

42



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES
CAMPUS "ARAGON"

296199

SISTEMA PARA LA ADMINISTRACION DE LAS
PARTIDAS ESPECIALES DEL GRUPO 300 DEL
PRESUPUESTO DE LA UNIVERSIDAD
NACIONAL AUTONOMA DE MEXICO

SIPES

(SISTEMA PARA LA ADMINISTRACION DE LAS
PARTIDAS ESPECIALES)

TESIS

QUE PARA OBTENER EL TITULO DE:
INGENIERO EN COMPUTACION

PRESENTA:

LEONARDO MUNGUIA HERNANDEZ

DIRECTOR DE TESIS:
INGENIERO AIDA ALICIA HERNANDEZ QUINTO

MEXICO. D. F.

2001



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



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.

TESIS CON FALLA DE ORIGEN

INTRODUCCIÓN

La Universidad Nacional Autónoma de México, como institución pública utiliza una enorme cantidad de recursos, para el desarrollo y logro de sus objetivos, debido a ello posee múltiples fuentes de financiamiento y objetos de gasto del orden de millones de pesos, el uso de la tecnología informática permite que sea adecuadamente administrado permitiendo así la consecución de sus metas, este sistema conocido como presupuesto por programas contempla sin lugar a dudas todos y cada uno de los objetivos de la universidad, convirtiéndolos en objetos de gasto perfectamente identificables, esto permite a la administración universitaria, destinar los recursos de manera prioritaria y en las áreas que así se requiera, sin descuidar los diferentes ámbitos del quehacer universitario.

El presente trabajo plantea en principio un estudio de la consistencia del presupuesto universitario para derivar en las partidas de gasto, objeto del sistema a desarrollar, es decir, parte de la premisa general del presupuesto universitario hasta aterrizar en el área particular de las partidas especiales, esto a través del desarrollo de un sistema de cómputo, que permita obtener resultados acerca de cómo se comportan dichas partidas, dentro de su grupo y del presupuesto en general permitiendo una planeación de las necesidades de recursos e inclusive de la preparación para cada uno de los ciclos anuales del presupuesto.

Esta herramienta debe permitir al área de planeación presupuestaria, tomar decisiones a partir de los datos obtenidos de procesar dicha información, además, de que permitirá utilizarla para su uso en otras dependencias del gobierno federal tales como: la Secretaría Educación Pública o la Secretaría de Hacienda y Crédito Público mismas que utilizan estos datos para la administración de recursos destinados al gasto universitario.

El programa plantea diferentes esquemas de trabajo, debido a la naturaleza de las partidas que estudia y las solicitudes de información que necesita cubrir, por lo tanto involucra diferentes módulos que serán analizados y desarrollados independientemente, para resolver los problemas particulares que son objeto de estudio en el presente trabajo; se plantean diferentes soluciones puesto que la naturaleza de las partidas así lo requieren, de esta manera el primer capítulo involucra la descripción del presupuesto por programas que se utiliza en la Universidad Nacional Autónoma de México, el capítulo 2 y 3 se encargan de la planeación y análisis del sistema, el capítulo 4 muestra el diseño del sistema que en el capítulo 5 se desarrollara bajo estas premisas, por último una serie de conclusiones nos permitirán dar una visión general del desarrollo realizado y las diferentes situaciones involucradas en la creación del sistema.

El trabajo concluye por una parte, con la bibliografía utilizada para el desarrollo del proyecto y por último con los apéndices que son una colección de documentos, que complementan el estudio del presupuesto universitario y el desarrollo del sistema en general.

El orden sugerido en el desarrollo del sistema esta basado, por un lado en la solicitud específica de la subdirección de planeación de la Dirección General de Programación Presupuestal y por otro su implícita integración a una serie de aplicaciones existentes para el manejo de la administración y control de la información sobre el presupuesto universitario.

De esta manera damos paso al primero de los capítulos, el estudio del Presupuesto Universitario.

ÍNDICE

Capítulo	Pagina
Índice	II
Introducción	VII
I El Presupuesto Universitario, Partidas del grupo 300.	1
1.1 Objetivo.	1
1.2 El presupuesto por programas en la Universidad Nacional Autónoma de México.	1
1.3 ¿Qué es el presupuesto por programas?.	2
1.4 Clasificaciones presupuestales.	6
1.4.1 Clasificación por funciones y programas.	6
1.4.2 Clasificación por ramos.	9
1.5 Código Programático.	11
1.5.1 Función y Programa.	11
1.5.2 Subprograma.	12
1.5.3 Dependencia.	13
1.5.4 Subdependencia.	13
1.5.5 Partida de gasto.	13
1.5.6 Dígito de control y Dígito verificador (DC).	14
1.6 Partidas del Grupo 300 (Partidas especiales).	16
1.7 Bases para el desarrollo del sistema.	18
II Levantamiento de Información.	20
II.1 Objetivo.	20
II.2 Estudio de Factibilidad.	20
II.2.1 Definición del Problema.	20
II.2.2 Sistema Actual.	20
II.2.3 Requerimientos del sistema.	21
II.3 Problemática del sistema.	23
II.4 Objetivo del sistema a Desarrollar.	24
II.4.1 Alternativas de solución.	24
II.4.2 Restricciones del sistema.	24
II.5 Requerimientos inmediatos.	25
II.5.1 Usuarios con los que se realizaran las entrevistas.	25
II.6 Cuestionarios aplicados a los usuarios.	26
II.6.1 Guía para la entrevista con usuarios finales del sistema SIPES.	26
II.6.2 Documentación por parte de los usuarios para el sistema SIPES.	27
II.6.3 Cuestionario Exhaustivo para Reportes e Información.	27
II.7 Respuestas obtenidas de los cuestionarios.	28
II.7.1 Actividades del encargado en computo de los programas PRIDE, PAIPA, CATEDRA, PEPASIG Y FOMDOC.	28
II.7.2 Actividades del encargado en computo de los programas EDPAC, CALIDAD Y EFICIENCIA.	29
II.8 Conclusiones del Levantamiento de información.	32
III Descripción del Sistema y Análisis.	33

III.1 Objetivo.	33
III.2 Plan general.	33
III.2.1 El módulo de administración.	35
III.2.2 El módulo SIPES I.	35
III.2.3 El módulo SIPES II.	35
III.2.4 Los módulos complementarios.	36
III.3 Modelo de contexto.	37
III.3.1 SIPES I – Partida 351 Estimulos a la productividad y al rendimiento del personal académico de asignatura y de fomento a la docencia para el personal de carrera.	38
III.3.1.1 DIPASI.	38
III.3.1.2 FOMDOC.	39
III.3.1.3 PEPASIG.	39
III.3.2 SIPES I – Partida 355 Prima de desempeño al personal Académico.	40
III.3.2.1 PRIDE Y PAIPA.	40
III.3.3 SIPES I – Partida 356 Reconocimiento catedrático UNAM.	40
III.3.3.1 CATEDRA.	40
III.3.4 SIPES II – Partida 341 Complemento al salario por calidad y eficiencia en el trabajo del personal administrativo.	41
III.3.4.1 CALIDAD Y EFICIENCIA.	41
III.3.5 SIPES II – Partida 344 Estimulos y gratificaciones complementarias.	41
III.3.5.1 EDPAC.	41
III.4 Modelo de eventos.	42
III.4.1 Los elementos del modelo de eventos.	42
III.4.2 Lista de eventos.	43
III.4.3 El diccionario de eventos.	45
III.4.3.1 SIPES I – Partida 351 Estimulos a la productividad y al rendimiento del personal académico de asignatura y de fomento a la docencia para el personal de carrera.	45
III.4.3.1.1 DIPASI.	45
III.4.3.1.2 FOMDOC.	48
III.4.3.1.3 PEPASIG.	50
III.4.3.2 SIPES I – Partida 355 Prima de desempeño al personal Académico.	53
III.4.3.2.1 PRIDE Y PAIPA.	53
III.4.3.3 SIPES I – Partida 356 Reconocimiento catedrático UNAM.	56
III.4.3.3.1 CATEDRA.	56
III.4.3.4 SIPES II – Partida 341 Complemento al salario por calidad y eficiencia en el trabajo del personal administrativo.	59
III.4.3.4.1 CALIDAD Y EFICIENCIA.	59
III.4.3.5 SIPES II – Partida 344 Estimulos y gratificaciones complementarias.	62
III.4.3.5.1 EDPAC.	62
III.5 Modelo de Información.	65

III.5.1 SIPES I – Partida 351 Estimulos a la productividad y al rendimiento del personal académico de asignatura y de fomento a la docencia para el personal de carrera.	67
III.5.1.1 DIPASI.	67
III.5.1.2 FOMDOC.	71
III.5.1.3 PEPASIG.	72
III.5.2 SIPES I – Partida 355 Prima de desempeño al personal Académico.	73
III.5.2.1 PRIDE Y PAIPA.	73
III.5.3 SIPES I – Partida 356 Reconocimiento catedrático UNAM.	77
III.5.3.1 CATEDRA.	77
III.5.4 SIPES II – Partida 341 Complemento al salario por calidad y eficiencia en el trabajo del personal administrativo.	79
III.5.4.1 CALIDAD Y EFICIENCIA.	79
III.5.5 SIPES II – Partida 344 Estimulos y gratificaciones complementarias.	80
III.5.5.1 EDPAC.	80
III.6 Conclusiones del análisis.	82
IV Diseño.	83
IV.1 Objetivo.	83
IV.2 Modelo arquitectónico.	84
IV.2.1 Distribución geográfica de los requerimientos de proceso.	86
IV.2.2 Los componentes de hardware para las maquinas cliente.	86
IV.2.3 Los componentes de hardware para las maquinas servidor.	87
IV.2.4 Configuración y cantidad de niveles de hardware cliente-servidor.	88
IV.2.5 Los mecanismos y lenguajes de comunicación de red.	88
IV.2.6 El sistema operativo.	88
IV.2.7 El paradigma de desarrollo.	89
IV.2.8 El lenguaje de presentación y de codificación de fondo.	89
IV.2.9 El sistema de administración de base de datos.	90
IV.2.10 La ubicación de los procesos.	91
IV.2.11 La ubicación de los datos físicos.	91
IV.3 Diseño de la base de datos.	93
IV.3.1 Bases de datos relacionales.	93
IV.3.2 Base de datos SISAP.	95
IV.3.3 Tablas de datos del sistema SIPES I y II.	95
IV.4 Diseño de los componentes internos y lógica de programación.	102
IV.4.1 SIPES I – Partida 351 Estimulos a la productividad y al rendimiento del personal académico de asignatura y de fomento a la docencia para el personal de carrera.	102

Índice

IV.4.1.1	DIPASI	102
IV.4.1.1.1	Diseño de interfaz	102
IV.4.1.1.2	Lógica de programación	103
IV.4.1.2	FOMDOC	103
IV.4.1.2.1	Diseño de interfaz	103
IV.4.1.2.2	Lógica de programación	104
IV.4.1.3	PEPASIG	105
IV.4.1.3.1	Diseño de interfaz	105
IV.4.1.3.2	Lógica de programación	105
IV.4.2	SIPES I - Partida 355 Prima de desempeño al personal académico.	106
IV.4.2.1	PRIDE Y PAIPA	106
IV.4.2.1.1	Diseño de interfaz	106
IV.4.2.1.2	Lógica de programación	107
IV.4.3	SIPES I - Partida 356 Reconocimiento catedrático UNAM.	109
IV.4.3.1	CATEDRA	109
IV.4.3.1.1	Diseño de interfaz	109
IV.4.3.1.2	Lógica de programación	109
IV.4.4	SIPES II - Partida 341 Complemento al salario por calidad y eficiencia en el trabajo del personal administrativo.	110
IV.4.4.1	CALIDAD Y EFICIENCIA	110
IV.4.4.1.1	Diseño de interfaz	110
IV.4.4.1.2	Lógica de programación	111
IV.4.5	SIPES II - Partida 344 Estímulos y gratificaciones complementarias.	112
IV.4.5.1	EDPAC	112
IV.4.5.1.1	Diseño de interfaz	112
IV.4.5.1.2	Lógica de programación	112
IV.5	Conclusión del Diseño.	114
V	Desarrollo e Implementación.	115
V.1	Objetivo.	115
V.2	Programación del sistema.	115
V.2.1	SIPES I - Partida 351 Estímulos a la productividad y al rendimiento del personal académico de asignatura y de fomento a la docencia para el personal de carrera.	115
V.2.1.1	Partida 351 - DIPASI	115
V.2.1.1.1	Formas y Unidades	115
V.2.1.2	Partida 351 - FOMDOC	122
V.2.1.2.1	Formas y Unidades	122
V.2.1.3	Partida 351 - PEPASIG	128
V.2.1.3.1	Formas y Unidades	128
V.2.2	SIPES I - Partida 355 Prima de desempeño al personal Académico.	130
V.2.2.1	Partida 355 - PRIDE Y PAIPA	130
V.2.2.1.1	Formas y Unidades	130
V.2.3	SIPES I - Partida 356 Reconocimiento catedrático UNAM.	144

Índice

V.2.3.1 Partida 356 - CATEDRA	144
V.2.3.1.1 Formas y Unidades	144
V.2.4 SIPES II - Partida 341 Complemento al salario por calidad y eficiencia en el trabajo del personal administrativo.	151
V.2.4.1 Partida 341 - CALIDAD Y EFICIENCIA	151
V.2.4.1.1 Formas y Unidades	151
V.2.5 SIPES II - Partida 344 Estimulos y gratificaciones complementarias	153
V.2.5.1 Partida 344 - EDPAC	153
V.2.5.1.1 Formas y Unidades	153
V.2.6 Módulos extras	156
V.3 Manual de operación.	157
V.3.1 Ejecución SIPES I	157
V.3.2 Operación SIPES I	159
V.3.3 SIPES I - Partida 351 Estimulos a la productividad y al rendimiento del personal académico de asignatura y de fomento a la docencia para el personal de carrera.	159
V.3.3.1 DIPASI	159
V.3.3.2 FOMDOC	161
V.3.3.3 PEPASIG	164
V.3.4 SIPES I - Partida 355 Prima de desempeño al personal académico.	166
V.3.4.1 Partida 355 - PRIDE Y PAIPA	166
V.3.5 SIPES I - Partida 356 Reconocimiento catedrático UNAM.	169
V.3.5.1 Partida 356 - CATEDRA	169
V.3.6 Módulos extras	171
V.3.7 Ejecución SIPES 2	174
V.3.8 Operación SIPES 2	175
V.3.9 SIPES II - Partida 341 Complemento al salario por calidad y eficiencia en el trabajo del personal administrativo.	175
V.3.9.1 CALIDAD Y EFICIENCIA	175
V.3.10 SIPES II - Partida 344 Estimulos y gratificaciones Complementarias	177
V.3.10.1 EDPAC	177
V.4 Pruebas	178
V.5 Mantenimiento	179
V.6 Conclusión del desarrollo e implementación	180
Conclusión.	181
Anexo A) Glosario de términos	182
Bibliografía.	185

Capítulo I

EL PRESUPUESTO UNIVERSITARIO, PARTIDAS DEL GRUPO 300.

I.1.- Objetivo :

En este capítulo se estudiará la estructura del presupuesto universitario y la composición e integración del grupo 300 (partidas especiales).

I.2.- El presupuesto por programas en la Universidad Nacional Autónoma de México.

En una primera fase de acercamiento debemos definir al presupuesto como un elemento fundamental para apoyar la realización de los programas de trabajo de cualquier organización o institución. El objeto del mismo es organizar los recursos asignados para la ejecución, cumplimiento de metas, objetivos específicos y bien delineados. Sirve para expresar y evaluar el grado de racionalidad en el uso de los recursos disponibles. En él se utilizan las asignaciones máximas de gasto para un período bien definido; su disponibilidad y origen de las fuentes de dicho financiamiento.

En el año de 1969 se estableció la política actual de presupuestación por programas debido a la expansión de los recursos educativos, así como a las tareas de investigación y extensión universitaria, esta reforma llevada a cabo por el Rector Javier Barrios Sierra dentro de la reforma administrativa de dicho año se implantó con los siguientes objetivos:

- a) Correlacionar los programas de trabajo con los recursos financieros disponibles.
- b) Establecer un orden de prioridades para normar las erogaciones.
- c) Obtener el máximo aprovechamiento económico y social de acuerdo con los recursos disponibles y los fines básicos de la Universidad.

1.3.- ¿Qué es el presupuesto por programas?

Es una técnica presupuestal que permite correlacionar los programas, y los subprogramas con los recursos financieros, teniendo como sustento de los objetivos y las metas de las dependencias que forman parte de la Institución.

Las funciones básicas de la universidad se definen como la parte del esfuerzo total, que se encamina a lograr los propósitos generales de la Institución y son:

- La docencia a nivel superior
- La docencia a nivel bachillerato y técnico
- La investigación
- La extensión universitaria
- La gestión institucional

La división en partes bien definidas de cada una de estas funciones se conoce como programa.

Por otro lado un programa comprende una serie de actividades homogéneas, encaminadas a cumplir con los propósitos expresados en una función en forma de subprograma; es la desagregación de un programa, ocasionado por las diferentes metas que cada una contempla.

Los objetivos son la expresión cualitativa de los fines que se pretenden alcanzar y conforman la directriz para el establecimiento de los programas de trabajo. Estos objetivos deben ser planteados de forma clara y realista, un ejemplo de esto lo constituye una dependencia del nivel bachillerato su objetivo es: Impartir enseñanza a nivel medio superior a fin de que los estudiantes continúen sus estudios a nivel licenciatura.

Las metas constituyen la expresión cuantitativa de los objetivos que se persiguen en cada dependencia para un ciclo de tiempo determinado. Por ejemplo: En el caso de la función docencia: el número de alumnos que se pretende atender; En el caso de la función investigación, el número de proyectos o asesorías a llevar a cabo, en el caso de la extensión universitaria, el número de eventos a realizar; Dentro de la función de gestión institucional, el número de servicios a proporcionar en algunas de las áreas administrativas.

La configuración de un presupuesto por programas se basa en los objetivos y metas a alcanzar, de los cuales se derivan los recursos presupuestales necesarios para su realización.

Alcanzar dichos objetivos es el aspecto medular para el proceso de evaluación presupuestaria. Estos dos aspectos constituyen las diferencias principales con el presupuesto tradicional, donde el énfasis se ubica en la identificación de los servicios y bienes por adquirir, así como los recursos financieros para lograrlos.

Podemos definir las diferencias entre un presupuesto tradicional y el presupuesto por programas de la siguiente manera:

Tipos de Presupuesto	
Tradicional	Por Programas
Es un informe financiero que relaciona los ingresos y los gastos.	Es un documento en el que se plantea explícitamente objetivos y metas a alcanzar, así como los recursos necesarios para su realización.
Pone especialmente énfasis en los bienes que se adquieren y los servicios que se contratan.	Los recursos se asignan de acuerdo a los objetivos y metas propuestos.
Se evalúa de acuerdo a criterios financieros y contables.	Se evalúa de acuerdo al cumplimiento de los objetivos y metas planteados y de acuerdo a la utilización de los recursos asignados.

Las ventajas de la aplicación del presupuesto por programas implican los siguientes rubros:

- a) Establecen y jerarquizan los objetivos y metas.
- b) Marca claramente los objetivos, metas y responsabilidades de cada una de las partes que forman a las organizaciones.
- c) Racionaliza los recursos y los procesos administrativos para alcanzar de manera eficaz y eficiente los objetivos y metas propuestos.
- d) Permiten obtener información de las acciones realizadas por la institución y de los recursos necesarios para su operación, esto implica ejercicios de evaluación más objetivos y precisos.
- e) Constituye un control financiero que permite la ejecución del gasto conforme a los objetivos y programas establecidos.
- f) Es un elemento de establecimiento de costo real, de un conjunto de actividades que dan forma a un programa o subprograma.
- g) Identifica los elementos necesarios para la planeación presupuestaria a corto, mediano o largo plazo.

Este conjunto de ventajas se basan esencialmente en varios principios, entre ellos y uno de los principales se encuentra la estructura programática, este constituye la parte medular del sistema de presupuesto por programas, en ellas se conjugan de manera sistemática los recursos financieros, con los programas y subprogramas que conducen a la realización de las funciones de la institución.

Debido a que los programas de trabajo de cada dependencia se relacionan directamente con las funciones institucionales, cada uno de ellas recae directamente en alguna de las siguientes:

La docencia a nivel superior

Se encamina a la formación de recursos humanos en el nivel superior (Licenciatura, Posgrado, etc.), por ejemplo:

Facultad de Ingeniería

División de estudios de profesionales

Objetivos :

Formar profesionales a nivel licenciatura en las carreras de Ingeniería Civil, Topografía y Geodésica, Mecánica (Industrial, Eléctrica y Electrónica); en Computación, Minería, Metalúrgica, Petrolera, Geológica y Geofísica.

Metas:

Atender a 11,378 alumnos

La docencia a nivel bachillerato y técnico

Dedicada a formar recursos humanos a nivel bachillerato, técnico y áreas afines. Por ejemplo:

Dirección General de la Escuela Nacional Preparatoria

Plantel No. 3 "Justo sierra"

Objetivos :

Formar alumnos a nivel bachillerato.

Metas :

Atender a 2,500 alumnos

La investigación

Se encarga de profundizar e incrementar los conocimientos, tanto en las ciencias exactas y naturaleza como en las sociales y humanidades. Por ejemplo:

Instituto de Física

Objetivo :

Llevar a cabo investigaciones básicas y aplicadas en el campo de Física Teórica y Experimental, a fin de colaborar al conocimiento dentro de sus campos, implementar mejores procesos que eleven el desarrollo tecnológico del país y capacitar a jóvenes investigadores desarrollando su creatividad.

Metas :

Desarrollar 51 proyectos de investigación

La extensión universitaria

Implica todas las actividades artísticas, científicas y culturales encaminadas a la difusión y extensión educativa, tanto en el ámbito universitario como en la sociedad en su conjunto. Por ejemplo:

Dirección de Actividades Musicales

Objetivo :

Difundir la música universal entre la comunidad universitaria y extrauniversitaria; fomentar la creatividad en los músicos y compositores mexicanos y difundir sus obras; Difundir la cultura musical a través de cursos y conferencias.

Metas :

Realizar 280 eventos

La gestión institucional

Constituida por los servicios inherentes a la dirección, planeación y administración, necesarios para la realización de todas las funciones universitarias. Por ejemplo:

Dirección General de Programación Presupuestal

Objetivo:

Colaborar con el patronato universitario en la formulación del presupuesto anual de ingresos y egresos, realizar los estudios necesarios para evaluar los programas presupuestales y generar la información necesaria, para una toma de decisiones adecuada propiciando así el mejor aprovechamiento de los recursos, con que cuenta la institución.

Metas:

Realizar 121 estudios.¹

¹ Los objetivos y metas descritos no coinciden necesariamente con los vigentes en las diferentes dependencias enunciadas; sirven solamente como guía de referencia de pasados ejercicios presupuestales, por lo tanto pueden existir variaciones o cambios en las metas o definición de los objetivos actuales, los datos actuales se emitirán al termino del presente ejercicio presupuestal en el Libro 99.

I.4.- Clasificaciones presupuestales.

La planeación, conformación, ejercicio y evaluación del presupuesto universitario requiere diferentes alternativas de clasificación que permitan su manipulación de manera objetiva, dependiendo de los requerimientos de información que deben de cubrirse, en primer lugar encontramos la clasificación por funciones, considerada la más importante, esto debido principalmente a que permite evaluar los recursos asignados a cada uno de los programas asociados a cada función, en segundo lugar tenemos una clasificación por objeto del gasto, esta permite obtener una visión sobre la forma en que se destinan los recursos de la Universidad, para el pago de remuneraciones personales (sueldos), gasto corriente y gasto de inversión; por último encontramos la clasificación por la naturaleza de las actividades a realizar, que se encuentra enlazado directamente con las funciones y programas, que cada dependencia realiza de acuerdo con sus objetivos y actividades cotidianas.

I.4.1.- Clasificación por funciones y programas.

Las funciones de la institución son el punto de partida, para identificar los niveles o las áreas que generan y dividen a los programas asociados, con cada una de ellas por ejemplo:

En la función docencia a nivel superior, existen 9 programas que corresponden a los niveles de enseñanza superior y a los programas relacionados, con las áreas destinadas a apoyar las actividades docentes tales como: Los programas de formación y actualización académica, servicios de apoyo académico y administrativo, etc.; derivado de lo anterior encontramos que una dependencia puede participar en mas de una función, por ejemplo: Una facultad imparte docencia, realiza investigación, desarrolla actividades de educación continua, o bien una dependencia administrativa donde su presupuesto se desagrega en todas las funciones tal como el presupuesto de la Dirección General de Obras o de la Dirección General de Protección a la Comunidad.

Esta clasificación permite obtener de manera clara y objetiva información de cómo esta distribuido el presupuesto universitario en las funciones encomendadas de manera sustantiva y adecuadamente en cada uno de los programas definidos.

La distribución de las funciones y los programas actuales de la Universidad Nacional Autónoma de México son:

Docencia

Nivel superior

Nivel bachillerato y técnico

Investigación

Extensión universitaria

Gestión institucional

La clasificación utilizando las funciones y programas sirve para determinar, el objeto y forma en que se distribuye el presupuesto de egresos, además indican la

magnitud de los recursos asignados y la participación porcentual, de cada función dentro del presupuesto total.

En el caso de la clasificación por objeto del gasto, sirve para identificar y obtener la naturaleza del gasto realizado en la universidad, es el resultado de la organización en diferentes grupos de gasto los que a su vez se dividen en subgrupos, integrados por conjuntos de partidas relacionadas entre sí, por el tipo de gasto al que hacen referencia por su naturaleza.

Esto se observa directamente en la lista de grupos de gasto utilizados para la organización del presupuesto:

Grupo	Descripción
100	Remuneraciones Personales
200	Servicios
300	Becas, Prestaciones y Estímulos
400	Artículos y Materiales de Consumo
500	Mobiliario y Equipo
600	Inmuebles y Construcciones
700	Erogaciones Condicionadas al Ingreso y Programas de Colaboración Académica

Grupo 100 Remuneraciones Personales

Comprende los recursos para pagar los sueldos del personal académico y administrativo de la Universidad Nacional Autónoma de México.

Grupo 200 Servicios

Incluye los recursos para la contratación de los diversos servicios (no personales, viáticos, derechos de autor, etc.) necesarios para la operación de la institución.

Grupo 300 Becas Prestaciones y Estímulos

Contempla los pagos para proporcionar al personal las prestaciones derivadas de los contratos colectivos de trabajo, así como los recursos destinados a los programas vigentes de becas nacionales y al extranjero, y de estímulos al personal académico. En este grupo se ubican las partidas que son objeto de estudio en este documento.

Grupo 400 Artículos y Materiales de Consumo

Son los gastos que se efectúan para la compra de diversos tipos de artículos, material y herramientas no inventariables, entre otros, que son necesarios para el desarrollo de las actividades en aulas, laboratorios, talleres y oficinas.

Grupo 500 Mobiliario y Equipo

Son las cantidades previstas para la adquisición de bienes muebles y equipo que incrementen el patrimonio universitario.

Grupo 600 Inmuebles y Construcciones

Son los recursos destinados a la adquisición de bienes inmuebles y a la construcción de obras por contrato.

Grupo 700 Erogaciones Condicionadas al Ingreso y Programas de Colaboración Académica

Incluye los recursos para hacer frente a los gastos para la implantación de proyectos, programas académicos y de extensión y otros específicos por su carácter temporal.

La clasificación por grupos de gasto, permite identificar la distribución del presupuesto de egresos en el pago de remuneraciones y prestaciones personales (grupos 100 y 300), gasto corriente (grupos 200, 400 y 700) y gasto de inversión (grupos 500 y 600)

Por la forma en que opera el presupuesto encontramos que las partidas pueden ejercerse en forma directa, centralizada o complementaria.

Partidas de operación directa

Son las partidas que las dependencias pueden ejercer en forma flexible y directa hasta por la cantidad asignada, pueden ser ejercidas por grupo hasta el monto total de las misma, por ejemplo:

Partida	Descripción
235	Servicios de Talleres Externos para Equipo de Transporte.
253	Telégrafos y Correos.

Partidas de ejercicio central

Son las partidas reguladas por alguna dependencia que con base en sus funciones centraliza y controla su ejercicio presupuestal, por ejemplo:

Partida	Descripción
216	Gastos por Reuniones de Trabajo.
233	Servicios de Mantenimiento para Equipo de Cómputo.

Partidas de ejercicio complementario

Son aquellas destinadas a un fin específico que se considera prioritario y cuyo ejercicio no debe rebasar, en principio, su asignación original. Estas partidas no forman parte del ejercicio directo por grupo, por ejemplo:

Partida	Descripción
221	Anuncios Varios
Ejercida por:	Dirección General de Información
241	Renta de Inmuebles
Ejercida por:	Dirección General del Patrimonio Universitario

I.4.2 Clasificación por ramos.

El criterio de organización para ordenar a las dependencias Universitarias, de acuerdo a la naturaleza de su actividad sustantiva permite la clasificación por ramos, de lo anterior obtenemos la siguiente clasificación:

Ramo	Descripción
100	Órganos de Dirección
200	Órganos de Investigación Humanística
300	Órganos de Investigación Científica
400	Facultades, Escuelas y Unidades Multidisciplinarias
500	Órganos Complementarios a la Docencia e Investigación
600	Órganos de Extensión y Vinculación Universitaria

Capítulo I - El presupuesto universitario, partidas del grupo 300

700	Órganos Para la Planeación, los Servicios Administrativos y Jurídicos
800	Órganos para Adaptación y Mantenimiento de Obras e Instalaciones
900	Órgano Coordinador de Servicios Institucionales

Esta clasificación permite obtener una descripción de la distribución del presupuesto, en los diferentes ramos que dan forma a la institución, además asocia a cada ramo, con la magnitud de los recursos asignados y su participación porcentual correspondiente.

1.5.- Código Programático.

El manejo operativo del presupuesto por programas, en cuanto a función, programa, subprograma, dependencia, subdependencia y partida de gasto requiere el código programático. Cada asignación de recursos dentro del presupuesto de la Universidad Nacional Autónoma de México, está identificado mediante este código. Los movimientos presupuestales, como el aumento o disminución de recursos o la evaluación en general del presupuesto universitario, no podrían estudiarse confiablemente de no contar con este instrumento.

El código programático está formado por un conjunto de dígitos ordenados, de forma sistemática y constituye el elemento central para procesar un cúmulo de información que requiere la técnica de presupuesto por programas, el cual, queda integrado de la siguiente manera:

PR	Función o Programa
SP	Subprograma
DEP	Dependencia
SD	Subdependencia
PAR	Partida de gasto
DC	Dígito de Control o Verificador

1.5.1.- Función y Programa.

La función se asocia a la naturaleza de los diferentes objetivos y funciones institucionales.

El programa se correlaciona, con el nivel o área de las actividades de trabajo de la dependencia, ubicados dentro de una determinada función.

Las Funciones y Programas actuales son :

Función Programa

Docencia

Nivel Superior

- 1.1 Educación de Licenciatura
- 1.2 Educación de Posgrado
- 1.3 Educación Complementaria
- 1.4 Coordinación Académica
- 1.5 Formación y Actualización Académica
- 1.6 Programas de desarrollo Académico
- 1.7 Servicios de Apoyo Académico
- 1.8 Servicios de Apoyo Administrativo
- 1.9 Educación Sistema Universidad Abierta

Nivel Bachillerato y técnico

- 2.1 Educación Media Superior
- 2.2 Educación Técnica
- 2.3 Educación Complementaria
- 2.4 Coordinación Académica
- 2.5 Formación y Actualización Académica
- 2.6 Programas de desarrollo Académico
- 2.7 Servicios de Apoyo Académico
- 2.8 Servicios de Apoyo Administrativo

Investigación

- 3.1 Investigación en Ciencias y Desarrollo Tecnológico
- 3.2 Investigación en Humanidades y Ciencias Sociales
- 3.3 Coordinación Académica
- 3.4 Formación Académica
- 3.5 Programas de Desarrollo Académico
- 3.6 Servicios de Apoyo Académico
- 3.7 Servicios de Apoyo Administrativo

Extensión Universitaria

- 4.1 Difusión de Actividades Artísticas, Científicas y Culturales
- 4.2 Coordinación
- 4.3 Extensión Educativa
- 4.4 Comunicación y Divulgación Universitaria
- 4.5 Vinculación con la Sociedad
- 4.6 Programas de Apoyo Técnico
- 4.7 Servicios de Apoyo Administrativo

Gestión Institucional

- 5.1 Dirección Institucional
- 5.2 Planeación y Normatividad
- 5.3 Programas de Apoyo Técnico
- 5.4 Prestaciones Contractuales
- 5.5 Servicios Administrativos Institucionales
- 5.6 Vigilancia y Fiscalización
- 5.7 Servicios de Apoyo a la Comunidad
- 5.8 Servicios de Apoyo Administrativo

I.5.2.- Subprograma.

Es la desagregación de un programa con el objeto de manejar metas homogéneas, esta operación permite la ejecución de determinadas acciones, para las cuales se ha efectuado una asignación presupuestal. Por ejemplo: Dentro del programa 11 se integran los subprogramas de cada una de las carreras, que se imparten en determinada facultad o escuela; es decir si nos referimos a la Facultad de Arquitectura explícitamente en su carrera Urbanismo, sabremos que esta se encuentra dentro del programa de Educación de licenciatura y los subprogramas que se describen a continuación:

PR.SP.DEP.SD.

424.00	Facultad de Arquitectura
424.01	División de Estudios Profesionales
11	Educación Licenciatura
11.01.424.01	Servicios de Apoyo
11.02.424.01	Arquitectura
11.03.424.01	Arquitectura del Paisaje
11.04.424.01	Urbanismo
11.20.424.01	Biblioteca
13	Educación Complementaria
13.02.424.01	Lenguas Extranjeras
13.03.424.01	Actividades
16	Programas de Desarrollo Académico
16.05.424.01	Desarrollo Académico
41	Difusión de Actividades Artísticas, Científicas y Culturales
41.02.424.01	Actividades Socio Culturales
43	Extensión Educativa
43.02.424.01	Educación Continua e Intercambio Académico

I.5.3.- Dependencia.

La dependencia es la unidad, que a partir de la estructura organizacional de la Universidad Nacional Autónoma de México, lleva a cabo diferentes actividades para el logro de metas y objetivos de acuerdo con la función general de la Institución.

I.5.4.- Subdependencia.

Es la unidad derivada de la partición de una dependencia, a fin de delimitar en forma clara y precisa, las funciones y responsabilidades que permiten un logro adecuado de las metas y objetivos preestablecidos por ejemplo:

PR.SP.DEP.SD.

111	01	Secretaría Ejecutiva de Junta de Gobierno y Consejo Universitario.
51		Dirección Institucional
51.02.111.01		Junta de Gobierno
112	01	Oficina del Rector
51		Dirección Institucional
51.02.112.01		Rectoría

I.5.5.- Partida de gasto.

Es el elemento presupuestario que se obtiene de la descripción de los subgrupos de gasto, que permite dosificar las erogaciones de acuerdo con la finalidad específica del mismo, por ejemplo:

Grupo	Subgrupo	Partida
100	Remuneraciones Personales	
	110	Sueldos a Profesores de Asignatura
		111 Sueldos a Profesores de Asignatura Ordinarios Definitivos
		112 Sueldos a Profesores de Asignatura Extraordinarios
		113 Complemento a Profesores Ordinarios de Asignatura
		114 Sueldos a Profesores de Asignatura Eméritos
		115 Sueldos por Asignación Académica
		116 Complementación a Profesores de Asignatura Ordinarios
		117 Sueldos a Profesores de Asignatura Ordinarios no Definitivos
		118 Sueldos para Regularización Académica
		119 Compensación por Antigüedad a Profesores de Asignatura Ordinarios

I.5.6.- Dígito de control y verificador (DC).

Los últimos dos dígitos identifican, en primer lugar a la dependencia centralizadora del gasto y el segundo constituye el dígito verificador, el cual permite comprobar la correcta integración del código programático de la siguiente forma:

PR	SP	DEP	SD	PAR	DC
13	03	411	01	151	01

La dependencia encargada de realizar el proceso del Presupuesto Universitario, es la Dirección General de Programación Presupuestal, esta se encuentra dividida en 4 subdirecciones que llevan a cabo los procesos de Planeación, Análisis, Operación, Registro y Control del presupuesto Universitario.

La identificación de cada una de las actividades, se realiza a través del código programático el cual contiene información acerca de los programas, subprogramas, dependencias, subdependencias y partidas que conforman en su totalidad al presupuesto por programas, una de las ventajas de utilizar esta técnica se observa en una completa organización, de la información por objetivos de la institución o bien por diferentes criterios tales como:

- Por dependencias
- Por subdependencias
- Por ramo (de dependencias)
- Por partidas
- Por grupos de partidas
- Por sistemas
- Por subsistemas

Estos niveles de organización aportan importante información sobre diferentes tópicos del presupuesto universitario, en este caso, en particular haremos referencia a las partidas especiales pertenecientes al grupo 300.

Las partidas de gasto son elementos presupuestarios, en las que se dividen, los subgrupos de gasto y que ilustran las erogaciones de acuerdo con el objeto específico del gasto, un grupo de gasto es el elemento presupuestal que agrupa los bienes y servicios que la institución adquiere conforme a su naturaleza intrínseca, esto nos indica como es que sistemáticamente se organizan los recursos.

I.6. - Partidas del Grupo 300 (Partidas especiales).

En este caso, en particular haremos referencia al grupo 300 el cual se conoce como Becas, Prestaciones y Estímulos.

La siguiente tabla muestra las partidas del grupo 300 también conocidas como especiales que serán estudiadas y analizadas durante el presente trabajo.

Grupo	Partida	Prestación	Descripción
340			Prestaciones Derivadas de los Contratos Colectivos de Trabajo.
	341	CALIDAD Y EFICIENCIA	Complemento al Salario por Calidad y Eficiencia en el Trabajo del Personal Administrativo.
	344	EDPAC	Estímulos y Gratificaciones Complementarias.
350			Estímulos al Personal Académico.
	351	DIPASI, FOMDOC, PEPASIG	Estímulos a la Productividad y Rendimiento del Personal Académico de Asignatura y de Fomento a la docencia para el Personal de Carrera.
	355	PRIDE Y PAIPA	Prima de Desempeño al Personal Académico.
	356	CATEDRA	Reconocimiento Catedrático UNAM.

La idea inicial del sistema solo abarca estas partidas, sin embargo se planea ampliar su ámbito a todas las que pertenecen al grupo 300.

Todas estas partidas se caracterizan por abarcar diferentes aspectos del presupuesto universitario, en general todas se refieren a percepciones relacionadas a las cláusulas del contrato colectivo de trabajo o bien a grupos y programas de apoyo y desarrollo institucional.

Debido a la naturaleza heterogénea de las mismas se requiere de un sistema que cumpla diferentes funciones y de esta manera cumplir con los requerimientos de información, de las diferentes subdirecciones y en general de las dependencias Universitarias.

Una vez establecido el vínculo con el presupuesto universitario damos paso a la descripción general del proceso que llevaremos a cabo en la creación del sistema, esto involucra las diferentes fases que serán desarrolladas conforme se avance en los capítulos de la tesis y plantean una metodología bien establecida para el desarrollo de este tipo de proyectos.

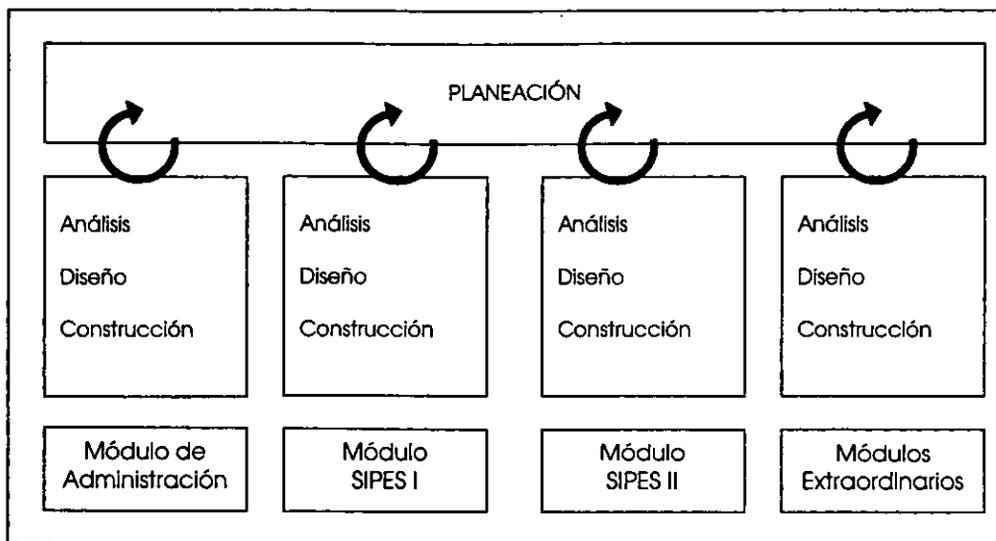
I.7.- Bases para el Desarrollo del Sistema.

