, sobre prototipos, sobre la historia y los recursos del proyecto, y sobre el texto organizativo.

Además de la unión entre el diseño automático y la programación automática, el depósito CASE convierte por primera vez, el concepto del software reutilizable en algo práctico. Los componentes del software reutilizable son la clave del vertiginoso aumento de la productividad del software. En lugar de ir creando cada vez nuevo software o de mejorar el dudoso, los profesionales de desarrollo de software pueden utilizar los componentes del software reutilizable almacenados en los depósitos CASE. No solamente podrán los profesionales reutilizar los módulos de código fuente, sino que también podrán reutilizar planes de proyectos, modelos de prototipos, modelos de datos y especificaciones de diseño.

3.1.6 Objetivo de los CASE

Para resumir, los CASE han cambiado radicalmente la forma de construir los sistemas de software al proporcionar tres avances principales:

1. Un entorno de desarrollo interactivo con un tiempo de respuesta rápido, recursos dedicados y una comprobación de errores desde el principio.
2. La automatización de muchas tareas de desarrollo y mantenimiento del software.
3. Una programación visual proporcionada por potentes interfaces gráficas.

El fin último de la tecnología CASE es automatizar todo el ciclo de vida del software. No obstante, para el estado tecnológico actual se trata de una meta a conseguir, no de una realidad.

3.2. Entorno de Desarrollo en CASE

Según se han ido sucediendo los nuevos avances tecnológicos en hardware y software, el entorno de desarrollo ha ido cambiando desde un entorno individual en los años cincuenta y principios de los sesenta hacia el modo de proceso por lotes (batch) en los años sesenta y principios de los setenta, y después al modo de tiempo compartido a finales de los años ochenta. A finales de los ochenta, un nuevo modo, basado en las estaciones de trabajo personales se ha impuesto como entorno de desarrollo de software. La introducción de potentes herramientas en las estaciones de trabajo con el comportamiento de metodología estructuradas y automatiza el proceso del desarrollo de software.

La estación de trabajo CASE es un entorno completo, que incluye hardware y software, cuya función es de proporcionar asistencia por ordenador para la producción, mantenimiento y gestión de los sistemas de software.

Además, la estación de trabajo proporciona el procesamiento de textos, el almacenamiento y la recuperación de la información, el correo electrónico y las funciones de ayuda y calendario. Es un computador personal dedicado a proporcionar soporte máximo posible individualmente al profesional de desarrollo de software.

El objetivo principal de la utilización de las estaciones de trabajo CASE es incrementar la productividad del software. Inmediatamente después un segundo objetivo es incrementar la calidad del software. Otros objetivos frecuentemente citados se encuentra en el cuadro 3.2.1

- Maximizar la productividad del programador
- Aumentar la calidad del software y reducir los errores
- Simplificar los procesos de desarrollo del software
- Posibilitar un empleo más eficiente de los recursos del computador
- Reducir los costos del software
- Aumentar la fiabilidad del entorno del desarrollo
- Proporcionar un entorno de desarrollo que tenga un interfaz humano mejor con el sistema
- Revolucionar los procesos de desarrollo del software
- Automatizar la generación de código ejecutable

- Soportar unos prototipos rápidos
- Automatizar la automatización el proceso de desarrollo
- Automatizar la gestión del proyecto
- Formalizar y normalizar el proceso de desarrollo de software
- Normalizar la documentación del software
- Integrar las herramientas de desarrollo
- Promover la reutilización del software
- Promover un control general sobre el desarrollo y mantenimiento del software
- Mejorar la portabilidad del software a través de los entornos

Cuadro 3.2.1 Los objetivos de las estaciones de trabajo

3.2.1 El Banco de Trabajo CASE

Un amplio conjunto de herramientas de software inteligentes e integradas llamadas banco de trabajo CASE compone el software del entorno de la estación de trabajo. Los hay a las medidas de las preferencias individuales del programador de desarrollo, de asistencias a tareas especializadas, como la gestión del proyecto, el diseño y el mantenimiento, y reemplazan las herramientas tradicionales basadas en el proceso por lotes y a los lenguajes de tercera generación de los años sesenta y setenta.

Mientras que los entornos de programación de los años setenta y ochenta se concentraron en herramientas para las facetas de codificación e implantación, los bancos de trabajo CASE proporcionan potentes herramientas para la especificación, el diseño, la implantación, la comprobación y la documentación. Son un entorno de soporte de software de propósito general destinado a soportar toda la gama de tareas del software y su gestión.

Para cumplir los objetivos de mejorar la productividad y la simplicidad de los procesos de desarrollo de software, el banco de trabajo CASE no puede constituir solamente en una colección de las mejores herramientas de los años setenta y ochenta. Esas herramientas no se diseñaron para utilizar en los entornos de los computadores personales permitiendo la particularización individual para cada usuario de la estación de trabajo. También fueron diseñadas para emplear potentes utilidades gráficas que potencien el interfaz humano. Ni fueron diseñadas para capturar la información sobre el progresivo desarrollo del software y la evolución de los productos de software. Finalmente, no fueron diseñadas como herramientas inteligentes capaces de muchas de las tareas del desarrollo por sí mismas.

Lo que hace el banco de trabajo CASE es ensamblar las herramientas que pueden proporcionar un soporte funcional completo al proceso del software:

- Creación de los requerimientos gráficos de sistema y de las especificaciones de diseño.
- Validación, análisis y referencias cruzadas de la información del sistema.
- Almacenamiento, gestión y comunicación de la información del sistema y de la gestión del proyecto.
- Construcción del prototipo de sistemas y simulación de sistemas.
- Imposición de la normalización y de los procedimientos.
- Prueba, validación y análisis de los programas
- Interfaces para la salida a los diccionarios y las bases de datos

Para proporcionar un soporte total de software, un banco de trabajo CASE completo debe tener las siguientes prestaciones:

- Gráficos
- Comprobación de errores
- Depósito de información
- Juego de herramientas totalmente integradas
- Cobertura total del ciclo de vida
- Soporte de prototipos
- Generación automática de código
- Soporte de metodología estructurada

En este tema se tratará de las cuatro primeras competencias de un banco de trabajo CASE, y en el siguiente las restantes.

3.2.2 Las Capacidades Gráficas

Cuando se usa una estación de trabajo CASE, lo primero que impresiona es el editor gráfico. La facilidad con la que consigue que los objetos aparezcan, desaparezcan y se muevan por la pantalla de la estación de trabajo con sólo apretar un botón es increíble. Sin embargo, tras los gráficos hay una presentación muy importante. El mejor interfaz gráfico es el que puede hacer al usuario más productivo.

Definición del programa del sistema de suscripciones

El sistema a suscripciones procesa las transacciones de suscripciones contra el fichero de suscripciones. Las transacciones son de tres tipos: nuevas suscripciones, renovaciones y cancelaciones. Cada transacción primero se valida y después se procesa contra el fichero maestro. En las sucesivas suscripciones, se crea un registro de cliente y se genera una anotación en el balance. En las renovaciones, se actualiza la fecha de expiración y se genera una anotación en el balance. En las cancelaciones se marca el registro como cancelado y se obtiene un saldo.

Figura 3.2.1 La descripción textual de las especificaciones del sistema de suscripciones.

La figura 3.2.1 es una descripción narrativa de las especificaciones de un sistema de suscripciones. Hay otra manera de describir estas especificaciones, como puede ser la forma gráfica (un diagrama de flujo de datos). Mucha gente prefiere la forma gráfica, ya que la mente humana está orientada a la imagen. Mientras que la narración es unidimensional, la imagen es multidimensional, tomando prestadas algunas propiedades del mundo físico, como el tamaño, la forma y el color. Ya que el lenguaje de las imágenes es más rico que el de los textos, se puede representar mucha más información en una imagen más rápidamente.

3.2.3 La Necesidad de los Diagramas

Las representaciones gráficas (o diagramas) siempre han desempeñado un papel muy importante en el desarrollo del software. Los diagramas se utilizan para definir las especificaciones de un programa y la representación del diseño de los programas. Proporcionan el proyecto original para la implantación del diseño en código y además son una forma importante de documentación del software.

Los diagramas claros son una parte esencial en el diseño de sistemas complejos y en el desarrollo de programas. Si sólo hay una persona desarrollando el diseño de un sistema o de un programa, los diagramas ayudan a un razonamiento claro. Una técnica de diagramación pobre en representaciones puede inhibir al razonamiento, en cambio una buena puede acelerar el trabajo y aumentar la calidad de los resultados.

Cuando trabaja muchas personas en sistema de software, los diagramas son una herramienta esencial de comunicación. Se necesita una técnica de diagramación formal para que los desarrolladores intercambien ideas y reúnan con precisión los distintos componentes del sistema.

Una vez corregidos los errores, los diagramas son una herramienta valiosa para saber cómo otro programador intentaba que funcionara el programa y para localizar el origen de los errores y el impacto del cambio.

Cuanto más grande sea el equipo de personas, mayor es la necesidad de precisión en la diagramación. Es muy difícil, casi imposible, que los componentes de un gran equipo puedan comprender en detalle el trabajo de los demás. Para resolver este problema, cada componente del equipo deberá estar familiarizado con el sistema, al menos con una visión general, y ver donde encaja su parte en el todo. Deberá tener claros y perfectamente definidos sus contactos y deberá ser capaz de ver el sistema o programa con varios grados de detalle. Es más, cada parte del sistema deberá describir en la forma normalizada para facilitar la comunicación y evitar las confusiones que puedan concluir al error.

Los diagramas son realmente un lenguaje de modelado del software, porque ofrecen para describirlo una forma concisa y sin ambigüedades de representación. Son fundamentales en una buena documentación del sistema (Ver cuadro 3.2.2). Además, son tan fundamentales para el análisis y el diseño del software que las diferentes metodologías pueden caracterizar a partir de las técnicas de diagramación utilizadas para modelar un sistema de software.

Cuadro 3.2.2 Requisitos de una buena documentación

Una buena documentación del sistema y de los programas es un componente esencial de la filosofía estructurada. La buena documentación cumple con los siguientes requisitos:

- Mejora la descripción de los sistemas de software.
- Es fácil y poco costosa de realizar y de actualizar, y puede producirse automáticamente.
- Proporciona una visión de alto nivel del sistema (programa), explicando su propósito y las relaciones entre los diversos componentes (datos y procedimientos).
- Describe detalladamente los componentes de los datos y procedimientos.
- Proporciona un proyecto para representar el problema de los requerimientos del diseño de un sistema y del traslado del sistema al código de programa.

3.2.3.1 Ventajas de las Técnicas de Diagramación Estructurada

Los formalismos gráficos utilizados en la construcción de los diagramas de flujo de datos y en el modelo entidad-relación, cumplen las condiciones básicas para producir una buena documentación. Primero combinan las anotaciones gráficas y de texto para facilitar la comprensión de los sistemas. Los gráficos son especialmente útiles porque tienden a ser menos ambiguos que las descripciones por escrito. Además, al tender a ser más concisos, los gráficos pueden dibujarse en mucho menos tiempo de lo que se tardaría en escribir un documento conteniendo la misma cantidad de información.

Segundo, las técnicas de diagramación estructurada son de jerarquía descendente (Top-down). Esto significa que pueden soportar una formulación de desarrollo estructurado "arriba-abajo". Pueden describir un sistema, un programa o una estructura de datos variando el grado de detalle durante cada paso de la composición del proceso proporcionando una forma normalizada de describir la lógica del proceso y las estructuras de los datos.

Finalmente, las técnicas de diagramación estructurada ayudan a los programadores a tratar con grandes volúmenes de detalles generados durante el proceso de desarrollo del programa. Proporcionan una representación lógica más que una orientada a la programación, lo que es mucho más significativo para los usuarios y para los directivos no técnicos.

3.2.3.2 Usos de los Diagramas Estructurados

Las técnicas de diagramación se han desarrollado a la par que las necesidades que las metodologías de programación. En los años cincuenta y sesenta, se utilizaron los organigramas para planificar detalladamente la complicada lógica de los programas. Perdieron terreno al no poder ofrecer una visión a alto nivel de la estructura de los programas. En los años setenta se extendieron las técnicas estructuradas y con ellas se introdujeron las técnicas estructuradas de diagramación.

Unas técnicas de diagramación se utilizan como herramienta de análisis; otras, como herramientas de diseño de ficheros y de bases de datos. En el cuadro 3.2.3 se exponen algunos ejemplos de técnicas de diagramación estructuradas de acuerdo a estas categorías.

Los diagramas de flujo de datos, los diagramas de descomposición y los diagramas de control son ejemplos de herramientas de análisis. Por ejemplo el diagrama de flujo de datos se utiliza para escribir la transformación que los datos experimentan en su flujo a través del sistema. el diagrama de flujo de control añade la información de control necesaria para describir los sistemas de tiempo real. El diagrama de descomposición se utiliza para identificar la estructura de una organización y sus funciones comerciales básicas.

Cuadro 3.2.3 Técnicas de diagramación estructurada.

	Tipo de documentación	
	de procedimiento	de datos
Herramientas de análisis		
Diagrama de flujo de datos	X	
Diagrama de flujo de control	X	
Tabla de decisión, árboles	X	X
Matrices	X	
Diagrama de dependencia	X	
Diagrama de descomposición		
Herramienta de diseño		
Diagrama de estructura	X	
Diagrama de acción	X	X
Diagrama de Warnier-Orr	X	
Diagrama de transmisión de estado	X	
Tablas de decisión, árboles	X	
Pseudo código	X	
Organigramas	X	
Diseños de pantallas	X	
Flujo de diálogo	X	X
Diseño de ficheros y bases de datos		
Modelo de datos		X
Diagrama entidad-relación		X
Estructura de datos		X
Registros lógicos		X
ficheros y bases de datos físicos		X

El diagrama de estructura se utiliza como una herramienta de diseño para definir la organización jerárquica de un programa, incluyendo sus componentes de procedimiento y los interfaces de datos que los conectan. los diagramas de acción y los de Warnier-Orr pueden utilizarse para mostrar la organización jerárquica del programa también como una lógica detallada de procedimiento.

Los diagramas de entidad-relación y sus modelos de datos se utilizan para describir estructuras de datos. Cada entidad de datos, sus atributos y sus relaciones con otras entidades de datos puede definirse con estos diagramas. Además la cardinalidad podría ser un indicativo para distinguir las relaciones opcionales y obligatorias cuando se capturen las reglas de las actividades del negocio.

3.2.3.3 Visión del Sistema Múltiple

Las técnicas de diagramación se utilizan para dar una visión de alto nivel y detallada de un sistema. Por ejemplo, los diagramas de estructura dan un panorámica de alto nivel de un programa.

Pueden utilizarse para explicar en términos generales las principales funciones que realizan el programa y que datos de alto nivel y procedimientos componen el programa. Por otra parte, los diagramas de acción y de pseudocódigo ofrecen una visión del programa a nivel de instrucción. Muestran donde la variable se inicializa, se comprueba o se referencia en el programa.

Una visión de alto nivel y/o detallada de un programa es importante según el propósito del lector. Si el lector está buscando un error, la documentación detallada puede guiarle a la localización exacta del error. Si la persona desea determinar en cual de varios programas se realiza una determinada función, una variable de alto nivel puede ser la mejor ayuda.

Los diferentes diagramas son más adecuados para la representación de alto nivel o detallada. Por ejemplo, los diagramas de estructura son los más adecuados para la representación global o de alto nivel que introduce al lector en el programa. Pero a un nivel de detalle, los diagramas de estructura están demasiado llenos de cajas y errores para representar claramente los detalles.

Para representar un sistema completo con diagramas, se necesitan una estructura del proyecto detallada y de alto nivel. La visión estructurada del proceso (o del sistema) identifica los componentes del proceso y las relaciones entre ellos mostrando el flujo de datos, el flujo de control, y el tiempo y la secuencia de la información. La visión de la estructura de los datos describe las entidades y sus atributos, así como las relaciones entre entidades. Con la ayuda de ambas visiones, el programador puede ver cómo los datos se derivan o se utilizan en el proceso; y en el caso de los sistemas en el tiempo real, cómo los hechos externos (señales) ocasionan los cambios de estado del sistema.

Diagramación Automática

El problema con la representación gráfica es el tiempo que lleva dibujarla manualmente y el tiempo que lleva cambiarla manualmente. La automatización elimina este problema. Cuando menos tiempo se emplee en el dibujar, más productivo será el programador.

La diagramación interactiva en la pantalla de un ordenador tiene importantes ventajas. acelera enormemente el proceso; fuerza la normalización, y puede realizar automáticamente la documentación del proceso. En el cuadro 3.2.4 se resumen las ventajas de las herramientas de diagramación estructurada automática.

Cuadro 3.2.4 Funciones de las herramientas de diagramación estructurada.

Las herramientas de diagramación estructurada automática proporcionan las siguientes funciones importantes:

- Una ayuda para aclarar el conocimiento y la solución del problema.
- Precisar y registrar la comunicación entre los componentes del equipo de desarrollo, los usuarios y los directivos.
- Normaliza la representación de las estructuras de las arquitecturas y de los datos del programa.
- Una ayuda en la detección de errores.
- Una ayuda para analizar y comprender la existencia de los programas
- Una ayuda para cambiar los sistemas y programas (mientras se están creando y durante su mantenimiento).
- Un desarrollado más rápido motivando la diagramación asistida por computador.
- Un mecanismo que capacita a los usuarios para revisar los requerimientos del programa y las especificaciones de su diseño.
- Un mecanismo de producción automática de la documentación del programa.

3.2.4 La Comprobación de Errores

La comprobación de errores que es utilizada a este nivel de análisis del sistema y de los programas es la realizada por los llamados Upper CASE, detectan los errores desde el principio, es un excelente modo de reducir los costos del sistema. La comprobación automática de errores ayuda al programador a encontrar los errores mucho antes en el ciclo de vida del sistema. La calidad de un sistema de información depende de sus especificaciones, y la calidad de las especificaciones de un sistema dependen de su integridad y de su consistencia. La comprobación automática de errores comienza tan pronto como el programador empieza a dibujar un diagrama estructurada en la pantalla de la estación de trabajo CASE.

3.3 El Soporte CASE a los Procesos de Desarrollo de Sistemas de Información

3.3.1 El soporte de las Metodologías y los Procesos de Desarrollo de Sistemas.

La función de un CASE en el nivel de desarrollo de los programas que componen el sistema la realizan los llamados Lower CASE que es un conjunto de herramientas integradas enfocadas a la validación de desarrollo de sistema real. Las herramientas CASE automatiza la producción de la documentación requerida en cada fase del desarrollo y del mantenimiento.

Por ejemplo, en las primeras fases del ciclo de vida del sistema, las herramientas CASE, como los lenguajes de generación de pantallas y de especificaciones ejecutables, se emplean para construir modelos, prototipos y simulaciones para descubrir los requerimientos del sistema y comprobar el comportamiento del mismo. En la fase de Implantación del ciclo de vida del sistema, las herramientas CASE, como los generadores de código y de bases de datos, se utilizan para escribir automáticamente programas documentados a partir de las especificaciones de diseño de alto nivel.

Una parte importante del soporte del ciclo de vida del sistema es apoyar las metodologías que estructuran los pasos del proceso dentro del ciclo de vida. Un CASE soporta el empleo de metodologías estructuradas automatizando la producción de la documentación requerida por la metodología y guiando al usuario en la correcta utilización de la metodología empleada.

En general los CASE a este nivel soportan tanto los procesos del ciclo de vida del sistema como las metodologías estructuradas analizando las cuatro prestaciones siguientes.

- La cobertura del ciclo de vida del software
- El soporte de prototipos
- La generación automática de código
- El soporte de la metodología estructurada

3.3.2 El Énfasis en las Primeras Fases del Ciclo de Vida

Una parte importante de la tecnología CASE se centra en la primeras fases del ciclo de vida del sistema. Este énfasis comienza con el reconocimiento de que las etapas de análisis y de diseño son las más críticas del ciclo de vida.

Los errores en las especificaciones pueden ser muy costosos si no se detectan y corrigen en la primera fase de mantenimiento puede llegar a ser cien veces más caro que si hubiera sido corregido durante la fase del análisis. La integridad y la exactitud de las especificaciones del sistema afectan al éxito del esfuerzo del desarrollo de todo sistema. La especificación es la base de la planificación y asignaciones del proyecto, de la planificación de las pruebas, de la documentación del usuario y del diseño del programa. Una comprensión deficiente del sistema es la causa de las fallas del sistema.

Los errores de diseño a menudo dominan los proyectos de sistemas de información dependiendo de su número y costo de corrección, especialmente cuando no se detectan pronto. En los proyectos grandes, los errores de diseño superan a menudo los de código. En un estudio de TRW sobre errores detectados durante o después de la prueba de aceptación, los errores de diseño fueron del 62% frente al 32% de los errores de código. No solamente ocurrieron más errores de diseño, sino que su corrección más costosa que la de los errores de código.

Cuando más cuidado se preste al diseño, menos costo y más fiable será el sistema. El diseño del sistema es la guía de su implantación. Si la guía no existe o es incorrecta, el sistema producido probablemente estará mal organizado, documentado y su mantenimiento será una pesadilla.

3.3.3 Los Prototipos

Las herramientas de prototipo son una parte muy importante de la automatización de las primeras fases del ciclo de vida. Se utilizan para determinar los requerimientos del sistema y para responder a las preguntas sobre el comportamiento del nascente sistema. El cuadro 3.3.1 muestra algunas herramientas CASE utilizadas para realizar los prototipos de varios aspectos del sistema de manejo de información.

Cuadro 3.3.1 Herramientas CASE para prototipos.

- Generadores de Pantallas
- Generadores de Informes
- Construcción de menús
- Lenguajes de Cuarta Generación
- Lenguajes de especificaciones ejecutables

Los generadores de pantallas, los generadores de informes y los constructores de menús se utilizan principalmente para hacer el prototipo del Interfaz de usuario como una forma rápida y agradable de clasificar los requerimientos del usuario. Los prototipos proporcionan al usuario un modelo correcto de cómo se verá el sistema desde la perspectiva del usuario. En este nivel de prototipos, el diálogo de las pantallas y la navegación de los datos puede simularse con o sin compilación. también pueden

generarse automáticamente el código fuente, las pantallas y las descripciones de los informes. Este es un modo efectivo para identificar y corregir los equívocos sobre las expectativas del usuario del sistema.

Los lenguajes de cuarta generación pueden utilizarse para desarrollar un modelo más completo del sistema. En este caso, el prototipo incluye las funciones principales del sistema pero no comprueba las excepciones ni los datos de entrada inválidos, ni se preocupa de los rendimientos de ejecución. Su propósito es proporcionar experiencia con el sistema al usuario al utilizar un modelo bastante completo. En ocasiones, el modelo es adecuado para utilizarlo como sistema real.

Los lenguajes de especificaciones ejecutables son herramientas de prototipo más sofisticadas. Cambian el desarrollo del sistema en un proceso iterativo donde el sistema se va especificando y las especificaciones se van ejecutando para determinar si el sistema está completo y es correcto. Después, basándose en la experiencia al utilizar el prototipo, las especificaciones se refinan y vuelven a ejecutarse. El proceso iterativo continúa hasta que el sistema es capaz de funcionar de forma que satisfaga totalmente todos los requerimientos del usuario.

3.3.4. La Simulación

Ciertos tipos de sistemas de información están incorporados en otros sistemas. Tales sistemas, llamados de tiempo real o embebidos, deben interactuar con el sistema del entorno mediante señales y controles.

En los sistemas de tiempo real o embebidos es importante simular también como prototipo el sistema en los estadios del análisis y de diseño. En tanto que un prototipo es un modelo de sistema de información, la simulación es un modelo del entorno al cual el sistema de información debe responder y controlar. La simulación proporciona una visión previa del comportamiento del sistema de información en su entorno verdadero de tiempo real. Las entradas verdaderas en el tiempo real desde el entorno deben estar provistas de mecanismos para verificar la exactitud e integridad del modelo del sistema. Con la ayuda de las herramientas CASE de simulación, el programador puede encontrar los errores de diseño incluso antes del comienzo de la fase de codificación. Los sistemas de tiempo real o embebidos se tratan más adelante.

3.3.5 La Generación de Código

Al otro extremo del prototipo en el ciclo de vida del sistema de información está la generación de código. Un CASE automatiza la fase de implantación del programa partiendo de las especificaciones de diseño del programa. Se genera un esquema o un programa completo.

En el caso de la generación del esquema, se generan automáticamente código para una base de datos, fichero, pantalla y descripción de informes, más una salida comentada de la lógica de procedimiento. La lógica adicional del programa debe codificarse manualmente para completar el programa.

En el caso de la generación completa del programa, desde las especificaciones del diseño, se genera automáticamente el programa completo más su documentación. Lo cual incluye:

- Código ejecutable
- Lenguaje de control

- Accesos a las bases de datos y a los ficheros (pantallas, direccionamientos, descripciones de las bases de datos).
- Pantallas de ayuda
- Mensajes de error
- Documentación del usuario y del programa
- Casos de prueba.

Todos los componentes del programa completo se juntan y se almacenan en el depósito CASE para su fácil acceso, actualización sincronizada y posterior almacenamiento.

El código generado puede ser código objeto o código fuente. Muchos profesionales prefieren el código fuente, porque en el caso de necesidad puede ser comprendido y programado por programadores humanos. Además es más portátil y puede ser más compatible con el sistema de información existente. Los lenguajes más extendidos (Cobol, PL/1, C, ADA, Fortran, etc.) y los prototipos del suministrador son enormemente generados por el CASE.

El código se genera para operar en entornos como MVS, CICS, IMS/DC, DB2, DC1, VSAM, VM/SQL, DS, TSO, IDMS, ADABAS, DBS y OS/1100. La posibilidad de generar códigos para varios entornos es una parte muy importante de las prestaciones de un CASE.

El código objeto ofrece la ventaja de una mayor eficiencia, y probablemente llegue a estar más extendido en el futuro entre los usuarios de los CASE, a medida que se convence de la fiabilidad y eficiencia de la generación automática de código. Además, como la capacidad del CASE aumenta el nivel del mantenimiento de los sistemas desde las especificaciones de diseño, el código objeto generado se considerará más práctico que el código fuente.

3.3.6 El Soporte de la Metodología Estructurada

El CASE soporta metodologías estructuradas como análisis estructurado, diseño estructurado y programación estructurada al proporcionar herramientas para automatizar las metodologías. Hay dos niveles de automatización:

- 1.- La preparación de la documentación.
- 2.- Los pasos del proceso de la metodología.

La preparación automatizada de la documentación significa proporcionar soporte gráfico para el dibujo de diagramas estructurados (de flujo, de entidad-relación, de transición de estado y de acción). también significa la producción automática de especificaciones textuales (miniespecificaciones y pseudo código). Las especificaciones textuales se emplean para proporcionar una información más detallada sobre los procedimientos del programa y de la estructura de los datos referenciados en los diagramas estructurados de un nivel más alto.

Comoquiera que las diferentes metodologías estructuradas soportadas por un CASE particular estarán determinadas por los tipos de diagramas estructurados y las convenciones de notación que pueda ofrecer. Por ejemplo, la metodología de diseño estructurado de Yourdon emplea un diagrama de estructura (struct chart) derivado de un diagrama de flujo para representar el diseño del programa; la metodología de diseño de Jackson utiliza el diagrama estructurado en árbol y la metodología de

desarrollo de sistemas de datos estructurado en árbol y la metodología de desarrollo de sistemas de datos estructurados de Orr emplea los diagramas de Warnier-Orr.

El segundo nivel de soporte de metodología, la automatización de los pasos del proceso de la metodología, significa que el CASE guía al usuario en el correcto uso de la metodologías estructurada. Esto quiere decir que al menos cierto nivel de comprensión de la metodología este incorporado en las herramientas del CASE. Esto podría ser tan simple como incorporar paneles de ayuda para describir cada paso de la metodología o de listas de comprobación que incluyan las entradas y salidas requeridas en cada paso. Podría incluir un mecanismo de comprobación para asegurar que cada salida requerida por la metodología está presente, correcta y completa antes de permitir al usuario ir al paso siguiente. En este caso, al usuario no solamente se le guía, sino que además, se le fuerza a seguir los pasos con un orden y método normalizado. El propósito, en definitiva, es normalizar y sistematizar el proceso de desarrollo del sistema.

Existen dos escuelas de pensamiento sobre si el segundo nivel de automatización de la metodología estructurada debe o no estar incorporado en la herramienta CASE. El argumento a favor es que se introduce un control sobre el proceso de desarrollo del sistema y la normalización impuesto por la herramienta CASE. Además, puede realizarse un control más completo y exhaustivo sobre la calidad e integridad, pues muchas comprobaciones de calidad son específicas de la metodología. Por ejemplo, las medidas de acoplamiento y cohesión son comprobaciones de diseño que forman parte no solamente de la metodología de diseño estructurado de Yourdon. En general, cuanto más pueda definirse específicamente y mayor potencia habrá en la automatización del sistema.

Por otra parte, el argumento en contra es que la no incorporación del proceso de la metodología en la herramienta CASE ofrece al usuario la flexibilidad de elegir la metodología (o parte de la metodología apropiada para desarrollar un sistema en pantalla). Esto es un débil argumento, ya que la mayoría de los profesionales del desarrollo apenas conocen bien una metodología y mucho menos están al día de las muchas metodologías para elegir la más apropiada para el desarrollo de un proyecto en particular.

Además, incorporar los pasos de la metodología en la herramienta CASE no significa que el usuario no tenga la opción de ignorar la sugerencia de la herramienta en cuanto al siguiente paso apropiado o un aviso sobre la deficiencia en la calidad.

3.3.6.1 La Clasificación de las Metodologías Estructuradas.

Las metodologías estructuradas pueden clasificarse en pertenecientes a la escuela de ingeniería de sistemas y la de la escuela de la ingeniería de la información; en orientadas a los procedimientos, a los datos o a la información, y en las que desarrollan sistemas de tiempo real o de información.

3.3.6.2. La Ingeniería de Sistemas

La ingeniería de sistemas es una fórmula descendente (top-down) de la implantación por fases para el desarrollo de programas. Sus premisas básicas se presentan en el cuadro 3.3.2. La ingeniería de sistemas es un proceso de desarrollo paso a paso que comienza con la visión funcional más general de lo que puede hacer el programa, descompone esta visión en sub-funciones y repite el proceso por cada sub-función hasta que todas las sub-funciones sean lo suficientemente pequeñas para ser implantadas en los códigos de los programas. El resultado es un programa modular estructurado jerárquicamente.

Aunque el diseño de las componentes funcionales y el diseño de la estructura de los datos se realizan en paralelo, el diseño de los procedimientos para proteger la independencia y claridad de los componentes funcionales del programa. De esta forma, la ingeniería de sistemas se enfoca hacia la visión funcional del programa.

Cuadro 3.3.2 Las premisas básicas de la ingeniería de sistemas.

- Un planteamiento descendente (top-down) de implantación por fases requiere un procedimiento gradual desde el nivel más alto de control y de definición de datos, descendiendo hasta módulos funcionales y las estructuras de los datos.
- Un aspecto muy importante del proceso descendente es la insistencia en la firmeza de los requerimientos.
- La solución al sistema se define en funciones; los datos se diseñan para preservar la funcionalidad del sistema y la independencia de los módulos del programa. Por tanto, los requerimientos de los datos se descubren a través del análisis de las funciones requeridas por el sistema.

Cuadro 3.3.3 Los diagramas de ingeniería de sistemas.

- Diagrama de flujo de datos: muestra los procesos funcionales desde el nivel más alto en un sistema y el flujo de datos desde la adquisición a través del proceso y las eventuales salidas.
- Diagramas estructurados en árbol: muestran las relaciones entre los que diseñan el programa y muestran la estructura jerárquica de los datos.
- Diagrama detallado de la lógica de procedimiento: muestran detalladamente la lógica del programa: secuencia, selección iteración.
- Diseño de pantallas y de informes: muestran el diseño del interfaz del usuario.

La ingeniería de sistemas representa uno de los primeros intentos para estructurar el proceso del desarrollo del software. Desde los años setenta se ha avanzado en el cambio de del desarrollo del sistema de información desde un arte manual a una disciplina casi mecanizada con el objetivo de introducir un enfoque normalizado del desarrollo del sistema. Como explica Bauer, la ingeniería de sistemas es "el establecimiento y empleo de los principios (métodos) de la ingeniería para obtener económicamente un sistema que sea fiable y que trabaje sobre máquinas reales. Los conceptos y técnicas involucrados en la ingeniería de sistemas incluyen:

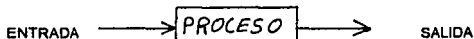
- El diseño descendente (top-down)
- La programación estructurada
- La modularidad
- EL refinamiento progresivo y la descomposición funcional.
- La métrica de la calidad del sistema de información
- La abstracción de los lenguajes de programación
- Aproximación descendente orientada al programa de desarrollo de sistemas basada en el análisis de flujo de los datos.
- Metodología para la gestión de la programación.

Las metodologías de análisis estructurado de Demarco y Gane Searson y la metodología de diseño de Yourdon son ejemplos de metodologías estructuradas populares pertenecientes a la escuela de la Ingeniería de sistemas.

La Ingeniería de sistemas es una disciplina general de desarrollo que puede aplicarse al desarrollo de los sistemas de información y a los sistemas de tiempo real. Por ejemplo, las metodologías de diseño para sistemas de tiempo real de Ward_Mellor y de Hatley están basadas en la metodología de diseño estructurado de Yourdon-Constantine empleada para desarrollar sistemas de información.

3.4 El desarrollo de Sistemas Orientado al Procedimiento Frente al Orientado a los Datos

La ingeniería de sistemas está fundada sobre el modelo básico de entrada/salida de un sistema:



Los datos se introducen en el sistema como entradas, se opera con ellos (se transforman) por medio de un proceso y lo abandonan como salidas.

Este modelo básico se utiliza en todas las metodologías de diseño estructurado. Sin embargo, el orden en que se construye el modelo depende de la metodología en particular que se sigue. La ingeniería de sistemas tradicional trata la parte del proceso como la fundamental del modelo del sistema por tanto se describe como un enfoque del desarrollo del software orientado al procedimiento.. Los datos de derivan de esa función.

En un enfoque de desarrollo orientado a los datos, la parte del modelo correspondiente a las entradas y las salidas se considera como la más importante. Por tanto, las estructuras de los datos se definen primero y los componentes procedimentales se derivan de las estructuras de los datos. En el cuadro 3.4 muestra las premisas básicas del enfoque de desarrollo dirigido por los datos. Los tipos de diagramas necesarios para soportar un enfoque de desarrollo dirigido por los datos son diagramas de árbol estructurado jerárquicamente para representar las estructuras de los datos y de los programas y el diagrama detallado de la lógica de procedimiento.

La metodología de diseño de Jackson es un ejemplo de metodología orientada a los datos. En el enfoque de Jackson, las estructuras de los datos de entrada y salida se definen primero, después, se define la estructura del programa mezclando todas las estructuras de los datos del programa.

La metodología de diseño de Warnier-Orr es otro ejemplo del enfoque orientado a los datos. Pero a diferencia del enfoque de Jackson, la formulación de Warnier-Orr se centra en las salidas del programa. Es una metodología orientada a las salidas, utilizando la formulación de Warnier-Orr, se comienza por la definición de las salidas del sistema hasta definir las partes del diseño correspondiente a los procesos y a las entradas.

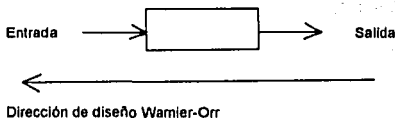
Cuadro 3.3.4 El enfoque dirigido por los datos para el desarrollo de sistemas

Las premisas básicas son:

- La estructura de control del programa debe ser jerárquica y debe derivarse de la estructura de los datos del programa.
- El proceso de diseño consiste en definir primero las estructuras de los datos, mezcladas luego toda una estructura jerárquica del programa y después ordenar detalladamente la lógica procedimental para que se ajuste a esta estructura.
- El diseño lógico debe estar separado del diseño físico y precederlo.

