



3
29
A. J. V.
2000

**UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO**

**FACULTAD DE ESTUDIOS SUPERIORES
CUAUTITLAN**

**DESARROLLO DE UN SISTEMA EN ARQUITECTURA
CLIENTE/SERVIDOR PARA EL SEGUIMIENTO DEL
EJERCICIO PRESUPUESTAL DE LA GERENCIA DE
INSPECCION Y MANTENIMIENTO, PEMEX-PEP.**

T E S I S

QUE PARA OBTENER EL TITULO DE:

LICENCIADO EN INFORMATICA

P R E S E N T A :

ADELA ADRIANA ESCOBEDO GARCIA

ASESOR: ING. FELIPE BELTRAN TREJO.

CUAUTITLAN IZCALLI, EDO. DE MEXICO.

1998.

**TESIS CON
FALLA DE ORIGEN**

265440



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.



UNIVERSIDAD NACIONAL
AVENIDA DE
MEXICO

FACULTAD DE ESTUDIOS SUPERIORES CUAUTITLAN
UNIDAD DE LA ADMINISTRACION ESCOLAR
DEPARTAMENTO DE EXAMENES PROFESIONALES

ASUNTO: VOTOS APROBATORIOS

U. N. A. M.
FACULTAD DE ESTUDIOS
SUPERIORES - CUAUTITLAN



Departamento de

DR. JUAN ANTONIO MONTARAZ CRESPO
DIRECTOR DE LA F.E.S.-CUAUTITLAN
P R E S E N T E .

AT'N: Ing. Jaime de Anda Montañez
Jefe del Departamento de Exámenes
Profesionales de la F.E.S.-C

Con base en el art. 28 del Reglamento General de Exámenes, nos permitimos comunicar a usted que revisamos la TESIS:

" Desarrollo de un Sistema en Arquitectura Cliente/Servidor para el seguimiento del Ejercicio Presupuestal de la Gerencia de Inspección y Mantenimiento PEMEX - PEP "

que presenta la pasante: Adela Adriana Escobedo García
con número de cuenta: 9057104 - 4 para obtener el TITULO de:
Licenciada en Informática

Considerando que dicho tesis reúne los requisitos necesarios para ser discutida en el EXAMEN PROFESIONAL correspondiente, otorgamos nuestro VOTO APROBATORIO.

A T E N T A M E N T E.
"POR MI RAZA HABLARA EL ESPIRITU"

Cuautitlán Izcalli, Edo. de México, a 18 de Marzo de 1998.

PRESIDENTE	Ing. Felipe Beltrán Trejo	
VOCAL	Lic. Epifanio Pineda Celis	
SECRETARIO	Ing. Sergio P. Acosta Torres	
PRIMER SUPLENTE	L.A. Araceli Nivón Zaghi	
SEGUNDO SUPLENTE	Ing. Marco A. Hernández Rodríguez	

*A*gradezco a todos aquellos
Que voluntaria e involuntariamente
Han sido mis maestros
Tanto en lo académico como en la vida.

Y en especial dedico mi tesis
A la memoria de mi Padre
Leonardo Escobedo Martínez
Que siempre esta con nosotros
“Gracias por tus enseñanzas y
gran amor hacia tu familia”

AGRADECIMIENTOS

Con gran cariño y admiración agradezco a mi Madre, ejemplo de tenacidad y lucha por lograr lo que queremos, a ella le dedico mi trabajo y esfuerzo en cualquier meta que me proponga alcanzar. También doy gracias a mis hermanos Omar y Leonardo por su gran apoyo.

Agradezco a mis amigos y amigas de los cuales me siento afortunada por tener su amistad y apoyo incondicional y siempre brindarme su experiencia y saber.

A mi asesor por su paciencia y dedicación, igualmente a mis profesores que invirtieron su conocimiento en mí, a ellos mi más sincero agradecimiento.

A mi universidad FES-Cuautitlán que me proporcionó las herramientas y el conocimiento necesario para ser una profesionista.

A todos ellos muchas gracias.

Adela Adriana Escobedo García.

CONTENIDO

INTRODUCCION

OBJETIVOS DE LA TESIS

CAPITULO I

ARQUITECTURA CLIENTE/SERVIDOR

INFRAESTRUCTURA DE COMPUTO DE SISTEMAS EN ARQUITECTURA CLIENTE/SERVIDOR

ASPECTOS GENERALES DE REDES

1.1. CONCEPTOS GENERALES DE REDES	1	
1.1.1. <u>DEFINICIÓN</u>	2	
1.1.2. <u>OBJETIVOS</u>	2	
1.1.3. <u>UTILIDAD</u>	2	
1.1.4. <u>VENTAJAS</u>	3	
1.1.5. <u>CLASIFICACIÓN</u>	3	
1.2. REDES DE AREA LOCAL	5	
1.2.1. <u>DEFINICIÓN</u>	5	
1.2.2. <u>CARACTERÍSTICAS BÁSICAS</u>	6	
1.2.3. <u>VENTAJAS</u>	6	
1.2.4. <u>COMPONENTES</u>	6	
1.2.5. <u>TOPOLOGÍAS</u>	8	
1.2.6. <u>PROTOCOLOS</u>	11	
1.2.7. <u>MEDIOS DE TRANSMISIÓN</u>	15	
1.2.8. <u>CABLEADO ESTRUCTURADO</u>	19	
1.2.9. <u>INTERCONECTIVIDAD</u>	20	
1.2.10. <u>TIPOS DE REDES LOCALES</u>	22	
1.2.11. <u>SISTEMAS OPERATIVOS DEL SERVIDOR</u>	27	
1.3. REDES DE AREA AMPLIA	29	
1.3.1. <u>REDES X.25</u>	29	
1.3.2. <u>FRAME RELAY</u>	30	
1.3.3. <u>ATM</u>	30	
1.4. EJEMPLO INFRAESTRUCTURA DE RED DE COMPUTO	31	

ASPECTOS GENERALES DE BASE DE DATOS

1.5. BASES DE DATOS	32	
1.5.1. ANTECEDENTES	32	
1.5.2. DEFINICIÓN	33	
1.5.3. CARACTERÍSTICAS	33	
1.5.4. VENTAJAS	34	
1.6. SISTEMAS MANEJADORES DE BASE DE DATOS (DBMS)	34	
1.6.1. DEFINICIÓN	34	
1.6.2. OBJETIVOS	35	
1.6.3. MODELO DE BASE DE DATOS	35	
1.6.4. LENGUAJES	39	
1.7. BASE DE DATOS DISTRIBUIDAS	40	
1.7.1. EVOLUCIÓN	40	
1.7.2. DEFINICIÓN	40	
1.7.3. CARACTERÍSTICAS	41	
1.7.4. PROCESAMIENTO COOPERATIVO	41	
1.7.5. REPLICACIÓN	41	
1.8. SISTEMA MANEJADOR DE BASE DE DATOS RELACIONALES (RDBMS)	42	
1.8.1. ORIGEN	42	
1.8.2. SQL	42	
1.8.3. CARACTERÍSTICAS	43	
1.8.4. ORACLE	44	
1.9. SISTEMA MANEJADOR DE BASE DE DATOS POR OBJETOS (ODBMS)	44	
1.9.1. DEFINICIÓN	44	
1.9.2. CARACTERÍSTICAS	45	

ASPECTOS DE LA ARQUITECTURA CLIENTE/SERVIDOR

1.10. ARQUITECTURA CLIENTE/SERVIDOR	45	
1.10.1. ANTECEDENTES	45	
1.10.2. MOVIMIENTO A CLIENTE/SERVIDOR	47	
1.10.3. DEFINICIÓN	49	
1.10.4. PROCESO COOPERATIVO	50	
1.11. COMPONENTES CLIENTE/SERVIDOR	50	
1.11.1. EL LADO CLIENTE	51	
1.11.2. EL LADO SERVIDOR	51	
1.11.3. LA RED	52	
1.11.4. MIDDLEWARE	52	

1.12. ARQUITECTURA	53	
1.12.1. <u>ARQUITECTURA TÉCNICA</u>	53
1.12.2. <u>ARQUITECTURA DE APLICACIÓN</u>	53
1.13. OPTAR POR CLIENTE/SERVIDOR	54	
1.13.1. <u>BENEFICIOS</u>	54
1.13.2. <u>MITOS</u>	56
1.13.3. <u>ADOPTAR CLIENTE/SERVIDOR</u>	57
1.13.4. <u>HERRAMIENTAS DE DESARROLLO</u>	58
1.14. ASPECTOS TECNICOS PARA INICIAR UN SISTEMA C/S	59	
1.15. ARQUITECTURA C/S SEGUNDA GENERACION	61	
1.15.1. <u>CARACTERÍSTICAS</u>	61
1.16. EJEMPLO: INFRAESTRUCTURA DE LA ARQUITECTURA C/S	62	

CAPITULO II

METODOLOGIA PARA EL DESARROLLO DE SISTEMAS CLIENTE / SERVIDOR
--

2.1. INTRODUCCION	64	
2.2. ANALISIS Y DISEÑO DE SISTEMAS	64	
2.3. PAPEL DEL LIDER DE PROYECTO	65	
2.3.1. <u>CUALIDADES DEL ANALISTA DE SISTEMAS</u>	65
2.3.2. <u>CUALIDADES DEL LÍDER DE PROYECTO DE SISTEMAS</u>	66
2.4. USUARIOS	67	
2.5. CONCEPTOS DE SISTEMAS ORGANIZACIONALES	68	
2.5.1. <u>LA NUEVA CONCEPTUALIZACIÓN DE LA EMPRESA</u>	68
2.5.2. <u>SISTEMAS DE INFORMACIÓN</u>	69
2.5.3. <u>CARACTERÍSTICAS IMPORTANTES DE LOS SISTEMAS</u>	69
2.5.4. <u>OBJETIVOS DE LOS SISTEMAS DE INFORMACIÓN</u>	70
2.5.5. <u>PAPEL DEL LOS SISTEMAS DE INFORMACIÓN</u>	70
2.5.6. <u>TIPOS DE SISTEMAS DE INFORMACIÓN</u>	71

2.6. FUNDAMENTOS DE LOS PROYECTOS	72
2.6.1. <u>CAUSAS PARA PROPONER PROYECTOS</u>	72
2.6.2. <u>FUENTES DE SOLICITUDES DE PROYECTO</u>	73
2.6.3. <u>CONTRATACIÓN DE SERVICIOS INFORMÁTICOS EXTERNOS</u>	73
2.6.4. <u>MANEJO DEL PROCESO DE SELECCIÓN Y REVISIÓN DE PROYECTOS</u>	75
2.6.5. <u>ADMINISTRACIÓN DEL PROCESO DE DESARROLLO DE SISTEMAS DE INFORMACIÓN</u>	75
2.7. CICLO DE VIDA ESTRUCTURADO PARA EL DESARROLLO DE SISTEMAS CLIENTE/SERVIDOR	76
2.7.1. <u>ESBOZO DE UNA METODOLOGÍA PARA LA REINGENIERIA DE SISTEMAS CLIENTE/SERVIDOR</u>	76
 <i>ESTRUCTURA DE LA METODOLOGIA DEL SISTEMA "SISEFIN"</i>	
2.8. ANTECEDENTES DEL SISTEMA SISEFIN	77
2.9. PLANEACION DEL SISTEMA	79
2.10. OBJETIVOS GENERALES	80
2.11. DETERMINAR SI EXISTEN PROYECTOS SIMILARES	80
2.12. DESIGNACION DE RESPONSABLES	80
2.13. INTEGRACION DEL EQUIPO DE TRABAJO	81
2.14. REINGENIERIA: ESTUDIO DE LA VIEJA APLICACIÓN	83
2.14.1. <u>DEFINICIÓN</u>	83
2.14.2. <u>BENEFICIOS</u>	84
2.14.3. <u>METODOLOGÍA</u>	84
2.15. EVALUACION DE LA SOLICITUD DEL PROYECTO	87
2.15.1. <u>INVESTIGACIÓN O ANÁLISIS PRELIMINAR</u>	87
2.15.2. <u>PRUEBA DE FACTIBILIDAD DEL PROYECTO</u>	88
2.15.3. <u>REINGENIERÍA: ANÁLISIS CON RESPECTO A LA ANTERIOR APLICACIÓN</u>	90
2.15.4. <u>EVALUACIÓN DE LA FACTIBILIDAD</u>	91
2.16. PROPUESTA TECNICO – ECONOMICA	79
2.16.1. <u>PRESENTACIÓN DE LA PROPUESTA</u>	92
2.16.2. <u>MANEJO DE PROYECTOS NO FACTIBLES</u>	96

CAPITULO III

ANÁLISIS DE SISTEMAS CLIENTE / SERVIDOR

3.1. ANÁLISIS ESTRUCTURADO	97	
3.1.1. <u>DEFINICIÓN</u>		97
3.1.2. <u>COMPONENTES</u>		98
3.1.3. <u>ANÁLISIS DE LA SITUACIÓN ACTUAL</u>		98
3.1.4. <u>ANÁLISIS DE LOS REQUERIMIENTOS</u>		100
3.2. PANORAMA DE LA REINGENIERIA DE PROCESOS	102	
3.2.1. <u>DEFINICIÓN</u>		103
3.2.2. <u>PROCESOS DE NEGOCIOS</u>		104
3.2.3. <u>CULTURA ORGANIZACIONAL</u>		105
3.2.4. <u>TECNOLOGÍA DE INFORMACIÓN</u>		106
3.2.5. <u>HERRAMIENTAS PARA DOCUMENTAR PROCESOS Y DECISIONES</u>		107
3.2.6. <u>CONCLUSIONES</u>		108
3.3. ANÁLISIS DE FLUJO DE DATOS	110	
3.3.1. <u>DEFINICIÓN</u>		110
3.3.2. <u>ESTRATEGIA DE FLUJO DE DATOS</u>		111
3.3.3. <u>DIAGRAMA DE FLUJO DE DATOS</u>		111
3.3.4. <u>DICCIONARIO DE DATOS</u>		112
3.3.5. <u>ESTRUCTURA DE DATOS (MODELO LÓGICO)</u>		114
3.3.6. <u>DIAGRAMA DE FLUJO ESTRUCTURADO O CARTAS ESTRUCTURADAS</u>		114
3.4. HERRAMIENTAS ASISTIDAS POR COMPUTADORA	115	
3.4.1. <u>DEFINICIÓN</u>		115
3.4.2. <u>VENTAJAS</u>		116
3.4.3. <u>FUTUROS DESARROLLOS CON CASE</u>		117
3.5. ANÁLISIS DE LA INFRAESTRUCTURA C/S	117	
3.5.1. <u>EVALUACIÓN DE LA INFRAESTRUCTURA ACTUAL</u>		117
3.5.2. <u>NECESIDADES DE INFRAESTRUCTURA</u>		118
3.5.3. <u>ESTABLECIMIENTO DE LA ARQUITECTURA C/S</u>		118
3.6. EJEMPLO: ANÁLISIS DEL SISTEMA SISEFIN	119	

CAPITULO IV

DISEÑO DE SISTEMAS CLIENTE/SERVIDOR

4.1. DISEÑO PRELIMINAR	133
4.1.1. <u>DEFINICIÓN</u>	133
<i>DISEÑO DE LA ARQUITECTURA CLIENTE/SERVIDOR</i>	
4.1.2. <u>APLICACIÓN DEL USO DE INFORMACIÓN</u>	134
4.1.3. <u>REDES PARA EL CLIENTE/SERVIDOR</u>	134
4.1.4. <u>HARDWARE PARA EL CLIENTE/SERVIDOR</u>	135
4.1.5. <u>SOFTWARE PARA EL CLIENTE/SERVIDOR</u>	136
4.1.6. <u>UBICACIÓN DEL EQUIPO Y ESPECIFICACIONES ELÉCTRICAS</u>	137
4.1.7. <u>COSTOS</u>	137
4.1.8. <u>INTEGRACIÓN DE PLANES DE TRABAJO</u>	138
<i>DISEÑO DE SISTEMAS CLIENTE/SERVIDOR</i>	
4.1.9. <u>CRITERIOS DE DISEÑO</u>	138
4.1.10. <u>OBJETIVOS DE DISEÑO</u>	140
4.1.11. <u>CARACTERÍSTICAS QUE SE DEBEN DISEÑAR</u>	142
4.1.12. <u>REGLAS DE NEGOCIO</u>	143
4.1.13. <u>PARTICIPACIÓN DE LOS USUARIOS</u>	143
4.2. DISEÑO DE SALIDAS DEL SISTEMA	144
4.2.1. <u>IDENTIFICACIÓN DE LAS NECESIDADES DE SALIDA DEL SISTEMA</u>	144
4.2.2. <u>PRESENTACIÓN DE LA INFORMACIÓN</u>	145
4.2.3. <u>DISEÑO DE SALIDA IMPRESA</u>	145
4.2.4. <u>DISEÑO DE SALIDA EN PANTALLA</u>	145
4.3. DISEÑO DE ENTRADAS Y CONTROLES DEL SISTEMA	146
4.3.1. <u>DISEÑO DE ENTRADAS</u>	146
4.3.2. <u>OBJETIVOS DEL DISEÑO DE ENTRADAS</u>	147
4.3.3. <u>DISEÑO DE CONTROLES</u>	147
4.4. DISEÑO DE DIALOGO	149
4.4.1. <u>DEFINICIÓN</u>	149
4.4.2. <u>RECEPCIÓN DE MENSAJES</u>	150
4.4.3. <u>DECISIONES EN EL DISEÑO DE DIALOGO</u>	150
4.5. DISEÑO DE PROCEDIMIENTOS	151
4.5.1. <u>DEFINICIÓN</u>	151
4.5.2. <u>TIPOS DE PROCEDIMIENTOS</u>	151

