

84



UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO

FACULTAD DE INGENIERÍA

SISTEMA INTEGRAL DE RECURSOS
HUMANOS

T E S I S
QUE PARA OBTENER EL TÍTULO DE
INGENIERO EN COMPUTACIÓN
P R E S E N T A
HORACIO ZALDÍVAR GAMBOA

DIRECTOR: M. en I. JUAN CARLOS ROA BEIZA



279415

MÉXICO, D.F.

2000



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

***A mis padres por su apoyo
cariño, consejos y paciencia.***

***A mi hermano
por sus recomendaciones.***

CAPITULO 1 REGLAS DEL NEGOCIO. 1

1.1	MARCO LEGAL QUE RIGE UNA NÓMINA.	1
1.1.1	NORMATIVIDAD.	3
1.1.2	CONSTITUCIÓN POLÍTICA DE LOS ESTADOS UNIDOS MEXICANOS.	4
1.1.3	LEY FEDERAL DEL TRABAJO.	6
1.1.4	LEY DEL ISR.	7
1.1.5	MISCELÁNEA FISCAL.	8
1.1.6	REGLAMENTO DEL ISR.	8
1.1.7	LEY DEL INFONAVIT.	9
1.1.8	ACUERDOS DEL HONORABLE CONSEJO DEL INFONAVIT	10
1.1.9	LEY DEL IMSS.	10
1.1.10	ACUERDOS DEL HONORABLE CONSEJO DEL IMSS.	12
1.1.11	CONTRATOS COLECTIVOS.	12
1.1.12	ACUERDOS SINDICALES.	12
1.1.13	REGLAMENTO INTERIOR DEL TRABAJO.	12
1.1.14	POLÍTICAS EMPRESARIALES.	13
1.1.15	PRACTICA COMÚN.	15

CAPITULO 2 TEORÍA DE SOFTWARE. 17

2.1	MODELO DE DATOS RELACIONAL.	17
2.1.1	ESTRUCTURA DE DATOS RELACIONAL.	17
2.1.1.1	Dominio.	17
2.1.1.2	Relación.	18
2.1.1.3	Propiedades de las relaciones.	19
2.1.1.4	Tipos de relaciones.	20
2.1.2	REGLAS DE INTEGRIDAD RELACIONAL.	21
2.1.2.1	Claves Primarias.	21
2.1.3	LA REGLA DE INTEGRIDAD DE LAS ENTIDADES.	23
2.1.4	LA REGLA DE INTEGRIDAD REFERENCIAL.	23
2.1.4.1	Claves Ajenas.	23
2.1.5	MANIPULACIÓN DE DATOS.	25
2.1.5.1	Algebra Relacional.	25
2.1.5.2	Calculo Relacional.	29
2.2	DISEÑO DE BASES DE DATOS RELACIONALES.	31
2.2.1	MODELO ENTIDAD RELACIÓN E/R.	31
2.2.2	NORMALIZACIÓN.	33
2.2.2.1	Dependencia Funcional.	34
2.2.2.2	Descomposición o Proyección.	35
2.2.2.3	Llave foránea.	35
2.2.2.4	Determinante.	35
2.2.2.5	Dependencia Transitiva.	36
2.2.3	FORMAS NORMALES.	36
2.2.3.1	Primera Forma Normal (1NF)	36

2.2.3.2	Segunda Forma Normal (2NF)	37
2.2.3.3	Tercera Forma Normal (3NF)	39
2.2.3.4	Forma Normal (Boyce/Codd)	39
2.2.3.5	Cuarta Forma Normal	40
2.2.3.6	Quinta Forma Normal	40
2.3	CONCEPTOS BÁSICOS DEL ANÁLISIS ESTRUCTURADO.	41
2.3.1	PROBLEMAS DEL ANÁLISIS.	43
2.3.1.1	Problemas de comunicación.	44
2.3.1.2	Cambio de los requerimientos.	45
2.3.1.3	La falta de herramientas.	45
2.3.1.4	Problemas con el Documento de Especificación.	45
2.3.1.5	El equipo de trabajo.	46
2.3.2	LA RELACIÓN ENTRE EL USUARIO / ANALISTA.	46
2.3.2.1	¿Quién es el usuario?	47
2.3.2.2	¿Quién es el analista?	47
2.3.2.3	División entre las responsabilidades del analista y el usuario.	48
2.3.3	¿QUÉ ES EL ANÁLISIS ESTRUCTURADO?	48
2.3.3.1	Diagramas de Flujo.	49
2.3.3.2	Diccionario de Datos.	52
2.3.3.3	Español Estructurado.	53
2.3.3.4	Tablas de decisión.	54
2.3.3.5	Arboles de Decisión.	54
2.4	INGENIERÍA DE SOFTWARE.	55
2.4.1	EL CICLO DE VIDA CLÁSICO.	56
2.4.2	CONSTRUCCIÓN DE PROTOTIPOS.	59
2.4.3	EL MODELO EN ESPIRAL.	60
2.4.4	TÉCNICAS DE CUARTA GENERACIÓN.	61
2.4.5	COMBINACIÓN DE PARADIGMAS.	62
2.4.6	VISIÓN GENÉRICA DE LA INGENIERÍA DEL SOFTWARE.	62
2.4.7	INGENIERÍA DE SOFTWARE Y DISEÑO DEL SOFTWARE.	64
2.4.7.1	Proceso de Diseño.	65
2.4.7.2	Fundamentos del diseño.	66

CAPITULO 3 HERRAMIENTAS DE DESARROLLO. **70**

3.1	DISEÑO DE WINDOWS NT.	70
3.1.1	LOS SUBSISTEMAS PROTEGIDOS.	72
3.1.1.1	El Subsistema Win32.	73
3.1.1.2	El subsistema POSIX.	73
3.1.1.3	El subsistema OS/2.	74
3.1.1.4	El subsistema Proceso de inicio (Logon Process).	74
3.1.1.5	El subsistema de Seguridad.	75
3.1.2	EL EXECUTIVE.	75
3.1.2.1	El Administrador de Objetos (Object Manager).	76
3.1.2.2	El Administrador de Procesos (Process Manager).	76
3.1.2.3	El Administrador de Memoria Virtual (Virtual Memory Manager).	77
3.1.2.4	Facilidad de Llamada a Procedimiento Local (LPC Facility).	77
3.1.2.5	Administrador de Entrada/Salida (I/O Manager).	77

3.1.2.6	Monitor de Referencias a Seguridad.	78
3.1.2.7	El Núcleo (Kernel).	78
3.1.2.8	Nivel de Abstracción de Hardware (HAL).	79
3.1.3	LLAMADAS A PROCEDIMIENTOS.	79
3.1.3.1	RPC (Remote Procedure Call).	79
3.1.3.2	LPC (Local Procedure Call).	80
3.1.4	PROCESOS.	82
3.1.5	ADMINISTRACIÓN DE LA MEMORIA.	83
3.1.5.1	Espacio de direcciones de un proceso.	83
3.1.5.2	Funcionamiento del VMM.	83
3.1.5.3	Archivos asignados en memoria.	86
3.1.5.4	Uso de memoria virtual por parte del programador.	88
3.1.5.5	Bloques (heaps).	88
3.1.6	SISTEMAS DE FICHEROS (ARCHIVOS).	89
3.1.7	ENTRADA Y SALIDA.	89
3.2	CONCEPTOS SOBRE REDES.	92
3.2.1	COMPONENTES DE UNA RED.	92
3.2.2	ARQUITECTURA DE LA RED.	95
3.2.2.1	Topología anillo.	95
3.2.2.2	Topología Bus.	96
3.2.2.3	Topología Árbol.	96
3.2.2.4	Topología Estrella.	97
3.2.2.5	Topología Malla.	97
3.2.3	COBERTURA DE LAS REDES.	98
3.2.3.1	Red de área Local (LAN).	99
3.2.3.2	Redes Interconectadas.	99
3.2.3.3	Red Metropolitana (MAN).	99
3.2.3.4	Red de gran alcance (WAN).	100
3.2.4	RAZONES PARA INSTALAR UNA RED DE COMPUTADORAS.	100
3.2.5	CARACTERÍSTICAS DE LOS SISTEMAS OPERATIVOS DE RED.	100
3.2.6	PROTOCOLO DE COMUNICACIÓN.	101
3.2.6.1	Nivel de Protocolo.	102
3.2.6.2	Paquetes de Información.	103
3.2.6.3	Jerarquía de protocolo OSI.	103
3.2.7	INTERCONEXIÓN DE REDES.	105
3.2.7.1	Repetidores.	106
3.2.7.2	Puentes.	107
3.2.7.3	Routers.	108
3.2.7.4	Switch.	108
3.2.7.5	Enlace Principal (Backbone).	109
3.2.7.6	FDDI Y ATM.	109
3.3	MICROSOFT ACCESS 97.	112
3.3.1	PROGRAMACIÓN CON ACCESS 97.	112
3.3.2	MEJORAR RENDIMIENTO.	117
3.3.3	TRABAJAR EN INTERNET O EN UNA INTRANET.	118
3.3.4	TRABAJAR CON OTRAS APLICACIONES	119
3.3.5	TRABAJAR CON DATOS	120
3.3.6	TRABAJAR CON FILTROS.	122
3.3.7	IMPRIMIR Y VISTA PRELIMINAR	123

CAPITULO 4 PLANTEAMIENTO DEL PROBLEMA Y PROPUESTA DE SOLUCIÓN. 124

4.1	SITUACIÓN ACTUAL.	124
4.1.1	ACTIVIDADES PRINCIPALES DE RECURSOS HUMANOS.	126
4.1.2	CARACTERÍSTICAS PRINCIPALES DEL SISTEMA DE NÓMINA HP-3000.	133
4.1.3	PROCESO ADMINISTRATIVO.	133
4.2	REQUERIMIENTOS DEL USUARIO.	136
4.3	RECOPIACIÓN Y ANÁLISIS DE LA INFORMACIÓN.	142
4.4	PLANTEAMIENTO DEL PROBLEMA.	163
4.4.1	MÓDULOS DEL SISTEMA.	163
4.4.2	CARACTERÍSTICAS DEL SISTEMA.	167
4.4.3	PROBLEMÁTICA POR ÁREAS.	168
4.5	DESCOMPOSICIÓN FUNCIONAL.	170
4.6	OPCIONES DE SOLUCIÓN.	176
4.7	ELECCIÓN DE SOLUCIÓN ÓPTIMA.	182

CAPITULO 5 DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA. 183

5.1	DESARROLLO DE LOS MÓDULOS DEL SISTEMA.	183
5.1.1	APLICACIÓN DE LA METODOLOGÍA.	183
5.1.2	DIAGRAMA DE CONTEXTO.	186
5.1.3	DIAGRAMAS DE FLUJO DE DATOS.	187
5.1.3.1	Generación del Salario Integrado.	187
5.1.3.2	Prestamos de Ahorro.	191
5.1.3.3	Tiempo Extra.	194
5.1.3.4	Deposito Bancario.	197
5.1.3.5	Aplicación Contable.	200
5.1.3.6	Procesa Evaluaciones.	204
5.1.3.7	Calculo de Gratificación Anual.	207
5.1.3.8	Liquidación de Ahorro.	210
5.1.3.9	Ajuste Anual de Impuestos.	213
5.1.3.10	Generación de una Nómina.	216
5.1.4	DICCIONARIO DE DATOS.	219
5.1.5	DIAGRAMA ENTIDAD RELACIÓN (DER).	246
5.1.6	NORMALIZACIÓN.	260
5.2	GENERACIÓN DE CÓDIGO PARA EL PROCESAMIENTO DE LA INFORMACIÓN	271
5.3	DISEÑO Y CONSTRUCCIÓN DEL SISTEMA.	302
5.3.1	DESARROLLO DE PANTALLAS	302
5.3.2	DESARROLLO DE INFORMES.	305
5.3.3	DESARROLLO DE CONSULTAS.	307
5.3.4	ADMINISTRACIÓN DE TABLAS.	308
5.3.5	ARQUITECTURA DEL SISTEMA.	309
5.4	DISEÑO E IMPLEMENTACIÓN DE RUTINAS DE DIAGNOSTICO Y EVALUACIÓN DE LOS DATOS.	311

5.4.1	EVALUACIÓN DE RUTAS INSTALADAS.	311
5.4.2	REPARACIÓN Y COMPACTACIÓN DE DATOS.	312
5.4.3	PRUEBAS PARA EL AÑO 2000.	313
5.4.4	EVALUACIÓN DEL RESULTADO DE UNA NÓMINA.	313
5.5	INTEGRACIÓN Y PRUEBAS DEL SISTEMA.	315
5.5.1	FLUJO DE INFORMACIÓN EN LA PRUEBA.	316
5.5.2	VERIFICACIÓN Y VALIDACIÓN.	317
5.5.3	APLICACIÓN DE PRUEBAS A SISTEMA.	318
5.6	FACTIBILIDAD TÉCNICA OPERATIVA.	321
5.6.1	REQUERIMIENTOS DE HARDWARE.	321
5.6.2	REQUERIMIENTOS DE INFORMACIÓN	322
5.6.3	CAMBIOS Y ADECUACIONES.	325
5.7	ANÁLISIS DEL COSTO Y VENTA DEL SISTEMA.	330
	CONCLUSIONES	332
	BIBLIOGRAFÍA	334
	MANUAL TÉCNICO	336
	MANUAL DEL USUARIO	341
	APENDICE 1 CÓDIGO FUENTE	395
	APENDICE 2 DICCIONARIO DE DATOS	421

CAPITULO 1 Reglas del Negocio.

1.1 Marco Legal que rige una nómina.

El elemento humano, tan vital hoy, como en cualquier época de la historia, es lo que verdaderamente da vida a un sistema empresarial. Es, por decirlo así, el factor dominante descrito como parte de los insumos que se incorporan al sistema de producción, tal como lo indica la figura 1.1-1.

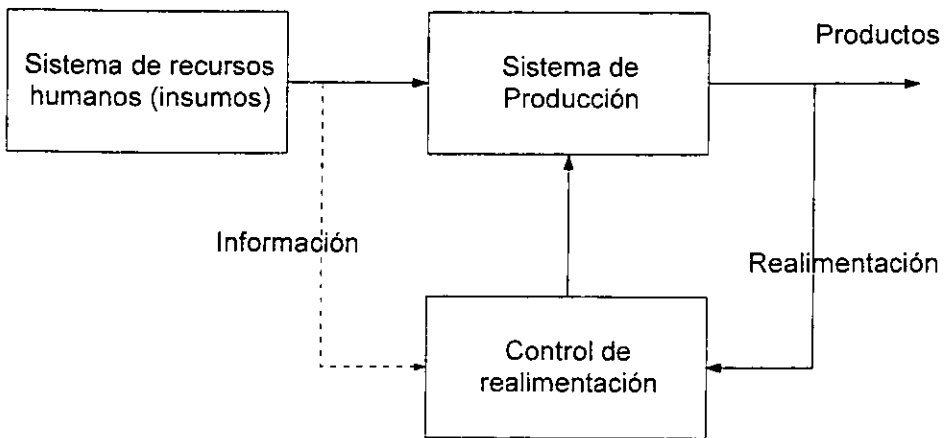


Figura 1.1- 1 Relación entre los sistemas de recursos humanos y producción.

Visto así, el sistema de producción recibe información de los insumos para construir una eficiencia de organización entre ambos sistemas, eficiencia que debe estar basada en la efectiva planeación y control de los objetivos, las políticas, los procedimientos y los programas que forman parte de la administración de personal.

La planeación y el control de la administración de personal, deben llevarse a cabo en función de la subdivisión clásica:

Objetivos: fundamentalmente, la obtención de la máxima eficiencia, el más alto grado de cooperación y el mayor bienestar posible del personal.

Políticas: reglas que reflejen el criterio de la empresa sobre problemas tales como adquisición de personal, salarios, incentivos, despidos, planes de retiro, indemnizaciones, servicios sociales, relaciones sindicales entre otros.

Procedimientos: sistemas técnicos sobre selección del personal, adiestramiento, capacitación de supervisores, análisis y valuación de puestos, calificación de méritos, medidas de seguridad e higiene entre otros.

Programas: planes de corto y largo plazo, tomando en consideración los problemas mas urgentes y de inmediata resolución, las posibilidades económicas de la empresa y la capacidad de expansión de las actividades del departamento especializado, incluyendo la organización y desarrollo de éste. Es decir, ambos sistemas requieren de funcionarios, trabajadores y empleados.

Respecto al personal operativo de una empresa, trabajadores y empleados se precisan los objetivos siguientes:

- Respeto a la dignidad humana.
- Los más altos salarios que permitan la estabilidad económica de la empresa y su normal desarrollo.
- Posibilidades de mejoramiento individual por méritos personales.
- Estabilidad en el empleo, orientaciones oportunas para mejorar su eficiencia y procurar la máxima integración de este personal a la empresa.

En lo que respecta a sus funcionarios, rigen los siguientes objetivos:

- Son aplicables, en lo que les concierne, los objetivos señalados en el punto anterior.

- La más alta competencia técnica y administrativa de acuerdo con las funciones básicas de los puestos que desempeñen
- La justa correspondencia entre la remuneración que perciban y la competencia demostrada en su trabajo.
- La máxima coordinación en sus actividades para acrecentar continuamente el trabajo en equipo.
- La responsabilidad solidaria en la realización de los objetivos generales de la empresa.

Para poder otorgar sueldos y prestaciones competitivas para la empresa y el trabajador, además de apegarse a lo establecido en la ley es necesario combinar y ajustar los cálculos a diferentes normas emitidas por el gobierno federal y estatal, para efectos de este trabajo se presentaran las leyes y una lista de elementos que contienen cada una para poder dar un panorama general del marco jurídico que rige una nómina.

1.1.1 Normatividad.

Los aspectos legales que rigen a una nómina están plasmados en las siguientes leyes

- Constitución Política de los Estados Unidos Mexicanos.
- Ley Federal del Trabajo.
- Ley de ISR.
- Miscelánea Fiscal.
- Reglamento del ISR.
- Ley del IMSS.
- Acuerdos del honorable consejo del IMSS
- Ley del Infonavit.
- Acuerdos del honorable consejo del Infonavit.

Otros aspectos son:

- Contratos Colectivos.
- Acuerdos Sindicalizados.
- Reglamento Interior de trabajo.
- Políticas de la empresa.
- Practica común.

1.1.2 Constitución Política de los Estados Unidos Mexicanos.

La constitución es la máxima ley que rige a México por lo que las demás leyes emanaran de esta y cualquiera que se contraponga quedará sin efecto alguno.

El artículo de la constitución que rige lo referente a las relaciones de trabajo es el 123 el cual se divide en dos partes fundamentales el párrafo A y el B.

TITULO SEXTO DEL TRABAJO Y DE LA PREVISION SOCIAL. Art. 123. Toda persona tiene derecho al trabajo digno y socialmente útil; al efecto, se promoverán la creación de empleos y la organización social para el trabajo, conforme a la Ley.

El Congreso de la Unión, sin contravenir a las bases siguientes, deberá expedir leyes sobre el trabajo, las cuales regirán:

A.- Entre los obreros, jornaleros, empleados domésticos, artesanos y de una manera general, todo contrato de trabajo

B.- Entre los Poderes de la Unión, el Gobierno del Distrito Federal y sus trabajadores.

Algunos de los rubros del párrafo A de este artículo se refieren a:

- Topes de jornadas y condiciones higiénicas de trabajo.
- Edad mínima para poder trabajar y jornadas permitidas para menores.
- Por cada 6 días de trabajo se otorgará uno de descanso.

- Restricción de algunos trabajos para mujeres embarazadas por poner en riesgo su salud.
- Definición y características de los salarios mínimos.
- Establece igualdad de remuneraciones sin distinción de sexo o nacionalidad.
- Participación de utilidades.
- Pago en efectivo y en moneda nacional.
- Tiempo extra.
- Obligación de la empresa a proporcionar vivienda; para lo cual se aportaran recursos al fondo nacional de la vivienda.
- La empresa deberá proporcionar capacitación y adiestramiento a su personal.
- Los empresarios serán responsables de los accidentes del trabajo y de las enfermedades profesionales de los trabajadores sufridas con motivo o en ejercicio de la profesión o trabajo que ejecuten.
- El patrón estará obligado a observar de acuerdo con la naturaleza de su negociación los preceptos legales sobre higiene y seguridad en las instalaciones propias de la empresa.
- Creación de sindicatos o asociaciones profesionales tanto de trabajadores como de patrones para defender sus derechos; así como el derecho a huelgas y paros.
- La huelga será lícita cuando tenga por objetivo conseguir el equilibrio entre los diversos factores de la producción.
- Los paros serán lícitos únicamente cuando el exceso de producción haga necesario suspender el trabajo para mantener los precios en un límite consteable.
- Establecer una junta de conciliación y arbitraje.
- Términos y causas por las que se podrá dar por terminada la relación laboral, así como las indemnizaciones correspondientes.
- Condiciones nulas y que no obligan a los contrayentes, aunque se expresen en el contrato.
- Creación de la ley del seguro social la cual comprenderá seguros de invalidez, de vejez de vida, de cesación involuntaria del trabajo, enfermedades, accidentes y guarderías.

- Sociedades cooperativas para la construcción de casa para ser adquiridas por los trabajadores en plazos determinados.
- La aplicación de estas leyes será en primer instancia estatal pero también será de competencia federal.

1.1.3 Ley Federal del Trabajo.

La presente ley es de observancia general en toda la república y rige la relaciones de trabajo comprendidas en el artículo 123, apartado A, de la Constitución.

Las normas de trabajo tienden a conseguir el equilibrio y la justicia en las relaciones entre trabajadores y patrones.

Se entiende por relación de trabajo, cualquiera que sea el acto que le dé origen, la prestación de un trabajo personal subordinado a una persona, mediante el pago de un salario. Se presume la existencia del contrato y de la relación de trabajo entre el que presta un trabajo personal y el que lo recibe.

Algunos puntos importantes que trata esta ley son:

- Suspensión de los efectos de las relaciones de trabajo
- Duración de las relaciones de trabajo.
- Rescisión y Terminación de las relaciones de trabajo.
- Jornada de trabajo.
- Días de descanso.
- Vacaciones.
- Normas protectoras y privilegios del salario.
- Participación de los trabajadores en las utilidades de las empresas.
- Obligaciones de los patrones.
- Obligaciones de los trabajadores.
- Habitaciones para los trabajadores.

- Derechos de preferencia, antigüedad y ascenso.
- Invecciones de los trabajadores.
- Trabajo de las mujeres.
- Trabajo de los menores.
- Trabajadores de confianza.
- Agentes de comercio.
- Relaciones colectivas de Trabajo.
- Coaliciones.
- Sindicatos, federaciones y confederaciones.
- Contrato colectivo de trabajo.
- Contrato Ley.
- Modificación colectiva de las condiciones de trabajo.
- Reglamento Interior de Trabajo.
- Suspensión colectiva de las relaciones de trabajo
- Terminación colectiva de las relaciones de trabajo.
- Objetivo y procedimientos de huelga.
- Riesgos de trabajo.
- Autoridades de Trabajo y Servicios sociales.
- Competencia constitucional de las autoridades del trabajo.
- Procuraduría de la defensa del trabajo.
- Del servicio nacional del empleo, capacitación y adiestramiento.
- Inspección del trabajo.
- Comisión nacional de los salarios mínimos.
- Representantes de los trabajadores y de los patrones.
- Derecho procesal del trabajo.

1.1.4 Ley del ISR.

Las personas físicas y las morales están obligadas al pago del impuesto sobre la renta en nuestro caso:

Los residentes en México respecto de todos sus ingresos cualquiera que sea la ubicación de la fuente de riqueza de donde procedan.

Se consideran ingresos por la prestación de un servicio personal subordinado (Ley de Impuesto Sobre la Renta LISR art. 78):

Los salarios y demás prestaciones que deriven de una relación laboral incluyendo la participación a los trabajadores en utilidades de las empresas y las prestaciones percibidas como consecuencia de la terminación de la relación laboral.

Para el calculo de impuesto se considerará como total de ingresos del trabajador el sueldo más prestaciones y todo tipo de ingresos; pero NO deben incluirse las aportaciones hechas por el patrón a las subcuentas del SAR, IMSS, INFONAVIT que establece la Ley del IMSS (LIMSS) y tampoco el crédito al salario (LISR Art. 77, 80-A y 80-B).

Estas cuentas no serán Ingresos Acumulables del trabajador en el ejercicio en que se aporten o generen, sino cuando se efectúen los retiros y entonces el ISR se pagará conforme al Art. 77-B (LISR Art. 77-A).

1.1.5 Miscelánea Fiscal.

La Secretaria de Hacienda con el fin de hacer del conocimiento publico los acuerdos, resoluciones o procedimientos del orden fiscal emite un documento que se publica en el Diario Oficial bajo el rubro de miscelánea fiscal.

1.1.6 Reglamento del ISR.

La parte del reglamento del ISR que interesa en este trabajo es el:

Título IV, capítulo I De los Ingresos por Salarios y en General por la Prestación de un Servicio Personal Subordinado, el cual trata los siguientes puntos:

- Ingresos por becas, ayuda, compensación para renta de casa y transporte.
- Total de percepciones por separación.
- Parte exenta de las pensiones o jubilaciones.
- Determinación del ingreso exento.
- Cálculo del ingreso gravable.
- Determinación de la parte acumulable a los demás ingresos.
- Determinación del impuesto de la parte no acumulable.
- Cálculo de la tasa aplicable.
- Retenciones por pago único de jubilaciones, pensiones y haberes de retiro.
- Cálculo de la retención.
- Pago provisional a cuenta del impuesto anual.
- Opción para retener por gratificación anual, PTU y primas
- Retención por previsión social gravada.
- Salario mínimo del área geográfica que corresponda.
- Opción para la retención por pagos de periodos de 7,10 ó 15 días.
- Opción de enteros mensuales.
- Procedimiento opcional para calcular las retenciones.
- Cálculo de pagos provisionales de sueldos acumulados.
- Compensación de saldos a favor de un trabajador con otro.
- Trabajadores obligados a presentar declaración anual.
- Opción de no presentar declaración anual.
- Obligación de expedir constancias.
- Datos del personal no obligado a pagar impuesto.

1.1.7 Ley del Infonavit.

La Constitución Política de los Estados Unidos Mexicanos, la Ley del Instituto del Fondo Nacional de la Vivienda para los Trabajadores, el Plan Nacional de Desarrollo, el Programa de Vivienda 1995-2000 y demás ordenamientos jurídicos y administrativos, constituyen el marco de referencia para definir que la Misión del INFONAVIT es:

"Consolidar los recursos del Fondo Nacional de la Vivienda con equidad social, responsabilidad financiera y cultura de servicio, para otorgar crédito barato y suficiente destinado a la vivienda de interés social y garantizar rendimientos competitivos a las cuentas individuales de los trabajadores".

El Instituto tiene por objeto:

1. Administrar los recursos del Fondo Nacional de la Vivienda;
2. Establecer y operar un sistema de financiamiento que permita a los trabajadores obtener crédito barato y suficiente para:
3. La adquisición en propiedad de habitaciones cómodas e higiénicas,
4. La construcción, reparación, ampliación o mejoramiento de sus habitaciones, y
5. El pago de pasivos contraídos por los conceptos anteriores;
6. Coordinar y financiar programas de construcción de habitaciones destinadas a ser adquiridas en propiedad por los trabajadores; y

Lo demás a que se refiere la Fracción XII del Apartado A del Artículo 123 constitucional y el Título Cuarto, Capítulo III de la Ley Federal del Trabajo, así como lo que esta ley establece._

1.1.8 Acuerdos del Honorable Consejo del Infonavit

El Honorable Consejo del Infonavit es una instancia cuya función es la de resolver controversias que se suscitan entre las partes involucradas, empresa, infonavit, trabajadores, así como emitir algunas resoluciones sobre aportaciones, prestamos y descuentos de infonavit.

1.1.9 Ley del IMSS.

La presente Ley es de observancia general en toda la República, en la forma y términos que la misma establece, sus disposiciones son de orden público y de interés social.

La Seguridad Social tiene por finalidad garantizar el derecho a la salud, la asistencia médica, la protección de los medios de subsistencia y los servicios sociales necesarios para el bienestar individual y colectivo, así como el otorgamiento de una pensión que, en su caso y previo cumplimiento de los requisitos legales, debe ser garantizada por el Estado.

La realización de la seguridad social está a cargo de entidades o dependencias públicas, federales o locales y de organismos descentralizados, conforme a lo dispuesto por esta Ley y demás ordenamientos legales sobre la materia.

Las disposiciones fiscales de esta Ley que establecen cargas a los particulares y las que señalan excepciones a las mismas, así como las que fijan las infracciones y sanciones, son de aplicación estricta. Se considera que establecen cargas las normas que se refieran a sujeto, objeto, base de cotización y tasa.

- Del Régimen Obligatorio.
- Del Seguro de Riesgos de Trabajo.
- Del Seguro de Enfermedades y Maternidad.
- Del Seguro de Invalidez y Vida.
- Del Seguro de Retiro, Cesantía en Edad Avanzada y Vejez.
- Del Seguro de Guarderías y de las Prestaciones Sociales.
- Del Régimen Voluntario.
- Del Instituto Mexicano del Seguro Social.
- De los Procedimientos, de la Caducidad y Prescripción.
- De las Responsabilidades y Sanciones.

1.1.10 Acuerdos del Honorable Consejo del IMSS.

El Honorable Consejo del IMSS es una instancia cuya función es la de resolver controversias que se suscitan entre las partes involucradas, empresa, IMSS, asegurados, así como emitir algunas resoluciones sobre aportaciones al IMSS.

1.1.11 Contratos Colectivos.

Contrato colectivo de trabajo es el convenio celebrado entre uno o varios sindicatos de trabajadores y uno o varios patrones o uno o varios sindicatos de patrones, con objeto de establecer las condiciones según las cuales debe prestarse el trabajo en una o más empresas o establecimientos.

1.1.12 Acuerdos Sindicales.

Los acuerdos sindicales son utilizados para establecer compromisos entre los trabajadores y la empresa con el fin de solucionar eventualidades no previstas en el contrato colectivo o por cuestiones extraordinarias que afecten las condiciones de trabajo. Un caso que se ha presentado por la inestabilidad económica del país ha sido el reducir los días de trabajo a cuenta de vacaciones con el fin de reducir la producción sin afectar la cantidad de trabajadores.

1.1.13 Reglamento Interior del Trabajo.

Reglamento interior de trabajo es el conjunto de disposiciones obligatorias para trabajadores y patrones en el desarrollo de los trabajos en una empresa o establecimiento.

No son materia del reglamento las normas de orden técnico y administrativo que formulen directamente las empresas para la ejecución de los trabajos.

El reglamento contendrá:

- Horas de entrada y salida de los trabajadores, tiempo destinado para las comidas y periodos de reposo durante la jornada.
- Lugar y momento que deben comenzar y terminar las jornadas de trabajo.
- Días y horas fijados para hacer la limpieza de los establecimientos, maquinaria, aparatos y útiles de trabajo.
- Días y lugares de pago.
- Normas para el uso de los asientos o sillas a que se refiere el artículo 132, fracción V de la LFT.
- Normas para prevenir los riesgos de trabajo e instrucciones para prestar los primeros auxilios.
- Labores insalubres y peligrosas que no deben desempeñar los menores y la protección que deben tener las trabajadoras embarazadas.
- Tiempo y forma en que los trabajadores deben someterse a los exámenes médicos, previos o periódicos, y a las medidas profilácticas que dicten las autoridades.
- Permisos y licencias.
- Disposiciones disciplinarias y procedimientos para su aplicación. La suspensión en el trabajo, como medida disciplinaria, no podrá exceder de ocho días. El trabajador tendrá derecho a ser oído antes de que aplique la sanción
- Las demás normas necesarias y convenientes de acuerdo con la naturaleza de cada empresa o establecimiento, para conseguir la mayor seguridad y regularidad en el desarrollo del trabajo.

1.1.14 Políticas Empresariales.

Las empresas tienen la necesidad de encontrar un balance entre el costo que implica pagar sueldos, salarios y prestaciones a los trabajadores; y la necesidad que tiene todo ser humano de recibir una justa retribución por su esfuerzo y su tiempo dedicado a trabajar en una empresa.

La empresa mexicana moderna busca este balance a través de establecer tres objetivos básicos en la administración de sueldos y salarios:

- Encontrar equidad interna, es decir, asegurarse que lo se paga refleje las diferencias en valor de los diferentes puestos dentro de la empresa.
- Encontrar competitividad, o sea, pagar de acuerdo a lo que se paga en el mercado.
- Diferenciar los niveles de actuación entre personas que ocupan puestos de similar valor, es decir, pagar de acuerdo al desempeño de los diferentes trabajadores.

Para lograr estos objetivos, existen tres elementos básicos, de los cuales se desprenden cinco técnicas que son indispensables en la administración de compensación moderna.

El primer elemento, la búsqueda de equidad interna, se logra aplicando la técnica conocida como Análisis y Valuación de Puestos. Esta técnica consiste en analizar funciones, responsabilidades, complejidades y requisitos de los diferentes puestos, y después, sistemáticamente valorar los diferentes puestos de la empresa, utilizando criterios consistentes para determinar el valor relativo de los puestos entre sí.

El segundo elemento, la competitividad, se refiere al estudio de lo que hacen otras empresas con las que compiten por atraer y retener al mejor personal de acuerdo a las necesidades. La técnica que se utiliza son las Encuestas de Compensación, las cuales son investigaciones de lo que se paga en sueldo y prestaciones en el mercado, así como de algunas prácticas de administración de remuneraciones (tipo de aumentos que se dan, cuándo, si tienen tabulador, etc.)

Una vez que se hubiera definido cuál es el valor relativo de los puestos, y que se sabe lo que paga la competencia, se puede desarrollar la tercera técnica, es decir, la elaboración de un tabulador de sueldos. El tabulador agrupa puestos de similar valor en

niveles de sueldo, los cuales generalmente establecen rangos de pagos mínimos y máximos a pagar a todos los puestos del mismo nivel.

Para cumplir con el pago por desempeño, es necesario establecer un enfoque sistemático en la Evaluación de Desempeño, que viene siendo la cuarta técnica utilizada. Estos programas generalmente establecen la manera de valuar resultados del trabajo y la forma de comunicar la valuaciones al personal afectado.

Con las cuatro técnicas ya mencionadas, se pueden establecer objetivos de sueldos, donde se relacionen el desempeño y la posición relativa del sueldo con respecto al tabulador, así como las estrategias necesarias para otorgar aumentos que aseguren que los sueldos pagados sean equitativos, competitivos y se relacionen con el desempeño individual de cada quien. Esto generalmente se logra utilizando la quinta técnica, que consiste en desarrollar una Matriz de Aumentos la cual indica cuánto y cuándo se deben otorgar aumentos de acuerdo a los objetivos salariales establecidos con anterioridad.

Una de las inquietudes mas comunes de las empresas es la de pagar prestaciones a sus empleados, las cuales no sean gravadas por impuestos, para que el personal goce de sus beneficios, sin que se les retengan impuestos, y que a la vez sean perfectamente deducibles para la empresa.

1.1.15 Practica Común.

La nómina como tal esta basada en preceptos legales los cuales están sujetos a interpretación lo que provoca que existan diferentes criterios para aplicarse, muchos de estos criterios están sustentados por jurisprudencias pero en la mayoría de los casos se fundamentan por la practica común ya sea por que el calculo es mas sencillo o por que los sistemas antiguos no podian hacer otro tipo de calculo, sin embargo aunque pudieran existir diferencias ninguna puede contravenir la ley o hacer un calculo inferior a lo estipulado en la ley, por esto cualquier sistema que se utilice deberá ser lo mas

CAPITULO 2 TEORÍA DE SOFTWARE.

2.1 Modelo de datos Relacional.

En 1970 E.F. Codd introdujo el Modelo de Datos Relacional, el cual revolucionó la forma en como se conceptualizaban las bases de datos, ya que propuso un modelo que permitía al usuario despreocuparse en gran medida de la estructura física de los datos. Codd también propuso dos lenguajes de manipulación de datos que permitían manipular y procesar la información de una manera más efectiva. Estos lenguajes son el álgebra y cálculo relacional los cuales se convirtieron en la base para los lenguajes relacionales usados por la mayoría de los Sistemas Administradores de Bases de Datos actuales.

"El Modelo de Datos Relacional organiza y presenta los datos en forma de relaciones. El termino de relación es un concepto matemático y representa lo que podríamos llamar una tabla de dos dimensiones consistente en columnas y renglones".

El modelo relacional se divide en tres partes, las cuales se ocupan de la estructura, la integridad y la manipulación de los datos.

2.1.1 Estructura de Datos Relacional.

2.1.1.1 Dominio.

Los valores escalares representan "la menor unidad semántica de la información en el sentido que son atómicos"; no poseen estructura interna (es decir no se pueden descomponer). Cabe subrayar que la carencia de estructura interna desde el punto de vista del modelo no implica la falta de estructura interna en términos absolutos. Por ejemplo, un nombre de ciudad sin duda tiene una estructura interna (está formado por una secuencia de letras); sin embargo, si se descompone un nombre de ciudad en sus letras componentes, se pierde el significado. El significado sólo se hace evidente si las letras aparecen todas juntas y en la secuencia correcta.

Se puede definir a un dominio como:

"Un conjunto de valores escalares, todos del mismo tipo, de los cuales uno o más atributos obtienen sus valores reales".

2.1.1.2 Relación.

Una relación (digamos R) sobre un conjunto de dominios D_1, D_2, \dots, D_n (no necesariamente distintos), se compone de dos partes, una cabecera y un cuerpo.

La **cabecera** está formada por un conjunto fijo de atributos, o, en términos más precisos, de pares de atributo-dominio.

$$\{ (A_1:D_1), (A_2:D_2), \dots, (A_n:D_n) \}$$

tales que cada atributo A_j corresponde a uno y sólo uno de los dominios subyacentes

$$D_j \quad (j=1, 2, \dots, n)$$

El **cuerpo** está formado por un conjunto de tuplas, el cual varía con el tiempo. Cada **tupla** a su vez está formada por un conjunto de parejas atributo-valor.

$$\{ (A_1:v_{i1}), (A_2:v_{i2}), \dots, (A_n:v_{in}) \}$$

($i = 1, 2, \dots, m$ donde m es el número de tuplas del conjunto). En cada una de estas tuplas hay uno de estos pares atributo-valor ($A_j:v_{ij}$) para cada atributo A_j de la cabecera. Para cada par atributo-valor ($A_j:v_{ij}$), v_{ij} es un valor del dominio único D_j asociado al atributo A_j .

Los valores m y n se llaman **cardinalidad** y **grado**, respectivamente, de la relación R. La cardinalidad varía con el tiempo, pero el grado no.

Una relación es lo que se ha dado en llamar como tabla, sin embargo, en términos formales una relación y una tabla no son la misma cosa. Más bien una relación es lo que indica la definición del párrafo anterior, una especie bastante abstracta del objeto; y una tabla es una representación concreta (casi siempre en papel) de tal objeto abstracto. Repitiendo: no son lo mismo. Por supuesto, son bastante

parecidas... y en contextos informales, por lo menos, es usual. Sin embargo hemos de reconocer que los dos conceptos no son idénticos.

2.1.1.3 Propiedades de las relaciones.

Las relaciones poseen ciertas propiedades, todas ellas consecuencia inmediata de la definición formal de relación, y todas ellas muy importantes las cuales son:

- No existen tuplas repetidas
- Las tuplas no están ordenadas (de arriba hacia abajo)
- Los atributos no están ordenados (de izquierda a derecha)
- Todos los valores de los atributos son atómicos

- No existen tuplas repetidas.

Esta propiedad es consecuencia del hecho de que el cuerpo de la relación es un conjunto matemático (es decir, un conjunto de tuplas), en matemáticas los conjuntos por definición no incluyen elementos repetidos.

- Las tuplas no están ordenadas (de arriba hacia abajo)

- Esta propiedad también se desprende del hecho de que el cuerpo de una relación es un conjunto matemático. Los conjuntos en matemáticas no son ordenados. Esta propiedad servirá también para ilustrar la diferencia entre una relación y una tabla, porque las filas de una tabla tienen un orden obvio de arriba hacia abajo, en tanto que las tuplas de la relación carecen de tal orden.

- Los atributos no están ordenados (de izquierda a derecha).

Esta propiedad se desprende del hecho de que la cabecera de la relación se define también como conjunto (es decir, un conjunto de atributos, dicho en forma más precisa, de pares atributo-dominio). Esta cuestión de ordenamiento de los atributos es otra área en la cual la representación concreta de una relación en forma de tabla sugiere algo que no se cumple en realidad: las

columnas de una tabla tienen un orden evidente de izquierda a derecha pero los atributos de una relación carecen de tal orden.

- Todos los valores de los atributos son atómicos.

Una forma más precisa de expresar esta última propiedad es: "todos los valores de los atributos simples son atómicos". Se trata desde luego, de una consecuencia del hecho de que todos los dominios subyacentes son a su vez simple; es decir, contienen sólo valores atómicos. (Aún si existen atributos compuestos, éstos no son sino una simple concatenación de atributos simples); en resumen las relaciones no contienen grupos repetitivos.

2.1.1.4 Tipos de relaciones.

A continuación se presentan muy brevemente los tipos de relaciones que puede contener un sistema relacional.

- Relaciones Base.

Las relaciones base son aquellas cuya importancia (para la aplicación en cuestión) es tal que el diseñador de la base de datos ha decidido darles un nombre y hacerlas parte directa de la base de datos en sí, a diferencia de otras relaciones cuya naturaleza es más efímera, como por ejemplo el resultado de una consulta.

- Vistas.

Son también llamadas relaciones virtuales, una vista es una relación derivada, con nombre, representada dentro del sistema exclusivamente mediante su definición en términos de otras relaciones con nombre; no posee datos almacenados propios, separados y distinguibles (a diferencia de las relaciones base)

- Instantáneas.

Una instantánea (snapshot) es también una relación derivada, con nombre, como una vista. Pero a diferencia de las vistas, las instantáneas son reales, no virtuales; es decir, están representadas no sólo por su definición en términos de otras relaciones con nombre, sino también por sus propios datos almacenados.

- Resultados intermedios.

Un resultado intermedio es una relación (casi siempre sin nombre) resultante de alguna expresión relación anidada dentro de alguna otra expresión relacional más grande.

- Relaciones temporales.

Es una relación temporal es una relación con nombre, similar a una relación base o visita o instantánea, pero (a diferencia de estas tres últimas) se destruye en forma automática en algún momento apropiado. Las relaciones base, vistas e instantáneas, en cambio son más permanentes, en cuanto a que sólo se destruyen como resultado de alguna acción explícita del usuario.

-

2.1.2 Reglas de integridad Relacional.

El modelo relacional incluye dos reglas generales de integridad: generales en el sentido de que se aplican no sólo a una base de datos específica, sino más bien a todas las bases de datos (o por lo menos a todas las bases de datos que digan apegarse al modelo). Estas dos reglas generales se refieren, respectivamente, a las **claves primarias** y a las **claves ajenas**.

2.1.2.1 Claves Primarias.

Primero es necesario definir el término de **clave candidata** de la siguiente manera. El atributo K (posiblemente compuesto) de la relación R es una clave

candidata de R si y sólo si satisface las siguientes dos propiedades, independientes del tiempo:

- **Unicidad:** En cualquier momento dado, no existen dos tuplas en R con el mismo valor de K.
- **Minimalidad :** Si K es compuesto, no será posible eliminar ningún componente de K sin destruir la propiedad de unicidad.

Es importante señalar que toda relación tiene por lo menos una clave candidata, por que las relaciones no contienen tuplas repetidas. En la práctica, las relaciones tienden a tener una y sólo una clave candidata, pero sin duda es posible que tengan más. Del conjunto de claves candidatas de una relación dada, se elige una y solo una como clave primaria de esa relación; las demás, si existen, se llamarán **claves alternativas**. Así, una clave alterna es una clave candidata que no es la clave primaria.

La importancia de las claves primarias radica principalmente en que constituyen el mecanismo de direccionamiento a nivel tupla básico en un sistema relacional. El único modo, garantizado por el sistema, de localizar alguna tupla específica es por el valor de su clave primaria. En consecuencia las claves primarias son tan indispensables para el funcionamiento exitoso de un sistema relacional como las direcciones de memoria principal lo son para el funcionamiento exitoso de la computadora".

En términos informales:

"Una clave primaria es un conjunto de atributos que fueron designados para identificar plenamente a cada tupla de la relación, su característica principal es que no acepta en su conjunto valores duplicados."

2.1.3 La regla de integridad de las entidades.

"Ningún componente de la clave primaria de una relación base puede aceptar nulos"¹

La justificación de esta regla radica en que las tuplas dentro de las relaciones base corresponden a entidades en el mundo real y por definición estas entidades dentro de la base de datos deben ser distinguibles, es decir, se les puede identificar de alguna manera. Las claves primarias realizan esta función de identificación única en el modelo relacional, en este sentido si la clave primaria contiene valores nulos es evidente que esa tupla no tiene sentido. Resumiendo este punto de la siguiente manera:

" En una base de datos relacional, nunca se registra información acerca de algo que no podamos identificar ".

2.1.4 La regla de integridad referencial.

2.1.4.1 Claves Ajenas.

"Una clave ajena es un atributo (quizá compuesto) de una relación R2 cuyos valores deben concordar con los de la clave primaria de alguna relación R1 (donde R1 y R2 no necesariamente son distintos)".

Un valor de clave ajena representa una referencia a la tupla donde se encuentra el valor correspondiente de la clave primaria (la tupla referida o tupla objetivo). Por lo tanto, el problema de garantizar que la base de datos no incluya valores no válidos de una clave ajena se conoce como el problema de la integridad referencial. La restricción según la cual los valores de una clave ajena determinada deben concordar con los valores de la clave primaria correspondiente se conoce como restricción referencial. La relación que contiene a la clave ajena se conoce como relación referencial y la relación

¹ Con nulos se quiere decir información faltante por alguna razón.

que contiene a la clave primaria correspondiente de denomina relación referida o relación objetivo.

Se puede definir a la clave ajena de una manera mas formal como:

El atributo LF (quizá compuesto) de la relación base R2 es una clave ajena si y sólo si satisface estas dos propiedades independientes del tiempo:

- Cada valor de LF es nulo del todo o bien no nulo del todo (con "nulo del todo" o "no nulo del todo" queremos decir que, si LF es compuesto, todos sus componentes son nulos o bien todos sus componentes son no nulos no una combinación.
- Existe una relación base R1 con clave primaria LP tal que cada valor no nulo de LF es idéntico al valor de LP en alguna tupla de R1.

Para poder entender el concepto es importante considerar los siguientes puntos:

- Una clave ajena dada y la clave primaria correspondiente deben definirse sobre el mismo dominio (el cual puede ser compuesto desde luego).
- La clave ajena no necesita ser un componente de la clave primaria de la relación que la contine, de hecho cualquier atributo (en una relación base) puede ser una clave ajena.
- Una relación dada puede ser desde luego tanto una relación referida como una relación referencial.
- Una relación podría incluir una clave ajena cuyos valores (no nulos) deben concordar con los valores de la clave primaria de esa misma relación, a este tipo especial de relaciones se les denomina relaciones autoreferenciales.
- Las claves ajenas, a diferencia de la claves primarias (en relaciones base), deben aceptar nulos en ocasiones.
- Las concordancias de la clave ajena con la clave primaria representan ciertas interrelaciones entre las tuplas.

La Regla de la Integridad Referencial se puede resumir como sigue:

"La base de datos no debe contener valores de clave ajena sin concordancia"

El término "valores de clave ajena sin concordancia" se refiere a valores no nulos de la clave ajena para el cual no existe un valor concordante de la clave primaria en la relación objetivo pertinente.

2.1.5 Manipulación de Datos.

2.1.5.1 Algebra Relacional.

La tercera y última parte del modelo relacional, la parte manipulativa, se divide en dos partes:

1. Un conjunto de operadores, como los de reunión que forman en conjunto la llamada álgebra relacional.
2. Una operación de asignación (por ejemplo, "C:= A reunión B) que asigna el valor de alguna expresión arbitraria del álgebra a una relación nombrada.

"El álgebra relacional consiste en un conjunto de operadores de alto nivel que operan sobre relaciones. Cada uno de estos operadores toma una o dos relaciones como entrada y produce una nueva relación como salida."

Codd definió un conjunto muy específico de ocho operadores de este tipo, en dos grupos de cuatro cada uno:

1. Las operaciones tradicionales de conjuntos **unión**, **intersección**, **diferencia** y **producto cartesiano** (todas estas con ligeras modificaciones debidas al hecho de tener relaciones con operandos, y no conjuntos arbitrarios; después de todo, una relación es un tipo especial de conjunto).

2. Las operaciones relacionales especiales **restricción**, **proyección**, **reunión** y **división**.

A continuación se explica brevemente cada uno de estos operadores:

1. **RESTRICCIÓN** o **SELECCIÓN**: Extrae las tuplas específicas de una relación dada (es decir, restringe la relación sólo a las tuplas que satisfagan una condición específica), ver figura 2.1-1.

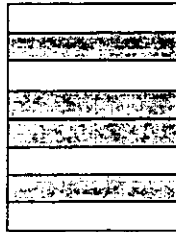


Figura 2.1- 1 Restricción.

2. **PROYECCIÓN**: Extrae los atributos específicos de una relación dada, ver la figura 2.1-2.

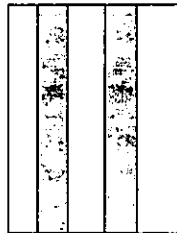


Figura 2.1- 2 Proyección

3. **PRODUCTO:** A partir de dos relaciones específicas, construye una relación que contiene todas las combinaciones posibles de tuplas, una de cada una de las dos relaciones, ver figura 2.1-3.

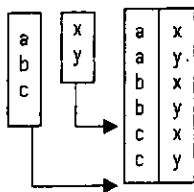


Figura 2.1- 3 Producto.

4. **UNIÓN:** Construye una relación formada por todas las tuplas que aparecen en cualquiera de las dos relaciones especificadas, ver figura 2.1-4(a).
5. **INTERSECCIÓN:** Construye una relación formada por todas las tuplas que aparecen en las dos relaciones especificadas, ver figura 2.1-4(b).
6. **DIFERENCIA:** Construye una relación formada por todas las tuplas de la primera relación que no aparezca en la segunda de las dos relaciones especificadas, ver figura 2.1-4(c).

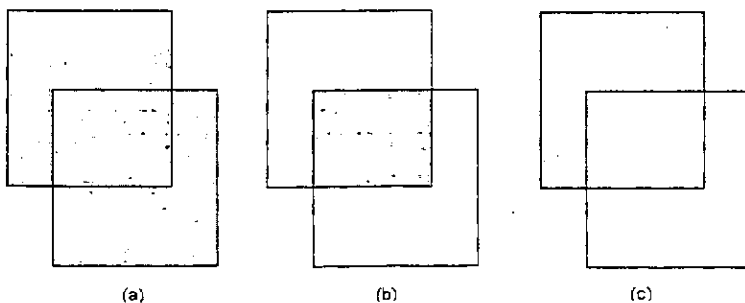


Figura 2.1- 4 (a)Unión, (b)Intersección, (c) Diferencia.

7. REUNIÓN: A partir de dos relaciones especificadas, construye una relación que contiene todas las posibles combinaciones de tuplas, una de cada una de las dos relaciones, tales que las dos tuplas participantes en una combinación dada satisfaga alguna condición especificada, ver figura 2.1-5.

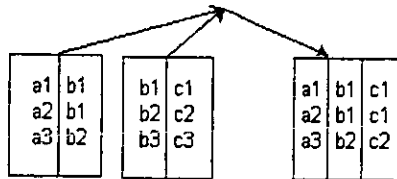


Figura 2.1- 5 Reunión.

8. DIVISIÓN: Toma dos relaciones, una binaria y una unaria, y construye una relación formada por todos los valores de un atributo de la relación binaria que concuerdan (en el otro atributo) con todos los valores en la relación unaria, ver figura 2.1-6.

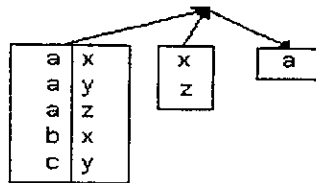


Figura 2.1- 6 División.

El propósito de la operación de asignación es el de poder recordar el valor de alguna expresión algebraica, y así modificar el estado de la base de datos.

2.1.5.2 Cálculo Relacional.

El álgebra relacional y el cálculo relacional son dos alternativas para establecer una base formal de la parte manipulativa del modelo. La diferencia entre ellas es la siguiente: mientras que el álgebra ofrece un conjunto de operaciones explícitas reunión, unión, proyección que pueden servir en la práctica para indicar al sistema la forma de construir alguna relación deseada a partir de las relaciones dadas en la base de datos, el cálculo sólo ofrece una notación para formular la definición de esa relación deseada en términos de esas relaciones dadas.

La formulación del cálculo es descriptiva donde la algebraica es prescriptiva; el cálculo sólo plantea el problema, el álgebra proporciona un procedimiento para resolver ese problema. O bien, en términos muy informales: el álgebra es de procedimientos (de alto nivel, pero de todos modos de procedimientos); el cálculo no es de procedimientos.

Se debe subrayar que tales distinciones son superficiales. En realidad, el álgebra y el cálculo son totalmente equivalentes. Para cada expresión del álgebra, existe una expresión equivalente en el cálculo; de manera similar, para cada expresión del cálculo, existe una expresión equivalente en el álgebra. Hay una correspondencia de uno a uno entre los dos. La diferencia de formalismos representa sólo diferentes estilos de expresión. El cálculo se acerca más al lenguaje natural, aunque esto es discutible, el álgebra quizá sea más parecida a un lenguaje de programación. El cálculo relacional se fundamenta en una rama de la lógica matemática llamada cálculo de predicados.

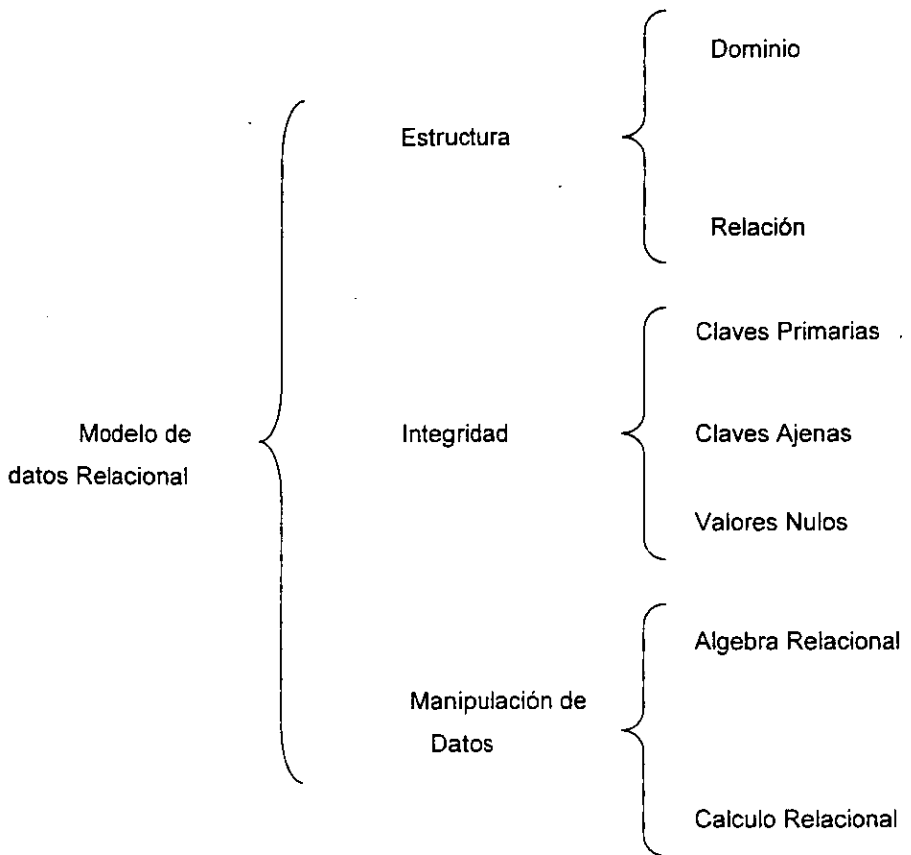


Figura 2.1- 7 Esquema del Modelo Relacional.

2.2 Diseño de Bases de Datos Relacionales.

El problema de diseñar una base de datos se puede expresar de manera muy sencilla, así: dado algún conjunto de datos que se deben representar en una base de datos, ¿cómo decidir cuál es la estructura lógica adecuada para esos datos? En otras palabras, ¿cómo decidir cuáles relaciones deberán existir y qué atributos deberán tener?.

En este punto es de mayor interés el problema del diseño lógico, no el de diseño físico. Ahora bien, no se intenta sugerir con este comentario que el diseño físico carece de importancia; todo lo contrario, el diseño físico es muy importante.

El presente capítulo se tratará lo que podría llamarse diseño independiente de las aplicaciones. Dicho de otro modo se tratará de diseñar el esquema conceptual; es decir, producir un diseño lógico abstracto independiente del equipo, independiente del sistema operativo, independiente lo mayor posible del DBMS, independiente del lenguaje.

2.2.1 Modelo Entidad Relación E/R.

Este modelo fue introducido por Peter Pin-Shan Chen en 1976 en una publicación titulada "El modelo entidad relación – hacia una visión de datos unificada".

El modelo Entidad Relacional (E/R) surge como una herramienta para el diseño de bases de datos el cual está basado en el Modelo Relacional pero con mayor riqueza semántica que permite una mejor comunicación entre los analistas, diseñadores y usuarios finales durante las fases de análisis de requerimientos y de diseño conceptual debido a que es simple y fácil de entender. Además de incorporar una sintaxis gráfica.

* ACM Transactions on DataBase Systems, Vol.1, No.1, Marzo 1976, Pag. 9-36

2.2.1.1.1 ELEMENTOS DEL MODELO E/R.

El Modelo Entidad Relación (E/R) se basa en una percepción de un mundo real que consiste en un conjunto de objetos llamados entidades y relaciones.

Entidades

Una entidad es una "cosa" que se puede distinguir claramente. Las entidades se pueden clasificar en diferentes tipo de entidades, como Empleado o Departamento. En un Diagrama E/R (DER) las entidades se representan como se muestra en la figura 2.2-1 Un sustantivo en español corresponde al nombre de la entidad en el DER.

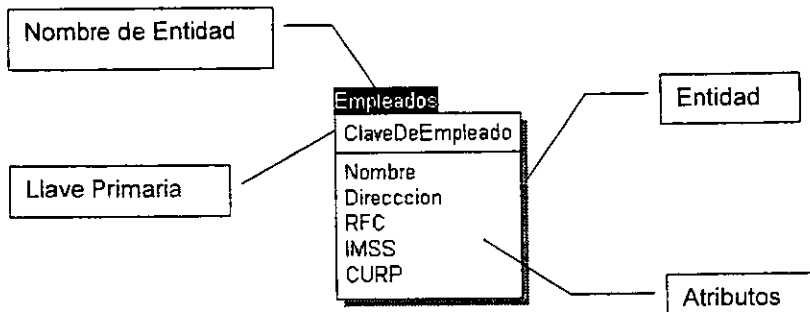


Figura 2.2- 1 Elementos de una entidad.

Relaciones

La conectividad de una relación especifica el tipo de asociación de las ocurrencias de las entidades de la relación. Los valores de la conectividad son "unos" o "muchos". El número real así asociado en el término "muchos" es llamado cardinalidad de la conectividad. Los tipos básicos de conectividad son los que se muestran en la figura 2.2-2:

Generalización/Especialización de relaciones (ISA).

Ocurre una generalización/especialización cuando una entidad se parte por diferentes valores de su atributo en común. Por ejemplo Sindicalizado y No Sindicalizado es una especialización de Empleados.

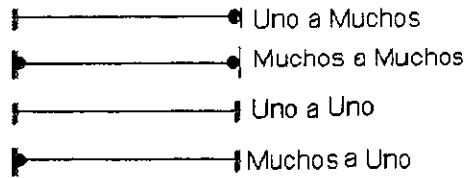


Figura 2.2- 2 Tipos de Relaciones.

Ambos comparten atributos comunes como su código, nombre, rfc entre otros, pero se distinguen por su tratamiento en la asignación de sueldo, este tipo de situaciones se representa en la figura 2.2-3.

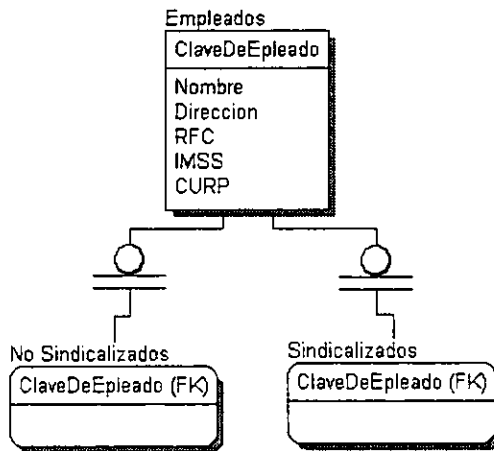


Figura 2.2- 3 Relaciones tipo ISA.

2.2.2 Normalización.

Las relaciones en una base de datos relacional siempre están normalizadas. En el sentido de que los dominios simples subyacentes contienen sólo valores atómicos (es decir relación sin grupos repetitivos), sin embargo es que una relación dada, aun cuando pudiera estar normalizada, podría poseer todavía ciertas propiedades

indeseables. La teoría de la normalización nos permite reconocer esos casos y nos muestra cómo podemos convertir esas relaciones a una forma más deseable.

La teoría de la normalización tiene como fundamento el concepto de formas normales. Se dice que una relación está en una determinada forma normal si satisface un cierto conjunto de restricciones.

Se ha definido un gran número de formas normales. Originalmente Codd definió la primera, segunda y tercera formas normales (1NF,2NF,3NF), todas las relaciones normalizadas están en 1NF; algunas relaciones 1NF están también en 2NF; algunas relaciones 2NF están también en 3NF. El motivo de las definiciones de Codd era que 2NF era "más deseable" que 1NF, y 3NF a su vez era más deseable que 2NF. Es decir, el diseñador de una base de datos en general debe tratar de lograr un diseño con relaciones en 3NF, no relaciones que estén sólo en 2NF o 1NF.

El llamado procedimiento de normalización (en términos más precisos, procedimiento de normalización adicional), se puede definir como la reducción sucesiva de un conjunto dado de relaciones a una forma más deseable, este procedimiento es reversible es decir se puede convertir un conjunto de relaciones en 3NF a un conjunto en 2NF original, esto garantiza que durante el proceso no se pierde información.

2.2.2.1 Dependencia Funcional.

Es importante definir el concepto de dependencia funcional ya que es el que sienta las bases para las tres formas normales de Codd y la forma normal de Boyce/Codd.

Dada una relación R, el atributo Y de R depende funcionalmente del atributo X de R si y sólo si siempre que dos tuplas de R concuerden en su valor de X, deben por fuerza concordar en su valor de Y.

En símbolos:

$$R.X \rightarrow R.Y$$

La cual se lee " R.X determina funcionalmente a R.Y" si y solo si un solo valor Y en R está asociado a cada valor X en R (en cualquier momento dado, los atributos X y Y pueden ser compuestos)"

Debe entenderse que la dependencia funcional es un concepto semántico. Reconocer las dependencias funcionales es parte del proceso de entender que significan los datos.

2.2.2.2 Descomposición o Proyección.

Es un proceso de dividir relaciones en múltiples relaciones para eliminar anomalías y mantener la integridad.

El primer paso en el procedimiento de normalización es sacar proyecciones para eliminar la dependencias funcionales no completas.

El proceso de reducción consiste en sustituir la relación en 1NF por proyecciones apropiadas; el conjunto de proyecciones así obtenido es equivalente a la relación original. En el sentido de que la relación original siempre se podrá recuperar efectuando la reunión (natural) de esas proyecciones, de manera que se pierde información con la reducción.

2.2.2.3 Llave foránea.

Un conjunto de atributos en una relación que constituye una llave en alguna relación; usada para indicar ligas lógicas entre relaciones.

2.2.2.4 Determinante.

Los atributos del lado izquierdo de una dependencia funcional; determina los valores de los demás atributos.

2.2.2.5 Dependencia Transitiva.

Ocurre cuando un atributo no-llave es funcionalmente dependiente de uno o más atributos no-llave.

2.2.3 Formas Normales.

2.2.3.1 Primera Forma Normal (1NF)

Una relación esta en primera forma normal (1NF) si y solo si todos los dominios simples subyacentes contienen sólo valores atómicos.

Por ejemplo en la figura 2.2-4 muestra una parte de una tabla PlantillaDePersonal en donde el campo ClaveDeEmpleado es la llave primaria, como se puede ver el sueldo y la FechaCambio se repiten ya que se mantiene un histórico, esto se contrapone con la 1NF por lo que se aplica una descomposición sobre esta tabla para generar dos tablas, en la figura 2.2-5 se muestra el resultado, la tabla DatosDePersonal mantiene como llave primaria el campo ClaveDeEmpleado y la tabla HistoricoDeSueldo la llave primaria se convierte en compuesta formada por ClaveDeEmpleado y FechaCambio si se realiza una reunión de ambas tablas (tomando como condición de reunión el campo ClaveDeEmpleado de ambas) se puede recuperar la tabla original sin perdida de datos.

ClaveDeEmpleado	Nombre	NombreDePuesto	FechaCambio	Sueldo
11767	CORONA REYES MATEO	Tecnico De Laboratorio	01/01/1997	\$124.25
			01/03/1997	\$139.15
			01/07/1997	\$154.00
12120	MARIN GARCIA SALOMON	Auxiliar De Cobranza	01/03/1997	\$198.19
			01/03/1999	\$282.23
12971	LOPEZ LOPEZ JUANA	Analista COMERCIAL	01/01/1997	\$170.44
			01/03/1997	\$192.60
			01/07/1997	\$200.77
			01/09/1997	\$222.85
13334	ARVIZU RIOS JUAN JOSE	Jefe De Laboratorio	01/03/1998	\$245.14
			01/01/1997	\$300.61
			01/03/1997	\$336.68

Figura 2.2- 4 Plantilla de Personal.

DatosDePersonal

ClaveDeEmpleado	Nombre	NombreDePuesto
11767	CORONA REYES MATEO	Tecnico De Laboratorio
12120	MARIN GARCIA SALOMON	Auxiliar De Cobranza
12971	LOPEZ LOPEZ JUANA	Analista COMERCIAL
13334	ARVIZU RIOS JUAN JOSE	Jefe De Laboratorio

HistoricoDeSueldo

ClaveDeEmpleado	FechaCambio	Sueldo
11767	01/01/1997	\$124.25
11767	01/03/1997	\$139.15
11767	01/07/1997	\$154.00
12120	01/01/1997	\$176.96
12120	01/03/1997	\$198.19
12120	01/03/1999	\$282.23
12971	01/01/1997	\$170.44
12971	01/03/1997	\$192.60
12971	01/07/1997	\$200.77
12971	01/09/1997	\$222.85
12971	01/03/1998	\$245.14
13334	01/01/1997	\$300.61
13334	01/03/1997	\$336.68

Figura 2.2- 5 Aplicación de la 1NF.

2.2.3.2 Segunda Forma Normal (2NF)

Una relación está en segunda forma normal (2NF) si y solo si esta en 1NF y todos los atributos no clave dependen por completo de la llave primaria.

Por ejemplo la figura 2.2-6 muestra los datos de la tabla EmpleadoNomina donde se identifican los procesos a los cuales están asociados los empleados, en esta tabla la llave primaria esta compuesta por ClaveDeEmpleado y ClaveDeNomina. El campo NombreDeNomina depende solo del campo ClaveDeNomina por lo cual no esta en 2NF, para solucionar este problema se descompone la tabla EmpleadoNomina en EmpleadoNomina y Nomina dejando la llave de EmpleadoNomina intacta y la de Nomina será la ClaveDeNómina. El resultado se muestra en la figura 2.2-7. Si se realiza una reunión de ambas tablas tomando como campo de reunión ClaveDeNomina se obtiene la tabla original.

ClaveDeEmpleado	ClaveDeNomina	NombreDeNomina
26	55	FINIQUITOS (MENSUAL)
26	78	PTU (FMENSUAL)
26	84	LIQ. FONDO AHORRO (FINIQUITOS)
596	55	FINIQUITOS (MENSUAL)
596	78	PTU (FMENSUAL)
1099	55	FINIQUITOS (MENSUAL)
1099	78	PTU (FMENSUAL)
11767	55	FINIQUITOS (MENSUAL)
11767	78	PTU (FMENSUAL)
12120	3	MENSUAL
12120	8	VALES (MENSUAL)
12120	10	GRATIFICACIÓN (MENSUAL)
12120	11	AGUINALDO (MENSUAL)

Figura 2.2- 6 EmpleadoNomina en 1NF pero no en 2NF.

EmpleadoNomina

ClaveDeEmpleado	ClaveDeNomina
26	55
26	78
26	84
596	55
596	78
1099	55
1099	78
11767	55
11767	78
12120	3
12120	8
12120	10
12120	11

Nomina

ClaveDeNomina	NombreDeNomina
55	FINIQUITOS (MENSUAL)
78	PTU (FMENSUAL)
84	LIQ. FONDO AHORRO (FINIQUITOS)
3	MENSUAL
8	VALES (MENSUAL)
10	GRATIFICACIÓN (MENSUAL)
11	AGUINALDO (MENSUAL)

Figura 2.2- 7 Descomposición EmpleadoNomina para que este en 2NF.

2.2.3.3 Tercera Forma Normal (3NF).

Una relación esta en tercera forma normal (3NF) si y solo si esta en 2NF y todos los atributos no clave dependen de manera no transitiva de la clave primaria.

Por ejemplo en la figura 2.2-8 se muestra la tabla de EmpleadoPuesto donde la llave primaria es ClaveDeEmpleado, aquí se identifica el puesto en el que se asigna cada empleado, sin embargo el NombreDePuesto depende de ClaveDePuesto (que no es llave) por lo que no esta en 3NF, para poder normalizarla se realiza una descomposición en dos tablas EmpleadoPuesto y Puestos esta última tendrá como llave primaria el campo ClaveDePuesto, en la figura 2.2-9 se muestran las dos tablas, aplicando una reunión ente EmpleadoPuesto y Puesto . Si se realiza una reunión de ambas tablas tomando como campo de reunión ClaveDePuesto se obtiene la tabla original.

EmpleadoPuesto

ClaveDeEmpleado	Nombre	ClaveDePuesto	NombreDePuesto
220038	ACEVEDO ROCHA BRAULIO	50	Asistente De Area
19414	ACOSTA RUBIO OSCAR SAMUEL	50	Asistente De Area
14209	AMBROSIO AYALA RUPERTO PEDRO	3	Tecnico De Laboratorio
18598	ANGULO CALDERON MARIA ESTHER	2	Secretaria Bilingue De Gerencia
250134	ARCADIA SANTOS SILVIA PATRICIA	55	Secretaria En Español
220046	ARELLANES SANCHEZ CESAR	19	Analista DE SISTEMAS
19364	ARRIOLA RIOS JOEL EDUARDO	50	Asistente De Area
13334	ARVIZU RIOS JUAN JOSE	29	Jefe De Laboratorio
18903	BALDERAS ROSALES MIGUEL ANGEL	112	Contralor
15313	BARRANCO PACHECO JESUS	114	Analista Comercial A

Figura 2.2- 8 EmpleadoPuesto en 1NF en 2NF pero no en 3NF.

2.2.3.4 Forma Normal (Boyce/Codd)

Una relación esta en forma normal (Boyce/Codd) (BNCF) si y solo si todo determinante es una clave candidata.

EmpleadoPuesto.

ClaveDeEmpleado	Nombre
220038	ACEVEDO ROCHA BRAULIO
18414	ACOSTA RUBIO OSCAR SAMUEL
14209	AMBROSIO AYALA RUPERTO PEDRO
18598	ANGULO CALDERÓN MARIA ESTHER
250134	ARCADIA SANTOS SILVIA PATRICIA
220046	ARELLANES SANCHEZ CESAR
18364	ARRIOLA RIOS JOEL EDUARDO
13334	ARVIZU RIOS JUAN JOSE
18503	BALDERAS ROSALES MIGUEL ANGEL
15313	BARRANCO PACHECO JESUS

Puesto

ClaveDePuesto	NombreDePuesto
50	Asistente De Area
3	Tecnico De Laboratorio
2	Secretaria Bilingue De Gerencia
55	Secretaria En Español
19	Analista DE SISTEMAS
29	Jefe De Laboratorio
112	Contralor
114	Analista Comercial A

Figura 2.2- 9 Descomposición de EmpleadoPuesto para esta en 3NF.

2.2.3.5 Cuarta Forma Normal.

Una relación R está en cuarta forma normal (4NF) si y solo si, siempre que existe una DMV en R, digamos $A \twoheadrightarrow B$, todos los atributos de R dependen también funcionalmente de A. en otras palabras, las únicas dependencias (funcionales o multivaluadas) en R son de la forma $K \twoheadrightarrow X$ (o sea, una dependencia funcional con respecto a una clave candidata K de algún otro atributo) o lo que es equivalente: R esta en 4NF si esta en BCNF y todas las dependencias multivaluadas en R son de hecho dependencias funcionales.

2.2.3.6 Quinta Forma Normal.

Una relación es en quinta forma normal (5NF) llamada también forma normal de proyección – reunión (PJ/NF) si y solo si toda dependencia de reunión de R es una consecuencia de las claves candidatas de R.

2.3 Conceptos Básicos del Análisis Estructurado.

¿Qué es el Análisis?

Es el estudio de un problema, que permite tomar una acción posterior

El propósito del análisis es decir que acción tomar, y describir esa acción completamente. En el dominio específico del desarrollo de sistemas de computo, el análisis se refiere al estudio de alguna área del negocio que requiere de un sistema, la acción que se tomará después de este análisis será la de implantar el sistema.

El producto más importante del análisis de sistemas es el documento de especificación, el cual establece las metas para el resto del proyecto, este documento establecerá los puntos que se deberán cumplir para considerar el proyecto exitoso en la figura 2.3-1 se muestra el flujo de la especificación.

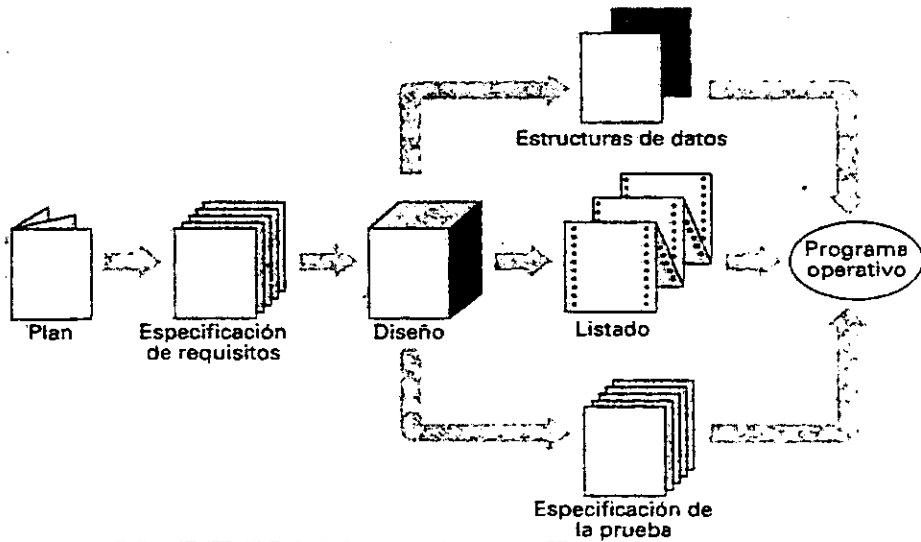


Figura 2.3- 1 Flujo de la Especificación.

El completo éxito de la fase del análisis involucra lo siguiente:

- Establecer el objetivo.
- Producir el documento de especificación, de tal manera que se pueda evaluar la implantación para ver si se logró el objetivo.
- Predecir el avance hacia el objetivo, sobre la base de costos, tiempo, beneficios, ejecución.
- Involucrar a las partes afectadas en cada uno de los tres puntos antes mencionados.

Las tareas del analista son muchas, pero como mínimo debe realizar:

- Vínculos con el usuario
- La especificación.
- El análisis Costo-Beneficio.
- El análisis de Factibilidad
- La Estimación.

Vínculos con el Usuario

Para lograr el éxito de un proyecto es indispensable la participación del usuario. El usuario debe estar enterado de cómo funcionara el sistema y como lo usará. Su experiencia en el área de negocio debe ser un ingrediente importante en el desarrollo del sistema. El usuario debe estar enterado de los avances y debe existir un canal a través del cual se puedan hacer las correcciones necesarias. Todo esto es responsabilidad del analista, el es el maestro, traductor, consejero del usuario, su función de intermediario es una de la actividades mas importantes del analista.

Especificación.

El éxito del proceso de especificación depende del producto final, el documento de especificación en nuestro caso, el cual deberá servir como un modelo del nuevo sistema, al punto que sirva para visualizar el sistema. El documento de especificación es el modelo del sistema.

Análisis de Costo - Beneficio.

Lo referente al estudio del costo-beneficio de los sistemas potenciales es un mecanismo de retro-alimentación usado por el analista para seleccionar la mejor opción. Un modelo del sistema ayuda en este punto para refinar las estimaciones del costo-beneficio.

Análisis de Factibilidad.

El análisis de factibilidad se refiere al proceso de prueba continua que el analista debe hacer para asegurarse que el sistema que se esta especificando pueda ser implementado dado un conjunto de restricciones.

Estimación.

Durante la fase del análisis generalmente solo se cuenta con un sistema en papel, pero también se involucran una serie de estimaciones, costo, duración, recursos necesarios, espacio en disco, tiempo de procesador etc. Una pobre estimación provocará que el proyecto fracase.

Algunos puntos a considerar en la estimación son:

- Nadie es un estimador experto.
- No podemos medir la habilidad del equipo, por que no se llevan estadísticas
- No siempre es un proceso de estimación, sino de ajuste de recursos a un calendario dado.

La estimación puede auxiliarse de la heurística, que a veces es un truco que funciona bien aunque no es garantía, no es un algoritmo ni un proceso que garantice resultados.

2.3.1 Problemas del análisis.

Un proyecto puede ir mal por muchas razones. El hecho de que se invierta mucho tiempo energía en el mantenimiento es una indicación de una falla como

diseñadores, el gastar mucho tiempo en depurar es una indicación de los métodos de diseño, codificación y prueba. Pero una falla en el análisis se vera reflejado en diferentes aspectos. Cuando el análisis esta mal no solo se gasta dinero y tiempo para lograr el objetivo, sino que se invierten demasiados recursos y generalmente no se logra cumplir con la meta.

Los problemas en el análisis mas comunes son los siguientes:

- Problemas de comunicación.
- El cambio de los requerimientos del sistema
- La falta de herramientas
- Problemas con el documento de especificación
- Problemas con el equipo de trabajo.
-

2.3.1.1 Problemas de comunicación.

Los problemas de comunicación se presentan, en nuestro caso, por la falta de un lenguaje común entre el usuario y el analista. Los elementos con los que el analista trabaja – especificaciones, descripción de formato de datos, diagramas de flujo, código entre otros – los cuales son totalmente ajenos al usuario. Uno de los aspectos del sistema en donde el usuario se siente más involucrado es cuando se habla acerca de las interfaces con las que inter - actua con el sistema.

En resumen los factores que contribuyen a los problemas de comunicación en la fase del análisis son:

- La dificultad natural de describir un procedimiento.
- Lo inapropiado de los métodos (Texto narrativo).
- La falta de un lenguaje común entre el analista y el usuario.
- La falta de tener rápidamente un modelo utilizable del sistema (Prototipo)

2.3.1.2 Cambio de los requerimientos.

La idea de congelar la especificación del sistema es algo que no se puede aplicar, ya que no se pueden ignorar los cambios, los cuales se presentan por cambios en la reglas del negocio, por una mayor participación del usuario o por fallas en la comunicación.

Es por tanto que se debe cambiar el enfoque, se deben construir documentos de especificación altamente flexibles, de hecho si se logra este objetivo será el primer paso para tener un sistema altamente flexible.

2.3.1.3 La falta de herramientas.

El trabajo del analista se vería mejorado si contara con las herramientas adecuadas para poder organizar y corregir el documento de especificación. Este punto esta muy ligado con la necesidad de tener un análisis flexible ya que si se requiere un cambio en las especificaciones será necesario cambiar el documento de especificación.

2.3.1.4 Problemas con el Documento de Especificación.

Resulta evidente que entre más grande sea el sistema más complejo se convertirá la tarea del análisis. Se puede hacer muy poco con respecto al tamaño del sistema, pero se puede atacar de distintas maneras, una de las más eficientes es el particionar. Esto es lo que hacen los diseñadores con un sistema muy grande para poderlo manejar apropiadamente – se descompone en partes pequeñas (módulos) y es exactamente lo que debe hacer el analista.

En la fase del análisis lo principal que debe particionar es el Documento de Especificación, se debe evitar el escribir “grandes novelas”, enormes documentos que para poder entenderlos se tendrán que leer de principio a fin. Es necesario aprender a hacer mini - especificaciones y organizarlas de tal manera que puedan ser estudiadas selectivamente.

Además de los problemas del tamaño existen otros:

- Que sea excesivamente redundante.
- Excesivamente grande (físicamente)
- Que sea tedioso de leer y difícil de escribir cambios.

2.3.1.5 El equipo de trabajo.

El análisis es una tarea que debe realizarse por un pequeño equipo, en ocasiones un sola persona, es importante seleccionar a la persona idónea para esta actividad desde el principio ya que hacer cambios durante el análisis resulta generalmente contraproducente.

2.3.2 La relación entre el Usuario / Analista.

Es importante identificar el papel que juegan en la fase del análisis el usuario y el analista, en la figura 2.3-2 se muestra.

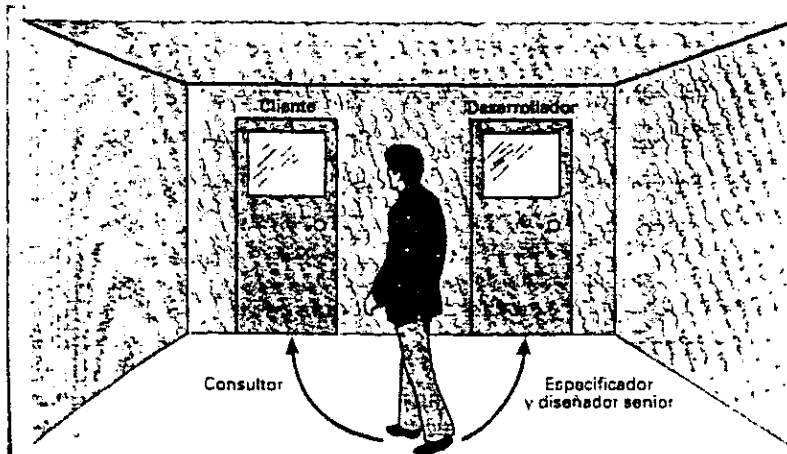


Figura 2.3- 2 Relación del Usuario-Analista.

2.3.2.1 ¿Quién es el usuario?

Difícilmente existe solo un usuario, de hecho el término de usuario puede clasificarse de acuerdo a su papel en las siguientes categorías:

1. El usuario final: Es el operador del sistema
2. El usuario responsable: Es el responsable de la operación o del área del negocio a que será automatizada por el sistema.
3. El dueño del sistema: Es generalmente el gerente o director del área.

En ocasiones estos papeles se combinan pero generalmente son personas distintas, y es importante identificar a los usuarios y el papel que desempeñan ya que es común omitir a alguno lo que provoca que no se logre el éxito del proyecto. Es común también que se nombre un "Usuario Representante" con el fin de evitar fricciones entre el equipo de desarrollo y los usuarios, esta figura sería conveniente si tuviera la autoridad de aceptar el sistema, generalmente no la tiene, y cuando llega el momento de aceptarlo es cuando se involucran los demás usuarios, cuando esto pasa es como si se volviera a empezar con todas las implicaciones que esto provoca.

2.3.2.2 ¿Quién es el analista?

El analista es la liga principal entre los usuarios y el equipo de desarrollo, el tiene que comunicar los requerimientos a los desarrolladores y retroalimentar a los usuarios acerca de cómo se están resolviendo los problemas, en ocasiones el analista propone opciones de solución a los problemas, o puede participara en la implementación asumiendo el papel de arquitecto que guía la construcción.

La mayoría de los analistas requieren:

- Facilidad con los elementos del desarrollo de sistemas
- Facilidad con los elementos del área de negocio.
- Ser capaz de comunicar estos conceptos.

2.3.2.3 División entre las responsabilidades del analista y el usuario.

En ocasiones no es muy clara la división de funciones o responsabilidades entre el analista y el usuario, pero se puede establecer; ¿Que es lo que se necesita realizar? es del dominio del usuario y ¿Como se va ha realizar? es del dominio del analista.

2.3.3 ¿Qué es el Análisis Estructurado?

Es una metodología para especificar funcionalmente un sistema, cuya principal meta es minimizar los errores en la fase del análisis, utilizando las herramientas del análisis estructurado.

Hasta este punto se ha discutido lo que se puede considerar como las fases de análisis clásico, sus problemas y debilidades. Pero ¿En qué es distinto el análisis estructurado?. Se puede decir que las diferencias se basan en:

- Establece nuevas metas para el análisis
- Utiliza herramientas estructuradas para el análisis.

Las nuevas metas para el Análisis.

- Los productos del análisis deben ser altamente flexibles, especialmente en lo que se refiere al Documento de Especificación.
- Los problemas de tamaño deben ser abordados usando un método efectivo de partición.
- Los gráficos deben usarse lo más posible.
- Se debe diferenciar entre las consideraciones lógicas y físicas, y delegar las responsabilidades, basándose en las diferencias entre los papeles que desempeñan el usuario y el analista.
- Construir modelos de sistemas para que el usuario se vaya familiarizando con las características del sistema antes de su implementación.

Herramientas estructuradas para el Análisis.

- Las nuevas herramientas para el Análisis Estructurado son:
- Diagramas de Flujo de Datos.
- Diccionario de Datos.
- Español Estructurado.
- Tablas de Decisión.
- Árboles de Decisión.

2.3.3.1 Diagramas de Flujo.

Un diagrama de flujo de datos (DFD) es una representación gráfica en forma de red de un sistema, el cual puede ser automático, manual o mixto. El diagrama de flujo de datos retrata al sistema en términos de sus componentes con todas las interfaces entre los componentes indicados.

Algunas de las características más importantes de los diagramas de flujo de datos son:

- Es gráfico
- Particionado
- Multidimensional
- Énfasis en el flujo de datos
- No enfatiza el flujo del control

Esta herramienta permite visualizar al sistema desde el punto de vista de los datos y no de quien trabaja con ellos.

Los elementos de un diagrama de flujo de datos son:

- Flujo de datos: representados por vectores etiquetados
- Procesos: representados por círculos o burbujas.
- Archivos: representados por cajas.

- Fuentes o Depósitos de datos: representadas por cajas con barras a los costados

La figura 2.3 -3 es una muestra de un diagrama de flujo de datos que incluye los cuatro elementos:

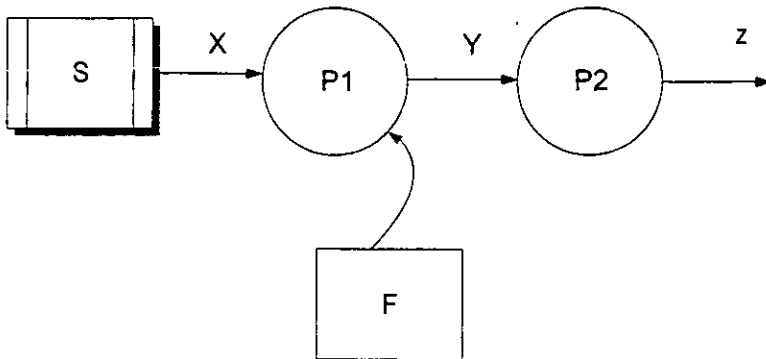


Figura 2.3- 3 Elementos de un DFD.

Este tipo de diagrama puede interpretarse de la siguiente manera:

"X proviene de la fuente S y es transformada en Y por el proceso P1 (el cual requiere obtener datos del archivo F para completar la tarea), después Y es transformada en Z por el proceso P2."

Con esta interpretación es claro que se enfatiza el flujo de los datos a través del sistema. A continuación se analizará con mas detenimiento cada uno de los elementos de los diagramas de flujo de datos.

Flujo de datos.

Un flujo de datos retrata una interfaz entre los componentes de un DFD, la mayoría de las veces los flujos de datos se mueven entre procesos pero también fluyen

desde o para los archivos o de las fuentes o depósitos de información, se puede definir como:

Un flujo de datos es un canal a través del cual los paquetes de información de composición conocida fluyen.

El nombre que se selecciona para los flujos de datos deben dar una idea del tipo de datos que se mueven, por ejemplo si se nombra un flujo como NumeroDeCuenta tiene la misma composición que NumeroDeCuenta_Valido pero semánticamente es mas clara la segunda opción. En ocasiones no es necesario nombrar los flujos de datos que provienen de los archivos ya que el nombre del archivo debe ser suficiente para identificar la información.

Es importante resaltar que los flujos de datos no representan control de flujo ni son disparadores de algún evento.

El Proceso.

Los procesos invariablemente representan alguna cantidad de trabajo hecho sobre los datos.

Generalmente a los procesos se les nombra en función de los datos de entrada y salida, se requiere de un nombre descriptivo de la operación que realiza aunque en otra parte del modelo (Español Estructurado) se describa con mayor detalle.

Un proceso es una transformación de flujos de datos de entrada en flujos de datos de salida.

El Archivo.

Para efectos del análisis estructurado se puede definir como:

Un archivo es un almacén de datos.

Este almacén puede ser una cinta, espacio en el disco, o registro de datos. Las bases de datos también son consideradas como archivos, sin embargo en este punto no se tomarán el detalle de la implementación física.

Fuentes o Depósitos de Datos.

Cualquier sistema o área del negocio puede ser descrito en un DFD con flujos de datos, procesos y archivos. Sin embargo algunas veces se puede incrementar la semántica del modelo mostrando el origen y el destino de los datos.

Una fuente o depósito es una persona u organización que esta fuera del contexto del sistema pero que es un generador o receptor de datos para el sistema.

2.3.3.2 Diccionario de Datos.

Un diccionario de datos es un almacén de datos acerca de los datos.

El Diccionario de Datos (DD) es una parte integral de la especificación estructurada; sin esta los diagramas de flujo de datos son solo diagramas que dan una idea de que es lo que pasa en el sistema. Es solo cuando cada elemento del DFD ha sido rigurosamente definido que el todo puede constituir una especificación. El conjunto de rigurosas definiciones de todos los elementos del DFD es un diccionario de datos.

El papel más importante de un diccionario de datos es proporcionar en un solo lugar las referencias de los términos que no se entienden en el DFD.

El DFD y el DD deben ser considerados como uno solo. Sin el DD los diagramas pierden rigor, sin los diagramas el DD no tiene ningún uso, se puede decir que la correlación entre ambos es:

Hay una entrada al DD por cada flujo de datos que aparecen en el DFD, hay una entrada en el DD por cada archivo que se hace referencia en el DFD.

2.3.3.3 Español Estructurado.

El español estructurado es un lenguaje de especificación que hace uso de un vocabulario limitado y una sintaxis también limitada.

El vocabulario del español estructurado consiste solo de:

- Verbos en imperativo
- Términos definidos en el Diccionario de Datos.
- Ciertas palabras reservadas para la formulación lógica.

La sintaxis del español estructurado esta limitada a estas posibilidades.

- Sentencias declarativas simples (secuencia).
- Sentencias de inicio y fin para estructuras de decisión.
- Sentencias de inicio y fin para estructuras de repetición.

O combinaciones de estas.

Otra forma de definir el español estructurado es como pseudocódigo, la ventaja de usar esta herramienta es que se puede uno concentrar en la lógica y en las estructuras de control y no preocuparse de las reglas de un lenguaje específico. Es también fácil modificar el pseudocódigo si se descubren errores o anomalías.

2.3.3.4 Tablas de decisión.

La sentencia de un programa puede ser descrita por una estructura de decisión, la tabla de decisión (TD).

Una TD es un documento de comunicación entre usuarios, analistas, programadores, etc., y un instrumento de análisis y programación, que se puede aplicar a numerosos y diversos problemas, permitiendo la representación de las diferentes situaciones de una modo fácil y lógico.

Una TD es una herramienta que permite presentar de forma concisa las reglas lógicas que hay que utilizar para decidir acciones a ejecutar en función de las condiciones y la lógica de decisión de un problema específico.

Una TD es un tipo de tabla que muestra lo que debe realizar el programa cuando se cumplen ciertas condiciones. Son especialmente útiles para describir procesos que incluyan muchas decisiones múltiples.

2.3.3.5 Árboles de Decisión.

Un árbol de decisión es una representación gráfica de una tabla de decisión, y es útil para ver de manera más clara esta representación.

2.4 Ingeniería de Software.

Una de las primeras definiciones de ingeniería del software fue la propuesta por Fritz Bauer:

El establecimiento y uso de principios de ingeniería robustos, orientados a obtener software económico que sea fiable y funcione de manera eficiente sobre máquinas reales.

Aunque se han propuesto muchas más definiciones generales, todas refuerzan la importancia de una disciplina de ingeniería para el desarrollo del software.

La ingeniería del software surge de la ingeniería de sistemas y de hardware. Abarca un conjunto de tres elementos clave – métodos, herramientas y procedimientos – que facilitan al gestor controlar el proceso del desarrollo del software y suministrar a los que practiquen dicha ingeniería las bases para construir software de alta calidad de una forma productiva.

Los métodos de la ingeniería de software indican como construir técnicamente el software. Los métodos abarcan un amplio espectro de tareas que incluyen: planificación y estimación de proyectos, análisis de los requerimientos del sistema y del software, diseño de estructuras de datos, arquitectura de programas y procedimientos algorítmicos, codificación, prueba y mantenimiento. Los métodos de la ingeniería del software introducen frecuentemente una notación especial orientada a un lenguaje o gráfica y un conjunto de criterios para la calidad del software.

Las herramientas de la ingeniería de software suministran un soporte automático o semiautomático para los métodos. Hoy existen herramientas para soportar cada uno de los métodos mencionados anteriormente. Cuando se integran las herramientas de forma tal que la información creada por una herramienta pueda ser usada por otra, se

establece un sistema para el soporte del desarrollo del software, llamado ingeniería del software asistida por computadora (del ingles CASE).

Los procedimientos de la ingeniería del software son el pegamento que junta los métodos y las herramientas y facilitan un desarrollo racional y oportuno del software de computadora. Los procedimientos definen la secuencia en la que se aplican los métodos, las entregas (documentos, informes, formas, etc.) que se requieren, los controles que ayudan a asegurar la calidad y coordinar los cambios, y las directrices que ayudan a los gestores del software a evaluar el progreso.

La ingeniería del software está compuesta por una serie de pasos que abarcan los métodos, las herramientas y los procedimientos antes mencionados. Estos pasos se denominan frecuentemente paradigmas de la ingeniería del software. La elección de un paradigma para la ingeniería del software se lleva a cabo de acuerdo con la naturaleza del proyecto de la aplicación, los métodos y herramientas a usar y los controles y entregas requeridos.

2.4.1 El ciclo de vida clásico.

La figura 2.4-1 ilustra el ciclo de vida clásico del desarrollo del software.

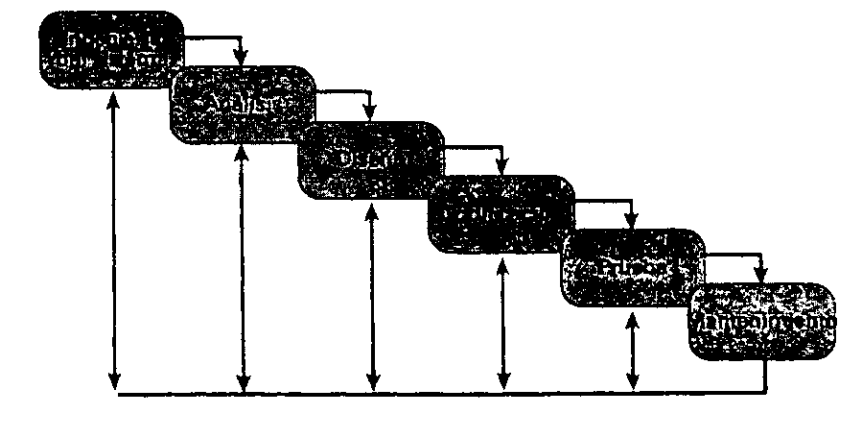


Figura 2.4- 1 Ciclo de vida Clasico.

También conocido como modelo de cascada, el paradigma del ciclo de vida exige un enfoque sistemático y secuencial del desarrollo del software que comienza en el nivel del sistema y progresa a través del análisis, diseño, codificación, prueba y mantenimiento. Modelizado a partir del ciclo convencional de una ingeniería, el paradigma del ciclo de vida abarca las siguientes actividades:

Ingeniería y análisis del sistema. Debido a que el software es siempre parte de un sistema mayor, el trabajo comienza estableciendo los requisitos de todos los elementos del sistema y luego asignando algún subconjunto de estos requisitos al software. Este planteamiento del sistema es esencial cuando el software debe interrelacionarse con otros elementos, tales como hardware, personas y bases de datos. La ingeniería y el análisis del sistema abarca los requisitos globales a nivel del sistema con una gran cantidad de análisis y de diseño a un nivel superior.

Análisis de los requisitos del software. El proceso de recopilación de los requisitos se centra e intensifica especialmente para el software. Para comprender la naturaleza de los programas que hay que construir, el ingeniero de software ("analista") debe comprender el ámbito de la información del software, así como la función, el rendimiento y las interfaces requeridas. Los requisitos, tanto del sistema como del software, se documentan y se revisan con el cliente.

Diseño. El diseño del software es realmente un proceso multipaso que se enfoca sobre cuatro atributos distintos del programa: la estructura de los datos, la arquitectura del software, el detalle procedural y la caracterización de la interfaz. El proceso de diseño traduce los requisitos en una representación del software que puede ser establecida de forma que obtenga la calidad requerida antes de que comience la codificación. Al igual que los requisitos, el diseño se documenta y forma parte de la configuración del software.

Codificación. El diseño debe traducirse en una forma legible para la máquina. El paso de codificación realiza esta tarea. Si el diseño se realiza de una manera detallada, la codificación puede realizarse mecánicamente.

Prueba. Una vez que se ha generado el código, comienza la prueba del programa. La prueba se centra en la lógica interna del software, asegurando que todas las sentencias se han probado, así como en las funciones externas, realizando pruebas que aseguren que la entrada definida produce los resultados que se requieren.

Mantenimiento. El software, indudablemente, sufrirá cambios después de que se entregue al cliente (una posible excepción es el software empotrado). Los cambios ocurrirán debido a que se hayan encontrado errores, a que el software deba adaptarse a cambios del entorno externo (por ejemplo, un cambio solicitado debido a que se tiene un nuevo sistema operativo o dispositivo periférico), o debido a que el cliente requiera ampliaciones funcionales o del rendimiento. El mantenimiento del software aplica a cada uno de los pasos precedentes del ciclo de vida de un programa existente en vez de a uno nuevo.

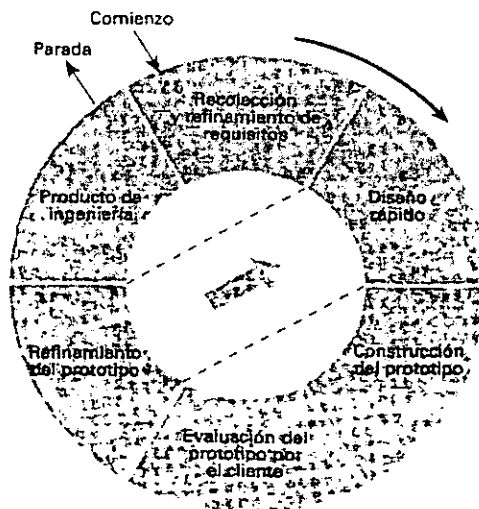


Figura 2.4- 2 Construcción de Prototipos.

2.4.2 Construcción de prototipos.

La construcción de prototipos es un proceso que facilita al programador la creación de un modelo del software a construir. El modelo tomará una de las tres formas siguientes: (1) un prototipo en papel o un modelo basado en PC que describa la interacción hombre-máquina, de forma que facilite al usuario la comprensión de cómo se producirá tal interacción; (2) un prototipo que implemente algunos subconjuntos de la función requerida del programa deseado, o (3) un programa existente que ejecute parte o toda la función deseada, pero que tenga otras características que deban ser mejoradas en el nuevo trabajo de desarrollo, en la figura 2.4-2 se muestra un esquema de la construcción de prototipos.

A continuación se muestra el ciclo de creación de prototipos.

- Recolección y refinamiento de requisitos
- Diseño rápido
- Construcción del prototipo
- Evaluación del prototipo por el cliente
- Refinamiento del prototipo
- Producto de Ingeniería

Idealmente, el prototipo sirve como mecanismo para identificar los requisitos del software. Si se va a construir un prototipo que funcione, el realizador intenta hacer uso de fragmentos de programas existentes o aplica herramientas (por ejemplo, generadores de informes, gestores de ventanas, etc.) que faciliten la rápida generación de programas que funcionen.

Aunque pueden aparecer problemas, la construcción de prototipos es un paradigma efectivo para la ingeniería del software. La clave está en definir al comienzo las reglas del juego; esto es el cliente y el técnico deben estar de acuerdo en que el prototipo se construya para servir sólo como un mecanismo de definición de los requisitos. Posteriormente, ha de ser descartado (al menos en parte) y debe construirse el software real, con los ojos puestos en la calidad y en el mantenimiento.

2.4.3 El modelo en espiral.

El modelo en espiral para la ingeniería del software ha sido desarrollado para cubrir las mejores características tanto del ciclo de vida clásico, como de la creación de prototipos, añadiendo al mismo tiempo un nuevo elemento; el análisis de riesgo, que falta en esos paradigmas. El modelo, representado mediante la espiral de la figura 2.4-3 define cuatro actividades principales, representadas por los cuatro cuadrantes de la figura:

1. Planificación: determinación de objetivos, opciones y restricciones
2. Análisis de riesgo: análisis de opciones e identificación/resolución de riesgos.
3. Ingeniería: desarrollo de producto de "siguiente nivel"
4. Evaluación del cliente: valoración de los resultados de la ingeniería.

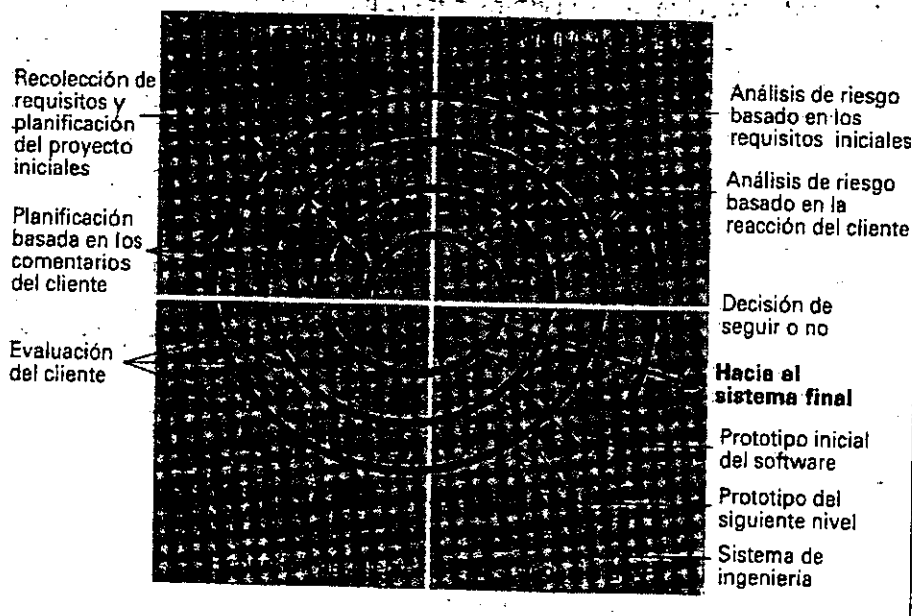


Figura 2.4- 3 Modelo en Espiral.