Los tipos de diagramas necesarios para soportar un enfoque dirigido por los datos son:

- Diagrama estructurado en árbol jerárquico
- Diagrama detallado de la lógica del procedimiento.



El enfoque del desarrollo en la información centrado en la información, un modelo de dato lógico que presenta la información utilizada a través de una organización es el punto de arranque de todo el desarrollo del sistema. El proceso empieza con un análisis de alto nivel de una corporación, sus objetivos comerciales y las necesidades que muestra información estratégica. Una visión global de las necesidades de información se presenta en un modelo que muestra todas las entidades de datos básicos de la organización y sus relaciones entre sí. Después, basándose en este modelo se constituyen individualmente los sistemas de información de la organización. Así los procedimientos se derivan de los datos.

3.4.1 La ingeniería de la información

La ingeniería de la información es un ejemplo de una formulación de desarrollo centrada en la información.

La ingeniería de la información es tanto un enfoque comercial como de ingeniería como una formulación para la construcción de sistemas de información. Es una formulación comercial con una

planificación estratégica de la organización. Es una formulación de ingeniería porque proporciona un procedimiento paso a paso para construir sistemas de información.

La ingeniería de información es una disciplina más reciente que la ingeniería de sistemas de información. En cierto sentido tiene un enfoque más amplio que la ingeniería de sistemas de información. Es una construcción del sistema, más alto (con la planificación estratégica) que la ingeniería de sistemas de información. Sin embargo, en la fase de diseño del programa, su formulación del desarrollo del programa es básicamente la misma que la ingeniería de sistemas de información.

En otro sentido, sin embargo, la ingeniería de la información tiene un enfoque más estrecho que la ingeniería de sistemas de información. Se emplea aquélla para construir sistemas de información, mientras que la ingeniería de sistemas de información se emplea para desarrollar toda clase de sistemas (tanto en tiempo real como comerciales).

La ingeniería de la información es un enfoque orientado a la información para el desarrollo del software y por tanto difiere de los enfoques orientados a los datos, como la de Jackson y de Warnier-Orr. La ingeniería de la información ha sido diseñada para con estructuras de datos no jerarquizadas. El modelado lógico de datos y de la normalización son pasos requeridos en la ingeniería de la información. La ingeniería de la información construye sistemas de información integrados, porque están construidos sobre el mismo modelo lógico de los datos.

Cuadro 3.3.5 Las premisas básicas de la ingeniería de la información

- Se necesita una estrategia general de desarrollo del sistema centrada en una planificación de los objetivos estratégicos del negocio para construir sistemas que satisfagan mejor las necesidades de la corporación.
- Sistemas de información que puedan integrarse mejor si los datos a compartir se controlan centralmente por ser parte de un mismo modelo lógico de datos.
- La representación lógica de datos es relativamente estable, en tanto que los procedimientos que utilizan los datos cambian frecuentemente. Por tanto, el modelo lógico de datos, que refleja lo que es la organización, no como trabaja, deberá ser la base del desarrollo del sistema.

Además el énfasis de la Ingeniería de la Información está en los datos desde el punto de vista de los requerimientos comerciales y no en los requerimientos del sistema de Información; Todo el enfoque del desarrollo de la ingeniería de la información se centra en la visión lógica de como el sistema utiliza los datos. La ingeniería de la información se trata más adelante en este capítulo, en la sección "Metodología de la ingeniería de la información de Martín".

3.5 Soporte de los Sistemas de Tiempo Real Frente a los Sistemas de Información.

Los diferentes tipos de sistemas requieren diferentes tipos de soporte de desarrollo. Dos ejemplos de tipos de sistemas son los de tiempo real/embebidos y los de información. El cuadro 3.6 muestra las diferencias principales entre ambos.

3.5.1 Los Sistemas de Tiempo real

Los sistemas de tiempo real típicamente son los sistemas que se controlan, y son controlados por, eventos externos. La respuesta a tales eventos (por ejemplo, señal, disparador, disponible/no disponible) de una forma oportuna es una de las principales funciones de un sistema de tiempo real. Cuando el sistema reconoce un evento al recibir una señal de entrada, realiza ciertos cálculos y actividades lógicas para responder al evento produciendo una salida.

Los ejemplos de sistemas de tiempo real incluyen: la navegación aérea, las redes de comunicaciones, los sistemas de software, el control de procesos de fabricación y los procesos químicos.

Para especificar los requerimientos de un sistema de tiempo real se incluyen los conceptos para:

- El manejo de las interrupciones.
- La comunicación y la sincronización entre tareas.
- El proceso concurrente.
- La respuesta oportuna a los eventos externos.
- Los requerimientos y las restricciones de los sistemas de información.
- Las interacciones entre el sistema/entorno.
- Los datos continuos y discretos.

El diseño de un sistema de tiempo real debe poder representar procesos que puedan ser interrumpidos por eventos externos y que pueden procesarse concurrentemente en distintos ordenadores.

Para representar los requerimientos de los sistemas de tiempo real se emplean tipos especiales de diagramas estructurados, entre los que se encuentran los de flujo de control, de transmisión de estado, gráfica de contexto, matrices estado/suceso, y las tablas de decisión. Por ejemplo, el diagrama de control de flujo se emplea para mostrar los procesos del sistema, el paso de las señales de los datos de entrada/salida entre los procesos y las relaciones oportunas entre los procesos. El diagrama de transición de estado mostrado en la figura 3.3.10 se utiliza para mostrar las condiciones y acciones que ocurren en el sistema al pasar de un estado a otro. La matriz proporciona una información más detallada al describir la lógica condicional empleada en un estado del sistema. El diagrama de contexto, como el mostrado en la figura 3.3.11, describe los límites del sistema de software y hardware

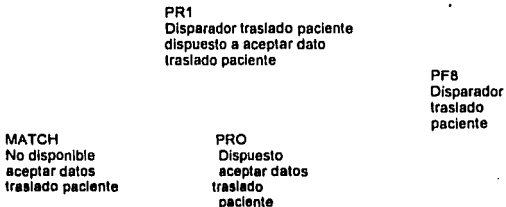


Figura 3.3.10 Este diagrama de transición de estado, que se utiliza para mostrar las condiciones y acciones que origina el cambio de un estado a otro del sistema.

La mayoría de estos diagramas son variaciones de los diagramas estructurados empleados para describir los sistemas de información. De modo similar, las metodologías de análisis y de diseño utilizadas en sistemas de tiempo real son variaciones de las utilizadas en sistemas de información como los de DeMarco, Yourdon y Jackson.

Por ejemplo, la metodología de Hatley es una extensión de la metodología de diseño de Yourdon-Constantine. La metodología de diseño de Jackson puede utilizarse tanto para diseñar sistemas de tiempo real de información.

Sistemas de Información	Sistemas de tiempo real
Dirigido por los datos.	Dirigido por los eventos
Estructuras complejas de los datos	Estructuras simples de los datos
Grandes volúmenes de datos de entrada	Pequeña cantidad de datos de entrada
Operaciones de entrada/salida intensivas	Operaciones de cálculo intensivas
Independiente de la máquina.	Dependiente de la máquina.

Cuadro 3.3.6 Características de los sistemas de tiempo real y los de la información

3.6 Las Metodologías Estructuradas de Utilización Más Extendida

En el cuadro 3.3.7 se listan las metodologías estructuradas de desarrollo más utilizadas de acuerdo con una investigación realizada entre más de mil empresas en E.U. En el mismo cuadro también se listan las principales metodologías que las organizaciones planean adoptar de acuerdo con la misma investigación. Obsérvese que la ingeniería de información, aunque no está dentro de las más utilizadas es una que en las organizaciones planean adoptar como estándar. esto probablemente se deba a que la ingeniería de información es una metodología más reciente en relación con las otras en el cuadro 3.3.7. Está ganado aceptación debido a la incorporación en su metodología del modelado de datos y la planificación estratégica.

Cuadro 3.3.7 Las metodologías más utilizadas

Las metodologías de desarrollo más utilizadas en los estados Unidos

• Diseño estructurado de Yourdon	29.1 %
• Análisis estructurado de Gane-Searson	13.8 %
• Análisis estructurado de DeMarco	12.3 %
• Diseño estructurado de Orr	7.2 %
• Diseño estructurado de Jackson	2.4 %

Elección de normalización de la metodología de desarrollo

• Yourdon	24.9 %
• Gane-searson	11.2 %
• Ingeniería de la información	11.2 %
• DeMarco	9.5 %
• Orr	7.1 %
• Jackson	3.6 %

El resto del capítulo se dedica a una breve exposición de los pasos de proceso de cada una de estas metodologías. En el cuadro 3.3.8 se clasifican estas metodologías estructuradas.

Cuadro 3.3.8 clasificación de las metodologías

Metodología	Clasificación
DeMarco Análisis estructurado	Ingeniería de software Orientado al procedimiento Sistemas de información Informal (extensiones tiempo real)
Gane-Searson Análisis estructurado	Ingeniería de software Orientado al procedimiento Sistemas de información Informal (extensiones tiempo real)
Yourdon Diseño estructurado	Ingeniería de software Orientado al procedimiento Sistemas de información Informal (extensiones tiempo real)
Jackson Diseño estructurado	Ingeniería de software Orientado a los datos Orientado a las salidas Sistemas de información Informal
Martín Ingeniería de la información	Ingeniería de información Orientado a la información Sistemas de información Informal

3.6.1 El Análisis Estructurado

El análisis estructurado propone un enfoque sistemático por pasos para la realización del análisis del sistema y la producción de las especificaciones del sistema. Hay dos versiones similares de análisis estructurado:

- Gane-searson
- DeMarco

Ambas representan una disciplina estructurada basada en los siguientes conceptos estructurados:

- Organización jerarquizada descendente
- Herramientas gráficas de comunicación/documentación

El análisis estructurado utiliza un método descendente de descomposición funcional para definir los requerimientos del sistema: La especificación del sistema producido por el análisis estructurado es un modelo descendente particionando del sistema a construir. El suministro de una descripción de los requerimientos del sistema a construir es la conexión entre el análisis y el diseño.

La unión entre el análisis y el diseño se proporciona con la descripción de los requerimientos del sistema a construir.

El objetivo primario del análisis es producir una especificación del sistema que defina la estructura del problema a resolver según la visión del usuario.

El propósito principal del análisis es producir una especificación del sistema que defina la estructura del problema a resolver según la visión del usuario. El propósito del diseño es definir la estructura de la solución de forma que sea compatible con la estructura del problema y con los requerimientos del usuario. Los defensores del análisis estructurado sugieren que al usar el mismo método de construcción (descomposición funcional descendente) en la especificación y en el diseño, los dos pueden unirse para representar un sistema que satisfaga las necesidades y expectativas del usuario.

3.6.2 La Especificación Estructurada

La especificación del sistema producida por el análisis estructurado es un modelo lógico, gráfico, descendente y jerárquico de los procesos del sistema y de los datos utilizados por estos procesos. Se compone de diagramas de flujo de datos, un diccionario de datos y las especificaciones de los procesos.

3.6.3 El diagrama de Flujo de Datos

Un diagrama de flujo de datos es un diagrama estructurado que representa los procesos (funciones o procedimientos) de un sistema y los datos que les conectan. Es la herramienta central del modelado del análisis estructurado y se emplea para fraccionar jerárquicamente el sistema en procesos. En los niveles inferiores, el proceso de nivel más alto se expande para ver si hay otros procesos y los datos más detallados involucrados.

3.6.4 El Diccionario de Datos

El diccionario de datos es un conjunto de definiciones de todos los datos que aparecen un diagrama de flujo de datos, en datos almacenados o como flujo de datos. La definición de cada dato consta de los componentes del dato que constituyen el dato y las relaciones entre ellos.

FICHERO DE CLIENTES	=	(registro de clientes)
REGISTRO DE CLIENTES	=	nombre-cliente + dirección-cliente información-del-pago + ordenes-de-pago + tipo-cliente

3.6.5 Las Especificaciones de Proceso

Una especificación de proceso (o miniespecificación) describe lo que ocurre dentro de un cuadro de proceso en un diagrama de flujo de datos. Una miniespecificación tiene generalmente una página de extensión y explica como se transforma los datos de entrada en datos de salida. La miniespecificación se escribe en lenguaje estructurado (seudocódigo) como una tabla de decisión o como un diagrama de acción. Se crea una miniespecificación para cada proceso definido en el diagrama de flujo de nivel en la especificación del sistema.

3.6.6 Los Pasos del Análisis Estructurado.

Como se muestra en el cuadro 3.3.9, DeMarco define el análisis estructurado en siete pasos:

Cuadro 3.3.9 Los pasos en los procesos de análisis estructurado

Los siete pasos del análisis estructurado de Demarco	Los cinco pasos del análisis estructurado de Gane y Searson
Paso 1: Construir el modelo físico	Paso 1: Construir un modelo lógico en curso
Paso 2: Construir un modelo lógico a partir de modelo físico.	Paso 2: Construir un modelo lógico del nuevo sistema.
Paso 3: Construir un modelo lógico del nuevo sistema Construir una especificación estructurada que incluye los diagramas de flujo de datos	Construir una especificación estructurada conteniendo los diagramas de flujo de datos, un diccionario de datos y las especificaciones de proceso. un diccionario de datos y las especificaciones de procesos. Construir un modelo lógico de datos que exprese en tercera forma normal el contenido de los datos almacenados
Paso 4: Crear una familia con los nuevos modelos físicos	Paso 3: diseñar la base de datos física
Paso 5: Estimar los costos y los tiempos para cada modelo	Paso 4: Crear un nuevo modelo físico del sistema.
Paso 6: Seleccionar un modelo.	Paso 5: Empaquetar la especificación en sub-sistemas.
Paso 7: Empaquetar la especificación en sub-sistemas	
Paso 1	

Documenta la forma en que las cosas en curso se realiza en el entorno actual del usuario. La salida de este paso es el diagrama de flujo físico los datos actuales. Muestra la situación, el personal, los nombres y los procedimientos manuales y automatizados, y utiliza los términos de los usuarios.

Paso 2

Utiliza el diagrama físico de flujo de datos del paso 1 para crear el modelo lógico del sistema actual. El modelo lógico, igual que el físico se representa como un conjunto de niveles jerarquizados de los diagramas de flujo de datos. La diferencia entre los dos modelos está en que el modelo físico muestra los detalles físicos de cómo un usuario particular hace las cosas, mientras que el modelo lógico muestra lo que se hace en un nivel abstracto, distinguiendo cualquier estado particular de los procedimientos fundamentales del negocio.

Paso 3

Desarrolla el nuevo modelo lógico del sistema a construir, siendo el punto de partida el modelo lógico del paso 2. Se modifica para incorporar los cambios para el nuevo sistema. Indica los procesos manuales y los automatizables y se representa como un conjunto por niveles de diagramas de flujo de datos que describen el sistema en variados niveles de detalle. El paso 3 conlleva la mayor parte del esfuerzo de trabajo del proceso de análisis estructurado.

Paso 4

Creas un nuevo modelo físico del sistema a construir identificando el interfaz hombre/máquina. Puede crearse más de un modelo posible para representar los distintos grados de automatización.

Paso 5

Produce una estimación de los costes y una planificación de cada uno de los modelos físicos alternativos desarrollados en el paso 4.

Paso 6.

Basándose en la estimación obtenida en el paso 5, se selecciona un modelo físico para utilizar en la especificación de las funciones y requerimientos del sistema a construir.

Paso 7.

Se empaqueta en su forma final la especificación estructurada, que incluye un conjunto por niveles de diagramas de flujo de datos, el diccionario de datos y las miniespecificaciones.

Gane y Searson definen un proceso similar para el análisis estructurado (ver cuadro 3.9). La principal diferencia es que Gane y Searson incluyen un paso del modelado de datos almacenados mostrados en los diagramas de flujo de datos, posiblemente en tercera forma normal. Gane y Searson incluyen el modelado de datos en el paso 2 de su versión de análisis estructurado. Después, en el paso 3, el modelo lógico de datos se emplea como entrada para el diseño físico de la base de datos.

3.7 El Diseño Estructurado de Yourdon

La metodología de diseño estructurado de Yourdon proporciona un proceso para el diseño paso a paso de sistemas y programas detallados. Unos pasos implican el desarrollo del diseño, otros la medición y la mejora de la calidad de diseño. Cada paso es soportado por un conjunto de estrategias de diseño, guías y técnicas de documentación. Por ejemplo, las estrategias de diseño, las de análisis de transformación, las de transacción guían al diseñador las decisiones de diseño. Las medidas de diseño, como el de acoplamiento y cohesión proporcionan al diseñador técnicas para medir la calidad del diseño. El principal producto obtenido por los procesos de diseño estructurado es un diagrama de estructura mostrando los componentes de procedimiento del programa, su ordenación jerárquica y los datos conectados a ellos.

3.7.1 El Diagrama de Estructura

El diagrama de estructura es un diagrama jerárquico o de árbol que define la arquitectura global de un programa mostrando sus componentes de procedimiento y sus interrelaciones. Los bloques básicos para construir el diagrama de estructura son cajas que representan los componentes de procedimiento y las flechas que los conectan. Los datos que pasan en los componentes de procedimientos se llaman pares y se escriben a lo largo de flechas de conexión.

3.7.2 Los Pasos de Diseño Estructurado de Yourdon

La figura 3.3.14 muestra los cuatro puntos básicos del proceso de diseño estructurado de Yourdon.

Paso. 1. Trazar el diagrama de flujo de datos

El primer paso en el proceso de diseño estructurado es representar el problema de diseño como el flujo de datos a través de un sistema. El sistema se compone de procesos que operan (transforman) con los datos. Estos procesos y los datos que los enlazan forman las bases para definir los componentes del programa. Se utiliza un conjunto por niveles de diagramas de flujo de datos que los enlazan para representar la primera visión del diseño del programa.

Paso 2. Trazar el diagrama de estructura

El segundo paso es representar el diseño del programa como una jerarquía de componentes de procedimiento. Se emplea un diagrama de estructura para mostrar esta visión del diseño. El diagrama de estructura se deriva del diagrama de flujo de datos obtenido en el paso 1. El diseño estructurado proporciona dos estrategias de diseño para guiar la transformación de uno o varios diagramas de flujo de datos a un diagrama de estructura: los análisis de transformación y de transacción. Estas dos estrategias proporcionan dos modelos de estructuras que pueden utilizarse individual o conjuntamente para dirigir el diseño jerárquico, así como un proceso paso a paso de transformación por cada estrategia.

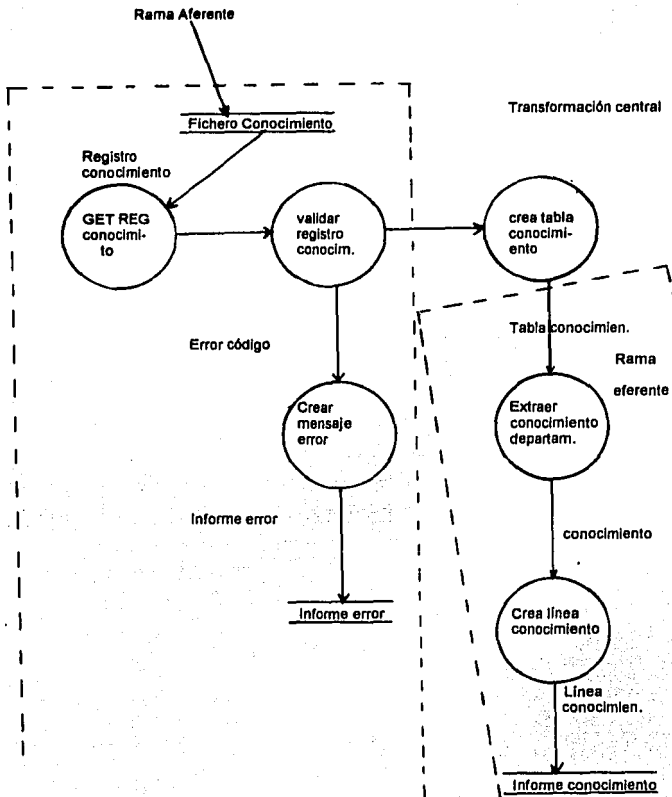


figura 3.3.15 diagrama de flujo de datos del sistema de conocimientos de empleados, indicando las ramas aferentes (entrada), las ramas eferentes (salida) y la transformación central (proceso lógico).

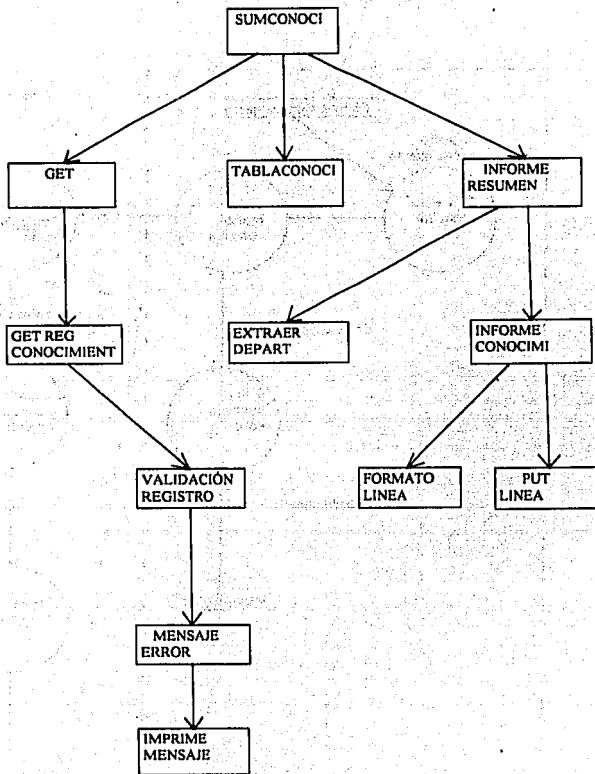


Figura 3.16 Diagrama de estructura del sistema de conocimientos de empleados derivado del diagrama de flujo de datos mostrado en la figura 3.3.15

El análisis de transacción se emplea cuando se diseñan programas con proceso de transacciones. El diagrama de estructura general para un programa con procesos de transacciones se muestra en la figura 3.3.17. En la parte superior de los diagramas de estructura está el módulo de la transacción central y debajo hay varios módulos de transacciones. Como se muestra en la figuras 3.3.18 y 3.3.19, el diagrama de estructura se obtiene definiendo la transformación central y las transacciones.

Paso 3: Evaluación del diseño.

El tercer paso en el diseño estructurado es la medición de la calidad del diseño. El acoplamiento y la cohesión son las dos técnicas de medición del diseño proporcionadas por el diseño estructurado.

El acoplamiento mide el grado de independencia entre los componentes del procedimiento (módulos) en el diagrama de estructura. Cuando existe poca interacción entre dos módulos, éstos describen como ligeramente acoplados. Cuando la interacción entre dos módulos es grande, los módulos se describen como estrechamente acoplados. Un diseño de alta calidad significa que los módulos están lo más ligeramente acoplados posible. Hay cinco niveles de acoplamiento: datos, por estampado, de control, común y por contenido. El acoplamiento por datos es el más ligero y por tanto el mejor tipo. Mientras que el acoplamiento por contenido es el más estrecho y por tanto, el peor tipo.

La cohesión mide la fuerza de las relaciones entre los elementos dentro de un módulo cuando más fuerte, mejor. Hay siete niveles de cohesión: funcional, secuencial, de comunicación, de procedimiento, temporal, lógica y de coincidencia. La cohesión funcional es la mejor, la coincidencia, la peor.

Paso 4: La preparación del diseño para la implantación

El último paso de un diseño estructurado es la preparación del diseño para la implantación. Esto se llama empaquetar el diseño. Empaquetar es el proceso de dividir el diseño de programa lógico en unidades físicas de implantación llamadas unidades de carga. Es un diseño físico del programa .

Es una descripción de la tradicional metodología de diseño estructurado de Yourdon . Hay variaciones y actualizaciones, algunas de las cuales han extendido las metodologías de diseño de Yourdon (que fue ideada originalmente para el diseño de sistemas de información) para soportar el diseño de sistemas de tiempo real.

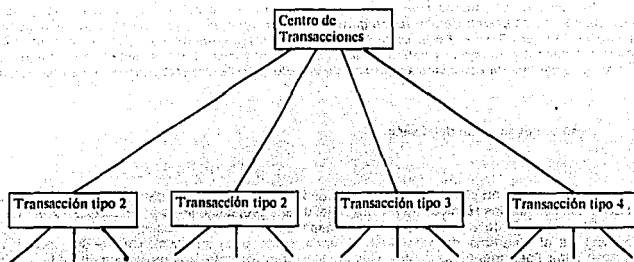


FIGURA 3.3.17 El modelo general de diagrama de estructura para un programa con proceso de transacciones, tiene un módulo "centro de transacciones" en lo más alto y por debajo un módulo de transacción para cada tipo distinto de transacción.

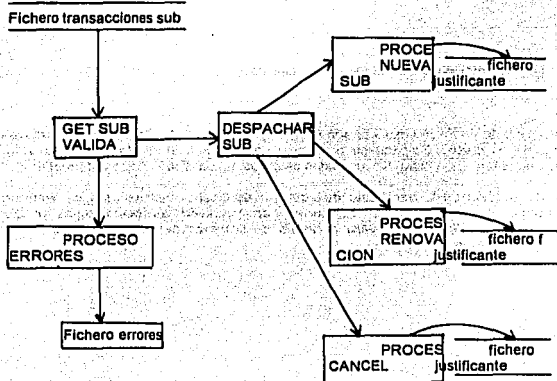


Figura 3.3.18 Diagrama de flujo de datos del sistema de suscripciones, mostrando el centro de transacción y los módulos de transacciones.

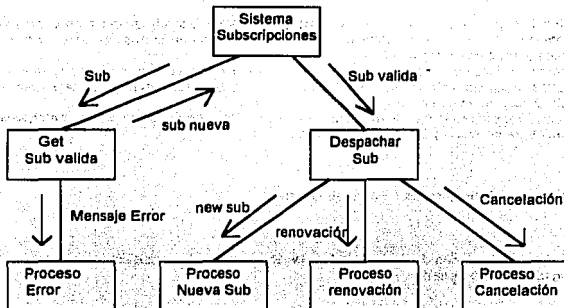


Figura 3.3.19 Diagrama de estructura a alto nivel derivado del diagrama de flujo de datos utilizando la estrategia de diseño de transacción

3.8 La Metodología de Diseño de Jackson

Como la metodología de diseño de Yourdon, la de Jackson proporciona un proceso descendiente por pasos de diseño. La principal diferencia entre las dos, es que la de Jackson está basada en el análisis de la estructura de los datos, mientras que la de Yourdon se basa en el análisis de flujo de los datos. La de Jackson está orientada a los datos y la de Yourdon, a procedimiento. La formulación de Jackson aboga por una visión estática de las estructuras, mientras que la de Yourdon utiliza una visión dinámica del flujo de datos.

La metodología de diseño de programas de Jackson es una técnica dirigida por los datos. Su objetivo es obtener la estructura del programa derivándola de la(s) estructura(s) de los datos, asumiendo que se ha especificado totalmente el problema.

Jackson considera el programa como un proceso secuencial. Tiene entradas y salidas que le consideran como corrientes secuenciales de la corriente de datos y después ordenar el proceso lógico (operaciones) para ajustar las estructuras de datos. Como se muestra en el cuadro 3.3.11, el proceso de diseño de Jackson consta de cuatro pasos.

Cuadro 3.3.11 Pasos básicos en el proceso de diseño de Jackson

- Paso de datos: Dibuja un diagrama estructurado por cada entrada y salida de datos en el programa.
- Paso de programa: Toma todas las estructuras de datos definidas en el paso de datos y forma con ellas una única estructura de programa.
- Paso de operación. Hace una lista con las operaciones necesarias para producir las salidas a partir de las entradas, después, asigna cada operación de la lista a un componente de la estructura del programa (progresivamente desde la salida hasta la entrada).
- Paso de texto. Transcribe la estructura del programa a texto, añadiendo la lógica condicional que rige la ejecución de los ciclos y las estructuras de selección.

3.8.1 Diagramas Estructurados y Pasos de Jackson

La metodología de diseño de Jackson utiliza las técnicas de diagramación estructurada en árbol para describir un diseño de programa:

- Diagrama de red: El flujo de corrientes de datos entre programas (ver figura 3.3.20).
- Diagrama estructurado en árbol: Representación jerárquica de estructura de datos y programas (ver fig. 3.3.21 y 3.3.22).
- Texto: En forma de pseudocódigo (ver figura 3.25).

Paso 1: Paso de Datos

Primero se dibuja un diagrama de red de sistema mostrando todas las corrientes de datos de entrada y de salida (ver figura 3.3.20), después, cada corriente de datos se representa como una estructura jerárquica.

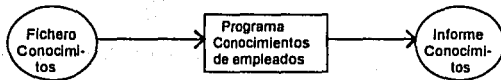


Figura 3.3.20. El diagrama de la red del sistema de conocimientos de los empleados muestra que hay una corriente de entrada, FICHERO CONOCIMIENTOS, una corriente de salida, INFORME CONOCIMIENTOS, y un programa, PROGRAMA CONOCIMIENTOS DE EMPLEADOS. En la terminología de Jackson se diría: EL FICHERO CONOCIMIENTOS se consume para producir el INFORME CONOCIMIENTOS.

Paso 2: Paso de programa

El segundo paso es combinar las estructuras de datos en una sola estructura de datos en una sola estructura de programa. Hay dos partes en el lapso de programación. Primera, todas las correspondencias entre los componentes de la estructura de datos se identifican encontrando la relación consumo/producción entre los componentes (ver figura 3.3.21). Se re construye la estructura del programa para que se corresponda con esas relaciones (ver figura 3.3.22).

La segunda parte consiste en verificar la corrección de la estructura de programa . Esto se consigue reduciendo la estructura del programa a sus estructuras de datos individuales. El proceso de verificación se aplica a cada corriente de datos para demostrar que el programa se puede reducir a cada una de sus estructuras de datos (ver figura 3.3.23). Si esto puede hacerse se asume que el diseño es correcto.

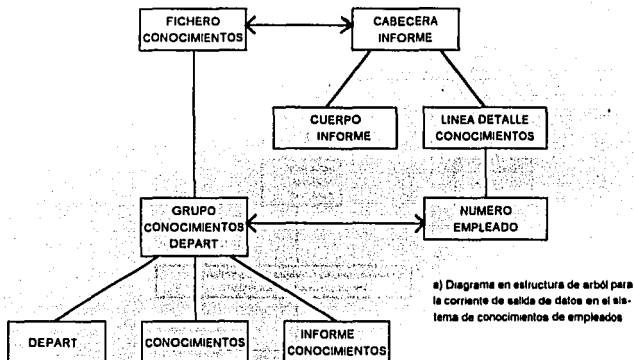


Figura 3.3.21 El programa de conocimiento de empleados tiene una corriente de entrada y una de salida. Se utiliza el diagrama estructurado en árbol para representar la estructura de los datos. Las flechas sombreadas muestran la relación consumo-producción entre los componentes en la estructura de datos.

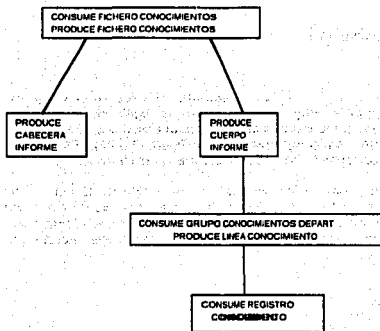


Figura 3.3.22 La estructura del programa del sistema conocimientos de empleados está compuesto por las estructuras de los datos de entrada y de salida mostrados en la figura 3.3.21.

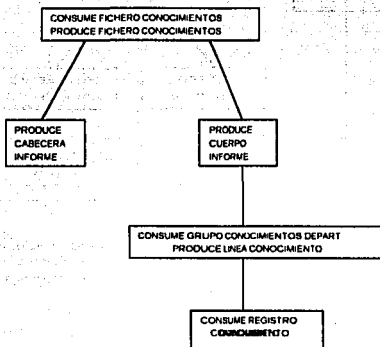


Figura 3.3.23 La exactitud de la estructura del programa se demuestra reduciéndolo a sus estructuras de datos. En el ejemplo la estructura del programa se reduce a la corriente de datos de salida, INFORME CONOCIMIENTOS. La estructura de los datos del INFORME CONOCIMIENTOS se muestra en la figura 3.21.

Paso 3: Paso de operaciones.

El paso de operaciones se compone de tres partes:

1.- Se listan las operaciones ejecutables requeridas por el programa para convertir su entrada en salida; esto se hace empezando en la salida y yendo hacia atrás hasta la entrada. Las operaciones se determinan por la referencia a las estructuras de datos, las cuales, a su vez, se reflejan en la estructura del programa.

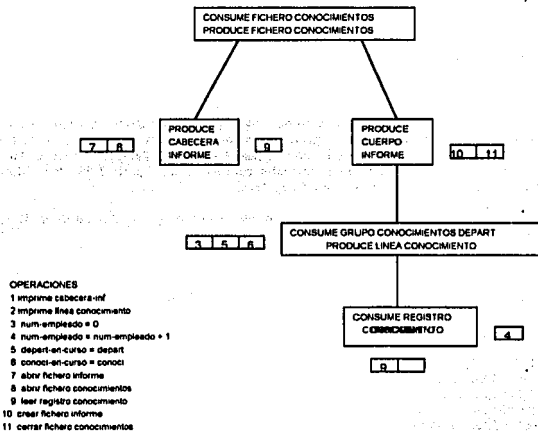


Figura 3.3.24 En el paso de operaciones se listan las operaciones ejecutables requeridas por el programa de conocimientos de los empleados.

2.- Cada operación se asigna al lugar apropiado de la estructura del programa.

3.- Se verifica la corrección comprobando si se producen todas las salidas y se consumen todas las entradas. La figura 3.3.24 muestra la estructura del programa incrementada con sus pasos de operación.

Paso 4: Paso del texto

El último paso en el diseño de Jackson es el texto. En este paso, el diagrama de estructura del programa, junto con sus correspondientes operaciones, se traduce a un texto de estructura.

El texto de estructura es un tipo formal de pseudocódigo. Tiene reglas estrictas para escribir el control de la secuencia, la selección y la iteración, pero las instrucciones a nivel de programación elemental se escriben informalmente.

Las consideraciones de control que rigen la ejecución de las estructuras selección e interacción se añaden a la estructura de texto en este paso. Al final del paso de texto, el diseño de programa está listo para su implantación. El diagrama de red del sistema, la estructura en árbol de datos, la estructura en árbol del programa y la estructura de texto conforman todo el paquete de diseño.

3.8.2 Procedimiento de Diseño Básico

El diseño de Jackson descrito antes se denomina proceso de diseño básico. Puede utilizarse en el diseño de programas complejos. Un programa complejo ha de dividirse antes en una secuencia de programas más simples, cada uno de los cuales se puede diseñar utilizando el procedimiento de diseño de Jackson para conducir el diseño de programas complejos. La figura 3.26 ofrece una visión más completa de los pasos del proceso de diseño de Jackson.

El proceso de diseño de Jackson se ha extendido al manejo de diseño de sistemas de tiempo real y a los sistemas de información.

3.9 Metodología de la Ingeniería de la Información de Martín

La metodología de la ingeniería de la información adoptada por James Martín es una formulación descendente por pasos para la construcción de sistemas de información. Los primeros pasos de esta metodología se centran en la información y el modelo de la empresa a alto nivel. El objetivo es gestionar el desarrollo del sistema y la iteración a través del control de los datos. Los datos compartidos en el sistema de información definen un modelo lógico y se controlan centralmente. Así, la cómo se utiliza y se comparte la información. Las funciones de alto nivel de la empresa se definen junto con la información del negocio. En los pasos posteriores, se definen e implantan los datos y la lógica del programa.