4.6. DISEÑO DE ESPECIFICACIONES PARA PROGRAMAS	151
4.6.1. <u>DEFINICIÓN</u>	151
4.7. DISEÑO DE BASE DE DATOS	151
4.7.1. <u>DEFINICIÓN</u>	151
4.7.2. <u>MODELO LÓGICO DE DATOS</u>	152
4.7.3. <u>DISEÑO FÍSICO DE LA BASE DE DATOS</u>	156
4.8. PROTOTIPOS	163
4.8.1. <u>DEFINICIÓN</u>	163
4.8.2. <u>USOS</u>	163
4.8.3. <u>RAZONES</u>	164
4.8.4. <u>CARACTERÍSTICAS</u>	164
4.8.5. <u>ETAPAS DEL MÉTODO DE PROTOTIPOS</u>	165
4.8.6. <u>PROPÓSITO</u>	166
4.9. DISEÑO FINAL	167
4.9.1. <u>DEFINICIÓN</u>	167
4.9.2. <u>CARPETA DE DESCRIPCIÓN DEL DISEÑO DEL SISTEMA</u>	167
4.10. EJEMPLO: DISEÑO DEL SISTEMA SISEFIN	167

CAPITULO V

CONSTRUCCION E IMPLANTACION DE SISTEMAS CLIENTE / SERVIDOR

5.1. CONSTRUCCION DE LA INFRAESTRUCTURA	184
5.1.1. <u>INSTALACIÓN DE LA RED</u>	184
5.1.2. <u>CONFIGURACIÓN DEL SERVIDOR</u>	184
5.1.3. <u>INSTALACIÓN Y CONFIGURACIÓN DEL SISTEMA OPERATIVO DE RED</u>	185
5.1.4. <u>INSTALACIÓN DEL SOFTWARE DE APLICACIÓN</u>	185
5.2. PANORAMA DE LA INGENIERIA DE SOFTWARE	185
5.2.1. <u>HISTORIA</u>	185
5.2.2. <u>DEFINICIÓN</u>	186
5.2.3. <u>METAS</u>	186
5.2.4. <u>PRINCIPIOS</u>	187
5.2.5. <u>ASEGURAMIENTO DE LA CALIDAD DEL SOFTWARE</u>	188
5.2.6. <u>MÉTRICAS DE SOFTWARE</u>	189

5.2.7. <u>AUDITORIA</u>	189
5.3. CONSTRUCCION DEL SOFTWARE 190	
5.3.1. <u>PRINCIPIOS</u>	190
5.3.2. <u>CONFIABILIDAD</u>	191
5.3.3. <u>CONSTRUCCIÓN DE LA BASE DE DATOS</u>	192
5.3.4. <u>CONSTRUCCIÓN DE LA APLICACIÓN</u>	192
5.3.5. <u>ELABORACIÓN DE LOS DISCOS DE INSTALACIÓN</u>	192
5.4. PRUEBA DEL SOFTWARE 193	
5.4.1. <u>DEFINICIÓN</u>	193
5.4.2. <u>OBJETIVOS</u>	193
5.4.3. <u>VERIFICACIÓN, VALIDACIÓN Y CERTIFICACIÓN</u>	194
5.4.4. <u>ESTRATEGIAS DE PRUEBA</u>	194
5.4.5. <u>MANEJO DE LAS PRACTICAS DE PRUEBA</u>	195
5.4.6. <u>DISEÑO DE DATOS DE PRUEBA</u>	197
5.4.7. <u>REVISIONES POR PARTE DEL USUARIO</u>	197
5.4.8. <u>CORRECCIONES DE LA APLICACIÓN</u>	198
5.5. DOCUMENTACION 198	
5.5.1. <u>TÉCNICAS DE DOCUMENTACIÓN</u>	198
5.5.2. <u>ELABORACIÓN DE MANUALES</u>	198
5.5.3. <u>EVALUACIÓN DE LA DOCUMENTACIÓN</u>	199
5.6. IMPLANTACION 199	
5.6.1. <u>DEFINICIÓN</u>	199
5.7. INSTALACION 200	
5.7.1. <u>DEFINICIÓN</u>	200
5.7.2. <u>MÉTODOS DE CONVERSIÓN</u>	200
5.7.3. <u>ELABORACIÓN DE UN PROGRAMA DE INSTALACIÓN</u>	202
5.8. CAPACITACION 203	
5.8.1. <u>DEFINICIÓN</u>	203
5.8.2. <u>CAPACITACIÓN DE OPERADORES DE SISTEMAS</u>	203
5.8.3. <u>CAPACITACIÓN DE USUARIOS</u>	204
5.8.4. <u>MÉTODOS DE CAPACITACIÓN</u>	204
5.8.5. <u>MANUALES</u>	205
5.8.6. <u>PLAN DE CAPACITACIÓN</u>	205
5.9. RESPALDO Y RECUPERACION DE DATOS 205	
5.9.1. <u>CAUSAS POTENCIALES DE LA PERDIDA DE DATOS</u>	205
5.9.2. <u>MÉTODOS DE RESPALDO</u>	206
5.9.3. <u>CONSIDERACIONES DE RESPALDO</u>	207

5.10. MONITOREO DEL SISTEMA	207
5.10.1. <u>DEFINICIÓN</u>	207
5.10.1. <u>PROPÓSITOS DEL MONITOREO</u>	207
5.11. LIBERACION	208
5.11.1. <u>ACTA DE ACEPTACIÓN</u>	208
5.12. MANTENIMIENTO	208
5.12.1. <u>DEFINICIÓN</u>	208
5.12.2. <u>PUNTOS DE MANTENIMIENTO</u>	209
5.12.3. <u>TIPOS DE MANTENIMIENTO</u>	209
5.12.4. <u>REUTILIZACIÓN DE SOFTWARE</u>	209
5.13. EJEMPLO: CONSTRUCCION E IMPLANTACION DEL SISTEMA SISEFIN	210
CONCLUSIONES	222
BIBLIOGRAFIA	229
ANEXO A	A1

INTRODUCCION

Actualmente la información tiene una connotación de recurso primordial dentro de las organizaciones. Los responsables de la toma de decisiones consideran que la información es un elemento decisivo, que en un momento dado, determina el éxito o fracaso de un negocio, por lo que la información debe ser oportuna, confiable y válida.

La capacidad de la dirección para el buen manejo de las actividades de la empresa, depende del análisis de los diferentes tipos de información, principalmente la información financiera que determina cuales son sus ingresos y egresos generados en cierto periodo, planeando y estimando mejor las utilidades y atacando o detectando pérdidas o costos excesivos.

Para una empresa tan grande como Petróleos Mexicanos (PEMEX), la información presupuestal es muy amplia y de vital importancia en su productividad ya que influye en la economía nacional. PEMEX ha recurrido a sistemas computarizados para administrar la información y así lograr beneficios óptimos conociendo a tiempo la solución y tomando la mejor decisión a través de un amplio panorama sobre: ! a dónde se va el presupuesto, cómo se maneja, cada cuándo se presenta reportes presupuestales! etc.

Los sistemas computarizados aplicados a PEMEX es un campo de gran dinamismo. Los sistemas de información se desarrollan con diferentes propósitos los cuales dependen de las necesidades de la empresa. En este caso PEMEX, a través de la Gerencia de Inspección y Mantenimiento (GIM) dio a conocer, al Instituto Mexicano del Petróleo (IMP) su preocupación por no disponer de un sistema que opere en ambiente de red con procedimientos adecuados para el Control Interno Presupuestal.

Actualmente para llevar esta actividad operan sistemas alternos, los cuales están en ambiente monousuario o multiusuario como el sistema SICEP (Sistema Institucional de Control del Ejercicio Presupuestal) o el SIPRE (Sistema Institucional de Presupuestos) que son sistemas Institucionales que involucran todo el presupuesto de la estructura organizacional de PEMEX, siendo que la GIM requiere de un sistema que solo contemple el control interno presupuestal de la gerencia representada en distintas regiones. Ante este problema el personal del IMP propuso desarrollar un sistema que realice el Control Interno Presupuestal bajo ambiente de red de computadoras a nivel Región aplicando la tecnología de arquitectura Cliente / Servidor.

Ante esta propuesta nació el SISEFIN (Sistema de Seguimiento Financiero para la GIM), el cual forma parte de todo un Sistema Integral de información de la GIM. El sistema SISEFIN fue desarrollado por el personal del IMP a cargo de una dependencia con experiencia en desarrollos de sistemas Cliente/Servidor.

En la presente Tesis se cubre ampliamente la metodología orientada al diseño de una arquitectura cliente/servidor para el desarrollo del sistema SISEFIN, describiendo cada una

de las etapas por la cuales atravesó el sistema hasta llegar a su terminación y operación. Cabe mencionar que el sistema SISEFIN se proyecta como un ejemplo práctico de la metodología seguida en la tesis.

En el primer capítulo se detallan los **componentes de la arquitectura cliente/servidor** y como se relacionan, involucrando conceptos de bases de datos y redes para su fácil entendimiento.

En el siguiente capítulo se explica la primera parte de la metodología que es la **planeación del sistema SISEFIN** para su desarrollo en arquitectura cliente/servidor, empezando a conocer las necesidades del usuario y realizando un análisis sobre la situación actual en que opera la GIM con respecto al control del ejercicio presupuestal, para después elaborar la propuesta especificando los objetivos, alcances y beneficios, así como el tiempo y costo por el sistema.

Una vez aceptada la propuesta se prosigue al **desarrollo del sistema** y esta es la etapa que consume más tiempo y se expone en el capítulo 3 y 4, donde el análisis y diseño juega un papel primordial para la construcción exitosa del sistema en ambiente cliente/servidor, éste análisis debe ser un estudio a detalle, cubriendo las expectativas del usuario. A fin de saber si el análisis y diseño se esta cumpliendo, es necesario crear un prototipo para que el usuario vea mediante pantallas funcionales como va a quedar el sistema y la conectividad que se va a implantar.

Siendo aprobado el análisis y diseño se procede a la construcción del sistema con todo lo que respecta a su programación y al ambiente cliente /servidor en que va a operar. En este último capítulo también se expone la implantación del sistema cuando esta ya es terminado, involucrando lo que es la instalación como la actividad donde se arma la arquitectura cliente/servidor y se instala el sistema SISEFIN. La capacitación es la siguiente actividad que tiene como finalidad adiestrar a los usuarios a comprender y manejar el nuevo sistema con el apoyo de manuales técnico y usuario. Una buena implantación del sistema asegura la operatividad del sistema, permitiendo al usuario obtener beneficios.

La liberación es la última actividad, y la culminación de todo sistema, en esta etapa se hace la entrega oficial del sistema al coordinador del proyecto y se negocia su mantenimiento evolutivo.

Las **conclusiones** de la tesis expondrán los resultados sobre los sistemas que operan en arquitectura cliente/servidor, principalmente del SISEFIN, además de las experiencias que se obtuvieron durante su desarrollo, concretando la importancia y futuro de la arquitectura Cliente/Servidor y el papel que tiene la informática en nuestro mundo actual.

OBJETIVOS DE LA TESIS

OBJETIVO GENERAL

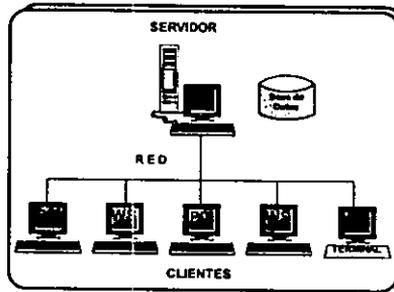
Analizar, diseñar y desarrollar un sistema que opere en arquitectura Cliente/Servidor para el control y seguimiento financiero de la Gerencia de Inspección y Mantenimiento en PEMEX-PEP.

OBJETIVOS ESPECIFICOS

- Conocer la plataforma Cliente/Servidor para el desarrollo de un sistema.
- Plantear la metodología de análisis y diseño para un sistema en arquitectura Cliente/Servidor.
- Establecer los procedimientos para la exitosa construcción e implantación del sistema Cliente/Servidor.
- Definir la tendencia informática sobre la operación en ambiente Cliente/Servidor.
- Introducir a las personas que tengan conocimientos en el área de desarrollo de sistemas los conceptos que se manejan en la arquitectura Cliente/Servidor

CAPITULO I

ARQUITECTURA CLIENTE/SERVIDOR



INFRAESTRUCTURA DE COMPUTO DE SISTEMAS EN ARQUITECTURA CLIENTE/SERVIDOR

Actualmente el éxito de una compañía depende de su forma de organizar y manejar adecuadamente sus recursos, obteniendo ventajas competitivas e incrementos de utilidades. Ante esta situación la tecnología informática tiende a proponer un panorama de alternativas para la implantación de plataformas de cómputo que faciliten el acceso confiable, veraz y oportuno de la información.

Una infraestructura de cómputo de sistemas en arquitectura Cliente/Servidor es un conjunto de instalaciones necesarias para el manejo automatizado de la información aplicando la tecnología C/S. El diseño y la construcción de la infraestructura dependen de los alcances y necesidades de la empresa cumpliendo con el principal objetivo servir a los usuarios de manera eficiente en las transacciones y generaciones de información para la toma de decisiones.

Para diseñar e implantar una buena infraestructura de cómputo Cliente/Servidor (C/S), se requiere conocer las características que nos ofrecen sus componentes y sobretodo tener cimentados los conocimientos sobre redes, bases de datos y la arquitectura C/S.

ASPECTOS GENERALES DE REDES

1.1. CONCEPTOS GENERALES DE REDES

Conforme el número de computadoras y usuarios crecieron, se hizo indispensable la creación de tecnologías que incorporan sistemas de comunicaciones, medios de gran almacenamiento, y equipos periféricos especializados. Fue así como surgieron las *Redes de*

Computadoras como un medio para llevar a cabo el óptimo aprovechamiento de los equipos y programas. En una Red de Computadora es posible transmitir información a un gran número de usuarios, compartir impresoras y discos duros, o bien realizar una transacción comercial vía satélite a cualquier parte del mundo.

Las Redes de Computadoras se han convertido en el medio de trabajo por excelencia de grandes y pequeñas empresas, debido a sus enormes ventajas. Esta importancia radica en el hecho de que las redes permiten a todos en una organización aprovechar todas las capacidades de computación, intercambiando y compartiendo los recursos sin importar su localización física.

1.1.1. DEFINICIÓN

Es la interconexión de sistemas de computadoras, canales y dispositivos de comunicación, los cuales tienen la misión de intercambiar información y servicios entre ellos, así como compartir recursos comunes. Utiliza un software que coordina el acceso a los dispositivos ubicados en diferentes posiciones geográficas. ^[1]

1.1.2. OBJETIVOS

- Eliminar el desplazamiento de los individuos en la búsqueda de información y en el acceso a equipo de alto costo.
- Proporcionar recursos computacionales a equipos distantes, tales como: Procesadores , Base de Datos, Periféricos, etc.
- Facilitar un proceso de comunicación entre los usuarios y datos.
- Servir como mecanismo de seguridad en el procesamiento de información por medio de respaldos.
- Proporcionar distribución a las operaciones.
- Presentar información adecuada para la toma de decisiones sobre los procesos que se realiza en la red.
- Proveer compatibilidad entre diferentes tecnologías de software y hardware.
- Aumentar la capacidad de procesamiento y almacenamiento disponible por cada uno de los usuarios en un momento determinado.
- Proponer alternativas de enrutamiento para el transporte de la información en caso de fallas en los medios de transmisión.
- Ofrecer acceso a servicios universales de datos.

1.1.3. UTILIDAD

Con las computadoras y las redes se ha entrado a la "Era de la Información". Estas herramientas revolucionarias han multiplicado la productividad y eficacia del trabajo, tanto

^[1] Vyles, Black.- *Redes de área local, Protocolos, Normas e Interfaces.*- 1ª Edición.- Editorial Macrobis.- México.- 1990.

para las empresas como usuarios individuales. Hoy por hoy, las computadoras en red facilitan el registro de transacciones que tienen lugar cada día una cadena comercial, se ocupan de las operaciones bancarias, gestionan las reservas de los hoteles, y existen muchas otras actividades económicas que dependen por completo de las redes.

1.1.4. VENTAJAS

- ✓ Permiten compartir datos y programas.
- ✓ Ayudan a compartir impresoras y recursos similares
- ✓ Brindan alta seguridad.
- ✓ Ayudan a la estandarización de software.
- ✓ Permiten una mejor integración hacia equipos mayores.
- ✓ Ayudan a la comunicación entre las personas.

Gracias a los sistemas de comunicaciones y a las redes de computadoras, hoy es posible el intercambio rápido de informaciones residentes en computadoras esparcidos por todo el país.

1.1.5. CLASIFICACIÓN

La clasificación de una red de computadoras depende del punto de vista a considerar, por tal motivo a continuación se presentan algunas de las principales clasificaciones:

Por el tipo de propietario de la red:

- a) **Redes Privadas:** Son las más comunes y normalmente pertenecen a universidades, bancos y empresas públicas y privadas. Se caracterizan porque sólo un grupo restringido de usuarios tiene acceso a la red (los propietarios de la red, socios, empleados o estudiantes).
- b) **Redes Públicas:** Son las administradas generalmente por el gobierno de cada país, utilizan la infraestructura de la red telefónica y ofrece sus servicios a cualquier organización que se suscriba a la red. Los servicios de transmisión de datos que se ofrece, son extremo económicos, debido a que se comparten canales de comunicación entre gran capacidad de usuarios.