El paradigma del modelo en espiral para la ingeniería del software es actualmente el enfoque más realista para el desarrollo de software y de sistemas a gran escala. Utiliza un enfoque evolutivo para la ingeniería de software, permitiendo al desarrollador y al cliente entender y reaccionar a los riesgos en cada nivel evolutivo. Utiliza la creación de prototipos como un mecanismo de reducción de riesgo, pero, lo que es más importante, permite a quien lo desarrolla aplicar el enfoque de creación de prototipos en cualquier etapa de la evolución del producto. Mantiene el enfoque sistemático correspondiente a los pasos sugeridos por el ciclo de vida clásico, incorporándola dentro de un marco de trabajo interactivo que refleja de forma más realista el mundo real. El modelo en espiral demanda una consideración directa de riesgos técnicos en todas la etapas del proyecto y, si se aplica adecuadamente, debe reducir los riesgos antes de que se conviertan en problemáticos.

2.4.4 Técnicas de Cuarta Generación.

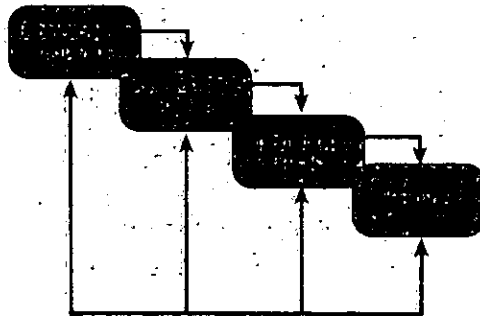


Figura 2.4- 4 Técnicas de 4ta. Generación.

El término técnicas de cuarta generación (TAG) abarca un amplio espectro de herramientas de software que tiene algo en común: todas facilitan, al que desarrolla el software, la especificación de algunas características del software a alto nivel. Luego, la herramienta genera automáticamente el código fuente tomando como base la especificación del técnico. Cada vez parece más evidente que cuanto mayor sea el nivel en que se especifique el software, más rápido se podrá construir el programa. El

paradigma T4G para la ingeniería de software se orienta hacia la posibilidad de especificar el software a un nivel más próximo al lenguaje natural o en una notación que proporcione funciones significativas, en la figura 2.4-4 se muestra el ciclo de T4G.

La implementación mediante un L4G permite, al que desarrolla el software, centrarse en la representación de los resultados deseados, que es lo que se traduce automáticamente en un código que produce dichos resultados. Obviamente, debe existir una estructura de datos con información relevante y a la que el L4G puede acceder rápidamente.

2.4.5 Combinación de paradigmas.

No hay necesidad de ser dogmático en la elección de los paradigmas para ingeniería del software; la naturaleza de la aplicación debe dictar el método a elegir. Mediante la combinación de paradigmas, el todo puede ser mejor que la suma de las partes.

2.4.6 Visión genérica de la ingeniería del software.

El proceso de desarrollo del software contiene tres fases genéricas, independientemente del paradigma de ingeniería elegido. Las tres fases, definición, desarrollo y mantenimiento, se encuentran en todos los desarrollos de software, independientemente del área de aplicación, del tamaño del proyecto o de la complejidad.

La fase de definición se cuenta sobre qué. Esto es, durante la definición, el que desarrolla el software intenta identificar qué información ha de ser procesada, qué función y rendimiento se desea, que interfaces han de establecerse, que restricciones de diseño y que criterios de validación se necesitan para definir un sistema correcto. Por tanto, han de identificarse los requisitos claves del sistema y del software.

La fase de desarrollo se centra en el cómo. Esto es, durante esta fase, el que desarrolla el software intenta descubrir cómo han de diseñarse las estructuras de datos y la arquitectura del software, como han de implementarse los detalles procedurales como ha de traducirse el diseño a un lenguaje de programación (o lenguaje no procedural) y cómo ha de realizarse la prueba. Los métodos aplicados durante la fase de desarrollo variarán, pero de alguna forma se producirán tres pasos concretos:

Diseño del software: El diseño traduce los requisitos del software a un conjunto de representaciones (algunas gráficas y otras tabulares o basadas en lenguajes) que describen la estructura de datos, la arquitectura, el procedimiento algorítmico y las características de la interfaz.

Codificación: La representación del diseño debe ser traducida a un lenguaje artificial (un lenguaje de programación convencional o un lenguaje no procedural usado en el contexto del paradigma T4G), dando como resultado unas instrucciones ejecutables por la computadora. El paso de la codificación es el que lleva a cabo esta traducción.

Prueba del software: Una vez que el software ha sido implementado en una forma ejecutable por la máquina, debe ser probado para descubrir los defectos que puedan existir en la función, en la lógica y en la implementación.

La fase de mantenimiento se centra en el cambio que va asociado a la corrección de errores, a las adaptaciones requeridas por la evolución del entorno del software y a las modificaciones debidas a los cambios de los requisitos del cliente dirigidos a reforzar o a ampliar el sistema. La fase de mantenimiento vuelve a aplicar los pasos de las fases de definición y de desarrollo, pero en el contexto del software ya existente. Durante la fase de mantenimiento se encuentran tres tipos de cambios:

Corrección: Incluso llevando a cabo las mejores actividades de garantía de calidad, es muy probable que el cliente descubra defectos en el software. El mantenimiento correctivo cambia el software para corregir los defectos.

Adaptación: Con el paso del tiempo es probable que cambie el entorno original (p.e., la UCP, el sistema operativo, los periféricos) para el que se desarrolló el software. El mantenimiento adaptativo consiste en modificar el software para acomodarlo a los cambios de su entorno externo.

Mejora: Conforme utilice el software, el cliente/usuario puede descubrir funciones adicionales que podría interesar que estuvieran incorporadas en el software. El mantenimiento perfectivo amplía más allá de sus requisitos funcionales originales.

Las fases y sus pasos relacionados descritos en esta visión genérica de la ingeniería del software, se complementan con varias actividades "protectoras": las revisiones que se realizan durante cada paso para asegurar que se mantiene la calidad. La documentación que se desarrolla y controla para asegurar que toda la información sobre el sistema y el software estará disponible para un uso posterior. El control de los cambios que se instituye de forma que los cambios puedan ser mejorados y registrados.

2.4.7 Ingeniería de software y diseño del software.

El diseño del software se asienta en el núcleo técnico del proceso de ingeniería del software y se aplica independientemente del paradigma de desarrollo utilizado. Cada una de las actividades del diseño (diseño, codificación y prueba) transforma la información de forma que finalmente se obtiene un software para computadora validado.

La figura 2.4-5 muestra el flujo de información durante la fase de desarrollo. Los requisitos del programa, establecidos mediante los modelos de información, funcional y de comportamiento, alimentan el paso del diseño. Mediante alguna metodología del diseño se realiza el diseño de datos, diseño arquitectónico y el diseño procedural. El diseño de datos transforma el modelo del campo de información, creado durante el análisis, en las estructuras de datos que se van a requerir para implementar el software.

El diseño arquitectónico define las relaciones entre los principales elementos estructurales del programa. El diseño procedural transforma los elementos estructurales en una descripción procedural del software. Se genera el código fuente y, para integrar y validar el software, se llevan a cabo las pruebas.

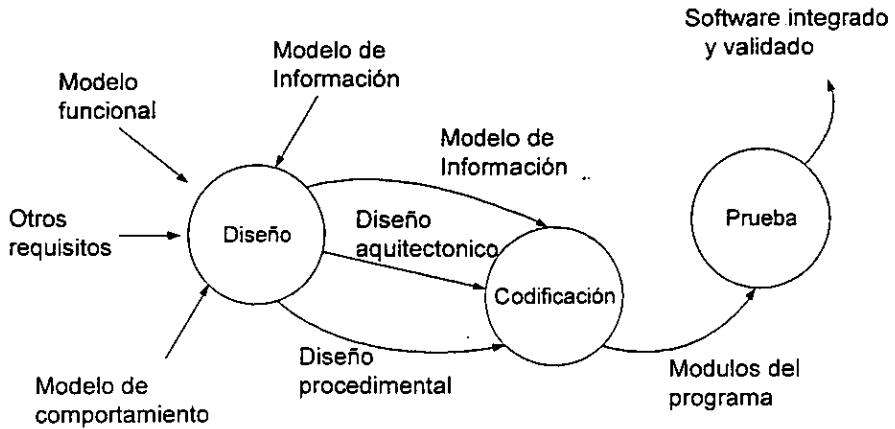


Figura 2.4- 5 Fases del diseño .

Las fases de diseño, codificación y prueba absorben el 75 por 100 o más del costo de la ingeniería de software (excluyendo el mantenimiento). Es aquí donde se toman las decisiones que afectan finalmente el éxito de la implementación del programa y con igual importancia, a la facilidad de mantenimiento que tendrá el software. Estas decisiones se llevan a cabo durante el diseño del software, haciendo que sea un paso fundamental de la fase de desarrollo.

La importancia del diseño del software se puede sentir con una única palabra "calidad". Sin diseño se corre el riesgo de construir un sistema inestable, un sistema que falle cuando se realicen pequeños cambios.

2.4.7.1 Proceso de Diseño.

Desde el punto de vista de la gestión del proyecto, el diseño del software se realiza en dos pasos. El diseño preliminar se centra en la transformación de los datos y la arquitectura del software. El diseño detallado se ocupa del refinamiento de la

representación arquitectónica que lleva a una estructura de datos detallada y a las representaciones algorítmicas del software.

Dentro del contexto de los diseños preliminar y detallado, se llevan a cabo varias actividades de diseño diferentes. Además del diseño de datos, del diseño arquitectónico y del diseño procedural, muchas aplicaciones modernas requieren una actividad distinta de diseño de la interfaz. El diseño de la interfaz establece la disposición y los mecanismos para la interacción hombre-máquina.

2.4.7.2 Fundamentos del diseño.

Abstracción.

Cuando se considera una solución modular para cualquier problema, pueden formularse muchos niveles de abstracción. En el nivel superior de abstracción, se establece una solución en términos amplios, usando el lenguaje del entorno del problema. En los niveles inferiores de abstracción de toma una orientación más procedural. La terminología orientada al problema se acompaña con una terminología orientada a la implementación, en un esfuerzo para establecer una solución.

Cada paso de un proceso de ingeniería del software es un refinamiento del nivel de abstracción de la solución del mismo. Durante el análisis de los requerimientos del software, se establece la solución del software en términos de "lo que es familiar al entorno del problema". Conforme se mueve desde lo preliminar al diseño detallado, se reduce el nivel de abstracción. Finalmente, se alcanza el nivel más bajo de abstracción, cuando se genera el código fuente.

Refinamiento.

El refinamiento sucesivo es una primera estrategia de diseño descendente propuesto por Niklaus Wirth. La arquitectura de un programa se desarrolla en niveles sucesivos de refinamiento de los detalles procedural. Se desarrolla una jerarquía descomponiendo una declaración microscópica de una función de una forma sucesiva, hasta que se llega a las sentencias del lenguaje de programación.

El proceso de refinamiento de programas propuesto por Wirth es análogo al proceso de refinamiento y de partición usados durante el análisis de requisitos. La diferencia está en el nivel de detalle que se considera, no en el método.

Modularidad.

El software se divide en componentes con nombres y ubicaciones determinados, que se denominan módulos y que se integran para satisfacer los requisitos del problema. El software monolítico (un gran programa compuesto de un único módulo) no puede ser fácilmente abarcado por el diseñador. El número de caminos de control, la expansión de las referencias, el número de variables y la complejidad global podrían hacer imposible su correcta comprensión, esta situación se puede ver en la figura 2.4-6.

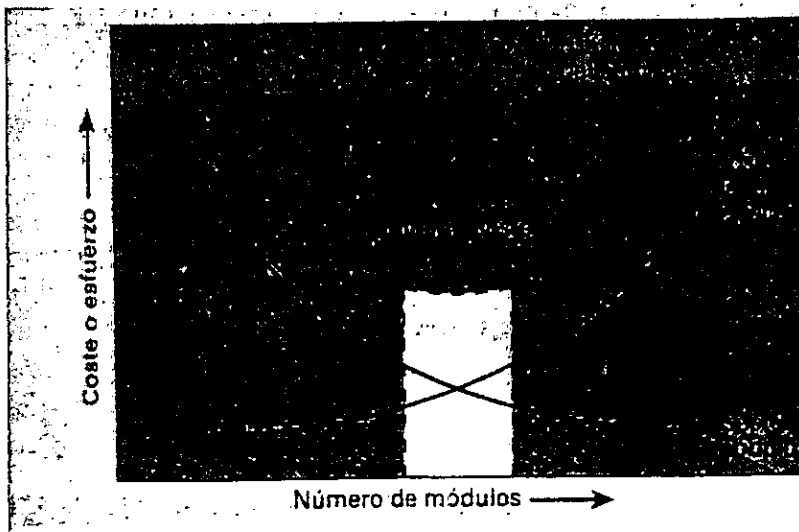


Figura 2.4- 6 Relación Módulos-Costo.

Dentro de una estructura de programa, un módulo puede ser clasificado como:

- Un módulo secuencial que se referencia y se ejecuta sin interrupción aparente por parte del software de la aplicación.

- Un módulo incremental que puede ser interrumpido, antes de que termine, por el software de la aplicación y, posteriormente, restablecida su ejecución en el punto en que se interrumpió.
- Un módulo paralelo que se ejecuta a la vez que otro módulo, en entornos de multiprocesadores concurrentes.

El concepto de independencia funcional es una derivación directa del de modularidad y de los conceptos de abstracción y ocultamiento de información. La independencia se mide con dos criterios cualitativos: la cohesión y el acoplamiento.

La cohesión es una extensión del concepto de ocultamiento de información. Un módulo cohesivo ejecuta una tarea sencilla de un procedimiento de software y requiere poca interacción con procedimientos que ejecutan otras partes del programa. Dicho de manera sencilla, un módulo cohesivo sólo hacer (idealmente) una cosa, en la figura 2.4-7 se muestra el espectro del grado de cohesión.

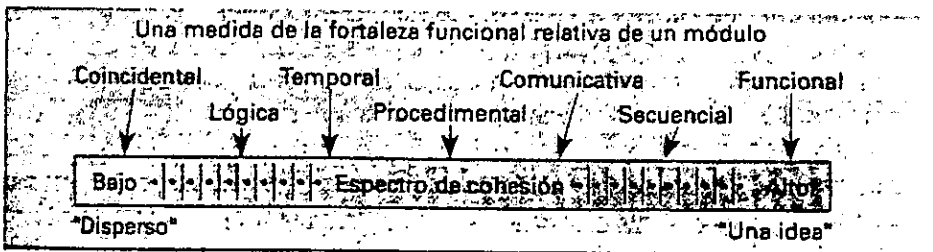


Figura 2.4- 7 Espectro del grado de cohesión.

El acoplamiento depende de la complejidad de las interfaces entre los módulos, del punto en el que se hace una entrada o referencia a un módulo y de los datos que pasan a través de la interfaz., en la figura 2.4-8 se muestra el espectro del grado de acoplamiento.

En el diseño del software se busca el más bajo acoplamiento posible. La conectividad sencilla entre módulos da como resultado un software que es más fácil de comprender y menos propenso al "efecto onda" producido cuando los errores aparecen en una posición y se propagan a lo largo del sistema.

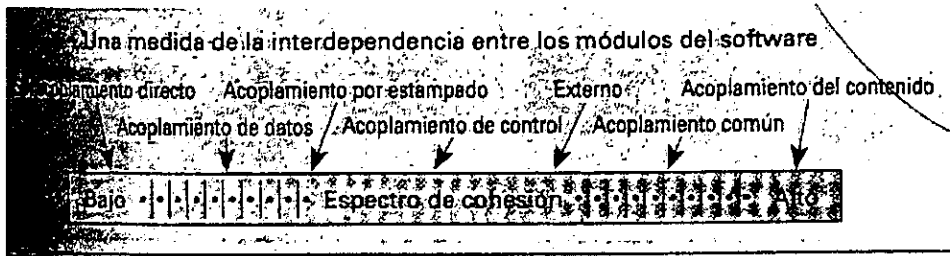


Figura 2.4- 8 Espectro del grado de acoplamiento.

CAPITULO 3 HERRAMIENTAS DE DESARROLLO.

3.1 Diseño de Windows NT.

Windows NT presenta una arquitectura del tipo cliente-servidor. Los programas de aplicación son contemplados por el sistema operativo como si fueran clientes a los que hay que servir, y para lo cual viene equipado con distintas entidades servidoras.

Uno de los objetivos fundamentales de diseño fue el tener un núcleo tan pequeño como fuera posible, en el que estuvieran integrados módulos que dieran respuesta a aquellas llamadas al sistema que necesariamente se tuvieron que ejecutar en modo privilegiado (también llamado modo kernel, modo núcleo y modo supervisor). El resto de las llamadas se expulsarían del núcleo hacia otras entidades que se ejecutarían en modo no privilegiado (modo usuario), y de esta manera el núcleo resultaría una base compacta, robusta y estable. Por eso se dice que Windows NT es un sistema operativo basado en micro-kernel.

Es por ello que en un primer acercamiento a la arquitectura se distingue un núcleo que se ejecuta en modo privilegiado, y se denomina Executive, y unos módulos que se ejecutan en modo no privilegiado, llamados subsistemas protegidos.

Los programas de usuario (también llamados programas de aplicación) interactúan con cualquier sistema operativo (SO) a través de un juego de llamadas al sistema, que es particular de cada SO. En el mundo Windows en general, las llamadas al sistema se denominan API (Application Programming Interfaces, interfaces para la programación de aplicaciones). En Windows NT y en Windows 95 se usa una versión del API llamada API Win32. Un programa escrito para Windows NT o Windows 95, y que por consiguiente hace uso del API Win32, se denomina genéricamente "programa Win32", y de hecho esta denominación es bastante frecuente en artículos y libros al respecto. Desgraciadamente, y conviene dejarlo claro cuanto antes, el término "Win32" tiene tres acepciones (al menos hasta ahora) totalmente distintas. Una es el API, otra es el nombre de uno de los subsistemas protegidos de Windows NT y por último se

denomina Win32s a una plataforma desarrollada por Microsoft, similar a Windows 3.1, pero que usa el API Win32 en vez del API Win16 del Windows 3.1.

Algunas de las llamadas al sistema, debido a su naturaleza, son atendidas directamente por el Executive, mientras que otras son desviadas hacia algún subsistema.

El hecho de disponer de un núcleo rodeado de subsistemas que se ejecutan en modo usuario permite además añadir nuevos subsistemas sin producir ningún tipo de confrontación.

En el diseño de Windows NT han confluído aportaciones de tres modelos: el modelo cliente-servidor, el modelo de objetos, y el modelo de multiprocesamiento simétrico.

- Modelo cliente-servidor. En la teoría de este modelo se establece un kernel que básicamente se encarga de recibir peticiones de procesos clientes y pasárselas a otros procesos servidores, ambos clientes y servidores ejecutándose en modo usuario. Windows NT pone el modelo en práctica pero no contempla el núcleo como un mero transportador de mensajes, sino que introduce en él aquellos servicios que sólo pueden ser ejecutados en modo kernel. El resto de servicios los asciende hacia subsistemas servidores que se ejecutan en modo usuario, independientes entre sí, y que por tanto pueden repartirse entre máquinas distintas, dando así soporte a un sistema distribuido (de hecho, el soportar los sistemas distribuidos fue otra de las grandes directivas de diseño de este SO).
- Modelo de objetos. Decir que no implementa puramente la teoría de este modelo, sino que más bien lo que hace es simplemente contemplar los recursos (tanto internos como externos) como objetos.. Brevemente, señalar que todo objeto ha de poseer identidad propia (es único y distinguible de todos los demás), y una serie de atributos (variables) y métodos (funciones) que modifican sus atributos. Los objetos

interaccionan entre sí a través del envío de mensajes. No sólo existen en Windows NT objetos software (lógicos), sino que los dispositivos hardware (físicos) que también son tratados como objetos (a diferencia de UNIX, que trata a los dispositivos como archivos).

- **Modelo de multiprocesamiento simétrico.** Un SO multiproceso (o sea, aquel que cuenta con varias CPU y cada una puede estar ejecutando un proceso) puede ser simétrico (SMP) o asimétrico (ASMP). En los sistemas operativos SMP (entre los que se encuentran Windows NT y muchas versiones de UNIX) cualquier CPU puede ejecutar cualquier proceso, ya sea del SO o no, mientras que en los ASMP se elige una CPU para uso exclusivo del SO y el resto de CPU quedan para ejecutar programas de usuario. Los sistemas SMP son más complejos que los ASMP, contemplan un mejor balance de la carga y son más tolerantes a fallos (de manera que si un subproceso del SO falla, el SO no se caerá pues podrá ejecutarse sobre otra CPU, cosa que en los ASMP no sería posible, con lo que se bloquearía el sistema entero).

3.1.1 Los Subsistemas Protegidos.

Son una serie de procesos servidores, como se ve en la figura 3.1-1, que se ejecutan en modo usuario como cualquier proceso de usuario, pero que tienen algunas características propias que los hacen distintos. Se inician al arrancar el SO. Los hay de dos tipos: integrales y de entorno. Un subsistema integral es aquel servidor que ejecuta una función crítica del SO (como por ejemplo el que gestiona la seguridad). Un subsistema de entorno da soporte a aplicaciones procedentes de SO distintos, adaptándolas para su ejecución bajo Windows NT. Existen tres de este tipo: Win32, que es el principal, y proporciona la interfaz para aplicaciones específicamente construidas para Windows NT; OS/2, que da el entorno a aplicaciones procedentes del SO del mismo nombre; y POSIX, que soporta aplicaciones UNIX.

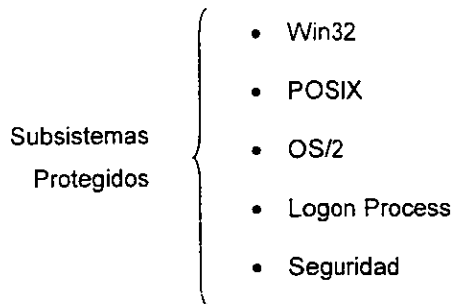


Figura 3.1- 1 Subsistemas Protegidos.

3.1.1.1 El Subsistema Win32.

Es el más importante, ya que atiende no sólo a las aplicaciones nativas de Windows NT, sino que para aquellos programas no Win32, reconoce su tipo y los lanza hacia el subsistema correspondiente. En el caso de que la aplicación sea MS-DOS o Windows de 16 bits (Windows 3.11 e inferiores), lo que hace es crear un nuevo subsistema protegido pero no servidor. Así, la aplicación DOS o Win16 se ejecutaría en el contexto de un proceso llamado VDM (Virtual DOS Machine, máquina virtual DOS), que no es más que un simulador de un ordenador funcionando bajo MS-DOS. Las llamadas al API Win16 serían correspondidas con las homónimas en API Win32. Microsoft llama a esto WOW (Windows On Win32).

El subsistema soporta una buena parte del API Win32. Así, se encarga de todo lo relacionado con la interfaz gráfica con el usuario (GUI), controlando las entradas del usuario y salidas de la aplicación. Por ejemplo, un buen número de funciones de las bibliotecas USER32 y GDI32 son atendidas por Win32, ayudándose del Executive cuando es necesario.

3.1.1.2 El subsistema POSIX.

La norma POSIX (Portable Operating System Interface for Unix) fue elaborada por IEEE para conseguir la portabilidad de las aplicaciones entre distintos entornos

UNIX. La norma se ha implementado no sólo en muchas versiones de UNIX, sino también en otros SO como Windows NT, VMS, etc. Se trata de un conjunto de 23 normas, identificadas como IEEE 1003.0 a IEEE 1003.22, o también POSIX.0 a POSIX.22, de las cuales el subsistema POSIX soporta la POSIX.1, que define un conjunto de llamadas al sistema en lenguaje C.

El subsistema sirve las llamadas interaccionando con el Executive. Se encarga también de definir aspectos específicos del SO UNIX, como pueden ser las relaciones jerárquicas entre procesos padres e hijos (las cuales no existen en el subsistema Win32, por ejemplo, y que por consiguiente no aparecen implementadas directamente en el Executive).

3.1.1.3 El subsistema OS/2.

Igual que el subsistema POSIX proporciona un entorno para aplicaciones UNIX, este subsistema da soporte a las aplicaciones OS/2. Proporciona la interfaz gráfica y las llamadas al sistema; las llamadas son servidas con ayuda del Executive.

3.1.1.4 El subsistema Proceso de inicio (Logon Process).

El proceso de inicio (Logon Process) recibe las peticiones de conexión por parte de los usuarios. En realidad son dos procesos, cada uno encargándose de un tipo distinto de conexión:

- El proceso de inicio local: gestiona la conexión de usuarios locales directamente a una máquina Windows NT
- El proceso de inicio remoto: gestiona la conexión de usuarios remotos a procesos servidores de Windows NT

3.1.1.5 El subsistema de Seguridad.

Este subsistema interacciona con el proceso de inicio y el llamado monitor de referencias de seguridad (del que trataremos en el Executive), y de esta forma se construye el modelo de seguridad en Windows NT.

El subsistema de seguridad interacciona con el proceso de inicio, atendiendo las peticiones de acceso al sistema. Consta de dos subcomponentes:

- La autoridad de seguridad local: es el corazón del subsistema. En general gestiona la política de seguridad local; así, se encarga de generar los permisos de acceso, de comprobar que el usuario que solicita conexión tiene acceso al sistema, de verificar todos los accesos sobre los objetos (para lo cual se ayuda del monitor de referencias a seguridad) y de controlar la política de auditorías, llevando la cuenta de los mensajes de auditoría generados por el monitor de referencias. Las auditorías son una facilidad que proporciona Windows NT para monitorear diversos acontecimientos del sistema por parte del Administrador.
- El administrador de cuentas: mantiene una base de datos con las cuentas de todos los usuarios (login, claves, identificaciones ...). Proporciona los servicios de validación de usuarios requeridos por el subcomponente anterior.

3.1.2 El Executive.

No se debe confundir el Executive con el núcleo de Windows NT, aunque muchas veces se usan (incorrectamente) como sinónimos. El Executive consta de una serie de componentes software, como se ve en la figura 3.1-2, que se ejecutan en modo privilegiado, y uno de los cuales es el núcleo. Dichos componentes son totalmente independientes entre sí, y se comunican a través de interfaces bien definidas. En el diseño se procuró dejar el núcleo tan pequeño como fuera posible, y, como se verá, la funcionalidad del núcleo es mínima.

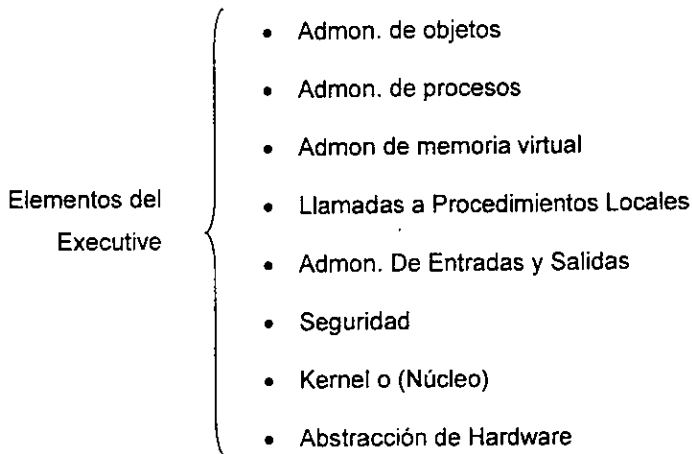


Figura 3.1- 2 Elementos del Executive.

3.1.2.1 El Administrador de Objetos (Object Manager).

Se encarga de crear, destruir y gestionar todos los objetos del Executive. Se tienen una gran cantidad de objetos: procesos, subprocesos, ficheros, segmentos de memoria compartida, semáforos, mutex, sucesos. Los subsistemas de entorno (Win32, OS/2 y POSIX) también tienen sus propios objetos. Por ejemplo, un objeto ventana es creado (con ayuda del administrador de objetos) y gestionado por el subsistema Win32. La razón de no incluir la gestión de ese objeto en el Executive es que una ventana sólo es innata de las aplicaciones Windows, y no de las aplicaciones UNIX o OS/2. Por tanto, el Executive no se encarga de administrar los objetos relacionados con el entorno de cada SO concreto, sino de los objetos comunes a los tres.

3.1.2.2 El Administrador de Procesos (Process Manager).

Se encarga (en colaboración con el administrador de objetos) de crear, destruir y gestionar los procesos y subprocesos. Una de sus funciones es la de repartir el tiempo de CPU entre los distintos subprocesos. Suministra sólo las relaciones más básicas

entre procesos y subprocesos, dejando el resto de las interrelaciones entre ellos a cada subsistema protegido concreto. Por ejemplo, en el entorno POSIX existe una relación filial entre los procesos que no existe en Win32, de manera que se constituye una jerarquía de procesos. Como esto sólo es específico de ese subsistema, el administrador de objetos no se entromete en ese trabajo y lo deja en manos del subsistema.

3.1.2.3 El Administrador de Memoria Virtual (Virtual Memory Manager).

Windows NT y UNIX implementan un direccionamiento lineal de 32 bits y memoria virtual paginada bajo demanda. El VMM se encarga de todo lo relacionado con la política de gestión de la memoria: determina los conjuntos de trabajo de cada proceso, mantiene un conjunto de páginas libres, elige páginas víctima, sube y baja páginas entre la memoria RAM y el archivo de intercambio en disco, etc.

3.1.2.4 Facilidad de Llamada a Procedimiento Local (LPC Facility).

Este módulo se encarga de recibir y enviar las llamadas a procedimiento local entre las aplicaciones cliente y los subsistemas servidores.

3.1.2.5 Administrador de Entrada/Salida (I/O Manager).

Consiste en una serie de subcomponentes, que son:

- el administrador del sistema de ficheros
- el servidor y el redirector de red
- los drivers de dispositivo del sistema
- el administrador de caches

Buena parte de su trabajo es la gestión de la comunicación entre los distintos drivers de dispositivo, para lo cual implementa una interfaz bien definida que permite el tratamiento de todos los drivers de una manera homogénea, sin que intervenga el cómo funciona específicamente cada uno.

Trabaja en conjunción con otros componentes del Executive, sobre todo con el VMM. Le proporciona la E/S síncrona y asíncrona, la E/S a archivos asignados en memoria y las caches de los ficheros.

El administrador de caches no se limita a gestionar unos cuantos buffers de tamaño fijo para cada fichero abierto, sino que es capaz de estudiar las estadísticas sobre la carga del sistema y variar dinámicamente esos tamaños de acuerdo con la carga. El VMM realiza algo parecido en su trabajo.

3.1.2.6 Monitor de Referencias a Seguridad.

Este componente da soporte en modo privilegiado al subsistema de seguridad, con el que interacciona. Su misión es actuar de alguna manera como supervisor de accesos, ya que comprueba si un proceso determinado tiene permisos para acceder a un objeto determinado, y monitoriza sus acciones sobre dicho objeto. De esta manera es capaz de generar los mensajes de auditorías. Soporta las validaciones de acceso que realiza el subsistema de seguridad local.

En UNIX, de la seguridad se encargaba un módulo llamado el Kerberos (Cancerbero), desarrollado por el MIT como parte del Proyecto Atenas. Kerberos se ha convertido en una norma de facto, y se incorporará a Windows NT en su versión 5.0.

3.1.2.7 El Núcleo (Kernel).

Situado en el corazón de Windows NT, se trata de un micro-kernel que se encarga de las funciones más básicas de todo el SO:

- ejecución de subprocesos
- sincronización multiprocesador
- manejo de las interrupciones hardware

3.1.2.8 Nivel de Abstracción de Hardware (HAL).

Es una capa de software incluida en el Executive que sirve de interfaz entre los distintos drivers de dispositivo y el resto del sistema operativo. Con HAL, los dispositivos se presentan al SO como un conjunto homogéneo, a través de un conjunto de funciones bien definidas. Estas funciones son llamadas tanto desde el SO como desde los propios drivers. Permite a los drivers de dispositivo adaptarse a distintas arquitecturas de E/S sin tener que ser modificados en gran medida. Además oculta los detalles de hardware que conlleva el multiprocesamiento simétrico de los niveles superiores del SO.

3.1.3 Llamadas a Procedimientos.

Windows NT, al tener una arquitectura cliente-servidor, implementa el mecanismo de llamada a procedimiento remoto (RPC) como medio de comunicación entre procesos clientes y servidores, situados ambos en máquinas distintas de la misma red. Para clientes y servidores dentro de la misma máquina, la RPC toma la forma de llamada a procedimiento local (LPC), en la figura 3.1-3 se mencionan estos procedimientos.

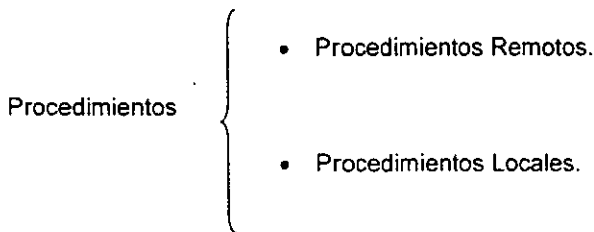


Figura 3.1- 3 Tipos de Procedimientos.

3.1.3.1 RPC (Remote Procedure Call).

Se puede decir que el sueño de los diseñadores de Windows NT es que algún día se convierta en un sistema distribuido puro, es decir, que cualquiera de sus componentes pueda residir en máquinas distintas, siendo el kernel en cada máquina el

coordinador general de mensajes entre los distintos componentes. El mecanismo de RPC permite a un proceso cliente acceder a una función situada en el espacio virtual de direcciones de otro proceso servidor situado en otra máquina de una manera totalmente transparente.

A continuación se ejemplifica: Se tiene un proceso cliente ejecutándose bajo una máquina A, y un proceso servidor bajo una máquina B. El cliente llama a una función "f" de una biblioteca determinada. El código de "f" en su biblioteca es una versión especial del código real; el código real reside en el espacio de direcciones del servidor. Esa versión especial de la función "f" que posee el cliente se denomina proxy. El código proxy lo único que hace es recoger los parámetros de la llamada a "f", construye con ellos un mensaje, y pasa dicho mensaje al Executive. El Executive analiza el mensaje, determina que va destinado a la máquina B, y se lo envía a través del interfaz de transporte. El Executive de la máquina B recibe el mensaje, determina a qué servidor va dirigido, y llama a un código especial de dicho servidor, denominado stub, al cual le pasa el mensaje. El stub desempaqueta el mensaje y llama a la función "f" con los parámetros adecuados, ya en el contexto del proceso servidor. Cuando "f" retorna, devuelve el control al código stub, que empaqueta todos los parámetros de salida (si los hay), forma así un mensaje y se lo pasa al Executive. Ahora se repite el proceso inverso; el Executive de B envía el mensaje al Executive de A, y este reenvía el mensaje al proxy. El proxy desempaqueta el mensaje y devuelve al cliente los parámetros de retorno de "f". Por tanto, para el cliente todo el mecanismo ha sido transparente. Ha hecho una llamada a "f", y ha obtenido unos resultados; ni siquiera tiene que saber si el código real de "f" está en su biblioteca o está en una máquina situada tres plantas más abajo. Esta es la elegancia de las RPC.

3.1.3.2 LPC (Local Procedure Call).

Las LPC se pueden considerar una versión descafeinada de las RPC. Se usan cuando un proceso necesita los servicios de algún subsistema protegido, típicamente Win32..

Por ejemplo: El proceso cliente tiene un espacio virtual de 4 Gb. Los 2 Gb inferiores son para su uso (excepto 128 Kb; ver el capítulo de la memoria). Los 2 Gb superiores son para uso del sistema.

El cliente realiza una llamada a la función `CreateWindow`. Dicha función crea un objeto ventana y devuelve un descriptor al mismo. No es gestionada directamente por el Executive, sino por el subsistema Win32 (con algo de colaboración por parte del Executive, por supuesto; por ejemplo, para crear el objeto). El subsistema Win32 va guardando en su propio espacio de direcciones una lista con todos los objetos ventana que le van pidiendo los procesos. Por consiguiente, los procesos no tienen acceso a la memoria donde están los objetos; simplemente obtienen un descriptor para trabajar con ellos. Cuando el cliente llama a `CreateWindow`, se salta al código de esa función que reside en la biblioteca `USER32.DLL` asignada en el espacio de direcciones del cliente. Por supuesto, ese no es el código real, sino el proxy. El proxy empaqueta los parámetros de la llamada, los coloca en una zona de memoria compartida entre el cliente y Win32, pone al cliente a dormir y ejecuta una LPC. La facilidad de llamada a procedimiento local del Executive captura esa llamada, y en el subsistema Win32 se crea un subproceso que va a atender a la petición del cliente. Ese subproceso es entonces despertado, y comienza a ejecutar el correspondiente código de stub. Los códigos de stub de los subsistemas se encuentran en los 2 Gb superiores (los reservados) del espacio virtual del proceso cliente. Aunque no se ha encontrado más documentación al respecto, es muy probable que dichos 2 Gb sean los mismos que se ven desde el espacio virtual de Win32. Sea como sea, el caso es que el stub correspondiente desempaqueta los parámetros del área de memoria compartida y se los pasa a la función `CreateWindow` situada en el espacio de Win32. Ése sí es el código real de la función. Cuando la función retorna, el stub continúa, coloca el descriptor en la ventana de la memoria compartida, y devuelve el control de la LPC al Executive. El subproceso del Win32 es puesto a dormir. El Executive despierta al subproceso cliente, que estaba ejecutando código proxy. El resto de ese código lo que hace es simplemente tomar el descriptor y devolverlo como resultado de la función `CreateWindow`.

3.1.4 Procesos.

Se debe tener cuidado con no confundir el proceso en Windows NT con el proceso en los SO más clásicos, como UNIX.

Un proceso es una entidad no ejecutable que posee un espacio de direcciones propio y aislado, una serie de recursos y una serie de subprocesos. En el espacio de direcciones hay colocado algún código ejecutable (entre otras cosas). En efecto, no puede ejecutar el código de su propio espacio de direcciones, sino que para esto le hace falta al menos un subproceso. Por consiguiente, un subproceso es la unidad de ejecución de código. Un subproceso está asociado con una serie de instrucciones, unos registros, una pila y una cola de entrada de mensajes (enviados por otros procesos o por el SO).

Cuando se crea un proceso, automáticamente se crea un subproceso asociado (llamado subproceso primario). Los subprocesos también se llaman "hebras de ejecución" (threads of execution). Debe quedar muy claro, pues, que lo que se ejecutan son subprocesos, no procesos. Los procesos son como el soporte sobre el que corren los subprocesos. Y entre los subprocesos se reparte el tiempo de CPU. Se puede pensar en los subprocesos de Windows NT como los procesos de los SO clásicos (aunque existen matices). A veces, por comodidad y por costumbre, se usaran ambos términos como sinónimos.

Un proceso tiene un espacio de direcciones virtuales de 4 Gb. En algún lugar de ese espacio se halla un código ejecutable (que quizás es la imagen de un programa en disco). Un proceso puede crear subprocesos, estando su número fijado por el sistema. Se dice que muere cuando todos sus subprocesos han muerto (incluso aunque el subproceso primario haya muerto, si aún existe algún subproceso propiedad del proceso, el proceso seguirá vivo).

3.1.5 Administración de la memoria.

La parte de Windows NT que soporta la gestión de la memoria se denomina Gestor de Máquina Virtual (VMM), y reside en el Executive, por encima del núcleo pero ejecutándose en modo supervisor.

3.1.5.1 Espacio de direcciones de un proceso.

Todo proceso que se crea en Windows NT posee un espacio de direcciones virtuales de 4 Gb exclusivos de él. Ningún otro podrá acceder a esas direcciones. Para un proceso, todo lo que ve es suyo, y ve virtualmente 4 Gb. Los 2 Gb superiores están reservados al SO, así como los 64 Kb primeros y los 64 Kb últimos de los 2 Gb inferiores. El resto de los 2 Gb inferiores son para uso del proceso.

- Direcciones para las DLL del sistema (NTDLL, KERNEL32, USER32, GDI32 ...).
- Direcciones para las DLL propias de la aplicación.
- Bloques y pilas de los subprocesos.
- Imagen del archivo ejecutable: código, datos, cabecera, información del depurador y tabla de activación imagen.

Además, cada proceso posee su propia tabla de implantación de páginas (TIP), a dos niveles. El objetivo de esa tabla (también llamada tabla de traducción) es, dada una dirección virtual, devolver su dirección física asociada. A veces la dirección virtual no tiene correspondencia en memoria física. Entonces se dice que se ha producido un fallo o defecto de página (page fault).

3.1.5.2 Funcionamiento del VMM.

Cada proceso tiene asignado un número que indica el máximo de páginas físicas que se le pueden conceder. Dicho número es ajustado por el llamado gestor del conjunto de trabajo, del que luego hablaremos. Además, cada proceso lleva asociada una lista que contiene referencias a las páginas físicas a las que se ha accedido menos últimamente.

Cuando el proceso accede a una dirección que no tiene dirección física asociada en la TIP, se produce un fallo de página. Entonces, el VMM consulta el número de páginas que el proceso tiene asignadas. Si no ha llegado al límite, se le concede una nueva página física y se escribe con la correspondiente desde el disco. Esto se denomina paginación bajo demanda. Si por el contrario ya había llegado al límite, entonces hay que descargar una página física al disco para subir a memoria la que ocasionó el fallo. El VMM elige la página víctima de la lista de menos recientemente usadas del proceso.

Nótese que aunque existan páginas libres en la memoria, si el proceso agota su número de páginas asignadas, se producirá el swapping, y además de una de sus propias páginas. Este mecanismo puede parecer ilógico, pero presenta tanto ventajas como inconvenientes. Como problema podemos pensar, por ejemplo, que si un proceso estuviera gran parte del tiempo dormido (en espera de un suceso o de una E/S), dicho proceso estaría ocupando páginas físicas que de otro modo podrían ser descargadas. Como ventaja se tiene que, con este esquema, procesos que requieran muchos recursos no dejarán fuera de juego a aquellos cuya demanda de memoria sea escasa. No obstante, el problema planteado es disminuido por una parte del VMM llamada gestor del conjunto de trabajo que realiza dos funciones fundamentales:

- Por un lado, periódicamente revisa las estadísticas sobre el uso de la CPU por cada proceso. Siguiendo este criterio, procede a ajustar el número que indica el máximo de páginas físicas asociadas a cada uno. A aquellos procesos con poca actividad se les bajará el número (lo que acarreará un descargue de sus páginas físicas, si las tiene, cuando sean necesarias), y a aquellos con mucha carga se les subirá el número.
- Por otro lado, y también de forma periódica, roba a los procesos páginas físicas, elegidas de entre las menos recientemente usadas por cada uno. Este procedimiento se realiza a la frecuencia necesaria para que los procesos no se ralenticen demasiado por los fallos de página. El objetivo de esto es mantener

una reserva de páginas para que una repentina demanda no ocasione una caída en prestaciones del sistema completo. Por ejemplo, cuando se inicia un proceso nuevo se suele requerir una cantidad considerable de páginas.

Para llevar a cabo estas tareas, las páginas físicas se clasifican en una de cuatro listas que son mantenidas por el gestor del conjunto de trabajo:

- lista de páginas modificadas
- lista de páginas no activas
- lista de páginas liberadas
- lista de páginas a cero

Cada página que el gestor roba a un proceso es borrada su lista de menos recientemente usadas e incluida en la lista de modificadas (y las entradas correspondientes de la TIP marcadas como no válidas, de forma que se produzca fallo de página al acceder el proceso a dichas direcciones). Dicha lista contiene páginas robadas que aún están en memoria RAM pero que no han sido escritas a disco (al archivo de paginación). Cuando se produzca un fallo de página, si la dirección física correspondiente estuviera en una de estas páginas, simplemente se volvería a insertar en la lista del proceso y se ajustaría su TIP.

Cuando la lista de modificadas se hace suficientemente grande, otra parte del VMM (el escritor de páginas) copia algunas páginas de la lista al archivo de paginación, las borra de la lista de modificadas y las añade a la lista de inactivas. Ahí están las páginas que han sido robadas, que están en RAM y además en el archivo de paginación. El tratamiento de un fallo de página aquí también sería muy rápido y simple.

Cuando un proceso libera memoria, sus páginas físicas asociadas se añaden a la lista de liberadas. Son, por tanto, potencialmente utilizables sin necesidad de escribirlas en el fichero de swapping pero su contenido no ha sido borrado (están tal y como las dejó el proceso propietario). Periódicamente, estas páginas van siendo

inicializadas con ceros, y añadidas a la lista de páginas a cero. El mecanismo de inicialización es para proteger la intimidad del antiguo proceso propietario. Cualquier página que se entrega a un proceso ha de haber sido convenientemente inicializada.

Cuando un proceso requiere memoria física, el VMM comienza cogiendo páginas de la lista de inicializadas. Cuando está vacía, toma de la lista de liberadas, y las inicializa. Cuando ésta se vacía, toma de la lista de inactivas, y las inicializa. Sólo como última opción recurre a las modificadas. Estas últimas requieren escritura en el archivo de paginación junto con inicialización, lo cual es un proceso lento.

En ambientes en los que la memoria es escasa, el gestor del conjunto de trabajo se centra en mantener un conjunto aceptable de páginas disponibles, más que en revisar las estadísticas de uso de la CPU.

3.1.5.3 Archivos asignados en memoria.

Un archivo asignado en memoria es todo aquel archivo para el que se ha reservado una región del espacio de direcciones virtuales de un proceso. Puede estar asignado el archivo completo o sólo una porción del mismo (llamada vista).

En principio, el VMM no asigna ninguna página física para el archivo asignado en memoria. El proceso simplemente supone que en ciertas direcciones de su espacio tiene cargado el fichero, así que cuando acceda a alguna se producirá un fallo de página. Es entonces cuando el VMM le asigna algunas páginas físicas y las copia desde el disco (paginación bajo demanda, como comentábamos antes). La gestión del VMM de los archivos asignados en memoria es como cualquier otra región del espacio direccionable del proceso, excepto que el swapping se hace directamente sobre el archivo (o sea, el archivo de intercambio es el propio archivo asignado en memoria).

Cuando se arranca un proceso con su código grabado en un fichero, el VMM asigna automáticamente dicho fichero en el espacio de direcciones del proceso. También asigna todas las bibliotecas incluidas explícitamente y todas aquellas a las que

se hace referencia en el código. Dentro del ejecutable existe una tabla llamada tabla de activación imagen, incluida por el enlazador, cuyas entradas contienen las funciones de biblioteca que se llaman durante el código. Una vez que se cargan las DLL (bibliotecas) en el espacio, el VMM completa la tabla escribiendo para cada entrada la dirección que ocupa la correspondiente función en el espacio del proceso. Por tanto, cada llamada a función implica una búsqueda en la tabla.

Suponiendo ahora que se arranca una segunda instancia del mismo proceso. Entonces el VMM pagina en el espacio de direcciones del nuevo proceso el fichero y las DLL, pero no vuelve a asignar páginas físicas, sino que ambas instancias comparten todo, al menos en principio. La ventaja de esto es ahorrar memoria, pero el inconveniente más claro es que si uno de los procesos modificara, por ejemplo, alguna variable global de su segmento de datos, el otro la tendría igualmente modificada. Esto ocurre evidentemente también en la pila e incluso en el mismo código (por ejemplo al ejecutar un depurador sobre una de las instancias para meter puntos de ruptura, se modificaría el código, lo cual implicaría que en la otra también).

Para solucionar el problema, Windows NT (y también UNIX) tiene una propiedad denominada "copiar antes de escribir". El VMM intercepta cualquier instrucción de escritura en el archivo mapeado en memoria por parte de las instancias. Cuando ocurre, asigna una o varias páginas físicas para la instancia escritora, y copia los contenidos de las páginas originales en las nuevas. A partir de ahora, esa instancia posee su propia región del archivo para modificar a su antojo. Análogamente para datos y pila.

UNIX también trabaja así, aunque duplica las zonas de datos y pila por defecto, de forma que múltiples instancias comparten, en principio, sólo el código.

Con respecto a la tabla de activación imagen, decir que el VMM asigna las DLL por defecto en direcciones fijas dentro del espacio de direcciones del proceso. Así, múltiples instancias del mismo pueden compartir la misma tabla. No obstante, un

proceso puede especificar la dirección base de cada DLL; en ese caso, el proceso tendría su propia tabla en memoria, con lo que resulta más eficiente que todos los procesos tengan las DLL asignadas en las mismas direcciones.

3.1.5.4 Uso de memoria virtual por parte del programador.

El mecanismo de la memoria virtual, que consiste en dos etapas la reserva y la asignación, permite situar objetos de grandes dimensiones en el espacio de direcciones del proceso, cuyo tamaño real se desconoce al tiempo de compilación. Por ejemplo, una hoja de cálculo.

- La reserva no es más que indicar al SO que una región del espacio de direcciones del proceso se va a destinar a un determinado objeto u objetos.
- La asignación consiste en hacer corresponder toda o parte de la memoria antes reservada con memoria física.

El problema de usar memoria virtual explícitamente es que el programador debe llevar una lista de la memoria que ha asignado de entre la que ha reservado. Otra solución es instalar un manejador de excepciones, de manera que al acceder a una dirección reservada pero no asignada se eleve una excepción. El manejador de esa excepción hará la asignación y se continuará la ejecución.

3.1.5.5 Bloques (heaps).

Los bloques de memoria son muy útiles cuando el programador no necesita usar memoria virtual explícitamente. Podemos tener estructuras de datos distintas en bloques distintos de memoria, de manera que un fallo en la manipulación de estructuras de un tipo no repercuta en las demás. Además, de este modo se gestionaría más eficientemente la memoria, ya que al borrar estructuras y escribir otras no se provocaría fragmentación interna dentro del bloque (al ser todas las de un tipo del mismo tamaño). Una última razón sería el poder minimizar los fallos de página. Estructuras en un mismo

bloque es probable que compartan la misma página, luego podríamos acceder a todas las de un mismo bloque con sólo un fallo de página.

3.1.6 Sistemas de Ficheros (Archivos).

Windows NT soporta cuatro tipos de sistemas de ficheros (SF) distintos, y puede trabajar con los cuatro a la vez (un disco con varias particiones, en cada una un SF distinto). Son los siguientes:

- NTFS (New-Technology File System). Es el sistema de ficheros nativo de Windows-NT. Como características podemos señalar:
 1. Permite nombres de archivo de hasta 255 caracteres, sensibles al tipo de letra.
 2. Permite la gestión de medios de almacenamiento extraordinariamente grandes.
 3. Incorpora mecanismos para garantizar la seguridad.
 4. Soporta el concepto de enlace (por compatibilidad con el estándar POSIX)
 5. Es capaz de recomponerse rápidamente después de una caída del sistema.
 6. Soporta el estándar Unicode
- HPFS (High-Performance File System). Es el que usa el sistema operativo OS/2. Se ha incluido para dar soporte a las aplicaciones OS/2 y complementar así al subsistema del mismo nombre. No es capaz de recomponerse del todo bien después de una caída del sistema ni de asegurar la no corrupción de los datos.
- FAT (File Allocation Table). El que usa MS-DOS y Windows 16 bits. Es el SF más pobre, y se mantiene para dar soporte a las aplicaciones DOS.
- CDFS (CD-ROM File System). Es un SF que Microsoft ha desarrollado exclusivamente para montarse sobre los CD-ROM.

3.1.7 Entrada y Salida.

Windows NT soporta trabajo en red con varios protocolos de comunicaciones. Lo más importante es que las facilidades de red están integradas en el SO, lo que lo

distingue de DOS y de la mayoría de las versiones de UNIX, en los que las interfaces con la red eran un añadido al SO, y frecuentemente no se adaptaban del todo bien al mismo.

A continuación se describirá brevemente qué ocurre en cada uno de los niveles de implementación del modelo OSI:

- **En el nivel 0** (es una capa donde se definen los medios de transporte: cable, fibra óptica ...) aparece un dispositivo que es la tarjeta de interfaz a la red (Network Interface Card, NIC). El NIC conecta el bus interno de la máquina con la red, sirviendo de interfaz entre el nivel 0 y el 1 (nivel físico). Es contemplado por el SO como un periférico más, controlado a través de su driver correspondiente.
- **En el nivel 2** (nivel de enlace de datos) aparece un software llamado NDIS (Network Device Interface Specification), que es un interfaz entre los drivers de dispositivo del NIC y los protocolos de transporte.
- **En los niveles 3 (nivel de red) y 4 (nivel de transporte)** Windows NT sitúa el software de los protocolos de transporte. Soporta TCP/IP, NBF, NWLink, DLC y AppleTalk.
- **En el nivel 5 (de sesión)** aparecen dos interfaces con los protocolos de transporte, que son los Windows Sockets (WinSock) y la NetBIOS.
 - Un socket es un mecanismo para establecer una conexión entre dos máquinas. Actúa como una especie de tubería bidireccional para los datos. Fueron introducidos por primera vez por el UNIX de Berkeley, y Windows NT incorpora una versión especial llamada WinSock. Se utilizan cuando se quiere una comunicación a través del protocolo TCP/IP, IPX/SPX ...
 - La interfaz NetBIOS es usada por aquellas aplicaciones que deseen usar protocolos que se adapten a NetBIOS (como el NBF). Establece conexiones entre distintas máquinas de la red y se encarga de que la transmisión sea fiable una vez establecida la conexión.

- En la misma capa de sesión están dos subsistemas integrales gestionados por el administrador de la E/S, denominados el redireccionador y el servidor. El redireccionador es el responsable de enviar peticiones de E/S a lo largo de la red, cuando el fichero o dispositivo solicitados son remotos. Al servidor le llegan peticiones desde los redireccionadores clientes, y las gestiona de modo que alcancen su destino. Tanto servidores como redireccionadores son implementados como drivers del sistema de ficheros. Así, cuando un proceso quiere realizar una E/S, usará las mismas llamadas al sistema para acceso local o remoto, con lo que no necesita conocer la ubicación del recurso (fichero o dispositivo). Pueden existir múltiples parejas de redireccionadores-servidores ejecutándose concurrentemente.
- **En el nivel 7 (capa de aplicación)** existe un proceso llamado suministrador por cada redireccionador de la capa 5. Cuando una aplicación hace una llamada de E/S, un software llamado enrutador de suministradores (multiple provider router) determina el suministrador adecuado, y le envía la petición. El suministrador, a su vez, se la pasará al correspondiente redireccionador. Por ejemplo, el gestor de ficheros (del administrador de E/S) es una aplicación que usa los servicios de los suministradores.

3.2 Conceptos sobre redes.

La más simple de las redes conecta dos computadoras, permitiéndoles compartir archivos e impresos. Una red mucho más compleja conecta todas las computadoras de una empresa o compañía en el mundo. Para compartir impresoras basta con un conmutador, pero si se desea compartir eficientemente archivos y ejecutar aplicaciones de red, hace falta tarjetas de interfaz de red (NIC, NetWare Interface Cards) y cables para conectar los sistemas. Aunque se puede utilizar diversos sistemas de interconexión vía los puertos series y paralelos, estos sistemas baratos no ofrecen la velocidad e integridad que necesita un sistema operativo de red seguro y con altas prestaciones que permita manejar muchos usuarios y recursos.

Una vez instalada la conexión se ha de instalar el sistema operativo de red (NOS, Network Operating System). Hay dos tipos básicos de sistemas operativos de red: punto a punto y con servidor dedicado.

- *Punto a Punto*: Este es un tipo de sistema operativo que le permite a los usuarios compartir los recursos de sus computadoras y acceder a los recursos compartidos de las otras computadoras. Microsoft Windows NT, Novell Lite son sistemas operativos punto a punto.
- *Con Servidor Dedicado*: Es un sistema operativo con servidor dedicado, como es NetWare de Novell, una o más computadoras se reservan como servidores de archivos no pudiendo ser utilizados para nada más.

3.2.1 Componentes de una red.

Una red de computadoras esta conectada tanto por hardware como por software. El hardware incluye tanto las tarjetas de interfaz de red como los cables que las unen, y el software incluye los controladores (programas que se utilizan para gestionar los dispositivos y el sistema operativo de red que gestiona la red. A continuación se listan los componentes, tal y como se muestra en la figura 3.2-1.

- Servidor
- Estaciones de trabajo
- Placas de interfaz de red (NIC)
- Recursos periféricos y compartidos.

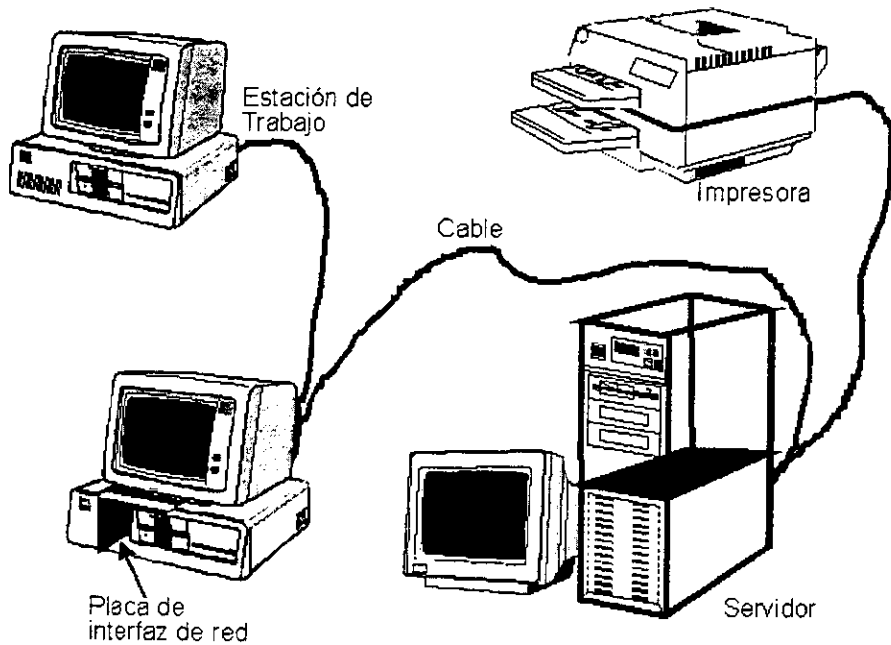


Figura 3.2- 1 Componentes de una red.

Servidor : este ejecuta el sistema operativo de red y ofrece los servicios de red a las estaciones de trabajo. **Estaciones de Trabajo:** Cuando una computadora se conecta a una red, la primera se convierte en un nodo de la última y se puede tratar como una estación de trabajo o cliente. Las estaciones de trabajos pueden ser computadoras personales con el DOS, Macintosh, Unix, OS/2 o estaciones de trabajos sin discos.

Tarjetas o Placas de Interfaz de Red: Toda computadora que se conecta a una red necesita de una tarjeta de interfaz de red que soporte un esquema de red específico, como Ethernet, ArcNet o Token Ring. El cable de red se conectara a la parte trasera de la tarjeta.

Sistema de Cableado: El sistema de la red esta constituido por el cable utilizado para conectar entre si el servidor y las estaciones de trabajo.

Recursos y Periféricos Compartidos: Entre los recursos compartidos se incluyen los dispositivos de almacenamiento ligados al servidor, las unidades de discos ópticos, las impresoras, los trazadores y el resto de equipos que puedan ser utilizados por cualquiera en la red.

Para realizar la conexión con una red son necesarias las tarjetas de interfaz de red y el cable (a menos que se utilice un sistema de comunicación sin cable). Existen distintos tipos de tarjetas de interfaz y de esquemas de cableados.

Hay tarjetas de interfaz de red disponibles de diversos fabricantes. Se pueden elegir entre distintos tipos, según se desee configurar o cablear la red. Los tres tipos mas usuales son ArcNet, Ethernet y Token Ring. Las diferencias entre estos distintos tipos de red se encuentran en el método y velocidad de comunicación, así como el precio. En los primeros tiempos de la informática en red (hace unos dos o tres años) el cableado estaba mas estandarizado que ahora. ArcNet y Ethernet usaban cable coaxial y Token Ring usaba par trenzado. Actualmente se pueden adquirir tarjetas de interfaz de red que admitan diversos medios, lo que hace mucho más fácil la planificación y configuración de las redes. En la actualidad las decisiones se toman en función del costo, distancia del cableado y topología.

El cable coaxial fue uno de los primeros que se usaron, pero el par trenzado ha ido ganando popularidad. El cable de fibra óptica se utiliza cuando es importante la velocidad, si bien los avances producidos en el diseño de las tarjetas de interfaz de red permiten velocidades de transmisión sobre cable coaxial o par trenzado por encima de lo normal. Actualmente el cable de fibra óptica sigue siendo la mejor elección cuando se necesita una alta velocidad de transferencia de datos.

3.2.2 Arquitectura de la red.

La arquitectura de una red viene definida por su topología, el método de acceso a la red y los protocolos de comunicación. Antes de que cualquier estación de trabajo pueda utilizar el sistema de cableado, debe definirse con cualquier otro nodo de la red.

La topología es el arreglo físico de los nodos y el medio de transmisión dentro de la estructura de red corporativa. La cuestión más importante al tener en cuenta la elegir el sistema de cableado es su costo, si bien también se ha de tener en cuenta el rendimiento total y su integridad.

3.2.2.1 Topología anillo.

Topología de red, ver figura 3.2-2, que consiste en una serie de repetidores conectados entre sí a través de enlaces de transmisión unidireccionales par formar un único ciclo cerrado. Cada estación en la red se conecta a la red en un REPETIDOR. Aunque conectadas de manera lógica en forma de anillo la topología de anillo suele estar organizada como una estrella de ciclo cerrado.

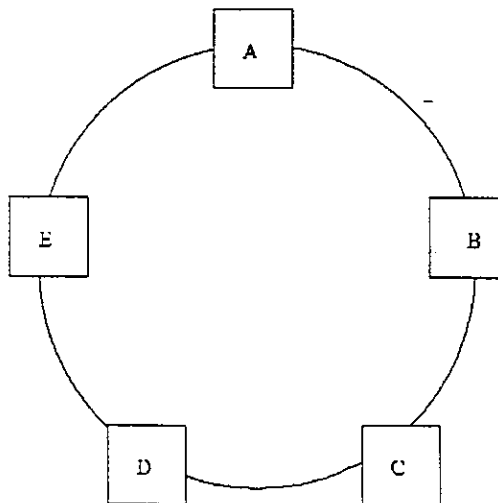


Figura 3.2- 2 Topología de anillo.

3.2.2.2 Topología Bus.

Arquitectura lineal de LAN, ver figura 3.2-3, en la cual los envíos desde las estaciones de la red se propagan a todo lo largo del medio y son recibidos por todas las demás estaciones.

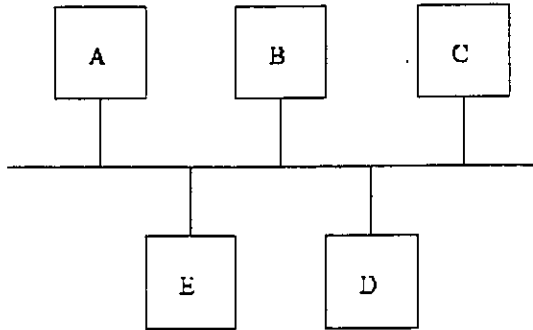


Figura 3.2- 3 Topología de Bus.

3.2.2.3 Topología Árbol.

Topología LAN, ver figura 3.2-4, similar a una topología bus excepto que las redes en árbol pueden contener ramas con nodos múltiples. Los envíos desde una estación se propagan por el medio y son recibidos por todas las demás estaciones.

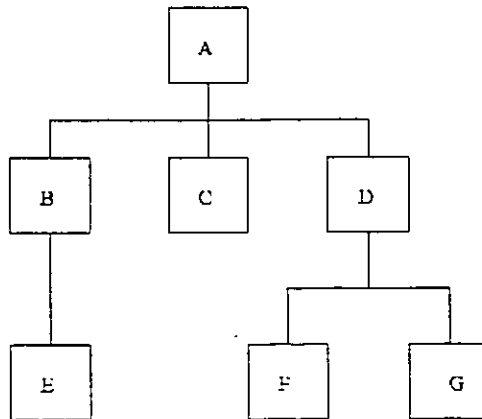


Figura 3.2- 4 Topología de árbol.

3.2.2.4 Topología Estrella.

Topología LAN, ver figura 3.2-5, en la que los puntos terminales de la red se conectan a una SWITCH central común a través de enlaces punto a punto. Una topología en anillo que está organizada como una estrella implementa una estrella unidireccional de lazo cerrado, en vez de enlaces punto a punto.

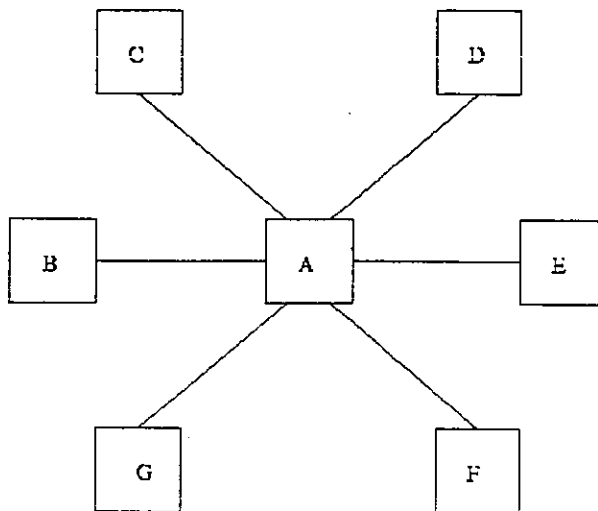


Figura 3.2- 5 Topología estrella.

3.2.2.5 Topología Malla.

Topología de red, ver figura 3.2-6 en la que los dispositivos están organizados de una manera segmentada con muchas o a menudo redundantes, interconexiones estratégicamente ubicadas entre los nodos de la red. Esta topología puede ser Parcial o Total. Una red parcial no proporciona el nivel de redundancia de una malla completa pero es menos cara de implementar.

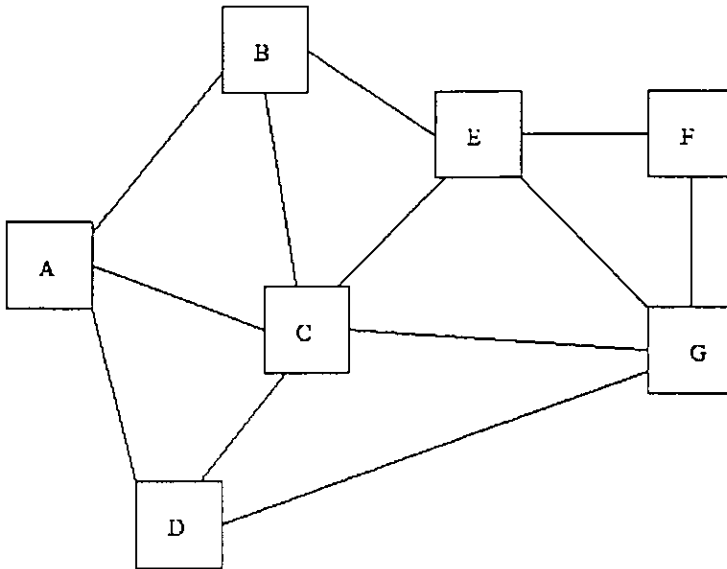


Figura 3.2- 6 Topología tipo malla.

3.2.3 Cobertura de las redes.

Existen redes de todos los tamaños. La red puede comenzar como algo pequeño y crecer con la organización. En la figura 3.2-7 se muestra el ámbito de cobertura de las redes.

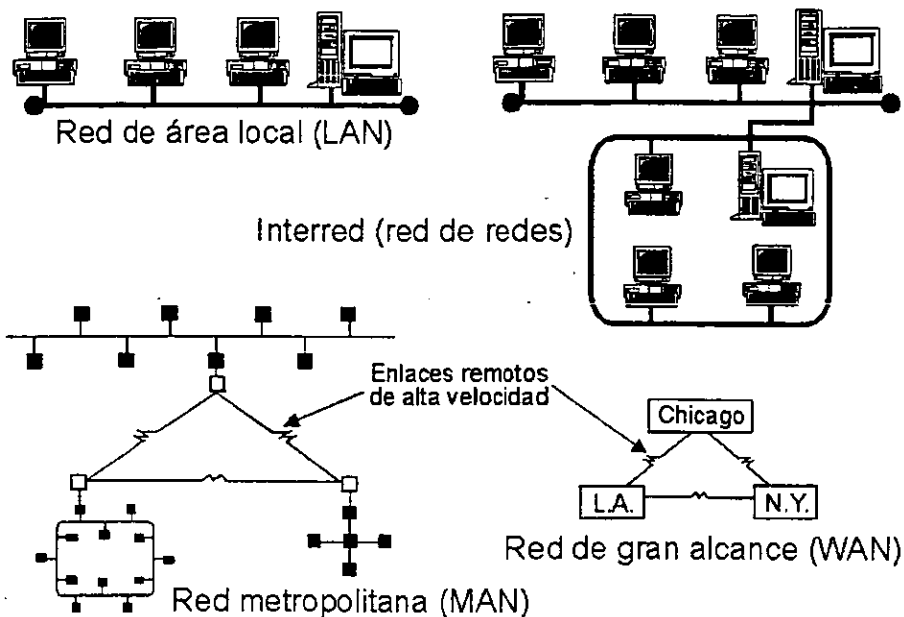


Figura 3.2- 7 Ambito de cobertura de las redes.

3.2.3.1 Red de área Local (LAN).

Red pequeña de 3 a 50 nodos, localizada normalmente en un solo edificio perteneciente a una organización.

3.2.3.2 Redes Interconectadas.

Una red de redes se encuentra formada por dos o mas segmentos de red local conectadas entre si para formar un sistema que puede llegar a cubrir una empresa.

3.2.3.3 Red Metropolitana (MAN).

Son normalmente redes de fibra óptica de gran velocidad que conectan segmentos de red local de una área específica, como un campus un polígono industrial o una ciudad.

3.2.3.4 Red de gran alcance (WAN).

Permiten la interconexión nacional o mundial mediante líneas telefónicas y satélites.

3.2.4 Razones para instalar una red de computadoras.

Instalar una red de computadoras puede ofrecer muchas ventajas para su trabajo. Estas son algunas ventajas ofrecidas al instalar una red de computadoras.

- *Compartición de programas y archivos.*
- *Compartición de los recursos de la red.*
- *Compartición de bases de datos.*
- *Expansión económica de una base en pc's.*
- *Posibilidad de utilizar software de red.*
- *Uso del Correo Electrónico.*
- *Creación de grupos de trabajo.*
- *Gestión centralizada.*
- *Seguridad.*
- *Acceso a mas de un sistema operativo.*
- *Mejoras en la organización de la empresa.*

3.2.5 Características de los sistemas operativos de red.

Los primeros S. O. de red ofrecían algunas utilidades de gestión de archivos de seguridad simples. Pero la demanda de los usuarios se ha incrementado de forma que los modernos sistemas operativos de red ofrecen una amplia variedad de servicios. Estos son algunos de ellos.

- *Adaptadores y cables de red.*
- *Nomenclatura global*
- *Servicios de archivos y directorios.*
- *Sistema tolerantes a fallos.*
- *Disk Caching (Optimización de acceso al disco).*
- *Sistema de control de transacciones (TTS, Transation Tracking System).*
- *Seguridad en la conexión.*
- *Bridges (Puentes) y Routers.*
- *Gateways (Pasarelas)*
- *Servidores Especiales*
- *Herramientas software de administración.*

3.2.6 Protocolo de comunicación.

Hace unos cuantos años parecía como si la mayor parte de los fabricantes de ordenadores y software fueran a seguir las especificaciones de la Organización internacional para el estándar (International Organization for Standarization, OSI). OSI define como los fabricantes pueden crear productos que funcionen con los productos de otros vendedores si la necesidad de controladores especiales o equipamientos opcionales. Su objetivo es la apertura. El único problema para implantar el modelo OSI fue que muchas compañías ya habían desarrollado métodos para interconectar su hardware y software con otros sistemas. Aunque pidieron un soporte futuro para los estándares OSI, sus propios métodos estaban a menudo tan atrincherados que el acercamiento hacia OSI era lento o inexistente. Novell y otras compañías de redes expandieron sus propios estándares para ofrecer soporte a otros sistemas, y relegaron los sistemas abiertos a un segundo plano. Sin embargo, los estándares OSI ofrecen un modo útil para comparar la interconexión de redes entre varios vendedores. En el

modelo OSI, hay varios niveles de hardware y el software. Podemos examinar lo que hace cada nivel de la jerarquía para ver como los sistemas se comunican por LAN.

3.2.6.1 Nivel de Protocolo.

Los protocolos de comunicaciones definen las reglas para la transmisión y recepción de la información entre los nodos de la red, de modo que para que dos nodos se puedan comunicar entre sí es necesario que ambos empleen la misma configuración de protocolos.

Entre los protocolos propios de una red de área local podemos distinguir dos principales grupos. Por un lado están los protocolos de los niveles físico y de enlace, niveles 1 y 2 del modelo OSI, que definen las funciones asociadas con el uso del medio de transmisión: envío de los datos en el ámbito de bits y trama, y el modo de acceso de los nodos al medio. Estos protocolos vienen unívocamente determinados por el tipo de red (Ethernet, Token Ring, etc.). El segundo grupo de protocolos se refiere a aquellos que realizan las funciones de los niveles de red y transporte, niveles 3 y 4 de OSI, es decir los que se encargan básicamente del encaminamiento de la información y garantizar una comunicación extremo a extremo libre de errores.

Estos protocolos transmiten la información a través de la red en pequeños segmentos llamados paquetes. Si un ordenador quiere transmitir un fichero grande a otro, el fichero es dividido en paquetes en el origen y vueltos a ensamblar en el ordenador destino. Cada protocolo define su propio formato de los paquetes en el que se especifica el origen, destino, longitud y tipo del paquete, así como la información redundante para el control de errores.

Los protocolos de los niveles 1 y 2 dependen del tipo de red, mientras que para los niveles 3 y 4 hay diferentes alternativas, siendo TCP/IP la configuración mas extendida. Lo que la convierte en un estándar de facto. Por su parte, los protocolos OSI representan una solución técnica muy potente y flexible, pero que actualmente esta escasamente implantada en entornos de red de área local.

3.2.6.2 Paquetes de Información.

La información es <<embalada>> en <<sobres>> de datos para la transferencia. Cada grupo, a menudo llamado paquetes incluye las siguientes informaciones - Datos a la carga. La información que se quiere transferir a través de la red, antes de ser añadida ninguna otra información. El término carga evoca a la pirotecnia, siendo la pirotecnia una analogía apropiada para describir como los datos son <<disparados>> de un lugar a otro de la red.

- Dirección. El destino del paquete. Cada segmento de la red tiene una dirección, que solamente es importante en una red que consista en varias LAN conectadas. También hay una dirección de la estación y otra de la aplicación. La dirección de la aplicación se requiere para identificar a que aplicación de cada estación pertenece el paquete de datos.
- Código de control. Informa que describe el tipo de paquete y el tamaño. Los códigos de control también códigos de verificación de errores y otra información.

3.2.6.3 Jerarquía de protocolo OSI.

Cada nivel de la jerarquía de protocolos OSI de la Figura 3.2-8 tiene una función específica y define un nivel de comunicaciones entre sistemas. Cuando se define un proceso de red, como la petición de un archivo por un servidor, se empieza en el punto desde el que el servidor hizo la petición. Entonces, la petición va bajando a través de la jerarquía y es convertida en cada nivel para poder ser enviada por la red.

Nivel Físico.

Define las características físicas del sistema de cableado, abarca también los métodos de red disponibles, incluyendo Token Ring, Ethernet y ArcNet. Este nivel especifica lo siguiente:

- Conexiones eléctricas y físicas.
- Como se convierte en un flujo de bits la información que ha sido empacada.

- Como consigue el acceso al cable la tarjeta de red.

7	Aplicación
6	Presentación
5	Sesión
4	Transporte
3	Red
2	Enlace de Datos
1	Física

Figura 3.2- 8 La jerarquía de protocolo OSI.

Nivel de Enlace de Datos.

Define las reglas para enviar y recibir información a través de la conexión física entre dos sistemas.

Nivel de Red.

Define protocolos para abrir y mantener un camino entre equipos de la red. Se ocupa del modo en que se mueven los paquetes.

Nivel de Transporte.

Suministra el mayor nivel de control en el proceso que mueve actualmente datos de un equipo a otro.

Nivel de Sesión.

Coordina el intercambio de información entre equipos, se llama así por la sesión de comunicación que establece y concluye.

Nivel de Presentación.

En este los protocolos son parte del sistema operativo y de la aplicación que el usuario acciona en la red.

Nivel de Aplicación.

En este el sistema operativo de red y sus aplicaciones se hacen disponibles a los usuarios. Los usuarios emiten ordenes para requerir los servicios de la red.

3.2.7 Interconexión de Redes.

Describe como extender una red utilizando repetidores, puentes, routers, adaptadores y otros dispositivos y métodos de interconexión de redes.

La Figura 3.2-9 muestra como se relaciona cada producto de interconexión de redes con el modelo de referencia OSI (Open System Interconexión). Las tareas que estos productos realizan sobre la red están relacionados con los niveles con los que son compatibles en la jerarquía de protocolos. Cuando mas alto se encuentre un producto en la pila de protocolo mas caro y complejo es.

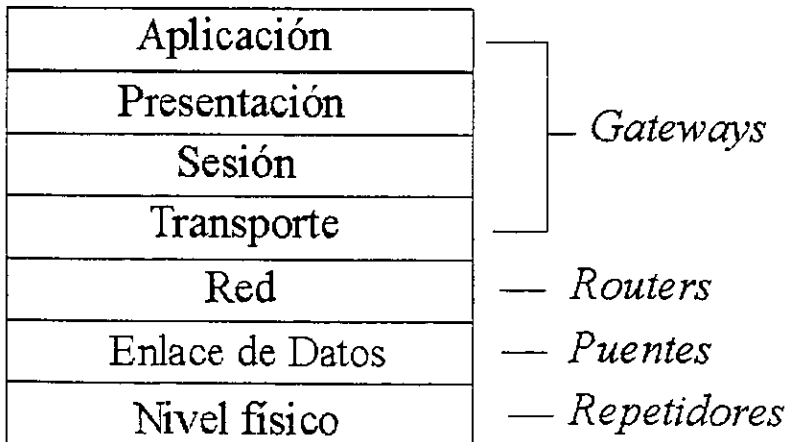


Figura 3.2- 9 Niveles de protocolos OSI utilizados por los dispositivos de interconexión de redes.

3.2.7.1 Repetidores.

A medida que las señales eléctricas se transmiten por un cable, tienden a degenerarse proporcionalmente a la longitud del cable. Este fenómeno se conoce como atenuación. Un repetidor es un dispositivo sencillo que se instala para amplificar las señales del cable, de forma que se pueda extender la longitud de la red ver figura 3.2-10. El repetidor normalmente no modifica la señal, excepto en que la amplifica para poder retransmitirla por el segmento de cable extendido. Algunos repetidores también filtran el ruido.

Un repetidor básicamente es un dispositivo "no inteligente" con las siguientes características:

- Un repetidor regenera las señales de la red para que lleguen mas lejos.
- Se utilizan sobre todo en los sistemas de cableado lineales como Ethernet.
- Los repetidores funcionan sobre el nivel mas bajo de la jerarquía de protocolos.
- Se utilizan normalmente dentro de un mismo edificio.
- Los segmentos conectados a un repetidor forman parte de la misma red. Los repetidores funcionan normalmente a la misma velocidad de transmisión que las redes que conectan.

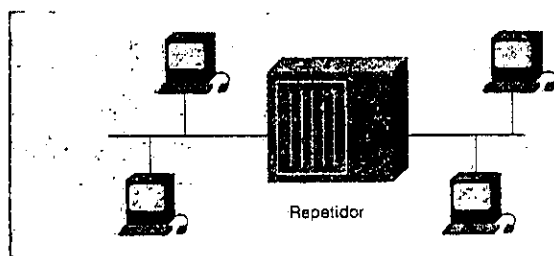


Figura 3.2- 10 Repetidores.

3.2.7.2 Puentes.

Un puente, ver figura 3.2-11, añade un nivel de inteligencia a una conexión entre redes. Conecta dos segmentos de red iguales o distintos. Podemos ver un puente como un clasificador de correo que mira las direcciones de los paquetes y los coloca en la red adecuada. Se puede crear un puente en un servidor NetWare instalando dos o mas tarjetas de interfaz de red. Cada segmento de red puede ser un tipo distinto (Ethernet, Token Ring, ArcNet). Las funciones de puente y routers incorporadas en el NetWare distribuyen el trafico de una red entre los segmento de LAN.

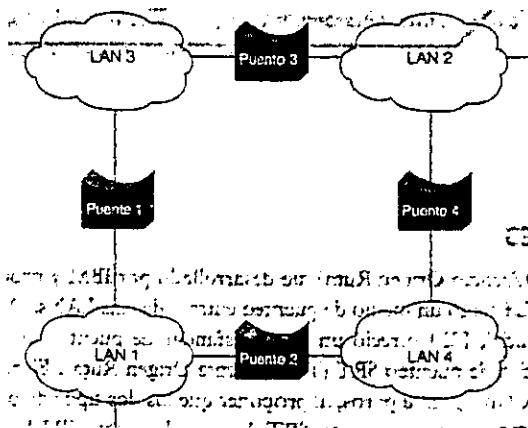


Figura 3.2- 11 Puentes.

Se puede crear un puente para dividir una red amplia en dos o mas redes mas pequeñas. Esto mejora el rendimiento al reducir el trafico, ya que los paquetes para estaciones concretas no tienen que viajar por toda la red. Los puentes también se usan para conectar distintos tipos de redes, como Ethernet y Token Ring. Los puentes trabajan en el nivel de enlace de datos. Cualquier dispositivo que se adapte a las especificaciones del nivel de control de acceso al medio (MAC, media Access Control) puede conectarse con otros dispositivos del nivel MAC. Recordemos que el nivel MAC es subnivel del nivel del enlace de datos.

3.2.7.3 Routers.

Son críticos para las redes de gran alcance que utilizan enlace de comunicación remotas, ver figura 3.2-12. Mantienen el tráfico fluendo eficientemente sobre caminos predefinido en una interconexión de redes compleja.

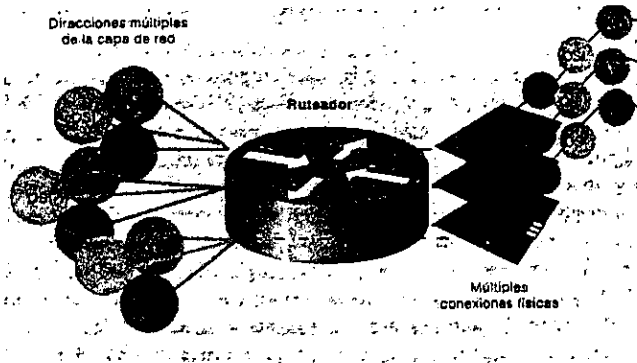


Figura 3.2- 12 Routers.

3.2.7.4 Switch.

Los switches son dispositivos, ver figura 2-13, de la capa de enlace de datos que, como los puentes permiten la interconexión de múltiples segmentos físicos de LAN en una sola red de gran tamaño. Los switches envían y distribuyen el tráfico con base en sus direcciones MAC. Hay muchos tipos de switches ATM, LAN y varios WAN.

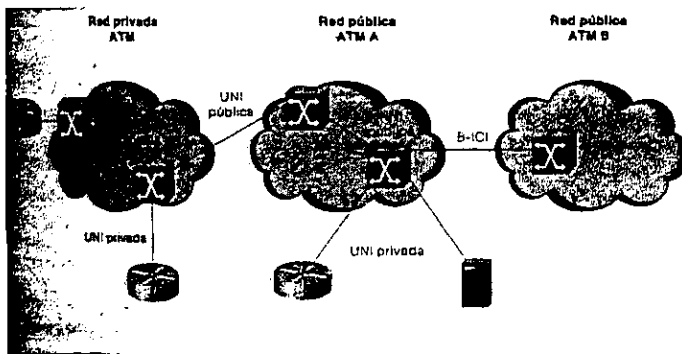


Figura 3.2- 13 Switch

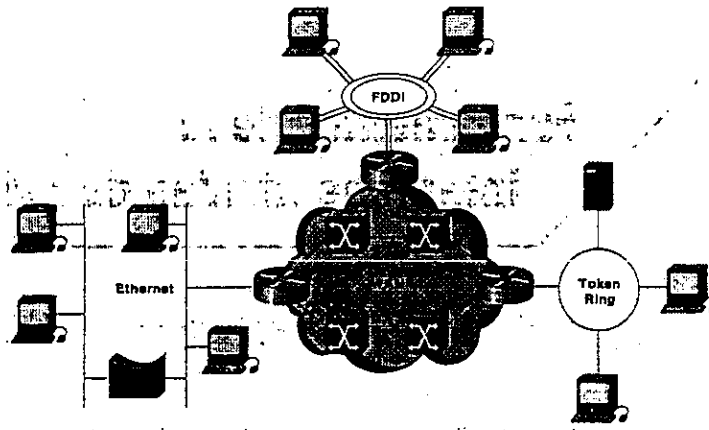


Figura 3.2- 14 Esquema general.

3.2.7.5 Enlace Principal (Backbone).

Un cable principal (Backbone) es un cable que conecta entre si dos o mas segmento de una red local y ofrece un enlace de datos de alta velocidad entre ellos. Mientras que un puente se establece instalando dos o mas tarjetas de red en un servidor, la interconexión de redes se realiza conectando varios servidores o segmentos de red local, generalmente con un enlace backbone. La figura 3.2-15 muestra un backbone basado en servidores. Cada servidor al backbone, ofrece conexión a los restantes segmentos de red conectados al backbone. Las otras tarjetas del servidor están conectadas a segmentos locales.

3.2.7.6 FDDI Y ATM.

En el nuevo entorno de conexiones de alta velocidad entre redes, se están usando como backbone dos tecnologías de transferencias de datos. Existe una creciente necesidad de más ancho de banda. Las estaciones de trabajo científicas y para ingeniería son comunes en las redes locales y globales. Estas requieren ancho de bandas al transferir grandes archivos gráficos y al conectarse a sistemas centrales (hosts). Las aplicaciones informáticas cliente- servidor que distribuyen en

procesamiento entre varias computadoras de una red también comparte la necesidad de un mayor ancho de banda. FDDI y ATM son posibles soluciones.

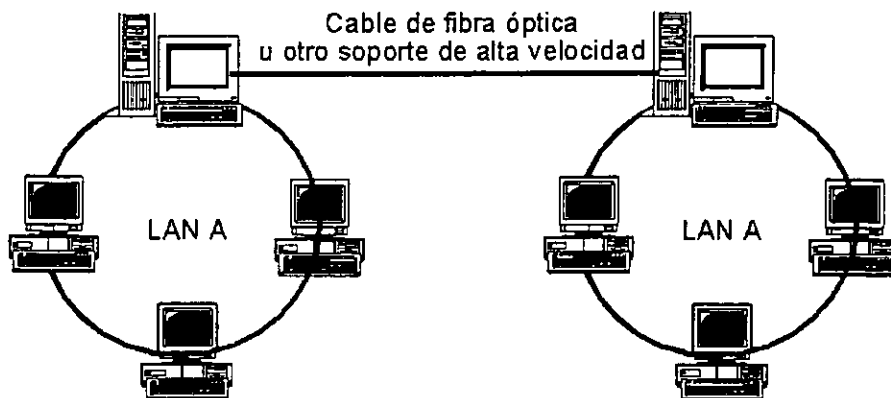


Figura 3.2- 15 Un backbone basado en servidores NetWare.

FDDI.

La Interfaz de datos distribuida de fibra (Fiber Distributed Data Interface, FDDI) es un estándar de cable de fibra óptica desarrollado por el comité X3T9.5 del American National Standards Institute (ANSI). Trabaja a 100 Mb/seg. Y utiliza una topología en anillo doble. FDDI se está implementando como backbone en redes en el ámbito de campus y de empresas.

Los anillos dobles en sentidos opuestos ofrecen redundancia. Si falla un anillo, el se reconfigura, como se muestra en la figura 3.2-16, de modo que se puede seguir aceptando tráfico en la red hasta que se corrija el error.

ATM.

ATM (Asynchronous Transfer Mode, Modo de transferencia asincrónica) es una tecnología de comunicación de datos de conmutación de paquetes de banda ancha diseñada para combinar las características de los multiplexores por división de tiempo con retardo dependiente (ATD) y redes locales de retardo variable. Los multiplexores por división de tiempo es un método para combinar señales separadas en una única

transmisión de alta velocidad. Con ATM se transmiten celdas provenientes de muchas fuentes. Pueden mezclarse, pero cada una tiene su dirección de destino específica, en la multiplexión por división de tiempo las señales llegan en orden en intervalos de tiempo regulares. En otras palabras, todas las celdas son del mismo tamaño, tanto en byte como en tiempo. El retardo variable es habitual en las redes locales, debido a que cada método de red puede utilizar un tamaño de paquete distinto. ATM divide los paquetes largos para adaptarlos a su tamaño de celda y los envía por el canal de datos; esto son reensamblados en el otro extremo.

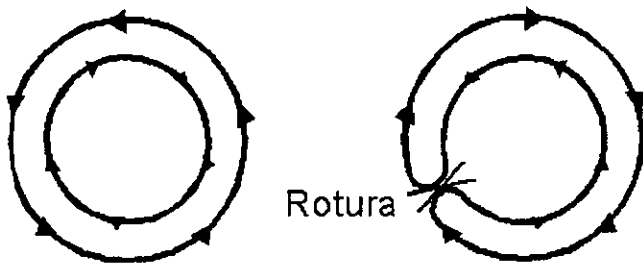


Figura 3.2- 16 FDDI se reconfigura automáticamente en un anillo normal cuando falla un enlace.

3.3 Microsoft Access 97.

3.3.1 Programación con Access 97.

Access 97 provee un conjunto de herramientas para el desarrollo de aplicaciones que hacen más eficiente esta actividad.

Una aplicación desarrollada en Access contempla los siguientes elementos:

- Tablas
- Consultas
- Formularios
- Informes
- Macros
- Módulos

En la figura 3.3-1 se muestra la pantalla principal del ambiente de desarrollo donde se indican los elementos de una aplicación en access 97.

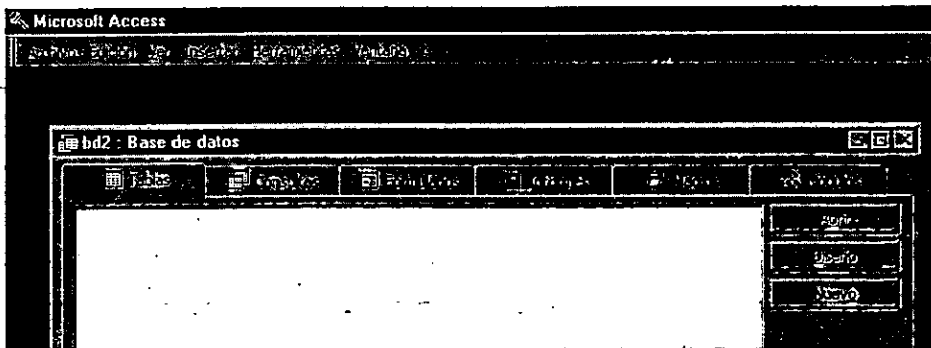
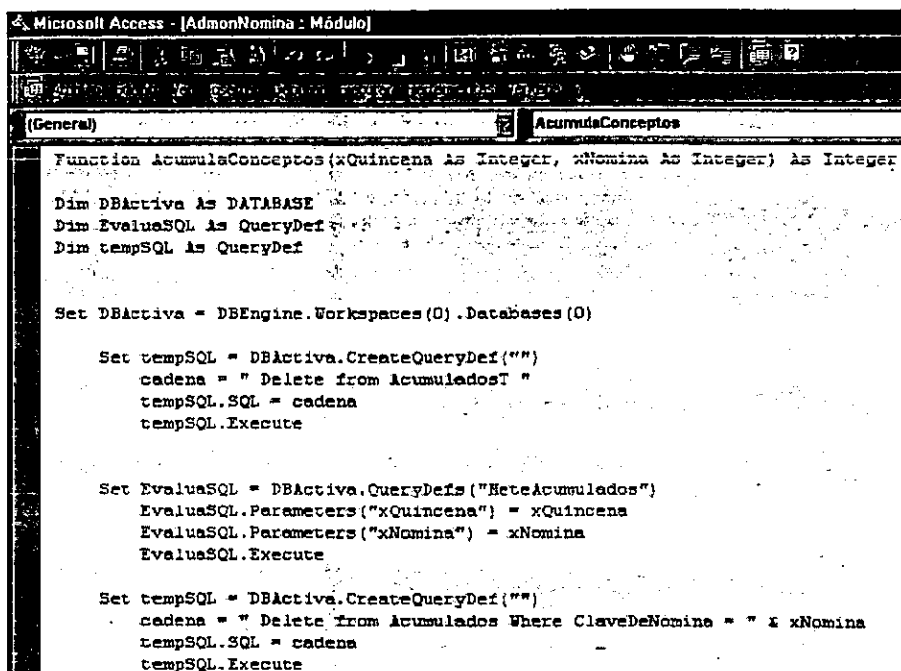


Figura 3.3- 1 Elementos de una aplicación en Access.

Aplicación de Visual Basic en Access 97.

La programación en Access 97 utiliza la sintaxis utilizada por Visual Basic, en la figura 3.3.-2 se muestra la pantalla y herramientas para crear código.



```
Microsoft Access - [AdmonNomina : Módulo]
AcumulaConceptos
Function AcumulaConceptos(xQuincena As Integer, xNomina As Integer) As Integer
Dim DBActiva As DATABASE
Dim EvalueaSQL As QueryDef
Dim tempSQL As QueryDef

Set DBActiva = DBEngine.Workspaces(0).Databases(0)

Set tempSQL = DBActiva.CreateQueryDef("")
cadena = " Delete from AcumuladosT "
tempSQL.SQL = cadena
tempSQL.Execute

Set EvalueaSQL = DBActiva.QueryDefs("HeteAcumulados")
EvalueaSQL.Parameters("xQuincena") = xQuincena
EvalueaSQL.Parameters("xNomina") = xNomina
EvalueaSQL.Execute

Set tempSQL = DBActiva.CreateQueryDef("")
cadena = " Delete from Acumulados Where ClaveDeNomina = " & xNomina
tempSQL.SQL = cadena
tempSQL.Execute
```

Figura 3.3- 2 Editor de código en Access 97.

Como se puede ver en la figura 3.3-2 el código que se utiliza en Access 97 combina las instrucciones o sentencias de cualquier lenguaje estructurado con las sentencias de acceso de datos.

El depurador es una herramienta que facilita la construcción del software ya que se puede verificar paso a paso la ejecución de los programas, en la figura 3.3- 3 las opciones que se tienen disponibles.

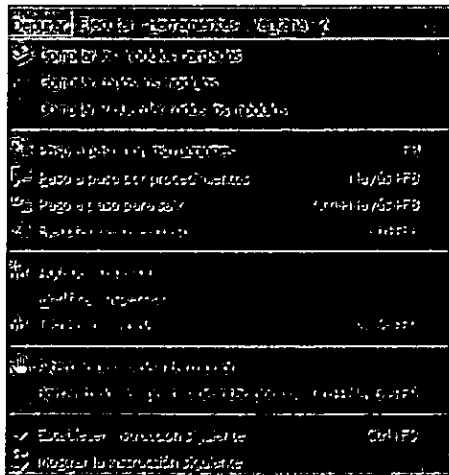


Figura 3.3- 3 Opciones de Depuración.

Las mismas instrucciones que se utilizan en la parte de módulos se pueden utilizar en formularios e informes, como se puede ver en la figura 3.3-4.

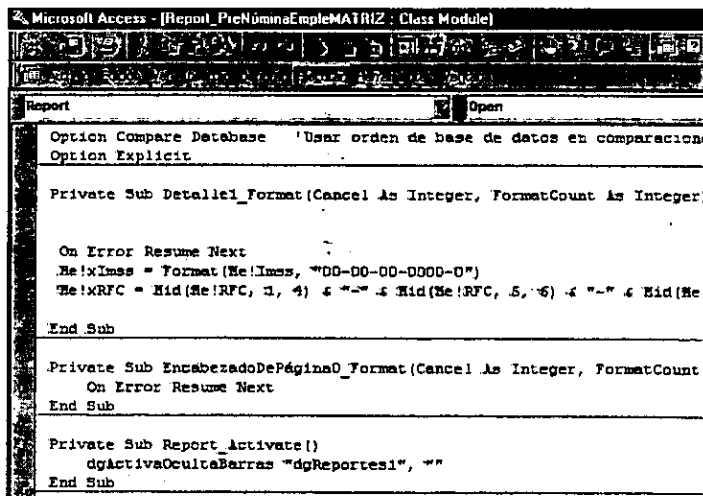


Figura 3.3- 4 Combinación de código en informes y formularios.

La administración de las tablas esta a cargo de la versión 3.5 del motor de base de datos Microsoft Jet el cual incluye las siguientes ventajas

- Un nuevo valor del registro, MaxLocksPerFile, permite acelerar la finalización de consultas grandes en servidores basados en NetWare y en Windows NT al forzar la confirmación parcial de una transacción. Use el método SetOption de DAO para suministrar un nuevo valor para este valor. El método SetOption permite anular ciertos valores del registro en tiempo de ejecución con el fin de ajustar la aplicación a entornos determinados.
- Las columnas indexadas presentan mejor concurrencia multiusuario, lo que significa que más usuarios pueden leer y actualizar las columnas indexadas sin obtener mensajes de conflicto de bloqueos.
- Las consultas grandes se ejecutan más rápidamente, debido a la mejora en el comportamiento de las transacciones de las instrucciones SQL del lenguaje de manipulación de datos (DML) y a los nuevos valores del registro que fuerzan la confirmación de las transacciones cuando se alcanza un cierto nivel de bloqueo.
- Las consultas que contienen el operador no igual (<>) en sus instrucciones de criterios son mucho más rápidas.
- Las lecturas secuenciales pueden ser más rápidas, ya que el motor de base de datos Jet puede preasignar ahora hasta 64 kilobytes de espacio en disco a la vez.
- Las consultas temporales se ejecutan más rápidamente.
- La eliminación de una tabla es más rápida cuando se usa la instrucción DROP de SQL o la instrucción DELETE de SQL sin predicado.

- La estructura de índices reduce la cantidad de espacio de almacenamiento requerido para los índices, y reduce la cantidad de tiempo requerido para crear índices complejos. Cuando se compacta una base de datos, sus índices se optimizan para un mejor rendimiento. Es necesario que compacte la base de datos de forma regular para mantener esta optimización.
- Las páginas indexadas ya no contienen bloqueos de lectura. Como resultado, el rendimiento se mejora cuando varios usuarios están trabajando con los mismos datos al mismo tiempo en una base de datos multiusuario.
- Las operaciones que incluyen la instrucción DELETE se ejecutan más rápidamente. Partes de una página pueden ser eliminadas todas al mismo tiempo, en lugar de fila a fila.
- El mecanismo de asignación de páginas del motor de base de datos Jet ha sido mejorado para incrementar la probabilidad de que los datos de la misma página se almacenen en páginas adyacentes. Como resultado, la capacidad de lectura previa del motor de base de datos Jet se ha mejorado.
- El motor de base de datos Jet es multiproceso. De forma predeterminada, un proceso realiza operaciones de lectura previa, otro realiza operaciones de escritura posterior y otro realiza mantenimiento del caché.
- Las transacciones implícitas se han incluido en el motor de base de datos Jet, de forma que puede obtener ventaja de la velocidad mejorada proporcionada por las transacciones, sin usar los métodos BeginTrans y CommitTrans en el código. Para retener todo el control cuando los datos se escriben al disco, debe generar transacciones explícitas usando los métodos BeginTrans y CommitTrans.
- El motor de base de datos Jet contiene un mecanismo de ordenación para mejorar el rendimiento de ordenación.

- El motor de base de datos Jet incluye una memoria caché configurada dinámicamente que se asigna en el inicio, basado en la memoria disponible del sistema. La memoria caché contiene los datos usados más recientemente para mejorar el rendimiento.
- El motor de base de datos Jet proporciona compatibilidad ISAM para los archivos HTML, para las versiones 5.0, 7.0, y 8.0 de Microsoft Excel, así como archivos de texto, hojas de cálculo Lotus 1-2-3 versión 3.0, Paradox 5.x, dBASE 5 y FoxPro 3.0.

3.3.2 Mejorar Rendimiento.

Microsoft Access 97 ofrece varias mejoras significativas en el rendimiento.

Retardo en la carga.	Microsoft Access sólo carga los componentes de software que son necesarios para las bases de datos, como Visual Basic para aplicaciones y DAO, cuando son necesarios. Esto reduce el tiempo que tarda en cargar las bases de datos y mejora el rendimiento global.
Formularios e informes simplificados.	Puede abrir muchos formularios e informes de manera más rápida, porque los formularios e informes que no tienen procedimientos de evento ya no incluyen un módulo de formulario o informe.
Propiedad Finalizar al producirse un error (FailOnError).	Puede optimizar las consultas de actualización en gran volumen para orígenes de datos ODBC enviando la consulta al servidor, donde todos los registros apropiados se procesan a la vez en lugar de registro por registro.
Carga sin llamadas mejorada.	Microsoft Access no carga los módulos, incluidos los módulos de formulario, hasta que se ejecuta el código de Visual Basic del módulo. Esto mejora el rendimiento global.
Administración de estado compilado mejorada.	Puede mantener el estado compilado de la base de datos incluso si la modifica. Sólo se descompilará el código modificado y aquel código que dependa del código modificado.
Velocidad de los controles ActiveX.	Microsoft Access ha mejorado el rendimiento de los controles ActiveX incrustados.
Cuadros combinados más rápidos.	Microsoft Access ha mejorado el rendimiento de los cuadros combinados en formularios.
Control de imagen	Puede mostrar imágenes independientes en los formularios e informes mucho más rápido utilizando un control de imagen en lugar de un marco de objeto independiente.
Analizador de rendimiento.	Este asistente analiza los objetos de base de datos y proporciona sugerencias para acelerar y mejorar su funcionamiento.

Vista previa de informes más rápida.	Puede ver los informes más rápidamente, porque los eventos sólo se activan la primera vez que pasa las páginas de un informe.
--------------------------------------	-------------------------------------------------------------------------------------------------------------------------------

3.3.3 Trabajar en Internet o en una Intranet.

Microsoft Access proporciona muchas características nuevas diseñadas para ayudar a utilizar fácilmente Internet y programar aplicaciones para el World Wide Web. Necesita un explorador de Web, tal como Microsoft Internet Explorer y un módem, una conexión de intranet u otra conexión de red para tener acceso a Internet y aprovechar algunas de estas nuevas características.

Importar o vincular archivos HTML.	<p>Importar o vincular tablas (de sólo lectura) o listas desde un archivo HTML utilizando el comando Obtener datos externos del menú Archivo.</p> <p>El Asistente para importación de HTML y el Asistente para vinculación de HTML también convierten los vínculos HTML a un campo con un tipo de datos Hipervínculo.</p>
Exportar objetos a formato HTML.	Exportar informes a formato HTML estático, y hojas de datos y formularios a formato HTML estático o dinámico utilizando el comando Guardar como o exportar del menú Archivo. Mejore la apariencia, la coherencia y el desplazamiento por sus páginas Web mediante un archivo de plantilla de HTML.
Asistente para publicar en el Web.	Crear una aplicación de Web con el Asistente para publicar en el Web mediante el comando Guardar como HTML del menú Archivo. Puede enviar hojas de datos, formularios o informes a formato HTML estático o dinámico utilizando uno o varios archivos de plantilla de HTML; crear una página principal; almacenar todos los archivos en una carpeta especificada como una publicación de Web, copiar los archivos a un servidor Web utilizando el Asistente para publicación en el Web y guarde un perfil de publicación de Web para utilizar posteriormente.
Acceso a servidores FTP y HTTP.	Importar o vincular datos (de sólo lectura) en servidores FTP o HTTP, exportar un objeto de base de datos a un servidor FTP y agregar o modificar una lista de ubicaciones de servidores FTP mediante el cuadro Buscar en en el cuadro de diálogo Abrir.
Hipervínculos.	Almacenar un hipervínculo en una tabla. Agregar la dirección del hipervínculo en una hoja de datos o formulario utilizando el botón Insertar hipervínculo. Un hipervínculo puede ser una ruta de acceso a un archivo en la unidad de disco duro, una ruta de acceso UNC o una dirección URL. Cuando hace clic en un campo de hipervínculo, Microsoft Access salta a un objeto, un documento, una página Web u otro destino.
Destinos de hipervínculos.	Saltar a una base de datos u objeto de Microsoft Access desde un hipervínculo en una hoja de datos, formulario o informe, o desde dentro de otros documentos de Microsoft Office, tales como un documento de Microsoft Word o una hoja de cálculo de Microsoft Excel. Saltar a un documento de Office desde un hipervínculo dentro de una hoja de datos,

	<p>formulario o informe. Hipervínculos en formularios e informes.</p> <p>Crear una etiqueta, control de imagen o botón de comando en un formulario o informe que salta a un destino de hipervínculo.</p>
Controles ActiveX.	Explorar World Wide Web, ver páginas Web, tener acceso a otros documentos en su PC, en la red o en Internet y transferir datos desde Internet a través de un formulario de la aplicación.
Controles ActiveX adicionales.	Programar aplicaciones Web brillantes y dinámicas. Disponible en Microsoft Office 97, edición Developer.
Herramientas de Office	Examinar fácilmente bases de datos y objetos de Microsoft Access o documentos de Office utilizando una funcionalidad similar a la de la barra de herramientas de Microsoft Internet Explorer.
Búsqueda rápida en el Web.	Ubicar bases de datos de Microsoft Access y otros documentos de Microsoft Office que están indexados mediante las propiedades de base de datos y archivo.
Réplica.	Sincronizar la réplica de una base de datos con una réplica o Diseño principal en un servidor de Internet.