El desarrollo del sistema se llevará, a cabo aplicando una serie de técnicas que permitirán alcanzar los objetivos por capítulo del mismo, la creación de un sistema GUI (Interfaz Gráfica de Usuario) cliente servidor requiere de una planeación e implementación diferente a la que se utiliza en los sistemas tradicionales, de esta manera hemos elegido un conjunto de técnicas que permiten la obtención de los objetivos establecidos, en cada uno de los capítulos divididos de la siguiente manera :

- 1) Levantamiento de información: Como parte del inicio de la creación del sistema se utilizarán cuestionarios, entrevistas, resúmenes, tablas de referencia y resultados, como los mecanismos de recolección de información inicial, los cuales proveen de un medio idóneo para investigar y observar el comportamiento actual del sistema basados en una serie de preguntas y cuestionamientos que nos permitan desarrollar en etapas posteriores, la solución a los diferentes problemas que se plantearán.
- 2) En la descripción del sistema y el análisis se llevarán a cabo el plan general del proyecto, el modelo de contexto, el modelo de eventos, el modelo de información y el modelo arquitectónico, los cuales constituyen en conjunto el análisis del sistema, los datos, las relaciones y los eventos que describen puntualmente cada uno de los módulos, hasta llegar al punto de la administración física de la información para los diferentes administradores y usuarios.
- 3) Dentro del diseño se ataca claramente el concepto GUI y la implementación en principio a través del diseño de la base de datos, la de interfaces externas y el de componentes internos, estos modelos de diseño permiten ir al detalle y explicar las respuestas a las preguntas y solicitudes planteadas en la fase de análisis, involucran una mezcla de información obtenida en las fases anteriores y proveerán a las siguientes de la información necesaria para la consecución de objetivos más particulares.
- 4) Por último el desarrollo e implementación basan toda información que requieren en los documentos anteriores, utilizan técnicas de programación GUI de OOP y programación tradicional, llamados al servidor SQL y diferentes pruebas de validación y comprobación que permitan al sistema el máximo nivel de funcionalidad.

Como se observa la creación del sistema de cómputo ha seguido un desarrollo en fases, este enfoque divide en partes funcionales el desarrollo del sistema completo.

En este tipo de desarrollo, se observa como a partir de la planeación inicial, la retroalimentación constante entre fases da como resultado la consecución total de los objetivos y permite su constante mantenimiento. Con la correspondiente independencia entre módulos podemos observar como las diferentes fases, requieren de su anterior para avanzar y sin embargo la retroalimentación constante, entre módulos crea un avance en experiencia que se refleja en una reducción de tiempos de creación conforme avanza el ciclo de desarrollo en cada módulo.



Capítulo I, Figura I.1 Enfoque de cascada en fases utilizado en el desarrollo del sistema

A la técnica de desarrollo ilustrada en la figura I.1 se le conoce como enfoque de cascada en fases, el cual fue utilizado como guión principal de desarrollo. Este esquema nos permite por un lado añadir módulos independientes y funcionales con una base de planeación común, esto último se relaciona directamente con las características del sistema, que desamos que cumpliera el sistema inicialmente: una serie de partidas (inicio común) del presupuesto universitario con características particulares en cuanto a manejo de información y bases de operación (fases del diagrama), que requieren un constante mantenimiento conforme los requerimientos particulares de información, lo cual implica modificaciones o ampliaciones (retroalimentación de las cascadas).



Capítulo II

LEVANTAMIENTO DE INFORMACIÓN.

II.1.- Objetivo.

En este capítulo se describirán las técnicas de levantamiento de información, utilizadas para el desarrollo del sistema y los resultados obtenidos.

Se describen los procesos de encuestas, recopilación de la información, análisis y evaluación de las necesidades que busca cubrir el sistema.

II.2.- Estudio de Factibilidad.

El desarrollo de este programa se basa en la necesidad de un sistema que ponga al alcance de los usuarios de la subdirección de planeación, una serie de datos necesarios para la administración del presupuesto universitario, en el caso de las partidas conocidas como especiales, esta necesidad involucra el uso de los recursos de cómputo al alcance en la Dirección General de Programación Presupuestal, que consisten principalmente en una plataforma cliente servidor multiusuario, ya utilizada hasta el momento en otras aplicaciones para el manejo del presupuesto universitario, en el presente capítulo se dará inicio a la creación de los módulos, que constituirán al sistema y en primer lugar se da cuenta de una investigación acerca de lo que se utiliza actualmente para resolver en mayor o menor medida las necesidades de información a diferentes niveles.

A continuación se estudiarán los datos iniciales que se poseen para la creación del sistema, se define el problema, el estado actual del sistema, los requerimientos del nuevo programa y las alternativas de iniciales de solución, esto definirá la factibilidad de uso y aplicación del sistema, lo que permitirá definir en posteriores capítulos los métodos a seguir para la conclusión de los objetivos.

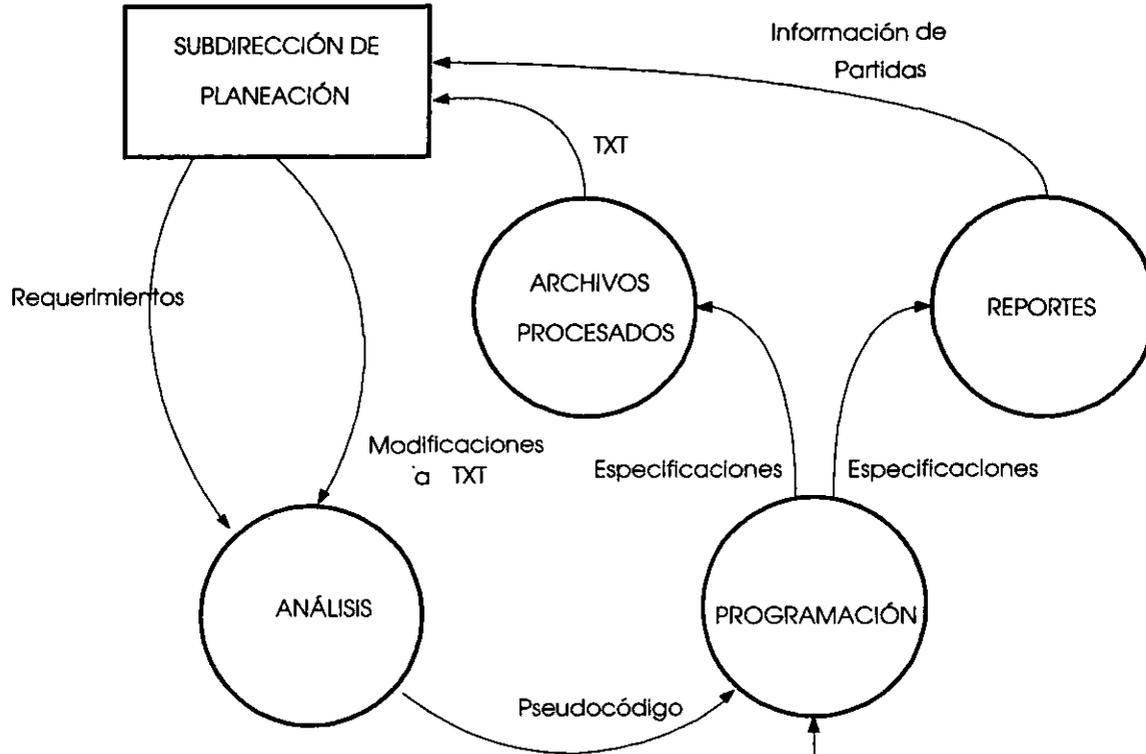
II.2.1.- Definición del Problema.

En este apartado se pretende dar una visión del problema a resolver, con el propósito de conocer el funcionamiento global del sistema actual, para dar cabida al nuevo sistema.

II.2.2.- Sistema Actual.

En el sistema de trabajo actual se realizan los siguientes procedimientos : (Vease Figura No. II.1)

Las solicitudes de información son principalmente de la Subdirección de Planeación Presupuestaria, estos requerimientos se basan en la información procesada por la Dirección General de Personal, Dirección General de Asuntos del Personal Académico entre otras, con el fin de obtener un control de gasto de las partidas, para



Capítulo II, Figura II.1

hacer proyecciones y en su caso presupuestarlas con los criterios de la Subdirección de Planeación de la Dirección General de Programación Presupuestal.

En la subdirección de cómputo, se analizan estos requerimientos para elaborar los programas necesarios, con el fin de cubrir las necesidades con archivos y reportes que solicite la Subdirección para su evaluación.

La subdirección de planeación, analiza los resultados acerca de las partidas presupuestales, efectuando los ajustes pertinentes y en caso necesario para que se lleven a cabo por la subdirección de cómputo.

La parte principal del proceso actual se lleva a cabo en una máquina mainframe A7 la cual produce los archivos de salida necesarios para la Subdirección de Planeación, estos archivos también son manipulados en equipo PC con programas como Dbase III Plus y Excel, este panorama inicial es el que busca superar con el nuevo sistema sobre una plataforma cliente - servidor.

El análisis del sistema de cómputo actual, nos permite dilucidar que partes del sistema pueden ser aprovechados y establecen una guía primaria de los flujos de información actuales, los resultados de este análisis sólo son posibles, con un adecuado levantamiento de información acerca de los procesos y métodos de trabajo actuales.

Para obtener una idea clara del estado actual del proceso de la información, a continuación, se muestra el diagrama de flujo de datos actual, el cual deberá ser modificado tomando en cuenta las características deseables del nuevo sistema.

Este diagrama muestra como funciona el sistema actual de trabajo, donde intervienen las diferentes subdirecciones y el flujo de datos entre ellas, se ha optado por generalizar en cuanto a su funcionamiento; en ciertas situaciones el esquema puede variar aunque no de manera significativa, si afecta susceptiblemente al proceso en general, además se debe considerar lo heterogéneo de las partidas que afectan de igual manera la operación de los procesos actuales.

Al ser realizado un requerimiento de información, la Subdirección de Planeación hace la petición a la Subdirección de Cómputo para satisfacer dicha solicitud, esto genera tiempos de espera ocasionados principalmente por las caídas del sistema A7, tiempo de programación (a cada nueva solicitud se requiere de la modificación del programa original), saturación de procesos, y/o enlaces excesivos por parte de los usuarios, además de tener una dependencia absoluta, de la Subdirección de Cómputo y en ciertas ocasiones no se satisfacen de manera optima los diversos requerimientos, actualmente los reportes y todo el proceso de la información lo realiza la Subdirección de Cómputo.

II.2.3.- Requerimientos del Sistema.

La creación del nuevo sistema se encuentra basada principalmente en la solicitud original de la Subdirección de Planeación la cual se describe en las tablas incluidas en el análisis de las partidas del grupo 300.

Esta solicitud aplica la generación de archivos, reportes e informes, además de los procesos que actualmente sigue la subdirección.

Capítulo II - Levantamiento de Información

Lo heterogéneo de la información impide un tratamiento general muy profundo, pero permite una visión general de lo que se desea, el sistema debe ser capaz de:

- a) Emitir reportes semanales, quincenales, mensuales o temporales de la información procesada.
- b) Generación de archivos de datos
- c) Proceso de la información sin la necesidad de intervención de la subdirección de cómputo.
- d) Emisión de reportes e informes, en pantalla, impresión o archivo en disco.
- e) Debe ser compatible con la plataforma cliente/servidor existente.

Estas solicitudes se han realizado en algunos casos de manera tácita, mientras que otros se derivan de las necesidades inminentes del nuevo diseño.

II.3.- Problemática del sistema.

Debido al cambio de tecnología empleada en la Universidad Nacional Autónoma de México que en el caso de la Dirección General Programación Presupuestal, se está erradicando el uso de la computadora Main Frame A7, que ejecuta diferentes programas realizados en Algol, a servidores Sun que atienden clientes de Power Builder y Delphi bajo Windows a través del manejador de base de datos para Sun Sybase, es necesario desarrollar un nuevo sistema que soporte esta nueva configuración.

Los formatos de reportes actuales requieren modificaciones de acuerdo a las nuevas solicitudes y requerimientos realizados por la Subdirección de Planeación y que se adapten a la nueva plataforma.

A esto se agrega la necesidad de los usuarios de no depender de las personas que controlan los procesos y ser independientes en la generación y obtención de la información requerida cotidianamente, esto implica la realización de un programa de fácil acceso para el usuario.

II.4.- Objetivo del Sistema a Desarrollar.

Desarrollar un sistema que interactúe con el usuario en la subdirección de planeación presupuestal, a fin de que permita generar la información sobre las partidas de prestaciones y estímulos a través de la generación de reportes o archivos, que reflejen el estado de los datos utilizando la nueva plataforma cliente-servidor.

II.4.1.- Alternativas de solución.

Este apartado nos permite describir las propuestas a la problemática detectada, con el propósito de plantear diferentes alternativas de solución que nos permitan tomar una serie de decisiones sobre el curso que tomará el proyecto.

El escenario principal plantea 2 alternativas:

1. La primera alternativa es crear un sistema, que permita satisfacer las necesidades de generación de reportes y archivos con información contenida en la nueva plataforma, tomando en cuenta únicamente los requerimientos de la Subdirección de Planeación.
2. La segunda alternativa se plantea como un complemento, a la solicitud original que permita satisfacer las necesidades de generación de reportes y archivos, tomando en cuenta los requerimientos de información, sobre las partidas correspondientes a Prestaciones y Estímulos que no están contempladas originalmente como son Gratificación por Separación, por Jubilación, Suministro y Ayuda para Guarderías, Despensas y Canastillas para el personal Académico y Administrativo, Uniformes y Ropa de Trabajo, Ropa y Artículos de Protección.

II.4.2.- Restricciones del sistema.

Se utilizará Delphi para generar al cliente del sistema, así como Sybase del lado del servidor, la comunicación lógica del cliente con el servidor será responsabilidad del sistema; mientras que las configuraciones de equipo, cableado y conexiones están fuera del alcance de este.

Como fuente de Información sobre las partidas se usará la Base de Datos SISAP, utilizada en la Dirección General de Programación Presupuestal como la fuente principal de información acerca del presupuesto universitario en la nueva plataforma.

II.5.- Requerimientos Inmediatos.

Los requerimientos propios de la Subdirección de Planeación Presupuestaria fueron planteados principalmente en las entrevistas realizadas a los encargados de las partidas del grupo 300, la integración de la solicitud mas las inquietudes resultado de las entrevistas con los usuarios se integraran en una tabla que describe la prestación objeto de estudio, su definición y el responsable, la subdirección que la maneja y los requerimientos hasta la fecha por parte de la misma, para dar seguimiento adecuado a la información.

Hasta el momento solo en algunos casos tenemos información disponible, de esta manera, necesitamos cubrir aun, diferentes requerimientos de información por parte de los futuros usuarios acerca de los archivos, reportes e informes del resto de las partidas.

II.5.1.- Usuarios con los que se realizaran las entrevistas.

Con el fin de obtener información general se deben concertar citas con las siguientes funcionarios:

Subdirector de Planeación Presupuestaria de la Dirección General de Programación Presupuestal

Coordinador de Estudios Presupuestales de la Subdirección de Planeación Presupuestal

Coordinador de Información de la Subdirección de Planeación Presupuestal.

Así mismo, para obtener información detallada y necesidades específicas para el buen control y seguimiento del gasto de las partidas, correspondientes a prestaciones y estímulos, se debe concertar cita con las siguientes personas:

Jefe de Unidad de la Coordinación de Estudios Presupuestales (Responsable de: PAPIIT, PAPIME, RECANT, EDPAB, EDPAC)

Jefe de Unidad de la Coordinación de Estudios Presupuestales (Responsable de: PRIDE, PEPASIG, FOMDOC)

Dichas citas serán acordadas con el Jefe de Proyectos especiales de la Subdirección de Cómputo.

II.6.- Cuestionarios aplicados a los usuarios.

El trabajo de acercamiento al proyecto da inicio con el levantamiento de información, como parte importante se realizaron una serie de cuestionarios y entrevistas con los usuarios finales del SIPES, en estas entrevistas se analizó la forma en que actualmente trabajan, en conjunto la Subdirección de Planeación y la Subdirección de Cómputo de la DGPP, se puso de manifiesto que los usuarios requieren por un lado una reducción de los tiempos de entrega de la información, una actualización urgente de las fuentes de información, debido al tránsito hacia el año 2000 el cual no es soportado por el sistema actual, también se planteo directamente una mayor interacción con el programa, que genera la información ya que existe una dependencia total entre las Subdirecciones de Cómputo y Planeación para la generación de información.

Los cuestionarios se dividieron en 2 partes principalmente. En primer lugar los generales, que nos indican de manera global el estado que guarda la generación de información, quien la maneja, en que momento la necesita y como interviene cada una de las partes que integran a los procesos. En segundo lugar una vez identificadas estas fuentes de información, se realizaron cuestionarios más detallados sobre los procesos particulares. La conjunción de ambos, dio como resultado una visión general y al detalle de cada una de las partidas que se pretende administrar con el programa, la identificación de esta información dio como resultado :

- 1) Quien genera y utiliza la Información requerida.
- 2) Saber como se opera actualmente.
- 3) La prioridad de los procesos a realizar.
- 4) Obtener reportes y archivos actuales como guías de trabajo propuesto.

De esta manera los datos obtenidos nos permiten tomar decisiones Iniciales acerca del sistema y la manera en que evoluciona. A continuación transcribimos las entrevistas más importantes y los resultados obtenidos, con fines de claridad no se incluirá todo el material recabado que involucra una serie de reportes y códigos, fuente de los programas actuales, por lo tanto se ha optado por colocar las entrevistas y los resultados mas relevantes en la siguiente sección.

II.6.1.- Guía para la entrevista con usuarios finales del sistema SIPES.

- Actividades en las cuales el sistema intervendrá y como lo afectará en sus funciones.
- 1.- De manera general, ¿Qué tramites realiza, para llevar a cabo sus funciones en la Subdirección de Cómputo en relación con las partidas especiales?.
 - 2.- ¿Cuál es su fuente Inicial de información?, ¿Quién solicita?, ¿Quién determina el seguimiento del control de información?, ¿Cuánto tiempo tarda ésta solicitud y el trámite en general?.
 - 3.- Existen solicitudes de Información extra o alguna relación con otras, Subdirecciones, Entidades Universitarias y Dependencias del Gobierno Federal.

Capítulo II – Levantamiento de información

- 4.- El resultado de los procesos de información es a través de: Reportes, Consulta, Archivos u otros (especifique).
- 5.- ¿Existe algún problema o queja recurrente acerca del sistema utilizado actualmente ?.
- 6.- ¿Cree que existe alguna forma de mejorar el proceso y tramites actuales?.
- 7.- ¿De las funciones que realiza, alguna de ellas no se ha contemplado para este sistema?.
- 8.- Comentarios y propuestas finales de usuarios.

II.6.2.- Documentación por parte de los usuarios para el sistema SIPES.

Justificación :

Debido al desarrollo de una aplicación que utilice la base de datos SISAP como fuente de información para cubrir los requerimientos de las diferentes Subdirecciones, suplicamos sea tan amable de llenar el siguiente formulario.

Subdirección : _____
Departamento: _____
Nombre: _____ Puesto: _____

En el desarrollo de sus actividades: ¿Cuales son las partidas, sistemas, actividades o procesos que tiene asignados con relación a los programas: PRIDE, PEPASIG, FOMDOC, BECAS, PAPIIT, PAPIME, RECANT, EDPAB, EDPAC?

Este cuestionario será modificado en relación con las personas a las que será aplicado, de manera personal a quien tenga participación en el sistema.

II.6.3.- Cuestionario Exhaustivo para Reportes e Información.

El siguiente cuestionario se aplicó a los encargados en cómputo de las partidas que hasta el momento utilizan el sistema A7 para la emisión de reportes y archivos de información sobre las partidas del grupo 300.

1. ¿De los reportes e información que le solicitan están basados en una necesidad? ¿la conoce?, ¿Cuál es su destino?.
- 2.- ¿Cuales son los trámites para realizarlo, quien interviene?.
- 3.- ¿Quién realiza estos tramites? ¿ Que dependencia – área realiza los tramites?
- 4.- ¿Cuánto tiempo toma realizar el proceso (incluyendo los tramites)?.
- 5.- ¿ Alguien más se encuentra relacionado (persona o departamento) en este proceso?.

II.7.- Respuestas obtenidas de los cuestionarios.

II.7.1.- Actividades del encargado en computo de los programas PRIDE, PAIPA, CATEDRA, PEPASIG Y FOMDOC.

El resultado obtenido del cuestionario escrito aplicado al encargado de cómputo, se transcribe para efectos de uso posterior y guía o bien como medio de comparación del sistema actual con la nueva plataforma, las referencias fueron tomadas directamente de los cuestionarios aplicados, por lo tanto, reflejan el conocimiento del encargado sobre cada una de las partidas objeto de estudio.

PRIDE Y PAIPA : De la nómina de percepciones se toman las partidas 355 y 356 (archivo plano, a partir de la quincena 18/97 todos los registros con categoría I21 y D21 pasaron a esta partida con un Nivel "V" y un porcentaje de 150), se generan los archivos por separado, el que contiene PAIPA y otro archivo que contiene PRIDE para las ENEP's y el resto de las dependencias. Se les asignan los porcentajes de acuerdo al nivel que nos indica nómina para hacer los cálculos y emitir los reportes.

El mismo tratamiento se aplica para el programa CATEDRA, ya que contiene el mismo tipo de información la diferencia es la categoría relacionada con los programas PRIDE, PAIPA o CATEDRA.

PEPASIG : La información se sustrae de las Percepciones y es ejecutado por el encargado en cómputo del sistema, con el se obtienen reportes generados por rango y niveles, así como por dependencia específica, por rango y/o niveles según sea solicitado.

FOMDOC : Con respecto a este archivo y a diferencia de los anteriores, DGAPA entrega un diskette conteniendo el archivo, el cual es copiado a la maquina A7 y se emiten 2 reportes, un resumen general y un reporte por dependencia con los totales calculados.

Cabe mencionar que a estos archivos tanto de la DGP como DGAPA, ya tiene nivel y costos incluidos, esto quiere decir que ni el responsable en la Subdirección de Planeación o el responsable en la Subdirección de Cómputo cuenta con los elementos ni antecedentes para decidir si alguien debe entrar a estos programas o si el pago corresponde o no y por lo tanto Dirección General de Personal lo asume como pagado.

Después de recibir este reporte se prosigue a la entrevista personal, a continuación se transcribe el resultado de la misma :

Los participantes en esta entrevista son :

El Jefe de Unidad de la Coordinación de Estudios Presupuestales, es el encargado de turnar a la Subdirección de Cómputo, las solicitudes y en su caso los archivos necesarios para el desarrollo de los procesos, los reportes o archivos requeridos.

El responsable de la subdirección de cómputo es encargado de los siguiente procesos:

1. En el caso de FOMDOC se recibe una copia del archivo proveniente de DGAPA ya procesado, en el no existen políticas que deban aplicarse debido a que se entrega con los datos requeridos para los reportes. Se anexan además las muestras de los reportes que se entregan actualmente a la Subdirección de Planeación.
2. En segundo término se platicó acerca del programa PEPASIG. El cual se obtiene directamente de la nómina de percepciones y corresponde a la partida 351, se procesa en las quincenas pares, de la nómina se obtienen los datos con la clave de cada asignación, así como claves especiales para el pago de honorarios, cuota fija, además se recibió una muestra de los reportes actuales, y la tabla de asignaciones académicas del estímulo.
3. Del programa PRIDE se analizaron, sus características y relación con PAIPA, las políticas que se aplican son diferentes, en el caso de las ENEP's y el resto de las dependencias, esto se ejemplifico por medio de una tabla que se aplica para cada uno de los porcentajes, a partir de los datos de la nómina se aplica dicho porcentaje para los programas PRIDE o PAIPA, a partir de los datos del académico o funcionario se aplica un filtro de programación al archivo fuente, para la asignación quincenal. De esta manera los documentos recibidos fueron: La impresión de los datos de las tablas fuente, tanto de datos como de clasificación de porcentajes y una muestra de los reportes que actualmente se realizan.

Conclusiones obtenidas de la entrevista :

- La entrevista siguió una estrategia previamente establecida sin embargo se aceptaron sugerencias sobre algún tema requerido por el encargado aunque se encuentre fuera del guión principal.
- Identificación de los usuarios que intervienen en el proceso a través de la persona encargada de la Subdirección de cómputo.
- Se obtuvieron diferentes materiales en el procesamiento de datos, como reportes o archivos que actualmente se generan.
- Se recomienda constante comunicación con los encargados para dar el seguimiento adecuado a la generación y metodología de los procesos que se pretende crear con la nueva plataforma.

II.7.2.- Actividades del encargado en Cómputo de los programas EDPAC , CALIDAD Y EFICIENCIA.

EDPAC : Control de seguimiento al pago por Calidad y Eficiencia del Personal Administrativo de Confianza. El tiempo de proceso actual estimado es de 3 a 5 días, la fuente de datos es la DGP, el responsable del programa es el Jefe de Unidad de la Coordinación de Estudios Presupuestales, solicita a la Subdirección de Cómputo los reportes correspondientes a cada quincena del programa de Estímulo al salario(EDPAC).

Proceso General : La Subdirección de Cómputo procesa el archivo de percepciones que entrega la DGP y sustrae el archivo correspondiente a este programa el cual, se cruza con la nómina y los catálogos de la Dirección General de Programación

Capítulo II - Levantamiento de Información

Presupuestal y así obtener los archivos necesarios para la emisión de reportes. Dentro de los archivos se contempla el pago por Calidad y Eficiencia, estos pagos pueden ser actuales o anteriores por lo cual los movimientos que son detectados como anteriores no cuadran con el importe del tabulador y procesar la emisión de reportes. La Subdirección de Planeación revisa y chequea que las cifras de los reportes cuadren con la relación de pagos, enviada por la Dirección General de Personal, si existen cambios o modificaciones para algún reporte se le notifica a la Subdirección de Cómputo.

CALIDAD Y EFICIENCIA : Control del seguimiento al pago de Calidad y Eficiencia del Personal Administrativo de Base. El tiempo de proceso actual estimado es de 2 a 5 días, la fuente original de información es la Nómina de la Dirección General de Personal. El Jefe de Unidad de la Coordinación de Estudios Presupuestales es el encargado de indicar a la Subdirección de Cómputo la quincena con la que se realizará el proceso o bien si hubiera alguna modificación o petición especial de información por parte del Gobierno Federal.

Proceso general : Generación de reportes por categoría del personal Administrativo de Base y administrativo de Confianza, así como resumen por RFC's del personal y turnarlo a la Subdirección de Planeación Presupuestal, la cual revisa y determina si requiere algún otro reporte que solicitar a la Subdirección de Cómputo con la información específica. Cuando la Subdirectora de Planeación autoriza el costo de este programa el responsable solicita a la Subdirección de Cómputo que genere los archivos con la información del mismo, a su vez genera la información autorizada en formato operativo y otros con el formato predefinido por el Gobierno Federal y se envía a la Subdirección de Planeación Presupuestal.

Tras esta serie de entrevistas se han recabado en la siguiente tabla los requerimientos básicos para el sistema.

PARTIDA	SOLICITUD
341 CALIDAD Y EFICIENCIA	<ul style="list-style-type: none">- Reporte de trabajadores por categoría, nivel salarial y estímulo.- Reporte de trabajadores por categoría y nivel salarial resumen general.- Reporte trabajadores por nivel, por estímulo, por dependencia y nivel salarial.
344 EDPAC	<ul style="list-style-type: none">- Reporte de trabajadores por categoría e importe por concepto de gasto.- Reporte de trabajadores por nivel de estímulo y dependencia.- Reporte de trabajadores por categorías EDPAC y nivel de estímulo.
351 DIPASI	<ul style="list-style-type: none">- Reporte de académicos por dependencia.- Reporte de académicos por persona e importe.

Capítulo II - Levantamiento de información

351 FOMDOC	<ul style="list-style-type: none">- Reporte de personas por dependencia.- Reporte de personas por nivel e importe.- Reporte de personas por dependencia y subramos.
351 PEPASIG	<ul style="list-style-type: none">- Reporte de personas por dependencia y nivel.- Reporte de personas por nivel de pago PEPASIG.- Reporte de Personas por horas académicas.
355 PRIDE Y PAIPA	<ul style="list-style-type: none">- Reporte de personas por categoría.- Reporte general por dependencia y subsistema.- Reporte de personas, nivel e importe por subramos.- Reporte de personas Unidades multidisciplinarias (ENEP's) únicamente.- Reporte de personas Resto UNAM (todas excepto ENEP's).- Reporte cuota fija por categoría.
356 CATEDRA	<ul style="list-style-type: none">- Reporte de personas por categoría.- Reporte de personas por nivel y dependencia.- Reporte de personas por subramo de dependencias.-

Además se solicitó que los resultados obtenidos en los reportes se graben en disco para manipularlos en programas alternos tales como EXCEL o WORD.

II.8.- Conclusiones del Levantamiento de Información.

Conclusiones obtenidas de este ejercicio son :

- Reducción de los tiempos de entrega de la información.
- Aumentar el flujo de información hacia los usuarios finales de la Subdirección de Planeación.
- Reestructurar el sistema de trabajo para adecuar a la nueva plataforma tanto para los usuarios como para el encargado en la Subdirección de Cómputo.
- Es muy apreciable la necesidad de comunicación entre los diversos actores en el proceso de generación y administración de la información necesaria para las labores cotidianas.

La ventaja de utilizar un sistema de retroalimentación continua nos permite estar en contacto constante con los usuarios, el analista y diseñador del sistema lo cual permite que aunque se de por terminada esta etapa continua con constantes consultas durante la marcha del sistema.

La información obtenida en el primer paso de este trabajo permite sentar bases sólidas para las siguientes fases las cuales hacen un uso extensivo de estos datos.

Capítulo III

DESCRIPCIÓN DEL SISTEMA Y ANÁLISIS.

III.1.- Objetivo:

Organizar sistemáticamente cada una de las partes que darán forma al SIPES y la manera de interactuar de los módulos propuestos para la solución del problema.

El análisis del sistema consiste en comprender y documentar las necesidades del usuario, este acto implica la motivación del descubrimiento proporcionando un marco de trabajo, en el que el analista puede escribir lo que los usuarios saben y evaluar este proceso de aprendizaje, en este punto el uso de técnicas bien definidas, permite la organización metodológica con tendencias al diseño y posibilita una forma susceptible de medición del avance en el desarrollo del sistema.

III.2.- Plan general.

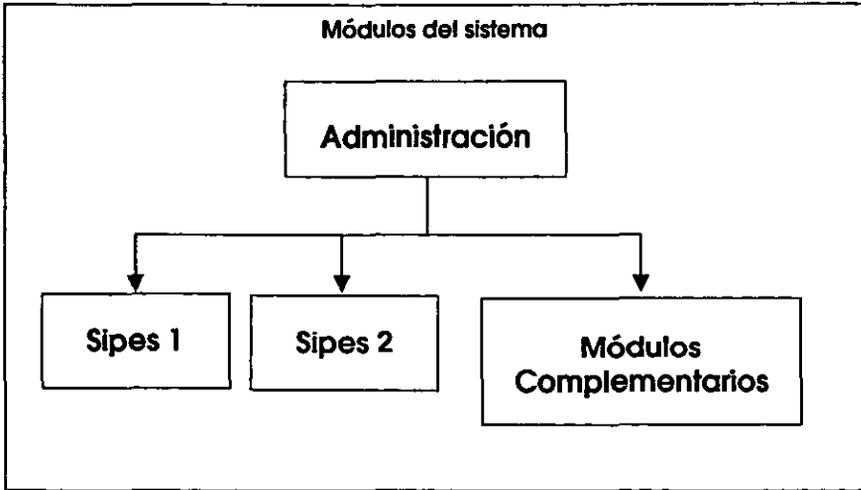
En primer lugar mencionaremos las metas del proyecto que planeamos realizar:

- a) Proporcionar información en forma de reportes o archivos a través de un sistema cliente servidor sobre las diferentes partidas del Presupuesto Universitario involucradas.
- b) Reducir de 4 días a solo medio día el proceso de los datos de nómina en el área de partidas especiales .
- c) Reducir la cantidad de reportes impresos con él consecuente ahorro en papel y tiempo, por medio de resúmenes o archivos.
- d) Permitir la operación directa del sistema por parte de los usuarios.
- e) Automatizar el proceso de las partidas especiales y la generación de reportes quincenales.

La primera fase del desarrollo nos ha permitido distinguir los diferentes componentes que interactúan para la obtención de información por parte del SIPES, por lo tanto una de las primeras decisiones acerca del análisis del sistema, estriba en la virtual separación del estudio de las partidas, esto implica que las soluciones propuestas estarán de acuerdo a cada una de las partidas que se planea manejar en el programa, es decir, se utilizara el mismo método de análisis y desarrollo, pero estos arrojaran diferentes procesos para cada uno de los módulos planteados, esto debido a la naturaleza intrínseca de las partidas, ya que cada una de ellas involucra diferentes tratamientos de los datos derivados de la naturaleza de las partidas, conforme se dé el avance del proceso podremos observar, como para una misma fuente de datos se plantean diferentes solicitudes.

Por lo tanto, el estudio general se tratara al principio de cada tema (en caso de ser necesario) y se harán las debidas anotaciones particulares para cada partida en el momento que se requiera.

En la aproximación general al sistema se han distinguido los siguientes módulos necesarios :



Capítulo III, Figura III.1 Módulos sugeridos del sistema SIPES

El diagrama de la figura III.1 involucra a 3 módulos principales y dos de ellos independientes entre sí, ya que en las entrevistas con los usuarios, se encontró que los Jefes de Unidad de la Subdirección de Planeación encargados de estas partidas, manejan la información de manera independiente de las partidas 351, 355 y 356 así que no debe estar disponibles en la misma aplicación con las partidas 341, 344 por lo tanto el manejo por separado de cada grupo implica la independencia total, esto no se aplica en el programa de administración ya que requiere de un nivel más genérico, en el manejo y flujo de información por lo tanto este módulo sí incluirá ambos grupos de partidas.

Otro de los fundamentos de la división involucra la necesidad de un control estricto del flujo de Información, mientras que los usuarios poseen módulos independientes entre sí, el administrador de la información puede preparar los datos necesarios con antelación reduciendo de manera significativa los tiempos de entrega, proceso y uso de la red.

De esta manera el diagrama muestra la forma en que se relacionan los módulos SIPES 1 y SIPES 2, los cuales son independientes entre sí y a su vez son atendidos por el módulo de administración del cual cubren sus necesidades de información.

Esta división permite además un control extra, sobre el flujo de la información lo cual reditúa en una facilidad de mantenimiento, ya que al tener necesidad de corregir un problema o hacer alguna actualización sólo un módulo a la vez es modificado y afectado (el único caso donde esto se aplica con restricción es cuando el módulo a modificar sea el de administración, ya que ambos módulos se verían afectados, sin embargo la división en modulos mas especificos reduciría al mínimo la afectación).

Otra de las situaciones que se presentan es el estado actual de la información por un lado existe el esquema anterior con archivos planos en un mainframe A7, la cual se pretende superar en el esquema actual con una base de datos relacional, mucho más poderosa y flexible en el manejo de la información

Los módulos complementarios implican reportes o solicitudes extras de información, que no involucran necesariamente a los dos módulos de operación propuestos y sin embargo, si complementan el trabajo de ambos, por lo tanto son considerados como parte del sistema en general.

Tras la separación en módulos funcionales del sistema, damos paso al análisis particular de cada uno de ellos, los cuales involucran a las diferentes partidas objeto del estudio.

III.2.1.- El módulo de administración.

Este se refiere a todas las partidas sin distinción, esta encargado de administrar, verificar y preparar los datos para la correcta operación del SIPES, consiste de una serie de rutinas para el control de los datos de las partidas, que verifican la validez de la información que se encuentra en la nómina y prepara la información de las tablas para obtener los reportes requeridos según las necesidades del usuario.

III.2.2.- El módulo SIPES I.

La división en SIPES I y II se debe a las características de las partidas, planteadas en el capítulo I y II, obedecen principalmente a la manera en que opera con los datos en la Subdirección de Planeación de la Dirección General de Programación Presupuestal, los responsables directos de las partidas se encargan cada uno y de manera independiente de los datos de las mismas por lo tanto se ha optado manejarlas por separado, dando continuidad a esta independencia.

El módulo I consta de apartados para las siguientes partidas 351 (DIPASI, PEPASIG, FOMDOC) , 355 (PRIDE y PAIPA) , 356 (CATEDRA) y no requiere del uso de SIPES II para su correcta operación.

III.2.3.- El módulo SIPES II.

Este módulo se encarga de la operación de datos de las partidas 341 (Calidad y Eficiencia) y 344 (EDPAC) las cuales son operadas de manera independiente, en otra área de la Subdirección de Planeación, otro motivo de separación de módulos es mantener la seguridad en la operación de los datos en módulos separados, de esta forma solo las unidades responsables de la operación, tendrán acceso a ellos, con esto se pretende evitar fugas de información, accesos no autorizados a los datos y promover la encapsulación, de manera que sólo las personas autorizadas por la Subdirección de Planeación tengan acceso al sistema. Otra de las situaciones que se presenta es la independencia de datos y el modo de operación en cada partida, por ser diferente la información que se maneja esta separación implica una operación lógica de los módulos respecto a la información y las personas que los operan.

III.2.4.- Los módulos complementarios.

Conforme se ha desarrollado el sistema una de las situaciones que ha prevalecido, es el constante crecimiento de sus partes y las reiteradas solicitudes de requerimientos de la Subdirección de Planeación. En algunas ocasiones este manejo de nuevas solicitudes necesita la implementación de módulos nuevos (por ejemplo: reportes de totales, peticiones de la Secretaría de Hacienda) la decisión de apoyar estas nuevas solicitudes fue la de crear módulos complementarios para evitar la constante modificación de los módulos principales y permitir que el usuario posea un flujo constante de información y al desarrollador le plantea un esquema completo de actualización de los módulos sin perder ni aumentar el tiempo de respuesta del sistema al hacer modificaciones de los módulos principales, esta característica se observará mas adelante en el capítulo V.

Cómo observamos existe una serie de razones físicas y lógicas de la división modular planteada como solución, entre las desventajas de la misma encontramos:

- a) La duplicación en algunas situaciones específicas de los procesos entre los módulos SIPES I y II.
- b) La complejidad en la operación del sistema.
- c) La independencia de los procesos no permite compartir código eficientemente entre módulos.
- d) Aumento de trabajo que implica el desarrollo independiente para cada módulo.

Entre las ventajas que podemos destacar se encuentran:

- a) La posibilidad de independencia de módulos que fomenta un manejo ordenado de la información solo del personal autorizado.
- b) Facilitar al usuario la información que acostumbra.
- c) Permitir el manejo adecuado de los módulos ya que separa físicamente unas operaciones de otras y los ejecutables del sistema, permanecen en tamaños manejables.
- d) Agrega unidad lógica a los módulos en desarrollo, ya que permite el manejo independiente de procesos.
- e) Aumenta la seguridad al definirse específicamente que personas tienen acceso a determinados procesos de cada módulo.
- f) Requiere de menor tiempo de aprendizaje por parte del usuario ya que restringe la cantidad de procesos que debe realizar.

III.3.- Modelo de contexto.

El modelo de contexto es un mecanismo de exploración, de la expansión y reducción del alcance del sistema, que funciona como fuente de información del estado actual, de la organización y relaciones con sistemas existentes. Estos procesos de exploración se convierten en el máximo valor del modelo de contexto y se refleja gráficamente a través del diagrama de flujo de datos que plantea como fluye la información y entidades que lo utilizan o se ven afectadas por los mismos.

El modelo del contexto se encuentra formado por un círculo o conjunto de ellos, que representan el área de estudio, los agentes externos que están fuera de control del sistema se representan usando rectángulos. Estos agentes externos envían flujos de estímulos al sistema o reciben flujos de respuesta, cada objeto del diagrama tiene una descripción escrita.

El diagrama de flujo de datos fue utilizado por primera vez en 1979 por Tom DeMarco, representan la ruta que siguen los datos a través de una organización, no dan tendencias a una implantación específica de los sistemas, solo muestran una tendencia general de la manera en como se trabaja con los datos y organiza su desarrollo posterior en alguna plataforma específica.

La notación de los diagramas de flujo de datos es muy simple, aunque implica un principio de caja negra donde los datos de entrada y de salida sufren transformaciones, que no se describen en el diagrama, sin embargo, se identifican todas las partes que constituyen dicho cambio.

El diagrama debe enunciar claramente las entidades y los flujos de datos entre ellas. Para completar el diagrama deben rotularse claramente cada uno de los agentes, flujos y procesos involucrados.

- a) **El proceso** : Representa módulos de transformación de la información entre procesos a través del paso del tiempo.
- b) **Flujo de datos**: Señala la relación entre procesos, agentes y almacenes de información a manera de estímulos y respuestas, en forma de datos para todos los componentes restantes del diagrama.
- c) **Agentes externos**: Son entidades que aunque forman parte del proceso son mas bien estáticas o independientes del proceso, llevado a cabo y sin embargo son indispensables para su consecución.
- d) **Almacenes de datos**: Son los lugares donde se encuentran los datos cuando no se utilizan y complementan o son fuente de los flujos de datos existentes en el sistema.

Una vez que obtenemos el diagrama de flujo de datos el siguiente paso es obtener el diagrama de contexto que utiliza los mismos principios de desarrollo, pero implica una delimitación de área de aplicación de cada uno de los módulos del sistema.

De manera general obtendremos un diagrama de contexto del sistema que deberá englobar la participación de cada una de las entidades y procesos que resuelven

el área de interés del sistema. Estas relaciones servirán como guía de fronteras del sistema estableciendo perfectamente sus límites y áreas de aplicación.

Con la información recabada hasta el momento se presenta el Diagrama de Contexto para el sistema SIPES (Sistema para el control de Información de las Partidas Especiales).

El Diagrama de Contexto (figura III.2) de la aplicación consta de las siguientes componentes :

El SIPES da alternativas de solución de acuerdo a los requerimientos y necesidades de información de la Subdirección de Planeación, en forma de archivos y reportes.