De acuerdo a las extensión geográfica que ocupan:

a) **Redes de Area Local (LAN - Local Area Netware)**

Son redes que están confinadas a un espacio físico restringido, tales como oficinas, escuelas, etc. Comparten periféricos de costo elevado (graficadores, impresoras lasser, dispositivos de almacenamiento). Estas redes utilizan la tecnología de base de banda, y

gracias a su transmisión digital permiten tasas de errores muy pequeñas. Por sus características proporcionan un medio de alta velocidad.

La IEEE 802 define a una *red LAN* como un sistema de comunicación de datos donde permite a cierto número de dispositivos comunicarse entre ellos y se distingue de otras redes porque están limitadas por un área geográfica de 1 a 20 Km., dentro de una oficina, un edificio, un grupo de edificios o un campo de trabajo. La red general puede depender de un canal de comunicaciones de 1 a 20 Mbits, bajo retardo y bajo rango de error, además de que sus dispositivos son autónomos (operados por una sola organización) y pueden ser conectados usando una topología en bus o en anillo.

En su definición más simple podemos decir que cuando varias computadoras se entrelazan entre sí, utilizando algún tipo de cable, de modo que entre ellas puedan compartir recursos (datos, programas, impresoras, etc.), en distancias relativamente cercanas, entonces podemos decir que estamos hablando de un a *red local*.

Una red local es la que se utilizó para el desarrollo e implantación del Sistema SISEFIN.

b) Redes de Area Metropolitana (MAN - Metropolitan Area Network)

Son redes que cubre áreas geográficamente limitadas al tamaño de las ciudades. Utilizan una infraestructura de la red basada en fibra óptica. Esta red proporciona servicios de secciones consistentes en 1.544 Mbps, ya que el backbone^[2] soporta hasta 80 Mbps o más. Lo anterior posibilita la utilización de transmisión de voz, datos y voz por el mismo medio.

Típicamente una *red MAN* conecta LANs de más baja velocidad a través de una ciudad o región, solucionando las limitaciones de ancho de banda. Para salvaguardar todos los datos transmitidos, las *redes metropolitanas* emplean mecanismos de autorecuperación para asegurar el grado más alto de disponibilidad y confiabilidad de la red. Las *MANs* son diseñadas de manera que el transporte sea fácilmente compartido por muchos clientes.

c) Redes de Area Amplia (WAN - Wide Area Network)

Son aquellas en las que es necesario conectar equipos de comunicación remota a las computadoras que integran la red. La extensión geográfica que abarca una *red WAN* puede ir desde una pequeña ciudad hasta cubrir en su totalidad el territorio de un país. Se destacan por la utilización de tecnología analógica para realizar sus enlaces, así como su baja velocidad de transmisión de 10 a 64 Kbps.

Con el fin de interconectar computadoras en áreas geográficamente muy alejadas, o redes LAN y MAN, se requiere contar con mecanismos de acceso especiales que caen

[2] El Backbone de una red es la parte medular y central de sistemas de comunicaciones en otras palabras es la estructura de la red comunicaciones.

dentro del concepto "internetworking". La intercomunicación remota requiere emplear enlaces de microondas, fibra óptica y satélites. Las *WAN* emplean conmutación de paquetes, inicialmente a baja velocidad (hasta 19,200 bps) con el protocolo X.25. Sin embargo, el advenimiento de tecnologías más rápidas y los requerimientos de las nuevas aplicaciones (por ejemplo: multimedia), impulsan el empleo de nuevos protocolos como Frame Relay, con los que se alcanzan velocidades de Megabits/segundo.

Un ejemplo de *red WAN* es la que pertenece a PEMEX (Figura 1.1), la cual recibe el nombre de PEMEX-PAQ. Esta Red abarca el territorio nacional proporcionando que PEMEX envíe información entre sus instalaciones. Una *red WAN* puede contener redes LAN, con lo cual se permite un proceso de comunicación entre usuarios y equipos contenidos en la *red WAN*.

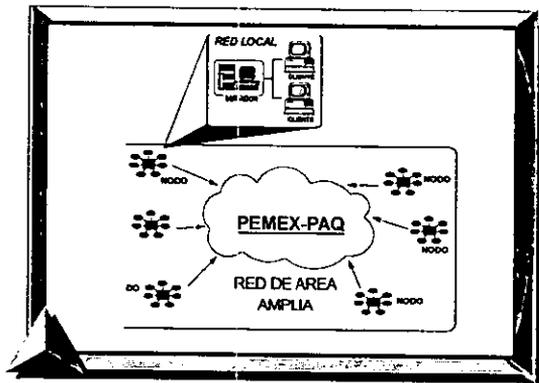


Figura (1.1) Red WAN

1.2. REDES DE AREA LOCAL

1.2.1. DEFINICIÓN

Las *redes locales* son las herramientas de comunicación por excelencia que permiten intercambiar información y compartir recursos. La topología y su protocolo de acceso son los dos conceptos básicos que definen su funcionamiento. Las más difundidas en la actualidad son Ethernet y Token Ring, y su protocolo de acceso está diseñado para la transmisión de datos a altas velocidades. Las redes locales del futuro serán aquellas capaces de transmitir tráfico de multimedia.

En menos de 20 años las redes locales dejaron de ser experimentos de laboratorio para convertirse en productos de enorme utilización en las empresas. Las redes locales son subredes de comunicación que permite la interconexión de diferentes dispositivos y su gran aceptación dentro de las empresas se debe a que facilitan el intercambio y permiten compartir recursos.

1.2.2. CARACTERÍSTICAS BÁSICAS

Las características principales que debe tener una red local son:

- Los equipos que la forman (computadoras) deben estar enlazados por algún medio físico, ordinariamente cable.
- Al menos uno de los equipos, deben ser capaz de compartir algún o algunos de los recursos hacia los demás equipos.
- Todos los equipos que conforman la red, deben estar físicamente cercanos (en el mismo edificio, planta, o en un mismo campus).

1.2.3. VENTAJAS

Las principales ventajas de una red local son:

- ✓ Permiten compartir datos y programas.
- ✓ Ayudan a compartir impresoras y recursos similares.
- ✓ Brindan alta seguridad con poderosos sistemas operativos.
- ✓ Ayudan a estandarización del software. Si todos los programas de aplicación residen en un servidor de archivos, y todos los usuarios accesan a sus programas desde él, entonces se podrá reglamentar un mejor el uso de software.
- ✓ Permiten una mejor integración hacia los equipos mayores, utilizando enlaces múltiples (gateways), facilitando que varios usuarios simultáneamente accesen los recursos de un equipo grande.
- ✓ Ayudan a la comunicación entre personas. Hablando de los ambientes de negocios lo común de una empresa es que existan archivos como memorándums u oficios que se compartan a través de la red.

1.2.4. COMPONENTES

Componentes de Hardware:

El Hardware se define como un conjunto de dispositivos físicos de los que consiste un sistema. Comprende componentes tales como: teclado, discos duros, impresoras, monitores, etc.

- **Estaciones de Trabajo:** Son microcomputadoras que utiliza el usuario para procesar su información. Para llevarlo acabo el usuario puede hacer uso de los recursos de su microcomputadora para utilizar unidades de memoria, impresoras de alta calidad, graficadores y módems.
- **Servidor de Red o de Archivos (File Server):** Es una computadora de alta capacidad y rendimiento (Mainframe, Minicomputadora o Estación de Trabajo) que tiene uno o

varios discos duros de alta velocidad, gran capacidad de memoria y varios puertos para conectar periféricos. Esta computadora permite compartir archivos (documentos, software, programas) que se encuentran en su disco duro y ofrece recursos a los usuarios, como el de impresión.

- **Tarjetas de Red:** Es un circuito electrónico que se instala en los equipos computacionales y permite transmitir y recibir bits de información a cierta velocidad a través del medio de comunicación que es el camino por donde fluye los datos. La tarjeta de red es muy importante porque además de enlazar la PC a la red, define el tipo de red.
- **Sistemas de Cableado:** Es la columna vertical de cualquier sistema de red ya que lleva la información de una nodo a otro.
- **Equipos de Comunicación:** Son dispositivos que enlazan redes, entre los más comunes están los módems, routers, bridges y gateways.

Componentes de Software:

El Software se define como un conjunto de instrucciones mediante las cuales la computadora realiza tareas. El software en la actualidad está tan extendido que sus usos abarcan sistemas operativos, lenguajes de programación de bajo y alto nivel así como programas que son producto de ellos mismos.

- **Sistema Operativo de Red:** El *sistema operativo* es un conjunto de programas que controla el flujo de información entre la computadora, dispositivos de Entrada/Salida y medios de almacenamiento. Por ello utiliza software de control que traduce los comandos a formatos que los dispositivos reconocen para realizar acciones. El *sistema operativo de red* es el conjunto de programas que se encargan de transferir archivos de una estación de trabajo a otra, administrar el acceso y seguridad de los archivos y controlar el uso de periféricos en la red. Hay muchos sistemas operativos, cada uno con características propias que las diferencian de otras.

El sistema operativo de red tiene dos partes:

La del Servidor de Archivos: Novell NetWare es uno de los más populares en el mercado. IBM PC-LAN es un sistema utilizado en redes Token Ring. LAN MANAGER es el sistema operativo patrocinado por Microsoft. Otros son SYNTAX, LAN Work Place, Windows 95, Windows NT.

La de las Estaciones de Trabajo. Casi siempre es MS-DOS de Microsoft Sistema Operativo de Disco o Windows 95.

- **Software de Aplicación:** Es el software con todos los paquetes que el usuario necesita para realizar su trabajo. Este software puede ser tan amplio como se requiera: procesadores de palabras, sistemas administrativos, de contabilidad y áreas afines, sistemas especializados (por ejemplo control de la producción o una cartera de clientes), correos electrónicos, etc.

Ya que estos equipos son utilizados por usuarios que desarrollan aplicaciones, es necesario que un especialista resuelva los problemas que a menudo se presentan. A esta persona o grupos de personas se les denomina *Administradores de Red*.

En la siguiente figura (1.2) se muestran los componentes propios para instalación y puesta en marcha de una Red.

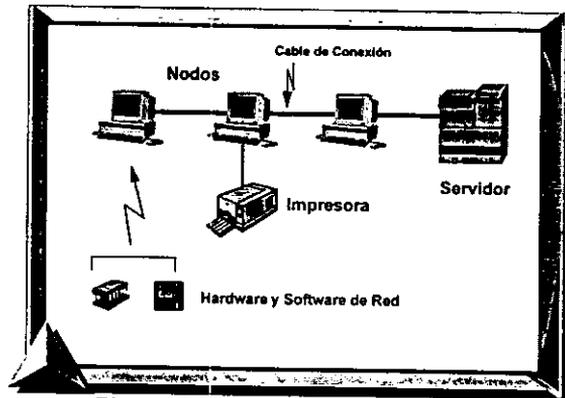


Figura (1.2) Componentes de una Red

1.2.5. TOPOLOGÍAS

La *topología* es la configuración de una red, la interconexión física de la misma. La topología es la forma en que se conecta físicamente las computadoras a la red, dependiendo de lo que se tiene y lo que se quiere. Las 3 topologías básicas de las redes de área local son: anillo, bus y estrella. Los medios de transmisión empleados con mayor frecuencia en estas topologías son el par trenzado, el cable coaxial y la fibra óptica. ^[3]

Las topologías de anillo y bus son las más empleadas en la actualidad. Originalmente las redes en anillo se desarrollaron en Europa y las redes en bus en los Estados Unidos.

Los tres objetivos principales de las topologías son:

- Proporcionar la máxima fiabilidad posible, para garantizar la recepción correcta de todo el tráfico. Ello incluye también la capacidad de recuperación de errores o información perdida en la red. La fiabilidad está relacionada también con el mantenimiento del sistema.
- Encaminar el tráfico entre la estación transmisora y receptora a través de la ruta más económica dentro de la red, evitando el menor número posible de componentes intermedios.

[3] Mayo Guzmán, Laura.- Topologías de red, tendencias en la selección de redes.- Revista Red.- Red, S.A de C.V.- No. 39.- 1994.- México.- pp. 20 a 25.

- Proporcionar al usuario final un tiempo de respuesta óptimo, reduciendo el retardo entre la transmisión y la recepción de los datos de una estación a otra.

Las topologías más comunes son:

- ↳ Topología en Anillo.
- ↳ Topología Horizontal (bus).
- ↳ Topología en Estrella.

TOPOLOGIA EN ANILLO

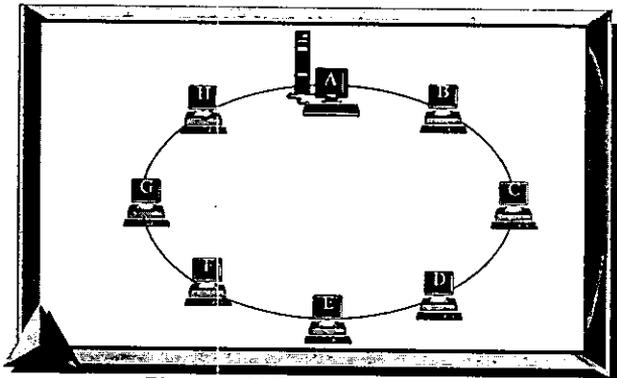


Figura (1.3) Topología en Anillo

Características

- Es una configuración bastante extendida.
- Se llama así por el aspecto circular del flujo de datos.
- Los datos fluyen en una sola dirección, y cada estación recibe la señal y la retransmite a la siguiente del anillo.
- Casi no se presentan embotellamientos.
- La lógica necesaria para poner en marcha una red de este tipo es relativamente simple.
- Cada componente lleva a cabo una serie de tareas muy sencillas: aceptar los datos, enviarlos a la estación conectado al anillo o retransmitirlos al próximo componente del mismo.
- Las estaciones que integran al anillo envían los datos en forma de tramas que también contienen información de control y las direcciones de las estaciones de origen y destino.

Desventajas

- Como todos los componentes del anillo están unidos por un mismo canal, si falla el canal entre dos estaciones, toda la red se interrumpe. Con el fin de evitar este problema se han ideado diseños especiales que incluyen canales de seguridad, o también, se han construido conmutadores que redirigen los datos automáticamente, saltándose la estación averiada, hasta la siguiente estación del anillo.

TOPOLOGIA HORIZONTAL - BUS

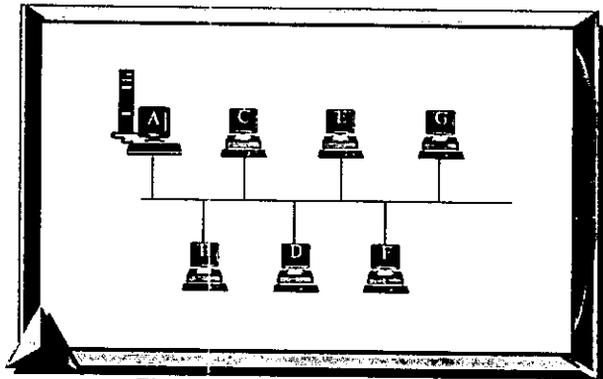


Figura (1.4) Topología Horizontal

Características

- Es frecuente en las redes de área local.
- Es fácil controlar el flujo de tráfico entre las distintas estaciones, ya que el bus permite que todas las estaciones reciban las transmisiones, es decir, una estación puede difundir la información a todas las demás.
- Suele existir un sólo canal de comunicaciones para todos los dispositivos de la red.

Desventajas

- Como existe un sólo canal de comunicaciones, si éste falla, toda la red deja de funcionar. Algunos fabricantes proporcionan canales redundantes por si falla el canal principal, y otros ofrecen conmutadores que permitan rodear un nodo en caso de que falle.
- Dificultad de aislar las averías de los componentes individuales conectados al bus.

TOPOLOGIA EN ESTRELLA

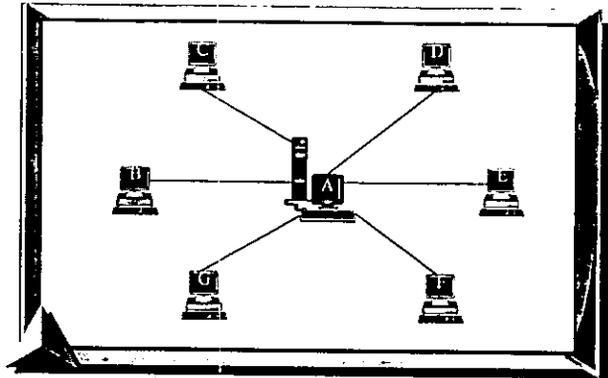


Figura (1.5) Topología en Estrella

Características

- Resulta fácil de controlar.
- Su software no es complicado.
- Su flujo de tráfico es sencillo. Todo el tráfico emana del núcleo de una estrella.
- La estación central es por lo general un servidor, posee el control total de las estaciones conectados a ella.
- Cada estación de trabajo puede comunicarse solamente con la estación central y no con los demás nodos de la red.
- La transferencia de información de un nodo a otro sólo es posible enviando los detalles al nodo central, el cual, a su vez, los envía al destino.
- La estación central es responsable de encaminar el tráfico hacia el resto de los componentes, además, se encarga de localizar las averías con sólo aislar las líneas para identificar el problema.

Desventajas

- Su capacidad de procesamiento distribuido es limitada.
- Sufre situaciones y problemas en caso de avería de la estación central.
- Presenta serios problemas de fiabilidad, debido a su carácter centralizado. Como medida de seguridad se hace necesario establecer redundancia en la estación central.

1.2.6. PROTOCOLOS

El término *protocolo* se refiere a las reglas que permiten a distintos dispositivos comunicarse entre sí de tal forma que cada uno pueda enviar y recibir señales comprensibles. El protocolo de comunicación es el que permite transportar los datos del

usuario a través de la red, así como es un elemento participe en el establecimiento de la infraestructura de sistemas distribuidos. Por lo tanto es necesario su estudio, ya que la elección correcta del protocolo, así como su configuración dependerá la interoperabilidad de los componentes de la arquitectura.^[4]

Para que dos computadoras se puedan comunicar se requiere un *protocolo* que coordine el envío y recepción de mensajes entre éstas. El propósito primario de un protocolo es el establecer el intercambio ordenado de información entre diferentes procesos y equipos, con la finalidad de optimizar la administración de recursos.