Las estructuras del programa se construyen en el nivel más alto de un modelo de la empresa que establece una infraestructura común de información que aúna los sistemas de información utilizados en toda la organización. Así pues, los procedimientos se derivan de los datos. El diseño lógico está separado del diseño físico y lo precede.

La figura 3.3.27 muestra los cuatro pasos principales de la metodología de la ingeniería de información de Martín. El cuadro 3.3.12 resume los tipos de diagramas estructurados utilizados en la metodología de la ingeniería de información de Martín.

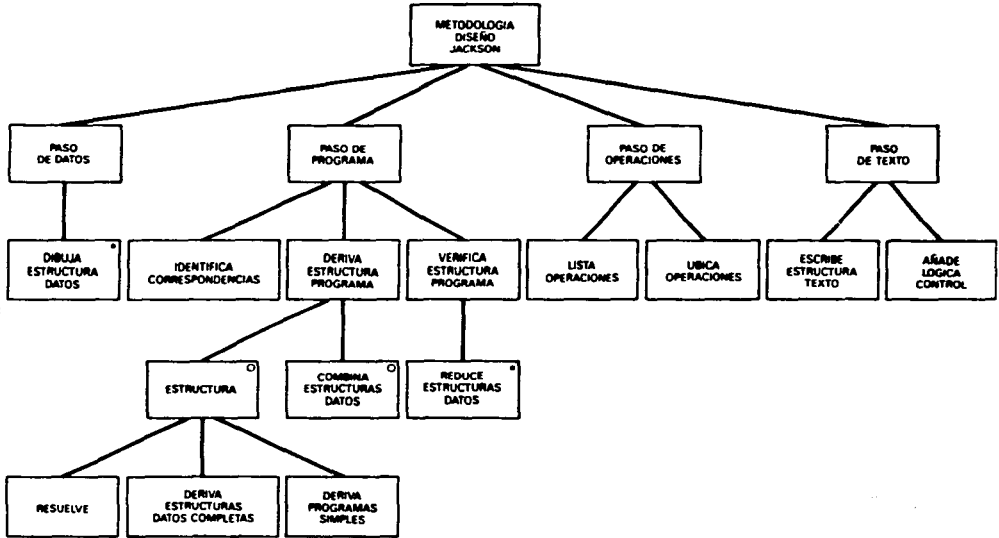


Figura 3.26. Diagrama estructurado en árbol para definir el proceso de Jackson para el diseño de programas

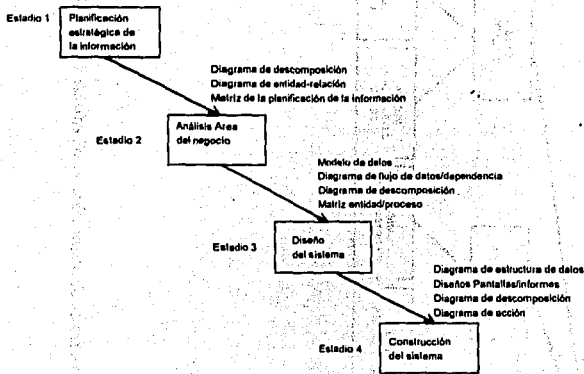


Figura 3.3.27 La metodología de la información de Martín, divide el proceso de desarrollo del software en cuatro fases básicas.

Cuadro 3.3.12 los diagramas de la Ingeniería de la Información de Martín

Los tipos de diagramas necesarios para soportar una formulación de ingeniería de la información para desarrollar un sistema son:

- Diagrama entidad/relación: muestra los tipos de entidades y sus relaciones.
- Diagrama de la estructura de los datos: muestra los tipos de entidades, sus atributos y sus relaciones.
- Diagrama de descomposición jerárquica estructurada en árbol: muestra el modelo corporativo de alto nivel y la estructura arquitectónica jerárquica de las actividades del negocio, de los procesos, de las aplicaciones, de los procedimientos y de los programas.
- Diagrama de flujo de datos y de dependencia: muestran el flujo de los datos entre procesos.
- Diseños de pantalla y de informes: muestran el diseño del interfaz del usuario.
- Diagramas de acción muestran detalladamente la lógica de los programas.

Estadio 1 Planificación de la estrategia de la información

Este estadio comienza con la creación de un plan estratégico de los sistemas del negocio en el que se definen los objetivos del negocio para los próximos cinco a diez años. Junto con el plan estratégico se construyen los modelos de alto nivel y datos de la empresa. El modelo de la empresa define las funciones básicas (por ejemplo, VENTAS) y la estructura de la organización. Un diagrama estructurado en árbol jerárquico, llamado diagrama de descomposición, es el que se emplea para definir las funciones del negocio y las estructuras de la organización. Para definir los tipos de datos de la organización se emplea un diagrama de entidad/relación.

Una entidad es cualquier información de la empresa que se pueda almacenar (como empleados, departamentos, clientes, facturas, etc.). Las entidades se descubren entrevistando los usuarios se analizan y luego se mezclan y clasifican para crear un modelo compuesto. Además, la ingeniería inversa de los sistemas existentes puede emplearse para revelar el modelo de datos existente, el cual puede utilizarse como punto de partida para crear el modelo de datos existente, el cual puede utilizarse como punto de partida para crear el modelo necesario para soportar la dirección estratégica de la empresa.

Finalmente, se determina en este estado el orden en el cual se van a desarrollar los sistemas de información que satisfagan los objetivos de la empresa. El usuario tiene una importante función activa en el estadio y en todo el proceso de desarrollo de los sistemas de ingeniería de la información.

Estadio 2: El análisis del área del negocio

Las funciones del negocio definidas en el estadio 1 se usan para dividir la empresa en áreas lógicas del negocio (por ejemplo, COMPRAS). después se definen y construyen los sistemas de información necesarios para soportar un área del negocio en particular.

Este estadio se concentra en la definición de los datos y procesos necesarios para satisfacer los objetivos de la empresa dentro de un área en particular del negocio. Una parte del diagrama entidad/relación desarrollada en el estadio 1 que concierne a un área en particular del negocio se extrae y desarrolla en un modelo de datos completamente normalizado. Se definen completamente las entidades, sus atributos y sus relaciones recíprocas. El diagrama de descomposición del estadio 1 que representa las funciones del negocio, se descompone en procesos por cada una de las áreas del negocio. El proceso puede representarse en diagramas de dependencias (un tipo de diagrama de flujo de datos), en diagramas de flujo de datos o en diagramas de descomposición. Se emplea una matriz de entidad/proceso para relacionar los datos con los procesos en los que intervienen.

Estadio 3: El diseño del sistema

El estadio 3 se ocupa de las consideraciones de diseño del sistema lógico. Los procesos del negocio definido en el estadio 2 conforman las bases para el desarrollo del sistema en este estadio. Se diseñan los procedimientos necesarios para realizar los procesos y las estructuras lógicas en sus datos. Se utiliza la descomposición funcional descendente para diseñar los procesos. En este estadio, la metodología de la ingeniería de la información se hace similar a la metodología tradicional de la ingeniería de software para el desarrollo de programas. El sistema diseñado se representa con diagramas de estructuras de datos, con diseños y con diagramas de actuación y descomposición. Los

diagramas de acción muestran detalladamente la lógica del procedimiento y los accesos a las bases de datos.

El prototipo del interfaz de usuario (en pantallas, diálogo o gráficos) se emplea para descubrir los requerimientos del usuario. Es necesaria la implantación del usuario para ayudar a construir y comprobar los modelos de prototipos.

Estadio 4: La construcción

Mientras que el estadio 3 trataba el diseño del sistema lógico, el estadio 4 se ocupa del diseño del sistema físico y de la implantación del programa. Durante el estadio 4, toda la información del sistema se entrega como una base de datos lógica, una base de datos física y un código documentado de trabajo. las herramientas CASE, como los lenguajes de cuarta generación, herramientas de soporte de decisiones y las generaciones de códigos, se emplean en el estadio 4.

10 La Metodología DSSD

La metodología DSSD (Data Structured Systems Development), metodología del desarrollo de sistemas estructurados de datos fue desarrollada por Jean Dominique Warnier en el CII Honeywell-Bull de París a finales de los años cincuenta. La metodología de Warnier, llamada LCP (Logical Construction of Programs), esta basada en el desarrollo de programas sobre la estructura de los datos del problema que le va a programar. A finales de los años setenta, Kent Orr modificó el LCP de Warnier creando una nueva metodología que llamó SPD (Structured Program Design), diseño estructurado de programas y que ha evolucionado en el DSSD.

10.1 El Diagrama de Warnier-Orr

El DSSD utiliza la teoría matemática de conjuntos para describir el diseño del programa. Un conjunto es una serie ordenada de objetos que comparten una o varias características comunes (por ejemplo, un conjunto de números enteros o el conjunto de departamentos de una compañía). En matemáticas, un conjunto se describe con la lista de sus miembros encerrada entre llaves. Por ejemplo, los departamentos de una compañía forman un conjunto indicado por:

{contabilidad, mercado, producción, compras}

El DSSD utiliza una notación similar para los conjuntos, excepto que se elimina la llave de la derecha y que los miembros se listan verticalmente:

{
contabilidad
mercado
producción
compras

El diagrama de Warnier-Orr, la herramienta central de la metodología DSSD, se compone de conjuntos anidados. Un ejemplo de diagramas de Warnier-Orr se muestra en la figura 3.3.21. Los diagramas de Warnier-Orr se emplean para mostrar la estructura jerárquica y el flujo de actividades, procedimientos o datos del proceso.

10.2 La Formulación Orientada a la Salida

El desarrollo de sistemas estructurados de datos es una metodología para el diseño de Jackson en el sentido de que ambas son enfoques orientados a los datos. En ambas la estructura del programa se deriva de la estructura de los datos, también en ambas se acentúa que el diseño lógico debe separarse del diseño físico y precederlo.

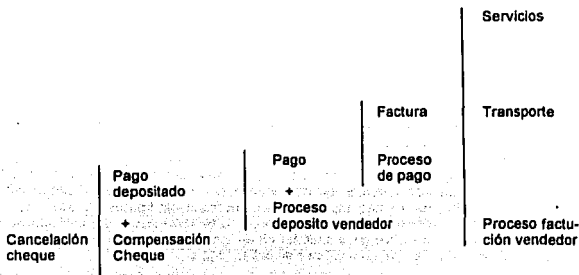


Figura 3.3.31 El diagrama de Warnier-Orr se utiliza para representar jerárquicamente los planes, los procedimientos y las estructuras de datos.

La principal diferencia entre las dos es que en la formulación de Jackson se mezclan todas las estructuras de datos de entrada y salida para formar una única estructura de programa, mientras que en la Warnier-Orr, la estructura del programa y las estructuras de los datos de entrada se derivan de las estructuras de datos de salida. La idea de Warnier-Orr es que la salida del sistema/programa determinada total y absolutamente la estructura de los datos y ésta, a su vez, determina la estructura del programa.

Según Ken Orr, "trabajando hacia atrás desde la salida es posible determinar exactamente las bases de datos y las entradas mínimas de sistema".

3.10.3 Los pasos básicos del DSSD

Como se muestra en el diagrama de Warnier-Orr de la figura 3.3.32, la metodología DSSD consta de nueve fases:

Fase de planificación del proyecto

Durante esta fase se desarrolla el plan del proyecto enlazando todos los objetivos del sistema. Se definen los resultados esperados del negocio que el sistema puede proporcionar y las limitaciones comerciales del sistema.

Fase de definición de requerimientos

Durante esta fase se definen los requisitos del usuario y organizativos para el sistema, también se define el entorno del ordenador necesario para el sistema. La fase de definición de requerimientos se divide en dos sub-fases: los requerimientos lógicos y los físicos, y, a su vez, estas sub-fases se dividen en pasos. El propósito principal de esta fase es definir el fundamento del diseño de bases de datos y sistemas durante la fase de diseño del DSSD.

El diagrama de entidad (un tipo de diagrama de flujo de datos) se emplea para definir las entidades del sistema y las transacciones que suceden entre ellas. El diagrama de entidad se utiliza para determinar la amplitud del sistema. El diagrama de línea de montaje (un tipo de diagrama de Warnier-Orr) se utiliza para definir los procesos funcionales (es decir, la transformación del conjunto de entrada en uno de salida). Las salidas necesarias para soportar en un diagrama de Warnier-Orr y un diccionario de datos. La figura 3.3.33 es un ejemplo de una base de salida lógica.

Fase de diseño

La metodología DSSD divide la fase de diseño en dos partes: el diseño lógico y el diseño físico.

1. Diseño de la base de datos lógica.
2. Diseño del proceso lógico.

Utilizando la base de salida lógica de la fase previa como entrada, el desarrollo de la base de datos lógica comienza por el desarrollo de la estructura de datos lógica que contienen los datos requeridos para producir una salida que contienen los datos mínimos requeridos para producir una salida. Seguidamente, se desarrollan los ficheros básicos lógicos y todos sus datos representando una arquitectura de datos normalizada. Puede demostrarse que cada sistema salida se produce a partir de

los ficheros básicos lógicos y que todos los datos son necesarios para producir la salida del sistema. Luego se definen la actualización de los datos lógicos, la edición y las transacciones de entrada.



figura 3.3.32 representación de la metodologías de desarrollo de sistemas por medio de un diagrama de Warnier-Orr.

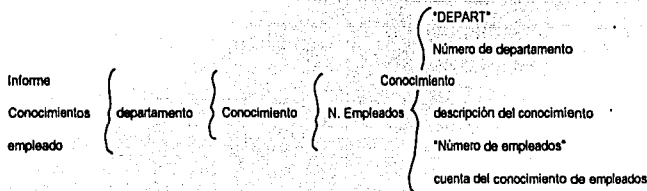


Figura 3.3.33 Se utiliza el diagrama de Warnier-Orr para representar la base de salida lógica de los datos del sistema de conocimientos de los empleados

En el paso del diseño de la base de datos lógica se desarrollan la base de actualización lógica y la base de entrada lógica se representan en el diagrama de Warnier-Orr. La figura 3.3.34 es un ejemplo de estructura de datos lógica.

Durante el paso del diseño lógico del proceso se desarrolla el proceso lógico, que también se representa con un diagrama de warnier-Orr. El objetivo del proceso lógico, es definir las transformaciones necesarias para producir las salidas del sistema desde las entradas.

Durante la fase de diseño físico, las restricciones del entorno físico (como las impuestas por el lenguaje de programación) se añaden a la base del proceso lógico,

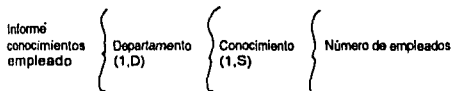


Figura 3.3.34 Se utiliza el diagrama de Warnier-Orr para representar la estructura lógica de los datos del sistema de conocimientos de los empleados.

Capítulo IV

DESARROLLO DEL SISTEMA MEDIANTE LENGUAJE SNAP

Introducción

El SNAP es una herramienta de alta productividad para el desarrollo de aplicaciones en un ambiente integrado de base de datos. Se enmarca dentro de las herramientas tipo CASE (Computer Aided Software Engineering), y comprende el desarrollo en un ciclo de vida del software desde el diseño hasta la implantación del sistema. La metodología del desarrollo en el ambiente SNAP involucra la definición de la base de datos, generación de programas de aplicación, un lenguaje de cuarta generación y la documentación en general del sistema. Esta metodología es sugerida por la interface de la herramienta al usuario.

Se pretende que al finalizar este capítulo, el lector tenga una idea clara de los pasos a seguir y las facilidades que provee el SNAP para el desarrollo de aplicaciones y en especial al desarrollo de el sistema COP (Control de Proyectos).

4.1 Pasos que se Recomiendan para el Desarrollo del Sistemas en SNAP

El SNAP se compone de cuatro grandes áreas: Modelo de Datos, Método de Desarrollo Acelerado (MDA), Utilitarios y Seguridad.

4.1.1 Modelo de Datos

En el Modelo de Datos, es una opción del menú principal de SNAP, donde se introduce el diseño conceptual o representación de la estructura de información de la aplicación, siguiendo paso a paso, los lineamientos de la metodología Entidad-Relación. Los pasos de modelamiento son muy sencillos: en primera instancia se identifican las entidades a participar en el sistema o modelo, luego se definen las interrelaciones que existen entre ellas para que finalmente, describir los atributos propios y específicos de cada entidad, La cual ya se definió en el capítulo III.

Una vez introducido el Modelo de Datos, SNAP genera automáticamente la base de datos compuesta por fuentes DDS y objetos nativos del AS-400. Al mismo tiempo, sin la necesidad de digitar ninguna línea de código, se queda habilitado para especificar los programas y dar mantenimiento integral al modelo, incluyendo aquellos de soporte de consulta, integridad referencial, navegación por lista y reportes necesarios para conformar un sistema.

Estos programas automáticos se generan utilizando todos los recursos y objetos disponibles que se administran y controlan desde el repositorio central, quedando a su vez registrados para su uso posterior por otros objetos o programas.

Todos los programas generados en el Modelo de Datos quedan totalmente funcionales; sin embargo, al analista tiene la posibilidad de ajuste por medio del Método del Desarrollo Acelerado (MDA), segundo componente del SNAP.

4.1.2 Método de Desarrollo Acelerado

El MDA es una plataforma de trabajo para ajustar, en forma individual, los programas generados automáticamente en el Modelo de Datos. El MDA permite crear y dar mantenimiento a programas nuevos.

Al igual que con el Modelo de Datos, al trabajar con el MDA el analista se sirve estratégicamente de los elementos del repositorio central y lo actualiza con el resultado de su gestión.

El MDA proporciona las herramientas necesarias para ajustar programas individuales, con un alto grado de productividad y rendimiento, sin necesidad de recurrir a lenguajes tradicionales. Esta herramienta incluyen, entre otras, un módulo para administrar las estructuras y bases de datos del programa, un diseñador de pantallas, un módulo para especificación de procesos, una lógica estructurada de módulos para los programas y un formateador de reportes.

4.1.3 Utilitarios

SNAP provee una serie de utilitarios para ayudar al analista a administrar y controlar el proceso de desarrollo de aplicaciones.

Entre las funciones que se ofrecen están las herramientas para definir normas de la organización en SNAP, entre estos están: utilitario de regeneración automática de sistemas como consecuencia de cambios en el modelo de datos, utilitarios de administración y control del repositorio central, documentación integral, y una gama de utilitarios misceláneos de apoyo a la gestión de los analistas.

4.1.4 Seguridad

SNAP incorpora un ambiente bastante sofisticado para controlar y ayudar a la administración del proceso del desarrollo de sistemas. Se proveen los elementos necesarios para proteger, hasta cinco niveles, las distintas definiciones y recursos del repositorio central.

4.2 Desarrollo del Sistema

Como se citó anteriormente, por medio de un ambiente de listas de opciones estructuradas, visto en el capítulo III, el modelo de datos se puede definir en forma lógica siguiendo una metodología formal, que permite aprovechar al máximo las características del desarrollo acelerado del SNAP.

Antes de empezar a capturar el modelo de datos es necesario tener una visión clara del sistema que se está desarrollando, es decir, debe existir una etapa de análisis previa para tener un total conocimiento de los requerimientos de información y de las reglas lógicas que gobiernan los estados válidos entre los datos y los procesos. Lo cual se puede ver en el capítulo anterior.

Una vez listo para la definición del modelo de datos en SNAP, se procede a seguir los siguientes pasos bajo la Metodología de Desarrollo en SNAP que son los mismos pasos que se definieron en el análisis del sistema del capítulo II, solamente que a diferencia estos pasos se definen dentro del SNAP.

- 1) Definir las entidades
- 2) Definir las relaciones entre entidades
- 3) Definir los atributos entre entidades
- 4) Definir las vías de acceso sobre las relaciones
- 5) Customizar el modelo de datos
- 6) Generar la base de datos

4.2.1 Definición de Entidades

El primer paso es definir las entidades que conforman el sistema dentro del SNAP. Aquí se pretende establecer en forma general los componentes del modelo de datos, en otras palabras, establecer simplemente cuáles van a ser los objetos sobre los que se desea mantener información, y no cómo se va a mantener esa información. Razón por la cual no se necesita definir la estructura de cada entidad, más bien se debe obtener una enumeración o lista de las entidades del sistema.

Entidades Fundamentales

Las entidades más fáciles de reconocer dentro de un sistema son las entidades fundamentales, puesto que no se derivan de ninguna otra.

Entidades Atributivas

Dentro del análisis del sistema ya se ha establecido cuáles datos son parte de una entidad y cuáles son opcionales. de la misma manera, se conoce cuáles datos son atómicos y cuáles respectivos. En estos casos, se debe definir nuevas entidades (atributivas) para almacenar los datos opcionales y los respectivos.

También existe la posibilidad de que una entidad que se haya definido en el sistema sea en realidad un componente de otra entidad (la entidad es más bien una atributiva), de modo que se debe definir la entidad principal a la cual depende.

Entidades Asociativas

Si existe alguna relación muchos a muchos conocida se debe crear una nueva entidad para que se pueda realizar esta relación.

Entidades Lógicas

Se debe crear además una entidad lógica por cada auto-referencia o se va a establecer más de una relación entre entidades.

Definición de Relaciones entre Entidades

Una vez definidas las entidades se procede a definir las relaciones entre ellas. Las relaciones sirven para establecer las reglas que permiten mantener la integridad referencial de la base de datos.

Dependencias

Se deben definir relaciones de dependencia en las entidades atributivas y entidades asociativas. En las entidades atributivas se debe definir sólo una dependencia hacia la entidad principal. Y en las entidades asociativas se debe definir más de una dependencia, una por cada entidad a relacionar. Es importante recordar que los campos de la dependencia van a formar parte de la llave primaria en la entidad donde se define.

Referencias

Las relaciones de referencia se pueden definir en cualquier tipo de entidad (el número de dependencias es el que establece el tipo de entidad), y se define cuando la entidad es independiente con respecto a la entidad con la que se relaciona. es importante recordar que los campos de la referencia pasan a formar parte de la entidad donde se define como campos no-llave.

Definición de Atributos de Entidades

Una vez definida la estructura lógica que habrá de tener el Modelo de Datos, se procede a definir la composición de las diversas entidades definidas en el primer paso.

Atributos de Identificación

Los atributos de identificación o "AID" son los que van a formar parte de la llave primaria de la entidad. Si la entidad tiene definida una dependencia, entonces, el "AID" va a diferenciar cada una de las ocurrencias que dependen de la misma ocurrencia en la entidad principal (los campos de la dependencia simple están antes de los atributos de identificación en la llave primaria).

Atributos Generales

Los atributos generales "ATR" son las características de cada entidad. Se debe incluir sólo aquellos atributos que pertenecen propiamente a la entidad.

Definición de las Vías de Acceso sobre las Relaciones

Para que el SNAP puede mantener la integridad referencial de la base de datos (controlar que exista o no relaciones sobre una ocurrencia en una entidad) necesita una vía de acceso por cada relación definida en las entidades.

La integridad referencial se va a validar por el programa automático (ELI). Este programa va a controlar la existencia o no de las vías de acceso requeridas.

Vías de Acceso sobre Referencias

Se debe definir una vía de acceso por cada referencia de la entidad.

Vías de sobre dependencias (dependencias no iniciales)

Cuando una entidad tiene una dependencia, la vía de acceso principal sirve para verificar la integridad de datos, por lo que no se necesita definir otra vía alterna.

Pero cuando existe más de una dependencia, la vía de acceso principal sólo sirve para verificar la integridad de la primera dependencia, pero las restantes (las dependencias no iniciales) necesitan una vía alterna para su verificación.

Por tanto, se debe definir una vía de acceso por cada dependencia no inicial de una entidad.

4.2.2 Customización del Modelo de Datos

Una vez completada la estructura conceptual de la base de datos, se puede proceder a customizar el modelo de datos, es decir, a realizar los ajustes necesarios para completar su definición.

Estos ajustes tienen que ver con especificaciones de características físicas y lógicas a nivel de entidades y de atributos

4.2.2.1 Customización de Entidades

Una entidad se puede ajustar tanto a nivel externo como a nivel físico, a nivel externo se pueden definir vías alternas de acceso o entidades lógicas, y a nivel físico se pueden definir ciertas características para la generación del archivo físico.

4.2.2.2 Definición de Vías Alternas de Acceso

Una vía de acceso permite leer secuencialmente la entidad ordenada por llave, o acceder directamente un registro si se conoce la llave o bien buscar el inicio o final de un grupo de registros que contengan parte de la llave (la parte inicial de la llave).

En muchas ocasiones las características de las aplicaciones de un sistema requieren que una entidad pueda ser accesada u ordenada por muchos atributos diferentes a la llave primaria. La llave primaria de una entidad representa la vía de acceso. Sin embargo, cuando se desarrolla se pueden definir vías alternas de acceso para poder manejar la información de la entidad de manera diferente, por ejemplo, para ordenar los registros por algún campo en particular, para definir una llave candidata (duplicados no permitidos en otros campos que no sean la llave primaria), para buscar información por algún campo que no sea la llave principal, etc.

Cuando se definen las vías de acceso se toman en cuenta:

- 1) Integridad referencial (explicado anteriormente)
- 2) Diferentes ordenamientos en los programas CON LIS.
- 3) Diferentes estructuras de reportes en los programas LIS.
- 4) Filtros de información de la entidad
- 5) Otros procesos u ordenamientos sobre la información.

4.2.2.3 Definición de Entidades Lógicas

Las entidades lógicas son un "espejo" de la entidad física a la que pertenece, en el sentido de que aunque SNAP las considera entidades independientes, físicamente sus datos son parte de los datos de la entidad física. Por esta razón, la llave principal de la entidad lógica es igual a la llave principal de la entidad física, es decir, puede tener muchos atributos que la física o atributos derivados de los físicos (atributos lógicos). Sin embargo, en la vía de acceso principal de la entidad lógica se puede definir filtros, de manera que no puede manejar todos los datos de la entidad física.

Cuando se definen entidades lógicas se debe tomar en cuenta:

- 1) Múltiples relaciones a una misma entidad o auto-referencias (explicado en la definición de entidades)
- 2) Diferentes formas de ver o manejar una entidad (Vista externa).

4.2.2.4 Modificación de Aspectos Físicos de las Entidades

Los aspectos físicos son características propias de la generación de los archivos de la entidad, para ello se debe conocer la arquitectura del AS-400 y su lenguaje de definición de datos DDS (Data Description Source).

Customización de Datos

Al igual que con la customización de una entidad, los atributos se pueden ajustar tanto a nivel externo como a nivel físico. a nivel externo se pueden definir una ampliación del dominio determinado por el tipo de dato, y a nivel físico se pueden definir ciertas características para la generación del campo físico.

Definición de Lista de Valores para el Atributo

Una lista de valores es un conjunto de códigos con sus correspondientes descripciones que tienen como finalidad establecer el conjunto de valores permitidos por el campo, y producirá automáticamente una ventana de ayuda para el usuario final, para escoger por pantalla uno de esos valores.

Modificar Aspectos Físicos del Atributo.

Con esta opción se puede modificar las características definidas en el tipo de datos (dominio) , a saber, longitud, número de decimales, código de edición, etc.

Modificar Aspectos DDS

Los aspectos DDS son características propias de la generación de los campos físicos de la entidad, para ello se debe conocer la arquitectura de AS-400 y su lenguaje de definición de datos DDS (Data Description Source).

4.2.3 Generación de la Base de Datos

Luego de que ya se haya definido todas las características de la Base de Datos, se procede a generar la Base de Datos física, lo cual se realiza dentro del SNAP por medio de una simple opción de generación de la entidad.

A este nivel, el desarrollador no ha tenido que digitar una sola línea de código para definir y generar la base de datos. Todas las definiciones fueron realizadas con una interface de muy alto nivel (lista de opciones), el SNAP se encarga ahora de traducir todas esas definiciones en el lenguaje nativo de AS-400, o sea, en las DDS de la entidad, o puede imprimir un reporte completo de toda la información definida de la entidad.

4.3 Documentación del Modelo de Datos

Además, en cada de estos pasos deben considerarse las diferentes opciones de documentación, las cuales como buena técnica de desarrollo, deben ser aprovechadas cada vez que se defina algún componente (entidad, vía de acceso, etc.). La documentación puede acompañarse desde la captura del campo ya sea campo llave o atributo y desde luego a nivel entidad.

4.4 Desarrollo Físico de el Sistema

Quando se inicia una sesión en SNAP se presenta el menú principal el cual contiene las cuatro opciones del ambiente, como se muestra en la siguiente figura.

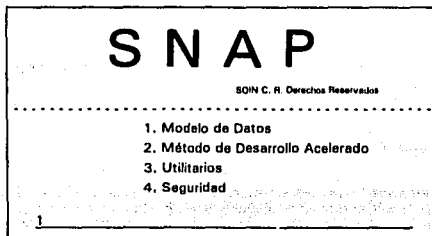


Figura 4.1.1 Pantalla principal de SNAP.

Al escoger la opción 1 de menú principal de SNAP, se ingresa al Modelo de Datos.

4.4.1 Sistema a Trabajar

En caso de que aún no se haya seleccionado el sistema a trabajar, se presenta otra pantalla la cual pide el sistema a trabajar. En nuestro caso el sistema se llama COP (Control de Proyectos).

4.4.2 Captura y Mantenimiento de Entidades

Una vez seleccionado el sistema de trabajo se presentará la pantalla principal del Mantenimiento de Entidades, la cual presenta la lista de las entidades ya capturadas y creadas y que permite manejarlas a través de opciones y teclas de mandatos, con el fin de realizar la definición de toda la base de datos.

SNAP Release 3.1 Derechos Reservados SOIN C.R.		10/05/94	10:41:18
Sistema Control de Proyectos			
Mantenimiento de Entidades			

2 = Cambiar 4 = Eliminar 5 = Visualizar 6 = Listar 7 = Vías de Acceso 8 = Entid.logicas			
9 = Relacio. 12 = Trabajar 13 = Documentar 14 = General 18 = Programas			
Señ.	Entidad	Descripción	
--	Actividades	Claves de Actividades	
--	Clientes	Catalogo de Clientes	
--	Comentarios	Comentarios de Productos	
--	Conocimientos	Tabla de Conocimientos	
--	ORA Detalle Reporte	Detalle de Reporte de Actividades	
--	Histórico de Planes	Histórico de Planes	
--	Personal	Catalogo de Personal	
--	Personal/Clientes	Personal Asignado a Clientes	
--	Personal/Conocimiento	Conocimientos del Personal	
--	Planes	Planes de Trabajo	

F6 = Agregar, F12 = Anterior, F17 = Subistema, F18 = Estado Generación			

En la pantalla se presenta la siguiente información por columnas:

Entidad:

Esta columna muestra el nombre de las diferentes entidades definidas en el subsistema que se escogió anteriormente.

Descripción:

Esta columna muestra la descripción de cada entidad.

Las opciones operan sobre la información de cada línea, es este caso entidades previamente creadas; colocando el número de la opción en la columna de selección de cada entidad, al comienzo de cada línea. En la parte superior, después de los encabezados de la pantalla, se indican las opciones válidas en esta pantalla y su función, por ejemplo, 2=Cambiar (cambiar la información general y aspectos físicos de una entidad).

Los mandatos realizan operaciones sobre el conjunto de información desplegada, en este caso sobre todas las entidades, o provocan una acción específica, por ejemplo, salir de la pantalla. Un mandato se acciona presionando la tecla de mandato. En la parte inferior de la pantalla se indican los diferentes comandos válidos en ella y su función, por ejemplo, F12=Anterior (Volver a la pantalla anterior, es decir, en este caso al menú principal).

En esta pantalla se tiene acceso a todas las opciones y mandatos que permiten crear, modificar y customizar el modelo de datos, los cuales se describen a continuación:

Mandatos

F6=Agregar

Permite agregar nuevas entidades al modelo de datos.

F12=Anterior

Permite retornar a la pantalla del menú principal. En general, esta tecla siempre estará presente en cualquier pantalla y su función es terminar el proceso actual y regresar a la pantalla anterior, es decir, a la pantalla donde se accede la actual.

F17=Subsistema

Permite escoger un subsistema para desplegar únicamente aquellas entidades asociadas al subsistema.

F18=Estado de generación

Permite visualizar el estado (En cola de Espera, Activo, Listo o Error) en el que se encuentran todos los programas y entidades que se han enviado a generar.

Opciones

2=Cambiar

Permite modificar la información general de la entidad y, además, permite modificar los aspectos físicos de una entidad.

4=Eliminar

Permite eliminar la definición de una entidad.

5=Visualizar fuente

Permite visualizar el archivo fuente DDS generado para cada entidad.

6=Listar

Permite generar un reporte con toda la información de la entidad.

7=Vías Acceso

Permite definir diferentes vías de acceso para las entidades.

8=Entidades Lógicas

permite definir diferentes entidades lógicas para las entidades.

9=Relaciones

Permite establecer las relaciones de dependencia o referencia entre diferentes entidades.

12=Trabajar

Permite trabajar con las entidades para definir los diferentes atributos y relaciones que la componen.

13=Documentar

Permite documentar las entidades del sistema.

14=Generar

Permite generar el correspondiente representación física de las entidades. Genera el fuente DDS y el archivo físico de la entidad.

18=Programas

Permite la creación de programas automáticos.

Estas opciones y comandos son explicados detalladamente en las siguientes secciones, el orden de aparición está determinado por la metodología de desarrollo SNAP:

4.4.2.1 Definición de Entidades

El proceso de definición de entidades implica agregar, cambiar y eliminar la información general de cada entidad y su documentación. A continuación se presenta en detalle la forma de realizar cada uno de estos procesos.

Inclusión de entidades

En este caso las entidades ya están creadas y documentadas, pero se dará un ejemplo de agregación con el objetivo de tener una visión completa del funcionamiento del sistema.

Para crear las entidades se utiliza el mandato F6=Agregar, al utilizar este mandato podrá incluirse una entidad dentro del modelo de datos que se definió. Luego de presionar la tecla F6 se presenta la pantalla mostrada en la siguiente figura.

SNAP Release 3.1 Derechos Reservados SOIN C.R.	10/05/94	0:41:18
Sistema Control de Proyectos		
Mantenimiento de Entidades		

ENTIDADES		
Entidad	:	Cientes

F4 = Lista, F12 = Anterior, F15 = Importar		

FALLA DE ORIGEN

En esta pantalla se solicita el nombre de la nueva entidad. El nombre está compuesto por un máximo de veinte caracteres. Con respecto al nombre existen dos restricciones:

1) El nombre debe ser único, es decir, no puede existir una entidad dentro del mismo sistema con el mismo nombre.

2) Las mayúsculas y minúsculas se toman como caracteres diferentes.

Además se dan las siguientes recomendaciones con respecto a la asignación de nombres:

1) Los nombres utilizados deben ser significativos lo que facilitará su utilización a través de todo el SNAP.

2) El primer carácter del nombre debe ser una letra mayúscula.

3) El nombre debe ser un sustantivo, debido a que este nombre será utilizado en los programas automáticos para enviar mensajes al usuario. Por ejemplo, el siguiente mensaje es generado automáticamente para un programa de mantenimiento simple a la entidad "Cliente":

"Cliente no está disponible INTRO para continuar"

Una vez que se ha definido el nombre de la entidad simplemente se debe pulsar la tecla INTRO, luego se desplegará la pantalla de cambiar la información de la entidad, explicado en el siguiente apartado.

Después de digitar la información de la entidad, existe la posibilidad de configurar el SNAP para que automáticamente se deba digitar su documentación. Esta opción es "Documentación obligatoria" definida en los parámetros de instalación.