Esta Subdirección notifica las necesidades a la Subdirección de Cómputo con la finalidad de obtener apoyo en el seguimiento del proceso.

La Subdirección de Cómputo determina las políticas para ofrecer una solución eficiente a los requerimientos de información de la Subdirección de Planeación.

En caso de que los requerimientos planteados estén fuera, del alcance original del SIPES la Subdirección de Cómputo los analiza para dar mantenimiento a SIPES y en su caso a la Base de datos SISAP.

La Base de Datos SISAP provee de información a SIPES y cubre con esto las necesidades de información de la Subdirección de Planeación.

En caso de que la Subdirección de Planeación utilice directamente el SIPES para la generación de archivos y reportes lo cual permite eliminar la dependencia de la Subdirección de Cómputo en los procesos cotidianos.

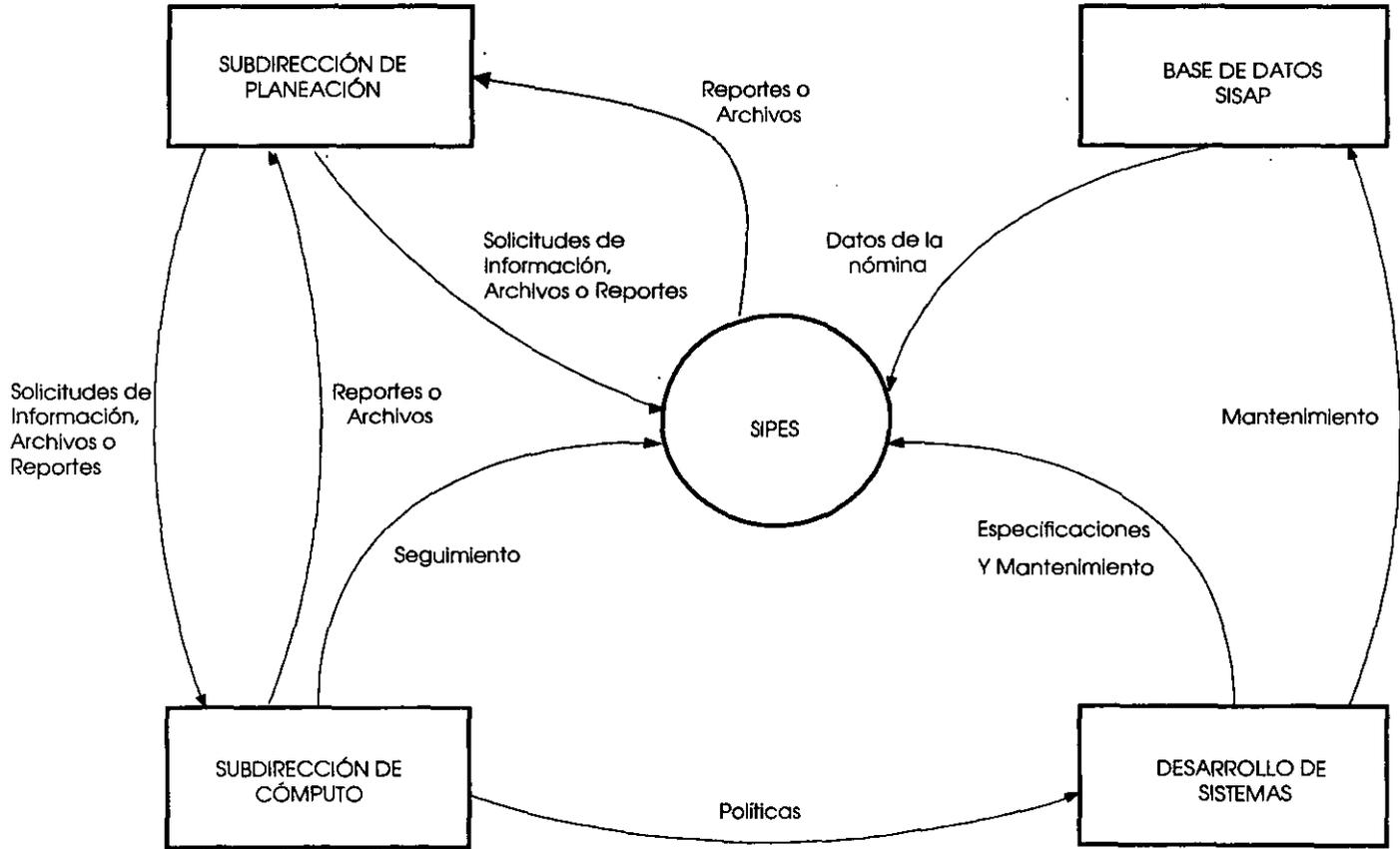
En el capítulo II mencionamos la dificultad de operar las diferentes partidas del sistema como una sola, debido a su naturaleza independiente unas de otras, esto desemboca en el manejo independiente de las constantes de cada una de ellas, por lo tanto cada partida tendrá su apartado de contexto y correspondiente diagrama de flujo de datos.

III.3.1.- SIPES I - Partida 351 Estimulos a la productividad y al rendimiento del personal académico de asignatura y de fomento a la docencia para el personal de carrera.

III.3.1.1.- DIPASI.

El estudio de esta partida se divide en diferentes programas cada uno con sus características particulares, el DIPASI es un programa que se aplica a los profesores de asignatura, por lo tanto es un programa únicamente académico, la asignación de recursos la realiza mensualmente la Dirección General de Personal a través de la nómina quincenal esta información junto con los catálogos de la base de datos SISAP nos permite distinguir los registros afectados, en especial en quincenas pares de las personas inscritas en el programa. Después de concentrar los datos y validar la información ocurren dos situaciones basadas en las solicitudes de la Subdirección de Planeación , en primer lugar la administración de los datos obtenidos y en segundo lugar la solicitud expresa de la

MODULO : SIPES I y II
DIAGRAMA : DE CONTEXTO
DEPARTAMENTO : DESARROLLO DE SISTEMAS
FECHA : 10 DE JUNIO DE 1998



Capítulo III, Figura III.2

MODELO DE CONTEXTO

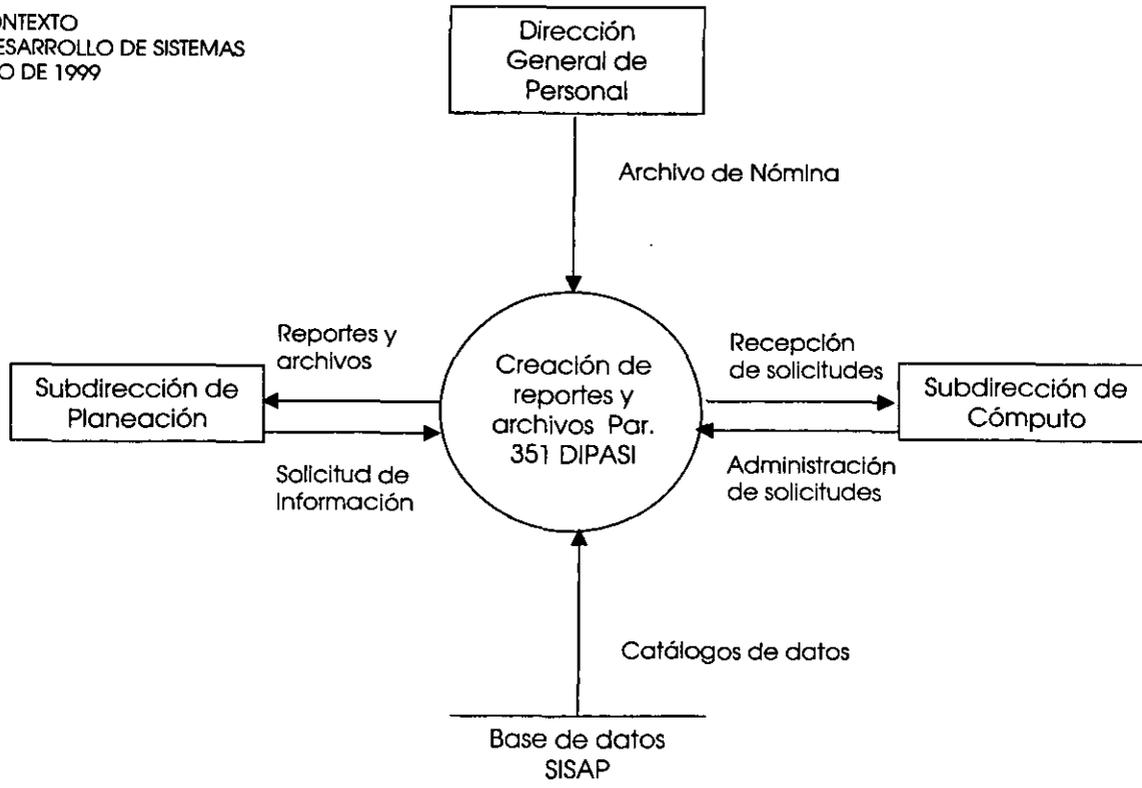
PARTIDA : 351 – DIPASI

MÓDULO : SIPES I

DIAGRAMA : DE CONTEXTO

DEPARTAMENTO : DESARROLLO DE SISTEMAS

FECHA : 15 DE ENERO DE 1999



Capítulo III, Figura III.3

MODELO DE CONTEXTO

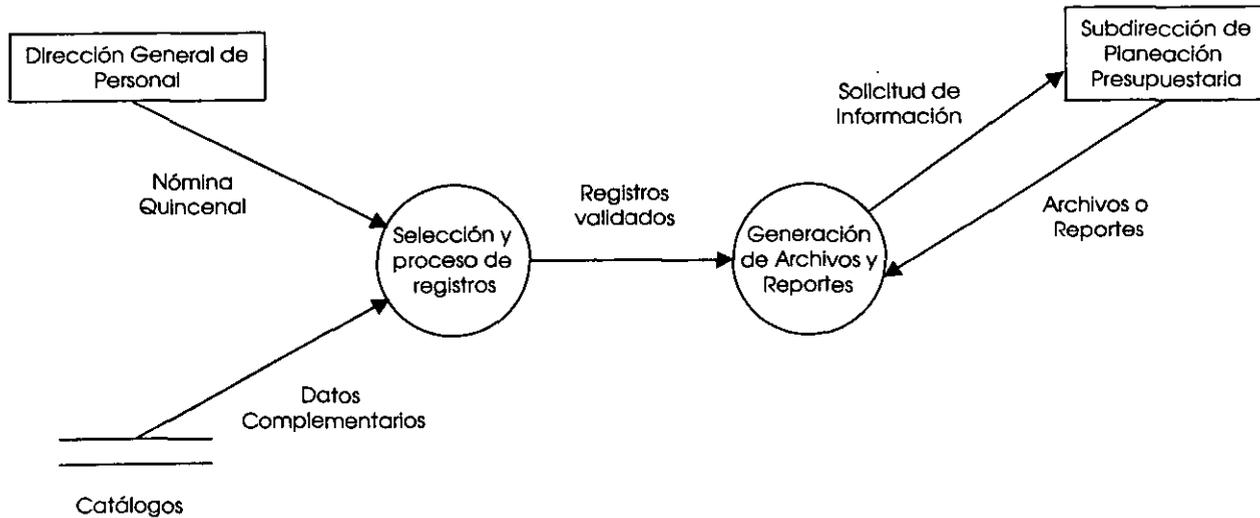
PARTIDA : 351 – DIPASI

MÓDULO : SIPES I

DIAGRAMA : FLUJO DE DATOS

DEPARTAMENTO : DESARROLLO DE SISTEMAS

FECHA : 15 DE ENERO DE 1999



Capítulo III, Figura III.4

Subdirección, la cual implica la generación de reportes o archivos que cumplen en específico con la solicitud realizada.

En el diagrama de contexto de las figuras III.3 y III.4 se plantea lo siguiente:

En él observamos a los principales participantes del proceso las Subdirecciones de Cómputo y Planeación de la Dirección General de Programación Presupuestal, las solicitudes directas de la primera desembocan en la consecuente administración, de la información de la segunda y esta información a su vez se convierte en los reportes o archivos solicitados.

Podemos observar el trabajo conjunto y bien delimitado de los procesos propuestos por un lado en el módulo de administración a cargo de la Subdirección de Cómputo y por el otro la operación por parte de la subdirección de Planeación, este patrón es constante en el resto de las partidas, aunque sufre variaciones sustanciales en el ámbito de los datos conforme cambian los programas y partidas en estudio.

III.3.1.2.- FOMDOC.

En el caso de este programa Fomento a la Docencia, nuevamente las solicitudes de información corren a cargo de la Subdirección de Planeación, en primer lugar se debe recibir un archivo externo de la Dirección General de Asuntos del Personal Académico que contiene la información de los académicos. Este archivo se recibe mensualmente, debe complementarse necesariamente con los datos contenidos en la base de datos SISAP, debido a que sólo contiene información sobre el importe del pago y el código programático afectado. Una vez completa y validada la información se pasa a la generación de reportes y archivos delimitando de esta manera el contexto del sistema.

Las entidades participantes en este módulo (fig III.5 y III.6) son las siguientes:

La Subdirección de Planeación con las solicitudes de información mensual, la Dirección General de Asuntos del Personal Académico que proporciona el archivo inicial de proceso de los datos y la Subdirección de Cómputo, a cargo de la administración de la información y su preparación para la generación de reportes o archivos.

III.3.1.3.- PEPASIG.

El origen del proceso de la información de PEPASIG parte de la nómina de la Dirección General de Personal los datos de la partida, quincenalmente son procesados y centralizados con la información de la base de datos SISAP para obtener una serie de registros validados y de totales que permiten obtener la consecuente información impresa o en archivo, según sea solicitado por la Subdirección de Planeación.

En el caso del diagrama de contexto (figs. III.7 y III.8) las solicitudes directas de la Subdirección de Planeación son atendidas directamente por la Subdirección de Cómputo apoyándose en los datos de nómina de la Dirección General de Personal para la generación de los reportes o archivos específicos.

MODELO DE CONTEXTO

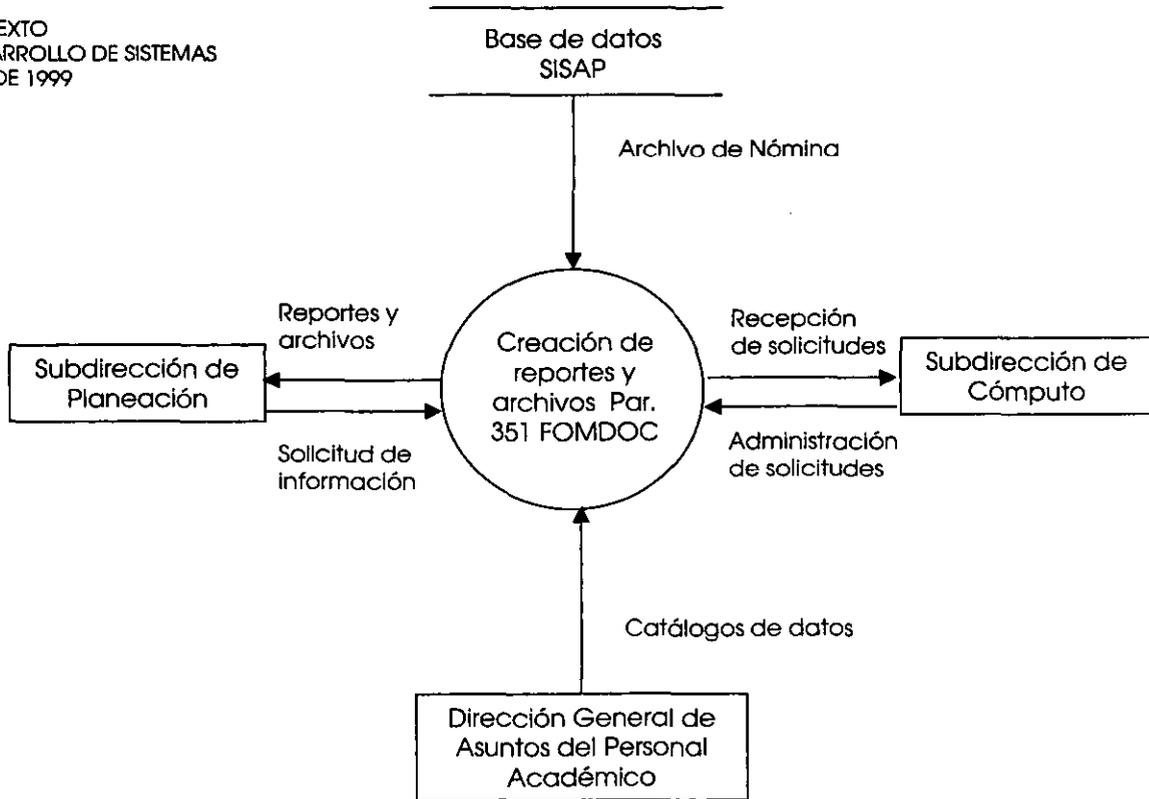
PARTIDA : 351 - FOMDOC

MÓDULO : SIPES I

DIAGRAMA : DE CONTEXTO

DEPARTAMENTO : DESARROLLO DE SISTEMAS

FECHA : 15 DE ENERO DE 1999



Capítulo III, Figura III.5

MODELO DE CONTEXTO

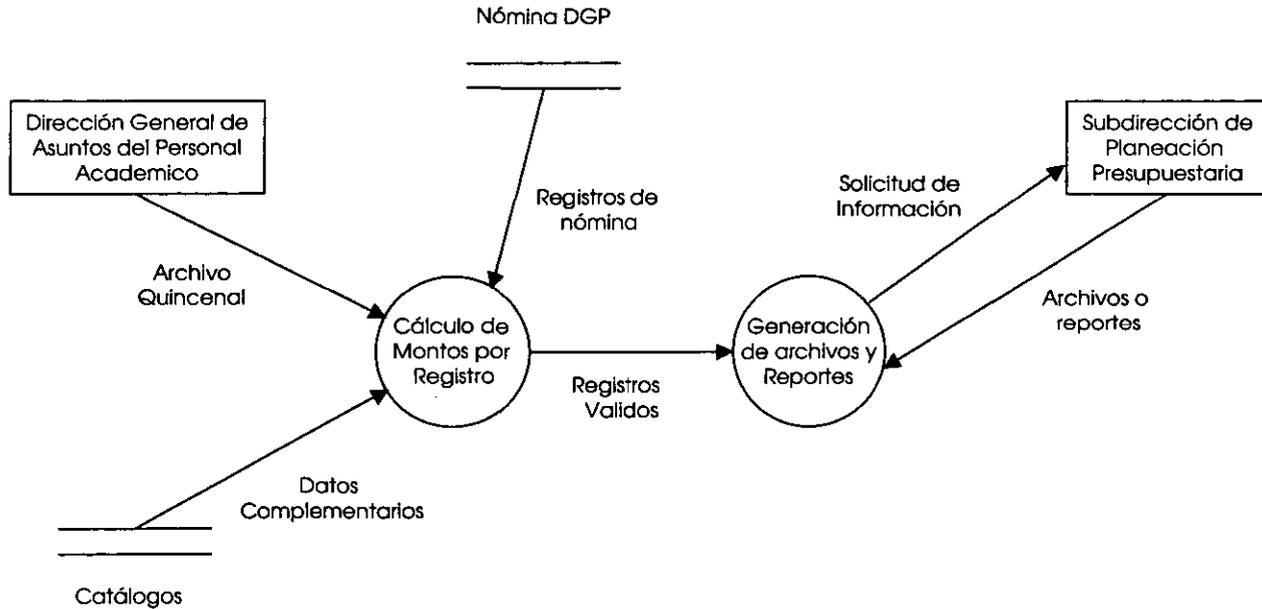
PARTIDA : 351 - FOMDOC

MÓDULO : SIPES I

DIAGRAMA : FLUJO DE DATOS

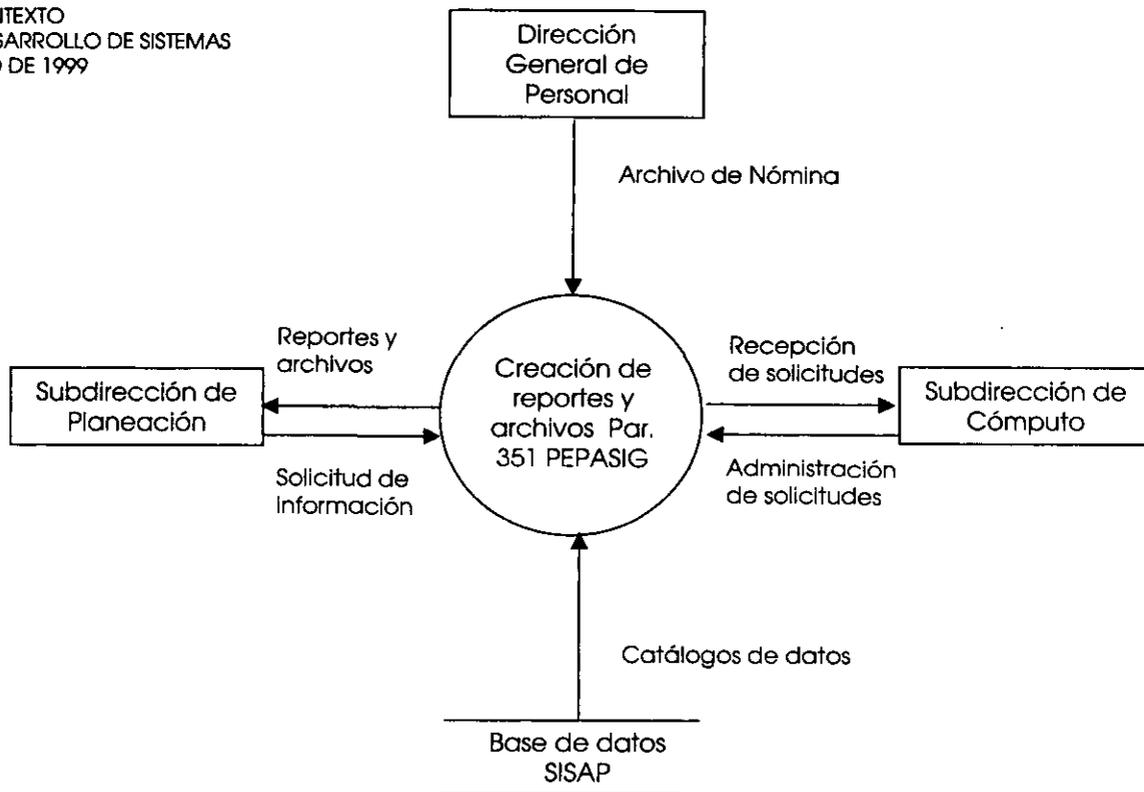
DEPARTAMENTO : DESARROLLO DE SISTEMAS

FECHA : 15 DE ENERO DE 1999



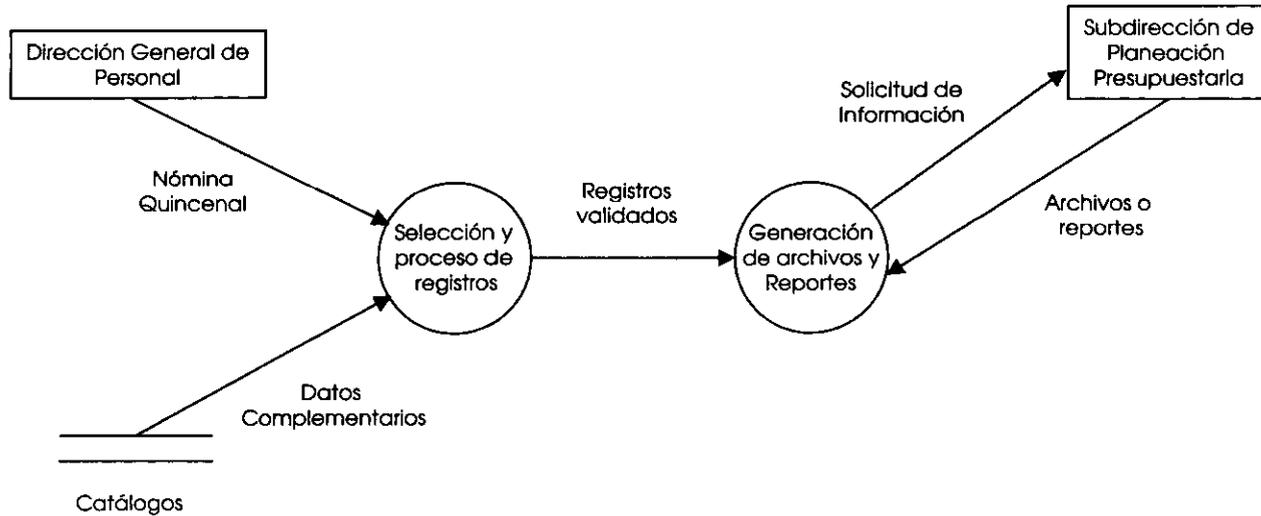
Capítulo III, Figura III.6

MODELO DE CONTEXTO
PARTIDA : 351 - PEPASIG
MÓDULO : SIPES I
DIAGRAMA : DE CONTEXTO
DEPARTAMENTO : DESARROLLO DE SISTEMAS
FECHA : 15 DE ENERO DE 1999



Capítulo III, Figura III.7

MODELO DE CONTEXTO
PARTIDA : 351- PEPASIG
MÓDULO : SIPES I
DIAGRAMA : FLUJO DE DATOS
DEPARTAMENTO : DESARROLLO DE SISTEMAS
FECHA : 15 DE ENERO DE 1999



Capítulo III, Figura III.8

III.3.2.- SIPES I – Partida 355 Prima de desempeño al personal Académico.

III.3.2.1.- PRIDE Y PAIPA.

La partida 355 se divide en 2 programas principalmente pero a diferencia de la partida 351 PEPASIG el tratamiento de la información es el mismo y cumple con las mismas características en ambos casos. Por lo tanto se decidió manejarlos en el mismo proceso y contexto.

Esto se debe a que se trata de programas con aplicación en la misma partida y de los mismos principios de proceso, la información inclusive se ha solicitado que ser manejada en conjunto como PRIDE, sin embargo existe un caso especial en cuanto al diseño ahí observaremos la necesidad de dividir en algunas situaciones los datos por un lado PRIDE y por otro de PAIPA, aunque en este nivel esa diferencia no se refleja implícitamente.

En las figuras III.9 y III.10 se observa que la principal fuente de datos es el archivo de nómina de la Dirección General de Personal, el cual contiene los registros de ambos programas que aunados a los catálogos de la base de datos SISAP nos permite elegir los registros adecuados para el proceso, una vez seleccionados estos registros se procesan para generar la información que sirve de fuente a los reportes y archivos solicitados, en especial se debe hacer una separación y validación de los datos por registro, lo cual permite un manejo de ambos programas en un mismo proceso de administración.

En el caso del Diagrama de contexto tenemos:

A partir de las solicitudes de información de la Subdirección de Planeación el proceso de entrega de reportes se basa en la información enviada a través de nomina de la Dirección General de Personal y el uso de los catálogos contenidos en el SISAP. Tras la validación y preparación de los datos a través de la administración se responde a los requerimientos iniciales.

III.3.3.- SIPES I – Partida 356 Reconocimiento catedrático UNAM.

III.3.3.1.- CATEDRA.

El proceso de la partida 356 es muy parecido al de la partida 355 PRIDE, pero a diferencia del mismo las condiciones varían en el ámbito de datos y su proceso.

Como se observa en el Diagrama de Flujo de Datos (figuras III.11 y III.12), el proceso es muy similar, sin embargo, la información debe cumplir con otros parámetros de solicitud por parte de la Subdirección de Planeación, la cual genera las solicitudes. Tras cubrir las con el proceso proveniente de la Dirección General de Personal y los catálogos de datos de SISAP para la generación de reportes y archivos que se requieren.

En el diagrama de contexto, el límite de trabajo se encuentra perfectamente delimitado a las solicitudes de información que son emitidas por la Subdirección de Planeación y recibidas por la Subdirección de Cómputo tras administrar la información a

MODELO DE CONTEXTO

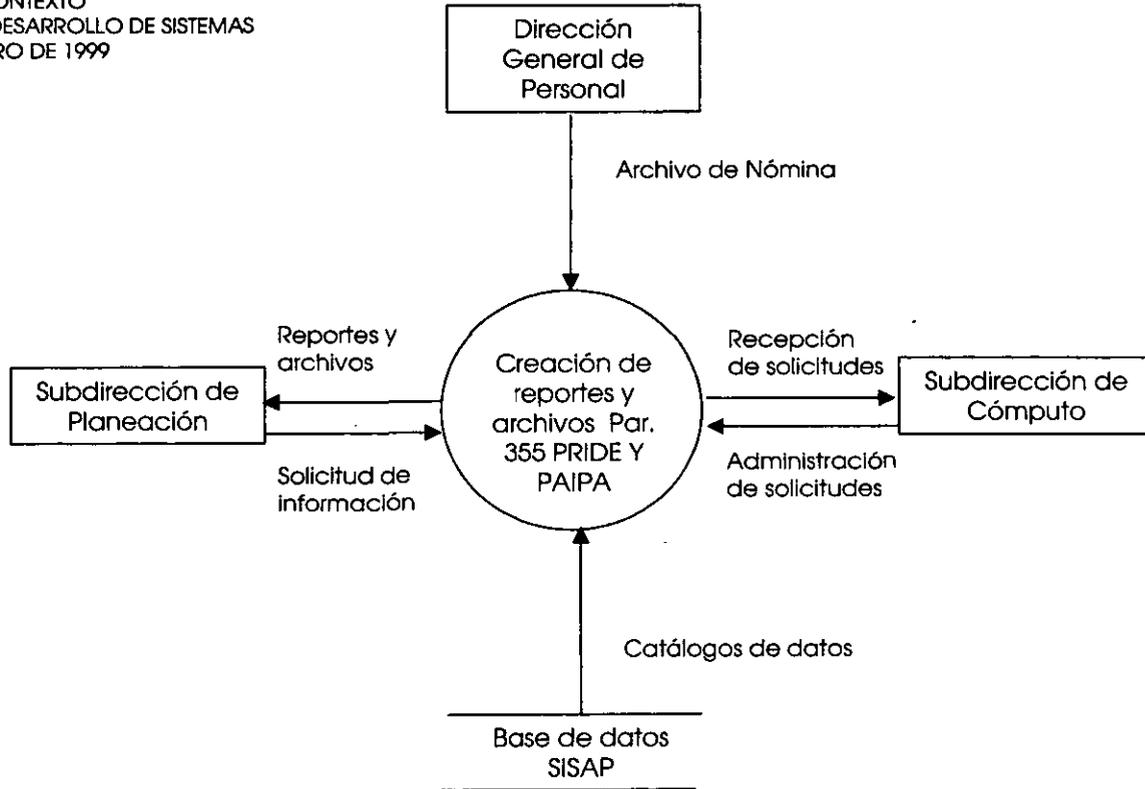
PARTIDA : 355 - PRIDE Y PAIPA

MÓDULO : SIPES I

DIAGRAMA : DE CONTEXTO

DEPARTAMENTO : DESARROLLO DE SISTEMAS

FECHA : 15 DE ENERO DE 1999



Capítulo III, Figura III.9

MODELO DE CONTEXTO

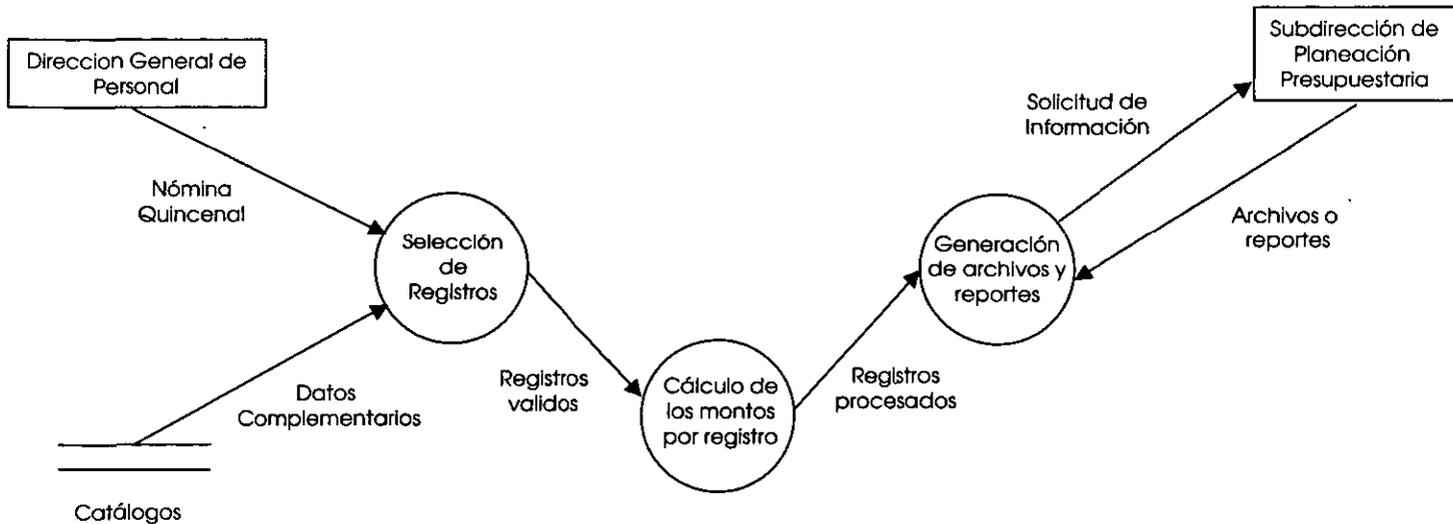
PARTIDA : 355 - PRIDE Y PAIPA

MÓDULO : SIPES I

DIAGRAMA : FLUJO DE DATOS

DEPARTAMENTO : DESARROLLO DE SISTEMAS

FECHA : 15 DE ENERO DE 1999



Capítulo III, Figura III.10

MODELO DE CONTEXTO

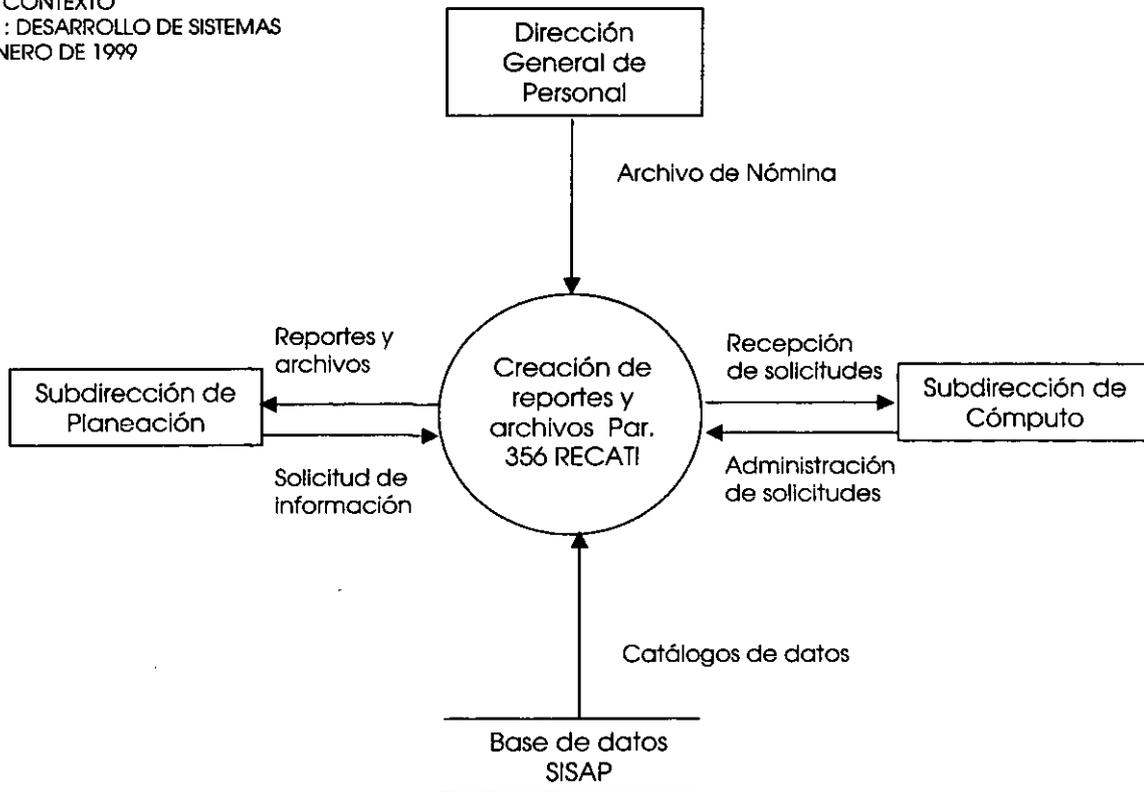
PARTIDA : 356 - RECATI

MÓDULO : SIPES I

DIAGRAMA : DE CONTEXTO

DEPARTAMENTO : DESARROLLO DE SISTEMAS

FECHA : 15 DE ENERO DE 1999



Capítulo III, Figura III.11

MODELO DE CONTEXTO

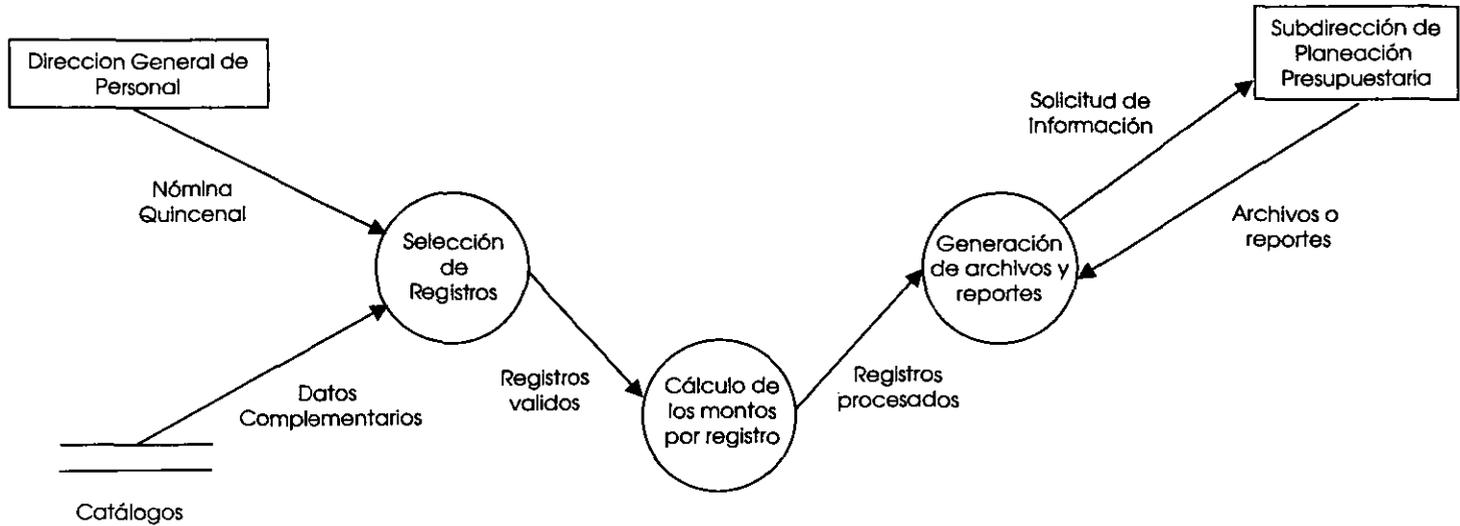
PARTIDA : 356- RECATI

MÓDULO : SIPES I

DIAGRAMA : FLUJO DE DATOS

DEPARTAMENTO : DESARROLLO DE SISTEMAS

FECHA : 15 DE ENERO DE 1999



Capítulo III, Figura III.12

partir de los catálogos de SISAP y la información de la nómina de la Dirección General de Personal se preparan los datos para la posterior emisión de los reportes.

III.3.4.- SIPES II - Partida 341 Complemento al salario por calidad y eficiencia en el trabajo del personal administrativo.

III.3.4.1.- Calidad y Eficiencia.

En este caso la partida tiene como fuente de información la Nómina quincenal emitida por la Dirección General de Personal, en la cual se incluyen los datos de los empleados que se requieren para el proceso de la partida, además se utilizan los catálogos de SISAP con la información complementarla que permite completar y validar los datos para ser utilizados por medio del SIPES 2 por la Subdirección de Planeación y generar los reportes o archivos requeridos.

El escenario de contexto (figuras III.13 y III.14) se repite conforme a las anteriores partidas aunque cabe resaltar que este módulo pertenece a SIPES 2 debido a la división logística que se comento al inicio del capítulo.

III.3.5.- SIPES II - Partida 344 Estimulos y gratificaciones complementarias.

III.3.5.1.- EDPAC.

En este caso solo mencionaremos las diferencias con la partida anterior con el ánimo de no ser reiterativo. En primer lugar los datos son de la misma fuente la Dirección General de Personal, el proceso de la información se cambia en las bases del programa y se refleja en los diferentes módulos, las solicitudes son generadas por la Subdirección de Planeación y cubiertas por la Subdirección de Cómputo de manera muy similar a la partida 341.

En el diagrama de contexto ilustrado en las figuras III.15 y III.16, observaremos los límites señalados con anterioridad, los cuales no son diferentes entre sí pero con la diferencia básica de que aplica a diferentes partidas y programas en cada una de ellas.