Entre las acciones que realizan los protocolos están:

- Controlar y sincronizar el flujo de información en el canal de comunicación.
- Contemplar ruidos e interferencia en la línea.
- Utilizan señales de confirmación de mensajes recibidos.
- Detectar errores, corregirlos y retransmitir el mensaje.
- Ordenar el intercambio de mensajes y sincronización.
- Establecer y terminar la sesión entre usuarios de la red.
- Administrar los recursos, monitoreo y protección.
- Transparencia entre los procesos de los usuarios con los otros nodos.
- Seguridad y privacidad.

Existen muchos protocolos de comunicación, en PEMEX los utilizados son: IPX y TCP/IP.

PROTOCOLO IPX (NETWARE)

IPX (Internetwork Packet Exchange) es el protocolo propio de las redes de Novell NetWare, fuerza dominante en la operación de redes en el mercado, y su protocolo IPX es un servicio que permite enviar y recibir información en una forma de mensajes a través de la red. IPX sigue los cinco niveles de XNS (Xerox Network Systems) que fue desarrollado por la corporación Xerox. Los protocolos XNS inicialmente fueron diseñados y operados para redes de área local pero estos no trabajaron eficientemente sobre redes de área amplia. Para compensar esta debilidad heredaron los protocolos IPX con características que hace más eficiente el diseño de redes locales.

IPX actúa a través de protocolos como SPX (Sequenced Packet Exchange) para mejorar la seguridad del servicio de flujo de datos. SPX es la capa de transporte que orienta la conexión y el servicio de circuito virtual entre las estaciones de la red. Pero como IPX no es un protocolo estándar, se debe recurrir a dispositivos convertidores de protocolos cuando se requiera enviar y recibir datos de otras redes, lo que incrementa considerablemente el costo de la red.

^[4] Vlyess, Black. - *Redes de área local, Protocolos, Normas e Interfaces* - 1ª Edición. - Editorial Macrobis. - México. - 1990.

PROTOCOLO TCP/IP

TCP/IP (Transmission Control Protocol/Internet Protocol - Protocolo de Control de Transmisión /Protocolo Internet) es un conjunto de protocolos creados en los 70s por Vince Cerf para un proyecto en el Departamento de Defensa de E.U., más tarde se convirtió en el protocolo por omisión de los sistemas UNIX. TCP/IP se basa en el modelo OSI (Open System Interconnect - Interconexión de sistemas abiertos) que consta de siete niveles definidos por ISO (International Standard Organization - Organización Internacional de Estándares) para facilitar la comunicación de ambientes heterogéneos. Por lo que TCP/IP es un protocolo estándar mundialmente aceptado.

Con los protocolos TCP/IP se logra independencia del hardware, proporcionando una comunicación continua, rápida y eficiente con otras computadoras. El protocolo TCP/IP es utilizado por las redes tipo Ethernet.

TCP/IP es una familia de protocolos que trabajan en conjunto para proporcionar la ruta de comunicación de datos entre redes. Sus cinco componentes son:

COMPONENTES	DEFINICIONES
IP (Internet Protocol - Protocolo Internet)	Es el encargado de definir la mejor ruta y enviar por ella los paquetes. Esta diseñado para proporcionar un servicio sin conexión. Es decir, el IP no informa al receptor sobre el la recepción correcta de paquetes, sólo entrega el mensaje. Esto significa que si el enlace de datos falla, el nivel IP no informará tal acción. En tal caso dependerá de un protocolo superior llevar a cabo la recuperación de la información. Cada equipo dentro de la red tendrá una dirección que lo identifique de los demás, esto se conoce como <i>dirección IP</i> (IP Adrees). El propósito del direccionamiento es permitir al IP comunicarse entre computadoras en la red o entre redes. Este número se conforma de cuatro octetos ^[5] de bits o cuatro números decimales separados por puntos. La organización y uso de estos números que van de 0 a 255 corresponden al tamaño y tipo de red. Por ejemplo: 198.168.5.13 es la dirección IP de un servidor del IMP.
ARP (Address Resolution Protocol - Protocolo para Determinación de Direcciones)	Permite el enlace entre la dirección IP a la determinación de la estación física. Esto se realiza con la finalidad de aprovechar las características de la dirección IP, la cual no identifica a las computadoras a nivel físico solo en el nivel de red.
RIP (Routing Information Protocol - Protocolo de Información de Enrutamiento)	Es un protocolo que permite automáticamente actualizar las tablas de enrutamiento para controlar el intercambio de información entre hosts y gateways. Con la información RIP cualquier ruteador conoce la longitud de la ruta más corta, que no necesariamente es la mejor.

[5] Un octeto es un conjunto de ocho bits

COMPONENTES	DEFINICIONES
ICMP (Internet Control Message Protocol - Protocolo de Mensajes para el control de Internet)	Genera mensajes de error o mensajes de control a través de la red. Existen diferentes tipos de mensajes, de los cuales uno de los más comunes es el programa PING (Packet Internet Gopher - Trama de búsqueda de Internet), es un mensaje que trata de localizar a otras estaciones en la red para ver si están activas o si la ruta es accesible
UDP (User Datagram Protocol - Protocolo de Datagrama de Usuario)	Es la transmisión de mensajes, ya sea de usuarios o de control. La transmisión UDP no genera mensajes de confirmación de recepción.
TCP (Transmission Control Protocol - Protocolo de Control de Transmisión)	Es un protocolo orientado a conexiones que permite que los datos sean intercambiados con cualquier otra estación de la red en forma segura. TCP garantiza la llegada de paquetes y su ordenamiento.

Niveles de Aplicación de TCP/IP.

- **TELNET** (Network Terminal Protocol - Protocolo Terminar de Red). Es un servicio de terminal virtual. Es decir proporciona acceso remoto en la red. La computadora Cliente se comporta como una terminal, pasando a ser una estación remota.
- **FTP** (File Transfer Protocol - Protocolo para la Transferencia de Archivos). Servicio de alto nivel bajo ambiente TCP que permite y controla el proceso de transferencia de archivos a través de una red.
- **SMTP** (Simple Mail Transfer Protocol - Protocolo de Transferencia de Correo Simple). Establece las transferencias de correo electrónico a través de una red.
- **DNS** (Domain Name Server - Servidor de Nombres de Dominio). También llamado servicio de nombres, esta aplicación mapea las direcciones IP para nombrar a los dispositivos de red.
- **NFS** (Network File System - Sistema de Archivos de Red). Es un sistema distribuido de archivos. Su función es la de compartir archivos entre varias computadoras asignadas a la red.

Ambos protocolos IPX y TCP/IP pueden trabajar sin ningún problema en una misma red LAN (Figura 1.6). En PEMEX esta situación es muy común, donde su file server (Servidor de Archivos) es administrado por Novell y su database server (Servidor de Base de Datos) con sistema operativo UNIX tiene como protocolo TCP/IP.

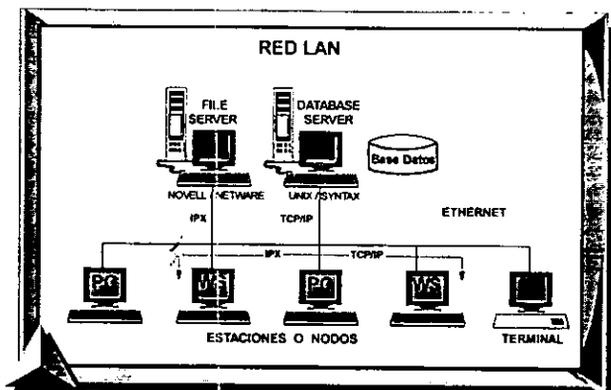


Figura (1.6) Red de Area Local

1.2.7. MEDIOS DE TRANSMISIÓN

El medio de transmisión o canal de comunicación es el camino físico entre el transmisor y el receptor, siendo una parte importante de la red. Las características de cada tipo de medio influyen en la velocidad, el costo y la distancia de transmisión. A continuación se exponen los principales medios de transmisión.

CABLE COAXIAL

Es similar a los cables usados en la TV. El *cable coaxial* se conforma por un alambre conductor básico cubierto por una placa metálica que actúa como tierra. El alambre conductor y la tierra se encuentran separados por un aislante plástico y, finalmente, todo el conjunto está protegido por una cubierta, a la que por lo común se llama jacket. Este cable sigue el esquema 10 base 2.

Un cable coaxial tiene un número importante de ventajas como medio de transmisión. El principal es su amplio ancho de banda de transmisión, lo cual permite transmitir altas frecuencias con mínimas pérdidas. Esto se aprovecha para enviar un gran número de datos a alta velocidad.

Los cables coaxiales pueden ser de varios tipos y anchos. Sin embargo, su principal característica es que pueden transportar una señal eléctrica a mayor distancia entre más grueso es el conductor. El cable grueso suele ser más caro y menos flexible.

Algunas de sus principales ventajas de este tipo de cable son:

- Fácil instalación.
- Mayor velocidad de transmisión: 500 Mbps.
- Compatibilidad con Ethernet y Arcnet.
- Ancho de banda del 10 Mbps - 350 Mhz.

- Distancias hasta de 600 metros sin necesidad de repetidores.
- Muy buena tolerancia a interferencias debidas a factores ambientales.

En general la alternativa de colocar cables coaxiales en redes locales tienen una relación de costo beneficio muy buena.

CABLE TELEFONICO

El *cable telefónico* se forma principalmente por dos alambres de cobre que se encuentran aislados por una cubierta plástica y torcidos uno contra el otro. El *par torcido*, a su vez, se encuentra cubierto por un aislante en la capa exterior denominada *jacket*. Su velocidad de transmisión es de 4 Mbps a 16 Mbps, con un ancho de banda de 350Mhz. Existen dos principales cables telefónicos:

UTP (Unshielded Twisted Pair - cables de par torcido no blindado): Son cables con los conductores de cobre más delgados y menos protegidos por un *jacket*, son baratos, flexibles y permiten manipular una señal a una distancia máxima de 110 metros. Hay cinco categorías de UTP, la categoría tres es la más usual en la construcción de la red, aunque la categoría cinco es la que permite la transmisión de voz, vídeo y datos. Básicamente, las computadoras y otros dispositivos están conectados a Hubs que a su vez estos se comunican con los servidores. **HUB** es aquel que tiene un agente SNMP (Simple Network Management Protocol - Protocolo para el Manejo de una Simple Red) y es un dispositivo que controla el tráfico, configuración, monitoreo y capacidad de manejo y rendimiento de la red.

STP(Shielded Twisted Pair - cables de par torcido blindado): Son más caros y menos flexibles que los UTP, pero permiten un rango de operación de hasta 500 metros y es mucho más inmune al ruido.

En resumen, los cables telefónicos tienen como principales ventajas:

- Aprovechamiento del cableado telefónico que ya existe
- Facilidad y rapidez de instalación.
- Compatibilidad con Ethernet y Token Ring.
- Ancho de banda de 10 Mbps.
- Esquema 10 base T.
- Distancias de hasta 110 metros con cables UTP y hasta 500 metros en caso de cable STP.
- Excelente relación de precio.
- Buena tolerancia a interferencias debido a factores ambientales.

Comparando el cable coaxial con el telefónico, el primero soporta velocidades de transmisión de datos mucho más altos y puede utilizarse para conectar estaciones más lejanas, además de que resiste la interferencia electromagnética.

FIBRA OPTICA

El cable de fibra óptica se compone de una fibra muy delgada de vidrio, de un décimo de milímetro de diámetro, dentro del cual circula una onda luminosa y cuya potencia se modula en función de la señal que va a ser transmitida. La fibra está elaborada por dos tipos de vidrio con diferentes índices de refracción, uno para la parte interior y otro para la parte exterior. Esta diferencia en la refracción previene que la luz penetre en una parte de la fibra óptica hasta la parte exterior, evitando así la pérdida de información. Una o más fibras pueden agruparse para formar un cable de fibra óptica. Así mismo para aumentar su resistencia mecánica, la fibra se encuentra cubierta por una placa aislante y protectora en la parte más exterior para darle mayor integridad estructural al cable. Es, sin embargo, extremadamente flexible.

Para la transmisión de la información en redes locales vía fibra óptica se utiliza una fibra como transmisor y otra como receptor. Las distancias máximas obtenidas para redes locales son de 2000 metros de nodo a nodo.

El elevado valor de la frecuencia de la onda portadora da una capacidad de transmisión muy alta; una sola fibra óptica puede transportar el equivalente de varios miles de canales telefónicos simultáneos. La gran pureza de los materiales utilizados y la tecnología hacen que las pérdidas en la línea sean muy pequeñas, aumentando considerablemente la distancia de transmisión. La fibra óptica, ligera, poco voluminosa, con una gran capacidad de transmisión es absolutamente insensible a las perturbaciones como interferencias de radio.

Normalmente se emplea por tres razones básicas:

- Para aquellos casos en donde las grandes distancias son un factor determinante para la implantación de una red local.
- Cuando se requiere una alta capacidad de aplicaciones de comunicación.
- Cuando el ruido o cualquier interferencia son factores a considerar.

Entre las principales ventajas de la fibra óptica se encuentran las siguientes:

- ✓ Transmisión de voz, video y datos por el mismo canal.
- ✓ Excelente velocidad de transmisión 2 Gbps.
- ✓ Aplicaciones de alta velocidad.
- ✓ No genera señales eléctricas o magnéticas.
- ✓ Inmune a interferencias eléctricas.
- ✓ Puede propagar una señal a distancias de hasta 2000 metros.

- ✓ Tiene un ancho de banda de 200 Mbps - 2 Ghz.
- ✓ Compatibilidad con Ethernet, Token Ring y FDDI .
- ✓ Excelente tolerancia a factores ambientales.
- ✓ Ofrece la mayor capacidad de adaptación a nuevas normas de rendimiento.

MEDIOS INALAMBRICOS

Este medio de transmisión no requiere de un camino físico sobre el cual se pueda propagar la información. Sin embargo esta ventaja no es posible aprovecharla extensamente debido a la absorción de la energía electromagnética por los medios materiales. Los principales medios inalámbricos son: el horno de microondas y enlaces satélites.

Microondas

La mayor parte de las transmisiones de datos y voz a larga distancia usa instalaciones de microondas. Las estaciones de envío y recepción llevan la transmisión por el aire. Las estaciones repetidoras, separadas por una distancia de 25 a 30 millas, mandan y reciben datos a lo largo de rutas de visión.

Puesto que no hay necesidad de colocar cables para las microondas, su uso es ideal en muchas áreas con terreno agreste (donde no bloquee la línea de visión). Sin embargo, en las áreas congestionadas hay demasiadas antenas instaladas, las cuales empiezan a interferir entre sí.

Satélite

Los datos se transmiten desde las instalaciones del usuario a una estación terrena, de donde se envían a un satélite localizado en el espacio, en una órbita controlada. Un analizador en el satélite recibe la señal y la retransmite a otro destino en la Tierra. En algunos casos, hay que utilizar un satélite intermedio cuando no es posible la transmisión directa a las instalaciones.

Un creciente número de satélites se están poniendo en órbita y la frecuencia de las transmisiones por satélite está aumentando rápidamente. Este tipo de transmisión es una buena alternativa a la transmisión por microondas cuando esta última se ve impedida debido a la interferencia por congestionamiento de tráfico. Este método tiene una muy alta velocidad y relativamente carece de errores.

Canales EI

Son canales de comunicación de datos vía microondas y soportan un máximo número de conexiones. A través de estos canales es más directa y rápida la comunicación, para lo cual se crea una ruta a través de circuitos virtuales.

1.2.8. CABLEADO ESTRUCTURADO

Actualmente, en la construcción de edificios destinados a oficinas corporativas, se considera imprescindible incluir un sistema de cableado estructurado para la formación de redes de voz y datos. Este sistema permitirá agregar posteriormente servicios de seguridad y alarmas para tener un edificio inteligente. Las empresas que ya cuentan con el inmueble, están realizando inversiones para aprovechar las ventajas que esta tecnología ofrece.

Los constantes cambios en la ubicación de los equipos, o la necesidad de contar con diferentes tipos de servicios en los puestos de trabajo (conexiones a redes locales, terminales de equipos mayores como mainframes, líneas telefónicas digitales o analógicas, etc.), ocasiona gastos elevados de materiales y mano de obra. Además de la suspensión total del servicio hasta que se complementaban las modificaciones.

Una de las ventajas principales del *cableado estructurado* radica en que los administradores de las redes pueden realizar cambios en la ubicación o tipo de servicios, a un costo sumamente bajo y sin interrupción del trabajo del resto de los equipos. A continuación se resume en una tabla las diferencias entre las instalaciones de cableado estructurado contra las normales.

ESTRUCTURADO	NO ESTRUCTURADO
➤ Independiente de las aplicaciones	➤ Tiende a ser dependiente de las aplicaciones
➤ Permite el movimiento del equipo sin cablear	➤ Se cablea al mover el equipo
➤ Facilidades para reconfigurar alambrado y equipo	➤ Difícil hacer reconfiguraciones.
➤ Diseño modular	➤ Diseño no modular
➤ Permite aislar equipo para la detección de fallas	➤ Difícil localización de fallas
➤ Soporta varias plataformas	➤ Dependiente del tipo de red.

El sistema de cableado estructurado consiste en varias familias de componentes que incluyen medios de transmisión (cables), paneles de parcheo para la administración de los servicios y accesorios como conectores, adaptadores y filtros.