Cambiar definición de entidades

La pantalla de cambiar la información de la entidad aparece después de agregar una entidad o escogiendo la opción 2=Cambiar desde la pantalla principal de mantenimiento de entidades:

SNAP Release 3.1 Derechos Reservados SOIN C.R.	10/05/94	0.41:18
Sistema Control de Proyectos		
Mantenimiento de Entidades		

ENTIDADES		
Entidad :	Clientes	
Descripción :	Catálogo de Clientes	
Subsistema :	COP	
Propietario :	AEIS	
Nivel de Seguridad :	5	
Nombre DDS :	CLIARC	
Nombre DDS regs :	CLIREG	
F4 = Lista, F10 = Aspectos Físicos, F12 = Anterior		

FALLA DE ORIGEN

A continuación se explica cada uno de sus campos

Descripción

En este campo de 40 caracteres se podrá digitar la descripción de la entidad en cuestión. Este campo de descripción es muy útil para ampliar la información de la entidad. El SNAP no permite dejar este espacio en blanco, por lo que es muy recomendable que se aproveche para incluir información descriptiva. Esta descripción es además presentada en la pantalla principal de mantenimiento del modelo de datos.

Subsistema

En esta campo se ingresa el código del subsistema donde se va a incluir la entidad (En este caso el subsistema es el COP). Automáticamente se utiliza el subsistema escogido para trabajar.

Propietario:

Este campo no puede modificarse, define el usuario que incluye la entidad.

Nivel de Seguridad

En este campo se define el nivel de seguridad de la entidad. La opción de seguridad de SNAP permite toda una gama de protección sobre los objetos (entidades y programas). Lo más importante es saber que el nivel de seguridad sobre las entidades se define en esta parte. El nivel de seguridad puede estar entre 0 y 5, siendo más restrictivo cada nivel conforme más alto sea. Es permitido el uso del mandato F4 para seleccionar el nivel de seguridad para la entidad.

Nombre DDS del archivo

Este campo establece el nombre que va a tener el archivo físico de la entidad. este nombre tiene 6 caracteres y va a ser único dentro del sistema. automáticamente el SNAP genera un nombre tentativo bajo el siguiente criterio: los primeros tres caracteres corresponden al sufijo "ARC" (Abreviación de archivo físico). En el caso de que existe el nombre generado se le agrega un dígito consecutivo al final. Por ejemplo, la entidad "Comentarios" generará un nombre DDS "COMARC" y luego "Comentarios Plan" generará "COMAR1". El nombre generado puede ser modificado sin ningún problema, con la condición de que el nuevo nombre no este ya incluido en el sistema.

Nombre DDS del registro:

Este campo establece el nombre que va a tener el registro físico de la entidad. Este nombre tiene 6 caracteres y va a ser único dentro del subsistema. Automáticamente el SNAP genera un nombre tentativo siguiendo el anterior, lo único que cambia son la tres primeras letras por "REG". El nombre generado puede ser modificado sin ningún problema, con la condición de que el nuevo nombre no este incluido en el sistema.

También se puede modificar los aspectos físicos de la entidad con el comando F10=Aspectos físicos. Este comando se explicará en la sección de Customización del modelo de datos.

Eliminar la Definición de una Entidad

La eliminación de una entidad borra los atributos, relaciones, aspectos físicos, vías de acceso, lista de valores, y cualquier otro elemento que pertenezca a la entidad. Por esta razón este proceso tarda un tiempo moderado en ejecutarse, existiendo la posibilidad de enviarlo a ejecutarse por lotes.

Una entidad no puede eliminarse si existe alguna relación sobre ella, por lo tanto, antes de ejecutarse una eliminación se debe excluir cualquier relación hacia la entidad.

En caso de que la entidad sea utilizada por algún programa, el proceso lo indicará para su confirmación, en caso de que se confirme la exclusión, se genera un listado con, con los programas que fueron modificados con la exclusión de la entidad.

Documentación de Entidades

Esta opción es muy útil para que se tenga documentación al orden que se requiera, esta puede ser utilizada cada vez que una entidad es creada. Si la opción de documentación es obligatoria está activa, cada vez que se incluya una nueva entidad se va a ejecutar este proceso.

Para documentar las entidades se utiliza la opción 13=Documentar de la pantalla de mantenimiento del Modelo de Datos. El desarrollador debe utilizar esta opción para incluir toda la información técnica necesaria con respecto a la entidad correspondiente.

Editor de Documentación

La pantalla de documentación muestra un editor básico de textos, cuyo fin es poder incluir o modificar la documentación de los diferentes elementos del Modelo de Datos. El SNAP permite, através de este editor, documentar entidades, relaciones y atributos. La documentación de entidades y relaciones es de naturaleza técnica, es decir, que la información digitada debe estar dirigida hacia desarrolladores para el mantenimiento del sistema. En cambio la documentación de los atributos es tanto técnica como dirigida al usuario, esto significa que parte de la documentación del atributo se convierte en ayuda en línea para el usuario al utilizar la aplicación final.

En caso de que ya exista documentación podrá ser modificada, ya sea para ser corregida o para ser ampliada.

Definición de Relaciones entre Entidades

La definición de relación entre entidades permite establecer las diferentes asociaciones entre las entidades del modelo de datos. Las relaciones pueden definirse por medio de DEPENDENCIAS o REFERENCIAS, según sea la naturaleza de la relación.

Para establecer una relación se utiliza la opción 9=Relaciones de la pantalla principal del mantenimiento de entidades. También es posible definir las relaciones a través del comando F7=Relaciones en la pantalla de trabajar con la entidad (opción 12=Trabajar en la pantalla de mantenimiento de entidades). Sin embargo, la opción 9 se utiliza para una mayor armonía con la metodología de desarrollo SNAP, donde se debe definir primero las relaciones y luego los atributos.

SNAP Release 3.1 Derechos Reservados SOIN C.R.		10/05/94	0:41:18
Sistema Control de Proyectos Mantenimiento de Entidades			
Entidad : Planes		Planes de Trabajo	
RELACIONES			
2 = Cambiar, 4 = Eliminar, 13 = Documentación, 20 = Relaciones Lógicas			
COD	Nombre	Descripción	
---	DEP	Clientes	
F8 = Agregar, F12 = Anterior			

En esta pantalla se presentan las siguiente información por columnas:

Código de Relación:

El código indica el tipo de relación definido, y esto sólo puede contener "DEP" para dependencias y "REF" para referencias.

Nombre:

Es el nombre de la entidad a la cual se refiere. En el caso de una dependencia, el nombre es el nombre de la entidad "uno" dentro de la relación "muchos-a-uno".

Descripción:

Es la descripción de la relación definida.

En el caso de que previamente se hayan definido dependencias aparecerá la información de las mismas en los campos situados anteriormente.

Mandatos

F8=agregar

Permite agregar nuevas relaciones en la entidad. El proceso de inclusión de relaciones se explica en el siguiente apartado.

Opciones

2=Cambiar

Permite modificar la información de una relación, y los aspectos DDS de los diferentes campos de la relación.

4=Eliminar

Permite eliminar la definición de entidades.

13=Documentar

Permite documentar una relación. El editor de documentación se explica en la documentación de entidades.

20=relaciones Lógicas

Permite escoger entre la entidad física y sus entidades lógicas, para relacionarse.

Agregar Relaciones

Para agregar nuevas referencias o dependencias a una entidad se utiliza el mandato F6=Agregar, luego de utilizar este mandato se despliega una pantalla de inclusión de entidades, la cual es una lista de campos para dirigir la información de cada relación explicada en la pantalla anterior.

La opción 5=Consultar actúa en una lista de campos de entrada (subarchivo de entrada) como el comando F4=Lista en un campo normal (que no pertenece a una lista o subarchivo), siempre que el campo a consultar es te en blanco. Si consulta el nombre de la relación aparecerá una lista de todas las entidades del modelo.

Para grabar las relaciones agregadas debe presionarse la tecla INTRO, y si se desea ignorar estas inclusiones presione F12. Si se confirma la inclusión de las relaciones y se tiene activa la opción de documentación obligatoria, se va a representar la pantalla de documentación para cada relación agregada.

Definición de Atributos

Para trabajar con la estructura de atributos y relaciones de una entidad se utiliza la opción 12=Trabajar desde la pantalla principal del mantenimiento del modelo de datos. Una vez que se ha utilizado esta opción se presenta la siguiente pantalla.

SNAP Release 3.1 Derechos Reservados SOIN C.R.		10/05/94	10:41:18
Sistema Control de Proyectos			
Mantenimiento de Entidades			
Entidad : Planes		Planes de Trabajo	
RELACIONES			
2 = Cambiar, 4 = Eliminar, 13 = Documentación, 20 = Relaciones Lógicas			
COD	Nombre	TIP	Descripción
- DEP 10	Clientes		Clientes
- AID 20	Plan Clave	A03	Clave del Plan
- AID 30	Modulo Clave	A03	Clave del Modulo
- AID 40	Submodulo Clave	A03	Clave del Submodulo
- ATR 50	Actividad Des.	A40	Descripción de actividad
- ATR 60	Tiempo Planeado	O70	Cantidad de tiempo planeado
- ATR 70	Gente Asignada	O20	Cantidad de gente asignada a un P.
- ATR 80	Dia Inicio Plan	S02	Dia de Inicio Plan
+			
F1 = Secuencia, F2 = Código, F6 = Agregar, F7 = Relaciones, F12 = anterior			

Con esta opción se puede dar mantenimiento tanto a atributos como a relaciones, sin embargo, como ya se explicó, la opción 9=relaciones se definió aparte para poder seguir más estrechamente la metodología de trabajo.

En esta pantalla aparece la información de los diferentes elementos que componen un entidad. Si ya existen atributos o relaciones definidas para la entidad se desplegará la información correspondiente.

En la parte superior aparece el nombre y la descripción de la entidad, e inmediatamente después, en forma columna aparecen los siguientes campos:

Código

Especifica el tipo de atributo o relación de que se trata, es decir, si es una dependencia, una referencia, de un atributo general, un atributo de identificación o un atributo lógico (los atributos lógicos sólo pertenecen a entidades lógicas).

Secuencia:

Especifica el orden cómo los campos van a generarse físicamente, y especifica a la vez el orden de la llave primaria, puesto que está se conforma por las dependencias (ordenadas por secuencia) y luego por los atributos de identificación (ordenados por secuencia).

Nombre:

Especifica el nombre del atributo o el nombre de la entidad en el caso de relaciones.

Tipo de Dato:

Especifica el tipo de dato de los atributos (no se utiliza para definición de relaciones); éste definirá las características físicas del atributo, así como los códigos de edición en pantallas y opciones de teclado que se utilizarán. El tipo de dato deberá estar previamente definido en el repositorio.

Descripción:

Especifica la descripción del atributo o relación que se define cuando se agregan.

Opciones

2=Cambiar

Permite cambiar el tipo de dato y la descripción del atributo. También permite modificar los aspectos físicos, incluir lista de valores y aspectos DDS. En casos de relaciones refiérase a cambiar relaciones en la sección anterior.

4=Eliminar

Permite eliminar la definición de un atributo o una relación.

13=Documentar

Permite documentar un atributo o una relación. En caso de atributos se recuerda que el recuadro responde a la ayuda en línea automática para el campo.

20=Relaciones Lógicas

En las relaciones permite escoger entre la entidad física y sus entidades lógicas, para relacionarse. Refiérase al mantenimiento de relaciones.

Mandatos

F1=Secuencia

Esta tecla de función, al igual que F2, tienen como propósito cambiar el orden de la forma cómo se visualiza los atributos y relaciones de la entidad. Un ordenamiento por secuencia muestra la entidad tal y como va a generarse físicamente.

F2=Código

Esta tecla de función, al igual que F1, tienen como propósito cambiar el orden de la forma cómo se visualiza los atributos y relaciones de la entidad. Un ordenamiento por código muestra la entidad en el orden natural que debe tener una entidad en un ambiente de base de datos, y es muy importante para conocer cómo se va a generar físicamente la llave primaria del archivo. Los códigos se ordenan de la siguiente forma: Dependencias, Atributos de Identificación, Atributos Generales, Atributos Lógicos y Referencias.

F8=Agregar Atributos

Permite agregar atributos de identificación y generales de la entidad.

F7=Agregar Relaciones

Permite agregar dependencias y referencias a la entidad.

Customatizar el Modelo de Datos

Definición de Vías Alternas de Acceso

Para definir vías alternas a las diferentes entidades se utiliza la opción 7=Vías de Acceso de la pantalla principal del Modelo de Datos. Antes de definir una vía de acceso alterna recuerde los factores de importancia a tomar en cuenta y que anteriormente se mencionaron:

- 1) Integridad referencial**
- 2) Diferentes ordenamientos en los programas CON LIS**
- 3) Diferentes estructuras de reportes en los programas LIS**
- 4) Filtros de la Información de la entidad**
- 5) Otros procesos u ordenamientos sobre la información**

Una vez escogida esta opción se presenta la pantalla de mantenimiento de vías alternas de acceso, la cual es similar a la de mantenimiento a entidades, y emplea las mismas teclas de función y de opciones (estas trabajan en forma por igual en todas las pantallas y por su puesto no todas las pantallas tendrán las mismas teclas y funciones). Una diferencia en esta pantalla es que tiene la opción 14=Generar, la cual genera la vía de acceso.

Aspectos Físicos de la vía de Acceso

Los aspectos físicos son características propias de la generación de los archivos físicos de la entidad, para ello se deben conocer la arquitectura del AS/400 y su lenguaje de definición de datos DDS (Data Description Surce) por lo que escapa a la intención de desarrollo de el sistema describir las o mencionadas ya que son un poco extensas y complicadas para quienes no conocen esta arquitectura.

Definición de Lista de Valores para un Atributo

El SNAP permite definir listas de valores para los atributos compuestas por un conjunto de valores, cada uno de ellos con su respectiva descripción con el fin de restringir los valores permitidos para el campo o para desplegar la lista con su significado. Por medio de esta opción puede definirse la serie de valores permisibles para el atributo, o en el caso de que no se desee restringir los valores del atributo a los de la tabla de valores, se puede utilizar como una herramienta para proveer rápidamente los valores de uso común.

Estas listas permiten que al añadir al sistema características de alta productividad, por medio de éstas el desarrollador puede mantener la integridad en los valores del atributo, agilizar el trabajo del usuario de la aplicación al tener rápidamente a su disposición los valores para un campo, generar más completos utilizando las descripciones de cada valor especificado para el atributo, etc.

La Customización del modelo de datos emplea estos y otros conceptos como la modificación de aspectos físicos del atributo y modificación de aspectos DDS pero como ya se mencionó son ya parámetros propios de la arquitectura del AS/400, aunque no son complejos son bastantes.

Generación de la Base de Datos

Por medio de la opción 14=generar de la pantalla de mantenimiento del Modelo de Datos puede generarse la base de datos, una vez que se ha definido, documentado y customizado o único por hacer es digitar 14 en la casilla de selección de las entidades y pulsar INTRO. Una vez hecho esto el SNAP se encarga de generar todos los elementos necesarios para formar la base de datos sobre la que se basará el sistema.

La generación de entidades tiene el siguiente efecto:

- 1) Si la entidad ya ha sido generada anteriormente, borra todas las vías de acceso y las entidades lógicas pertenecientes a la entidad a generar, y al final renombra el archivo físico de la entidad para dejar un respaldo de la información. Este respaldo empieza con el prefijo "TP" junto con un número aleatorio que maneja internamente el SNAP.
- 2) Genera las especificaciones fuente DDS de la entidad y las vías de acceso, y si existe, los fuentes de las entidades lógicas y sus respectivas vías de acceso. Estas definiciones DDS se almacenan en el archivo de miembros QDDSRC de la biblioteca definida para datos del sistema.

3) Crea los archivos físicos y lógicos en la biblioteca del sistema definida para almacenar los datos (compila los fuentes DDS).

4) Si la generación terminó satisfactoriamente, copia los datos del archivo de respaldo al nuevo archivo físico. En caso de que exista algún error, el estado de generación mostrará el nombre del archivo de respaldo, por ejemplo, TP03438758.

Con el estado de generación se conoce el resultado de la generación de la entidad o entidades a generar.

Una vez generada la entidad, se puede visualizar el fuente DDS de su archivo físico y de sus archivos lógicos (visualizar DDS en vías de acceso y de entidades lógicas), y puede generar un reporte completo de la entidad.

4.4.3 Creación de Programas Automáticos

Una de las opciones que acelera enormemente el trabajo del desarrollador es la opción 18=Programas, de la pantalla de Mantenimiento de Modelo de Datos.

Esta opción es un generador de programas automáticos (generador de aplicaciones) que permiten obtener toda una gama de procesos sobre la base de datos, los cuales son inteligentemente interconectados por SNAP y, lo más importante, no requieren ningún esfuerzo de programación, disminuyendo el tiempo de desarrollo considerablemente.

Esta es una de las dos áreas principales de la metodología para el desarrollo de sistemas, la primera consiste en el modelamiento de la información, la segunda se relaciona directamente con la definición de la aplicación, obteniéndose como los programas necesarios para administrar la información de la base de datos.

SNAP proporciona dos herramientas poderosas para definir la aplicación. La primera ya se mencionó que es la generación de programas automáticos, los cuales únicamente requieren una breve especificación de muy alto nivel sobre su función. La otra herramienta, llamada Método de Desarrollo Acelerado o MDA, proporciona un ambiente de trabajo y lenguaje de programación diseñado especialmente para ajustar cualquier programa automático a una necesidad específica o bien, crear los programas de carácter transaccional del sistema.

En el siguiente cuadro se muestra en resumen la metodología para definir una aplicación en SNAP.

METODOLOGIA DE DESARROLLO SNAP

Area 1: Modelamiento de la información

1. Definición de Entidades
2. Definición de Relaciones entre Entidades
3. Definición de Atributos
4. Definición de Vías de Acceso sobre relaciones
5. Customización del Modelo de Datos
6. Generación de la BAsE de Datos

Area 2: Definición de la Aplicación

1. Definición de Programas Automáticos
2. Generación de Programas Automáticos

Area 3: Ajuste de la Aplicación

1. Ajuste de Programas Automáticos Vía MDA
2. Construcción de nuevos Programas Vía MDA

4.4.3.1 Programas automáticos

Los programas automáticos son programas generados partiendo de especificaciones de muy alto nivel las cuales determinan la función en el manejo de la información de la base de datos. Para la definición de estos se requiere de la combinación de tres elementos básicos:

- 1) El tipo de Programa
- 2) La Entidad Asociada
- 3) Especificaciones de muy Alto nivel

Tipos de Programas Automáticos

El tipo de programa automático es el elemento que aporta la lógica al proceso, es decir específicamente las "acciones" que se requiere sobre la entidad. En una aplicación se puede reconocer que una gran mayoría de programas encajan en un patrón lógico muy similar, sobre todo cuando se trata

de programas interactivo (Consulta, Mantenimiento, Generación de reportes, etc.) Cada tipo de programa automático resume las características de un patrón lógico específico, ajustándose a cada entidad y al conjunto de características definidas para el programa.

Por otra parte los programas automáticos hacen uso de parámetros generales definidos por el usuario para cada tipo, lo que permite una muy alta normalización de las aplicaciones.

Un aspecto importante de la normalización de las aplicaciones en los parámetros de programas automáticos son por ejemplo, las teclas de función, pantalla modelo, proceso inicial, y otros. Los programas automáticos ofrecen una característica de alta flexibilidad, permitiendo ser modificados tanto en su forma de presentación como en ciertos aspectos importantes de su algoritmo.

SNAP provee cinco tipos de programas automáticos que satisfacen la necesidad básica de cualquier sistema de información, además dentro de estos tipos se definen subtipos que aportan diferentes modalidades específicas de procesamiento y recuperación de información.

Estos programas cuentan con un 100% de correctitud permitiendo que una aplicación se encuentre rápidamente en un ambiente de producción. Además se basan sobre algoritmos de fácil entendimiento por lo que cualquier modificación en el Lenguaje SNAP generado es bastante simple. Al final de este capítulo se dará un ejemplo de la utilización de estos programas con un ejemplo real y que abarca la mayoría de estos programas pero ahora se explicaran cada uno de ellos.

Eliminación

Este primer tipo de programa automático es muy importante, ya sea su función es velar por la integridad de la información en la base de datos. El mantener dicha integridad no es tarea fácil en bases de datos complejas, la integridad debe controlarse en dos áreas básicas que son la inclusión y eliminación de registros, especialmente la eliminación de registros que involucra un control más extenso.

Sin embargo SNAP vela por la integridad de la base de datos cuando se elimina registros, ya que por medio de los programas automáticos de verificación de la eliminación antes de realizar el borrado de registros.

Los subtipos de los programas de eliminación son los siguientes:

Tipo	Subtipo	Descripción
ELI		Verificación de Eliminación
EL	AS	Verificación de Eliminación en Cascada

Consulta

Este segundo tipo de programa se refiere a programas de consulta a la información de las entidades. Para este tipo existen dos subtipos de programas automáticos.

Tipo	Subtipo	Descripción
CON	LIS	Consulta por Lista
CON	SIM	Consulta Simple

El primer subtipo sirve de apoyo a los demás programas automáticos y a los que se definen con un propósito específico en la aplicación deseada, este recibe parámetros que permiten desplegar y retornar la información elegida. El propósito de este programa es listar los valores o un grupo de registros de una entidad y tener la posibilidad de escoger el registro buscado.

El segundo subtipo son programas de consulta con múltiples opciones de diseño, que permite visualizar un registro en particular así como crear diversos tipos de recuperación de información del registro así como crear diversos tipos de recuperación de información del registro en caso de relacionarse con otras entidades, con un mínimo esfuerzo.

Mantenimiento

Este tipo de programa automático es el que se dedica al mantenimiento interactivo de la información de la base de datos que consiste en la inclusión, modificación y eliminación de registros de las entidades definidas. Existen varios tipos de programas de mantenimiento que se basa en los siguientes criterios.

- 1) Forma de Entrada (Masiva o por Registro).
- 2) Naturaleza de la información (basada en las relaciones existentes)
- 3) Forma de Navegación a través de la aplicación.

Los subtipos de programas automáticos de mantenimiento son:

Tipo	Subtipo	Descripción
MAN	SIM	Mantenimiento Simple
MAN	SAB	Mantenimiento Simple "ABC"
MAN	LIN	Mantenimiento con Líneas de Detalle
MAN	LAB	Mantenimiento con Líneas de Detalle
MAN	MAS	Mantenimiento Masivo

El primer subtipo de programa MAN SIM permite dar mantenimiento a un único registro a la vez, por medio de un sistema de navegación en múltiples pantallas. Con este subtipo de programa se puede trabajar con los campos de las entidades relacionadas, por medio de una referencia (REF) o una dependencia (DEP), con la entidad para la cual se define el programa, el tipo de acceso para el programa se puede dar de dos formas.

1) Pasando el campo llave del registro como parámetro. En este caso el programa no despliega la pantalla de acceso.

2) No pasando la llave del registro como parámetro lo que causa que el programa despliegue la pantalla de acceso para poder capturar la llave de registro a trabajar.

El segundo subtipo de programa MAN SAB permite dar mantenimiento a un único registro a la vez, por medio de un sistema de navegación en múltiples pantallas. Con este subtipo de programa se puede trabajar con los campos de las entidades relacionadas, por medio de una referencia (REF) o una dependencia (DEP), con la entidad para la cual se define el programa. El tipo de acceso para el programa requiere como parámetros la llave del registro al que se desea dar mantenimiento y el "modo" en que operará. El parámetro de "modo" se expone a confinación.

1. Permite únicamente la inserción de registros.
2. Permite únicamente el borrado de registros
3. Permite únicamente la modificación de la información del registro.

Este subtipo de programa es siempre secundario, ósea su forma de acceso será por invocación desde otro programa, por lo general el SEL NAV que hace la función de programa primario, siguiendo la metodología de programas SNAP.

El tercer subtipo de programa MAN LIN permite dar mantenimiento a registros por medio de líneas de detalle bajo un sistema de pantallas denominado Encabezado y su Detalle. Con este subtipo de programas se puede trabajar con los campos de las entidades relacionadas, por medio de una referencia o una dependencia con la entidad para la cual principalmente en el mantenimiento de la información de entidades relacionadas por medio de dependencias. El tipo de acceso para el programa se puede dar de dos formas.

1) Pasando la llave del registro como parámetro lo que provoca que el programa no despliegue la pantalla de acceso.

2) No pasando la llave del registro como parámetro lo que causa que el programa despliegue la pantalla de acceso para poder capturar la llave del registro.

El cuarto subtipo de programa MAN LAB permite dar mantenimiento a registros por medio de líneas de detalle bajo el sistema de pantallas: Encabezado y Detalle. Con este subtipo de programa se puede trabajar con los campos de las entidades relacionadas, por medio de una referencia o una dependencia, con la entidad para la cual se define el programa. El uso de este subtipo de programa se da principalmente en el mantenimiento de la información de entidades relacionadas por medio de dependencias.

El tipo de acceso para el programa requiere como parámetro la llave del registro al que se desea dar mantenimiento y el "modo" en que operará. El parámetro de "modo" tiene el siguiente significado:

1. Permite únicamente la inserción de registro.
2. Permite únicamente el borrado del registro.
3. Permite únicamente la modificación de la información del registro.

El subtipo de programa es siempre secundario, o sea su forma de acceso será por invocación desde otro programa, por lo general el SEL NAV que hace la función de programa automático.

El quinto subtipo de programa MAN MAS permite dar mantenimiento en forma masiva a los registros de una entidad por medio de líneas de detalle.

De esta forma se puede agregar nuevos registros de una entidad por medio de líneas de detalle.

De esta forma se puede agregar nuevos registros, eliminar registros y actualizar los registros. este subtipo de programa permite también trabajar bajo el esquema denominado Encabezado y su Detalle, con la diferencia de otros subtipos, que la información del encabezado debe existir antes de la invocación. este subtipo de programa puede trabajar con los campos de las entidades relacionadas, por medio de una referencia o una dependencia, con la entidad la cual se define el programa. El uso de este subtipo de programa se da principalmente en el mantenimiento de la información de entidades relacionadas por medio de dependencias.

Reportes

El cuarto tipo de programa es el programa automático de reportes (listados).

Este tipo de programa brinda la facilidad de generar cualquier tipo de reporte que se necesite sobre una entidad y, además, sobre las entidades relacionadas.

Tipo	Subtipo	Descripción
LIS		Reportes

Selección de Navegación

El quinto tipo de programa está especialmente diseñado para el procesamiento de la información, permitiendo ejecutar opciones sobre ítems de información seleccionados interactivamente. el propósito de este programa es el de integrar procesos definidos alrededor de la información, creando un ambiente de navegación por listas.

Este tipo de programa, automáticamente genera la lógica necesaria para poder acceder otros programas automáticos, que previamente se definieron para la entidad sobre la cual se está trabajando, dichos programas pueden ser: CON SIM, MAN SAB, MAN LAB, PRO BAT.

Entidad

Los programas automáticos están directamente asociados a una entidad, sin embargo esto no restringe de ninguna manera la fuente de la información manipulada en ellos, sino que, pueden ser elegidas todas aquellas entidades que guardan relación directa con la entidad base del programa.

Por lo anterior, el hecho que una entidad tenga referencias, dependencias o entidades dependientes, hará variar el algoritmo de los programas, pero se mantendrá la normalización en la

descripción de los módulos de proceso del programa, permitiendo una integración vía programa, de la información de la base de datos.

4.4.3.2 Especificaciones de Alto Nivel

Las especificaciones de alto nivel se utilizan para indicar que atributos de la entidad estarán presentes en el programa, definen las características de invocación del programa, pudiendo definir parámetros o una invocación directa del mismo.

Adicionalmente se podrá indicar la ubicación de cada atributo, ya sea en pantalla o en un reporte, así como texto asociado a este despliegue. Por otro lado los modos de despliegue en pantalla pueden ser definidos con las especificaciones de alto nivel.

En algunos casos se podrá indicar el uso de vías adicionales para implantar diferentes ordenamientos de la información (CON LIS Y LIS).

4.4.3.3 Normalización

Los parámetros de instalación de programas automáticos definen, para cada tipo de programa automático, ciertas características con el objetivo de normalizar la aplicación desarrollada por SNAP.

Estas normas tienen que ver con los procesos asociados a los mandatos o las opciones en un subarchivo, el nombre de las diferentes pantallas generadas automáticamente, el texto asociado a los mandatos en las líneas 22 y 23, etc. además, se establece el programa dentro del sistema que servirá de modelo para copiar un proceso inicial y para establecer el formato general que tendrán todas las pantallas o formatos de impresión en el caso de los reportes.

Normalización de Programas Automáticos

Como se explico al principio, SNAP ofrece la posibilidad de definir toda una serie de características normales para todos los programas automáticos generados en el sistema.

Las características propias de los parámetros de instalación de los programas automáticos son presentados en la opción 3 de la pantalla principal de SNAP, utilitarios. Sin embargo, se resume brevemente la forma de definir el programa modelo o programa base indicado en los parámetros de instalación de cada tipo de programa automático.

Programa Modelo o Programa Base

El programa base que se indica en los parámetros de instalación es un programa SNAP que debe estar definido en todos los sistemas, de otro modo se utilizará un modelo predefinido por SNAP en el sistema donde no existe. Cabe indicar, que no es necesario que este programa sea generado, porque lo que se utiliza son las definiciones fuentes y no los objetos.

Cada vez que se genera un programa automático se copia del programa base las estructuras de datos, la pantalla PANT01 (como encabezado de todas las pantallas de los programas automáticos a definir), y el modelo inicial indicado en los programas automáticos.

Quando el programa definido es un reporte automático (LIS) al crear la estructuras del reporte (y no durante la generación) se copia del programa base los formatos de impresión FMT@E1 para el encabezado del reporte y el FMT@F1 para el final del reporte. En caso de que no existan estos formatos en el programa base o no exista el programa base, se incluyen esos dos formatos de impresión en blanco y deben ser modificados por el usuario antes de generar el programa.

Quando un programa automático es generado se realiza los siguientes procesos con respecto al programa base:

Si Existe el programa base en el sistema

Se incluye todas las estructuras de datos del programa base.

Si no es un programa automático tipo LIS

Si existe la pantalla PANT01 en el programa base se copia la pantalla como encabezado de todas las pantallas sustituyendo los caracteres especiales.

Sino

Se copia una pantalla predefinido como encabezado de todas las pantallas sustituyendo los caracteres especiales.

Fin

Sino

Se copian los formatos de impresión

Fin

Si existe el modulo inicial indicado en parámetros de instalación.

Se incluye el modulo inicial que será ejecutado desde el MAIN (modulo principal)

Fin

Sino

Se incluye una pantalla predefinida por SNAP como encabezado de todas las pantallas sustituyendo los caracteres especiales.

Fin

Estructuras de Datos del Programa Base

Se definen en el programa base todas las entidades y variables de trabajo que se utilizarán en el módulo inicial. En la pantalla modelo o en los formatos de impresión de Encabezado y Fin de Reporte. Las entidades o variables definidas en el programa base pueden o no estar relacionadas en el modelo de datos con la entidad que se generará automáticamente.

En los programas de mantenimiento simple (MAN SIM/SAB) y los de mantenimiento con líneas de datare (MAN LIN/LAB) se puede definir en los parámetros de instalación una variable para contener el texto del modulo que se proceso (Alta, Baja o Cambio o el texto ahí indicado), por tanto, si se desea que ésta aparezca en la pantalla debe incluirse en las estructuras de datos y la pantalla del programa base.

Módulo Inicial

Se debe definir un módulo con el editor de módulos, en el programa base. El módulo definido no puede tener el nombres MAIN, M0001... M99999, pues estos están reservados para la generación de procesos en los programas automáticos SNAP.

Este módulo debe tener la finalidad de procesar u obtener cierta información para desplegar en la pantalla modelo o en los formatos de impresión de Encabezado y fin de reporte; o bien puede tener el propósito de realizar un proceso de verificación de autoridad o cualquier otro proceso que se desee incluir al inicio de todos los programas generados automáticamente.

Pantalla Modelo

Se debe definir una pantalla de formato libre con el editor de pantallas que llamada PANT01 en el programa base, cuyo objetivo es incluir un encabezado (parte superior) y que va a ser la modelo en todas las pantallas de los programas automáticos generados. Por tanto, se debe colocar aquella información adecuada para el funcionamiento del sistema en la parte superior de las pantallas definidas en los programas.

Formatos de Encabezado y Fin de Reporte

Para programas de reportes se definen dos formatos de impresión: FMT@E1 y FMT@F1 en el programa base.

En estos formatos de impresión se pueden incluir cualquier constante, o una de las variables definidas o atributos de las entidades incluidas en las estructuras de datos. La utilización de estas estructuras de datos, incluyendo los caracteres especiales, son exactamente igual que en la pantalla modelo.

Últimas consideraciones

El programa base es el mismo para todos los programas definidos en el sistema. Sin embargo es recomendable que ciertos tipos de programas automáticos tengan su propio programa base, esto es debido existen ciertas características propias del tipo de programa. Por ejemplo, los programas de reporte pueden necesitar ciertas variables o ciertos procesos que no utilizarán en los demás programas automáticos, por lo que se debería construir un programa base especial para ellos.

Otro ejemplo es que los programas de mantenimiento simple y con línea de detalle se quiere que el texto de modo aparezca en reverso, en los demás programas automáticos va aparecer ese campo en reverso, por lo que se debe utilizar otro programa base.

4.4.4. Método de desarrollo Acelerado

El MDA o Método de Desarrollo Acelerado es una opción en la cual se escoge el programa para su modificación o ajuste en la aplicación del sistema, para esto se emplean conceptos que emplea el SNAP que a continuación se explican:

Programación Estructurada

El SNAP provee las estructuras necesarias para crear programas sencillos de comprender, con lo que se disminuye así el tiempo requerido en las etapas de programación y mantenimiento.

El flujo de la lógica de un módulo siempre empieza con la primera instrucción, y sigue hacia abajo con las demás instrucciones una tras otra, excepto en el caso de una condicional (SI....SINO....FIN o CASO....CASO....SINO....FINCASOS), con la que se puede escoger que se ejecute un bloque de instrucciones, otro o ninguno, o en un ciclo (MIENTRAS....FIN o REPITA....FIN) donde se repite el bloque de instrucciones.

Programación Modular

La filosofía de programación SNAP es completamente modular; es decir, que los procesos deben descomponerse en sub-procesos más pequeños y sencillos de comprender, para facilitar la programación y mantenimiento. De esta manera, se construirán programas pequeños que ejecuten otros programas para simplificar los procesos. Así mismo, un programa debe componerse de varios módulos o rutinas de proceso que ejecuten partes específicas.

Todo programa SNAP empieza en el módulo 'MAIN', a partir del cual se puede llamar a los demás módulos del programa (Instrucción EJECUTAR módulo).

Se recomienda crear un módulo de inicialización del programa, donde se realicen todas aquellas operaciones iniciales, necesarias para realizar el resto del programa, como por ejemplo, inicialización de variables, lectura de la información de la base de datos, etc. Y luego, otro módulo donde se realice el cuerpo del programa, el cual puede utilizar otros módulos. Estos dos módulos deberán ser llamados por el módulo MAIN. Adicionalmente, el módulo MAIN se encarga de obtener la información especial que se le pasa al programa para su ejecución por medio de parámetros.

También se puede llamar a cualquier programa SNAP que esté en el repositorio central o a cualquier otro programa existente en el sistema, sin importar el lenguaje en que fue programado (Instrucción LLAMAR programa).

Por último, existen procesos de uso común que se han como funciones SNAP, y que están a disposición del usuario para que se utilicen en cualquier programa (Instrucción F Función). además, se puede programar nuevas funciones e incorporarlas al repositorio central.