3.3.4 Trabajar con otras aplicaciones

Microsoft Access 97 ofrece muchas características nuevas para trabajar con otros productos.

Ayudante de Office	El Ayudante de Office le ofrece Ayuda para su programa de Microsoft Office. Cuando necesite Ayuda, haga clic en el botón Ayudante de Office. El Ayudante proporciona de forma automática temas de Ayuda relacionados con aquello en lo que se encuentra trabajando, incluso antes de que haga alguna pregunta. También puede hacer una pregunta determinada escribiéndola en el Ayudante. Puede elegir un Ayudante que se adapte a su personalidad y a su forma de trabajar. El Ayudante es compartido por todos los programas de Office, por lo que cualquier opción que cambie afectará al Ayudante de todos los programas de Office.
Seguimiento de la actividad de la base de datos con Outlook Journal	Outlook Journal se utiliza para realizar el seguimiento de cuándo se abrió o cerró una base de datos, o cuando se imprimió un objeto. Para obtener detalles, consulte la Ayuda que se proporciona con Microsoft Outlook.
Obtener los contactos de Outlook desde una base de datos de Microsoft Access	Se pueden importar datos desde una base de datos de Microsoft Access a la carpeta Contactos de Outlook. También puede volver a exportar datos de Outlook a Microsoft Access. Para obtener detalles, consulte la Ayuda que se proporciona con Microsoft Outlook.
Retención del formato del texto	Cuando se copian o cortan datos desde formularios y hojas de datos de Microsoft Access, se retienen los valores de la fuente, alineación y formato de los números para los encabezados de las columnas y los datos cuando se pegan en hojas de cálculo de Microsoft Excel.
Envío de subinformes a otros formatos de archivo	Cuando se envía un informe con un subinforme a un formato de archivo .txt, .rtf, o .xls, el subinforme se envía junto con el informe principal.

Formato e impresión de datos de Microsoft Excel en informes de Microsoft Access	Cuando se necesita la funcionalidad de agrupación y formato de los informes de Microsoft Access para los datos que tenga en hojas de cálculo de Microsoft Excel, puede hacer clic en Informe de Access en el menú Datos de Microsoft Excel para iniciar el Asistente para informes de Microsoft Access. Para obtener los detalles, consulte la Ayuda que se proporciona con Microsoft Excel.
Mover de forma permanente datos de Microsoft Excel a Microsoft Access	Cuando una hoja de cálculo de Microsoft Excel deja de ser la herramienta apropiada para los datos, se puede crear una base de datos de Microsoft Access desde la hoja de cálculo haciendo clic en Convertir a Access en el menú Datos de Microsoft Excel. Para obtener los detalles, consulte la Ayuda que se proporciona con Microsoft Excel.
Creación de un control de tabla dinámica en un formulario de Microsoft Access	Cree un control de tabla dinámica en un formulario de Microsoft Access. Mediante el Asistente para tablas dinámicas, se puede crear un control en un formulario que permite resumir grandes cantidades de datos usando un formato y el método de cálculo que elija. Una tabla dinámica es como una consulta de tabla de referencias cruzadas, pero en la que puede cambiar dinámicamente los encabezados de las filas y de las columnas para obtener una vista distinta de los datos. (Para obtener ventaja de este asistente, necesita tener instalado Microsoft Excel.)
Vincular una hoja de cálculo	Se pueden vincular datos de una hoja de cálculo Microsoft Excel o Lotus 1-2-3, siempre que los datos estén organizados en el formato de tabla apropiado.
Uso de Microsoft Access como un componente ActiveX.	Cuando se utiliza Microsoft Access como un componente ActiveX, se pueden manipular objetos de Microsoft Access desde otra aplicación, tal como Microsoft Excel.
Arrastrar y colocar objetos de base de datos.	La siguiente funcionalidad de arrastrar y colocar está ahora disponible en Microsoft Access: <ul style="list-style-type: none"> • Puede arrastrar y colocar objetos de base de datos entre bases de datos abiertas de Microsoft Access. (Debe tener en ejecución dos instancias de Microsoft Access). • Puede arrastrar y colocar tablas y consultas de Microsoft Access en otras aplicaciones, tal como Microsoft Word y Microsoft Excel. • Puede crear una tabla arrastrando y colocando un rango de celdas desde una hoja de cálculo de Microsoft Excel a la ventana Base de datos. • Puede arrastrar y colocar objetos OLE en un campo de objeto OLE de un formulario en la vista Formulario, o de un formulario o informe en la vista Diseño.

3.3.5 Trabajar con datos

Microsoft Access 97 incorpora las siguientes características nuevas para hacer aún más fácil el trabajo con los datos.

Actualización de los datos en un cuadro de lista o cuadro combinado de un campo de Búsqueda	Para mejorar el rendimiento, los datos del cuadro de lista o cuadro combinado de un campo de Búsqueda que provienen de otra tabla ya no se actualizan de forma automática cuando se actualiza la otra tabla. Puede actualizar los datos del
---------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

	campo Búsqueda presionando la tecla F9.
Nueva interpretación de las fechas introducidas en un formato de año reducido	En los controles de hojas de datos y de modificación, Microsoft Access interpreta ahora de la siguiente forma las fechas introducidas con un formato de año reducido, para significar el siglo veintiuno.
Intervalo de fechas para el formato de año reducido Interpretación	1/1/00 hasta 12/31/29 1/1/2000 hasta 12/31/2029 1/1/30 hasta 12/31/99 1/1/1930 hasta 12/31/1999
Utilice réplica de base de datos para sincronizar múltiples copias de la base de datos	Mediante la réplica, múltiples usuarios en distintas ubicaciones pueden trabajar en sus propias copias de una base de datos y luego sincronizar los datos comunes de todas las réplicas. Puede crear una réplica parcial que contenga sólo un subconjunto de los registros de una réplica completa usando un filtro para restringir los datos a sincronizar. Por ejemplo, puede mantener el Diseño principal de la base de datos de ventas completa en las oficinas centrales y réplicas parciales de los datos regionales en las oficinas regionales. Los beneficios de este enfoque incluyen una mayor seguridad, réplicas menores de bases de datos, tráfico de red reducido, y menor costo de las telecomunicaciones.
Compatibilidad con IntelliMouse al desplazarse por una hoja de datos o formulario	Si tiene IntelliMouse, puede utilizarlo para desplazarse en una hoja de datos o formulario.
Arrastre y coloque los datos o envíelos a otras aplicaciones	Puede seleccionar los datos de un formulario u hoja de datos y arrastarlos y colocarlos en Microsoft Word o Microsoft Excel. También puede seleccionar datos de un formulario u hoja de datos y guardarlo como texto, como texto enriquecido (que puede abrirse con Microsoft Word y con la mayoría de los procesadores de texto de Microsoft Windows y de los programas de diseño electrónico), o como una hoja de trabajo de Microsoft Excel.
Compruebe la ortografía y corrija automáticamente los errores de escritura	Puede comprobar la ortografía de todos los campos Texto o Memo o de entradas seleccionadas en hojas de datos y formularios. También puede hacer que Microsoft Access corrija de forma automática los errores comunes de ortografía y de escritura mientras escribe, y convertir a mayúsculas los nombres de los días de la semana.
Orden de los subformularios	Ahora también se pueden ordenar los registros de un subformulario.
Guarde el orden de una tabla, consulta o formulario.	Cuando guarde una tabla, consulta o formulario, Microsoft Access guarda cualquier orden que haya establecido y lo vuelve a aplicar de forma automática cada vez que abra ese objeto. Si basa un nuevo formulario o informe en una tabla o consulta que tenga guardado un determinado orden con la misma, el formulario o informe hereda el orden. Cada vez que abra el formulario o informe, Microsoft Access vuelve a aplicar ese orden de forma automática.
Formato de hojas de datos	Las características de una hoja de datos se pueden cambiar de las siguientes formas:

	Cambiando la apariencia de las celdas para que parezcan elevadas o hundidas, cambiando el color de las líneas de la cuadrícula, mostrando sólo las líneas de la cuadrícula horizontales o verticales, o ambas.
Mostrando datos en un color y estilo de fuente distinto.	Animando las filas de tal forma que pueda verlas deslizarse hacia arriba y rellenar el hueco de un registro eliminado. Esta es la opción predeterminada, aunque puede desactivar la animación si lo prefiere.

3.3.6 Trabajar con filtros.

Microsoft Access 97 hace que filtrar sea más fácil incluso con las siguientes nuevas formas de filtrar los registros.

Filtrar por entrada.	Hacer clic con el botón secundario en un campo de formulario u hoja de datos, y luego en el menú contextual, escribir en el cuadro Filtrar por el valor exacto que está buscando en ese campo, o una expresión cuyo resultado desea usar como criterios.
Filtro por selección.	Directamente en el formulario u hoja de datos, seleccionar todo o parte de un valor, y luego hacer clic en el botón Filtro por selección en la barra de herramientas para buscar todos los registros que tengan el valor seleccionado.
Filtro excluyendo la selección.	Directamente en el formulario u hoja de datos, seleccionar todo o parte de un valor, y luego hacer clic con el botón secundario en el valor seleccionado, y luego hacer clic en Filtro excluyendo la selección en el menú contextual para buscar todos los registros que no contengan el valor seleccionado.
Filtro por formulario.	Especificar el valor o valores que está buscando en una vista del formulario u hoja de datos con campos en blanco (en lugar de datos). Especifique un valor que esté buscando escribiéndolo o eligiéndolo de una lista del campo. Nota Todavía puede crear filtros avanzados en la ventana Filtro u orden avanzado usando la cuadrícula de diseño, llamada anteriormente cuadrícula Consulta por ejemplo (Query By Example, QBE).
Filtrar registros de una tabla o consulta.	Usando cualquiera de las técnicas de filtrado, ahora puede filtrar los registros mostrados en una vista Hoja de datos de tabla o consulta.
Guardar un filtro con una tabla, consulta o formulario.	El filtro se guarda cuando se guarda la tabla, consulta o formulario; sin embargo, no se aplica de forma automática cuando se vuelven a abrir cualquiera de estos objetos. Se aplican cuando sean necesarios.
Basar un nuevo formulario o informe en los datos filtrados.	Puede crear un formulario o informe directamente desde un conjunto filtrado de registros al mismo tiempo que se muestran los registros en una tabla abierta o consulta, o puede basar el formulario o informe en una tabla cerrada o consulta que se haya guardado con un filtro. En cualquier caso, el formulario o informe hereda el filtro.
Aplicar, eliminar, o cambiar un filtro o un orden heredado.	Las siguientes propiedades le permiten manipular un filtro u orden que haya heredado un informe: Filtro (Filter), ActivarFiltro (FilterOn), OrdenarPor (OrderBy), y ActivarOrden (OrderByOn). Filtro (Filter) y OrdenarPor (OrderBy) se aplican también a los formularios, tablas, y consultas; sin embargo,

	estas propiedades se pueden cambiar usando las características de filtro y orden en lugar de establecer las propiedades directamente.
Aplicar un filtro a un subformulario.	El filtro limita los registros del subformulario sin limitar los registros del formulario principal.

3.3.7 Imprimir y vista preliminar

Microsoft Access 97 ofrece las siguientes mejoras para la impresión y vista preliminar de los objetos de base de datos.

Características de la vista preliminar	<ul style="list-style-type: none"> • El botón Vista de la barra de herramientas hace que sea fácil cambiar a las demás vistas del objeto de base de datos. • El botón Una página muestra una página. • El botón Dos páginas muestra dos páginas. • El botón Varias páginas le permite seleccionar el número de páginas de la vista preliminar. • El cuadro Zoom enumera el porcentaje de zoom en el que puede ver el objeto. • Si tiene Microsoft IntelliMouse, puede usar el botón redondo para moverse por la página actual.
Comandos y cuadros de diálogo	<p>Los siguientes comandos y cuadros de diálogo se han rediseñado o se les ha cambiado el nombre para que sean más fáciles de usar y para que coincidan con los utilizados en las demás aplicaciones de Microsoft Office:</p> <ul style="list-style-type: none"> • El cuadro de diálogo Imprimir es el mismo que se utiliza en las demás aplicaciones de Microsoft Office. • El comando Configurar impresión del menú Archivo se llama ahora Configurar página. • El botón Imprimir de la barra de herramientas imprime el objeto seleccionado sin mostrar el cuadro de diálogo Imprimir. • El comando Muestra preliminar se llama ahora Vista previa del diseño y está disponible en el menú Ver y haciendo clic en la flecha junto al botón Vista de la barra de herramientas. • El comando Definición de impresión se llama ahora Documentador y está disponible apuntando al comando Analizar del menú Herramientas.

CAPITULO 4 PLANTEAMIENTO DEL PROBLEMA Y PROPUESTA DE SOLUCIÓN.

4.1 Situación Actual.

Este grupo de empresas esta inmerso en un proceso de mejora continua que le permita mantener sus estándares de calidad, lo cual reditúa en mejores condiciones de competencia, sobre todo si se toma en cuenta que son altamente exportadoras.

Dentro de esta dinámica de mejora continua el área de recursos humanos comenzó un proceso de modernización para brindar un mejor servicio, para tal se procedió a realizar un diagnostico de la situación que guardaba el área de recursos humanos y poder detectar los puntos de mejora de una manera integral.

El diagnostico mostró que el departamento de recursos humanos requiere de una importante colaboración de los departamentos de contraloría y de sistemas así como del área corporativa compuesta esta ultima por un departamento de auditoría y otra de recursos humanos, en la figura 4.1-1 muestra un organigrama con el personal involucrado. A continuación se hará una descripción de las principales actividades.

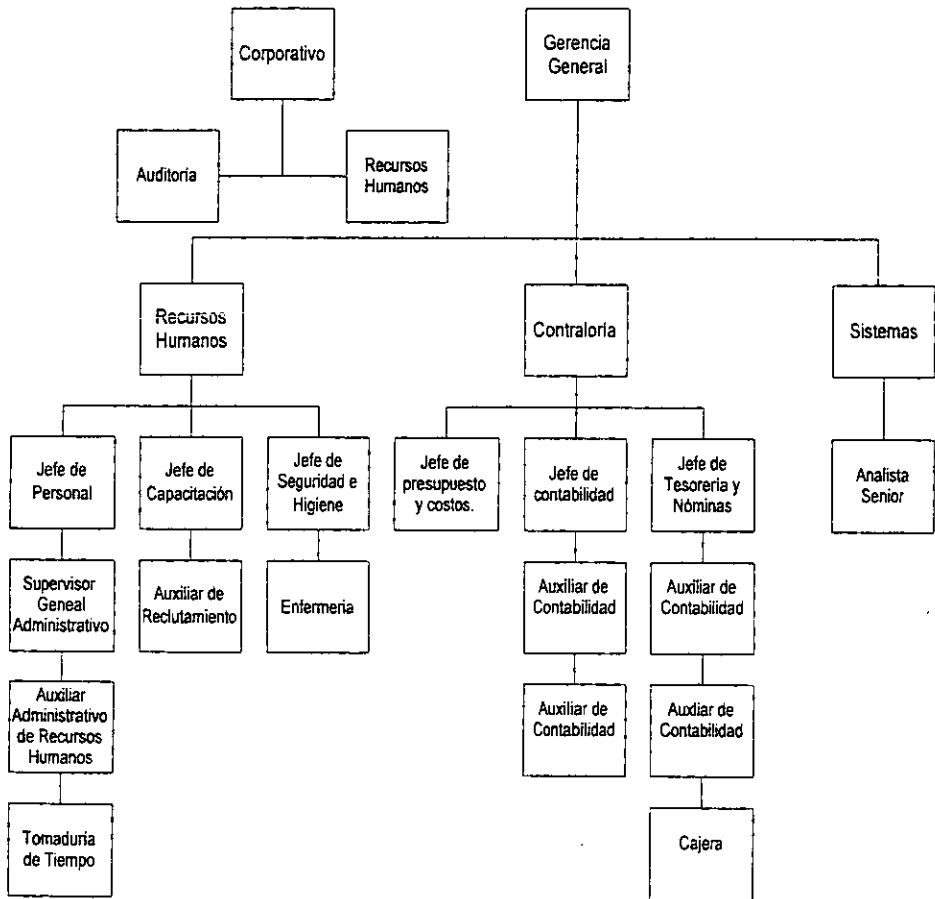


Figura 4.1- 1 Organigrama del personal involucrado en la administración de la nómina.

4.1.1 Actividades principales de Recursos Humanos.

Proceso	Personal que interviene	Información y Procesos que realizan.
Reclutamiento y Selección		
	Gerente de cualquier área	Envía memorándum al jefe de personal solicitando el personal.
	Jefe de personal	Analiza la solicitud y envía el memorándum a que se autorice por el gerente general.
	Gerente General	Con base en las restricciones del presupuesto y producción se regresa el memorándum al jefe de personal para iniciar el proceso de selección.
	Jefe de Personal	Con el memorándum autorizado lo canaliza al auxiliar de reclutamiento.
	Auxiliar de Reclutamiento	Con base en el perfil del memorándum consulta su cartera de aspirantes (La cual consta de un archivo en papel), para complementar esta lista se recurre a publicaciones para obtener más aspirantes. Realiza las entrevistas necesarias a cada aspirante, a los que reúnen los requerimientos se turnan al servicio médico para practicarles un examen.
	Servicio Medico	Registra en su archivo (en papel) los resultados del examen y envía solo un resumen del resultado al Auxiliar de Reclutamiento.
	Auxiliar de Reclutamiento	Selecciona al candidato y envía su expediente al Supervisor General Administrativo.
	Supervisor General Administrativo.	Revisa que el expediente este correctamente conformado y envía al Auxiliar Administrativo de Recursos Humanos para su contratación.
	Auxiliar	Llena solicitud de alta para capturarla en el sistema

	Administrativo de Recursos Humanos	de nómina de la HP-3000. Obtiene el código del trabajador de una lista previamente generada. Mecanografía su contrato. Calcula con base a su salario su nuevo integrado. Mecanografía su alta al IMSS con el integrado calculado. Mecanografía su credencial de la empresa. El trabajador queda contratado.
Administración de datos personales y familiares.		
	Auxiliar Administrativo de Recursos Humanos.	Lleva un registro en excel de los familiares directos del empleado: Padre, Madre, Cónyuge e hijos. Los datos que controla son nombre, sexo, parentesco y fecha de nacimiento, esto con el fin de presentar la información a la aseguradora de riesgos. Administra en el sistema de nómina HP-3000 los datos como dirección, teléfono, rfc, registro del IMSS, CURP.
Administración y mantenimiento de la plantilla de personal.		
	Supervisor General Adiministrativo.	Recibe la información de cambios de sueldo, departamento y puesto y los actualiza en la HP-3000.
	Auxiliar Administrativo de Recursos Humanos.	Si el cambio fue de sueldo se calcula manualmente el nuevo salario integrado y mecanografía el aviso de modificación de sueldo.
Evaluación al desempeño y Capacitación.		
	Gerente de cualquier área	El gerente junto con su personal establecen los objetivos de manera individual y se envían los documentos al Jefe de Capacitación. Al final del periodo establecido se califican con base

		<p>en el desempeño y se vuelve a enviar los resultados al Jefe de Capacitación.</p> <p>Establece las necesidades y del tipo de capacitación que requiere el personal y las comunica por memorándum al Jefe de Capacitación</p>
	Jefe de Capacitación	<p>Recibe las evaluaciones y requerimientos de capacitación para establecer el programa anual de capacitación, este control lo lleva en papel.</p> <p>Lleva un control en excel, del tiempo invertido en capacitación.</p> <p>Emite las constancias de capacitación que pide la Secretaría de Trabajo y Previsión Social, las cuales se mecanografian.</p>
Administración de Nominas.		
Tiempo Extra.		
	Supervisores de cada área	<p>Los supervisores de cada área envían un documento con la relación de los trabajadores que tienen tiempo extra en donde se especifica la fecha, la hora de inicio y la hora de termino. Esta información se entrega al Tomador de tiempo.</p>
	Tomador de tiempo.	<p>El tomador de tiempo recibe la relación y la codifica en papel para su captura en la HP-3000, este trabajo resulta muy tardado por que se requiere decirle al sistema cuantas son dobles cuantas triples y el costo de cada hora, además se separar el calculo que integra para el imss y el que grava para impuesto.</p>
Control de asistencia		
	Supervisor de cada área.	<p>Los supervisores de cada área envían al tomador de tiempo formatos varios con los datos que se requieren para la nómina y que se refieren a :</p> <p>Faltas y Permisos</p>

		Cambios temporales de categoría. Jornadas de trabajo. Control de vacaciones.
	Tomador de tiempo.	El tomador de tiempo recibe los formatos y los codifica para su captura en la HP-3000.
Administración de incapacidades		
	Enfermería	Recibe los formatos de incapacidad que emite el IMSS, las codifica y registran en excel para su control interno, y envía el formato al Auxiliar Administrativo de Recursos Humanos.
	Auxiliar Administrativo de Recursos Humanos	Recibe el formato y lo registra en una hoja de excel para su control interno. Registra las incapacidades en el SUA para presentar la información al IMSS. Envía la incapacidad al Tomador de tiempo.
	Tomador de tiempo.	Recibe el formato y codifica para su captura en la HP-3000.
Otras Incidencias.		
	Tomador de Tiempo.	Registra las incidencias varias como total de comedores (existe comedor en la empresa)
Administración de Ahorro.		
	Auxiliar de Contabilidad.	Recibe las solicitudes de inscripción, bajas y presentamos sobre el ahorro, las registra en excel para su control interno y envía un concentrado al capturista que registra en la HP-3000. Administra los acumulados de ahorro para su liquidación la cual es procesa por el Analista Senior de sistemas. Lo mismo aplica para Caja y Fondo de ahorro.
Administración de descuentos de fonacot		
	Auxiliar de	Recibe la información de FONACOT codifica en excel

	contabilidad	y captura en la HP-3000.
Administración de descuentos de infonavit.		
	Auxiliar de contabilidad.	Recibe la información de Infonavit codifica en excel, calcula el descuento por periodo y captura en la HP-3000.
Calculo de Finiquitos.		
	Gerente de cualquier área.	Comunica al jefe de personal la solicitud de baja.
	Jefe de personal.	Establece los criterios de negociación del finiquito.
	Supervisor Administrativo de Recursos Humanos	Realiza el calculo del finiquito, solicita información a contabilidad y al tomador de tiempo sobre algún adeudo a la compañía o de algún otro tipo. Mecanografía el finiquito Envía el finiquito al Auxiliar de contabilidad para el calculo de los impuestos.
	Auxiliar de contabilidad.	Verifica el calculo y calcula en un hoja de excel el impuesto por finiquito, y regresa al Supervisor Administrativo de Recursos Humanos. Se captura la póliza correspondiente del finiquito
	Supervisor Administrativo de Recursos Humanos.	Mecanografía el finiquito definitivo y prepara la carta de baja, envía el documento al Auxiliar Administrativo de Recursos Humanos.
	Auxiliar Administrativo de Recursos Humanos	Actualiza en la HP-3000 la baja. Mecanografía el aviso de baja para el IMSS.
Calculo de Liquidación al SUA.		
	Auxiliar Administrativo	Obtiene de su registro de excel las incapacidades. Solicita a sistemas prepare la información para la

	de Recursos Humanos	transferencia de información al SUA.
Calculo de integrados.		
	Auxiliar Administrativo de Recursos Humanos	<p>Solicita a sistemas que prepare la información para el calculo de integrados.</p> <p>Procesa la información y emite una sola vez los reportes.</p> <p>Para efectos de auditorías captura algunos resultados en excel para trabajar sobre ellos.</p> <p>Sistemas general el archivo que se envía al IMSS.</p>
Calculo de Nómina		
	Tomador de Tiempo. Sistemas Contabilidad.	<p>Es el que coordina que todas las incidencias se hayan capturado, cuando el lo indica solicita a sistemas que genera la nómina, una vez que termina se imprime una pre-nómina que se revisa por todos los involucrados, si existen cambios se realizan y se vuelve a generar el proceso. Una vez que ya no hay cambios se genera la nomina definitiva, al terminar se imprimen los reportes correspondientes, estos solo se imprimen una vez.</p> <p>La póliza se genera de manera automática. Si hubiera algún cambio se tendría que modificar manualmente.</p>
	Auxiliar de Contabilidad.	Una vez que se tiene el monto a pagar se envía a Servicios Panamericanos el listado para que entreguen los sobres correspondientes
Calculo Anual de Impuestos.		
	Auxiliar de Contabilidad.	<p>Verifica en el último mes los acumulados de las percepciones y deducciones, así como la parte gravable, los impuestos y créditos de cada trabajador.</p> <p>En el ultimo proceso del año se solicita a sistemas que active el proceso de ajuste anual.</p>

		Las constancias de percepciones y deducciones se mecanografian.
Declaración anual de impuestos.		
	Auxiliar de contabilidad.	En enero de cada año se debe presentar una declaración de impuestos sobre todo lo ejercido por conceptos de nómina, esta información se consolida y genera el archivo de manera manual. Se solicita a sistemas ayuda para obtener la información.
Calculo de Presupuesto de Nómina.		
	Jefe de presupuestos y costos.	Recopila la información de recursos humanos para poder estimar el costo del siguiente año, una vez que tiene los datos solicita a sistemas que prepare la aplicación para procesar el presupuesto. Esta aplicación genera una póliza para el presupuesto.
Elaboración de información para Recursos Humanos Corporativo		
	Supervisor General Administrativo.	La información la recolecta de todos los controles internos y la codifica en excel para enviarla a las oficinas corporativas, esta información es requerida cada mes.
Elaboración de información para auditorías internas y externas		
	Auxiliar de contabilidad. Auxiliar Administrativo de Recursos Humanos	Recolectan la información de sus controles internos para llenar los formatos establecidos por los auditores, esta actividad requiere de mucho tiempo por parte del personal de la empresa ya que no es posible obtener información histórica del sistema HP-3000.

4.1.2 Características principales del sistema de nómina HP-3000.

El sistema de nómina HP-3000 es un sistema cuya primera versión es de 1980, la cual se desarrollo sobre una plataforma UNIX y programada en Informix, el sistema se identifica como nomina HP-3000 por que el equipo en que se ejecuta es precisamente una HP-3000.

La situación que actualmente guarda este sistema son los siguientes:

- No esta preparado para el año 2000.
- El equipo HP-3000 ya no cuenta con el soporte del proveedor.
- Las adecuaciones al sistema se han hecho con parches por lo que no funciona correctamente.
- Los cálculos que se programaron no están totalmente apegados a ley.
- No cuenta con información histórica de personal.
- Requiere de un especialista en sistemas para cambiar cualquier paramento.
- Es dificii de utilizar.
- Cualquier información que se solicite requiere de la participación del área de sistemas.

Debido a que el sistema que esta funcionando actualmente no contempla todas las necesidades de información se lleva un control externo de mucha información que es en la mayoría de los casos duplicada y que no pueden compartir. Esto provoca que cuando se pide información consolidada resulte muy difícil de proporcionar además de no ser 100% confiable.

4.1.3 Proceso Administrativo.

El proceso administrativo que se sigue actualmente para el funcionamiento de recursos humanos involucra 13 personas, siendo el proceso de nómina el que mas recursos consume, esto principalmente al trabajo previo que requiere la nómina.

Existen una intervención de 5 personas del departamento de contraloría en el proceso de nómina, cuya actividad es la de intermediario de la información fuente ya que por requerimientos operativos requieren llevar un control de la información que el sistema HP-3000 no contempla.

Se realizó un estudio en el que se midió el tiempo que es necesario para generar una nómina, a continuación se muestra una tabla donde se resume:

Actividad	Realiza	Duración horas
Preparar Nómina para proceso	Sistemas (Analista Senior)	0.2
Codificar tiempo extra	Tomador de Tiempo	10.0
Capturar tiempo extra	Capturista (Contraloría)	2.0
Codificar Ausentismo	Tomador de Tiempo	2.5
Capturar Ausentismo	Tomador de Tiempo	1.0
Codificar Información de Ahorro.	Auxiliar de Contabilidad	0.5
Capturar información de Ahorro	Capturista(Contraloría)	0.2
Codificar información de Infonavit y fonacot.	Auxiliar de Contabilidad	0.1
Capturar información de infonavit y fonacot.	Capturista(Contraloría)	0.1
Codificar Información de Seguro de Autos.	Auxiliar de Contabilidad	0.2
Capturar información de Seguro de Autos.	Capturista(Contraloría)	0.1
Proceso de generación de nómina e impresión de reportes	Sistemas (Analista Senior)	3.0 (tiempo maquina)
Revisión de prenómina	Tomador de Tiempo.	2.0

Proceso de generación de nómina e impresión de reportes (2da ocasión)	Sistemas (Analista Senior)	3.0 (tiempo maquina)	
Revisión de prenómina	Tomador de Tiempo	0.5	
Generación de nómina y reportes definitivos.	Sistemas (Analista Senior)	2 (tiempo maquina)	
		Hombre	Maquina
Total de Tiempo		19.4	8
Gran total de tiempo		27.4	

4.2 Requerimientos del usuario.

Una vez que se presentó el estado actual del departamento de recursos humanos se decidió que el corporativo sería quien estableciera los nuevos lineamientos de manera general.

Con base en la información recolectada se decidió que se debería realizar un nuevo software que contemplara todas las necesidades de información de recursos humanos, pero deberá contemplar una serie de requerimientos.

El software deberá satisfacer las necesidades de manera integral de los departamentos de:

- Recursos Humanos
- Corporativo
- Contraloría
- Sistemas.

1. Características Técnicas.

- El sistema deberá ser gráfico.
- Usar un manejador de bases de datos relacional.
- Exportar datos de manera sencilla a Texto, Excel y Word.
- Importar datos de manera sencilla desde Excel y Texto
- Establecer niveles de seguridad por módulo y pantalla.
- La plataforma del manejador de bases de datos deberá ser UNIX o WindowsNT
- La aplicación deberá ejecutarse desde una computadora con Windows 95 o posterior.
- Cubrir el requisito para el año 2000.
- Participación de sistemas solo para el soporte (rendimiento, conectividad), pero no para la operación del sistema.

2. Poder parametrizar por el usuario los siguientes conceptos:

- Salarios mínimos
- Tasas de interes para prestamos
- Factor de subsidio.
- Factores para calculo del imss.
- Dias de Vacaciones.
- Dias de Aguinaldo.
- Porcentajes de Despensa.
- Porcentajes de Fondo y Caja de Ahorro.
- Factores fijos de Integración.
- Montos variables de integración.
- Tablas del ISR.
- Definir que concepto grava o cual se exenta.
- Definir base de calculo para el 2% estatal sobre nómina.
- Definir el esquema contable
- Definir porcentajes y cuentas de provisión contable.

3. Contemplar los siguientes procesos.

- Administración de contratos.
- Apegarse a la política de sueldos para empleados (puesto-nivel).
- Aumento de empleados por evaluación al desempeño.
- Calculo de nómina de gratificación con base al desempeño.
- Aumento automático de tabuladores para sindicalizados.
- Calculo automático de finiquitos.
- Emisión automática de avisos al imss
- Generación automática de salarios integrados
- Administración de Ahorro (Prestamos, Aportaciones, Liquidación, Acumulados).
- Calculo automático del ajuste anual de impuestos.
- Generar archivo de movimientos de ahorro para el banco
- Generación de archivo de póliza contable por proceso para carga en el MFG-Pro.

- Cancelación de recibos.
- Generar un archivo para depositar individualmente el monto de la nómina.
- Generar un archivo para solicitar vales de despensa.
- Generar el archivo de transferencia de datos al SUA.
- Generar el archivo de información de movimientos de salarios al IMSS.

4. Los módulos de:

- Reclutamiento y Selección que contemple todas las fases.
- Capacitación
- Consultas Gerenciales.
- Captura de incidencias por parte de supervisores y Jefes de Departamentos.
- Servicio Medico
- Administración de Acumulados (Declaración anual de impuestos)
- Presupuestos.
- Evaluaciones al desempeño.
- Auditoria.

5. Otros requerimientos.

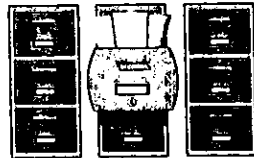
- Disminuir al máximo la intervención del área de sistemas en la operación.
- Designar un administrador del sistema.
- Transladar la operación del sistema de nómina a recursos humanos.
- Asignar la tarea de auditoría a contabilidad.
- Eliminar al máximo el uso de papel.
- Quien genere la incidencia que la registre en el sistema.
- Acceso vía red a información consolidada por parte del corporativo.
- Disminuir el tiempo de proceso y de respuesta al cliente (empleados, auditores, corporativo).
- Ofrecer una alternativa de registro electrónico de incidencias (relojes checadores)

En la figura 4.2-1 se muestra de manera gráfica la situación actual, donde como se explico en los puntos anteriores existe un sistema que solo resuelve un conjunto limitado de operaciones que esta fijas y no permiten fácil actualización, de igual manera se emite demasiado papel que debe ser recapturado por cada área con el fin de generar sus propia información que generalmente tiene errores y se presenta a destiempo.

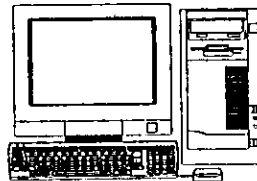
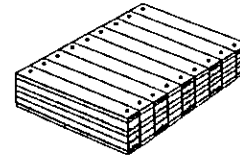
En la figura 4.2-2 se muestra la alternativa de solución de un sistema que realmente procese los datos para generar información para cada área evitando al máximo la recaptura y retrabajo con la información, la premisa de la propuesta es "Quien genera la información la captura en el sistema" y "Quien necesita la información la obtiene".

Consultas y Revisión En Papel

Registro en Papel



Demasiado Papel



Sistema que solo procesa la información

Figura 4.2-1 Sistema Actual



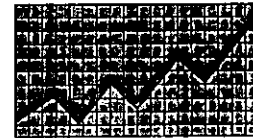
Corporativo



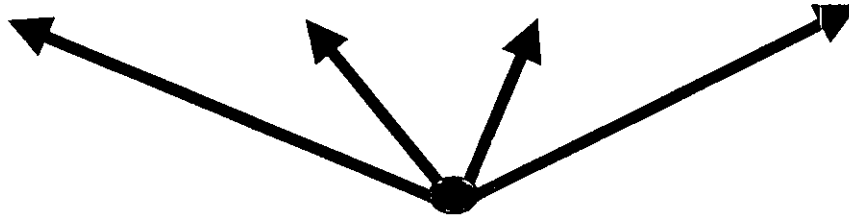
Auditorias



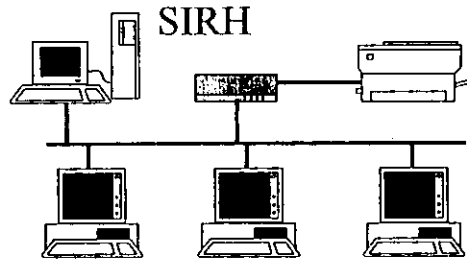
Gerencias



Consultas



141



SIRH

Captura en Planta
Contabilidad
Recursos Humanos
Servicio Medico



Figura 4.2-2 Sistema Propuesto

4.3 Recopilación y Análisis de la información.

La información que genera y que recibe el departamento de recursos humanos se lleva a cabo básicamente a través de documentos llenados manualmente, estos formatos varían del departamento que los envía ya que no existe una regla para su presentación y menos para la información mínima que debe contener; la situación más crítica es que incluso la comunicación es verbal o por costumbre, esta situación provoca que sea muy difícil rastrear el origen o el destino de la información y comprobar que es realmente correcta dicha información. En este apartado se presentan los formatos más importantes que se utilizan dando una breve explicación sobre su uso y contenido.

Recibo de Incapacidades.

En la figura 4.3-1 se muestra el formato que es llenado por el servicio médico (enfermera) cuando el trabajador entrega su incapacidad que le entrega el IMSS, el formato se le entrega al trabajador para comprobar cuando se presentó a la compañía.

Fecha _____

Nombre:	
Código Depto:	Turno:
Diagnostico:	-
No. de Incapacidad:	
A partir de:	Al:
Recibio:	Entrego:

E.G. ()

A.M.T. ()

Obrero ()

Figura. 4.3- 1 Recibo de Incapacidades de Servicio Médico.

Control Mensual de Incapacidades.

Después que se llena el Recibo de Incapacidades se vacía la información de la misma en el formato que se muestra en la figura 4.3-2, esta información es para uso exclusivo de servicio médico ya que requiere tener un concentrado de esta información.

Registro de incapacidades.

Servicio medico envía al departamento de recursos humanos cada semana todos los formatos de incapacidad que los trabajadores entregan, los cuales son codificados en el formato de la figura 4.3-4 que es el que utiliza el tomador de tiempo de R.H. para calcular manualmente el costo de la incapacidad y después capturarlo en la nómina HP-3000.

Fecha _____

Nombre del Departamento Solicitante:	Numero	Turno

Tiempo Extra para ser utilizado el día _____ de: _____

Nombre	Codigo	Hora		Tiempo	
		Entra(1)	Entra(2)	Solicitado	Real

Observaciones (Motivo y Trabajo a Realizar)	
Solicitante (Supervisor o Jefe de Detpo)	Autorizado Gerente

(1) La hora de entrada la anota el solicitante

(2) La hora de salida la anota vigilancia.

Figura 4.3- 3 Solicitud de Tiempo Extra

Solicitud de Tiempo Extra.

El tiempo extra es un proceso que tiene un gran interés para la empresa controlar, por su volumen, por su costo, por lo que representa desde el punto de vista productivo y por lo complicado de calcular para apegarse al 100% a la ley. En la figura 4.3-3 se muestra el formato que utilizan los supervisores de cada área para reportar el tiempo extra a recursos humanos, este formato contiene la fecha, hora de inicio y hora de fin del tiempo extra, este formato es codificado por el tomador de tiempo de R.H. para su captura en la nómina HP-3000, para su codificación no existe formato alguno sino que se utiliza una hoja tamaño carta para hacer sus anotaciones.

VELES TRACTIVOS, S.A. DE C.V.
 INCAPACIDADES RECORRIDAS EN

FICHA N.º _____
 NOMBRE DEL TRABAJADOR: _____
 CATEGORIA: _____

Código	N.º O. M. S. N.º 2	DIA	MES	AÑO	DIA	MES	AÑO

* CALCULAR SOBRE MISMO INTERVALO COMO SI FUERAN UNO (UNO) DÍA Y CONTINUO INCAPACITADO.

Figura 4.3- 4 Registro de Incapacidades.

CODIGO		PERIODO ANI	
NOMBRE			CA.
DEPARTAMENTO		CTA. CARGO	TNO.
PERIODO DEL		AL	

D I A	PRIMER TURNO		SEGUNDO TURNO		TERCER TURNO		HORAS TRABAJADAS	
	ENTRADA	SALIDA	ENTRADA	SALIDA	ENTRADA	SALIDA	NORMALES	HOR/ EXTR.
16								
17								
18								
19								
20								
21								
22								
23								
24								
25								
26								
27								
28								
29								
30								
31								
15								

CONFORME:

FIRMA DEL EMPLEADO

Figura 4.3- 5 Tarjeta Checadora.

Aviso de Asistencia.

Para el control de asistencia se utilizan tarjetas checadoras como se muestra en la figura 4.3-5 las cuales son codificadas en una hoja tamaño carta para su captura en nómina HP-3000, junto con este formato se utiliza el de la figura 4.3-6 que son los avisos de faltas, permisos y retardos en que incurrir los empleados que no checan tarjeta; este formato es llenado por el supervisor o jefe de departamento y es enviado a recursos humanos, en caso de no estar ninguno de los dos se llena por el personal de vigilancia ya que sin este documento no pueden entrar a la planta (no aplica a faltas).

1.Faltas		2.Permisos		3.Retardos		4.TiempoExtra	
Departamento				Fecha:			
Código	Nombre	Concepto	Motivo	Horas	Se Paga	Se Desc.	

Figura 4.3- 6 Aviso de asistencia.

Roll de Turno Defasados.

Por razones de mantenimiento y operación de las plantas existen trabajadores que su día de descanso no es el domingo sino un día entre semana, la ley contempla que si no se descansa el domingo se debe pagar una compensación o prima dominical, la relación del personal en esta situación es enviada por el supervisor al tomador de tiempo de R.H. usando el formato de la figura 4.3-7. El tomador de tiempo lo codifica en una hoja tamaño carta para después capturarlo en la nómina HP-3000. También es importante saber el día en que se descansa para tener un control preciso del personal que esta en la planta, sobre todo en caso de accidente.

Fecha _____

Nombre	Código	Día Trabajo	Día Descanso

Figura 4.3- 7 Roll de turno desfasado.

Constancia de Vacaciones.

El trabajador puede solicitar tomar sus vacaciones en cualquier momento sin embargo el otorgarlas dependerá de las necesidades de producción y de los días disponibles, pero al año deberá disfrutarlas, para solicitarlas el trabajador pide un formato a R.H. que se muestra en la figura 4.3-8, después lo debe autorizar su jefe inmediato de ser así lo entrega a R.H. para que revise que tiene días disponibles en su saldo de vacaciones, de ser así se autoriza, el tomador de tiempo de R.H. actualiza sus datos en su reporte de vacaciones, que se muestra en la figura 4.3-9, ya que es de ahí donde consultara si tiene días disponibles. También codificara en una hoja tamaño carta la información para capturar en la nómina HP-3000.

Fecha: _____ Período: _____
 Nombre: _____
 Código: _____ Depto. _____
 Fecha De Ingreso _____ Antigüedad _____
 No. de días a que tiene derecho: _____
 No. de días de adeudo de
 Período(s) anterior(es): _____
 No. de días ya disfrutados _____
 No. de días que quedan pendientes: _____
 Fecha de reanudación de labores: _____

Conforme Autoriza Vo.Bo.
 _____ _____ _____
 Trabajador Jefe Inmediato Rec. Hum.

Figura 4.3- 8 Constancia de Vacaciones.

Fecha _____

Código	Nombre	Saldo	Días Tomados	Semanas	Días Habiles
Cifras control					

Figura 4.3- 9 Reporte de Vacaciones.

Reporte de Percepciones por Aviso.

Existen conceptos que se pagan en nómina que no tienen un formato establecido para lo cual utilizan uno genérico, el cual se muestra en la figura 4.3-10, dicho formato puede ser llenado por el tomador de tiempo o por contabilidad para avisar de algún pago extraordinario que se fuera a realizar.

Fecha	
Nómina	
Proceso	

Código	Nombre	Concepto	Variable	Monto
Cifras control				

Figura 4.3- 10 Reporte de percepciones por aviso.

Solicitud de préstamo sobre ahorro.

Existen dos ahorros del trabajador caja y fondo se diferencian por el manejo de deducibilidad que la empresa puede hacer, como política de este grupo de empresas se pueden pedir prestamos hasta del 80% de lo ahorrado, para hacerlo se debe llenar la solicitud que se muestra en la figura 4.3-11 la cual se entrega al administrador del

ahorro quien de acuerdo a sus controles (en una hoja de excel) revisa si es solvente, de ser así se autoriza y emite un cheque, comunica a recursos humanos llenando el formato de cambios de constantes de deducción que se muestra en la figura 4.3-12.

FOLIO:

COMO SOCIO ACTIVO DE LA CAJA DE AHORROS DEL PERSONAL DE : _____ SOLICITO UN PRÉSTAMO POR LA CANTIDAD Y PLAZO QUE MENCIONO EN EL ESPACIO CORRESPONDIENTE, ASÍ MISMO, ACEPTO QUE LA TASA DE INTERES QUE PAGARE POR ESTE CONCEPTO, SERA DEL C.C.P. MAS 10 PUNTOS. AUTORIZO QUE LOS ABONOS E INTERESES ME SEAN DESCONTADOS POR NOMINA, PARA LO CUAL PROPORCIONO LOS DATOS SIGUIENTES:

DATOS DEL SOLICITANTE

CÓDIGO:	NOMBRE:
DEPTO:	
SDO. MENSUAL:	TIPO DE NÓMINA:
PRÉSTAMO:	INICIO:
PLAZO:	TASA: FIRMA:

DATOS DEL AVAL (ES)

Vg. Bg. DIRECTIVA

DE ACUERDO A LOS DATOS Y REFERENCIAS DEL SOLICITANTE, DETERMINO QUE EL PRESTAMO A QUE SE HACE MENCION SEA : () ACEPTADO () RECHAZADO

AUTORIZA
P R E S I D E N T E

POR ESTE PAGARE DEBO Y ME COMPROMETO A PAGAR A LA ORDEN INCONDICIONAL DE LA CAJA DE AHORROS DEL PERSONAL DE :
EN SU DOMICILIO UBICADO EN :
LA CANTIDAD DE :
BAJO LOS TERMINOS ESTIPULADOS EN ESTA SOLICITUD. EN CASO DE QUE ESTE PAGARE NO SEA LIQUIDADO A A MAS TARDAR EL DÍA 22 DE CAUSARA EL DOBLE DE INTERESES QUE AQUI SE SEÑALA POR EL PERIODO TRANSCURRIDO ENTRE EL DÍA 22 Y LA FECHA DE LIQUIDACIÓN DE LA CAJA DE AHORROS.

LUGAR Y FECHA :

FIRMA DEUDOR :

Figura 4.3- 11 Solicitud de Préstamo Sobre Ahorro

Cambios a Constantes de Deducción.

Para el caso de prestamos sobre ahorro, seguro de auto, fonacot, entre otros, son descuentos que se hacen de manera programada es decir se parte de un saldo a cubrir en cierto número de descuentos conocidos, cuando existe alguna modificación o en un nuevo descuento se llena el formato de la figura 4.3-12 para que se registrado por la capturista y una copia para R.H. ya que confirman que el descuento procede.

Concepto _____ Descripción _____ Fecha _____

Código	Nombre	Constante	Saldo
Cifras control			

Figura 4.3- 12 Cambios a constantes de deducción.

Aviso de Sobre Sueldo.

Para el caso de personal sindicalizado solo pueden realizar las actividades de su categoría, pero por razones de producción o de entrenamiento es necesario que realicen funciones en otra categoría, lo que implica pagar temporalmente un sobre sueldo, para este efecto el supervisor debe enviar el aviso usando el formato de la figura 4.3-13.

Para: Relaciones Industriales.
De: _____

Fecha _____

Por medio de la presenta le solicito se le pague sobresueldo al Sr _____ Código _____ del Departamento _____, Por haber realizado una operación Categoría _____ los días _____ Lo anterior debido a (explicar el motivo) _____ El salario que recibira será el de : Categoría _____ Etapa _____

Firma del Supervisor
Jefe de Área

Firma Gerente de Área.

Figura 4.3- 13 Aviso de Pago de Sobre Sueldo.

Programa de ajustes de sueldo bimestrales.

Para el caso de los empleados se hacen dos ajustes de sueldo, esto aplicando el proceso de evaluación, sin embargo por alguna promoción o por ajuste de nivel se hacen cambios de sueldo pero de manera bimestral, el gerente del área envía el formato de la figura 4.3-14 a recursos humanos para que se autorice y programe el aumento.

Sueldo Empleados.
Fecha: _____

Nombre: _____ Código _____
Fecha de Ingreso _____
Puesto Actual _____ Depto. _____
Puesto Nuevo _____ Depto. _____
Rango Actual _____ Nuevo _____
Sueldo Actual _____ Sueldo Objetivo _____
Penetración Rango actual: _____
Penetración final vs. Nuevo Rango _____
Motivos de acuerdo a políticas _____

Gerente de Rel. Ind. Gerente General. Gerente de Área.

Figura 4.3- 14 Programa de ajustes de sueldo bimestral.

Aviso de Vencimiento de Contrato.

Para los trabajadores de contrato eventual (tiempo y obra determinada) es necesario enviar un aviso cuando va a terminar su contrato para que el gerente del área indique si se renueva o no, recursos humanos llena el formato de la figura 4.3-15 y los envía a las gerencias quienes a su vez devuelven con los cambios correspondientes, es entonces cuando recursos humanos mecanografian los nuevos contratos o genera el finiquito según sea el caso.

Para: _____

De: Recursos Humanos

Me permito comunicarle que el día _____ se vence el contrato por tiempo determinado (Eventual) que tiene el empleado(a)

_____ Código _____, por lo que le pido exprese su voluntad con respecto a esta situación con base en las siguientes opciones.

- Dar contrato por tiempo determinado () 1 mes () 2 meses () 3 meses
- Dar contrato por tiempo indeterminado (planta)
- Dar por terminada la relación de trabajo.
- Enviar remplazo.

Firma de Gerente de Área

Figura 4.3- 15 Aviso de Vencimiento de Contrato.

Control de Personal (Cardex).

Para efectos de tener un registro de los días de ausentismo o vacaciones de los trabajadores se lleva un cardex como se muestra en la figura 4.3-16 donde se vacía la información de los formatos de vacaciones, faltas, permisos en incapacidades. Este registro es llevado por el tomador de tiempo de recursos humanos.



CONTROL DE PERSONAL

MES	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31			
ENE																																		
FEB																																		
MAR																																		
ABR																																		
MAY																																		
JUN																																		
JUL																																		
AGO																																		
SEP																																		
OCT																																		
NOV																																		
DIC																																		
NUM.	N-02-14															Total																		
	NOMBRE DEL TRABAJADOR															REG. FED. CAUS.																		

S- PERMISO SIN SUELDO
 E- EXPIRADO
 F- FOLGAS
 P- PERMISO CON SUELDO
 1- INJUSTIFICADA
 V- VACACIONES
 D- DEFERIDO
 R- RETARDO

Figura 4.3- 16 Control de Personal (Cardex).

Alta de trabajadores.

Cuando se va a dar un alta el auxiliar de nómina llena el formato de la figura 4.3-17 con los datos del trabajador para su captura en el sistema de nómina HP-3000.

Proceso		
Código		
R.F.C.		
IMSS		
Cartilla		
Licencia		
Nombre		
Ape. Paterno		
Ape. Materno		
Sexo		
Escolaridad		
Fecha de nacimiento		
Dirección		
Telefono		
Fecha de ingreso		
Fecha de Venc. Contrato.		
Tipo de Contrato		
Departamento		
Puesto		
Sueldo Diario	Sueldo Integrado	Sueldo Mensual

Figura 4.3- 17 Alta de Trabajadores.

Calculo de Finiquitos.

Cuando se da una baja se debe calcular el finiquito que es el pago de las prestaciones a las que tiene derecho el trabajador, estas dependen del tipo de contrato, motivo de renuncia y antigüedad, además se le debe liquidar lo ahorrado y descontar sus adeudos y cuotas al IMSS. Para efectos fiscales el impuesto tiene un tratamiento especial. Para estos cálculos se utiliza el formato de la figura 4.3-18 que llena R.H. y lo envía al administrador de caja para que anote lo ahorrado y lo adeudado, después es enviado a contabilidad para que calcule los impuestos y las cuotas del IMSS, usando los formatos de las figuras 4.3-19 y 4.3-20 respectivamente se regresa a R.H. para su autorización y se devuelve a contabilidad que codifique y capture su póliza contable usando el formato de la figura 4.3-21.

Fecha _____

El Sr. Srita:	Código:
Departamento:	Planta () Eventual ()
Renuncia Voluntaria () Terminación de Contrato () Rescisión de Contrato ()	
Ultimo día de trabajo:	Semana () Quincena ()
Fecha de Ingreso:	Antigüedad:

Sueldo Diario \$

Semana de Fondo \$ _____

Salario del _____ al _____ de _____ 19____

Horas Extras \$ _____

Premio de Puntualidad _____ días \$ _____

Vacaciones \$ _____

Prima Vacacional _____ días

Aguinaldo Proporcional _____ días \$ _____

_____ \$ _____

Sub Total Percepciones Normales \$ _____

OTRAS PERCEPCIONES:

_____ \$ _____

_____ \$ _____

_____ \$ _____

Sub Total Otras Percepciones \$ _____

TOTAL DE PERCEPCIONES \$ _____

DEDUCCIONES Y AHORROS

ISPT Percepción Normal \$ _____

ISPT Indemnización \$ _____

Adeudo a la Empresa \$ _____

Adeudo Caja de Ahorro \$ _____

Adeudo Fondo de Ahorro \$ _____

Aportación Caja de Ahorro \$ _____

Interes Caja de Ahorro \$ _____

10% Fondo Caja de Ahorro \$ _____

Fondo de Ahorro \$ _____

_____ % Fondo de Ahorro \$ _____

TOTAL \$ _____

Neto a Pagar \$ _____

Autorizo _____ Elaboró _____

Figura 4.3- 18 Calculo de Finiquitos.

Nombre				
Código				
Calculo de impuesto para finiquitos				
		Montos		Montos
	PERCEPCIÓN		INDEMNIZACION	
MENOS	P. VAC. EXE. (15)		INDEMN. EXENTA	
MENOS	AGUI. EXE. (30)		INDEMN. GRAV.	
MENOS	DESPENSA			
IGUAL	SUBTOTAL			
MAS	PERCEP.ACUM.			
IGUAL	TOTAL		SUELDO. MENS	
MENOS	LIMITE INFERIOR		LIMITE INFERIOR	
IGUAL	EXCED. S/LIM.INF.		EXCED. S/LIM.INF	
	IMPTO.MARGINAL		IMPTO MARGINAL	
MAS	CUOTA FIJA		CUOTA FIJA	
IGUAL	I.S.P.T.		ISPT	
	SUBS.IMP.MARG.		SUBS.IMP. MARG.	
MAS	SUB. CUOTA FIJA		SUB.CUOTA FIJA	
IGUAL	TOTAL SUBSIDIO		TOTAL SUBSIDIO	
MENOS	REDUCC. 70.06 %		REDUC.70.06 %	
IGUAL	SUBSIDIO NETO		SUBSIDIO NETO	
	ISPT		ISPT	
MENOS	SUBSIDIO		SUBSIDIO	
IGUAL	ISPT S/SUBS.		ISPT S/SUBS	
MENOS	CRED. SALARIO.		CRED. SALARIO	
IGUAL	ISPT S/CRED.		ISPT S/CRED.	
MENOS	ISPT RETENIDO			
IGUAL	ISPT X PAGAR		ISPT/SAL.MENS	
			FACT.X INDEMN.	
			ISPT X INDEMN.	

Figura 4.3- 19 Calculo de Impuesto para finiquitos.

SDI 25	437.93	861.25	SDI 25	348.24				
SDI 16	437.93	551.20	SDI 16	348.24				
DIAS PER.	30		DIAS PER.	31				
FALTAS			FALTAS					
INCAP.			INCAP.					
SMG	34.45		SMG	34.45				
RIES TRAB	0.031855		RIES TRAB	0.041855				
					SUA			
					TRABAJADOR	PATRON		
ENFERMEDAD Y MATERNIDAD			ENFERMEDAD Y MATERNIDAD					
CUOTA FIJA			CUOTA FIJA		155.39	\$155.39		
PREST ESP	13320.37		PREST ESP	10795.44	10795.44			
	3143.56			3203.85	3203.85			
	167.25			139.69	418.30	\$557.98		
PREST.DIN.	33.30		PREST.DIN.	26.99	75.57	\$102.56		
CUOTA ADIC.	49.95		CUOTA ADIC.	40.48	113.35	\$153.84		
IVALIDEZ Y VIDA			IVALIDEZ Y VIDA					
	83.25			67.47	188.92	\$256.39		
VEJEZ Y CESANTIA			VEJEZ Y CESANTIA					
	149.85			121.45	340.06			
RIESGO TRABAJO			RIESGO TRABAJO					
					451.84	\$451.84		
GUARDERIA			GUARDERIA					
					107.95	\$107.95		
SAR			SAR					
					215.91			
INFONAVIT			INFONAVIT					
					539.77			
TOTAL		503.61	TOTAL		396.076956	2607.06	\$1,785.95	

Figura 4.3- 20 Calculo de aportaciones del IMSS, SAR e INFONAVIT.

Poliza No. _____

Fecha: _____

Mayor	Cuenta	SubCuenta	Concepto	Debe	Haber
Sumas iguales					

Figura 4.3- 21 Póliza de Diario.

Examen Médico de Admisión.

Quando se va a contratar una persona es indispensable que se realice un examen medico el cual es realizado por el servicio médico y se registra en el formato de la figura 4.3-22, el cual queda en resguardo del médico, para comunicar el resultado del examen a recursos humanos se llena el formato de la figura 4.3-23 para que tomen la decisión de contratarlo o no.

Programa de Exámenes Médicos.

Una vez que se ha integrado el personal a la empresa es necesario practicarle cada año un examen para lo cual se lleva un programa anual, el cual se registra en el formato de la figura 4.3-23 y se emite a cada departamento para que programe su producción con el personal disponible.

Dependiendo de la actividad el examen puede ser programado dos veces al año o de acuerdo a su estado de salud se puede hacer un seguimiento mas estrecho.

Fecha _____

Registro Num.
Nombre de la negociación
Clase de Industria
Ubicación
Nombre del trabajador
Edad:
Peso: Estatura:

Datos generales

Aparato Respiratorio

Aparato Circulatorio

Aparato Genito-Urinario

Sistema Nervioso

Psiquis

Examen de los Oídos

Aparato Locomotor

Datos del Médico

Figura 4.3- 22 Examen Médico de Admisión.

Departamento de Relaciones Industriales

Presente:

El Sr., (Srita) _____ ha

Presentado el examen médico de admisión el día _____

De _____ de 19__

Con los siguientes resultados:

Apto: _____

No Apto: _____

Observaciones _____

Recibio:

ATENTAMENTE

Relaciones Industriales

Jefe de servicio médico

Figura 4.3- 23 Resultado de Examen Médico.

Depto	No. Personas	Ene	Feb	Mar	Abr	May	Jun	Jul	Ago	Sep	Oct	Nov	Dic
Total													

Figura 4.3- 24 Programa de Exámenes Médicos.

4.4 Planteamiento del Problema.

Como resultado del análisis de los requerimientos y de información es necesario encaminar los esfuerzos a dar una solución integral a los problemas de administración de información en el área de recursos humanos, para lo cual se plantea desarrollar un sistema de computo.

4.4.1 Módulos del sistema.

Recursos Humanos
<ul style="list-style-type: none">• Reclutamiento y Selección.• Administración de datos personales del personal.• Administración de datos familiares del personal.• Administración de renovación de contratos.• Integración de fotografía del personal.

Control de Asistencia (Vía Relojes y Captura en Planta).
<ul style="list-style-type: none">• Administración de Tiempo Extra• Faltas y Permisos• Control de Turnos• Turnos Desfasados• Pagos de sobre sueldo• Administración de Vacaciones.• Control de Incapacidades.

Administración de Movimientos Programados.
Administración de Movimientos Adicionales.
Captura de incidencias.

Administración de Ahorro.

- Acumulados de Puntos y Aportaciones
- Cambios de Aportaciones
- Administración de Prestamos.
- Liquidación de Ahorro.
- Control de Inversión.

Calculo de varias nóminas:

- Ordinarias
- Vales de Despensa
- Prima Vacacional
- P.T.U.
- Gratificación Anual.
- Aguinaldo
- Premio de Puntualidad.
- Otras (No Establecer Limite)

Salarios integrados.

- Generación de Integrados
- Impresión de avisos al IMSS
- Transferencia electrónica de salarios integrados
- Movimientos al SUA
- Conciliación de Aportaciones.

Generación de archivo para deposito electrónico de nóminas (Banamex)

- Administración de Número de Cuentas Bancarias

Administración de Personal.

- Aumentos programados de sueldo.
- Cambios de Puesto y Departamentos
- Aumentos generales de sueldo por Evaluación.
- Aumentos generales de sueldo por negociación contractual.

Calculo de Finiquitos

- Calculo de Liquidaciones
- Emisión de Recibo
- Emisión de carta de separación
- Emisión automática de póliza.
- Emisión de Constancia de Baja.
- Emisión de Aviso de Baja al IMSS.

Pólizas contables

- Configuración de cuentas
- Emisión de póliza por proceso de nómina
- Archivo de transferencia electronica a MFG-Pro
- Cancelación de recibos.

Capacitación

- Administración de Plan anual de capacitación
- Administración de cursos
- Administración de Evaluaciones al desempeño
- Emisión de Constancias de Capacitación.

Servicio Médico

- Exámenes médicos a aspirantes
- Emisión electrónica de resultados de exámenes
- Programación de Exámenes Médicos
- Control de Incapacidades.

Contabilidad

- Revisión y Actualización de acumulados
- Emisión de constancias de percepciones y deducciones
- Proceso de ajuste anual de impuestos.
- Declaraciones anuales para la SHCP.
- Calculo de presupuesto y póliza del mismo.

Consultas Gerenciales

- Administración de evaluaciones al desempeño.
- Administración de vacaciones.
- Consultas de datos generales de su personal.
- Consultas de avance de capacitación.
- Requisiciones de personal.

Consultas Corporativas

- Reporte mensual de personal.
- Reporte de prestaciones.
- Reporte de movimientos acumulados.
- Consultas personalizadas del personal

Auditorías

- Movimientos acumulados
- Bases de integración de IMSS, SAR, Infornavit.
- Bases gravadas para impuestos

4.4.2 Características del Sistema.

- Evitar la múltiple captura de información.
- Reducir al máximo el manejo de papel (conjunto reducido de reportes)
- Poder transferir información (por el usuario final) del sistema a herramientas como Excel o Word.
- Reducir el número de horas hombre en el manejo de la información.

- Ser parametrizable por el usuario
- Confiabilidad al 100 % en los cálculos.

- Establecer niveles de seguridad para el acceso.
- Tener un nivel adicional para las nominas confidenciales.

- Arquitectura Cliente/Servidor
- Plataforma del Servidor Windows NT.
- Plataforma de los clientes Windows 95 o posterior.
- Base de datos relacional.
- Interfaz gráfica.
- Satisfacer las necesidades del año 2000
- Contar con herramientas de administración del sistema

4.4.3 Problemática por Areas.

Area	Problemática
Recursos Humanos	Excesivo trabajo de decodificación de incidencias de nómina: <ul style="list-style-type: none"> • Tiempo Extra • Faltas • Permisos • Cambios temporales de Categoría • Incapacidades
	Poca confiabilidad en los salarios integrados.
	Poca confiabilidad en los cálculos del sistema
	Problemas de generar la información que se le solicita.
	Excesivo trabajo administrativo de contratos.
	Calculo manual de finiquitos
	Dependencia del área de sistemas
	Problemas para obtener información para auditorias
Contabilidad	Ajuste manual a las pólizas contables
	Calculo manual de impuesto para finiquitos
	Llenado manual de constancias de retención de impuestos.
	Administración manual de Ahorros: <ul style="list-style-type: none"> • Prestamos • Intereses • Aportaciones • Liquidaciones
	Poca confiabilidad en los acumulados de conceptos.
	Dependencia del área de sistemas
	Problemas para obtener información para auditorias
Servicio Médico	Control manual de incapacidades y exámenes médicos.
Capacitación	Llenado manual de constancias de capacitación
	Control manual de cursos.
	Control manual de evaluaciones

Gerencias	Control manual de evaluaciones
	Control manual de vacaciones
	Dificultad para obtener información consolidada.
Sistemas	Excesiva carga de trabajo por servicio a los usuarios
	La HP-3000 ya no cuenta con servicio del proveedor
	Problemas con el año 2000

4.5 Descomposición Funcional.

En la figura 4.5-1 se muestra un esquema de las necesidades de información que el departamento de recursos humanos debe satisfacer.

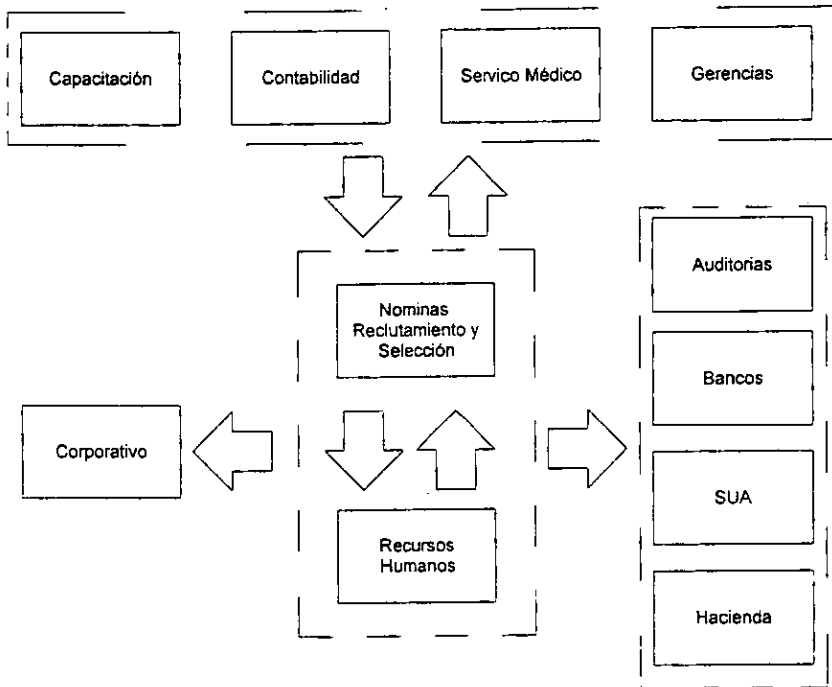


Figura 4.5- 1 Esquema funcional.

El componente más importante es el de Nomina ya que este es el origen de toda la información que se comparte y complementa por los demás elementos en la figura 4.5-2 se muestra el esquema de la nómina y como esta constituida.

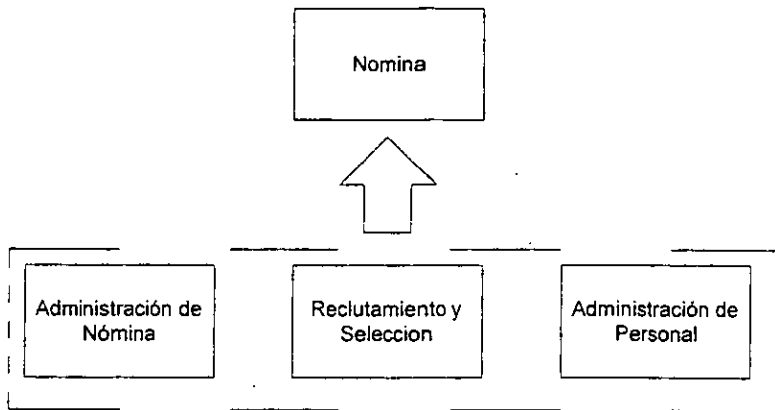


Figura 4.5- 2 Proceso de Nómina

Para poder hacer una nómina se requiere de los módulos que se muestran en la figura 4.5-2, el de Administración de Nómina esta compuesto por una serie de sub-módulos que permiten recopilar la información necesaria para este proceso, en la figura 4.5-3 se muestran estos esquemas.

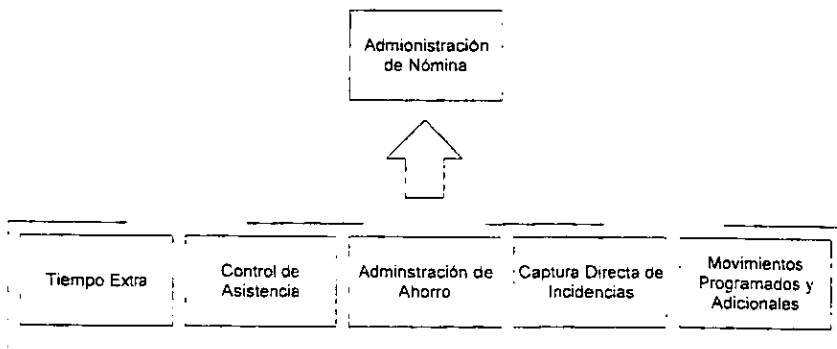


Figura 4.5- 3 Administración de Nómina.

Los módulos de Reclutamiento y Selección, así como en el de Administración de personal proveerán la información acerca de los empleados, por ejemplo departamento, puesto, sueldo, salario integrado. Es en este módulo donde se administran los datos generales del empleado, en la figura 4.5-4 se muestra este esquema.

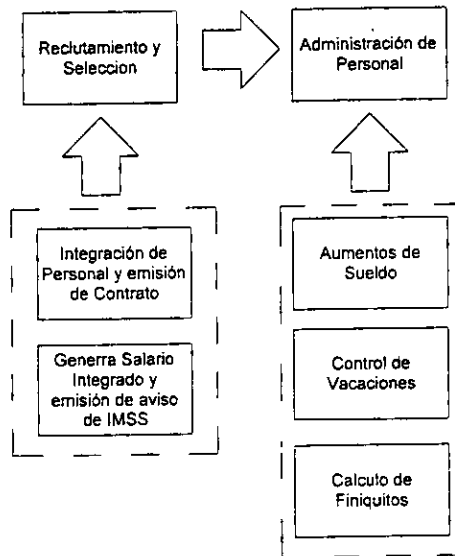


Figura 4.5- 4 Reclutamiento – Selección y Administración de Personal.

El módulo de contabilidad tiene como objetivo explotar la información que genera el módulo de nómina, sobre todo para efectos de pólizas contables, acumulados de conceptos, declaración de impuestos además de la estimación presupuestal, en la figura 4.5-5 se muestra este módulo.

La información de los módulos de hacienda y auditoría se generan de manera independiente ya que son entidades externas a la empresa es por eso que en la figura 4.5-1 se clasifican en un bloque aparte lo mismo que el módulo de SUA y Bancos.

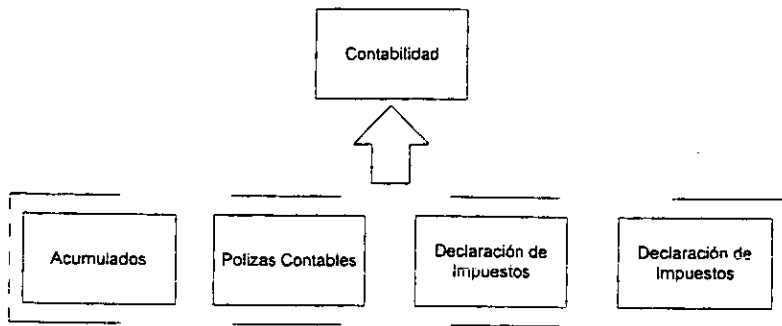


Figura 4.5- 5 Módulo de Contabilidad.

El modulo de Capacitación administra principalmente lo referente a evaluaciones al desempeño y control de cursos, además de relacionarlos con los de Administración de Personal, en la figura 4.5-6 se muestra este esquema.

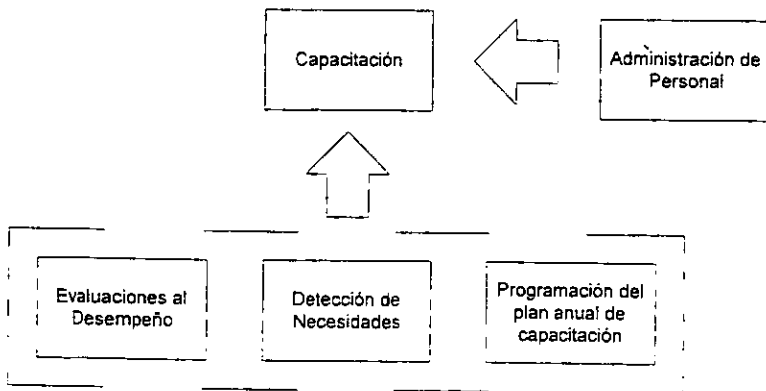


Figura 4.5- 6 Modulo de Capacitación.

El módulo de servicio médico se encarga de administrar los exámenes médicos para que sean analizados y ponderados por recursos humanos, de igual manera controla las fechas de los siguientes exámenes, en la figura 4.5-7 se muestra este esquema.

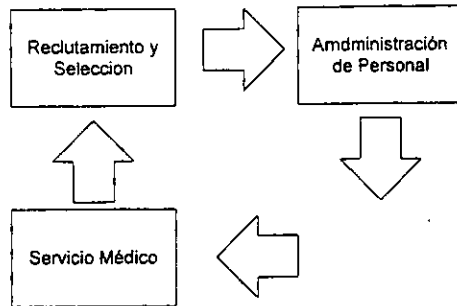


Figura 4.5- 7 Servicio Médico.

El módulo corporativo tiene la finalidad de consolidar la información general de cada empresa. Este modulo deberá proporcionar información de personal, prestaciones incidencias de ausentismo, avances de capacitación y riesgos de trabajo. En la figura 4.5- 1 se sitúa en una región independiente porque es un consolidado de información tal como se muestra en la figura 4.5- 8.

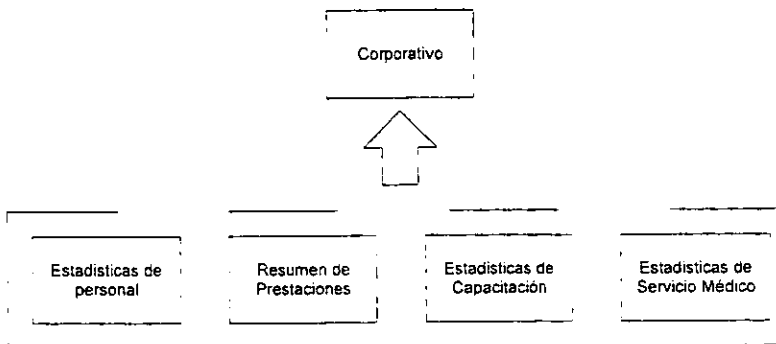


Figura 4.5- 8 Modulo Corporativo.

4.6 Opciones de Solución.

Para el desarrollo del sistema se tiene que seleccionar una herramienta para el back-end (manejador de bases de datos) y otra para el front-end (Interfaz y aplicación). Con base en las necesidades y la política informática del cliente el sistema operativo de red sobre el cual deberá correr la aplicación será Windows NT 4.0

El manejador de bases de datos tendrá que soportar el modelo relacional. Para la aplicación se utilizará una herramienta que permita implementar un ambiente gráfico además de permitir combinar información con la suite de herramientas de microsoft office.

Las opciones de implementación del sistema son las siguientes:

Back-End	Front-End
SQL-Server	Access 97
SQL-Server	Visual Basic
Access 97	Visual Basic
Access 97	Access 97

En el cuadro 4.6-1 se muestra una comparación entre las principales características de Visual Basic 5.0 y Access 97 y en el cuadro 4.6-2 se muestra la relación que guardan con respecto a la conectividad a las bases de datos.

Característica	Visual Basic 5.0	Access 97
Fabricante	Microsoft	Microsoft
Herramienta	De propósito general	Aplicaciones de Bases de Datos
Ambiente gráfico de desarrollo	Si	Si

Lenguaje de programación	Visual Basic para Aplicaciones (VBA)	Visual Basic para Aplicaciones (VBA)
Soporte para el año 2000	Sí	Sí
Distribución de la aplicación	Genera Ejecutables	Genera archivos objeto (.mde) que requieren un Run-Time
Combina rutinas externas	Sí	Sí
Herramienta de Depuración	Sí	Sí
Asistentes para desarrollo	Sí	Sí (se pueden adicionar de terceros)
Soporte de Bases de Datos	Sí	Sí (Interconstruido)
Formas o pantallas de captura.	Sí (Asistente muy limitado)	Sí (Asistente muy robusto)
Creación de módulos	Sí (Asistente)	Sí (Sin Asistente pero con ayuda en línea)
Importación y Exportación de archivos.	Sí (Sin Asistente)	Sí (Asistente muy robusto)
Generación de Reportes	Sí (Usando Crystal Report)	Sí (Asistente muy Robusto)
Generación de Consultas	Sí (Se escribe toda la sentencia SQL)	Sí (Asistente gráfico)
Administración de Bases de Datos	Sí (Por código)	Sí (por código e interfaz gráfica)
Seguridad	Sí utiliza archivos .mdw (Por código)	Sí utiliza archivos .mdw (por código y asistente)
Desempeño de ejecución de código compilado.	Excelente	Buena
Utilerías comerciales	Sí (Amplio mercado)	Sí (muy restringido)

Cuadro 4.6- 1 Características de VB5.0 y Access 97.

Parámetros	Visual Basic 5.0	Access 97
Soporte de Bases de datos	<p>Visual Basic SQL Server Library (VBSQL) soporte de 16 y 32 bits</p> <p>ODBC API soporte de 16 y 32 bits.</p> <p>Data Access Objets (DAO) a través de Microsoft Jet Engine de 32 bits.</p> <p>RemoteData Control (RDC) de 32 bits.</p> <p>Remote Data Objects (RDO) de 32 bits.</p> <p>ODBCDirec (DAO y RDO) de 32 bits.</p> <p>Proveedores Externos.</p>	<p>DAO vía Microsoft Jet Engine 3.5+</p> <p>ODBCDirect</p>
Bases de Datos que se pueden usar.	Cualquiera que provea un controlador ODBC.	
Eficiencia en la manipulación de datos	Mayor ya que tiene mayor número de controladores y solo carga el necesario.	Para bases de datos externos se recomienda usar ODBCDirec ya que así no carga el Jet Engine.
Codificación	Requiere mayor programación.	Programación con el Jet Engine 3.5+ casi

		transparente. Con el ODBCDirect mayor programación.
Conexión a SQL-Server	Sí (Se explota mejor)	Sí (Se Restringen las capacidades de SQL-Server)

Cuadro 4.6- 2 Conectividad a las Bases de Datos.

En el cuadro 4.6-3 se muestra la comparación entre los Manejadores de Bases de Datos SQL-Server 6.5 y Access 97.

Característica	SQL-Server 6.5	Access 97
Fabricante	Microsoft	Microsoft
Plataforma	Windows NT o Windows Workstation	Windows 95 o posterior.
Seguridad	Estándar: SQL valida el login a la base de datos y guarda esta información de manera independiente a la del sistema operativo. Integrada: SQL utiliza el sistema de autenticación que usa Windows NT: Combinada: es una combinación de las dos anteriores.	A través del administrador de grupos de trabajo.
Arquitectura	Cliente/Servidor	Servidor de Archivos
Para Aplicaciones	Medianas y Grandes	Medianas

Soporte	Soporte especializado	Casi libre de soporte (Solo compactar y reparar)
Stored Procedures	Sí	No
Triggers	Sí	No
Replicación de Datos	Sí	Sí (Muy Limitado)
Ambiente multiUsuario	Sí	Sí (El performance disminuye considerablemente)
Sistema de respaldo y recuperación.	Sí (Cuenta con herramientas especializadas)	No
Administración de Base de datos.	Sí	Sí (A través de una interfaz gráfica)
Combina bases de datos de diferentes proveedores	Sí	Sí (A través de Adjuntos)
Las aplicaciones de Office pueden obtener información de la base de datos.	Sí	Sí

Cuadro 4.6- 3 Manejadores de Bases de Datos.

Con base en los datos de los cuadros anteriores las cuatro opciones se pueden calificar como se muestra en el cuadro 4.6-4

Opción	Rendimiento	Opciones	Complejidad	Costo
SQL-Server-Visual Basic	Se obtiene el mejor rendimiento (10 puntos)	Cumple con todas las necesidades	Presenta mayores complejidades en el desarrollo.	El más caro por Equipo, Licencias, Soporte y tiempo de desarrollo.

SQL-Server- Access 97	Se obtiene el segundo mejor rendimiento (7 Puntos)	Cumple con todas las necesidades	Por parte del back-end persisten las mismas complejidades pero el front-end presenta una mayor facilidad de desarrollo.	El costo sigue elevado por los requerimientos del SQL-Server, las licencias de Access 97 son más económicas
Access 97 – Visual Basic.	Tercer mejor rendimiento (5 Puntos)	Cumple con todas las necesidades	La administración del Back-end se disminuye casi al mínimo, pero los asistentes de Visual Basic no son tan poderosos.	El costo se disminuye a casi el mínimo.
Access 97- Access 97	Cuarto mejor rendimiento (4 Puntos)	Cumple con todas las necesidades	Esta opción lleva al mínimo la administración del back- end y al tiempo de desarrollo por las ventajas de los asistentes y de las herramientas de programación en Access.	El costo es el mínimo.

Cuadro 4.6- 4 Calificación de la comparación

4.7 Elección de solución óptima.

Para seleccionar la opción de desarrollo se tomo en cuenta la premura que tenían las empresas de actualizar sus sistemas del área de recursos humanos, esto debido que no estaban preparados para funcionar en el año 2000 y sus clientes internacionales condicionaron sus compras a que concluyeran la reconversión de sus sistemas a más tardar al 3 trimestre de 1999. También fue considerado el presupuesto disponible para dicha reconversión ya que no era el único que deberían hacer.

La opción de Access97 – Access97 fue la que se selecciono ya que:

- Las empresas cuentan con una licencia corporativa la cual les da derecho a un número ilimitado de licencias de Office y Access 97 así como sus futuras actualizaciones.
- El costo del proyecto (en tiempo) se redujo por la facilidad de la herramienta.
- Se cumple con la premisa interna de eliminar al máximo la dependencia del departamento de sistemas para la generación de información.
- Queda la posibilidad de migrar en un futuro a SQL-Server.
- Esta herramienta de desarrollo combina las dos partes fundamentales que se buscan un manejador de bases de datos robusto y la posibilidad de crear aplicaciones gráficas.

Para eficientar el rendimiento de la aplicación se decidió dividir la aplicación en módulos de tal manera que los programas objetos (.mde) se instalen en los clientes de cada usuario y los datos en un servidor determinado, esto reducirá el trafico en la red y elevara el desempeño de la aplicación.

CAPITULO 5 DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA.

5.1 Desarrollo de los módulos del sistema.

5.1.1 Aplicación de la metodología.

La metodología que se aplicará es la propuesta por Yourdon/DeMarco, la cual es discutida ampliamente en el tema 2.3.3, a continuación se muestra una síntesis de sus dos elementos fundamentales.

Diagramas de Flujo.

Un diagrama de flujo de datos (DFD) es una representación gráfica en forma de red de un sistema, el cual puede ser automático, manual o mixto. El diagrama de flujo de datos retrata al sistema en términos de sus componentes con todas las interfaces entre los componentes indicados.

Algunas de las características más importantes de los diagramas de flujo de datos son:

- Es gráfico
- Particionado
- Multidimensional
- Énfasis en el flujo de datos
- No enfatiza el flujo del control

Esta herramienta permite visualizar al sistema desde el punto de vista de los datos y no de quien trabaja con ellos.

- Los elementos de un diagrama de flujo de datos son:
- Flujo de datos: representados por vectores etiquetados
- Procesos: representados por círculos o burbujas.
- Archivos: representados por cajas.
- Fuentes o Depósitos de datos: representadas por cajas con barras a los costados

La figura 5.1-1 es una muestra de un diagrama de flujo de datos que incluye los cuatro elementos:

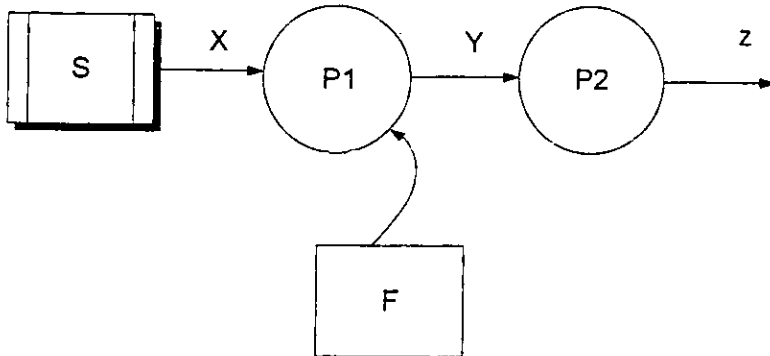


Figura 5.1- 1Elementos de un DFD

Diccionario de Datos.

Un diccionario de datos es un almacén de datos acerca de los datos.

El Diccionario de Datos (DD) es una parte integral de la especificación estructurada; sin esta los diagramas de flujo de datos son solo diagramas que dan una idea de que es lo que pasa en el sistema. Es solo cuando cada elemento del DFD ha sido rigurosamente definido que el todo puede constituir una especificación. El conjunto de rigurosas definiciones de todos los elementos del DFD es un diccionario de datos.

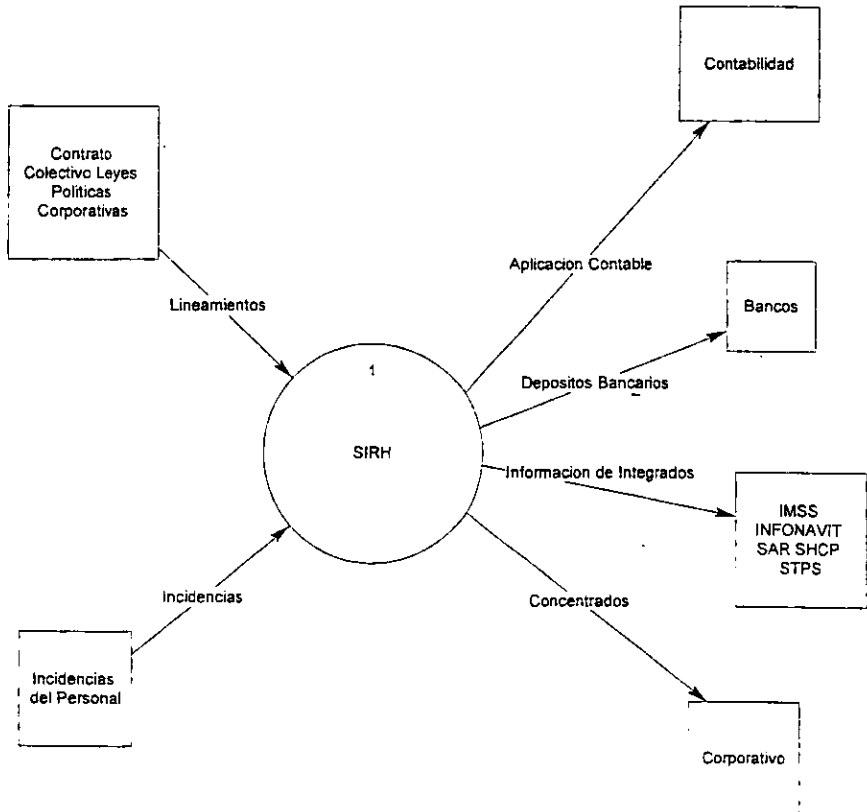
El papel más importante de un diccionario de datos es proporcionar en un solo lugar las referencias de los términos que no se entienden en el DFD.

El DFD y el DD deben ser considerados como uno solo. Sin el DD los diagramas pierden rigor, sin los diagramas el DD no tiene ningún uso, se puede decir que la correlación entre ambos es:

Hay una entrada al DD por cada flujo de datos que aparecen en el DFD, hay una entrada en el DD por cada archivo que se hace referencia en el DFD.

5.1.2 Diagrama de Contexto.

El diagrama de contexto delimita el alcance del proyecto, en este sentido en la figura 5.1.2-1 se muestra el correspondiente al SIRH. En este diagrama se puede ver que el sistema básicamente recibe insumos de que se denomina como incidencias (Sueldo, faltas, incapacidades, préstamos etc.) y los lineamientos (leyes, políticas etc.) para ser transformados en flujos direccionados a Bancos, Contabilidad, Gobierno y Corporativo.



5.1.3 Diagramas de Flujo de Datos.

Los diagramas de flujo que se presentan en esta sección son los siguientes:

5.1.3.1 Generación del Salario Integrado.

Este modulo tiene la finalidad de generar lo que conoce como salario diario integrado (SDI) el cual se utiliza para calcular las aportaciones al IMSS, Infonavit, SAR y en su caso el pago de incapacidades.

Los ingresos del empleado están compuestos de su sueldo diario más prestaciones como vales de despensa, fondo de ahorro, prima de vacaciones, aguinaldo, gratificaciones, horas extras y cualquier otra percepción que se tenga derecho. El SDI integrado es un sueldo promedio con todas estas prestaciones, la ley indica que las prestaciones se deben clasificar en tres tipos:

- Sueldo Diario: Sueldo que percibe de manera diaria el empleado.
- Factor fijo: Las prestaciones a las cuales tiene derecho el trabajador y que se conocen previamente como son aguinaldos y primas de vacaciones, estas prestaciones varían de acuerdo a la antigüedad por lo es necesario conocer la fecha de ingreso de cada trabajador.
- Parte Variable : Las prestaciones o remuneraciones que se presentan de manera eventual o que dependen de factores desconocidos como son vales, fondo de ahorro, horas extras etc. Los montos variables serán los de mes inmediato anterior a la fecha de integración y se dividirán entre los días del mes para obtener un promedio.

De manera general el SDI se calcula de la siguiente manera:

$$\text{SDI} = (\text{Sueldo Diario}) * (\text{Facto Fijo}) / [(\text{Parte Variable}) / (\text{Días del Mes})]$$

La Ley también marca un tope es decir que nadie puede tener un integrado que sobrepase a 25 veces el salario mínimo general del D.F. y este tope puede variar de acuerdo a las disposiciones legales.

Se debe calcular el SDI cuando se presente al menos uno de los siguientes casos:

- Es 1 de mes.
- Aumento de Sueldo Diario
- Cambio en los topes
- Cambian los montos de prestaciones de Aguinaldo y Prima Vacacional
- La antigüedad hace que cambie en la tabla de presentaciones

En las figuras 5.1.3-1 y 5.1.3-2 se muestra el diagrama de flujo de este módulo.

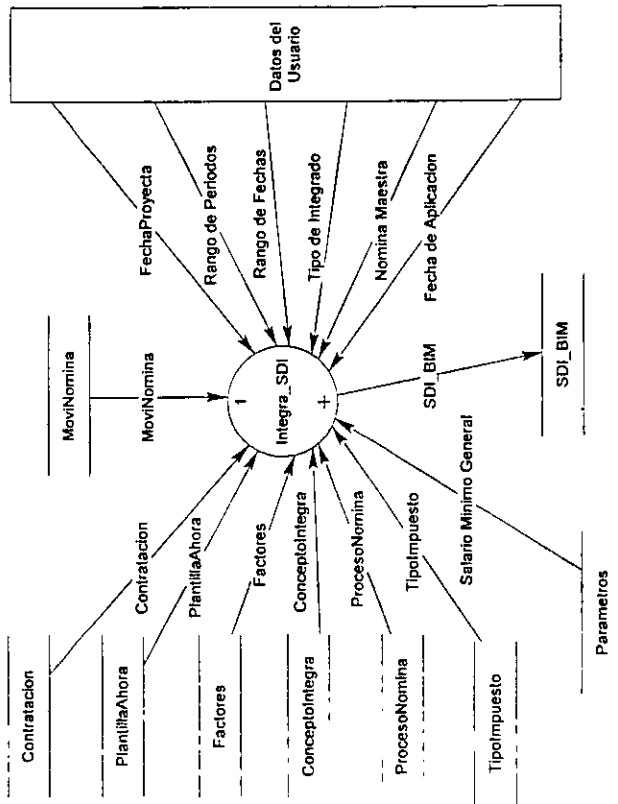


Figura 5.1.3-1 DFD Salario Integrado Nivel 1

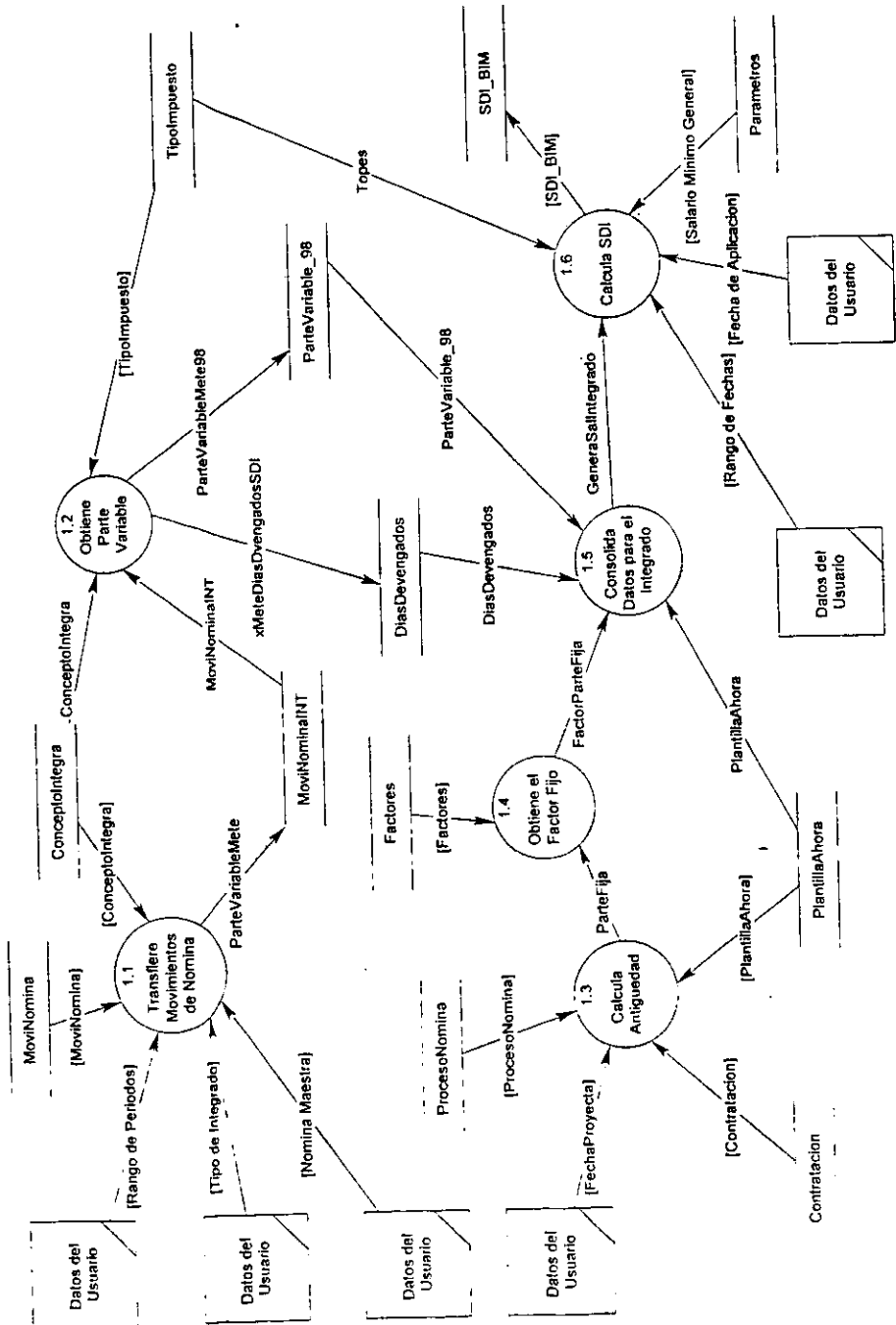


Figura 5.13-2 DFD Salario Integrado Nivel 2

5.1.3.2 Prestamos de Ahorro.

Los empleados que estén inscritos a la caja y fondo de ahorro pueden pedir prestamos sobre su ahorro, estos prestamos son descontados por nómina en pagos pero también se le descuentan los intereses generados (sobre saldos insolutos).

Existen dos tipos de prestamos:

- Descuento Normal
Son los que periodo a periodo de nómina se descuenta capital e interés con tasa ajustable al mes, es decir que la tasa cambia de mes a mes.
- Un Solo pago
Solo se descuentan intereses ajustables al mes y se liquida el saldo al final del ejercicio.

También existe la posibilidad de hacer pagos por caja, es decir que se abone a capital fuera de los periodos de nómina, en este caso el sistema calcula los intereses sobre dos saldos o tantos como abonos hubiera hecho.

Si por alguna razón no pudieran descontarse capital e interés el sistema reajusta el monto a descontar de capital y recalcula los intereses en el siguiente periodo.

En las figuras 5.1.3-3 y 5.1.3-4 se muestra el diagrama de flujo de este módulo.

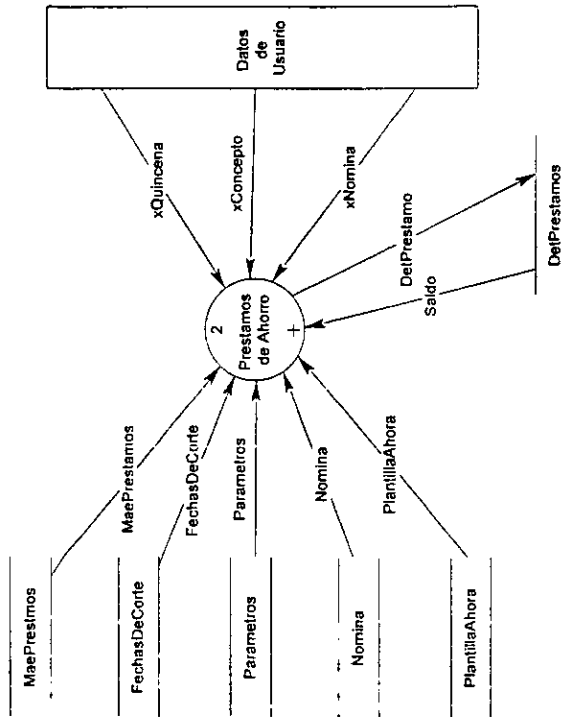


Figura 5.1.3-3 DFD Prestamos de Ahorro Nivel 1

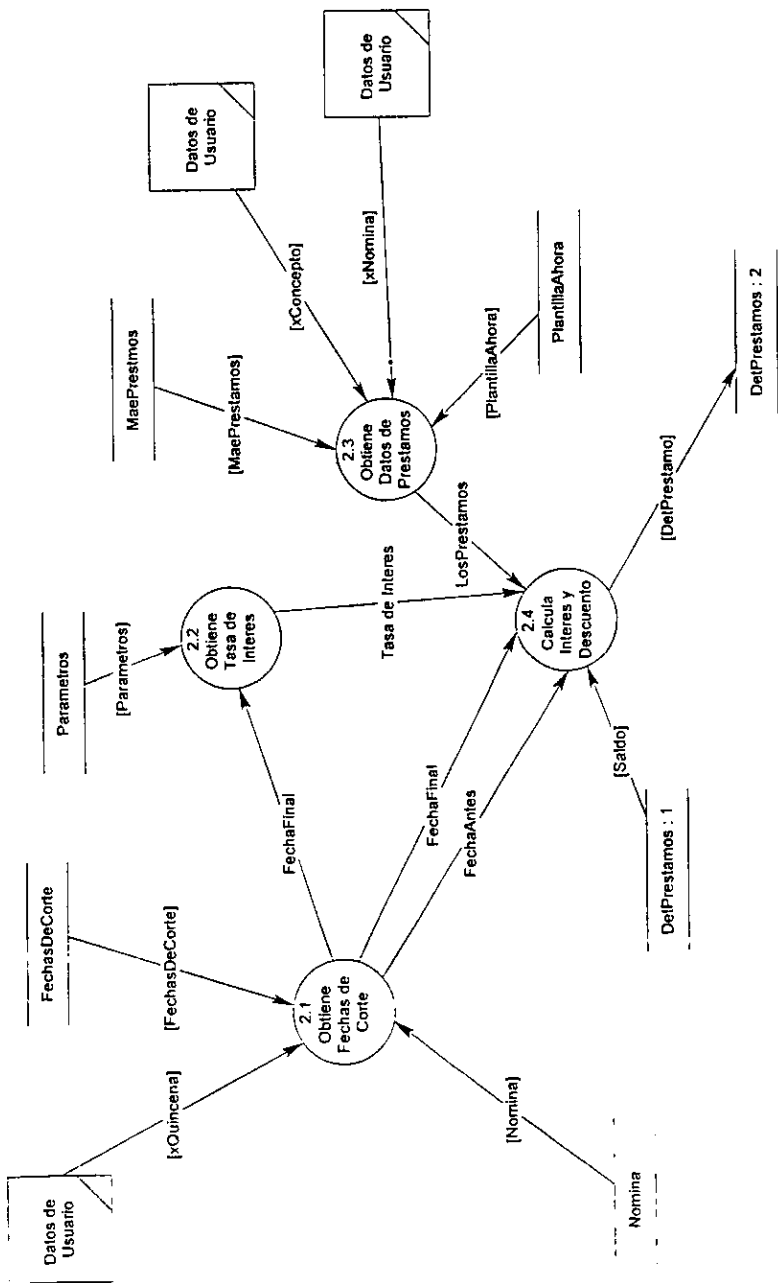


Figura 5.1.3-4 DFD Prestamos de Ahorro Nivel 2

5.1.3.3 Tiempo Extra.

El tiempo extra es un proceso que esta definido en la ley y por lo tanto debe apegarse de manera estricta.

El tiempo extra se registra por el usuario con la fecha, hora de inicio y hora final, este registro puede ser por medio de relojes checadores o por medio del modulo de captura en planta, a este registro es necesario procesarlo para:

- Separa el tiempo extra por turno ya que se paga de manera distinta.
- Separar lo que es jornada extraordinaria del tiempo extra.
- Separar el tiempo doble del triple de cada semana.
- Clasificar el tiempo extra doble o triple cual integra y cual no.
- Calcular el costo del tiempo extra
- Transferir datos para calcular la parte gravable y la exenta.

En las figuras 5.1.3-5 y 5.1.3-6 se muestra el diagrama de flujo de este módulo.

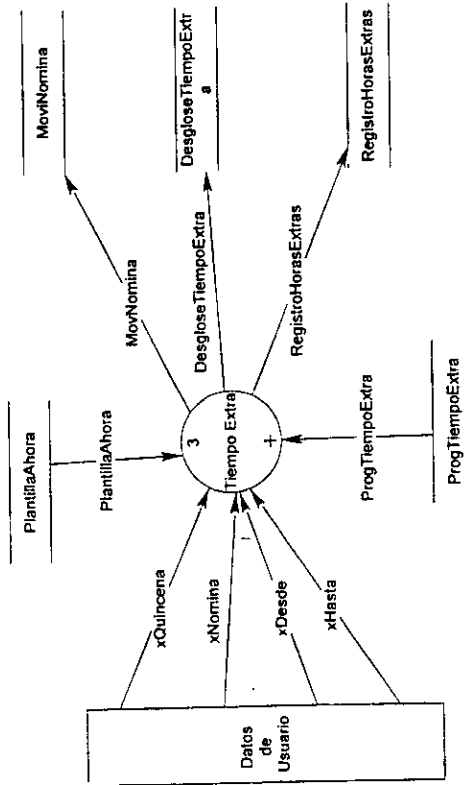


Figura 5.1.3-5 DFD Tiempo Extra Nivel 1

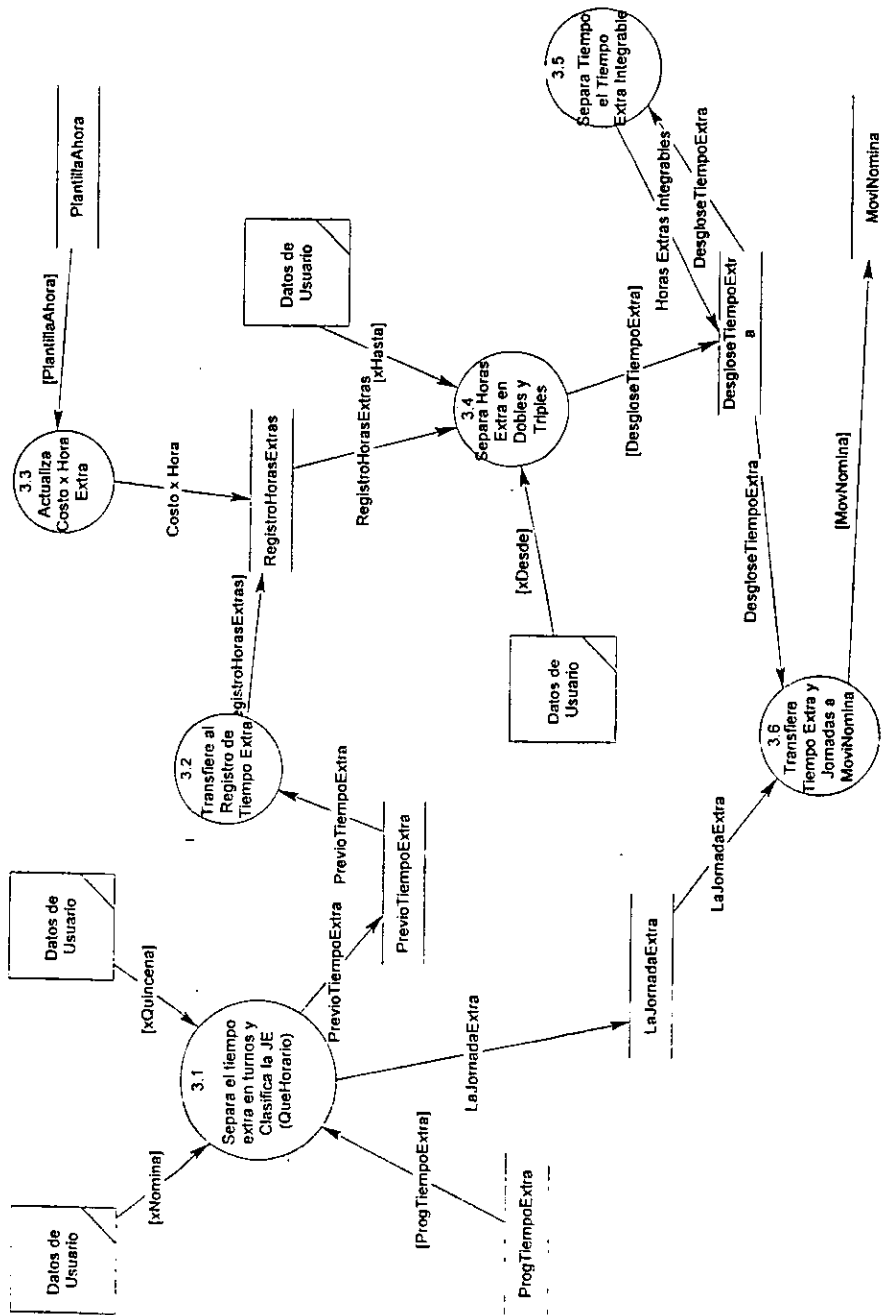


Figura 5.1.3-6 DFD Tiempo Extra Nivel 2

5.1.3.4 Deposito Bancario.

Los pagos de nómina se realizan por medio de un deposito bancario para lo cual el sistema debe generar un archivo texto con un formato establecido para que pueda transferirse electrónicamente. Para tal efecto se registran las cuentas individuales y la de la empresa.