Para la instalación de este sistema se emplea una topología en estrella, con lo cual se tienen las siguientes ventajas:

- ✓ Es aplicable tanto a sistemas centralizados como distribuidos.
- ✓ Proporciona una configuración flexible para la introducción de nuevos servicios y segmentación de las redes a fin de dividir el tráfico.
- ✓ Proporciona puntos centrales para el mantenimiento.
- ✓ Permite la expansión de cableado a un costo muy bajo.
- ✓ Ofrece facilidades para mudarse a otras tecnologías.

Las nuevas definiciones de tipo de cable que las redes pueden usar (ethernet y token ring, por ser las más comunes), permiten que la topología estrella en el cableado estructurado soporte las configuraciones lógicas de redes: bus y anillo.

Estos edificios cuentan con una vertical. Esta es la zona en donde se instala todos los dispositivos de interconexión y los paneles de parcheo. Desde ahí se distribuye el cableado hasta los puestos de trabajo, formando así lo que se llama la horizontal. En esta horizontal se emplea cable par trenzado no blindado (UTP) que se conectan a concentradores modulares uno con otro y usualmente a la sala central de computadoras. El cableado estructurado facilita la incorporación de sistemas de administración con el fin de detectar los problemas en la red justo en el momento en que estos ocurren.^[6]

CONCENTRADOR (HUB)

Es un dispositivo que controla el tráfico, configuración, monitoreo y capacidad de manejo y rendimiento de la red a través del protocolo SNMP (Simple Network Management Protocol). Un concentrador necesita ser flexible para expandir el área de red, para soportar protocolos de diferentes redes locales y manejar cada protocolo dentro del concentrador.

Los concentradores son diseñados con el fin de soportar redes que combinan topologías y tipos de cableado. Mas usuarios, más computadoras físicas a conectar condujo al desarrollo del primer concentrador de red (HUB). Este concentrador combinado con algunos repetidores multipuerto, permite compartir energía y señales a través de una plataforma común.

Para asegurar la interoperatividad, la industria desarrollo el estándar "10baseT" para Ethernet sobre un par trenzado no blindado (UTP) que sirve como segmento de 10 mts., esta tecnología elimina las molestias del cableado, haciendo a la Ethernet menos cara y más eficiente

1.2.9. INTERCONECTIVIDAD

En una arquitectura Cliente/Servidor los sistemas son estructurados para formar un conjunto de componentes interconectados. Siempre que se transmiten datos, debe existir un medio de interconexión entre los componentes de cómputo y los canales de comunicación (Figura 1.7). Cuando se enlazan redes LAN para formar una red más grande, a menudo se utilizan los siguientes componentes:

- a) **Modems (Modulador/Demodulador):** Dispositivo que convierte las señales digitales de la computadora a su forma analógica para la transmisión (modulación). Al recibir los datos en el otro extremo, se convierte de forma analógica a digital (desmodulación) de tal forma que la computadora que los recibe los pueda procesar.

[6] Oropesa Talavera, Enrique.- *El sistema de Cableado Estructurado.*- Revista Personal Computing.- Sayrols, S.A de C.V.- No. 72.- Mayo de 1994.- México.- pp. 76 a 78.

- b) **Repetidores (Repeater):** Dispositivo que retransmite y amplifica la señal eléctrica entre secciones de cableado, por lo que permite la interconexión de segmentos individuales de redes pequeñas, para formar grandes configuraciones. Actúa solamente en el nivel uno del modelo OSI y se utiliza cuando la distancia pasa de 100 mts entre PCs.
- c) **Puentes (Bridge):** Dispositivo que permite conectar dos redes similares a distintos medios físicos. Por ejemplo puede permitir el acceso de una señal proveniente de cable coaxial a fibra óptica.
- d) **Ruteadores (Router):** Dispositivo que toma un paquete y lo envía del punto A al punto B, después de analizar cual es el camino óptimo entre varios nodos intermedios para llegar a su destino. Esto se logra gracias a la información que cada ruteador almacena sobre todos los nodos de la red. Existen otros equipos conmutados (*switches*) que tienen la misma funcionalidad que los ruteadores. El tiempo de convergencia^[7] de un equipo conmutador es mucho menor al de un ruteador, ya que el conmutador es un dispositivo que se ubica en el nivel 2 del modelo OSI y que sólo trabaja interpretando direcciones físicas de las tramas, mientras que un ruteador necesariamente va a procesar direcciones lógicas de la aplicación, ya que opera al nivel 3 del modelo OSI. Por esta razón el ruteador formará parte de la red de acceso que requiere mayor inteligencia.
- e) **Puertas (Gateway):** Dispositivo que permite conectar redes que funcionan con diferentes protocolos. Para ello direcciona datos y maneja la conversión de protocolos necesarios. La ventaja del gateway es su flexibilidad, sin embargo implica un costo mayor, así como un diseño de la red más elaborado. El gateway opera a nivel software, supongamos que se requiere conectar una base de datos SQL Server residente sobre Windows NT y una base de datos DB/2 puesta en un mainframe. Para lograr esto se necesita un gateway que traduzca entre protocolos TCP/IP y SNA (Systems Network Architecture - Arquitectura de Sistema de Redes) usando un gateway SNA.^[8]

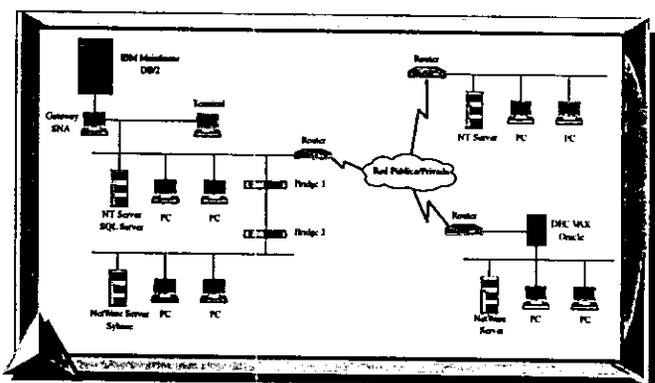


Figura (1.7) Componentes de Interconectividad

[7] El tiempo de convergencia es el tiempo que toma el proceso de selección averiguar la mejor ruta.

[8] SNA es una arquitectura de red creada por IBM.

1.2.10. TIPOS DE REDES LOCALES

Actualmente existen miles de opciones tanto en software como en hardware para sistematizar una oficina. La planeación de una instalación de cómputo se vuelve una condición para ofrecer un buen servicio como empresa.

En México se han realizado instalaciones de redes locales en los cuatro sectores más importantes de la economía: financiera, gubernamental, de servicios e industrial, pero es obvio que cada infraestructura de red presenta variantes como: espacio, equipo, software, personal propuesto, necesidades de comunicación, etc. Cada uno de estos elementos conforman las bases de una decisión, son las pesas que inclinan la balanza. Esto significa que las necesidades de un usuario cambian de acuerdo a sus actividades y a pesar de las múltiples opciones, sólo habrá un tipo de red que cubra al 100% las expectativas de una empresa.

A la fecha, las redes locales más importantes son: Ethernet, Token Ring y FDDI.

TOKEN RING

Token Ring trabaja en una *topología de anillo* conectada en forma de estrella, donde las estaciones de trabajo se enlazan alrededor de un concentrador/repetidor o MAU (Multiple Acces Unit - Unidad de Acceso Múltiple) que se encarga de unir dispositivos a la red. Esta línea de conectividad la ofrece IBM.

Esta topología corre a una *velocidad de 4 y 16 Mbps*, velocidades que representan su máxima en transferencia de datos, utiliza el *protocolo de comunicación Token Passing y cumple con la norma 802.5 de la IEEE* que incluye una capacidad de prioridad, la cual permite que algunas estaciones tengan más acceso que otras.

En Token Ring, el envío de datos a través de la red se hace por medio de una ficha (token) que circula permanentemente en el anillo a cierta velocidad; cada vez que la ficha pasa por alguna estación verifica si sus servicios son requeridos para llevar algún paquete de datos al servidor o a otra estación de trabajo (Figura 1.8).

Mientras no éste transmitiendo ninguna estación, un token estará circulando por el anillos y todas las estaciones estarán en estado de repetición. La recepción del token ofrece la oportunidad de transmitir una trama o una secuencia de tramas.

Cuando una estación tenga uno o varios paquetes listos para transmitir, les deberá agregar información de control y las direcciones de origen y destino, y esperará la llegada de un token. En el momento que lo tome, la estación pasará del estado de repetición al estado de transmisión. Las otras estaciones continuarán en estado de repetición y ninguna de éstas podrá transmitir datos por el anillo dado que el token esta ocupado. Para evitar que una estación transmita indefinidamente, se especifica un tiempo máximo de posesión del token, después del cual deberá emitirse un nuevo token que proporcione la oportunidad de

transmitir tramas a las otras estaciones, aunque cada estación puede poseer determinada prioridad y el token tiene indicadores de preferencia.

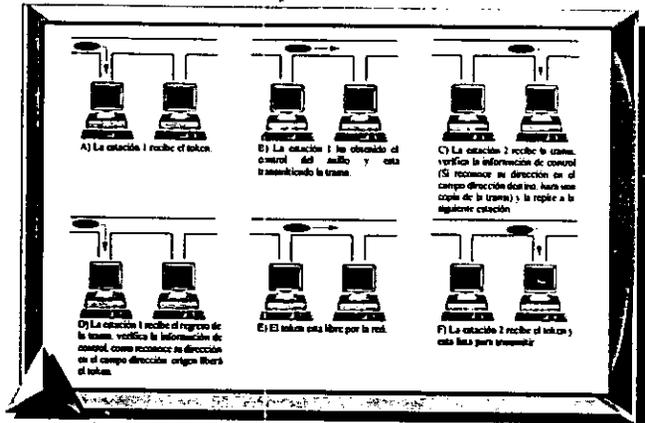


Figura (1.8) Funcionalidad de la Red Token Ring

Cada trama transmitida se repetirá de una estación a otra y la estación que reconozca su dirección en el campo de dirección destino, hará una copia de la trama. Una estación termina la transmisión de una o varias tramas, cuando la trama transmitida ha dado una vuelta al anillo, mientras esto no suceda la estación continuará en estado de transmisión. Si se cumple la condición la estación libera el token y pasa a estado de repetición.

Algunos usuarios hablan de la superioridad de estas redes sobre Ethernet, pues argumentan que su funcionamiento en un número considerable de estaciones de trabajo es superior a cualquier sistema de red. Una razón ante esta situación, es la *gran demanda* que *Token Ring tiene con los potenciales usuarios del sector financiero*, quienes manejan impresionantes cantidades de información y usuarios con tiempos de respuesta aceptables.

La velocidad es una de las grandes cualidades de Token Ring 16 Mbps y además, es un punto a favor sobre Ethernet que trabaja a 10 Mbps y presenta colisiones frecuentes al trabajar en ambientes demasiado concurrentes, pues el protocolo que utiliza CSM/CD (Carrier Sense Multiple access with Collision Detection) es quien la determina. *Token Ring se encuentra libre de colisiones* y su desempeño se degrada menos en redes no segmentadas; pero esta topología todavía tiene un defecto: su precio.

La expansión de *las redes Token Ring resulta cara* pues se necesita incrementar el número de repetidores en el anillo, lo cual limita su flexibilidad de despliegue. Un detalle como este, se compensa con la casi nula probabilidad de que la red se paralice; ya que *Token Ring utiliza las tarjetas controladoras con microprocesadores* que minimizan la posibilidad de la transferencia de información incorrecta.

A nivel cableado, factor trascendente en la instalación de una red, *Token Ring utiliza cable par torcido (twisted pair) o de fibra óptica*, lo cual es un atractivo más, ya que el costo del par torcido es bajo y de fácil manejo y se recomienda para ambientes con poca

interferencia electromagnética, donde será posible aislar grupos de estaciones de trabajo con un radio menor a los 110 m cuando estén conectadas a otros grupos más lejanos.

Cuando la instalación Token Ring se hace con fibra óptica, el precio es más elevado pues actualmente es la opción más cara, pero las ventajas en alta velocidad para la transmisión de datos y su inmunidad a la interferencia también la hacen la mejor decisión; sobre todo, si se toma en cuenta que una red con un tráfico de datos considerable requiere gran velocidad de procesamiento. Esta es otra razón de peso para el sector financiero, conformado en su mayoría por Bancos y Casa de Bolsa, donde se encuentra gran inclinación por Token Ring.

ETHERNET

Aunque sólo cuenta con una parte del sector financiero, *Ethernet* es la topología que posee más seguidores entre los grandes usuarios pues domina con un 55.9% el total del mercado mexicano.

Las razones son claras, Ethernet es un ambiente de comunicación entre computadoras que cumple con la norma IEEE.802.3, se basa en el principio de que cada estación de trabajo tiene la misma oportunidad de usar la red. La especificación incluye un algoritmo de justicia que impide que cualquier estación o grupos de estaciones monopolice la red.

Ethernet también se puede utilizar con distintas opciones de cableado, no requiere aditamentos especiales, su precio es menor que el Token Ring y brinda entre otras ventajas, la posibilidad de administrar recursos de hardware más fácilmente. El protocolo que utiliza Ethernet es CSMA/CD (Carrier Sense Multiple Access/Collision Detect - Acceso Múltiple con Sensado de Portadora y Detección de Colisiones) y con transferencia de datos a una velocidad de 10 Mbps.

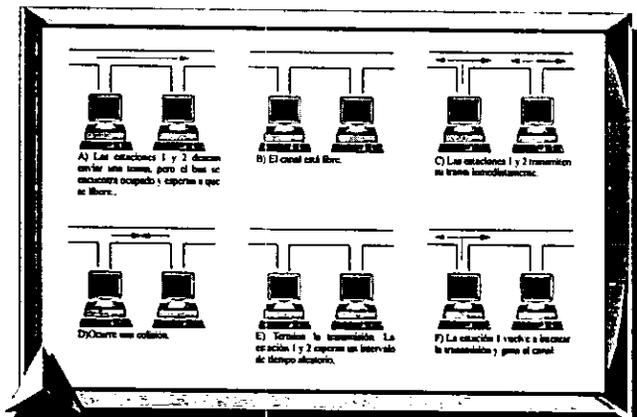


Figura (1.9) Funcionalidad de la Red Ethernet

Una estación que desea transmitir una trama escucha primero si alguien está transmitiendo (está sensando la presencia portadora). Si encuentra libre el medio envía la

trama. En caso de encontrarla ocupada, espera a que ésta se libere y transmite su trama inmediatamente (Figura 1.9).

Con el comportamiento anterior, es posible imaginar el siguiente escenario:

- El tiempo T1 la estación X inicia una transmisión.
- Poco después, en T2, la estación 1 desea enviar una trama pero encuentra el bus ocupado y espera.
- Más adelante, en T3, otra estación 2 también desea enviar una trama y encuentra ocupado el bus por X, por lo que también espera.
- En T4, X termina de transmitir y el cable queda libre, esto lo notan las estaciones 1 y 2 que inician la transmisión de sus respectivas tramas simultáneamente y las tramas interfieren entre sí. Se dice entonces que ha ocurrido una colisión.

Al detectar la colisión, la estación transmisora aborta su trama y en su lugar envía una señal para garantizar que las demás estaciones involucradas en la interferencia puedan detectar la colisión. Después de enviar esta señal, la estación espera durante un intervalo de tiempo "aleatorio" y vuelve a intentar la transmisión de la trama escuchando en el bus para ver si éste se encuentra libre. Conforme se va incrementando el tránsito, las probabilidades de colisión aumentan así como los tiempos de espera, precisamente para tratar de disminuir la posibilidad de colisiones múltiples.

El procesamiento y transmisión de datos es más rápido y atractivo si se parte del hecho de que en Token Ring, el uso de repetidores es extenso y por lo tanto la velocidad se reduce. A nivel cableado Ethernet ofrece más opciones de operación respecto a Token Ring, como: cable coaxial grueso o delgado, par torcido o fibra óptica. Físicamente Ethernet, se parece a las redes Token Ring pues las estaciones de trabajo *se conectan a través de un centro de cableado o concentradores* y estos podrán enlazarse a un bus de cable coaxial, fibra óptica o par torcido. Este último por la sencillez que ofrece en cuanto a instalación.

Fast Ethernet

Se trata de una tarjeta Ethernet que *pretende acelerar la velocidad de transmisión de la información* 10 veces más que el estándar actual, es decir, *de los 10 Mbps a los 100 Mbps.* Fast Ethernet, *trabaja con el protocolo Access Collision Detection de Apple* en el CSMA/CD (Carrier Sense Multiple Access/Collision Detection) que utilizan las PC's.

Para volver más capaz la transmisión de datos con Ethernet, esta tarjeta *permite tener comunicaciones por fibra óptica sin necesidad de utilizar FDDI o ATM* y esto coloca a Ethernet nuevamente como una opción muy atractiva.

FDDI

FDDI (Fiber Distributed Data Interface - Interfase de Datos Distribuidos por Fibra) es una *red de alta velocidad* y gran desempeño, desarrollada por la ANSI (American National

Standard - Instituto Estadounidense de Estándares Nacionales). *Esta red fue creada principalmente para soportar las nuevas aplicaciones de gráficas y visualización que requieran la transferencia de millones de bytes en intervalos de tiempo muy cortos. FDDI es una evolución natural de la red Token Ring.*

FDDI es una red local con una *velocidad de 100 Mbps basada en fibra óptica*. El funcionamiento de FDDI está *basado en mecanismos de operación* de otras redes locales, en particular los *especificados en la norma IEEE 802.5 (Token Ring)*.