Datos Disponibles

Dentro de la lógica de un programa se puede utilizar cualquier estructura de datos definida en el programa; a saber, campos DDS de las entidades seleccionadas, variables y arreglos de trabajo definidos en el programa, indicadores y constantes. Además, se utilizan las pantallas y formatos de impresión diseñados anteriormente. También puede utilizarse cualquier programa SNAP o cualquier otro programa del sistema, así como las funciones SNAP del repositorio central.

Utilización de Atributos y Entidades

Todo programa sólo tiene disponibles aquellas entidades que el modelo de datos son seleccionados para ese programa.

Tanto las entidades como los atributos se utilizan dentro de la lógica del programa por medio del nombre DDS asignado en los generadores del modelo de datos. Estos nombres son códigos de 6 caracteres comenzando con una letra.

Los nombres DDS de los atributos en una vía alterna de acceso son los mismos de la entidad a la que pertenece. Los nombres DDS de los atributos en una entidad lógica son diferentes a los de la entidad física a la que pertenece.

Utilización de Variables de trabajo

Las variables de trabajo se identifican con el código de 6 caracteres especificado en la definición de variables de trabajo. Existen varias variables reservadas con un uso específico, que puede utilizar el usuario si son declaradas en el programa. Estas variables se utilizan sobre todo para el manejo de pantallas y subarchivos.

Utilización de Arreglos de Trabajo

Los arreglos de trabajo pueden procesarse de dos maneras. Por tanto, pueden procesarse por elemento, es decir, procesando un dato a la vez. Para identificar un elemento de un arreglo se coloca el nombre del arreglo e inmediatamente le uno de los subíndices que indican el elemento requerido. El número de subíndices que se deben colocar se determina por las dimensiones del arreglo. Por ejemplo: VECTOR[ÍNDICE] ó MATRIZ[3, 20]. Por otro lado, puede procesarse en forma completa, o sea, que se procesan todos los datos en una sola operación. Para identificar un arreglo completo se coloca el nombre del arreglo completo e inmediatamente le sigue los paréntesis abierto y cerrado sin subíndices. Por ejemplo VECTOR[] ó MATRIZ[].

Utilización de Constantes

Las constantes o literales son datos digitados dentro de un programa cuyo valor no puede cambiar y está determinado. Pueden ser numéricas o alfanuméricas.

Existen 2 constantes predefinidas que se pueden utilizar en el SNAP, a saber, la constante alfanumérica *BLANCO, que se reemplaza por una hilera de blancos de tamaño indefinido (todo dato contiene blancos). Y la constante numérica o alfanumérica *CERO, que se reemplaza por el valor cero.

Funciones Intrínsecas

En el SNAP, las funciones intrínsecas sirven para realizar ciertos procesos u obtener cierta información especial, sin tener que definir variables especiales o llamar programas externos.

Existen 3 tipos de funciones intrínsecas: las que actúan como variables, las que actúan como constantes y las que son el resultado de un proceso. Las funciones que actúan como variables y constantes pueden utilizarse en cualquier instrucción del SNAP que permita el uso de variables y constantes. Las funciones que son el resultado de un proceso sólo pueden utilizarse en funciones específicas según el tipo de dato, es decir, aquellas cuyo resultado es numérico sólo pueden utilizarse en la instrucción CALCULAR, y aquellas cuyo resultado es carácter sólo pueden utilizarse en la instrucción Obtener. Sería muy amplio explicar cada una de ellas por lo que se recomienda consultar el manual de Referencia al Lenguaje SNAP.

Compartir Datos entre Módulos y Programas

Hay que tomar en cuenta que las estructuras de datos son globales para un programa, lo que quiere decir que para compartir un dato entre procesos hay que tomar en cuenta lo siguiente:

- Ejecución de un Módulo dentro del programa: todas las variables de un programa están disponibles en todo programa, por lo que no se necesita realizar ningún proceso especial para utilizar el dato requerido.
- Llamada a otro programa o función SNAP: Todas las variables de un programa son propias de éste e independientes de otro, por lo que para pasar un dato de un programa a otro se debe realizar por medio de parámetros.

Utilización de Indicadores

Junto a las estructuras de datos, los programas también pueden utilizar los indicadores para condicionar ciertos procesos.

Un indicador es un dato que puede almacenar dos únicos valores:

'0' Apagado
'1' Encendido

Existen 99 indicadores que identifican con los códigos del *F01 al *F99.

Se utilizan principalmente para:

- Teclas de mandato o Función
- Verificación automática en pantalla
- Indicador de evento asociado a una entidad
- Indicadores especiales
- Verificación en pantalla de sub-archivos en programas automáticos
- Condicionar los atributos de los campos de pantalla

Utilización de Condiciones

Las condiciones son expresiones que se utilizan para controlar el flujo de instrucciones que se van a ejecutar en una instrucción condicional.

Utilización de Hileras

En el SNAP las hileras son de tamaño fijo, es decir, su longitud es determinado por el usuario al definir el atributo o variable de trabajo. En el caso de constantes de caracteres, su longitud se determina por la cantidad de caracteres entre los apóstrofes que lo delimitan.

Sin embargo, el SNAP provee la facilidad de manejar las hileras por medio de funciones intrínsecas desde las instrucciones OBTENER y CALCULAR que permiten manejar esas hileras con una longitud dada en la instrucción o la longitud eliminando por el sufijo en el nombre de la función.

Instrucciones y Macro-Instrucciones

El lenguaje de programación del SNAP tiene un conjunto de instrucciones completo para realizar cualquier operación o proceso. Adicionalmente, proporciona un conjunto de macro-instrucciones y funciones de biblioteca que realizan procesos muy complejos y tediosos, con el fin de liberar al programador de hacerlos. Estas instrucciones de muy alto nivel que hacen del SNAP uno de los lenguajes más poderosos y versátiles existentes en el ambiente de herramienta CASE.

El lenguaje SNAP es completamente auto suficiente; es decir, que aún cuando genere programas fuente RPG, el programador no tiene que recurrir a éstos para incluir ciertos procesos, sino que la programación del SNAP produce programas completos y listos para su ejecución.

Otra gran ventaja de este lenguaje es que deja atrás la utilización del inglés para incorporar instrucciones en español, facilitando la utilización de cada una de ellas por intuición sin necesidad de conocer un idioma diferente.

4.5 Ejemplo de Utilización de programas automáticos mediante un requerimiento

En esta parte se tomará un requerimiento real, tomado del sistema COP el cuál no es nada complicado con el objetivo de introducir al lector a la utilización de estos diferentes tipos de programas y en que se simulará un nuevo desarrollo para satisfacer este requerimiento, también se mencionará la metodología que debe seguirse para el desarrollo y conclusión de los diferentes programas y en caso de que no se ajusten en el diseño requerido, su modificación mediante el método de desarrollo acelerado.

Definición de requerimiento:

Se necesita una pantalla en donde aparezcan los diferentes clientes en forma de lista, en donde aparecen ordenados con su código asignado, junto con éste, su nombre o denominación social. En esta pantalla se podrán hacer altas, cambios y bajas de clientes; en altas y cambios se podrá dar mantenimiento a todos los campos o atributos que estén en la entidad de "Clientes". Mediante una opción se podrá hacer una consulta por cada cliente en donde aparecerán todas las características de éste. Con otra opción se podrá consultar la lista de el personal asignado con el cliente, una opción diferente también servirá para dar mantenimiento de altas bajas y cambios de el personal asignado a un determinado cliente. Mediante una tecla de función se podrá mandar listar los clientes junto con este el personal asignado.

4.5.1 Definición de Programas Automáticos

Para definir programas automáticos se utiliza la opción 18=Programas de la pantalla de mantenimiento del Modelo de Datos. Esta opción nos lleva a la siguiente pantalla.

SNAP Rovers 3.0 Derechos Reservados SOR C. A. 6/01/1995 16 30 42

Sistema Control de Proyectos

Mantenimiento de Programas Automáticos

.....

Inidad: CLIENTES Clientes de la Empresa

.....

2 = Cambiar 4 = Borrar 8 = Listar 12 = Trabaja 13 = Detenerse

18 = MDA

Ser	Tipo	Subto	Subdes	Programa	Nombre
-----	------	-------	--------	----------	--------

F4 = Agregar, F12 = Anular, F18 = Estado de Generación

Pantalla de mantenimiento de programas automáticos

Importante: La secuencia en que se deben definir los programas de una entidad para aprovechar al máximo las posibilidades de la herramienta debe ser la siguiente:

Tipo	Sub-tipo	Descripción
ELI		Verificación de Eliminación
ELI	CAS	Eliminación en Cascada
CON	LIS	Consulta por Lista
CON	SIM	Consulta Simple
MAN	SAB	Mantenimiento Simple ABC
MAN	LAB	Mantenimiento con Líneas de Detalle ABC
MAN	SIM	Mantenimiento Simple
MAN	MAS	Mantenimiento Masivo
PRO	BAT	Programas de ejecución Transaccional
LIS		Programa de Listado
SEL	NAV	Selección por Navegación

El motivo de seguir dicha secuencia es que a la hora de definir cada uno de los programas se tengan definidos los programas que puedan ser asociados automáticamente.

Tomando del requerimiento se observa que se realizarán siete tipos de programas, los cuales son:

SEL	NAV	Sesión de clientes
MAN	SAB	Mantenimiento de alta, cambio y baja de clientes
CON	SIM	Consulta simple de clientes
MAN	LIN	Mantenimiento a del personal asignado a clientes
ELI		Verificación de eliminación de clientes
CON	LIS	Consulta personal asignado a clientes
LIS		Listado de clientes con personal asignado

Para definir estos nuevos programas se utiliza el mandato F8=Agregar de la pantalla de mantenimiento de programas automáticos, después de esto se despliega la siguiente pantalla de definición de programas.

SNAP Release 3.0 Derechos Reservados SOH C R 6/01/1995 19 30 42

Sistema Control de Proyectos

Mantenimiento de Programas Automáticos

Entidad: CLIENTES Clientes de la empresa

Tipo: COD

Subtipo: LIS

Subprograma: COP

Programa: CPO004

F8 = Listar. F12 = Ayuda.

FALLA DE ORIGEN

En la pantalla anterior se trabajan los siguientes campos para definición de programa automático. Estos campos no podrán ser cambiados posteriormente con excepción del sub-sistema donde se agrupará el programa.

Tipo: Especifica el tipo de programa automático que se crea.

Subtipo: Especifica el subtipo de programa automático para el tipo de programa elegido.

Sub-sistema: Indica el sub-sistema al cual estará asociado el programa. Los programas del sistema se agrupan en sub-sistemas definidos en esté, en el caso del sistema COP el sub-sistema es sólo uno y este tiene el mismo nombre del sistema. El campo es inicializado con el nombre del sub-sistema donde reside la entidad base del programa.

Programa: Especifica el nombre del programa.

Una vez que se ha seleccionado el programa automático debe darse INTRO y se desplegará los siguientes campos adicionales como se muestra a continuación. Estos campos pueden ser modificados posteriormente con la opción 2=Cambiar en la pantalla de mantenimiento de programas automáticos tal como se explicó anteriormente.

SNAP Release 3.0 Derechos Reservados SOH C. R.		01/1996	18 30 42
Sistema Central de Proyectos			
Mantenimiento de Programas Automáticos			
.....			
Entidad	CLINNES	Clientes de la empresa	
.....			
Tipo	COH		
Subtipo	LIS		
Subsistema	COP		
Programa	CDP006		
Descripción: Consulta lista de personal asignado			
Tipo			
Propietario	SOH		
Nivel de Seg.	5		
F8=Listar, F12=Atender.			

Descripción: Permite incluir la descripción del programa dentro de la aplicación. Este nombre podrá contener la función del programa y el nombre de la entidad base para que el programa puede ser identificado fácilmente. Cuando el programa de estándares se haya definido el despliegue de la descripción del programa, el valor de este campo será el que se sustituya en las pantallas.

Tipo: Este campo es de salida e indica si el sistema fue definido automáticamente o bien fue alterado en el MDA.

Propietario: Especifica el usuario que definió el programa automático. El valor de este campo contiene el nombre del perfil de usuario que abrió la sesión.

FALLA DE ORIGEN

Nivel: Especifica el nivel de seguridad del programa, este nivel puede estar entre 0 y 5: El significado de este nivel de seguridad se detalla en el manual de Utilitarios y Seguridad.

Adicionalmente, en las pantallas de definición de programas automáticos de listados, tipo LIS, aparecen otros campos para definir estos programas que a continuación se definen:

Vía de Acceso: Este campo captura el grupo de la vía de acceso sobre la cual se definirá el programa de reporte definido, al solicitar una lista de valores en el campo, mandato F4; la lista de valores desplegada contendrá todas las vías de acceso definidas sobre la entidad base en la que se define el programa. A esta lista se agregan dos valores importantes:

- 00 Este valor especifica que el programa trabajará sobre el ordenamiento principal definido en la entidad.
- XX Cuando se define este valor se indica que el programa trabajará sobre el un acceso que se definirá en el momento de ejecución desde un programa en lenguaje de control generado en el menú de comandos utilitarios del SNAP: OPNQRYF.

Quando el programa definido se trate de un Mantenimiento Simple con Líneas de Detalle o un mantenimiento Simple con Líneas de Detalle ABC, se debe incluir también el nombre de la entidad dependiente a la que mantendrá junto con cada registro.

Entidad Detalle: este campo se captura el nombre de la entidad dependiente a la que se dará mantenimiento en el detalle; esta entidad debe estar relacionada con la entidad asociada al programa. El mandato F4 en el campo despliega una lista de escogencia con entidades válidas relacionadas.

Si el programa que se desea incluir es de tipo Selección por Navegación se desplegará un campo adicional que se define a continuación:

Vía de Acceso. En este campo se debe indicar el número de la vía de acceso, por la cual se desea ordenar los campos en la pantalla de despliegue de este programa. Si se desea utilizar la llave primaria de la entidad se debe indicar la vía de acceso 01.

4.5.2 Programa de Verificación de Eliminación de Clientes

Concepto. El programa automático de Verificación de Eliminación, verifica la consistencia de la base de datos cuando se elimina un registro de la entidad a la que está asociado. Este programa recorre los registros de las entidades dependientes o referentes a la entidad a la que está asociado, revisando que el valor campo llave se encuentra en una entidad dependiente o referente, el programa devuelve un indicador de violación de la consistencia de la base de datos.

Este programa no borra el registro; el programa encargado de ello, normalmente un programa de mantenimiento, efectúa la invocación a este programa implícitamente al ejecutar la instrucción BORRARE (refiérase al manual de referencia al Lenguaje SNAP.)

Pasos Previos

El programa de verificación de eliminación normalmente es el primer programa automático que se define a una entidad junto con el programa de Consulta de Lista de Valores (CON LIS). La razón de

ello es que son programas secundarios que no necesitan otros programas para generarse correctamente, sino más bien son invocados por otros.

Para lograr el máximo aprovechamiento de un programa de verificación de eliminación se deben de tomar las siguientes consideraciones:

- El modelo de datos del sistema que se trabaja debe estar completamente terminado, por cuanto la generación del programa se basa en las relaciones de éste y de no estar terminado la verificación de la consistencia de la base de datos no tomará en cuenta relaciones incluidas, modificadas o eliminadas posteriormente a la generación del programa. Para solucionar problemas de este tipo basta con volver a generar automáticamente el programa.

Uso del Programa

El programa de verificación de eliminación es uno de los programas automáticos más importantes, se recomienda utilizar éste para todas las entidades definidas en el sistema.

Definición

Para crear un programa de Verificación de Eliminación para la Entidad de Clientes, únicamente se debe definir un programa tipo ELI. Los programas de este tipo solamente tienen que incluirse y luego generarse con la opción 14=Generar.

Generación

El programa de Verificación de eliminación de Clientes generará automáticamente:

a) Las vías de acceso en las entidades que dependen o se refieren a la entidad base, necesarias para realizar las verificaciones. Estas vías de acceso se generarán con los grupos a partir del número 50.

b) Entradas a las estructuras de datos con la entidad primaria del programa y cada una de las entidades en las que se haga referencia a la entidad base, las entidades serán del tipo L (Lectura).

4.5.3 Programa Consulta de Personal Asignado a Clientes

Concepto

El propósito de este tipo de programa es brindar un soporte de consulta a nivel de campo en la pantalla del programa invocado. Este programa retorna el valor de las llaves de un registro escogido de la lista desplegada.

Una vez definido el programa de consulta por lista sobre la entidad, la cual esta vez el programa se realiza en la entidad PERSONAL/CLIENTES debido a que la entidad CLIENTES tiene una relación de dependencia de la PERSONAL/CLIENTES.

Todos los programas interactivos con alguna relación con ésta, podrán hacer uso de esta consulta mediante la utilización del mandato F4=Lista en la pantalla en donde se presenta este campo. Los programas automáticos del SNAP se encargan de realizar la integración de éste programa en las pantallas donde se definan los campos de la entidad automáticamente.

Cuando el programa de consulta por lista de valores es accesado despliega una lista con los registros de la entidad base, un encabezado con campos de entidades de las que depende y un campo de selección junto a cada registro que determinará cual de ellos, es elegido por el usuario, será devuelto al programa que lo invocó. Adicionalmente brinda un ambiente en donde el usuario podrá seleccionar el campo por el cual se realizará el ordenamiento de la lista; este estará definido por una vía de acceso en la entidad.

Pasos Previos

Se debe asegurar que los parámetros de generación para el tipo de programa sean los correctos; éstos pueden ser modificados para programas automáticos en los utilitarios del SNAP. El programa de estándares para la generación de programas automáticos deberá ser el indicado para el sistema donde residirá el programa de consulta de lista de valores. La importancia de este programa radica en que, según la definición de este, se establecerá el formato de pantallas y el módulo de inicialización del programa, así como las variables necesarias para su correcta generación.

Para aprovechar al máximo las facilidades de este tipo de programa se sugiere revisar los siguientes puntos:

- Se deben definir las vías de acceso sobre la entidad sobre la que se quiere permitir el ordenamiento en el programa de consulta de lista de valores. Con ello se garantiza su disponibilidad cuando se definan las especificaciones de alto nivel para el programa.

Uso del Programa

El programa de consulta de Lista de personal asignado a clientes debe ser definido antes de cualquier otro programa automático interactivo, pues SNAP los asocia automáticamente a los programas posteriormente generados.

El programa de consulta lista de clientes debe utilizarse cuando se encuentre en al menos una de las siguientes situaciones:

- La entidad a la que se asociará tiene entidades referentes o dependientes.
- El mantenimiento de la entidad se hará interactivamente por el usuario del sistema. En otras palabras, los registros de la entidad no serán generados por un proceso dentro del sistema.
- La entidad tiene procesos interactivos donde el usuario debe ingresar los valores del campo llave del registro.

Definición

Para la definición del programa de consulta de lista de personal asignado a clientes se debe seguir una serie de pasos, los cuales en términos generales especifican que información debe estar presente en la pantalla, que ordenamientos posibles tendrá a su disposición el usuario en el momento de ejecución de esta consulta y de cuál registro iniciará la lista según el ordenamiento de las llaves de la entidad.

Lo anterior se especifica utilizando la opción 12=Trabajar de la pantalla de mantenimiento de programas. Luego de utilizar esta opción sobre el programa de Consulta de Lista de valores se despliega la pantalla mostrada en la siguiente figura:

SNAP Revise 3.0 Derechos Reservados SOH C. R.		8/01/1988	19 30 42
Sistema Consult de Proyectos			
Programa Consulta por lista			
.....			
Entidad	:	PERSONALCLIENTES	
.....			
1 = Seleccionar			
Sub	:	Descripción	
-	:	Especificar Atributos	
-	:	Especificar Vistas de Acceso	
-	:	Mantenimiento	
.....			
F12 = Avanzar			

Especificar Atributos

En el proceso de Especificar Atributos se encuentran las opciones para seleccionar los campos que formarán parte en la pantalla de la consulta de lista de valores. Para cada campo seleccionado, se especifican las características que tendrá el programa generado, con su ubicación en el despliegue o valor de posicionamiento en la ejecución.

La pantalla para especificar atributos de este tipo de programa se muestra a continuación.

SNAP Release 3.0 Derechos Reservados BOM C. R. 6/01/1998 18 30 42

Sistema Control de Proyectos
Programa Consulta por lista

Entidad PERSONALCLIENTES

4 = (Detalle)

Seq	SEC	Entidad/Atributo	TIP	Or	ENC	DET	USO	Descripción	Len
-	10	PERSONALCLIENTES							
		Código Cliente	AID	L	S	N		Cliente	
	20	PERSONAL							
		Código Personal	AID	L	N	S		Cód. Personal	
	30	PERSONAL							
		Nombre	ATR	L	N	S		Nombre	

F6 = Agregar. F12 = Anterior

Para seleccionar los atributos que se desplegarán en la pantalla de la consulta se utiliza el mandato F6=Agregar, con esta opción se desplegará una lista con campos de la entidad y las entidades con que se relaciona por una dependencia o referencia

Para cada atributo seleccionado se definirán las siguientes características.

SEC: Indica la secuencia de aparición del campo en la pantalla generada por el programa, este campo variará su significado según la posición donde se defina el campo en la pantalla utilizando los campos USO ENC y USO DET, así, se define:

- Como el orden de aparición en el encabezado de la pantalla cuando el valor de USO ENC sea "S".
- Como la posición en la línea de l sub-archivo que desplegará la lista de los registros de la entidad cuando el valor de USO DET sea "S" o "A".

ORI: Este campo indica el origen del campo seleccionado, donde se desplegará el campo: en el encabezado o el detalle. Los valores posibles para el campo son los siguientes:

E: Indica que el campo seleccionado se desplegará en el encabezado de la pantalla. Un valor "E" es inicializado cuando el campo pertenece a una entidad con una relación de dependencia.

L: Indica que el campo seleccionado se desplegará en la línea de detalle. El valor por omisión será "L" cuando el campo sea un atributo de la entidad o un campo en una relación de referencia de la entidad.

Este campo es inicializado por SNAP dependiendo del origen del campo seleccionado, sin embargo puede ser modificado. Los campos que se presentan en el encabezado actúan como filtros de los registros que se presentan en el detalle, siempre y cuando la estructura de los campos llave de la entidad permita implantarlo, es decir los campos llaves desplegados en el encabezado deben preceder a los definidos en la línea de detalle.

FALLA DE ORIGEN

USO ENC: Este campo es utilizado para definir la característica que tendrá el atributo seleccionado en el encabezado de la pantalla. Los posibles valores para este campo son los siguientes:

N: El atributo no se muestra en la parte superior de la pantalla.

S: El atributo se desplegará en la parte superior de la pantalla, como un campo de salida únicamente.

USO DET: Este campo es utilizado para definir las características del campo seleccionado en la línea de detalle del sub-archivo. Los valores que pueden definirse para este campo son los siguientes:

N: El atributo no se muestra en la línea de detalle del sub-archivo generado.

S: El atributo aparecerá en la línea de detalle de la pantalla como campo de salida.

A: El atributo se despliega en el detalle de la consulta como campo de salida, pero adicionalmente se define un campo con las mismas características en el encabezado sobre la posición del campo. Este campo del encabezado permitirá capturar un valor del campo seleccionado a partir del cual desplegará la lista de registros del sub-archivo.

Descripción: En este campo se captura el texto asociado a cada campo en el encabezado de la pantalla donde se definió con un atributo de despliegue. Esta descripción generará una constante en la pantalla justo antes del despliegue del campo a partir de las columnas 1 a 40 de la línea donde se ubique éste. El valor del este campo es inicializado automáticamente por SNAP a partir del nombre dado a cada atributo en el Modelo de datos.

En el caso de que este se deje en blanco SNAP no asignará una constante antecedente a la variable.

Lin: Indica el número de líneas que separan a un campo del anterior desplegado en el encabezado de la pantalla.

El valor por omisión es cero, en este caso SNAP acomodará los campos en las pantallas donde se halla definido el campo automáticamente. Esto es acomodado tantos campos por línea como su longitud le permita separando cada uno de ellos con un espacio en blanco. Cuando estos campos tengan una descripción asociada ésta se desplegará a partir de la columna 1 a 40 de la pantalla si hay espacio disponible.

Vías de Acceso

Esta opción permite seleccionar vías de acceso sobre la cuál se especifica el programa. Al acceso por las vías seleccionadas se le asigna un mandato, permitiendo al usuario elegir diferentes ordenamientos sobre la información desplegada en la pantalla de la consulta.

En la opción de vías de acceso se despliega la siguiente pantalla, en ella se muestran las vías de acceso definidas la entidad para elegir los posibles ordenamientos del sub-archivo de detalle.

SNAP Release 3.0 Derechos Reservados SOW C. R. 8/31/1995 18:30:42

Sistema Control de Proyección
Programa Consulto por lista

Entidad PERSONAL CLIENTES

DEFINICIÓN DE VÍAS DE ACCESO

1 = Selección, 4 = Eliminar

Del	ID	Descripción	Tecla de función	Incluir
-	00	ACCESO PRIMARIO	0	SI
-	01	Numero Personal	1	SI

F12 = Anterior

Para incluir el ordenamiento por una vía de acceso en el programa se selecciona la vía con la opción 1=Selección y se asigna el mandato a la que se asociará esté. Para asignar un mandato a un ordenamiento se indica el número de la tecla de función correspondiente en el campo.

Mantenimiento

En la opción de mantenimiento se define el texto de las líneas 22 y 23 de la pantalla del programa de consulta, además se puede escoger un campo de la entidad para realizar búsqueda fonética sobre el valor del campo en los registros del detalle y filtra la información por la ocurrencia de una sub-hilera en ese campo.

Al elegir la opción se presenta la siguiente pantalla en la que se define el campo sobre el que se hará la búsqueda fonética y las líneas de mandato.

SNAP Release 3.0 Derechos Reservados SOW C. R. 8/31/1995 18:30:42

Sistema Control de Proyección
Programa Consulto por lista

Entidad PERSONAL CLIENTES

Programa EDPOS2

Descripción : Consulta Personal Asignada a Clientes

Atributo de Búsqueda : Nombre del personal _____

Descripción línea 24 : _____

Incluir = Código, 11 = Por nombre, F12 = Anterior _____

F12 = Anterior

Atributo de búsqueda: este campo se especifica un campo de entidad. No es necesario que este campo tenga relación con las vías de acceso definidas. El campo seleccionado tiene que estar en el detalle de la consulta y se utiliza para búsquedas fonéticas de sub-hileras o por contenido.

La búsqueda fonética consiste en localizar una sub-hilera de caracteres, capturados por el programa dentro del campo especificado, haciendo caso omiso entre mayúsculas, minúsculas y la tabla de equivalencia de caracteres definida en SNAP, referirse al manual de Utilitarios SNAP en Parámetros de Instalación.

Con la búsqueda fonética el programa desplegará en la línea de detalle solamente aquellos registros donde haya encontrado ocurrencia de la hilera ingresada por el usuario, sin importar el ordenamiento de los registros.

Descripción Línea 24 En este campo se especifica la información que el programa de consulta de lista de valores definido desplegará en la línea de mandatos. Usualmente esta hilera contiene los mandatos permitidos para el programa, sean estos las vías de acceso definidas y los mandatos de control.

Generación

El programa de Consulta Lista de Clientes generará:

a) Entradas en la estructura de datos con la entidad primaria del programa, las vías de acceso definidas en cada uno de los ordenamientos y cada una de las entidades a las que hace referencia en alguno de sus campos, las entidades serán del tipo L (Lectura). Además incluirá las entidades que se encuentren definidas en el programa de estándares.

b) Variables de trabajo para la manipulación de datos y las definidas en el programa de estándares.

c) Macros para las sustituciones dentro del archivo del SNAP, sólo si el SNAP está configurado con esta opción en los parámetros de generación de esta herramienta.

d) Un archivo fuente de pantallas con un registro dividido en: la pantalla del encabezado, el sub-archivo y las líneas de mandatos.

e) Los módulos de ejecución del programa, la estructura modular del programa se muestra en el diagrama de referencia del programa de consulta de lista de valores. A los módulos generados se agregan los módulos definidos en el programa de estándares definido en el sistema.

Este programa será modificado por la opción 18=MDA en la pantalla de mantenimiento de programas automáticos, ya que este programa normalmente cuando es invocado recibe parámetros de la entidad encabezado y parámetros en blanco de la entidad detalle, cuando se ha elegido en registro de la entidad de detalle el programa retorna el valor del registro de detalle que se ha seleccionado. Para nuestro caso como este programa es de consulta solamente, no necesitamos este valor de parámetro por lo cual con la opción 18= MDA podemos quitar estos parámetros de retorno de la entidad de detalle en el módulo MAIN en la primera instrucción PARÁMETROS. Basta generar el programa para que quede listo esta parte del requerimiento y la pantalla generada y que se presenta al llamar al programa de consulta es la siguiente..

SNAP Pantalla 3 0 Derechos Reservados SCN C. R.		BG111995	18 30 42
Sistema Central de Proyectos		CDP0058	PANT01
Programa Consulta por lista		Consulta	

Cliente	12 Fondo de Cultura Económica		
	Nombre _____		
Código	Nombre del personal		
001	José Ramos Contreras		
002	Isabel Fábrega Ramos		
003	Marco Siles Pacheco		
004	Sergio Vargas Sanchez		

Intro = Código, F1 = Por nombre, F12 = Anterior			

4.5.4 Consulta Simple a Clientes

Concepto

El programa consulta simple despliega en una o varias pantallas definidas en su especificación los valores de los campos de un registro de la entidad a la que se define. Estos campos no podrán ser modificados por el usuario.

Este programa permite consultar el valor de un registro o de un cliente a la vez. Este registro puede ser parametrizado al programa, en cuyo caso lo convierte en un programa secundario para ser invocado por otro; o bien el mismo programa podrá capturar la llave del registro deseado.

Pasos Previos

Se debe asegurar que los parámetros de generación para este tipo de programa sean los correctos, éstos pueden ser modificados en los programas automáticos en los utilitarios del SNAP. Este programa se debe de realizar en la entidad CLIENTES.

El programa de estándares para la generación de programas automáticos deberá ser indicado para el sistema donde residirá el programa de consulta simple, la importancia de este programa radica en que, según lo definido en éste, se establecerá el formato de pantallas y el módulo de inicialización del programa, así como las variables necesarias para su correcta generación.

Para aprovechar al máximo las facilidades del programa de consulta simple, se sugiere revisar los siguientes puntos:

- En caso de que la entidad dependa de otras entidades y además estos valores no sean parámetro, es conveniente que los programas automáticos CON LIS de estas entidades ya hayan sido generados, para que el programa de consulta simple los asocie automáticamente a la verificación de los campos correspondientes en la pantalla de acceso para su invocación por medio del mandato F4. En caso contrario, en mandato F4 no tendrá funcionalidad.
- En caso que los campos llave de la entidad a la cual se asocia el programa CON SIM tengan lista de valores, es necesario que estas listas de valores estén previamente definidas con al menos un valor para que el programa lo asocie automáticamente. En caso contrario no se contará con la función de la tecla F4 para desplegar está lista.

Uso del Programa

Este programa deberá utilizarse en aquellos casos donde se presenten algunas de las siguientes circunstancias:

- Se requiere un programa de consulta para una entidad que pueda ser invocado desde un menú o línea de mandato, pues en la generación de este sub-tipo de programa puede especificarse el uso o no de sus campos llave como parámetro.
- La consulta de registros de una entidad requiere de una acción única para cada uno de ellos, ya que las características del programa definen esta consulta para procesar un registro a la vez.
- Se requiere de un programa de consulta para los registros de un programa de lista, tipo SEL NAV. En este caso se debe generar el programa de consulta simple con todos sus campos llave como parámetro.

Definición

Una vez seleccionado este tipo de programas, con la opción 12=Trabajar, se presenta la pantalla de especificaciones de alto nivel de programas de consulta simple mostrada a continuación.

SNAP Release 3.0 Derechos Reservados SON C R				6/01/1998			18 30 47	
Sistema Control de Proyectos								
Programa Consulta Clientes								

Entidad: CLIENTES								

# = Elemento				ACCES			DATOS	
Seq	SIC	Entidad/Atributo	COD	PO1	PO2	PO3	Descripción	Lim
10		CLIENTES						
		Código Cliente	AD	N	S	N	Cliente	
20		CLIENTES						
		Nombre del Cliente	ATR	N	S	N	Nombre	
30		CLIENTES						
		Dirección	ATR	N	S	N	Dirección	
40		CLIENTES						
		Teléfono	ATR	N	S	N	Teléfono	
50		CLIENTES						
		Encargado semana	ATR	N	S	N	Encargado	

F8 = Agragar. F12 = Actualizar

FALLA DE ORIGEN

SNAP Puntos 3 y Derechos Reservados SDIN C. R.		0-01/1989	18 30 42
Sistema Control de Proyectos		CGP0088	PAN001
Programa Consulta por lista		Consulta	

Código: 12 Fondo de Cultura Económica			
Nombre: _____			
Código	Nombre del personal		
001	Jesús Romero Contreras		
002	Isabel Fonseca Ramirez		
003	Maria Mercedes Pacheco		
004	Sergio Vargas Sanchez		
Inicio = Código, F1 = Por nombre, F12 = Anterior			

4.5.4 Consulta Simple a Clientes

Concepto

El programa consulta simple despliega en una o varias pantallas definidas en su especificación los valores de los campos de un registro de la entidad a la que se define. Estos campos no podrán ser modificados por el usuario.

Este programa permite consultar el valor de un registro o de un cliente a la vez. Este registro puede ser parametrizado al programa, en cuyo caso lo convierte en un programa secundario para ser invocado por otro; o bien el mismo programa podrá capturar la llave del registro deseado.

Pasos Previos

Se debe asegurar que los parámetros de generación para este tipo de programa sean los correctos, éstos pueden ser modificados en los programas automáticos en los utilitarios del SNAP. Este programa se debe de realizar en la entidad CLIENTES.

El programa de estándares para la generación de programas automáticos deberá ser indicado para el sistema donde residirá el programa de consulta simple, la importancia de este programa radica en que, según lo definido en éste, se establecerá el formato de pantallas y el módulo de inicialización del programa, así como las variables necesarias para su correcta generación.

Para aprovechar al máximo las facilidades del programa de consulta simple, se sugiere revisar los siguientes puntos:

FALLA DE ORIGEN

- En caso de que la entidad dependa de otras entidades y además estos valores no sean parámetro, es conveniente que los programas automáticos CON LIS de estas entidades ya hayan sido generados, para que el programa de consulta simple los asocie automáticamente a la verificación de los campos correspondientes en la pantalla de acceso para su invocación por medio del mandato F4. En caso contrario, en mandato F4 no tendrá funcionalidad.
- En caso que los campos llave de la entidad a la cual se asocia el programa CON SIM tengan lista de valores, es necesario que estas listas de valores estén previamente definidas con al menos un valor para que el programa lo asocie automáticamente. En caso contrario no se contará con la función de la tecla F4 para desplegar está lista.

Uso del Programa

Este programa deberá utilizarse en aquellos casos donde se presenten algunas de las siguientes circunstancias:

- Se requiere un programa de consulta para una entidad que pueda ser invocado desde un menú o línea de mandato, pues en la generación de este sub-tipo de programa puede especificarse el uso o no de sus campos llave como parámetro.
- La consulta de registros de una entidad requiere de una acción única para cada uno de ellos, ya que las características del programa definen esta consulta para procesar un registro a la vez.
- Se requiere de un programa de consulta para los registros de un programa de lista, tipo SEL NAV. En este caso se debe generar el programa de consulta simple con todos sus campos llave como parámetro.

Definición

Una vez seleccionado este tipo de programas, con la opción 12=Trabajar, se presenta la pantalla de especificaciones de alto nivel de programas de consulta simple mostrada a continuación.