En este proceso el sistema valida la existencia de cuenta para solo enviar datos correctos, también inserta el neto que se depositará con el concepto 298 el cual se utilizará para cuadrar la póliza contable. A toda nómina es posible generarle su deposito.

En las figuras 5.1.3-7 y 5.1.3-8 se muestra el diagrama de flujo de este módulo.

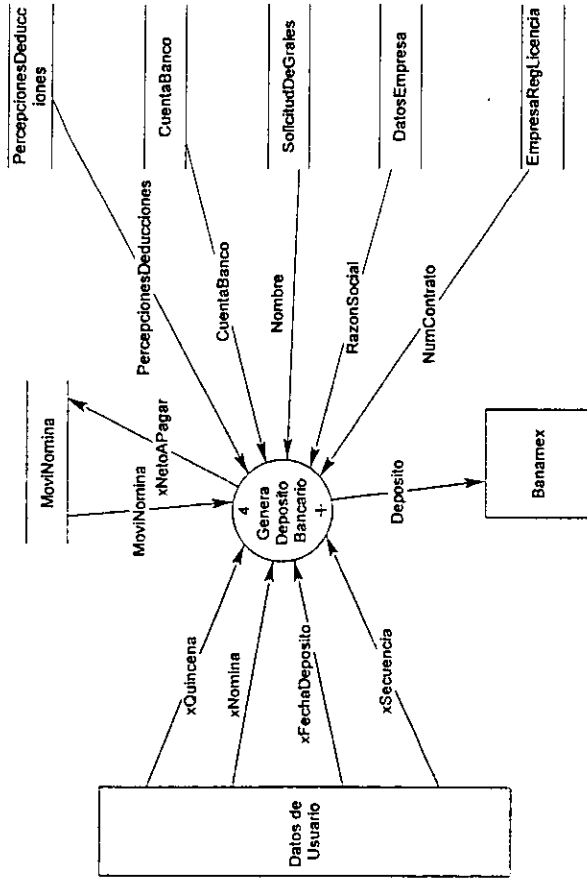


Figura 5.1.3-7 DFD Depósito Bancario Nivel 1

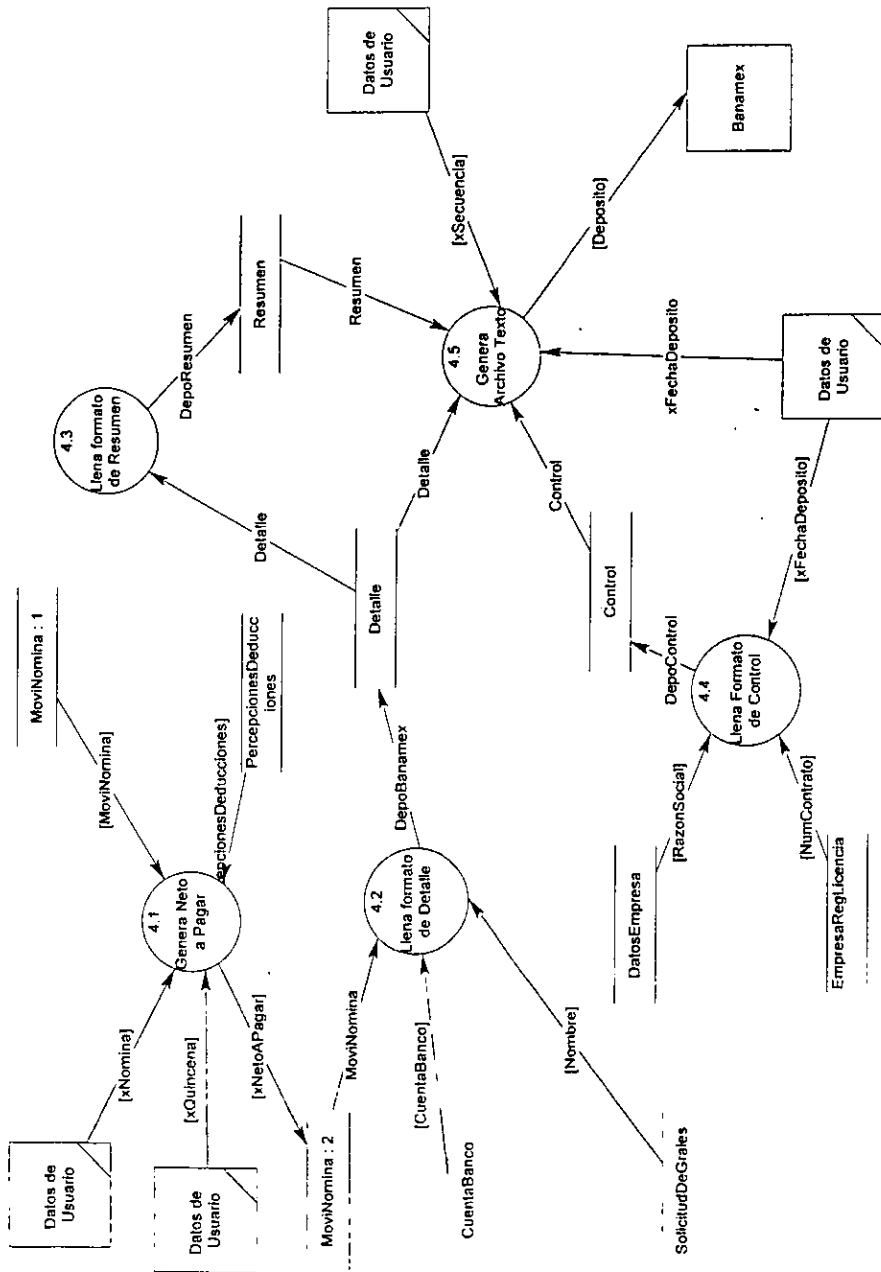


Figura 5.1.3- 8 DFD Deposito Bancario Nivel 2

5.1.3.5 Aplicación Contable.

El sistema puede generar la póliza contable de cualquier nómina que este registrada, por lo mismo es necesario generar el deposito bancario (aunque no se leposite) para así calcular el neto a pagar, dato que siempre debe contener una póliza.

A cada concepto de nómina se le puede definir la manera de conformar su asiento contable, que tendrá que ser 000-00-000, y básicamente puede ser de tres formas distintas:

- Proceso

Los cinco primeros dígitos serán del departamento al que corresponde el trabajador.

Los últimos tres se le asigna al concepto en cuestión

APLICACIÓN

Los ocho dígitos se asignan al concepto en cuestión sin importar el departamento del trabajador.

DEUDOR DIVERSO.

Los primero cinco dígitos corresponden al concepto en cuestión y los últimos tres a los tres dígitos menos significativos del código del trabajador

Así mismo se puede hacer el calculo provisiones contables estableciendo un factor, como por ejemplo: Existe un impuesto de 2% sobre nómina, en este caso se define el factor de 0.02 a todas las percepciones para que se provisione este impuesto que se debe pagar.

Al final el sistema emite un archivo de formato específico (ZIM) para poder ser transferido al sistema de contabilidad.

En las figuras 5.1.3-9, 5.1.3-10 y 5.1.3-11 se muestra el diagrama de flujo de este módulo.

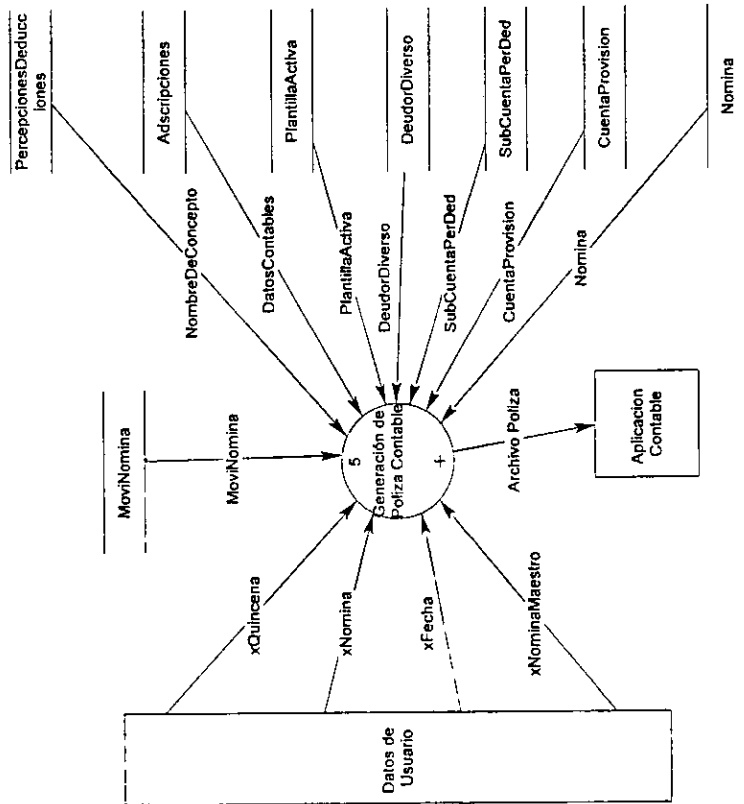


Figura 5.13- 9 DFD Aplicación Contable Nivel 1

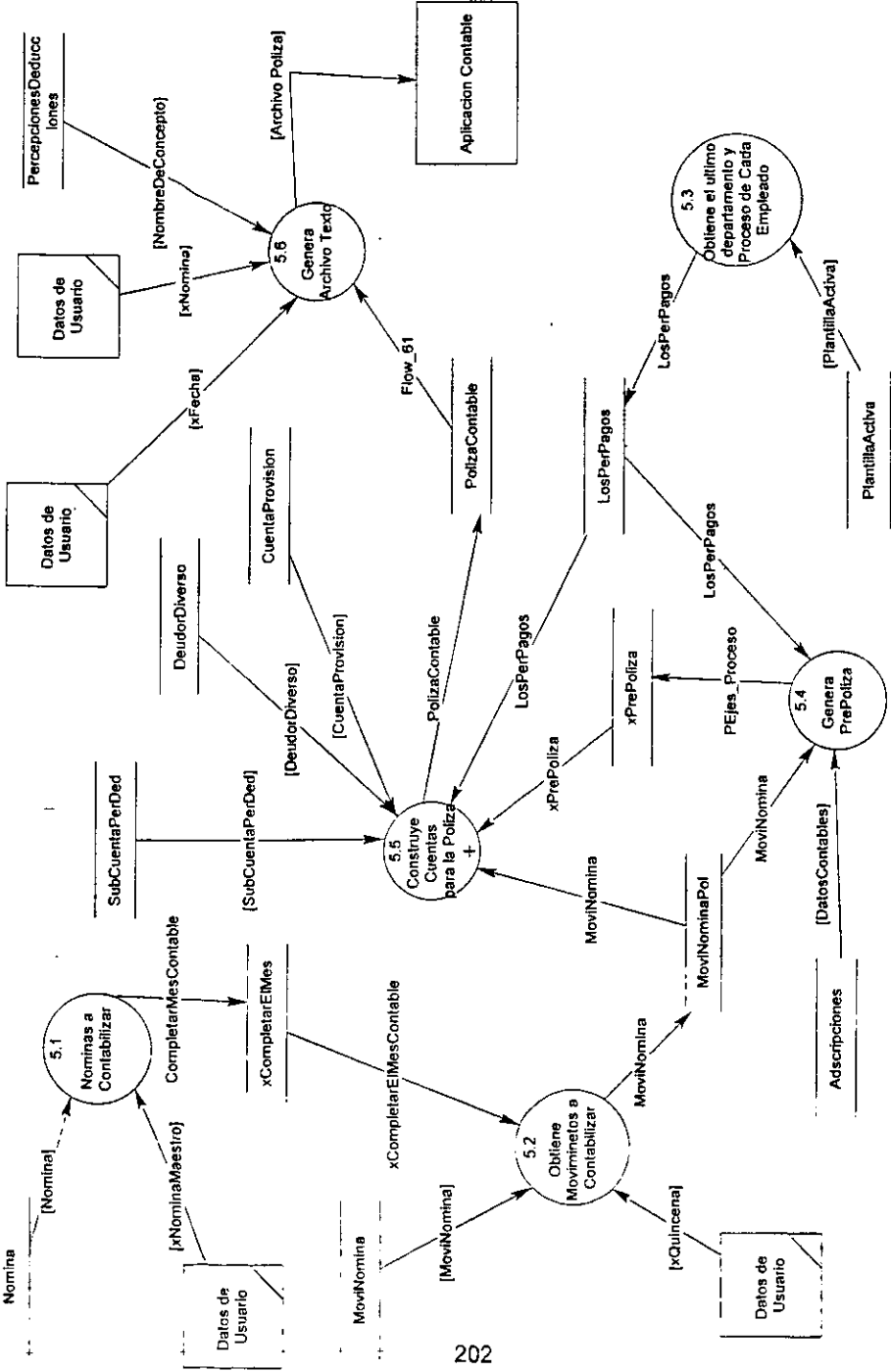


Figura 5.1.3- 10 DFD Aplicación Contable Nivel 2

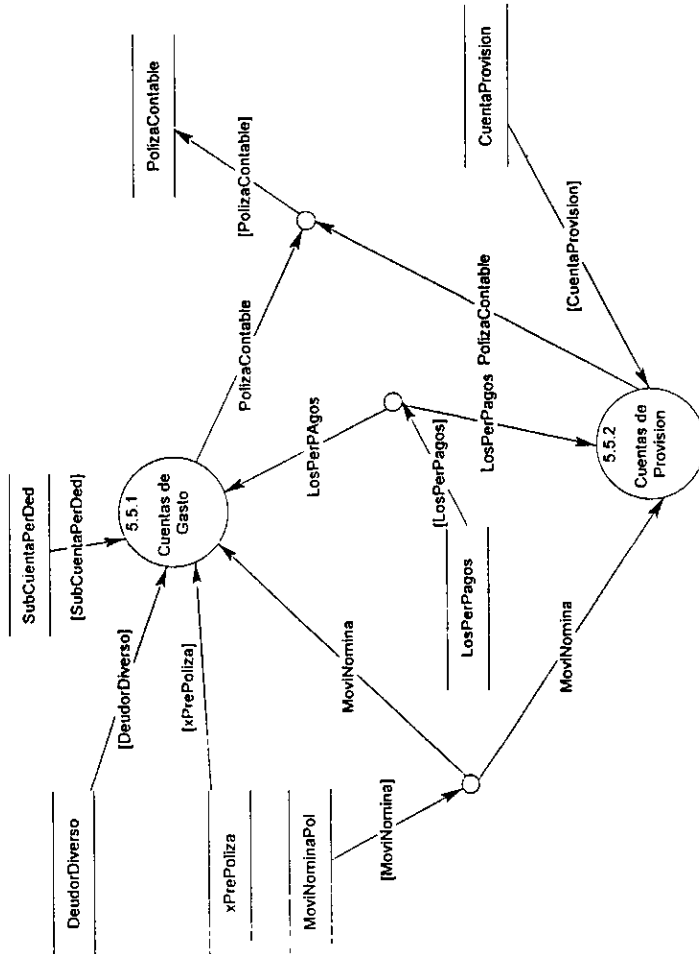


Figura 5.1.3- 11 DFD Aplicación Contable Nivel 3

5.1.3.6 Procesa Evaluaciones.

El desempeño del empleado se mide por medio de evaluaciones las cuales se registran en el sistema y también son utilizadas para calcular el aumento de sueldo, este proceso esta definido dentro de las políticas de este grupo de empresas, a continuación se explica brevemente.

Cada empleados esta asignado a un puesto el cual tiene uno y solo un nivel que es como la empresa identifica el rango de sueldo que puede tener dicho empleado, de igual manera el corporativo emite una tabla donde se relaciona el nivel, % de penetración (proporción del sueldo con respecto al nivel) y calificación con el aumento correspondiente.

El modulo toma los datos del corporativo y los datos actuales del personal para calcular el nuevo sueldo y % de penetración, los cuales presenta en un reporte para su autorización o modificaciones, una vez que se da por bueno el calculo se afecta el cambio de sueldo para que el usuario genere el nuevo integrado.

En las figuras 5.1.3-12 y 5.1.3-13 se muestra el diagrama de flujo de este módulo.

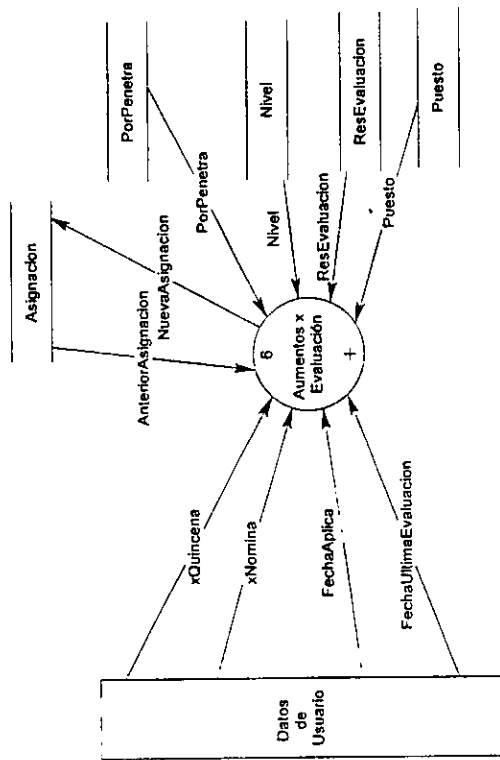


Figura 5.1.3- 12 DFD Procesa Evaluaciones Nivel 1

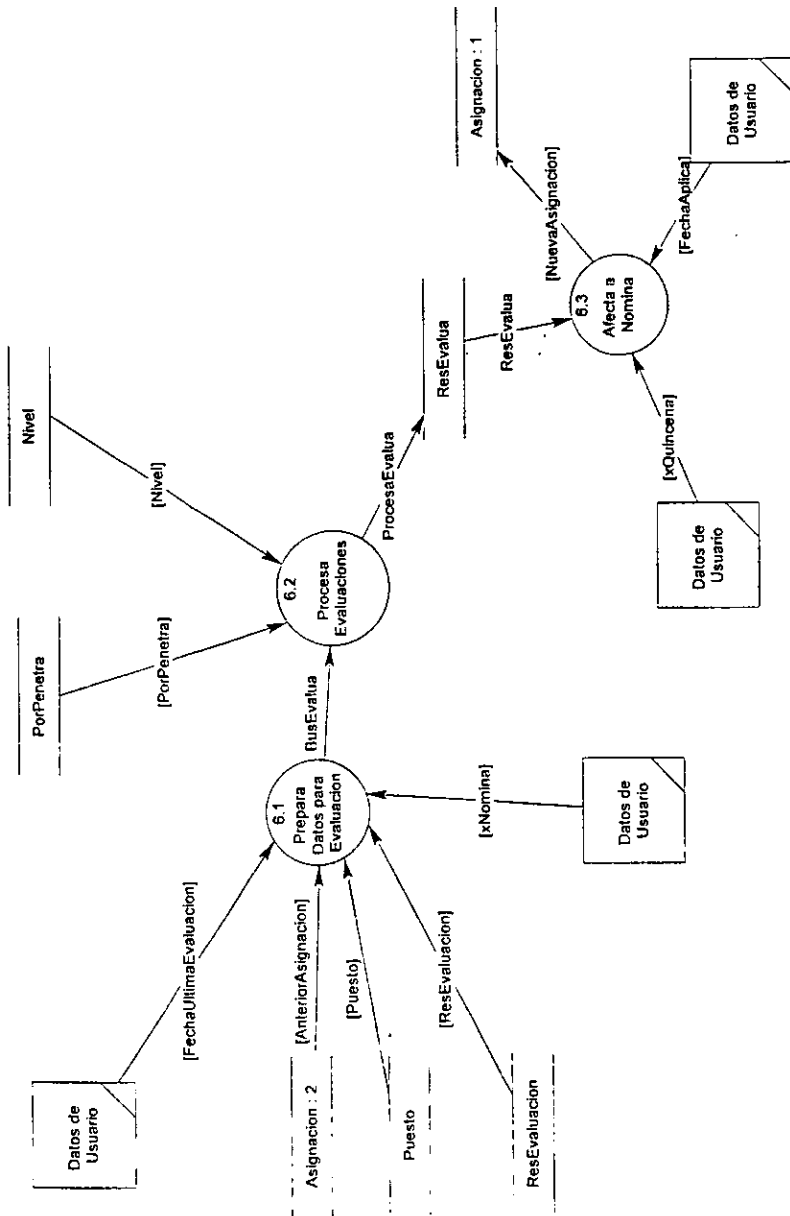


Figura 5.1.3- 13 DFD Procesa Evaluaciones Nivel 2

5.1.3.7 Calculo de Gratificación Anual.

Con base en desempeño y nivel de cada empleado a final de año se paga una gratificación de acuerdo a una tabla que emite el corporativo.

Para el calculo de los días de gratificación se toman en cuenta el promedio de calificaciones de las ultimas evaluaciones, si no tuviera se asume que tiene la media. El corporativo emite una tabla de Plan de Bonos que relaciona Nivel y Promedio de calificaciones con los días de gratificación.

La función del modulo es calcular los días y emitir un reporte para su autorización, al final transfiere la información a la nómina para que se calculen los montos de impuesto.

En las figuras 5.1.3-14 y 5.1.3-15 se muestra el diagrama de flujo de este módulo.

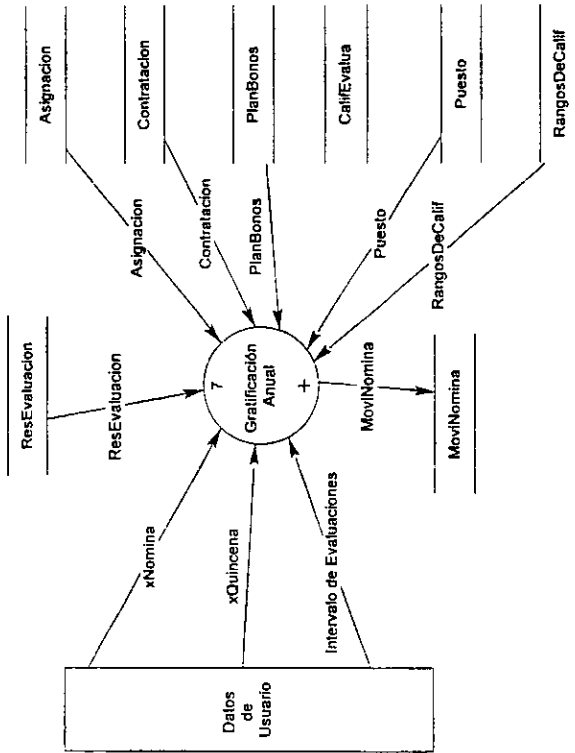


Figura 5.1.3-14 DFD Calculo de Gratificación Anual Nivel 1

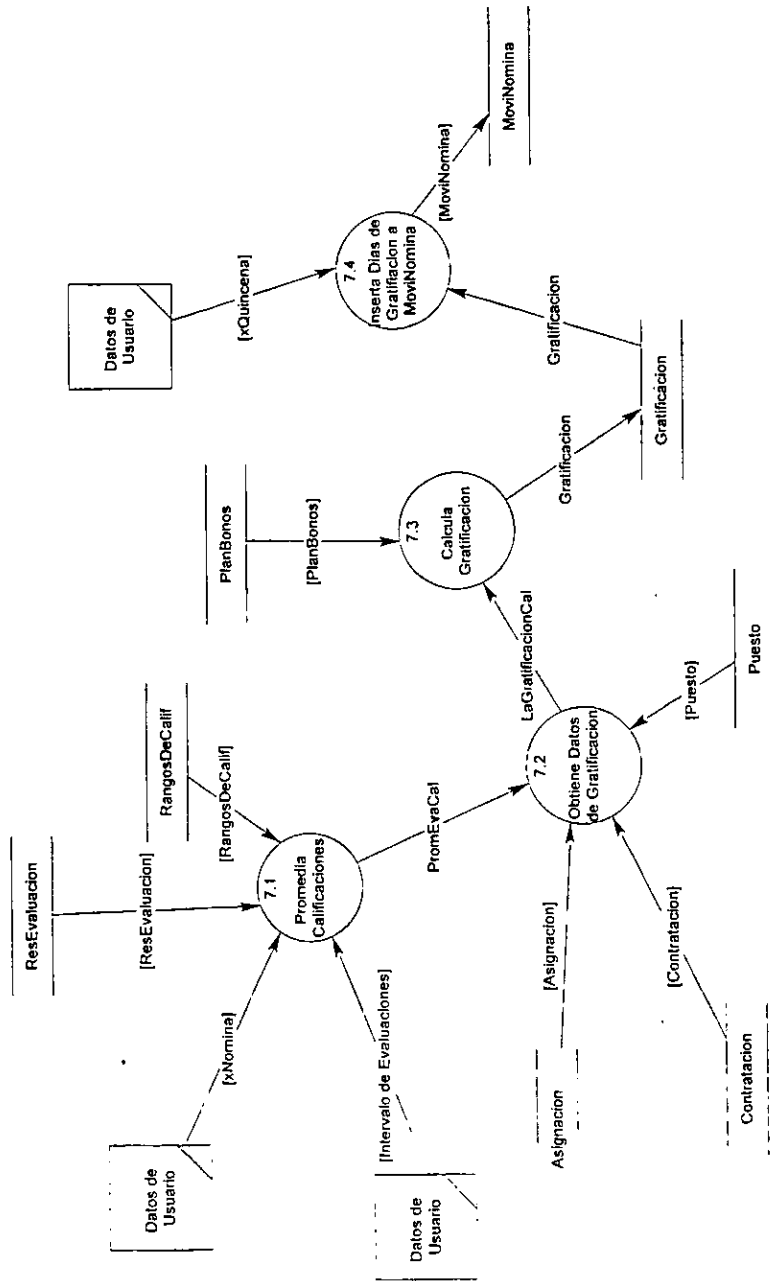


Figura 5.1.3-15 DFD Calculo de Gratificacion Anual Nivel 2

5.1.3.8 Liquidación de Ahorro.

Los empleados aportan periodo a periodo una cantidad que se acumula y se transfiere a una cuenta que genera intereses, al final del ciclo de ahorro se devuelve su aportación y los intereses proporcionales por cada empleado.

En la liquidación también es posible detectar los adeudos que se tengan por prestamos que no se hubieran cubierto, los cuales son descontados de la devolución.

Emite un reporte para validar los datos y proceder a insertar los datos a la nómina y generar los recibos.

En las figuras 5.1.3-16 y 5.1.3-17 se muestra el diagrama de flujo de este módulo.

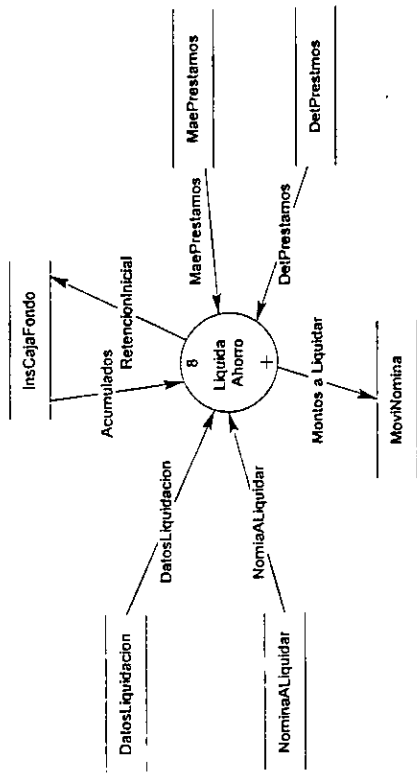


Figura 5.1.3-16 DFD Liquidacion de Ahorro Nivel 1

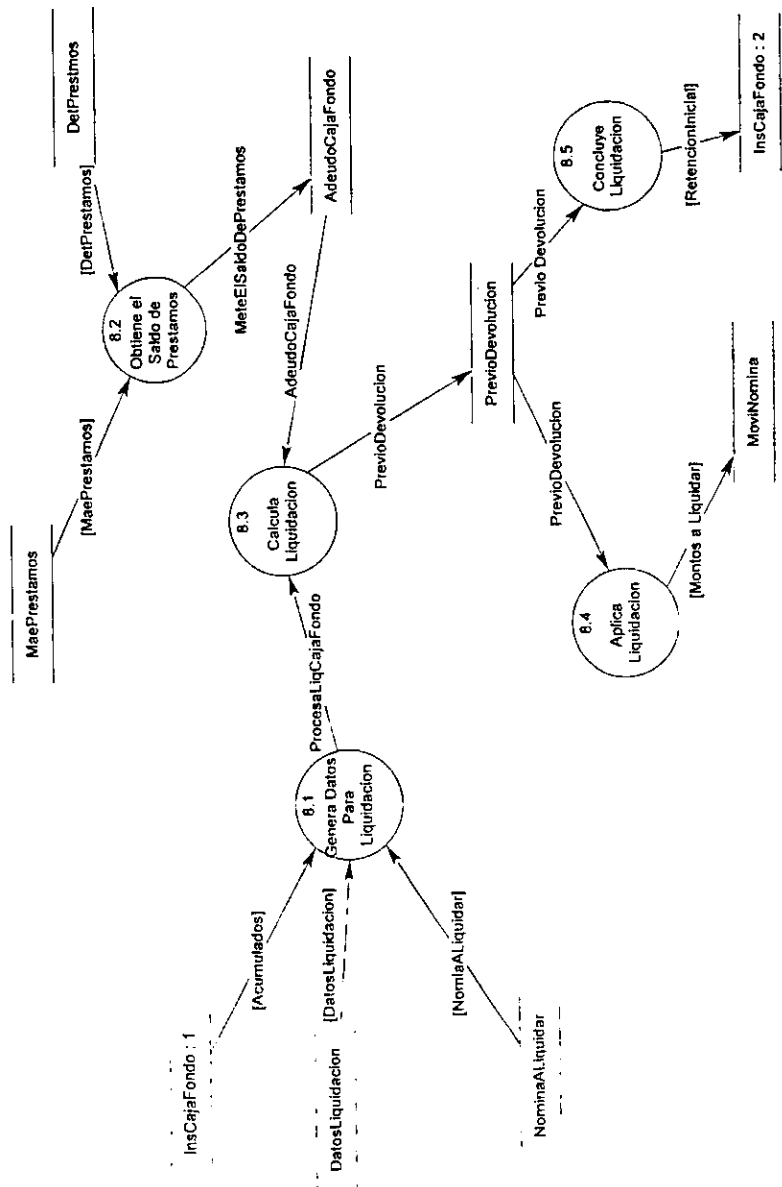


Figura 5.1.3- 17 DFD Liquidación de Ahorro Nivel 2

5.1.3.9 Ajuste Anual de Impuestos.

El ajuste anual como su nombre lo indica es un proceso que se ejecuta cada fin de año para calcular el impuesto real que y en caso de existir alguna diferencia a cargo o a favor aplicar el movimiento en la ultima nómina del año.

La información que toma es la que existe en los acumulados que se van generando con cada proceso de nómina, hacienda emite las tablas de impuesto a aplicar y la compañía calcula el factor de subsidio correspondiente. El calculo de ajuste anual puede ser por dos procedimientos, Ley y Reglamento, se escoge la que mas convenga al empleado.

El ajuste anual no se aplica a todos los empleados sólo aquellos que tienen un año completo trabajando en la empresa. Lo que si se emite para todos es la constancia de percepciones en el formato especificado por Hacienda.

En las figuras 5.1.3-18 y 5.1.3-19 se muestra el diagrama de flujo de este módulo.

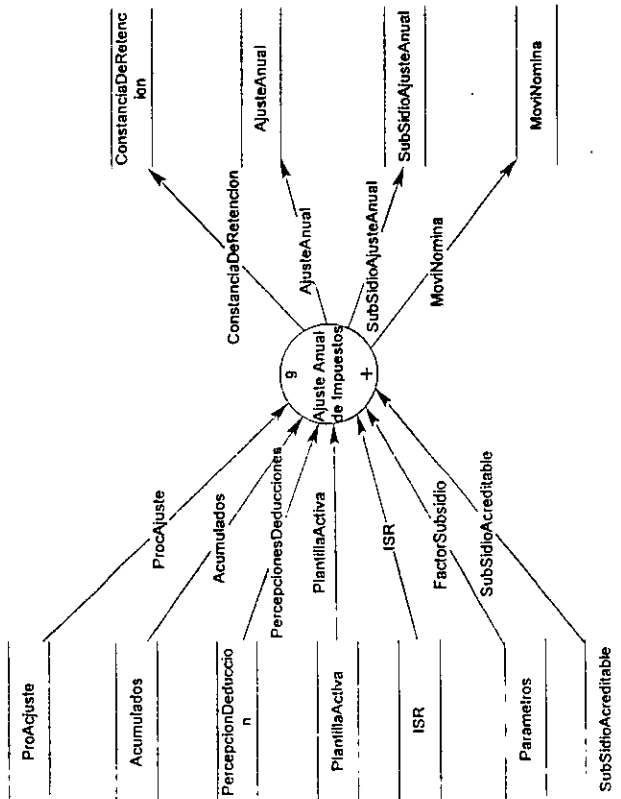


Figura 5.1.3- 18 DFD Ajuste Anual de Impuestos Nivel 1

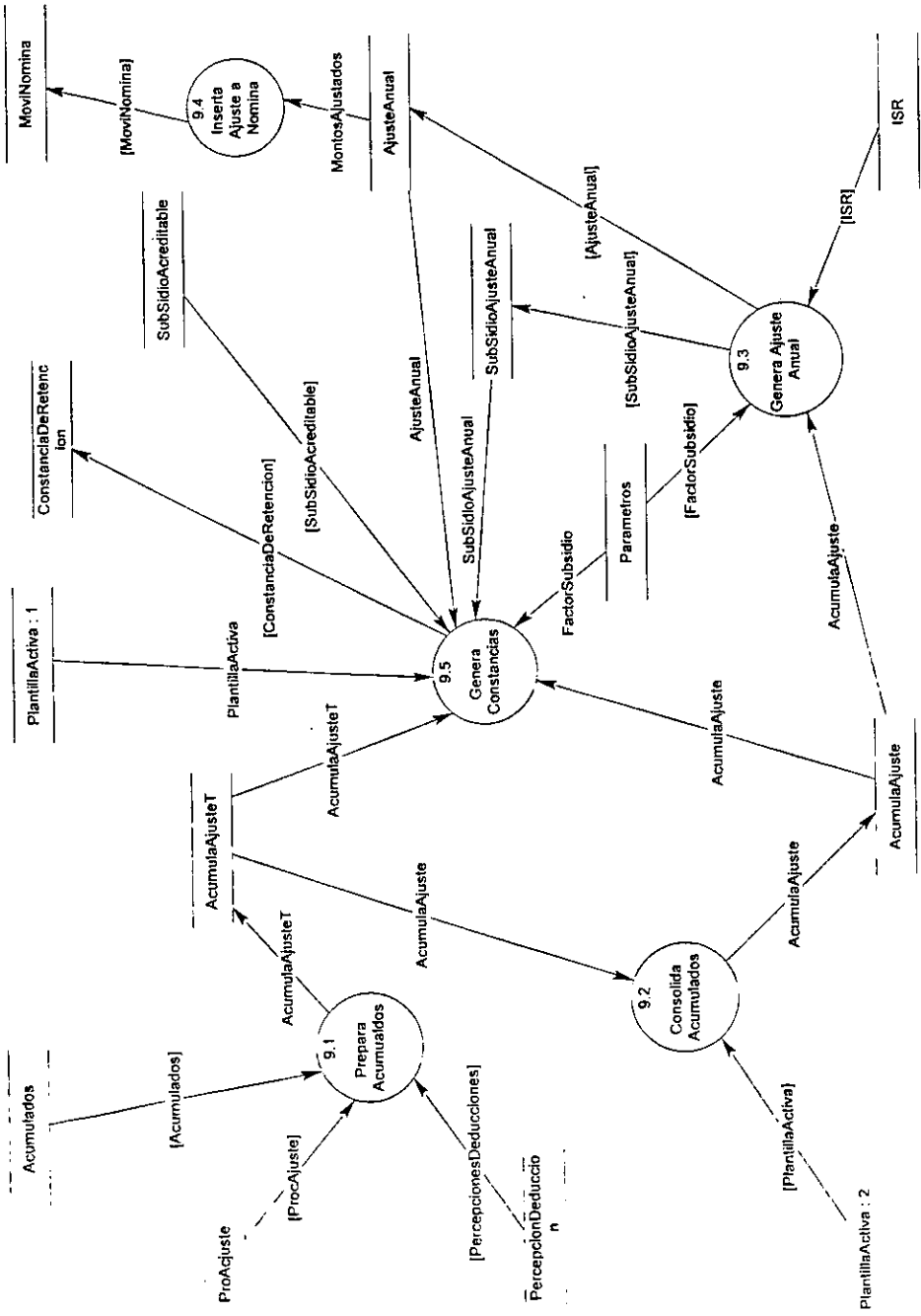


Figura 5.1.3- 19 DFD Ajuste Anual de Impuestos Nivel 2

5.1.3.10 Generación de una Nómina.

La nómina es el proceso centralizador de varios módulos en donde diferentes usuarios capturan y pre-procesan su información como son los de ahorro, tiempo extra, movimientos programados entre otros.

El proceso de nómina toma la mayoría de sus datos de la tabla MoviNomina que es donde los módulos periféricos dejan sus datos.

La solución de la nómina esta basada en la solución de las formulas de la nómina a continuación se presenta el algoritmo básico para esta solución

```
Para todos los Empleados Activos
  Para todos los Conceptos de Nomina
    Monto = ResuelveFormula(xEmpleado,xConcepto,xFormula)
    Insertar_resultado(Monto,xEmpleado,xConcepto)
    Siguiente Concepto
  Fin Para todos los Conceptos
  Siguiente Empleado
Fin Para Todos los Empleados
```

En las figuras 5.1.3-20 y 5.1.3-21 se muestra el diagrama de flujo de este módulo.

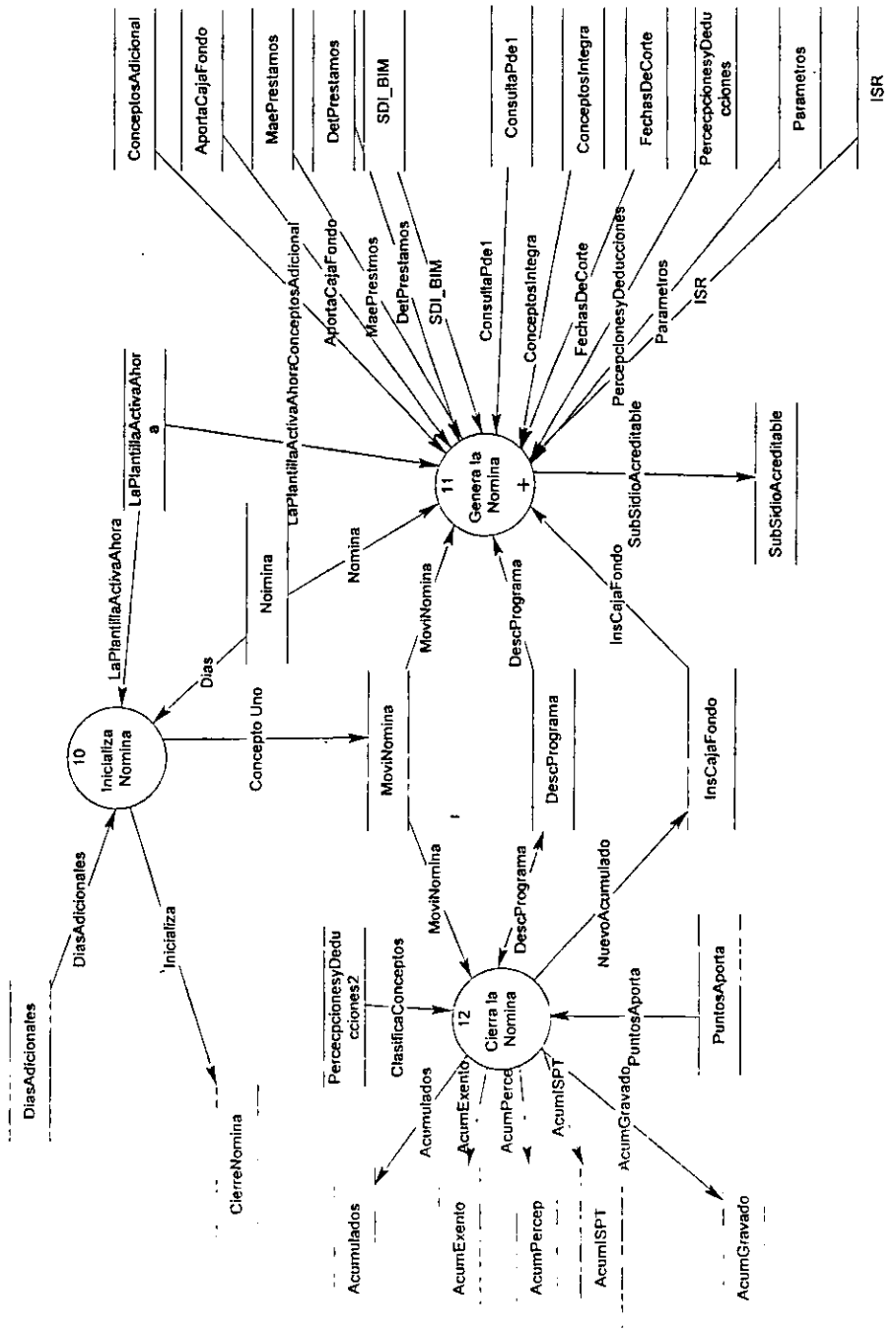


Figura 5.1.3- 20 DFD Generación de una Nómina Nivel 1

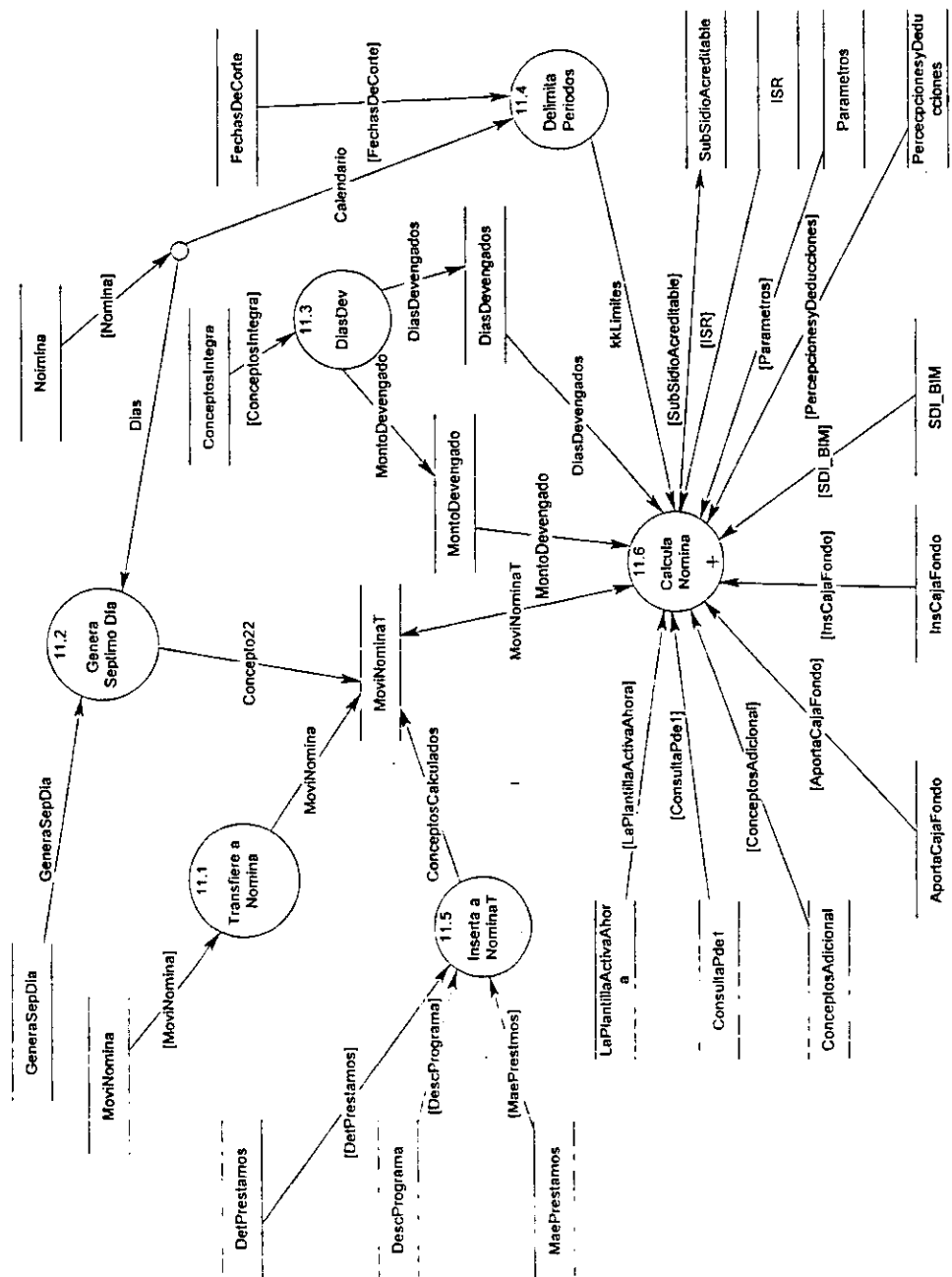


Figura 5.1.3-21 DFD Generación de una Nómina Nivel 2

5.1.4 Diccionario de datos.

En el siguiente listado se muestra el diccionario de datos con el siguiente formato:

Entity Name: <Nombre de la Entidad o Tabla>

Column Names : <Nombre de la columna> <Tipo De Dato>

Primary key: <Campo(s) llave de la tabla>

Foreing key: <Campos(s) llave pero que provienen de otra tabla>

References: <Nombre de Tabla que proporciona los datos> <(Campos)>

1 Entity Name: Adscripciones

Primary Key: ClaveDeAdscripcion
Column Names: ClaveDeAdscripcion Long Integer
Foreign Keys: ClaveDeGerencia
References: Gerencias (ClaveDeGerencia)

Column Names: NombreDeAdscripcion Text(55)

Column Names: AdscripcionAbreviada Text(20)

Column Names: ClaveDeEscalafon Double

Column Names: CuentaContable Text(14)

Column Names: EntradaComedor Date/Time

Column Names: SalidaComedor Date/Time

Column Names: Tipo Text(15)

Column Names: Funciones Memo

Column Names: Fecha_UltimaRev Date/Time

Column Names: TipoProductivo Text(50)

Column Names: ClaveDeGerencia Integer

Column Names: CuentaContableSIN Text(14)

Column Names: ParFondo Double

Column Names: NomResponsable Text(50)

2 Entity Name: Asignacion

Primary Key: ClaveDeEmpleado
Column Names: ClaveDeEmpleado Long Integer NOT NULL
Foreign Keys: ClaveDeNomina
References: Nomina (ClaveDeNomina)

Primary Key: FechaCambio
Column Names: FechaCambio Date/Time NOT NULL
Foreign Keys: ClaveDeSucursal
References: EmpresSucursal (ClaveDeSucursal)

Column Names: ClaveDeNomina Integer
Foreign Keys: ClaveDePuesto
References: Puesto (ClaveDePuesto)

Column Names: ClaveDeAdscripcion Long Integer
Foreign Keys: ClaveDeAdscripcion
References: Adscripciones (ClaveDeAdscripcion)

Column Names: NumeroDeQuincena Integer
Foreign Keys: ClaveDeEmpleado
References: Contratacion (ClaveDeEmpleado)

Column Names: ClaveDePuesto Integer

Column Names: ClaveDeSucursal Integer

Column Names: PorPenetra Double

Column Names: Sueldo Currency

Column Names: Estatus Text(1)

3 Entity Name: BajaDefinitiva

Primary Key: ClaveDeEmpleado
Column Names: ClaveDeEmpleado Long Integer NOT NULL
Foreign Keys: ClaveDeEmpleado
References: Finiquitos (ClaveDeEmpleado)

4 Entity Name: Categorias

Primary Key: ClaveDeCategoria
Column Names: ClaveDeCategoria Integer
Foreign Keys: ClaveTabulador, NumeroDeQuincena
References: Tabuladores (ClaveTabulador, NumeroDeQuincena)

Column Names: ClaveTabulador Integer

Column Names: NumeroDeQuincena Integer

Column Names: NombreDeCategoria Text(30)

Column Names: ClaveDeTabulador Integer

Column Names: TipoManoDeObra Double

5 Entity Name: Contratacion

Primary Key: ClaveDeEmpleado
Column Names: ClaveDeEmpleado Long Integer
Foreign Keys: ClaveDeSolicitud
References: SolicitudDGrales (ClaveDeSolicitud)

Column Names: ClaveDeSolicitud Long Integer

Column Names: Tipo Text(10)

Column Names: TipoEvantual Text(15)

Column Names: Foto OLE Object

Column Names: FechaAlta Date/Time

Column Names: FechaIniReno Date/Time

Column Names: FechaReno Date/Time

6 Entity Name: CuentaBanco

Primary Key: ClaveDeEmpleado

Column Names: ClaveDeEmpleado Long Integer NOT NULL

Foreign Keys: ClaveDeEmpleado

References: Contratacion (ClaveDeEmpleado)

Column Names: Moneda Text(3)

Column Names: TipoDeCuenta Text(2)

Column Names: Sucursal Text(4)

Column Names: Cuenta Text(16)

7 Entity Name: DatosEmpresa

Primary Key: ClaveDeEmpresa

Column Names: ClaveDeEmpresa Integer

Column Names: Corporativo Text(50)

Column Names: ClaveDeGrupo Integer

Column Names: Grupo Text(50)

Column Names: RazonSocial Text(50)

Column Names: Constitucion Date/Time

Column Names: InicioOperaciones Date/Time

Column Names: Siglas_Em Text(10)

Column Names: Giro Text(60)

Column Names: RFC_Em Text(15)

Column Names: Direccion Text(60)

Column Names: NoExt Text(10)

Column Names: Ubica Text(15)

Column Names: Colonia Text(60)

Column Names: Poblacion Text(50)

Column Names: DelegaMunici Text(60)

21 Entity Name: JornadasLaborables

Primary Key: ClaveDeJornada

Column Names: ClaveDeJornada Integer

Column Names: Descripcion Text(30)

Column Names: Horas Double

Column Names: Ent_labores Date/Time

Column Names: Sal_labores Date/Time

Column Names: HorasNormal Date/Time

Column Names: Ent_Sabado Date/Time

Column Names: Sal_Sabado Date/Time

Column Names: HorasSabado Date/Time

Column Names: Tot_labores Integer

Column Names: MaximaTolerancia Double

Column Names: Rotacion Byte

Column Names: OtroDia Yes/No

Column Names: TrabajaDom Yes/No

22 Entity Name: MaeVacaciones

Primary Key: ClaveDeEmpleado

Column Names: ClaveDeEmpleado Long Integer

Foreign Keys: ClaveDeEmpleado

References: Contratacion (ClaveDeEmpleado)

Column Names: ClaveDeNomina Integer

Column Names: DiasDelPeriodo Integer

Column Names: DiasPeriodoAnterios Integer

Column Names: DiasDisfrutados Integer

Column Names: DiasSaldo Integer

Column Names: DiasXDisfrutar Long Integer

Column Names: PrimaExenta Double

Column Names: PrimaExentaT Double

Column Names: PrimaExentaU Double

23 Entity Name: MotivFiniquito

Primary Key: Motivo

Column Names: Motivo Integer

Column Names: DesMotivo Text(30)

24 Entity Name: Nivel

Primary Key: ClaveDeNivel

Column Names: ClaveDeNivel Text(8) NOT NULL

Column Names: Mon_Min Double

Column Names: Mon_Med Double

Column Names: Mon_Max Double

25 Entity Name: Nomina

Primary Key: ClaveDeNomina

Column Names: ClaveDeNomina Integer

Column Names: ClaveDeCalendario Text(18)

Column Names: ClaveDePeriodo Text(18)

Column Names: NombreDeNomina Text(50)

Column Names: Dias Integer

Column Names: Tipo Text(1)

Column Names: Activa Yes/No

Column Names: TablaSR Double

Column Names: FechaCorte Integer

Column Names: Quincenas Integer

Column Names: NominaMaestro Integer

Column Names: Confidencial Yes/No

26 Entity Name: NomTabul

Primary Key: ClaveTabulador

Column Names: ClaveTabulador Integer

Column Names: NombreDeTabulador Text(70)

Column Names: ClaveDeEmpresa Integer

27 Entity Name: ProgramacionXTurno

Primary Key: ClaveDeEmpleado
Column Names: ClaveDeEmpleado Long Integer
Foreign Keys: ClaveDeJornada
References: JornadasLaborables (ClaveDeJornada)

Primary Key: ClaveDeJornada
Column Names: ClaveDeJornada Integer
Foreign Keys: ClaveDeEmpleado
References: Contratacion (ClaveDeEmpleado)

Primary Key: Fechalnicio
Column Names: Fechalnicio Date/Time

Column Names: ClaveDeNomina Integer

Column Names: FechaFinal Date/Time

28 Entity Name: Puesto

Primary Key: ClaveDePuesto
Column Names: ClaveDePuesto Integer
Foreign Keys: ClaveDeNivel
References: Nivel (ClaveDeNivel)

Column Names: ClaveDeAdscripcion Long Integer

Column Names: ClaveDeArea Integer

Column Names: ClaveDeNivel Text(8)

Column Names: NombreDePuesto Text(50)

Column Names: PropPuesto Text(250)

Column Names: PropSubor Text(250)

Column Names: Valores Currency

Column Names: Libertad Text(250)

Column Names: Posicion Text(50)

Column Names: EnDireccion Double

Column Names: EnTramites Currency

29 Entity Name: RSDatosGenerales

Primary Key: ClaveDeSolicitud
Column Names: ClaveDeSolicitud Long Integer NOT NULL
Foreign Keys: ClaveDeSolicitud
References: SolicitudDGrates (ClaveDeSolicitud)

Column Names: TipoDeDato Text(50)

Column Names: Descripcion Memo

30 Entity Name: RSEscolaridad

Primary Key: ClaveDeSolicitud

Column Names: ClaveDeSolicitud Long Integer NOT NULL

Foreign Keys: ClaveDeSolicitud

References: SolicitudDGrales (ClaveDeSolicitud)

Column Names: TipoEscolaridad Text(50)

Column Names: NombreEscuela Text(50)

Column Names: CuentaEscolaridad Yes/No

Column Names: TiempoEstudio Double

Column Names: Calificacion Text(20)

31 Entity Name: RSFamiliares

Primary Key: ClaveDeSolicitud

Column Names: ClaveDeSolicitud Long Integer NOT NULL

Foreign Keys: ClaveDeSolicitud

References: SolicitudDGrales (ClaveDeSolicitud)

Primary Key: Familiar

32 Column Names: Familiar Text(50)

Column Names: Tipo Text(2)

Column Names: Sexo Integer

Column Names: FechaNac Date/Time

33 Entity Name: SDI_BIM

Primary Key: ClaveDeEmpleado

Column Names: ClaveDeEmpleado Long Integer

Foreign Keys: ClaveDelmpuesto

References: TipoImpuesto (ClaveDelmpuesto)

Primary Key: ClaveDelmpuesto

Column Names: ClaveDelmpuesto Text(15)

Foreign Keys: ClaveDeEmpleado

References: Contratacion (ClaveDeEmpleado)

Primary Key: Fecha_Alta

Column Names: Fecha_Alta Date/Time

Column Names: Fecha_Baja Date/Time

Column Names: NumeroDeQuincena Integer

Column Names: Bimestre Integer

Column Names: MontoSDI_1 Double

Column Names: Estado Text(50)

Column Names: SDI Double

Column Names: MontoSDI Double

34 Entity Name: SolicitudDFamilia

Primary Key: ClaveDeSolicitud

Column Names: ClaveDeSolicitud Long Integer

Foreign Keys: ClaveDeSolicitud

References: SolicitudDGrales (ClaveDeSolicitud)

Column Names: Padre Text(60)

Column Names: Vive1 Text(3)

Column Names: Edad Double

Column Names: FechaNac Date/Time

Column Names: Ocupacion Text(50)

Column Names: Madre Text(60)

Column Names: Vive2 Text(3)

Column Names: Edad2 Double

Column Names: FechaNac2 Date/Time

Column Names: Ocupacion2 Text(50)

Column Names: Conyuge Text(60)

Column Names: Vive3 Text(3)

Column Names: Edad3 Double

Column Names: FechaNac3 Date/Time

Column Names: Ocupacion3 Text(50)

35 Entity Name: SolicitudDGrales

Primary Key: ClaveDeSolicitud

Column Names: ClaveDeSolicitud Long Integer

Foreign Keys: ClaveDeEmpresa
References: DatosEmpresa (ClaveDeEmpresa)

Column Names: ClaveDeEmpresa Integer

Column Names: ClaveDeEmpleado Long Integer

Column Names: Nombre Text(50)

Column Names: Paterno Text(50)

Column Names: Materno Text(50)

Column Names: Nombres Text(50)

Column Names: TipoTraba Text(20)

Column Names: ClaveDePuesto Integer

Column Names: MotivSoli Text(40)

Column Names: DeptoSoli Long Integer

Column Names: Sexo Text(10)

Column Names: Fec_nac Date/Time

Column Names: Lug_nac Text(50)

Column Names: Nacionalidad Text(30)

Column Names: Con_migra Text(50)

Column Names: FM Text(5)

Column Names: NumeroFM Double

Column Names: Edo_Civil Text(15)

Column Names: Lug_Matrimonio Text(50)

Column Names: Fec_Matrimonio Date/Time

Column Names: Direccion Text(18)

Column Names: CodPos Text(18)

Column Names: Municipio Text(50)

Column Names: Entidad Text(50)

Column Names: Telefono Text(15)

Column Names: Tel_Urgencia Double

Column Names: RFC Text(15)

Column Names: Imss Text(20)

Column Names: Cartilla Text(20)

Column Names: Pasaporte Text(30)

Column Names: Ven_Pasaporte Date/Time

Column Names: Licencia Text(20)

Column Names: Tip_Licencia Text(15)

Column Names: Religion Text(20)

Column Names: CURP Text(50)

36 Entity Name: Tabuladores

Primary Key: ClaveTabulador

Column Names: ClaveTabulador Integer

Foreign Keys: ClaveTabulador

References: NomTabul (ClaveTabulador)

Primary Key: NumeroDeQuincena

Column Names: NumeroDeQuincena Integer

Column Names: MontoTabulador Currency

Column Names: FechaIniV Date/Time

Column Names: FechaTerV Date/Time

_

37 Entity Name: TipoFiniquito

Primary Key: TipoFiniquito

Column Names: TipoFiniquito Integer NOT NULL

Column Names: DesFiniquito Text(30)

38 Entity Name: TipoImpuesto

Primary Key: ClaveDelimpuesto

Column Names: ClaveDelimpuesto Text(15)

Column Names: Descripcion_de_impuesto Memo

Column Names: TopeSMG1 Double

Column Names: TopeSMG2 Double

39 Entity Name: TurnosDiarios

Primary Key: ClaveDeJornada
Column Names: ClaveDeJornada Integer NOT NULL
Foreign Keys: ClaveDeJornada
References: JornadasLaborables (ClaveDeJornada)

Primary Key: Dia
Column Names: Dia Long Integer

Column Names: HoraEntrada Date/Time

Column Names: HoraSalida Date/Time

Column Names: CambiaDia Byte

40 Entity Name: Vacaciones

Primary Key: ClaveDeEmpleado
Column Names: ClaveDeEmpleado Long Integer NOT NULL
Foreign Keys: ClaveDeEmpleado
References: MaeVacaciones (ClaveDeEmpleado)

Primary Key: Fecha_Inicial
Column Names: Fecha_Inicial Date/Time NOT NULL

Column Names: ClaveDeNomina Integer

Column Names: Fecha_Final Date/Time

Column Names: FechaRegreso Date/Time

Column Names: DiasTomados Double

Column Names: PrimaExe Double

Column Names: PrimaGra Double

Column Names: ISPTPrima Double

Column Names: NumeroDeQuincena Integer

Column Names: QuincenaInicia Long Integer

Column Names: QuincenaTermina Long Integer

Column Names: TotalQuincena Long Integer

Column Names: Sta_tus Yes/No

41 Entity Name: Acumulados

Column Names: ClaveDeNomina Integer NOT NULL
Primary Key: ClaveDeNomina
Foreign Keys: ClaveDeNomina ClaveDeEmpleado

References: EmpleadoNomina (ClaveDeNomina, ClaveDeEmpleado)

Column Names: ClaveDeConcepto Integer NOT NULL

Primary Key: ClaveDeConcepto

Foreign Keys: ClaveDeConcepto, ClaveDeNomina

References: PercepcionesDeducciones (ClaveDeConcepto, ClaveDeNomina)

Column Names: ClaveDeEmpleado Long Integer NOT NULL

Primary Key: ClaveDeEmpleado

Column Names: MontoAcumulado Double

Column Names: Variables Double

Column Names: ClaveDeConceptoADEP Integer

42 Entity Name: AportaCajaFondo

Column Names: ClaveDeConcepto Integer

Primary Key: ClaveDeConcepto

Foreign Keys: ClaveDeConcepto, ClaveDeNomina, ClaveDeEmpleado

References: InsCajaFondo (ClaveDeConcepto, ClaveDeNomina, ClaveDeEmpleado)

Column Names: NumeroDeQuincena Integer

Primary Key: NumeroDeQuincena

Column Names: ClaveDeNomina Integer NOT NULL

Primary Key: ClaveDeNomina

Column Names: ClaveDeEmpleado Long Integer NOT NULL

Primary Key: ClaveDeEmpleado

Column Names: Monto Double

43 Entity Name: Calendarios

Column Names: ClaveDeCalendario Integer NOT NULL

Primary Key: ClaveDeCalendario

Column Names: NombreDeCalendario Text(18)

44 Entity Name: CambiosTemporalesC

Column Names: ClaveDeEmpleado Long Integer

Primary Key: ClaveDeEmpleado

Foreign Keys: ClaveDeEmpleado

References: Contratacion (ClaveDeEmpleado)

Column Names: NumeroDeQuincena Double

Primary Key: NumeroDeQuincena

Column Names: FechaInicio Date/Time

Primary Key: FechaInicio

Column Names: FechaTermina Date/Time

Column Names: NuevaCategoria Integer

Column Names: CategoriaActual Integer

Column Names: MotivoCambio Text(15)

Column Names: NumDeHoras Double

Column Names: ClaveDeJornada Integer

Column Names: DiasPorPagar Integer

Column Names: TiempoDescontar Double

45 Entity Name: CatParametros

Column Names: ClaveDeParametro Integer

Primary Key: ClaveDeParametro

Column Names: NombreDeParametro Text(50)

Column Names: Descripcion Text(255)

46 Entity Name: CierreNomina

Column Names: NumeroDeQuincena Double

Primary Key: NumeroDeQuincena

Foreign Keys: ClaveDeNomina

References: Nomina (ClaveDeNomina)

Column Names: ClaveDeNomina Integer NOT NULL

Primary Key: ClaveDeNomina

Column Names: FechaDelInicializa Date/Time

Column Names: FechaDeGeneracion Date/Time

Column Names: FechaDeCierre Date/Time

Column Names: Estado Text(2)

47 Entity Name: ConceptoIntegra

Column Names: ClaveDeConcepto Integer

Primary Key: ClaveDeConcepto

Foreign Keys: ClaveDeConcepto, ClaveDeNomina

References: PercepcionesDeducciones (ClaveDeConcepto, ClaveDeNomina)

Column Names: ClaveDelImpuesto Text(15)

Primary Key: ClaveDelImpuesto

Column Names: ClaveDeNomina Integer NOT NULL
Primary Key: ClaveDeNomina

Column Names: TipoTope Text(15)

Column Names: FormulaTope Text(100)

48 Entity Name: ConceptosAdicional

Column Names: NumeroDeQuincena Integer
Primary Key: NumeroDeQuincena
Foreign Keys: ClaveDeNomina, ClaveDeEmpleado
References: EmpleadoNomina (ClaveDeNomina, ClaveDeEmpleado)

Column Names: ClaveDeConcepto Integer NOT NULL
Primary Key: ClaveDeConcepto
Foreign Keys: ClaveDeConcepto, ClaveDeNomina
References: PercepcionesDeducciones (ClaveDeConcepto, ClaveDeNomina)

Column Names: ClaveDeNomina Integer NOT NULL
Primary Key: ClaveDeNomina

Column Names: ClaveDeEmpleado Long Integer NOT NULL
Primary Key: ClaveDeEmpleado

Column Names: MontoDeConcepto Double

Column Names: Estado Text(12)

Column Names: Num_Credito Text(50)

49 Entity Name: Contratacion

Column Names: ClaveDeEmpleado Long Integer
Primary Key: ClaveDeEmpleado

Column Names: ClaveDeSolicitud Long Integer

Column Names: Tipo Text(10)

Column Names: TipoEventual Text(15)

Column Names: Foto OLE Object

Column Names: FechaAlta Date/Time

Column Names: FechaIniReno Date/Time

Column Names: FechaReno Date/Time

50 Entity Name: CuentaProvision

Column Names: ClaveDeConcepto Integer
Primary Key: ClaveDeConcepto

Foreign Keys: ClaveDeConcepto, ClaveDeNomina
References: PercepcionesDeducciones (ClaveDeConcepto, ClaveDeNomina)

Column Names: ClaveDeNomina Integer
Primary Key: ClaveDeNomina

Column Names: TipoProductivo Text(1) NOT NULL
Primary Key: TipoProductivo

Column Names: CuentaProvision Text(8)

Column Names: SubDeCargo Text(8)

Column Names: Factor Double

51 Entity Name: DescPrograma

Column Names: FolioSistema Long Integer
Primary Key: FolioSistema

Foreign Keys: ClaveDeNomina, ClaveDeEmpleado
References: EmpleadoNomina (ClaveDeNomina, ClaveDeEmpleado)

Column Names: ClaveDeConcepto Integer NOT NULL
Primary Key: ClaveDeConcepto

Foreign Keys: ClaveDeConcepto, ClaveDeNomina
References: PercepcionesDeducciones (ClaveDeConcepto, ClaveDeNomina)

Column Names: ClaveDeNomina Integer NOT NULL
Primary Key: ClaveDeNomina

Column Names: ClaveDeEmpleado Long Integer NOT NULL
Primary Key: ClaveDeEmpleado

Column Names: FolioDescuento Double

Column Names: FechaDescuento Date/Time

Column Names: Estado Text(1)

Column Names: Descripcion Text(50)

Column Names: NumDesc Integer

Column Names: PerIni Integer

Column Names: PerFin integer

Column Names: MontoDescuento Double

Column Names: MontoDescontar Double

Column Names: Abonado Double

Column Names: Num_Pagos Integer

52 Entity Name: DesgloseHorasExtras

Column Names: ClaveDeEmpleado Long Integer NOT NULL

Primary Key: ClaveDeEmpleado

Foreign Keys: ClaveDeEmpleado, ClaveDeNomina, FechaDelIncidencia, NumeroDeQuincena, ClaveDeTurno, CostoPorHora

References: RegistroHorasExtras (ClaveDeEmpleado, ClaveDeNomina, FechaDelIncidencia, NumeroDeQuincena, ClaveDeTurno, CostoPorHora)

Column Names: FechaDelIncidencia Date/Time NOT NULL

Primary Key: FechaDelIncidencia

Column Names: NumeroDeQuincena Long Integer NOT NULL

Primary Key: NumeroDeQuincena

Column Names: ClaveDeNomina Integer NOT NULL

Primary Key: ClaveDeNomina

Column Names: ClaveDeTurno Integer NOT NULL

Primary Key: ClaveDeTurno

Column Names: CostoPorHora Double NOT NULL

Primary Key: CostoPorHora

Column Names: Tipo Text(1)

Column Names: NumeroSemana Integer

Column Names: NumHoras Double

Column Names: Pagas Yes/No

Column Names: Integran Double

Column Names: EITipo Text(1)

53 Entity Name: DetPrestamos

Column Names: FechaDePago Date/Time

Primary Key: FechaDePago

Foreign Keys: FolioDePrestamo, ClaveDeConcepto, ClaveDeNomina, ClaveDeEmpleado

References: MaePrestamos (FolioDePrestamo, ClaveDeConcepto, ClaveDeNomina, ClaveDeEmpleado)

Column Names: TipoDePago Text(1)

Primary Key: TipoDePago

Column Names: FolioDePrestamo Long Integer NOT NULL

Primary Key: FolioDePrestamo

Column Names: ClaveDeConcepto Integer NOT NULL

Primary Key: ClaveDeConcepto

Column Names: ClaveDeNomina Integer NOT NULL

Primary Key: ClaveDeNomina

Column Names: ClaveDeEmpleado Long Integer NOT NULL
Primary Key: ClaveDeEmpleado

Column Names: NumeroDeQuincena Integer

Column Names: MontoPagado Double

Column Names: InteresPagado Double

54 Entity Name: EmpleadoNomina

Column Names: ClaveDeNomina Integer
Primary Key: ClaveDeNomina

Column Names: ClaveDeEmpleado Long Integer
Primary Key: ClaveDeEmpleado

55 Entity Name: factores

Column Names: ClaveDeProceso Integer
Primary Key: ClaveDeProceso
Foreign Keys: ClaveDeProceso
References: Procesos (ClaveDeProceso)

Column Names: niv_min Double
Primary Key: niv_min

Column Names: niv_max Double
Primary Key: niv_max

Column Names: dias_vac Double

Column Names: porc_primaVac Double

Column Names: pri_vac Double

Column Names: fac_vac Double

Column Names: dias_contrato Double

Column Names: fac_Contrato Double

Column Names: aguinaldo Double

Column Names: fac_agu Double

Column Names: pri_produc Double

Column Names: fac_produc Double

Column Names: Tot_DiasVac Double

Column Names: fac_tot Double

Column Names: NumeroDeQuincena Double

Column Names: VacacionesReales Double

56 Entity Name: FechasDeCorte

Column Names: ClaveDeCalendario Integer NOT NULL

Primary Key: ClaveDeCalendario

Foreign Keys: ClaveDeCalendario

References: Calendarios (ClaveDeCalendario)

Column Names: NumeroDeQuincena Integer NOT NULL

Primary Key: NumeroDeQuincena

Column Names: FechaInicio Date/Time

Column Names: FechaFinal Date/Time

Column Names: EIMes Integer

57 Entity Name: FormasDeCalculo

Column Names: FormaDeCalculo Text(2)

Primary Key: FormaDeCalculo

Column Names: Descripcion Text(50)

58 Entity Name: InsCajaFondo

Column Names: ClaveDeConcepto Integer

Primary Key: ClaveDeConcepto

Foreign Keys: ClaveDeEmpleado

References: Contratacion (ClaveDeEmpleado)

Column Names: ClaveDeNomina Integer NOT NULL

Primary Key: ClaveDeNomina

Foreign Keys: ClaveDeConcepto, ClaveDeNomina

References: PercepcionesDeducciones (ClaveDeConcepto, ClaveDeNomina)

Column Names: ClaveDeEmpleado Long Integer NOT NULL

Primary Key: ClaveDeEmpleado

Column Names: PuntosAcumulados Double

Column Names: AportacionAcumulada Double

Column Names: RetencionP Double

Column Names: PuntosReten Double

Column Names: Estado Text(1)

59 Entity Name: ISR

Column Names: Articulo Text(4)
Primary Key: Articulo
Foreign Keys: ClaveDePeriodo
References: Periodos (ClaveDePeriodo)

Column Names: Fechalnicial Integer
Primary Key: Fechalnicial

Column Names: LimiteInferior Currency
Primary Key: LimiteInferior

Column Names: LimiteSuperior Currency
Primary Key: LimiteSuperior

Column Names: ClaveDePeriodo Integer NOT NULL
Primary Key: ClaveDePeriodo

Column Names: CuotaFija Currency

Column Names: Porcentaje Double

60 Entity Name: MaePrestamos

Column Names: FolioDePrestamo Long Integer
Primary Key: FolioDePrestamo
Foreign Keys: TipoDePago
References: TipoDePagoFondoAhorro (TipoDePago)

Column Names: ClaveDeConcepto Integer NOT NULL
Primary Key: ClaveDeConcepto
Foreign Keys: ClaveDeConcepto, ClaveDeNomina, ClaveDeEmpleado
References: InsCajaFondo (ClaveDeConcepto, ClaveDeNomina, ClaveDeEmpleado)

Column Names: ClaveDeNomina Integer NOT NULL
Primary Key: ClaveDeNomina

Column Names: ClaveDeEmpleado Long Integer NOT NULL
Primary Key: ClaveDeEmpleado

Column Names: TipoDePago Text(50)

Column Names: FechaDePrestamo Date/Time

Column Names: Estado Text(1)

Column Names: NumeroDeCheque Text(12)

Column Names: NumeroDePrestamo Long Integer

Column Names: NumeroDePagos Integer

Column Names: PeriodoDePago Integer

Column Names: Per_Ini Integer

Column Names: Per_Fin Integer

Column Names: TipoDeInteres Text(10)

Column Names: NumeroDeTasa Integer

Column Names: MontoDePrestamo Double

Column Names: SaldoDePrestamo Double

Column Names: Descuento Double

61 Entity Name: MoviNomina

Column Names: NumeroDeQuincena Integer

Primary Key: NumeroDeQuincena

Foreign Keys: ClaveDeNomina, ClaveDeEmpleado

References: EmpleadoNomina (ClaveDeNomina, ClaveDeEmpleado)

Column Names: ClaveDeConcepto Integer NOT NULL

Primary Key: ClaveDeConcepto

Foreign Keys: ClaveDeConcepto, ClaveDeNomina

References: PercepcionesDeducciones (ClaveDeConcepto, ClaveDeNomina)

Column Names: ClaveDeNomina Integer NOT NULL

Primary Key: ClaveDeNomina

Column Names: ClaveDeEmpleado Long Integer NOT NULL

Primary Key: ClaveDeEmpleado

Column Names: VariableParaPercepYDed Double

Column Names: MontoMovi Currency

Column Names: FechaIncidencia Date/Time

62 Entity Name: Nomina

Column Names: ClaveDeNomina Integer

Primary Key: ClaveDeNomina

Foreign Keys: ClaveDeCalendario

References: Calendarios (ClaveDeCalendario)

Column Names: ClaveDeCalendario Integer

Foreign Keys: ClaveDePeriodo

References: Periodos (ClaveDePeriodo)

Column Names: ClaveDePeriodo Integer

Column Names: NombreDeNomina Text(50)

Column Names: Dias Integer

Column Names: Tipo Text(1)

Column Names: Activa Yes/No

Column Names: TablaISR Double

Column Names: FechaCorte Integer

Column Names: Quincenas Integer

Column Names: NominaMaestro Integer

Column Names: Confidencial Yes/No

63 Entity Name: Parametros

Column Names: ClaveDeParametro Integer
Primary Key: ClaveDeParametro
Foreign Keys: ClaveDeParametro
References: CatParametros (ClaveDeParametro)

Column Names: ClaveDeSucursal Integer
Primary Key: ClaveDeSucursal

Column Names: NumeroDeQuincena Integer
Primary Key: NumeroDeQuincena

Column Names: MontoParametro Double

Column Names: FechaIniP Date/Time

Column Names: FechaFinP Date/Time

64 Entity Name: PercepcionesDeducciones

Column Names: ClaveDeConcepto Integer
Primary Key: ClaveDeConcepto
Foreign Keys: PerDed
References: TiposDeConceptos (PerDed)

Column Names: ClaveDeNomina Integer NOT NULL
Primary Key: ClaveDeNomina
Foreign Keys: FormaDeCalculo
References: FormasDeCalculo (FormaDeCalculo)

Column Names: PerDed Text(15)
Foreign Keys: ClaveDeNomina
References: Nomina (ClaveDeNomina)

Column Names: NombreDeConcepto Text(50)

Column Names: NombreCorto Text(15)

Column Names: DescripcionConcepto Memo

Column Names: sta_tus Yes/No
Column Names: TipoDeConcepto Yes/No
Column Names: TipoDeConceptoM Yes/No
Column Names: FormaDeCalculo Text(2)
Column Names: AceptaNegativos Yes/No
Column Names: FormulaGenMovi Text(100)
Column Names: RequiereTope Yes/No
Column Names: FormulaTope Text(100)
Column Names: Secuencia Integer
Column Names: VariableComoFalta Yes/No
Column Names: VariableComoIncapa Yes/No
Column Names: VariableComoDiaLab Yes/No
Column Names: Redondeo Yes/No
Column Names: Truncamiento Yes/No
Column Names: GravalImpuestoEstatal Yes/No
Column Names: Prioridad Byte

65 Entity Name: Periodos

Column Names: ClaveDePeriodo Integer NOT NULL
Primary Key: ClaveDePeriodo
Column Names: NombreDePeriodo Text(18)

66 Entity Name: ProcesoNomina

Column Names: ClaveDeProceso Integer NOT NULL
Primary Key: ClaveDeProceso
Foreign Keys: ClaveDeNomina
References: Nomina (ClaveDeNomina)
Column Names: ClaveDeNomina Integer NOT NULL
Primary Key: ClaveDeNomina
Foreign Keys: ClaveDeProceso
References: Procesos (ClaveDeProceso)

5.1.5 Diagrama Entidad Relación (DER).

En la siguiente sección se muestra el diagrama entidad relación del sistema, en el punto 2.2.1 se explica la notación utilizada. A continuación se dará una breve explicación de dicho diagrama.

DER	Descripción
Los diagramas 1/13-2/13-3/13-5/13 representan al módulo de administración de personal	
1/13	Información relativa a la ubicación del personal dentro de la empresa (Departamento, Puesto o Categoría, Sueldo, Sucursal, Nómina).
2/13	Información relativa a los integrados, Contratos, Fecha de Transferencia, de los procesos disponibles por persona y de la Cuenta de Deudor Diverso.
3/13	Información relativa a los datos generales del empleado o trabajador (Nombre, RFC, IMSS, Fecha de Nacimiento, Escolaridad, Datos de Familiares, etc.) y sobre los datos de la Compañía.
5/13	Información relativa a Vacaciones, Turnos Incapacidades, y Finiquitos
Los diagramas 4/13-6/13-7/13-8/13-9/13 representan al módulo de administración de nómina	
4/13	Información correspondiente a Administración de Ahorro, Parámetros y Cambios Temporales de Categoría.
6/13	Información correspondiente al control de asistencia, Faltas, Permisos, Tiempo Extra etc.
7/13	Información correspondiente a la configuración de nóminas, factores de integración y tablas de ISR
8/13	Información correspondiente a la parametrización de los conceptos de nómina.
8/13	Información de Movimientos de las nóminas, Acumulados, Descuentos programados, Adicionales y SubSidios
Los diagramas 12/13 y 13/13 representan al módulo Servicio Médico	
10/13	Información relativa a Cursos – Puesto – Finalidades y asignación de Grupos
11/13	Información relativa a las evaluaciones al desempeño
Los diagramas 12/13 y 13/13 representan al módulo Servicio Médico	

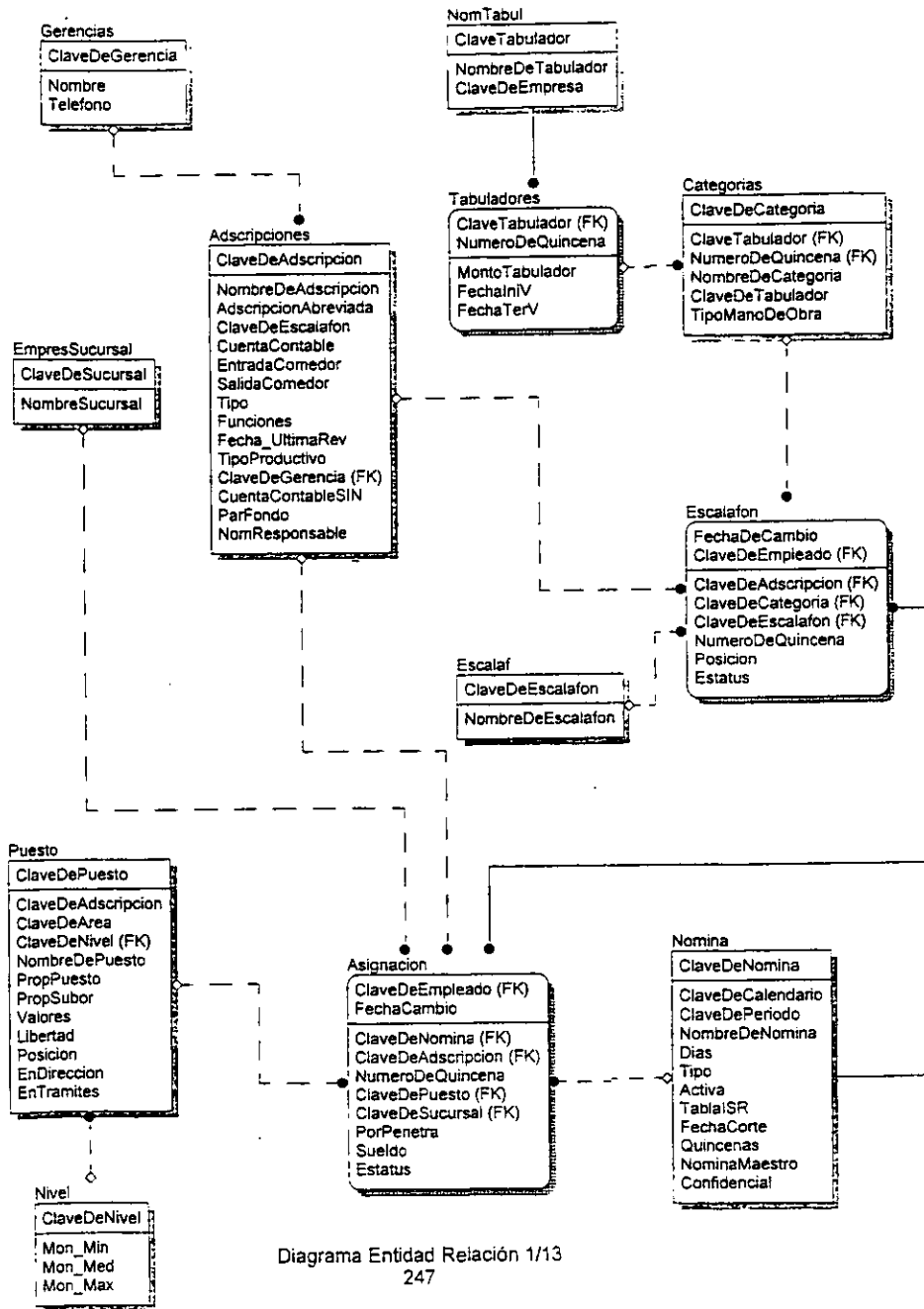


Diagrama Entidad Relación 1/13
 247

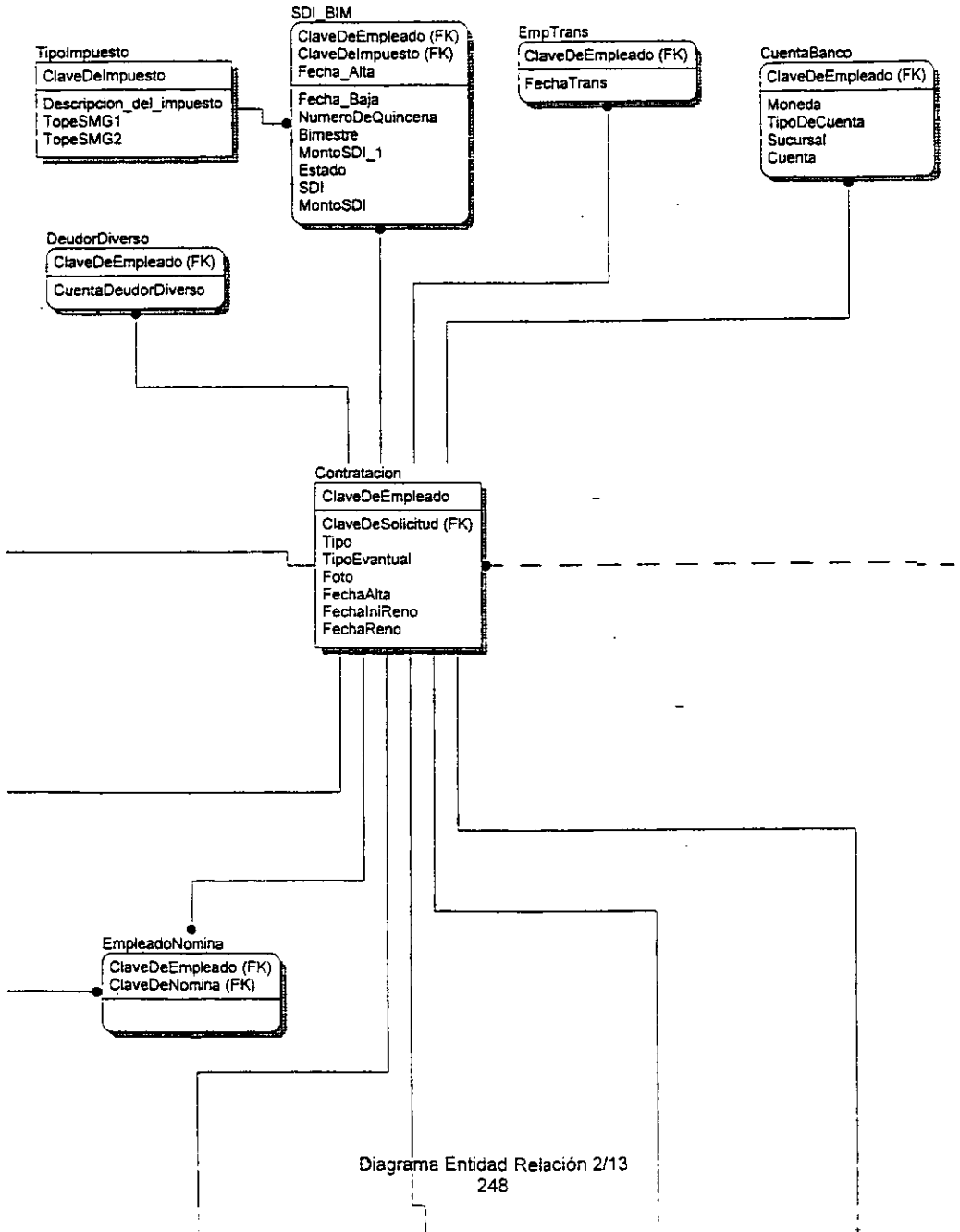


Diagrama Entidad Relación 2/13
248

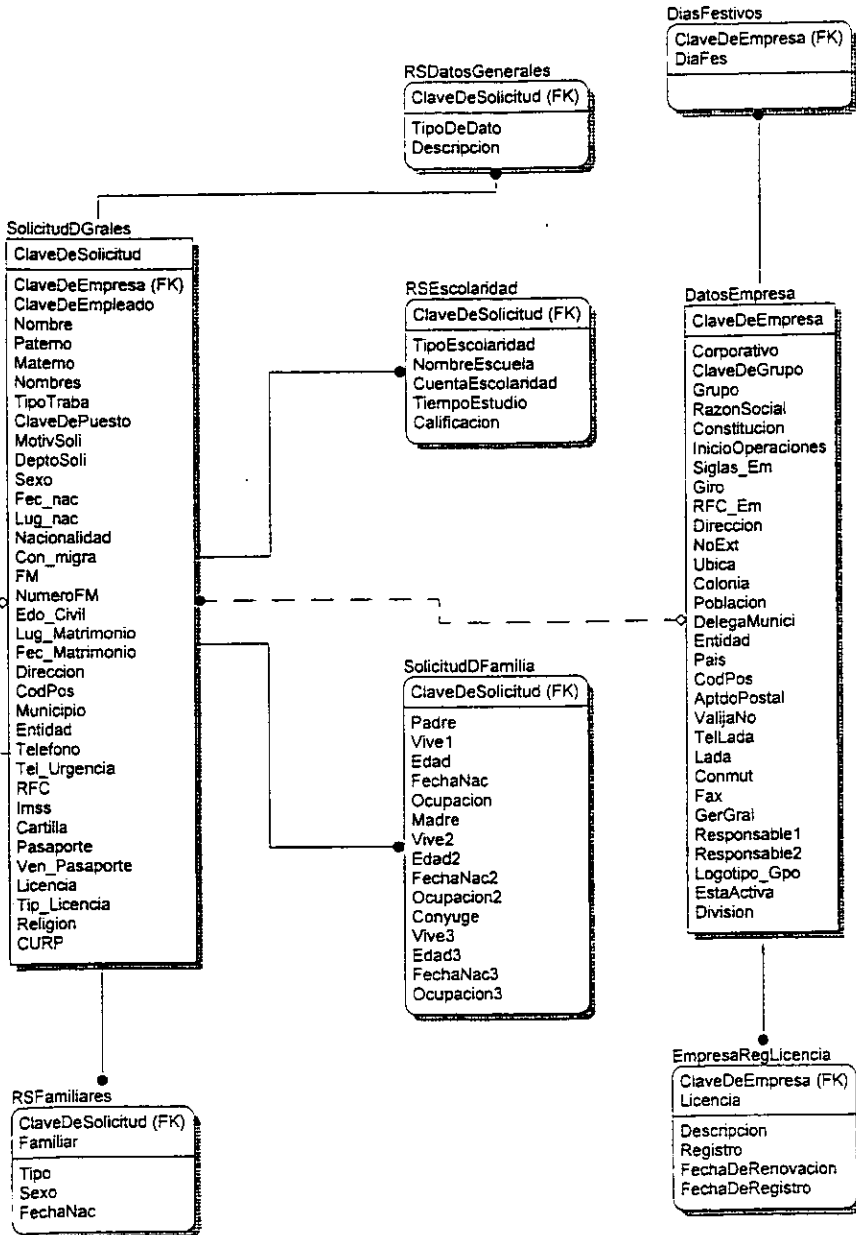


Diagrama Entidad Relación 3/13
249

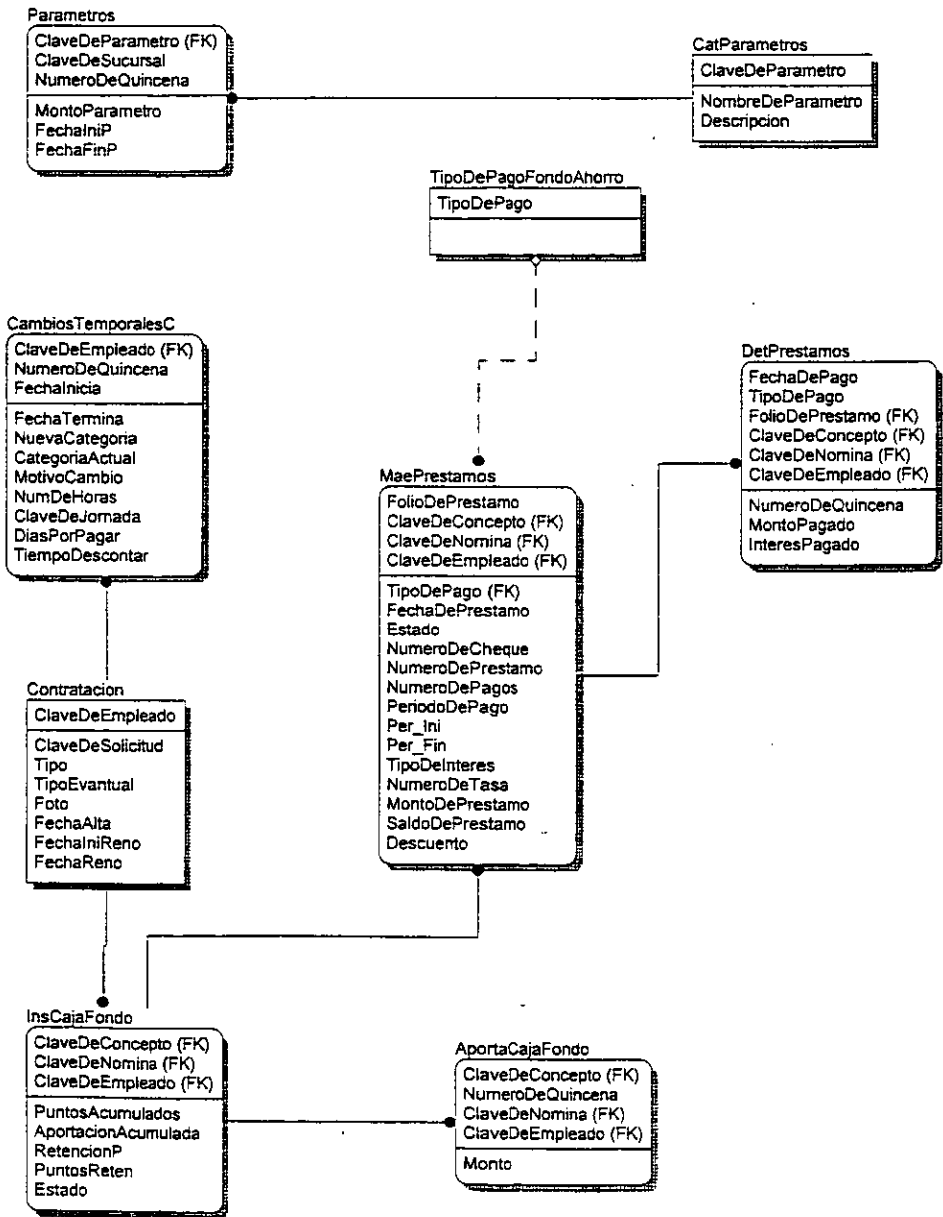


Diagrama Entidad Relación 4/13
250

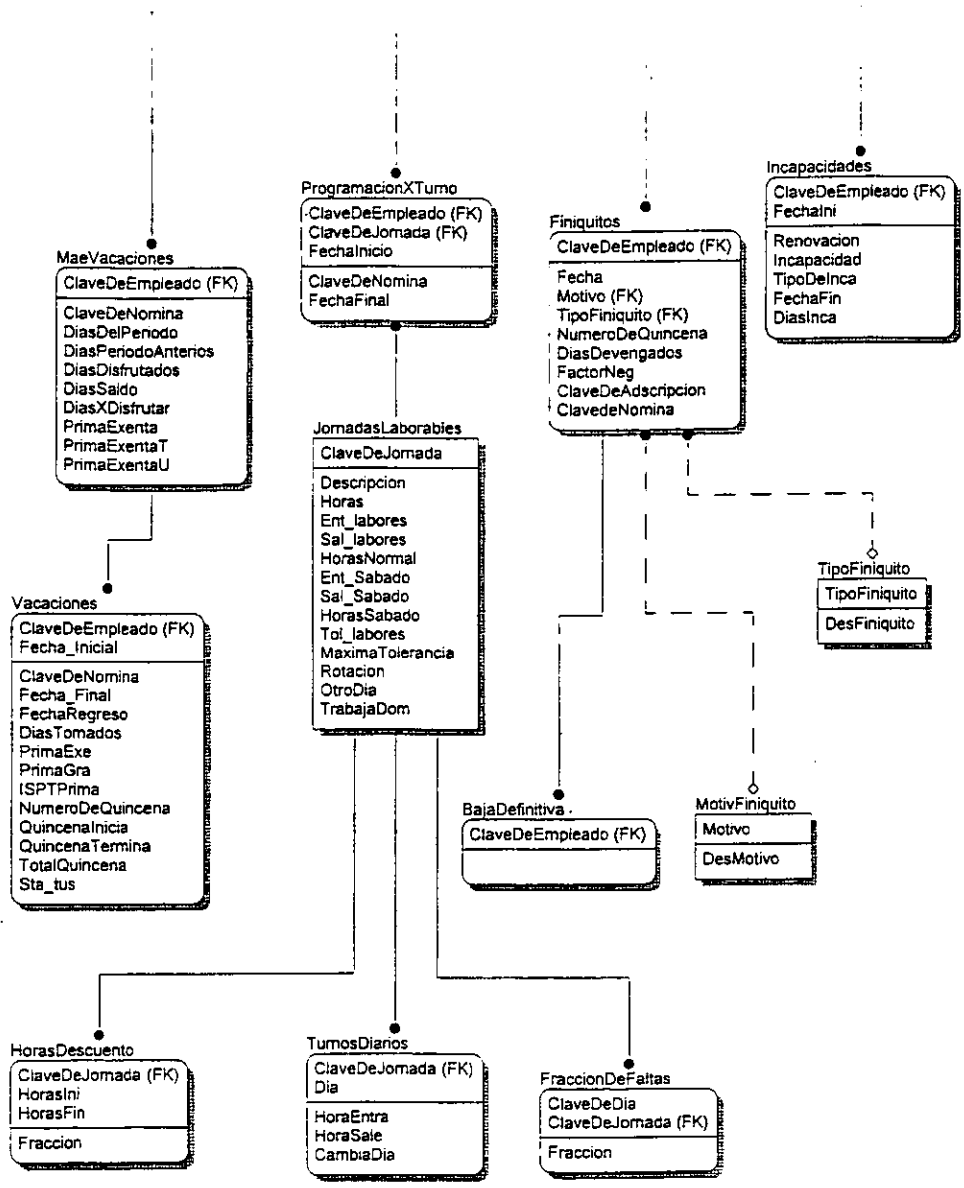


Diagrama Entidad Relación 5/13

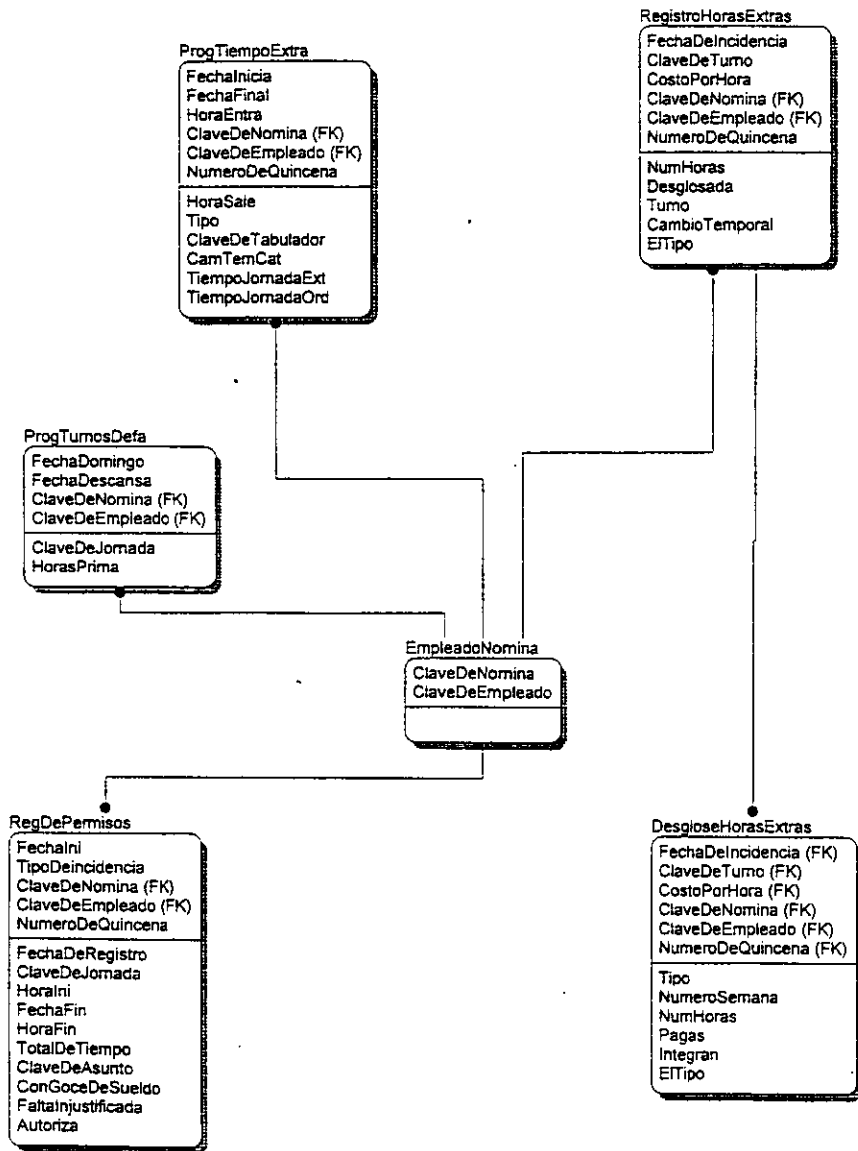


Diagrama Entidad Relación 6/13

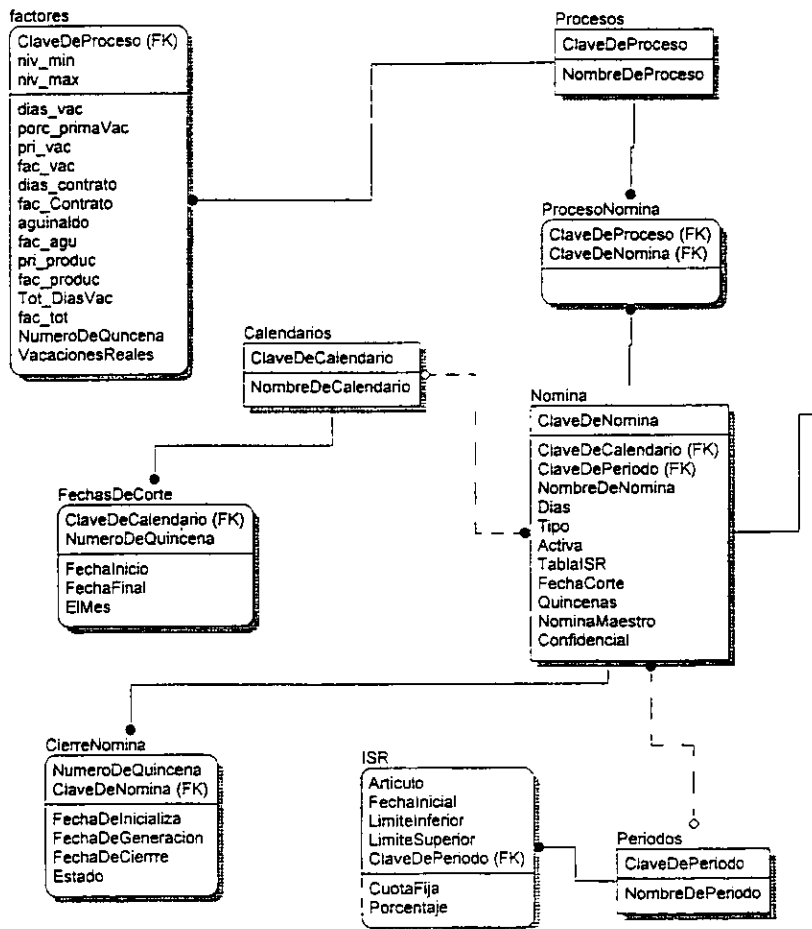


Diagrama Entidad Relación 7/13

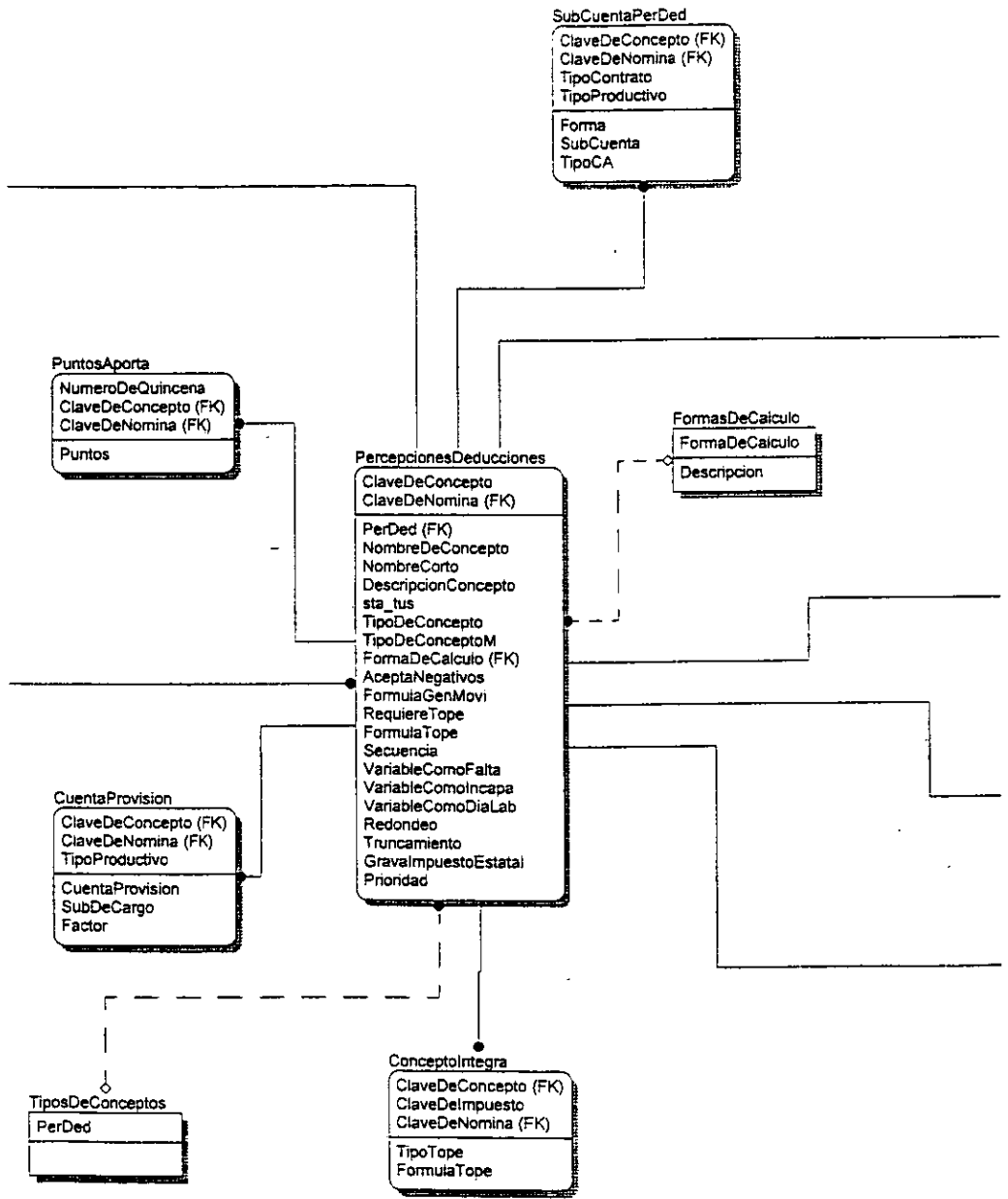
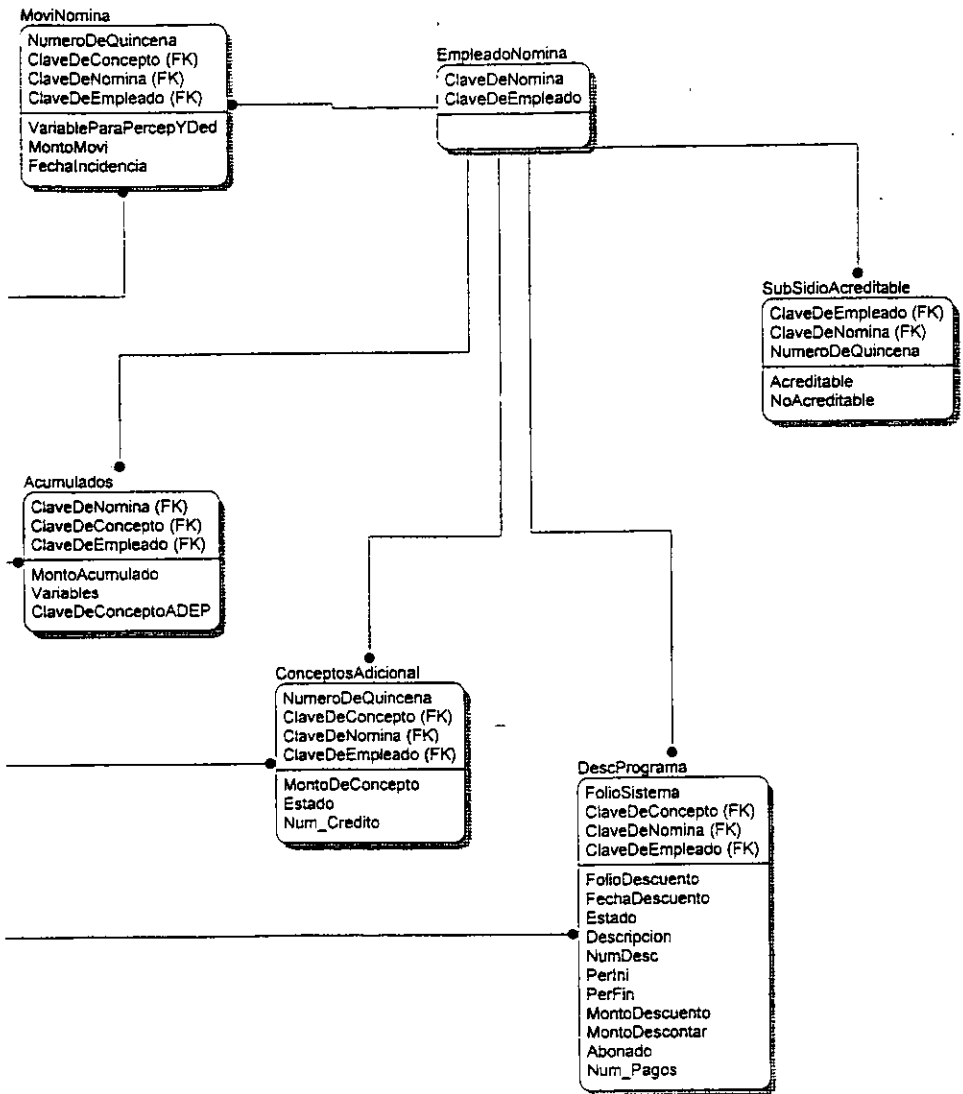