Las redes FDDI pueden operar en *distancias de hasta 200 kms y a 125 Mhz*. Esta habilidad manipula eficientemente a alta velocidad múltiples paquetes. Un anillo de fibra óptica *puede incluir hasta 1000 nodos*, que pueden estar separados hasta 2 kilómetros, y la circunferencia del anillo puede llegar hasta 200 kilómetros y no exige que todos sus nodos sean necesariamente de fibra óptica.

En FDDI se especifica una topología en la que existen dos anillos de fibra óptica independientes y de rotación inversa, donde se da una velocidad global de 200 Mbps, 100 para cada uno de los canales. El canal interno enlaza solo determinados dispositivos, que están conectados a ambos anillos (internos y externos), tienen clasificación A. Los dispositivos tipo B solo están unidos a un anillo. Lo interesante de esta especificación es que permite designar con la clasificación A a las estaciones críticas que necesiten apoyo adicional y canales de mayor velocidad. Las estaciones de menor importancia, como estaciones de trabajo aisladas, pueden dejarse como estaciones de clasificación B, trayendo como consecuencia un costo inferior (Figura 1.10).

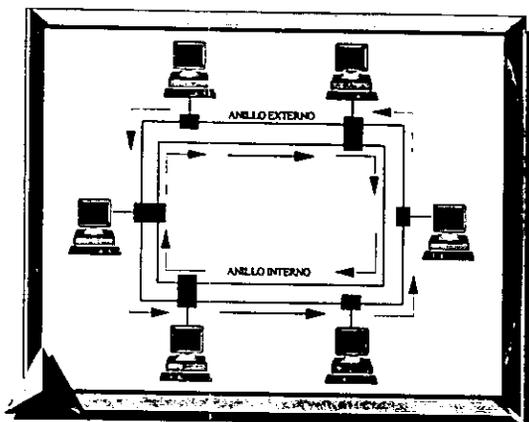


Figura (1.10) Funcionalidad de la Red FDDI

FDDI recurre a un **protocolo de entrega de token múltiples**. El token circula por el anillo detrás del último paquete transmitido desde un dispositivo. Si una estación desea enviar datos, capturará al token, coloca su paquete o paquetes en el anillo y vuelve a depositar el token justo a continuación de la corriente de datos. Sin embargo, cuando una estación captura el token, el anillo permanece inactivo durante un breve período de tiempo, mientras se prepara el paquete. Esto proporciona a la estación y a la unidad de interfaz con el anillo algo más de tiempo para estructurar el paquete y hacerlo pasar por la interfaz.

1.2.11. SISTEMAS OPERATIVOS DEL SERVIDOR

Los sistemas operativos más importantes para servidores que han acaparado el mercado son: UNIX y Windows NT.

UNIX

Una piedra angular en el diseño y desarrollo de los sistemas operativos (SO) sin duda lo constituye UNIX, SO creado por K. Thompson y D. M. Ritchie de los Laboratorios Bell de AT&T a principios de los años 70s.

Los objetivos iniciales de UNIX eran bastante modestos: un sistema operativo para programadores hecho por programadores; sin embargo, y a pesar de que inicialmente corría en minicomputadoras, desde un principio ofrecieron características propias de los SO de las computadoras más grandes de esa época, tales como: multiusuario y multiprocesamiento.

Evidentemente, las características actuales de UNIX han evolucionado muchísimo con respecto a las primeras versiones, entre las cuales se encuentran: proporcionar un "lock" a los archivos o porciones de ellos; proveer una comunicación ágil, poderosa y elegante entre los procesos; utilizar archivos localizados en máquinas remotas (NFS); proporcionar una interfaz gráfica (X Windows); compartir tanto el espacio de direcciones como los recursos de un proceso UNIX entre varias actividades de control, y agregar algoritmos apropiados para procesos en tiempo real.

UNIX es un sistema operativo que destaca principalmente por su manejo de redes multiusuario. Ha sido aceptado por ambientes académicos, científicos y corporativos como VAX de Digital, IS/1, XENIX y SUN, entre muchos otros.

Unix ha demostrado escalabilidad confiabilidad y rendimiento. Escalabilidad lineal de hasta 64 procesadores en un sistema, tiene bases de datos de más de 5 terabytes y permite miles de usuarios conectados. Por ejemplos los sistemas Unix de Sun permanecen en línea por meses, manejando transacción tras transacción. Esto define retos de misión crítica de negocios y computación en Internet.

Aunque esta plataforma operativa se mantuvo como líder por muchos años, en la actualidad se enfrenta cara a cara contra el Gigante del Software: Microsoft, ante la nueva tecnología del sistema operativo conocido en 1993 como Windows New Technology (Windows NT)

WINDOWS NT

Microsoft, desde el posicionamiento de sus versiones 3.x de interfase gráfica Windows, pasando por el ahora exitoso sistema operativo Windows 95, sin olvidar el legendario MS-DOS, hasta todos sus productos que integran a lo que denominan el BackOffice, ha causado el temor de cualquier desarrollador de sistemas operativos.

UNIX y Windows NT, son sorprendentemente similares en diseño y capacidades, pero sus diferencias son significativas. Los servidores basados en Intel están dirigiendo el crecimiento de Windows NT de acuerdo a distribuidores y usuarios. Aunque NT corre en tres diferentes arquitecturas (el Mips Rx100, el Digital Alpha, y el IBM/Motorola PowerPc), la versión que corre en sistemas basados en x86 es la más popular. NT facilita la integración de servidores y PCs de escritorio con las más populares suites Office y BackOffice de Microsoft. Además los costos del hardware y software de los sistemas basados en NT son más bajos.

Un número cada vez mayor de administradores de sistemas está eligiendo NT. Un factor es la fácil integración de Windows NT con el hardware y software existente. Las instalaciones que ya tiene PCs Windows en sus escritorios, se inclinan hacia NT cuando tienen que elegir opciones para servidor. Otro es que Windows NT es rápido de instalar aprender, usar y mantener.

Windows NT proporciona poderosos servicios de archivos e impresión, además de robustos y confiables servicios de aplicación ya que soporta 32 procesadores. Con Windows NT se pueden usar una gran cantidad de herramientas RAD (Rapid Application Development -- Desarrollo rápido de Aplicaciones), tales como, Visual Basic de Microsoft, Power Builder de PowerSoft, y Delphi de Borland. Unix no soporta tanta variedad de opciones de herramientas RAD populares. NT corre código de fuente idéntico a través de tipos de CPU, es decir, las aplicaciones de una máquina Intel, se pueden correr en una máquina RISC.

El estándar para archivos Macintosh y el soporte de archivos NT es un beneficio para realizaciones de plataformas mezcladas y es más fácil encontrar controladores de dispositivos NT para hardware nuevo. Windows NT tiene excelentes características estándar de seguridad y la lista de control de accesos no sólo se aplican a los archivos sino a todos los objetos manejados por el sistema operativo

Sin embargo, la ascensión de NT no significa que UNIX esté tambaleándose. Otras fuentes hacen notar que aún existe un hueco en cuanto a desempeño entre el rango medio y el nivel alto. Las limitaciones arquitectónicas limitan a los servidores de bajo precio basados en x86, cuando intentan enfrentar tareas realmente grandes.

Cuando UNIX se une con software de base de datos de ligas mayores (como Oracle, Sysbase, o Informix), Windows NT y SQL Server se enfrentan a una competencia formidable.

Para elegir un sistema operativo en el servidor, los expertos deben tener el suficiente conocimiento y mente abierta como adoptar un sistema operativo que solucione el problema de conectividad e interoperabilidad en una organización.

1.3. REDES DE AREA AMPLIA

1.3.1. REDES X.25

Las redes de telefonía pública han satisfecho en forma parcial la demanda de servicios para la comunicación de datos, ya que su alcance mundial las hacen atractivas, sin embargo, las altas tarifas en comunicación de larga distancia y la falta de flexibilidad en sus conexiones obligaron a pensar en una creación de redes dedicadas exclusivamente a ofrecer servicios de transporte de datos.

La técnica empleada por las redes X.25 es la conmutación de paquetes, en la cual los paquetes transferidos son separados en bloques de longitud variable formados por datos y por información de control. Los paquetes así integrados son transmitidos por el origen y retransmitidos por los equipos ruteadores hasta alcanzar su destino. La transmisión de los paquetes es supervisada para controlar el flujo de información de los enlaces, así como para detectar y corregir los errores que pudieran ocurrir durante la transmisión.

Las redes X.25 no sólo dan servicio a una sola corporación pueden interconectarse con otras redes dentro de un sólo país o internacionalmente. *Las redes X.25 utilizan el principio de almacenamiento y envío (Store and Forward)* que tiene como unidad de transferencia al paquete, con el tamaño normalmente de 128 bytes. El usuario fuente es el responsable de realizar la segmentación de sus datos antes de entregarlos a la red. La red se encarga de enviar los paquetes al usuario destino, el cual reensambla los datos a partir de los paquetes que recibe.

Debido a que se establece un *circuito virtual (CV)* ^[9], cada ruteador dentro de la red conoce quién debe entregarle cada paquete de datos. Así mediante saltos, se lleva cada paquete desde su origen hasta su destino Existen paquetes de control que sirven para supervisar la buena operación de la comunicación. Las redes de paquetes permiten la utilización simultánea de una infraestructura común entre diferentes equipos. ^[10]

X.25 define una interfaz hacia las redes por conmutación de paquetes de área amplia. Cada fabricante resuelve de manera particular los aspectos de ruteo y control de fallas y congestión. *Las redes X.25 son diseñadas para conectar equipos terminales remotos a minicomputadoras y mainframes*, empleando enlaces telefónicos conmutados y directos para acceder a la red. El protocolo X.25 realiza gran cantidad de procesamiento, lo que ocasiona retrasos limitando la velocidad de transferencia disponible en el servicio básico de entrega de paquetes. Esta desventaja no permite que X.25 sea considerado para interconectar eficazmente redes locales y se opte por conexiones punto a punto.

Para satisfacer los criterios de alta velocidad de transmisión y bajo costo se considera la opción del *Frame Relay* (Relevo en Tramas) esencialmente es una variación de X.25 a velocidad mucho mayor y con procesamiento en los nodos, enfocado a obtener altas

[9] Un *circuito virtual* es una conexión lógica sobre un medio físico.

[10] Soto Sumuano, Leonardo y Sánchez Sandoval, Alfonso. - Red X.25. - Revista Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios. - Xview, S.A. de C.V. - No. 7. - Enero-Febrero de 1994. - México. - pp. 16 a 19.

velocidades de transmisión, minimizando las funciones de corrección de error y control de flujo.

1.3.2. FRAME RELAY

Forma parte de la familia de las tecnologías de “Fast Packet Switching” (Conmutación rápida de paquetes), donde a diferencia de las tecnologías de conmutación de paquetes tradicionales (X.25), *permiten el manejo de un mayor número de circuitos virtuales* por puerto físico. (un circuito virtual es una conexión lógica sobre un medio físico).

X.25 maneja la posibilidad de tener hasta 128 circuitos virtuales en un puerto físico de un equipo de datos, lo que significa que hasta 128 sesiones se pueden tener en forma simultánea accediendo a un equipo por el mismo puerto, mientras que Frame Relay nos ofrece el poder tener hasta 975 circuitos virtuales por conexión física (puerto).

Aunque X.25 es un protocolo muy robusto y complejo, desarrollado para operar sobre redes telefónicas analógicas, donde las inducciones (ruido) y atenuaciones, son muy comunes, por lo que se requiere un mecanismo de corrección de errores. *Frame Relay* opera en forma nativa sobre circuitos digitales, donde no hay errores por inducciones o atenuaciones, evitando el manejo de “acknowledgment” o confirmaciones en cada nodo (como lo hace X.25), por tal motivo es más eficiente, ya que *deja a protocolos como TCP la detección y retransmisión por causa de errores.*

1.3.3. ATM

El *modo de transferencia asíncrona* (Asynchronous Transfer Mode), es uno de los mayores logros que la industria de la conectividad ha dado a los usuarios, debido a los beneficios que ofrece en la conectividad local remota haciendo **uso de sistemas de red terrestres y satelitales.**

Es una tecnología de transferencia de datos conmutados a gran velocidad. Mientras redes como Token Ring y Ethernet, transmiten datos a velocidades que oscilan entre 10 y 16Mbps, ATM lo hace a *velocidades que van desde los 155 hasta los 622 Mbps* e incluso se espera que lo haga a 10 Gbps.

El beneficio principal de *ATM* y todas las topologías que utilizan fibra óptica no es su velocidad, sino la gran cantidad de servicios que se pueden acceder contando con él por ancho de banda que manejan y *que hace posible la integración de multimedia a sus comunicaciones.* Algunos ejemplos son: videoconferencias de escritorio a escritorio, animación, envío de mensajes, trabajo cooperativo y acceso a computadoras de gran capacidad. En otras palabras, ATM le permitirá integrar voz, vídeo y datos en un mismo sistema de comunicación que conecta sus estaciones de trabajo en **topología de estrella con cable de par torcido o fibra óptica.**

ATM es una opción cara, pero sus servicios lo justifican sobretodo en aquellos lugares donde el flujo de información es muy grande y las necesidades de resolver problemas en cuestión de segundos son la prioridad, como en el caso del sector financiero.

Al parecer, la velocidad se vuelve un factor cada vez más importante en la comunicación de datos y las tendencias en telecomunicación se inclinan seriamente al uso de fibra óptica. Se busca la comunicación compartida con velocidad en los enlaces que permitan contar con estaciones de trabajo más poderosas y computadoras personales con un número mayor de aplicaciones gráficas.

1.4. EJEMPLO: INFRAESTRUCTURA DE RED DE COMPUTO

La Gerencia de Inspección y Mantenimiento (GIM) de PEMEX cuenta con la siguiente infraestructura de red de computo.

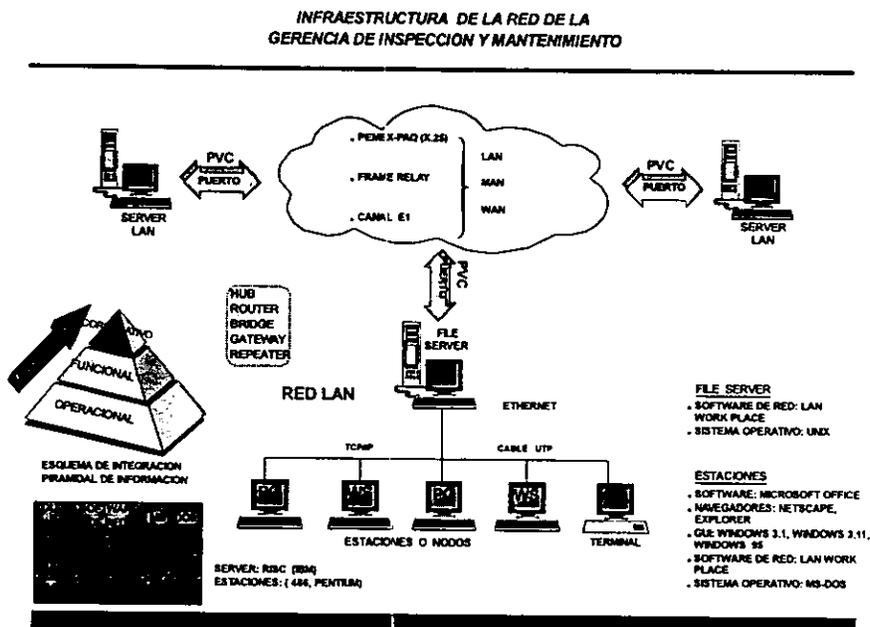


Figura (1.11) Red de Cómputo de la GIM

Es una Red de Area Local (LAN) con topología Ethernet, compuesta en la parte del servidor de archivos de un RISC de IBM, con sistema operativo UNIX y en la parte de los nodos intervienen computadoras personales, estaciones de trabajo y terminales, aceptando cualquier marca, excepto Macintosh, con procesadores 486 y Pentium, y en memoria RAM 8 MB mínimo.

El protocolo por el cual se comunican todos los equipos es por medio de TCP/IP, a través del cable UTP. En general el cableado esta diseñado de forma estructurada, existiendo un concentrador conocido como HUB. El servidor tiene salida a la red PEMEX-PAQ a través del protocolo X.25, o por medio de Canal E1, creándose la comunicación con otras redes locales.

Con respecto al software, las estaciones de trabajo manejan como sistema operativo a MS-DOS o Windows 95 y como Interfaz Gráfica de Usuario a Windows en sus diferentes versiones, además de diverso software para trabajos de oficina. El software de red para estos equipos (nodos y servidor) es LAN WORK PLACE. En el servidor su software es compartido, principalmente almacenan herramientas como gráficas, procesadores de palabras, manejadores de imágenes y otros, que requieren de considerable espacio en disco duro y recursos de mayor rendimiento, razón principal para que estén en el servidor. El servidor también actúa como administrador de impresión.

La mayoría del equipo se conecta a un UPS (Uninterruptable Power Supply – Suministro de Energía Ininterrumpible) para garantizar la continuidad de la corriente eléctrica.

ASPECTOS GENERALES DE BASE DE DATOS

1.5. BASES DE DATOS

1.5.1. ANTECEDENTES

Desde siempre, el hombre se ha visto en la necesidad de crear abstracciones con la finalidad de representar algunos objetos del mundo real, estas abstracciones son conocidas como **DATOS**. Además de generar datos, el hombre también a buscado la manera de almacenarlos de forma ordenada con el fin de facilitar su modificación y posterior consulta.

En los últimos años, se ha incrementado la generación y demanda de datos en todos los niveles de una empresa. Estos requerimientos abarcan desde las agendas personales, hasta las nominas, pasando por estados de cuenta, catálogos de productos, listas de precios, inventarios, control de asistencia, etc.. Por lo que podemos decir que todo mundo requiere manejar "datos ". Dichos requerimientos de manejo de información pueden ser cubiertos de diferentes maneras, dependiendo de su volumen y complejidad.