SNAP Release 3.0 Derechos Reservados SOW C. R.									
Sistema Control de Proyectos									
Programa Consulta Clientes									

Entidad CLIENTES									

4 = Entidad									
ACCES DATOS									
Seq	SEC	Entidad/Atributo	COD	PO1	PO2	PO3	Descripción	Lim	
	10	CLIENTES							
		Código Cliente	AD	N	S	N	Clave		
	20	CLIENTES							
		Nombre del Cliente	ATR	N	S	N	Nombre		
	30	CLIENTES							
		Dirección	ATR	N	S	N	Dirección		
	40	CLIENTES							
		Teléfono	ATR	N	S	N	Teléfono		
	50	CLIENTES							
		Encargado sistema	ATR	N	S	N	Encargado		
F8 = Ayudar		F12 = Anterior							

Para poder seleccionar los atributos que se desplegarán en el proceso de consulta se utiliza el mandato F6=Agregar de la pantalla anterior.

Una vez seleccionados los atributos, se desplegarán con valores por omisión dados por SNAP. Para modificar estos valores sobre los atributos, se debe tomar en cuenta las siguientes consideraciones sobre los campos desplegados.

SEC: Indica la secuencia de aparición del campo en las pantallas generadas por el programa.

P01: Esta columna corresponde a la pantalla de acceso al programa. En este campo se definen las características de despliegue de los campos llave de la entidad en la primera pantalla del programa o pantalla de acceso. En esta pantalla también se pueden definir atributos de la entidad o entidades relacionadas, pero en este caso el atributo sólo puede ser de "Salida".

Los valores posibles para este campo son:

S: Indica que el campo es de Salida
N: Indica que el campo no aparece en pantalla
A: Indica que el campo es de Entrada/Salida.

Si el valor es "A" para los campos llave, SNAP generará la lógica necesaria para capturar la llave antes de desplegar el contenido del registro, pero si se indica "S" o "N" para estos, se incluirá el valor de este campo como parámetro del programa. El valor por omisión para el campo será de "A" para los campos llave, en este caso el programa no será invocado con parámetros.

P02: Indica la característica de despliegue del campo en la segunda pantalla generada por el programa.

Los valores posibles de este campo son:

S: Indica que el campo es de "Salida"
N: Indica que el campo no aparece en la pantalla

P03: Este campo cumple con la misma función que el anterior pero para la tercera pantalla generada.

Descripción En este campo se captura el texto asociado a cada campo en las pantallas donde aparezca el campo justo antes del despliegue de éste a partir de las columnas 1 o 40 de la línea donde se ubique el campo.

El valor de este campo es inicializado automáticamente por SNAP a partir del nombre dado a cada atributo en el Modelo de Datos.

En el caso de que este campo se deje en blanco SNAP no asignará una constante antecedente a la variable.

LIN: Indica el número por omisión es cero, en este caso SNAP acomodará los atributos en las pantallas donde se halla definido el campo automáticamente. Esto es acomodando tantos campos por línea como su longitud lo permita separando cada uno de ellos con un espacio en blanco. Cuando estos campos tengan una descripción asociada ésta se desplegará a partir de la columna 1 a 40 si hay espacio disponible.

En muchos casos no será necesario utilizar las tres últimas pantallas definidas por los campos P02, P03 y P04. En este caso SNAP generará el programa para que solamente utilice las pantallas que contengan campos de despliegue, es decir, si para todos los campos seleccionados se indica "N" en alguna de estas pantallas, ésta no se utilizará en el programa generado.

Generación

Este programa de consulta simple genera:

- a) Entradas a la estructura de datos con la entidad primaria del programa y cada una de las entidades a las que se haga referencia en alguno de sus campos, y las entidades serán de tipo L (Lectura). Además incluirá las entidades que se encuentren definidas en el programa de estándares.
- b) Variables de trabajo para la manipulación de datos y las definidas en el programa de estándares.
- c) Macros para las sustituciones dentro del archivo fuente SNAP, sólo si el SNAP está configurado con esta opción en los parámetros de generación de la herramienta.
- d) Un archivo fuente de pantallas con cuatro registros: las cuatro pantallas definidas en el programa.
- e) Los módulos de ejecución del programa, además de los módulos definidos en el programa de estándares definido en el sistema.

Una vez generado y cuando es invocado el programa, este presenta la siguiente pantalla de consulta:

SNAP Release 3.0 Derechos Reservados SOH C. R.		BOI/1988	18 30 42
Sistema Control de Proyectos		COPO037	PANT01
Programa Consulta Clientes		Consulta	
.....			
Cliente:	12		
Nombre:	Fondo de Cultura Económica		
Dirección ..:	Calle 4 y Santa Teresita # 48		
Teléfono ..:	4 48 26 22		
Encargado ..:	Jesus Ramirez Contreras		
.....			
F12 - Aceptar			

4.5.5 Mantenimiento Simple

Concepto

Por medio de un programa de mantenimiento simple el usuario podrá crear, modificar o eliminar registros de la entidad a la que se asocia el programa, para este ejemplo se tomará la entidad de CLIENTES. Este programa además, permite utilizar atributos de otras entidades relacionadas a ésta por una dependencia o referencia como campos de salida.

El mantenimiento de los registros de la entidad se realiza con un registro a la vez. Para dar mantenimiento a un registro se debe especificar el valor o los valores de los campos llave de éste; si existe se podrá modificar los atributos no llave o eliminar registro, en caso contrario permite crear el registro.

Pasos Previos

Se debe asegurar que los parámetros de generación para el tipo de programa sean los correctos, estos pueden ser modificados en los parámetros automáticos en utilitarios SNAP.

El programa de estándares para la generación de programas automáticos deberá ser indicado en el sistema en el cuál residirá el programa de mantenimiento simple. La importancia de este programa radica en que según lo definido en él, se establecerá el formato de pantallas y el módulo de inicialización del programa, así como las variables necesarias para su correcta generación.

Para aprovechar al máximo las características del programa de mantenimiento simple, se sugiere revisar los siguientes puntos:

- En caso que la entidad dependa o se refiera a otras entidades, es conveniente que los programas automáticos CON LIS de éstas entidades ya hayan sido generados, para que el programa de mantenimiento simple los ligue a los campos correspondientes en las pantallas para su invocación por medio del mandato F4. En caso contrario, el mandato F4 no tendrá funcionalidad.
- En caso que los atributos de la entidad a la cuál se define el programa tengan lista de valores, es necesario que estas listas de valores estén previamente definidas con al menos un valor para que el programa asocie la validación del caso en forma automática. En caso contrario, no se contará con el mandato F4 para desplegar ésta lista.
- En caso de que existan entidades que sean dependientes o referentes a la entidad a la cuál se define el programa, y se desee mantener la integridad de los datos del sistema, se debe definir previamente el programa ELI asociado a esta entidad de CLIENTES, lo cuál ya se realizó. En caso de no definirse este programa, el usuario podrá eliminar registros sin que exista la verificación de integridad referencial en el sistema.

Uso del Programa

Este programa debe utilizarse en aquellos casos donde se presente alguna de las siguientes circunstancias:

- Cuando se requiere un programa de mantenimiento para una entidad que pueda ser invocado desde un menú o una línea de comandos, pues en la generación de este sub-tipo de programa puede especificarse el uso de las llaves como parámetro.
- El mantenimiento de una entidad requiere de una acción única para cada uno de los registros ya que las características del programa los define para el mantenimiento de un registro a la vez. Este es nuestro caso para la realización de este tipo de programa y satisfacer nuestro requerimiento.
- Cuando se necesite un programa de mantenimiento de rápido acceso, para una entidad cuyo volumen de registros hace obsoleto el uso de listas.

Generación

El programa de mantenimiento simple genera:

- a) Entradas a las estructuras de datos con la entidad primaria del programa y en cada una de las entidades a las que se haga referencia en alguno de sus campos, la entidad base tendrá el tipo "A" (Actualización) y las otras "L" (Lectura), además de las entidades que se encuentren definidas en el programa de estándares.
- b) Variables de trabajo para la manipulación de datos y las definidas en el programa de estándares.
- c) Macros para sustituciones dentro del archivo fuente del SNAP, sólo si el SNAP está configurado con esta opción en los parámetros de generación de la herramienta.
- d) Un archivo fuente de pantallas con cinco registros: las cuatro pantallas definidas en programa, como se mencionó previamente, y otra pantalla de confirmación de la eliminación tendrá la misma secuencia de campos definida para la segunda pantalla del programa.
- e) Los módulos de ejecución del programa: la estructura modular del programa así como los módulos que pertenecen al programa de estándares.

Definición

Una vez generado el programa de mantenimiento simple se debe definir sus características por medio de las especificaciones de alto nivel, para ello se utiliza la opción F12=Trabajar. Para definir las especificaciones de alto nivel, se despliega la siguiente pantalla.

SNAP Roles 3.0 Derechos Reservados SOH C. R.				BDI/1996			19 30 42	
Sistema Central de Procesos								
Programa Mantenimiento Simple a Clientes								

Entidad : CLIENTES								

4 = Tiempo				ACCES			DATOS	
Seq	SEC	Entidad/Atributo	COD	PO1	PO2	PO3	Descripción	Len
-	10	CLIENTES						
		Código Cliente	AID	A	S	N	Cliente	1
-	20	CLIENTES						
		Nombre del Cliente	ATR	N	A	N	Nombre	1
-	30	CLIENTES						
		Dirección	ATR	N	A	N	Dirección	1
-	40	CLIENTES						
		Teléfono	ATR	N	A	N	Teléfono	1
-	50	CLIENTES						
		Encargado sistema	ATR	N	A	N	Encargado	1
F6 = Agregar		F12 = Anular						

Para seleccionar los atributos que serán presentados durante el tiempo de ejecución del programa se usa la tecla de función F6=Agregar. En este punto se podrán seleccionar atributos de la propia entidad como también atributos de entidades relacionadas. Los atributos que se seleccionen de una entidad relacionada únicamente podrán ser desplegados.

Una vez seleccionado los atributos debe definirse los siguientes aspectos:

SEC: Indica la secuencia de aparición del campo en las pantallas generadas por el programa, donde el mismo haya sido definido de "Entrada/Salida" o "Salida".

PO1: Esta columna corresponde a la pantalla de acceso al programa. En este campo se definen las características de despliegue de los campos llave de la entidad en el primer pantalla del programa; en esta pantalla también se pueden definir atributos de la entidad o entidades relacionadas, pero en este caso el atributo solo puede ser de "Salida".

Los valores posibles para este campo son:

S: Indica que el campo es de "Salida"

N: Indica que el campo no aparece en pantalla

A: Indica que el campo es de "Entrada/Salida".

Si el valor es "A" para los campos llave, SNAP generará la lógica necesaria para capturar la llave antes de desplegar el contenido del registro, pero si se indica "S" o "N" para estos, se incluirá el valor de este campo como parámetro del programa.

PO2: Indica la característica de despliegue del campo en la segunda pantalla generada por el programa.

Los valores posibles de este campo son:

S: Indica que el campo es de "Salida"

N: Indica que el campo no aparece en pantalla

A: Indica que el campo es de "Entrada/Salida".

P03: Este campo cumple la misma función que el campo anterior pero para la tercera pantalla generada.

Descripción: En este campo se captura el texto asociado a cada campo en las pantallas donde se definió con un atributo de despliegue, esta descripción generará una constante en las pantallas donde aparezca el campo justo antes del despliegue de éste a partir de las posiciones 1 o 40 de la línea donde se encuentre.

El valor de este campo es inicializado automáticamente por SNAP a partir de la descripción dada a cada atributo del modelo de datos.

En el caso que este campo se deje en blanco SNAP no asignará una constante antecedente a la variable.

Lin: Indica el número de líneas que separan a un atributo del anterior desplegado en la pantalla.

El valor por omisión es cero, en este caso SNAP acomodará los atributos en las pantallas donde se halla definido el campo automáticamente. Esto es acomodado tantos campos por línea como su longitud lo permita separando cada uno de ellos con un espacio en blanco, cuando estos campos tengan una descripción asociada estas descripciones empezarán en las columnas 1 a 40 de la pantalla.

En muchos casos no será necesario utilizar las últimas tres pantallas definidas por los campos P02 y P03: en este caso SNAP generará el programa para que solamente utilice las pantallas que contengan campos de despliegue ("S" o "A").

Cuando sean utilizadas más de una de las últimas tres pantallas el programa generado por SNAP no permitirá la grabación del registro hasta que todas ellas sean desplegadas, con esto se garantiza la integridad de la base de datos al obligar a la validación de los campos seleccionados en pantallas donde cada uno de ellos aparezca.

Si un mismo campo es definido con el atributo "A" en más de una pantalla el valor que contendrá el registro será el valor del campo de despliegue en la última pantalla desplegada por el programa antes de grabar los datos.

Generación

El programa de mantenimiento simple genera:

Entradas en la estructura de datos con la entidad primaria del programa y cada una de las entidades a las que se haga referencia en alguno de sus campos, la entidad base tendrá el tipo "A" (Actualización) y las otras "L" (Lectura). Además de las entidades que se encuentran definidas en el programa de estándares.

Variables de trabajo para la manipulación de datos y las definidas en el programa de estándares.

Macros para las sustituciones dentro del fuente SNAP, sólo si el SNAP esta configurado con esta opción en los parámetros de generación de la herramienta.

Un fuente de pantallas con cinco registros: las cuatro pantallas definidas en el programa, como se mencionó previamente, y otra pantalla de confirmación de la eliminación que se desplegará cuando

está acción sea tomada por el usuario. La pantalla de confirmación de la eliminación cual contendrá la secuencia de campos definida para la segunda pantalla del programa.

Los módulos de ejecución del programa, la estructura modular del programa incluyendo los módulos generados en el programa de estándares definido en el sistema.

El programa una vez generado y evocado entra directamente a la siguiente pantalla:

SNAP Release 3.0 Derechos Reservados S/IN C. R.	06/1996	19 30 42
Sistema Control de Pluicias	CDPO08	PANT01
Programa Mantenimiento a Clientes		Combo

Cliente	12 Fondo de Cultura Económica	
Nombre	Fondo de Cultura Económica	
Dirección ..	Ciudad de Santa Teresita # 85	
Encargado:	Isabel Fabroca Pérez	
F12 = Anterior		

4.5.6 Mantenimiento Alta, Baja, Cambio de Personal Asignado a Clientes

Concepto

El programa de mantenimiento MAN LIN (mantenimiento con líneas de detalle efectúa modificaciones, inclusiones o borrados en la entidad donde esté asociado y otra entidad depende de esta, para nuestro caso este programa se dará de alta en la entidad de PERSONAL/CLIENTES ya que está tiene una dependencia que es la entidad CLIENTES. El concepto de mantenimiento con líneas de detalle está directamente relacionado con el concepto de dependencia.

Un programa de mantenimiento simple con líneas de detalle se divide en dos partes: la primera es el mantenimiento de los registros de la entidad dependiente de ella donde se relacione el valor del campo llave. El mantenimiento de los registros dependientes se realiza por medio de una línea de detalle en un sub-archivo generado.

Pasos Previos

Se debe asegurar que los parámetros para el tipo de programa sean los correctos, éstos pueden ser modificados en los programas automáticos en los utilitarios SNAP.

FALLA DE ORIGEN

El programa de estándares para la generación deberá ser el indicado para el sistema donde radicaré el programa de mantenimiento simple con líneas de detalle. La importancia radica en que según lo definido en éste, se establecerá el formato de pantallas y el módulo de inicialización del programa, así como las variables necesarias para su correcta generación.

Para aprovechar al máximo las características del programa de mantenimiento simple con líneas de detalle, se sugiere revisar los siguientes puntos:

- En caso que la entidad dependa o se refiera a otras entidades es conveniente que los programas automáticos de consulta de lista de valores de estas entidades ya hayan sido generados, para que el programa de mantenimiento simple con líneas de detalle los asocie a los campos correspondientes en las pantallas para su invocación por medio del mandato F4. En caso contrario el mandato F4 no tendrá funcionalidad.
- En caso de que los campos de la entidad a la cuál se define el programa tengan lista de valores, es necesario que estas listas de valores estén previamente definidas con al menos un valor para que el programa lo ligue automáticamente. En caso contrario no se contará con la función de la tecla F4 para desplegar esta lista.
- Los programas de verificación de eliminación de las entidades dependientes o referentes a la entidad base, así como el programa de verificación de eliminación asociado a ésta, deben definirse previamente al programa de mantenimiento simple con líneas de detalle. En caso de no definirse dicho programa, el usuario podrá eliminar registros sin que exista la verificación de integridad referencial en el sistema y no se verificará la consistencia de la base de datos al borrar.

Uso del Programa

Este programa deberá utilizarse en aquellos casos donde se presenten algunas de las siguientes circunstancias:

- Cuando se requiere de un programa de mantenimiento para una entidad y una entidad dependiente de ella, que pueda ser invocado desde un menú o línea de mandato, pues en la generación de este sub-tipo de programa puede especificarse el uso o no de valores de sus llaves como parámetro.
- El mantenimiento de una entidad requiere de una acción única para cada uno de sus registros, y para cada uno de ellos se definen inmediatamente los valores de los registros dependientes.
- El uso del programa no requiere previamente, conocer la acción a seguir en el mantenimiento de un registro del programa. En otras palabras mientras se ejecuta éste pueden utilizarse aleatoriamente inserciones, borrados o modificaciones sobre registros de la entidad y sus registros dependientes caso por el cuál se va a utilizar este tipo de programa para el mantenimiento de personal para cada cliente.
- Se necesita un programa de mantenimiento de rápido acceso, para una entidad cuyo volumen de registros hacen obsoleto el uso de listas.

Definición

Para la definición del programa de mantenimiento simple con línea de detalle se debe seguir una serie de pasos, los cuales en términos generales especifican que información debe estar presente en la pantalla, en su encabezado y en su detalle.

Lo anterior se especifica utilizando la opción 12=Trabajar de la pantalla de mantenimiento de programas. Luego de utilizar esta opción sobre el programa de mantenimiento simple con líneas de detalle se despliega la pantalla mostrada en la siguiente figura.

SNAP Petate 3.0 Derechos Reservados SOH C. R. 8/01/1995 19 30 42

Sistema Control de Proyectos
Programa Mantenimiento Personal a Clientes

Entidad PERSONAL/CLIENTES

4 = Eliminar

Sel	SEC	Entidad/Atributo	ACCES				U S O				Lin
			CGO	Dir	PD1	ENC	DEF	Descripción			
-	10	PERSONAL/CLIENTES									
		Código Cliente	AID	E	A	S	N		Cliente		
-	20	CLIENTES									
		Nombre Cliente	ATR	E	S	S	N		Nombre		
-	30	PERSONAL									
		Clave Personal	AID	L	N	N	A		Clave		
-	40	PERSONAL									
		Nombre	ATR	L	N	N	S		Nombre		

F8 = Agregar, F12 = Anterior

Con el mandato F6=Agregar en esta pantalla se encuentran las opciones para seleccionar los campos que formarán parte en la pantalla del programa. Para cada campo seleccionado, se especifican las características que tendrá el programa generado, como su ubicación en el despliegue o valor de posicionamiento en la ejecución. La pantalla para especificar atributos de este tipo de programa se muestra a continuación.

SNAP Petate 3.0 Derechos Reservados SOH C. R. 8/01/1995 19 30 42

Sistema Control de Proyectos
Programa Mantenimiento Personal a Clientes

Entidad PERSONAL/CLIENTES Destino PERSONAL

1 = Selección

Sel	Entidad	CGO	Atributo	Dir
1	PERSONAL/CLIENTES	AID	Código Cliente	E
1	CLIENTES	ATR	Nombre Cliente	E
1	PERSONAL	AID	Clave Personal	L
2	PERSONAL	ATR	Nombre	L
	PERSONAL	ATR	Base	L

F12 = Anterior

Los atributos se seleccionan con la posición 1=Selección, luego de seleccionar los atributos deseados, estos se desplegarán en la pantalla anterior. Para cada atributo seleccionado se definen las siguientes características:

FALLA DE ORIGEN

SEC: Indica la secuencia de aparición del campo en la pantalla generada por el programa, este campo variará su significado según la posición donde se defina el campo en la pantalla utilizando los campos USO ENC y USO DET, así se define:

- Como el orden de aparición en el encabezado de la pantalla cuando el valor de USO ENC sea "S".
- como la posición en la línea del sub-archivo que desplegará la lista de los registros de la entidad cuando el valor de USO DET sea "S" o "A".

P01 Esta columna corresponde a la pantalla de acceso al programa. En este campo se definen las características de despliegue de los campos llave del registro de la entidad en el encabezado de la pantalla del programa y en la pantalla inicial donde se capturan los valores de la llave. En esta pantalla inicial donde se capturan los valores de la llave. En esta pantalla también se pueden definir atributos de la entidad o entidades relacionadas, pero en este caso el atributo solo puede ser de "Salida".

Los valores posibles para este campo son:

S: Indica que el campo es de "Salida"

N: Indica que el campo no pertenece en esta pantalla.

A: Indica que el campo es de "Entrada/Salida".

Si el valor es "A" para los campos llave, SNAP generará la lógica necesaria para capturar la llave antes de desplegar el contenido del registro, pero si se indica "S" o "N" para estos, se incluirá el valor de este campo como parámetro del programa.

ORI: Este campo indica el origen del campo seleccionado, donde se desplegará el campo: en el encabezado o el detalle. Los valores posibles para el campo son los siguientes:

E: Indica que el campo seleccionado se desplegará en el encabezado de la pantalla. Un valor "E" es inicializado cuando el campo pertenece a la entidad base.

L: Indica que el campo seleccionado se desplegará en la línea de detalle. El valor por omisión será "L" cuando el campo sea un campo de la entidad dependiente o uno en una entidad dependiente o uno en una entidad de referencia.

Este campo es inicializado por SNAP dependiendo del origen del campo seleccionado, sin embargo puede ser modificado. Los campos que se presentan en el encabezado actúan como filtro de los registros que se muestran en el detalle, siempre y cuando la estructura de los campos llave de la entidad permita implantarlo, es decir los campos llaves desplegados en el encabezado deben preceder a los definidos en la línea de detalle.

USO ENC: Este campo es utilizado para definir la característica que tendrá el atributo seleccionado en el encabezado de la pantalla. El encabezado constituye también la pantalla de acceso con la misma distribución de campos. Los posibles valores para este campo son:

N: El atributo no se muestra en la parte del encabezado de la pantalla.

S: El atributo se desplegará en la parte del encabezado de la pantalla, como un campo de salida únicamente.

USO DET: Este campo es utilizado para definir las características del campo seleccionado en la línea del sub-archivo. Los valores posibles para este campo son los siguientes:

N: El atributo no se muestra en la línea de detalle del sub-archivo generado.

A: El atributo se despliega en el detalle del programa como un campo de actualización.

Los valores de este campo son inicializados por SNAP y no pueden ser modificados.

Descripción: En este campo se captura el texto asociado a cada campo en el encabezado de la pantalla donde se definió con un atributo de despliegue. Esta descripción generará una constante en la pantalla justo antes del despliegue del campo a partir de la columna 1 a 40 de la línea de donde se ubique el éste.

El valor de este campo es inicializado únicamente por SNAP a partir del nombre dado a cada atributo en el Modelo de Datos.

En el caso que este campo se deje en blanco SNAP no asignará una constante antecedente a la variable.

Lín: Indica el número de líneas que separan a un campo del anterior desplegado en el encabezado de la pantalla.

El valor por omisión es cero, en este caso SNAP acomodará los campos en las pantallas donde se halla definido el campo automáticamente, esto es acomodado tantos campos por línea como su longitud lo permita separando cada uno de ellos con un espacio en blanco. Cuando estos campos tengan una descripción asociada ésta se desplegará a partir de la columna 1 a 40 de la pantalla si hay espacio disponible.

Generación

El programa mantenimiento simple con líneas de detalle genera:

a) Entradas en la estructura de datos con la entidad primaria del programa con el tipo de acceso "A", cada una de las entidades a las que se haga referencia en algunos de sus campos, las entidades serán tipo "L", la entidad dependiente que se mantendrá en el detalle con acceso "A" y cada una de las entidades que se relacionen con ésta con acceso "L". Además incluirá las entidades que se encuentren definidas en el programa de estándares.

b) Variables de trabajo para la manipulación de datos y las definidas en el programa de estándares

c) Macros para las sustituciones dentro del archivo fuente del SNAP, sólo si el SNAP está configurado con esta opción en los parámetros de generación de esta herramienta.

d) Un fuente de pantallas con registros divididos en: pantalla de acceso. La pantalla del encabezado y una pantalla para mostrar el registro y sus dependencias por borrar.

e) Los módulos de ejecución del programa modular del programa. Los módulos generados incluyen los módulos definidos en el programa de estándares definido para el sistema.

Una vez generado el programa y al ser invocado desde donde se selecciono un cliente determinado se muestra la siguiente pantalla en modo de cambio:

SIAP Paises 3 O Derechos Reservados SGM C B		6/01/1985	18 30 42
Sistema Control de Proyectos		COPO08	PAN102
Programa mantenimiento de Personal a Clientes		Cambio	

Cliente 12 Fondo de Cultura Económica			
4 = Eliminar, 8 = Consultar			
Sal	Código	Nombre del personal	
-	001	Jesús Romero Contreras	
-	002	Isabel Fátima Romero	
-	003	María Siles Pacheco	
-	004	Sergio Vargas Sanchez	
-			
-			
-			
-			
F8 = Agroup, F10 = Contar, F12 = Anterior			

4.5.7 Programa Listado

Concepto

Un reporte es un conjunto de componentes o elementos que cumplen un fin determinado en un momento específico. Cada componente está compuesto por una o más líneas de impresión, dependiendo de la entidad de información que se desea imprimir en un momento dado. A su vez cada línea de impresión está compuesta por campos de salida que pueden ser tanto:

Constantes

Variables

Atributos de la entidad en proceso

Atributos de otras entidades relacionadas.

Campos de resultado definidos por el programador.

Cada vez que se imprime un componente se realizarán automáticamente las operaciones de lectura y de cálculo necesarias para obtener la información que se imprimirá en la o las líneas del componente.

Tipos de componentes

Encabezado: El objetivo de este componente es el de imprimir el encabezado del reporte. Se imprimirá con cada cambio o salto de página y automáticamente, la variable PAGE (número de página) se incrementará en 1. Las características de este componente pueden ser inicializadas automáticamente o modificadas por el programador.

Corte N(Cn): Se genera un componente de corte por cada campo llave excepto el último, hasta un máximo de ocho niveles o cortes de control.

Detalle: Este componente se inicializa automáticamente con los atributos de la entidad listada, se podrán incluir o eliminar campos, constantes o de resultado.

Total N(Tn): Este componente actúa junto con sus respectivo componente Corte N y su objetivo es imprimir los totales respectivos.

TOTAL: El objetivo de este componente es imprimir los totales generales del reporte.

Fin Reporte: El objetivo de este componente es imprimir el mensaje de fin de reporte, sus características pueden ser inicializadas automáticamente o modificadas por el programador.

Pasos Previos

Se debe asegurar que los parámetros de generación para el tipo de programa sean los correctos, estos pueden ser modificados en los parámetros de instalación y a la definición del programa de estándares para reportes .

El programa de estándares para la generación de programas automáticos deberá ser el indicado para el sistema donde residirá el programa de Selección por Navegación, la importancia de este programa radica en que, según lo definido en éste, se establecerá el formato de pantallas y el módulo de inicialización del programa, así como las variables necesarias para su correcta generación.

Para aprovechar al máximo las facilidades del programa de Listado, se sugiere revisar los siguientes puntos:

- El modelo de datos debe estar totalmente terminado, es decir todas las relaciones de la entidad base deben estar definidas.
- Si se utilizarán programas de cálculo de resultados para su uso dentro de la estructura del reporte es conveniente que éstos estén definidos.
- Se deben definir las vías de acceso en el caso que el programa se defina sobre el acceso determinado por una de ellas, esto debe ser definido antes de la definición del reporte pues de escogerse una de estas vías los cortes de control y la estructura del reporte dependerá del acceso que se defina en ella.

Uso del Programa

La utilización de un programa de Listado debe darse cuando se encuentre en al menos una de las siguientes situaciones:

- La entidad a la que se asociará requiere de la salida impresa de parte de su información.
- Se requiere establecer un esquema de cálculos sobre la información contenida en una o varias entidades. Cuando una estructura de dependencias sea listada el programa automático de listado debe ser asociado a la última entidad de la estructura.

Definición

Para incluir un programa de Listado debe escogerse el tipo LIS. Una vez especificado el tipo se desplegará la pantalla de características, mostrada en la siguiente figura:

SNAP Release 3.0 Derechos Reservados SOM S. R.		8/01/1988	19 30 43
Sistema Control de Proyectos			
Listado Personal asignado a clientes			

Entidad	PERSONAL/CLIENTES	Personal Asignado a clientes	

Form	LIS		
Subsistema	LIS		
Subprograma	CDP		
Programa	COFOORO		

Descripción	Listado Personal Asignado a Clientes		

Propietario	SOM		
Nivel de Seg.	5		
Via de Acceso	XX = Temporal definido		
Long de Imp.	80, 132, 198		

F12 = Anterior.			

Además de los campos que se especifican para cualquier otro programa automático, deben especificarse dos campos adicionales:

Vía de acceso: En este espacio se indica la vía de acceso que se utilizará para definir el listado. En caso de que desee utilizar la entidad física y no una vía de acceso, se debe indicar con "00". También podrá definirse una vía de acceso temporal (No definida en el modelo de datos), exclusiva para este programa, para ello seleccione la vía de acceso XX.

Precaución: Si se selecciona la vía de acceso XX siempre invocarse el programa a través del módulo OPNQRYF generado automáticamente, para más información consultar el manual de programas automáticos del SNAP.

Longitud de impresión: En este campo se deberá indicar la longitud de la línea de impresión del reporte ya sea esta (80, 132, 198), y de acuerdo a esto se escoge el formato impresión del programa base que corresponda.

Una vez definido el programa se utiliza la opción 12=Trabajar de la pantalla de mantenimiento de programas automáticos. La primera vez que se ejecuta esta opción el SNAP generará la estructura del reporte de la siguiente forma:

FALLA DE ORIGEN

SNAP Report 3.0 Derechos Reservados SOW C. R. 6/01/1985 18 30 42
 Sistema Control de Proyectos CDPQ080
 Manejamiento de Programa de Reportes

ENTIDAD PERSONAL/CLIENTES Personal que actúa con clientes

12 = Trabaja

Sel	Componente	Tp	Descripción
--	ENCABEZADO	E1	
--	COORTE 1	C1	Cliente
--	COORTE 2	C2	Personal
--	DETALLE	D1	
--	TOTAL 2	T2	
--	TOTAL 1	T1	
--	TOTALES		
--	FIN REPORTE		

F12 = Anterior, F13 = Diseño

En esta pantalla se tiene la opción de visualizar la salida diseñada por el reporte utilizado el mandato F13=Diseño. En esta pantalla aparecen los componentes del reporte, los cuales pueden ser eliminados o modificados a través de la opción 12=Trabajar. Los campos que se despliegan en esta pantalla son:

Componente: Campo de salida que indica el nombre del componente del reporte que se especifica.

Tp: Este campo de salida indica el tipo o nivel del componente, este código de dos caracteres será utilizados como parte del nombre del formato de impresión en el programa generado.

Descripción: Campo de salida que contiene la descripción o atributo que genera un corte de control.

Cada componente del reporte está compuesto por una o varias líneas de impresión, como se muestra a continuación:

SNAP Report 3.0 Derechos Reservados SOW C. R. 6/01/1985 18 30 42
 Sistema Control de Proyectos CDPQ080
 Manejamiento de Programa de Reportes

ENTIDAD PERSONAL/CLIENTES Personal que actúa con clientes

Componente DETALLE

1 = Campos, 4 = Líneas, 12 = Trabaja

Sel	N1	Descripción	Detalle	Sel	N2	Descripción	Detalle
--	10	Ases		--	10	Ases	
--	20			--	20		

F6 = Agragar Nueva Línea F12 = Anterior

FALLA DE ORIGEN

Cada línea de impresión mostrada puede tener los siguientes parámetros:

NL: Identifica una secuencia de línea de impresión dentro de un componente, su rango válido varía entre 01 y 99

Brincar líneas: Se especifican las líneas que se deben dejar nulas cada vez que se imprima el formato de impresión correspondiente al componente.

Antes: Número de líneas que se deben de brincar antes de imprimir la línea, el valor puede ser 1, 2 o 3. También puede ser dejado en blanco.

Después: Número de líneas que se deberán brincar después de haber impreso la línea, sus valores posibles son 1, 2, 3 o blanco.

Saltar a Línea: se especifica el número de línea dentro de la página que se imprime actualmente. Si esta línea ya ha sido alcanzada por el contador de líneas se incrementará hasta la línea correspondiente en la siguiente página.

Antes Número de línea en que se imprimirá, generalmente se utiliza con el valor 01 para la primera línea de impresión del componente ENCABEZADO. Los componentes para este campo van del 01 al 66 o blanco.

Después: Número de líneas en la que debe quedar posicionada la impresora para la siguiente línea, los valores posibles van del 01 al 66.

Las opciones para manejar las líneas definidas para un componente son las siguientes:

F4=Eliminar Elimina una línea del componente

F6=Agregar Agregar una nueva línea

F12=Trabajar Para cada una de las líneas de impresión existentes en un componente puede seleccionarse con la opción F12=Trabajar y definir los campos o constantes que la componen.

Una línea de impresión está compuesta por campos de salida, que pueden ser constantes o variables, ya sean atributos de entidades o campos de resultado calculados en el programa, como muestra a continuación.

SINAP Petrol 3.0 Derechos Reservados SINAP C. R.				001/1995		18 30 42	
Sistema Control de Proyectos				COROORO			
Mantenimiento de Programa de Noticias							
ENTIDAD PERSONAL CLIENTES				Personal que actúa con clientes			
Componente				DETALLE		Línea 10	
Sec	Exp	Im	I	Campo	Constante	Jul 74	
10	B	3	N		Puesto		
20	20	23	A	PIRPLU	Puesto		
F8 = Agrega Nueva Línea. F12 = Anterior							

FALLA DE ORIGEN

Cada campo tiene especificado los siguientes aspectos:

Sec: Define el orden en que se definen los campos en la línea.

Esp: define la separación o número de espacios de un campo con respecto al anterior.

Fin: Este campo es de salida e indica la última posición ocupada por el campo en la línea de impresión. Utilizando el mandato F9 automáticamente calculará la posición final del campo, esto en el caso de que este campo se necesite como un campo de referencia para posesionar un campo en algún otro componente, por ejemplo, verificar si el total de un campo está alineado con la salida de los montos de detalle.

T: Tipo de campo o elemento definido en la línea de impresión según su naturaleza. Los posibles valores para este campo son:

C: Constante

A: Atributo de la entidad o campo de resultado definido en otro nivel o componente.

R: campo de resultado definido en este nivel o componente.

E: Código de edición escondido, el campo se utiliza en el componente pero no se imprime ni se toma en cuenta en la formación de la línea de impresión, se utiliza solamente cuando el tipo de dato es numérico.

X: Código de edición que se utiliza únicamente para los campos numéricos, se debe sustituir X por un código de edición definido.

Atrib/Constante; Representa ya sea, el texto de una constante si el tipo es "C", o el nombre del atributo si el tipo de campo es "A".

Jus.Tit: Indica si el texto del campo de Atrib/Constante será o no utilizado como título, hacia que lado se justificará en caso de que el título sea menor que la longitud sea menor que la longitud del campo de salida. Los valores posibles para este campo son:

I: Ajusta descripción a la izquierda

D: Ajusta descripción a la derecha

*Blanco No define la descripción.