MODELO DE CONTEXTO

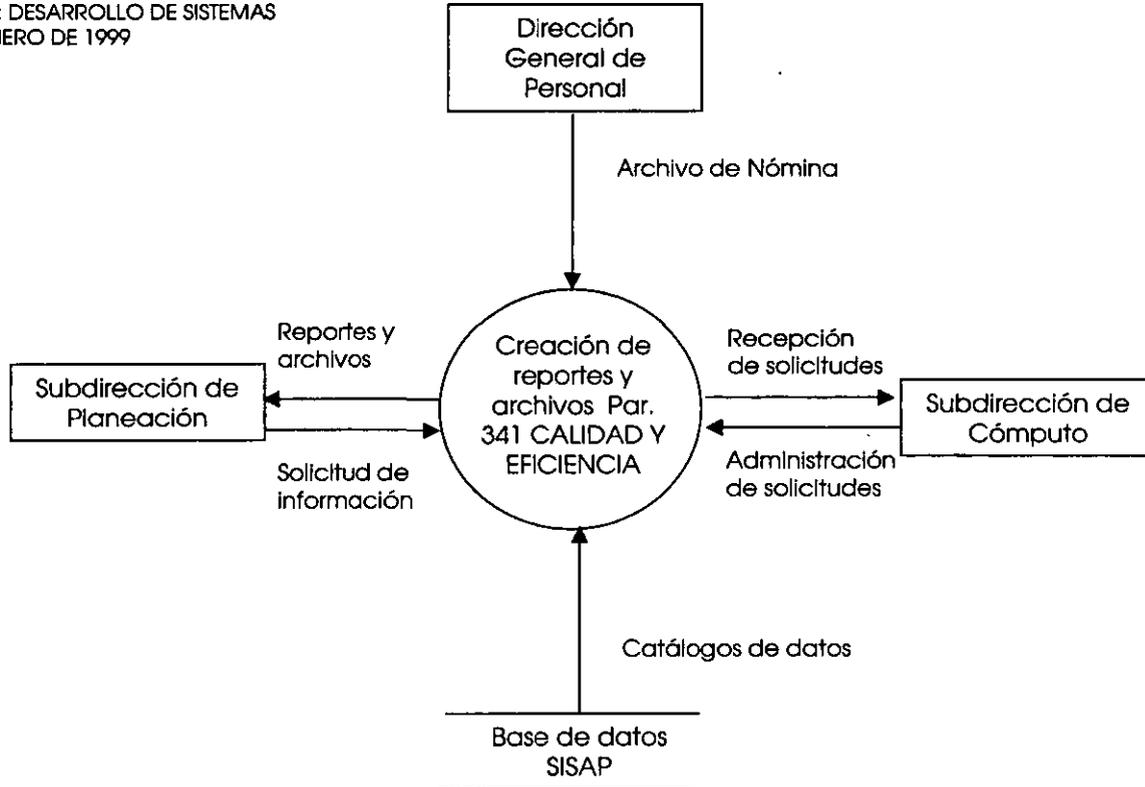
PARTIDA : 341 - CALIDAD Y EFICIENCIA ADMVO.

MÓDULO : SIPES II

DIAGRAMA : DE CONTEXTO

DEPARTAMENTO : DESARROLLO DE SISTEMAS

FECHA : 15 DE ENERO DE 1999



Capítulo III, Figura III.13

MODELO DE CONTEXTO

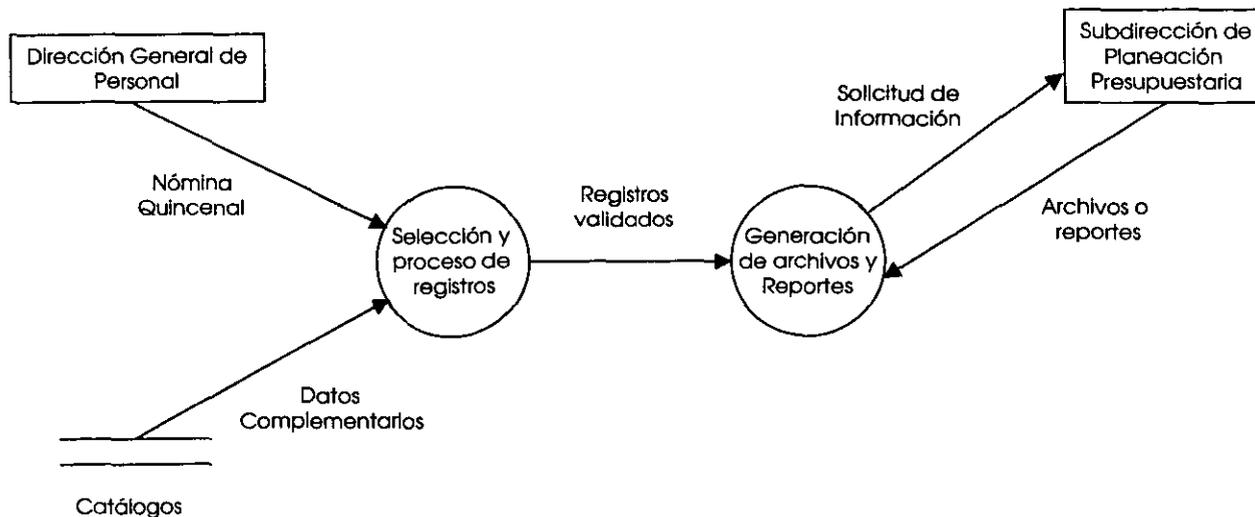
PARTIDA : 341 - CALIDAD Y EFICIENCIA ADMINISTRATIVO

MÓDULO : SIPES II

DIAGRAMA : FLUJO DE DATOS

DEPARTAMENTO : DESARROLLO DE SISTEMAS

FECHA : 15 DE ENERO DE 1999



Capítulo III, Figura III.14

MODELO DE CONTEXTO

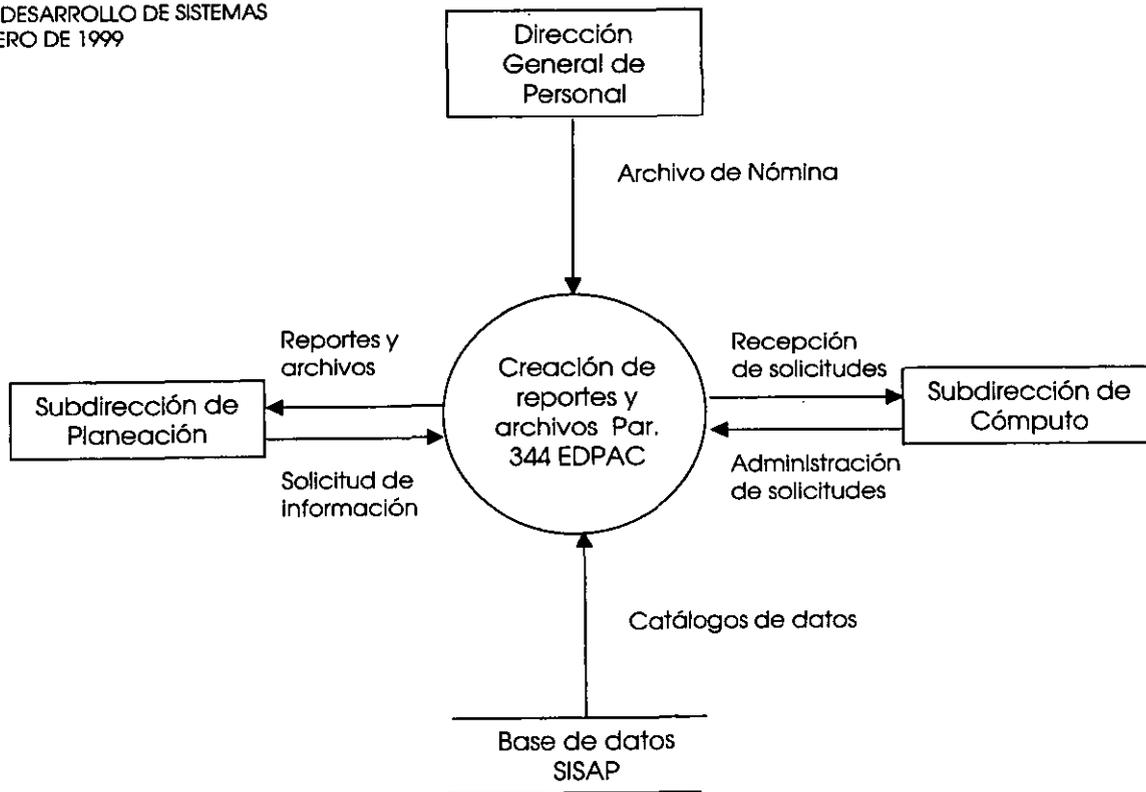
PARTIDA : 344 - EDPAC

MÓDULO : SIPES II

DIAGRAMA : DE CONTEXTO

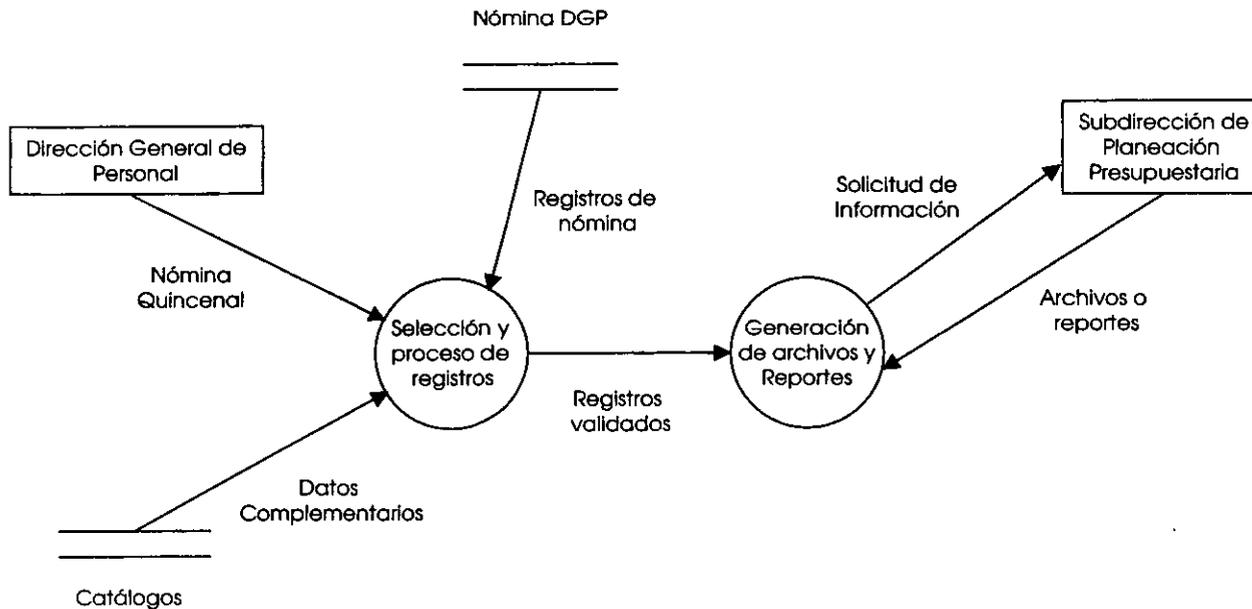
DEPARTAMENTO : DESARROLLO DE SISTEMAS

FECHA : 15 DE ENERO DE 1999



Capítulo III, Figura III.15

MODELO DE CONTEXTO
PARTIDA : 344 - EDPAC
MÓDULO : SIPES II
DIAGRAMA : FLUJO DE DATOS
DEPARTAMENTO : DESARROLLO DE SISTEMAS
FECHA : 15 DE ENERO DE 1999



Capítulo III, Figura III.16

III.4.- Modelo de eventos.

Este modelo es la parte que permite a los encargados del sistema responder a los sucesos ocurridos en el mundo real, es decir, abstrae las situaciones que permiten al programa reaccionar ante los sucesos que puedan ocurrir por parte de los usuarios durante el tiempo o ante valores específicos de los datos.

Los eventos son situaciones que ocurren durante el tiempo, cosas que suceden y que disparan o generan solicitudes al programa.

Los eventos pueden ser esperados o inesperados dependiendo de su ocurrencia dentro de la ejecución del sistema o peticiones de información.

La manera en que se implementa el modelo de eventos, es a través de una lista y el diccionario de eventos, la primera es una herramienta que permite la identificación y la segunda una descripción de los eventos que ocurren durante la ejecución del programa.

El modelo de eventos indicará posteriormente a los programadores el orden de espacio o tiempo en que se deben ejecutar los procesos, es importante distinguir los eventos y su relación con las entidades que ya hemos delimitado en el modelo de contexto.

Las técnicas propuestas se basan en el principio de la caja negra es decir los eventos producen como resultado una acción, sin embargo, los procesos en sí no se encuentran definidos aún, esto permite dar un segundo paso al análisis, el primero fue principalmente delimitar el ámbito del sistema, el segundo paso consiste en establecer el tiempo/lugar las actividades que se solicitan.

El modelo de eventos es el enlace entre los diferentes componentes de diseño y desarrollo del sistema, ya que involucra la capacidad del sistema de información de responder a los sucesos del mundo real.

III.4.1.- Los elementos del modelo de eventos.

La lista de eventos: Es una lista de formato sujeto-verbo-objeto que describe las acciones que durante el tiempo ocurren en la ejecución del programa.

El diccionario de eventos: se basa en la lista de eventos para especificar cada proceso en el sistema, se define también el significado en texto claro y conciso, e incluye una serie de datos de estímulo (o entradas) requeridos para activar el evento, a su vez involucra la actividad realizada a partir de estos datos y su correspondiente respuesta o salida de manera escrita lo que da forma completa al evento es decir describe todas las características deseadas.

El diccionario de eventos consta de las siguientes partes:

El identificador: Permite identificar a un evento específico dentro del sistema, no posee ningún atributo especial y no es recomendable darle un sentido significativo para facilitar la modificación durante el desarrollo del modelo.

El nombre del evento: Se toma directamente de las listas de eventos con un formato sujeto-verbo-objeto.

La descripción: Describe claramente las políticas del negocio para el evento.

El estímulo: Es el flujo de datos de entrada que activa el evento.

La actividad: Es el procesamiento realizado por el sistema a partir de los datos obtenidos como estímulo.

La respuesta: Es el flujo de datos obtenido al concluir la actividad realizada en el sistema.

El efecto: Es la manera en la que el flujo de datos afecta al ambiente que rodea al sistema en acciones y decisiones por tomar con la respuesta obtenida.

III.4.2.- Lista de eventos.

En la siguiente tabla se encuentran las listas de eventos principales del sistema:

PARTIDA / PROGRAMA	EVENTOS
351 DIPASI	110 La Subdirección de Planeación hace una solicitud de información. 120 La Subdirección de Cómputo carga nómina. 130 La Subdirección de Cómputo procesa datos de nómina. 140 La Subdirección de Planeación emite reportes o archivos.
351 FOMDOC	210 La Subdirección de Planeación hace una solicitud de información. 220 La Subdirección de Cómputo ingresa los datos de FOMDOC y de nómina. 230 La Subdirección de Cómputo procesa datos de FOMDOC y notifica a la Subdirección de Planeación el estado de la información. 240 La Subdirección de Planeación emite reportes o archivos.

Capítulo III - Descripción del sistema y análisis

<p>351 PEPASIG</p>	<p>310 La Subdirección de Planeación solicita información a la Subdirección de Cómputo. 320 La Subdirección de Cómputo carga nómina. 330 La Subdirección de Cómputo procesa y notifica estado de los datos de la nómina. 340 La Subdirección de Planeación emite reportes o archivos.</p>
<p>355 PRIDE Y PAIPA</p>	<p>410 La Subdirección de Planeación solicita información a la Subdirección de Cómputo. 420 La Subdirección de Cómputo carga nómina. 430 La Subdirección de Cómputo procesa y notifica el estado de los datos de la nómina. 440 La Subdirección de Planeación emite reportes o archivos.</p>
<p>356 CATEDRA</p>	<p>510 La Subdirección de Planeación solicita información a la Subdirección de Cómputo. 520 La Subdirección de Cómputo carga nómina. 530 La Subdirección de Cómputo procesa y notifica el estado de los datos de la nómina. 540 La Subdirección de Planeación emite reportes o archivos.</p>
<p>341 CALIDAD Y EFICIENCIA</p>	<p>610 La Subdirección de Planeación hace una solicitud de información. 620 La Subdirección de Cómputo carga los datos de nómina. 630 La Subdirección de Cómputo procesa y notifica el estado de los datos de la nómina. 640 La Subdirección de Planeación emite reportes o archivos.</p>
<p>344 EDPAC</p>	<p>710 La Subdirección de Planeación hace una solicitud de información. 720 La Subdirección de Cómputo carga los datos de nómina. 730 La Subdirección de Cómputo procesa y notifica el estado de los datos de la nómina. 740 La Subdirección de Planeación emite reportes o archivos.</p>

El periodo aproximado de tiempo entre los eventos de las diferentes partidas es el siguiente:

PARTIDA / PROGRAMA	PERIODICIDAD
351 DIPASI	Mensual (Quincenas pares).
351 FOMDOC	Mensual (Quincenas pares).
351 PEPASIG	Mensual (Quincenas pares).
355 PRIDE Y PAIPA	Quincenal.
356 CATEDRA	Quincenal.
341 CALIDAD Y EFICIENCIA	Quincenal (pagos especiales cuatrimestrales).
344 EDPAC	Quincenal (pagos especiales cuatrimestrales).

En las dos tablas anteriores se observa el funcionamiento requerido del sistema para cubrir las necesidades de información y los objetivos planteados inicialmente.

III.4.3.- El diccionario de eventos.

A continuación se encuentra el diccionario de eventos del sistema:

III.4.3.1.- SIPES I - Partida 351 Estimulos a la productividad y al rendimiento del personal académico de asignatura y de fomento a la docencia para el personal de carrera.

III.4.3.1.1.- DIPASI.

Id del evento :	110
La Subdirección de Planeación hace una solicitud de información.	
Descripción:	Debido a sus funciones la Subdirección de Planeación hace solicitudes

Capítulo III - Descripción del sistema y análisis

	de información acerca de la quincena actual o quincenas anteriores sobre la partida 351 en el programa DIPASI dependiendo de sus necesidades.
Estimulo:	Necesidades de información presupuestal de la Subdirección de Planeación.
Actividad:	El personal de la Subdirección de Planeación hace una petición oral o escrita de los datos de la partida 351 DIPASI.
Respuesta:	Solicitud oral o escrita a la Subdirección de Cómputo.
Efecto:	La Subdirección de Cómputo ejecuta los procesos necesarios para la obtención de la información.

Id del evento :	120
La Subdirección de Cómputo carga nómina.	
Descripción:	Tras la solicitud de la Subdirección de Planeación, la Subdirección de Cómputo toma los datos de nómina de la Dirección General de Personal y los ingresa en la Base de Datos SISAP para poder utilizarlos.
Estimulo:	Petición oral o escrita de los reportes de DIPASI para la quincena actual.
Actividad:	Se carga de manera masiva en la tabla bnomdgp2 los datos de nómina de la Dirección General de Personal a partir del archivo bnomdgp.txt (de la quincena solicitada). Se crean los índices de la tabla.
Respuesta:	Nómina lista para el proceso de administración.
Efecto:	La Subdirección de Cómputo procede a procesar los datos de la partida en específico.

--	--

Capítulo III - Descripción del sistema y análisis

Id del evento :	130
La Subdirección de Cómputo procesa datos de nómina.	
Descripción:	Este paso intermedio permite procesar la información de la partida 351 para obtener los datos necesarios para los reportes.
Estimulo:	Datos cargados y actualizados de la nómina.
Actividad:	Se actualizan los datos clave de DIPASI en la tabla bnomdgp2 para distinguirlos del resto. Se verifican los totales de la partida los cuales deben coincidir con los datos de la Dirección General de Personal.
Respuesta:	Notificación a la Subdirección de Planeación del estado de los datos después del proceso.
Efecto:	La Subdirección de Planeación procede a emitir reportes o archivos necesarios.

Id del evento :	140
La Subdirección de Planeación emite reportes o archivos.	
Descripción:	Una vez preparados los datos la Subdirección de Planeación ejecuta el programa SIPES I para emitir los reportes que requiera sobre el programa DIPASI.
Estimulo:	Notificación de la Subdirección de Cómputo del estado de los datos.
Actividad:	Ejecutar programa elige opción Dipasi Elegir Reporte por persona Reporte por dependencia Con el reporte elegido Impresión o Guardado en disco Cerrar el Reporte Cerrar la opción Dipasi

Capítulo III - Descripción del sistema y análisis

	Fin del programa Sipes I
Respuesta:	Reportes o archivos solicitados.
Efecto:	Fin del proceso.

III.4.3.1.2.- FOMDOC.

Id del evento :	210
La Subdirección de Planeación hace una solicitud de información.	
Descripción:	Según los requerimientos de información de la Subdirección de Planeación se solicitan los reportes del programa FOMDOC
Estimulo:	Necesidades de información de la Subdirección de Planeación.
Actividad:	Emitir la solicitud de información a la Subdirección de Cómputo. Entrega del disco de datos con el archivo fodoxxx.txt
Respuesta:	Solicitud oral o escrita de información. Archivo de la DGAPA con información mensual del programa FOMDOC.
Efecto:	La Subdirección de Cómputo toma el control en el proceso de los datos de FOMDOC.

Id del evento :	220
La Subdirección de Cómputo carga datos de la nómina.	
Descripción:	Después de recibida la solicitud de información la Subdirección de Cómputo carga los datos necesarios de nómina de la quincena adecuada.

Capítulo III - Descripción del sistema y análisis

Estimulo:	Archivo de nómina de la Dirección General de Personal.
Actividad:	Se carga la nómina de manera masiva en el archivo bnomdgp2 de la base de datos SISAP. Se crean los índices en la tabla bnomdgp2 351.
Respuesta:	La Subdirección de Cómputo procesa los datos de la partida .
Efecto:	Archivo de FOMDOC y nómina, listos para la emisión de los reportes o archivos.

Id del evento :	230
La Subdirección de Cómputo procesa los datos del programa FOMDOC.	
Descripción:	Una vez obtenido el archivo de FOMDOC se debe poner la información en la base de datos SISAP que se complementa con los catálogos de información y la nómina de la base de datos SISAP.
Estimulo:	Tabla de nómina bnomdgp2 de la base de datos SISAP. Catálogos actualizados en la base de datos SISAP. Archivo FOMDOC fodoxxx.txt (donde de xxx es el mes y año de generación de los datos).
Actividad:	Lee el primer renglón de archivo fodoxxx.txt. Mientras no sea fin del archivo fodoxxx.txt haz Completa el registro a armar con los catálogos Carga registro completo en la tabla bfomdoc Siguiete renglón del archivo fodoxxx.txt fin mientras Se verifican los totales de FOMDOC contra los datos de la DGAPA.
Respuesta:	Notificación de los datos del programa FOMDOC.
Efecto:	La Subdirección de Planeación se dispone a generar los reportes que requiere.

Capítulo III - Descripción del sistema y análisis

Id del evento :	240
La Subdirección de Planeación emite reportes o archivos.	
Descripción:	Ya que los datos de FOMDOC han sido debidamente preparados el usuario en la Subdirección de Planeación procede a emitir o en su caso salvar los datos de la partida 351 FOMDOC.
Estimulo:	Tabla bfomdoc de la base de datos SISAP. Catálogos de la base de datos SISAP.
Actividad:	Ejecuta programa SIPES I Elige opción fomdoc. Elige tipo del reporte General Por dependencia Por categoría Elige mes del reporte Vista preliminar del reporte elegido. Impresión o grabado de datos en disco. Cerrar vista preliminar Cerrar opción fomdoc Finalizar ejecución SIPES I
Respuesta:	Reportes o archivos partida 351 FOMDOC.
Efecto:	Se da por terminado el proceso de la información por parte de la Subdirección de Planeación.

III.4.3.1.3. - PEPASIG.

Id del evento :	310
La Subdirección de Planeación solicita Información a la Subdirección de Cómputo sobre programa PEPASIG partida 351.	
Descripción:	En el caso de PEPASIG la Subdirección de Planeación realiza la petición de Información a la Subdirección de Cómputo en forma oral o escrita.

Capítulo III - Descripción del sistema y análisis

Estimulo:	Requerimientos de información por parte de la Subdirección de Planeación.
Actividad:	El personal de la Subdirección de Planeación hace la petición oral o escrita de información a la Subdirección de Cómputo.
Respuesta:	Solicitud de proceso de datos a la Subdirección de Cómputo.
Efecto:	Inicio del proceso de información por parte de la Subdirección de Cómputo.

Id del evento :	320
La Subdirección de Cómputo carga nómina de la Dirección General de Personal.	
Descripción:	Tras recibir la solicitud de Información de la Subdirección de Planeación se cargan los datos necesarios de la nómina en la base de datos SISAP de la quincena solicitada.
Estimulo:	Archivo de nómina de la Dirección General de Personal nóminaxxxx.txt
Actividad:	Se carga el archivo de nómina de manera masiva en la tabla bnomdgp2 de la base de datos SISAP. Se crean los índices en la tabal bnomdgp2.
Respuesta:	La Subdirección de Cómputo procede a procesar los datos de la partida específica.
Efecto:	La Subdirección de Planeación cuenta con la información preparada para la emisión de sus reportes.

Id del evento :	330
------------------------	-----

Capítulo III - Descripción del sistema y análisis

La Subdirección de Cómputo procesa y notifica el estado de los datos a la Subdirección de Planeación.	
Descripción:	Una vez preparados los datos de la nómina el proceso actualiza la información de la partida 351 PEPASIG y llena las tablas requeridas para los archivos y reportes.
Estímulo:	Tabla bnomdgp2 en datos de la quincena solicitada. Catálogos de la base de datos SISAP.
Actividad:	<p>Se actualizan los datos de clave de PEPASIG en al tabla bnomdgp2 para distinguirlos del resto de información de la partida.</p> <p>Solicitud de las diferentes dependencias en el programa Ir al primer registro</p> <p style="padding-left: 40px;">Mientras (no sean fin de la solicitud)</p> <p style="padding-left: 80px;">Calcula el total de persona - importe por dependencia</p> <p style="padding-left: 80px;">Inserta datos en tabla Tpepdep</p> <p style="padding-left: 80px;">Siguiete dependencia</p> <p style="padding-left: 40px;">Fin del mientras.</p> <p>Comparación de los totales de la partida obtenidos contra los datos de nómina.</p> <p>Notificación de la Subdirección de Cómputo a la Subdirección de Planeación sobre el estado de los datos.</p>
Respuesta:	La Subdirección de Planeación recibe el visto bueno para la generación de reportes y archivos.
Efecto:	La Subdirección de Planeación esta en condiciones de emitir sus reportes o archivos según le convenga.

Id del evento :	340
La Subdirección de Planeación emite reportes o archivos.	
Descripción:	Una vez que los datos de PEPASIG han sido debidamente procesados la Subdirección de Planeación utiliza el SIPES I para obtener los reportes o archivos que requiere.

Capítulo III - Descripción del sistema y análisis

Estímulo:	Tablas bnomdgp2 y tpepedep de la base de datos SISAP Catálogos de la base de datos SISAP.
Actividad:	Ejecutar el programa SIPES I Elegir opción PEPASIG Elegir tipo del reporte. Por rango de horas Generar por horas académicas. Por dependencia y rango de horas. Vista preliminar del reporte elegido Impresión o salvado de los datos en archivo Cerrar vista preliminar Cerrar opción PEPASIG Finalizar ejecución SIPES I
Respuesta:	Reportes o archivos deseados.
Efecto:	Fin del proceso de la partida 351 PEPASIG.

III.4.3.2.- SIPES I – Partida 355 Prima de desempeño al personal Académico.

III.4.3.2.1.- PRIDE Y PAIPA.

Id del evento :	410
La Subdirección de Planeación solicita información a la Subdirección de Cómputo.	
Descripción:	Según sus necesidades de información la Subdirección de Planeación realiza una solicitud acerca de PRIDE y PAIPA en la Subdirección de Cómputo.
Estímulo:	Necesidades de información de la Subdirección de Planeación.
Actividad:	Los usuarios de la Subdirección de Planeación realizan una solicitud escrita u oral a la Subdirección de Cómputo sobre los datos que se requieren acerca de la partida 355.

Capítulo III - Descripción del sistema y análisis

Respuesta:	Solicitud entregada a la Subdirección de Cómputo.
Efecto:	La Subdirección de Cómputo inicia sus procesos de preparación de la información.

Id del evento :	420
La Subdirección de Cómputo carga información de la nómina.	
Descripción:	Una vez recibida la solicitud de información de la Subdirección de Planeación la Subdirección de Cómputo inicia la preparación de los datos necesarios para los reportes y archivos de ambos programas. El primer paso consiste en actualizar el archivo de nómina al requerimiento actual.
Estímulo:	Archivo de nómina Dirección General de Personal nominaxxx.txt.
Actividad:	Se carga el archivo de nómina de manera masiva en la tabla bnomdgp2 de la base de datos SIPES. Se crean los índices necesarios en la tabla bnomdgp2.
Respuesta:	La Subdirección de Cómputo una vez concluida la carga procesa la información de la partida.
Efecto:	Se da paso al proceso de los datos de la partida.

Id del evento :	430
La Subdirección de Cómputo procesa y notifica el estado de los datos a la Subdirección de Planeación.	
Descripción:	Después de preparar la información fuente de la nómina el encargado de administrarla utiliza estos datos mas los catálogos de la base de datos SISAP, con el fin de crear o en su caso actualizar las tablas referentes al programa PRIDE y PAIPA.

Capítulo III - Descripción del sistema y análisis

Estimulo:	Tabla bnomdgp2 con datos de la quincena solicitada. Catalogo de información de la base de datos SISAP.
Actividad:	Se elimina el contenido de bpride Se solicitan los datos de partida 355 de la nómina bnomdgp2 Mientras (no sea fin de la solicitud) Solicita los datos que completan el registro de bpride Inserta registro en bpride Siguiete registro Fin del mientras Comparación de totales con los datos de nómina de la Dirección General de Personal. Notificación del estado de la información a la Subdirección de Planeación.
Respuesta:	Los datos procesados y listos para emitir los reportes o archivos de información.
Efecto:	La Subdirección de Planeación dispone de la información para generar los reportes o archivos adecuados.

Id del evento :	440
La Subdirección de Planeación emite reportes o archivos.	
Descripción:	En la ultima parte del proceso de la partida 355 la Subdirección de Planeación utiliza el sistema SIPES para imprimir o guardar a disco la información ya procesada, en el caso de esta partida los datos requeridos pueden ser de PRIDE Y PAIPA o solo PAIPA por lo tanto el usuario tiene ambas opciones.
Estimulo:	Tablas bnomdgp2, bpride de la base de datos SISAP Catálogos de la base de datos SISAP.
Actividad:	Ejecutar el programa SIPES I Elegir opción PRIDE O PAIPA Si elige PRIDE Elige Reporte

Capítulo III - Descripción del sistema y análisis

	<p>Por categoría Por subramo Pos subsistema Por categoría y nivel Por subramo nivel y porcentaje Por dependencia nivel y porcentaje Por categoría y cuota fija Preliminar del reporte elegido Impresión o datos guardados en archivo Cerrar vista preliminar Cerrar opción PRIDE Fin del Si Si elige PAIPA Elige reporte General PAIPA ENEP's sin PAIPA Resto UNAM sin PAIPA Vista preliminar del reporte elegido Impresión de datos guardados en archivo Cerrar vista preliminar Cerrar opción PAIPA Fin del Si Finalizar ejecución de Sipes I</p>
Respuesta:	Reportes o archivos deseados.
Efecto:	Fin del proceso de datos PRIDE, PAIPA.

III.4.3.3.- SIPES I – Partida 356 Reconocimiento catedrático UNAM.

III.4.3.3.1.- CATEDRA.

Id del evento :	510
La Subdirección de Planeación solicita información a la Subdirección de Cómputo.	
Descripción:	Según sus necesidades de información la Subdirección de Planeación realiza una solicitud acerca de CATEDRA en la Subdirección de Cómputo.
Estimulo:	Necesidades de información de la Subdirección de Planeación.

Capítulo III - Descripción del sistema y análisis

Actividad:	Los usuarios de la Subdirección de Planeación realizan una solicitud escrita y oral a la Subdirección de Cómputo sobre los datos que se requieren acerca de la partida 356.
Respuesta:	Solicitud entregada a la Subdirección de Cómputo.
Efecto:	La Subdirección de Cómputo inicia sus procesos de preparación de la información.

Id del evento :	520
La Subdirección de Cómputo carga información de la nómina.	
Descripción:	Una vez recibida la solicitud de información de la Subdirección de Planeación la Subdirección de Cómputo inicia la preparación de los datos necesarios para los reportes y archivos del programa. El primer paso consiste en actualizar el archivo de nómina al requerimiento actual.
Estimulo:	Archivo de nómina Dirección General de Personal nominaxxx.txt.
Actividad:	Se carga el archivo de nómina de manera masiva en la tabla bnomdgp2 de la base de datos SIPES. Se crean los índices necesarios en la tabla bnomdgp2.
Respuesta:	La Subdirección de Cómputo una vez concluida la carga procesa la información de la partida.
Efecto:	Se da paso al proceso de los datos de la partida.

Id del evento :	530
------------------------	-----

Capítulo III - Descripción del sistema y análisis

La Subdirección de Cómputo procesa y notifica el estado de los datos a la Subdirección de Planeación.	
Descripción:	Después de preparar la información fuente de la nómina el encargado de administrarla utiliza estos datos mas los catálogos de la base de datos SISAP, con el fin de crear o en su caso actualizar las tablas referentes al programa CATEDRA.
Estimulo:	Tabla bnmdgp2 con datos de la quincena solicitada Catalogo de Información de la base de datos SISAP.
Actividad:	Se elimina el contenido de brecati Se solicitan los datos de partida 356 de la nómina bnmdgp2 Mientras (no sea fin de la solicitud) Solicita los datos que completan el registro de brecati Inserta registro en brecati Siguiete registro Fin del mientras Comparación de totales con los datos de nómina de la Dirección General de Personal Notificación del estado de la Información a la Subdirección de Planeación
Respuesta:	Los datos procesados y listos para emitir los reportes o archivos de información.
Efecto:	La Subdirección de Planeación dispone de la información para generar los reportes o archivos adecuados.

Id del evento :	540
La Subdirección de Planeación emite reportes o archivos.	
Descripción:	En la ultima parte del proceso de la partida 355 la Subdirección de Planeación utiliza el sistema SIPES para imprimir o guardar a disco la información ya procesada del programa CATEDRA.
Estimulo:	Tablas bnmdgp2, brecati de la base de datos SISAP. Catálogos de la base de datos SISAP.

Capítulo III - Descripción del sistema y análisis

Actividad:	Ejecutar el programa SIPES I Elegir opción CATEDRA Si elige CATEDRA Elige Reporte Por categoría Por subramo Por subsistema Por categoría y nivel Por persona Preliminar del reporte elegido Impresión o datos guardados en archivo Cerrar vista preliminar Cerrar opción CATEDRA. Fin del Si Finalizar ejecución de Sipes I
Respuesta:	Reportes o archivos deseados.
Efecto:	Fin del proceso de datos CATEDRA.

III.4.3.4.- SIPES II - Partida 341 Complemento al salario por calidad y eficiencia en el trabajo del personal administrativo.

III.4.3.4.1.- CALIDAD Y EFICIENCIA.

Id del evento :	610
La Subdirección de Planeación hace una solicitud de información.	
Descripción:	El evento activador del proceso es la solicitud de la Subdirección de Planeación de información la cual implica una serie de labores para resolver esta necesidad inicial.
Estimulo:	Petición oral o escrita de datos por parte de la Subdirección de Planeación.
Actividad:	Los usuarios de la Subdirección de Planeación emiten una petición oral o escrita a la Subdirección de Cómputo.

Capítulo III - Descripción del sistema y análisis

Respuesta:	La Subdirección de Cómputo recibe solicitud y procede a preparar la información requerida.
Efecto:	Inicio del proceso de datos Calidad y Eficiencia.

Id del evento :	620
La Subdirección de Cómputo carga datos de nómina.	
Descripción:	Tras recibir la solicitud de información de la Subdirección de Planeación se deben cargar los datos de la nómina solicitada para poder procesar la información requerida de la partida 341.
Estimulo:	Archivo de nómina de la Dirección General de Personal nominaxxx.txt.
Actividad:	Se carga de manera masiva el archivo nominaxxx.txt en la tabla bnomdgp2 de la base de datos SISAP. Se crean los índices en la tabla bnomdgp2.
Respuesta:	Tabla bnomdgp2 actualizada.
Efecto:	Permite la continuación del proceso de la partida.

Id del evento :	630
La Subdirección de Cómputo procesa y notifica el estado de los datos a la Subdirección de Planeación.	
Descripción:	Tras obtener la tabla bnomdgp2 el proceso pasa ahora a los datos sobre la partida en especial; así se obtienen los datos de la tabla bcalyefi que pondrá la información a disposición del sistema SIPES II.
Estimulo:	La tabla bnomdgp2 con datos de la quincena solicitada catálogos de información de la base de datos SISAP.

Capítulo III - Descripción del sistema y análisis

Actividad:	<p>Se elimina el contenido de bcalyefi Se solicitan los datos de la partida 341 Ir al primer registro Mientras (no sea fin de la solicitud) Solicita datos que complementan el registro de bcalyefi Inserta datos en bcalyefi Siguiendo registro Fin del mientras Actualización de los datos faltantes de la tabla bcalyefi Comparación de totales con los datos de nómina de la Dirección General de Personal. Notificación del estado de la información a la Subdirección de Planeación.</p>
Respuesta:	Los datos de bcalyefi actualizados a la quincena solicitada.
Efecto:	La Subdirección de Planeación dispone ahora de los datos para generar de reportes o archivos.

Id del evento :	640
La Subdirección de Planeación genera reportes o archivos.	
Descripción:	El evento final en esta partida consiste en la obtención de los reportes solicitados por la Subdirección de Planeación, cabe mencionar que estos reportes ya pertenecen a SIPES II.
Estimulo:	Tablas bnomdgp2, bcalyefi con datos de la partida. Catálogos actualizados de la base de datos SISAP.
Actividad:	<p>Ejecutar el SIPES II Elegir la opción Calidad y Eficiencia Elige reporte Totales Cal y Eficiencia Vista preliminar del reporte elegido Impresión o salvar datos en archivo Cerrar vista preliminar Cerrar opción cal y eficiencia Finalizar ejecución de SIPES II</p>

Capítulo III - Descripción del sistema y análisis

Respuesta:	Datos solicitados originalmente.
Efecto:	Fin del proceso de datos Calidad y Eficiencia.

III.4.3.5.- SIPES II - Partida 344 Estimulos y gratificaciones complementarias.

III.4.3.5.1.- EDPAC.

Id del evento :	710
La Subdirección de Planeación hace una solicitud de información.	
Descripción:	El evento activador del proceso es la solicitud de la Subdirección de Planeación de información la cual implica una serie de labores para resolver esta necesidad inicial.
Estimulo:	Petición oral o escrita de datos por parte de la Subdirección de Planeación.
Actividad:	Los usuarios de la Subdirección de Planeación emiten una petición oral o escrita a la Subdirección de Cómputo.
Respuesta:	La Subdirección de Cómputo recibe solicitud y procede a preparar la información requerida.
Efecto:	Inicio del proceso de datos EDPAC.

Id del evento :	720
La Subdirección de Cómputo carga datos de nómina.	
Descripción:	Tras recibir la solicitud de información de la Subdirección de Planeación se deben cargar los datos de la nómina solicitada para poder procesar la información requerida de la partida 341.

Capítulo III - Descripción del sistema y análisis

Estimulo:	Archivo de nómina de la Dirección General de Personal nóminaxxxx.txt.
Actividad:	Se carga de manera masiva el archivo nóminaxxxx.txt en la tabla bnomdgp2 de la base de datos SISAP. Se crean los índices en la tabla bnomdgp2.
Respuesta:	Tabla bnomdgp2 actualizada.
Efecto:	Permite la continuación del proceso de la partida.

Id del evento :	730
La Subdirección de Cómputo procesa y notifica el estado de los datos a la subdirección de Planeación.	
Descripción:	Tras obtener la tabla bnomdgp2 el proceso pasa ahora a los datos sobre la partida en especial así se obtienen los datos de la tabla bedpac que pondrá la información a disposición del sistema SIPES II.
Estimulo:	La tabla bnomdgp2 con datos de la quincena solicitada catálogos de información de la base de datos SISAP.
Actividad:	Se elimina el contenido de bedpac Se solicitan los datos de la partida 344 Ir al primer registro Mientras (no sea fin de la solicitud) Solicita datos que complementan el registro de bedpac Inserta datos en bedpac Siguiete registro Fin del mientras Actualización de los datos faltantes de la tabla bedpac. Comparación de totales con los datos de nómina de la Dirección General de Personal. Notificación del estado de la información a la Subdirección de Planeación.
Respuesta:	Los datos de bedpac actualizados a la quincena solicitada.

Capítulo III - Descripción del sistema y análisis

Efecto:	La Subdirección de Planeación dispone ahora de los datos para generar de reportes o archivos.
----------------	---

Id del evento :	740
La Subdirección de Planeación genera reportes o archivos.	
Descripción:	El evento final en esta partida consiste en la obtención de los reportes solicitados por la Subdirección de Planeación, cabe mencionar que estos reportes ya pertenecen a SIPES II.
Estimulo:	Tablas bnomdgp2, bedpac con los datos de la partida. Catálogos actualizados de la base de datos SISAP.
Actividad:	Ejecutar el SIPES II Elegir la opción Calidad y Eficiencia Elegir reporte Totales EDPAC Vista preliminar del reporte elegido Impresión o salvar datos en archivo Cerrar vista preliminar Cerrar opción EDPAC Finalizar ejecución de SIPES II
Respuesta:	Datos solicitados originalmente.
Efecto:	Fin del proceso de datos EDPAC.