Al principio la información se registraba en papel y se almacenaba en archiveros o gavetas. Posteriormente, cuando los volúmenes de información crecieron, se buscaron medios más adecuados de almacenamiento y de esta manera surgieron las microfichas o microfilms, resolviéndose el problema de espacio, pero se complicaba su registro y modificación. Con el creciente uso de las computadoras personales dentro de las empresas, surgieron también paquetes o programas que permitieron almacenar y recuperar grandes volúmenes de información a bajo costo:

Evolución de las Base de Datos

Antes de que aparecieran las primeras computadoras de la tercera generación, la mayoría de los archivos se organizaban de modo secuencial. El programador de aplicaciones diseñaba la distribución física de los datos y la incorporaba a los programas de aplicación. De ahí que los datos difícilmente se comparten entre las distintas aplicaciones.

A mediados y finales de la década de los setentas, se reconoció la naturaleza cambiante de los archivos y de los dispositivos de almacenamiento. Los archivos estaban por lo general, diseñados para una aplicación determinada o para un grupo de aplicaciones muy similares. Aunque el software provee "métodos de acceso" (secuencial, directo) a los registros no tiene administración de datos.

En la década de los ochentas y principios de los noventas, de los mismos datos físicos se derivan múltiples bases de datos lógicas. Se puede tener acceso a los mismos datos de diferentes maneras, según los requisitos de la aplicación. El software provee los medios para disminuir la redundancia. Se utilizan formas de organización de datos muy complejos sin que ello se refleje en los programas de aplicación.

En la etapa actual, el software procura la independencia lógica y física de los datos. Se facilita la administración e integración de las aplicaciones. Y los datos pueden evolucionar sin que se incurra en costos de mantenimiento excesivos.

1.5.2. DEFINICIÓN

Una base de datos es el conjunto de datos relacionados entre si, almacenados, estructurados, no redundantes y de fácil acceso para múltiples aplicaciones.

Una *base de datos* es una colección de datos interrelacionados almacenados en conjunto, sin redundancias perjudiciales o innecesarias. Su finalidad es la de servir a una o más aplicaciones; los datos son independientes de los programas que los usan. Se emplean métodos bien determinados para incluir datos nuevos y modificar o extraer los datos almacenados.

Los registros para distintas entidades se almacenan comúnmente en una base de datos (mientras que los archivos almacenan registros para una única entidad). Por ejemplo, en una base de datos de una universidad, se interrelacionan las entidades de estudiantes, cursos y profesores.

Un *sistema de base de datos* es un sistema de información el cual hace uso de ciertos modelos, metodologías y técnicas que nos ayudan a estructurar los datos; además aplica ciertas herramientas que nos permiten manejar estas estructuras de datos. Con esto se pretende eliminar los errores de los sistemas tradicionales.

1.5.3. CARACTERÍSTICAS

- Conjunto de datos interrelacionados.
- Almacenados en conjunto.
- No redundantes.
- De fácil acceso.
- Compatibles entre aplicaciones.
- Independientes de los aplicaciones que los usen.

1.5.4. VENTAJAS

El enfoque de Base de Datos provee una solución poderosa y simple al procesamiento de información, evitando muchos de los problemas que representa el que cada aplicación tenga sus propias estructuras de datos.

El enfoque de Bases de Datos nos permite:

- ✓ Controlar la redundancia.
- ✓ Mantener la consistencia.
- ✓ Lograr la integración de los datos.
- ✓ Compartir datos entre las diferentes aplicaciones.
- ✓ Cumplir con estándares.
- ✓ Facilitar el desarrollo de aplicaciones.
- ✓ Uniformar los controles de seguridad, primacía y seguridad.
- ✓ Independencia entre los datos y los programas.
- ✓ Reducir el mantenimiento a los programas.

1.6. SISTEMA MANEJADOR DE BASE DE DATOS (DBMS)

1.6.1. DEFINICIÓN

Un *DBMS* (Data Base Management System - Sistema Manejador de Base de Datos) consiste en un conjunto de datos relacionados entre si y un grupo de programas para tener acceso a esos datos. El conjunto de datos se conoce comúnmente como *base de datos*. Esta contiene información acerca de una organización determinada. El objetivo primordial de un *DBMS* es crear un ambiente en que sea posible guardar y recuperar información de la base de datos en forma conveniente y eficiente.

Los sistemas manejadores de base de datos fueron creadas para aventajar el mecanismo de almacenamiento de datos compartidos.

La esencial tarea de un *DBMS* es mejorar la definición de datos para el sistema, almacenando datos en dispositivos físicos y permitiendo que usuarios accesen y cambien los datos guardados. Un *DBMS* también debe mejorar la seguridad sobre los privilegios de acceso a la base de datos, la recuperación y respaldo de la información, la concurrencia para que múltiples usuarios puedan acceder la base de datos al mismo tiempo y checar la integridad de los datos para que permanezca la consistencia lógica. Implementar estos principios de los *DBMS* dentro de un sistema C/S, asegura su éxito en la funcionalidad y operación.

El manejador actúa como un intermediario del ambiente de proceso de datos, ya que sirve de interfaz entre los archivos y la gente que busca información de ellos.

1.6.2. OBJETIVOS

- *Minimizar la redundancia de los datos:* No tener datos repetidos. No almacenar datos derivados.
- *Garantizar la consistencia de los datos:* Obtener la misma información por peticiones similares en un momento dado.
- *Integridad de los datos:* Las reglas dictadas por política o normas de la empresa y que los datos deben cumplir (Reglas de Negocio).
- *Seguridad de los datos:* La protección de los datos contra accesos, modificaciones o pérdidas, ya sea en forma intencional o no intencional.
- *Controlar la concurrencia:* Múltiples usuarios pueden acceder a la misma información al mismo tiempo, sin que con ello se tengan problemas con los datos.
- *Proteger los datos contra fallas del sistema:* Es la capacidad de restaurar la integridad y consistencia después de una falla del sistema.
- *Diccionario de datos:* Es la capacidad que da el DBMS de poder tener la descripción de los datos que están almacenados en la base de datos. Un diccionario de datos es un conjunto de tablas que sirven de referencia en la base de datos para saber las características de los usuarios y los objetos.
- *Interfaz de alto nivel con los programadores:* Se refiere al manejo de un lenguaje de cuarta generación, como lo es SQL (Structured Query Language).
- *Características adicionales:*
 - * Independencia de los programas respecto a los cambios en la estructura de los datos.
 - * Programas de utilería para la administración de la base de datos.
 - * Mecanismos de seguridad para imponer límites de acceso.
 - * Facilidades para afinación (tuning) de la base de datos.

1.6.3. MODELO DE BASE DE DATOS

Existen enfoques alternativos para visualizar y manejar datos a un nivel lógico independientemente de cualquier estructura física de soporte en que se basen. El conocimiento de qué modelo utilizará un DBMS determinará cómo debe estructurarse un diseño y las formas para representar las relaciones entre los datos.

El modelo expresa las entidades y sus relaciones y es usado para representar los datos. El modelo es una herramienta de comunicación entre los diversos usuarios de los datos, y como tal se desarrolla sin tomar en cuenta la representación física de los mismos. El modelo de datos se utiliza para organizar, visualizar, planear y comunicar ideas.

Estos modelos han evolucionado junto con los medios y métodos de almacenamiento.

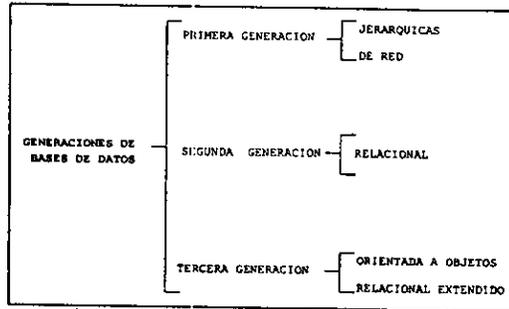


Figura (1.11) Modelos de Base de Datos

Los modelos de bases de datos que existen son:

- Modelo Jerárquico
- Modelo de Red
- Modelo Relacional
- Modelo Orientado a Objetos

MODELO JERARQUICO

La estructura lógica en la cual se sustenta la base de datos jerárquica es el *árbol*. Un árbol se compone de un nodo *raíz* y varios nodos *sucesores*, ordenados jerárquicamente. Cada nodo representa una *entidad* (tipo de registro) y las *relaciones* entre entidades son conexiones entre los nodos. El nodo colocado en la parte superior es llamado *padre* y los nodos inferiores son los *hijos*, creándose la relación *Padre/Hijo* (Figura 1.12).

Un modelo jerárquico de datos permite dos tipos de relación:

- *Uno a Uno*: Una entidad en un nivel se relaciona con una entidad en el siguiente nivel.
- *Uno a muchos*: Una entidad en un nivel se relaciona con una, muchas o ninguna entidad del siguiente nivel. Entre un padre y varios hijos, en otras palabras, cada hijo tiene un padre.

A partir de estas relaciones se generan las siguientes condiciones:

- *Regla de Inserción*: Un registro dependiente no se puede añadir a la base de datos sin un padre.
- *Regla de Borrado*: Al borrar un padre de la base de datos también se borran todos sus descendientes.
- *Regla de Modificación*: No se puede modificar la llave primaria del padre, porque de ahí depende varios hijos. Por lo que es necesario hacer una copia hijos.

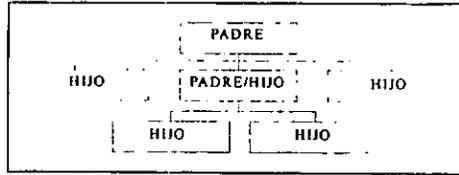


Figura (1.12) Modelo Jerárquico

Las desventajas en el enfoque jerárquico son:

- No modela sencillamente las relaciones muchos a muchos.
- Como resultado del estricto ordenamiento jerárquico, las operaciones conocidas como de inserción y de supresión se vuelven complejas.
- La raíz es el tipo de nodo dominante. Cualquier hijo es accesible solamente a través de su nodo padre.

MODELO RED

Es análogo al modelo jerárquico, excepto que una entidad puede tener más de un padre. Los datos se representan como registros ligados formando un conjunto de datos interceptados (Figura 1.13).

Esta capacidad introduce el uso de un tipo adicional de relación entre los datos:

- *Muchos a muchos*: Una entidad se puede relacionar con una, muchas o ninguna entidad en otro nivel. Un hijo puede tener varios padres y varios hijos a la vez.

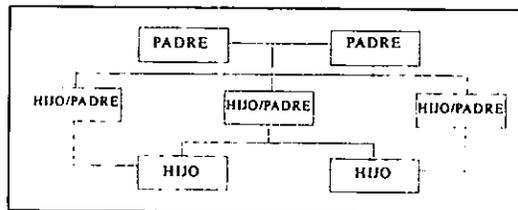


Figura (1.13) Modelo de Red

El modelo cumple con las reglas de inserción y borrado descritas en el modelo jerárquico.

Desventajas en el enfoque de red:

- Resulta difícil definir nuevas relaciones.
- Es complejo darle mantenimiento ya que cualquier cambio en la estructura requiere una descarga de los datos.
- Representa desperdicio de recursos.

MODELO RELACIONAL

El modelo relacional de datos, desarrollado en 1970 por E.F. Codd, se basa en una *relación*: una tabla bidimensional. Los renglones de la tabla representan los *registros* y las columnas muestran los *atributos de la entidad* o también llamados *campos* (Figura 1.14).

Las *relaciones entre tablas* se llevan a cabo a través de un conjunto de columnas que se tengan en común, logrando una conexión dinámica entre un número ilimitado de ellas a través del contenido de las columnas.

El orden de los datos en la tabla no es significativo y tampoco implica un orden cuando los registros están incluidos en la relación. Análogamente, los detalles físicos de almacenamiento (ya sea una organización secuencial o indexada) no son de interés para el analista. Las tablas relacionales muestran las relaciones lógicas, no físicas.

El *modelo relacional* está construido sobre fundamentos relacionales: teoría de conjuntos, álgebra relacional y cálculo relacional, y gracias a esto tiene una ventaja significativa sobre sus competidores: El *formalismo relacional* proporciona un marco de referencia conceptual, en donde los problemas adoptan una formulación más comprensible y a menudo se vuelven más fáciles de manejar.

Las características de este enfoque son:

- Representación de datos a través de tablas.
- Flexibilidad en el mantenimiento de las estructuras.
- Flexibilidad en el mantenimiento de los datos.
- Flexibilidad en el tipo de consultas.

Col 1	Col 2	
		Renglon 1
		Renglon 2
		Renglon 3
		Renglon 4

Figura (1.14) Modelo Relacional

Sus ventajas son las siguientes:

- ✓ Fácil de usar.
- ✓ Fácil de obtener respuestas.
- ✓ Fácil de insertar y actualizar datos.
- ✓ Fácil de cambiar la estructura de los datos.
- ✓ La navegación es responsabilidad del DBMS.

MODELO ORIENTADO A OBJETOS

La creciente demanda del uso de la computadoras para el almacenamiento masivo de información ha incursionado en áreas que directamente no caen dentro del ámbito del procesamiento de datos. Estas áreas de aplicación incluyen el *Diseño Asistido por Computadora* (CAD Computer Aided Design), *Ingeniería de Software Asistido por Computadora* (CASE Computer Aided Software Engineering), *Multimedia*, *Sistemas de información de Oficina* (OIS Office Information System), entre otros.

Estas aplicaciones demandan que la diversa información que generan sea almacenada en la computadora, la diferencia radica principalmente en el hecho de que no hacen uso directo de elementos de información convencionales (cadenas de caracteres, números o fechas). La demanda de almacenamiento se enfoca hacia datos más complejos denominados objetos.

Estos objetos complejos incluyen:

CAD: Componentes de diseño de datos, relaciones entre componentes y versiones anteriores de los diseños.

CASE: Código fuente, dependencia entre módulos de software, definiciones y los históricos del desarrollo del sistema de software.

MULTIMEDIA: Datos de audio, video y similares, éstos datos derivan de sistemas de correo de voz, geofísicos y aplicaciones gráficas.

OIS: Creación y recuperación de documentos, software para mantener calendarios y citas.

1.6.4. LENGUAJES

El DBMS cuenta con tres lenguajes importantes:

- *Lenguajes de Definición de Datos* (DDL - Data Definition Lenguaje): Se encuentra tanto en la base de datos como en las aplicaciones.
- *Lenguaje Manipulador de Datos* (DML - Data Manage Lenguaje): Este lenguaje nos da las instrucciones mínimas para poder acceder los datos, entre ellas se encuentran aquellas que nos permiten:
 - Abrir y cerrar la Base de Datos.
 - Consultar y actualizar datos.
 - Iniciar y terminar transacciones.
 - Controlar concurrencia, etc.
- *Lenguajes de Anfitrión:* Es el que proporciona instrucciones adicionales a un programa de aplicación, por ejemplo, instrucciones de control de secuencia (ciclos y bifurcaciones), instrucciones de entrada y/o salida a dispositivos físicos (como terminales, lectoras ópticas, etc.), funciones matemáticas, etc.. Este lenguaje anfitrión puede ser un lenguaje algorítmico como Pascal, Cobol, C, etc. o bien un lenguaje de 4a generación. Así, los programas de aplicación de un sistema de base de datos contienen dos tipos de código: código del lenguaje anfitrión y código del DML.

1.7. BASE DE DATOS DISTRIBUIDAS

1.7.1. EVOLUCIÓN

PROCESO CENTRALIZADO

- La aplicación reside en un nodo remoto.
- El usuario utiliza una terminal con emulación y un módem para conexión.
- Es común utilizar redes públicas.

PROCESO DISTRIBUIDO

- La aplicación reside en un nodo diferente al de la base de datos y a su manejador.
- El manejador de base de datos es requerido sólo donde la base de datos reside.
- La aplicación conoce la localidad de los datos.
- Instrucciones SQL accesan datos en una localidad a la vez.

PROCESO COOPERATIVO (BASES DE DATOS DISTRIBUIDAS)

- Se presenta frecuentemente el caso de que en las organizaciones, los usuarios, la fuente de información y los recursos, en cuanto a equipo se refiere, se encuentran geográficamente en diferentes lugares físicos.
- En este tipo de bases de datos, cada máquina de la red posee capacidad de procesamiento autónomo y puede efectuar aplicaciones locales. Igualmente puede participar en la ejecución de cuando menos una aplicación global, que requiere accesar datos de varias máquinas por medio de un subsistema de comunicaciones.

1.7.2. DEFINICIÓN

Una *Base de Datos de Distribuida* es una colección de sitios de almacenamiento con procesamiento autónomo, tal vez a distancia uno de otro y conectados a través de un *sistema de red*. En estas bases de datos los servidores hacen el papel de *administradores* de la información e interactúan entre sí para proporcionar al usuario la imagen de una sola base de datos.

Una base de datos distribuida es una red de *Bases de Datos Locales* almacenadas en múltiples máquinas pero vistas y manejadas por el usuario como una sola *Base de Datos Lógica*.

1.7.3. CARACTERÍSTICAS

- Permite ver múltiples Bases de Datos Físicas como una sola Base de Datos Lógica.
- El manejador de base de datos se encuentra en cada lugar donde hay una Base de Datos Física.
- Cada manejador de base de datos sabe la localidad de los datos.

1.7.4. PROCESAMIENTO COOPERATIVO

El procesamiento cooperativo es subconjunto de las bases de datos distribuidas, o sea para que exista procesamiento cooperativo debe haber un ambiente de base de datos distribuidas.

El *Proceso Cooperativo* permite que porciones de una aplicación pueden correr en diferentes computadoras, y por tanto los datos pueden estar ubicados en diferentes localidades; de esta forma varias computadoras comparten el procesamiento del programa.