Este campo únicamente es válido en el componente de DETALLE, TOTAL y TOTALES.

En el componente de DETALLE los títulos de los campos de salida generarán los formatos de impresión FMT@H0 y FMT@H1 y se imprimirán inmediatamente después del encabezado o después del último título de corte de control, dependiendo de como se haya especificado en los parámetros de generación para este tipo de programa. En la segunda línea de detalle se hará lo mismo pero se incluirá el campo constante "Última evaluación de el personal ." y en seguida el campo de entidad "PEREVA"

En los componentes de TOTAL y TOTALES los últimos se imprimirán encima de la variable de salida, su uso más práctico es el de colocar una raya sobre los campos de totales para dicho efecto se puede indicar como título un carácter, por ejemplo @= o @_ que colocarán una raya de longitud del campo con el carácter "=" sobre el campo total.

Para agregar nuevos campos se utiliza el mandato F6=Agregar. Al utilizar este mandato se presentará la siguiente pantalla:

SNAP Release 3.0 Derechos Reservados SDIN C. R.		0/01/1998	19 30 42
Sistema Control de Proyectos		COP0080	
Mantenimiento de Programa de Reportes			
.....			
ENTIDAD	PERSONAL/CLIENTES	Personal que actúa con clientes	
.....			
Componente	CORTE1	No. Línea 10	
.....			
1 = Seleccionar			
Sal	Accede		
	Seleccionar un Campo		
	Seleccionar una Entidad Esterna		
	Definir un Campo Constante		
	Definir un Campo de Resultado		
F12 = Aceptar			

Seleccionar un campo: Se utiliza para agregar atributos de la entidades seleccionadas o campos de resultado predefinidos en otros niveles o componentes.

Al seleccionar esta opción se despliega la pantalla mostrada en la siguiente figura:

SNAP Release 3.0 Derechos Reservados SDIN C. R.		0/01/1998	19 30 42
Sistema Control de Proyectos		COP0080	
Mantenimiento de Programa de Reportes			
.....			
ENTIDAD	PERSONAL/CLIENTES	Personal que actúa con clientes	
.....			
Componente	CORTE1	No. Línea 10	
.....			
Agregando Atributo/Campo			
Campo	CINDIA		
F4 = Lista, F12 = Aceptar			

Debe digitarse el nombre del campo a incluir, este puede ser un atributo o un campo de resultado definido en otro nivel. Se puede hacer uso del mandato F4=Lista el cual permite visualizar la lista de todos los campos disponibles y seleccionar el deseado.

Una vez seleccionado el campo, se despliega la siguiente pantalla.

SNAP Paises 2.0 Derechos Reservados SONY C R		8/01/1998	18 30 47
Sistema Control de Proyectos		COPO080	
Mantenimiento de Programa de Reportes			

ENTIDAD	PERSONAL/CLIENTES	Personal que Actua con Cliente	

Componente	CURTEL	No. Linea : 10	

Agregando Atributo/Campo			
Campo	CLINOM	Nombre del Campo	
Secuencia	00		
Espacios A	3		
Código de Ed.			

F4 = Lista, F12 = Anterior			

En la pantalla anterior se debe especificarse:

Secuencia: Indica el orden en que se imprimirá el campo en la línea del componente, sobre el que se ha trabajado.

Espacios: Indica el número de espacios que separarán a este campo con el campo anterior de la línea.

Código de edición: Código de edición que tendrá el campo únicamente es valido en campos numéricos.

Estos valores pueden ser modificados luego desde la línea de impresión donde se definen.

Seleccionar una Entidad. Se utiliza para seleccionar una entidad externa, como resultado se cuenta con los atributos de esta nueva entidad y una operación de lectura se realizará cada vez que se ejecute este componente. Se podrá ligar esta nueva entidad siempre y cuando existan atributos de referencia.

Luego de seleccionar esta opción se despliega una pantalla que lista las entidades activas en este nivel, mostrada seguidamente.

FALLA DE ORIGEN

SNAP Proceso 20 Derechos Reservados SON C. R. 6/01/1995 19 30 42
 Sistema Control de Proyectos CPO0000
 Mantenimiento de Programa de Reportes

ENTIDAD PERSONAL/CLIENTES Personal que Actua con Clientes

Componente CORTE1 No. Lineas 10

ENTIDADES ACTIVAS EN ESTE NIVEL
 4 = Eliminar, 12 = Trabajar

Sal	Entidad
---	Personal de la Empresa
---	Personal Asignado a Clientes
---	Clientes de la Empresa

F4 = Lista, F12 = Anterior

Para cada una de las Entidades de la lista existen dos posible opciones:

- 12=Trabajar Se define el acceso para la entidad
- 4=Eliminar Utiliza para eliminar la Entidad de la lista
- F8=Agregar Entidad Se utiliza este mandato para agregar una nueva entidad a la lista y se desplegará la siguiente pantalla.

SNAP Proceso 20 Derechos Reservados SON C. R. 6/01/1995 19 30 42
 Sistema Control de Proyectos CPO0000
 Mantenimiento de Programa de Reportes

ENTIDAD PERSONAL/CLIENTES Personal Asignado a Clientes

Componente CORTE1 No. Lineas 10

Agregando Entidad Externa

Entidad Personal de la Empresa

F8 = Lista, F12 = Anterior

FALLA DE ORIGEN

En esta pantalla se debe digitar el nombre de la entidad que se desea ligar, naturalmente deberá existir relación entre la entidad sobre la que se está generando el reporte y la que se está agregando.

Por medio del mandato F4=Lista se obtiene la lista de las entidades definidas en la base de datos y seleccionar la deseada.

Una vez seleccionada, se despliega una pantalla con los campos necesarios para accederla, con su tipo y longitud (Ver siguiente figura).

```
SAAP Pagina 30 Derechos Reservados SOIN C R                               8.07/1995       19 30 42
Sistema Control de Proyectos                                           COPO000
Mantenimiento de Programa de Reportes
-----
ENTIDAD ... PERSONALCLIENTES                                           Personal Agregado a Clientes
-----
Complemento ... : CDRT01                                               No Lines ... : 10
-----
Agregando Entidad Externa
-----
Entidad ... : Personal de la Empresa
Opciones ... : C=Consulta
-----CAMPOS LLAVE ENTIDAD EXTERNA-----
01  Código Cliente           CUCLA           J N           -----CAMPOS INTERNOS-----
PERCLA
F12 = Anterior, F23 = Eliminar Entidad Externa
```

En la pantalla anterior se debe especificar:

Campos Internos: Se indican los campos internos con los que se hará el acceso. No se podrá continuar si hay coincidencia de los campos internos con respecto a los campos llave de la entidad.

C=Consulta: Digitado una C antes del campo solicitado se obtendrá una lista con los atributos y los campos definidos para el programa, permitiendo seleccionar el deseado.

F23=Eliminar: Con este mandato se eliminará la Entidad Externa de la definición

Definir un campo Constante: Se utiliza para definir un campo constante, su longitud puede ser hasta de 40 caracteres.

Una vez seleccionada esta opción se despliega la pantalla de definición de campos constantes mostrada enseguida:

SNAP Paseo 3 D Derechos Reservados S.O.H.C.R.		6/01/1985	19 30 42
Sistema Control de Proyectos		CDPO000	
Mantenimiento de Programa de Reportes			

ENTIDAD	PERSONALCIENTES	Personal que actua con cheques	

Componente	CORF1	No. Linea 10	

Agregando Campo Constante			
Tq410	Fecha de Término de Proyecto :		
Secuencia	30		
Espacios A	6		

F12 - Anterior			

Se deberá especificar lo siguiente:

Texto Para agregar un campo constante se deberá digitar el texto del campo, hasta un máximo de cuarenta caracteres.

Secuencia: Indica el orden en que se imprimirá la constante en la línea componente.

Espacios: Indica el número de espacios que separarán este campo con el valor anterior en la línea de impresión.

Estos valores pueden ser modificados luego desde la línea de Impresión donde se definen.

Definir campo de resultado: Esta opción se utiliza para definir un campo de resultado que se imprimirá en un componente posterior a la definición, o en el mismo componente haciendo uso de los datos del registro. A cada uno se le asocia una instrucción SNAP, como por ejemplo: CALCULAR, ASIGNAR, EDTDDS, EDTTAB, LLAMAR, etc., o cualquier función definida en las funciones SNAP. Los campos resultado se definen en la siguiente pantalla:

SNAP Paseo 3 D Derechos Reservados S.O.H.C.R.		6/01/1985	19 30 42
Sistema Control de Proyectos		CDPO000	
Mantenimiento de Programa de Reportes			

ENTIDAD	PERSONALCIENTES	Personal que actua con cheques	

Componente	CORF1	No. Linea 10	

Agregando Campo de Resultado			
Campo	FECHA		

F12 - Anterior			

FALLA DE ORIGEN

Una vez seleccionada esta opción se desplegará la pantalla de especificación de campos de resultado, mostrada a continuación

SNAP Release 3.0 Derechos Reservados SOH C R		6/01/1995	19:30:42
Sistema Control de Proyectos		CUP0081	
Mantenimiento de Programa de Reportes			

INICIAZ:	PERSONALCIENTES	Personal que está con clientes	

Componente	CORTE1	No. Línea 10	

Agregando Campo de Resultado			
Campo	FICHA4	+ CALCULO + +	
Secuencia	100	Espacios A	2
Tipo	A	Longitud	8
Inicializar Campo después de Imprimir		Numero Decimales	0
		Secuencia de Cálculo	01

INSTRUCCION SNAP			

OBTENER FICHA4 = PERAÑO // PERMES // PERMA			

F4 = Lista, F10 = Oubrir Instrucción, F12 = Anterior			

Para definir un campo de resultado debe indicarse el nombre del campo, este campo no debe corresponder a un atributo o campo predefinido anteriormente.

Una vez digitado el campo, se despliega la pantalla de especificaciones del campo. En la pantalla se deberá digitar tanto información relacionada con las características del campo, como las especificaciones de cálculo que se realizarán para obtener el valor del campo de resultado

Secuencia: Indica el orden en que se imprimirá el campo en la línea del componente en que se trabaja.

Espacios: Indica el número de espacios que separarán a este campo con el campo anterior de la línea de impresión.

Código de edición: Código de edición que tendrá el campo, únicamente es válido en campos numéricos. Con el código de edición "E" se logra realizar el cálculo asociado a este campo en el momento de ejecución e imprimir el valor en un componente posterior.

Tipo: Se refiere al tipo de campo numérico o alfanumérico.

Longitud: Se debe indicar la longitud del campo.

Decimales: Únicamente se debe indicar el número de los decimales si el tipo de campo es numérico.

Inicializar campo después de imprimir: Se indica "S" si se desea que el campo sea inicializado después de haberse impreso. Los cálculos se realizan antes de la impresión, o sea este campo se podría acumular a otro y después de imprimirse inicializarse.

Secuencia de cálculo: este campo indica la secuencia de cálculo en caso de que existan más campos de resultado para este componente. Las instrucciones válidas para obtener el resultado del campo son:

CALCULAR
ASIGNAR
EDTDDS
EDTTAB

FALLA DE ORIGEN

LLAMAR FUNCIONES SNAP, ETC.

Si se define más de una instrucción SNAP para obtener el resultado del campo en la línea de instrucción mostrada, se debe separar cada una por un carácter ":" (dos puntos).

Lo anterior se realiza también para el corte de control de personal asignado en donde irá como constante "Personal" y enseguida los campos "PERCLA" y "PERNOM", que son la clave de el personal y el nombre de el personal respectivamente. En el detalle, en la segunda línea se incluirá como constante: "Puesto de el personal" enseguida el campo de "PERPUE", este campo define el puesto del personal y está incluido en la entidad de Personal asignado a Cliente.

Generación

El programa de Listado genera:

Entradas en la estructura de datos con la entidad primaria del programa, cada una de las entidades definidas a las que se haga referencia en alguno de sus campos y las entidades definidas en cada componente en su especificación, las entidades serán del tipo "L" (Lectura). Además incluirá las entidades que se encuentren definidas en el programa de estándares.

Variables de trabajo para la manipulación de datos, los campos de resultado definidos en los componentes y las definidas en el programa de estándares. Las variables de trabajo generadas por el programa son las siguientes:

ANT@@@n Estas son variables que corresponden a cada uno de los campos de la entidad definidos como corte de control para el programa. En éstas se almacena el valor del último corte efectuado en ese nivel para compararlo con el valor del nivel en cada registro leído, si el valor no coincide significa que se debe imprimir otro corte de control para ese nivel.

T@n A 1 Este es un indicador para definir la impresión del respectivo total en cada corte de control definido.

VVt@@@ A 8 Esta variable almacenará un valor, correspondiente a una constante según el modo que se utilice.

Formatos de impresión correspondientes a las especificaciones de cada uno de los componentes del reporte. El formato para el encabezado y el fin de reporte son definidos previamente.

Los módulos de ejecución del programa, la estructura modular del programa se muestra en el diagrama de referencia del programa Listado del manual de referencia de programas automáticos del SNAP.

Una vista de el reporte es la siguiente:

Sistema Control de Proyectos		6/01/1995	19 30 42
Estado de Personal Asignado a Clientes		CGP0000	Página 1
Cliente:	001 FONDO DE CULTURA ECONOMICA	FIN DE PLAN:	1995001
Personal:	028 Amanda Zambrano		
	PUESTO: PROGRAMADOR EN SHAP		
Personal:	030 MARCOS BEES PACHECO		
	PUESTO: PROGRAMADOR EN SHAP		
Personal:	036		
	PUESTO: PROGRAMADOR EN SHAP		
Cliente:	005 SEGUROS INTERAMERICANA	FIN DE PLAN:	1995012
Personal:	005 EMILIO ALAMOS SANCHEZ		
	PUESTO: LIDER DE PROYECTO		
Personal:	010 SERGIO VARGAS ANDRADE		
	PUESTO: PROGRAMADOR PPO		

4.5.8 Programa de Sesión de Clientes

Concepto

El programa automático de selección por navegación despliega una lista con los registros de la entidad a la cual se asocia y entidades a la que ésta se relacione por una dependencia o una referencia. Su salida se define en dos partes un encabezado que contiene la información de las entidades de las que depende y que conforman parte de su llave y líneas de detalle que corresponden cada una a un registro en la entidad.

El programa de selección por navegación implanta el concepto de "Trabajar con", el que consiste en seleccionar el objeto y luego definir cual es la acción a efectuar sobre él. A la lista de registros desplegada se aplican procesos en otros programas asociados a un mandato o una opción, se parametrizan los campos llaves del registro seleccionado y se invoca el programa.

El propósito de este tipo de programa es el crear un ambiente de navegación dentro de la información contenida en la base de datos y los procesos. Para navegar por la información en una base de datos basta definir programas de selección por navegación en cada entidad, parametrizando los campos llaves correspondientes a las dependencias e invocar cada programa SEL NAV definido para la entidad de la que depende.

Este programa asocia los procesos que se aplicarán a los registros de una entidad en forma individual, por opciones, o en forma grupal. Los programas de mantenimiento y consulta simple son asociados automáticamente a un programa de selección por navegación.

Pasos Previos

Se debe asegurar que los parámetros de generación para el tipo de programa sean los correctos. Estos pueden ser modificados en los parámetros para programas automáticos en los utilitarios SNAP. Refiérase al Apéndice de Parámetros de Instalación.

El programa de estándares para la generación deberá ser el indicado para el sistema donde radicará el programa de selección por navegación. La importancia radica en que según lo definido en éste, se establecerá el formato de pantallas y el módulo de inicialización del programa, así como las variables necesarias para su correcta generación.

Para aprovechar al máximo las características del programa de sesión por navegación, se sugiere revisar los siguientes puntos:

- Se debe definir los programas automáticos de consulta simple, mantenimiento de la entidad, cada uno de estos con las llaves locales de la entidad parametrizados, con el fin de aprovechar la facilidad de SNAP de asociarlos automáticamente a las opciones y mandatos correspondientes.
- Se debe definir cada uno de los programas de aplicación sobre los datos de un registro de la entidad, ya que las especificaciones de alto nivel del programa de selección por navegación estos pueden ser asociados y el programa generado contará con ellos. Sin embargo en las especificaciones de alto nivel puede definirse un programa que aún no exista pero en este caso no se podrá contar con el comando "LL" para formar la invocación de programa.
- Si las entidades dependientes tienen un programa de selección por navegación definido el cual reciba como parámetros las llaves de la entidad base a la que se asocia el programa, es conveniente que estos hayan sido definidos para asociarlos a una opción, sin embargo no es necesario.

Uso del Programa

El programa de selección por navegación debe ser definido después de cualquier otro programa automático interactivo sobre los registros de la entidad, pues SNAP asocia automáticamente éste a los programas definidos del tipo:

- Consulta simple, opción de consulta
- Mantenimiento simple "ABC", opción de modificación, opción de borrado y mandato de inclusión.
- Mantenimiento línea de detalle "ABC", opción de modificación, opción de borrado y mandato de inclusión.

Además cualquier otro programa no automático de estos tipos puede ser definido en las especificaciones de alto nivel del programa con ayuda de mandato "LL" que se encarga de asociar un programa existente con otro por medio de la instrucción "LLAMAR" y los parámetros definidos.

La utilización de un programa de selección por navegación se debe seguir una serie de pasos, los cuales en términos generales especifican que información debe estar presente en la pantalla y desde cual registro iniciará la lista según el ordenamiento de las llaves de la entidad.

Lo anterior se especifica utilizando la opción 12=Trabajar de la pantalla de mantenimiento de programas. Luego de utilizar esta opción sobre el programa se selección por navegación se despliega la pantalla mostrada en la siguiente figura:

SNAP Paises 3 B Derechos Reservados SDH C. R.		8-01/1998		19 30 42					
Sistema Control de Proyectos									
Programa Sesión de Clientes									

Entidad CLIENTES									

A = Elemento		ACCES		USO					
Seq	SEC	Entidad Atributo	COD	Or	POI	ENC	DET	Descripción	Lin
	10	CLIENTES							
		Código Cliente	AG	L	A	N	S	Cliente	
	20	CLIENTES							
		Nombre Cliente	ATR	L	S	N	S	Nombre	1

16 - Ayudar, 17 - Ligar Programas, F12 - Anterior

Los atributos se seleccionan con la opción 1=selección, luego los atributos deseados, estos se desplegarán en la pantalla anterior. Para cada atributo seleccionado se definirán las siguientes características:

SEC: Indica la secuencia de aparición del campo en la pantalla por el programa, este campo variará su significado según la posición donde se defina el campo en la pantalla utilizando los campos USO ENC y USO DET: así, se define:

- Como el orden de aparición en el encabezado de la pantalla cuando el valor de uso sea "S". Para nuestro ejemplo este uso encabezado no será utilizado.
- Como la posición en la línea del sub-archivo que desplegará la lista de los registros de la entidad cuando el valor de USO DET sea "S" o "A".

ORI: Este campo indica el origen del campo seleccionado, donde se desplegará el campo en el encabezado o el detalle. Los valores posibles para el campo son los siguientes:

E: Indica que el campo seleccionado se desplegará en el encabezado de la pantalla. Un valor "E" es inicializado cuando el campo pertenece a una entidad con una relación de dependencia.

L: Indica que el campo seleccionado se desplegará en la línea de detalle. el valor por omisión será "L" cuando el campo sea un campo de entidad o uno en una relación de referencia de la entidad.

Este campo es inicializado por SNAP dependiendo del origen del campo seleccionado, sin embargo puede ser modificado. Los campos que se presentan en el encabezado actúan como filtro de los registros que se muestran en el detalle, siempre y cuando la estructura de los campos llave de la entidad permita implantarlo, es decir los campos llaves desplegados en el encabezado deben preceder a los definidos en la línea de detalle.

USO ENC: Este campo es utilizado para definir la característica que tendrá el atributo seleccionado en el encabezado de la pantalla. Los posibles valores para este campo son los siguientes:

N: El atributo no se muestra en la parte superior de la pantalla.

FALLA DE ORIGEN

S: El atributo se desplegará en la parte superior de la pantalla, como campo de salida únicamente.

USO DET: Este campo es utilizado para definir las características del campo seleccionado en la línea de detalle del sub-archivo. Los valores posibles pueden definirse para el campo son los siguientes:

N: El atributo no se muestra en la línea de detalle del sub-archivo generado.

S: El atributo aparecerá en la línea de detalle de la pantalla como campo de salida.

A: El atributo se despliega en el detalle del programa como campo de salida, pero adicionalmente se define un campo con las mismas características en el encabezado sobre la posición del campo. Este campo del encabezado permitirá capturar un valor del campo seleccionado a partir del cual se desplegará la lista de registros del sub-archivo.

Descripción: En este campo se captura el texto asociado a cada campo en el encabezado de la pantalla donde se definió con un atributo de despliegue. esta descripción generará una constante en la pantalla justo antes del despliegue del campo a partir de las columnas 1 o 40 de la línea donde se ubique este.

El valor de este campo es inicializado automáticamente por SNAP a partir del nombre dado a cada atributo en el modelo de datos.

En el caso que este campo se deje en blanco SNAP no asignará una constante antecedente a la variable.

LIN: Indica el número de líneas que separan a un campo del anterior desplegado en el encabezado de la pantalla.

El valor por omisión es cero, en est caso SNAP acomodará los campos en las pantallas donde se halla definido el campo automáticamente. Esto es acomodando tantos campos por la línea como su longitud lo permita separando cada uno de ellos con un espacio en blanco. Cuando estos campos tengan una descripción asociada ésta se desplegará a partir de la columna 1 o 40 de la pantalla si hay espacio disponible.

Establecer relación con otros programas

En la pantalla anterior, de definición de especificaciones de alto nivel para el programa de selección por navegación, se definen mandatos para establecer las relaciones de invocación con otros programas

F7=Ligar programas: Este mandato permite ligar el programa mediante el uso de opciones o mandatos. La diferencia entre una liga por opción y por mandato se define a continuación:

- Una liga por opción se establece cuando el programa que se asocia recibirá los parámetros necesarios para trabajar sobre un único registro de los desplegados en la lista del programa de selección por navegación
- Una liga por mandato se establece cuando el programa asociado trabajará con todos los registros desplegados en la lista, o bien la información generada por el programa invocado se refiere a más de un registro de los desplegados por el programa invocador.

Al seleccionar este mandato se presenta la siguiente pantalla

SNAP Versión 2.0 Derechos Reservados SOIN C. R.	6/01/1995	18 30 42		
Sistema Control de Proyectos				
Programa Base de Clientes				

Entidad CLIENTES				

2 = Cambiar, 6 = Eliminar				
Seq	Función	Val	Par	Instrucciones
-	FUNCION	F06	S	LLAMAR COPO08 (PERCLA_MODO_ALTA)
-	OPCION	2	S	LLAMAR COPO08 (PERCLA_MODO_CAMBIO)
-	OPCION	4	S	LLAMAR COPO08 (PERCLA_MODO_BAJA)
-	OPCION	5	N	LLAMAR COPO07 (PERCLA)
-	FUNCION	F16	N	LLAMAR COPO06
-	OPCION	6	N	LLAMAR COPO08 (PERCLA_MODO_CAMBIO)
F6 = Agregar, F7 = Liga Programas, F12 = Anterior				

En esta pantalla se despliega una lista con los programas asociados con el programa de selección por navegación. En esta lista aparecen los programas que SNAP ligó automáticamente y los definidos por el usuario.

El SNAP liga automáticamente los programas del tipo MAN SAB y CON SIM definidos para la entidad, asignándolos a las funciones y opciones definidas en los parámetros automáticos de selección por navegación:

Función de agregar:	MAN SAB con parámetro: Alta
Opción de cambiar:	MAN SAB con parámetro de: Cambio
Opción de Eliminar:	MAN SAB con parámetro de: Baja
Opción de Visualizar	CON SIM con todos los campos llave como parámetros.

Si no existe un programa MAN SAB asociado a la entidad, el SNAP ligará un programa MAN LAB si esta definido. Si el programa CON SIM se definió sin parámetros o solo parte de la llave como parámetro, no se asociará automáticamente.

En la pantalla de ligas del programa se presentan las siguientes opciones y mandatos para definir la relación con los programas secundarios.

2=Cambiar: Esta opción permite entrar en la definición de una liga seleccionada y modificar datos relacionados con la renovación de la pantalla o el programa a invocar.

4=Eliminar: Con esta opción se elimina una opción o mandato del programa de selección por navegación.

F6=Agregar: Este mandato incluye nuevas ligas al programa, sean opciones o mandatos. Cuando se elige este mandato se despliega la siguiente pantalla.

FALLA DE ORIGEN

SNAP Release 3.0 Derechos Reservados SGM C. R.		6/01/1995	19 30 42
Sistema Control de Proyectos			
Liga de programas automáticos			

Entidad	CLIENTES	Programa	CDP0001

Tipo de Liga	F		
Valor	15		
F4 = Lista, F12 = Anular			

Los campos solicitados en esta pantalla son los siguientes:

Tipo de liga: En este campo se define la liga a crear. Especifica si la liga se hará por una opción de selección o bien por mandato. Los valores posibles para el campo son:

- F: La liga se establece por medio de mandato.
- O: La liga se establece por medio de una opción de selección.

Valor: En este campo se incluirá el número del mandato (Referirse Tabla 1) o el valor de la opción de selección elegida, este valor será el especificado para efectuar el acceso al programa. Una vez especificado la liga se despliega otra pantalla solicitando los siguientes campos:

SNAP Release 3.0 Derechos Reservados SGM C. R.		6/01/1995	19 30 42
Sistema Control de Proyectos			
Liga de programas automáticos			

Entidad	CLIENTES	Programa	CDP0001

Tipo de Liga	F	FUNCION	
Valor	15		
Reservar Datos	N		
Instrucción		SI = Llamar, F = Función, Y = Variable	
LLAMAR CDP0001			

F4 = Lista, F12 = Anular			

Renovar Pantalla: Este campo define si la lista desplegada por el programa deberá desplegarse de nuevo al terminar la ejecución de la liga o no, con el fin de refrescar los datos que pudieron alterarse. Los valores posibles para este campo son:

S: Actualiza la lista desplegada en el sub-archivo del programa.

N: Mantiene los datos del sub-archivo sin actualizar la información desplegada.

Instrucción: Se coloca la instrucción necesaria para invocar el programa o función a ejecutar con los parámetros respectivos. En realidad se puede definir cualquier instrucción del Lenguaje SNAP, que se ejecutará cuando se elija la liga a la que se asocia.

Existe un campo de ayuda para formar la instrucción, donde según la instrucción ligada permite formarla consultando los valores necesarios para que funcione correctamente. Los valores posibles son:

LL: Este valor se utiliza para construir la instrucción LLAMAR cuando se trate de la invocación a un programa.

F: Si se desea invocar una función permite escogerla y parametrizarla.

V: Para consultar las variables usadas por el programa, en este caso se debe utilizar el carácter "?" donde se requiera la variable dentro de la instrucción.

F7=Texto Opciones y Mandatos. Cuando se hayan definido las opciones y mandatos ligados en un programa se selección por navegación se debe parametrizar la pantalla del programa generado para que muestre al usuario las diferentes ligas que puede hacer. La pantalla generada por el programa tendrá dos partes para ello: justo después del despliegue de los campos del encabezado y antes de los nombres de las columnas de la línea de detalle se describen las opciones de los nombres de las columnas de la líneas 22 y 23 de la pantalla de mandatos.

Una vez ingresado este mandato se despliega la siguiente pantalla:

SNAP Release 3.0 Director Reservado SOIN C. R.		6/01/1988	19 30 42
Sistema Control de Proyectos			
Liga de programas automáticos			
Entidad	CIENTER	Programa	CDP0001
Texto de opciones :			
2 = Combinar, 4 = Eliminar, 5 = Consultar, 6 = Mantenimiento de Personal e Clientes			

Texto Líneas 22 y 23			
F8 = Renovar, F9 = Agragar, F10 = Incluir, F11 = Focal, F12 = Anterior, F13 = Listado			

F12 = Anterior, F23 = Eliminar Texto			

En esta pantalla se definen seis líneas para la hilera que describirá las opciones o mandatos:

- Las dos primeras líneas describirán las opciones de selección para las ligas del programa.

FALLA DE ORIGEN

- Las líneas tres y cuatro especificarán la función de los mandatos seleccionados y otros utilizados por el programa.
- Finalmente las dos últimas serán utilizadas para describir mandatos y habilitan el mandato F24 para desplegar alternativamente éstas con las líneas tres y cuatro. En el caso de utilizar estas dos líneas es conveniente agregar a la hilera la descripción del mandato F24.

Generación

El programa de selección por navegación genera:

Entradas en la estructura de datos con la entidad primaria del programa, y cada una de las entidades a las que se haga referencia en alguno de sus campos con tipo de acceso "L" (Lectura). Además se incluirán las entidades que se encuentren definidas en el programa de estándares.

Variables de trabajo para la manipulación de datos y las definidas en el programa de estándares.

Macros para las sustituciones dentro del archivo fuente del SNAP, sólo si el SNAP está configurado con está opción en los parámetros de generación de la herramienta.

Un archivo fuente de pantallas con un registro dividido en la pantalla del encabezado, el sub-archivo y las líneas de mandatos.

Los módulos de ejecución del programa: la estructura modular del programa además de los módulos del programa de estándares definido para el sistema.

Modo Experto en Programas de selección por Navegación.

Este modo de operación permite que desde el programa, se indique la llave del registro con el que se desea trabajar, es decir, el modo experto se puede utilizar conjuntamente con las opciones tanto de adición como de cambio, eliminación y consulta o para la invocación a otro programa con otra función particular.

El modo experto se logra indicando la opción deseada en la columna de selección (SEL) y la llave del registro que se desea incluir en el campo de actualización de ésta.

Una vez generado el programa y al ser invocado, este presenta la siguiente pantalla:

SNAP Release 3.0 Derechos Reservados SINH C. R.			B01/1995	10 30 42
Sistema Control de Proyectos				
Sección de Cheques			PANT10	CP0071
.....				
J = Cambiar, E = Eliminar, C = Consultar, B = Mantenimiento Personal Asegurado				
SEL	Código	Nombre		
--	---			
--	001	Seguros Internacionales		
--	002	Fondo de Cultura Económica		
--	003	La Quindiana		
--	004	Man Power		
--	005	La Nacional Constructora		
--	006	kratt		
F8 = Renovar, F9 = Agregar, F10 = Imprimir, F11 = Final, F12 = Anterior, F15 = Salir				

La realización de este procedimiento de generación de programas automáticos reporta una considerable ventaja en la construcción de estos programas con un mínimo de esfuerzo y tiempo. Para programas en los cuales no se ajusta la solución mediante el empleo de estos tipos de programas hay un tipo de programa llamado PRO BAT el cuál se podrá realizarse desde cero, pero con la ventaja de poder copiar desde otro programa un determinado módulo o pantalla sin necesidad de diseñarlas, sólo ajustando las variables de pantalla o de los módulos de él programa. Esto se lleva a cabo, debido a que todos los programas automáticos están, estandarizados y su estructura modular es muy semejante y de fácil comprensión, lo que permite detectar y aprenderse la estructura particular de cada módulo de cada programa y ajustarla al programa PRO BAT.

Generación de programas automáticos

Para generar los programas automáticos simplemente se utiliza la opción 14=Generar sobre cada programa definido para la entidad desde la lista de los programas automáticos definidos para una entidad.

El SNAP se encargará de generar el correspondiente código en SNAP que puede modificarse por medio del MDA utilizando la opción 18=MDA. La generación automática de macrosustituciones puede eliminarse con los parámetros de generación explicados en el manual de utilitarios del SNAP.

SNAP generará el código fuente RPG correspondiente al programa, cuyo nombre es el mismo definido para el programa SNAP más el prefijo TR en la biblioteca del programa en el archivo QRPGRSRC. Además del fuente en Lenguaje SNAP y el objeto en la biblioteca de programas.

Estos programas generados automáticamente se encuentran 100 % libres de errores de procesarse como se indicó anteriormente.

Capítulo V

Implantación y Mantenimiento

Introducción

En el capítulo III y IV se estudiaron muchas de las actividades específicas del diseño, necesarias para desarrollar un sistema de información. Para implantar con éxito el nuevo sistema, se deben de realizar entre otras actividades que no forman parte precisamente del trabajo de diseño, y que comprenden la capacitación y la instrucción del personal, así como la prueba del sistema. Por otra parte, debido a que todo sistema nuevo debe implantarse en el ambiente dinámico de una organización, hay un aspecto especial, llamado conversión de sistemas que interviene en la implantación. En realidad en el sistema COP no se necesita hacer esta conversión debido a que no había nada que controlara y documentara las actividades de todo el personal de la empresa resumido en un sistema debido a que es enteramente nuevo, pero se mencionara debido a que en actualidad generalmente se sustituye un sistema nuevo por uno anterior.

Entre estas actividades se efectúan pruebas parciales a muchos niveles, el manejo de los cuales los sustituye la prueba general del sistema. En las primeras fases pueden seguirse diversos métodos, que reflejan un equilibrio entre el costo de preparación de las pruebas y la confiabilidad que se establecen con ellas. La prueba se ha convertido en una actividad muy importante en la implantación de nuevos sistemas.

La implantación de muchos sistemas nuevos implica un proceso de conversión que requiere una o más consideraciones: 1) conversión del equipo, 2) conversión del método de procesamiento de datos y 3) conversión de procedimientos. Hay cuatro enfoques básicos de la conversión: 1) directa, 2) en paralelo, 3) modular y 4) por fases. El banco de datos exige una consideración especial durante el proceso de conversión.

La planeación es un aspecto importante de la conversión. Por lo común, se elabora un plan general antes de iniciar las actividades específicas de diseño y desarrollo. El plan específico de conversión se formula poco antes de la implantación real del sistema.

Una vez implantado el sistema, el analista desempeña la función de consultor. Inmediatamente después de la implantación, ayuda al personal de operación y a los usuarios a entender el nuevo sistema, asistiéndolos en la solución de los problemas que se presenten. Y en lo que se refiriere al sistema, este vendrá acompañado en toda su vida útil por el la actividad de mantenimiento que la realizan los programadores.

V.1 Capacitación e Instrucción del personal

En todo elemento principal de todo sistema lo constituyen las personas. Son estas las que diseñan, desarrollan, operan y conservan el sistema, para utilizar después los resultados que éste genera. Si se requiere implantar con éxito un nuevo sistema de información, a toda persona que participe en él se le debe informar, en primer lugar, de las responsabilidades que asume, y segundo, de lo que va a obtener del sistema. La responsabilidad primordial de este proceso de instrucción pertenece al analista.

En el análisis de la capacitación e instrucción del personal se verá lo siguiente: 1) qué clase de personal requiere capacitación, 2) los diversos enfoques del proceso de instrucción y 3) algunas consideraciones generales relativas a la elección de un método de capacitación.

Clasificación del personal que debe capacitarse

El personal que debe recibir alguna clase de instrucción puede clasificarse en dos categorías generales:

1) Los usuarios de la información. Esta categoría comprende a la administración general, de las distintas áreas funcionales, que incluye contadores, programadores etc. Generalmente se usa el término "instrucción" cuando se trata de informar a los usuarios con respecto a lo que el sistema requiere y lo que puede proporcionar. De hecho, el proceso de grupo comienza en la fase de análisis, que es información. En la última etapa, de la cual se habla ahora, se explica la forma en que el sistema de información, podrá satisfacer esas necesidades.