III.5.- Modelo de información.

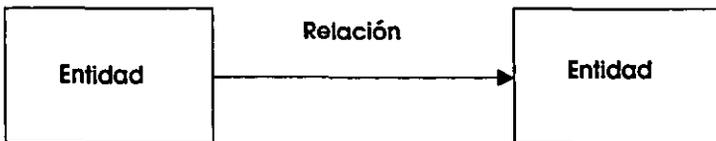
La última fase de análisis involucra directamente a los datos, en los modelos anteriores se estudiaron las entidades que participan en el sistema (modelo de contexto), a continuación lo que ocurre durante la ejecución y alrededor del sistema (modelo de eventos) y como consecuencia damos paso al estudio de la información que este maneja.

Esta sección del análisis sirve para distinguir las entidades relaciones y atributos de los datos que intervienen en el proceso que desarrollamos, a través de diferentes herramientas tales como los diagramas entidad-relación, las estimaciones de volumen y retención para cada entidad, el listado de atributos, la definición de cada entidad, relación y atributo, las propiedades de los atributos y los diagramas estado-transición, entre otros; esto permite un amplio espectro de información para los desarrolladores y programadores del sistema, la creación de cualquiera de las guías anteriores depende en gran medida del alcance que se requiera en cuanto a la información que se necesite.

La primera herramienta utilizada en este trabajo es el diagrama de entidad relación, el cual es un elemento gráfico del análisis que nos permite observar las entidades y las relaciones que tienen entre sí.

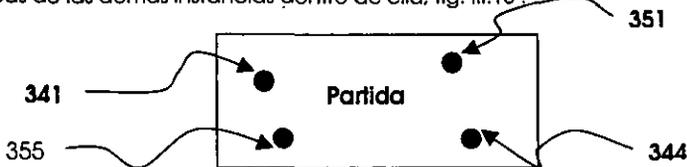
Entidad: Es una persona, lugar, cosa o idea abstracta sobre la que el sistema necesita recordar algo. Las instancias a una entidad son las representaciones de la entidad dentro del sistema, cada una de ellas es congruente con la entidad de la que derivan en características y comportamiento .

Relación: La asociación entre instancias de las entidades se conoce como relación, representada como una línea que va de una instancia de entidad, a otra permite distinguir cuales son los puntos de unión entre ellas, fig. III.17.



Capítulo III, Figura III.17 Modelo entidad relacion

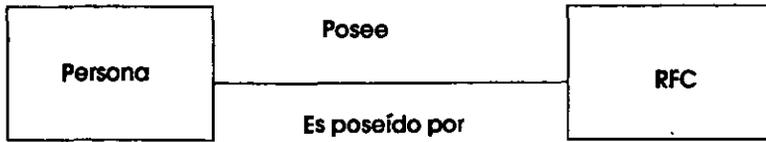
Cada miembro de una entidad (instancia) es único y comparte las mismas características de las demás instancias dentro de ella, fig. III.18.



Capítulo III, Figura III.18 Entidades

Cada entidad posee un nombre y descripción de lo que es o representa en la realidad o el proceso del sistema.

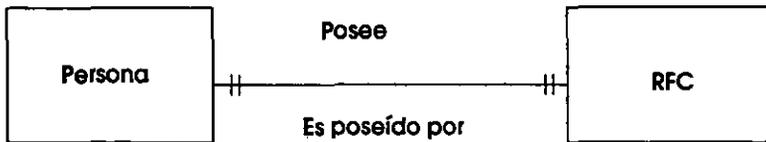
Las relaciones poseen una dirección que les da sentido dentro del contexto del sistema, fig III.19; Por ejemplo:



Capítulo III, Figura III.19 Relaciones

La dirección en este ejemplo esta dada de izquierda a derecha que permite la relación: Persona posee un RFC y de derecha a izquierda, RFC es poseído por una persona, con lo cual se establece claramente en que relación quedan las entidades entre sí e inclusive la cardinalidad entre ellas.

La cardinalidad consiste en el valor mínimo y máximo en una relación entre instancias retomando el caso anterior tenemos, fig III.20:



Capítulo III, Figura III.20 Cardinalidad

Que se lee: Una persona posee uno y solo un RFC o bien en el sentido contrario: RFC es poseído por una y solo una persona. El resto de las cardinalidades se resume en la siguiente tabla.

Intervalo	
Min - Max	Notación
Cero a uno	—○
Cero a muchos	—○⋈
Uno a uno	— —
Uno a muchos	— ⋈

Aunado al concepto de cardinalidad tenemos el de normalización que es un método para la asignación de cada elemento de dato en un modelo de información hacia la entidad que mejor corresponde. Es decir es una forma de representación de los datos en un formato relacional no redundante. Desarrollada por Edgar F. Codd para extender la expectativa de vida de las aplicaciones al permitir extensiones a los datos, funcionalidad y cambios en la naturaleza de las consultas con un mínimo de trastornos.

La siguiente parte del modelo involucra a los atributos de las entidades descritas en los Diagramas de Entidad Relación, en estas tablas encontraremos a detalle las entidades y los atributos que las conforman con la estructura Nombre, Requerido (si es campo necesario o no), Tiempo de Dato y Definición, estas características nos permitirán en una fase posterior de desarrollo la creación y operación de una base de datos relacional de manera eficiente.

A continuación como en el caso de los modelos anteriores se estudia cada partida en particular aplicando los conceptos concebidos sobre el modelo de información.

Nota: Por convención se utilizarán el nombre de entidades que serán necesarios en el desarrollo y programación formal del sistema.

III.5.1.- SIPES I - Partida 351 Estimulos a la productividad y al rendimiento del personal académico de asignatura y de fomento a la docencia para el personal de carrera.

III.5.1.1.- DIPASI.

Diagrama Entidad Relación

Véase Figura III.21.

Definición de entidad y atributos

Entidad: **bnomdgp2**

Definición: Este es el archivo más importante del sistema SIPES involucra los datos de todas las personas que reciben remuneración quincenal en la UNAM, también conocido como archivo de percepciones contiene información sobre el código programático, registro federal de causantes, importe de nómina entre otros.

Atributos :

Nombre	Requerido	Tipo de dato	Definición
Clave de programa	Sí	Entero	Parte del código programático que distingue el programa del pago.
Clave de subprograma	Sí	Entero	Parte del código programático que distingue el subprograma del pago.

MODELO DE INFORMACIÓN

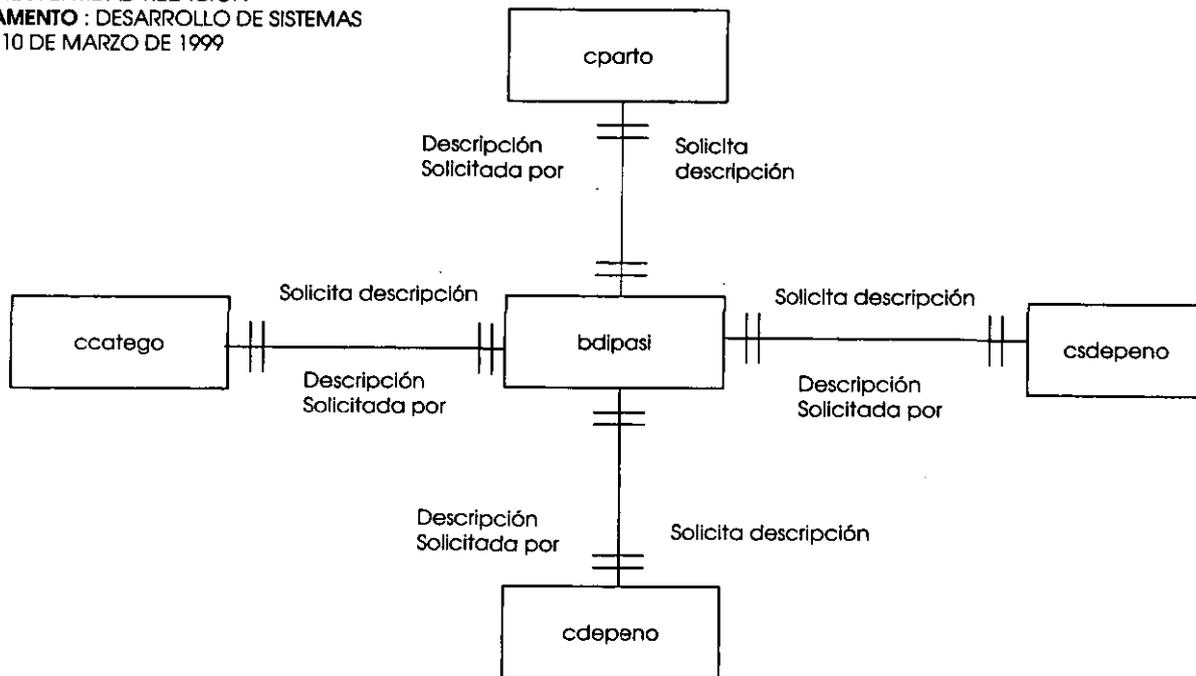
PARTIDA : 351 - DIPASI

MÓDULO : SIPES I

DIAGRAMA : ENTIDAD RELACIÓN

DEPARTAMENTO : DESARROLLO DE SISTEMAS

FECHA : 10 DE MARZO DE 1999



Capítulo III, Figura III.21

Capítulo III - Descripción del sistema y análisis

Clave de dependencia	Sí	Entero	Parte del código programático que distingue a la dependencia del pago.
Clave de subdependencia	Sí	Entero	Parte del código programático que distingue a la subdependencia del pago.
Clave tipo de percepción	Sí	Entero	Tipo de percepción que se emite en el pago. NE= No especificado. DE= Devolución. DS= Diferencia de sueldo. PD= Pride. PE= Percepciones. DI= Devolución de impuesto. PV= Pago vales.
Clave de partida	Sí	Entero	Parte del código programático que distingue a la partida del pago.
Clave número de nómina	No	Entero	Número de referencia de la nómina.
Clave de categoría nominal	Sí	Carácter (2)	Clave en letras de la categoría de la persona en el pago.
Clave de categoría tipo	Sí	Entero	Clave en números de la categoría es la segunda parte del código que distingue a las categorías dentro del tabulador.
Clave de categoría nivel	Sí	Entero	Clave del nivel de la categoría, con este código se completa la clave de categoría, estas tres claves se utilizan como referencia para los tabuladores del personal académico o administrativo.
Descripción categoría en nómina	Sí	Carácter (14)	Este atributo es agregado por la Dirección General de Personal para describir el pago realizado.
Clave del subnivel de categoría	No	Carácter (3)	Es la clave que define un subnivel dentro de una categoría o tipo de pago, por ejemplo : A,B,C = Dipasl.
Clave PRIDE	No	Carácter (2)	Clave de pride en la nómina para distinguir el nivel de pago.
Clave lugar de pago	No	Entero	Es la clave del lugar donde se realiza el pago.
Registro Federal de Causantes.	Sí	Carácter (15)	Cada empleado académico o administrativo posee un registro federal de contribuyentes que lo identifica ante la Secretaría de Hacienda para el pago de impuestos.
Clave digito de control	No	Entero	Es el digito que distingue pagos especiales dentro de la nómina.
Clave nómina actual	Sí	Entero	Es el número de la quincena en que se emitió el pago.
Clave numero de año	No	Entero	Es el año en que emitió el pago.
Clave numero de plaza	No	Entero	Es el número de la plaza que ocupa la persona que recibe el pago.
Precio de antigüedad	No	Moneda	Es el pago de antigüedad de la persona a la fecha actual.
Precio de importe en nómina	Sí	Moneda	Es el importe del pago del trabajador.

Capítulo III - Descripción del sistema y análisis

Fecha de movimiento	No	Fecha y hora	Fecha en que se realizó el movimiento del pago.
Fecha límite en nómina	No	Fecha y hora	Fecha límite en que se ejerció el pago.

Entidad: cparto

Definición: Es el catálogo que contienen todos los datos acerca de las partidas del presupuesto universitario.

Atributos :

Nombre	Requerido	Tipo de dato	Definición
Clave de partida	Sí	Entero	Es la clave de la partida necesaria para obtener la descripción.
Clave de tipo de ejercicio	No	Entero	Es el tipo de ejercicio de la partida X= No identificado. D= Operación directa. C= Centralizada. S= Complementaria
Clave del responsable	No	Entero	Es la clave de la persona responsable en la Subdirección de Planeación.
Clave del tipo de gasto	No	Entero	Es la clave para la distinción entre gastos existentes.
Clave del tipo de prestación	No	Entero	Esta clave distingue el tipo de prestación que representa el ejercicio de la partida.
Descripción de la partida	Sí	Carácter (80)	Es el título completo de la partida.
Clave del analista encargado	No	Entero	Es la clave del encargado de la partida en Subdirección de Análisis y Operación Presupuestal.

Entidad: ccatogo

Definición: Este catálogo contiene todas las categorías de los tabuladores existentes en la universidad.

Atributos :

Nombre	Requerido	Tipo de dato	Definición
Llave del archivo	Sí	Entero	Es llave índice de la tabla.
Clave de categoría nómina	Sí	Carácter (2)	Es la clave en letras de la categoría.
Clave de categoría tipo	Sí	Entero	Clave en números de la categoría se utiliza en conjunción clave nominal y la de nivel para ubicar a las personas en el tabulador .
Clave de categoría nivel	Sí	Entero	Clave del nivel de la categoría.

Capítulo III - Descripción del sistema y análisis

Clave de categoría	No	Entero	Es la clave que distingue las diferencias entre rangos de categorías.
Importe de la categoría	Sí	Moneda	Es el precio mensual de la categoría.
Importe material didáctico	No	Moneda	Es el precio de la prestación conocida como material didáctico según la categoría.
Número de horas de la categoría	No	Entero	Son las horas asignadas en cada categoría.
Fecha de última actualización	No	Fecha y hora	Es la fecha en que inicia la vigencia del tabulador, el día de la última actualización.
Descripción de la categoría	Sí	Carácter (40)	Es el título completo de la categoría.
Descripción abreviada	Sí	Carácter (15)	Es el título abreviado de la categoría (se utiliza en ciertos reportes).
Nivel de la categoría	Sí	Carácter (3)	Es el título del nivel de la categoría A, B, C.
Tiempo asignado a la categoría	No	Carácter (3)	Es el título del tiempo que implica la categoría. MT= Medio tiempo. TC= Tiempo Completo.

Entidad: cdepeno

Definición: Es el catálogo que contiene los datos de todas las dependencias de la UNAM y sus descripciones.

Atributos :

Nombre	Requerido	Tipo de dato	Definición
Clave dependencia	Sí	Entero	Es la clave de la dependencia permite distinguir a que dependencia esta haciendo referencia el código programático.
Clave subsistema	Sí	Entero	Es la clave del subsistema de dependencias al que esta pertenece.
Clave de subramo	Sí	Entero	Es la clave del subramo de dependencias al que esta pertenece.
Descripción de la dependencia.	Sí	Carácter (85)	Es el nombre completo de la dependencia.

Entidad: csdepeno

Definición: Es el catálogo que contiene información acerca de las subdependencias universitarias.

Atributos:

Nombre	Requerido	Tipo de dato	Definición
Clave de dependencia	Sí	Entero	La clave de la dependencia permite distinguir a que dependencia se hace referencia en el código programático.
Clave de subdependencia	Sí	Entero	Es la clave de cada una de las subdependencias que forman la Universidad Nacional Autónoma de México.
Clave área económica	No	Entero	Cada una de las subdependencias forma parte de una área económica dependiendo de su zona de ubicación geográfica.
Clave zona geográfica	No	Entero	Debido al carácter nacional de la universidad los distintos campus se encuentran ubicados en diferentes zonas geográficas perfectamente identificadas con esta clave.
Clave analista	No	Entero	Correspondiente a la clave del analista responsable de la subdependencia en la Subdirección de Análisis y Operación Presupuestal.
Descripción subdependencia	Sí	Carácter (80)	Es el nombre completo de la subdependencia.

En el caso del DIPASI y las siguientes partidas los datos han sido normalizados y no se duplicaran las entidades una vez definidas.

Esto se hace con el fin de evitar la repetición de definiciones en las entidades ya que como veremos mas adelante el uso de una Base de Datos relacional implica una serie de relaciones entre entidades para evitar la duplicidad.

III.5.1.2.- FOMDOC.

Diagrama Entidad - Relación

Véase Figura III.22.

Entidad: bfomdoc

Definición: En esta entidad encontraremos los datos enviados por la DGAPA sobre el programa FOMDOC los cuales se completan con la información que encontramos en la nómina de la DGP y cuyo contenido nos permite almacenar los datos mensuales del programa.

Atributo :

MODELO DE INFORMACIÓN

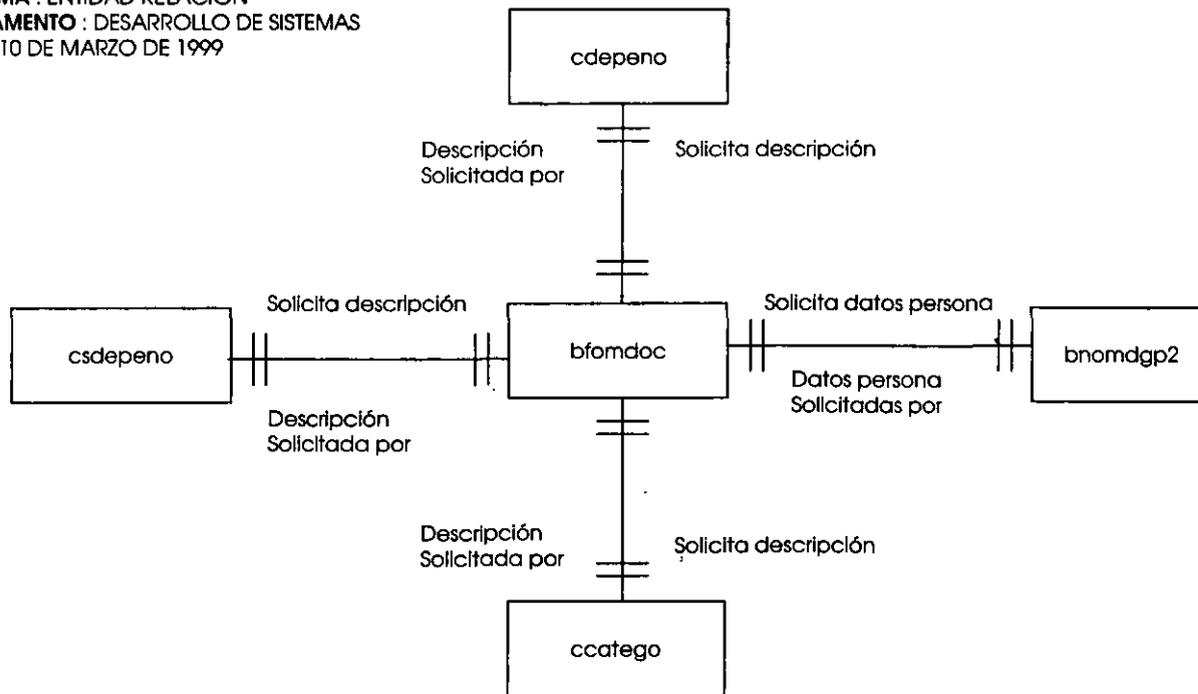
PARTIDA : 351 - FOMDOC

MÓDULO : SIPES I

DIAGRAMA : ENTIDAD RELACIÓN

DEPARTAMENTO : DESARROLLO DE SISTEMAS

FECHA : 10 DE MARZO DE 1999



Capítulo III, Figura III.22

MODELO DE INFORMACIÓN

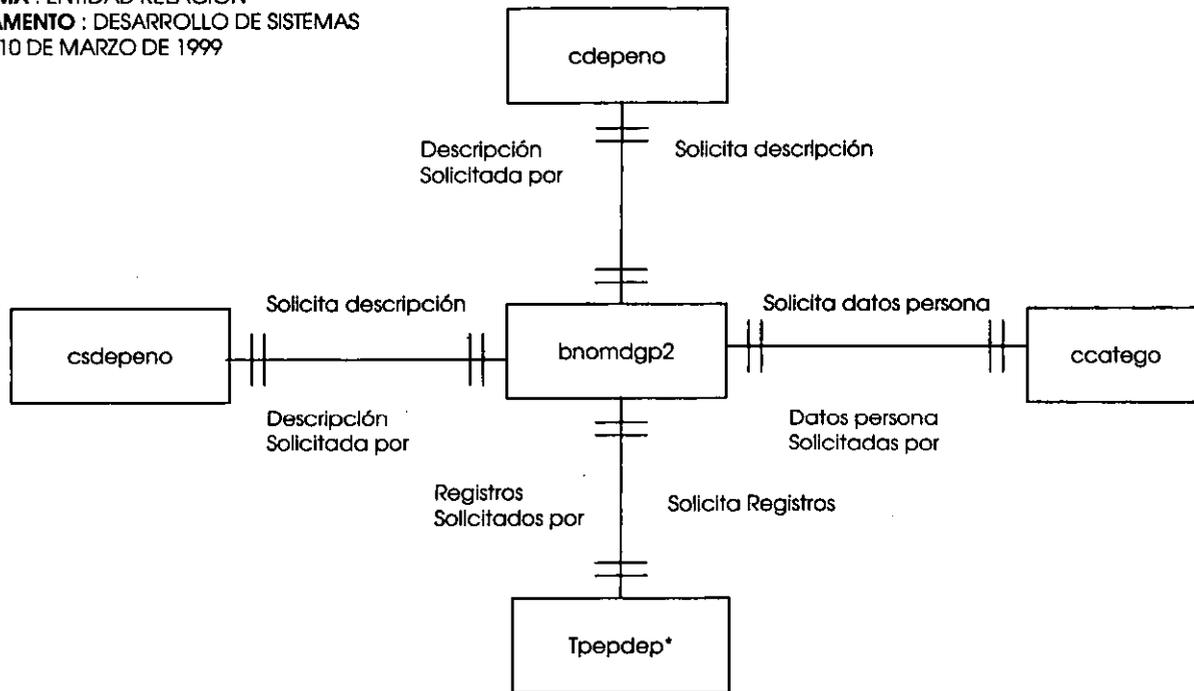
PARTIDA : 351 - PEPASIG

MÓDULO : SIPES I

DIAGRAMA : ENTIDAD RELACIÓN

DEPARTAMENTO : DESARROLLO DE SISTEMAS

FECHA : 10 DE MARZO DE 1999



* Contiene totales por dependencia

Capítulo III, Figura III.23

MODELO DE INFORMACIÓN

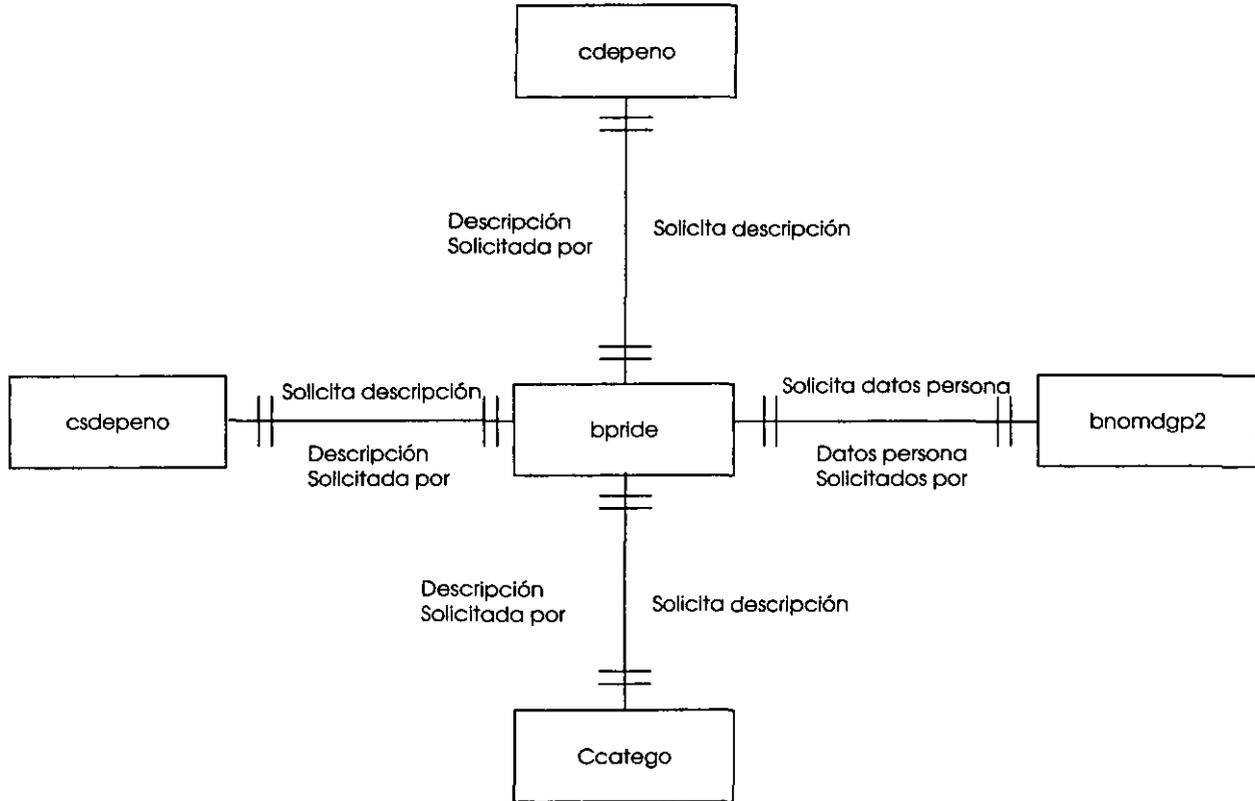
PARTIDA : 355 - PAIPA

MÓDULO : SIPES I

DIAGRAMA : ENTIDAD RELACIÓN

DEPARTAMENTO : DESARROLLO DE SISTEMAS

FECHA : 10 DE MARZO DE 1999



Capítulo III, Figura III.25

Capítulo III - Descripción del sistema y análisis

Clave de categoría nivel	Sí	Entero	Es la clave del nivel de la categoría dentro del tabulador.
Clave de pride	Sí	Moneda	Cada uno de los pagos de PRIDE tienen un nivel y porcentaje definido en base a las condiciones de ingreso al programa.
Numero de quincena	Sí	Entero	Es el número de la quincena actual en bnomdgp2.
Importe pride pagado	Sí	Moneda	Es el importe pagado en la nómina actual de PRIDE.
Descripción de categoría	Sí	Carácter (40)	Se requiere para tener el nombre de la categoría en la que esta el académico que recibió el pago.
Nivel de pride	Sí	Entero	Es el nivel A, B, C, D, en que puede tener el profesor dentro de PRIDE.
Porcentaje pride	Sí	Entero	Es el porcentaje sobre su sueldo integrado que recibe por concepto de PRIDE el académico.
Clave del pago	No	Entero	Se actualiza para el tipo de pago = cpago= 100 pride; cpago= 101 = paipa.
Importe mensual de la categoría	Sí	Moneda	Es el importe que recibe la categoría a la que el académico pertenece.
Años de antigüedad	Sí	Entero	Son los años de antigüedad del docente al momento del pago.
Importe por antigüedad	Sí	Moneda	Es el importe obtenido de la nómina de la antigüedad del trabajador.
Importe por antigüedad calculado.	No	Moneda	Es el importe calculado del pago por antigüedad utilizado para estimaciones.
Registro Federal de Causantes.	Sí	Carácter (15)	Cada empleado (académico o administrativo) posee un registro federal de contribuyentes que lo identifica ante la Secretaría de Hacienda para el pago de impuestos.

Entidad: Cpride.

Definición: Es el catalogo que hace referencia a los niveles y porcentajes que se utilizan en la asignación de estímulo en el programa PRIDE Y PAIPA.

Atributos :

Nombre	Requerido	Tipo de dato	Definición
Clave pride	Sí	Carácter (2)	Cada uno de los pagos de PRIDE tienen un nivel y porcentaje definido en base a las condiciones de ingreso al programa.
Clave porcentaje	Sí	Entero	Es el porcentaje sobre su sueldo integrado que recibe por concepto de PRIDE el académico.
Clave nivel	Sí	Entero	Es el nivel A, B, C que puede tener el profesor dentro de PRIDE.

Entidad: Csubramo

Definición: Este catalogo contiene las claves y las descripciones de todos los subramos de dependencias de la UNAM.

Atributos :

Nombre	Requerido	Tipo de dato	Definición
Clave de subramo	Sí	Entero	Cada subramo es identificado mediante esta clave para ubicar a las diferentes dependencias dentro de alguno de ellos.
Descripción de subramo	Sí	Carácter (80)	Es el título completo o nombre del subramo referido.

Entidad: Csubsistema

Definición: Este catalogo contiene las claves y las descripciones de todos los subsistemas de dependencias de la UNAM.

Nombre	Requerido	Tipo de dato	Definición
Clave del subsistema	Sí	Entero	Cada subsistema es identificado mediante esta clave, para ubicar a las diferentes dependencias dentro de alguno de ellos.
Descripción del subsistema	Sí	Carácter (40)	Es el título completo o nombre del subsistema referido.

Entidad: Tpride01

Definición: Es una entidad que permite almacenar los datos de los académicos cuyo estímulo es de cuota fija y requieren ser manejados de manera independiente para cumplir con los objetivos planteados.

Nombre	Requerido	Tipo de dato	Definición
Clave del programa	Sí	Entero	Parte del código programático que distingue al programa del pago.
Clave del subprograma	Sí	Entero	Parte del código programático que distingue al subprograma del pago.
Clave de la dependencia	Sí	Entero	Parte del código programático que distingue a la dependencia del pago.
Clave de la subdependencia	Sí	Entero	Parte del código programático que distingue a la subdependencia del pago.
Clave del tipo de percepción	No	Carácter (2)	Tipo de percepción que se emite en el pago NE= No especificado, DE= Devolución, DS= Diferencia de sueldos, PD= PRIDE, PE= Percepciones, DI= Devolución de impuesto, PV= Pago de valores.

Capítulo III - Descripción del sistema y análisis

Clave de la partida	Sí	Entero	Parte del código programático que distingue a la partida del pago.
Clave de categoría nominal	Sí	Carácter (2)	Clave en letras de la categoría de la persona en el pago.
Clave de categoría tipo	Sí	Entero	Clave en números de la categoría es la segunda parte del código que distingue a las categorías dentro del tabulador.
Clave de categoría nivel	Sí	Entero	Clave del nivel de la categoría con este código se completa la clave de categoría. Para referencia se encuentran los tabuladores del personal académico o administrativo.
Clave pride	Sí	Entero	Cada uno de los pagos de pride tiene un nivel y porcentaje definido en base a las condiciones de ingreso al programa PRIDE.
Descripción de categoría	Sí	Carácter (40)	Es el título de la categoría del académico que recibe el pago.
Clave del pago	No	Entero	Se actualiza para el tipo de pago= cpago= 100 Pride; cpago= 101 = Paipa.
Registro Federal de Causantes.	Sí	Carácter (15)	Cada empleado (académico o administrativo) posee un registro federal de contribuyentes que lo identifica ante la Secretaría de Hacienda para el pago de impuesto.
Número de quincena	Sí	Entero	Es el número de la quincena actualmente en proceso.
Importe del pago.	Sí	Moneda	Es la cantidad quincenal en pesos por concepto del estímulo PRIDE.

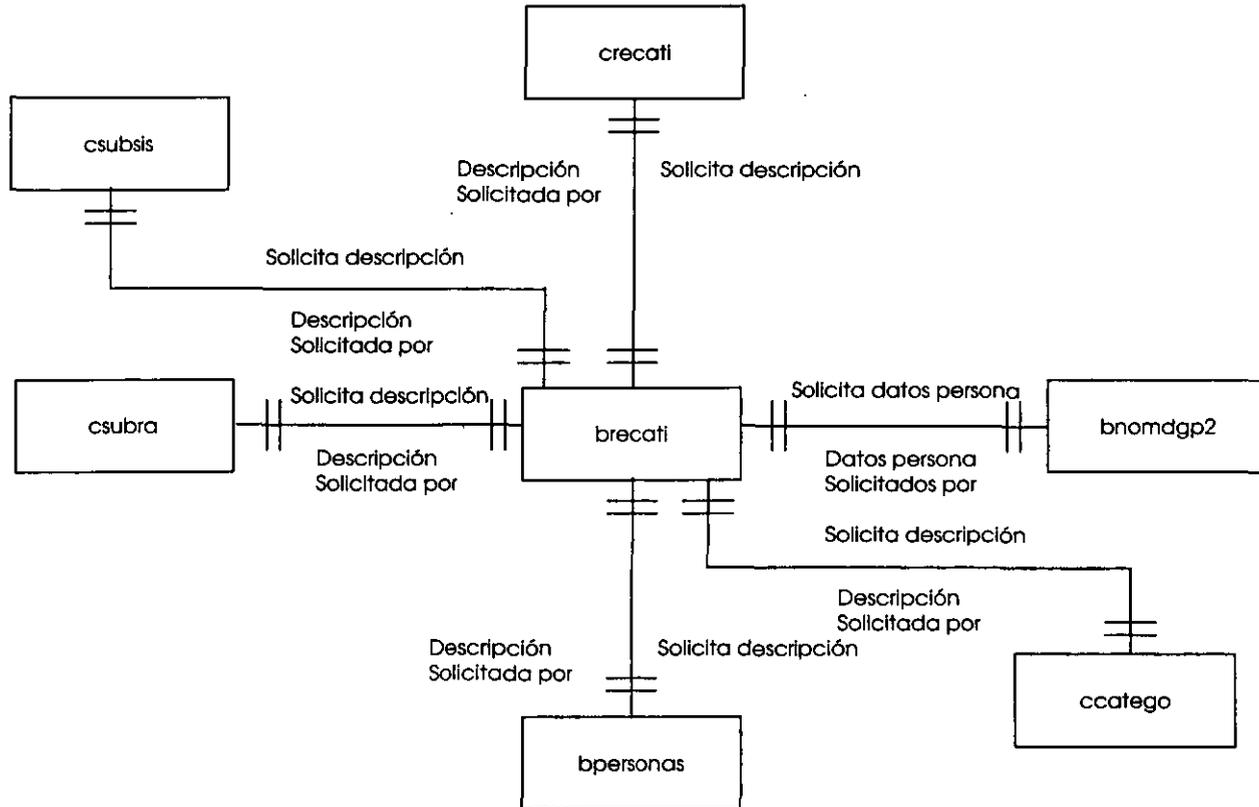
Entidad: bpersonas

Definición: Es la entidad que contiene los datos de cada una de las personas vigentes que laboran en la Universidad.

Atributos:

Nombre	Requerido	Tipo de dato	Definición
Registro federal de causantes	Sí	Carácter (15)	Cada empleado (académico o administrativo) posee un registro federal de causantes que lo identifica ante la Secretaría de Hacienda para el pago de impuestos.
Clave Sindicato	No	Entero	Es la clave que identifica el sindicato al que pertenece la persona.
Clave sexo	No	Carácter (1)	Es la clave del sexo de la persona F= Femenino, M= Masculino, X= No disponible.
Quincenas antigüedad	Sí	Entero	Número de quincenas de antigüedad en la UNAM.
Número de quincenas administrativa	No	Entero	Número de quincenas de labor administrativa de la persona en la Universidad.

MODELO DE INFORMACIÓN
PARTIDA : 356 - RECATI
MÓDULO : SIPES I
DIAGRAMA : ENTIDAD RELACIÓN
DEPARTAMENTO : DESARROLLO DE SISTEMAS
FECHA : 10 DE MARZO DE 1999



Capítulo III, Figura III.26

Capítulo III - Descripción del sistema y análisis

Clave de categoría nivel	Sí	Entero	Es la clave del nivel de la categoría dentro del tabulador.
Clave recati	Sí	Carácter (2)	Esta clave sirve para distinguir el nivel y porcentaje de estímulo del pago.
Numero de quincena	Sí	Entero	Número de la quincena actual en nómina.
Importe recati	Sí	Moneda	Es el importe quincenal del estímulo.
Descripción categoría	Sí	Carácter (40)	Es el título de la categoría del profesor que recibe el estímulo.
Nivel de estímulo	Sí	Entero	Es el nivel de recati al que pertenece el académico.
Porcentaje de estímulo	Sí	Entero	El porcentaje sobre el sueldo integrado que recibe el profesor.
Clave del pago	No	Entero	Es la clave del pago cpago= 120 = Recati.
Importe de la categoría	Sí	Moneda	Es el importe que recibe por la categoría a la que pertenece.
Años de antigüedad	Sí	Entero	Es el número de años de antigüedad del profesor.
Importe de antigüedad	Sí	Moneda	Este es el importe que recibe por antigüedad el trabajador.
Registro federal de causantes	Sí	Carácter (15)	Cada empleado (académico o administrativo) posee un registro federal de contribuyentes que lo identifica ante la Secretaría de Hacienda para el pago de impuestos.

Entidad: crecati

Descripción: Esta entidad contiene las diferentes claves que se manejan en el programa CATEDRA para asignar niveles y porcentajes a los académicos que se les otorga el estímulo.

Atributos:

Nombre	Requerido	Tipo de dato	Definición
Clave de recati	Sí	Carácter (2)	Es la clave que sirve para distinguir el nivel y el porcentaje de estímulo del pago.
Porcentaje	Sí	Entero	Es el nivel de recati al que pertenece el académico.
Nivel	Sí	Entero	Es el porcentaje sobre el sueldo integrado que recibe el profesor.

III.5.4.- SIPES II – Partida 341 Complemento al salario por calidad y eficiencia en el trabajo del personal administrativo.

III.5.4.1.- Calidad y Eficiencia.

Diagrama Entidad – Relación

Véase Figura III.27.

Entidad: bcalyefi

Definición: Es la entidad en la que se depositan los datos de las personas que reciben el pago de calidad y eficiencia en la nomina vigente al momento del proceso.

Atributos:

Nombre	Requerido	Tipo de dato	Definición
Clave del programa	Sí	Entero	Parte del código programático que distingue al programa del pago.
Clave del subprograma	Sí	Entero	Parte del código programático que distingue al subprograma del pago.
Clave de la dependencia	Sí	Entero	Parte del código programático que distingue a la dependencia del pago.
Clave de la subdependencia	Sí	Entero	Parte del código programático que distingue a la subdependencia del pago.
Clave del subsistema	Sí	Entero	Es la clave del subsistema de dependencias al que pertenece el pago.
Clave del subramo	Sí	Entero	Es la clave del subramo de dependencias al que pertenece el pago.
Clave de categoría nominal	Sí	Entero	Clave en letras de la categoría de la persona en el pago.
Clave de categoría tipo	Sí	Carácter (2)	Clave en números de la categoría y la segunda parte del código que distingue a la categoría dentro del tabulador.
Clave de categoría nivel	Sí	Entero	Con este código se completa la clave de categoría, para referencia se encuentran los tabuladores del personal académico o administrativo.
Importe categoría	Sí	Moneda	Es el pago mensual correspondiente a la categoría del pago.
Importe nómina	Sí	Moneda	Es el importe pagado por concepto del estímulo en la quincena actual.
Descripción categoría nómina	Sí	Carácter (14)	Es la descripción del pago hecho por la DGP sirve como referencia de tipo de percepción.
Registro federal de causantes	Sí	Carácter (15)	Cada empleado (académico o administrativo) posee un registro federal de contribuyentes

**ESTA TESIS NO SALE
DE LA BIBLIOTECA**

Capítulo III - Descripción del sistema y análisis

			que lo identifica ante la Secretaría de Hacienda para el pago de impuestos.
Número de la quincena	Sí	Entero	Es el número de la quincena actual.
Nivel calidad y eficiencia	Sí	Entero	Es el nivel que el trabajador ocupa en el programa.
Descripción de la categoría	Sí	Carácter (40)	Es el título de la categoría a la que pertenece el profesor objeto del estímulo.

III.5.5.- SIPES II - Partida 344 Estímulos y gratificaciones complementarias.

III.5.5.1.- EDPAC

Diagrama Entidad - Relación

Véase Figura III.28.

Entidad: bedpac.

Descripción: Es la entidad en la que se encuentran los datos de las personas que reciben el pago de calidad y eficiencia para el personal de confianza en la nómina vigente al momento del proceso.

Atributos :

Nombre	Requerido	Tipo de dato	Definición
Clave del programa	Sí	Entero	Parte del código programático que distingue al programa del pago.
Clave del subprograma	Sí	Entero	Parte del código programático que distingue al subprograma del pago.
Clave de la dependencia	Sí	Entero	Parte del código programático que distingue a la dependencia del pago.
Clave de la subdependencia	Sí	Entero	Parte del código programático que distingue a la subdependencia del pago.
Clave del subsistema	Sí	Entero	Es la clave del subsistema de dependencias al que pertenece el pago.
Clave del subramo	Sí	Entero	Es la clave del subramo de dependencias al que pertenece el pago.
Clave de categoría nominal	Sí	Entero	Clave en letras de la categoría de la persona en el pago.
Clave de categoría tipo	Sí	Carácter (2)	Clave en números de la categoría y la segunda parte del código que distingue a la categoría dentro del tabulador.
Clave de categoría nivel	Sí	Entero	Con este código se completa la clave de categoría, para referencia se encuentran los

III.6.- Conclusión del análisis

El modelo de contexto propuesto nos define el alcance del sistema en el ámbito de procesos y entidades que intervienen en el sistema, nos ha permitido distinguirlos y colocarlas en la posición adecuada, tanto al nivel expandido como reducido, en lo general del sistema y en cada módulo particular que se planea.

Estas operaciones plantean la base junto con los siguientes modelos para dar forma al sistema total y permite ver cómo se desarrollará paso a paso el sistema.