Diagrama Entidad Relación 8/13



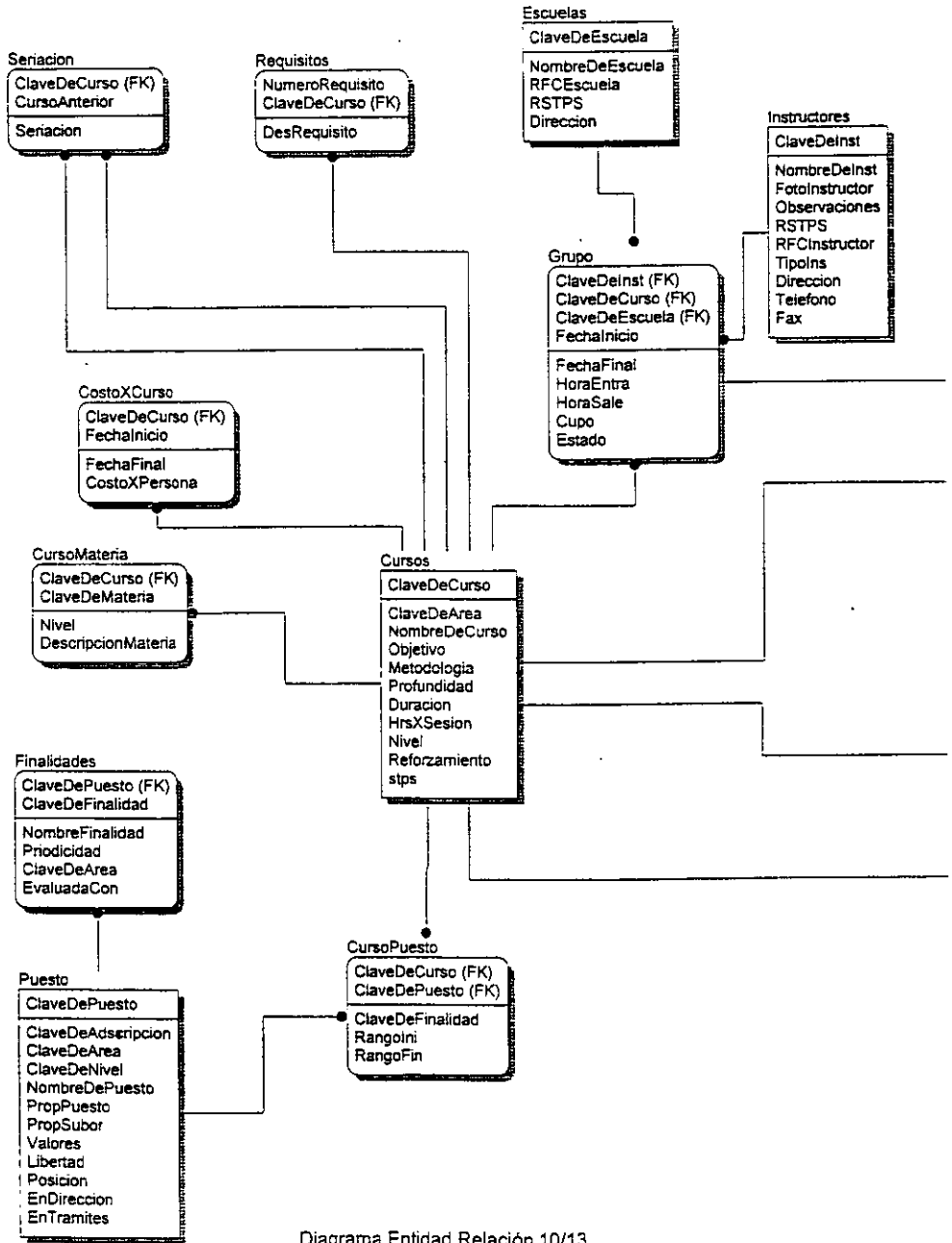


Diagrama Entidad Relación 10/13

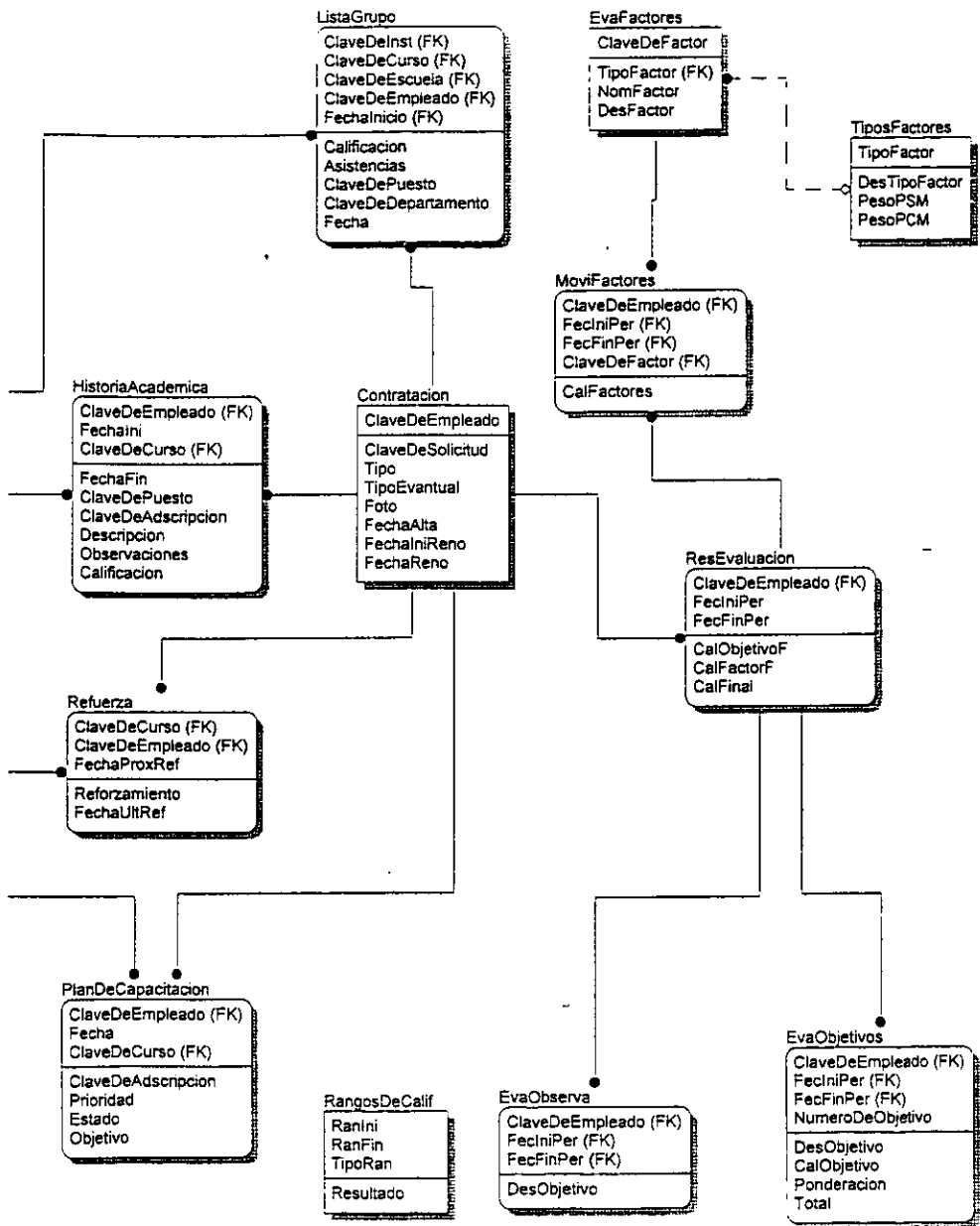


Diagrama Entidad Relación 11/13

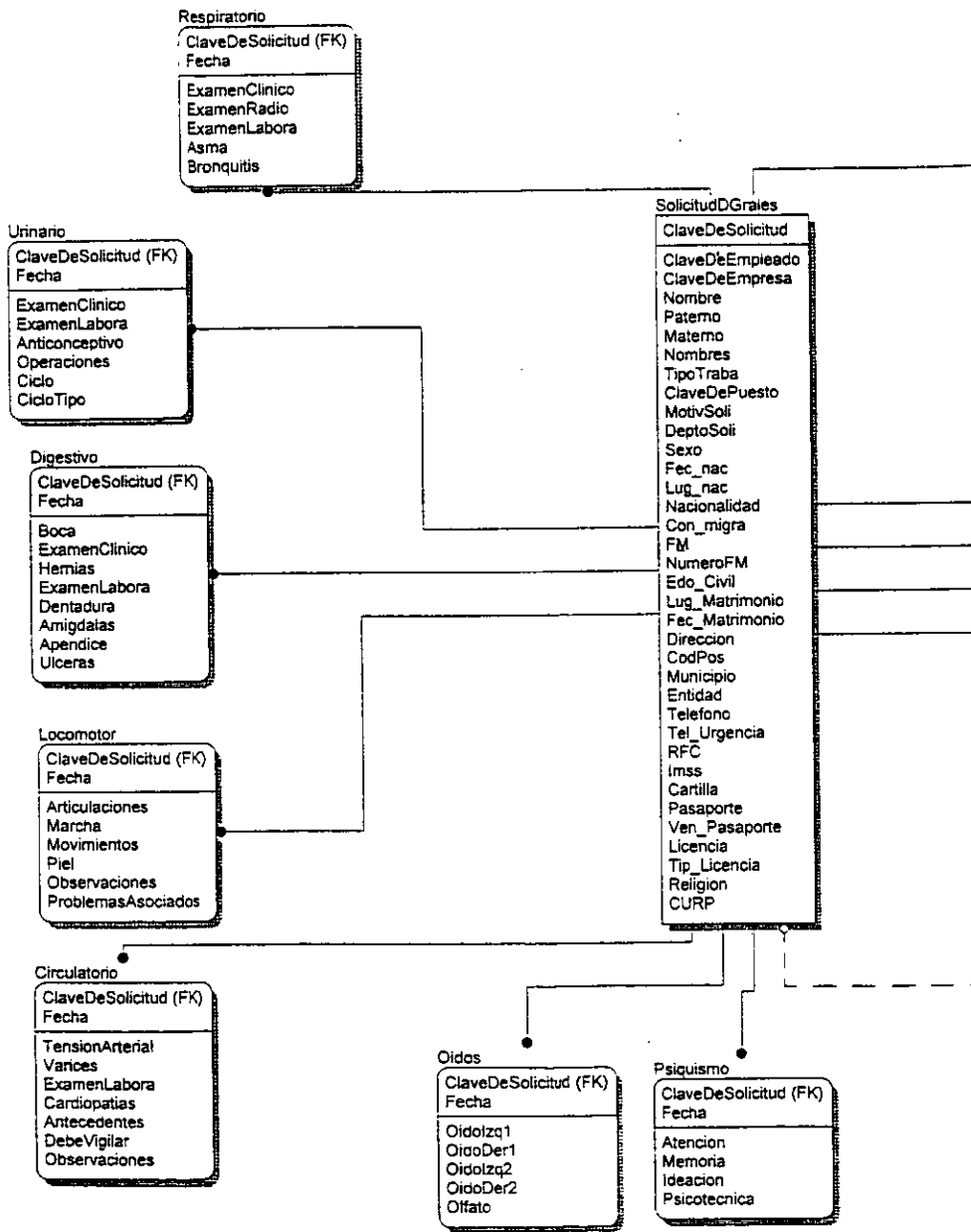
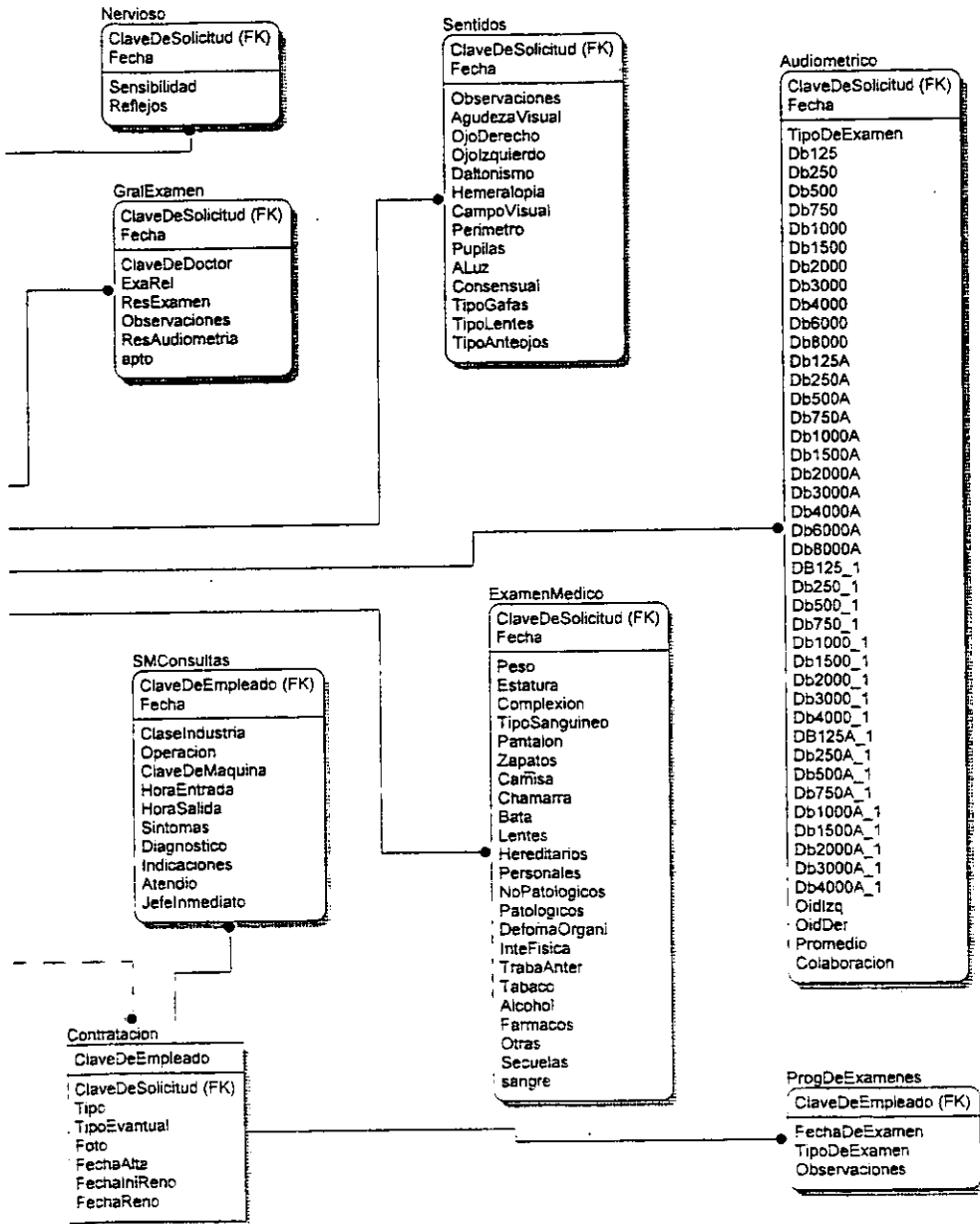
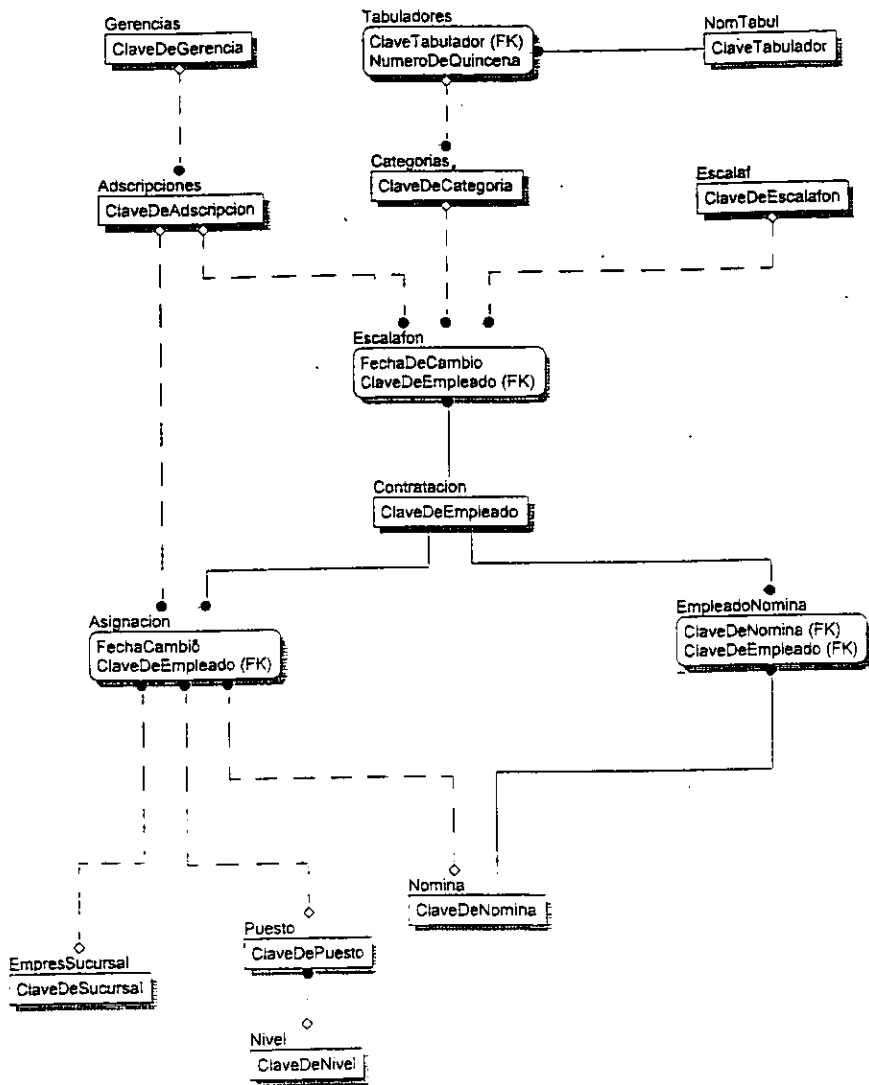


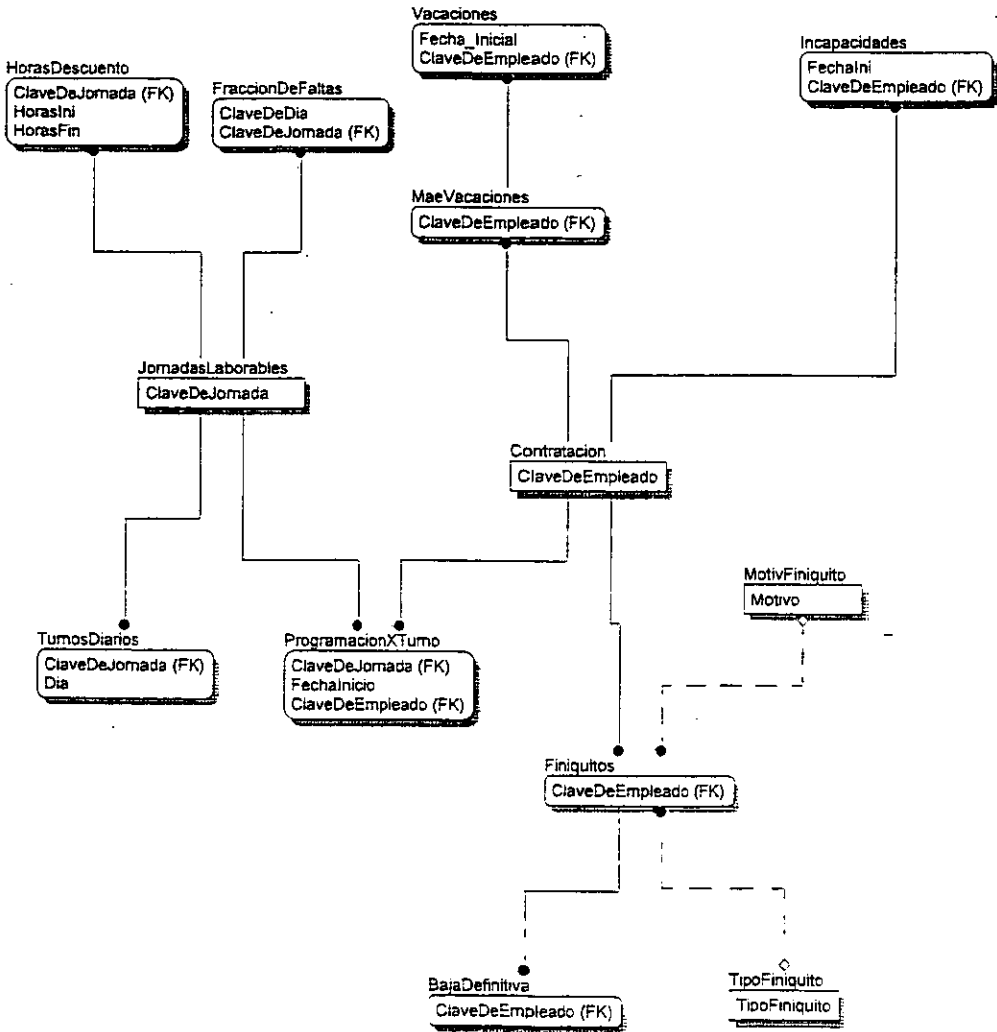
Diagrama Entidad Relación 12/13
258

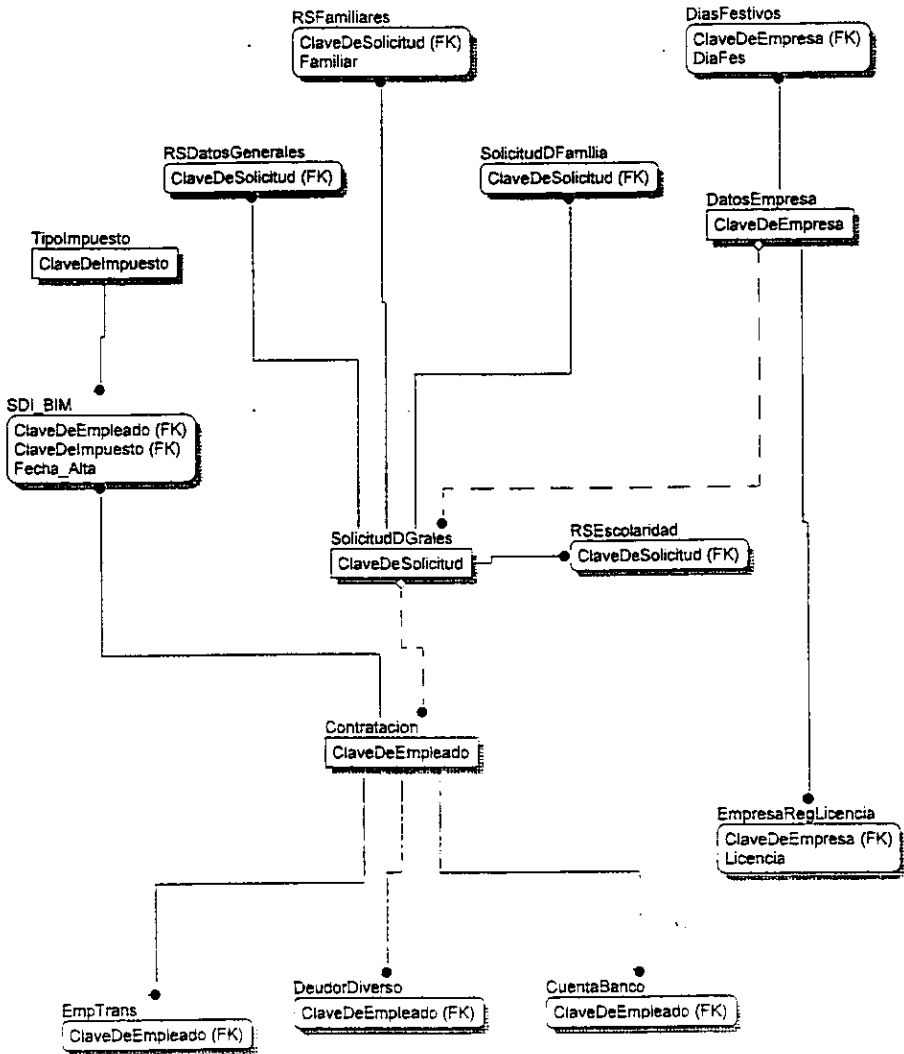


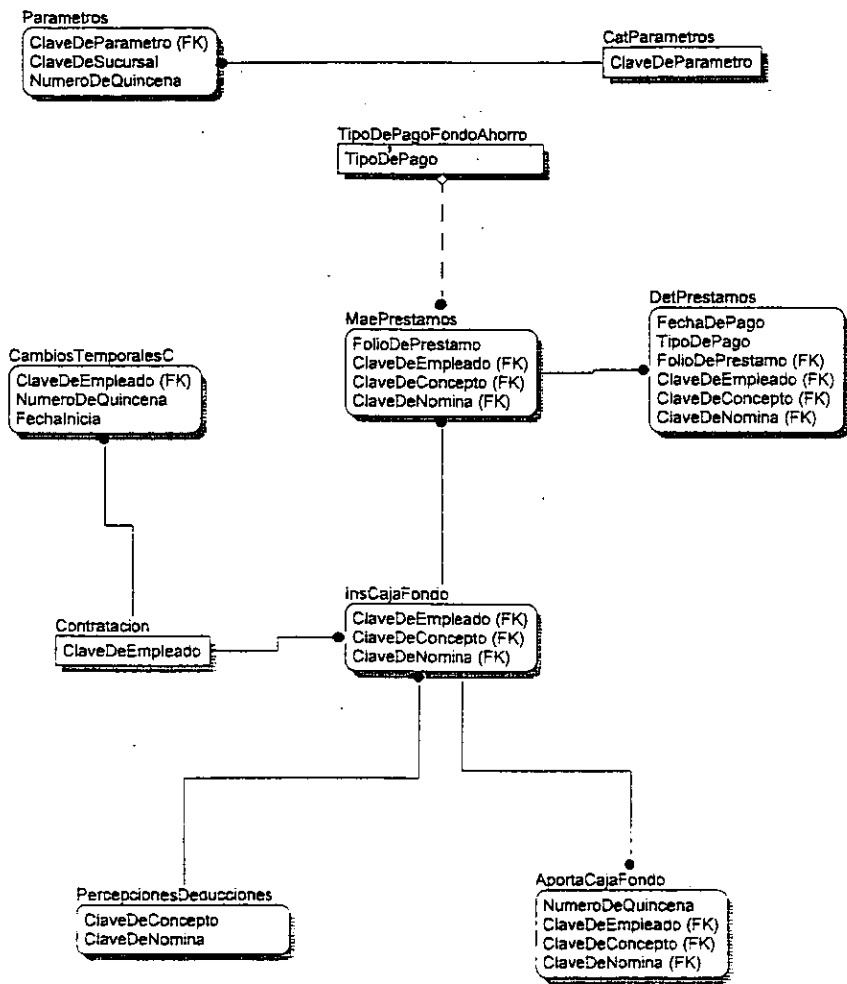
5.1.6 Normalización.

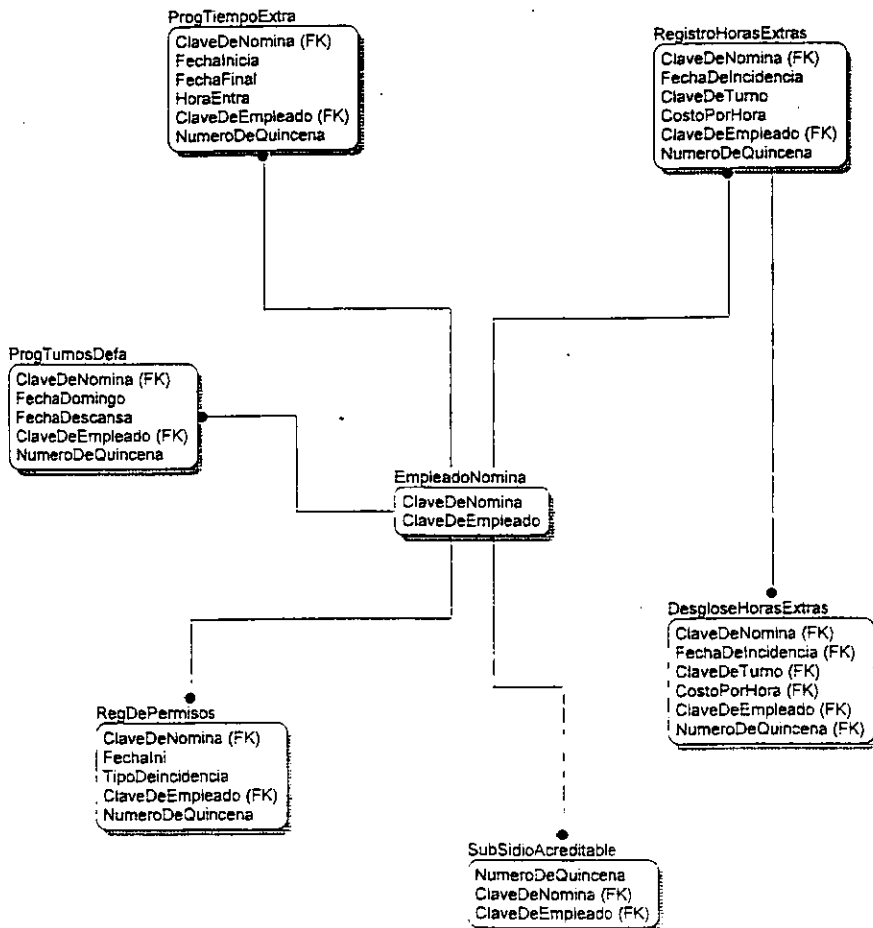
Los diagramas entidad relación de la sección anterior ya fueron normalizados conforme a lo descrito en el punto 2.2.4, a continuación se muestran los diagramas donde se muestran las llaves primarias, los cuales con ayuda de los DER sirven para verificar la normalización.

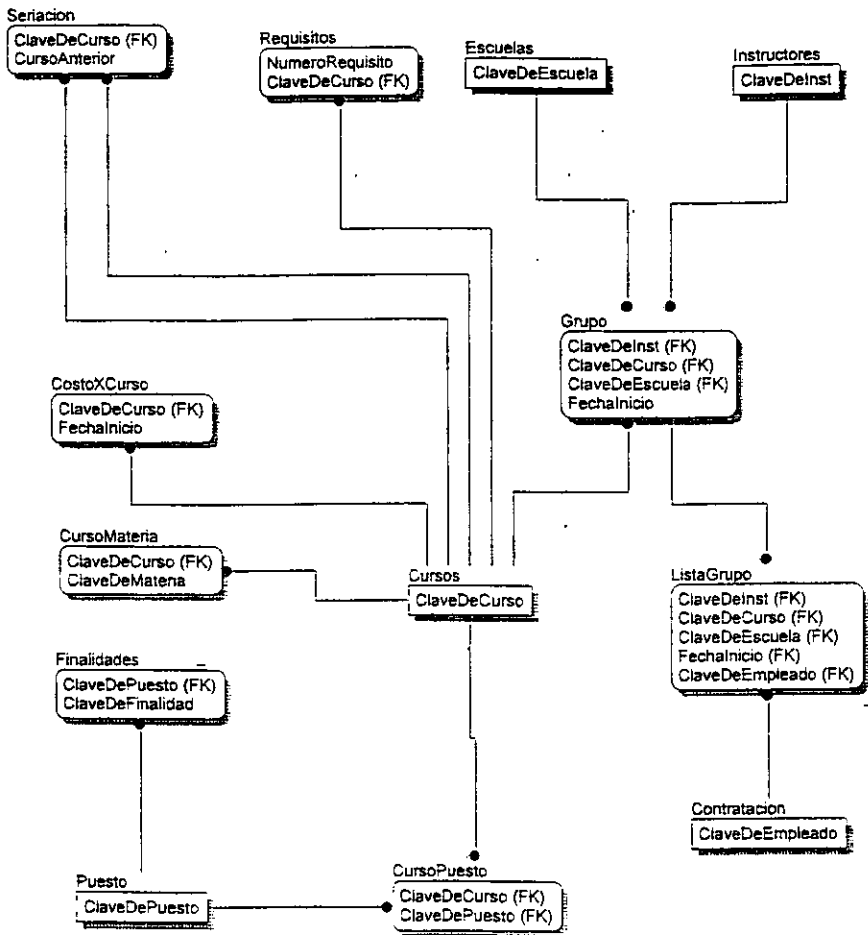


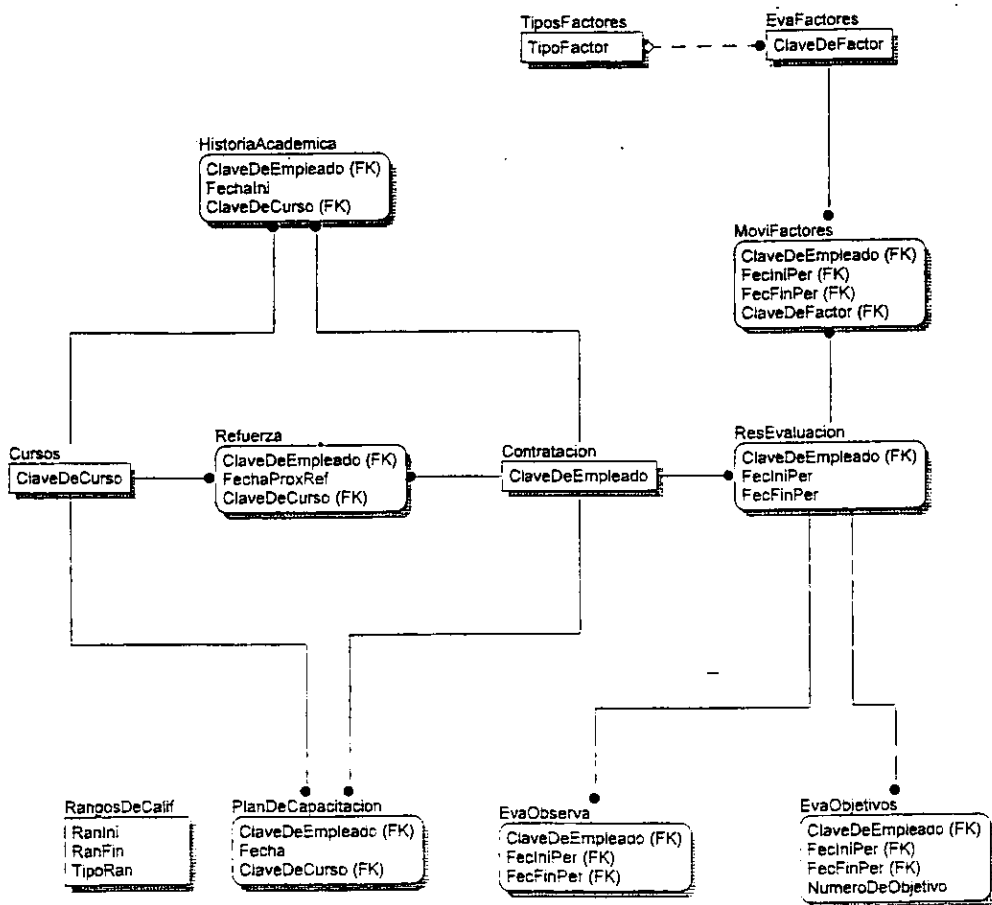


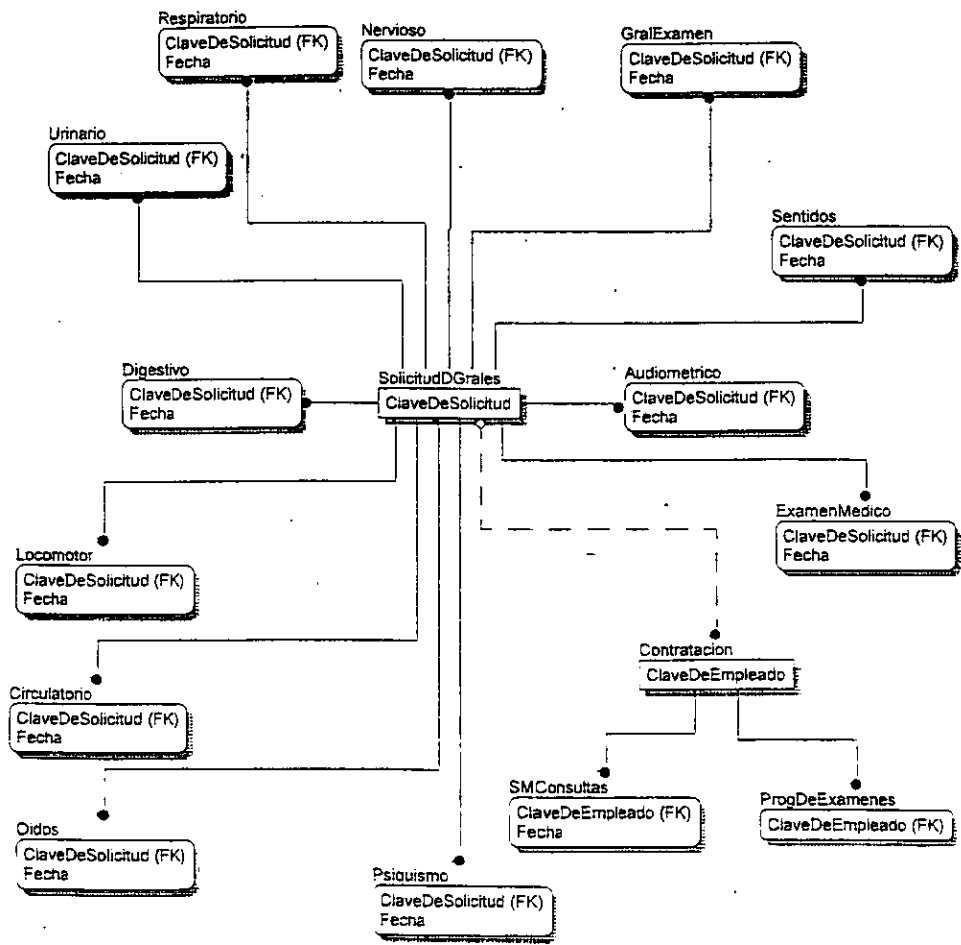












5.2 Generación de código para el procesamiento de la información.

En esta parte se muestra un listado de los módulos incluidos en sistema en la figura 5.2-1 se muestra el índice de los módulos.

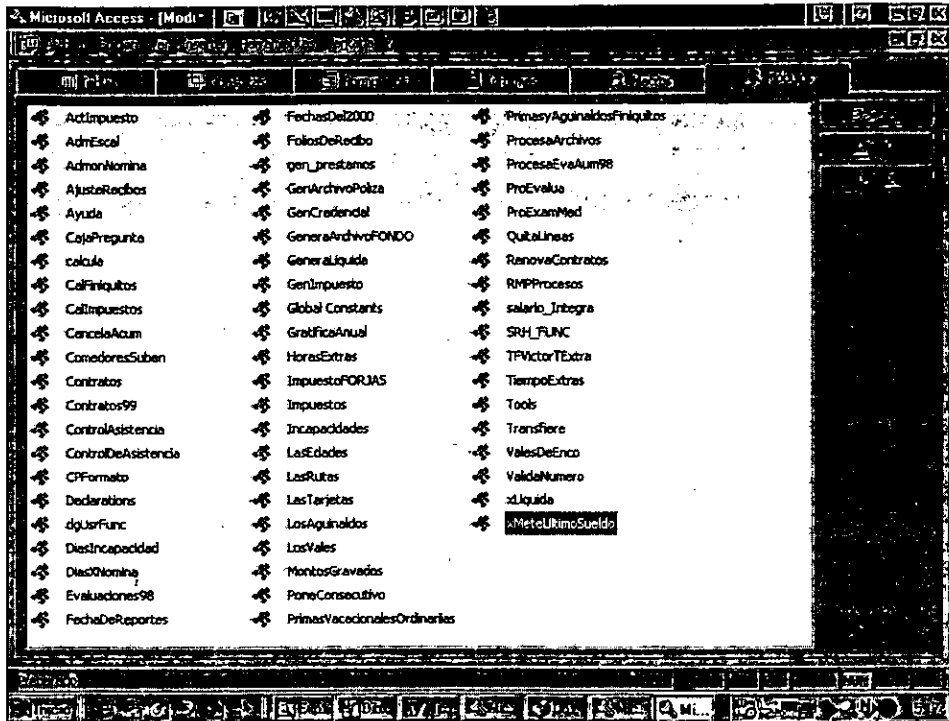


Figura 5.2- 1 Índice de Código Fuente

1. Option Compare Database
2. Option Explicit

Function ActualizaDesDelSPT(NumPeri As Integer, TipoNomi As Integer) As Integer

3. Dim cadena As String
 4. Dim TempSQL As QueryDef
 5. Dim DBActiva As DATABASE

 6. Set DBActiva = DBEngine.Workspaces(0).Databases(0)

 7. Set TempSQL = DBActiva.CreateQueryDef("")
 8. cadena = " Delete * from DesDelSPT "
 9. TempSQL.SQL = cadena
 10. TempSQL.Execute

 11. cadena = "INSERT INTO DesDelSPT (ClaveDeEmpleado, MontoMovi)"
 12. cadena = cadena & " SELECT MoviNomina.ClaveDeEmpleado, MoviNomina.MontoMovi"
 13. cadena = cadena & " FROM MoviNomina"
 14. cadena = cadena & " WHERE (((MoviNomina.NumeroDeQuincena)= " & NumPeri & ") AND
 ((MoviNomina.NumeroDeNómina)= " & TipoNomi & ") AND ((MoviNomina.ClaveDeConcepto)=345));"
 15. TempSQL.SQL = cadena
 16. TempSQL.Execute

 17. cadena = " UPDATE ConceptosAdicional INNER JOIN DesDelSPT ON
 ConceptosAdicional.ClaveDeEmpleado = DesDelSPT.ClaveDeEmpleado SET
 ConceptosAdicional.MontoDeConcepto = [MontoDeConcepto]+[MontoMovi]"
 18. cadena = cadena & " WHERE (((ConceptosAdicional.ClaveDeConcepto)=6028));"
 19. TempSQL.SQL = cadena
 20. TempSQL.Execute

 21. End Function
-
22. Option Compare Database 'Usar orden de base de datos en comparaciones de cadenas
 - 23.

Public Function CambiaLetras(CadenaDeNombre As Variant) As String

24. If IsNull(CadenaDeNombre) Then
25. CadenaDeNombre = ""
26. End If
27. tamaño = Len(CadenaDeNombre)
28. For i = 1 To tamaño
29. letra = Mid\$(CadenaDeNombre, i, 1)
30. If letra = "Ñ" Or letra = "ñ" Then
31. Mid\$(CadenaDeNombre, i, 1) = "N"
32. Else
33. Select Case Mid\$(CadenaDeNombre, i, 1)
34. Case "Á": Mid\$(CadenaDeNombre, i, 1) = "A"


```

35.         Case "É": Mid$(CadenaDeNombre, i, 1) = "E"
36.         Case "Í": Mid$(CadenaDeNombre, i, 1) = "I"
37.         Case "Ó": Mid$(CadenaDeNombre, i, 1) = "O"
38.         Case "Ú": Mid$(CadenaDeNombre, i, 1) = "U"
39.         Case ",": Mid$(CadenaDeNombre, i, 1) = ""
40.         Case " ": Mid$(CadenaDeNombre, i, 1) = " "
41.
42.     End Select
43.
44. End If
45. Next i
46.
47. CambiaLetras = CadenaDeNombre
48.
49.
50.
51.
52. End Function

```

Public Function SUACambiaLetras(CadenaDeNombre As Variant) As String

```

53. If IsNull(CadenaDeNombre) Then
54.     CadenaDeNombre = ""
55. End If
56. tamaño = Len(CadenaDeNombre)
57. For i = 1 To tamaño
58.     letra = Mid$(CadenaDeNombre, i, 1)
59.     If letra = "Ñ" Or letra = "ñ" Then
60.         Mid$(CadenaDeNombre, i, 1) = "&"
61.     Else
62.         Select Case Mid$(CadenaDeNombre, i, 1)
63.             Case "Á": Mid$(CadenaDeNombre, i, 1) = "A"
64.             Case "É": Mid$(CadenaDeNombre, i, 1) = "E"
65.             Case "Í": Mid$(CadenaDeNombre, i, 1) = "I"
66.             Case "Ó": Mid$(CadenaDeNombre, i, 1) = "O"
67.             Case "Ú": Mid$(CadenaDeNombre, i, 1) = "U"
68.             Case ",": Mid$(CadenaDeNombre, i, 1) = ""
69.             Case " ": Mid$(CadenaDeNombre, i, 1) = " "
70.
71.         End Select
72.
73.     End If
74. Next i
75.
76. SUACambiaLetras = CadenaDeNombre
77.
78.
79.
80.
81. End Function
82.
83. Function mueveEscal(nue_esca As Integer, nue_cat As Integer,  

ant_esca As Integer, ant_cat As Integer, m_pos As Integer) As Integer
84. Dim DBActiva As DATABASE

```

```

85. Dim EvaluaSQL As QueryDef
86. Dim EvaluaReg As Recordset
87. Dim nue_num As Integer
88.
89. Set DBActiva = DBEngine.Workspaces(0).Databases(0)
90.   Stop
91.
92.   Set EvaluaSQL = DBActiva.QueryDefs("BusMaxEscal")
93.   EvaluaSQL.Parameters("nue_esca") = nue_esca
94.   EvaluaSQL.Parameters("nue_cat") = nue_cat
95.   Set EvaluaReg = EvaluaSQL.OpenRecordset()
96.   If EvaluaReg.EOF = -1 Then
97.     nue_num = 1
98.   Else
99.     nue_num = EvaluaReg![MaxPos] + 1
100.   End If
101.
102.   EvaluaSQL.Close
103.   If mi_pos > 0 Then
104.     Set EvaluaSQL = DBActiva.QueryDefs("ActEscal")
105.     EvaluaSQL.Parameters("ant_esca") = ant_esca
106.     EvaluaSQL.Parameters("ant_cat") = ant_cat
107.     EvaluaSQL.Parameters("mi_pos") = mi_pos
108.     EvaluaSQL.Execute
109.   End If
110.     mueveEscal = nue_num
111.
112. End Function
113. Option Compare Database 'Usar orden de base de datos en comparaciones de cadenas
114.
115.

```

Function AcumulaConceptos(xQuincena As Integer, xNomina As Integer) As Integer

```

116.
117. Dim DBActiva As DATABASE
118. Dim EvaluaSQL As QueryDef
119. Dim TempSQL As QueryDef
120.
121.
122. Set DBActiva = DBEngine.Workspaces(0).Databases(0)
123.
124.   Set TempSQL = DBActiva.CreateQueryDef("")
125.   cadena = " Delete from AcumuladosT "
126.   TempSQL.SQL = cadena
127.   TempSQL.Execute
128.
129.
130.   Set EvaluaSQL = DBActiva.QueryDefs("MeteAcumulados")
131.   EvaluaSQL.Parameters("xQuincena") = xQuincena
132.   EvaluaSQL.Parameters("xNomina") = xNomina
133.   EvaluaSQL.Execute
134.
135.   Set TempSQL = DBActiva.CreateQueryDef("")
136.   cadena = " Delete from Acumulados Where ClaveDeNomina = " & xNomina

```

```

137.     TempSQL.SQL = cadena
138.     TempSQL.Execute
139.
140.     Set TempSQL = DBActiva.CreateQueryDef("")
141.     cadena = " insert into acumulados select * from AcumuladosT "
142.     TempSQL.SQL = cadena
143.     TempSQL.Execute
144.
145.     AcumulaConceptos = 1
146.
147.
148. End Function
149.

```

Sub yCierreNomina(TipoNomi As Integer, NumPeri As Integer, Operacion As String, zEmpresa As Integer)

```

150. Dim DBActiva As DATABASE
151. Dim mov_act As Recordset
152. Dim TempSQL As QueryDef
153. Dim EvaluaSQL As QueryDef
154. Dim EvaluaReg As Recordset
155.
156. Set DBActiva = DBEngine.Workspaces(0).Databases(0)
157.
158. ' Verificar si ya ha sido inicializada o no
159.
160. x_estado = DLookup("Estado", "CierreNomina", "TipoDeNomina = " & TipoNomi & " and
NumeroDeQuincena = " & NumPeri)
161.
162. DoCmd.Hourglass True
163. If IsNull(x_estado) Then
164.     x = prueba("Admon. Nómina", " Se procede a Inicializar Nómina ")
165.     If x = 1 Then
166.         Set TempSQL = DBActiva.QueryDefs("InicializaLaNomina")
167.         TempSQL.Parameters("NumPeri") = NumPeri
168.         TempSQL.Parameters("TipoNomi") = TipoNomi
169.         TempSQL.Parameters("xEmpresa") = zEmpresa
170.
171.         TempSQL.Execute
172.         TempSQL.Close
173.
174.         Set TempSQL = DBActiva.QueryDefs("InsertaDiasAdicionales")
175.         TempSQL.Parameters("NumPeri") = NumPeri
176.         TempSQL.Parameters("TipoNomi") = TipoNomi
177.
178.         TempSQL.Execute
179.         TempSQL.Close
180.
181.
182.         Set TempSQL = DBActiva.QueryDefs("EstadoIncializa")
183.         TempSQL.Parameters("NumPeri") = NumPeri
184.         TempSQL.Parameters("TipoNomi") = TipoNomi
185.         TempSQL.Execute
186.         TempSQL.Close
187.         'MsgBox " -Horacio Acuérdate de activar la integración "

```

```

188.         'z = GeneralIntegraPromo(NumPeri, TipoNomi)
189.         MsgBox "Genere el Proceso "
190.     End If
191.
192.
193. Else
194.     Select Case (x_estado)
195.
196.     Case "I": ' Nomina inicializada
197.         Select Case (Operacion)
198.
199.             Case "C": MsgBox " No Se Puede Cerrar Nómina "
200.             Case "G": x = prueba("Admon. Nómina", " Se procede a Generar Nómina ")
201.                 If x = 1 Then
202.                     Call gen_nom(NumPeri, TipoNomi, zEmpresa)
203.                 End If
204.             Case "I": x = prueba("Admon. Nómina", " Se procede a ReIncializar Nómina ")
205.
206.                 If x = 1 Then
207.
208.                     Set TempSQL = DBActiva.QueryDefs("BorraLaNomina")
209.                     TempSQL.Parameters("NumPeri") = NumPeri
210.                     TempSQL.Parameters("TipoNomi") = TipoNomi
211.                     TempSQL.Execute
212.                     TempSQL.Close
213.
214.
215.                     Set TempSQL = DBActiva.QueryDefs("InicializaLaNomina")
216.                     TempSQL.Parameters("NumPeri") = NumPeri
217.                     TempSQL.Parameters("TipoNomi") = TipoNomi
218.                     TempSQL.Parameters("xEmpresa") = zEmpresa
219.                     TempSQL.Execute
220.                     TempSQL.Close
221.
222.                     Set TempSQL = DBActiva.QueryDefs("InsertaDiasAdicionales")
223.                     TempSQL.Parameters("NumPeri") = NumPeri
224.                     TempSQL.Parameters("TipoNomi") = TipoNomi
225.
226.                     TempSQL.Execute
227.                     TempSQL.Close
228.
229.
230.                     Set TempSQL = DBActiva.QueryDefs("IniciaLaNomina")
231.                     TempSQL.Parameters("NumPeri") = NumPeri
232.                     TempSQL.Parameters("TipoNomi") = TipoNomi
233.                     TempSQL.Execute
234.                     TempSQL.Close
235.                     MsgBox " -Horacio Acuere date de activar la integraci3n "
236.                     'z = GeneralIntegraPromo(NumPeri, TipoNomi)
237.                 End If
238.
239.             End Select
240.
241.
242.         Case "G": ' Nomina generada
243.             Select Case (Operacion)

```

```

244.
245. Case "C": x = prueba("Admon. Nómina", "Se procede a Cerrar Nómina")
246.     If x = 1 Then
247.
248.         'Programa Adelanto Vacaciones
249.         Set TempSQL = DBActiva.QueryDefs("Programa Adelanto Vacaciones")
250.         TempSQL.Parameters("xQuincena") = NumPeri
251.         TempSQL.Parameters("xNomina") = TipoNomi
252.         TempSQL.Execute
253.         TempSQL.Close
254.
255.         Set TempSQL = DBActiva.QueryDefs("Cierra La Nomina")
256.         TempSQL.Parameters("NumPeri") = NumPeri
257.         TempSQL.Parameters("TipoNomi") = TipoNomi
258.         TempSQL.Execute
259.         TempSQL.Close
260.         ' Actualiza Puntos
261.         Set TempSQL = DBActiva.QueryDefs("Borra Antes")
262.         TempSQL.Parameters("elPeriodo") = NumPeri
263.         TempSQL.Parameters("laNomina") = TipoNomi
264.         TempSQL.Execute
265.         TempSQL.Close
266.
267.         Set TempSQL = DBActiva.QueryDefs("Los Puntos Del Periodo")
268.         TempSQL.Parameters("elPeriodo") = NumPeri
269.         TempSQL.Parameters("laNomina") = TipoNomi
270.         TempSQL.Execute
271.         TempSQL.Close
272.
273.         Set TempSQL = DBActiva.QueryDefs("Actualiza Los Puntos")
274.         TempSQL.Parameters("elPeriodo") = NumPeri
275.         TempSQL.Parameters("laNomina") = TipoNomi
276.         TempSQL.Execute
277.         TempSQL.Close
278.
279.         ' Actualiza los descuentos programados
280.         Set TempSQL = DBActiva.QueryDefs("qActDesProg")
281.         TempSQL.Parameters("num_qui") = NumPeri
282.         TempSQL.Parameters("num_nom") = TipoNomi
283.         TempSQL.Execute
284.         TempSQL.Close
285.
286.         'Actualiza el acumulado gravable
287.         Set TempSQL = DBActiva.QueryDefs("xPercepcionGravada")
288.         TempSQL.Parameters("Periodo") = NumPeri
289.         TempSQL.Parameters("xNomina") = TipoNomi
290.         TempSQL.Execute
291.
292.         Set TempSQL = DBActiva.QueryDefs("xPercepcionExenta")
293.         TempSQL.Parameters("Periodo") = NumPeri
294.         TempSQL.Parameters("xNomina") = TipoNomi
295.         TempSQL.Execute
296.
297.         Set TempSQL = DBActiva.QueryDefs("xPercepcionPercep")
298.         TempSQL.Parameters("Periodo") = NumPeri
299.         TempSQL.Parameters("xNomina") = TipoNomi

```

```

300.      TempSQL.Execute
301.      TempSQL.Close
302.          'Actualiza el impuesto
303.      Set TempSQL = DBActiva.QueryDefs("xMetelSPT")
304.      TempSQL.Parameters("xPeriodo") = NumPeri
305.      TempSQL.Parameters("xNomina") = TipoNomi
306.      TempSQL.Execute
307.      TempSQL.Close
308.      ' Acumula Vales de Despensa
309.      z = ValesEnco(CInt(TipoNomi), CInt(NumPeri))
310.
311.      'Actualiza el Tiempo Extra Transferido
312.      If TipoNomi = 1 Then
313.          Set TempSQL = DBActiva.QueryDefs("TerminaSindTE")
314.          TempSQL.Execute
315.          TempSQL.Close
316.      Else
317.          Set TempSQL = DBActiva.QueryDefs("TerminaEmpITE")
318.          TempSQL.Parameters("xClaveDeNomina") = TipoNomi
319.          TempSQL.Execute
320.          TempSQL.Close
321.      End If
322.
323.      z = AcumulaConceptos(NumPeri, TipoNomi)
324.      z = ActualizaDesiSPT(CInt(NumPeri), CInt(TipoNomi))
325.
326.      Set TempSQL = DBActiva.CreateQueryDef("")
327.      cadena = " Delete from DesgloseIncapacidad where Procesada = -1 and
ClaveDeNomina = " & TipoNomi
328.      TempSQL.SQL = cadena
329.      TempSQL.Execute
330.
331.      If NumPeri Mod 100 = 12 Then
332.          NumPeri = (NumPeri - 12) + 101
333.      Else
334.          NumPeri = NumPeri + 1
335.      End If
336.      'MsgBox " -Horacio Acuérdate de activar la integración "
337.      'z = GeneralIntegraPromo(NumPeri, TipoNomi)
338.
339.
340.      End If
341.
342.
343.      Case "G": x = prueba("Admon. Nómina", "Se procede a Generar Nómina")
344.      If x = 1 Then
345.          Call gen_nom(NumPeri, TipoNomi, zEmpresa)
346.      End If
347.
348.
349.
350.      Case "I": x = prueba("Admon. Nómina", "Se procede a Incialiazar Nómina")
351.      If x = 1 Then
352.
353.          Set TempSQL = DBActiva.QueryDefs("BorraLaNomina")
354.          TempSQL.Parameters("NumPeri") = NumPeri

```

```

355.         TempSQL.Parameters("TipoNomi") = TipoNomi
356.         TempSQL.Execute
357.         TempSQL.Close
358.
359.
360.         Set TempSQL = DBActiva.QueryDefs("InicializaLaNomina")
361.         TempSQL.Parameters("NumPeri") = NumPeri
362.         TempSQL.Parameters("TipoNomi") = TipoNomi
363.         TempSQL.Parameters("xEmpresa") = zEmpresa
364.         TempSQL.Execute
365.         TempSQL.Close
366.
367.         Set TempSQL = DBActiva.QueryDefs("InsertaDiasAdicionales")
368.         TempSQL.Parameters("NumPeri") = NumPeri
369.         TempSQL.Parameters("TipoNomi") = TipoNomi
370.
371.         TempSQL.Execute
372.         TempSQL.Close
373.
374.         Set TempSQL = DBActiva.QueryDefs("IniciaLaNomina")
375.         TempSQL.Parameters("NumPeri") = NumPeri
376.         TempSQL.Parameters("TipoNomi") = TipoNomi
377.         TempSQL.Execute
378.         TempSQL.Close
379.         'MsgBox " -Horacio Acuérdate de activar la integración "
380.         'z = GeneralIntegraPromo(NumPeri, TipoNomi)
381.     End If
382.
383.     End Select
384.
385.     Case "C": MsgBox "Nómina Cerrada No se Puede Hacer Ningun Proceso"
386.
387.     End Select
388. End If
389. DoCmd.Hourglass False
390. MsgBox "Fin del Proceso"
391.
392. End Sub
393.

```

Function generaCalendario(xFechaInicio As Variant, xPeriodo As Integer, XDiasPer As Integer, el_dia As Integer) As Integer

```

394. Dim DBActiva As DATABASE
395. Dim mov_act As Recordset ' movimientos
396. Set DBActiva = DBEngine.Workspaces(0).Databases(0)
397. Set mov_act = DBActiva.OpenRecordset("FechasDeCorte")
398.
399. If xPeriodo = 1 Then
400.     yMes = 0
401.     For i = 1 To 52 Step 1
402.         If yMes = 0 Then
403.             yMes = 1
404.         Else
405.             If yMes <> Month(xFechaInicio) Then
406.                 yMes = Month(xFechaInicio)

```

```

407. End If
408. End If
409.
410. yFecha = DateAdd("d", XDiasPer - 1, xFechaInicio)
411. mov_act.AddNew
412. mov_act![Periodo] = xPeriodo
413. mov_act![FechaInicio] = xFechaInicio
414. mov_act![FechaFinal] = yFecha
415. mov_act![NumeroDeQuincena] = (Year(yFecha) Mod 100) * 100 + i
416. mov_act![EMes] = yMes
417. mov_act.Update
418. xFechaInicio = DateAdd("d", 1, yFecha)
419. Next i
420. Else
421. If xPeriodo = 2 Then
422. Stop
423. año = Year(xFechaInicio) Mod 100
424.
425. For i = 1 To 12 Step 1
426. For j = 1 To 2 Step 1
427. If j = 1 Then
428. xFechaInicio = "#01/" & Format(i, "00") & "/" & año & "#"
429. yFecha = "#15/" & CStr(i) & "/" & año & "#"
430. Else
431. xFechaInicio = "#16/" & Format(i, "00") & "/" & año & "#"
432. Select Case i
433. Case 1: Dias = 31
434. Case 2: Dias = 28
435. Case 3: Dias = 31
436. Case 5: Dias = 31
437. Case 7: Dias = 31
438. Case 8: Dias = 31
439. Case 10: Dias = 31
440. Case 12: Dias = 31
441. Case Else: Dias = 30
442. End Select
443.
444.
445. yFecha = "#" & Dias & "/" & CStr(i) & "/" & año & "#"
446. End If
447.
448.
449. xFechaInicio = Format(xFechaInicio, "dd/mm/yyyy")
450. yFecha = Format(yFecha, "dd/mm/yyyy")
451.
452. mov_act.AddNew
453. mov_act![Periodo] = xPeriodo
454. mov_act![FechaInicio] = CDate(xFechaInicio)
455. mov_act![FechaFinal] = yFecha
456. mov_act![NumeroDeQuincena] = (Year(yFecha) Mod 100) * 100 + (i * 2 - (2 - j))
457. mov_act![EMes] = i
458. mov_act.Update
459. Next j
460. Next i
461. Else
462.

```



```

463. For i = 1 To 12 Step 1
464.     yFecha = DateAdd("d", XDiasPer - 1, xFechaInicio)
465.     yFecha = "#" & el_dia & "/" & CStr(i) & "/" & Year(yFecha) Mod 100 & "#"
466.     mov_act.AddNew
467.     mov_act![Periodo] = xPeriodo
468.     mov_act![FechaInicio] = xFechaInicio
469.     mov_act![FechaFinal] = yFecha
470.     mov_act![NumeroDeQuincena] = (Year(yFecha) Mod 100) * 100 + i
471.     mov_act![ElMes] = i
472.     mov_act.Update
473.
474.     xFechaInicio = DateAdd("d", 1, yFecha)
475.     Next i
476.
477. End If
478. End If
479. End Function
480.

```

Sub Liquidacion_Ahorro(FechaLiquida As Variant, concepto As Integer, Nomina As Integer)

```

481. Dim DBActiva As DATABASE
482. Dim EvaluaSQL As QueryDef
483. Dim nueva_quincena As Variant
484. Dim nuevo_tabulador As Variant
485. Dim criterios As String, MIRS As Recordset
486. Dim nuevoCargo As String
487. Set DBActiva = DBEngine.Workspaces(0).Databases(0)
488.
489.     Set EvaluaSQL = DBActiva.QueryDefs("BorraLiquidacion")
490.     EvaluaSQL.Parameters("LaFecha") = FechaLiquida
491.     EvaluaSQL.Parameters("Elconcepto") = concepto
492.     EvaluaSQL.parameters("LaNomina") = Nomina
493.     EvaluaSQL.Execute
494.     EvaluaSQL.Close
495.
496.
497.     Set EvaluaSQL = DBActiva.QueryDefs("LiquidacionAhorro")
498.     EvaluaSQL.Parameters("LaFecha") = FechaLiquida
499.     EvaluaSQL.Parameters("Elconcepto") = concepto
500.     EvaluaSQL.parameters("LaNomina") = Nomina
501.     EvaluaSQL.Execute
502.     EvaluaSQL.Close
503.
504.     Set EvaluaSQL = DBActiva.QueryDefs("QuitaLosNulosDeLiquidacion1")
505.     EvaluaSQL.Parameters("LaFecha") = FechaLiquida
506.     EvaluaSQL.Parameters("Elconcepto") = concepto
507.     EvaluaSQL.parameters("LaNomina") = Nomina
508.     EvaluaSQL.Execute
509.     EvaluaSQL.Close
510.
511.     Set EvaluaSQL = DBActiva.QueryDefs("QuitaLosNulosDeLiquidacion2")
512.     EvaluaSQL.Parameters("LaFecha") = FechaLiquida
513.     EvaluaSQL.Parameters("Elconcepto") = concepto
514.     EvaluaSQL.parameters("LaNomina") = Nomina

```

```

515.   EvaluaSQL.Execute
516.   EvaluaSQL.Close
517.
518.   Set EvaluaSQL = DBActiva.QueryDefs("QuitaLosNulosDeLiquidacion3")
519.   EvaluaSQL.Parameters("LaFecha") = FechaLiquida
520.   EvaluaSQL.Parameters("Elconcepto") = concepto
521.   'EvaluaSQL.parameters("LaNomina") = Nomina
522.   EvaluaSQL.Execute
523.   EvaluaSQL.Close
524.
525.   Set EvaluaSQL = DBActiva.QueryDefs("ActualizaLiquidacion")
526.   EvaluaSQL.Parameters("LaFecha") = FechaLiquida
527.   EvaluaSQL.Parameters("Elconcepto") = concepto
528.   'EvaluaSQL.parameters("LaNomina") = Nomina
529.   EvaluaSQL.Execute
530.   EvaluaSQL.Close
531.
532.
533.
534.   End Sub
535.

```

Function ObtenFechaInicial(FechaDato As Variant, NumNomina As Integer) As Variant '

```

536.   xEIPeriodo = DLookup("Periodo", "Nómina", "ClaveDeNomina = " & NumNomina)
537.   If IsNull(xEIPeriodo) Then
538.       xEIPeriodo = 3
539.   End If
540.
541.   cadena = " Periodo = " & xEIPeriodo
542.   cadena = cadena & " and #" & Format(FechaDato, "mm/dd/yyyy") & "# between FechaInicio and
FechaFinal"
543.   xFechaInicial = DLookup("FechaInicio", "FechasDeCorte", cadena)
544.   If IsNull(xFechaInicial) Then
545.       xFechaInicial = Format(FechaDato, "dd/mm/yyyy")
546.   End If
547.   ObtenFechaInicial = xFechaInicial
548.
549.
550.   End Function
551.

```

Function ObtenFechaInicial_P(FechaDato As Variant, NumNomina As Integer) As Integer

```

552.   Dim xFechaInicial As Integer
553.
554.   xEIPeriodo = DLookup("Periodo", "Nómina", "ClaveDeNomina = " & NumNomina)
555.
556.
557.   If IsNull(xEIPeriodo) Then
558.       xEIPeriodo = 3
559.   End If
560.
561.   cadena = " Periodo = " & xEIPeriodo

```

```

562.  cadena = cadena & " and #" & Format(FechaDato, "mm/dd/yyyy") & "# between FechaInicio and
      FechaFinal"
563.  xFechaInicial = DLookup("NumeroDeQuincena", "FechasDeCorte", cadena)
564.  If IsNull(xFechaInicial) Then
565.      xFechaInicial = (Year(FechaDato) \ 100) * 100 + Month(FechaDato)
566.  End If
567.  ObtenFechaInicial_P = xFechaInicial
568.
569.  End Function
570.

```

Function que_dias(zzFechaCambio As Variant, zzNumeroDeQuincena As Integer)

```

571.      If Day(zzFechaCambio) <= 15 Then
572.          x = 1
573.          xBase = 1
574.      Else
575.          x = 0
576.          xBase = 16
577.      End If
578.      xper = ((Year(zzFechaCambio) Mod 100) * 100) + ((Month(zzFechaCambio) * 2) - x)
579.      If xper = zzNumeroDeQuincena Then
580.          If Day(zzFechaCambio) > 30 Then
581.              xDias = 30
582.          Else
583.              xDias = Day(zzFechaCambio)
584.          End If
585.          el_sig_per = (xDias - xBase) * -1
586.      Else
587.          If x = 1 Then
588.              el_sig_per = 15 - Day(zzFechaCambio) + 1
589.          Else
590.              el_sig_per = 30 - Day(zzFechaCambio) + 1
591.          End If
592.      End If
593.      que_dias = el_sig_per
594.
595.  End Function
596.

```

Function xprueba()

```

597.  DoCmd.TransferText A_EXPORTDELIM, "ContratoEventual", "GeneraContratoEventual",
      "Z:\etraci\CONTRA\GeneraC.Txt", True
598.
599.  End Function
600.  Option Compare Database

```

Function AjustarRecibos(xMaxPer As Integer, Secuencia As Integer, yConcepto As Integer, xTipo As String) As Integer

```

601.
602.  Dim DBActiva As DATABASE
603.  Dim Sql1 As QueryDef
604.  Dim EvaluaSQL As QueryDef

```

```

605. Dim EvaluaReg As Recordset
606. Dim xEmpleado As Long
607. Dim xSuma As Double
608. Dim xCuenta As Integer
609. Dim xConcepto As Integer
610. Dim cad As String
611.
612. Set DBActiva = DBEngine.Workspaces(0).Databases(0)
613. Set Sql1 = DBActiva.CreateQueryDef("")
614.
615.
616.
617. If Secuencia = 0 Then
618.   cad = " Delete * from ConceptosABorrar"
619.   Sql1.SQL = cad
620.   Sql1.Execute
621. End If
622.
623.   cad = " Delete * from LosAjustes"
624.   Sql1.SQL = cad
625.   Sql1.Execute
626.
627.   cad = " INSERT INTO LosAjustes ( ClaveDeEmpleado, PerDed, CuentaConceptos )"
628.   cad = cad & " SELECT MovinNomina_Reportes.ClaveDeEmpleado,
PercepcionesDeducciones.PerDed, Count(MovinNomina_Reportes.ClaveDeConcepto) AS
CuentaDeClaveDeConcepto"
629.   cad = cad & " FROM MovinNomina_Reportes INNER JOIN PercepcionesDeducciones ON
(MovinNomina_Reportes.ClaveDeConcepto = PercepcionesDeducciones.ClaveDeConcepto) AND
(MovinNomina_Reportes.NumeroDeNómina = PercepcionesDeducciones.NumeroDeNomina)"
630.   cad = cad & " GROUP BY MovinNomina_Reportes.ClaveDeEmpleado,
PercepcionesDeducciones.PerDed"
631.   cad = cad & " HAVING (((PercepcionesDeducciones.PerDed)= "" & xTipo & "") AND
((Count(MovinNomina_Reportes.ClaveDeConcepto))> " & xMaxPer & " ));"
632.
633.
634.   Sql1.SQL = cad
635.   Sql1.Execute
636.
637.
638.   Set EvaluaSQL = DBActiva.QueryDefs("RecorreConceptos")
639.   Set EvaluaReg = EvaluaSQL.OpenRecordset()
640.   Stop
641.   If EvaluaReg.EOF Then
642.     MsgBox " No hay Conceptos a Ajustar"
643.   Else
644.     EvaluaReg.MoveLast
645.     EvaluaReg.MoveFirst
646.     CuentaRegistro = EvaluaReg.RecordCount
647.     EvaluaCue = 0
648.     Cuenta = SysCmd(SYSCMD_INITMETER, "Ajustando Conceptos ", CuentaRegistro)
649.     DoCmd.Hourglass False
650.     xEmpleado = 0
651.     Stop
652.     Do Until EvaluaReg.EOF
653.       xQuincena = EvaluaReg!NumeroDeQuincena
654.       xNomina = EvaluaReg!NumeroDeNómina

```

```

655.
656.     If xEmpleado = 0 Then
657.         xEmpleado = EvaluaReg!ClaveDeEmpleado
658.         xCuenta = 0
659.         xSuma = 0
660.     End If
661.
662.     If xEmpleado = EvaluaReg!ClaveDeEmpleado Then
663.
664.     Else
665.         Stop
666.         cad = "INSERT INTO MovinNomina_Reportes ( NumeroDeQuincena, ClaveDeEmpleado,
ClaveDeConcepto, NumeroDeNómina, VariableParaPercepYDed, MontoMovi )"
667.         cad = cad & " values (" & EvaluaReg!NumeroDeQuincena & "," & xEmpleado & "," &
yConcepto & "," & EvaluaReg!NumeroDeNómina & ",0," & xSuma & ")"
668.         Sql1.SQL = cad
669.         Sql1.Execute
670.         xSuma = 0
671.         xCuenta = 0
672.         xEmpleado = EvaluaReg!ClaveDeEmpleado
673.     End If
674.
675.     If xCuenta < xMaxPer Then
676.         xCuenta = xCuenta + 1
677.     Else
678.         Stop
679.         xSuma = xSuma + EvaluaReg!MontoMovi
680.         xConcepto = EvaluaReg!ClaveDeConcepto
681.         cad = " insert into ConceptosABorrar (ClaveDeEmpleado,ClaveDeConcepto) values (" &
xEmpleado & "," & xConcepto & ")"
682.         Sql1.SQL = cad
683.         Sql1.Execute
684.
685.         'cad = "INSERT INTO MovinNomina_Reportes ( NumeroDeQuincena,
ClaveDeEmpleado, ClaveDeConcepto, ClaveDeEmpresa, NumeroDeNómina, TipoDeNomina,
VariableParaPercepYDed, MontoMovi )"
686.         'cad = cad & " values (" & EvaluaReg!NumeroDeQuincena & "," & xEmpleado & "," &
yConcepto & ",8," & EvaluaReg!NumeroDeNómina & ",N',0," & xSuma & ")"
687.         'SQL1.SQL = cad
688.         'SQL1.Execute
689.
690.
691.     End If
692.
693.
694.     EvaluaReg.MoveNext
695.     EvaluaCue = EvaluaCue + 1
696.     Cuenta = SysCmd(SYSCMD_UPDATERMETER, EvaluaCue)
697.     Loop
698.     cad = "INSERT INTO MovinNomina_Reportes ( NumeroDeQuincena,
ClaveDeEmpleado, ClaveDeConcepto, NumeroDeNómina, VariableParaPercepYDed, MontoMovi )"
699.     cad = cad & " values (" & xQuincena & "," & xEmpleado & "," & yConcepto & "," &
xNomina & ",0," & xSuma & ")"
700.     Sql1.SQL = cad
701.     Sql1.Execute
702.

```

```

703.     EvaluaReg.Close
704.     Cuenta = SysCmd(SYSCMD_REMOVEMETER)
705.     DoCmd.Hourglass False
706. End If
707. -----
708.     If Secuencia = 1 Then
709.         cad = "DELETE Movinomina_Reportes.""
710.         cad = cad & " FROM ConceptosABorrar INNER JOIN Movinomina_Reportes ON
(ConceptosABorrar.ClaveDeConcepto = Movinomina_Reportes.ClaveDeConcepto) AND
(ConceptosABorrar.ClaveDeEmpleado = Movinomina_Reportes.ClaveDeEmpleado);"
711.         Sql1.SQL = cad
712.         Sql1.Execute
713.         Sql1.Close
714.     End If
715.
716.
717. End Function
718. Option Compare Database 'Usar orden de base de datos en comparaciones de cadenas
719.

```

Function prueba(Titulo As String, Mensaje As String)

```

720.
721.     Const MB_ACEPTAR = 0, MB_ACEPTARCANCELAR = 1 ' Define botones.
722.     Const MB_SINOCANCELAR = 3, MB_SÍNO = 4
723.     Const MB_ICONOALTO = 16, MB_ICONINTERROGACIÓN = 32 ' Define iconos.
724.     Const MB_ICONADMIRACIÓN = 48, MB_ICONINFORMACIÓN = 64
725.     Const MB_DEFBOTÓN2 = 256, IDSÍ = 6, IDNO = 7 ' Define otros.
726.     Titulo = Titulo
727.     ' Crea un cuadro de mensaje de muestra con todos los componentes apropiados.
728.     msg = "Este es un ejemplo de un mensaje de error crítico."
729.     msg = Mensaje
730.     msg = msg & " ¿Desea continuar?"
731.
732.     dgdef = MB_SÍNO + MB_ICONOALTO + MB_DEFBOTÓN2 ' Describe diálogo.
733.     Respuesta = MsgBox(msg, dgdef, Titulo) ' Obtiene respuesta de usuario.
734.     If Respuesta = IDSÍ Then ' Evalúa la respuesta y
735.         prueba = 1 ' msg = "Elegió Sí." ' realiza la acción
736.     Else ' apropiada.
737.         prueba = 0 ' msg = "Elegió No o presionó Entrar."
738.     End If
739.     'MsgBox msg ' Presenta la acción realizada.
740.
741.
742. End Function
743. Option Compare Database
744. Option Explicit
745.

```

Function BotonDeEleccion(Titulo As String, Mensaje As String) As Integer

```

746. Dim Estilo, Ayuda, Ctxt, Respuesta, MiCadena
747.
748. Estilo = vbYesNo + vbCritical + vbDefaultButton2 ' Define los botones.
749.

```

```

750. Ayuda = "DEMO.HLP" ' Define el archivo de ayuda.
751. Cbxt = 1000 ' Define el tema
752.     ' el contexto
753.     ' Muestra el mensaje.
754. Respuesta = MsgBox(Mensaje & " ¿Desea Continuar?", vbYesNo, Titulo)
755.
756.
757. If Respuesta = vbYes Then ' El usuario eligió el botón Sí.
758.     BotonDeEleccion = 1 ' Ejecuta una acción.
759. Else ' El usuario eligió el botón No.
760.     BotonDeEleccion = 0 ' Ejecuta una acción.
761. End If
762. End Function
763. Declare Function Lib1 Lib "formul.dll" (ByVal sDato As String) As Double
764. Option Compare Database 'Usar orden de base de datos en comparaciones de cadenas
765. Global hzgxxElFacFondo As Double
766. Global Aniv(4) As Integer
767. Global kkQui_ini As Integer
768. Global kkQui_fin As Integer
769. Global kkDiasAlPer As Double
770. Global KKDiasDePer As Double
771.
772.

```

Function bus_val(Clave As String, cla_emp As Long, quincena As Integer, cla_mov As Integer, el_sueldo As Variant, numero As Integer, xSucursal As Integer) As String

```

773. Dim DBActiva As DATABASE
774. Dim tam As Integer
775. 'Dim num_cla As Double
776. Dim num_cla As Variant
777.
778. Dim smg_df As Variant
779. Dim clave1 As String
780. Dim conemp As QueryDef
781. Dim sel_mov As String ' variable de selección
782. Dim res_sel As Recordset ' respuesta de selección
783. Dim isrSQL As QueryDef
784. Dim aux_qui As Integer
785. Dim aux_mov As Integer
786. Dim tip_nom As String
787. Dim nume_p As Integer
788. Dim Valor As Currency
789. Dim cad_val As String
790. Dim dob_val As Double
791. Dim valore As Currency
792. Dim fecha_1 As Variant
793. Dim fec_ing As Variant
794. Dim criterio As String
795.
796. Dim TempSQL As QueryDef
797.
798. Dim año As Double
799. Dim dia As Double
800. Dim consulta As String

```

```

801. Dim num_hij As Integer
802. 'Dim montos1 As Currency
803.
804.
805. Set DBActiva = DBEngine.Workspaces(0).Databases(0)
806. tam = Len(Clave)
807. aux_mov = cla_mov
808. num_cla = Null
809. Select Case Mid$(Clave, 1, 2)
810.
811. Case "@A" ' @A9705 01
812.   consulta = ""
813.   aux_qui = quincena
814.   tip_nom = "N"
815.   consulta = "NumeroDeQuincena = " & Mid(Clave, 3, 4) 'aux_qui
816.   consulta = consulta & " and NumeroDeNomina = " & Mid(Clave, 7, tam)
817.   consulta = consulta & " and ClaveDeEmpleado = " & cla_emp
818.   montos = DSum("MontoGravado", "AcumGravado", consulta)
819.   If IsNull(montos) Then
820.     bus_val = "0"
821.   Else
822.     clave1 = CStr(montos)
823.     bus_val = clave1
824.   End If
825.
826. Case "@N" ' @N9705 01 22 Monto
827.
828.   consulta = ""
829.   aux_qui = quincena
830.   tip_nom = "N"
831.   consulta = "NumeroDeQuincena = " & Mid(Clave, 3, 4) 'aux_qui
832.   consulta = consulta & " and ClaveDeEmpleado = " & cla_emp
833.   consulta = consulta & " and NumeroDeNomina = " & CInt(Mid(Clave, 7, 2))
834.   consulta = consulta & " and ClaveDeConcepto = " & Mid(Clave, 9, tam)
835.   consulta = consulta & " and TipoDeNomina = " & tip_nom & ""
836.   montos = DLookup("MontoMovi", "MoviNomina", consulta)
837.   If IsNull(montos) Then
838.     bus_val = "0"
839.   Else
840.     clave1 = CStr(montos)
841.     bus_val = clave1
842.   End If
843. Case "@C" ' @C 01 22 Monto de la nomina y periodo dado
844.
845.   consulta = ""
846.   aux_qui = quincena
847.   tip_nom = "N"
848.   consulta = "NumeroDeQuincena = " & aux_qui
849.   consulta = consulta & " and ClaveDeEmpleado = " & cla_emp
850.   consulta = consulta & " and NumeroDeNomina = " & CInt(Mid(Clave, 3, 2))
851.   consulta = consulta & " and ClaveDeConcepto = " & Mid(Clave, 5, 4)
852.   consulta = consulta & " and TipoDeNomina = " & tip_nom & ""
853.   montos = DLookup("MontoMovi", "MoviNomina", consulta)
854.   If IsNull(montos) Then
855.     bus_val = "0"
856.   Else

```



```

857.     clave1 = CStr(montos)
858.     bus_val = clave1
859.     End If
860.
861. Case "@D" ' @D9705 01 22 Variable
862.
863.     consulta = ""
864.     aux_qui = quincena
865.     tip_nom = "N"
866.     consulta = "NumeroDeQuincena = " & Mid(Clave, 3, 4) 'aux_qui
867.     consulta = consulta & " and ClaveDeEmpleado = " & cla_emp
868.     consulta = consulta & " and NumeroDeNómina = " & Cint(Mid(Clave, 7, 2))
869.     consulta = consulta & " and ClaveDeConcepto = " & Mid(Clave, 9, tam)
870.     consulta = consulta & " and TipoDeNomina = "" & tip_nom & ""
871.     montos = DLookup("VariableParaPercepYDed", "MoviNomina", consulta)
872.     If IsNull(montos) Then
873.         bus_val = "0"
874.     Else
875.         clave1 = CStr(montos)
876.         bus_val = clave1
877.     End If
878.
879. Case "@T" ' @T 01 22 Variable
880.
881.     consulta = ""
882.     aux_qui = quincena
883.     tip_nom = "N"
884.     consulta = "NumeroDeQuincena = " & aux_qui
885.     consulta = consulta & " and ClaveDeEmpleado = " & cla_emp
886.     consulta = consulta & " and NumeroDeNómina = " & Cint(Mid(Clave, 3, 2))
887.     consulta = consulta & " and ClaveDeConcepto = " & Mid(Clave, 5, tam)
888.     consulta = consulta & " and TipoDeNomina = "" & tip_nom & ""
889.     montos = DLookup("VariableParaPercepYDed", "MoviNomina", consulta)
890.     If IsNull(montos) Then
891.         bus_val = "0"
892.     Else
893.         clave1 = CStr(montos)
894.         bus_val = clave1
895.     End If
896.
897. Case "@M" ' Monto del Periodo Actual
898.     consulta = ""
899.     aux_qui = quincena
900.     tip_nom = "N"
901.     consulta = "NumeroDeQuincena = " & aux_qui
902.     consulta = consulta & " and ClaveDeEmpleado = " & cla_emp
903.     consulta = consulta & " and ClaveDeConcepto = " & Mid(Clave, 3, tam)
904.     consulta = consulta & " and NumeroDeNómina = " & numero
905.     consulta = consulta & " and TipoDeNomina = "" & tip_nom & ""
906.     montos = DLookup("MontoMovi", "MoviNominaT", consulta)
907.     If IsNull(montos) Then
908.         bus_val = "0"
909.     Else
910.         clave1 = CStr(montos)
911.         bus_val = clave1
912.     End If

```

```

913.
914. Case "@I" 'Impuesto de la cantidad
915.     consulta = ""
916.     aux_qui = quincena
917.     tip_nom = "N"
918.     Dim zzMontos As Currency
919.     Stop
920.     consulta = "NumeroDeQuincena = " & aux_qui
921.     consulta = consulta & " and ClaveDeEmpleado = " & cla_emp
922.     consulta = consulta & " and ClaveDeConcepto = " & Mid(Clave, 3, tam)
923.     consulta = consulta & " and NumeroDeNómina = " & numero
924.     consulta = consulta & " and TipoDeNomina = "" & tip_nom & ""
925.     montos = DLookup("MontoMovi", "MoviNominaT", consulta)
926.     If IsNull(montos) Then
927.         bus_val = "0"
928.     Else
929.         zzMontos = montos
930.         montos = cal_ispt(zzMontos, quincena, cla_emp, "N", 3, 3, 0, 0)
931.         If montos <= 0 Then
932.             montos = 0
933.         End If
934.         clave1 = CStr(montos)
935.         bus_val = clave1
936.     End If
937. Case "@K" 'Impuesto de la cantidad
938.     consulta = ""
939.     aux_qui = quincena
940.     tip_nom = "N"
941.     Dim zzzMontos As Currency
942.     consulta = "NumeroDeQuincena = " & aux_qui
943.     consulta = consulta & " and ClaveDeEmpleado = " & cla_emp
944.     consulta = consulta & " and ClaveDeConcepto = " & Mid(Clave, 3, tam)
945.     consulta = consulta & " and NumeroDeNómina = " & numero
946.     consulta = consulta & " and TipoDeNomina = "" & tip_nom & ""
947.     montos = DLookup("MontoMovi", "MoviNominaT", consulta)
948.     If IsNull(montos) Then
949.         bus_val = "0"
950.     Else
951.         zzzMontos = montos
952.         montos = cal_ispt(zzzMontos, quincena, cla_emp, "N", 3, 3, 1, 0)
953.         If montos <= 0 Then
954.             montos = 0
955.         End If
956.         clave1 = CStr(montos)
957.         bus_val = clave1
958.     End If
959. Case "@V" ' Variable del periodo actual
960.     consulta = ""
961.     aux_qui = quincena
962.     tip_nom = "N"
963.     consulta = "NumeroDeQuincena = " & aux_qui
964.     consulta = consulta & " and ClaveDeEmpleado = " & cla_emp
965.     consulta = consulta & " and ClaveDeConcepto = " & Mid(Clave, 3, tam)
966.     consulta = consulta & " and NumeroDeNómina = " & numero
967.     consulta = consulta & " and TipoDeNomina = "" & tip_nom & ""
968.     montos = DLookup("VariableParaPercepYDed", "MoviNominaT", consulta)

```

```

969.     If IsNull(montos) Then
970.         bus_val = "0"
971.     Else
972.         clave1 = CStr(montos)
973.         bus_val = clave1
974.     End If
975.
976.
977.     Case "@B"
978.
979.         Set isrSQL = DBActiva.QueryDefs("BusPar")
980.         isrSQL.Parameters("num_qui") = quincena
981.         isrSQL.Parameters("num_par") = 16
982.         Set res_Sel = isrSQL.OpenRecordset()
983.         If res_Sel.EOF Then
984.             montos = 0
985.         Else
986.             montos = res_Sel![MontoParametro]
987.         End If
988.
989.         If eI_sueldo <= montos Then
990.
991.             consulta = ""
992.             aux_qui = quincena
993.             tip_nom = "N"
994.             consulta = "NumeroDeQuincena = " & aux_qui
995.             consulta = consulta & " and ClaveDeEmpleado = " & cla_emp
996.             consulta = consulta & " and ClaveDeConcepto = " & Mid(Clave, 3, tam)
997.             consulta = consulta & " and NumeroDeNómina = " & numero
998.             consulta = consulta & " and TipoDeNomina = " & tip_nom & ""
999.             montos = DLookup("MontoMovi", "MoviNominaT", consulta)
1000.         If IsNull(montos) Then
1001.             bus_val = "0"
1002.         Else
1003.             DoCmd.RunSQL "Delete from MoviNominaT where " & consulta & ";"
1004.             clave1 = CStr(montos)
1005.             bus_val = clave1
1006.         End If
1007.     Else
1008.         bus_val = "0"
1009.     End If
1010.
1011.
1012.     Case "@R"
1013.         consulta = ""
1014.         aux_qui = quincena
1015.         tip_nom = "N"
1016.         consulta = "NumeroDeQuincena = " & aux_qui
1017.         consulta = consulta & " and ClaveDeEmpleado = " & cla_emp
1018.         consulta = consulta & " and ClaveDeConcepto = " & Mid(Clave, 3, tam)
1019.         consulta = consulta & " and NumeroDeNómina = " & numero
1020.         consulta = consulta & " and TipoDeNomina = " & tip_nom & ""
1021.         montos = DLookup("VariableParaPercepYDed", "MoviNominaT", consulta)
1022.         If IsNull(montos) Then
1023.             montos = 0
1024.         End If

```

```

1025.
1026.   ajuste_mon = (montos * 100) Mod 100
1027.   If ajuste_mon >= 50 Then
1028.       ajuste_mon = -1
1029.   Else
1030.       ajuste_mon = 0
1031.   End If
1032.
1033.       montos = Int(montos * ajuste_mon)
1034.       montos = montos * ajuste_mon
1035.       clave1 = CStr(montos)
1036.       bus_val = clave1
1037.
1038.
1039.
1040.   Case "@P" ' Por Sucursal
1041.
1042.       Set isrSQL = DBActiva.QueryDefs("BusParSUC")
1043.       isrSQL.Parameters("num_qui") = quincena
1044.       isrSQL.Parameters("num_par") = Cint(Mid(Clave, 3, tam))
1045.       isrSQL.Parameters("xSucursal") = xSucursal
1046.       Set res_Sel = isrSQL.OpenRecordset()
1047.       If res_Sel.EOF Then
1048.           'IsNull(res_sel![montoParametro]) Then
1049.               montos = 0
1050.           Else
1051.               montos = res_Sel![MontoParametro]
1052.           End If
1053.
1054.
1055.       clave1 = CStr(montos)
1056.       bus_val = clave1
1057.
1058.   ----- Concepto Parametrizables -----
1059.
1060.   Case Else
1061.       Select Case Clave
1062.           Case "DIASNAT"
1063.               xxMes = quincena Mod 100
1064.               criterio = "ClaveDeNomina =" & numero
1065.               xxPer = DLookup("Periodo", "Nómina", criterio)
1066.               If IsNull(xxPer) Then
1067.                   xxPer = 3
1068.               End If
1069.
1070.               criterio = "mes =" & xxMes & " and ClaveDeNomina =" & xxPer
1071.               num_cla = DLookup("dias", "DiasMes", criterio)
1072.               If IsNull(num_cla) Then
1073.                   num_cla = 30
1074.               End If
1075.
1076.               bus_val = CStr(num_cla)
1077.
1078.           Case "PROVPRIMA"
1079.               criterio = "ClaveDeEmpleado =" & cia_emp
1080.               num_cla = DLookup("DiasSaído", "MaeVacaciones", criterio)

```

```

1081.     If IsNull(num_cla) Then
1082.         num_cla = 0
1083.     End If
1084.     clave1 = CStr(num_cla)
1085.     bus_val = clave1
1086.     Case "SD"
1087.         num_cla = el_sueldo
1088.         clave1 = CStr(num_cla)
1089.         bus_val = clave1
1090.     Case "SDI"
1091.         criterio = "ClaveDeEmpleado = " & cla_emp & " and Estado = 'A' and ClaveDelmpuesto =
'IMSS'"
1092.         num_cla = DLookup("MontoSDI", "SDI_BIM", criterio)
1093.         If IsNull(num_cla) Then
1094.             num_cla = 0
1095.         End If
1096.         clave1 = CStr(num_cla)
1097.         bus_val = clave1
1098.
1099. 'codigo agregado por v.h. 12/06/98
1100.     Case "SDIIN"
1101.         criterio = "ClaveDeEmpleado = " & cla_emp & " and Estado = 'A' and ClaveDelmpuesto =
'SDIIN'"
1102.         num_cla = DLookup("MontoSDI", "SDI_BIM", criterio)
1103.         If IsNull(num_cla) Then
1104.             num_cla = 0
1105.         End If
1106.         clave1 = CStr(num_cla)
1107.         bus_val = clave1
1108.     ' termina agregado
1109.
1110.
1111.     Case "SDIINFO"
1112.         criterio = "ClaveDeEmpleado = " & cla_emp & " and Estado = 'A' and ClaveDelmpuesto =
'INFO'"
1113.         num_cla = DLookup("MontoSDI", "SDI_BIM", criterio)
1114.         If IsNull(num_cla) Then
1115.             num_cla = 0
1116.         End If
1117.         clave1 = CStr(num_cla)
1118.         bus_val = clave1
1119.
1120.     Case "SDINFOPOR"
1121.         criterio = "ClaveDeEmpleado = " & cla_emp & " and Estado = 'ACTIVO'"
1122.         criterio = criterio & " and NumeroDeQuincena <= " & quincena & " and ClaveDeConcepto = "
& cla_mov
1123.         num_cla = DLookup("MontoDeConcepto", "ConceptosAdicional", criterio)
1124.
1125.         If IsNull(num_cla) Then
1126.             num_cla = 0
1127.             xRango = 0
1128.         Else
1129.             Set isrSQL = DBActiva.QueryDefs("BusParSUC")
1130.             isrSQL.Parameters("num_qui") = quincena
1131.             isrSQL.Parameters("num_par") = 16
1132.             isrSQL.Parameters("xSucursal") = 1

```

```

1133.      Set res_Sel = isrSQL.OpenRecordset()
1134.      If res_Sel.EOF Then
1135.          montos = 1
1136.      Else
1137.          montos = res_Sel![MontoParametro]
1138.      End If
1139.
1140.      criterio = "ClaveDeEmpleado = " & cla_emp & " and Estado = 'A' and
ClaveDeImpuesto = 'INFO'"
1141.      xnum_cla = DLookup("MontoSDI", "SDI_BIM", criterio)
1142.      If IsNull(xnum_cla) Then
1143.          xnum_cla = 0
1144.      End If
1145.
1146.      xRango = Cint((xnum_cla / montos) * 100) / 100
1147.
1148.      criterio = " PorDeAviso = " & num_cla & " and " & xRango & " between RanIni and
ranFin "
1149.      xRango = DLookup("PorAplicar", "PorAvisoInfo", criterio)
1150.      If IsNull(xRango) Then
1151.          xRango = 0
1152.      End If
1153.
1154.      End If
1155.
1156.
1157.
1158.      clave1 = CStr(xRango)
1159.      bus_val = clave1
1160.      Case "ANTICIPO"
1161.          criterio = "ClaveDeEmpleado = " & cla_emp & " and NumeroDeQuincena = " & quincena
1162.          num_cla = DLookup("Adelanto", "RepAdelantos", criterio)
1163.          If IsNull(num_cla) Then
1164.              num_cla = 0
1165.          End If
1166.          clave1 = CStr(num_cla)
1167.          bus_val = clave1
1168.      Case "DIASDEV"
1169.          Stop
1170.          criterio = "ClaveDeEmpleado = " & cla_emp
1171.          num_cla = DLookup("Dias", "DiasDevengados", criterio)
1172.          If IsNull(num_cla) Or num_cla <= 0 Then
1173.              num_cla = 1
1174.          End If
1175.          clave1 = CStr(num_cla)
1176.          bus_val = clave1
1177.
1178.      Case "DIASNDEV"
1179.          criterio = "ClaveDeEmpleado = " & cla_emp
1180.          num_cla = DLookup("Dias", "DiasNoDevengados", criterio)
1181.          If IsNull(num_cla) Then
1182.              num_cla = 0
1183.          End If
1184.          clave1 = CStr(num_cla)
1185.          bus_val = clave1
1186.