Por diversas razones, a menudo esta instrucción se reduce al mínimo o se omite durante el periodo de implantación de un sistema. El analista, que ha pasado varios meses pensando en el nuevo sistema y trabajando con él, no considera que los usuarios potenciales han estado poco tiempo en contacto con el mismo. Por otra parte, cuando los planes de lo que parece ser una instrucción masiva se presentan a la administración con frecuencia éstos se reducen o se eliminan, aduciendo que resultan demasiado costosos. Aunque es fácil calcular el costo de la instrucción, a veces es difícil definir todos los beneficios que aporta, así cuando ciertos recursos, como la información no se utilizan adecuadamente por falta de comprensión. El primer problema puede resolverlo el analista, quien reconoce que la instrucción del usuario y los planes siguientes forman parte del proceso de implantación del sistema. También puede superar el segundo problema mediante planes de instrucción bien concebidos, que presenta a la administración como parte del proceso.

2) El personal de operación. Este personal comprende a todas las personas relacionadas con la preparación de los elementos de entrada, con el procesamiento de datos y con el mantenimiento y operación de los componentes lógicos y físicos. También comprende a las personas responsables del control directo del sistema. Al proceso de instrucción de estas personas, generalmente se llama "capacitación". La capacitación del personal de operación presenta dos aspectos que el analista debe considerar. Primero, este personal debe ser entrenado en el manejo del nuevo sistema. Segundo, el entrenamiento debe continuar a medida que se va modificando el sistema o se va necesitando nuevo personal. La importancia que tiene el hecho de reconocer este segundo aspecto del entrenamiento se comprenderá mejor a medida que se estudien los diversos métodos que se pueden aplicar con el fin de proporcionar una capacitación aceptable cuando se implanta el sistema por primera vez.

5.1.1 Métodos de Instrucción y Capacitación del Personal

Los psicólogos y los educadores han demostrado que los diferentes objetivos de instrucción y capacitación exigen una gran variedad de métodos. Una conferencia será adecuada cuando se trata de explicar a un grupo de usuarios la forma en que opera el nuevo sistema; pero habrá de explicar el método de "aprender haciendo" cuando se trata de capacitar al nuevo personal de operación. Por otra parte, una gran cantidad de personas son capaces de realizar satisfactoriamente una determinada tarea después de practicarla una sola vez o unas cuantas veces. En consecuencia, para instruir y capacitar debidamente en el uso y operación de un nuevo sistema, el analista debe seguir muchos y diversos métodos, entre los cuales figuran los siguientes:

1. **Seminarios e instrucción en grupo.** Este método permite al analista llegar simultáneamente a un gran número de personas, y es muy útil cuando se trata de presentar un panorama general del sistema, además, es muy valioso en las grandes empresas, donde muchas personas realizan las mismas tareas.

2. **Capacitación a base de procedimientos.** Mediante este método se proporcionan al empleado los procedimientos escritos que describen las actividades, como modo fundamental de aprendizaje. Generalmente se le da oportunidad de hacer preguntas y plantear problemas que conciernen al procedimiento, ya sea en forma individual o en grupo, para completar sus conocimientos. Una extensión de esta técnica consiste en proporcionar una descripción formal del sistema, sobre todo los elementos de salida, a cada uno de los usuarios. El uso del correo para informar a los usuarios externos es otro ejemplo de la instrucción basada en procedimientos.

3) **Preceptoría.** Como el término implica, este método de capacitación es de naturaleza más personal y, en consecuencia, algo costoso. Sin embargo, combinado con otros métodos, puede eliminar cualquier vacío que impida la comprensión satisfactoria. En aquellos sistemas en que ciertas tareas son sumamente complejas o particularmente importantes para la buena operación, la Preceptoría puede ser indispensable para lograr buenos resultados. En la práctica, el analista capacita e instruye no sólo al personal de operación, sino también a los usuarios de los productos del sistema.

4) **Simulación.** Una técnica muy importante para la capacitación del personal de operación consiste en simular el ambiente de trabajo. esto puede hacerse con relativa facilidad reproduciendo los datos, procedimientos y equipos necesarios y dejando que el aprendiz ejecute las actividades del caso hasta lograr un nivel aceptable de rendimiento. Aunque el método de simulación parece resultar algo costoso, la práctica real demuestra que se cometen menos errores y se evitan las repeticiones cuando al empleado se enfrenta a la verdadera situación.

5) **Entrenamiento directo.** Posiblemente el método que más se emplea para capacitar al personal de operación consiste en ponerlo a realizar el trabajo. Por lo común, al aprendiz se le señalan algunas tareas sencillas, dándole instrucciones específicas con respecto a lo que debe hacer y a la forma de hacerlo. A medida que va dominando las tareas, se le asignan otras más complicadas. Con este método, la curva de aprendizaje puede prolongarse mucho y, en algunos casos puede resultar muy costoso lo que parece ser un aprovechamiento inmediato una rápida producción. Por otra parte, si alguna operación en particular es bastante compleja, la persona en cuestión llega a sentirse frustrada y prefiere renunciar.

5.2 Realización de Pruebas del Sistema.

A medida que la fase de desarrollo llega a su fin, es normal que las personas relacionadas estrechamente con el nuevo sistema experimenten un gran deseo de verlo en operación, general del movimiento de la empresa. Sin embargo, para que esto pueda hacerse con éxito, el analista debe asegurarse de que el sistema funcionará como está previsto. La prueba del sistema es una actividad que, igual que la capacitación del personal, exige una planeación cuidadosa por parte del analista. En esta sección se estudiarán los diferentes niveles en que puede realizarse la prueba, las distintas clases de pruebas que pueden aplicarse y la importancia creciente que tiene el someter a prueba todo nuevo sistema de información.

5.2.1 Niveles de Prueba.

La prueba de un sistema de información recién desarrollado debe efectuarse en varios niveles.

Generalmente, la prueba de un módulo lógico aislado corresponde al programador. Esta actividad es el nivel de prueba más específico. Como ejemplos de prueba de un módulo lógico se tienen los siguientes: verificar si todas las transacciones introducidas han quedado incorporadas, verificar si una transacción específica actualiza correctamente el archivo maestro, verificar si los encabezados se imprimen en cada una de las páginas, y verificar si todos los archivos quedan cerrados al finalizar el proceso. Generalmente, la prueba del programa también corresponde al programador. Igual que en el caso de la prueba de un módulo lógico, el programador verifica si produce exactamente el resultado previsto y si todas las tareas normales (por ejemplo, la apertura y el cierre de los archivos) se ejecutan de acuerdo con lo planeado.

La prueba secuencial consiste en probar dos o más programas que se procesan en secuencia, con el fin de ver si pueden depender uno de otro para su correcta operación. Una secuencia de programas puede incluir el ciclo de validación, los programas de mantenimiento de los archivos o los programas de actualización. La responsabilidad de la prueba secuencial puede corresponder a una sola persona, que puede ser el analista, o bien, la pueden compartir los programadores participantes.

La prueba del sistema de computadora implica probar todos los programas que se implantarán para apoyar el sistema. Todos los elementos de entrada se preparan y se implantarán para apoyar el sistema. Todos los elementos de entrada se preparan y se procesan con el fin de producir los resultados previstos, los cuales se verifican en cuanto a exactitud y confiabilidad. También este nivel de prueba implica una responsabilidad, pero quien la coordina es el analista de sistemas o el programador en jefe.

La prueba del sistema se diferencia del nivel anterior en el que se incluyen todos los procedimientos de oficina. Se produce un evento que se registra y prepara para el procesamiento; se procesa, se corrigen los errores, se prepara y ejecuta el mantenimiento del archivo maestro, se actualiza el banco de datos y se formulan los informes. Toda discrepancia que se observen en los resultados en el banco de datos es seguida a través de las etapas de procesamiento hasta que el problema queda definido y resuelto. Existe la misma probabilidad de que el error aparezca tanto en el procedimiento de oficina como en el programa de computación.

5.2.2 Clases de Prueba

Hay cinco clases generales de prueba del sistema, que pueden efectuarse en cada nivel.

La prueba lógico-simulada se efectúa cuando el programador o el analista prueban el sistema siguiendo el programa o programas con una presentación mental de transacción. Los programadores llaman a este método "prueba de escritorio". El analista de sistemas lo usa continuamente cuando está desarrollando el sistema. Su ventaja está principal consiste en que no cuesta casi nada, y su desventaja consiste en que el nivel de confiabilidad es relativamente bajo.

El uso de datos al azar significa que el analista o el programador preparan una serie de transacciones con objeto de ver si el programa las ejecuta y llega hasta el final. También este método es relativamente económico, pero tiene un grado limitado de confiabilidad.

Para la tercera prueba se utilizan datos reales, se eligen transacciones reales para el procesamiento. Generalmente, el programador o el analista piden al personal de oficina que proponga algunas transacciones que constituyan diversos tipos de entradas. Puesto que es poco probable que el encargado de las entradas sepa hasta ese momento lo que el nuevo sistema va a aceptar o a rechazar, la entrada representa un muestreo al azar, sólo que a escala mucho mayor. La ventaja de usar datos reales reside en que su costo de preparación es bajo, aunque no deja de ser mayor que el de los métodos antes citados. El grado de confiabilidad asociado con este método es mediano; pero debe considerarse todavía como una desventaja.

Al cuarto método se le llama prueba de producción. El sistema debe procesar todos los elementos de entrada. Si al verificarse los resultados aparecen errores, se corrigen y se corre de nuevo la prueba de producción. Si no se detectan errores, se corrigen y se corre de nuevo la prueba de producción. Si no se detectan errores, los resultados se distribuyen entre los usuarios de la organización. La ventaja de este método es que se presenta un grado de confiabilidad entre mediano y alto y, en un momento dado, produce realmente buenos resultados. La desventaja está en que la preparación de los datos de entrada es costosa, así también lo es la verificación de los resultados, el rastreo de errores y la repetición de pruebas.

Al quinto método se le llama prueba controlada. El control se refiere a la manera de preparar los datos de entrada. Se prepara cada una de las operaciones de entrada para todas las combinaciones que los campos de datos pueden incluir. Esto se hace para asegurarse que el sistema funciona no sólo para los pasos válidos de procesamiento, sino también para las transacciones no válidas y equivocadas. Como regla, los sistemas funcionan mal no porque pueda realizarse aquello para lo que se diseñaron, sino porque no pueden manejar situaciones no válidas o extraordinarias. La evidente ventaja de la prueba controlada es que da al sistema un alto grado de confiabilidad. Otra ventaja, no tan evidente, es que el procedimiento de verificación es relativamente rápido y poco costoso. Es así porque el programador o analista conocen los datos que se introdujeron y pueden comprobar rápidamente si los resultados son correctos. La desventaja está en el costo elevado de preparación de los datos y que, en la mayoría de los casos es imposible probar todas las combinaciones de cada transacción.

5.2.3 La Importancia Creciente de las Pruebas

La importancia de las pruebas, como actividad fundamental del desarrollo, es cada vez mayor, debido a varias razones.

1. La tendencia hacia una mayor integración de los sistemas que componen la organización, exige que cada nuevo sistema implantado funciones con éxito desde el principio, no sólo con respecto a sus propios fines, sino en forma que no degrade a los otros sistemas.

2. Debido a que para tomar decisiones y solucionar problemas, los usuarios dependen cada vez más de la información generada por la computadora, resulta que el funcionamiento de la organización depende directamente del rendimiento de los sistemas.

3. El amplio uso y la familiaridad que se ha alcanzado con los sistemas basados en la computadora han dado lugar a mayores expectativas por parte de los usuarios.

4. La tendencia inflacionaria del costo de otras actividades del desarrollo se puede contrarrestar mediante los procedimientos mejorados de prueba.

5. La inversión en recursos para el mantenimiento del sistema puede reducirse estableciendo mejores procedimientos de prueba antes de instalarlo.

En un intento de mejorar los procedimientos de prueba del sistema, el analista debe poner en juego una gran creatividad. La participación de los usuarios en el periodo de prueba es un método que se utiliza mucho. Otros han desarrollado, o adquirido, otros medios que, en curso de la prueba, utilizan a la propia computadora para detectar los defectos (o fallas potenciales).

5.3 Conversión de sistemas

El término conversión sirve para describir el proceso de cambiar, de una manera de hacer las cosas, a otra diferente. Cuando se aplica al sistema de información de una empresa, describe los cambios introducidos en las actividades relacionadas con la recompilación, almacenamiento, recuperación y procesamiento de datos, y con la presentación de la información. Al analizar el proceso se irán identificando las diversas clases de conversión, los diferentes métodos para lograrla, las consideraciones especiales relativas al banco de datos y la importancia que tiene el planear la conversión.

5.3.1 Clases de Conversión.

hay que señalar tres clases generales de conversión, con las cuales debe estar familiarizado el analista: 1) la conversión del equipo, 2) la conversión del método de procesamiento de datos y 3) la conversión de procedimientos.

La conversión del equipo, esta conversión implica la sustitución de una parte del equipo, y puede incluirse también la reposición de partes, como máquinas de escribir, copadoras etc. Anteriormente la conversión de una computadora a otra implicaba la reestructuración o la recompilación de los programas existentes que se irían a procesar en la nueva computadora. Esta conversión no significaba necesariamente cambiar la lógica del programa, aunque ésta podría ser afectada, sino se trataba más bien de dar a la lógica una estructura de codificación que fuera procesada en la nueva máquina. Actualmente la conversión rara vez exige la reprogramación. En muchos casos, la nueva computadora o el dispositivo periférico son de conexión compatible con el equipo antiguo. Dicho de otro modo, la estructura física y la operación del nuevo equipo no tienen ningún efecto en la lógica de los programas. En otros, puede hacerse que el nuevo equipo imite las operaciones del anterior. Con

frecuencia, estas dos mejoras en el diseño de equipos han permitido adquirir una máquina más veloz o menos costosa sin incurrir en grandes gastos de conversión.

Conversión del método de procesamiento de datos. Esta conversión describe el cambio de un método de procesamiento de datos a otro diferente. Como se había mencionado en el primer capítulo, los primeros avances en la automatización del procesamiento de datos consistieron en la sustitución, una por una, de ciertas actividades o grupos de actividades. No se pretendía alterar el diseño general del sistema. El enfoque inicial del uso de la computadora era también el de "automatizar" o "computarizar" determinadas actividades. Actualmente, aplicando el enfoque de sistemas, el diseño del sistema de la información requiere, en la mayoría de las empresas, bastante más que la simple conversión del método de procesamiento. Generalmente, con la conversión del método va aparejada una serie de extensos cambios de procedimiento.

La conversión de procedimientos. La conversión de procedimientos puede abarcar tanto en el cambio de las actividades como en la alteración de la secuencia en que se realizan esas actividades. Los procedimientos sujetos a conversión pueden ser manuales o automatizados. La conversión de procedimientos se puede efectuar sola o en combinación con la conversión del equipo o del método de procesamiento de datos. Por ejemplo, la implantación de una nueva forma de recoger los datos, así como la modificación o reestructuración de un programa de computación, no son sino cambios de procedimiento.

Quando se trata de implantar un sub-sistema principal del sistema de información, el analista se enfrentará seguramente a los tres tipos de conversión. Si no es capaz de relacionar lógicamente las actividades de conversión de acuerdo con el tipo que representan, el analista tendrá una buena idea del enfoque que debe aplicar a la conversión general de los sistemas.

5.3.2 Enfoques de las Conversiones.

Hay cuatro enfoques básicos para la conversión a un nuevo sistema: 1) directo, 2) en paralelo, 3) modular y 4) por fases.

1) **Conversión directa.** La conversión directa es la implantación del nuevo sistema con la supresión inmediata del anterior. A veces, se le llama método "si contemplaciones". Este método tiene sentido: 1) cuando el sistema no sustituye a ningún otro sistema, 2) cuando se considera que el sistema anterior no tiene ya valor alguno, 3) cuando el nuevo sistema es muy reducido, o bien, muy sencillo, y 4) cuando el diseño del nuevo sistema es totalmente diferente del anterior y no puede hacerse ninguna comparación. La ventaja principal de este enfoque es que resulta relativamente barato, y la desventaja principal radica en que ofrece un gran riesgo de fracasar. Cuando es preciso recurrir a la conversión directa, las pruebas que se estudiaron en la sección anterior adquieren mayor importancia.

2) **Conversión en paralelo.** La conversión en paralelo describe una situación en la que el viejo y el nuevo sistema operan simultáneamente durante cierto tiempo. Los resultados de cada sistema se comparan, y se origin las diferencias. La ventaja de este método es que ofrece un alto grado de protección contra las fallas del nuevo sistema. La desventaja obvia está en el costo que implica la duplicación de instalaciones y el personal necesario para atender el doble sistema. Este método de conversión es ampliamente empleado debido a la gran cantidad de problemas que las empresas afrontaban anteriormente al implantar un nuevo sistema. Cuando el proceso de conversión incluya la operación en paralelo, el analista debe planear la revisión periódica, junto con el personal de operación y los usuarios, del funcionamiento del nuevo sistema, fijando una fecha razonable para la aceptación de éste y la supresión del anterior.

3) **Conversión modular.** La conversión modular, llamada a veces "enfoque piloto", se refiere a la implantación por partes de un sistema. Por ejemplo, un sistema de registro de pedidos se puede instalar en una determinada área, y así sucesivamente. El sistema de inventarios sería otro ejemplo; podría convertirse para un grupo seleccionado de productos, o para todos los productos pero únicamente en una localidad cuando la empresa opera en varias. Las ventajas de este enfoque son las siguientes: 1) el riesgo de que falle el sistema queda localizado, 2) los problemas detectados se pueden corregir antes de proseguir con la implantación, y 3) otro personal de operación se puede entrenar en un medio "vivo" antes de que el sistema se implante en su propia área. Una de las desventajas de este método es que el periodo de conversión puede prolongarse demasiado y, lo que es más importante, en el caso de ciertos sistemas u organizaciones, no siempre se puede practicar.

4) **Conversión por fases.** La conversión por fases es algo similar a la conversión modular, pero difiere de esta última en que se fragmenta el sistema y no la organización. Por ejemplo, pueden implantarse las nuevas actividades de recopilación de datos, estableciendo un mecanismo de relación con el sistema anterior. Esto permite al antiguo sistema operar con los nuevos datos de entrada. Posteriormente se instalan el acceso al nuevo banco de datos y las actividades de almacenamiento y recuperación, provyendo de nuevo un mecanismo de relación con el antiguo sistema. Así se va instalando cada segmento del nuevo sistema, hasta implantarlo todo. Cada vez que se agrega un nuevo segmento hay que establecer una relación con el sistema anterior. Las ventajas de este enfoque son las siguientes: 1) la tasa del cambio, en una organización dada, puede reducirse al mínimo, 2) puede implantarse un sistema integrado de información en un tiempo largo y con un presupuesto mínimo, y 3) los recursos para el procesamiento de datos pueden adquirirse gradualmente en un periodo prolongado. Las desventajas son: 1) el costo que implica el establecimiento de relaciones temporales con el antiguo sistema, 2) la aplicabilidad limitada y 3) la sensación desmoralizadora de que "nunca se acaba de completar el sistema".

5.3.3 Consideraciones Relativas al Banco de Datos durante la Conversión

El éxito de la conversión de un sistema depende en gran parte de la forma de que el analista prepare la creación y la conversión de los archivos de datos que requiere el nuevo sistema. Esta preparación es muy importante cuando el sistema de información de la empresa está altamente integrado a través de su banco de datos. En algunas compañías grandes, donde existe un sistema integrado de información, el analista puede colaborar estrechamente con el administrador del banco de datos en la preparación de la creación y conversión de archivos. Sin embargo, para nuestro propósito, se pondrá que corresponde al analista hacer todos los preparativos para la conversión de sistemas.

Crear un archivo quiere decir simplemente recoger y organizarlos datos, con un formato conveniente, en un determinado dispositivo de almacenamiento. Convertir un archivo significa que alguna ya existente se debe modificar por los menos en una de estas tres maneras: 1) en su formato, 2) en su contenido y 3) en el medio que lo contiene. En una conversión de sistemas, es muy probable que algunos archivos se tengan que someter simultáneamente a las tres formas de conversión.

Cuando se crea un archivo que se va a procesar por medio de la computadora, algunas veces es necesario disponer un sistema de utilería especial de arranque que defina y etiquete una ubicación específica, tanto física como lógica, para dicho archivo. A este proceso se le llama crear un archivo "ficticio". Una vez creado, el nuevo sistema procesará y almacenará los datos pertinentes a este archivo.

Cuando se convierte un archivo que se va a procesar por medio de la computadora, también se requiere un sistema de utilería especial de arranque, el cual contiene una lógica que permite introducir los datos existentes con su formato anterior y en el medio anterior, para obtenerlos a la salida con un nuevo formato y sobre un nuevo medio. Con respecto a la conversión del contenido de un archivo, el sistema de utilería para arranque puede simplemente inicializar (poner en cero) los nuevos campos del

archivo con objeto de poder actualizarlos en forma cuando el sistema empiece a procesar las transacciones.

A menudo, durante la conversión de archivos es necesario establecer procedimientos complicados de control para garantizar la integridad de los datos disponibles después de la conversión. Utilizando la clasificación de archivos que se presentó en el segundo capítulo, se harán algunas observaciones generales con respecto a los diferentes tipos de archivo durante la conversión:

1) Archivos maestros. Los archivos maestros son la clave del banco de datos y generalmente es preciso crear o convertir por lo menos uno en la etapa de conversión de sistemas. Cuando un archivo maestro se debe convertir al nuevo sistema, el analista debe disponer una serie de totales fragmentarios y de control, para compararlos con todos los campos del archivo convertido, con el fin de asegurarse de que la conversión se hizo con la debida propiedad. Deben establecerse procedimientos especiales de protección por separado para cada paso del procedimiento. Esta precaución evitará el tener que reiniciar innecesariamente la conversión desde el principio en el caso de que después se detecte algún error en la lógica. Las consideraciones de tiempo son particularmente importantes en la conversión de archivos maestros, sobre todo en los sistemas en línea. Si el archivo convertido no va a implantarse inmediatamente después de la conversión, deben adoptarse medidas especiales para tomar nota de toda actividad de actualización que se realice entre el momento de la conversión y la fecha de implantación. En aquellos sistemas en los que las funciones de administración de datos se realizan mediante programas de aplicación individual, el analista debe asegurarse de que cada uno de los programas que tengan acceso al archivo convertido también se haya modificado de manera que acepte el nuevo formato y el nuevo medio.

2) Archivos de transacciones. Los archivos de transacciones se crean generalmente mediante el procesamiento de un sub-sistema individual dentro del sistema de información y por lo tanto se pueden verificar cuidadosamente durante la prueba de los sistemas. Sin embargo, los archivos de transacciones generados en otras partes del sistema de información aparte del nuevo sub-sistema se tendrán que convertir si los archivos maestros que actualizan se han convertido en formato o en medio.

3) Archivos de índices. Los archivos de índices contienen las llaves y direcciones que vinculan a los diversos archivos maestros. Por lo tanto, es preciso crear otros nuevos cuando los archivos maestros correspondientes han sufrido una conversión que invalide las llaves y direcciones contenidas en los índices anteriores.

4) Archivos de tablas. También los archivos de tablas se pueden crear o convertir durante la conversión de sistemas. Las mismas consideraciones hechas con respecto a los archivos maestros son aplicables a los archivos de tablas.

5) Archivos resúmenes. Durante el procesamiento del nuevo sistema, en forma similar los archivos de transacciones. Generalmente, los archivos de resúmenes creados en otras áreas del sistema de información no se tendrán que convertir cuando se implanta un nuevo sub-sistema.

6) Archivos históricos. Los archivos históricos pertenecen a una categoría similar a la de los archivos maestros. Las consideraciones hechas con respecto a estos últimos son aplicables a los archivos históricos, con dos excepciones: 1) las consideraciones de tiempo no son tan severas con los archivos históricos como lo son para los archivos maestros, aun cuando el procesamiento es en línea, y 2) generalmente el volumen de los registros que aparecen en los archivos históricos es mucho mayor que el que contienen los archivos maestros.

7) Archivos de protección. El propósito de estos archivos es proteger al sistema en el caso de que se cometan errores de procesamiento o de que ocurra un desastre en el centro de información. Por lo tanto, cuando se convierte o se crea un archivo, es necesario crear también un archivo de protección. Es muy probable que los procedimientos de protección para el archivo sean los mismos que se sugirieron con el original. Se haría una excepción en el caso de que se efectuara un cambio de medio.

archivo con objeto de poder actualizarlos en forma cuando el sistema empiece a procesar las transacciones.

A menudo, durante la conversión de archivos es necesario establecer procedimientos complicados de control para garantizar la integridad de los datos disponibles después de la conversión. Utilizando la clasificación de archivos que se presentó en el segundo capítulo, se harán algunas observaciones generales con respecto a los diferentes tipos de archivo durante la conversión:

1) Archivos maestros. Los archivos maestros son la clave del banco de datos y generalmente es preciso crear o convertir por lo menos uno en la etapa de conversión de sistemas. Cuando un archivo maestro se debe convertir al nuevo sistema, el analista debe disponer una serie de totales fragmentarios y de control, para compararlos con todos los campos del archivo convertido, con el fin de asegurarse de que la conversión se hizo con la debida propiedad. Deben establecerse procedimientos especiales de protección por separado para cada paso del procedimiento. Esta precaución evitará el tener que reiniciar innecesariamente la conversión desde el principio en el caso de que después se detecte algún error en la lógica. Las consideraciones de tiempo son particularmente importantes en la conversión de archivos maestros, sobre todo en los sistemas en línea. Si el archivo convertido no va a implantarse inmediatamente después de la conversión, deben adoptarse medidas especiales para tomar nota de toda actividad de actualización que se realice entre el momento de la conversión y la fecha de implantación. En aquellos sistemas en los que las funciones de administración de datos se realizan mediante programas de aplicación individual, el analista debe asegurarse de que cada uno de los programas que tengan acceso al archivo convertido también se haya modificado de manera que acepte el nuevo formato y el nuevo medio.

2) Archivos de transacciones. Los archivos de transacciones se crean generalmente mediante el procesamiento de un sub-sistema individual dentro del sistema de información y por lo tanto se pueden verificar cuidadosamente durante la prueba de los sistemas. Sin embargo, los archivos de transacciones generados en otras partes del sistema de información aparte del nuevo sub-sistema se tendrán que convertir si los archivos maestros que actualizan se han convertido en formato o en medio.

3) Archivos de índices. Los archivos de índices contienen las llaves y direcciones que vinculan a los diversos archivos maestros. Por lo tanto, es preciso crear otros nuevos cuando los archivos maestros correspondientes han sufrido una conversión que invalide las llaves y direcciones contenidas en los índices anteriores.

4) Archivos de tablas. También los archivos de tablas se pueden crear o convertir durante la conversión de sistemas. Las mismas consideraciones hechas con respecto a los archivos maestros son aplicables a los archivos de tablas.

5) Archivos resúmenes. Durante el procesamiento del nuevo sistema, en forma similar los archivos de transacciones. Generalmente, los archivos de resúmenes creados en otras áreas del sistema de información no se tendrán que convertir cuando se implanta un nuevo sub-sistema.

6) Archivos históricos. Los archivos históricos pertenecen a una categoría similar a la de los archivos maestros. Las consideraciones hechas con respecto a estos últimos son aplicables a los archivos históricos, con dos excepciones: 1) las consideraciones de tiempo no son tan severas con los archivos históricos como lo son para los archivos maestros, aun cuando el procesamiento es en línea, y 2) generalmente el volumen de los registros que aparecen en los archivos históricos es mucho mayor que el que contienen los archivos maestros.

7) Archivos de protección. El propósito de estos archivos es proteger al sistema en el caso de que se cometan errores de procesamiento o de que ocurra un desastre en el centro de información. Por lo tanto, cuando se convierte o se crea un archivo, es necesario crear también un archivo de protección. Es muy probable que los procedimientos de protección para el archivo sean los mismos que se sugirieron con el original. Se haría una excepción en el caso de que se efectuara un cambio de medio.

La determinación de cuáles se deben crear o convertir cuando se implanta un nuevo sistema, y de la forma en que dichos archivos se crearán, es parte del proceso de razonamiento necesario para elaborar un plan de conversión.

5.3.4 Planeación de la Conversión

Aunque un sistema se haya diseñado y desarrollado adecuadamente, la mayor parte de su éxito dependerá de la forma en que se lleve a cabo la conversión. Cuando el nuevo sistema produce información inexacta o poco oportuna, debido a que ciertas actividades no se realizaron en la forma prevista, puede crear un estigma que sigue latente mucho después de que el problema quedo resuelto. Para evitar que se produzca la desconfianza entre el nuevo sistema y los resultados que produce, el analista de sistemas debe planear cuidadosamente la conversión.

En la práctica, generalmente el plan de conversión señala la programación del trabajo de sistemas que se efectúa en la fase de desarrollo. El plan específico, elaborado en las etapas finales de prueba, indica los procedimientos especiales de entrenamiento del personal, el plan y el programa para la creación y conversión de archivos, los criterios iniciales para la aceptación y los procedimientos especiales para iniciar el control.

La elaboración de planes formales para efectuar la conversión de sistemas es otra tarea fundamental de comunicación para el analista. Es muy importante que la administración de la empresa comprenda cabalmente el enfoque que se va a darse a la conversión. Como regla, las recomendaciones del analista en este sentido se sopesarán cuidadosamente en relación con otros compromisos y actividades planeados durante el mismo periodo. Para lograr una exitosa implantación del sistema, ayudará la preparación de planes de conversión bien concebidos, que luego se comunicarán a todo el personal interesado de la organización.

5.4 Secuela de la Implantación

Una vez implantado el nuevo sistema, la participación del analista no terminará necesariamente, se mencionaran las tareas durante el periodo que sigue inmediatamente a la implantación.

En primer lugar, debe verificar regularmente si se están cumpliendo los programas de entrada, procesamiento y salida. Cuando parezca que se ha establecido una rutina, las verificaciones pueden hacerse con menos regularidad, dirigiendo la atención hacia los puntos problemáticos que se lleguen a encontrar.

Periódicamente se deben revisar las actividades del personal encargado de la preparación de los datos de entrada. Si casi seguro que algunos procedimientos manuales requieran una aclaración adicional. También puede detectarse algún problema de programación que requiera solución inmediata. A veces se encontrará que ciertos procedimientos, manuales o de computación, son un tanto ineficientes. Un pequeño cambio puede eliminar los problemas difíciles en las operaciones del sistema.

Quizá la más importante actividad que realiza el analista consiste en comprobar si los controles del sistema funcionan correctamente. En muchos casos en que se ha implantado un gran sistema, al principio se procesan inadvertidamente muchos errores de entrada. Por lo común, no existe un mecanismo eficiente de mantenimiento de archivos en que haya gran cantidad de errores. Durante el periodo de aprendizaje, el analista puede ayudar a las operaciones del sistema, ya se han indicado un

método ágil para corregir errores, o señalando a los correspondientes supervisores los puntos donde hace falta ayuda adicional.

Uno de los problemas que afrontará el analista durante el periodo posterior a la implantación consiste en distinguir entre las mejoras y refinamientos que se le sugieren, y los verdaderos problemas que presenta el sistema. En el periodo de instrucción de los usuarios, inicialmente la información se hará hincapié en la corrección de errores. Las sugerencias de mejoras al sistema son bien recibidas y alentadas; pero dichas sugerencias se reunirán y evaluarán una vez terminada la implantación. En esta forma, todos los usuarios tendrán la oportunidad de ser escuchados y las mejoras posteriores al sistema se podrán efectuar en conjunto por lo que respecta a los sub-sistemas (o módulos) del nuevo sistema. Si no se hace esta distinción, las actividades de implantación se prolongarán indefinidamente.

Una actividad final, que el analista llevará a cabo durante este periodo consiste en retirar los procedimientos, programas, formas, etc. anticuados y de relación, que formaron parte del sistema anterior o del periodo de conversión, que formaron parte del sistema anterior o del periodo de conversión. Esto eliminará la posibilidad de que alguien, inadvertidamente, haga referencia o utilice el procedimiento o programa equivocado.

El analista debe tener una junta de aceptación con la administración del sistema y con los usuarios del mismo, en cuyo momento se da por terminado oficialmente el proyecto. A partir de entonces, el analista queda disponible para otra comisión.

Conclusiones

Con este texto se pretende ayudar a conocer lo que es una herramienta CASE y como es que este tipo de herramienta esta revolucionando la manera de desarrollar sistemas de información en la actualidad, pasando de una labor artesanal a una disciplina de ingeniería mucho más rigurosa y previsible ya que este tipo de herramientas emplea técnicas y metodologías que están probadas ampliamente en la actualidad y la novedad es que estas técnicas y metodologías están totalmente integradas en un entorno de ingeniería de software. Lo cual acelera substancialmente la productividad de desarrollo de sistemas sin perder la flexibilidad y control sobre estos proyectos de desarrollo.

Sin embargo este nivel de mejoramiento de la productividad es alcanzado en forma proporcional a los siguientes factores:

- Organización de trabajo
- Utilización de la metodología entidad-relación

Las ventajas que proporcionan este tipo de herramienta es que dan soporte a las distintas fases del ciclo de vida de un proyecto: planificación del proyecto, definición de requerimientos, análisis, diseño, definición de pruebas, codificación, gestión de la configuración del software y el mantenimiento. Sin embargo para tener un buen éxito en la Utilización de estas herramientas se deben de hacer consideraciones importantes como: conocer exactamente la situación actual de la empresa, entender perfectamente lo que se quiere conseguir con la automatización del desarrollo de un determinado proyecto, realizar una adecuada selección de las herramientas disponibles en el mercado y finalmente elaborar un estudio detallado con los pasos a seguir para la implantación de la nueva herramienta.

El motivo por el cual se tomo el sistema COP (Control de Proyectos) fue para dar una explicación de cada uno de los pasos a seguir en el desarrollo de sistemas de información y tomándose a este sistema como ejemplo, ya que este fue realizado en el SNAP. Paralelamente se trato de hacer correcciones a algunos programas para alcanzar uno de los objetivos principales de esta obra que era el de arrancar el sistema. Desgraciadamente no se pudo realizar la implantación debido a que se presento un cambio de trabajo a tres cuartas partes de entregado el proyecto para su corrección. A pesar de que se tenía todavía el apoyo para seguir con el proyecto no se pudo continuar por el motivo antes citado.

Sin embargo el motivo principal del texto es explicar la metodología que se sigue para el desarrollo de proyectos de sistemas de información usando este tipo de herramientas lo cual se considera que fue alcanzado.

Bibliografía

CASE la automatización del software
Autor: Carma McClure
Editorial: Addison-Wesley Iberoamericana
Edición: Primera Edición 1993

Sistemas de Información Teoría y práctica
Autor: John G. Burch Jr.
Felix R. Strater Jr.
Editorial: Limusa
Edición: Primera Edición 1981

Introducción a la Ingeniería de Software Un Enfoque Práctico
Autor: Reyes S. Presman
Editorial: Limusa
Edición: Primera Edición 1992

SNAP Modelo de Datos
SNAP Método de Desarrollo Acelerado
SNAP Referencia al Lenguaje SNAP
SNAP Instalación y Configuración
SNAP Utilitarios
SNAP Seguridad
Autor:
Editorial: SNAP CO.
Edición: Segunda Edición 1990

Herramientas CASE, Metodología Estructurada para Desarrollo de Sistemas
Autor: Williams S. Davis
Editorial: Paraninfo
Edición: Primera Edición 1990

Ingeniería de Sistemas
Autor: Arthur D. Hall
Editorial: Compañía Editorial Continental
Edición: sexta Edición 1977

Sistemas: Concepto, Metodología y Aplicaciones
Autor: Brian Wilson
Editorial: Wiley Megabyte
Edición Primera Edición 1993

Pensamiento de Sistemas
Autor: Peter Checkland
Editorial: Wiley Megabyte
Edición: Primera Edición 1993

El Enfoque de Sistemas
Autor: Jerez-Grijalvo
Editorial: Limusa
Edición: Cuarta Edición 1990