En la siguiente parte del capítulo se abordará el segundo modelo de análisis requerido, veremos como las fronteras planteadas en este modelo limitaran la manera de tomar direcciones, al estar vinculado directamente con las fases de diseño e implementación.

Queda claro que se acota completamente el alcance del sistema hasta el momento. Algo sumamente valioso para el logro de las metas, pues permite evaluar el avance conforme se crea el sistema.

Una vez concluidos los 3 modelos que forman parte del análisis tenemos una enorme cantidad de información acerca de lo que queremos obtener del sistema en diferentes niveles, por un lado el contexto en el cual afectara a los procesos existentes actualmente en la Dirección General de Programación Presupuestal delimitando perfectamente sus alcances e incluso permitiendo su expansión en caso de que sea necesario a mediano o largo plazo.

En segundo lugar gracias a el modelo de eventos ahora sabemos en que momentos durante la operación de la Dirección General de Programación Presupuestal se va a utilizar el sistema y como tras la sucesión de diferentes operaciones se dará cause a las necesidades continuas de información de la Subdirección de Planeación.

El ultimo modelo, el de información, es sumamente importante ya que nos ha permitido trabajar por primera vez durante la creación del sistema con datos previamente normalizados, esto permite su aplicación casi inmediata en la creación de la base de datos del siguiente capítulo, nos muestra además las restricciones aplicables a los datos, el conjunto de esta información permite el control durante el diseño y la implementación de variables tales como la administración de datos de entrada y de salida a la base de datos, los rangos validos de datos, la cohesión necesaria para el uso de las relaciones entre datos de las diferentes tablas que constituirán la base.

El siguiente paso una vez concluido el análisis consiste en utilizar toda esta información para establecer físicamente nuestros depósitos de datos y del sistema, para crear e implementar en el programa lo que se desea obtener.

Capítulo IV

DISEÑO.

IV.1.- Objetivo :

Describir las decisiones de implementación del sistema SIPES utilizando técnicas de decisión que satisfaga las necesidades planteadas en el análisis.

Esta fase permite crear los "planos" del sistema que se utilizaran para la creación del programa que buscamos realizar y su debida prueba de funcionamiento.

Mientras que la fase de análisis nos permite organizar todo el conocimiento sobre la forma en que debe operar el sistema y la forma en la que opera actualmente la dependencia, el diseño debe aterrizar todo este conocimiento en el mundo real para el equipo que se posee y para las situaciones reales de puesta en marcha lenguajes de programación e infraestructura tecnológica que requiere el sistema para su funcionamiento adecuado.

Nuevamente basaremos este capítulo en una serie de técnicas probadas en el área de desarrollo de sistemas que nos permite crear un estudio estructurado y lógico como fuente para el programador del sistema.

Entre las técnicas que utilizaremos se encuentran:

- a) El modelo arquitectónico.
- b) El diseño de la base de datos.
- c) El diseño de los componentes internos.

Estas técnicas tienen como premisa los modelos de contexto, eventos e información desarrollados en el capítulo anterior ya que para su implementación requieren la información previamente compilada y se convierten en el único paso previo antes de la creación del sistema.

IV.2.- Modelo Arquitectónico.

La primera fase del modelo arquitectónico se refiere a que es lo que entendemos por sistema cliente / servidor y como afecta esta definición en la creación y administración del sistema SIPES.

En principio la computación cliente / servidor es el procesamiento cooperativo de información de negocios mediante un conjunto de procesadores en donde clientes múltiples geográficamente distribuidos inician peticiones que son procesadas por uno o más servidores centrales, por ejemplo cuando se utiliza un servidor de Internet, se establece un proceso en el cual entrarán en juego dos partes por un lado el usuario ejecuta una aplicación en el ordenador local que es el denominado programa cliente. Este programa se pone en contacto con el ordenador remoto para solicitar la información deseada.

El ordenador remoto a su vez responderá al pedido realizado por el programa cliente, a través de otro programa o aplicación: el programa servidor.

Los términos cliente y servidor se usan tanto por referirse a los programas que cumplen estas funciones, como a los ordenadores donde son ejecutados dichos programas.

El programa cliente cumple 2 funciones distintas: por un lado gestiona la comunicación con el servidor, solicita un servicio y recibe los datos enviados por aquel. Por otro, maneja la interfaz con el usuario, presenta los datos en el formato adecuado y brinda las herramientas y comandos necesarios para que el usuario pueda usar las prestaciones del servidor de forma sencilla.

El programa servidor en cambio, básicamente solo tiene que encargarse de transmitir la información de manera eficiente. No tiene que atender al usuario. De esta forma un mismo servidor puede atender a varios clientes al mismo tiempo.

La arquitectura Cliente / Servidor puede dividirse en mas de 2 Niveles, hasta el momento se ha tratado el punto de vista de 2 niveles. Sin embargo la complejidad, ventajas y desventajas cambian al agregarse niveles, por ejemplo: en el caso de un sistema de 3 niveles existen diferentes solicitudes al primer nivel normalmente de aplicaciones y al segundo como servidor de datos centrales o distribuido, esta configuración requiere de un manejo mas detallado de los servidores por la dependencia de aplicaciones de un servidor distinto al principal sin embargo libera de ciertos procesos al servidor central y con lo cual se reparte la carga de trabajo en diferentes maquinas.

Continuando con esta filosofía podemos encontrar configuraciones de n-niveles que dependerán de la complejidad o necesidad del problema a resolver o del sistema deseado.

El proceso de modelo arquitectónico permite mapear los requerimientos esenciales de la fase de análisis hacia una arquitectura tecnológica específica. Esto implica un aterrizaje de la Información obtenida anteriormente para hacer la propuesta de configuración optima de los recursos de hardware para llevar a cabo las tareas necesarias y descritas en el análisis.

Este modelado involucra diferentes condiciones para formar las decisiones necesarias, entre ellas encontramos.

- a) La distribución geográfica de los requerimientos de proceso.
- b) Los componentes de hardware para las maquinas cliente.
- c) Los componentes de hardware para las maquinas servidor.
- d) La configuración y cantidad de niveles de hardware cliente / servidor.
- e) Los mecanismos y lenguajes de comunicación de la red.
- f) El sistema operativo.
- g) El paradigma de desarrollo (orientado a objetos, 4gl, 3gl, mezclado).
- h) El lenguaje de presentación.
- i) El lenguaje de codificación de fondo.
- j) El sistema de administración de base de datos.
- k) La ubicación o ubicaciones de los procesos.
- l) La ubicación o ubicaciones de los datos físicos.

En este caso particular muchas de estas decisiones ya han sido tomadas y solo serán ilustradas puesto que ya existe la infraestructura basada en la base de datos SISAP que es un Banco de información de la Dirección General de Programación Presupuestal el cual se describirá posteriormente a detalle.

Estos puntos brindan una panorámica real de los recursos y medios para obtener el sistema a nivel físico, hasta el momento no se había tocado el tema del hardware necesario para la obtención del objetivo, sin embargo esto ha sido con el propósito de tener el máximo de información sobre lo que se desea hacer y los recursos existentes hasta el momento, por lo tanto, y bajo estas premisas se considera este como el momento adecuado para llevar a cabo esta parte del diseño.

En el caso del SIPES I y II esta decisión ya ha sido aplicada antes para el SISAP (Sistema de Administración Presupuestal) que es un sistema con el que cuenta la dependencia para el manejo de la información presupuestal de la UNAM.

Bajo este esquema se utiliza un sistema cliente / servidor de 2 niveles donde varios clientes hacen peticiones a un solo servidor que centraliza el total de los datos presupuestales.

En cuanto a las capas de software que se contempla en esta configuración tenemos.

- a) La capa de presentación: la presentación de los datos del SIPES en los clientes será utilizando un GUI (Interfaz Gráfica de Usuario) bajo el sistema operativo Windows 98.
- b) La capa de administración de datos: Esta será llevada a cabo en un servidor de bases de datos SYBASE, con una base de datos relacional.

Esta parte se ha tratado de manera superficial debido a que se tratara a detalle en cada una de las divisiones de modelado arquitectónico.

Una de las características importantes del modelo cliente / servidor es sin duda en que parte recae la capa lógica más grande, por lo tanto se le da el nombre de "peso" de la capa y se divide en dos partes por un lado un cliente pesado (como es el caso de

SIPES I y II) y un servidor delegado lo cual significa que la parte importante del software del SIPES I y II se ejecuta en la maquina cliente y el servidor esta relegado a dar los datos como esclavo cuando se los solicitan o regresarlos a la base de datos cuando el cliente se lo instruye.

La decisión fue tomada en base a la existencia de clientes pesados del sistema SISAP que cubren diferentes situaciones del presupuesto universitario tales como Ejercicio, Transferencias, Previo etc.

IV.2.1.- Distribución geográfica de los requerimientos de proceso.

La dirección general de programación presupuestal se encuentra ubicada geográficamente en el circuito interior de la ciudad universitaria a espaldas de la Dirección General de la Escuela Nacional del Colegio de Ciencias y Humanidades, ocupa el primer piso del edificio y esta dividido físicamente en 4 subdirecciones, Planeación Presupuestaria, Registro y Control, Análisis y Operación Presupuestal, Cómputo; la ubicación de los servidores de datos es en el área de soporte técnico de la subdirección de cómputo los clientes se encuentran en el área de desarrollo y en la Subdirección de Planeación dentro del mismo edificio, eso facilita enormemente la situación de requerimientos puesto que el manejo de la información no implica una distribución externa de los mismos.

IV.2.2.- Los componentes de hardware para las maquinas cliente.

Debido a la multitud de configuraciones actualmente disponibles en el mercado se ha buscado reducir a solo dos configuraciones generales de los clientes como medio de estandarización.

Configuración		
Componentes	Mínima	Recomendada
Memoria	16 M	64M
Procesador	Pentium	Pentium III
Disco duro	30 Mb (libres)	50 Mb (libres)
Disqueteira	3 1/2 Pulgadas	3 1/2 Pulgadas
Tarjeta de Red	Ethemet	Ethemet
Teclado	Estándar	Estándar
Mouse	Estándar	Estándar
Monitor	Vga Color	Svga color
Sistema Operativo	Windows 95	Windows 98

Este par de configuraciones, permiten una amplia cobertura sobre marcas y medios específicos y solo es una guía de requerimiento para los encargados de instalar o proveer del hardware.

Estos servidores basan su operación en las necesidades de un sistema multiusuario y de alta disponibilidad para los diferentes usuarios del sistema. Además que el software posee una serie de características que se describirán en los siguientes incisos y que cubren las actuales y las sugeridas para el sistema en desarrollo.

IV.2.4.- Configuración y cantidad de niveles de hardware cliente - servidor.

Se utiliza una configuración cliente servidor de dos niveles con cliente - pesado, servidor - ligero donde la mayoría de procesos de datos se llevan a cabo en el cliente, lo cual no afecta de manera negativa a la velocidad de respuesta por la cercanía entre las fuentes de datos y los clientes del sistema.

El problema de esta configuración estriba en la pobre reutilización de los módulos creados bajo esta perspectiva, sin embargo también otorgan una completa independencia al programador que permite cumplir con las particularidades del sistema, tales como las existentes en el modelo de eventos.

IV.2.5.- Los mecanismos y lenguajes de comunicación de red.

Este apartado se refiere a la forma en que viajará la información de los clientes a los servidores y viceversa para ello se ha utilizado el recurso de red ethernet ya existente en la dependencia y se utilizan los protocolos TCP/IP (también conocido como protocolo Internet) y el protocolo NETBEUI (Protocolo de Windows). El primero es el nivel mas bajo de transferencia de datos entre clientes con diferentes sistemas operativos entre sí normalmente se opera a nivel de computadoras, los usuarios no tienen relación directa o requieren del protocolo TCP/IP, sin embargo el sistema operativo gestiona las llamadas a servicio y operaciones en este protocolo para permitir la operación en red (local o Internet) del sistema.

En un caso similar cae el protocolo NETBEUI aunque a diferencia de TCP/IP, este es utilizado para la comunicación entre computadoras con sistema operativo Windows de manera local para la operación de las comunicaciones entre las computadoras de la red de área local utilizando por ejemplo para impresión o intercambio de archivos, ambos protocolos son de muy bajo nivel solo utilizados por el sistema.

Además de estos dos protocolos se utiliza un protocolo de mas alto nivel conocido como protocolo sybase o bien cliente sybase esta es la capa de software que permite que el programa cliente se conecte desde la maquina cliente al servidor de base de datos y realice las transacciones adecuadas para el intercambio de Información entre ellos.

IV.2.6.- El sistema operativo.

Aquí se hicieron dos distinciones la primera hacia el sistema operativo de los clientes y la segunda hacia el de los servidores:

- a) Sistema operativo cliente elegido: Windows 98 por ser el sistema operativo más extendido para computadoras personales y es el más disponible en la dependencia y por lo tanto es el elegido para crear los clientes del sistema SIPES I y II además de ser el S. O. con que están equipadas las computadoras en las que se utilizaran los clientes y el lenguaje de programación elegido.

Por lo tanto la decisión se baso en disponibilidad, facilidad de uso, capacidad del equipo cliente y el estado del mercado actual.

- b) Sistema Operativo servidor elegido: El Solaris es una GUI (Interfaz Grafica de usuario) para el Kernel del Sistema Operativo UNIX, versión creada por SUN para sus modelos SPARC y ULTRA, es un sistema operativo multiusuario con soporte TCP/IP de alto rendimiento adecuado para el manejo de grandes volúmenes de datos y múltiples usuarios simultáneamente. Además el sistema de base de datos elegido tiene una versión muy optimizada para este sistema operativo por lo tanto se espera un optimo rendimiento en la relación Sistema Operativo / Sistema manejador de Base de Datos.

IV.2.7.- El paradigma de desarrollo.

Después de elegir los niveles del sistema (cliente pesado / servidor ligero), el proceso del paradigma es mucho menos complejo ya que al observar las características del sistema concluimos que es necesario una herramienta de desarrollo que permita el uso de lenguaje orientado a objetos (la GUI de desarrollo por ejemplo) El soporte 4GL (a través de uso de SQL (Structured Query Language) para las solicitudes de datos al DBMS), y de soporte 3gl para el desarrollo de Rutinas o el cumplimiento de reglas de negocio necesarios para la manipulación de los datos. Los siguientes incisos permiten describir cada una de las partes mencionadas a nivel de detalle.

IV.2.8.- El lenguaje de presentación y de codificación de fondo.

En este caso se llevo a la conclusión de la utilización de un mismo lenguaje en ambos casos, ya que tanto la presentación así como la codificación de las operaciones en el cliente han sido desarrolladas mediante Delphi Versión 5 de Borland entre sus características tenemos:

- a) Es una herramienta de desarrollo RAD (Rapid Application Development) que permite en un mismo paquete editar y crear el código depurarlo y hacer ejecutables.
- b) Tiene acceso a los componentes API de Win 32 (Application Program Interface) para Windows 9x.
- c) VCL: (Visual Component Library) que es la base para la presentación, es una serie de componentes que permiten el uso de los controles más comunes dentro de Windows 9x.
- d) OOD (Object Oriented Development) permite desarrollar aplicaciones con la filosofía orientada a objetos.
- e) Desarrollo en codificación de fondo con Object Pascal (una versión aumentada y moderna de Pascal Standard).

- f) Soporte de transacciones con los sistemas manejadores de Bases de Datos (DBMS) más comunes.

Existen otras características de desarrollo como componentes para Internet y desarrollo en más de 2 niveles, sin embargo, estas características básicas nos permiten dilucidar la manera en como la herramienta permite desarrollar los paradigmas orientados a objetos 4GL y 3GL respectivamente convirtiéndose en un programa de desarrollo ideal en este caso.

La decisión de utilizar Delphi fue tomada después de evaluar el desempeño de un par de herramientas más: Power Builder versión 6.5 y Visual Basic versión 6.0 los cuales poseen características similares, sin embargo, al evaluarlos se obtuvo como resultado que la conexión con el manejador de base de datos es más rápida y transparente a en segundo lugar el tiempo de compilación del código y la generación de ejecutables es considerablemente menor en Delphi.

IV.2.9.- El sistema de administración de base de datos.

El DBMS (Sistema de administración de base de datos) es el motor central de manejo de la información, el DBMS elegido es Sybase versión 11.09.2, este es un manejador de bases de datos relacional multiusuario de alto rendimiento, además soporta el standard SQL (Structurd Query Lenguaje), este es un lenguaje de administración de base de datos que permite la modificación y definición de datos y el control del acceso por parte del usuario además de permitir la consulta y recuperación de los datos. El SQL nació de los conceptos formados por el doctor Edgar Codd que dieron lugar al modelo de datos relacional. La tecnología relacional se deriva de la aplicación de principios matemáticos a la administración de bases de datos publicada primeramente por Codd en 1970.

Una base de datos relacional se compone de una colección de tablas.

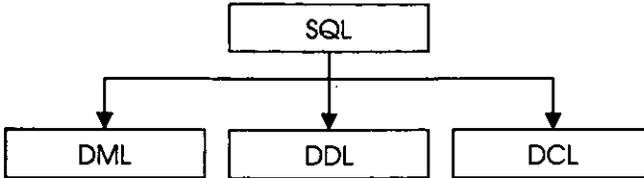
Toda la información se almacena en dichas tablas cada una es una estructura de datos bidimensional que consiste de cero o más renglones y de una o más columnas. Una de las principales características del modelo relacional es que el SQL se encarga de todos los procesos de obtención de datos de las tablas, la operación más sencilla tal como obtener los datos de una tabla se lleva a cabo por SQL este último solo requiere el nombre de la tabla para obtener los renglones que la constituyen independientemente de su organización o posición física.

El SQL se encuentra formado por 3 componentes de lenguaje:

- a) **El lenguaje de definición de datos (DDL)** define todos los objetos de la base de datos. Un objeto se define como cualquier cosa que pueda ser manipulado por SQL.
- b) **El lenguaje de control de datos (DCL)** proporciona privilegios a los usuarios potenciales de objetos SQL a través de los dos estatutos primarios GRANT Y REVOKE. Esto contrasta con los privilegios implícitos obtenidos por los usuarios sin el uso de DCL. Más comúnmente en casos donde la propiedad de un objeto SQL ha sido establecida durante su creación o cuando aplica la autoridad administrativa en general.
- c) **El lenguaje de manipulación de datos (DML)** se utiliza para recuperar, actualizar, insertar y borrar cualquier información almacenada en tablas SQL mediante cierto

lenguaje de formato libre que puede ser ejecutado interactivamente o a través de código escrito en un programa.

En la figura IV.1 encontramos los componentes SQL.



Capítulo IV, Figura IV.1 Componentes SQL

IV.2.10.- La ubicación de los procesos.

Esta parte de modelo se basa en el paradigma decidido con anterioridad al tratarse de un nivel de cliente pesado y servidor ligero la carga principal de proceso caerá sobre los clientes ubicados en diferentes lugares dentro de la dependencia. La ventaja principal de la ubicación estiba en la facilidad de transferencia de la información entre clientes y servidores sin necesidad de utilizar esquemas más complejos tales como replicación de los datos o sincronización de envió para actualización en línea.

Otra de las ventajas de la ubicación recae en la posibilidad de ser necesario soporte sobre alguno de los procesos el tiempo respuesta ante contingencias se ve reducido enormemente debido a las facilidades de traslado.

Ante la posibilidad de ser necesario el uso de la información fuera de la dependencia la utilización del protocolo TCP/IP se convierte en una gran ventaja puesto que al tener el cliente la mayoría de la lógica del sistema es teóricamente posible que desde cualquier maquina compatible se obtengan los mismos resultados utilizando la Internet como medio de comunicación.

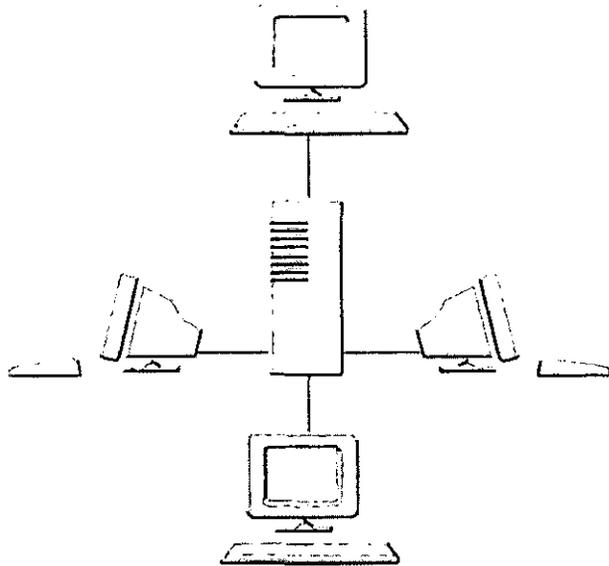
IV.2.11.- La ubicación de los datos físicos.

Tal y como se ha referido los datos serán centralizados en el manejador de base de datos en un servidor principal, estos equipos se encuentran ubicados en la Subdirección de Cómputo en el área de soporte técnico de la DGPP. El nombre de la base de datos es SISAP, en ella se encuentran almacenados los datos de los catálogos y tabuladores presupuestales la información sobre dependencias, subdependencias, ramos, subramos y demás constituyentes presupuestales de la Universidad Nacional Autónoma de México.

Se toma la decisión de centralizar los datos como medio de seguridad para la actualización constante de la información de esta manera todos los clientes del servidor poseerán los datos mas actualizados previniendo el uso de información inadecuada.

Una de las desventajas de este esquema es lo vulnerable a fallas del sistema, una política de constante respaldo y un esquema de seguridad para el acceso a la información deber ser implementado para asegurar la confiabilidad de los datos a los usuarios. Vease figura IV.2

Esquema de
centralizado de
datos



Capítulo IV, Figura IV.2 Esquema de datos centralizado

En primer lugar describiremos que es una base de datos y sus características principales.

IV.3.- Diseño de la base de datos.

IV.3.1.- Base de datos relacionales.

Una base de datos es un conjunto colección o depósito de datos almacenados en su soporte informático de acceso directo los datos deben estar interrelacionados y estructurados de acuerdo con un modelo capaz de recoger el máximo contenido semántico.

Dada la importancia que tienen en el mundo real existen además restricciones semánticas, a las que se está concediendo una importancia creciente y que en los sistemas actuales tienden a almacenarse junto con los datos al igual que ocurrió con las relaciones entre ellos.

La redundancia de los datos debe ser controlada, de forma que no existan duplicidades perjudiciales ni innecesarias (normalización) y que las redundancias físicas, convenientes muchas veces a fin de responder a objetivos de eficiencia sean tratadas por el usuario del sistema, de modo que no puedan producirse incoherencias. Esto podría resumirse diciendo que en las bases de datos no debe existir redundancia lógica aunque sí se admite cierta redundancia física por motivos de eficiencia.

Por lo tanto un dato se actualizara lógicamente por el usuario de forma única, y el sistema se preocupara de cambiar físicamente todos aquellos campos en los que el dato estuviese repetido en caso de existir redundancia física.

Las bases de datos deben servir al conjunto de la organización manejando los datos como un recurso que vienen a añadirse a los ya tradicionales, por tanto las bases de datos han de atender a múltiples usuarios y a diferentes aplicaciones en contraposición a los sistemas de archivos planos, en los que cada archivo está diseñado para responder a las necesidades de una determinada aplicación.

Otro aspecto importante de las bases de datos es la independencia, tanto física como lógica, entre datos y procesos. Esta independencia, objetivo fundamental de las bases de datos es una característica esencial que distingue a las bases de datos de los archivos planos. La definición y la descripción del conjunto de datos contenidos en la base deben ser integrados con los mismos. En los sistemas basados en archivos planos los datos se encuentran almacenados en soporte magnético, mientras su descripción (muy somera) está separada de los mismos formando parte de los programas. Suele haber, además una documentación adicional habitualmente en soporte de papel y en general insuficiente y atrasada. Este tipo de organización da origen a infinidad de problemas, ya que se presentan ocasiones en las que no se sabe cuál es la descripción de determinado archivo bien por pérdida del mismo bien por que no se ha actualizado debidamente la correspondiente documentación y tampoco se conoce exactamente el programa que lo trataba. En las bases de datos la descripción y en algunos casos también una definición y documentación completas (Meta -Datos) se almacenan junto con los datos, de manera que estos se encuentran auto documentados y cualquier cambio que se produzca se ha de reflejar y quedar recogido en el sistema con todas las ventajas que de este hecho se derivan.

Por último la actualización y recuperación en las bases de datos deben realizarse mediante procesos bien determinados, procedimientos que han de estar diseñados de modo que se mantenga la integridad seguridad y confidencialidad de la base de datos.

Después de enunciar las características de las bases de datos encontramos útil establecer la siguiente definición formal:

Base de datos: Es una colección o depósito de datos integrados con redundancia controlada y con una estructura que refleje las interrelaciones y restricciones existentes en el mundo real; los datos, que han de ser compartidos por diferentes usuarios y aplicaciones, deben mantenerse independientes de estas, y su definición y descripción únicas para cada tipo de datos han de estar almacenadas junto con los mismos.

Los procedimientos de actualización y recuperación, comunes y bien determinados habrán de ser capaces de conservar la integridad, seguridad y confidencialidad del conjunto de los datos.

En segundo lugar haremos referencia al modelo relacional.

Este modelo está encaminado a la obtención de una serie de ventajas u objetivos en el manejo de datos:

Independencia física: El modo en el que se almacenan los datos no influye en su manipulación lógica y por lo tanto los usuarios que acceden a esos datos no tienen que modificar sus programas por cambios en el almacenamiento físico.

Independencia lógica: Esto es que el añadir eliminar o modificar objetos de la base de datos no repercute en los programas y/o usuarios que están accediendo a subconjuntos parciales de los mismos (vistas, excepto en los casos donde las consultas sean de los datos en modificación).

Flexibilidad: En el sentido de poder presentar a cada usuario los datos de la forma en que este prefiera.

Uniformidad: Las estructuras lógicas de los datos presentan en aspecto uniforme lo que facilita la concepción y manipulación de la base de datos por parte de los usuarios.

Sencillez: Las características anteriores, así como unos lenguajes de usuario muy sencillos, producen como resultado que el modelo de datos relacional sea fácil de comprender y utilizar por parte del usuario final.

En el modelo relacional la relación es el elemento básico y se representa como una tabla.

En ella podemos distinguir un conjunto de columnas, denominadas atributos, que representan las propiedades de la misma y que están caracterizadas por un nombre y un conjunto de filas llamadas tuplas que son las ocurrencias de la relación. El número de filas de la relación se denomina cardinalidad, mientras que el número de columnas es el grado. Existen también dominios de donde los atributos toman sus valores.

El esquema general de las tablas se observa en la figura IV.3 :

Nombre de la tabla

Atributo 1	Atributo 2	...	Atributo n
xxxx	xxxx	...	xxxx
		...	
		...	
		...	

← Tupla 1
 ← Tupla 1
 ...
 ← Tupla n

Capítulo IV, Figura IV.3 Vista esquemática de una tabla.

La unión de los principios relacionales y los de bases de datos dan como resultado el moderno concepto DBMS, sistemas manejadores de base de datos, que es la parte principal del sistema de control y acumulación de la información dentro del SIPES I y II .

IV.3.2.- Base de datos SISAP.

Cómo se ha mencionado en apartados anteriores el sistema SIPES esta íntimamente, ligado a la base de datos SISAP la cual sirve como fuente de información de diferentes aplicaciones utilizadas actualmente en la Dirección General de Programación Presupuestal, en las aplicaciones, generación del presupuesto, operación del presupuesto, nómina, plantilla y administración. Estas aplicaciones se encuentran en funcionamiento actualmente actualizando y operando constantemente, utilizando SISAP como fuente de datos principal.

La base de datos SISAP es el resultado de la necesidad de actualización del sistema de cómputo utilizado anteriormente que respondía a la infraestructura tecnológica, tanto en equipos como en sistemas (hardware y software) de los años 70's y si bien esta plataforma de trabajo cumplió satisfactoriamente con los requerimientos de información y control planteados en su momento; simplemente llegaron al nivel de no responder adecuadamente a las expectativas presentes y futuras.

La implantación de esta base de datos corrió a cargo de empresa privada Ultrast. En un servidor UNIX y con el Servidor de base de datos Sybase, actualmente la organización, mantenimiento y actualización corre a cargo del departamento de soporte técnico de la Dirección General de Programación Presupuestal.

IV.3.3.- Tablas de la base de datos del sistema SIPES I y II.

Una vez definidos los principios de bases de datos relacionales, podemos dar paso al diseño de la base de datos en esta sección utilizaremos el modelo de información (cap 3), en el cual se asientan las entidades que forman parte del sistema de información SIPES I y II. Las tablas siguientes contienen los nombres de objeto de base de datos, los atributos de las tablas y sus características que permitirán en el momento de programar establecer las características finales de la base de datos.

Tomando como referencia el modelo de información y utilizando los principios aplicables al manejo de bases de datos relacionales a continuación haremos el desglose de las tablas que serán utilizadas en todo el sistema SIPES I y II, el orden en que se encuentran coincide completamente con el modelo de información y poseen además las reglas de validación necesarios para asegurar la confiabilidad y disposición de los datos.

La tabla contiene los siguientes datos :

- a) **Campo** : Es el nombre del campo definido.
- b) **Tipo de dato** : Es alguno de los tipos predefinidos en el manejador de base de datos para manejar la información.
- c) **Tamaño** : Es la longitud en bytes del registro según su tipo.
- d) **Validación** : Es la regla que debe de seguir el dato, es decir, los valores para los cuales el dato es valido.
- e) **Omisión** : Es el valor que adquiere el campo en caso de que no sea asignado ningún otro al llenar la tabla o crearla.

A continuación se describen las tablas y sus características:

Campo	Tipo de dato	Tamaño	Validación	Omisión
Cparti	smallint	2	1 - 999	not null
Ctejer	smallint	2	> 0	not null
Crespo	smallint	2	> 0	not null
Cgasto	smallint	2	> 0	not null
Ctpres	smallint	2	> 0	not null
Dparti	char	80		not null
Canalis	smallint	2	> 1	NULL

Campo	Tipo de dato	Tamaño	Validación	Omisión
Cllave	smallint	2	> 1	not null
Cl	Char	2		not null
Ctt	smallint	2	> 0	not null
Cnn	smallint	2	> 0	NULL
Ccateg	smallint	2	> 0	not null
Pcateg	money	8	> 0	not null
Pmatdl	money	8	>= 0	not null
Nhrscatego	smallint	2	> 0	not null
Finivi	smalldatetime	4	> 1990	not null
Dcateg	Char	40		not null
Dabrev	Char	15		not null
Dnivel	Char	8		not null
Dtiempo	char	4		not null

Campo	Tipo de dato	Tamaño	Validación	Omisión
Cprogr	smallint	2	>= 1	not null,
Csprog	Smallint	2	>= 1	not null,
Cdepen	smallint	2	1 - 999	not null,
Csdepe	smallint	2	1 - 999	not null

Capítulo IV Diseño

Ctiperc	smallint	2	> 0	not null
Cparti	smallint	2	1 - 999	not null
Cnumnom	smallint	2	>= 1	not null
Cil	Char	2		not null
Ctt	smallint	2	> 0	not null
Cnn	smallint	2	> 0	not null
Dcategn	Char	14		not null
Csnc	char	3		not null
Cpride	char	2		not null
Cpago	smallint	4	>= 0	Default 0,not null
Crfc	Char	15		not null
Ndicon	Char	1	Letra	not null
Nquinc	smallint	2	>= 1	default 0,not null
Nyear	smallint	2	> 1980	not null
Nplaza	int	6	>= 0	not null
Padant	money	8	>= 0	Default 0,not null
Pimpor	money	8	>= 0	default 0 not null
Fmovimient	smalldatetime	4		not null
Flimite	smalldatetime	4		Not null

Tabla : dbo.cdopeno

Campo	Tipo de dato	Tamaño	Validación	Omisión
Copen	smallint	2	1 - 999	not null
Csubsi	smallint	2	>= 1	not null
Csubra	smallint	2	> 0	not null
Dopen	char	85		not null

Tabla : dbo.csdepeno

Campo	Tipo de dato	Tamaño	Validación	Omisión
Cdepen	smallint	2	1 - 999	not null
Csdepe	Smallint	2	1 - 999	not null
Careco	smallint	2	>= 1	not null
Czogeno	Smallint	2	>= 1	not null
Canalist	Smallint	2	>= 1	NULL
Dsdepe	Char	80		not null

Tabla : dbo.bfomdoc

Campo	Tipo de dato	Tamaño	Validación	Omisión
Crfc	Char	15		not null
Inivel	Money	8	>= 0	not null
Cdepen	Smallint	2	1 - 999	not null
Csdepe	Smallint	2	1 - 999	not null
Cmes	Smallint	2	0 - 11	not null
Ppagado	Money	8	>= 0	not null
Cil	Char	2		not null
Ctt	Smallint	2	> 0	not null

Capítulo IV Diseño

Cnn	Smallint	2	> 0	not null
-----	----------	---	-----	----------

Tabla : dbo.tpepdep

Campo	Tipo de dato	Tamaño	Validación	Omisión
Ddep1	Char	85		not null
Ddep2	Char	6		not null
Dsdep	Char	90		not null
Dperson*	Int	6		not null
Dlmporn**	Money	8		not null

* Es el número de personas por dependencia, son varios campos distinguidos por nivel donde "n" es el número del campo

** Es el monto del importe por nivel, son varios campos donde "n" es el número del campo.

Tabla : dbo.bpride

Campo	Tipo de dato	Tamaño	Validación	Omisión
Csubsi	smallint	2	>= 1	not null
Cdepen	smallint	2	1 - 999	not null
Csdepe	smallint	2	1 - 999	not null
Csubra	smallint	2	> 0	not null
Cil	Char	2		not null
Clt	smallint	2	> 0	not null
Cnn	smallint	2	> 0	not null
Cpride	char	2		not null
Ngulnc	Smallint	2	>= 1	not null
Pmpor	Money	8	> 0	not null
Dcateg	Char	40		not null
Nivel	Smallint	2	> 0	not null
Nporcen	Smallint	2	> 0	not null
Cpago	Smallint	2	>= 0	not null
Pcateg	Money	8	> 0	not null
Nantig	Smallint	2	>= 0	not null
Pantig	Money	8	>= 0	not null
Pcantig	Money	8	>= 0	not null
Crc	Char	15		not null

Tabla : dbo.cpride

Campo	Tipo de dato	Tamaño	Validación	Omisión
Cpride	Char	2		not null
Nporcen	Smallint	2	> 0	not null
Nivel	Smallint	2	> 0	not null

Tabla : dbo.csubramo

Campo	Tipo de dato	Tamaño	Validación	Omisión
Csubra	Smallint	2	1 - 999	not null

Capítulo IV Diseño

Dsubra	Char	80		not null
--------	------	----	--	----------

Tabla : dbo.csubsis

Campo	Tipo de dato	Tamaño	Validación	Omisión
Csubsi	Smallint	2	>= 1	not null
Dsubsi	Char	40		not null

Tabla : dbo.fpride01

Campo	Tipo de dato	Tamaño	Validación	Omisión
Cprogr	Smallint	2	>= 1	not null
Csprog	Smallint	2	>= 1	not null
Cdepen	Smallint	2	1 - 999	not null
Csdepe	Smallint	2	1 - 999	not null
Ctiperc	Smallint	2	> 0	not null
Cparti	Smallint	2	1 - 999	not null
Cil	Char	2		not null
Ctt	Smallint	2	> 0	not null
Cnn	Smallint	2	> 0	not null
Cpride	Char	2		not null
Dcateg	Char	40		not null
Cpago	Smallint	2	>= 0	not null
Crfc	Char	15		not null
Nquinc	Smallint	2	>= 1	not null
Pimpor	Money	8	> 0	not null

Tabla : dbo.bpersonas

Campo	Tipo de dato	Tamaño	Validación	Omisión
Crfc	Char	15		not null
Csindi	Smallint	2	>= 1	not null
Csexo	Char	1	F o M	not null
Nqanti	Smallint	2	>= 1	not null
Nqadmi	Smallint	2	>= 1	not null
Nqacad	Smallint	2	>= 1	not null
Cipago	Int	4	>= 1	not null
Fulgna	Smalldatetime	4	> 1985	not null
Falta	Smalldatetime	4	> 1910	not null
Dnombr	Char	40		not null
Bmusico	Bit	8	S o N	not null
Bvige	Bit	8	S o N	not null

Tabla : dbo.brecati

Campo	Tipo de dato	Tamaño	Validación	Omisión
Csubsi	Smallint	2	>= 1	not null
Cdepen	Smallint	2	1 - 999	not null
Csdepe	Smallint	2	1 - 999	not null
Csubra	Smallint	2	>= 1	not null
Cil	Char	2		not null

Capítulo IV Diseño

Ctt	Smallint	2	> 0	not null
Cnn	Smallint	2	> 0	not null
Crecati	Char	2		not null
Nquinc	smallint	2	>= 1	not null
Pimpor	money	8	> 0	not null
Dcateg	char	40		not null
Nnivel	Smallint	2	> 0	not null
Nporcen	Smallint	2	>= 0	not null
Cpago	Smallint	2	>= 0	not null
Pcateg	Money	8	> 0	not null
Nanting	Smallint	2	>= 1	not null
Pantig	Money	8	> 0	not null
Crfc	Char	15		not null

Tabla : dbo.bcatyefi

Campo	Tipo de dato	Tamaño	Validación	Omisión
Cprogr	Smallint	2	>= 1	not null
Csprog	Smallint	2	>= 1	not null
Cdepen	Smallint	2	1 - 999	not null
Csdepe	Smallint	2	1 - 999	not null
Csubsi	Smallint	2	>= 1	not null
Csubra	Smallint	2	>= 1	not null
Ctt	Smallint	2	>= 1	not null
Cll	Char	2		not null
Cnn	Smallint	2	>= 1	not null
Pimpcateg	Money	8	> 0	not null
Pimpnom	Money	8	> 0	not null
Dcategn	Char	14		not null
Crfc	Char	15		not null
Nquinc	Smallint	2	>= 1	not null
Nnivel	Smallint	2	>= 1	not null
Dcateg	Char	40		not null

Tabla : dbo.bedpac

Campo	Tipo de dato	Tamaño	Validación	Omisión
Cprogr	Smallint	2	>= 1	not null
Csprog	Smallint	2	>= 1	not null
Cdepen	Smallint	2	1 - 999	not null
Csdepe	Smallint	2	1 - 999	not null
Csubsi	Smallint	2	>= 1	not null
Csubra	Smallint	2	>= 1	not null
Ctt	Smallint	2	>= 1	not null
Cll	Char	2		not null
Cnn	Smallint	2	>= 1	not null
Pimpcateg	Money	8	> 0	not null
Pimpnom	Money	8	> 0	not null
Dcategn	Char	14		not null
Crfc	Char	15		not null

IV.4.1.1.2.- Lógica de programación.

Eventos	Entidades	Acciones
Reporte por dependencia	Bdlpasi Cdepeno Csdepeo Cparto Ccatego	Evento Click en botón Reporte Selecciona todos los registros de dlpasi en Bdlpasi Ordenados por dependencia Emite reporte en pantalla Fin del evento Evento click en botón terminar Cierra la ventana dipasi Fin del evento
Reporte por persona	Bdlpasi Cdepeno Csdepeno Ccatego	Evento Click en botón Reporte Selecciona todas las personas de bdlpasi Ordenados por RFC Emite reporte en pantalla Fin del evento Click Evento Click en botón terminar Cierra la ventana Dipasi Fin del Evento Click
Reporte por nivel	Bdlpasi Cdepeno Csdepeno	Evento Click en botón Reporte Selecciona todas los registros por nivel de bdlpasi Ordenados por nivel dipasi Emite reporte en pantalla Fin del evento Click Evento Click en botón terminar Cierra la ventana Dipasi Fin del Evento Click

Además de estos eventos existe la posibilidad eventualmente de que exista mas de una quincena disponible para los reportes por lo tanto se recomienda manejar el numero de quincena en la entidad bdlpasi, y se propone la utilización de una lista de datos para cubrir esta situación.

IV.4.1.2.- FOMDOC.

IV.4.1.2.1.- Diseño de interfaz.

La interfaz propuesta para este programa es el siguiente:

Reportes Partida 351 Fomdoc:

Reporte general FOMDOC

Reporte por dependencia

Reporte por categoría

Mes del reporte

Enero
Febrero
.....

Reporte

Terminar

Capítulo IV, Figura IV.5 Interfaz sugerida partida 351

En el se incluye la posibilidad de elegir el mes del reporte planteada originalmente en el objeto bfomdoc que contiene el campo cmes y permite diferenciar los registros en diferentes meses de operación del sistema.

IV.4.1.2.2.- Lógica de programación.