```

```

1187. Case "MONDEV"
1188.     criterio = "ClaveDeEmpleado = " & cla_emp
1189.     num_cla = DLookup("Monto", "MontoDevengado", criterio)
1190.     If IsNull(num_cla) Then
1191.         num_cla = 0
1192.     End If
1193.     clave1 = CStr(num_cla)
1194.     bus_val = clave1
1195.
1196. Case "NPM"
1197.     num_cla = (kkQui_fin - kkQui_ini) + 1
1198.     clave1 = CStr(num_cla)
1199.     bus_val = clave1
1200.
1201. Case "SDSAR"
1202.     criterio = "ClaveDeEmpleado = " & cla_emp & " and Estado = 'A' and ClaveDeImpuesto =
'SAR'"
1203.     num_cla = DLookup("MontoSDI", "SDI_BIM", criterio)
1204.     If IsNull(num_cla) Then
1205.         num_cla = 0
1206.     End If
1207.     clave1 = CStr(num_cla)
1208.     bus_val = clave1
1209.
1210. Case "ADICIONAL"
1211.     criterio = "ClaveDeEmpleado = " & cla_emp & " and Estado = 'ACTIVO'"
1212.     criterio = criterio & " and NumeroDeQuincena <= " & quincena & " and ClaveDeConcepto = "
& cla_mov
1213.     num_cla = DLookup("MontoDeConcepto", "ConceptosAdicional", criterio)
1214.     If IsNull(num_cla) Then
1215.         num_cla = 0
1216.     End If
1217.     clave1 = CStr(num_cla)
1218.     bus_val = clave1
1219.
1220. Case "TOTALPER"
1221.     criterio = "ClaveDeEmpleado = " & cla_emp & " and Estado = 'ACTIVO'"
1222.     criterio = criterio & " and NumeroDeQuincena <= " & quincena & " and ClaveDeConcepto = "
& cla_mov
1223.     num_cla = DLookup("MontoDeConcepto", "ConceptosAdicional", criterio)
1224.     If IsNull(num_cla) Then
1225.         num_cla = 0
1226.     Else
1227.         aux_qui = quincena
1228.         Set isrSQL = DBActiva.QueryDefs("BusTotPerTotal")
1229.         isrSQL.Parameters("num_qui") = quincena
1230.         isrSQL.Parameters("num_emp") = cla_emp
1231.         isrSQL.Parameters("num_nom") = numero
1232.         Set res_Sel = isrSQL.OpenRecordset()
1233.         If res_Sel.EOF Or IsNull(res_Sel![monto]) Then
1234.             num_cla = 0
1235.         Else
1236.             num_cla = res_Sel![monto]
1237.         End If
1238.
1239.     End If

```

```

1240.     clave1 = CStr(num_cla)
1241.     bus_val = clave1
1242.
1243.     Case "APORTAFONDO"
1244.         criterio = "ClaveDeEmpleado = " & cla_emp & " and ClaveDeConcepto = 290"
1245.         num_cla = DLookup("Aportacion", "LiquidaAhorra", criterio)
1246.         If IsNull(num_cla) Then
1247.             num_cla = 0
1248.         End If
1249.         clave1 = CStr(num_cla * 2)
1250.         bus_val = clave1
1251.
1252.     Case "INTERESFONDO"
1253.         criterio = "ClaveDeEmpleado = " & cla_emp & " and ClaveDeConcepto = 290"
1254.         num_cla = DLookup("Interesind", "LiquidaAhorra", criterio)
1255.         If IsNull(num_cla) Then
1256.             num_cla = 0
1257.         End If
1258.         clave1 = CStr(num_cla)
1259.         bus_val = clave1
1260.
1261.     Case "VARIABLE"
1262.         consulta = ""
1263.         aux_qui = quincena
1264.         tip_nom = "N"
1265.         consulta = "NumeroDeQuincena = " & aux_qui
1266.         consulta = consulta & " and ClaveDeEmpleado = " & cla_emp
1267.         consulta = consulta & " and ClaveDeConcepto = " & cla_mov
1268.         consulta = consulta & " and NumeroDeNómina = " & numero
1269.         consulta = consulta & " and TipoDeNomina = " & tip_nom & ""
1270.         montos = DLookup("VariableParàPercepyDed", "MoviNominaT", consulta)
1271.
1272.         If IsNull(montos) Then
1273.             bus_val = "0"
1274.         Else
1275.             clave1 = CStr(montos)
1276.             If montos < 1 Then
1277.                 clave1 = "0" & clave1
1278.             End If
1279.             bus_val = clave1
1280.         End If
1281.
1282.     Case "SMG" ' Por Sucursal
1283.
1284.         Set isrSQL = DBActiva.QueryDefs("BusPar")
1285.         isrSQL.Parameters("num_qui") = quincena
1286.         isrSQL.Parameters("num_par") = 16
1287.         'isrSQL.Parameters("xSucursal") = xSucursal
1288.         Set res_Sel = isrSQL.OpenRecordset()
1289.         If res_Sel.EOF Then
1290.             'IsNull(res_sel![montoParametro]) Then
1291.                 montos = 0
1292.             Else
1293.                 montos = res_Sel![MontoParametro]
1294.             End If
1295.

```



```

1296.
1297.     clave1 = CStr(montos)
1298.     bus_val = clave1
1299.
1300. Case "SMGDF" ' El del distrito federal
1301.
1302.     Set isrSQL = DBActiva.QueryDefs("BusPar")
1303.     isrSQL.Parameters("num_qui") = quincena
1304.     isrSQL.Parameters("num_par") = 19
1305.     'isrSQL.Parameters("xSucursal") = 1
1306.     Set res_Sel = isrSQL.OpenRecordset()
1307.     If res_Sel.EOF Then
1308.         'IsNull(res_sel![montoParametro]) Then
1309.             montos = 0
1310.         Else
1311.             montos = res_Sel![MontoParametro]
1312.         End If
1313.
1314.
1315.     clave1 = CStr(montos)
1316.     bus_val = clave1
1317.
1318. Case "SMGM" ' el del distrito federal
1319.
1320.     Set isrSQL = DBActiva.QueryDefs("BusParSUC")
1321.     isrSQL.Parameters("num_qui") = quincena
1322.     isrSQL.Parameters("num_par") = 16
1323.     isrSQL.Parameters("xSucursal") = 1
1324.     Set res_Sel = isrSQL.OpenRecordset()
1325.     If res_Sel.EOF Then
1326.         'IsNull(res_sel![montoParametro]) Then
1327.             montos = 0
1328.         Else
1329.             montos = res_Sel![MontoParametro]
1330.         End If
1331.
1332.
1333.     clave1 = CStr(montos)
1334.     bus_val = clave1
1335.
1336.
1337. Case "CAJA"
1338.     consulta = "ClaveDeEmpleado = " & cla_emp
1339.     consulta = consulta & " and NumeroDeQuincena <= " & quincena
1340.     consulta = consulta & " and ClaveDeConcepto = " & cla_mov
1341.
1342.     z_per = DMax("NumeroDeQuincena", "LaAportaCaja", consulta)
1343.     If IsNull(z_per) Then
1344.         num_cia = "0"
1345.     Else
1346.         consulta = "ClaveDeEmpleado = " & cla_emp
1347.         consulta = consulta & " and NumeroDeQuincena = " & z_per
1348.         consulta = consulta & " and ClaveDeConcepto = " & cla_mov
1349.         num_cia = DLookup("Monto", "LaAportaCaja", consulta)
1350.     End If
1351.     If IsNull(num_cia) Then

```

```

1352.         bus_val = "0"
1353.     Else
1354.         clave1 = CStr(num_cla)
1355.         bus_val = clave1
1356.     End If
1357.
1358. Case "SALDOCAJA"
1359.     consulta = "ClaveDeEmpleado = " & cla_emp
1360.     consulta = consulta & " and ClaveDeConcepto = " & cla_mov
1361.
1362.     num_cla = DLookup("RetencionP", "InsCajaFondo", consulta)
1363.
1364.     If IsNull(num_cla) Then
1365.         bus_val = "0"
1366.     Else
1367.         clave1 = CStr(num_cla)
1368.         bus_val = clave1
1369.     End If
1370.
1371. '----- Hasta aqui totalmente validado
1372.
1373. Case "VALES"
1374.     consulta = ""
1375.     aux_qui = quincena
1376.     tip_nom = "N"
1377.     consulta = "NumeroDeQuincena BETWEEN " & quincena - 1 & " and " & quincena
1378.     consulta = consulta & " and ClaveDeEmpleado = " & cla_emp
1379.     consulta = consulta & " and ClaveDeConcepto = " & cla_mov
1380.     consulta = consulta & " and NumeroDeNómina = 2 "
1381.     consulta = consulta & " and TipoDeNomina = " & tip_nom & ""
1382.     montos = DSum("MontoMovi", "MoviNomina", consulta)
1383.     If IsNull(montos) Then
1384.         bus_val = "0"
1385.     Else
1386.         clave1 = CStr(montos)
1387.         bus_val = clave1
1388.     End If
1389.
1390. Case "FALTAS"
1391.     Set isrSQL = DBActiva.QueryDefs("ComoFalta")
1392.     isrSQL.Parameters("empleado") = cla_emp
1393.     isrSQL.Parameters("lanomina") = numero
1394.     isrSQL.Parameters("elperiodo") = quincena
1395.     Set res_Sel = isrSQL.OpenRecordset()
1396.     If res_Sel.EOF Then
1397.         'IsNull(res_sel![Variable]) Then
1398.             montos = 0
1399.         Else
1400.             montos = res_Sel![variable]
1401.         End If
1402.         clave1 = CStr(montos)
1403.         bus_val = clave1
1404.
1405.
1406. Case "FALTASACUMULA"
1407.     Set isrSQL = DBActiva.QueryDefs("ComoFaltaAcumula")

```

```

1408.   isrSQL.Parameters("empleado") = cla_emp
1409.   'IsrSQL.parameters("lanomina") = numero
1410.   isrSQL.Parameters("qui_ini") = 9601
1411.   isrSQL.Parameters("qui_fin") = quincena
1412.
1413.   Set res_sel = isrSQL.OpenRecordset()
1414.   If res_sel.EOF Then
1415.     'IsNull(res_sel![Variable]) Then
1416.       montos = 0
1417.     Else
1418.       montos = res_sel![variable]
1419.     End If
1420.   clave1 = CStr(montos)
1421.   bus_val = clave1
1422.
1423.
1424. Case "INCAPA"
1425.
1426.   Set isrSQL = DBActiva.QueryDefs("ComoIncapa")
1427.   isrSQL.Parameters("empleado") = cla_emp
1428.   isrSQL.Parameters("lanomina") = numero
1429.   isrSQL.Parameters("elperiodo") = quincena
1430.   Set res_sel = isrSQL.OpenRecordset()
1431.   If res_sel.EOF Then
1432.     'IsNull(res_sel![Variable]) Then
1433.       montos = 0
1434.     Else
1435.       montos = res_sel![variable]
1436.     End If
1437.   clave1 = CStr(montos)
1438.   bus_val = clave1
1439.
1440.
1441.
1442. Case "INCAPAACUMULA"
1443.
1444.   Set isrSQL = DBActiva.QueryDefs("ComoIncapaAcumula")
1445.   isrSQL.Parameters("empleado") = cla_emp
1446.   'IsrSQL.parameters("lanomina") = numero
1447.   isrSQL.Parameters("qui_ini") = 9601
1448.   isrSQL.Parameters("qui_fin") = quincena
1449.   Set res_sel = isrSQL.OpenRecordset()
1450.   If res_sel.EOF Then
1451.     'IsNull(res_sel![Variable]) Then
1452.       montos = 0
1453.     Else
1454.       montos = res_sel![variable]
1455.     End If
1456.   clave1 = CStr(montos)
1457.   bus_val = clave1
1458.
1459.
1460. Case "DEVENGA"
1461.
1462.   Set isrSQL = DBActiva.QueryDefs("ComoDevenga")
1463.   isrSQL.Parameters("empleado") = cla_emp

```

```

1464.     isrSQL.Parameters("lanomina") = numero
1465.     isrSQL.Parameters("elperiodo") = quincena
1466.     Set res_Sel = isrSQL.OpenRecordset()
1467.     If res_Sel.EOF Then
1468.         'IsNull(res_sel![Variable]) Then
1469.             montos = 0
1470.         Else
1471.             montos = res_Sel![variable]
1472.         End If
1473.     clave1 = CStr(montos)
1474.     bus_val = clave1
1475.
1476.     Case "FAL15"
1477.         Set isrSQL = DBActiva.QueryDefs("ComoFalta")
1478.         isrSQL.Parameters("empleado") = cla_emp
1479.         isrSQL.Parameters("elconcepto") = cla_mov
1480.         isrSQL.Parameters("lanomina") = numero
1481.         isrSQL.Parameters("elperiodo") = quincena
1482.         Set res_Sel = isrSQL.OpenRecordset()
1483.         If res_Sel.EOF Then
1484.             montos = 1
1485.         Else
1486.             montos = res_Sel![variable] - 15
1487.             If montos < 0 Then
1488.                 montos = 0
1489.             Else
1490.                 montos = 1
1491.             End If
1492.         End If
1493.     clave1 = CStr(montos)
1494.     bus_val = clave1
1495.
1496.     Case "SDI"
1497.         criterio = "ClaveDeEmpleado = " & cla_emp & " and NumeroDeQuincena = " & quincena & "
1498.         and ClaveDelImpuesto = 'IMSS'"
1499.         num_cla = DLookup("MontoSDI", "SDI_BIM", criterio)
1500.         If IsNull(num_cla) Then
1501.             num_cla = 0
1502.         End If
1503.         criterio = "NumeroDeQuincena <= " & quincena & " and ClaveDeParametro = " & 14
1504.         smg_df = DLookup("MontoParametro", "Parametros", criterio)
1505.         If IsNull(smg_df) Then
1506.             smg_df = 0
1507.         End If
1508.         If num_cla > smg_df * 10 Then
1509.             num_cla = smg_df * 10
1510.         End If
1511.         clave1 = CStr(num_cla)
1512.         bus_val = clave1
1513.
1514.     Case "SUST"
1515.         consulta = ""
1516.         aux_qui = quincena
1517.         consulta = "NumeroDeQuincena = " & quincena
1518.         consulta = consulta & " and ClaveDeEmpleado = " & cla_emp

```

```

1519.      consulta = consulta & " and ClaveDeEmpresa = " & 1
1520.      montos = DLookup("MontoDeSustitucion", "Sustitucion", consulta)
1521.      If IsNull(montos) Then
1522.          bus_val = "0"
1523.      Else
1524.          clave1 = CStr(montos)
1525.          bus_val = clave1
1526.      End If
1527.
1528.      -----Calculo del PTU '98 -----
1529.      Case "FACTORDIAS"
1530.          consulta = ""
1531.          aux_qui = quincena
1532.          montos = DLookup("FactorPorDia", "DatosPTU", "")
1533.          If IsNull(montos) Then
1534.              bus_val = "0"
1535.          Else
1536.              clave1 = CStr(montos)
1537.              bus_val = clave1
1538.          End If
1539.
1540.      Case "FACTORSUELDO"
1541.          consulta = ""
1542.          aux_qui = quincena
1543.          montos = DLookup("FactorPorPeso", "DatosPTU", "")
1544.          If IsNull(montos) Then
1545.              bus_val = "0"
1546.          Else
1547.              clave1 = CStr(montos)
1548.              bus_val = clave1
1549.          End If
1550.
1551.      Case "TOTALDINERO"
1552.          consulta = ""
1553.          aux_qui = quincena
1554.          SalBase = DLookup("SalarioBase", "DatosPTU", "")
1555.          If IsNull(SalBase) Then
1556.              SalBase = 0
1557.          End If
1558.          consulta = " ClaveDeEmpleado = " & cla_emp
1559.          xTotalDias = DLookup("TotalDias", "AcumuladosPTU", consulta)
1560.          montos = DLookup("TotalDinero", "AcumuladosPTU", consulta)
1561.          If IsNull(montos) Then
1562.              bus_val = "0"
1563.          Else
1564.              If montos > SalBase * xTotalDias Then
1565.                  montos = SalBase * xTotalDias
1566.              End If
1567.
1568.              clave1 = CStr(montos)
1569.              bus_val = clave1
1570.          End If
1571.
1572.      Case "TOTALDIAS"
1573.          consulta = ""
1574.          aux_qui = quincena

```

5.3 Diseño y Construcción del Sistema.

El sistema integral de recursos humanos SIRH fue desarrollado en Access 97 lo que facilita la construcción de la interfaz, ya que cuenta con una serie de herramientas para dicho propósito.

5.3.1 Desarrollo de Pantallas

En el desarrollo del sistema se utilizó el formato estandar que proporciona Access esto con el fin de mantener uniformidad en la interfaz con el usuario. En la figura 5.3-1 se muestra parte del asistente de access para la creación de las pantallas.

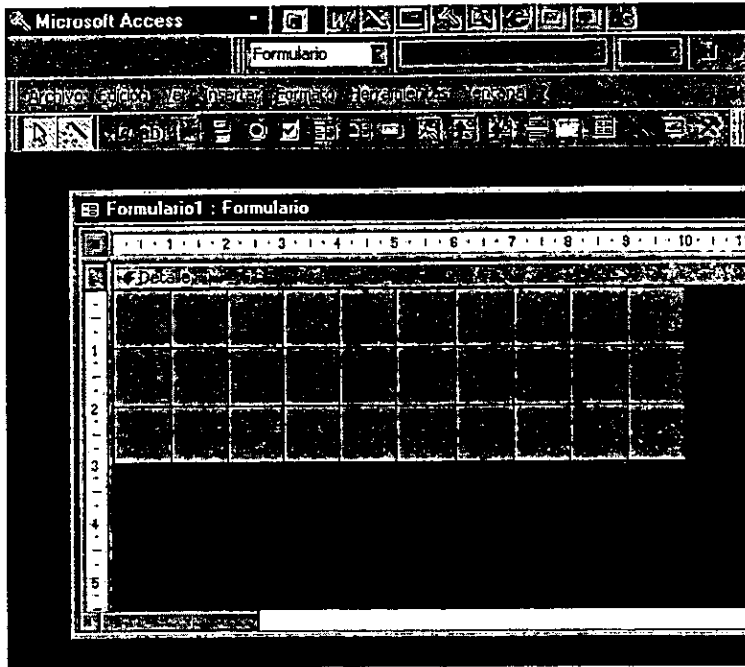


Figura 5-3- 1 Asistente de Formularios o Pantallas

Las pantallas que se utilizan en el sistema se pueden dividir en los siguientes grupos:
Presentación de Opciones

Estas pantallas son las que se utilizaron para presentar los Menús principales tal como se muestra en la figura 5.3-2 que es la segunda pantalla que ve en el SIRH. Cada botón esta asociado a alguna tarea en específico, que no requiere de parámetros

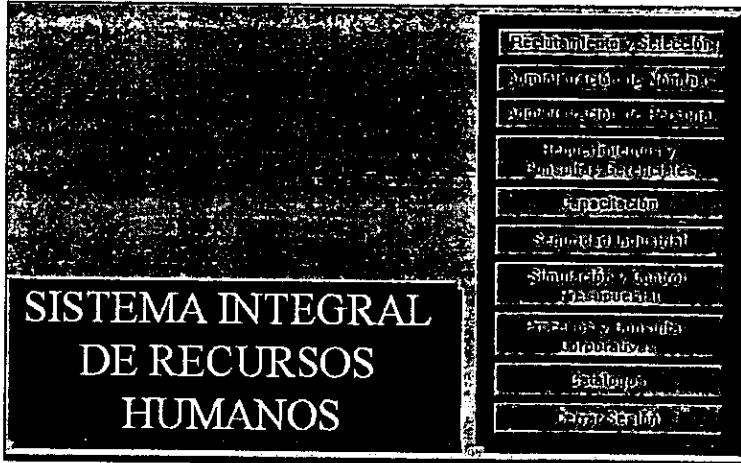


Figura 5-3- 2 Pantalla de Presentación de Opciones (zGeneral)

Solicitud de Criterios

Estas pantallas se utilizan para que el usuario proporcione datos para ejecutar un proceso o establecer criterios de búsqueda, como se puede ver en la figura 5.3-3 estas pantallas tienen asociados otros botones que disparan el evento y uno de salida.

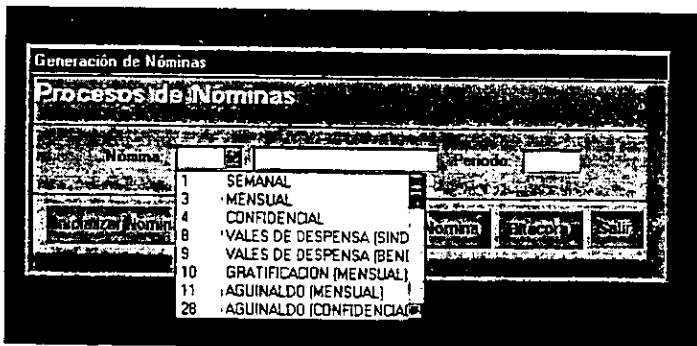


Figura 5.3- 3 Pantalla de solicitud de Criterios (CierreNomina).

En el caso de la pantalla de la figura 5.3-3 se utiliza para Inicializar, Generar y Cerrar una nómina, solicitando al usuario la Clave de la nomina y el periodo o Quincena a procesar, en todas las pantallas siempre que se solicite escoger algún dato que esta referenciado a alguna tabla o consulta se presentara un menú colgante para validar la selección.

Solo consulta.

Las pantallas de solo consulta se utilizan para presentar los datos de una consulta o tabla solo para ver o exportar los datos pero no para modificar, en la figura 5.3-4 se muestra un ejemplo de este tipo de pantalla.

EXPEDIENTE DE PERSONAL
SISTEMA INTEGRAL DE RECURSOS HUMANOS

EMPRESA: **FORJAS**

ID: **2014** NOMBRE: **DANCHEZ VIZQUEZ MANUEL A** SALARIO: **2002-01X** SOLICITUD: **4300**

DEPTO: **2** CATEGORIA: **FORJA**

PUESTO: **34** SUPERVISOR DE PRODUCCION

REGIMEN: **7** MENSUAL IMSS: **01-00-72-1440-0** CARTILLA:

LICENCIA: PASAPORTE: ESCOLARIDAD:

FECHA NACIMIENTO: EDAD: SEXO: ESTADO CIVIL:

DIRECCION: CIUDAD: COLONIA:

TEL. RESIDENCIAL: ENTIDAD: TELEFONO:

CONTRATO: **4** INGRESO: **00/1/1000** ANTIGUEDAD: **0 Años 7 Meses 3 Dias**

TIPO EVENTUAL: FECHA INI. REN: EXPIRA EL:

SALARIO DIARIO: **\$374.67** INTEGRADO: **\$200.04** SALARIO MENSUAL: **\$2,740.10**

FECHA ULTIMA EVALUACION: CALIFICACION: FECHA ULTIMA MODIFICACION SALARIO: **01/01/2000**

Figura 5.3- 4 Pantalla de solo consulta (Expediente de Personal)

Este tipo de pantalla siempre tendrá un botón de salida uno de buscar y las barras de desplazamiento.

Administración de Datos.

Las pantallas del tipo Administración de datos son utilizadas para las cuatro operaciones Altas, Bajas, Cambios y Consultas. Como se muestra en la figura 5.3-5 pueden estar incluidas varias sub-pantallas asociadas a datos maestros.

SISTEMA INTEGRAL DE RECURSOS HUMANOS

Empleados

CONTRATO	200014	CERVANTES F. HERIBERTO
DEPARTAMENTO	44	CONTROL DE CALIDAD
PUESTO	01	INSPECTOR/CONTROL CALIDA
PERIODO	3	MENSUAL
SUBSECTOR	1	TLAXCALA
FECHA INGRESO	01/01/1999	9901
SALARIO	\$170.96	\$5,128.00

PERIODO	ESTADO	TIPO
01/01/2000	A	INFO
01/01/2000	A	IMSS

Botones: **STAD HISTORICO**, **BOLETA**, **Salir**

Figura 5.3- 5 Pantalla de administración de datos (Empleados)

5.3.2 Desarrollo de informes.

Los informes o reportes se desarrollan utilizando el asistentes que proporciona access como se muestra en la figura 5.3-6. Sin embargo para la mayoría de los reportes que se requiere impresión rápida se configuró un tipo estándar con este arreglo se pueden imprimir en matriz y lasser. En la figura 5.3-7 se muestra el encabezado que todos los reportes deben tener.

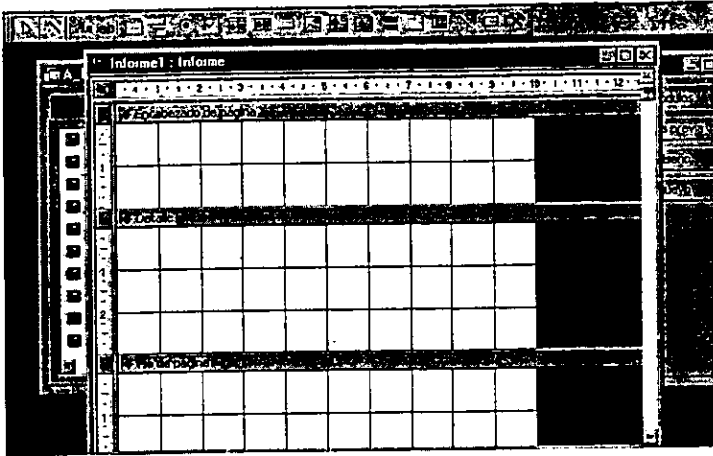


Figura 5.3- 6 Asistente de informes

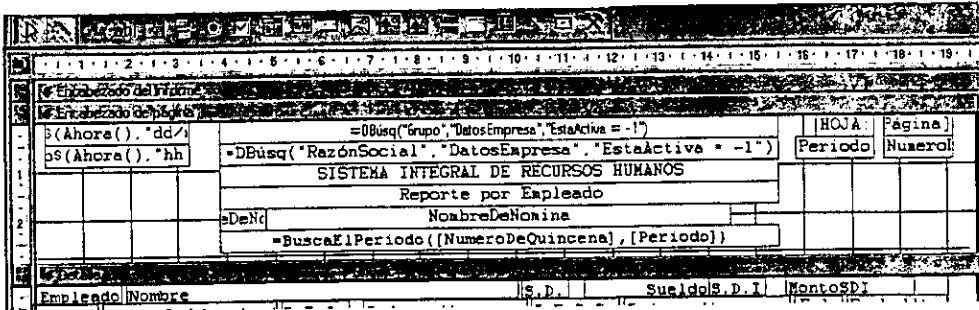


Figura 5.3- 7 Encabezado estándar de los reportes (PreNominaEmp)

La filosofía de los reportes esta basada en tener un conjunto reducido de reportes que pueden exportarse a las herramientas de Excel, Word o PowerPoint, para tal efecto se procuró en lo más posible establecer la distancia adecuada entre las columnas.

5.3.3 Desarrollo de Consultas.

Tal vez el elemento más poderoso de access es su asistente de consultas, que pueden utilizarse en el código, pantallas, y reportes; en la figura 5.3-8 se muestra el asistente.

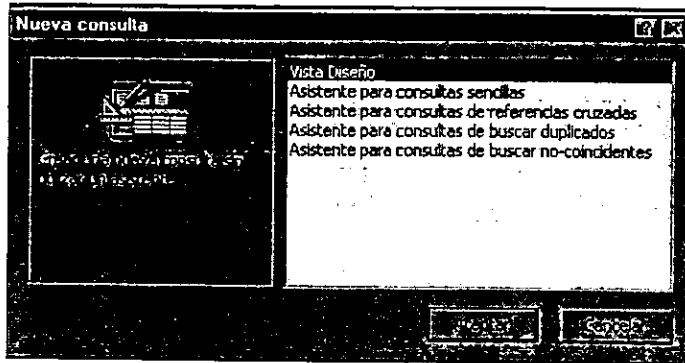


Figura 5.3- 8 Asistente de Consultas.

La interfaz para generar consultas permite asociar llamadas de funciones en la consultas un ejemplo se muestra en la figura 5.3-9.

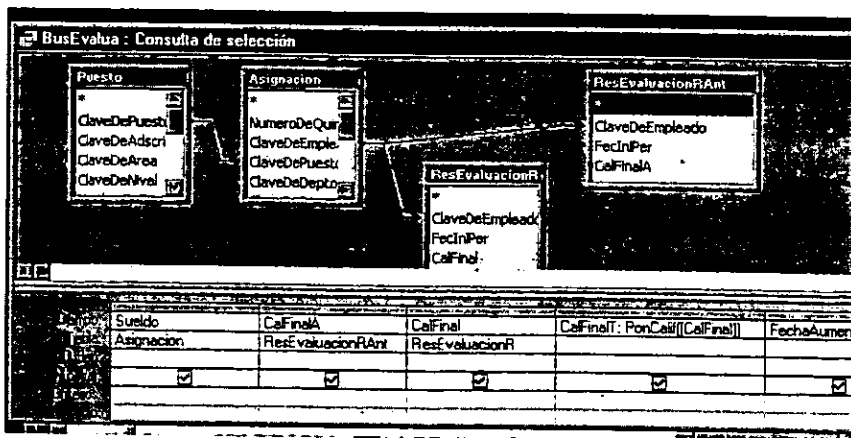


Figura 5.3- 9 Ejemplo de consultas con funciones (BusEvaluat)

5.3.4 Administración de Tablas.

La administración de tablas se puede hacer mediante el asistente como se muestra en la figura 5.3-10 y en la figura 5.3-11.

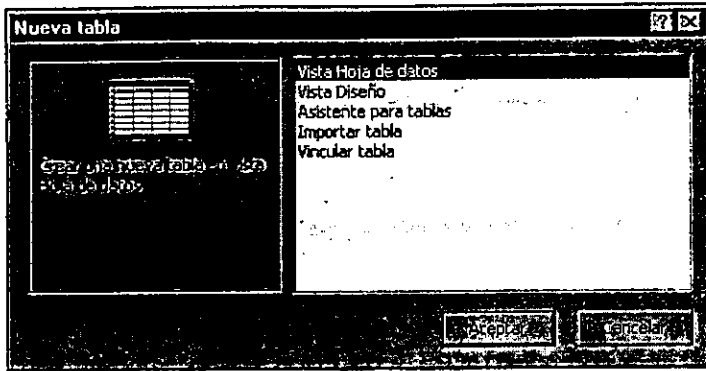


Figura 5.3- 10 Asistente de Tablas.

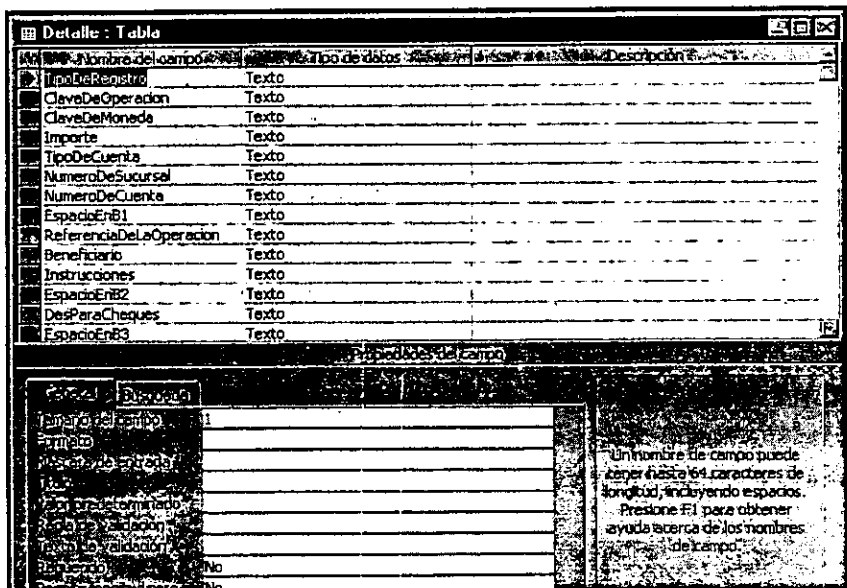


Figura 5.3- 11 Manipulacion de Tablas

5.3.5 Arquitectura del Sistema.

Para poder mejorar el rendimiento y hacer mas sencillo el mantenimiento de los programas se decidió separar los datos de la aplicación es decir una base de datos para consultas, código, pantallas, informes y macros, otra u otras para datos. En la figura 5.3-12 se muestra la estructura de archivos del cliente del sistema.

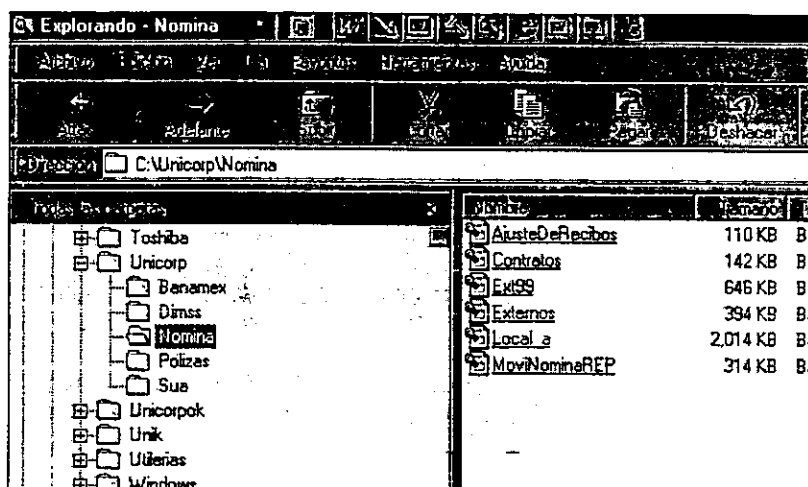


Figura 5.3- 12 Estructura del directorio Cliente.

En la carpeta Nomina se ponen archivos temporales que se utilizan para reportes o algún proceso en específico y disminuir el tráfico en la red. En la figura 5.3-13 se muestra la estructura del servidor.

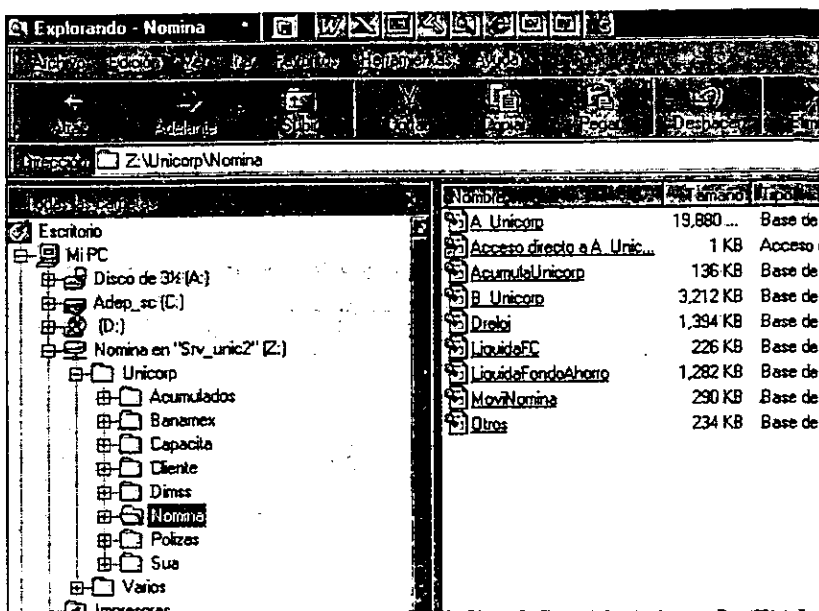


Figura 5.3- 13 Estructura de Servidor

5.4 Diseño e Implementación de rutinas de diagnostico y evaluación de los datos.

5.4.1 Evaluación de rutas instaladas.

Debido a que la arquitectura del sistema esta fundamentada en instalar una parte en el cliente y otra en el servidor es importante saber en cualquier momento el direccionamiento de las tablas, para este efecto Access cuenta con un asistente que se muestra en la figura 5.4-1.

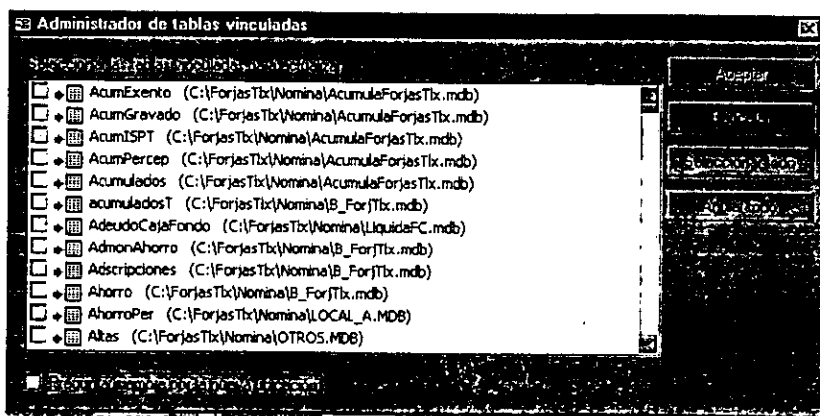


Figura 5.4- 1 Administrador de Tablas Vinculadas

Sin embargo resulta muy tardado el usarlo por lo cual se desarrollo un programa que indica estas rutas y permite cambiarlas de manera más sencilla y con menos posibilidad de error, este modulo se encuentra dentro del sistema.

Esta rutina esta diseñada para buscar las rutas actuales y presentar la alternativa de indicar la nuevas rutas, así como mostrar un reporte con el total de tablas y direcciones, una vez que se indican las nuevas rutas se ejecuta el proceso de verificarlas.

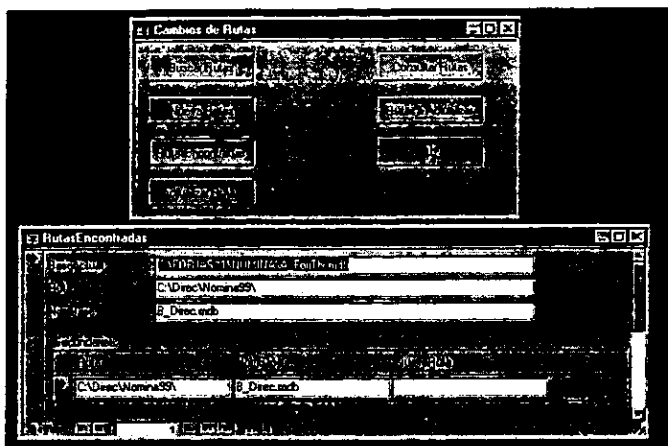


Figura 5.4- 2 Administrador de Rutas (SIRH)

5.4.2 Reparación y Compactación de Datos.

Access provee una herramienta para corregir errores físicos en los archivos de las bases de datos pero se corre el riesgo de ejecutarlo erróneamente por lo que diseño una rutina sencilla de utilizar el menú principal se muestra en la figura 5.4-3.

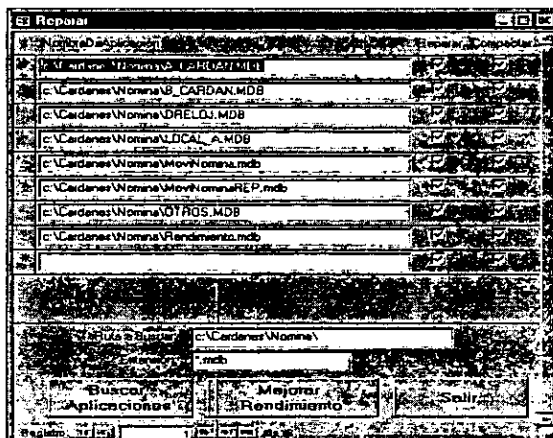


Figura 5.4- 3 Rendimientos de la Base de Datos

5.4.3 Pruebas para el año 2000.

Para el funcionamiento adecuado en el manejo de fechas se estableció como máscara de captura 99/99/0000;0;_ , la cual obliga a capturar 4 dígitos y asegurar guardarlos.

El código que haga referencias a fechas deberá usar dd/mm/yyyy para asegurar el manejo de cuatro dígitos.

Las nominas se numeran en base a la cantidad de procesos se hagan, por ejemplo las mensuales son 12, quincenas 24, semanales 52 entre otras, para indicar estos periodos se tomo la siguiente convención: Para periodos antes del 2000 AAPP es decir los dos últimos dígitos del año (AA) y los dígitos del periodo (PP), por ejemplo para el mes de enero de 1999 en una nómina semanal será 9912 para el mismo caso pero del 2000 será 20001 (20AAPP)

5.4.4 Evaluación del resultado de una nómina.

La nómina es el proceso que más cálculos involucra en volumen y en complejidad. Para hacer pruebas sobre este resultado se recomienda que se sigan estas recomendaciones.

Para revisar Detalle.

Seleccionar empleados que sean:

- a) de los mas bajos en sueldo
- b) de la media en sueldo
- c) de los mas altos sueldos
- d) con faltas
- e) incapacidades

f) Tiempo extra

Calculando esto se puede evaluar el comportamiento con respecto a los topes (vales y fondo), así como las aportaciones al IMSS, Infonavit SAR y el Impuesto (ISR)

Para revisar totales.

- 1) Imprimir el reporte de nómina y el contador de empleados debe ser igual al numero de empleados que se tengan
- 2) Generar deposito bancario el neto debe ser igual al del reporte de nómina.
- 3) Generar póliza contable que debe cuadrar y el neto debe ser igual al del deposito.
- 4) Imprimir reporte de acumulados antes y después de cerrar la nómina.

Si los puntos anteriores son correctos no hay movimientos improcedentes y la nomina es correcta.

5.5 Integración y pruebas del sistema.

Las pruebas constituyen una parte integral y vital del ciclo del desarrollo de sistemas. Se realizan con el propósito de descubrir defectos y se establecen para mejorar la calidad del sistema.

Las pruebas permiten:

- Sentar las bases para determinar los objetivos y un plan específico de pruebas.
- Asegurar la obtención y formalización de los requerimientos del usuario y verificar que son adquiridos de una manera completa, correcta y constante.
- Verificar los requerimientos funcionales así como los estructurales y establecerlos con fundamento para realizar las pruebas del sistema.
- Detectar y registrar defectos asociados a los requerimientos establecidos.
- Establecer la documentación de reportes para pruebas realizadas.

Las pruebas pueden clasificarse en:

1. Unitarias: Se realizan sobre un programa o modulo con la intención de encontrar problemas funcionales en la lógica, así como problemas técnicos en el código, esta prueba centra el proceso de verificación en la menor unidad del diseño del software.
2. Integración: Se realizan a un grupo de programas para asegurar que los datos y controles sean pasados adecuadamente ente controles.
3. Regresión: Se utilizan para detectar fallas que pudieran presentarse en algún modulo como consecuencia de modificaciones realizadas.
4. Volumen: Se realizan para verificar el funcionamiento adecuado y eficiente de una aplicación bajo condiciones extremas de operación.

5. Caja Negra: Se realizan con base en los requerimientos sin conocimiento sobre cómo fue construido el sistema y usualmente dirigidas a los datos.
6. Caja Blanca: Pruebas basadas en el conocimiento sobre la lógica y estructura interna, usualmente dirigidas a la lógica.
7. Aceptación del Usuario: Realizadas por el usuario final para asegurar que el sistema satisfaga las necesidades requerida.
8. Estáticas: Revisión y validación de los documentos generados en las distintas fases de la vida de un proyecto.
9. Funcionales: Validar los requerimientos de la organización (lo que se supone que el sistema debe hacer).
10. Estructurales: Validar la arquitectura del sistema confirmando que todas sus partes funcionen de manera armónica y que la tecnología este siendo usada apropiadamente. Se refieren a las características técnicas, como su comportamiento con grandes volúmenes de información, tiempos de respuesta etc.

5.5.1 Flujo de información en la prueba.

El esquema descrito en la figura 5.5-1 muestra el flujo de información para una prueba. Se proporcionan dos clases de entradas al proceso de prueba:

1. Configuración del software, la especificación del diseño y el código fuente.
2. Configuración de prueba que incluye un plan y procedimiento de prueba.

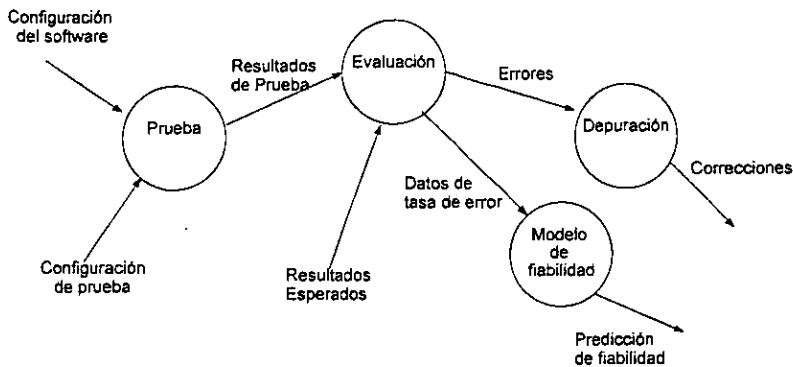


Figura 5.5- 1 Flujo de información en la prueba.

Se llevan a cabo las pruebas y se evalúan los resultados. O sea, se comparan los resultados de la prueba con los esperados. Cuando se descubren datos erróneos implica que hay un error y comienza la depuración. El proceso de la depuración es la parte más impredecible del proceso de prueba. Un error que indique una discrepancia de un 0.01 por 100 entre los resultados esperados y los reales puede llevar una hora, un día o un mes de diagnóstico y corrección. Es la inherente incertidumbre de la depuración lo que hace difícil planificar la prueba de forma eficaz.

5.5.2 Verificación y validación.

La prueba del software es un elemento de un concepto más amplio que, a menudo, se referencia como verificación y validación (V&V). La verificación se refiere al conjunto de actividades que aseguran que el software implementa correctamente una función específica. La validación se refiere a un conjunto diferente de actividades que aseguran que el software construido se ajusta a los requisitos del cliente, lo anterior se puede expresar de la siguiente manera:

Verificación: "¿Estamos construyendo el producto correctamente?"

Validación: "¿Estamos construyendo el producto correcto?"

5.5.3 Aplicación de pruebas a sistema.

Pruebas al cada modulo.

Los datos de prueba que se aplicaron a los módulos se definieron de manera conjunta entre el usuario y el equipo de diseño.

Ajuste anual de impuestos.

Los datos de entrada que tiene este modulo son:

Dato de entrada	Descripción
Bispt	Base de impuesto que se acumulo durante el año.
Bcredito	Base de crédito que se acumulo durante el año
ISPTAnual	Impuesto generado aplicando el procedimiento anual

Los datos de salida son:

Dato de salida	Descripción
DevISPT	Impuesto a cargo o a favor
DevCredito	Crédito a cargo o a favor

La combinación de resultados pueden ser:

ISPT Anual	Bispt	Bcredito	DevISPT	DevCredito
<0	No importa	<ISPTAnual	$(-Bispt) - (BCredito - ISPTAnual)$	0
<0	No importa	>ISPTAnual	-Bispt	$ISPTAnual - Bcredito$
<0	No importa	= ISPTAnual	-Bispt	0
>=0	No importa	No Importa	$ISPTAnual - Bispt - BCredito$	0

Con base en estas tablas e pueden definir los datos a probar y los resultados a esperar. El valor de ISTPAnual proviene de un modulo externo el cual escoge el monto menor de los dos impuestos que calcula.

Salario Diario Integrado.

Para la prueba del modulo del salario diario integrado se aplico el esquema de caja negra, haciendo énfasis en los datos de salida , para los empleados denominados como confidenciales con los sueldos altos sin importar el monto de los variables siempre serán topados.

Otro punto a probar fue la conformación de la base integrable, para lo cual se establecieron datos simulando cada uno de los tipos de integración y validar que realmente se sumen en el rubro adecuado.

Tiempo Extra.

Para la prueba de este modulo se aplicó el esquema de caja negra para efectos de validar el correcto funcionamiento del proceso en general, sin embargo era necesario por la complejidad del modulo aplicarle también el esquema de caja blanca es decir datos que pusieran en evidencia los puntos importantes del modulo, estos son:

- Tiempo extra que inicia en un turno y termina en otro, para verificar la correcta asignación de costo por turno.
- Tiempo extra que inicia un día (en la noche) y termina al otro en la mañana, para verificar el correcto cambio de fecha.
- Tiempo extra que se trabaja con un sueldo distinto al que tienen en el momento de generación de nómina.
- Tiempo extra de periodos atrasados.

Volumen de información.

Para garantizar el funcionamiento del sistema con respecto al incremento en el volumen de información se insertaron datos simulando 12 meses de nóminas y con un 100% excedente del máximo de personas de cada planta.

Pruebas con el usuario final.

El sistema esta compuesto de módulos específicos para cada actividad, por lo cual se procedió a probar cada modulo con el usuario correspondiente, estas pruebas se realizaron para verificar:

- El correcto funcionamiento con base en las especificaciones.
- El funcionamiento de la interfaz.

Pruebas con el sistema integrado.

Una vez que se probó de manera individual se procedió a la prueba integral del sistema, esta prueba se baso en generar un proceso de nómina paralelo al que se utilizaba en ese momento, la prueba en paralelo no tenía como objetivo comparar resultados entre ambos sino en evaluar el resultado del nuevo sistema con base los nuevos esquemas, la comparación serviría para identificar y comprobar la correcta participación de los usuarios en la nueva dinámica.

Pruebas del corporativo.

Una de las razones por las que se decidió realizar un nuevos sistemas fue para afrontar el problema del año 2000 por lo cual el sistema fue sometido a una serie de pruebas por parte del departamento de sistemas del corporativo, estas pruebas consistieron en :

- Verificación manual de las mascararas de captura de fechas con 4 dígitos en el año.
- Simular el proceso de la primera semana de enero del 2000.
- Aplicar una prueba al código del sistema con el software OnMark ver figura 5.5-2

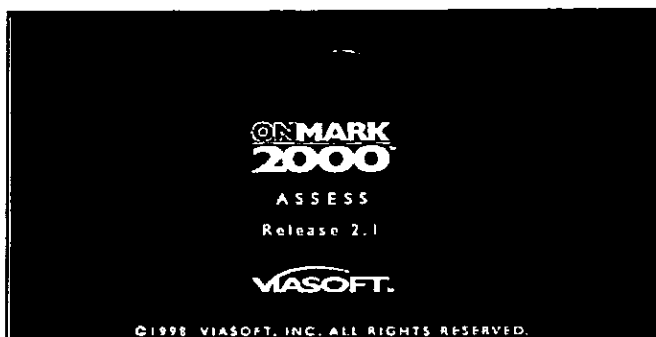


Figura 5.5- 2 Software de prueba de código año 2000

5.6 Factibilidad técnica operativa.

El sistema fue desarrollado para ser instalado en un conjunto de 11 empresas que reportan a un corporativo, sin embargo debido a negociaciones con el sindicato existen algunas diferencias que deben ser analizadas en el momento de la implementación.

5.6.1 Requerimientos de Hardware.

Los requerimientos de equipo que se solicitan para la instalación del sistema en cada empresa es el siguiente:

Servidor	Red	Clientes	Impresoras
Windows NT	Alambrica	Windows 95 o posterior	Matriz de Puntos (DFX-5000+)
Pentium 400 Mhz		Access 97 ver. Español	
RAM 128Mb		Pentium 400Mhz	Lasser
Disco Duro de 1GB		64 MB en RAM	
Tarjeta de Red.		Disco duro de 300 MB Libres	
		Tarjeta de Red 10/100	

El equipo con que cuentan las empresas satisfacen y en algunos casos exceden los mínimos requeridos no siendo esto limitante para la implementación del sistema.

5.6.2 Requerimientos de Información

Para poder iniciar el proceso de implementación del sistema se tuvo que solicitar a cada empresa información para poderla transferir evitando al máximo la captura inicial, para poder agilizar esta actividad se solicito dicha información con el formato siguiente.

Tipo : Texto delimitado

Delimitador de texto: Ninguno

Delimitador de campo: coma (,)

Datos Generales

Campo	Tipo	Observaciones
Código	Numérico	Código del trabajador
Nombre	Texto	Nombre del trabajador
RFC	Texto	RFC del trabajador
Imss	Texto	Numero de afiliación
Calle	Texto	Calle y numero
Colonia	Texto	Colonia
CP	Texto	Código Postal
Municipio	Texto	Municipio o Delegación
Entidad	Texto	Entidad Federativa
Teléfono	Texto	Teléfono particular
EdoCivil	Texto	Soltero(a) Casado(a) Otro
Sexo	Texto	Masculino

		Femenino
Nacionalidad	Texto	Mexicana Otra
Lugar De Nacimiento	Texto	Estado en que nació

Datos para Nómina

Campo	Tipo	Observaciones
Código	Numérico	Código del trabajador
Proceso	Numérico	Clave Del proceso 1 Sindicalizados 2 Empleados (Quincenal) 3 Empleados (Mensual) 4 Confidencial 53 Fin. Sindicalizados 55 Fin. Empleados 56 Fin. Confidencial
Fecha De Alta	Fecha(dd/mm/yyyy)	Fecha de Ingreso al Grupo
Fecha Transferencia	Fecha(dd/mm/yyyy)	Fecha de Ingreso a la Planta
Tipo De Contrato	Texto	Eventual Obra Eventual Tiempo Planta
Fecha Renovación	Fecha(dd/mm/yyyy)	Fecha de renovación de Contrato Eventual
Depto	Numérico	Clave Del Depto al que esta asignado
Puesto o Categoría	Numérico	Clave del Puesto o categoría al que esta asignado.
Sueldo Diario	Numérico	Percepción Diaria
Sueldo Diario	Numérico	Sueldo de cotización al IMSS.

Integrado		
Porcentaje de Penetración	N Numérico	Solo para empleados, es el % de penetración conforme al nivel.
Clave De Turno	N Numérico	Identificador del turno

Acumulados

Campo	Tipo	Observaciones
ClaveDeNomina	N Numérico	Proceso de Nomina
ClaveDeEmpleado	N Numérico	Clave del trabajador
ClaveDeConcepto	N Numérico	Clave del Concepto de Nómina
Descripción	T Texto	Descripción deConcepto
Variables	N Numérico	Monto de Variables p.e. del concepto 1 el total de días
Monto	N Numérico	Monto del Concepto

Acumulado de Infonavit

Campo	Tipo	Observaciones
ClaveDeEmpleado	N Numérico	Clave del trabajador
MontoAcumulado	N Numérico	Monto Acumulado de Infonavit

Acumulados de Caja y Fondo

Campo	Tipo	Observaciones
ClaveDeNomina	N Numérico	Proceso de Nomina
ClaveDeEmpleado	N Numérico	Clave del trabajador
ClaveDeConcepto	N Numérico	Clave del Concepto de Nómina 279 Caja Empleados 285 Caja Sindicalizados 290 Fondo de Ahorro Empleados y Sindicalizados

PuntosAcumualdos	Numérico	Acumulado de Puntos, por la aportación
MontoAcumulados	Numérico	Monto Acumulado, solo parte del empleado
Aportacion	Numérico	Aportación por periodo, solo si es fija, si depende de las incidencias se deja en cero.

Catalogo de Departamentos

Campo	Tipo	Observaciones
Depto	Numérico	Clave Del Departamento
Descripción	Texto	Nombre del Departamento

Catalogo de Puestos y Categorías

Campo	Tipo	Observaciones
Puesto o Categoría	Numérico	Clave Del Puesto o Categoría
Descripción	Texto	Nombre del Puesto o Categoría

Esta información tuvo que ser procesada por cada empresa ya que si bien su contaba con ella se tuvo que obtener de diferentes bases ya que no están integradas en una sola, incluso tuvo que usarse programas externos como el SUA, (Sistema Único de Auto Determinación) que proporciona el IMSS.

5.6.3 Cambios y Adecuaciones.

El sistema debió sufrir algunas adecuaciones en cada implementación debido a que algunos cálculos no podían cambiarse por ser propios del sindicato, pero siempre se procuró que fueran los menos, para este tipo de situaciones se contó con la colaboración de los responsables corporativos del área.

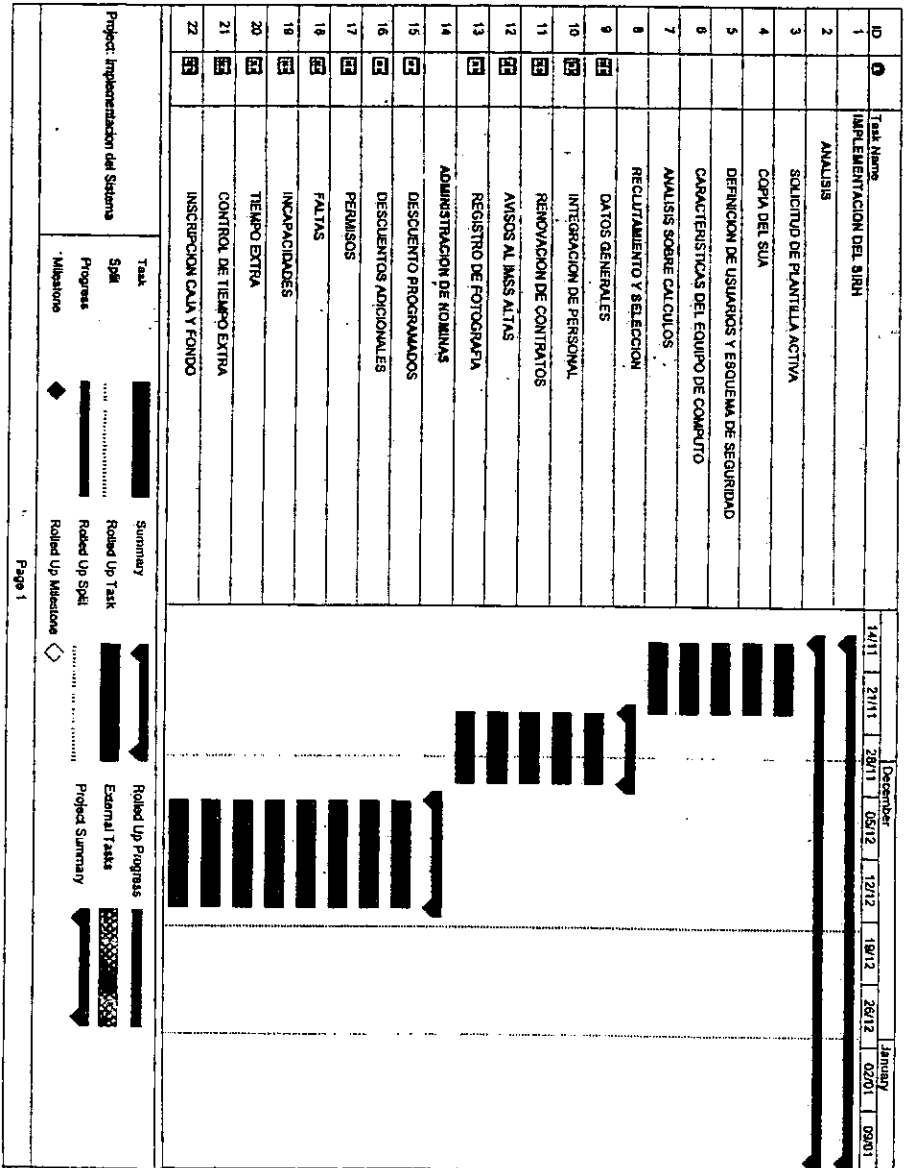


Figura 5.6- 1 Programa de Implementación.

ID	Task Name	14/11	21/11	28/11	05/12	12/12	19/12	26/12	January
23	PRESTAMOS CADA Y FONDO								
24	GENERACION DE NOMINA								
25	VALIDACION DE NOMINA								
26	DEPOSITO BANCARIO								
27	IMPRESION DE NOMINA								
28	IMPRESION DE RECIBOS								
29	REPORTES DE NOMINA								
30	GENERACION DE POLIZA CONTABLE								
31	VALIDACION								
32	PROCESO DE INTEGRACION								
33	VALIDACION								
34	GENERACION DE ARCHIVO DE MODIFICACIONES DE SALARIO								
35	GENERACION DE ARCHIVO MOVIMIENTOS AL SUA								
36	VALIDACION								
37	ADMINISTRACION DE PERSONAL								
38	MOVIMIENTOS DE PERSONAL								
39	CONTROL DE VACACIONES								
40	REGISTRO DE CUENTAS BANCARIAS								
41	CALCULO DE FINANCIOS								
42	REPORTES DE PERSONAL								
43	CATALOGOS								
44	ADMINISTRACION DE CATALOGOS								

Task	Spik	Progress	Milestone	Summary	Roller Up Task	Roller Up Spik	Roller Up Milestone	Roller Up Progress	External Testis	Project Summary

Project: Implementacion del Sistema

Page 2

Figura 5.6- 2 Programa de Implementación.

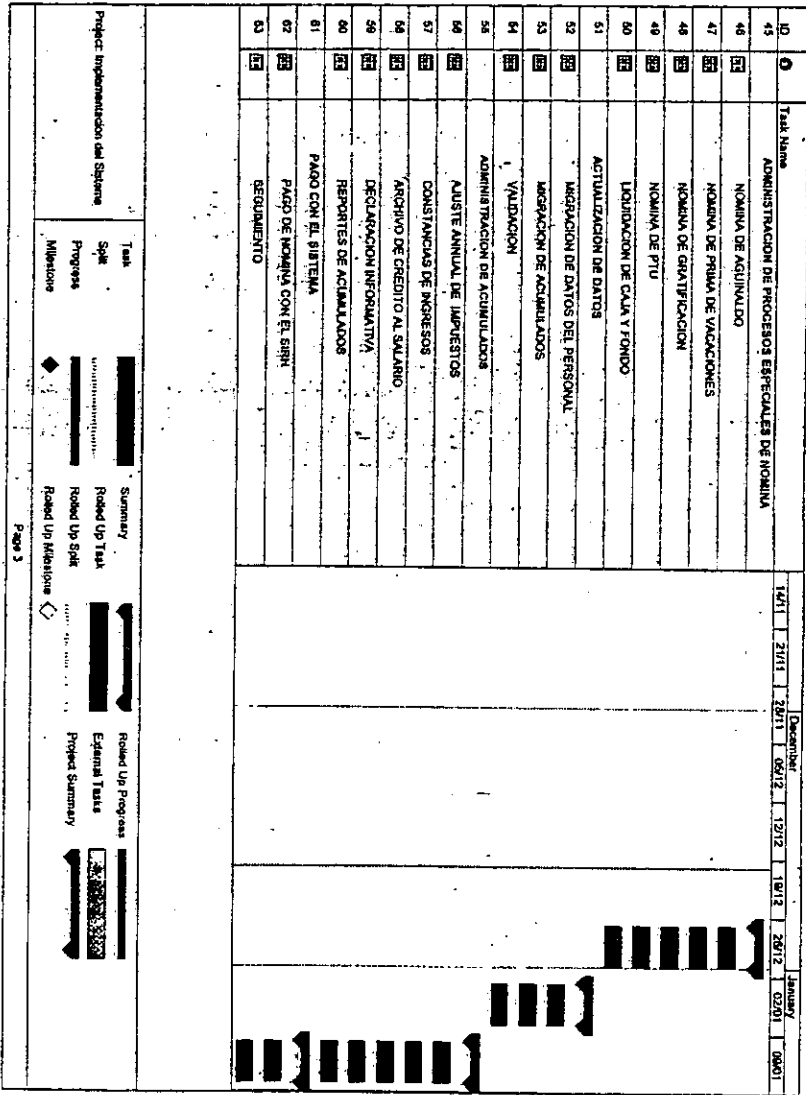


Figura 5.6- 3 Programa de Implementación

Programa de implementación.

En las figuras 5.6-(1,2,3) se muestra el programa de implementación que se siguió en cada empresa, el cumplimiento de estas actividades en tiempo y forma fue fundamental para el éxito del proyecto. El tiempo puede variar dependiendo del tamaño de la empresa y de su disponibilidad.

Al momento de realizar el presente trabajo se tenía concluido al 100% el proceso de implementación en las 12 empresas en cuestión y se comenzaba con la fase de mantenimiento.

5.7 Análisis del costo y venta del sistema.

El costo total del sistema puede realizarse con base en los siguientes puntos:

- Número de líneas de código.
- Hora – Hombre (Analista, Diseñador, Programador)
- Por Proyecto

El costo de una línea de código oscila entre los 20 y 30 dólares. Para el Sistema Integral de Recursos Humanos son aproximadamente 10,000 líneas de código.

El costo podría calcularse tomando a 25 dólares por línea, para obtener un costo de 250,000 dólares, al tipo de cambio actual.

El costo Hora - Hombre se calcula con base al tiempo que invierte cada persona que participa en el proyecto, se puede establecer un costo por persona como se muestra en la siguiente tabla.

Tipo	Costo x (Día de 8 Horas) Dólares
Líder de Proyecto	900
Programador Senior	800
Especialista Administrativo	600
Programador Junior	550

Figura 5.7- 1 Tabla de costo Hora-Hombre.

En el desarrollo del sistema integral de recursos humanos se invirtieron 4 meses (22 días promedio por mes) distribuidos de la siguiente manera.

Número	Tipo	Costo x (Día de 8 Horas) Dólares	Meses (22 días por mes)	Costo Total Dólares
1	Lider de Proyecto	900	4	79,200
1	Programador Senior	800	4	70,400
1	Especialista Administrativo	600	4	52,800
1	Programador Junior	550	4	48,400
				250,800

Figura 5.7- 2 Análisis de costos por hora-hombre

Con base en las estimaciones anteriores se puede calcular que el costo del proyecto oscila entre los 250,000 y 260,000 dólares.

Costo del Proyecto	260,000
--------------------	---------

El cliente para la cual se desarrolló el sistema tiene como política adquirir totalmente los derechos de los programas realizados, es decir, requiere que le sea entregado la totalidad del código del sistema, manuales de usuario y manuales técnicos.

Con base al costo y los requerimientos del cliente se estableció un esquema de facturación por planta de acuerdo a los cambios del sistema base con un ilimitado número de licencias por planta, es decir se facturo aproximadamente 40,000 dólares por planta para cubrir los costos de instalación, capacitación, adecuación y puesta en marcha del sistema.

Conclusiones.

- El sistema Integral de recursos humanos cubrió las expectativas de información que se plantearon al principio.
- La implementación del Sistema Integral de Recursos Humanos en las 12 empresas de referencia se concluyó satisfactoriamente en el tiempo convenido.
- Se ha reducido en un 60% el tiempo de proceso de una nómina.
- Se ha reducido al 0% las observaciones por auditores.
- Se eliminó la participación del área de sistemas en los procesos normales de nómina para enfocarla hacia el soporte.
- Se proporcionó una herramienta para facilitar el manejo de la información, sin embargo la capacitación del usuario en su área (impuestos, pagos al imss, prestaciones etc) debe ser fundamental para el correcto uso de la herramienta.
- El sistema se desarrolló bajo la premisa de "*...quien genera la información la registra y quien requiere la información la obtiene..*", es decir que no debe existir la múltiple captura ni dificultad para obtenerla.
- Access 97 es una herramienta de desarrollo con la que se pueden realizar aplicaciones profesionales de una manera sencilla y con un rendimiento aceptable; además de ser escalable, es decir, a un mayor rendimiento del hardware mayor rendimiento de la aplicación.
- Las normas legales que rigen la relaciones laborales no son claras y están expuestas a la interpretación y jurisprudencia, además de que en algunos casos se

Bibliografía

1. C.J. Date. "Introducción a los sistemas de bases de datos Volumen 1", Quinta Edición, Addison- Wesley Iberoamericana, 1993. 860 pp.
2. Gary W. Hansen. "DataBase Management And Desing", Segunda Edición, Prentice Hall, 1996. 582 pp.
3. Tom De Marco. "Structured Analysis and System Spicification", Prentice Hall, 1979 347 pp.
4. Roger S. Pressman. "Ingeniería del Software" Un enfoque practico, Tercera Edición, Mc. Graw Hill, 1993. 824 pp.
5. Andrew S. Tanenbaum. "Redes de Computadoras", Tercera Edición, Prentince Hall, 1997. 813 pp.
6. Cisco Systems. "Tecnología de Interconectividad de redes", Prentice Hall, 1998. 716 pp.
7. Ari Kaplan. "NT5 La próxima Revolución", Paraninfo, 1999, 480 pp.
8. Willian R. Vaughn. "Visual Basic & SQL-Server", Quinta Edición, Microsoft Press, 1997. 784 pp.
9. Timothy M. O'Brien. "Microsoft Access 97", Developer's Handbook, Microsoft Press, 1997 596 pp.
10. G. Velázquez Mastretta. "Administración de los Sistemas de Producción", Quinta Edición, Limusa Grupo Noriega Editores, 1992. 290 pp.
11. Guillermo Sander. "Productividad y Salarios", Primera Edición, Editorial Iniza, 1995, 111 pp.
12. Carmen Cardenas Peña. "Pago de Impuestos en Español 1999", Novena Edición, Rocar Ediciones, 1999. 360 pp.
13. Constitución Política de los Estados Unidos Mexicanos, 1917, Reformada en diciembre de 1994.
14. LEY DEL INSTITUTO DEL FONDO NACIONAL DE LA VIVIENDA PARA LOS TRABAJADORES, Publicada en el Diario Oficial de la Federación el día 24 de abril de 1972.

15. LEY FEDERAL DEL TRABAJO, Publicada en el Diario Oficial de la Federación el día 1 de abril de 1970.
16. LEY DEL IMPUESTO SOBRE LA RENTA, vigente a partir del 1o. de enero de 1981.
17. LEY DEL SEGURO SOCIAL, publicada en el Diario Oficial de la Federación el 21 de diciembre de 1995.

MT-1 Manual Técnico.

MT-1.1 Descripción del Sistema.

El Sistema Integral de Recursos Humanos (SIRH) cuenta con los siguientes módulos:

Nómina	
Reclutamiento y Selección	<ul style="list-style-type: none"> • Contratación • Datos Generales • Administración de Contratos • Fotografías
Administración de Nóminas	<ul style="list-style-type: none"> • Control de Asistencia y Tiempo Extra • Control de Ahorros • Incidencias de Nómina • Generación de Integrados • Depósitos Bancarios • Póliza Contable • Reportes de Nominas, Recibos, Vales. • Archivo Fondo de Ahorro • Datos SUA • Archivo IMSS
Administración de Personal	<ul style="list-style-type: none"> • Aumentos de Sueldos <ul style="list-style-type: none"> • Directos • Por Evaluación • Gratificación Anual • Finiquitos • Reportes de Personal, Avisos IMSS
Catálogos	

2. Acumulados	
	<ul style="list-style-type: none"> • Administración de Acumulados • Constancias de Percepciones • Ajuste Anual de Impuestos • Declaración de Impuestos

3. Liquidación de Ahorro	
	<ul style="list-style-type: none"> • Calculo de Liquidación de Ahorro • Generación de Recibos de Liquidación • Ajuste de Acumulados de Ahorro

4. Presupuesto	
	<ul style="list-style-type: none"> • Calculo de Presupuesto • Matriz de Horas Disponibles • Póliza de Presupuesto

5. Capacitación	
	<ul style="list-style-type: none"> • Control de Cursos y Programa de capacitación • Control de Evaluaciones al desempeño • Constancias de capacitación

6. Servicio Médico	
	<ul style="list-style-type: none"> • Control de exámenes médicos • Control de incapacidades

7. Captura en Planta	
	<ul style="list-style-type: none"> • Control de incidencias

8. Gerencial	
	<ul style="list-style-type: none"> • Consultas de Personal

MT-1.2 Instalación del SIRH.

MT-1.2.1 Requerimientos Técnicos.

Servidor	Red	Clientes	Impresoras
Windows NT Pentium 400 Mhz RAM 128Mb Disco Duro de 1GB Tarjeta de Red.	Alambrica	Windows 95 o posterior Access 97 ver. Español Pentium 400Mhz 64 MB en RAM Disco duro de 300 MB Libres Tarjeta de Red 10/100	Matriz de Puntos (DFX-5000+) Lasser

MT-1.2.2 Tipos de instalación.

El sistema fue diseñado para operar en un ambiente de red, y por lo tanto compartir de manera simultanea con distintos usuarios y equipos los programas y datos. Sin embargo dependiendo del tipo de red y de su velocidad se pueden hacer dos tipos de instalaciones, que se describen en la tabla siguiente, las cuales no restringen el funcionamiento multi usuario.

Local.	Remota
<ul style="list-style-type: none">• La instalación Local se recomienda cuando se cuenta con un excesivo trafico en la red, ya sea por baja velocidad o fallas en la arquitectura.• En esta arquitectura se instalan programas y datos en el mismo equipo (Cliente) y de este se conectan los demás usuarios.	<ul style="list-style-type: none">• La instalación Servidor se recomienda cuando la velocidad de la red es a 100 Mbs o mayor y el tráfico es correctamente administrado.• En esta arquitectura se instalan los datos y algunos programas en el servidor y en el cliente solo algunos archivos que sirven como accesos temporales.

MT-1.2.3 Instalación del Servidor.

El SIRH se distribuye en un CD que contiene los programas y bases de datos necesarios.

1. Definir tipo de instalación.

2. Modificar en Windows:

Panel de Control->Configuración Regional->

Fecha : El formato de fecha corta a dd/mm/yyyy

Hora: El formato de hora a: HH:mm:ss

3. Instalar Access 97 ver. Español (No instalar en el Servidor)

4. Insertar el CD de distribución.

5. Ejecutar el programa Instalar.exe y seguir las instrucciones.

La función principal de este programa de instalación es de copiar todos los archivos necesarios para el sistema y configurar las rutas de acceso.

MT-1.2.4 Instalación del Cliente.

Una vez instalado el servidor se puede instalar clientes en tantas maquinas como se requieran, los pasos son los siguientes:

1. Instalar Access 97 ver. en el cliente que se desea.

2. Iniciar una sesión de red con acceso al servidor designado

3. Conectar una unidad de red (z:) al directorio del sistema en el servidor

4. Desde el cliente ejecutar el programa z:InstalaCliente.exe y seguir las instrucciones

MT-1.3 Seguridad.

El sistema viene acompañado con un archivo SegSirh.mdw que contiene la configuración de la seguridad el cual siempre debe residir en el directorio Nomina del servidor. Si se requiere cambiar la ubicación deberá "Unirse al Grupo de Trabajo" ejecutando el programa "C:\WINDOWS\SYSTEM\WRKGADM.EXE" y seguir las instrucciones, es importante señalar que la ubicación de este programa puede cambiar dependiendo de donde se hubiera instalado Windows.

MT-1.4 Respaldos.

Los respaldos del sistema deberán ejecutarse únicamente a la información del servidor y se podrán hacer simplemente copiando en cinta o en otro medio el directorio donde se instalo.

MT-1.5 Reparar y Compactar la base de datos.

Compactar

Al realizar operaciones en los archivos del sistema estos crecen lo que disminuye el desempeño de la aplicación para lo cual es necesario Compactar las bases de datos.

Reparar

La base de datos cuando esta siendo utilizada y ocurre un error en la aplicación o sistema operativo requiere ser Reparada para evitar la perdida de información.

Estos dos procesos descritos están disponibles en las utilidades de Access 97, sin embargo dentro de los archivos de instalación se encuentra un programa para facilitar esta actividad Rendimiento.mdb . El cual se ubica en el directorio Varios del ser servidor.

Se recomienda aplicar el proceso de Compactar cada mes a las bases del cliente y cada dos meses a la de los servidores.

**SISTEMA INTEGRAL
DE RECURSOS
HUMANOS**

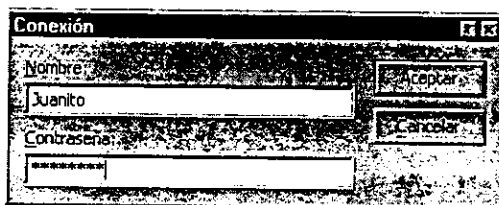
MANUAL DEL USUARIO

MU-1 Descripción General de Pantallas y Módulos Principales



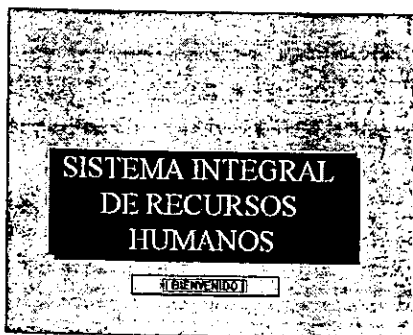
SIRH

Lo primero que debemos hacer es identificar la ruta de acceso al programa. Para dar inicio a una sesión de trabajo con **SIRH**, es necesario que desde el escritorio de trabajo de **Windows**, localice un icono con la figura de un escritorio con una lámpara y la palabra **SIRH**, como aquí se muestra. Una vez que haya dado doble clic sobre el mencionado icono se comenzara a cargar el programa **Access 97**, que es la plataforma a partir de la cual se desarrollo el presente sistema. Acto seguido el programa pedirá que introduzca su Contraseña, para poder acceder a la aplicación.

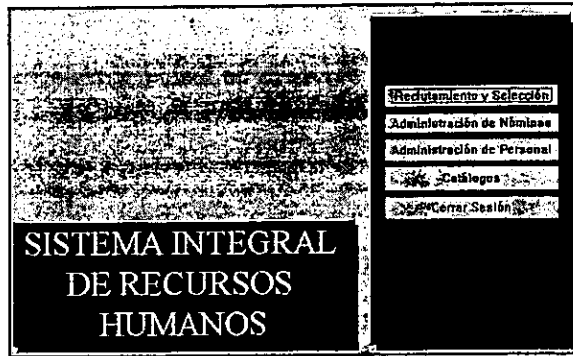


Algo muy importante que debe notar es que este cuadro de diálogo le muestra dos identificaciones de su persona, el Nombre y la Contraseña. Cuando recién aparezca esta pantalla, el cursor estará colocado de inmediato en la sección de Contraseña, en la cual usted deberá escribir su Contraseña, pero no olvide verificar que el Nombre que aparezca sea realmente el suyo, o de lo contrario no importara si el Contraseña es o no correcto, igual le negara el acceso. Si este es el caso, escriba su nombre de usuario y Contraseña asignados.

Esto es importante debido a que cada usuario del sistema tiene asignados derechos y permisos, es decir, los módulos en los que puede trabajar y los procesos que puede generar.



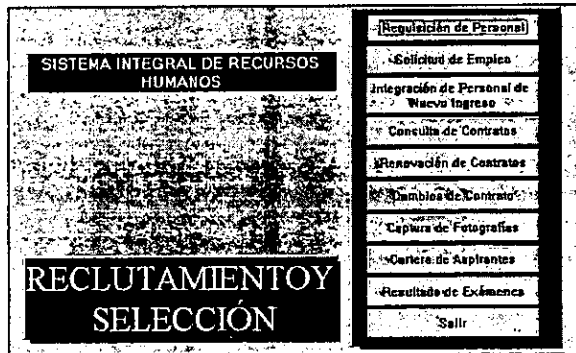
Después de la etapa de identificación del usuario (la cual esta conformada de esta manera por motivos de seguridad en la integridad de la información; no olvide que después de todo se trata de su trabajo y no querrá que cualquiera tenga acceso a el, ¿o sí?), el sistema nos mostrara la pantalla inicial o de bienvenida, en el cual solo debe dar un clic en el botón denominado "¡BIENVENIDO!" y acto seguido nos muestra el menú principal.



En esta pantalla usted podrá iniciar su navegación a lo largo y ancho de nuestro sistema. Como podrá darse cuenta en esta se dispone de varios botones, cada uno de ellos destinado a un modulo diferente. A continuación revisaremos uno a uno los más importantes.

Reclutamiento y Selección

La finalidad de este modulo es la de brindarle las herramientas necesarias, para poder llevar el registro y control del personal que labora en su empresa, iniciando los procedimientos desde que llega a solicitar el empleo, pasando por el proceso de selección, la



integración y posterior contratación de los mismos. Se cuenta además con la posibilidad de realizar cambios en las características de los contratos, así como la impresión y renovación de contratos del personal eventual de forma tal, que las condiciones laborales se pueden modificar cuantas veces sea necesario sin mayor problema.

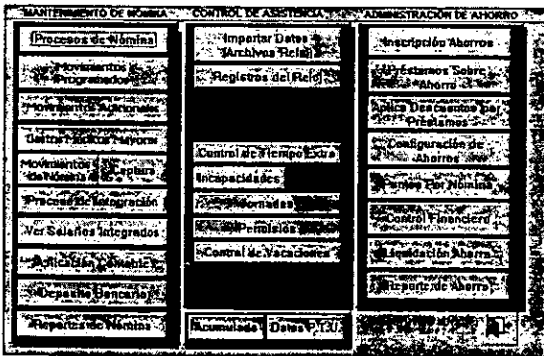
Además existe una sección en la cual puede cargar en el sistema la fotografía de la persona, de forma tal que cuente con el apoyo para poder identificar a su personal.

Otra sección importante que requiere de mención es la *Cartera De Aspirantes*, en la cual se encuentran registrados todas aquellas personas que solicitaron empleo, y que en su momento no fueron contratados, pero que por su perfil denotan características atractivas para la empresa y que en un futuro podrían ser objeto de contratación.

Administración de Nóminas

Sin duda uno de los módulos que más utilizara es el de *ADMINISTRACIÓN DE NOMINAS*. Las posibilidades de trabajo en este modulo son diversas, pues todos aquellos movimientos financieros, contables y administrativos que son necesarios para realizar correctamente los pagos de salario a su personal cumpliendo con las normas jurídicas de competencia, se encuentran contenidos en este modulo.

Como se observa en la figura aquí mostrada, la pantalla principal de *ADMINISTRACIÓN DE NOMINAS* se encuentra dividida en tres bloques principales como son Mantenimiento De Nomina, Control De Asistencia y Administración De Ahorro.

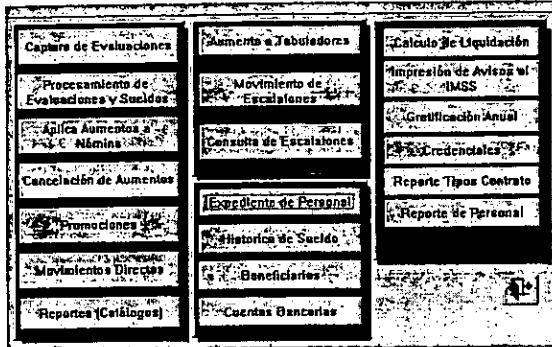


En *Mantenimiento De Nomina* se localizan las funciones necesarias para calcular y generar la nomina de su empresa, así como la impresión de reportes y archivos de información para los bancos, en caso de que en su empresa se pague vía depósito bancario. *Control De Asistencia* esta preparado para llevar el control del personal en cuanto a su tiempo de permanencia física en el lugar

de trabajo, comenzando por la asistencia diaria, tiempo extra, incapacidades, permisos, faltas, salir o llegar tarde, etc. En el caso que su empresa cuente con un sistema de caja o fondo de ahorro, *Administración De Ahorro* es el modulo en el que usted realizara todos los movimientos necesarios para tales efectos.

Administración de Personal

Un módulo que sin duda será de gran utilidad, aunque de aplicación más esporádica en comparación con el de Administración De Nóminas, es el de Administración De Personal. Dicho módulo está diseñado pensando en todas aquellas incidencias en las que se requiere de realizar consultas, evaluaciones, modificaciones a las características de uno o varios de sus empleados, así como los finiquitos o avisos al IMSS, procesos que aunque comunes, no tan frecuentes como los de una nómina.



Al igual que el módulo anterior este también requiere de un tratamiento especial por parte del usuario, razón por la cual es explicado con más detalle en el proceso 3 del capítulo 2.

Catálogos

Existen diversos factores que influyen en los diferentes procesos que se realizan en su empresa y que no son operables por el usuario, pero que si son necesarios en los diversos procesos. Estos factores son considerados como tablas, tabuladores, esquemas particulares, en otras palabras catálogos en los que el usuario puede dar de alta, modificar, dar de baja o consultar alguna de las diversas tablas que se encuentran contempladas en este módulo.

CATALOGOS			
[Datos de la Empresa]		Tipos de Nómina	
Comité	Departamentos	Percepciones y Deducciones	
Organigrama		Parámetros	Clasificador
Tipos de Trabajadores		Tablas del ISR	Tablas Info.
Niveles y Tabulador de Sueldos		Factores para Integrar Salario	
Descripciones de Puestos		Jornadas Laborales	
Pólizas de Compensaciones		Fracciones de Jornada Laboral	
Categorías		Días Festivos	
Tabulador Contractual		Zonas Geográficas	
Categorías/Tabuladores		Contratos	
Pólizas de Seguros		Calificación de Evaluaciones	
Escalafones		Salir	

El objetivo aquí no es un proceso en sí, es el de determinar parámetros y límites bajo los cuales se

realizaran muchos de los procesos que se llevan a cabo en el entorno del SIRH. Lo que necesita saber es que tipo de tabla va a manejar, en que procesos aplica y de que manera se involucra.



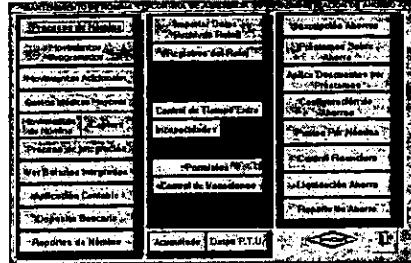
El ultimo botón de esta pantalla esta destinado a terminar la aplicación, así que en el momento en que usted decida que su trabajo con el SIRH ha terminado de un clic en él y regresara al escritorio de trabajo.



Procesos De Nómina

Uno de los aspectos más importantes y lo que se considera como el cuerpo central de todo el SIRH, es precisamente este módulo, y más aun este proceso. **Procesos de Nómina**, es vital por que aquí hablamos de aspectos financieros, y esa es una de las cuestiones que más preocupan a todos.

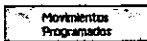
En este módulo nos encargamos de realizar todas las operaciones pertinentes en el proceso de pago, por lo mismo detallaremos las operaciones y su relación entre ellas. De cualquier forma si existen dudas en el procedimiento, puede verificar los diagramas de flujo de procedimiento, en el último capítulo de este manual.



Aunque los botones en la pantalla se encuentran dispuestos en un orden específico, nosotros los analizaremos en una secuencia lógica de procedimientos.

En cualquier proceso de nómina, se cuenta con movimientos o incidencias especiales, tales como préstamos, créditos, INFONAVIT, FONACOT, etc. Estos procesos pueden ser descontados al trabajador de su salario corriente, de forma permanente o amortizada en un número determinado de periodos. Para manejar estos dos casos el SIRH los divide en dos procesos independientes.

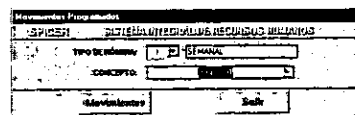
1)



Movimientos Programados, es el primero de estos procedimientos que analizaremos. Estos pueden ser una percepción o deducción, según se haya estipulado en la tabla Percepciones y Deducciones del módulo Catálogos (el cual se analiza a detalle en el proceso 4), y en el cual se le indica al sistema si tenemos incidencias que serán amortizadas hasta saldar el adeudo o bien una percepción que se dará como importe igual periodo a periodo hasta cubrir este.

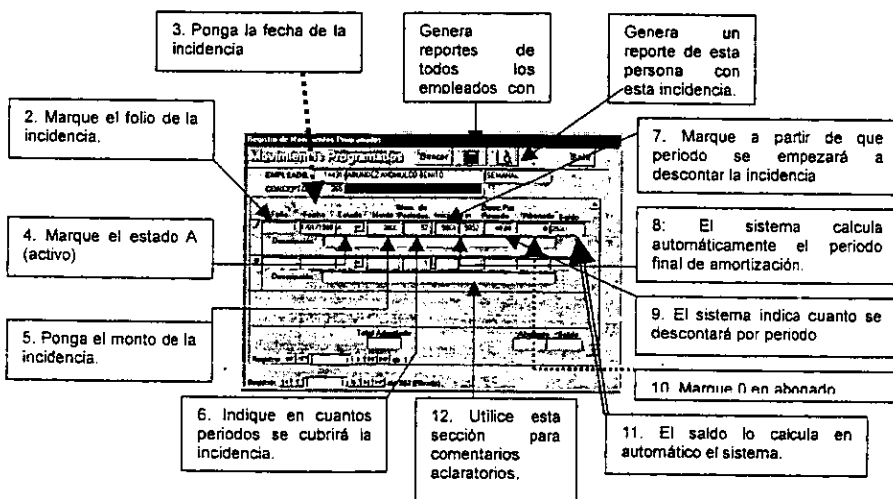
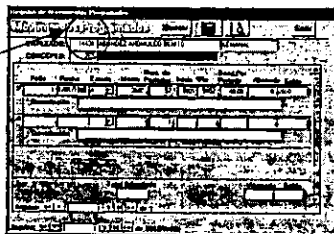
Las incidencias que se permiten en este procedimiento, se encuentran descritas en la tabla PERCEPCIONES Y DEDUCCIONES del módulo CATALÓGOS, marcadas como descuentos programados "DP". En el proceso 4, de este capítulo analizaremos a detalle este aspecto.

En este proceso deberá primero indicar el tipo de nómina en la que se localiza al trabajador, el concepto del cual capturara las incidencias y luego dar clic en "Movimientos". Hecho lo anterior aparece la pantalla mostrada en la página siguiente.



Es importante que siga uno a uno los pasos descritos en la siguiente pantalla para que sepa el correcto llenado de esta y no tenga problemas cuando lo intente por sí mismo.

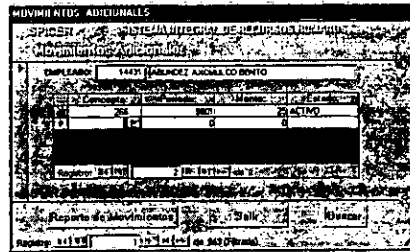
1. En esta pantalla localice primero al trabajador mediante el uso del botón "Buscar", que proporciona una ventana de búsqueda idéntica a la que se utiliza en WORD o EXCEL.



ESTE PROCESO DEBE REALIZARLO ANTES DE GENERAR LA NÓMINA, porque el sistema, toma las incidencias, las procesa, calcula los movimientos correspondientes, y al cerrar la nómina actualiza los saldos.

2) **Movimientos Adicionales**

El segundo tipo de incidencias que veremos se refiere a los descuentos o percepciones que son de carácter fijo, es decir, que nómina tras nómina se repetirán con las mismas características. Estas NO son amortizadas.



Al igual que en muchos de los procesos que veremos durante este manual, lo primero que haremos es indicar el tipo de nómina a procesar, y después dar clic en el botón de proceso de la pantalla, en este caso en particular, "Movimientos Adicionales" en el que nos presenta una segunda pantalla del proceso.

Nuevamente siga los pasos de llenado para evitar equivocaciones:

3. Marque a partir de que periodo se comenzara a aplicar la

4. Indique cual será el valor de la variable "Adicional" para el cálculo en fórmulas. Verifique primero el tipo de concepto y formula a aplicar.

5. ponga "ACTIVO" en el estado del concepto.

1. Localice primero al trabajador en el que marcara las incidencias.

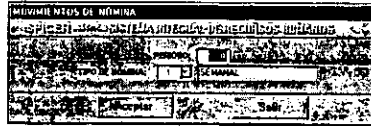
Concepto	Formula	Periodo	Estado
42	IBECAS		ADICIONAL
252	CREDITO INFONAVIT (COFINANCI)	SDINFOP(SM1)*1.4-@M6010-@M6008PADIC	
253	MANTENIMIENTO INFONAVIT (A)		
252	PENSION ALIMENTICIA (A)	(TOTALPER)/ADICIONAL/100	
256	CREDITO INFONAVIT	SDINFOPOR-SDINFOP(SM1)*1.4-@M6008	
297	FUNERALES		ADICIONAL
308	INFONAVIT CUOTA FUA		ADICIONAL
316	COFINANCIAMIENTO INFONAVIT	SDINFOP(SM1)*1.4-@M6010-@M6008PADIC	

Como puede ver en la sección aumentada, el concepto se refiere a todos aquellos que en la tabla PERCEPCIONES Y DEDUCCIONES de la sección de CATÁLOGOS, tienen marcada la clasificación Adicional "A", los cuales se calculan a partir de formulas específicas que consideran como "Adicional" la cantidad que usted estipula en el cuadro Monto de la pantalla.

Nuevamente la observación: **ESTE PROCESO DEBE REALIZARLO ANTES DE GENERAR LA NOMINA**, porque el sistema, toma las incidencias, las procesa y calcula los movimientos correspondientes.

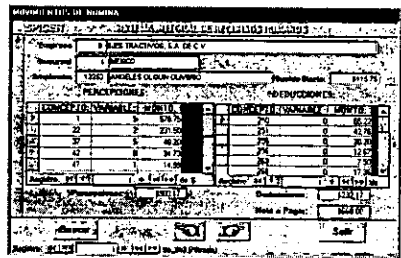


3) Otro aspecto importante es el manipular algunas variables para el cálculo de la NÓMINA y esto lo haremos con la opción de Movimientos de Nómina.



Esta opción le permite realizar cálculos directos, antes de generar la NÓMINA. Los cálculos directos son aquellos que están especificados en la tabla de PERCEPCIONES Y DEDUCCIONES de la sección de CATÁLOGOS como movimientos directos, o aquellos que son resultado de fórmulas en donde la variable puede ser modificada.

Para manipular esta sección, lo primero que debe hacer cuando de clic en el botón de "Movimientos de Nómina", es indicar el periodo o número de nómina que desea consultar, y el tipo de nómina, pues en esta sección se puede además de realizar modificaciones a la NÓMINA actual, hacer consultas sobre los resultados de las incidencias de las nóminas anteriores.



Cuando de clic en "Aceptar", pasa a la pantalla principal de este procedimiento. Se distingue en la parte superior la sección de identificación de la empresa y empleado.

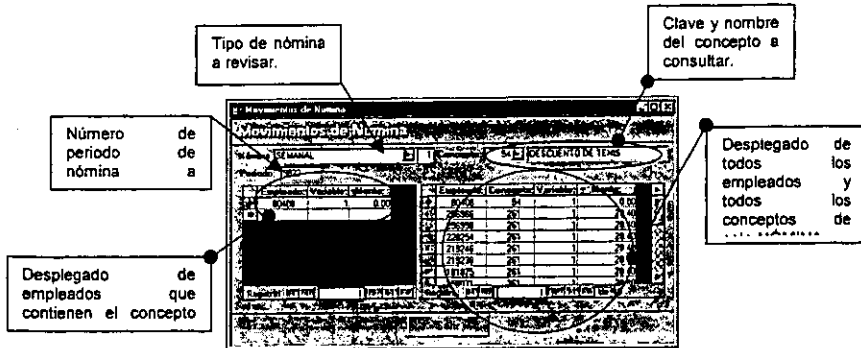
En la parte central existen dos secciones tabulares. La del lado izquierdo muestra y desglosa el total de conceptos de percepción del trabajador, en la nómina solicitada y en el tabular del lado derecho el desglose de deducciones. Debajo de estas secciones el SIRH muestra los totales de cada concepto y el total de la nómina para esta persona.

Si desea consultar a otra persona, puede hacer uso de las dos manos que se encuentran en la sección inferior de la pantalla, y que lo llevan a un registro antes o después, según sea el caso. Puede hacer uso también del botón de búsqueda rápida, denominado "Buscar", y del cual ya explicamos su funcionamiento. Una tercera opción de desplazamiento es la sección inferior con las flechas y el indicador de registro.



4) Una posibilidad útil de trabajo es la que se brinda por medio

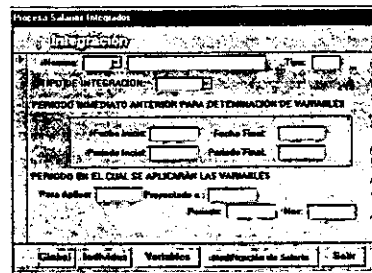
del botón de **CAPTURA**. La pantalla de este procedimiento se compone de la siguiente manera:



- 1) Indique la nómina.
- 2) Para realizar modificaciones a las variables, marque el concepto a consultar.
- 3) Indique el periodo.
- 4) En el empleado a modificar, de clic.
- 5) En el cuadro variable, cambie el valor existente por el que usted desea actualizar.
- 6) En el cuadro monto es muy importante que marque 0 siempre, cuando es una fórmula o si es directo en variable ponga 1 y en Monto el importe que desee, pues el sistema se encarga de recalcular este valor.

Proceso de Integración

5) **PROCESO DE INTEGRACIÓN.** Este es uno de los procedimientos más comunes, pues en él se tendrán que reflejar muchos de los cambios realice en otros procedimientos del SIRH. En proceso 3, hay varios procedimientos que lo referirán a esta sección, por lo que le recomendamos tener muy presente este apartado.



El proceso de integración se encarga de realizar el cálculo de los salarios integrados. Un aspecto importante que hay que resaltar es que el SIRH calcula tres tipos principales de salario diario integrado (SDI), y estos son IMSS, INFONAVIT (INFO), y las INCAPACIDADES (SDIIN). Es necesario que recuerde cuales son las

siglas de identificación de cada uno, ya que será con ellas con las que indicara al sistema, que tipo de integrado generar.

El SDI del IMSS se deberá calcular cada vez que sea modificado el sueldo de uno o varios trabajadores, o cada día primero de mes, como lo requiere el IMSS. El procedimiento puede ser individual o global. Será global cuando sea resultado de las evaluaciones corporativas, de aumentos en tabuladores, o de cambios al salario mínimo del DF. y será individual, cuando solo se afecte a una o más personas, pero no a todos.

El SDI del INFO, se genera solamente cada dos meses, es decir bimestralmente (enero, marzo, mayo, julio, septiembre, noviembre). Este proceso es global. El SDI del SDIIN se genera cuando una persona esta incapacitada, y es individual. Para el vaciado en la pantalla, la información que requiere, son los datos con los cuales se generó el último SDI del IMSS, del cual tiene conocimiento el IMSS.

Hagamos un ejemplo de un integrado SDIIN, para hacer más claro esto. Una persona se incapacita el 15 de febrero, por lo que la pantalla se llenara de la siguiente manera:

The screenshot shows a software window titled 'Proceso Salarios Integrados' with a sub-window 'Integración'. The interface includes a dropdown menu for 'TIPO DE INTEGRACIÓN' set to 'SDIIN', a section for 'PERIODO INMEDIATO ANTERIOR PARA DETERMINACIÓN DE VARIABLES' with dates from 01/ene/90 to 31/feb/90, and a section for 'PERIODO EN EL CUAL SE APLICARÁN LAS VARIABLES' with dates from 15/feb/90 to 01/feb/90. A worker's name 'SANCHEZ MONROY GABRIELA TATIANA' is visible. At the bottom, there are buttons for 'Global', 'Individual', 'Variables', 'Modificación de Salario', and 'Salir'. The callout boxes provide the following instructions:

1. Indique el tipo de nómina, y el tipo de integración SDIIN (será incapacidad).
2. Indique las fechas del periodo anterior. En este caso del 1° al 31 de enero.
3. Los periodos los proporciona en automático el sistema. Verifique que sean correctos.
4. Marque la fecha inicial de la incapacidad.
5. Ponga la fecha del 1° del mes en que ocurre la
6. Verifique el periodo en el cual se aplicará la incapacidad.
7. Clic en individual. El sistema manda un mensaje de aviso. Clic en aceptar.
8. Concluido el paso 7, el sistema muestra una sección nueva para indicar de que empleado
9. Seleccione del menú colgante al trabajador incapacitado.
10. Haga clic en Individual nuevamente, y Si al mensaje de confirmación que manda el
11. Al terminar haga clic en Salir.

Con esto el integrado de la incapacidad será el mismo que entró en su proceso de integración en el mes de febrero al IMSS.

El objetivo de todo este proceso es que el SDIIN, que se subsidie vía nómina, sea el mismo con el cual el IMSS pagará el importe correspondiente por este concepto.

El resto de los botones que aparecen en la pantalla son como siguen:

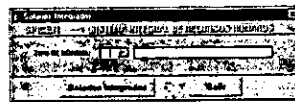
Global: dispara la integración de salario de todos los empleados activos en la empresa por el concepto escogido.

Variables: es un reporte que muestra por que conceptos esta formando el nuevo SDI.

Modificación al salario: es un reporte en medio magnético de las operaciones realizadas.

6)  **VER SALARIOS INTEGRADOS.**


Aquí usted puede de manera más concreta ver cual es el monto del SDI en cada uno de los tres conceptos principales, por cada uno de sus trabajadores. El proceso es sencillo, haga clic en el botón antes mostrado, elija el tipo de nómina en la desea consultar, y haga clic en Salarios Integrados.



que

Fecha	Fecha Emisión	Sal. P.	Periodo	Estado	Tipo
12/02/1998	1997	8271	999	A	PPD
21/02/1998	1997	1821	999	A	PPD
		1400			

En la nueva pantalla localice al trabajador que desea consultar, mediante el uso de "Buscar", o con las manos apuntando. Cuando lo tenga en pantalla, en la sección tabular puede ver cual es el SDI, por cada uno de los conceptos manejados por el SIRH.

7)  **DEPÓSITO BANCARIO.** Si en su empresa se paga vía depósito bancario, es importante llevar este proceso a cabo. En él se genera el pago

de toda la planta trabajadora, es decir, que todos y cada uno de los elementos de su personal deben tener una cuenta bancaria, la cual registra en **Administración de Personal** (este procedimiento se explica más adelante en el proceso 3). El procedimiento es sencillo, seleccione el tipo de NÓMINA que va a pagar, anote la fecha de creación del archivo, el periodo al que corresponde, y en la sección de secuencia, marque uno diferente por cada tipo de NÓMINA, para evitar borrar o sobre escribir información.

Depositos Bancarios

Nomina: SEMANAL Tipo: []

Fecha Creación: [] Periodo: [] Secuencia: 01

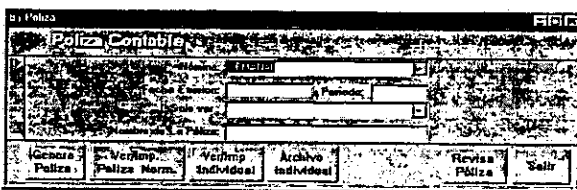
Generar Archivo Salir

se

Quando termine, de un clic en Generar Archivo, el cual crea un archivo en medio magnético (disco) de la información y un reporte impreso de los movimientos. Este reporte debe checar con el neto del reporte de nómina, siempre y cuando todos los trabajadores tengan una cuenta bancaria.

8) **Aplicación Contable** **APLICACIÓN CONTABLE.** Este es el 2° paso después de la generación del Depósito Bancario. Aquí se genera una póliza contable por Nómina y por Periodo. Además genera un archivo en disco para enviar al sistema de información contable de la empresa y generar un reporte de la misma.

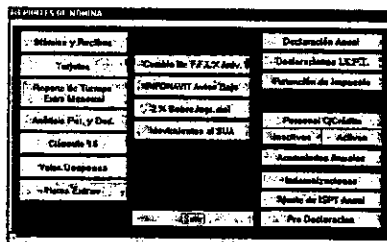
Algo importante que debe tener en cuenta que por cada NÓMINA que genere, cualquiera que esta sea (semanal, quincenal, mensual, confidencial, finiquitos, aguinaldo, PTU, etc.), se debe generar el Depósito Bancario y la Aplicación Contable o póliza.



es

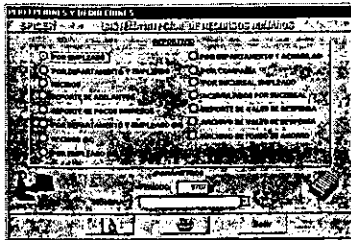
9) **Reportes de Nóminas** **REPORTES DE NÓMINA.** En lo referente a la generación de reportes, el SIRH le proporciona una amplia variedad de opciones y posibilidades. En este caso describiremos los más usuales y principales a la vez.

Quando entre a este apartado, lo primero que vera es la pantalla del lado derecho, en la cual puede elegir la naturaleza básica del reporte que desea imprimir.



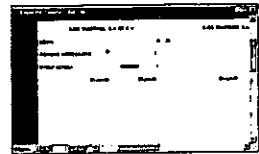
Nómina y Recibos

Este botón muestra la sección principal de opciones de reportería, en la que solo debe decidir cual es el que necesita, dar un clic en el botón de selección, indicar el tipo de NÓMINA, y el periodo, posteriormente en la lupa dar un clic para pasar a vista preliminar o directamente en el botón con la impresora y se imprimirá en papel. Le recomendamos que procure siempre ir primero a preliminar y después imprimir, solo para verificar que lo que va a mandar es realmente lo que necesita.



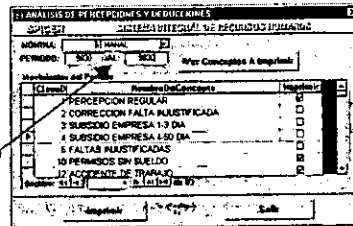
Tarjetas

El botón Tarjetas esta destinado a la impresión de las tarjetas checadoras de asistencia para las personas que en Escalafón de Administración de Personal (procedimiento que analizaremos posteriormente) fueron señaladas como que debían checar su entrada y salida en un reloj. Indique el tipo de nómina (que por regla casi siempre es semanal) y el periodo para el cual se usaran las tarjetas, después mande la impresión.



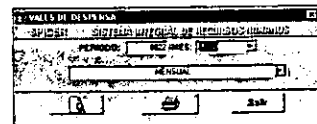
Análisis Per. y Ded.

Este reporte le permite conocer las cifras de control de cada uno de los conceptos involucrados en la NÓMINA del periodo correspondiente, ya sean percepciones o deducciones, usted elige. Indique los periodos que desea analizar, así como el intervalo de los conceptos a analizar.



Vales Despejados

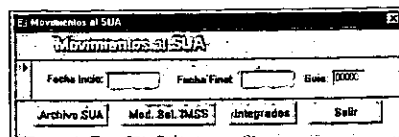
Este es un reporte bastante sencillo, pues nos muestra la distribución que se ha de realizar de los vales entre los trabajadores del tipo de NÓMINA que seleccione.



Indique el periodo, el mes correspondiente y el tipo de NÓMINA, después mande a imprimir.

Movimientos al SUA

El SUA (Sistema Unico de Autodeterminación) es un requisito del IMSS, para el cual el mismo tiene un software especializado, por medio del cual se le informa



de los movimientos e incidencias de la empresa.

Este reporte por así llamarlo, genera el archivo en el cual se incluye la información de los movimientos del mes anterior (bajas, altas, movimientos de salario, incapacidades, faltas, etc.) para que esta sea transferida al software ya antes mencionado y se entere al IMSS de las incidencias.

El botón "Archivo SUA" genera un archivo en medio magnético (disco). Cuando este es generado, en las casillas de Fecha Inicio y Fecha Final, se pondrán las fechas del día primero y último del mes anterior, respectivamente en la pantalla del SIRH.

El botón "Mod. Sal. IMSS" genera un archivo en medio magnético para enterar al IMSS de los movimientos de salario. Indique el rango de fechas y el número de guía y oprima el botón indicado.

El botón *Integrados*, es un reporte de los integrados registrados en el sistema.

10)

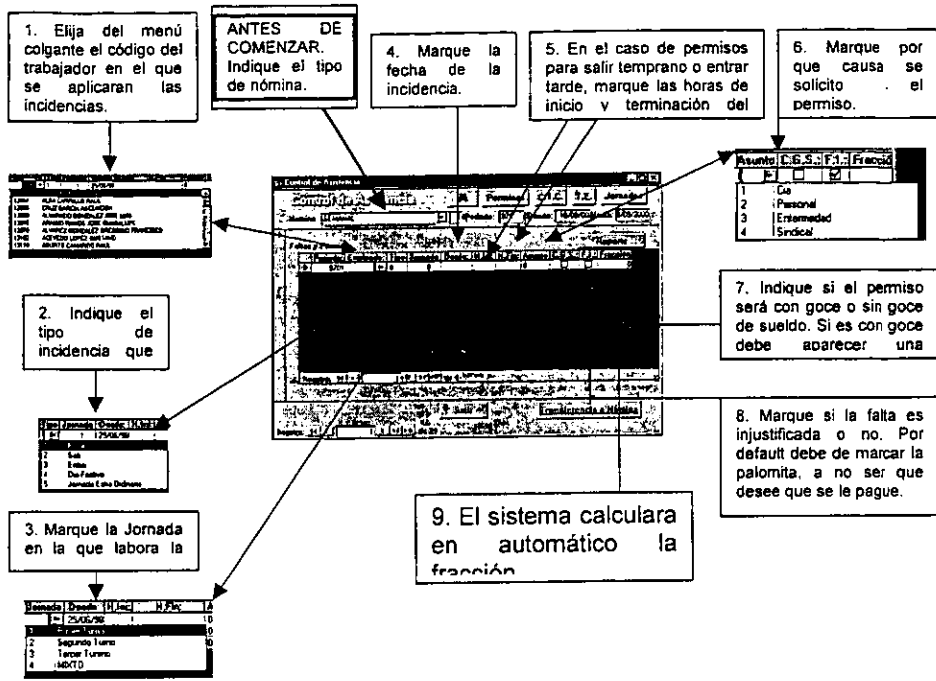


PERMISOS. Tal vez uno de los aspectos más importantes para el cálculo de cualquier nómina, sea el que se refiere a la asistencia del personal, y para ello se cuenta con una serie de incidencias que deben de ser evaluadas para determinar los ingresos o deducciones adicionales originadas de este aspecto. el SIRH le proporciona los medios necesarios para que usted solo tenga que capturar los aspectos básicos, y en el resto del proceso usted ya no tenga problemas de cálculo.

Cuando ingrese a este apartado, lo primero que vera es una pantalla con una serie de botones en el encabezado de la misma, que le permiten capturar las incidencias que se pueden presentar con este respecto. Veamos una a una como se registran.

1)Primero escoja la nómina, después indique el periodo en el que se aplicarán las incidencias.

Empecemos por permisos, que es la sección que por default proporciona primero el sistema.



Un ejemplo para un permiso de salir y regresar, sería el siguiente:

1. Elija al trabajador.
2. Marque el tipo de permiso 2.
3. Indique la jornada en que labora. Por ejemplo la 1.
4. Marque la fecha del permiso.
5. En H. Inic. Ponga las 12:00
6. En H. Fin. Ponga las 13:30, suponiendo que regreso a la 1:30 de la tarde. Por lo cual solo se habrá ausentado hora y media. Pero si ya no regresa deje en blanco este cuadro.
7. Ponga en asunto 2, suponiendo que fue por razones personales.
8. Deje sin marcar el cuadro de C.G.S. para indicar que ese tiempo no se le pagara.
9. En Falta injustificada deje sin marcar, puesto que le aviso cual era la razón de su ausencia.
10. El sistema calcula el porcentaje correspondiente a la hora y media que no trabajo.

Un ejemplo para una falta, sería el siguiente:

1. Elija al trabajador.
2. Marque el tipo de permiso 1.
3. Indique la jornada en que labora. Por ejemplo la 1.

4. Marque la fecha de la falta.
5. Deje sin marcar el cuadro de C.G.S. para indicar que ese tiempo no se le pagara.
6. En Falta Injustificada márkuela también, salvo que le haya avisado cual era la razón de su ausencia.
7. El sistema calcula el porcentaje correspondiente a la fecha que no trabajo como 1.

Veamos ahora como se llena la pantalla para el caso de un cambio temporal de categoría.

Nuevamente marque el periodo para el cual se consideran las incidencias.

1. Elija del menú colgante del trabajador en el que se aplicarán las incidencias.

2. Marque las fechas de inicio y culminación del

3. El sistema indica cuantos días permanecerá en esa categoría.

4. El sistema marca de manera automática la categoría original de

5. Elija en el menú colgante la categoría a la que pasará el

6. Indique en que jornada laborará el C.T.C.

7. Marque la causa que origino el

Mostrar: 1
PERMISO
INCAPACIDAD
CAPACITACION

Ahora hagamos un ejemplo de tiempo extra de un solo día.

Suponga que un trabajador vino a trabajar de las 10:00 PM de un lunes, y se fue a las 5:00 AM del martes.

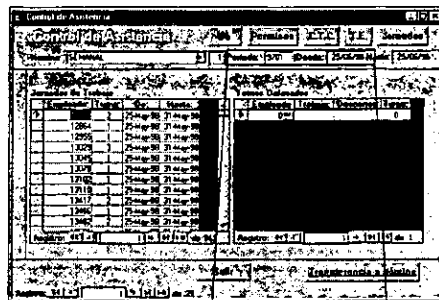
1. Marque el periodo en el que se devengarán las horas extras
2. Seleccione al trabajador.
3. Ponga en el cuadro "Desde" la fecha del día lunes.
4. En el cuadro "Hasta", marque la misma fecha del día lunes. Esto es importante porque si marca la fecha del martes, le estará diciendo al sistema que el trabajador vino a trabajar dos días de las 10:00 a las 5:00, cuando en realidad solo fue uno.

Mostrar: 1
1 TIEMPO EXTRA
2 JORNADA EXTRA
3 COMISION

Mostrar: 1
1 PERM
2 CAPACITACION "F"
3 CAPACITACION "C"
4 PRODUCCION "C" Y TR.
5 HAYTE DE MANTENIME
6 PRODUCCION "S" Y MD
7 PRODUCCION "A"
8 PRODUCCION "C"

5. Ponga 2200 en el cuadro Entra y 0500 en el cuadro Sale, el sistema los convertirá a horas con formato de 12 horas (es decir AM y PM).
6. Indique el tipo de incidencia de que se trata. En este caso deje Tiempo extra (note que el sistema le permite manejar otros dos tipos de incidencias).
7. Si el tiempo extra será trabajado en C.T.C. indíquelo activando la casilla correspondiente.
8. Si el paso 7 es verdadero, indique en que tabulador de categoría será trabajado el tiempo extra, de lo contrario déjelo sin modificaciones.

Al personal se le rolan turnos, y el sistema tiene que saber cuando trabajará alguien en el primer, segundo o tercer turno, por que de lo contrario, si se registro a alguien en el primer turno y se le rola al segundo turno, y no se notifica al sistema, el marcará faltas a esa persona, aunque en realidad no sea así.



O tal vez vino a trabajar un día domingo, y por lo tanto tiene derecho a descansar otro día.

Desde: 30/05/98 Hasta: 30/05/98

Recuerde siempre marcar las fechas de inicio y final de las incidencias a capturar.

Para ello debe seguir los siguientes pasos:

1. Marque el periodo de la incidencia
2. En los cuadros Desde y Hasta, ponga las fechas del periodo comprendido en el cual se dio origen a los días domingos laborados en forma ordinaria.

JORNADAS DE TRABAJO:

3. Seleccione al empleado para el cual especificara el nuevo turno.
4. Seleccione del menú colgante el nuevo turno en que se presentará a trabajar.
5. Marque en los cuadros Desde Hasta, las fechas en las que se

presentará a ese nuevo turno.

TURNOS DESFASADOS

3. Seleccione el código del trabajador que laboró un día domingo.
4. Capture en el cuadro "Trabajo", la fecha del domingo que se presentó a trabajar.
5. En el cuadro "Descansa" digite la fecha del día que se le concederá para descansar.
6. Indique en que turno laboró el día domingo.

Una vez que hemos terminado de explorar las posibilidades de este procedimiento denominado Procesos, es importante hacer mención de la necesidad de dar un clic en el botón que lleva el título de "**Transferencia a NÓMINA**" en color azul. Este botón se encarga de transferir la información que hemos capturado a una sección donde el sistema la buscare para procesar las horas extras tal y como lo veremos en el procedimiento siguiente. Por esto siempre que capture algo en esta pantalla, antes de dar clic en salir debe oprimir el botón de transferencia a Nómina, para que el sistema sepa de la existencia de la información, de lo contrario corre el riesgo de que no se procese adecuadamente su nómina.

NOTA: Es importante que siempre marque las fechas de inicio y final de las incidencias que está capturando, para que el sistema realice los procesos adecuadamente.

Otra de las posibilidades de esta pantalla es que por cada uno de los conceptos que acabamos de ver, es factible la generación de un reporte impreso. Para ello seleccione el tipo de incidencia que desea y de un clic en el botón con la leyenda "Reporte" y le proporcionara una vista previa del mismo y la posibilidad de la impresión en papel.

- 11) **Control de Tiempo Extra** CONTROL DE TIEMPO EXTRA. Este procedimiento se encarga de separar las horas extras registradas en el sistema, en dobles, triples, turno en que fueron laboradas o devengadas y determina cuantas de ellas se integrarán.

Un aspecto importante que debe tener en cuenta es que el tiempo máximo de postergación para captura y registro de tiempos extras es de cinco semanas por incidencia. Por lo anterior debe evitar que este concepto sea excesivamente desfasado en sus incidencias. El orden de operación en esta pantalla es como sigue (tenga cuidado con el paso número 2)¹:

¹ sábado a viernes, verifique la política de su compañía. En el año un día libre en un periodo anual de lunes a domingo, y si el trabajador laboró el día domingo o día miércoles, tendrá que poner la Fecha Inicial del día lunes y la Fecha Final del día domingo.

1. Marque la nómina a generar.

2. Marque la fecha de inicio y Fecha Final de la semana laboral, conforme lo especifique

3. Oprima el botón "Poner el periodo".

4. Ahora un clic en Desglosa Horas Extras.

5. Como último paso haga clic en "Inserta Horas extras a Nómina".

6. Cuando el sistema termine de trabajar, haga clic en "Salir" para terminar este

12)

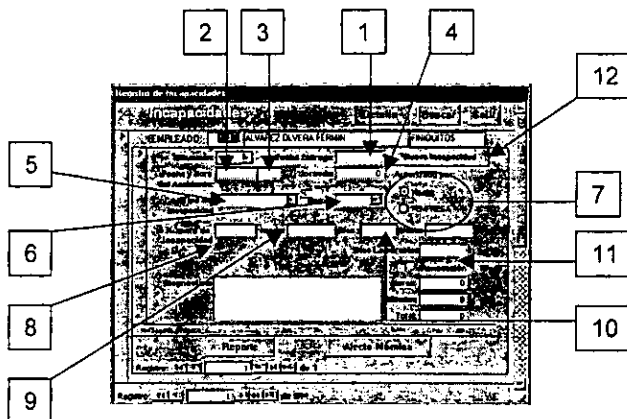
Incapacidades

INCAPACIDADES.

Las incapacidades son un tipo de incidencia que aunque nada deseables, se presentan en toda empresa la forma en el que el SIRH le pide que le proporcione la información para su procesamiento es la de la pantalla de la derecha. En ella como puede observar lo primero que debe hacer es localizar al trabajador, que sufre la incapacidad, para ello se puede auxiliar del botón Buscar, que aparece en la parte superior de la pantalla. A continuación se describe el pocedimiento:

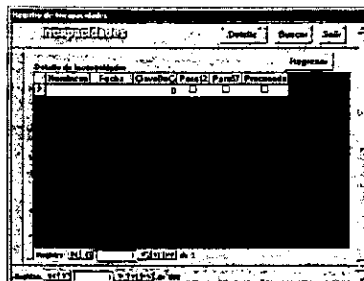
1. Marque la fecha en la que se enteró al IMSS de la incapacidad
2. Marque la fecha en que se produjo el motivo de la incapacidad
3. Marque la hora en que se produjo el motivo de la incapacidad
4. Ponga la jornada en la que trabaja la persona.
5. En "Tipo de Incapacidad" marque siempre como NORMAL, en caso de que sea necesario modificarla, lo puede hacer posteriormente.
6. Determine cuál es el Ramo de la incapacidad, seleccionando la opción de entre las disponibles en el menú colgante.
7. En "Autorizado por:" marque siempre IMSS
8. De él número de incapacidad en caso de quiera llevar un folio.
9. En el cuadro "Desde" indique a partir de cuando ampara el IMSS la incapacidad.
10. En el cuadro "Días", indique cuantos días autorizó el IMSS la incapacidad.
11. En el caso de que se trate de una Renovación, active la casilla de selección que lleva este nombre, de no ser así, déjelo en blanco.

12. Si existiera una incapacidad adicional que desee capturar para la misma persona, utilice el botón de "Nueva Incapacidad".



Quando termine oprima el botón "Afecta a Nómina", para que la incapacidad sea procesada. Cuando esto ocurra observe que el sistema cambia automáticamente la casilla "Situación" de una "P" por una "A". Esto significa que la incapacidad ya fue procesada.

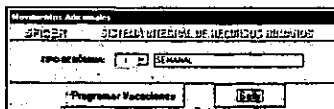
Si quisiera ver como procesó las incapacidades de esta persona, puede utilizar el botón "Detalle", que lo lleva a la pantalla de la izquierda, en la cual en forma tabular, usted podrá ver el número de incapacidades, la fecha y para que conceptos aplica.



13) Control de Vacaciones

CONTROL DE VACACIONES.

Quando tenga que procesar las vacaciones de alguien sólo de un clic en el botón arriba mostrado, y en la caja de diálogo que ahí aparece, indicar como es costumbre, el tipo de NÓMINA a la que pertenece el trabajador.

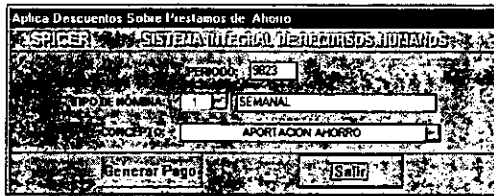


En el caso de que exista un segundo o tercer préstamo, tiene el botón con la leyenda "Nuevo Préstamo", el cual le proporciona la oportunidad de realizar un nuevo registro de préstamo para la misma persona.


18) 

APLICA DESCUENTOS POR PRÉSTAMOS. Una

vez que se ha concedido un préstamo y se ha registrado en el sistema, es importante que se generen las incidencias correspondientes, para que el sistema sepa cuanto, cuando y como debe realizar los ajustes para el cobro de los préstamos. El proceso es similar al que realizamos con las horas extras. Después de capturar los préstamos, debe aplicarlos. Para ello utilice el botón mostrado al inicio de párrafo, el cual despliega la pantalla aquí mostrada, en la que debe nuevamente indicar en que NÓMINA concedió los préstamos, en que concepto los cargo, y Generar el Pago. El sistema trabajara un momento en los cálculos correspondientes, tras lo cual cuando termine puede dar un clic en el botón de "Salir", y volverá al menú principal de *Administración de Nóminas*. Este proceso se debe de realizar independientemente de si existen o no préstamos en el periodo.

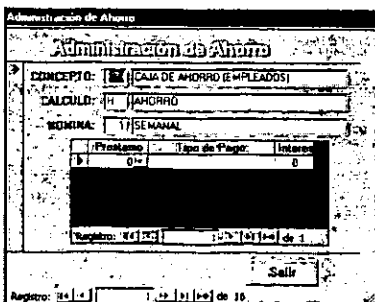


NOTA: Verifique si ya capturó en catálogos en el botón de parámetros la tasa correspondiente a aplicar en el mes por concepto de caja y/o fondo, de no haberlo hecho dirijase a éste y registre las tasas, posteriormente siga al procedimiento de **APLICA DESCUENTOS POR PRÉSTAMOS** como lo hemos explicado anteriormente.

19) 

CONFIGURACIÓN DE AHORROS. En esta sección

usted le indicara al sistema como se relacionaran los conceptos de parámetro de cálculo en la NÓMINA, con las características de los préstamos concedidos.



La pantalla muestra aquellos conceptos que en la tabla Percepciones y deducciones de la sección de Catálogos (la cual analizaremos posteriormente con más detalle), están registrados como ahorro, o "H" en la denominación de Cálculo, y el usuario puede generar una correlación de los conceptos de préstamo aquí mostrados con los tipos de interés permitidos en el sistema (Un Solo Pago y Descuento Normal). De forma tal que usted

aportación será aquel cuando el trabajador comenzó a aportar y cuanto aporta por periodo, esto normalmente se aplica en la caja de ahorro.

17)



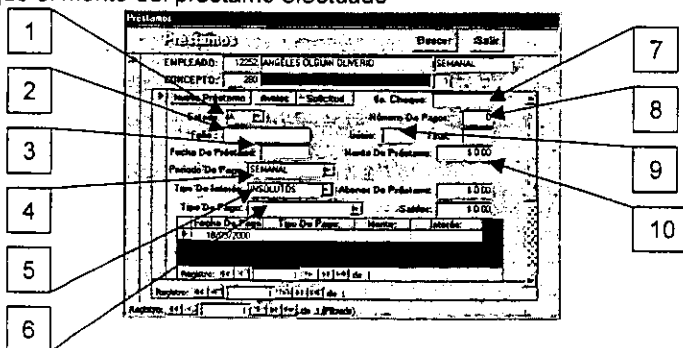
PRÉSTAMOS SOBRE AHORRO. Hay ocasiones en

que por situaciones diversas algunos trabajadores, se ven en la necesidad de solicitar préstamos. Estos pueden ser concedidos de las aportaciones que hacen a la caja o al fondo de ahorro. Para indicar al SIRH, que alguien tiene un préstamo, usted sólo debe entrar en este procedimiento y llenar la pantalla que ahí muestra. Para ello debe primero, indicar el tipo de nómina, y el concepto sobre el cual se le concedió el préstamo.

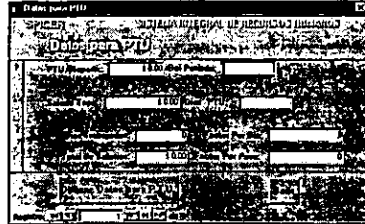


Los préstamos pueden ser concedidos de la caja de ahorro, del fondo de ahorro, o de ambos. Usted sólo indique al sistema sobre cual prestara. Cuando entre la pantalla localice primero al trabajador, indique lo siguiente:

1. Marque el estado como Activo
2. Marque el folio que seguirá el registro
3. Indique la fecha del préstamo
4. Seleccione del menú colgante la periodicidad con se efectuarán los pagos.
5. Seleccione por regla general INSOLUTOS en el Tipo de Interés.
6. En el tipo de pago (UN SOLO PAGO, indica que en la nómina se le descontarán únicamente los intereses, el monto se cobrara por otro medio, y de manera independiente. DESCUENTO NORMAL, indica que se descontaran intereses más capital.
7. Ponga el número de cheque expedido.
8. Indique en cuantos periodos se devengara el préstamo
9. A partir de que periodo se comienza a cobrar
10. Marque el monto del préstamo efectuado



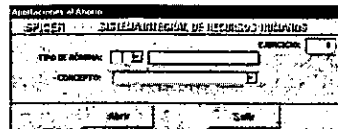
15) **Datos P.T.U.** **DATOS PTU.** Mediante el uso de este módulo, usted integrara al sistema los datos necesarios para realizar los cálculo del reparto en la Participación de los Trabajadores en las Utilidades (PTU).



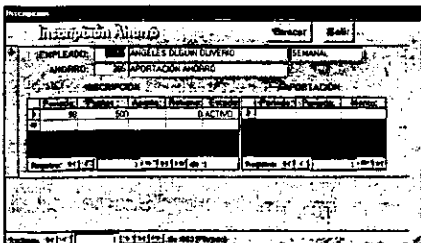
En esta pantalla usted ingresara la información de la siguiente manera:

1. Inserte la cantidad a repartir entre los trabajadores, en el cuadro PTU Repartir.
2. Inserte el Salario tope Diario en la sección del mismo nombre
3. Indique 365 ó 366 si es año bisiesto en la sección Días PTU, y el sistema determinara los días totales laborales por todo el personal y su factor, además define el total de salarios y el factor correspondiente.
4. Oprima el botón con la leyenda "Obtiene Datos para PTU"
5. Cuando el sistema termine de calcular, haga clic en Salir.

16) **Inscripción Ahorros** **INSCRIPCIÓN AHORROS.** Ya habíamos mencionado la existencia de una herramienta para el manejo y control de la caja y fondo de ahorro en su empresa. Para ello sirve este botón, él cuál cuando usted tenga alguien nuevo en su empresa, y comenzara a aportar para la caja y/o fondo de ahorro, deberá inscribirlo para que el sistema tome las incidencias correspondientes y las aplique en la NÓMINA del trabajador.



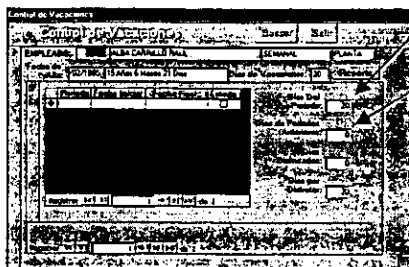
Lo primero que debe hacer es indicar en que tipo de nómina se encuentra registrado el trabajador, y después en que concepto lo inscribirá, ya sea caja o fondo. Posteriormente debe de:



1. Buscar al trabajador mediante el uso del botón con el nombre "Buscar".
2. Indique el periodo de inicio de la caja (el año en el que empezó el ejercicio de la caja o fondo).
3. En las casillas de Puntos, Aporta y Retiene marque con 0 (importantísimo).
4. En la casilla de "Estado" marque ACTIVO.

En el caso del fondo de ahorro, cuando la aportación se calcule en base a sueldo, únicamente se llenara la sección con el título "Inscripción". Pero si no es ese el caso, se deberán llenar las dos secciones tabulares, y el periodo de la sección de

En la nueva pantalla localice al trabajador que se va a ir a descansar. Cuando lo tenga en pantalla, podrá ver cual es su antigüedad en la empresa, y cuantos días de vacaciones le corresponden. También puede generar un reporte del registro de sus vacaciones.

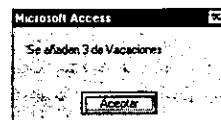


Indica cuantos días de vacaciones le corresponden en este periodo al trabajador en base a su antigüedad.

Indica cuantos días de vacaciones de periodos anteriores, puede disfrutar en este.

Indica cuantos días de vacaciones ya ha disfrutado en este periodo.

En la sección media marque el periodo en el que se cubrirán las vacaciones, ponga la fecha inicial y final de sus vacaciones, haga clic en el cuadrado, de la sección "Estado", para que aparezca una palomita.



Cuando haya dado clic en estado, aparece un mensaje como él de la caja de diálogo de la derecha.

Pero si se arrepiente y no se tomaran esas vacaciones, o no se han aprobado aun, puede dar otro clic estado para desactivar la palomita, tras lo cual el sistema le informa de la modificación.

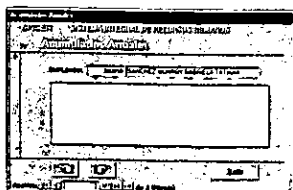


en

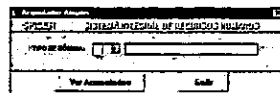
En cualquiera de los dos casos el sistema actualizara los días de vacaciones disponibles para el trabajador, con la finalidad de llevar un control exacto de este aspecto.

14)

Acumulado



ACUMULADO. En esta sección usted sólo debe elegir el tipo de NÓMINA sobre la que efectúara los cálculos, y el sistema le mostrara los acumulados de sueldo y días pertenecientes al ejercicio inmediato anterior fiscal del PTU, correspondientes al trabajador que tiene en pantalla. Si desea ver otro trabajador, puede desplazarse de un registro a otro mediante el uso de las dos manos apuntando.

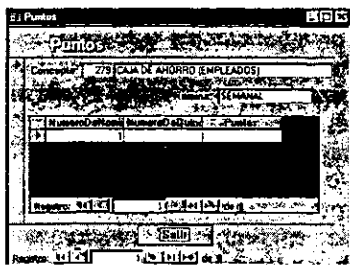


Cuando termine de un clic en Salir y regresara al menú principal de **Administración de Nóminas**.

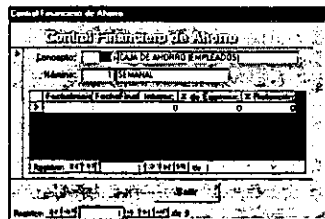
controlará mejor la forma en que maneja los préstamos asignados a que concepto quiere que se vaya el interés a un solo pago o bien el descuento normal.

20) **Puntos Por Nómina** *PUNTOS POR NÓMINA*. Una forma para llevar el rendimiento o beneficio por las aportaciones realizadas por los trabajadores a los conceptos de caja y fondo de ahorro, es la de brindarles puntos en base a las cantidades aportadas.

En esta sección el usuario podrá dar de alta el catálogo de puntos por los conceptos de caja y fondo de ahorro por nómina. Con este precepto, la tabla se debe de llenar en base al tipo de NÓMINA que afecta, el periodo del ejercicio, y cuantos puntos le corresponden a cada periodo. Al terminar solo debe dar un clic en Salir, y habrá terminado.

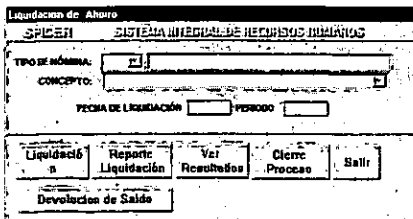


21) **Control Financiero** *CONTROL FINANCIERO*. El control financiero o control de ahorro, es una herramienta en la que usted puede de manera sencilla dar los parámetros iniciales de cada ejercicio. Para ello debe 1. Poner la fecha inicial y final del ejercicio en cuestión. 2. Cuanto aportara la empresa por ese concepto en el cuadro % Empresa. 3. % Retención, es la casilla en la que indica cuanto será retenido del total del ejercicio, para poder dar inicio a las operaciones del próximo ejercicio.



22) **Liquidación Ahorro** *LIQUIDACIÓN AHORRO*. Aquí usted cerrara las operaciones del ejercicio en cuestión. Para ello debe seguir los pasos siguientes, en el orden indicado.

1. Indicar el tipo de nómina
2. Establecer el tipo de concepto que será cerrado.
3. Dar un clic en "Liquidación".
4. Clic en "Reporte Liquidación" para realizar un respaldo escrito de las operaciones.
5. Clic en "Ver Resultados" para corroborar la veracidad de los mismos.
6. Cuando todo este correcto esta en condiciones de Cerrar el ejercicio. De clic en "Cierre Proceso".



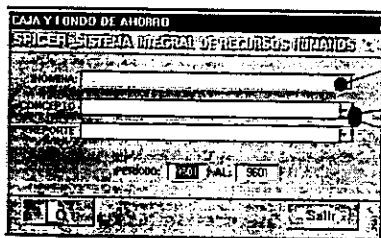
7. Haga clic en "Devolución de Saldo", para que el sistema reintegre a los trabajadores, el capital correspondiente al 10% retenido en el ejercicio anterior para dar inicio a este.
8. Al terminar, de clic en "Salir", y habrá terminado.

23)



REPORTE DE AHORRO. Al igual que la sección de

reporteria antes descrita, existen un apartado especial para los movimientos correspondientes a los ejercicios de caja y fondo de ahorro. Los cuales pueden ser mostrados en tres etapas diferentes: Préstamos, Acumulados y Preliminar. En la pantalla generada por el clic en el botón, usted puede elegir los periodos que desea analizar, así como el tipo de nómina de la que realizará el reporte. Préstamos le muestra un desplegado de todos los préstamos concedidos en el periodo solicitado. Acumulados tiene la función de mostrar el desglose de puntos en los periodos comprendidos, este deberá sacarlo cuando cierre la NÓMINA. Preliminar es un listado de aportaciones, préstamos y puntos de la NÓMINA en el periodo solicitados.



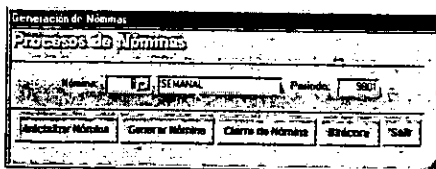
SEMANAL	1
QUINCENAL	2
MESESUAL	3
COOPERACIONAL	4
FINQUITOS	5
INDIVIDUAL CONEXIONISTAS	6
REPRESENTANTES	7
VALORES DEL BONO (PUNTO)	8

CAJA DE AHORRO (EMPLEADOS)	278
APORTACION AHORRO	285
FONDO DE...	...
ACUMULADOS	
PRESTAMOS	
PRELIMINAR	
UN SOLO PAGO	
FINQUITOS	

Procesos de Nómina

24) **PROCESOS DE NÓMINA.** Cuando ha terminado con todos los procedimientos anteriores, esta en condiciones de generar su nómina y solicitar al sistema el desglose de incidencias, cálculo de impuestos, retenciones, aportaciones, análisis de asistencia, etc.

Cuando esta seguro de que todos los movimientos para una nómina han sido realizados y transferidos a Nómina, puede dar un clic en el botón antes mostrado y obtendrá una pantalla como la que aquí se muestra.



1. Lo primero que debe hacer es inicializar la NÓMINA que pagara. Para ello ya debió previamente haber seleccionado el tipo de NÓMINA a cubrir y el periodo que se pagara.

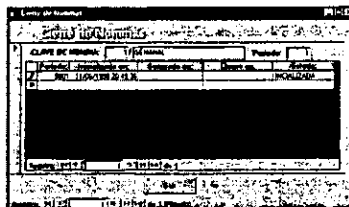
Para inicializar su nómina debe haber terminado de capturar todas las contrataciones y bajas sufridas en el periodo a pagar, pues el sistema realiza el cálculo de los días a cubrir como resultado del concepto 1 percepción regular. Si no lo hiciera así, es muy probable que surgan discrepancias en el resultado final de la nómina, pues

si quiere realizar un finiquito o contratar a alguien después de haber inicializado su nómina y pagar los conceptos correspondientes a esas incidencias en la misma nómina, estará realizando una contradicción lógica al sistema.

2. Después de haber inicializado, y habiéndose asegurado de que de los movimientos hasta ahora descritos, no omitido ninguno, está en condiciones de **Generar la Nómina**, mediante el uso del botón del mismo nombre. Este procedimiento tarda un poco, pues el se encarga de realizar todos los cálculos pertinentes y necesarios para el pago de la nómina.

3. Recuerde que No debe Cerrar NÓMINA, hasta que este seguro 100% de que ya no hay modificaciones a los conceptos y cálculos realizados en el punto No. 2, pues una vez realizado el depósito bancario y Cerrada la Nómina, no hay forma alguna de dar marcha atrás y corregir alguna inconsistencia. Por ello debe tener muy en cuenta cuando puede cerrar la nómina con toda tranquilidad.

4. Tenemos un botón que es un acceso a una bitácora de nóminas, para que sepa la fecha, hora y tipo de movimiento realizados en cada una de las nóminas que maneje con el SIRH. La pantalla generada es como la que aquí se muestra.



25) **SALIR**. El último botón que analizaremos en este "Proceso 2 Procesos de Nómina", es el que contiene la puerta con la flecha azul, y que no es otra cosa que el botón de Salir, con el cual termina su trabajo en el Módulo de Administración de Nóminas, y lo regresa al menú principal del SIRH.

Recuerde que para mayor claridad de los procesos y secuencias aquí descritos, puede consultar el capítulo 3 de este manual con los pasos uno a uno que debe realizar en la generación de sus nóminas.



Administración De Personal

El siguiente proceso a analizar es el que se refiere a las herramientas mediante las cuales podrá llevar un control muy específico de la gente que labora en su empresa a través de Evaluaciones de requerimiento corporativo, Movimientos a sueldo, Cambios en tabuladores, Requerimientos del IMSS, Cálculo de finiquitos, Revisión de históricos de sueldo, Generación de reportes entre otros.

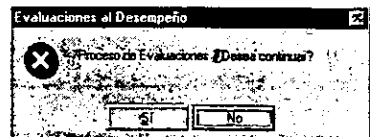



Comencemos ahora a revisar uno por uno todos los procedimientos permitidos en este proceso.

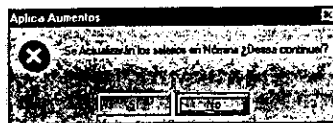
- 1) **Procesamiento de Evaluaciones y Sueldos** Existe en el ambiente del **SIRH** un módulo satélite denominado **CAPACITA**, el cual esta pensado para llevar el control de la capacitación tomada y asignada, así como las evaluaciones de carácter corporativo de todos y cada uno de los empleados de su empresa. Cuando ya se hayan efectuado las evaluaciones y generado las calificaciones correspondientes esta en posibilidades de afectar la nómina basándose en esas calificaciones.

Por política corporativa, **UNIK** ha establecido que a todos los empleados de las empresas pertenecientes al grupo se les apliquen dos evaluaciones de desempeño durante el año. Esta evaluación con todos los aspectos a considerar, ya esta contemplado en el módulo antes mencionado, este módulo y su interacción con el cuerpo central del **SIRH** es importante porque de los resultados obtenidos por el personal en esas evaluaciones, se moverán los niveles de salario.


Quando los gerentes le han avisado que ya se terminaron las evaluaciones, usted esta listo para dar clic en el botón que aparece al inicio de este apartado. Tras los cual aparecerá la pantalla aquí mostrada. Cuando de clic en Sí, el sistema realizara una serie de cálculos, destinados a estimar cual será el nuevo porcentaje de penetración del nivel y el sueldo de todos los empleados. Cuando termine estamos listos para el siguiente paso.



2)  Acto seguido dará clic en el botón con la leyenda "Aplica Aumentos a Nómina", en el que el sistema vuelve a mostrar una pantalla de confirmación, esta vez ya con los nuevos salarios estimados y calculados se procede a insertar estos en la nómina, para a partir de este momento realizar los pagos con el nuevo sueldo.



NOTA: Cuando se ha terminado de aplicar el cambio en los salarios, es importante que **ANTES DE GENERAR CUALQUIER NÓMINA**, vaya al módulo de Administración de Nóminas y ahí a Proceso de Integración, para generar el Sueldo Diario Integrado (SDI). Esto es importante, porque acaba de cambiar los sueldos, por lo tanto **DEBE** actualizar el SDI correspondiente, sino lo hace así, el resultado de su nómina tendrá varios errores importantes que le pueden causar problemas de tipo legal.


3)  Vamos a suponer que existe alguien a quien no se le aplicarán los cambios de sueldo como al resto del personal. Se trata de una excepción. Para tales efectos tenemos este procedimiento que brinda la oportunidad de deshacer las modificaciones que se efectuarán como resultado de las evaluaciones, pero antes de aplicar el proceso del punto 2, por que si lo quiere hacer posterior ya no le permitirá hacerlo, a menos que utilice otra de las opciones del sistema; pero, de ella hablaremos después.

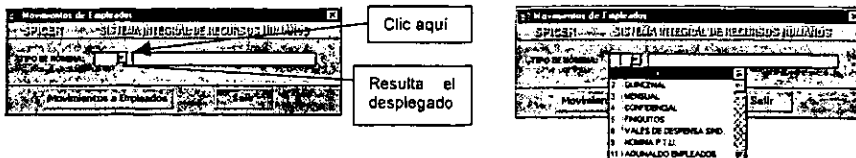
Cancelación de Aumentos

Nombre	Apellido	Fecha de Aumento	% Pasa	Estado	Cancelar
21008	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21009	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21010	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21011	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21012	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21013	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21014	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21015	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21016	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21017	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21018	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21019	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21020	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21021	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21022	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21023	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21024	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21025	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21026	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21027	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21028	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21029	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21030	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21031	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21032	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21033	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21034	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21035	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21036	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21037	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21038	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21039	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21040	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21041	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21042	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21043	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21044	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21045	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21046	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21047	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21048	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21049	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]
21050	ESPINALADO ESTEVA JOSE LUI	15/01/08	102.00	[A]	[X]

Quando requiere de cancelar algunas de las modificaciones tendrá que hacerlas en la pantalla resultado de oprimir el botón "Cancelación de Aumentos", e ir uno por uno de los empleados a modificar, es decir de manera individual.

Solo deberá tener cuidado de no modificar el estado del trabajador.

4)  Los movimientos Directos nos permiten realizar cambios en las características de desempeño de cada una de las personas empleadas en la empresa. Primero debe dar clic en el botón correspondiente y en la ventana que aparece dar clic en la flecha hacia abajo, para desplegar el menú colgante, como se muestra aquí.

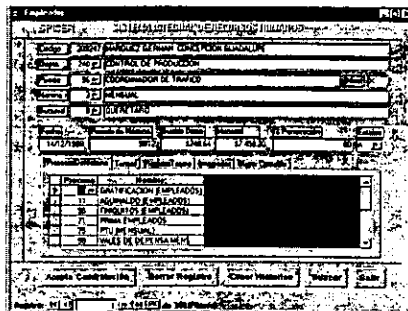


Seleccione el tipo de nómina que desea consultar y después en "Movimientos a Empleado" haga clic. Como resultado aparece la siguiente pantalla:

Aquí usted puede modificar el departamento, puesto, sueldo diario, etc. pero será de manera individual.

Para ello debe:

- ☞ Buscar a la persona que va a sufrir cambios en sus características, mediante el uso de las flechas de registro o con el botón para búsquedas rápidas .
- ☞ Clic en el botón Nuevo registro
- ☞ Realizar los cambios deseados, y **PONER LA FECHA Y EL PERIODO** a partir de la cual se tomara en cuenta el cambio.



Algo importante que debe notar es que cuando modifique el sueldo, la penetración se modificara de manera automática, por lo que **NUNCA** trate de modificar manualmente la penetración. Otro aspecto que modifica la penetración es el cambio de puesto, por lo que, cuando realice modificaciones a este respecto, el sistema siempre proporcionara el 80% de penetración, para tal efecto y en caso de que sea necesario, modifique nuevamente el sueldo diario para que se ajuste a los requerimientos reales.

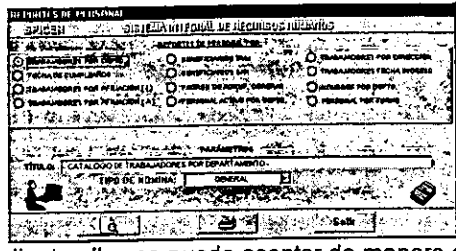
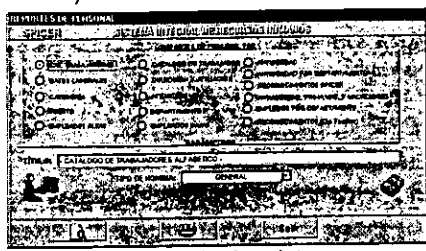
Recuerde que cuando termine sus cambios debe *Generar Los Integrados* mediante el uso del botón correspondiente, para posteriormente revisarlos con el botón *Ver Integrados*.

Posteriormente para generar los avisos correspondientes al IMSS, utilice el botón "Reportes" de la misma pantalla, y vera el generador de reportes correspondiente. Elija "Extemporáneo" si el aviso es posterior al plazo correspondiente, y el tipo de salario que le corresponde al trabajador en turno, el cual podrá verificar mediante el código que le muestra en el cuadro de desplegado al centro de la pantalla.




NOTA: Nuevamente, cuando se ha terminado de aplicar el cambio en los salarios, es importante que **ANTES DE GENERAR CUALQUIER NÓMINA**, vaya al módulo de Administración de Nóminas y ahí a Proceso de Integración, para generar el **NUEVO Sueldo Diario integrado (SDI)**. **Esto lo debe realizar cada vez que modifique el sueldo de alguien.**

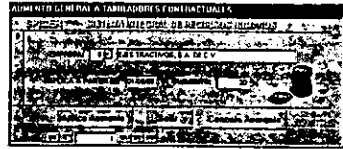
5) **Reportes (Cálculos)** Los reportes son un aspecto **Otros Reportes**




importante en cualquier sistema, pues mediante ellos se puede asentar de manera clara los resultados que se generan en el mismo, además es el objetivo final de todo sistema de información. Para ello el SIRH le proporciona dos grupos de reportes en este módulo, y se generan de la siguiente manera:

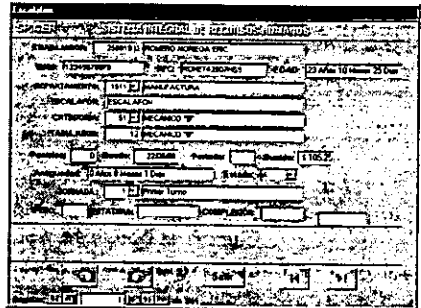
- ☞ Clic en el tipo de reporte que desea
- ☞ Clic en el Tipo de Nómina _
- ☞ Clic en el botón de vista preliminar (el que tiene la lupa)
- ☞ Clic en botón de Impresión (el de la impresora) de la pantalla de vista preliminar o si lo desea
- ☞ Desde la pantalla generadora de reportes Clic en el botón de impresión directa (el que tiene la impresora)

6)  Ahora veamos lo procedente a los aumentos de salario mediante el uso de tabuladores. Para ello el botón "Aumento a Tabuladores" nos permite indicar a partir de que fecha se aplica el aumento, el porcentaje a incrementar y los botones para confirmar el proceso. "Aplica aumento", se encarga de incrementar en la tabla de catálogos correspondiente el porcentaje indicado, mientras que "Cancela Aumento" deshace las indicaciones hechas.



NOTA: Recuerde que, cuando ha terminado de aplicar el cambio en los salarios, es importante que **ANTES DE GENERAR CUALQUIER NÓMINA**, vaya al módulo de Administración de Nóminas y ahí a Proceso de Integración, para generar el **NUEVO Sueldo Diario Integrado (SDI)**. **ESTO LO DEBE REALIZAR CADA VEZ QUE MODIFIQUE EL SUELDO DE ALGUIEN.**


7)  Cuando requiere de consultar únicamente la información de su personal sindicalizado, la pantalla idónea para hacerlo es ésta. Ya que aquí puede revisar de manera visual toda la información laboral de la gente sin el riesgo de modificar algo.

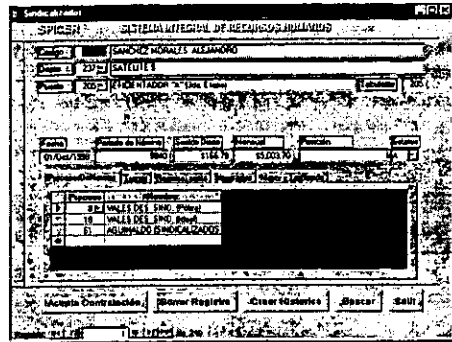


Su manejo es bastante sencillo de un clic en las manos apuntando para desplazarse un registro a la vez, ya sea hacia adelante o atrás, o las flechas con línea, para ir al primero o último registro respectivamente. Cuando termine, haga clic en "Salir" y regresara al menú principal de "Administración de Personal".

Movimiento de Escalafones

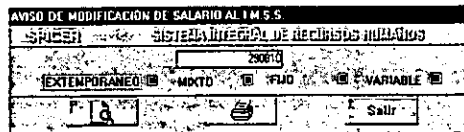
8) "Movimiento de escalafones" es un procedimiento con el cual el usuario puede modificar el departamento, escalafón, categoría, sueldo diario, si checa tarjeta o no, el turno, etc. Pero será de manera individual. Para ello debe:

- Buscar a la persona que va a sufrir cambios en sus características, mediante el uso de las flechas de registro o con el botón  para búsquedas rápidas.
- Clic en el botón "Nuevo registro".
- Realizar los cambios deseados, y **PONER LA FECHA Y EL PERIODO** a partir de la cual se tomara en cuenta el cambio.



Recuerde que cuando termine sus cambios igual que como lo hizo en Asignación, debe *Generar Los Integrados* mediante el uso del botón "Genera Integrado", para posteriormente revisarlos con el botón "Ver Integrados".

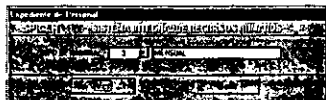
Posteriormente para generar los avisos correspondientes al IMSS, utilice el botón "Reportes" de la misma pantalla, y vera el generador de reportes correspondiente. Elija Extemporáneo si el aviso es posterior al plazo correspondiente y el tipo de salario que le corresponde al trabajador en turno, el cual podrá verificar mediante el código que le muestra en el cuadro de desplegado al centro de la pantalla.



NOTA: Nuevamente igual que en Asignación, cuando se ha terminado de aplicar el cambio en las características del trabajador, es importante que **ANTES DE GENERAR CUALQUIER NÓMINA**, vaya al módulo de Administración de Nóminas y ahí a Proceso de Integración, para generar el **NUEVO Sueldo Diario integrado (SDI)**. **Esto lo debe realizar cada vez que modifique el sueldo de alguien.**

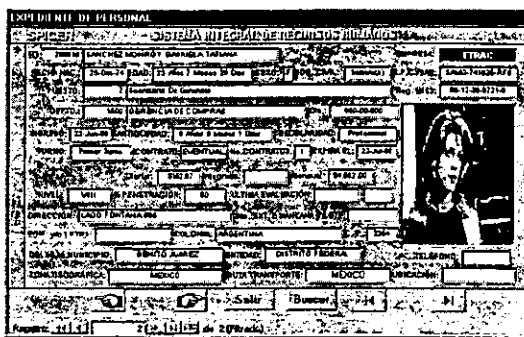
9) Expediente de Personal

La siguiente actividad que analizaremos es la de *Expediente De Personal* en la que también se puede consultar de manera detallada la información del personal ya sea sindicalizado o empleado, pero con la ventaja de que en este apartado podemos además de la información laboral revisar la información personal más relevante de cada persona.



Primero debe indicar el tipo de nómina que queremos consultar y después en la pantalla de trabajo, buscar a la persona, y revisar la información ahí mostrada.

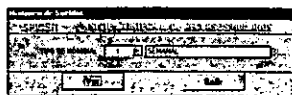
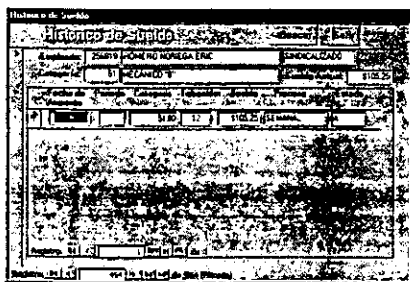
El manejo de la pantalla como podrá ver es idéntico a Consulta de Escalafón.



10) Historio de Sueldo

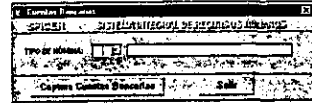
Un aspecto importante en las valoraciones y análisis del desempeño de su personal es también el histórico de sueldos que ha llevado durante su permanencia en la empresa, y para ello el procedimiento destinado a controlar este aspecto le

proporciona una pantalla donde

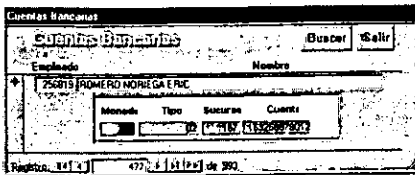


indicara a que nómina pertenece la persona a consultar, y una pantalla accesible en su presentación pues organiza toda esta información en registros escalonados, de tal forma que pueda ver cual es el avance de su gente en cuanto a salarios se refiere.

11) Cuentas Bancarias Posiblemente en su empresa se acostumbre pagar vía depósito bancario y para ello es necesario llevar un control muy estricto del número de cuenta de cada persona. El SIRH cuenta con un proceso destinado a esta función.

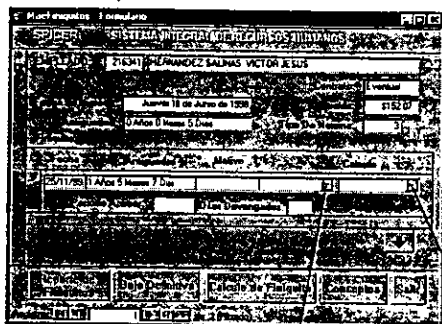
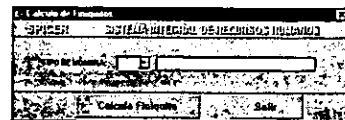


Lo único que se requiere de usted como usuario, es que cuando haya avisado al banco que va a realizar los pagos de la gente en su institución, y este le proporcione el número cuenta de cada persona, al dar clic en el botón correspondiente indique el tipo de nómina, posteriormente localice al trabajador en la pantalla de la derecha y digite con cuidado los números que le fueron asignados a cada quien. Para el proceso de pago, el sistema se encargara de generar el archivo de información para el banco, tal y como lo vimos con anterioridad en el proceso destinado a nóminas.



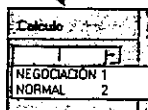
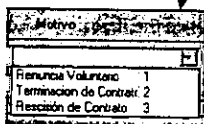
12) Cálculo de Liquidación Pasemos a uno de los aspectos más interesantes del SIRH, el cálculo de finiquitos. En este proceso le indicaremos al sistema que alguien va a dejar de laborar en la empresa por alguna razón, y queremos que realice todos los cálculos correspondientes para realizar su pago.

Primero como ya es costumbre indique el tipo de nómina en la que se encuentra la persona. Posterior a esto localice a la persona de la que se elaborara el finiquito. Esto lo puede hacer usando las flechas de registro en la parte inferior de la pantalla, o colocando el Mouse sobre el campo donde aparece



el código del empleado y dando clic con el botón derecho, esto invoca al menú de acción rápida de Windows, y en este elija la opción de orden ascendente o descendente según le convenga.

Ya con el empleado localizado, verifique que la información que se despliega sea la correcta. Posterior a esto digite la fecha en la cual se llevara a cabo la liquidación de la persona.



La antigüedad se calcula de manera automática. Ahora debe indicar cual es el motivo de la liquidación. En este aspecto cabe hacer la aclaración pertinente respecto a los tipos de liquidación que se muestran.

Renuncia voluntario: Este tipo de renuncia se aplica cuando el trabajador es quien decide retirarse de la empresa y el sistema se encarga de calcular

- Percepción del periodo
- Aguinaldo correspondiente
- Prima vacacional
- Aportación de caja más la aportación de la empresa
- Aportación de fondo más la aportación de la empresa
- Saldo de préstamos
- Préstamo de caja
- Préstamo de fondo
- Adeudo a la compañía
- Etc.

Terminación de contrato: Este tipo de renuncia se aplica cuando el contrato celebrado entre el trabajador (que debe ser de tipo eventual) y la empresa llega a su terminación y alguna de los partes o ambas deciden no renovar el convenio y el sistema se encarga de calcular

- Percepción del periodo

- Aguinaldo correspondiente
- Prima vacacional
- Aportación de caja más la aportación de la empresa
- Aportación de fondo más la aportación de la empresa
- Saldo de préstamos
- Préstamo de caja
- Préstamo de fondo
- Adeudo a la compañía
- Etc.

Rescisión de contrato: Este tipo de renuncia se aplica cuando la empresa decide rescindir el contrato por causa injustificada.

Un aspecto importante que cabe aclarar es el de prima vacacional el cual si se pagó de manera anticipada el sistema calcula la parte proporcional a descontar pagada en demasía en el periodo ordinario. En caso de no ser así pagara el proporcional correspondiente a los periodos y días devengados.

En la sección de cálculo también tiene dos opciones, negociación y normal. Estos dos procesos se encargan de realizar parámetros de cálculo para los pagos correspondientes.

Periodo a aplicar, es la casilla en la que usted le indica al sistema en que periodo se realizara el pago. Días devengados, aquí le indica al sistema cuantos días del periodo se le pagarán, puesto que no se le pagara todo el periodo.

Ahora bien, cuando vaya a finiquitar a la persona, y después de haber llenado la información descrita, los pasos que seguirá serán los siguientes:

1. Localización de la persona
2. Llenado de datos

The screenshot shows a form for employee data. Callouts indicate the following steps:

- 3. Clic para calcular el finiquito. (Points to a button labeled 'Calcula el Finiquito')
- 4. Clic aquí para pasar a la siguiente página. (Points to a right arrow button)
- 7. Clic aquí y genere sus reportes para revisión de datos. (Points to a button labeled 'Reporte de Flujos')
- 9. De clic en baja definitiva. (Points to a button labeled 'Baja Definitiva')

8. Generar Depósito Bancario en Administración De Nóminas.

The screenshot shows a form for generating a bank deposit. Callouts indicate the following steps:

- 5. Clic en este botón para avanzar un registro (Points to a right arrow button)
- 6. Clic en este botón para regresar un (Points to a left arrow button)

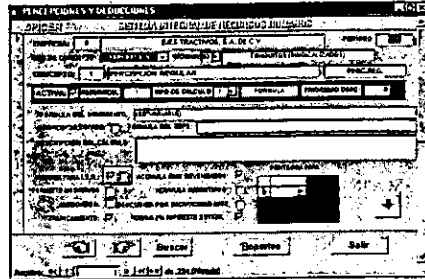
El objetivo de estos dos es para actualizar el registro en el desplegado en

Este proceso se va a realizar por todas y cada una de las personas que se vayan a finiquitar.

10. Cuando todos los finiquitos del periodo se hayan realizado y estén correctos, esta listo para terminar el proceso, cierre la nómina, en el proceso correspondiente en *Administración De Nóminas*.

Una aclaración pertinente es el caso de Baja Definitiva. Este botón es importante que tenga bien presente cuando dar el clic, debido a que el proceso es irreversible. Cuando da clic en este botón el trabajador es eliminado de las nóminas corrientes, pero su información de acumulados de aportación siguen activos en el sistema para los procesos de PTU, caja, fondo, etc. hasta la terminación del periodo.

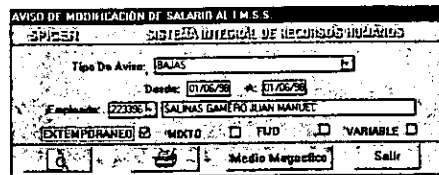
Un punto importante es el correspondiente al botón de "CONCEPTOS", el cual lo lleva a una pantalla en la que se pueden consultar los cálculos parametrizados de los conceptos involucrados en el cálculo de finiquitos. Sin embargo estos conceptos pertenecen a un módulo aparte denominado Catálogos, el cual analizaremos en el capítulo siguiente.



Impresión de Reportes al IMSS

14)

Una vez que hemos realizado cambios en el salario del personal, o finiquitado a alguien es una obligación legal, notificar al IMSS, respecto a estos movimientos. Para ello contamos con un botón especial que nos lleva a un generador de reportes, en el que indicamos el tipo de movimientos a realizar, las fechas en las que se generaron los movimientos, el empleado a quien se le realizaron los movimientos, si es extemporáneo o no, y el tipo de salario que se percibe.

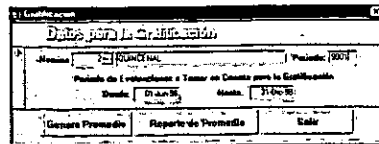


NOTA: Si el aviso es baja, no se indica el tipo de salario.

Observe que tiene la opción de Medio Magnético, la cual nos sirve cuando la notificación va a ser masiva, y en casos de modificación de salarios. Si se trata de casos particulares o bajas el reporte es impreso.

15) En grupo SPICER existe una política de gratificación anual al personal empleado. Esta gratificación es determinada con base al nivel de la persona y de las calificaciones obtenidas en sus evaluaciones corporativas.

Gratificación Anual



Entre a la pantalla e indique la nómina Quincenal, confidencial, etc. registre el periodo en que se realizaron las evaluaciones, y oprima el botón de Generar Promedio, posterior a esto genere el reporte correspondiente. Cuando termine, dirijase a Administración de nóminas, Procesos de Nómina y genere la nómina de gratificación.

16) **SALIR.**

Cuando terminó con todos estos procesos de *Administración De Personal* solo nos resta terminar los movimientos en este módulo, para ello oprima el botón mostrado aquí y regresara al menú principal del SIRH





Catálogos

Uno de los aspectos que hemos estado mencionado con mucha frecuencia, es el que se refiere a las tablas de parámetros que topan muchos de los conceptos de las incidencias que se manejan en los diversos procedimientos que se han descrito

hasta este momento.



La sección de catálogos es un módulo especial que aunque no requiere de mucha operación por parte del usuario, es muy importante tenerlo bajo un control estricto para evitar que los diversos cálculos efectuados por el sistema, puedan resultar equivocados, por los parámetros utilizados.

CATÁLOGOS			
Comités		Tipos de Nómina	
Departamentos	Percepciones y Deducciones		
Organizaciones	Parámetros		Clasificadores
Tipos de Trabajadores		Tablas Int.	
Niveles y Tabulador de		Factores para los gral Salario	
Descripciones de Puestos		Jornales Laborales	
Políticas de Compensaciones		Fracciones de Jornada Laboral	
Categorías		Días Festivos	
Estados/Controlados		Zonas Geográficas	
Cargos/Empleados		Contratos	
Políticas de Seguro		Calificación de Evoluciones	
Exclusiones		Salir	

Cuando en la pantalla del menú principal del SIRH, haga clic en el botón que lleva la leyenda CATÁLOGOS, aparece la pantalla principal de este módulo, tal y como se muestra en la pantalla lateral de este párrafo.



1) **DATOS DE LA EMPRESA.** En muchos de los procesos que se realizan como avisos a dependencias gubernamentales (IMSS; INFONAVIT, HACIENDA, etc.), así como en la generación de reportes, se registra información referente a la empresa, para ello, es necesario que le indique al sistema todas las características que identifican a su organización.

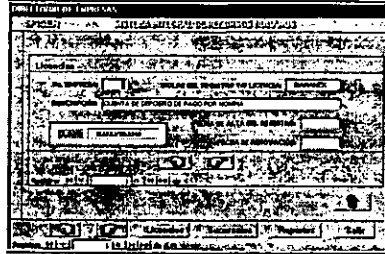
Esto lo puede hacer mediante el empleo del botón mostrado antes, y la captura de los datos solicitados en la pantalla generada a partir de él.

Los nombres del primer y segundo responsables son vitales. El nombre del primer responsable es el que se maneja como representante en los avisos a la diferentes dependencias gubernamentales, por lo que su captura debe ser de carácter obligatoria.

Como se puede ver la información solicitada es de carácter general pero vital para muchos de los procesos. En la parte inferior de la pantalla cuenta con tres botones principales, como son "Licencias", "Sucursales" y "Reportes".

LICENCIAS

Aquí puede dar de alta las claves de diversas cuentas, como puede ser la del banco donde realice sus depósitos, pagos o cobros; la despensa, el registro con el IMSS, el fondo de ahorro, etc.



SUCURSALES



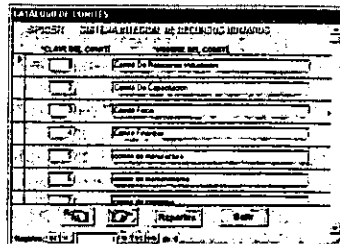
Puede darse el caso de que su empresa cuenta con una o más sucursales, para ello debe indicar al sistema cuales son estas. Para ello utilice este botón que le permite dar la clave de identificación de la sucursal, así como su localización general. Una vez terminado de un clic en la flecha negra apuntando hacia arriba para regresar a la pantalla inicial o en "Salir" si desea terminar la captura.

REPORTES

En este botón obtendremos un informe escrito de todas las características antes capturadas.

2.) **Comités** COMITÉS.

Se pueden crear por diversas razones en su empresa diferentes comités, en los que se involucre a ciertos departamentos para realizar actividades muy específicas, razón por la cual es necesario llevar un control de cuales son los comités que existen y su nombre distintivo.

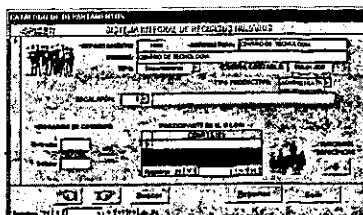


Cuando entre a esta sección verá la pantalla mostrada a la derecha, y en la cual a manera tabular, podrá dar de alta, baja o realizar cambios en las características distintivas de dichos comités. Así mismo si lo desea puede generar un reporte impreso de la información contenida en este apartado.

3.) **Departamentos** DEPARTAMENTOS.

Cuando se le indica al sistema de la existencia de un departamento es necesario indicarle algunos aspectos adicionales para que pueda reconocerlo y clasificarlo adecuadamente.

En varios de los reportes que se generan en el sistema se incluyen opciones de clasificación vía departamento; cuando da de alta algún trabajador debe indicar a que departamento pertenece; cuando genera controles presupuestales, debe manejar aspectos departamentales. Por todo lo anterior es necesario dar de alta cualquier departamento que exista en su empresa.

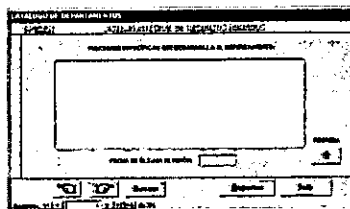


Si observa la pantalla adjunta al párrafo anterior vera una sección tabular denominada "COMITÉS", en ella indicara a que comités pertenece este departamento (comités que por supuesto fueron dados de alta en el procedimiento anterior).

El cuadro denominado Cuenta Contable, no es otra cosa que el centro de costo, que se designe a cada departamento. Así mismo el menú colgante llamado "Tipo Productivo", identifica a este como directo, indirecto o administrativo, información necesaria para la generación de la póliza contable. Además asigne el escalafón a que pertenece el departamento.

El resto de la información como puede ver es de carácter informativo y su llenado es bastante sencillo, pues la pantalla le pide de manera muy clara la información que necesita.

Si observa con cuidado encontrara una flecha de color azul, que como en las otras ocasiones que la hemos tenido, nos conduce a una segunda página de la misma pantalla.



Esta segunda página, contiene una sección destinada a la descripción detallada de las actividades del departamento. Esto es con carácter informativo a manera de comentario.

5) NIVELES Y TABULADOR DE SUELDO.

Aquí usted indicara al sistema sobre la base de las políticas corporativas, cuales son los topes paramétricos de los diferentes niveles en los puestos que estén registrados en el catálogo correspondiente.

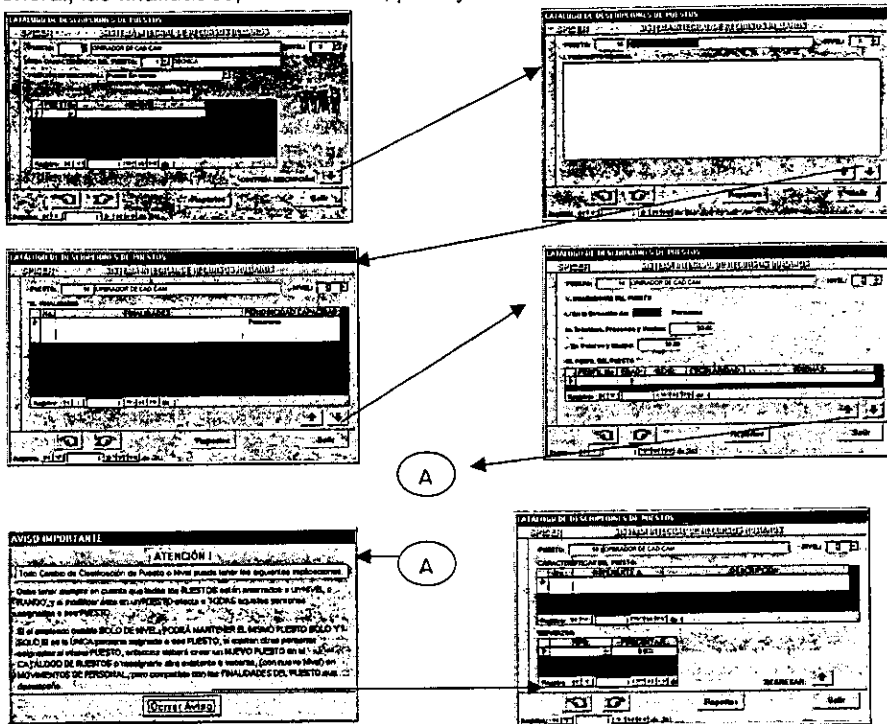
The image shows a screenshot of a software interface titled "CASILLAS DE NIVELES". It contains a grid of input fields for defining salary parameters for different levels. The grid has 4 rows and 4 columns of main cells. Each cell contains a sub-table with 4 columns: "Máximo", "Mínimo", "Porcentaje", and "Índice". The "Índice" column contains the value "0" in all cells. The "Porcentaje" column contains values like "80%", "100%", and "120%". The "Máximo" and "Mínimo" columns contain numerical values. The interface also includes a "TABULADOR" section at the top right and a "TABLA DE NIVELES" section at the bottom.

La información que aquí se requiere la emite el corporativo e implica el porcentaje de penetración que se aplicará (80%,100%,120%). Recuerde que cuando el porcentaje de penetración sobrepase el tope del nivel, debe incrementar el nivel del trabajador.

El cuadro denominado AMPLITUD, esta destinado a estimar el diferencial entre el tabulador mínimo y el tabulador máximo, tomando como base el tabulador medio. Por lo que, por políticas corporativas, en esta casilla siempre debe poner el valor 50. En lo referente a la casilla Índice de Programación, se estipulan los porcentajes de incremento salarial entre un nivel y otro. Por esta razón marque siempre 0 en este cuadro.

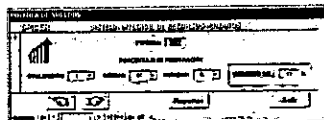
6) **Descripciones de Puestos** DESCRIPCIONES DE PUESTOS.

Para indicarle al sistema cuales son los puestos que existen en la organización, es necesario dar de alta cada uno de estos, asignándole su nivel salarial, las características, la posición estructural, que puestos dependen de el, el propósito general, las finalidades, la dimensión, perfil y esfuerzos del mismo.



7) **Política de Compensaciones** POLÍTICA DE COMPENSACIONES.

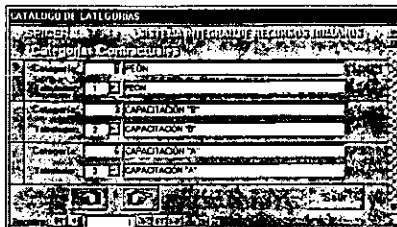
Aquí el usuario capturara basándose en los preceptos del corporativo, cual es el porcentaje de penetración máximo y mínimo, para determinar cual es el porcentaje de incremento para la calificación de la evaluación corporativa designada en el cuadro evaluación.



De tal forma que para todos aquellos empleados que obtengan la calificación en cuestión obtengan este porcentaje de incremento en su salario.

8) **Categorías** CATEGORÍAS

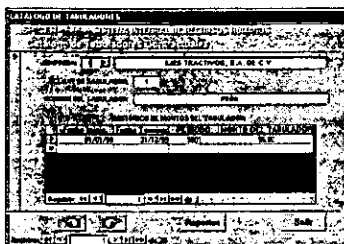
Hemos mencionado anteriormente la existencia de categorías, y estas determinan algunas de las características contractuales de los trabajadores. Ahora a estas categorías se les debe asignar un tabulador, por lo cual del menú colgante escoja el que le pertenece a cada categoría.



NOTA: primero de alta los tabuladores contractuales en TABULADOR CONTRACTUAL.

9) **Tabulador Contractual** TABULADOR CONTRACTUAL

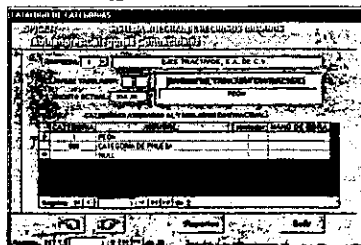
En esta sección se dan de alta los diferentes tabuladores que manejamos en la sección anterior. Lo que tiene que hacer es digitar la clave y nombre del tabulador, el periodo inicial y las fechas entre las cuales se maneja el monto que usted le indique en la sección final de este tabulador.



Categorías/Tabuladores

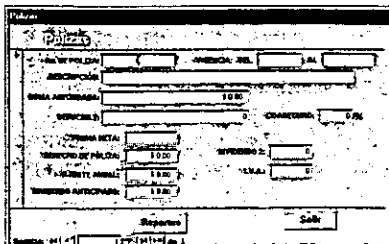
10) CATEGORÍAS / TABULADORES.

En la sección anterior usted dio de alta los tabuladores, y al entrar aquí los relacionara con las categorías dadas de alta. De forma que correlacionara los tabuladores con las categorías en las que estarán registrados los diversos puestos de los trabajadores que laboran en la empresa.



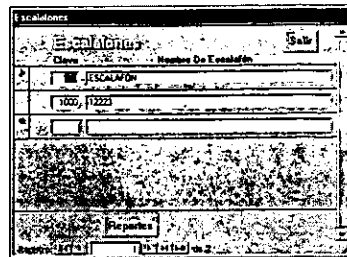
11) **Pólizas de Seguros** PÓLIZAS DE SEGUROS

Es común que las empresas cuenten con diversas pólizas de seguro para proteger los bienes de las mismas. Si en su organización este es el caso, aquí puede registrar toda la información referente a ellos para que usted tenga un control exacto.



12) **Escalafones** ESCALAFONES

Para el caso de los escalafones que se mencionaron cuando manejamos los de los Departamentos, también puede crear su catálogo correspondiente. Solo tiene que digitar la clave y nombre del escalafón y habrá terminado.

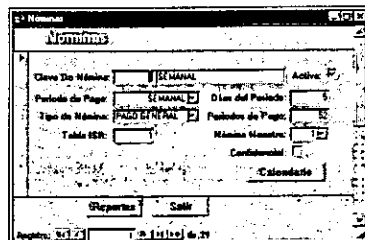


13) **Tipos de Nómina** TIPOS DE NÓMINA

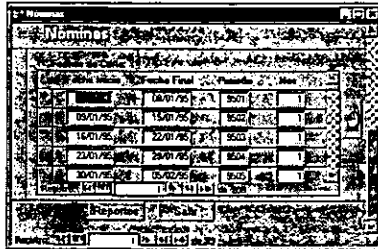
Habrà notado que a lo largo de los diferentes tópicos tratados en este manual, en varias ocasiones se ha hecho una diferencia entre empleados y sindicalizados. Ello se debe principalmente a que se manejan en nóminas diferentes. Aquí puede decirle al sistema los diferentes tipo de nómina que se manejan en la empresa, estos pueden ser Semanal, Catorcenal, Quincenal, Mensual, Confidencial, etc., debe recordar que estas son nóminas maestras, pues dentro de ellas puede tener nóminas de finiquitos, prima vacacional, etc., que son en realidad subnóminas, por ello debe marcar a que nómina maestra pertenecen.

Vamos a dar de alta una nómina maestra:

1. En clave de nómina registre el código y posteriormente la descripción.
2. En periodo de pago seleccione si será semanal, quincenal, etc.
3. En tipo de nómina seleccione si va a ser para PAGO GENERAL O ADICIONAL.
4. En tablas de ISR, marque 1, 2, y/o 3 si quiere que esta nómina calcule el impuesto con tarifas semanales, quincenales o mensuales respectivamente.
5. En ACTIVA marque la casilla para que aparezca la "palomita", para indicarle al sistema que esta nómina esta activa para cualquier proceso.
6. En DIAS DEL PERIODO registre el número de días que quiere que el sistema inserte al momento de inicializar nómina en el concepto 1, percepción regular
7. En periodos de pago, registre cuantos periodos de pago va a realizar usted en el año. Por ejemplo en la semanal 52, quincenal 24, y mensual 12.
8. En NÓMINA MAESTRA, registre de que nómina maestra es subnómina ésta.



9. En CONFIDENCIAL, active la casilla para que aparezca la "palomita", y se indique al sistema si la nómina será de acceso limitado para ciertas personas autorizadas.
10. Ahora bien dé de alta el calendario anual de la nómina dando un clic en el botón "CALENDARIO", el cual lo lleva a la pantalla siguiente:

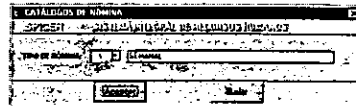


11. En fecha de inicio registre el día, mes y año en el cual inicia la semana o bien el periodo laboral.
12. En fecha final registre el día, mes y año de la terminación de esta.
13. En periodo registre año y periodo que identifican a estas fechas.
14. En mes registre 1, 2, 3, etc. para enero, febrero, marzo, etc., respectivamente.

14) Percepciones y Deduciones PERCEPCIONES Y DEDUCCIONES

En este procedimiento se determina la naturaleza de los diferentes conceptos que se verán involucrados en el cálculo de sus nóminas, como recordara en el procedimiento anterior dimos de alta los tipos de nóminas que se manejan en su empresa, y se determinó cuales serian las nóminas maestras.

En esta sección se pueden parametrizar de manera independiente los cálculos para cada tipo de nómina, para ello, cuando ingrese a este procedimiento lo primero que deberá indicar, es el tipo de nómina que va manejar. Cuando este por dar de alta un nuevo concepto, SIEMPRE debe indicar a partir de que periodo se comenzará a contemplar dicho concepto.

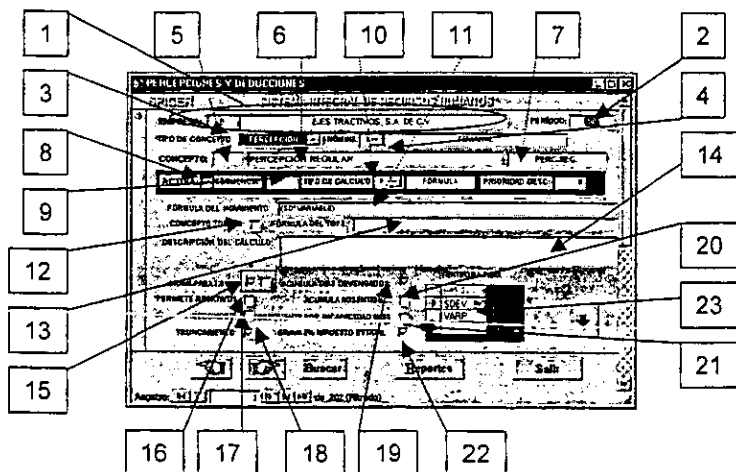


Este procedimiento por su naturaleza y alcance es el corazón del SIRH. A continuación se describen las partes de la pantalla.

1. Clave y nombre de la empresa. el sistema por default le da el nombre de su empresa que registro en datos de la empresa
2. Periodo a partir del cual se comienza el concepto. Registre el periodo en el cual dio de alta al concepto
3. Tipo de concepto. La forma en que aplicara en el proceso de cálculo (percepción, deducción, cálculo, estadístico). Escoja percepción o deducción si el concepto es de esta naturaleza y si desea que aparezca

- en su nómina como tal y si necesita realizar algún cálculo por este concepto pero que no se registre en su nómina como percepción o deducción escoja cálculo o estadístico y que el sistema realice la operación indicada, más sin embargo no lo mostrará en la nómina
4. Verifica el tipo de nómina en el que aplicara el concepto. El sistema por default le proporciona la nómina en la cual esta realizando este movimiento.
 5. Clave del concepto. Registre el número o código del concepto para que el sistema identifique a este, como tal el SÍRH únicamente acepta valores numéricos y no podrá repetirse.
 6. Nombre del concepto. Registre la descripción de concepto.
 7. Abreviatura del concepto. Registre la descripción abreviada del concepto para que aparezca en su recibo de pago.
 8. Indicación de activa, es decir, si el concepto se calcula o no en el proceso de nómina.
 9. Secuencia, registre en que orden se calculara el concepto. El sistema primero busca la secuencia de cálculo posteriormente busca el número del concepto y en ese orden el sistema calcula cada uno de los conceptos.
 10. Tipo de cálculo. De que forma se parametriza el concepto para algunos procesos particulares (recuerde que en varios procedimientos del proceso 2 se hizo mención de que algunos conceptos dependían de la forma en que estuvieran registradas en esta tabla).
 11. Formula del movimiento. Aquí el usuario indica la operación aritmética.
Claves utilizadas en la parametrización
 - (a) "@M" sirve para que el sistema registre en este concepto el monto de otro concepto.
 - (b) "@V" sirve para que el sistema registre en este concepto la variable de otro concepto.
 - (c) "SD" indica al sistema que traiga el sueldo de cada trabajador.
 - (d) "VARIABLE" indica al sistema que tome el valor registrado en este para realizar alguna operación aritmética
 - (e) "SDI" indica al sistema que traiga el salario diario integrado del IMSS de cada trabajador.
 - (f) "SDINFOPOR" indica al sistema que remita esta macro a la tabla INFO. Para identificar que porcentaje de crédito le corresponde.
 - (g) "SDIINFO" indica al sistema que traiga el salario diario integrado del INFONAVIT de cada trabajador.
 - (h) "*" Indica al sistema que multiplique.
 - (i) "+" Indica al sistema una suma
 - (j) "SMG" Indica al sistema que traiga el salario mínimo general de la zona geográfica registrado en parámetros
 - (k) "SMGDF" Indica al sistema que traiga el salario mínimo general del Distrito Federal registrado en parámetros.
 12. Concepto topado. Aquí se indica si existe algún valor máximo para el concepto en cuestión. Es decir cuanto es el importe máximo que se debe

- registrar en este concepto como resultado final, por ejemplo, si quiere que el máximo del concepto, sea 100, debe escribirlo como 1*100.
13. Descripción del cálculo. Aquí pondrá alguna nota aclaratoria de las características del concepto (opcional).
 14. Marque sólo si el total del concepto servirá como base gravable para el cálculo de ISR.
 15. Se activa sólo en el caso de que quiera que el resultado se registre en nómina cuando sea menor a cero.
 16. Acerca la cifra decimal al entero más próximo.
 17. Elimina los decimales de la cantidad obtenida, a sólo dos dígitos. Márquelo siempre.
 18. Sirve para acumular las variables del concepto.
 19. Indique si el concepto no se debe considerar para cotizar por ausentismo, como por ejemplo las faltas injustificadas.
 20. Indique si el concepto no se debe considerar para cotizar por incapacidades, como por ejemplo el concepto 57 incapacidades.
 21. Este caso es particular para algunas entidades federativas, en las cuales se debe considerar un impuesto estatal. Determine si en su estado se aplica este impuesto, de ser así active o desactive la casilla para cada concepto que desee considerar para base de impuesto estatal.
 22. Integra para. En esta casilla, determine que conceptos son sujetos para que se tome su importe como variable para el IMSS, INFO, SDIIN.

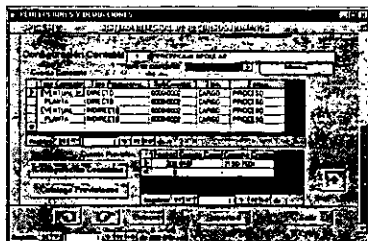


Cada concepto de nómina debe tener asignada una cuenta contable, para que el sistema pueda proporcionarle la información veraz cuando usted genere la póliza contable.

Vamos a analizar las tres formas en que se pueden parametrizar las cuentas contables:

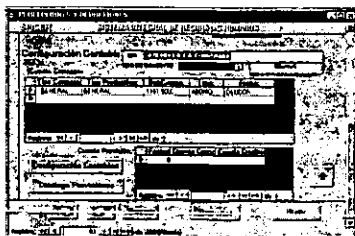
- 1) En la pantalla que se muestra abajo se cuenta con un título que dice "FORMA", aquí el usuario le indicara como quiere que se conforme la cuenta contable, es decir:
 1. PROCESO. Parametricemos la cuenta contable del concepto 1 (percepción regular) de una nómina semanal.
 - a. En tipo de contrato, seleccione EVENTUAL
 - b. En tipo productivo DIRECTO
 - c. En SUBCUENTA, a que subcuenta se va este concepto
 - d. En TIPO, CARGO
 - e. En FORMA, PROCESO
 2. En la siguiente fila, capture PLANTA
 - a. En tipo productivo DIRECTO
 - b. En SUBCUENTA, a que subcuenta se va este concepto
 - c. En TIPO, CARGO
 - d. En FORMA, PROCESO
 3. Capture EVENTUAL, en tipo productivo indirecto
 - a. En tipo productivo INDIRECTO
 - b. En SUBCUENTA, a que subcuenta se va este concepto
 - c. En TIPO, CARGO
 - d. En FORMA, PROCESO
 4. en PLANTA:
 - a. En tipo productivo INDIRECTO
 - b. En SUBCUENTA, a que subcuenta se va este concepto
 - c. En TIPO, CARGO
 - d. En FORMA, PROCESO

Es decir, el sistema identificara cada trabajador en base a su tipo de contrato, en que departamento y el tipo productivo de este, conformando así la cuenta contable, la cual se forma agregando la cuenta de mayor más el centro de costo (que capturó el usuario en el catálogo de departamentos) más la subcuenta que acaba de registrar.



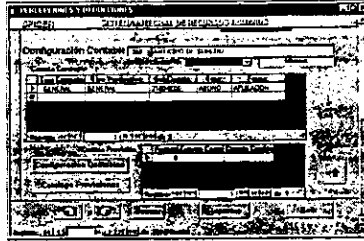
- 2) En la pantalla que se muestra abajo se cuenta con un título que dice "FORMA", aquí el usuario le indicara como quiere que se conforme la cuenta contable, es decir:
1. **DEUDOR.** Parametricemos la cuenta contable del concepto 258 (adeudo a la compañía) de una nómina semanal.
 - e. En tipo de contrato, seleccione GENERAL
 - f. En tipo productivo GENERAL
 - g. En SUBCUENTA, a que subcuenta se va este concepto
 - h. En TIPO, ABONO
 - i. En FORMA, DEUDOR

Es decir, el sistema no hará distinción alguna del tipo de contrato del trabajador, del departamento, ni del tipo productivo, conformando la cuenta contable con la cuenta que registro, más las 4 últimas cifras del código del trabajador.



- 3) En la pantalla que se muestra abajo se cuenta con un título que dice "FORMA", aquí el usuario le indicara como quiere que se conforme la cuenta contable, es decir:
1. **APLICACIÓN.** Parametricemos la cuenta contable del concepto 259 (anticipo de sueldo), de una nómina semanal.
 - j. En tipo de contrato, seleccione GENERAL
 - k. En tipo productivo GENERAL
 - l. En SUBCUENTA, a que subcuenta se va este concepto

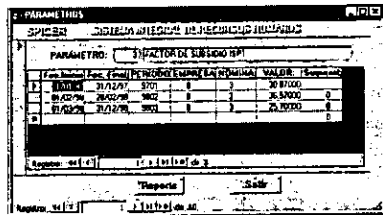
- m. En TIPO, ABONO
- n. En FORMA, APLICACIÓN



Es decir, el sistema no hará distinción alguna del tipo de contrato del trabajador, del departamento, ni del tipo productivo, conformando la cuenta contable con la que registro.

15) **Parámetros** PARÁMETROS

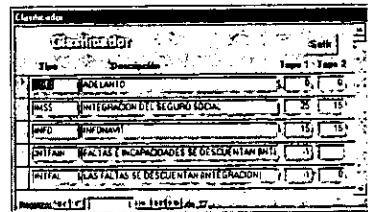
Aquí el usuario dará de alta los parámetros generales a aplicar en el sistema. Los aspectos que debe contemplar son: Factor De Subsidio acreditable de ISPT, Base De Cotización Del Deducible Para G.M.M., Salario Diario Mínimo General (De La Entidad), Días Bimestres, SAR, Salario Mínimo General (D.F.), Tasa Fija Sindicato, Tasa Interés Fondo Ahorro, Tasa Caja De Ahorro (Empleados), Días Del Mes.



Clasificador

16) CLASIFICADOR

En la sección de clasificador, el usuario debe dar de alta macros de procedimiento que se apliquen en el sistema. No se preocupe por el manejo de estas pues están dadas por el sistema, en el caso de que se requirieran macros adicionales son contempladas durante el diseño del SIRH, contacte al personal de ADEP para su atención.



En esta pantalla deberá cambiar el tope del salario mínimo general del D.F. para los procesos de integración del IMSS, y del INFONAVIT, así como en

percepciones y deducciones el tope de la fórmula del concepto 60001 de todas las nóminas cada 1° de Julio.

17) **Tablas del ISR:** TABLAS DEL ISR

En el Diario Oficial de la Federación, cada seis meses se emite de manera regular una nueva tabla de tarifas para el cálculo del ISR. Estas están dadas en función de los artículos 80, 80A, y 80B. El usuario las captura tal cual se publican en el mencionado diario. La única excepción es la que se refiere al artículo 80B, el cual menciona en una de sus columnas el título Crédito al salario, este en el SIRH lo capturara en la casilla de Cuota Fija y en la casilla de Impuesto marginal registre el valor 0.

Artículo	Rango de Salario	Tarifa	Impuesto Marginal
80	0 - 150.00	2%	15%
80	150.01 - 300.00	6%	16%
80	300.01 - 600.00	10%	17%
80	600.01 - 1,200.00	15%	18%
80	1,200.01 - 2,400.00	20%	19%
80	2,400.01 - 4,800.00	25%	20%
80	4,800.01 - 9,600.00	30%	21%
80	9,600.01 - 19,200.00	35%	22%
80	19,200.01 - 38,400.00	40%	23%
80	38,400.01 - 76,800.00	45%	24%
80	76,800.01 - 153,600.00	50%	25%
80	153,600.01 - 307,200.00	55%	26%
80	307,200.01 - 614,400.00	60%	27%
80	614,400.01 - 1,228,800.00	65%	28%
80	1,228,800.01 - 2,457,600.00	70%	29%
80	2,457,600.01 - 4,915,200.00	75%	30%
80	4,915,200.01 - 9,830,400.00	80%	31%
80	9,830,400.01 - 19,660,800.00	85%	32%
80	19,660,800.01 - 39,321,600.00	90%	33%
80	39,321,600.01 - 78,643,200.00	95%	34%
80	78,643,200.01 - 157,286,400.00	100%	35%

18) **Tablas Info:** TABLAS DEL INFONAVIT

En esta sección sucede la misma situación que en la anterior, solo que aquí las tablas las emite el mismo Instituto. La captura será idéntica a las tablas emitidas.

Categoría	Valor
1	15%
2	16%
3	17%
4	18%
5	19%
6	20%
7	21%
8	22%
9	23%
10	24%
11	25%
12	26%
13	27%
14	28%
15	29%
16	30%
17	31%
18	32%
19	33%
20	34%
21	35%

19) **Factores para Integrar Salario:** FACTORES PARA INTEGRAR SALARIO

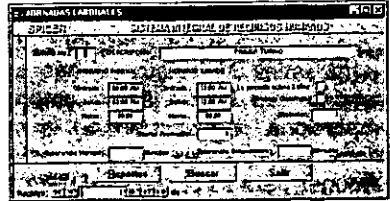
El objetivo de esta pantalla es que el usuario capture en base a las tablas de antigüedad propias de la empresa cuantos días de vacaciones, que porcentaje de prima y cuantos días de aguinaldo le corresponden a los trabajadores en base a su antigüedad. Recuerde que la antigüedad debe ser registrada en meses, y en la casilla denominada Mínima, debe siempre para el primer registro indicar -1, en lugar de 0, esto es sobre la base de requisitos de cálculo del SIRH:

Form fields include:

- Antigüedad (Meses)
- Porcentaje de Prima
- Días de Vacaciones
- Días de Aguinaldo
- Factores de Integración
- Botones: Guardar, Cancelar, Salir

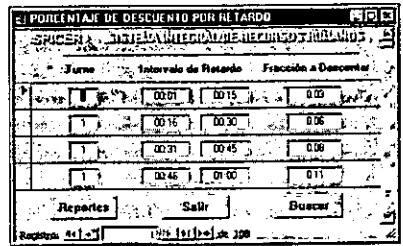
20) **Jornadas Laborales** JORNADAS LABORALES

En el proceso 2, en el procedimiento de permisos, le mencionamos la importancia de indicar los turnos en los que el trabajador desempeña sus actividades, pero para que el sistema sepa en que horario considera cada turno, debe dar de alta todos y cada uno de los turnos que se manejen en su empresa en este módulo. Además tiene la opción de generar un reporte de estos.



21) **Fracciones de Jornada Laboral** FRACCIONES DE JORNADA LABORAL

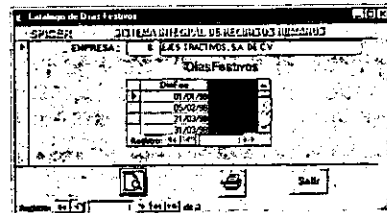
En el mismo proceso 2, procedimiento permisos, se mencionó que el sistema le proporcionaba en automático las fracciones a descontar correspondientes a faltas o permisos. Estas fracciones se dan de alta basándose en tolerancias y criterios que se estipulen en la empresa, y es en esta tabla donde le indicara el sistema cual es el rango de tiempo a considerar por cada fracción de tiempo.



El usuario debe indicar el turno, la fracción de tiempo y la fracción a descontar. Cuando haya terminado haga clic en "Salir".

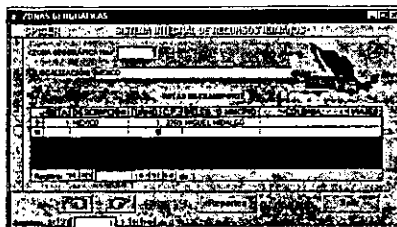
22) **Días Festivos** DÍAS FESTIVOS

Existen por ley, días en que se debe de permitir al trabajador descanso obligatorio, y otros días que por razones particulares la empresa concede. Estas fechas deben de indicarse al sistema en la tabla de Días Festivos, llenando en la sección tabular central las fechas que serán concedidas para tal efecto.



23) **Zonas Geográficas** ZONAS GEOGRÁFICAS

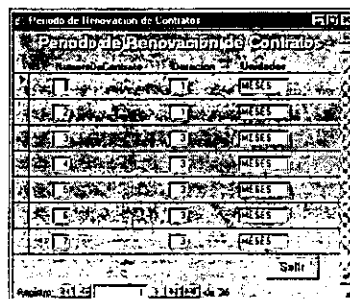
Una de las intenciones del SIRH, es la de mantener un registro lo más exacto de cada persona que labore en la empresa, para ello proporciona una sección en la que el usuario puede indicar cuales son las rutas de acceso más comunes para su personal al dirigirse a laborar.



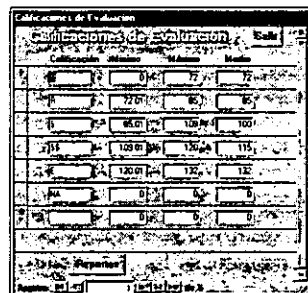
Esta pantalla es opcional, y no requiere un llenado obligatorio.

24) **Contratos** CONTRATOS

Cuando realice contratos eventuales, estos deberán renovarse con cierta periodicidad (si recuerda en el proceso1, Cambios de contrato, existe una sección en la que se indica la fecha de renovación de contrato), el sistema sabra cuando debe generar la renovación de contratos solo a través de esta tabla, en la cual usted indicara cuales son las políticas de periodicidad de contratación de su empresa.



25) **Calificación de Evaluaciones** CALIFICACIÓN DE EVALUACIONES. Cuando se generan las evaluaciones corporativas se obtienen calificaciones que deben ser contempladas de entre un rango de puntuaciones. Esos rangos deben ser capturados en la tabla Calificación de Evaluaciones, basándose en los lineamientos corporativos para tales efectos.



26) **Salir** SALIR

Cuando ha terminado de revisar, generar y modificar sus tablas de parámetros para todo el entorno del SIRH, esta listo para abandonar el módulo. Y para ello debe utilizar el botón con la leyenda "Salir".