Eventos	Entidades	Acciones
Reporte General	Bfomdoc Bnomdgp2	Evento Open de la ventana Selecciona los meses disponibles y llena la lista Fin del evento Open Evento Click en botón Reporte Selecciona los registros de fomdoc correspondientes al numero de mes menos 1 Selecciona y cuenta los registros por nivel de pagado Emite reporte en pantalla Fin del evento Evento click en botón terminar Cierra la ventana Fomdoc Fin del evento
Reporte por dependencia	Bfomdoc Cdepeno Csdepeno	Evento Open de la ventana Selecciona los meses disponibles y llena la lista Fin del evento Open Evento Click en botón Reporte Selecciona todas los registros de Fomdoc ordenados por dependencia Emite reporte en pantalla Fin del evento Click Evento Click en botón terminar Cierra la ventana Fomdoc Fin del Evento Click
Reporte por categoría	Bfomdoc Ccatego	Evento Open de la ventana Selecciona los meses disponibles y llena la lista Fin del evento Open Evento Click en botón Reporte Selecciona todas los registros por categoría Ordenados por Cl, Ct y Cnn Emite reporte en pantalla Fin del evento Click Evento Click en botón terminar Cierra la ventana Fomdoc Fin del Evento Click

Reporte General	Bpride Ccatego Bnomdgp2	Evento Click en botón Reporte Selecciona los registros de Paipa ordenados por categoría Emite reporte en pantalla Fin del evento Evento click en botón terminar Cierra la ventana Paipa Fin del evento
Reporte por Unidades	Bpride Cdepeno	Evento Click en botón Reporte Selecciona los registros de Paipa

Multidisciplinarias (Enep's)	Csdepeno Ccatego Bnomdgp2	agrupados por dependencia ordenados por categoría Emite reporte en pantalla Fin del evento Click Evento Click en botón terminar Cierra la ventana Paipa Fin del Evento Click
Reporte resto UNAM (sin ENEP's)	Bpride Cdepeno Csdepeno Ccatego Bnomdgp2	Evento Click en botón Reporte Selecciona los registros de Paipa agrupados por dependencias y ordenados por categoría Emite reporte en pantalla Fin del evento Click Evento Click en botón terminar Cierra la ventana Paipa Fin del Evento Click

IV.4.3.- SIPES I - Partida 356 Reconocimiento catedrático UNAM.

IV.4.3.1.- Partida 356 - CATEDRA.

IV.4.3.1.1.- Diseño de interfaz.

Reportes Partida 356 CATEDRA

Reporte por categoría Reporte

Reporte por subsistemas

Reporte por subramos Terminar

Reporte por categoría y antigüedad

Reporte por personas

Capítulo IV, Figura IV.10 Interfaz sugerida partida 356

IV.4.3.1.2.- Lógica de programación

Eventos	Entidades	Acciones
Reporte por categoría	Brecati Ccatego	Evento Click en botón Reporte Selecciona los registros de Catedra ordenados por categoría Emite reporte en pantalla Fin del evento Evento click en botón terminar

Reporte por categoría y antigüedad	Bnomdgp2	Evento Click en botón Reporte Selecciona los registros de Catedra ordenados por categoría y antigüedad Emite reporte en pantalla Fin del evento Evento click en botón terminar Cierra la ventana Catedra Fin del evento
Reporte por personas	Bnomdgp2	Evento Click en botón Reporte Selecciona los registros de Catedra ordenados por persona Emite reporte en pantalla Fin del evento Evento click en botón terminar Cierra la ventana Catedra Fin del evento

IV.4.4.- SIPES II - Partida 341 Complemento al salario por calidad y eficiencia en el trabajo del personal administrativo.

IV.4.4.1.- CALIDAD Y EFICIENCIA.

IV.4.4.1.1.- Diseño de interfaz.

Reportes Partida 341 Calidad v Eficiencia

Reporte por Cateadoría Reporte

Reporte por Subsistema v dependencia Terminar

Reporte por nivel

Reporte por Subramo v dependencia

Capítulo IV, Figura IV.11 Interfaz sugerida

IV.4.4.1.2.- Lógica de programación.

Eventos	Entidades	Acciones
Reporte por categoría	Bcalyefi Ccatago	Evento Click en botón Reporte Selecciona los registros de Calyefi ordenados por categoría Emite reporte en pantalla Fin del evento Evento click en botón terminar Cierra la ventana Calyefi Fin del evento
Reporte por subsistema y dependencia	Bcalyefi Cdepeno Csubsis	Evento Click en botón Reporte Selecciona los registros de Calyefi ordenados por subsistema y dependencia Emite reporte en pantalla Fin del evento Evento click en botón terminar Cierra la ventana Calyefi Fin del evento
Reporte por nivel de estímulo	Bcalyefi	Evento Click en botón Reporte Selecciona los registros de Calyefi ordenados por nivel Emite reporte en pantalla Fin del evento Evento click en botón terminar Cierra la ventana Calyefi Fin del evento
Reporte subramo y dependencia	Bcalyefi Csubra Cdepeno Cdepeno	Evento Click en botón Reporte Selecciona los registros de Calyefi ordenados por subramo y dependencia Emite reporte en pantalla Fin del evento Evento click en botón terminar Cierra la ventana Calyefi Fin del evento

Capítulo V

DESARROLLO E IMPLEMENTACIÓN.

V.1.- Objetivo :

Desarrollar la programación de los módulos, probar el sistema con los usuarios y describir el mantenimiento del sistema.

V.2.- Programación del sistema.

La programación del sistema se basó principalmente en los datos que fueron recopilados en la fase de análisis y en la serie de documentos y tablas que se crearon durante el diseño, estos elementos se aplicarán directamente para la creación de una serie de unidades y formas que son los pilares de la programación basada en Delphi, las siguientes tablas hacen una descripción de las formas que se utilizaron y el recuento de las unidades que involucran a cada una de las partidas y en general las que se utilizan para el funcionamiento programa.

V.2.1.- SIPES I – Partida 351 Estimulos a la productividad y al rendimiento del personal académico de asignatura y de fomento a la docencia para el personal de carrera.

V.2.1.1.- DIPASI.

V.2.1.1.1.- Formas y Unidades.

Las unidades y formas que se utilizan en el sistema son :

Módulo	Unidad	Forma	Descripción
Sipes I	Dipasi	F_dipasi	Unidad principal para el manejo de los reportes de dipasi

El siguiente es el listado de la unidad dipasi

{*****}

Modulo principal de Dipasi se utilizan los métodos y formas necesarios para el menu de Dipasi y sus reportes

Ultima modificación : 11/Febrero/2000

*****}

```
unit Dipasi;           /* Zona de Declaraciones */
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
```

ExtCtrls, quickrpt, QrCtrls, Db, DBTables, StdCtrls, Mask, DBCtrls,
ComCtrls, QRExport;

type

```
TF_dipasi = class(TForm)
  QuickRep1: TQuickRep;
  Query1: TQuery;
  QRBand1: TQRBand;
  QRImage1: TQRImage;
  QRLabel11: TQRLabel;
  QRLabel12: TQRLabel;
  QRLabel13: TQRLabel;
  QRSysData2: TQRSysData;
  QRSysData1: TQRSysData;
  QRLabel26: TQRLabel;
  QRLabel117: TQRLabel;
  QRGroup1: TQRGroup;
  QRBand3: TQRBand;
  QRExpr3: TQRExpr;
  QRExpr4: TQRExpr;
  QRShape3: TQRShape;
  QRBand4: TQRBand;
  QRLabel15: TQRLabel;
  QRGroup2: TQRGroup;
  QRBand5: TQRBand;
  QRExpr5: TQRExpr;
  QRExpr6: TQRExpr;
  QRBand6: TQRBand;
  QRShape1: TQRShape;
  QRLabel16: TQRLabel;
  QRLabel14: TQRLabel;
  QRLabel111: TQRLabel;
  QRShape2: TQRShape;
  QRLabel18: TQRLabel;
  QRLabel19: TQRLabel;
  QRLabel10: TQRLabel;
  QRSysData3: TQRSysData;
  QRBand7: TQRBand;
  QRLabel12: TQRLabel;
  QRExpr7: TQRExpr;
  QRExpr8: TQRExpr;
  QRDBText4: TQRDBText;
  QuickRep2: TQuickRep;
  QRBand2: TQRBand;
  QRImage2: TQRImage;
  QRLabel17: TQRLabel;
  QRLabel13: TQRLabel;
  QRLabel14: TQRLabel;
  QRSysData4: TQRSysData;
  QRSysData5: TQRSysData;
  QRLabel15: TQRLabel;
  QRLabel16: TQRLabel;
  Query2: TQuery;
  QRBand8: TQRBand;
  QRBand9: TQRBand;
  QRBand10: TQRBand;
  QRLabel21: TQRLabel;
  QRLabel22: TQRLabel;
  PageControl1: TPageControl;
  TabSheet1: TTabSheet;
  Button1: TButton;
  Button2: TButton;
```

```
DetailBand1: TQRBand;
QRDBText2: TQRDBText;
QRShape5: TQRShape;
TitleBand1: TQRBand;
QRExpr2: TQRExpr;
QRCSVFilter1: TQRCSVFilter;
Query3: TQuery;
Query4: TQuery;
QRLabel17: TQRLabel;
QRLabel18: TQRLabel;
QRLabel19: TQRLabel;
QRLabel20: TQRLabel;
QRLabel23: TQRLabel;
QRLabel24: TQRLabel;
QRLabel25: TQRLabel;
QRLabel27: TQRLabel;
QRDBText1: TQRDBText;
QRDBText3: TQRDBText;
QRDBText6: TQRDBText;
QRDBText7: TQRDBText;
QRDBText8: TQRDBText;
QRDBText9: TQRDBText;
QRDBText10: TQRDBText;
QRExpr1: TQRExpr;
QRDBText11: TQRDBText;
QRShape4: TQRShape;
QRLabel28: TQRLabel;
QRExpr9: TQRExpr;
QRExpr10: TQRExpr;
QRExpr11: TQRExpr;
PageFooterBand1: TQRBand;
QRSysData6: TQRSysData;
QRExpr12: TQRExpr;
QRDBText5: TQRDBText;
QRShape6: TQRShape;
QRShape7: TQRShape;
QRDBText12: TQRDBText;
QRLabel30: TQRLabel;
QRLabel31: TQRLabel;
QRDBText13: TQRDBText;
QRDBText14: TQRDBText;
QRDBText15: TQRDBText;
GroupBox1: TGroupBox;
ListBox1: TListBox;
GroupBox2: TGroupBox;
RadioButton1: TRadioButton;
RadioButton2: TRadioButton;
QRLabel29: TQRLabel;
QuickRep3: TQuickRep;
QRBand11: TQRBand;
QRImage3: TQRImage;
QRLabel32: TQRLabel;
QRLabel33: TQRLabel;
QRLabel34: TQRLabel;
QRSysData7: TQRSysData;
QRSysData8: TQRSysData;
QRLabel35: TQRLabel;
QRLabel36: TQRLabel;
QRLabel38: TQRLabel;
RadioButton3: TRadioButton;
QRBand12: TQRBand;
QRShape8: TQRShape;
```

```

QRShape9: TQRShape;
QRLabel37: TQRLabel;
QRLabel39: TQRLabel;
QRLabel40: TQRLabel;
QRBand13: TQRBand;
QRBand14: TQRBand;
QRDBText17: TQRDBText;
QRDBText18: TQRDBText;
QRDBText19: TQRDBText;
QRShape10: TQRShape;
QRLabel41: TQRLabel;
QRExpr13: TQRExpr;
QRExpr14: TQRExpr;
QRLabel42: TQRLabel;
Procedure Arma_reporte(quincena : string);
Procedure CreaReporteporPersona (quincena : string);
Procedure CreaReportepornivel(quincena : string);
procedure Button2Click(Sender: TObject);
procedure Button1Click(Sender: TObject);
procedure FormShow(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  /* Variables del procedimiento */
  F_dipasi: TF_dipasi;
  Paginas : String;
  tdcateg, tcsnc, tcrfc, tc11 : String;
  tcdepen, tcsdepe, tc1t : Integer;
  tpepasig, tpdipasi : Currency;

implementation

{$R *.DFM}

/* Procedimiento para conjuntar los datos del reporte */

Procedure TF_dipasi.Arma_reporte(quincena : string);
Begin
  QrLabel21.caption := quincena;
  If (StrToInt(QrLabel21.caption)= 1) or (StrToInt(QrLabel21.caption)= 2) then
    QrLabel22.caption := ' Enero ' ;
  If (StrToInt(QrLabel21.caption)= 3) or (StrToInt(QrLabel21.caption)= 4) then
    QrLabel22.caption := ' Febrero ' ;
  If (StrToInt(QrLabel21.caption)= 5) or (StrToInt(QrLabel21.caption)= 6) then
    QrLabel22.caption := ' Marzo ' ;
  If (StrToInt(QrLabel21.caption)= 7) or (StrToInt(QrLabel21.caption)= 8) then
    QrLabel22.caption := ' Abril ' ;
  If (StrToInt(QrLabel21.caption)= 9) or (StrToInt(QrLabel21.caption)= 10) then
    QrLabel22.caption := ' Mayo ' ;
  If (StrToInt(QrLabel21.caption)= 11) or (StrToInt(QrLabel21.caption)= 12) then
    QrLabel22.caption := ' Junio ' ;
  If (StrToInt(QrLabel21.caption)= 13) or (StrToInt(QrLabel21.caption)= 14) then
    QrLabel22.caption := ' Julio ' ;
  If (StrToInt(QrLabel21.caption)= 15) or (StrToInt(QrLabel21.caption)= 16) then
    QrLabel22.caption := ' Agosto ' ;
  If (StrToInt(QrLabel21.caption)= 17) or (StrToInt(QrLabel21.caption)= 18) then
    QrLabel22.caption := ' Septiembre ' ;
  If (StrToInt(QrLabel21.caption)= 19) or (StrToInt(QrLabel21.caption)= 20) then
    QrLabel22.caption := ' Octubre ' ;

```

```

If (StrToInt(QrLabel21.caption)= 21) or (StrToInt(QrLabel21.caption)= 22)then
    QrLabel22.caption := ' Noviembre ';
If (StrToInt(QrLabel21.caption)= 23) or (StrToInt(QrLabel21.caption)= 24)then
    QrLabel22.caption := ' Diciembre ';
Query1.Close;
Query1.Sql.Clear; /* Selección de datos para el reporte*/
Query1.Sql.Add ('select bdipasi.*, cdepen.ddepen, csdepen.dsdepe from ');
Query1.Sql.Add ('bdipasi, cdepen, csdepen where ');
Query1.Sql.Add ('(bdipasi.csdepe = csdepen.csdepe ');
Query1.Sql.Add ('and bdipasi.cdepen = csdepen.cdepen ');
Query1.Sql.Add ('and cdepen.cdepen = bdipasi.cdepen ');
Query1.Sql.Add ('and bdipasi.nquinc = '+ quincena);
Query1.Sql.Add (' order by bdipasi.cdepen, bdipasi.csdepe');
Query1.Open;
QuickRepl.Prepare;
Paginas := InttoStr(QuickRepl.QRPrinter.AvailablePages);
QuickRepl.QRPrinter.Free;
QuickRepl.QRPrinter := nil;
QrLabel10.Caption := Paginas;
QuickRepl.Preview;
Query1.Close;
End;

procedure TF_dipasi.Button2Click(Sender: TObject);
begin
    Close;
end;
procedure TF_dipasi.CreaReporteporPersona(quincena: string);
Begin
If (StrToInt(quincena)= 1) or (StrToInt(quincena)= 2) then
    QrLabel29.caption := ' Enero ';
If (StrToInt(quincena)= 3) or (StrToInt(quincena)= 4) then
    QrLabel29.caption := ' Febrero ';
If (StrToInt(quincena)= 5) or (StrToInt(quincena)= 6)then
    QrLabel29.caption := ' Marzo ';
If (StrToInt(quincena)= 7) or (StrToInt(quincena)= 8)then
    QrLabel29.caption := ' Abril ';
If (StrToInt(quincena)= 9) or (StrToInt(quincena)= 10)then
    QrLabel29.caption := ' Mayo ';
If (StrToInt(quincena)= 11) or (StrToInt(quincena)= 12)then
    QrLabel29.caption := ' Junio ';
If (StrToInt(quincena)= 13) or (StrToInt(quincena)= 14)then
    QrLabel29.caption := ' Julio ';
If (StrToInt(quincena)= 15) or (StrToInt(quincena)= 16)then
    QrLabel29.caption := ' Agosto ';
If (StrToInt(quincena)= 17) or (StrToInt(quincena)= 18)then
    QrLabel29.caption := ' Septiembre ';
If (StrToInt(quincena)= 19) or (StrToInt(quincena)= 20)then
    QrLabel29.caption := ' Octubre ';
If (StrToInt(quincena)= 21) or (StrToInt(quincena)= 22)then
    QrLabel29.caption := ' Noviembre ';
If (StrToInt(quincena)= 23) or (StrToInt(quincena)= 24)then
    QrLabel29.caption := ' Diciembre ';

Query2.Close;
Query2.SQL.Clear;
Query2.SQL.Add('Select t1.*, t2.ddepen, t3.dnombr from bdipasi t1, cdepen
t2, bpersonas t3 where t1.cdepen = t2.cdepen and t1.crfc = t3.crfc and t1.nquinc
= '+ quincena +' order by t1.crfc ');
Query2.Open;
QuickRep2.Preview;
Query2.Close;

```

```

End;
                                /* Reporte por Nivel */
procedure TF_dipasi.CreaReportePorNivel(quincena: string);
Begin
If (StrToInt(quincena)= 1) or (StrToInt(quincena)= 2) then
    QRLabel38.caption := ' Enero ';
If (StrToInt(quincena)= 3) or (StrToInt(quincena)= 4) then
    QRLabel38.caption := ' Febrero ';
If (StrToInt(quincena)= 5) or (StrToInt(quincena)= 6) then
    QRLabel38.caption := ' Marzo ';
If (StrToInt(quincena)= 7) or (StrToInt(quincena)= 8) then
    QRLabel38.caption := ' Abril ';
If (StrToInt(quincena)= 9) or (StrToInt(quincena)= 10) then
    QRLabel38.caption := ' Mayo ';
If (StrToInt(quincena)= 11) or (StrToInt(quincena)= 12) then
    QRLabel38.caption := ' Junio ';
If (StrToInt(quincena)= 13) or (StrToInt(quincena)= 14) then
    QRLabel38.caption := ' Julio ';
If (StrToInt(quincena)= 15) or (StrToInt(quincena)= 16) then
    QRLabel38.caption := ' Agosto ';
If (StrToInt(quincena)= 17) or (StrToInt(quincena)= 18) then
    QRLabel38.caption := ' Septiembre ';
If (StrToInt(quincena)= 19) or (StrToInt(quincena)= 20) then
    QRLabel38.caption := ' Octubre ';
If (StrToInt(quincena)= 21) or (StrToInt(quincena)= 22) then
    QRLabel29.caption := ' Noviembre ';
If (StrToInt(quincena)= 23) or (StrToInt(quincena)= 24) then
    QRLabel29.caption := ' Diciembre ';
QrLabel42.caption := quincena;
Query2.Close;
Query2.SQL.Clear;
Query2.SQL.Add(' Select count (crfc), sum (pimpdip), nnivel from bdipasi
where nquinc = '+ quincena +' group by nnivel ');
Query2.Open;
QuickRep3.Preview;
Query2.Close;
End;
                                /* Método Click del boton Reporte */
procedure TF_dipasi.Button1Click(Sender: TObject);
var quincena : String;
begin
//1) chequea que quincena
//2) chequea que reporte

If Listbox1.Itemindex >= 0 then
begin
quincena := listbox1.Items[listbox1.Itemindex];

If Radiobutton1.Checked then
Begin
Query2.Close;
Query2.Sql.Clear;
Query2.Sql.Add('Select count(*) from bdipasi where nquinc = '+quincena);
Query2.Open;
If Query2.FieldByName('Column1').AsInteger > 0 Then
    Arma_reporte(quincena)
else
    MessageDlg('No hay datos para reportar',mtWarning,mbOkCancel,0);
Query2.Close;
end;

If Radiobutton2.Checked then

```

```

Begin
  Query2.Close;
  Query2.Sql.Clear;
  Query2.Sql.Add('Select count(*) from bdipasi where nquinc = '+ quincena);
  Query2.Open;
  If Query2.FieldByName('Column1').AsInteger > 0 Then
    CreaReporteporPersona(quincena)
  else
    MessageDlg('No hay datos para reportar',mtWarning,mbOkCancel,0);
  Query2.Close;
End;

If Radiobutton3.Checked then
Begin
  Query2.Close;
  Query2.Sql.Clear;
  Query2.Sql.Add('Select count(*) from bdipasi where nquinc = '+ quincena);
  Query2.Open;
  If Query2.FieldByName('Column1').AsInteger > 0 Then
    CreaReportepornivel(quincena)
  else
    MessageDlg('No hay datos para reportar',mtWarning,mbOkCancel,0);
  Query2.Close;
End;
end
else
  begin
    MessageDlg('Debera Elegir una quincena antes de continuar',
      mtWarning, {mbOk},0);
  end;
end;

procedure TF_dipasi.FormShow(Sender: TObject);
var num : Integer;
begin
  ListBox1.Clear;
  Query4.Close;
  Query4.SQL.Clear;
  Query4.SQL.Add('Select count(*) from bdipasi');
  Query4.Open;
  num := Query4.FieldbyName('Column1').AsInteger;
  Query4.Close;
  Query4.SQL.Clear;
  Query4.SQL.Add('Select distinct nquinc from bdipasi order by nquinc');
  Query4.Open;
  If num > 0 then
  begin
    Query4.DisableControls;
    try
      Query4.First;
      while not Query4.EOF do
      begin
        Listbox1.Items.Add(Query4.FieldbyName('nquinc').AsString);
        Query4.Next;
      end;
    finally
      end;
      Query4.EnableControls;
      Query4.Close;
    end;
  end;
end;
end;

```

end.

V.2.1.2.- Partida 351 - FOMDOC.

V.2.1.2.1.- Formas y Unidades.

Módulo	Unidad	Forma	Descripción
Sipes I	Fomdocr	F_fomdoc	Unidad principal para el manejo de los reportes de fomdoc

El estado de la unidad Fomdocr es el siguiente

```

(*****
Modulo principal de Fomdoc se utilizan los métodos
y formas necesarios para el menu de Fomdoc y sus reportes

Ultima modificación : 10/Septiembre/99
*****)

unit Fomdocr;

interface

uses
    Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
    QrCtrls, quickrpt, Db, DBTables, ExtCtrls, StdCtrls, DBCtrls, ComCtrls,
    QRExport;

type
    TF_fomdoc = class(TForm)
        QuickRep1: TQuickRep;
        QRBand1: TQRBand;
        QRImage1: TQRImage;
        QRLabel1: TQRLabel;
        QRLabel2: TQRLabel;
        QRLabel3: TQRLabel;
        QRSysData2: TQRSysData;
        QRSysData1: TQRSysData;
        QRBand2: TQRBand;
        QRLabel12: TQRLabel;
        Query1: TQuery;
        QRShape2: TQRShape;
        QRLabel4: TQRLabel;
        QRLabel5: TQRLabel;
        QRLabel6: TQRLabel;
        QRLabel25: TQRLabel;
        QRShape1: TQRShape;
        QRLabel14: TQRLabel;
        QRLabel26: TQRLabel;
        QRGroup1: TQRGroup;
        QRGroup2: TQRGroup;
        QRGroup3: TQRGroup;
        QRBand3: TQRBand;
        QRExp4: TQRExp;
    end;

```

```
QRExpr5: TQRExpr;
QRExpr6: TQRExpr;
QRExpr7: TQRExpr;
QRExpr8: TQRExpr;
QRExpr9: TQRExpr;
QRLabel7: TQRLabel;
QRLabel8: TQRLabel;
QRBand6: TQRBand;
QRLabel9: TQRLabel;
QRExpr10: TQRExpr;
QRExpr11: TQRExpr;
QRExpr12: TQRExpr;
QRBand7: TQRBand;
QRLabel10: TQRLabel;
QRSysData3: TQRSysData;
QRGroup4: TQRGroup;
QRDBText2: TQRDBText;
QRDBText5: TQRDBText;
QRDBText4: TQRDBText;
QRDBText1: TQRDBText;
QRDBText6: TQRDBText;
QRDBText7: TQRDBText;
QRExpr2: TQRExpr;
QRExpr3: TQRExpr;
QRExpr1: TQRExpr;
QRDBText3: TQRDBText;
QrBand8: TQRBand;
QrBand5: TQRBand;
QrBand4: TQRBand;
QRExpr13: TQRExpr;
QRExpr14: TQRExpr;
QRExpr15: TQRExpr;
QRLabel13: TQRLabel;
QuickRep2: TQuickRep;
Query2: TQuery;
QRBand9: TQRBand;
QRImage2: TQRImage;
QRLabel15: TQRLabel;
QRLabel16: TQRLabel;
QRLabel17: TQRLabel;
QRSysData4: TQRSysData;
QRSysData5: TQRSysData;
QRLabel18: TQRLabel;
QRLabel19: TQRLabel;
QRBand10: TQRBand;
QRLabel20: TQRLabel;
QRShape3: TQRShape;
QRLabel21: TQRLabel;
QRLabel22: TQRLabel;
QRLabel23: TQRLabel;
QRLabel24: TQRLabel;
QRShape4: TQRShape;
QRLabel27: TQRLabel;
QRBand11: TQRBand;
QRBand12: TQRBand;
QRDBText8: TQRDBText;
QRDBText9: TQRDBText;
QRExpr16: TQRExpr;
QRExpr17: TQRExpr;
QRExpr18: TQRExpr;
QRDBText10: TQRDBText;
QRDBText11: TQRDBText;
```

```
QRBand13: TQRBand;
QRExpr19: TQRExpr;
QRExpr20: TQRExpr;
QRExpr21: TQRExpr;
QRBand14: TQRBand;
QRExpr22: TQRExpr;
QRExpr23: TQRExpr;
QRExpr24: TQRExpr;
QRShape5: TQRShape;
QRShape7: TQRShape;
QRLabel128: TQRLabel;
QRLabel129: TQRLabel;
QRShape6: TQRShape;
QRShape8: TQRShape;
QRShape9: TQRShape;
QRShape10: TQRShape;
QRLabel130: TQRLabel;
QRSysData6: TQRSysData;
QRLabel132: TQRLabel;
Query3: TQuery;
Query4: TQuery;
QRLabel133: TQRLabel;
QRLabel134: TQRLabel;
QRLabel135: TQRLabel;
QRLabel136: TQRLabel;
QRLabel137: TQRLabel;
QRLabel111: TQRLabel;
PageControl1: TPageControl;
TabSheet1: TTabSheet;
GroupBox1: TGroupBox;
RadioButton1: TRadioButton;
RadioButton2: TRadioButton;
RadioButton3: TRadioButton;
GroupBox2: TGroupBox;
ListBox1: TListBox;
GroupBox3: TGroupBox;
CheckBox1: TCheckBox;
QuickRep3: TQuickRep;
QRBand15: TQRBand;
QRLabel131: TQRLabel;
QRLabel138: TQRLabel;
QRLabel139: TQRLabel;
QRLabel140: TQRLabel;
QRLabel141: TQRLabel;
QRImage3: TQRImage;
QRSysData7: TQRSysData;
QRSysData8: TQRSysData;
QRBand16: TQRBand;
QRShape11: TQRShape;
QRShape12: TQRShape;
QRLabel142: TQRLabel;
QRLabel143: TQRLabel;
QRLabel144: TQRLabel;
QRLabel145: TQRLabel;
QRGroup8: TQRGroup;
QRBand17: TQRBand;
QRDBText12: TQRDBText;
QRExpr25: TQRExpr;
QRExpr26: TQRExpr;
QRExpr27: TQRExpr;
QRBand18: TQRBand;
QRLabel147: TQRLabel;
```

```

QRLabel48: TQRLabel;
QRSysData9: TQRSysData;
QRLabel49: TQRLabel;
QRBand19: TQRBand;
QRExpr28: TQRExpr;
QRExpr29: TQRExpr;
QRLabel46: TQRLabel;
QRExpr30: TQRExpr;
QRShapel3: TQRShape;
DetailBand1: TQRBand;
Button1: TButton;
Button2: TButton;
DetailBand2: TQRBand;
DetailBand3: TQRBand;
QRHTMLFilter1: TQRHTMLFilter;
procedure Button2Click(Sender: TObject);
procedure FormShow(Sender: TObject);
procedure Button1Click(Sender: TObject);
private
  ( Private declarations )
public
  ( Public declarations )
end;

var
  F_fomdoc: TF_fomdoc;
  SqlFinal : String;
implementation

($R *.DFM)

      /* Método click en el boton terminar*/
procedure TF_fomdoc.Button2Click(Sender: TObject);
begin
  Close;
end;

      /* Método que muestra la ventana FOMDOC */
procedure TF_fomdoc.FormShow(Sender: TObject);
Var
  num : integer;
begin
  Listbox1.ItemIndex := -1;
  If Listbox1.items.count <= 0 then
  begin
    SqlFinal := 'Select count(*) from bfomdoc';
    Query3.Close;
    Query3.SQL.Clear;
    Query3.SQL.Add(Sqlfinal);
    Query3.Open;
    num := Query3.FieldByName('Column1').AsInteger;
    Query3.Close;
    Query3.SQL.Clear;
    Query3.SQL.Add('Select distinct cmes from bfomdoc order by cmes');
    Query3.Open;
    If num > 0 then begin
      Query3.DisableControls;
      try
        Query3.First;
        while not Query3.EOF do
        begin
          If Query3.FieldByName('cmes').AsString = '0' then
            Listbox1.Items.Add('Enero');
        end;
      finally
        Query3.Next;
      end;
    end;
  end;
end;

```

```

If Query3.FieldByName('cmes').Asstring = '1' then
Listbox1.Items.Add('Febrero');
If Query3.FieldByName('cmes').Asstring = '2' then
Listbox1.Items.Add('Marzo');
If Query3.FieldByName('cmes').Asstring = '3' then
Listbox1.Items.Add('Abril');
If Query3.FieldByName('cmes').Asstring = '4' then
Listbox1.Items.Add('Mayo');
If Query3.FieldByName('cmes').Asstring = '5' then
Listbox1.Items.Add('Junio');
If Query3.FieldByName('cmes').Asstring = '6' then
Listbox1.Items.Add('Julio');
If Query3.FieldByName('cmes').Asstring = '7' then
Listbox1.Items.Add('Agosto');
If Query3.FieldByName('cmes').Asstring = '8' then
Listbox1.Items.Add('Septiembre');
If Query3.FieldByName('cmes').Asstring = '9' then
Listbox1.Items.Add('Octubre');
If Query3.FieldByName('cmes').Asstring = '10' then
Listbox1.Items.Add('Noviembre');
If Query3.FieldByName('cmes').Asstring = '11' then
Listbox1.Items.Add('Diciembre');
Query3.Next;
end;
finally
end;
Query3.EnableControls;
Query3.Close;
end;
end;
end;

```

```

procedure TF_fomdoc.Button1Click(Sender: TObject);
Var
tmes, titmes, tcmetodo, Paginas : String;
begin
tmes := '';
If Listbox1.Itemindex >= 0 then
begin
str (listbox1.Itemindex,tmes);
titmes := listbox1.Items[listbox1.Itemindex];
If titmes = 'Enero' then tmes := '0';
If titmes = 'Febrero' then tmes := '1';
If titmes = 'Marzo' then tmes := '2';
If titmes = 'Abril' then tmes := '3';
If titmes = 'Mayo' then tmes := '4';
If titmes = 'Junio' then tmes := '5';
If titmes = 'Julio' then tmes := '6';
If titmes = 'Agosto' then tmes := '7';
If titmes = 'Septiembre' then tmes := '8';
If titmes = 'Octubre' then tmes := '9';
If titmes = 'Noviembre' then tmes := '10';
If titmes = 'Diciembre' then tmes := '11';
If checkbox1.checked = false then
tcmetodo := intostr(1)
else
tcmetodo := intostr(2);
If RadioButton1.Checked then
Begin
Query4.Close;
Query4.SQL.Clear;

```

```

    Query4.SQL.Add('Select * from bfomdoc where cmes = '+tmes+' and cmetodo
    ='+tcmetodo+' order by pnivel');
    Query4.Open;
    QrLabel41.Caption := titmes;
    QuickRep3.Prepare;
    Paginas := InttoStr(QuickRep3.QRPrinter.AvailablePages);
    QuickRep3.QRPrinter.Free;
    QuickRep3.QRPrinter := nil;
    QrLabel47.Caption := Paginas;
    QuickRep3.Preview;
end;
If RadioButton2.Checked then
Begin
    Query1.Close;
    Query1.SQL.Clear;
    Query1.SQL.Add('Select
bfomdoc.*, cdepeno.ddepen, csdepeno.dsdepe, cdepeno.csubsi, ');
    Query1.SQL.Add('csubsis.dsubsi from bfomdoc, cdepeno, csdepeno, csubsis
where (bfomdoc.cdepen = cdepeno.cdepen)');
    Query1.SQL.Add('and (bfomdoc.cdepen = csdepeno.cdepen) and
(bfomdoc.csdepe = csdepeno.csdepe) and (cdepeno.csubsi = csubsis.csubsi) and
bfomdoc.cmes = '+tmes+' and cmetodo ='+tcmetodo+' order by cdepeno.csubsi,
bfomdoc.cdepen, bfomdoc.csdepe, bfomdoc.pnivel');
    Query1.Open;
    QrLabel32.Caption := titmes;
    QuickRep1.Prepare;
    Paginas := InttoStr(QuickRep1.QRPrinter.AvailablePages);
    QuickRep1.QRPrinter.Free;
    QuickRep1.QRPrinter := nil;
    QrLabel35.Caption := Paginas;
    QuickRep1.Preview;
end;
If RadioButton3.Checked then
Begin
    Query2.Close;
    Query2.SQL.Clear;
    Query2.SQL.Add('Select bfomdoc.*, ccatego.dcateg from bfomdoc, ccatego');
    Query2.SQL.Add('where bfomdoc.c11 = ccatego.c11 and bfomdoc.ctt =
ccatego.ctt and bfomdoc.cnn = ccatego.cnn and bfomdoc.cmes = '+tmes+' and
cmetodo ='+tcmetodo);
    Query2.SQL.Add('order by bfomdoc.c11, bfomdoc.ctt, bfomdoc.pnivel');
    Query2.Open;
    QrLabel33.Caption := titmes;
    QuickRep2.Prepare;
    Paginas := InttoStr(QuickRep2.QRPrinter.AvailablePages);
    QuickRep2.QRPrinter.Free;
    QuickRep2.QRPrinter := nil;
    QrLabel37.Caption := Paginas;
    QuickRep2.Preview;
end;
end
else
begin
    MessageDlg('Debera Elegir un mes antes de continuar',
    mtWarning, [mbOk], 0);
end;
end;
end.

```

V.2.1.3.- Partida 351 – PEPASIG.

V.2.1.3.1.- Formas y Unidades.

Módulo	Unidad	Forma	Descripción
Sipes I	Mreppepasig	F_rpepasig	Unidad con el menú de reportes de pepasig
	Pepasigr1	F_r1pepasig	Unidad para la creación del reporte por general por rango, nivel e importe
	Pepasigr4	F_r4pepasig	Unidad para la creación del reporte por nivel de horas
	Pepasigr5	F_pepasig	Unidad para el manejo del reporte por dependencias

Nota: El orden de las unidades en este caso no es consecutivo debido a que se eliminaron un par de reportes previos que los usuarios consideraron su discontinuación.

El listado de la unidad principal se encuentra a continuación :

```

{*****}

Módulo principal de reportes PEPASIG, Se realiza la
elección del reporte a realizar y posteriormente se arma
cada uno de ellos.

Ultima modificación : 25/Noviembre/2000

*****}

unit MRepPepasig;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  StdCtrls, ExtCtrls, Db, DBTables, DBCtrls, ComCtrls;

type
  TF_rpepasig = class(TForm)
    Query1: TQuery;
    PageControl1: TPageControl;
    TabSheet1: TTabSheet;
    GroupBox1: TGroupBox;
    RadioButton1: TRadioButton;
    RadioButton2: TRadioButton;
    RadioButton3: TRadioButton;
    Button2: TButton;
    Button3: TButton;
    procedure Button3Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
  private

```

```

    ( Private declarations )
public
    ( Public declarations )
end;

var
    F_rpepasig: TF_rpepasig;

implementation

uses PepasigR1, PepasigR4, pepasigR5;
{$R *.DFM}

procedure TF_rpepasig.Button3Click(Sender: TObject);
begin
    Close;
end;

procedure TF_rpepasig.Button2Click(Sender: TObject);
begin
    If Radiobutton1.checked then
        Begin
            Query1.Close;
            Query1.Sql.Clear;
            Query1.Sql.Add('Select count(*) from bnomdgp2 where cparti = 351');
            Query1.Open;
            If Query1.FieldName('Column1').AsInteger > 0 Then
                F_r1pepasig.Crea_repo
            else
                MessageDlg('No hay datos para reportar',mtWarning,mbOkCancel,0);
            Query1.Close;
        end;
    If Radiobutton2.checked then
        Begin
            Query1.Close;
            Query1.Sql.Clear;
            Query1.Sql.Add('Select count(*) from bnomdgp2 where cparti = 351');
            Query1.Open;
            If Query1.FieldName('Column1').AsInteger > 0 Then
                F_r4pepasig.Crea_repo
            else
                MessageDlg('No hay datos para reportar',mtWarning,mbOkCancel,0);
            Query1.Close;
        end;
    If Radiobutton3.checked then
        Begin
            Query1.Close;
            Query1.Sql.Clear;
            Query1.Sql.Add('Select count(*) from bnomdgp2 where cparti = 351');
            Query1.Open;
            If Query1.FieldName('Column1').AsInteger > 0 Then
                F_pepasig.Showmodal
            else
                MessageDlg('No hay datos para reportar',mtWarning,mbOkCancel,0);
            Query1.Close;
        end;
end;
end;
End.

```

V.2.2.- SIPES I – Partida 355 Prima de desempeño al personal Académico.

V.2.2.1.- PRIDE Y PAIPA.

V.2.2.1.1.- Formas y Unidades.

Módulo	Unidad	Forma	Descripción
Sipes I	Prider	F_pride	Esta unidad contiene los reportes que se requieren para la operación del PRIDE
	Paipar	F_paipa	Unidad para la operación de los reportes de PAIPA

{*****

Modulo principal de reportes PRIDE. Se realiza la elección del reporte a realizar y posteriormente se arma cada uno de ellos.

Ultima modificación : 28/Mayo/99

*****}

unit Prider1;

interface

uses

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs, QrCtrls, quickrpt, ExtCtrls, Db, DBTables, StdCtrls, ComCtrls;

type

```
TF_pride = class(TForm)
  QuickRep1: TQuickRep;
  QRBand1: TQRBand;
  QRLabel1: TQRLabel;
  QRLabel2: TQRLabel;
  QRLabel3: TQRLabel;
  QRSysData1: TQRSysData;
  QRSysData2: TQRSysData;
  QRLabel4: TQRLabel;
  QRImage1: TQRImage;
  QRLabel49: TQRLabel;
  Query1: TQuery;
  QRBand2: TQRBand;
  QRLabel5: TQRLabel;
  QRDBText6: TQRDBText;
  QRLabel7: TQRLabel;
  QRLabel8: TQRLabel;
  QRLabel9: TQRLabel;
  QRLabel10: TQRLabel;
  QRShape2: TQRShape;
  QRGroup1: TQRGroup;
  QRGroup2: TQRGroup;
  QRGroup3: TQRGroup;
  QRGroup4: TQRGroup;
```

```
QRLabel6: TQRLabel;
QRBand3: TQRBand;
QRBand4: TQRBand;
QRBand5: TQRBand;
QRBand6: TQRBand;
QRLabel12: TQRLabel;
QRExpr2: TQRExpr;
QRExpr4: TQRExpr;
QRShape1: TQRShape;
QRExpr1: TQRExpr;
QRExpr3: TQRExpr;
QRBand7: TQRBand;
QRExpr5: TQRExpr;
QRLabel11: TQRLabel;
QRExpr6: TQRExpr;
QRShape3: TQRShape;
QRBand8: TQRBand;
QRLabel13: TQRLabel;
QRSysData3: TQRSysData;
QRShape4: TQRShape;
QRDBText7: TQRDBText;
QRDBText8: TQRDBText;
QRDBText1: TQRDBText;
QRDBText2: TQRDBText;
QRDBText3: TQRDBText;
QuickRep2: TQuickRep;
QRBand9: TQRBand;
QRLabel14: TQRLabel;
QRLabel15: TQRLabel;
QRLabel16: TQRLabel;
QRSysData4: TQRSysData;
QRSysData5: TQRSysData;
QRLabel17: TQRLabel;
QRImage2: TQRImage;
QRLabel18: TQRLabel;
QRDBText4: TQRDBText;
ColumnHeaderBand1: TQRBand;
QRLabel19: TQRLabel;
QRLabel20: TQRLabel;
QRLabel21: TQRLabel;
QRLabel22: TQRLabel;
QRLabel23: TQRLabel;
QRLabel24: TQRLabel;
QRLabel25: TQRLabel;
QRLabel26: TQRLabel;
QRShape5: TQRShape;
QRShape6: TQRShape;
QRGroup5: TQRGroup;
QRDBText5: TQRDBText;
QRDBText9: TQRDBText;
QRGroup6: TQRGroup;
QRBand10: TQRBand;
QRLabel27: TQRLabel;
QRExpr7: TQRExpr;
QRExpr8: TQRExpr;
QRShape7: TQRShape;
QRGroup7: TQRGroup;
QRBand11: TQRBand;
QRDBText10: TQRDBText;
QRDBText11: TQRDBText;
QRDBText12: TQRDBText;
QRDBText13: TQRDBText;
```

```
QRDBText14: TQRDBText;
QRExpr9: TQRExpr;
QRExpr10: TQRExpr;
QRShape8: TQRShape;
SummaryBand1: TQRBand;
QRShape9: TQRShape;
QRLabel128: TQRLabel;
QRExpr11: TQRExpr;
QRExpr12: TQRExpr;
QRBand12: TQRBand;
QRLabel129: TQRLabel;
QRSysData6: TQRSysData;
Query2: TQuery;
Query3: TQuery;
QuickRep3: TQuickRep;
QRBand13: TQRBand;
QRLabel130: TQRLabel;
QRLabel131: TQRLabel;
QRLabel132: TQRLabel;
QRSysData7: TQRSysData;
QRSysData8: TQRSysData;
QRLabel133: TQRLabel;
QRImage3: TQRImage;
QRLabel134: TQRLabel;
QRDBText15: TQRDBText;
QRGroup8: TQRGroup;
QRDBText16: TQRDBText;
QRDBText17: TQRDBText;
QRGroup9: TQRGroup;
QRGroup10: TQRGroup;
QRDBText18: TQRDBText;
QRDBText19: TQRDBText;
QRDBText20: TQRDBText;
QRDBText21: TQRDBText;
QRGroup11: TQRGroup;
QRGroup12: TQRGroup;
QRDBText22: TQRDBText;
QRDBText23: TQRDBText;
QRDBText24: TQRDBText;
QRGroup13: TQRGroup;
QRGroup14: TQRGroup;
QRBand14: TQRBand;
QRDBText25: TQRDBText;
QRDBText26: TQRDBText;
QRExpr13: TQRExpr;
QRExpr14: TQRExpr;
QRShape10: TQRShape;
QRBand15: TQRBand;
QRLabel135: TQRLabel;
QRExpr15: TQRExpr;
QRExpr16: TQRExpr;
QRShape11: TQRShape;
QRBand16: TQRBand;
QRLabel136: TQRLabel;
QRExpr17: TQRExpr;
QRExpr18: TQRExpr;
QRShape12: TQRShape;
QRBand17: TQRBand;
QRLabel137: TQRLabel;
QRExpr19: TQRExpr;
QRExpr20: TQRExpr;
QRShape13: TQRShape;
```

```
QRBand18: TQRBand;
QRLabel138: TQRLabel;
QRExpr21: TQRExpr;
QRExpr22: TQRExpr;
QRShape14: TQRShape;
QRBand19: TQRBand;
QRBand20: TQRBand;
QRLabel140: TQRLabel;
QRLabel141: TQRLabel;
QRLabel142: TQRLabel;
QRLabel143: TQRLabel;
QRLabel144: TQRLabel;
QRLabel145: TQRLabel;
QRLabel146: TQRLabel;
QRShape15: TQRShape;
QRShape16: TQRShape;
QRShape17: TQRShape;
QRLabel147: TQRLabel;
QRLabel148: TQRLabel;
QRLabel150: TQRLabel;
QRLabel151: TQRLabel;
QuickRep4: TQuickRep;
QRBand21: TQRBand;
QRLabel154: TQRLabel;
QRLabel155: TQRLabel;
QRLabel156: TQRLabel;
QRSysData10: TQRSysData;
QRSysData11: TQRSysData;
QRLabel157: TQRLabel;
QRImage4: TQRImage;
QRLabel158: TQRLabel;
QRDBText27: TQRDBText;
QRBand22: TQRBand;
QRLabel159: TQRLabel;
Query4: TQuery;
QRLabel161: TQRLabel;
QRLabel162: TQRLabel;
QRShape18: TQRShape;
QRShape19: TQRShape;
QRGroup16: TQRGroup;
QRGroup17: TQRGroup;
QRGroup18: TQRGroup;
QRDBText28: TQRDBText;
QRBand23: TQRBand;
QRExpr23: TQRExpr;
QRExpr24: TQRExpr;
QRBand24: TQRBand;
QRBand25: TQRBand;
QRExpr25: TQRExpr;
QRExpr26: TQRExpr;
QRLabel163: TQRLabel;
QRExpr27: TQRExpr;
QRExpr28: TQRExpr;
QRShape20: TQRShape;
QRShape21: TQRShape;
QRLabel164: TQRLabel;
QRLabel165: TQRLabel;
QRBand26: TQRBand;
QRLabel166: TQRLabel;
QRSysData12: TQRSysData;
QRLabel167: TQRLabel;
QRLabel168: TQRLabel;
```

```
QRDBText32: TQRDBText;
QRGroup19: TQRGroup;
QRDBText34: TQRDBText;
QRDBText33: TQRDBText;
QRBand27: TQRBand;
QRDBText29: TQRDBText;
QRDBText30: TQRDBText;
QRDBText31: TQRDBText;
QRExpr29: TQRExpr;
QRExpr30: TQRExpr;
QRShape22: TQRShape;
QRLabel69: TQRLabel;
QRShape23: TQRShape;
QRLabel60: TQRLabel;
QRBand28: TQRBand;
QRShape24: TQRShape;
QRLabel70: TQRLabel;
QRExpr31: TQRExpr;
QRLabel39: TQRLabel;
QRSysData9: TQRSysData;
QRLabel52: TQRLabel;
QRLabel53: TQRLabel;
QRExpr32: TQRExpr;
QRExpr33: TQRExpr;
Query5: TQuery;
QuickRep5: TQuickRep;
QRBand29: TQRBand;
QRLabel71: TQRLabel;
QRLabel72: TQRLabel;
QRLabel73: TQRLabel;
QRSysData13: TQRSysData;
QRSysData14: TQRSysData;
QRLabel74: TQRLabel;
QRImage5: TQRImage;
QRLabel75: TQRLabel;
QRDBText35: TQRDBText;
QRBand30: TQRBand;
QRShape25: TQRShape;
QRShape26: TQRShape;
QRLabel76: TQRLabel;
QRLabel77: TQRLabel;
QRLabel78: TQRLabel;
QRLabel79: TQRLabel;
QRLabel80: TQRLabel;
QRGroup20: TQRGroup;
QRGroup21: TQRGroup;
QRDBText36: TQRDBText;
QRDBText37: TQRDBText;
QRGroup22: TQRGroup;
QRBand31: TQRBand;
QRDBText38: TQRDBText;
QRDBText39: TQRDBText;
QRExpr34: TQRExpr;
QRExpr35: TQRExpr;
QRBand32: TQRBand;
QRLabel81: TQRLabel;
QRExpr36: TQRExpr;
QRExpr37: TQRExpr;
QRBand33: TQRBand;
QRLabel82: TQRLabel;
QRExpr38: TQRExpr;
QRExpr39: TQRExpr;
```

```
QRShape27: TQRShape;
QRShape28: TQRShape;
QRShape29: TQRShape;
QRBand34: TQRBand;
QRLabel83: TQRLabel;
QRSysData15: TQRSysData;
QRLabel84: TQRLabel;
QRLabel85: TQRLabel;
Query6: TQuery;
QuickRep6: TQuickRep;
QRBand35: TQRBand;
QRLabel86: TQRLabel;
QRLabel87: TQRLabel;
QRLabel88: TQRLabel;
QRSysData16: TQRSysData;
QRSysData17: TQRSysData;
QRLabel89: TQRLabel;
QRImage6: TQRImage;
QRLabel90: TQRLabel;
QRDBText40: TQRDBText;
QRBand36: TQRBand;
QRShape30: TQRShape;
QRShape31: TQRShape;
QRLabel91: TQRLabel;
QRLabel92: TQRLabel;
QRLabel93: TQRLabel;
QRLabel94: TQRLabel;
QRLabel95: TQRLabel;
QRGroup23: TQRGroup;
QRGroup24: TQRGroup;
QRGroup25: TQRGroup;
QRGroup26: TQRGroup;
QRGroup27: TQRGroup;
QRBand37: TQRBand;
QRDBText45: TQRDBText;
QRDBText46: TQRDBText;
QRExpr40: TQRExpr;
QRExpr41: TQRExpr;
QRBand38: TQRBand;
QRLabel96: TQRLabel;
QRExpr42: TQRExpr;
QRExpr43: TQRExpr;
QRBand39: TQRBand;
QRGroup28: TQRGroup;
QRDBText41: TQRDBText;
QRDBText42: TQRDBText;
QRDBText43: TQRDBText;
QRDBText44: TQRDBText;
QRLabel97: TQRLabel;
QRExpr44: TQRExpr;
QRExpr45: TQRExpr;
QRBand40: TQRBand;
QRLabel98: TQRLabel;
QRExpr46: TQRExpr;
QRExpr47: TQRExpr;
QRBand42: TQRBand;
QRLabel99: TQRLabel;
QRSysData18: TQRSysData;
QRLabel100: TQRLabel;
QRLabel101: TQRLabel;
QRShape32: TQRShape;
QRShape33: TQRShape;
```

```

QRShape34: TQRShape;
QRShape35: TQRShape;
PageControl1: TPageControl;
TabSheet1: TTabSheet;
DetailBand1: TQRBand;
DetailBand2: TQRBand;
DetailBand3: TQRBand;
DetailBand4: TQRBand;
DetailBand5: TQRBand;
DetailBand6: TQRBand;
TitleBand1: TQRBand;
GroupBox1: TGroupBox;
RadioButton1: TRadioButton;
RadioButton2: TRadioButton;
RadioButton3: TRadioButton;
RadioButton4: TRadioButton;
RadioButton5: TRadioButton;
RadioButton6: TRadioButton;
RadioButton7: TRadioButton;
Button1: TButton;
Button2: TButton;
procedure Button2Click(Sender: TObject);
procedure Button1Click(Sender: TObject);
private
  ( Private declarations )
public
  Procedure Arma_Reporte(numero_reporte : integer);
  ( Public declarations )
end;

var
  F_pride: TF_pride;

implementation

uses Paipar;

{$R *.DFM}

Procedure TF_pride.Arma_Reporte(numero_reporte : integer);
Var Paginas : String ;
Begin
  If numero_reporte = 1 then
  Begin
    Query1.Open;
    QuickRep1.Prepare;
    Paginas := IntToStr(QuickRep1.QRPrinter.AvailablePages);
    QuickRep1.QRPrinter.Free;
    QuickRep1.QRPrinter := nil;
    Qrlabel48.Caption := Paginas;
    QuickRep1.Preview;
    Query1.close;
  End;
  If numero_reporte = 2 then
  Begin
    Query2.Open;
    QuickRep2.Prepare;
    Paginas := IntToStr(QuickRep2.QRPrinter.AvailablePages);
    QuickRep2.QRPrinter.Free;
    QuickRep2.QRPrinter := nil;
    Qrlabel51.Caption := Paginas;
    QuickRep2.Preview;
  End;
End;

```

```
    Query2.Close;
End;
If numero_reporte = 3 then
Begin
    Query3.Open;
    QuickRep3.Prepare;
    Paginas := IntToStr(QuickRep3.QRPrinter.AvailablePages);
    QuickRep3.QRPrinter.Free;
    QuickRep3.QRPrinter := nil;
    Qrlabel53.Caption := Paginas;
    QuickRep3.Preview;
    Query3.Close;
End;
If numero_reporte = 4 then
Begin
    Query4.Open;
    QuickRep4.Prepare;
    Paginas := IntToStr(QuickRep4.QRPrinter.AvailablePages);
    QuickRep4.QRPrinter.Free;
    QuickRep4.QRPrinter := nil;
    Qrlabel68.Caption := Paginas;
    QuickRep4.Preview;
    Query4.Close;
End;
If numero_reporte = 5 then
Begin
    Query5.Open;
    QuickRep5.Prepare;
    Paginas := IntToStr(QuickRep5.QRPrinter.AvailablePages);
    QuickRep5.QRPrinter.Free;
    QuickRep5.QRPrinter := nil;
    Qrlabel85.Caption := Paginas;
    QuickRep5.Preview;
    Query5.Close;
End;
If numero_reporte = 6 then
Begin
    Query6.Open;
    QuickRep6.Prepare;
    Paginas := IntToStr(QuickRep6.QRPrinter.AvailablePages);
    QuickRep6.QRPrinter.Free;
    QuickRep6.QRPrinter := nil;
    Qrlabel101.Caption := Paginas;
    QuickRep6.Preview;
    Query6.Close;
End;
End;

procedure TF_pride.Button2Click(Sender: TObject);
begin
    Close;
end;

procedure TF_pride.Button1Click(Sender: TObject);
begin
    If Radiobutton1.checked then
        Arma_Reporte(1);
    If Radiobutton2.checked then
        Arma_Reporte(2);
    If Radiobutton3.checked then
        Arma_Reporte(3);
    If Radiobutton4.checked then
```

```

    F_paipa.Arma_Reporte(4);
  If Radiobutton5.checked then
    Arma_Reporte(5);
  If Radiobutton6.checked then
    Arma_Reporte(6);
  {PR 01}
  If Radiobutton7.checked then
    Arma_Reporte(4);
end;

end.
```

Código fuente de la unidad Paipar, que contiene el control para los reportes del programa PAIPA

```

{*****}

Modulo principal de reportes PAIPA, Se realiza la
elección del reporte a realizar y posteriormente se arma
cada uno de ellos.

Ultima modificación : 28/Junio/98      Lester :)
*****}
Unit Paipar;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  QrCtrls, quickrpt, ExtCtrls, Db, DBTables, ComCtrls, StdCtrls;

type
  TF_paipa = class(TForm)
    Query1: TQuery;
    QuickRep1: TQuickRep;
    QRBand13: TQRBand;
    QRLabel130: TQRLabel;
    QRLabel131: TQRLabel;
    QRLabel132: TQRLabel;
    QRSysData7: TQRSysData;
    QRSysData8: TQRSysData;
    QRLabel133: TQRLabel;
    QRImage3: TQRImage;
    QRLabel134: TQRLabel;
    QRDBText15: TQRDBText;
    ColumnHeaderBand1: TQRBand;
    QRLabel1: TQRLabel;
    QRLabel2: TQRLabel;
    QRLabel3: TQRLabel;
    QRLabel4: TQRLabel;
    QRLabel5: TQRLabel;
    QRLabel6: TQRLabel;
    QRLabel7: TQRLabel;
    QRLabel18: TQRLabel;
    QRGroup4: TQRGroup;
    QRGroup5: TQRGroup;
    QRGroup6: TQRGroup;
    QRGroup7: TQRGroup;
    QRBand3: TQRBand;
    QRDBText3: TQRDBText;
```

```
QRExpr1: TQRExpr;
QRExpr6: TQRExpr;
QRDBText4: TQRDBText;
QRDBText1: TQRDBText;
QRBand2: TQRBand;
QRExpr3: TQRExpr;
QRExpr4: TQRExpr;
QRShape1: TQRShape;
QRShape2: TQRShape;
QRExpr7: TQRExpr;
QRExpr8: TQRExpr;
QRExpr9: TQRExpr;
QRBand1: TQRBand;
QRExpr10: TQRExpr;
QRExpr11: TQRExpr;
QRExpr12: TQRExpr;
QRExpr13: TQRExpr;
QRExpr14: TQRExpr;
QRLabel10: TQRLabel;
QRLabel11: TQRLabel;
QRShape4: TQRShape;
QRShape5: TQRShape;
QRBand4: TQRBand;
QRBand5: TQRBand;
QRSysData1: TQRSysData;
QRLabel12: TQRLabel;
QRExpr16: TQRExpr;
QRExpr17: TQRExpr;
QRExpr18: TQRExpr;
QRExpr19: TQRExpr;
QRExpr20: TQRExpr;
QRLabel13: TQRLabel;
QRDBText8: TQRDBText;
QRDBText7: TQRDBText;
QRDBText9: TQRDBText;
QRShape3: TQRShape;
QuickRep2: TQuickRep;
QRBand6: TQRBand;
QRLabel14: TQRLabel;
QRLabel15: TQRLabel;
QRLabel16: TQRLabel;
QRSysData2: TQRSysData;
QRSysData3: TQRSysData;
QRLabel17: TQRLabel;
QRImage1: TQRImage;
QRLabel18: TQRLabel;
QRDBText10: TQRDBText;
QRBand7: TQRBand;
QRLabel19: TQRLabel;
QRLabel20: TQRLabel;
QRLabel21: TQRLabel;
QRLabel22: TQRLabel;
QRLabel23: TQRLabel;
QRLabel24: TQRLabel;
QRLabel25: TQRLabel;
QRLabel26: TQRLabel;
QRShape6: TQRShape;
QRShape7: TQRShape;
QRGroup1: TQRGroup;
QRGroup2: TQRGroup;
QRDBText11: TQRDBText;
QRDBText12: TQRDBText;
```

```
QRDBText23: TQRDBText;
QRDBText24: TQRDBText;
QRShape13: TQRShape;
QRGroup11: TQRGroup;
QRGroup12: TQRGroup;
QRBand16: TQRBand;
QRDBText26: TQRDBText;
QRExpr44: TQRExpr;
QRExpr45: TQRExpr;
QRDBText29: TQRDBText;
QRDBText30: TQRDBText;
QRBand17: TQRBand;
QRExpr46: TQRExpr;
QRExpr47: TQRExpr;
QRExpr49: TQRExpr;
QRExpr50: TQRExpr;
QRExpr51: TQRExpr;
QRLabel151: TQRLabel;
QRShape14: TQRShape;
QRBand18: TQRBand;
QRExpr52: TQRExpr;
QRExpr53: TQRExpr;
QRExpr54: TQRExpr;
QRExpr55: TQRExpr;
QRExpr56: TQRExpr;
QRLabel152: TQRLabel;
QRShape15: TQRShape;
QRBand19: TQRBand;
QRExpr58: TQRExpr;
QRExpr59: TQRExpr;
QRExpr60: TQRExpr;
QRExpr61: TQRExpr;
QRExpr62: TQRExpr;
QRLabel153: TQRLabel;
QRBand20: TQRBand;
QRSysData9: TQRSysData;
QRLabel154: TQRLabel;
Query3: TQuery;
QuickRep4: TQuickRep;
QRBand21: TQRBand;
QRLabel155: TQRLabel;
QRLabel156: TQRLabel;
QRLabel157: TQRLabel;
QRSysData10: TQRSysData;
QRSysData11: TQRSysData;
QRLabel158: TQRLabel;
QRImage4: TQRImage;
QRLabel159: TQRLabel;
QRDBText31: TQRDBText;
QRBand22: TQRBand;
QRLabel160: TQRLabel;
QRLabel161: TQRLabel;
QRLabel162: TQRLabel;
QRLabel163: TQRLabel;
QRLabel164: TQRLabel;
QRLabel165: TQRLabel;
QRLabel166: TQRLabel;
QRLabel167: TQRLabel;
QRShape16: TQRShape;
QRShape17: TQRShape;
QRGroup13: TQRGroup;
QRGroup14: TQRGroup;
```

```
-----: TQRShape;
TabSheet1: TTabSheet;
QRExpr5: TQRExpr;
```

```

QRExpr15: TQRExpr;
DetailBand1: TQRBand;
DetailBand2: TQRBand;
DetailBand3: TQRBand;
DetailBand4: TQRBand;
GroupBox1: TGroupBox;
RadioButton1: TRadioButton;
RadioButton2: TRadioButton;
RadioButton3: TRadioButton;
Button1: TButton;
Button2: TButton;
procedure Button2Click(Sender: TObject);
procedure Button1Click(Sender: TObject);
private
  { Private declarations }
public
  Procedure Arma_Reporte(numero_reporte : integer);
  { Public declarations }
end;

var
  F_paipa: TF_paipa;

implementation

{$R *.DFM}

Procedure TF_paipa.Arma_Reporte(numero_reporte : integer);
Var Paginas : String ;
Begin
  If numero_reporte = 1 then
  Begin
    Query1.Open;
    QuickRep1.Prepare;
    Paginas := InttoStr(QuickRep1.QRPrinter.AvailablePages);
    QuickRep1.QRPrinter.Free;
    QuickRep1.QRPrinter := nil;
    Qrlabel74.Caption := Paginas;
    QuickRep1.Preview;
    Query1.Close;
  End;
  If numero_reporte = 2 then
  Begin
    Query2.Open;
    QuickRep2.Prepare;
    Paginas := InttoStr(QuickRep2.QRPrinter.AvailablePages);
    QuickRep2.QRPrinter.Free;
    QuickRep2.QRPrinter := nil;
    Qrlabel76.Caption := Paginas;
    QuickRep2.Preview;
    Query2.Close;
  End;
  If numero_reporte = 3 then
  Begin
    Query3.Open;
    QuickRep3.Prepare;
    Paginas := InttoStr(QuickRep3.QRPrinter.AvailablePages);
    QuickRep3.QRPrinter.Free;
    QuickRep3.QRPrinter := nil;
    Qrlabel78.Caption := Paginas;
    QuickRep3.Preview;
    Query3.Close;
  End;
End;

```

```

End;
If numero_reporte = 4 then
Begin
    Query4.Open;
    QuickRep4.Prepare;
    Paginas := IntToStr(QuickRep4.QRPrinter.AvailablePages);
    QuickRep4.QRPrinter.Free;
    QuickRep4.QRPrinter := nil;
    QrLabel80.Caption := Paginas;
    QuickRep4.Preview;
    Query4.Close;
End;

End;

procedure TF_paipa.Button2Click(Sender: TObject);
begin
    Close;
end;

procedure TF_paipa.Button1Click(Sender: TObject);
begin
    If Radiobutton1.checked then
        Arma_reporte(1);
    If Radiobutton2.checked then
        Arma_reporte(2);
    If Radiobutton3.checked then
        Arma_reporte(3);
end;
end.

```

V.2.3.- SIPES I - Partida 356 Reconocimiento catedrático UNAM.

V.2.3.1.- Partida 356 - CATEDRA.

V.2.3.1.1.- Formas y Unidades .

Módulo	Unidad	Forma	Descripción
Sipes I	Recatir	F_recati	En esta unidad se almacenan los reportes de Catedra

```

{*****}
Modulo principal de reportes CATEDRA, Se realiza la
elección del reporte a realizar y posteriormente se arma
cada uno de ellos.

Ultima modificación : 15/Agosto/2000

{*****}
unit RecatiR1;

interface

```

```

PC1: TPageControl;
TabSheet1: TTabSheet;
QuickRep3: TQuickRep;
QRBand13: TQRBand;
QRLabel30: TQRLabel;
QRLabel31: TQRLabel;
QRLabel32: TQRLabel;
QRSysData7: TQRSysData;
QRSysData8: TQRSysData;
QRLabel33: TQRLabel;
QRImage3: TQRImage;
QRLabel34: TQRLabel;
QRDBText15: TQRDBText;
QRGroup8: TQRGroup;

```

```

QRLabel19: TQRLabel;

```

```

QRBand24: TQRBand;
QRExpr67: TQRExpr;
QRExpr68: TQRExpr;
QRExpr72: TQRExpr;
QRLabel69: TQRLabel;

```

```
QRShape19: TQRShape;
QRBand26: TQRBand;
QRExpr79: TQRExpr;
QRExpr81: TQRExpr;
QRExpr83: TQRExpr;
QRLabel171: TQRLabel;
QRBand27: TQRBand;
QRSysData12: TQRSysData;
QRLabel172: TQRLabel;
QRLabel179: TQRLabel;
QRLabel180: TQRLabel;
Query5: TQuery;
QRLabel19: TQRLabel;
QRLabel112: TQRLabel;
QRDBText7: TQRDBText;
QRDBText8: TQRDBText;
QRBand29: TQRBand;
QRLabel23: TQRLabel;
QRExpr2: TQRExpr;
QRExpr4: TQRExpr;
QRShape4: TQRShape;
QRGroup19: TQRGroup;
QRGroup21: TQRGroup;
QRLabel24: TQRLabel;
QRLabel35: TQRLabel;
QRDBText13: TQRDBText;
QRDBText14: TQRDBText;
QRDBText25: TQRDBText;
QRLabel43: TQRLabel;
QRLabel44: TQRLabel;
QRLabel54: TQRLabel;
QRDBText27: TQRDBText;
QRLabel60: TQRLabel;
QRExpr15: TQRExpr;
QRExpr16: TQRExpr;
QRExpr23: TQRExpr;
QRDBText26: TQRDBText;
QRDBText28: TQRDBText;
QRLabel64: TQRLabel;
QRLabel65: TQRLabel;
QuickRep5: TQuickRep;
Query6: TQuery;
QRBand11: TQRBand;
QRLabel66: TQRLabel;
QRLabel68: TQRLabel;
QRLabel73: TQRLabel;
QRSysData13: TQRSysData;
QRSysData14: TQRSysData;
QRLabel74: TQRLabel;
QRImage5: TQRImage;
QRLabel75: TQRLabel;
QRDBText29: TQRDBText;
QRBand15: TQRBand;
QRLabel77: TQRLabel;
QRLabel78: TQRLabel;
QRLabel81: TQRLabel;
QRLabel82: TQRLabel;
QRLabel83: TQRLabel;
QRLabel84: TQRLabel;
QRLabel85: TQRLabel;
QRBand30: TQRBand;
QRDBText30: TQRDBText;
```

```

QRDBText35: TQRDBText;
QRDBText36: TQRDBText;
QRDBText38: TQRDBText;
QRBand31: TQRBand;
QRDBText41: TQRDBText;
QRExpr24: TQRExpr;
QRExpr25: TQRExpr;
QRExpr26: TQRExpr;
QRBand32: TQRBand;
QRLabel89: TQRLabel;
QRExpr27: TQRExpr;
QRExpr28: TQRExpr;
QRLabel90: TQRLabel;
QRExpr29: TQRExpr;
QRExpr30: TQRExpr;
QRLabel86: TQRLabel;
QRDBText37: TQRDBText;
QRLabel87: TQRLabel;
QRSysData15: TQRSysData;
QRLabel88: TQRLabel;
QRLabel91: TQRLabel;
QRLabel92: TQRLabel;
QRExpr34: TQRExpr;
QRShape10: TQRShape;
QRShape11: TQRShape;
QRShape21: TQRShape;
QRExpr35: TQRExpr;
QRExpr36: TQRExpr;
QRExpr37: TQRExpr;
QRExpr38: TQRExpr;
QRShape12: TQRShape;
QRShape13: TQRShape;
QRShape14: TQRShape;
QRShape20: TQRShape;
DetailBand1: TQRBand;
DetailBand2: TQRBand;
DetailBand3: TQRBand;
DetailBand4: TQRBand;
GroupBox1: TGroupBox;
RadioButton1: TRadioButton;
RadioButton2: TRadioButton;
RadioButton3: TRadioButton;
RadioButton4: TRadioButton;
RadioButton5: TRadioButton;
Button1: TButton;
Button2: TButton;
procedure Button2Click(Sender: TObject);
procedure Button1Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  F_recati: TF_recati;

implementation

{$R *.DFM}

```

- 149 -

```

Paginas := IntToStr(QuickRep5.QuickRep1.Printer.PrinterPages);
QuickRep5.QRPrinter.Free;
QuickRep5.QRPrinter := nil;
QRLabel91.Caption := Paginas;
QuickRep5.Preview;

```

- 150 -

- 151 -

```

end;

```

- 152 -

```

procedure TF_Edpab.TabSheet3Show(Sender: TObject);
begin
    Query1.Close;
end;

procedure TF_Edpab.Button2Click(Sender: TObject);
begin
    If radiobutton1.Checked then
        F_redpabcat.showmodal;
    If radiobutton2.Checked then
        F_redpabdep.showmodal;
    If radiobutton3.Checked = true then
        F_redpab.showmodal;
    If radiobutton4.Checked = true then
        F_edpabniv.showmodal;
    If radiobutton5.Checked = true then
        F_edpabsubra.showmodal;
    If radiobutton12.Checked = true then
        Begin;
            F_Edpacpre.Query1.Open;
            F_Edpacpre.QuickRepl.Preview;
            F_Edpacpre.Query1.Close;
        End;
end;
end;
end.

```

V.2.5.- SIPES II - Partida 344 Estimulos y gratificaciones complementarias.

V.2.5.1.- EDPAC.

V.2.5.1.1.- Formas y Unidades.

Módulo	Unidad	Forma	Descripción
Sipes II	EDPAC	F_Edpac	Unidad que contiene el control de los reportes del programa de estímulos al personal de confianza

{*****}

Módulo principal de reportes EDPAC, Se realiza la elección del reporte a realizar y posteriormente se arma cada uno de ellos.

Ultima modificación : // Lester :)

*****}

unit Edpac;

interface

uses

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
 Db, DBCtrls, Grids, DBGrids, DBTables, ComCtrls, ExtCtrls, StdCtrls,
 QrCtrls, quickrpt;

V.3.- Manual de Operación.

Una vez concluida la programación se da paso a la instalación y operación por parte de los usuarios a continuación se describe la forma en el que el programa opera y su apariencia después de la programación realizada.

V.3.1.- Ejecución SIPES I.

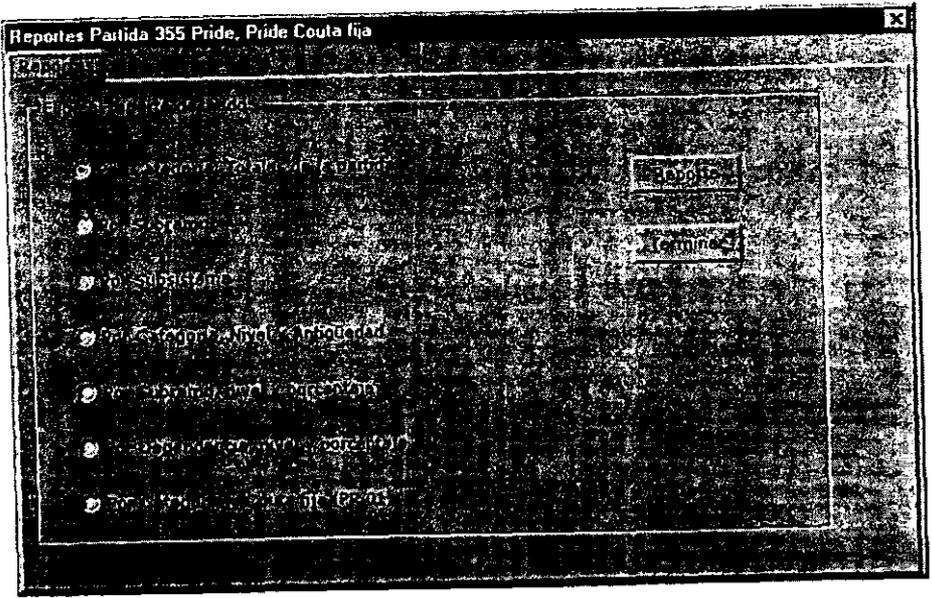
Para iniciar la ejecución del programa SIPES, debe seleccionar entre los iconos de Sipes 1 o Sipes 2, los cuales permiten el acceso a los programas y se encuentran sobre el escritorio de Windows. A continuación se mostrará la pantalla de presentación :



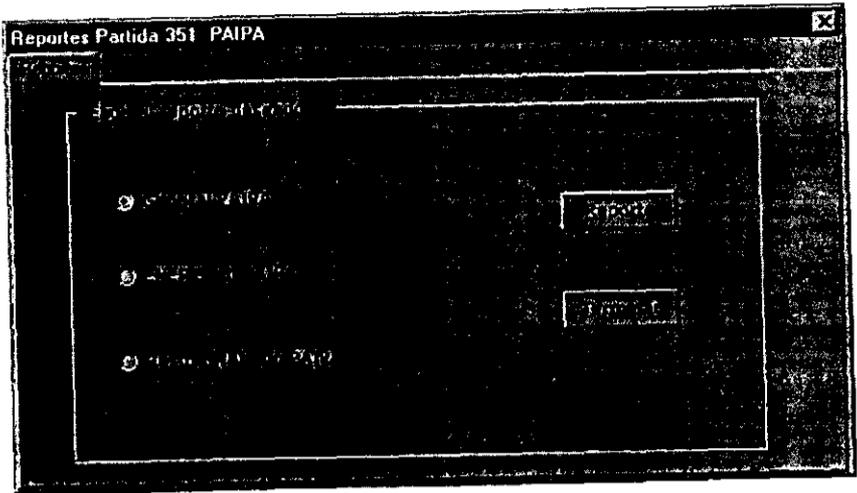
Capítulo V, Figura V.1 Ventana de presentación

Una vez que se oprimió el botón aceptar se da paso a la autenticación de seguridad del programa que permite la conexión con la base de datos a través de la siguiente pantalla :

parte de las decisiones de programación sin embargo para continuar con el formato que hemos seguido a lo largo del documento estos programas de la partida 355 se manejarán conjuntamente, las pantallas principales de PRIDE y PAIPA se muestran a continuación.



Capítulo V, Figura V.17 Ventana de selección de reportes PRIDE



Capítulo V, Figura V.18 Ventana de selección de reportes PAIPA

La cantidad de reportes varía considerablemente dado las necesidades de información de los usuarios recopiladas en las fases de levantamiento de la información y análisis. Sin embargo el tratamiento por parte de los operadores del sistema es mas simple que en las partidas anteriores, en este caso están a disposición 10 diferentes reportes que

51	2	112	1	0	158	1 CA	43
51	2	112	1	4	143	1 CF	81
51	2	112	1	4	143	1 CA	81
51	2	112	1	2	198	1 CA	81
51	2	112	1	0	345	1 XX	1
51	2	113	1	4	161	1 C	12
51	2	113	1	4	165	1 C	12
51	2	113	1	0	345	1 XX	1
51	2	113	1	0	345	1 XX	1
51	2	113	1	4	143	1 CF	81
51	2	113	1	4	143	1 CA	81
51	2	113	1	2	198	1 CA	81
51	2	113	1	0	345	1 XX	1
51	2	443	1	4	117	1 D	41
51	2	443	1	4	119	1 D	41

Capítulo V, Figura V.25 Ventana que contiene los datos de la nómina en curso.

Las últimas dos opciones son de carácter informativo se trata de la pantalla Nomina DGP y el catalogo de categorías de la UNAM.

Categoría	Descripción	Salario	Beneficio	Fecha
01	01/01/00
02	01/01/00
03	01/01/00
04	01/01/00
05	01/01/00
06	01/01/00
07	01/01/00
08	01/01/00
09	01/01/00
10	01/01/00
11	01/01/00
12	01/01/00
13	01/01/00
14	01/01/00
15	01/01/00
16	01/01/00
17	01/01/00
18	01/01/00
19	01/01/00
20	01/01/00

Capítulo V, Figura V.26 Datos del tabulador de categorías de la UNAM.

partidas en el futuro (a un año o más), esto cierra el ciclo presupuestal al cual el sistema provee de información constantemente.

El desarrollo y la implementación constituyen la etapa culminante de todo nuestro proceso de desarrollo, sin embargo, se ha visto ampliamente beneficiada con la adecuada aplicación de los conceptos de desarrollo de sistemas en los capítulos anteriores así que solo resta mencionar las bondades del metodo de desarrollo planteado al final del primer capítulo que nos ha permitido un crecimiento constante de la aplicación para la resolución de los problemas (nuevos) implícitos en el manejo de la información presupuestal de la Universidad Nacional Autónoma de México.

Anexo A

Glosario de términos

A

Asignación de recursos : Es el importe autorizado en programas, subprogramas, grupos y partidas, destinado a sufragar las erogaciones de la dependencia o subdependencia para el desarrollo de sus actividades.

C

Calidad y Eficiencia : Complemento al salario por calidad y eficiencia en el trabajo del Personal Administrativo de la Universidad Nacional Autónoma de México.

Catálogo presupuestal : Es el documento que proporciona un conjunto de elementos auxiliares, para la operación del Sistema de presupuesto.

CATEDRA : Reconocimiento catedrático UNAM, es el estímulo que se otorga a los profesores eméritos de la Institución.

Centralizadora : Es la dependencia que por sus funciones controla el ejercicio de una o varias partidas presupuestales, emitiendo un dictamen técnico sobre el gasto en que se incurra.

Código Programático : Es un conjunto de dígitos que, ordenados en forma sistemática, se constituye en el elemento central para procesar el cúmulo de información que demanda el Sistema de Presupuesto por Programas.

Ejemplo : PR SP DEP SD PAR DV
 22 01 211 02 255 07

De izquierda a derecha, los dígitos representan:

PR	La función y el número del programa.
SP	El subprograma.
DEP	La dependencia.
SD	La subdependencia.
PAR	La partida de gasto.
DV	De estos dos campos, el primero identifica a la dependencia centralizadora del gasto y el segundo constituye el dígito verificador que permite comprobar la correcta integración del código programático.

D

Delphi : Lenguaje de programación RAD (Rapid Aided Design) para windows desarrollado por Borland - Inprise, basado en Pascal y C++.

DIPASI : Distinción a los Profesores de Asignatura, programa de estímulos para los académicos de la Universidad Nacional Autónoma de México.

Disponible : Es el saldo o remanente de los recursos por ejercer en un código programático.

Dependencia : Cada una de las unidades derivadas de la estructura organizacional de la UNAM. La cual lleva a cabo una serie de actividades para el logro de metas y objetivos acordes a una función de la institución.

DGAPA : Dirección General de Asuntos del Personal Académico.

DGP : Dirección General de Personal.

DGPP : Dirección General de Programación Presupuestal. Dependencia universitaria encargada del control del presupuesto por programas de la UNAM.

E

EDPAC : Estimulos y gratificaciones complementarias para los trabajadores de confianza de la Universidad Nacional Autónoma de México.

Estructura programática Universitaria : La estructura programática es la parte central de la técnica de presupuesto por programas de la Universidad; en ella se conjugan sistemáticamente los programas y subprogramas con los recursos encaminados al cumplimiento de las funciones de docencia, investigación, extensión universitaria y gestión institucional.

Esta estructura es el marco conceptual en el cual las dependencias presentan los objetivos y metas que pretenden lograr, así como los recursos humanos, materiales y financieros que se requieran para su cumplimiento.

F

FOMDOC : Fomento a la docencia, es un programa de estímulos para los profesores de carrera de la Universidad Nacional Autónoma de México.

G

Grupo de gasto : Es el elemento presupuestario que agrupa los bienes y servicios que la Institución adquiere conforme a la naturaleza intrínseca de los mismos.

GUI : (Graphic User Interface) La Interfaz Grafica de Usuario es una serie de clases fundamentales para el desarrollo de programas en entornos gráficos tales como Windows o Gnome.

O

OOP : (Oriented Object Programming) Programación orientada a objetos, es un paradigma de programación que involucra principios de herencia, encapsulación y polimorfismo para el desarrollo de sistemas computacionales.

P

Partida : Es el elemento presupuestario en que se dividen los subgrupos de gasto y que clasifican las erogaciones de acuerdo con el objeto específico del gasto.

PEPASIG : Programa de estímulos a la productividad y rendimiento del personal académico de asignatura.

Presupuesto operativo : Es el documento que contiene a detalle, los niveles de programación y los recursos asignados a cada dependencia, y de la Institución para un año de calendario.

PRIDE : Prima de desempeño al personal académico de tiempo completo de la Universidad Nacional Autónoma de México.

S

SISAP (Sistema para la Administración Presupuestal) : Es el nombre de la base de datos y un conjunto de programas que constituyen el banco de información presupuestal de la DGPP, esta constituido por los diferentes sistemas de administración presupuestal y catálogos de información sobre el presupuesto universitario.

SQL : (Structured Query Language) Lenguaje estructurado de peticiones, es un lenguaje para acceso y creación de objetos en bases de datos relacionales tales como Sybase u Oracle.

Subgrupo de Gasto : Es el elemento presupuestario en que se dividen los grupos de gasto y clasifican las erogaciones afines, de acuerdo con el objeto de gasto.

PEPASIG : Programa de estímulos a la productividad y rendimiento del personal académico de asignatura.

Presupuesto operativo : Es el documento que contiene a detalle, los niveles de programación y los recursos asignados a cada dependencia, y de la Institución para un año de calendario.

PRIDE : Prima de desempeño al personal académico de tiempo completo de la Universidad Nacional Autónoma de México.

S

SISAP (Sistema para la Administración Presupuestal) : Es el nombre de la base de datos y un conjunto de programas que constituyen el banco de información presupuestal de la DGPP, esta constituido por los diferentes sistemas de administración presupuestal y catálogos de información sobre el presupuesto universitario.

SQL : (Structured Query Language) Lenguaje estructurado de peticiones, es un lenguaje para acceso y creación de objetos en bases de datos relacionales tales como Sybase u Oracle.

Subgrupo de Gasto : Es el elemento presupuestario en que se dividen los grupos de gasto y clasifican las erogaciones afines, de acuerdo con el objeto de gasto.

Disponible : Es el saldo o remanente de los recursos por ejercer en un código programático.

Dependencia : Cada una de las unidades derivadas de la estructura organizacional de la UNAM. La cual lleva a cabo una serie de actividades para el logro de metas y objetivos acordes a una función de la institución.

DGAPA : Dirección General de Asuntos del Personal Académico.

DGP : Dirección General de Personal.

DGPP : Dirección General de Programación Presupuestal. Dependencia universitaria encargada del control del presupuesto por programas de la UNAM.

E

EDPAC : Estimulos y gratificaciones complementarias para los trabajadores de confianza de la Universidad Nacional Autónoma de México.

Estructura programática Universitaria : La estructura programática es la parte central de la técnica de presupuesto por programas de la Universidad; en ella se conjugan sistemáticamente los programas y subprogramas con los recursos encaminados al cumplimiento de las funciones de docencia, investigación, extensión universitaria y gestión institucional.

Esta estructura es el marco conceptual en el cual las dependencias presentan los objetivos y metas que pretenden lograr, así como los recursos humanos, materiales y financieros que se requieran para su cumplimiento.

F

FOMDOC : Fomento a la docencia, es un programa de estímulos para los profesores de carrera de la Universidad Nacional Autónoma de México.

G

Grupo de gasto : Es el elemento presupuestario que agrupa los bienes y servicios que la Institución adquiere conforme a la naturaleza intrínseca de los mismos.

GUI : (Graphic User Interface) La Interfaz Grafica de Usuario es una serie de clases fundamentales para el desarrollo de programas en entornos gráficos tales como Windows o Gnome.

O

OOP : (Oriented Object Programming) Programación orientada a objetos, es un paradigma de programación que involucra principios de herencia, encapsulación y polimorfismo para el desarrollo de sistemas computacionales.

P

Partida : Es el elemento presupuestario en que se dividen los subgrupos de gasto y que clasifican las erogaciones de acuerdo con el objeto específico del gasto.