Apéndice 1

Código Fuente

```

1575.
1576.     consulta = " ClaveDeEmpleado = "
        & cla_emp
1577.     montos = DLookup("TotalDias",
        "AcumuladosPTU", consulta)
1578.
1579.     If IsNull(montos) Then
1580.         bus_val = "0"
1581.     Else
1582.         clave1 = CStr(montos)
1583.         bus_val = clave1
1584.     End If
1585.     ----- Hasta Aquí
        Calculo de PTU - '98
1586.
1587.
1588.     Case "DIASAGUINALDO"
1589.         montos = DLookup("aguinaldo",
        "FactorFin", "ClaveDeEmpleado = " &
        cla_emp)
1590.     If IsNull(montos) Then
1591.         montos = 0
1592.     End If
1593.     clave1 = CStr(montos)
1594.     bus_val = clave1
1595.
1596.
1597.     Case "DIASVAC"
1598.         montos = DLookup("Dias_vac",
        "FactorFin", "ClaveDeEmpleado = " &
        cla_emp)
1599.     If IsNull(montos) Then
1600.         montos = 0
1601.     End If
1602.     clave1 = CStr(montos)
1603.     bus_val = clave1
1604.
1605.     Case "PRIMAVAC"
1606.         montos = DLookup("pri_vac",
        "FactorFin", "ClaveDeEmpleado = " &
        cla_emp)
1607.     If IsNull(montos) Then
1608.         montos = 0
1609.     End If
1610.     clave1 = CStr(montos)
1611.     bus_val = clave1
1612.
1613.     Case "PORCEVAC"
1614.         montos =
        DLookup("porC_primaVac", "FactorFin",
        "ClaveDeEmpleado = " & cla_emp)
1615.     If IsNull(montos) Then
1616.         montos = 0
1617.     End If
1618.     clave1 = CStr(montos)
1619.     bus_val = clave1

1620.
1621.     Case "DIASANTI"
1622.         montos = DLookup("DiasAño",
        "FactorFin", "ClaveDeEmpleado = " &
        cla_emp)
1623.     If IsNull(montos) Then
1624.         montos = 0
1625.     End If
1626.
1627.     montos1 = DLookup("TodosDias",
        "FactorFin", "ClaveDeEmpleado = " &
        cla_emp)
1628.     If IsNull(montos1) Then
1629.         montos1 = 0
1630.     End If
1631.
1632.     If montos > montos1 Then
1633.         montos = montos1
1634.     End If
1635.
1636.     clave1 = CStr(montos)
1637.     bus_val = clave1
1638.
1639.     Case "XDIAGUI"
1640.
1641.         xLaFecha = DLookup("FechaAlta",
        "Contratación", "ClaveDeEmpleado = " &
        cla_emp)
1642.     If IsNull(xLaFecha) Then
1643.         xDias = 0
1644.     Else
1645.         xper = DLookup("Periodo",
        "Nómina", "ClaveDeNomina = " & numero)
1646.         If IsNull(xper) Then
1647.             xper = 3
1648.         End If
1649.
1650.         cadena = "Periodo = " & xper & "
        and NumeroDeQuincena = " & quincena
1651.         XFecha =
        DLookup("FechaInicio", "FechasDeCorte",
        cadena)
1652.         x año = CVDate("31/12/" &
        Year(XFecha))
1653.         If Year(xLaFecha) <
        Year(XFecha) Then
1654.             xlni = CVDate("01/01/" &
        Year(XFecha))
1655.         Else
1656.             xlni = xLaFecha
1657.         End If
1658.
1659.         xFin = CVDate("31/12/" &
        Year(XFecha))
1660.         xDias = DateDiff("d", xlni, xFin) +
        1

```

```

1661.      End If
1662.
1663.      xMes = DateDiff("yyyy",
xLaFecha, xFin)
1664.      'xFecVir = DateAdd("yyyy",
xmes, xLaFecha)
1665.      xDiasVir = 0 'DateDiff("d",
xFecVir, xFin) + 1
1666.      xMes = CLng(((xMes * 12) +
(xDiasVir / 30)) * 100) / 100
1667.      xAgui = DLookup("Aguinaldo",
"Factores", xMes & "between niv_min and
niv_max ")
1668.      If IsNull(xAgui) Then
1669.          xAgui = 0
1670.      End If
1671.
1672.      xDias = xAgui * xDias
1673.
1674.      bus_val = CStr(xDias)
1675.
1676.      Case "XDIAGUIPRO"
1677.          ' Stop
1678.          xLaFecha = DLookup("FechaAlta",
"Contratación", "ClaveDeEmpleado = " &
cla_emp)
1679.          If IsNull(xLaFecha) Then
1680.              xDias = 0
1681.          Else
1682.              xper = DLookup("Periodo",
"Nómina", "ClaveDeNomina = " & numero)
1683.              If IsNull(xper) Then
1684.                  xper = 3
1685.              End If
1686.
1687.              cadena = "Periodo = " & xper & "
and NumeroDeQuincena = " & quincena
1688.              XFecha =
DLookup("FechaInicio", "FechasDeCorte",
cadena)
1689.              año = CVDDate("31/12/" &
Year(XFecha))
1690.              If Year(xLaFecha) <
Year(XFecha) Then
1691.                  xlni = CVDDate("01/01/" &
Year(XFecha))
1692.              Else
1693.                  xlni = xLaFecha
1694.              End If
1695.
1696.              xFin = CVDDate("31/12/" &
Year(XFecha))
1697.              xDias = DateDiff("d", xlni, xFin) +
1
1698.      End If
1699.
1700.
1701.      xTipoCon = DLookup("Tipo",
"Contratación", "ClaveDeEmpleado = " &
cla_emp)
1702.      If IsNull(xTipoCon) Then
1703.          xTipoCon = "PLANTA"
1704.      End If
1705.
1706.      If xTipoCon = "PLANTA" Then
1707.          xDias = 365
1708.      Else
1709.          xDias = xDias
1710.      End If
1711.
1712.
1713.
1714.          bus_val = CStr(xDias)
1715.      Case "XPrimaVac"
1716.          montos = LaPrimaVac(numero,
quincena, cla_emp)
1717.          If IsNull(montos) Then
1718.              montos = 0
1719.          End If
1720.          bus_val = CStr(montos)
1721.
1722.
1723.      Case "xPrimaVacPro"
1724.          montos = LaPrimaVacPro(numero,
quincena, cla_emp)
1725.          If IsNull(montos) Then
1726.              montos = 0
1727.          End If
1728.          bus_val = CStr(montos)
1729.          -----
1730.          '
1731.          ' Impuestos
1732.          '
1733.          -----
1734.          Case "ISPTEjes"
1735.
1736.          If numero = 4 Then
1737.              xnum = -1 ' Para Usar Tabla
Mensual
1738.          Else
1739.              If numero = 2 Then
1740.                  xnum = 0 ' Para Usar Tabla
Quincenal
1741.              Else
1742.                  xnum = 0 ' Para Usar Tabla
Semanal
1743.              End If
1744.          End If
1745.
1746.          aux_qui = quincena
1747.          Set isrSQL =
DBActiva.QueryDefs("BusTotPerT")

```



```

1748.     isrSQL.Parameters("num_qui") =
    quincena
1749.     isrSQL.Parameters("num_emp") =
    cia_emp
1750.     isrSQL.Parameters("num_nom") =
    numero
1751.
1752.     Set res_Sel =
    isrSQL.OpenRecordset()
1753.     If res_Sel.EOF Or
    IsNull(res_Sel![monto]) Then
1754.         Valor = 0
1755.     Else
1756.         valore = res_Sel![monto]
1757.     End If
1758.
1759.     Valor = cal_ispt(valore, quincena,
    cla_emp, "N", 1, numero + (xnum), 1, 0)
1760.
1761.     criterio = "ClaveDeEmpleado = " &
    cla_emp & " and NumerodeQuincena = " &
    quincena & " and ClaveDeConcepto = " & 37
1762.     monto1 = DLookup("MontoMovi",
    "MovinominaT", criterio)
1763.     xBase = valore
1764.     If Not IsNull(monto1) Or monto1 > 0
    Then
1765.         valore = valore - monto1
1766.         aumento = cal_ispt(valore,
    quincena, cla_emp, "N", 1, numero +
    (xnum), 1, 0)
1767.     If (Valor - aumento) > 0 Then
1768.         aumento = Valor - aumento
1769.         Set TempSQL =
    DBActiva.QueryDefs("ModifComedorT")
1770.
    TempSQL.Parameters("num_qui") =
    quincena
1771.     TempSQL.Parameters("cia_con")
    = 37
1772.
    TempSQL.Parameters("cia_emp") =
    cia_emp
1773.
1774.     TempSQL.Parameters("mon_act") =
    CDbI(Format(aumento, "000000.00"))
1775.     TempSQL.Execute
1776.     TempSQL.Close
1777.
1778.     End If
1779.     Else
1780.         aumento = 0
1781.     End If
1782.
1783.
1784.     valore = xBase + aumento
1785.     Valor = cal_ispt(valore, quincena,
    cla_emp, "N", 1, numero + (xnum), 1, 1)
1786.     If Valor < 0 And (numero <> 20 And
    numero <> 99) Then
    '1
1787.         Call inserta_exe(Valor, quincena,
    cla_emp, "N", numero, 8)
1788.     End If
1789.
1790.
1791.     dob_val = CDbI(Valor)
1792.     cad_val = CStr(dob_val)
1793.     bus_val = cad_val
1794.
1795.     Case "ISPT"
1796.     'Stop
1797.     montos =
    ImpuestoSR(CLng(cla_emp), CInt(numero),
    CInt(quincena))
1798.     bus_val = CStr(montos)
1799.
1800.     Case "ISPTM"
1801.     ' Stop
1802.     KKDiasDePer = 30.41667
1803.     aux_qui = quincena
1804.     Set isrSQL =
    DBActiva.QueryDefs("BusTotPerT")
1805.     isrSQL.Parameters("num_qui") =
    quincena
1806.     isrSQL.Parameters("num_emp") =
    cla_emp
1807.     isrSQL.Parameters("num_nom") =
    numero
1808.
1809.     Set res_Sel =
    isrSQL.OpenRecordset()
1810.     If res_Sel.EOF Or
    IsNull(res_Sel![monto]) Then
1811.         Valor = 0
1812.     Else
1813.         valore = res_Sel![monto]
1814.     End If
1815.     zznumero = DLookup("Periodo",
    "Nómina", "ClaveDeNomina = " & numero)
1816.
1817.     If zznumero = 3 Then
1818.         xAcumulado = Null
1819.         KKDiasDePer = 30
1820.     Else
1821.         cadena = " NumeroDeQuincena >=
    " & kkQui_ini & " and NumeroDeQuincena
    <= " & (quincena - 1)
1822.         cadena = cadena & " and
    NumeroDeNomina = " & numero

```

```

1823.     cadena = cadena & " and
         ClaveDeEmpleado = " & cla_emp
1824.     xAcumulado =
         DSum("MontoGravado", "AcumGravado",
         cadena)
1825.     End If
1826.
1827.     kkDiasAlPer = DLookup("Dias",
         "DiasDevengados", "ClaveDeEmpleado = "
         & cla_emp)
1828.     If kkDiasAlPer <= 0 Then
1829.         kkDiasAlPer = 1
1830.     End If
1831.     If IsNull(xAcumulado) Then
1832.         xAcumulado = 0
1833.     End If
1834.     xvalore = valore
1835.     valore = ((xvalore + xAcumulado) /
         kkDiasAlPer) * KKDiasDePer
1836.     Valor = cal_ispt(valore, quincena,
         cla_emp, "N", 3, 3, 1, 0)
1837.     Valor = (Valor / KKDiasDePer) *
         kkDiasAlPer
1838.
1839.     If zznumero = 3 Then
1840.         'cadena = " NumeroDeQuincena >=
         " & kkQui_ini & " and NumeroDeQuincena
         <= " & (quincena)
1841.         xAcum = Null
1842.         xCre = Null
1843.         Else
1844.
1845.             cadena = " NumeroDeQuincena
             >= " & kkQui_ini & " and
             NumeroDeQuincena <= " & (quincena - 1)
1846.             cadena = cadena & " and
             NumeroDeNomina = " & numero
1847.             cadena = cadena & " and
             ClaveDeEmpleado = " & cla_emp
1848.             cadena = cadena & " and
             ClaveDeConcepto in (250 , 294)"
1849.             xAcum =
             DSum("MontoGravado", "AcumiSPT",
             cadena)
1850.             xCre = Null
1851.
1852.         End If
1853.         If IsNull(xAcum) Then
1854.             xAcum = 0
1855.         End If
1856.
1857.         If IsNull(xCre) Then
1858.             xCre = 0
1859.         End If
1860.
1861.         Valor = Valor - (xAcum) - (xCre)
1862.
1863.         If Valor < 0 Then
1864.             Call inserta_exe(Valor,
             quincena, cla_emp, "N", numero, 8)
1865.             Valor = 0
1866.             End If
1867.
1868.             dob_val = CDbl(Valor)
1869.             cad_val = CStr(dob_val)
1870.             bus_val = cad_val
1871.
1872.         Case "ISPTMR" '
1873.             Stop
1874.             KKDiasDePer = 30.41667
1875.             aux_qui = quincena
1876.             Set isrSQL =
             DBActiva.QueryDefs("BusTotPerT")
1877.             isrSQL.Parameters("num_qui") =
             quincena
1878.             isrSQL.Parameters("num_emp") =
             cla_emp
1879.             isrSQL.Parameters("num_nom") =
             numero
1880.
1881.             Set res_Sel =
             isrSQL.OpenRecordset()
1882.             If res_Sel.EOF Or
             IsNull(res_Sel![monto]) Then
1883.                 Valor = 0
1884.             Else
1885.                 valore = res_Sel![monto]
1886.             End If
1887.             zznumero = DLookup("Periodo",
             "Nómina", "ClaveDeNomina = " & numero)
1888.
1889.             If zznumero = 3 Then
1890.                 xAcumulado = Null
1891.                 KKDiasDePer = 30
1892.             Else
1893.                 cadena = " NumeroDeQuincena >=
                 " & kkQui_ini & " and NumeroDeQuincena
                 <= " & (quincena - 1)
1894.                 cadena = cadena & " and
                 NumeroDeNomina = " & numero
1895.                 cadena = cadena & " and
                 ClaveDeEmpleado = " & cla_emp
1896.                 xAcumulado =
                 DSum("MontoGravado", "AcumGravado",
                 cadena)
1897.             End If
1898.
1899.             kkDiasAlPer = DLookup("Dias",
             "DiasDevengados", "ClaveDeEmpleado = "
             & cla_emp)
1900.             If kkDiasAlPer <= 0 Then

```

```

1901.    kkDiasAlPer = 1
1902.    End If
1903.    If IsNull(xAcumulado) Then
1904.        xAcumulado = 0
1905.    End If
1906.    xvalore = valore
1907.    valore = ((xvalore + xAcumulado) /
    kkDiasAlPer) * KKDiasDePer
1908.    'ImpuestoREGLA(MontoGravado
    As Double, xEmpledo As Long, xNomina As
    Integer, xQuincena As Integer
1909.        Valor =
    ImpuestoREGLA(CDbI(valore), cla_emp, 1,
    quincena)
1910.        'Valor = cal_ispt(valore, quincena,
    cla_emp, "N", 3, 3, 1)
1911.        Valor = (Valor / KKDiasDePer) *
    kkDiasAlPer
1912.
1913.    If zznumero = 3 Then
1914.        'cadena = " NumeroDeQuincena >=
    " & kkQui_ini & " and NumeroDeQuincena
    <= " & (quincena)
1915.        xAcum = Null
1916.        xCre = Null
1917.    Else
1918.
1919.        cadena = " NumeroDeQuincena
    >= " & kkQui_ini & " and
    NumeroDeQuincena <= " & (quincena - 1)
1920.        cadena = cadena & " and
    NumeroDeNomina = " & numero
1921.        cadena = cadena & " and
    ClaveDeEmpleado = " & cla_emp
1922.        cadena = cadena & " and
    ClaveDeConcepto in (250 , 294)"
1923.        xAcum =
    DSum("MontoGravado", "AcumiSPT",
    cadena)
1924.        xCre = Null
1925.
1926.    End If
1927.    If IsNull(xAcum) Then
1928.        xAcum = 0
1929.    End If
1930.
1931.    If IsNull(xCre) Then
1932.        xCre = 0
1933.    End If
1934.
1935.    Valor = Valor - (xAcum) - (xCre)
1936.    If Valor < 0 Then
1937.        Call inserta_exe(Valor,
    quincena, cla_emp, "N", numero, 8)
1938.        Valor = 0
1939.    End If

1940.
1941.
1942.        dob_val = CDbI(Valor)
1943.        cad_val = CStr(dob_val)
1944.        bus_val = cad_val
1945.    '-----
1946.        '-----
    Fin de Impuestos
1947.    '-----
1948.    Case "FACTOR"
1949.        fec_ani = DLookup("FechaAlta",
    "Contratación", "ClaveDeEmpleado = " &
    cla_emp)
1950.    If Not IsNull(fec_ani) Then
1951.        Meses = fec_fac(quincena, fec_ani,
    numero)
1952.        condicion = " " & Meses & "
    between niv_min and niv_max "
1953.        Factor = DLookup("fac_tot",
    "factores", condicion)
1954.        If IsNull(Factor) Then
1955.            Factor = 0
1956.        End If
1957.        cad_val = CStr(Factor)
1958.        bus_val = cad_val
1959.    Else
1960.        bus_val = "0"
1961.    End If
1962.
1963.    Case "TOTPEREXT"
1964.        aux_qui = quincena
1965.        Set conemp =
    DBActiva.QueryDefs("TotPerExt")
1966.        conemp.Parameters("cla_emp") =
    cla_emp
1967.        conemp.Parameters("num_qui") =
    aux_qui
1968.        Set res_Sel =
    conemp.OpenRecordset()
1969.        If Not res_Sel.EOF Then
1970.            res_Sel.MoveFirst
1971.            If Not IsNull(res_Sel![monto]) Then
1972.                cad_val = CStr(res_Sel![monto])
1973.            Else
1974.                cad_val = "0"
1975.            End If
1976.            bus_val = cad_val
1977.        Else
1978.            bus_val = "0"
1979.        End If
1980.        res_Sel.Close
1981.    '-----
    Finiquitos
1982.    Case "DEVCAJA"
1983.        Stop

```

```

1984.      consulta = "ClaveDeEmpleado =
           " & cla_emp
1985.      consulta = consulta & " and
           ClaveDeConcepto = " & 285
1986.
1987.      num_cla =
           DLookup("AportacionAcumulada",
           "InsCajaFondo", consulta)
1988.
1989.      If IsNull(num_cla) Then
1990.          bus_val = "0"
1991.      Else
1992.          clave1 = CStr(num_cla)
1993.          bus_val = clave1
1994.      End If
1995.      Case "RETCAJA"
1996.          Stop
1997.          consulta = "ClaveDeEmpleado =
           " & cla_emp
1998.          consulta = consulta & " and
           ClaveDeConcepto = " & 285
1999.
2000.      num_cla =
           DLookup("RetencionP", "InsCajaFondo",
           consulta)
2001.
2002.      If IsNull(num_cla) Then
2003.          bus_val = "0"
2004.      Else
2005.          clave1 = CStr(num_cla)
2006.          bus_val = clave1
2007.      End If
2008.
2009.      Case "DEVCAJAE"
2010.          Stop
2011.          consulta = "ClaveDeEmpleado =
           " & cla_emp
2012.          consulta = consulta & " and
           ClaveDeConcepto = " & 279
2013.
2014.      num_cla =
           DLookup("AportacionAcumulada",
           "InsCajaFondo", consulta)
2015.
2016.      If IsNull(num_cla) Then
2017.          bus_val = "0"
2018.      Else
2019.          clave1 = CStr(num_cla)
2020.          bus_val = clave1
2021.      End If
2022.
2023.      Case "RETCAJAE"
2024.          Stop
2025.          consulta = "ClaveDeEmpleado =
           " & cla_emp
2026.          consulta = consulta & " and
           ClaveDeConcepto = " & 279
2027.
2028.          num_cla =
           DLookup("RetencionP", "InsCajaFondo",
           consulta)
2029.
2030.          If IsNull(num_cla) Then
2031.              bus_val = "0"
2032.          Else
2033.              clave1 = CStr(num_cla)
2034.              bus_val = clave1
2035.          End If
2036.          Case "DESPRECAJ"
2037.              Stop
2038.              Set TempSQL =
           DBActive.CreateQueryDef("")
2039.              cad = "Delete From
           EISaldoDePrestamos"
2040.              TempSQL.SQL = cad
2041.              TempSQL.Execute
2042.
2043.              Set TempSQL =
           DBActive.QueryDefs("MeteEISaldoDePresta
           mos")
2044.              TempSQL.Parameters("xNomina") = numero
           - 52
2045.              TempSQL.Parameters("xPrestamo") = 280
2046.              TempSQL.Execute
2047.              cadena = "ClaveDeEmpleado = "
           & cla_emp
2048.
2049.              xPrestamo =
           DLookup("xMontoDePrestamo",
           "ResumenSaldo", cadena)
2050.              xDescuento =
           DLookup("xMontoPagado",
           "ResumenSaldo", cadena)
2051.              If IsNull(xPrestamo) Then
2052.                  xPrestamo = 0
2053.              End If
2054.
2055.              If IsNull(xDescuento) Then
2056.                  xDescuento = 0
2057.              End If
2058.
2059.              num_cla = xPrestamo -
           xDescuento
2060.
2061.              clave1 = CStr(num_cla)
2062.              bus_val = clave1
2063.
2064.              Case "DEVFONDO"
2065.                  Stop

```

```

2066.      consulta = "ClaveDeEmpleado =
" & cla_emp
2067.      consulta = consulta & " and
ClaveDeConcepto = " & 290
2068.
2069.      num_cla =
DLookup("AportacionAcumulada",
"InsCajaFondo", consulta)
2070.
2071.      If IsNull(num_cla) Then
2072.          bus_val = "0"
2073.      Else
2074.          clave1 = CStr(num_cla)
2075.          bus_val = clave1
2076.      End If
2077.
2078.      Case "DESPREFON"
2079.          Stop
2080.          Set TempSQL =
DBActiva.CreateQueryDef("")
2081.          cad = "Delete From
EISaldoDePrestamos"
2082.          TempSQL.SQL = cad
2083.          TempSQL.Execute
2084.
2085.          Set TempSQL =
DBActiva.QueryDefs("MeteEISaldoDePresta
mos")
2086.          TempSQL.Parameters("xNomina") = numero
- 52
2087.          TempSQL.Parameters("xPrestamo") = 291
2088.          TempSQL.Execute
2089.          cadena = "ClaveDeEmpleado = "
& cla_emp
2090.
2091.          xPrestamo =
DLookup("xMontoDePrestamo",
"ResumenSaldo", cadena)
2092.          xDescuento =
DLookup("xMontoPagado",
"ResumenSaldo", cadena)
2093.          If IsNull(xPrestamo) Then
2094.              xPrestamo = 0
2095.          End If
2096.
2097.          If IsNull(xDescuento) Then
2098.              xDescuento = 0
2099.          End If
2100.
2101.          num_cla = xPrestamo -
xDescuento
2102.
2103.          clave1 = CStr(num_cla)
2104.          bus_val = clave1
2105.      Case "FACNEG"
2106.          cadena = "ClaveDeEmpleado =
" & cla_emp
2107.          xFacNeg =
DLookup("FactorNeg", "Finiquitos", cadena)
2108.          If IsNull(xFacNeg) Then
2109.              xFacNeg = 100
2110.          End If
2111.          xFacNeg = xFacNeg / 100
2112.          clave1 = CStr(xFacNeg)
2113.          bus_val = clave1
2114.          Case "ADEUDOC"
2115.              cadena = "ClaveDeEmpleado =
" & cla_emp
2116.              cadena = cadena & " and
ClaveDeConcepto = " & cla_mov
2117.              xFacNeg = DLookup("Saldo",
"AdeudoCom", cadena)
2118.              If IsNull(xFacNeg) Then
2119.                  xFacNeg = 0
2120.              End If
2121.              clave1 = CStr(xFacNeg)
2122.              bus_val = clave1
2123.          Case "ADEUDOCC"
2124.              cadena = "ClaveDeEmpleado =
" & cla_emp
2125.              cadena = cadena & " and
ClaveDeConcepto = " & 280
2126.              xFacNeg = DLookup("Saldo",
"AdeudoCom", cadena)
2127.              If IsNull(xFacNeg) Then
2128.                  xFacNeg = 0
2129.              End If
2130.              clave1 = CStr(xFacNeg)
2131.              bus_val = clave1
2132.
2133.          Case "ADEUDOCF"
2134.              cadena = "ClaveDeEmpleado =
" & cla_emp
2135.              cadena = cadena & " and
ClaveDeConcepto = " & 291
2136.              xFacNeg = DLookup("Saldo",
"AdeudoCom", cadena)
2137.              If IsNull(xFacNeg) Then
2138.                  xFacNeg = 0
2139.              End If
2140.              clave1 = CStr(xFacNeg)
2141.              bus_val = clave1
2142.
2143.          Case "SALDOVAC"
2144.              montos = DLookup("DiasSaldo",
"MaeVacaciones", "ClaveDeEmpleado = " &
cla_emp)
2145.              If IsNull(montos) Then
2146.                  montos = 0
2147.              End If

```

```

2148.     clave1 = CStr(montos)
2149.     bus_val = clave1
2150.
2151.
2152. ' Fin finiquitos ETRAC
2153.
2154.     Case "APCAJA"
2155.         sel_mov = "ClaveDeEmpleado = " &
CStr(cla_emp)
2156.         sel_mov = sel_mov & " AND
ClaveDeConcepto = " & cla_mov
2157.         xTot_Per =
DSum("AportacionAcumulada",
"InsCajaFondo", sel_mov)
2158.         If IsNull(xTot_Per) Then
2159.             xTot_Per = 0
2160.         End If
2161.         bus_val = CStr(xTot_Per)
2162.
2163.     Case "RCAJA"
2164.         sel_mov = "ClaveDeEmpleado = " &
CStr(cla_emp)
2165.         sel_mov = sel_mov & " AND
ClaveDeConcepto = " & cla_mov
2166.         xTot_Per = DSum("Retencion",
"InsCajaFondo", sel_mov)
2167.         If IsNull(xTot_Per) Then
2168.             xTot_Per = 0
2169.         End If
2170.         bus_val = CStr(xTot_Per)
2171.
2172.
2173.     _Case "APFONDO"
2174.         sel_mov = "ClaveDeEmpleado = " &
CStr(cla_emp)
2175.         sel_mov = sel_mov & " AND
ClaveDeConcepto = " & 290
2176.         xTot_Per =
DSum("AportacionAcumulada",
"InsCajaFondo", sel_mov)
2177.         If IsNull(xTot_Per) Then
2178.             xTot_Per = 0
2179.         End If
2180.         bus_val = CStr(xTot_Per)
2181.
2182.     Case "ADEUDOCAJA"
2183.
2184.         sel_mov = "ClaveDeEmpleado = " &
CStr(cla_emp)
2185.         sel_mov = sel_mov & " AND
ClaveDeConceptoPres = " & cla_mov
2186.         xTot_Per = DLookup("NetoDebe",
"EstadoBenditox", sel_mov)
2187.
2188.         If IsNull(xTot_Per) Then
2189.             xTot_Per = 0
2190.
2191.         End If
2192.         bus_val = CStr(xTot_Per)
2193.
2194.     Case "ADEUDOFONDO"
2195.         'Stop
2196.         sel_mov = "ClaveDeEmpleado = "
& CStr(cla_emp)
2197.         sel_mov = sel_mov & " AND
ClaveDeConceptoPres = " & cla_mov
2198.         xTot_Per = DLookup("NetoDebe",
"EstadoBenditox", sel_mov)
2199.
2200.         If IsNull(xTot_Per) Then
2201.             xTot_Per = 0
2202.         End If
2203.         bus_val = CStr(xTot_Per)
2204.
2205.     Case "ADEUDOEMPRESA"
2206.         'Stop
2207.         sel_mov = "ClaveDeEmpleado = " &
CStr(cla_emp) & " and ClaveDeConcepto = "
& cla_mov
2208.         xTot_Per =
DSum("MontoDescuento", "DescPrograma",
sel_mov)
2209.         ytot_per = DSum("Abonado",
"DescPrograma", sel_mov)
2210.
2211.         If IsNull(xTot_Per) Then
2212.             xTot_Per = 0
2213.         End If
2214.         If IsNull(ytot_per) Then
2215.             ytot_per = 0
2216.         End If
2217.         xTot_Per = xTot_Per - ytot_per
2218.         bus_val = CStr(xTot_Per)
2219.     Case "PRIMAVACACIONAL"
2220.         montos = LaPrimaVac(numero,
quincena, cla_emp)
2221.         If IsNull(montos) Then
2222.             montos = 0
2223.         End If
2224.         bus_val = CStr(montos)
2225.
2226.     Case "XDIAGUI"
2227.
2228.         xLaFecha = DLookup("FechaAlta",
"Contratación", "ClaveDeEmpleado = " &
cla_emp)
2229.         If IsNull(xLaFecha) Then
2230.             xDias = 0
2231.         Else
2232.             xper = DLookup("Periodo",
"Nómina", "ClaveDeNomina = " & numero)
2233.             If IsNull(xper) Then

```

```

2234.         xper = 3
2235.         End If
2236.
2237.
2238.
2239.         cadena = "Periodo = " & xper & "
and NumeroDeQuincena = " & quincena
2240.         XFecha =
DLookup("FechaInicio", "FechasDeCorte",
cadena)
2241.         año = "#31/12/" & Year(XFecha)
& "#"
2242.         If Year(xLaFecha) <
Year(XFecha) Then
2243.             xlni = "#01/01/" &
Year(XFecha) & "#"
2244.             Else
2245.                 xlni = xLaFecha
2246.             End If
2247.
2248.             xFin = "#31/12/" & Year(XFecha)
& "#"
2249.             xDias = DateDiff("d", xlni, xFin) +
1
2250.         End If
2251.         'Stop
2252.         xMes = DateDiff("yyyy",
xLaFecha, xFin)
2253.         xFecVir = DateAdd("yyyy", xMes,
xLaFecha)
2254.         xDiasVir = DateDiff("d", xFecVir,
xFin) + 1
2255.         xMes = CLng(((xMes * 12) +
(xDiasVir / 30)) * 100) / 100
2256.         xAgui = DLookup("Aguinaldo",
"Factores", xMes & "between niv_min and
niv_max ")
2257.         If IsNull(xAgui) Then
2258.             xAgui = 0
2259.         End If
2260.
2261.         xDias = xAgui * xDias
2262.
2263.         bus_val = CStr(xDias)
2264.
2265.
2266.
2267.         Case "PRIMADEANTIGUEDAD"
2268.
2269.             xTot_Per = 0
2270.             sel_mov = "ClaveDeEmpleado = " &
CStr(cla_emp)
2271.             x = DLookup("Motivo", "Finiquitos",
sel_mov)
2272.             ' If x <> 2 Then
2273.
2274.             sel_mov = "ClaveDeEmpleado = "
& CStr(cla_emp)
2275.             Fecha_Ing = DLookup("FechaAlta",
"Contratación", sel_mov)
2276.             Fecha_fin = DLookup("Fecha",
"FactorFin", sel_mov)
2277.             xTot_Per = DateDiff("m",
Fecha_Ing, Fecha_fin)
2278.             xFechaPro = DateAdd("m",
xTot_Per, Fecha_Ing)
2279.
2280.             xDias = DateDiff("d", xFechaPro,
Fecha_fin) + 1
2281.
2282.             If xDias < 0 Then
2283.                 xTot_Per = (xTot_Per) + (xDias /
30)
2284.             Else
2285.                 xTot_Per = (xTot_Per) + (xDias /
30)
2286.             End If
2287.
2288.             xTot_Per = xTot_Per / 12
2289.             'xtot_per = Format(xtot_per, "yy")
2290.             'End If
2291.             If Forms!MaeFiniquitos![Tipo] =
"Planta" Then
2292.                 If x = 1 Then
2293.                     If xTot_Per < 15 Then
2294.                         montos = 0
2295.                     Else
2296.                         montos = xTot_Per
2297.                     End If
2298.                 Else
2299.                     montos = xTot_Per
2300.                 End If
2301.             Else
2302.                 montos = 0
2303.             End If
2304.             bus_val = CStr(montos)
2305.
2306.             Case "INDEMNIZAEVENTUAL"
2307.                 xTot_Per = 0
2308.                 xtot_ded = 0
2309.
2310.                 sel_mov = "ClaveDeEmpleado = " &
CStr(cla_emp)
2311.                 x = DLookup("Motivo", "Finiquitos",
sel_mov)
2312.                 If x = 3 Then
2313.                     If Forms!MaeFiniquitos![Tipo] =
"Eventual" Then
2314.                         sel_mov = "ClaveDeEmpleado = "
& CStr(cla_emp)
2315.                         Fecha_Ing = DLookup("FechaAlta",
"Contratación", sel_mov)

```

```

2316. Fecha_fin = DLookup("Fecha",
"FactorFin", sel_mov)
2317. xTot_Per = DateDiff("yyyy",
Fecha_Ing, Fecha_fin)
2318. xFechaPro = DateAdd("yyyy",
xTot_Per, Fecha_Ing)
2319. xDias = DateDiff("d", xFechaPro,
Fecha_fin)
2320.
2321. If xDias < 0 Then
2322. xTot_Per = (xTot_Per) - 1 '+
(xDias / 365)
2323. Else
2324. xTot_Per = (xTot_Per) - 0 '+
(xDias / 365)
2325. End If
2326.
2327. 'xtot_per = Format(xtot_per, "yy")
2328. If xTot_Per < 1 Then
2329. sel_mov = "ClaveDeEmpleado =
" & CStr(cla_emp) & " AND
ClaveDeConcepto = 1 AND
NumeroDeNomina = 3"
2330. xTot_Per = DSum("MontoMovi",
"MoviNomina", sel_mov)
2331. If Not IsNull(xTot_Per) Then
2332. xTot_Per = xTot_Per / 2 + 90 *
el_sueldo
2333. Else
2334. xTot_Per = 90 * el_sueldo
2335. End If
2336. Else
2337. xTot_Per = (270 + (20 *
(xTot_Per - 1))) * el_sueldo
2338. End If
2339. End If
2340. Else
2341. xTot_Per = 0
2342. End If
2343. bus_val = CStr(xTot_Per)
2344.
2345. Case "VACRES"
2346. xfecha1 = DLookup("DiasSaldo",
"MaeVacaciones", "ClaveDeEmpleado = " &
cla_emp)
2347. If IsNull(xfecha1) Then
2348. xfecha1 = 0
2349. End If
2350. clave1 = CStr(xfecha1)
2351. bus_val = clave1
2352.
2353.
2354. Case "ANTCUM"
2355.
2356. xfecha1 = DLookup("Fecha",
"FactorFin", "ClaveDeEmpleado = " &
cla_emp)
2357.
2358. xfecha2 = DLookup("FechaAlta",
"FactorFin", "ClaveDeEmpleado = " &
cla_emp)
2359.
2360. xmeses = DateDiff("m", xfecha2,
xfecha1)
2361. yMeses = xmeses \ 12
2362.
2363. xmeses = (xmeses Mod 12)
2364. If xmeses >= 6 Then
2365. yMeses = yMeses + 1
2366. End If
2367. clave1 = CStr(yMeses)
2368. bus_val = clave1
2369.
2370. Case "INDEMNIZAPLANTA"
2371.
2372. xTot_Per = 0
2373. sel_mov = "ClaveDeEmpleado = " &
CStr(cla_emp)
2374. x = DLookup("Motivo", "Finiquitos",
sel_mov)
2375.
2376. cadena = "ClaveDeEmpleado = " &
cla_emp
2377. cadena = cadena & " and
ClaveDeConcepto = 40 "
2378. cadena = cadena & " and
NumeroDeNomina = 4"
2379.
2380. xxVirtual =
DLookup("MontoDeConcepto",
"ConceptosAdicional", cadena)
2381. If IsNull(xxVirtual) Then
2382. xxVirtual = 0
2383. End If
2384.
2385.
2386. If x = 3 Then
2387.
2388.
2389. If Forms!MaeFiniquitos![Tipo] =
"Planta" Then
2390. sel_mov = "ClaveDeEmpleado = "
& CStr(cla_emp)
2391. Fecha_Ing = DLookup("FechaAlta",
"Contratación", sel_mov)
2392. Fecha_fin = DLookup("Fecha",
"FactorFin", sel_mov)
2393. xTot_Per = DateDiff("yyyy",
Fecha_Ing, Fecha_fin)

```



```

2394.      xFechaPro = DateAdd("yyyy",
xTot_Per, Fecha_Ing)
2395.      xDias = DateDiff("d", xFechaPro,
Fecha_fin)
2396.
2397.      If xDias < 0 Then
2398.          xTot_Per = (xTot_Per) - 1 '+
(xDias / 365)
2399.      Else
2400.          xTot_Per = (xTot_Per) - 0 '+
(xDias / 365)
2401.      End If
2402.
2403.      'xtot_per = Format(xtot_per, "yy")
2404.      If xTot_Per >= 1 Then
2405.          xTot_Per = 20 * (el_sueldo +
xxVirtual) * xTot_Per + (90 * (el_sueldo +
xxVirtual))
2406.      Else
2407.          xTot_Per = 90 * (el_sueldo +
xxVirtual)
2408.      End If
2409.      End If
2410.      Else
2411.          xTot_Per = 0
2412.      End If
2413.      bus_val = CStr(xTot_Per)
2414.
2415.      Case "DIASAGUINALDO"
2416.
2417.          montos = DLookup("aguinaldo",
"FactorFin", "ClaveDeEmpleado = " &
cla_emp)
2418.      If IsNull(montos) Then
2419.          montos = 0
2420.      End If
2421.          clave1 = CStr(montos)
2422.          bus_val = clave1
2423.
2424.
2425.      Case "DIASVAC"
2426.          montos = DLookup("Dias_vac",
"FactorFin", "ClaveDeEmpleado = " &
cla_emp)
2427.      If IsNull(montos) Then
2428.          montos = 0
2429.      End If
2430.          clave1 = CStr(montos)
2431.          bus_val = clave1
2432.
2433.      Case "PRIMAVAC"
2434.          montos = DLookup("pri_vac",
"FactorFin", "ClaveDeEmpleado = " &
cla_emp)
2435.      If IsNull(montos) Then
2436.          montos = 0
2437.      End If
2438.          clave1 = CStr(montos)
2439.          bus_val = clave1
2440.
2441.      Case "PORCEVAC"
2442.          montos =
DLookup("porC_primaVac", "FactorFin",
"ClaveDeEmpleado = " & cla_emp)
2443.      If IsNull(montos) Then
2444.          montos = 0
2445.      End If
2446.          clave1 = CStr(montos)
2447.          bus_val = clave1
2448.
2449.      Case "DIASANTI"
2450.
2451.          montos = DLookup("DiasAño",
"FactorFin", "ClaveDeEmpleado = " &
cla_emp)
2452.      If IsNull(montos) Then
2453.          montos = 0
2454.      End If
2455.
2456.          montos1 = DLookup("TodosDias",
"FactorFin", "ClaveDeEmpleado = " &
cla_emp)
2457.      If IsNull(montos1) Then
2458.          montos1 = 0
2459.      End If
2460.
2461.      If montos > montos1 Then
2462.          montos = montos1
2463.      End If
2464.          bus_val = CStr(montos)
2465.      Case "ISPTF"
2466.          'Stop
2467.          aux_qui = quincena
2468.          Set isrSQL =
DBActiva.QueryDefs("BusTotPerT")
2469.          isrSQL.Parameters("num_qui") =
quincena
2470.          isrSQL.Parameters("num_emp") =
cla_emp
2471.          isrSQL.Parameters("num_nom") =
numero
2472.
2473.          Set res_Sel =
isrSQL.OpenRecordset()
2474.          If res_Sel.EOF Or
IsNull(res_Sel![monto]) Then
2475.              valore = 0
2476.          Else
2477.              valore = res_Sel![monto]
2478.          End If
2479.

```

```

2480.     Set isrSQL =
        DBActiva.QueryDefs("BusTotPerTFin")
2481.     isrSQL.Parameters("num_qui") =
        quincena
2482.     isrSQL.Parameters("num_emp") =
        cla_emp
2483.     isrSQL.Parameters("num_nom") =
        numero
2484.
2485.     Set res_Sel =
        isrSQL.OpenRecordset()
2486.     If res_Sel.EOF Or
        IsNull(res_Sel[monto]) Then
2487.         ValoreF = 0
2488.     Else
2489.         ValoreF = res_Sel[monto]
2490.     End If
2491.
2492.     valore = valore - ValoreF
2493.
2494.     Valor = cal_ispt(valore, quincena,
        cla_emp, "N", 3, 3, 1, 0)
2495.     dob_val = CDbl(Valor)
2496.     cad_val = CStr(dob_val)
2497.     bus_val = cad_val
2498.     '-----Termina
        Finiquitos
2499.     Case "FACFONDOAH"
2500.         bus_val = CStr(hzgxElFacFondo)
2501.
2502.     Case "VALESIDEAL"
2503.         sel_mov = "ClaveDeEmpleado = " &
        CStr(cla_emp)
2504.         xTot_Per = DLookup("ValesIdeal",
        "ValesIdeal", sel_mov)
2505.
2506.     If IsNull(xTot_Per) Then
2507.         xTot_Per = 0
2508.     End If
2509.     bus_val = CStr(xTot_Per)
2510.
2511.     Case "VALESANTES"
2512.         sel_mov = "ClaveDeEmpleado = " &
        CStr(cla_emp)
2513.         xTot_Per = DLookup("ValesReal",
        "ValesReal", sel_mov)
2514.
2515.     If IsNull(xTot_Per) Then
2516.         xTot_Per = 0
2517.     End If
2518.     bus_val = CStr(xTot_Per)
2519.     Case Else
2520.         bus_val = "0"
2521.     End Select
2522. End Select
2523. End Function

```

```

2524.
2525.

```

2526. Public Function DiasDev(xNomina As Integer, xPini As Integer, xPfin As Integer)

```

2527.
2528. Dim DBActiva As DATABASE
2529. Dim TempSQL As QueryDef
2530.
2531. Set DBActiva =
        DBEngine.Workspaces(0).Databases(0)
2532.
2533.
2534.     cad = "Delete From DiasDevengados
        "
2535.     Set TempSQL =
        DBActiva.CreateQueryDef("")
2536.     TempSQL.SQL = cad
2537.     TempSQL.Execute
2538.     TempSQL.Close
2539.
2540.     cad = "Delete From
        DiasNoDevengados "
2541.     Set TempSQL =
        DBActiva.CreateQueryDef("")
2542.     TempSQL.SQL = cad
2543.     TempSQL.Execute
2544.     TempSQL.Close
2545.
2546.     cad = "Delete From MontoDevengado
        "
2547.     Set TempSQL =
        DBActiva.CreateQueryDef("")
2548.     TempSQL.SQL = cad
2549.     TempSQL.Execute
2550.     TempSQL.Close
2551.
2552.     Set TempSQL =
        DBActiva.QueryDefs("xMeteDiasDevengado
        s")
2553.     TempSQL.Parameters("xNomina") =
        xNomina
2554.     TempSQL.Parameters("xlni") = xPini
2555.     TempSQL.Parameters("xFin") = xPfin
2556.
2557.     TempSQL.Execute
2558.     TempSQL.Close
2559.
2560.     Set TempSQL =
        DBActiva.QueryDefs("xMeteMontoDevenga
        do")

```

```

2847. cadena = " NumeroDeQuincena = " &
xQuincena
2848. cadena = cadena & " and Periodo = " &
xEIPer
2849.
2850. ' Busca el Mes correspondiente
2851. xEIMes = DLookup("EIMes",
"FechasDeCorte", cadena)
2852.
2853.
2854. cadena = " Periodo = " & xEIPer
2855. cadena = cadena & " and EIMes = " &
xEIMes
2856. cadena = cadena & " and
NumeroDeQuincena \ 100 = " & xQuincena \
100
2857.
2858. xQuince = DMax("NumeroDeQuincena",
"FechasDeCorte", cadena)
2859. If Not IsNull(xQuince) Then
2860. kkQui_fin = xQuince
2861. End If
2862. xQuince = DMin("NumeroDeQuincena",
"FechasDeCorte", cadena)
2863. If Not IsNull(xQuince) Then
2864. kkQui_ini = xQuince
2865. End If
2866.
2867. cadena = " Periodo = " & xEIPer
2868. cadena = cadena & " and EIMes = " &
xEIMes
2869. cadena = cadena & " and
NumeroDeQuincena = " & kkQui_ini
2870. xQuince = DLookup("FechaInicio",
"FechasDeCorte", cadena)
2871.
2872. cadena = " Periodo = " & xEIPer
2873. cadena = cadena & " and EIMes = " &
xEIMes
2874. cadena = cadena & " and
NumeroDeQuincena = " & kkQui_fin
2875.
2876. yQuince = DLookup("FechaFinal",
"FechasDeCorte", cadena)
2877.
2878. xDias = DateDiff("d", xQuince, yQuince)
+ 1
2879. If Not IsNull(xDias) Then
2880. KKDiasDePer = xDias
2881. End If
2882.
2883. cadena = " Periodo = " & xEIPer
2884. cadena = cadena & " and EIMes = " &
xEIMes
2885. cadena = cadena & " and
NumeroDeQuincena = " & xQuincena

```

```

2886. yQuince = DLookup("FechaFinal",
"FechasDeCorte", cadena)
2887. If xNomina = 2 Then
2888. If CInt(Day(yQuince)) > 16 Then
2889. zdias = 30 - CInt(Day(yQuince))
2890. yQuince = DateAdd("d", zdias,
yQuince)
2891. End If
2892. End If
2893.
2894.
2895.
2896. xDias = DateDiff("d", xQuince, yQuince)
+ 1
2897. If Not IsNull(xDias) Then
2898. kkDiasAlPer = xDias
2899. End If
2900. ' Stop
2901. DelimitaLosPeriodos = 0
2902. End Function
2903.

```

2904.Sub yxEDad(FechaF As Variant, Fechal As Variant)

```

2905. Dim Dias As Integer
2906. Dim Meses As Integer
2907. Dim Años As Integer
2908. Dim xfecha1 As Variant
2909. Dim xfecha2 As Variant
2910. Dim cad As String
2911. ' Stop
2912. Aniv(1) = 0
2913. Aniv(2) = 0
2914. Aniv(3) = 0
2915. Aniv(4) = 0
2916. Dias = Day(FechaF) - Day(Fechal)
2917. Meses = Month(FechaF) -
Month(Fechal)
2918. Años = Year(FechaF) - Year(Fechal)
2919. cad = Format("01/01/" &
CStr(Year(FechaF)), "dd/mm/yyyy")
2920. xfecha1 = CVDDate(cad)
2921. Aniv(4) = DateDiff("d", xfecha1,
FechaF)
2922.
2923. If Meses < 0 Then
2924. Meses = 12 + Meses
2925. Años = Años - 1
2926. If Dias < 0 Then
2927. cad = "01/" & CStr(Month(Fechal))
& "/" & CStr(Year(Fechal)) & ""
2928. Fechal = CVDDate(cad)
2929. If Month(Fechal) <> 12 Then

```

```

2742. cadena = CStr(quincena)
2743.
2744. año = Mid$(cadena, 1, 2)
2745. año = "19" + año
2746.
2747. día = Mid$(cadena, 3, 2)
2748.
2749.
2750. tempo = CVar(día)
2751. Select Case Periodo
2752. Case 1
2753.   If tempo Mod 2 Then
2754.     If tempo Mod 4 Then
2755.       día = "30"
2756.     Else
2757.       día = "15"
2758.     End If
2759.   Else
2760.     If tempo Mod 3 Then
2761.       día = "21"
2762.     Else
2763.       día = "7"
2764.     End If
2765.   End If
2766.
2767.
2768.   Select Case Mid$(cadena, 3, 2)
2769.
2770.     Case "01" To "04"
2771.       mes = "enero"
2772.     Case "05" To "08"
2773.       mes = "febrero"
2774.     Case "09" To "12"
2775.       mes = "marzo"
2776.     Case "13" To "16"
2777.       mes = "abril"
2778.     Case "17" To "20"
2779.       mes = "mayo"
2780.     Case "21" To "24"
2781.       mes = "junio"
2782.     Case "25" To "28"
2783.       mes = "julio"
2784.     Case "29" To "32"
2785.       mes = "agosto"
2786.     Case "33" To "36"
2787.       mes = "septiembre"
2788.     Case "37" To "40"
2789.       mes = "octubre"
2790.     Case "41" To "44"
2791.       mes = "noviembre"
2792.     Case "45" To "48"
2793.       mes = "diciembre"
2794.     End Select
2795.
2796.   Case 2
2797.     If tempo Mod 2 Then
2798.       día = "15"
2799.     Else
2800.       día = "30"
2801.     End If
2802.
2803.   Select Case Mid$(cadena, 3, 2)
2804.
2805.     Case "01" To "02"
2806.       mes = "enero"
2807.     Case "03" To "04"
2808.       mes = "febrero"
2809.     Case "05" To "06"
2810.       mes = "marzo"
2811.     Case "07" To "08"
2812.       mes = "abril"
2813.     Case "09" To "10"
2814.       mes = "mayo"
2815.     Case "11" To "12"
2816.       mes = "junio"
2817.     Case "13" To "14"
2818.       mes = "julio"
2819.     Case "15" To "16"
2820.       mes = "agosto"
2821.     Case "17" To "18"
2822.       mes = "septiembre"
2823.     Case "19" To "20"
2824.       mes = "octubre"
2825.     Case "21" To "22"
2826.       mes = "noviembre"
2827.     Case "23" To "24"
2828.       mes = "diciembre"
2829.     End Select
2830.
2831.
2832.   End Select
2833. End Select
2834.
2835.
2836. cadena = día & " " & mes & " " & año
2837. MsgBox "Fecha " & cadena
2838.
2839. Con_fec = cadena
2840.
2841. End Function
2842.

```

2843.Function

DelimitaLosPeriodos(xQuincena As Integer, xNomina As Integer)

```

2844. ' Busca el Periodo correspondiente
2845.
2846. xEIPer = DLookup("Periodo", "Nómina",
"ClaveDeNomina = " & xNomina)

```

```

2742. cadena = CStr(quincena)
2743.
2744. año = Mid$(cadena, 1, 2)
2745. año = "19" + año
2746.
2747. dia = Mid$(cadena, 3, 2)
2748.
2749.
2750. tempo = CVar(dia)
2751. Select Case Periodo
2752. Case 1
2753. If tempo Mod 2 Then
2754. If tempo Mod 4 Then
2755. dia = "30"
2756. Else
2757. dia = "15"
2758. End If
2759. Else
2760. If tempo Mod 3 Then
2761. dia = "21"
2762. Else
2763. dia = "7"
2764. End If
2765. End If
2766.
2767.
2768. Select Case Mid$(cadena, 3, 2)
2769.
2770. Case "01" To "04"
2771. mes = "enero"
2772. Case "05" To "08"
2773. mes = "febrero"
2774. Case "09" To "12"
2775. mes = "marzo"
2776. Case "13" To "16"
2777. mes = "abril"
2778. Case "17" To "20"
2779. mes = "mayo"
2780. Case "21" To "24"
2781. mes = "junio"
2782. Case "25" To "28"
2783. mes = "julio"
2784. Case "29" To "32"
2785. mes = "agosto"
2786. Case "33" To "36"
2787. mes = "septiembre"
2788. Case "37" To "40"
2789. mes = "octubre"
2790. Case "41" To "44"
2791. mes = "noviembre"
2792. Case "45" To "48"
2793. mes = "diciembre"
2794. End Select
2795.
2796. Case 2
2797. If tempo Mod 2 Then
2798. dia = "15"
2799. Else
2800. dia = "30"
2801. End If
2802.
2803. Select Case Mid$(cadena, 3, 2)
2804.
2805. Case "01" To "02"
2806. mes = "enero"
2807. Case "03" To "04"
2808. mes = "febrero"
2809. Case "05" To "06"
2810. mes = "marzo"
2811. Case "07" To "08"
2812. mes = "abril"
2813. Case "09" To "10"
2814. mes = "mayo"
2815. Case "11" To "12"
2816. mes = "junio"
2817. Case "13" To "14"
2818. mes = "julio"
2819. Case "15" To "16"
2820. mes = "agosto"
2821. Case "17" To "18"
2822. mes = "septiembre"
2823. Case "19" To "20"
2824. mes = "octubre"
2825. Case "21" To "22"
2826. mes = "noviembre"
2827. Case "23" To "24"
2828. mes = "diciembre"
2829. End Select
2830.
2831.
2832.
2833. End Select
2834.
2835.
2836. cadena = dia & " " & mes & " " & año
2837. 'MsgBox "Fecha " & cadena
2838.
2839. Con_fec = cadena
2840.
2841. End Function
2842.

```

2843.Function DelimitaLosPeriodos(xQuincena As Integer, xNomina As Integer)

```

2844. ' Busca el Periodo correspondiente
2845.
2846. xElPer = DLookup("Periodo", "Nómina",
"ClaveDeNomina = " & xNomina)

```

```

2847. cadena = " NumeroDeQuincena = " &
      xQuincena
2848. cadena = cadena & " and Periodo = " &
      xEIPer
2849.
2850. ' Busca el Mes correspondiente
2851. xEIMes = DLookup("EIMes",
      "FechasDeCorte", cadena)
2852.
2853.
2854. cadena = " Periodo = " & xEIPer
2855. cadena = cadena & " and EIMes = " &
      xEIMes
2856. cadena = cadena & " and
      NumeroDeQuincena \ 100 = " & xQuincena \
      100
2857.
2858. xQuince = DMax("NumeroDeQuincena",
      "FechasDeCorte", cadena)
2859. If Not IsNull(xQuince) Then
2860.   kkQui_fin = xQuince
2861. End If
2862. xQuince = DMin("NumeroDeQuincena",
      "FechasDeCorte", cadena)
2863. If Not IsNull(xQuince) Then
2864.   kkQui_ini = xQuince
2865. End If
2866.
2867. cadena = " Periodo = " & xEIPer
2868. cadena = cadena & " and EIMes = " &
      xEIMes
2869. cadena = cadena & " and
      NumeroDeQuincena = " & kkQui_ini
2870. xQuince = DLookup("FechaInicio",
      "FechasDeCorte", cadena)
2871.
2872. cadena = " Periodo = " & xEIPer
2873. cadena = cadena & " and EIMes = " &
      xEIMes
2874. cadena = cadena & " and
      NumeroDeQuincena = " & kkQui_fin
2875.
2876. yQuince = DLookup("FechaFinal",
      "FechasDeCorte", cadena)
2877.
2878. xDias = DateDiff("d", xQuince, yQuince)
      + 1
2879. If Not IsNull(xDias) Then
2880.   KKDiasDePer = xDias
2881. End If
2882.
2883. cadena = " Periodo = " & xEIPer
2884. cadena = cadena & " and EIMes = " &
      xEIMes
2885. cadena = cadena & " and
      NumeroDeQuincena = " & xQuincena

```

```

2886. yQuince = DLookup("FechaFinal",
      "FechasDeCorte", cadena)
2887. If xNNomina = 2 Then
2888.   If CInt(Day(yQuince)) > 16 Then
2889.     zdias = 30 - CInt(Day(yQuince))
2890.     yQuince = DateAdd("d", zdias,
      yQuince)
2891.   End If
2892. End If
2893.
2894.
2895.
2896. xDias = DateDiff("d", xQuince, yQuince)
      + 1
2897. If Not IsNull(xDias) Then
2898.   kkDiasAiPer = xDias
2899. End If
2900.   Stop
2901. DelimitaLosPeriodos = 0
2902. End Function
2903.

```

2904.Sub yxEDad(FechaF As Variant, FechaI As Variant)

```

2905. Dim Dias As Integer
2906. Dim Meses As Integer
2907. Dim Años As Integer
2908. Dim xfecha1 As Variant
2909. Dim xfecha2 As Variant
2910. Dim cad As String
2911. Stop
2912. Aniv(1) = 0
2913. Aniv(2) = 0
2914. Aniv(3) = 0
2915. Aniv(4) = 0
2916. Dias = Day(FechaF) - Day(FechaI)
2917. Meses = Month(FechaF) -
      Month(FechaI)
2918. Años = Year(FechaF) - Year(FechaI)
2919. cad = Format("01/01/" &
      CStr(Year(FechaF)), "dd/mm/yyyy")
2920. xfecha1 = CVDate(cad)
2921. Aniv(4) = DateDiff("d", xfecha1,
      FechaF)
2922.
2923. If Meses < 0 Then
2924.   Meses = 12 + Meses
2925.   Años = Años - 1
2926.   If Dias < 0 Then
2927.     cad = "01/" & CStr(Month(FechaI))
      & "/" & CStr(Year(FechaI)) & ""
2928.     FechaI = CVDate(cad)
2929.     If Month(FechaI) <> 12 Then

```

```

2930.      cad = "01/" &
      CStr(Month(Fecha) + 1) & "/" &
      CStr(Year(Fecha)) & ""
2931.      Else
2932.      cad = "31/" &
      CStr(Month(Fecha)) & "/" &
      CStr(Year(Fecha)) & ""
2933.      End If
2934.      FechaF = CVDate(cad)
2935.      Dias = DateDiff("d", Fecha,
      FechaF) + Dias
2936.      Meses = Meses - 1
2937.      End If
2938.      Else
2939.      If Meses = 0 Then
2940.      If Dias < 0 Then
2941.      Fecha = DateAdd("m", 1,
      FechaF)
2942.      Dias = DateDiff("d", FechaF,
      Fecha)
2943.      End If
2944.      Else
2945.      If Dias < 0 Then
2946.      cad = "01/" &
      CStr(Month(Fecha)) & "/" &
      CStr(Year(Fecha)) & ""
2947.      Fecha = CVDate(cad)
2948.      If Month(Fecha) <> 12 Then
2949.      cad = "01/" &
      CStr(Month(Fecha) + 1) & "/" &
      CStr(Year(Fecha)) & ""
2950.      Else
2951.      cad = "31/" &
      CStr(Month(Fecha)) & "/" &
      CStr(Year(Fecha)) & ""
2952.      End If
2953.      FechaF = CVDate(cad)
2954.      Dias = DateDiff("d", Fecha,
      FechaF) + Dias
2955.      Meses = Meses - 1
2956.      End If
2957.      End If
2958.      End If
2959.      Aniv(1) = Dias
2960.      Aniv(2) = Meses
2961.      Aniv(3) = Años
2962.      End Sub
2963.

```

2964.Function EsAñoBis(quincena As Integer, numero As Integer) As Double

2965. Dim añoActual As Integer

```

2966. Dim tipoaño As Double
2967. Dim la_fecha As Variant
2968. Dim la_fec As String
2969. If quincena = 9504 Then
2970. quincena = 9501
2971. End If
2972.
2973. la_fec = Con_fec(quincena, numero)
2974. la_fecha = CVDate(la_fec)
2975.
2976. añoActual = Year(la_fecha) ' Obtiene
año actual.
2977. If añoActual Mod 4 = 0 And añoActual
Mod 100 = 0 Then
2978. If añoActual Mod 400 = 0 Then '
¿Puede dividirse exactamente por 400?
2979. tipoaño = 6 ' añoBisiesto = "es un
año bisiesto."
2980. Else ' No se puede
dividir exactamente.
2981. tipoaño = 5 ' añoBisiesto = "es un
año centesimal pero no es bisiesto."
2982. End If
2983. Elseif añoActual Mod 4 = 0 Then
2984. tipoaño = 6 ' añoBisiesto = "es un año
bisiesto."
2985. Else
2986. tipoaño = 5 ' añoBisiesto = "no es un
año bisiesto."
2987. End If
2988.
2989. EsAñoBis = tipoaño
2990. End Function
2991.

```

2992.Function fec_fac(quincena As Integer, Fecha As Variant, numero As Integer) As Double

```

2993.
2994. Dim la_fecha As Variant
2995. Dim cad_fec As String
2996. Dim quince As Integer
2997. Dim semanas As Double
2998. quince = quincena
2999.
3000. cad_fec = Con_fec(quince, numero)
3001.
3002. la_fecha = CVDate(cad_fec)
3003.
3004. If quincena = 9504 Then
3005. el_día = 30 - Day(Fecha)
3006. End If
3007.

```

```

3008.   semanas = DateDiff("y", Fecha,
      la_fecha)
3009.   semanas = semanas / 30
3010.   fec_fac = semanas
3011.
3012. End Function
3013.

```

3014.Function fecha_aniv(Fecha As Variant, quincena As Integer, años As Integer, Tipo As Integer) As Double

```

3015.
3016. Dim el_año As Integer
3017. Dim el_mes As Integer
3018. Dim el_día As Integer
3019. Dim la_fecha As Variant
3020. Dim cad_fec As String
3021. 'Dim fecha As Variant
3022. Dim Valor As Double
3023. Dim Dias As Double
3024. Dim quince As Integer
3025.   quince = quincena
3026.   If quincena = 9504 Then
3027.     quince = quince - 1
3028.   End If
3029.   cad_fec = Con_fec(quince, Tipo)
3030.   la_fecha = CDate(cad_fec)
3031.   el_año = Year(la_fecha) -
      Year(Fecha)
3032.   el_mes = Month(la_fecha) -
      Month(Fecha)
3033.
3034.   If quincena = 9504 Then
3035.     el_día = 30 - Day(Fecha)
3036.   End If
3037.
3038.
3039.
3040.
3041.
3042.   If el_año >= años Then
3043.     If el_mes > 0 And el_año > años
then
3044.       Valor = 1
3045.     Else
3046.       If el_mes = 0 Then
3047.         If el_día = 0 Then
3048.           Valor = 1
3049.         Else
3050.           Valor = el_día / 15
3051.         If Valor < 0 Then
3052.           Valor = -1
3053.         End If

```

```

3054.     End If
3055.   Else
3056.     Valor = -1
3057.   End If
3058. End If
3059. If el_año - años = 0 And Valor = -1
Then
3060.   Valor = 0
3061. Else
3062.   If Tipo = 1 Then
3063.     Valor = el_año - años + Valor
3064.   Else
3065.     If Valor = -1 Or el_año > años
Then
3066.       Valor = 1
3067.     End If
3068.   End If
3069. End If
3070. Else
3071.   Valor = 0
3072. End If
3073.
3074.   fecha_aniv = Valor
3075. End Function
3076.
3077. -----
3078. ' Modulo : Calcula
3079. ' Objetivo : Procesor los empleados para
      el pago de nomina
3080. ' Fecha : 12/octubre/1995
3081. ' Parametros : la_quincena -> Número
      de Quincena a Procesas
3082. '           Tipo -> 1 Semanal
3083. '           2 Quincenal
3084. '           3 Mensual
3085. -----
3086. '

```

3087.Sub gen_nom(la_quince As Integer, numero As Integer, la_empr As Integer)

```

3088.
3089. Dim DBActiva As DATABASE
3090. Dim TempSQL As QueryDef
3091. Dim tempSQ1 As QueryDef
3092. Dim emp_act As Recordset
      empleados activos
3093. Dim pde_act As Recordset
      percepciones y deducciones
3094. Dim mov_act As Recordset
      movimientos
3095. Dim pde_con As String
      consulta de
      percepciones y deducciones

```



```

3096. Dim miArea As Workspace
3097. Dim num_transa As Integer
3098.
3099. Dim msg As String      ' variable de
mensaje
3100. Dim la_quin1 As Integer ' quincena
numerica
3101. Dim el_tipo As String  ' Tipo de
nomina
3102. Dim CuentaRegistro As Integer
3103. Dim Cuenta As Variant
3104. Dim cue_emp As Integer
3105.
3106. Dim el_emp As Long     ' numero del
empleado
3107. Dim EL_MOV As Integer ' numero de
movimiento
3108.
3109. Dim la_for01 As String ' formula
3110. Dim final As Double  ' resultado final
3111. Dim Empresa As Integer
3112.
3113. Dim xTotalPercibe As Double
3114. Dim xTotalDeduca As Double
3115. Dim xSucursal As Integer
3116.
3117. 'Stop
3118. Set DBActiva =
DBEngine.Workspaces(0).Databases(0)
3119. Set miArea = DBEngine.Workspaces(0)
3120. num_transa = 0
3121. la_quin1 = la_quince
3122. el_tipo = "N"
3123. 'x = MeteIncapacidades(numero,
la_quince)
3124.
3125. x = BorraSubAcreditable(CInt(numero),
CInt(la_quince))
3126. x = MeteLosAguinaldos(numero,
la_quince, 338)
3127.
3128.
3129. ' Esta Funcion borra los dias de ajuste
de incapacidad 7001
3130. ' y refresca los datos con el valor del
concepto 12
3131. x = AjusteDelIncapacidades(numero,
la_quince)
3132. ' Borra todos los movimientos de la
nomina a generar
3133. '-----
3134. ' Incluir a la empresa
3135. '-----
3136.
3137. ' Borra los datos de MoviNominaT
3138.
3139. Set TempSQL =
DBActiva.QueryDefs("TemporalBorra")
3140. TempSQL.Parameters("num_nom") =
numero
3141. TempSQL.Parameters("num_qui") =
la_quince
3142. TempSQL.Execute
3143. TempSQL.Close
3144.
3145. 'Inserta los datos a MoviTemporalT
3146.
3147. Set TempSQL =
DBActiva.QueryDefs("MoviTemporal")
3148. TempSQL.Parameters("num_qui") =
la_quince
3149. TempSQL.Parameters("num_nom") =
numero
3150. TempSQL.Execute
3151. TempSQL.Close
3152. ' Se Insertan las Variables de las horas
Extras
3153.
3154. 'Set tempSQL =
DBActiva.QueryDefs("BorraHorasExtras")
3155. 'tempSQL.parameters("num_qui") =
la_quince
3156. 'tempSQL.parameters("num_nom") =
numero
3157. 'tempSQL.Execute
3158. 'tempSQL.Close
3159.
3160.
3161.
3162.
3163. ' Actualiza los descuentos programados
3164. Set TempSQL =
DBActiva.QueryDefs("qInserDesProg")
3165. TempSQL.Parameters("num_qui") =
la_quince
3166. TempSQL.Parameters("num_nom")
= numero
3167. TempSQL.Parameters("num_empr")
= la_empr
3168. TempSQL.Execute
3169. TempSQL.Close
3170.
3171. 'Inserta los Prestamos
3172.
3173. Set TempSQL =
DBActiva.QueryDefs("InsertaInteresPrestam
oNUEVO")
3174. TempSQL.Parameters("num_qui") =
la_quince
3175. TempSQL.Parameters("cla_nom") =
numero

```

```

3176. TempSQL.Parameters("xEmpresa") =
    la_empr
3177. TempSQL.Execute
3178. TempSQL.Close
3179.
3180. Set TempSQL =
    DBActiva.QueryDefs("InsertaPagoPrestamo
    NUEVO")
3181. TempSQL.Parameters("num_qui") =
    la_quince
3182. TempSQL.Parameters("cla_nom") =
    numero
3183. TempSQL.Parameters("xEmpresa") =
    la_empr
3184. TempSQL.Execute
3185. TempSQL.Close
3186.
3187. 'If numero = 1 Then
3188.     Set TempSQL =
        DBActiva.QueryDefs("InsertaCeroCajaFond
        o")
3189.     TempSQL.Parameters("xPeriodo") =
        la_quince
3190.     TempSQL.Parameters("xNomina") =
        numero
3191.     TempSQL.Execute
3192.     TempSQL.Close
3193. 'End If
3194.
3195. x = DelimitaLosPeriodos(la_quin1,
    numero)
3196. If numero = 1 Or numero = 53 Then
3197.     x = GeneraSeptimoDia(la_quince,
        numero)
3198. End If
3199.
3200. If numero = 8 Then
3201.     If la_quince = 9903 Then
3202.         kkQui_ini = 9852
3203.         x = DiasDev(1, kkQui_ini, la_quin1)
3204.     End If
3205. Else
3206.     If numero = 98 Then
3207.         x = DiasDev(3, kkQui_ini, la_quin1)
3208.     Else
3209.         If numero = 88 Then
3210.             x = DiasDev(2, kkQui_ini,
                la_quin1)
3211.         Else
3212.             If numero = 87 Then
3213.                 x = DiasDev(4, kkQui_ini,
                    la_quin1)
3214.             Else
3215.                 x = DiasDev(numero,
                    la_quin1, la_quin1)
3216.             End If
3217.         End If
3218.     End If
3219. End If
3220. ' Stop
3221. ' Llama a la consulta para buscar a los
    empleados
3222.     Set TempSQL =
        DBActiva.QueryDefs("LaPlantillaActiva")
3223.     TempSQL.Parameters("lanomina") =
        numero
3224.     TempSQL.Parameters("periodo")
        = la_quince
3225.     TempSQL.Parameters("num_qui") =
        la_quince
3226.
3227.
3228. '-----
3229. ' Ahora se seleccionan los conceptos
    afectados
3230. '-----
3231.
3232.
3233. Set tempSQ1 =
    DBActiva.QueryDefs("ConsultaPde1")
3234. tempSQ1.Parameters("num_qui") =
    la_quin1
3235. tempSQ1.Parameters("num_nom") =
    numero
3236. tempSQ1.Parameters("CLA_empr") =
    la_empr
3237. 'tempSQ1.Execute
3238.
3239. '-----
3240. ' Establece el periodo a generar
3241.
3242.
3243.
3244. '-----
3245.
3246.
3247. 'MsgBox "Comienza a Generar "
3248. 'Stop
3249. Set emp_act =
    TempSQL.OpenRecordset()
3250. Set pde_act =
    tempSQ1.OpenRecordset()
3251.
3252. If emp_act.EOF Then
3253.     MsgBox "Error " & " No hay empleados
        activos "
3254. Else

```

```

3255.
3256.   emp_act.MoveLast
3257.   emp_act.MoveFirst
3258.   CuentaRegistro =
emp_act.RecordCount
3259.   cue_emp = 0
3260.   Cuenta =
SysCmd(SYSCMD_INITMETER,
"Generando Nomina ", CuentaRegistro)
3261.   DoCmd.Hourglass True
3262.   Set mov_act =
DBActiva.OpenRecordset("MoviNominaT")
3263.   Set TempSQL =
DBActiva.QueryDefs("CamMoviT")
3264.   Do Until emp_act.EOF
3265.
3266.     If pde_act.EOF Then
3267.       MsgBox "Error " & "No hay
movimientos "
3268.     Else
3269.       xTotalPercibe = 0
3270.       xTotalDeduce = 0
3271.       pde_act.MoveFirst
3272.       el_emp =
emp_act![ClaveDeEmpleado]
3273.       el_sueldo = emp_act![Sueldo]
3274.       xSucursal =
emp_act![ClaveDeSucursal]
3275.       If IsNull(emp_act!FacFondo) Then
3276.         hzgxxEIFacFondo = 0
3277.       Else
3278.         hzgxxEIFacFondo =
emp_act!FacFondo
3279.       End If
3280.
3281.     End If
3282.     Do Until pde_act.EOF
3283.
3284.       If
IsNull(pde_act![FormulaGenMovi]) Then
3285.         la_for01 = "0*1"
3286.       Else
3287.         la_for01 =
pde_act![FormulaGenMovi]
3288.       End If
3289.
3290.     EL_MOV =
pde_act![ClaveDeConcepto]
3292.     Debug.Print EL_MOV
3293.     Debug.Print el_emp
3294.     Debug.Print "-----"
3295.
3296.
3297.
3298.

```

```

3299.     la_for01 = parser(la_for01,
el_emp, la_quin1, EL_MOV, el_sueldo,
numero, xSucursal)
3300.     final = cal_mon(la_for01)
3301.     If pde_act![RequiereTope] = -1
Then
3302.       la_for01 =
pde_act![FormulaTope]
3303.       la_for01 = parser(la_for01,
el_emp, la_quin1, EL_MOV, el_sueldo,
numero, xSucursal)
3304.       final_tope = cal_mon(la_for01)
3305.
3306.     If final > final_tope Then
3307.       final = final_tope
3308.     End If
3309.   End If
3310.
3311.   If final <= 0 And
pde_act![AceptaNegativos] = 0 Then
3312.     final = 0
3313.   End If
3314.
3315.   ' Verificar si esta inscrito en el
fondo de Ahorro
3316.
3317.   If pde_act![FormaDeCalculo] =
"H" Then
3318.     x_ahorro =
DLookup("ClaveDeEmpleado",
"InsCajaFondo", "ClaveDeEmpleado = " &
el_emp & " and Estado = 'A' and
ClaveDeConcepto = " & EL_MOV)
3319.     -If IsNull(x_ahorro) Then
3320.       final = 0
3321.     End If
3322.   End If
3323.
3324.   ' Marcado Ajuste Moneda
3325.   If pde_act![Redondeo] = -1 Then
3326.     x_ajuste = 1
3327.     If EL_MOV > 0 Then
3328.       x_vales = (final * 100) Mod 100
3329.       If x_vales < 50 Then
3330.         x_ajuste = 0
3331.       Else
3332.         x_ajuste = 1
3333.       End If
3334.       final = Int(final) + x_ajuste
3335.     End If
3336.   End If
3337.
3338.   ' Marcado a Redondeo a truncamiento
de 2 cifras
3339.   If pde_act![Truncamiento] = -1 Then

```

```

3340.         final = Format(final,
"00000000.00")
3341.
3342.     End If
3343.
3344.
3345.         consulta = "NumeroDeQuincena
= " & la_quin1
3346.         consulta = consulta & " and
ClaveDeEmpleado = " &
emp_act![ClaveDeEmpleado]
3347.         consulta = consulta & " and
ClaveDeConcepto = " &
pde_act![ClaveDeConcepto]
3348.         consulta = consulta & " and
NumeroDeNómina = " & numero
3349.         cue_emp =
DLookup("ClaveDeEmpleado",
"MoviNominaT", consulta)
3350.
3351.
3352.     If IsNull(cue_emp) Then
3353.         If final <> 0 Then
3354.             mov_act.AddNew
3355.             mov_act![ClaveDeEmpleado]
= el_emp
3356.             mov_act![ClaveDeConcepto] =
EL_MOV
3357.             mov_act![NumeroDeQuincena] = la_quin1
3358.             mov_act![ClaveDeEmpresa] =
la_empr
3359.             mov_act![TipoDeNomina] =
el_tipo
3360.             mov_act![NumeroDeNómina]
= numero
3361.             mov_act![VariableParaPercepYDed] = 0#
3362.             mov_act![MontoMovi] = final
3363.             mov_act.Update
3364.         End If
3365.     Else
3366.
3367.         TempSQL.Parameters("num_qui") =
la_quin1
3368.         TempSQL.Parameters("cve_emp") =
emp_act![ClaveDeEmpleado]
3369.         TempSQL.Parameters("cve_con") =
pde_act![ClaveDeConcepto]
3370.         TempSQL.Parameters("num_nom") =
numero
3371.         TempSQL.Parameters("monto")
= final
3372.         TempSQL.Execute
3373.     End If
3374.
3375.     pde_act.MoveNext
3376.     Loop
3377.
3378.
3379.
3380.     pde_act.MoveFirst
3381.
3382.     emp_act.MoveNext
3383.     cue_emp = cue_emp + 1
3384.     Cuenta =
SysCmd(SYSCMD_UPDATEMETER,
cue_emp)
3385.     Loop
3386.
3387.     emp_act.Close
3388.     pde_act.Close
3389.     Cuenta =
SysCmd(SYSCMD_REMOVEMETER)
3390.     DoCmd.Hourglass False
3391. End If
3392.
3393. ' Borra los datos de MoviNomina
3394.
3395. Set TempSQL =
DBActiva.QueryDefs("BorraMoviNomina")
3396. TempSQL.Parameters("num_nom") =
numero
3397. TempSQL.Parameters("num_qui") =
la_quin1
3398. TempSQL.Execute
3399. TempSQL.Close
3400.
3401. ' Inserta los datos a MoviNomina de
MoviNominaT
3402.
3403. Set TempSQL =
DBActiva.QueryDefs("TemporalInserta")
3404. TempSQL.Parameters("num_qui") =
la_quince
3405. TempSQL.Parameters("num_nom") =
numero
3406. TempSQL.Execute
3407. TempSQL.Close
3408.
3409.
3410. Set TempSQL =
DBActiva.QueryDefs("GeneraLaNomina")
3411. TempSQL.Parameters("NumPeri") =
la_quince
3412. TempSQL.Parameters("TipoNomi") =
numero

```

```

3413. TempSQL.Execute
3414. TempSQL.Close
3415.
3416. If numero = 1 Or numero = 3 Or numero
    = 4 Then
3417.     Set TempSQL =
        DBActiva.QueryDefs("xElimina241")
3418.     TempSQL.Parameters("xQuincena") =
        la_quince
3419.     TempSQL.Parameters("xNomina") =
        numero
3420.     TempSQL.Execute
3421.     TempSQL.Close
3422. End If
3423.
3424. Set TempSQL =
        DBActiva.QueryDefs("xNetoAPagarNom")
3425. TempSQL.Parameters("formularios!Dep
        Bancario!NumeroDeQuincena") = la_quince
3426. TempSQL.Parameters("formularios!Dep
        Bancario!ClaveDeNomina") = numero
3427. TempSQL.Execute
3428. TempSQL.Close
3429.
3430. x = InsertaVacPro(numero, la_quince)
3431.
3432. Set TempSQL =
        DBActiva.QueryDefs("xNetoAPagarAjus")
3433. TempSQL.Parameters("formularios!Dep
        Bancario!NumeroDeQuincena") = la_quince
3434. TempSQL.Parameters("formularios!Dep
        Bancario!ClaveDeNomina") = numero
3435. TempSQL.Execute
3436. TempSQL.Close
3437.
3438. Set TempSQL =
        DBActiva.QueryDefs("xNetoAPagarAjusNeg
        ")
3439. TempSQL.Parameters("formularios!Dep
        Bancario!NumeroDeQuincena") = la_quince
3440. TempSQL.Parameters("formularios!Dep
        Bancario!ClaveDeNomina") = numero
3441. TempSQL.Execute
3442. TempSQL.Close
3443.
3444. End Sub
3445.
3446.
3447.

```

3448.Function

GeneraSeptimoDia(num_qui As Integer, Num_nom As Integer) As Integer

```

3449. Dim DBActiva As DATABASE
3450.
3451. Dim mov_act As Recordset
3452. Dim mov_actT As Recordset
3453.
3454. Dim EvaluaSQL As QueryDef
3455. Dim EvaluaReg As Recordset
3456. Dim cla_emp As Long
3457.
3458. Set DBActiva =
        DBEngine.Workspaces(0).Databases(0)
3459.
3460. If Num_nom = 53 Then
3461.     Else
3462.         Set EvaluaSQL =
            DBActiva.QueryDefs("BorraSepDia")
3463.         EvaluaSQL.Parameters("num_nom")
            = Num_nom
3464.         EvaluaSQL.Parameters("num_qui") =
            num_qui
3465.         EvaluaSQL.Execute
3466.         EvaluaSQL.Close
3467.
3468.         Set EvaluaSQL =
            DBActiva.QueryDefs("BorraSepDiaM")
3469.         EvaluaSQL.Parameters("num_nom")
            = Num_nom
3470.         EvaluaSQL.Parameters("num_qui") =
            num_qui
3471.         EvaluaSQL.Execute
3472.         EvaluaSQL.Close
3473.     End If
3474.         Set mov_act =
            DBActiva.OpenRecordset("MoviNominaT")
3475.         Set mov_actT =
            DBActiva.OpenRecordset("MoviNomina")
3476.
3477.         Set EvaluaSQL =
            DBActiva.QueryDefs("GeneraSepDia")
3478.         EvaluaSQL.Parameters("num_nom")
            = Num_nom
3479.         EvaluaSQL.Parameters("num_qui") =
            num_qui
3480.         Set EvaluaReg =
            EvaluaSQL.OpenRecordset()
3481.
3482. If EvaluaReg.EOF Then
3483.     MsgBox " No hay Promocionados "
3484. Else
3485.     DoCmd.Hourglass False

```

```

3486.     elDia = DLookup("dias", "Nómina",
"ClaveDeNomina = " & Num_nom)
3487.     If IsNull(elDia) Then
3488.         elDia = 5
3489.     End If
3490.     Do Until EvaluaReg.EOF
3491.         'Stop
3492.         cla_emp =
EvaluaReg![ClaveDeEmpleado]
3493.         montos =
EvaluaReg![NoTrabajados]
3494.         montos1 =
EvaluaReg![CorreccionDias]
3495.         xDias = EvaluaReg![Dias]
3496.
3497.         If IsNull(xDias) Then
3498.             xDias = elDia
3499.         Else
3500.             xDias = elDia - xDias
3501.         End If
3502.
3503.         If IsNull(montos) Then
3504.             montos = 0
3505.         End If
3506.         If IsNull(montos1) Then
3507.             montos1 = 0
3508.         End If
3509.
3510.         'FactorxDia = (CInt(((14 / elDia) *
1000)) / 1000) - 1
3511.         FactorxDia = (7 / elDia) - 1
3512.
3513.         yyElSepDia = 7 - elDia
3514.         montos = yyElSepDia - (montos *
FactorxDia) + (montos1 * FactorxDia) -
(xDias * FactorxDia)
3515.         'montos = CDbl(Format(montos,
"00.00"))
3516.
3517.         'Montos = Montos * 1000
3518.         '     If Montos Mod 10 = 0 Then
3519.         '         Montos = Montos / 1000
3520.         '     Else
3521.         '         Montos = Montos / 1000
3522.         '         Montos = Int(Montos * (-
100)) / (-100)
3523.         '     End If
3524.
3525.         If Num_nom = 53 Then
3526.             cadena = "ClaveDeEmpleado = " &
cla_emp
3527.             cadena = cadena & " and
NumeroDeNómina = " & Num_nom
3528.             cadena = cadena & " and
NumeroDeQuincena = " & num_qui
3529.             cadena = cadena & " and
ClaveDeConcepto = " & 22
3530.             xxx22 =
DLookup("VariableParaPercepYDed",
"MoviNomina", cadena)
3531.             If Not IsNull(xxx22) Then
3532.                 montos = 0
3533.             End If
3534.             End If
3535.
3536.             If montos > 0.01 Then
3537.                 mov_act.AddNew
3538.                 mov_act![ClaveDeEmpleado] =
cla_emp
3539.                 mov_act![ClaveDeConcepto] =
22
3540.                 mov_act![NumeroDeQuincena] =
num_qui
3541.                 mov_act![ClaveDeEmpresa] = 8
3542.                 mov_act![TipoDeNomina] = "N"
3543.                 mov_act![NumeroDeNómina] =
Num_nom
3544.                 mov_act![VariableParaPercepYDed] =
montos
3545.                 mov_act![MontoMovi] = 0
3546.                 mov_act.Update
3547.
3548.                 mov_actT.AddNew
3549.                 mov_actT![ClaveDeEmpleado] =
cla_emp
3550.                 mov_actT![ClaveDeConcepto] =
22
3551.                 mov_actT![NumeroDeQuincena]
= num_qui
3552.                 mov_actT![ClaveDeEmpresa] = 8
3553.                 mov_actT![TipoDeNomina] = "N"
3554.                 mov_actT![NumeroDeNómina] =
Num_nom
3555.                 mov_actT![VariableParaPercepYDed] =
montos
3556.                 mov_actT![MontoMovi] = 0
3557.                 mov_actT.Update
3558.
3559.
3560.             End If
3561.
3562.             EvaluaReg.MoveNext
3563.         Loop
3564.
3565.         EvaluaReg.Close
3566.
3567.         DoCmd.Hourglass False
3568.
3569.     End If

```

```

3570. GeneraSeptimoDia = 0
3571.
3572.
3573. End Function

```

3574.Function

GeneraSeptimoDiaW(num_qu i As Integer, Num_nom As Integer) As Integer

```

3575. Dim DBActiva As DATABASE
3576.
3577. Dim mov_act As Recordset
3578. Dim mov_actT As Recordset
3579.
3580. Dim EvaluaSQL As QueryDef
3581. Dim EvaluaReg As Recordset
3582. Dim cla_emp As Long
3583. Dim xFactor As Double
3584. Set DBActiva =
    DBEngine.Workspaces(0).Databases(0)
3585. 'Stop
3586. If Num_nom = 53 Then
3587. Else
3588. Set EvaluaSQL =
    DBActiva.QueryDefs("BorraSepDia")
3589. EvaluaSQL.Parameters("num_nom")
    = Num_nom
3590. EvaluaSQL.Parameters("num_qui") =
    num_qui
3591. EvaluaSQL.Execute
3592. EvaluaSQL.Close
3593.
3594. Set EvaluaSQL =
    DBActiva.QueryDefs("BorraSepDiaM")
3595. EvaluaSQL.Parameters("num_nom")
    = Num_nom
3596. EvaluaSQL.Parameters("num_qui") =
    num_qui
3597. EvaluaSQL.Execute
3598. EvaluaSQL.Close
3599. End If
3600. Set mov_act =
    DBActiva.OpenRecordset("MoviNominaT")
3601. Set mov_actT =
    DBActiva.OpenRecordset("MoviNomina")
3602.
3603. Set EvaluaSQL =
    DBActiva.QueryDefs("GeneraSepDia")
3604. EvaluaSQL.Parameters("num_nom")
    = Num_nom
3605. EvaluaSQL.Parameters("num_qui") =
    num_qui
3606. Set EvaluaReg =
    EvaluaSQL.OpenRecordset()

```

```

3607.
3608. If EvaluaReg.EOF Then
3609. 'MsgBox " No hay Promocionados "
3610. Else
3611. DoCmd.Hourglass False
3612. eIDia = DLookup("dias", "Nómina",
    "ClaveDeNomina = " & Num_nom)
3613. If IsNull(eIDia) Then
3614. eIDia = 5
3615. End If
3616. Do Until EvaluaReg.EOF
3617. 'Stop
3618. cla_emp =
    EvaluaReg![ClaveDeEmpleado]
3619. montos =
    EvaluaReg![NoTrabajados]
3620. montos1 =
    EvaluaReg![CorreccionDias]
3621. xDias = EvaluaReg![Dias]
3622. xFactor = EvaluaReg![xFactor]
3623. If xFactor = 1.4 Then
3624. eIDia = 5
3625. Else
3626. eIDia = 6
3627. End If
3628.
3629. If IsNull(xDias) Then
3630. xDias = eIDia
3631. Else
3632. xDias = eIDia - xDias
3633. End If
3634.
3635. If IsNull(montos) Then
3636. montos = 0
3637. End If
3638. If IsNull(montos1) Then
3639. montos1 = 0
3640. End If
3641.
3642. FactorxDia = xFactor - 1
3643. yyEISepDia = 7 - eIDia
3644. montos = yyEISepDia - (montos *
    FactorxDia) + (montos1 * FactorxDia) -
    (xDias * FactorxDia)
3645.
3646. If Num_nom = 53 Then
3647. cadena = "ClaveDeEmpleado = " &
    cla_emp
3648. cadena = cadena & " and
    NumeroDeNómina = " & Num_nom
3649. cadena = cadena & " and
    NumeroDeQuincena = " & num_qui
3650. cadena = cadena & " and
    ClaveDeConcepto = " & 22

```

```

3651.     xxx22 =
        DLookup("VariableParaPercepYDed",
        "MoviNomina", cadena)
3652.     If Not IsNull(xxx22) Then
3653.         montos = 0
3654.     End If
3655.     End If
3656.
3657.     If montos > 0.01 Then
3658.         mov_act.AddNew
3659.         mov_act![ClaveDeEmpleado] =
            cla_emp
3660.         mov_act![ClaveDeConcepto] =
            22
3661.         mov_act![NumeroDeQuincena] =
            num_qui
3662.         mov_act![ClaveDeEmpresa] = 8
3663.         mov_act![TipoDeNomina] = "N"
3664.         mov_act![NumeroDeNómina] =
            Num_nom
3665.         mov_act![VariableParaPercepYDed] =
            montos
3666.         mov_act![MontoMovi] = 0
3667.         mov_act.Update
3668.
3669.         mov_actT.AddNew
3670.         mov_actT![ClaveDeEmpleado] =
            cla_emp
3671.         mov_actT![ClaveDeConcepto] =
            22
3672.         mov_actT![NumeroDeQuincena]
            = num_qui
3673.         mov_actT![ClaveDeEmpresa] = 8
3674.         mov_actT![TipoDeNomina] = "N"
3675.         mov_actT![NumeroDeNómina] =
            Num_nom
3676.         mov_actT![VariableParaPercepYDed] =
            montos
3677.         mov_actT![MontoMovi] = 0
3678.         mov_actT.Update
3679.
3680.
3681.     End if
3682.
3683.     EvaluaReg.MoveNext
3684. Loop
3685.
3686.     EvaluaReg.Close
3687.
3688.     DoCmd.Hourglass False
3689.
3690. End If
3691. GeneraSeptimoDiaw = 0
3692.
3693.
3694. End Function
3695.
3696.

```


67 Entity Name: Procesos

Column Names: ClaveDeProceso Integer NOT NULL
Primary Key: ClaveDeProceso

Column Names: NombreDeProceso Text(18)

68 Entity Name: ProgTiempoExtra

Column Names: ClaveDeEmpleado Long Integer
Primary Key: ClaveDeEmpleado
Foreign Keys: ClaveDeEmpleado, ClaveDeNomina, NumeroDeQuincena
References: SubSidioAcreditable
(ClaveDeEmpleado, ClaveDeNomina, NumeroDeQuincena)

Column Names: ClaveDeNomina Integer NOT NULL
Primary Key: ClaveDeNomina

Column Names: NumeroDeQuincena Long Integer
Primary Key: NumeroDeQuincena

Column Names: FechaInicio Date/Time
Primary Key: FechaInicio

Column Names: FechaFinal Date/Time
Primary Key: FechaFinal

Column Names: HoraEntrada Date/Time

Primary Key: FechaDomingo

Column Names: FechaDescansa Date/Time
Primary Key: FechaDescansa

Column Names: ClaveDeJornada Integer

Column Names: HorasPrima Double

70 Entity Name: PuntosAporta

Column Names: NumeroDeQuincena Double NOT NULL
Primary Key: NumeroDeQuincena
Foreign Keys: ClaveDeConcepto, ClaveDeNomina
References: PercepcionesDeducciones
(ClaveDeConcepto, ClaveDeNomina)

Column Names: ClaveDeConcepto Integer NOT NULL
Primary Key: ClaveDeConcepto

Column Names: ClaveDeNomina Integer NOT NULL
Primary Key: ClaveDeNomina

Column Names: Puntos Double

71 Entity Name: RegDePermisos

Column Names: ClaveDeEmpleado Long Integer
Primary Key: ClaveDeEmpleado

Primary Key: HoraEntrada

Column Names: HoraSale Date/Time

Column Names: Tipo Text(1)

Column Names: ClaveDeTabulador Integer

Column Names: CamTemCat Yes/No

Column Names: TiempoJornadaExt Double

Column Names: TiempoJornadaOrd Double

69 Entity Name: ProgTurnosDefa

Column Names: ClaveDeEmpleado Long Integer
Primary Key: ClaveDeEmpleado
Foreign Keys: ClaveDeEmpleado, ClaveDeNomina, NumeroDeQuincena
References: SubSidioAcreditable
(ClaveDeEmpleado, ClaveDeNomina, NumeroDeQuincena)

Column Names: ClaveDeNomina Integer
Primary Key: ClaveDeNomina

Column Names: NumeroDeQuincena Long Integer
Primary Key: NumeroDeQuincena

Column Names: FechaDomingo Date/Time

Foreign Keys: ClaveDeEmpleado, ClaveDeNomina, NumeroDeQuincena
References: SubSidioAcreditable
(ClaveDeEmpleado, ClaveDeNomina, NumeroDeQuincena)

Column Names: ClaveDeNomina Integer
Primary Key: ClaveDeNomina

Column Names: NumeroDeQuincena Long Integer
Primary Key: NumeroDeQuincena

Column Names: FechaIni Date/Time NOT NULL
Primary Key: FechaIni

Column Names: TipoDeIncidencia Long Integer
Primary Key: TipoDeIncidencia

Column Names: FechaDeRegistro Date/Time

Column Names: ClaveDeJornada Integer

Column Names: HoraIni Date/Time

Column Names: FechaFin Date/Time

Column Names: HoraFin Date/Time

Column Names: TotalDeTiempo Double

Column Names: ClaveDeAsunto Integer

Column Names: ConGoceDeSueldo Yes/No

Column Names: Faltainjustificada Yes/No

Column Names: Autoriza Text(15)

72 Entity Name: RegistroHorasExtras

Column Names: ClaveDeEmpleado Long Integer
Primary Key: ClaveDeEmpleado
Foreign Keys: ClaveDeEmpleado, ClaveDeNomina,
NumeroDeQuincena

References: SubSidioAcreditable
(ClaveDeEmpleado, ClaveDeNomina,
NumeroDeQuincena)

Column Names: ClaveDeNomina Integer NOT NULL
Primary Key: ClaveDeNomina

Column Names: FechaDeIncidencia Date/Time
Primary Key: FechaDeIncidencia

Column Names: NumeroDeQuincena Long Integer
Primary Key: NumeroDeQuincena

Column Names: ClaveDeTurno Integer
Primary Key: ClaveDeTurno

Column Names: CostoPorHora Double
Primary Key: CostoPorHora

Column Names: TipoCA Text(1)

74 Entity Name: SubSidioAcreditable

Column Names: ClaveDeEmpleado Long Integer
Primary Key: ClaveDeEmpleado
Foreign Keys: ClaveDeNomina, ClaveDeEmpleado
References: EmpleadoNomina (ClaveDeNomina,
ClaveDeEmpleado)

Column Names: ClaveDeNomina Integer
Primary Key: ClaveDeNomina

Column Names: NumeroDeQuincena Long Integer
Primary Key: NumeroDeQuincena

Column Names: Acreditable Double

Column Names: NoAcreditable Double

75 Entity Name: TipoDePagoFondoAhorro

Column Names: TipoDePago Text(50) NOT NULL
Primary Key: TipoDePago

76 Entity Name: TiposDeConceptos

Column Names: PerDed Text(15)
Primary Key: PerDed

Column Names: NumHoras Double

Column Names: Desglosada Yes/No

Column Names: Turno Integer

Column Names: CambioTemporal Yes/No

Column Names: EITipo Text(1)

73 Entity Name: SubCuentaPerDed

Column Names: ClaveDeConcepto Integer NOT NULL
Primary Key: ClaveDeConcepto
Foreign Keys: ClaveDeConcepto, ClaveDeNomina
References: PercepcionesDeducciones
(ClaveDeConcepto, ClaveDeNomina)

Column Names: ClaveDeNomina Integer NOT NULL
Primary Key: ClaveDeNomina

Column Names: TipoContrato Text(1) NOT NULL
Primary Key: TipoContrato

Column Names: TipoProductivo Text(1) NOT NULL
Primary Key: TipoProductivo

Column Names: Forma Text(1)

Column Names: SubCuenta Text(8)

77 Entity Name: Contratacion

Column Names: ClaveDeEmpleado Long Integer
Primary Key: ClaveDeEmpleado

Column Names: ClaveDeSolicitud Long Integer

Column Names: Tipo Text(10)

Column Names: TipoEventual Text(15)

Column Names: Foto OLE Object

Column Names: FechaAlta Date/Time

Column Names: FechaIniReno Date/Time

Column Names: FechaReno Date/Time

78 Entity Name: CostoXCurso

Column Names: ClaveDeCurso Integer
Primary Key: ClaveDeCurso
Foreign Keys: ClaveDeCurso
References: Cursos (ClaveDeCurso)

Column Names: FechaInicio Date/Time
Primary Key: FechaInicio

Column Names: FechaFinal Date/Time

Column Names: CostoXPersona Double

79 Entity Name: CursoMateria

Column Names: ClaveDeCurso Integer
Primary Key: ClaveDeCurso
Foreign Keys: ClaveDeCurso
References: Cursos (ClaveDeCurso)

Column Names: ClaveDeMateria Integer
Primary Key: ClaveDeMateria

Column Names: Nivel Text(10)

Column Names: DescripcionMateria Text(125)

80 Entity Name: CursoPuesto

Column Names: ClaveDeCurso Integer
Primary Key: ClaveDeCurso
Foreign Keys: ClaveDeCurso
References: Cursos (ClaveDeCurso)

Column Names: ClaveDePuesto Integer
Primary Key: ClaveDePuesto
Foreign Keys: ClaveDePuesto
References: Puesto (ClaveDePuesto)

Column Names: ClaveDeFinalidad Byte

Column Names: Rangoini Integer

Column Names: NombreDeEscuela Text(50)

Column Names: RFCEscuela Text(13)

Column Names: RSTPS Text(50)

Column Names: Direccion Text(60)

83 Entity Name: EvaFactores

Column Names: ClaveDeFactor Byte
Primary Key: ClaveDeFactor
Foreign Keys: TipoFactor
References: TiposFactores (TipoFactor)

Column Names: TipoFactor Byte

Column Names: NomFactor Text(25)

Column Names: DesFactor Memo

84 Entity Name: EvaObjetivos

Column Names: ClaveDeEmpleado Long Integer
Primary Key: ClaveDeEmpleado
Foreign Keys: ClaveDeEmpleado, FeciniPer, FecFinPer
References: ResEvaluacion (ClaveDeEmpleado, FeciniPer, FecFinPer)

Column Names: RangoFin Integer

81 Entity Name: Cursos

Column Names: ClaveDeCurso Integer
Primary Key: ClaveDeCurso

Column Names: ClaveDeArea Text(5)

Column Names: NombreDeCurso Text(50)

Column Names: Objetivo Memo

Column Names: Metodologia Memo

Column Names: Profundidad Text(50)

Column Names: Duracion Double

Column Names: HrsXSesion Double

Column Names: Nivel Text(15)

Column Names: Reforzamiento Integer

Column Names: stps Text(2)

82 Entity Name: Escuelas

Column Names: ClaveDeEscuela Integer
Primary Key: ClaveDeEscuela

Column Names: FecIniPer Date/Time
Primary Key: FecIniPer

Column Names: FecFinPer Date/Time
Primary Key: FecFinPer

Column Names: NumeroDeObjetivo Byte
Primary Key: NumeroDeObjetivo

Column Names: DesObjetivo Text(255)

Column Names: CalObjetivo Double

Column Names: Ponderacion Double

Column Names: Total Double

85 Entity Name: EvaObserva

Column Names: ClaveDeEmpleado Long Integer
Primary Key: ClaveDeEmpleado
Foreign Keys: ClaveDeEmpleado, FeciniPer, FecFinPer
References: ResEvaluacion (ClaveDeEmpleado, FeciniPer, FecFinPer)

Column Names: FecIniPer Date/Time
Primary Key: FeciniPer

Column Names: FecFinPer Date/Time
Primary Key: FecFinPer

Column Names: DesObjetivo Memo

86 Entity Name: Finalidades

Column Names: ClaveDePuesto Integer

Primary Key: ClaveDePuesto

Foreign Keys: ClaveDePuesto

References: Puesto (ClaveDePuesto)

Column Names: ClaveDeFinalidad Byte

Primary Key: ClaveDeFinalidad

Column Names: NombreFinalidad Memo

Column Names: Priodicidad Text(18)

Column Names: ClaveDeArea Text(5)

Column Names: EvaluadaCon Text(20)

87 Entity Name: Grupo

Column Names: ClaveDeInst Long Integer

Primary Key: ClaveDeInst

Foreign Keys: ClaveDeInst

References: Instructores (ClaveDeInst)

Column Names: ClaveDeCurso Integer

Primary Key: ClaveDeCurso

Foreign Keys: ClaveDeEscuela

Column Names: ClaveDeCurso Integer

Primary Key: ClaveDeCurso

Column Names: FechaFin Date/Time

Column Names: ClaveDePuesto Integer

Column Names: ClaveDeAdscripcion Integer

Column Names: Descripcion Text(100)

Column Names: Observaciones Text(100)

Column Names: Calificacion Double

89 Entity Name: Instructores

Column Names: ClaveDeInst Long Integer

Primary Key: ClaveDeInst

Column Names: NombreDeInst Text(60)

Column Names: FotoInstructor OLE Object

Column Names: Observaciones Memo

Column Names: RSTPS Text(50)

Column Names: RFCInstructor Text(13)

References: Escuelas (ClaveDeEscuela)

Column Names: ClaveDeEscuela Integer

Primary Key: ClaveDeEscuela

Foreign Keys: ClaveDeCurso

References: Cursos (ClaveDeCurso)

Column Names: FechaInicio Date/Time

Primary Key: FechaInicio

Column Names: FechaFinal Date/Time

Column Names: HoraEntra Date/Time

Column Names: HoraSale Date/Time

Column Names: Cupo Text(18)

Column Names: Estado Text(18)

88 Entity Name: HistoriaAcademica

Column Names: ClaveDeEmpleado Long Integer

Primary Key: ClaveDeEmpleado

Foreign Keys: ClaveDeEmpleado

References: Contratacion (ClaveDeEmpleado)

Column Names: FechaIni Date/Time

Primary Key: FechaIni

Foreign Keys: ClaveDeCurso

References: Cursos (ClaveDeCurso)

Column Names: Tipoins Text(1)

Column Names: Direccion Text(50)

Column Names: Telefono Long Integer

Column Names: Fax Double

90 Entity Name: ListaGrupo

Column Names: ClaveDeInst Long Integer

Primary Key: ClaveDeInst

Foreign Keys: ClaveDeEmpleado

References: Contratacion (ClaveDeEmpleado)

Column Names: ClaveDeCurso Integer

Primary Key: ClaveDeCurso

Foreign Keys: ClaveDeInst, ClaveDeCurso,
ClaveDeEscuela, FechaInicio

References: Grupo (ClaveDeInst, ClaveDeCur
ClaveDeEscuela, FechaInicio)

Column Names: ClaveDeEscuela Integer

Primary Key: ClaveDeEscuela

Column Names: ClaveDeEmpleado Long Integer

Primary Key: ClaveDeEmpleado

Column Names: FechaInicio Date/Time

Primary Key: FechaInicio

Column Names: Calificacion Double

Column Names: Asistencias Byte

Column Names: ClaveDePuesto Integer

Column Names: ClaveDeDepartamento Integer

Column Names: Fecha Date/Time

91 Entity Name: MovFactores

Column Names: ClaveDeEmpleado Long Integer

Primary Key: ClaveDeEmpleado

Foreign Keys: ClaveDeFactor

References: EvaFactores (ClaveDeFactor)

Column Names: FecIniPer Date/Time

Primary Key: FecIniPer

Foreign Keys: ClaveDeEmpleado, FecIniPer, FecFinPer

References: ResEvaluacion (ClaveDeEmpleado, FecIniPer, FecFinPer)

Column Names: FecFinPer Date/Time

Primary Key: FecFinPer

Column Names: ClaveDeFactor Byte

Primary Key: ClaveDeFactor

Column Names: CalFactores Double

Column Names: ClaveDeArea Integer

Column Names: ClaveDeNivel Text(8)

Column Names: NombreDePuesto Text(50)

Column Names: PropPuesto Text(250)

Column Names: PropSubor Text(250)

Column Names: Valores Currency

Column Names: Libertad Text(250)

Column Names: Posicion Text(50)

Column Names: EnDireccion Double

Column Names: EnTramites Currency

94 Entity Name: RangosDeCalif

Column Names: RanIni Double

Primary Key: RanIni

Column Names: RanFin Double

Primary Key: RanFin

Column Names: TipoRan Byte

Primary Key: TipoRan

92 Entity Name: PlanDeCapacitacion

Column Names: ClaveDeEmpleado Long Integer

Primary Key: ClaveDeEmpleado

Foreign Keys: ClaveDeCurso

References: Cursos (ClaveDeCurso)

Column Names: Fecha Date/Time

Primary Key: Fecha

Foreign Keys: ClaveDeEmpleado

References: Contratacion (ClaveDeEmpleado)

Column Names: ClaveDeCurso Integer

Primary Key: ClaveDeCurso

Column Names: ClaveDeAdscripcion Long Integer

Column Names: Prioridad Integer

Column Names: Estado Text(15)

Column Names: Objetivo Text(20)

93 Entity Name: Puesto

Column Names: ClaveDePuesto Integer

Primary Key: ClaveDePuesto

Column Names: ClaveDeAdscripcion Long Integer

Column Names: Resultado Double

95 Entity Name: Refuerza

Column Names: ClaveDeCurso Integer

Primary Key: ClaveDeCurso

Foreign Keys: ClaveDeCurso

References: Cursos (ClaveDeCurso)

Column Names: ClaveDeEmpleado Long Integer

Primary Key: ClaveDeEmpleado

Foreign Keys: ClaveDeEmpleado

References: Contratacion (ClaveDeEmpleado)

Column Names: FechaProxRef Text(18) NOT NULL

Primary Key: FechaProxRef

Column Names: Reforzamiento Integer

Column Names: FechaUltRef Text(18)

96 Entity Name: Requisitos

Column Names: NumeroRequisito Text(18) NOT NULL

Primary Key: NumeroRequisito

Foreign Keys: ClaveDeCurso

References: Cursos (ClaveDeCurso)

Column Names: ClaveDeCurso Integer NOT NULL

Primary Key: ClaveDeCurso

Column Names: DesRequisito Text(50)

References: Cursos (ClaveDeCurso)

97 Entity Name: ResEvaluacion

Column Names: ClaveDeEmpleado Long Integer
Primary Key: ClaveDeEmpleado
Foreign Keys: ClaveDeEmpleado
References: Contratacion (ClaveDeEmpleado)

Column Names: FecIniPer Date/Time
Primary Key: FecIniPer

Column Names: FecFinPer Date/Time
Primary Key: FecFinPer

Column Names: CaiObjetivoF Double

Column Names: CalFactorF Double

Column Names: CalFinal Double

98 Entity Name: Seriacion

Column Names: ClaveDeCurso Integer NOT NULL
Primary Key: ClaveDeCurso
Foreign Keys: ClaveDeCurso
References: Cursos (ClaveDeCurso)

Column Names: CursoAnterior Integer
Primary Key: CursoAnterior
Foreign Keys: ClaveDeCurso

Column Names: Db500 Double

Column Names: Db750 Double

Column Names: Db1000 Double

Column Names: Db1500 Double

Column Names: Db2000 Double

Column Names: Db3000 Double

Column Names: Db4000 Double

Column Names: Db6000 Double

Column Names: Db8000 Double

Column Names: Db125A Double

Column Names: Db250A Double

Column Names: Db500A Double

Column Names: Db750A Double

Column Names: Db1000A Double

Column Names: Db1500A Double

Column Names: Db2000A Double

Column Names: Seriacion Integer

99 Entity Name: TiposFactores

Column Names: TipoFactor Byte
Primary Key: TipoFactor

Column Names: DesTipoFactor Text(40)

Column Names: PesoPSM Double

Column Names: PesoPCM Double

100 Entity Name: Audiometrico

Column Names: ClaveDeSolicitud Long Integer
Primary Key: ClaveDeSolicitud
Foreign Keys: ClaveDeSolicitud
References: SolicitudDGrales (ClaveDeSolicitud)

Column Names: Fecha Date/Time
Primary Key: Fecha

Column Names: TipoDeExamen Text(10)

Column Names: Db125 Double

Column Names: Db250 Double

Column Names: Db3000A Double

Column Names: Db4000A Double

Column Names: Db6000A Double

Column Names: Db8000A Double

Column Names: DB125_1 Double

Column Names: Db250_1 Integer

Column Names: Db500_1 Double

Column Names: Db750_1 Double

Column Names: Db1000_1 Double

Column Names: Db1500_1 Double

Column Names: Db2000_1 Double

Column Names: Db3000_1 Double

Column Names: Db4000_1 Double

Column Names: DB125A_1 Double

Column Names: Db250A_1 Double

Column Names: Db500A_1 Double
Column Names: Db750A_1 Double
Column Names: Db1000A_1 Double
Column Names: Db1500A_1 Double
Column Names: Db2000A_1 Double
Column Names: Db3000A_1 Double
Column Names: Db4000A_1 Double
Column Names: OidIzq Double
Column Names: OidDer Double
Column Names: Promedio Double
Column Names: Colaboracion.Text(7)

101 Entity Name: Circulatorio

Column Names: ClaveDeSolicitud Long Integer
Primary Key: ClaveDeSolicitud
Foreign Keys: ClaveDeSolicitud References:
SolicitudDGrales (ClaveDeSolicitud)

Column Names: Fecha Date/Time
Primary Key: Fecha

Column Names: FechaIniReno Date/Time
Column Names: FechaReno Date/Time

103 Entity Name: Digestivo

Column Names: ClaveDeSolicitud Long Integer
Primary Key: ClaveDeSolicitud
Foreign Keys: ClaveDeSolicitud References:
SolicitudDGrales (ClaveDeSolicitud)

Column Names: Fecha Date/Time
Primary Key: Fecha

Column Names: Boca Text(150)
Column Names: ExamenClínico Text(150)
Column Names: Hernias Text(150)
Column Names: ExamenLabora Text(150)
Column Names: Dentadura Text(150)
Column Names: Amigdalas Yes/No
Column Names: Apendice Yes/No
Column Names: Ulceras Yes/No

Column Names: TensionArterial Text(20)
Column Names: Varices Text(50)
Column Names: ExamenLabora Text(100)
Column Names: Cardiopatias Text(50)
Column Names: Antecedentes Text(50)
Column Names: DebeVigilar Yes/No
Column Names: Observaciones Text(50)

102 Entity Name: Contratacion

Column Names: ClaveDeEmpleado Long Integer
Primary Key: ClaveDeEmpleado
Foreign Keys: ClaveDeSolicitud References:
SolicitudDGrales (ClaveDeSolicitud)

Column Names: ClaveDeSolicitud Long Integer
Column Names: Tipo Text(10)
Column Names: TipoEventual Text(15)
Column Names: Foto OLE Object
Column Names: FechaAlta Date/Time

104 Entity Name: ExamenMedico

Column Names: ClaveDeSolicitud Long Integer
Primary Key: ClaveDeSolicitud
Foreign Keys: ClaveDeSolicitud References:
SolicitudDGrales (ClaveDeSolicitud)

Column Names: Fecha Date/Time
Primary Key: Fecha

Column Names: Peso Integer
Column Names: Estatura Single
Column Names: Complexion Text(10)
Column Names: TipoSanguineo Text(15)
Column Names: Pantalon Double
Column Names: Zapatos Double
Column Names: Camisa Double
Column Names: Chamarra Double
Column Names: Bata Double
Column Names: Lentes Yes/No
Column Names: Hereditarios Text(100)

Column Names: Personales Text(100)
Column Names: NoPatologicos Text(100)
Column Names: Patologicos Text(100)
Column Names: DefomaOrgani Text(100)
Column Names: InteFisica Text(100)
Column Names: TrabaAnter Text(100)
Column Names: Tabaco Text(10)
Column Names: Alcohol Text(10)
Column Names: Farmacos Text(15)
Column Names: Otras Text(20)
Column Names: Secuelas Yes/No
Column Names: sangre Text(50)

105 Entity Name: GrafExamen

Column Names: ClaveDeSolicitud Long Integer
Primary Key: ClaveDeSolicitud
Foreign Keys: ClaveDeSolicitud References:
SolicitudDGrales (ClaveDeSolicitud)

Column Names: Piel Text(100)
Column Names: Observaciones Text(100)
Column Names: ProblemasAsociados Text(200)

107 Entity Name: Nervioso

Column Names: ClaveDeSolicitud Long Integer
Primary Key: ClaveDeSolicitud
Foreign Keys: ClaveDeSolicitud References:
SolicitudDGrales (ClaveDeSolicitud)

Column Names: Fecha Date/Time
Primary Key: Fecha
Column Names: Sensibilidad Text(100)
Column Names: Reflejos Text(100)

108 Entity Name: Oidos

Column Names: ClaveDeSolicitud Long Integer
Primary Key: ClaveDeSolicitud
Foreign Keys: ClaveDeSolicitud References:
SolicitudDGrales (ClaveDeSolicitud)

Column Names: Fecha Date/Time
Primary Key: Fecha
Column Names: Oidoizq1 Double

Column Names: Fecha Date/Time
Primary Key: Fecha
Column Names: ClaveDeDoctor Integer
Column Names: ExaRel Text(70)
Column Names: ResExamen Text(15)
Column Names: Observaciones Text(200)
Column Names: ResAudiometria Text(50)
Column Names: apto Text(50)

106 Entity Name: Locomotor

Column Names: ClaveDeSolicitud Long Integer
Primary Key: ClaveDeSolicitud
Foreign Keys: ClaveDeSolicitud References:
SolicitudDGrales (ClaveDeSolicitud)

Column Names: Fecha Date/Time
Primary Key: Fecha
Column Names: Articulaciones Text(100)
Column Names: Marcha Text(100)
Column Names: Movimientos Text(100)

Column Names: OidoDer1 Double
Column Names: OidoIzq2 Text(5)
Column Names: OidoDer2 Text(5)
Column Names: Olfato Text(50)

109 Entity Name: ProgDeExámenes

Column Names: ClaveDeEmpleado Long Integer NOT
NULL
Primary Key: ClaveDeEmpleado
Foreign Keys: ClaveDeEmpleado References:
Contratacion (ClaveDeEmpleado)

Column Names: FechaDeExamen Text(18)
Column Names: TipoDeExamen Text(18)
Column Names: Observaciones Text(18)

110 Entity Name: Psiquismo

Column Names: ClaveDeSolicitud Long Integer
Primary Key: ClaveDeSolicitud
Foreign Keys: ClaveDeSolicitud References:
SolicitudDGrales (ClaveDeSolicitud)

Column Names: Fecha Date/Time

Primary Key: Fecha

Column Names: Atencion Text(100)

Column Names: Memoria Text(50)

Column Names: Ideacion Text(50)

Column Names: Psicotecnica Text(50)

111 Entity Name: Respiratorio

Column Names: ClaveDeSolicitud Long Integer
Primary Key: ClaveDeSolicitud
Foreign Keys: ClaveDeSolicitud References:
SolicitudDGrales (ClaveDeSolicitud)

Column Names: Fecha Date/Time
Primary Key: Fecha

Column Names: ExamenClinico Text(150)

Column Names: ExamenRadio Text(150)

Column Names: ExamenLabora Text(150)

Column Names: Asma Text(150)

Column Names: Bronquitis Text(150)

Column Names: TipoGafas Text(40)

Column Names: TipoLentes Text(40)

Column Names: TipoAnteojos Text(30)

113 Entity Name: SMConsultas

Column Names: ClaveDeEmpleado Long Integer
Primary Key: ClaveDeEmpleado
Foreign Keys: ClaveDeEmpleado References:
Contratacion (ClaveDeEmpleado)

Column Names: Fecha Date/Time
Primary Key: Fecha

Column Names: ClasIndustria Text(100)

Column Names: Operacion Text(70)

Column Names: ClaveDeMaquina Text(70)

Column Names: HoraEntrada Date/Time

Column Names: HoraSalida Date/Time

Column Names: Sintomas Memo

Column Names: Diagnostico Memo

Column Names: Indicaciones Memo

112 Entity Name: Sentidos

Column Names: ClaveDeSolicitud Long Integer
Primary Key: ClaveDeSolicitud
Foreign Keys: ClaveDeSolicitud References:
SolicitudDGrales (ClaveDeSolicitud)

Column Names: Fecha Date/Time
Primary Key: Fecha

Column Names: Observaciones Text(180)

Column Names: AgudezaVisual Text(15)

Column Names: OjoDerecho Text(20)

Column Names: Ojolzquierdo Text(20)

Column Names: Daltonismo Text(100)

Column Names: Hemeralopia Text(100)

Column Names: CampoVisual Text(100)

Column Names: Perimetro Text(100)

Column Names: Pupilas Text(100)

Column Names: ALuz Text(100)

Column Names: Consensual Text(100)

Column Names: Atendio Memo

Column Names: JefeInmediato Memo

114 Entity Name: SolicitudDGrales

Column Names: ClaveDeSolicitud Long Integer
Primary Key: ClaveDeSolicitud

Column Names: ClaveDeEmpleado Long Integer

Column Names: ClaveDeEmpresa Text(18)

Column Names: Nombre Text(50)

Column Names: Paterno Text(50)

Column Names: Materno Text(50)

Column Names: Nombres Text(50)

Column Names: TipoTraba Text(20)

Column Names: ClaveDePuesto Integer

Column Names: MotivSoli Text(40)

Column Names: DeptoSoli Long Integer

Column Names: Sexo Text(10)

Column Names: Fec_nac Date/Time
Column Names: Lug_nac Text(50)
Column Names: Nacionalidad Text(30)
Column Names: Con_migra Text(50)
Column Names: FM Text(5)
Column Names: NumeroFM Double
Column Names: Edo_Civil Text(15)
Column Names: Lug_Matrimonio Text(50)
Column Names: Fec_Matrimonio Date/Time
Column Names: Direccion Text(18)
Column Names: CodPos Text(18)
Column Names: Municipio Text(50)
Column Names: Entidad Text(50)
Column Names: Telefono Text(15)
Column Names: Tel_Urgencia Double

Column Names: Anticonceptivo Text(150)
Column Names: Operaciones Text(150)
Column Names: Ciclo Text(150)
Column Names: CicloTipo Text(50)

Column Names: RFC Text(15)
Column Names: Imss Text(20)
Column Names: Cartilla Text(20)
Column Names: Pasaporte Text(30)
Column Names: Ven_Pasaporte Date/Time
Column Names: Licencia Text(20)
Column Names: Tip_Licencia Text(15)
Column Names: Religion Text(20)
Column Names: CURP Text(50)

115 Entity Name: Urinario

Column Names: ClaveDeSolicitud Long Integer
Primary Key: ClaveDeSolicitud
Foreign Keys: ClaveDeSolicitud References:
SolicitudDGrales (ClaveDeSolicitud)
Column Names: Fecha Date/Time
Primary Key: Fecha
Column Names: ExamenClinico Text(150)
Column Names: ExamenLabora Text(150)