Los esquemas de negocios actuales requieren información totalmente actualizada y con mayor rapidez posible, lo que hace de vital importancia que la información que se encuentra dispersa en varias plataformas pueda unirse en algún momento del proceso.

Dentro del procesamiento cooperativo, sin importar qué plataformas se utilicen, el usuario debe ver la información de la misma manera que si fuera una sola base de datos. Existen varios manejadores de base de datos que pueden radicar en distintos servidores de diferentes plataformas y que le dará la funcionalidad antes mencionada al usuario. Entre los más populares están: **ORACLE, SQLBase, DB2, Sybase, Informix, Ingres, Progress y SQL-Server.**

1.7.5. REPLICACIÓN

La *replicación* es una capacidad que los manejadores de base de datos poseen para cambiar o actualizar los datos desde una base de datos padre a todos sus hijos, en un ambiente de base de datos distribuidas (servidor a servidor).

La replicación toma lugar basada sobre el tiempo de frecuencia que se determine. No obstante, no es conveniente enviar grandes volúmenes de datos a otra base de datos remota. Deberán considerarse varias opciones de réplica de los datos para reducir la cantidad de datos que se envían por la red.

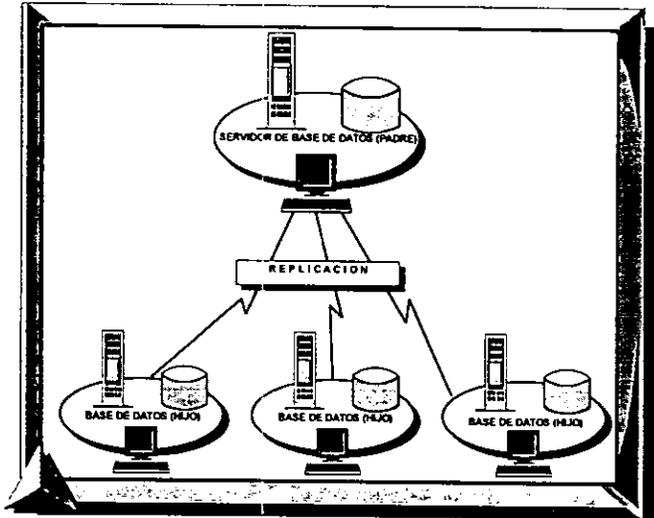


Figura (1.15) Replicación

1.8. SISTEMA MANEJADOR DE BASE DE DATOS RELACIONALES (RDBMS)

1.8.1. ORIGEN

Los RDBMS (Relational Database Management Systems - Sistemas Manejadores de Base de Datos Relacionales) tuvo su origen en la primera teoría avanzada por E.F. Codd en IBM en 1969. El *modelo relacional de Codd* estableció un sistema formal para almacenar información que separa a los datos desde su representación lógica y su acceso. A finales de los ochenta los RDBMS se habían adelantado sobre esas viejas tecnologías que empezaron a tocar el tema sobre integración en el emergente paradigma de C/S. En la actualidad, la enorme mayoría de los sistemas C/S están enfocados a transacciones de información con bases de datos casi siempre relacionales.^[11]

Los RDBMS han sido parte del ambiente de C/S, frecuentemente la gente piensa en base de datos relacionales cuando escuchan el termino servidor. Hoy en día los RDBMS representan uno de los más grandes y rápidos mercados dentro de la industria del software. Oracle es líder indiscutible del mercado de RDBMS sostiene el 56%, otras bases de datos relacionales importantes son: Sybase, Informix, Sql Server de Microsoft y DB2 de IBM

1.8.2. SQL

SQL es el único lenguaje de programación estándar NCITS/ISO (Nacional Committee for Information Technology Standards / International Standard Organization- Comité

[11] Jenkins, Neil y otros.- *Client/Server Unleashed*. - 1ª Edición.- Editorial Sams.- E.U.A.- 1996.

Nacional para Estándares de Tecnologías de Información / Organización Internacional de Estándares) para bases de datos relacionales.

El *Lenguaje Estructurado de Consulta* (SQL- Structured Query Language) es el idioma de todos los RDBMS para formular preguntas y recibir información de la base de datos. En algunos casos se utiliza como lenguaje principal y en otros como lenguaje alternativo o para comunicación con otros RDBMS.

SQL contiene tanto instrucciones de DDL (Lenguaje de definición de datos) como de DML (Lenguaje manipulador de datos) y se puede utilizar a nivel intérprete o bien inmerso en un lenguaje anfitrión. Las operaciones de SQL funcionan casi siempre en términos de conjuntos y no de registros individuales, y tampoco incluyen ninguna referencia a rutas de acceso explícitas, lo que lo hace bastante amigable para los usuarios.

1.8.3. CARACTERÍSTICAS

La arquitectura de los RDBMS tienen numerosas características que de otros modelos de almacenamiento de datos. Las más importantes son:

- *Separación del punto de vista físico (manipulación de los datos) y lógico (acceso a datos) de los datos.* Dentro de una base de datos relacional, todos los datos están lógicamente almacenados en tablas, formadas por *renglones y columnas*. Las tablas se relacionan entre ellas mediante las columnas que tiene en común. Puede utilizarse la base de datos para imponer estas relaciones por medio de la *integridad referencial* (referencial integrity), la cual se establece al nivel de la base de datos por medio de las *restricciones* (constraints) de la tabla.
- *Fácil manejo de los datos.* Los usuarios no necesitan aprender tareas de bajo nivel, definir estructuras de datos en memoria o escribir código sobre el disco para acceder datos. Los usuarios sólo necesitan aprender un poco de los sencillos comandos de SQL y permitir la manipulación y acceso de datos a las rutinas del RDBMS.
- *Adecuada administración de los datos.* Es otro atributo para definir a un RDBMS porque presenta el catalogo del sistema, el cual contiene información acerca de las tablas, índices, asignación de disco físico, usuarios, permisos, tipos de datos, etc., es decir, proporciona información sobre el mismo RDBMS. Cuando los objetos de la base de datos son creados, cambiados o borrados, el catalogo del sistema es modificado automáticamente.
- *Manejo de Índices.* Afortunadamente los RDBMS proporcionan un mecanismo por índices que mejora la búsqueda para localizar un valor en particular. Un *índice* es una estructura de datos separado de la tabla donde se almacenan valores ordenados de una columna dada y su dirección física. Un índice puede reducir significativamente las entradas y salidas (E/Ss) requeridas para encontrar un renglón del dato. En vez de leer los datos directamente sobre el disco, el RDBMS primero ubica el valor dentro de un índice, encuentra la localización física y entonces va directamente al registro sobre el disco.

- **Notación de Restricciones de Integridad.** Las tablas son relacionadas por llaves. Una llave es alguna columna o combinación de columnas que identifican los registros de cada tabla. Una llave única de una tabla puede existir como una columna no única dentro de otra tabla. Por instancia, dentro de una tabla de empleados, un empleado puede identificarse únicamente por su número de seguro social. Este número se repite varias veces en la tabla nómina, donde el registro es insertado cada vez que se emite el pago de nomina. Las *restricciones de integridad* son reglas construidas por el diseñador de la base de datos que automáticamente se invocan para mantener la integridad de la base de datos.
- **Definición de esquemas de seguridad.** Los esquemas de seguridad definidos para un RDBMS son altamente eficientes, proporcionando:
- *Seguridad de cuentas:* Son cuentas para entrar a la base de datos.
 - *Privilegios de base de datos:* Son perfiles de seguridad definidos, para la manipulación de la base de datos.
 - *Seguridad de los objetos:* Es la posibilidad de crear concesiones sobre sus objetos.

Para concluir, el RDBMS debe proporcionar los servicios de definición, mantenimiento, manipulación, desplegado e integridad de datos.

1.8.4. ORACLE

Oracle se fundó en E.U.A. por Larry Ellison y actualmente es la base de datos institucional de PEMEX. Oracle es un producto concebido para organizaciones que tienen mainframes, minicomputadoras y redes por todos lados con Workstations, PCs y MACs.

Una base de datos Oracle es un conjunto de datos relacionados. Oracle proporciona la capacidad de almacenar y acceder a estos datos de forma consecuente con un modelo definido conocido como Modelo Relacional (Relational Model). Por ello, Oracle se conoce como un sistema de gestión de bases de datos relacionales (RDBMS). Para más información sobre éste manejador consultar el punto 4.7.3 Diseño Físico de la Base de Datos en el Capítulo IV.

En un futuro no muy lejano, Oracle será una base de datos totalmente orientada a objetos (OO), ofreciendo apoyo en migrar aplicaciones relacionales a OO.

1.9. SISTEMA MANEJADOR DE BASE DE DATOS POR OBJETOS (ODBMS)

1.9.1. DEFINICIÓN

Los ODBMS (Object Database Management Systems - Sistemas para Administración de Base de Datos de Objetos) son una consecuencia natural de la tendencia orientada a objetos dentro del desarrollo del software. Esta clase de productos de manejo de datos acoplan los lenguajes de programación orientado a objetos (OOP – Objeto-Oriented Programming), permitiendo a los desarrolladores almacenar y acceder estructuras complejas de datos,

conocidas como objetos. Los ODBMS manejan datos como diagramas, imágenes y series de tiempo que no son fácilmente llevados a cabo en un modelo relacional.

La principal habilidad de un objeto es representar las entidades del mundo real y almacenarlas en objetos dentro de la base de datos. Los objetos no sólo contienen datos pueden también almacenar programación lógica que describe el comportamiento de un objeto. La promesa de la orientación a objetos es lograr que los módulos del software sean reusables para automatizar y modernizar el desarrollo de las aplicaciones.

1.9.2. CARACTERÍSTICAS

Las bases de datos fueron diseñadas para que los datos se almacenaran y pudieran ser recuperados. Los RDBMSs resolvieron el problema del manejo de los datos dentro de la forma más básica con estructuras bidimensionales llamadas tablas. Con el advenimiento de la programación orientada a objetos, las estructuras de los datos han ido incrementando su complejidad, organizándose de acuerdo a los principios de orientación a objetos dificultando el traslado dentro del formato relacional. El modelo de objetos permite el modelado complejo de objetos tal y como existen, en lugar de tratar de reducir los objetos a una estructura de dos dimensiones. El creciente interés en aplicaciones multimedia y en Internet han creado nuevos mercados para ODBMS.

La diferencia entre un ODBMS y un RDBMS es la forma en que ambos almacenan los datos y la manera en que estos son accedados. Dentro de un ODBMS los objetos de datos se construyen desde un número de componentes de elementos de datos y sus relaciones se representan usando conceptos orientados a objetos como encapsulación, herencia, polimorfismo, jerarquía de clases, instancias de los objetos y tipos de datos definidos por el usuario. El RDBMS almacena los datos dentro en un nivel más atómico permitiendo al desarrollador agruparlos dentro de objetos significativos usando SQL. Los ODBMS no tienen un lenguaje estándar, por lo que se tiene que trabajar con un extraño lenguaje propio y aprender uno nuevo cada vez.

Actualmente la tecnología de los ODBMSs no ha madurado lo suficiente como el mercado de los RDBMSs, pero la tendencia es enfocar las bases de datos a la orientación de objetos y muchas compañías fuertes en esta área están invirtiendo tiempo y dinero en la construcción de eficientes ODBMSs

ASPECTOS DE LA ARQUITECTURA CLIENTE / SERVIDOR

1.10. ARQUITECTURA CLIENTE/SERVIDOR

1.10.1. ANTECEDENTES

Debido al aumento de procesamiento de información entre empresas ubicadas en diferentes zonas geográficas, fue necesario diseñar sistemas que pudieran acceder a bases

de datos remotas con mayor rapidez y eficacia. Lo anterior propicio el desarrollo de métodos y tecnologías que permitieron llevar a cabo esta función.

A partir de la década de los setenta resultó factible operar ambientes *multiusuario* con el servidor (computadora central) y terminales tontas (Figura 1.16). A este modelo también se le conoce con el nombre de *Procesamiento Maestro/Esclavo*. Las características principales de este ambiente son:

El **servidor** tiene gran capacidad de almacenamiento, alta velocidad, buenos esquemas de seguridad y memoria suficiente. Ejecuta los comandos introducidos en la terminal. Es un equipo costoso en su operación y mantenimiento.

Las **terminales tontas** sin capacidad de procesamiento ni almacenamiento. Es el enlace entre el usuario y el equipo central. Sirven sólo para entrada/salida y visualización de errores, por lo tanto, la mayor parte del trabajo la realiza el servidor.

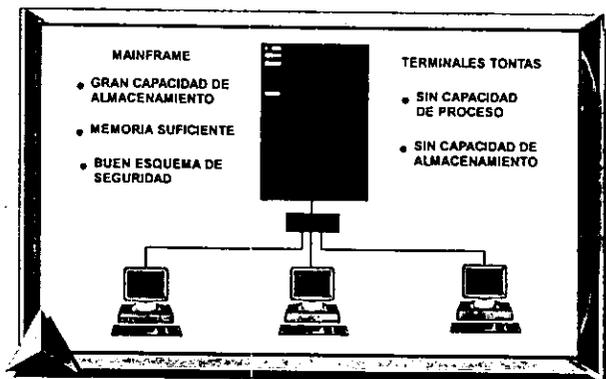


Figura (1.16) Ambiente Multiusuario

En tiempos más recientes surge la *computadora personal* (PC's) con capacidad de procesamiento propio, almacenamiento y memoria pero con menor velocidad que un servidor, propio de un sistema multiusuario (Figura 1.17).

Estas computadoras, por su facilidad de programación, dieron lugar a gran cantidad de paquetes, entre los que sobresalieron Lotus 123 y Dbase, haciendo que el uso de la PC casera se transformara en una herramienta alterna para usuarios independientes dentro de una oficina.

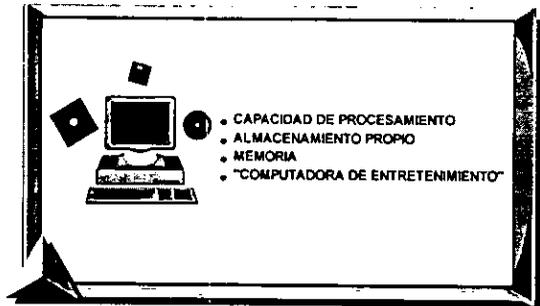


Figura (1.17) Computadora Personal

La llegada de las *redes de computadoras* personales dio un giro importante en la industria de computadoras (Figura 1.18), ya que permitió que los usuarios independientes de PC's pudieran compartir su información y recursos.

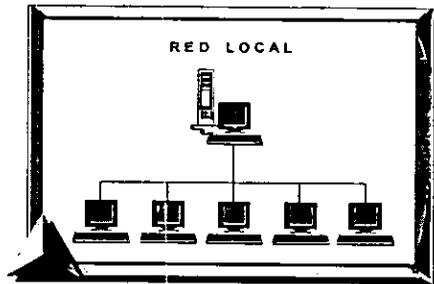


Figura (1.18) Red de Cómputo

A estos eventos, siguió el mejoramiento en velocidad de procesamiento, incremento de memoria, ambiente gráfico y otras capacidades que hicieron de la PC una herramienta indispensable en los grupos de trabajo de cualquier compañía. Claro que los sistemas multiusuarios no se quedaron atrás, y siguieron mejorando en cuando al rendimiento y seguridad.

Sin menospreciar a ninguna de las plataformas, se necesitaba mayor interconexión de los usuarios de PC's con los sistemas multiusuarios, lo cual fue formando el concepto que actualmente conocemos como *Arquitectura Cliente/Servidor*.

1.10.2. MOVIMIENTO A CLIENTE/SERVIDOR

El movimiento a Cliente/Servidor ha marcado un mayor desarrollo en la evolución de la computadora personal. Este desarrollo se origino porque varias compañías habían invertido grandes sumas de dinero en la compra de computadoras para optimizar sus funciones organizacionales. Pero a pesar de que el principal uso de las PCs y las redes locales se dirige a usuarios que normalmente realizan funciones de oficina así como de compartición de archivos y de impresión, no se justificaba el gasto del equipo.

La necesidad de hacer un mejor uso de las PCs condujo a la idea de que las aplicaciones tenían que procesarse localmente y remotamente sobre un servidor. Esto significó que las compañías se fueran alejando de un sistema de computo centralizado y confiaran en un sistema de red de clientes y servidores. La promesa de implantar sistemas cliente/servidor robustos, rápidos y distribuidos significó que muchas organizaciones empezaran a desarrollar aplicaciones en cliente/servidor.

Hoy en día Cliente/Servidor se ha convertido en la más grande tecnología dentro del ambiente de los negocios. Las estrategias organizacionales están cambiando rápidamente y con el desarrollo o modificación de un sistema centralizado no se podía atender rápidamente las necesidades de la organización.

De acuerdo a recientes estudios el 85% de 1000 empresas están usando la tecnología cliente/servidor para dar soluciones a sus necesidades. Algunas ya han implementado sofisticados sistema cliente/servidor obteniendo resultados benéficos. Otros están activamente involucrados en el desarrollo de procesos, entusiastas a los beneficios que el sistema completo pudiera proporcionar. Esta fuerte inversión financiera aplicada a los sistemas cliente/servidor muestra un claro compromiso de las compañías.

Las grandes corporaciones no solo están cambiando a cliente/servidor, medianas y pequeñas compañías también están adoptando esta arquitectura. En el mercado se ha incrementado diariamente la aparición de nuevos productos y herramientas para trabajar en cliente/servidor. Esta competencia ha permitido la reducción de costos así como que diferentes tipos de compañías puedan beneficiarse de esta tecnología. Los sistemas cliente/servidor pueden implantarse para cualquier compañía a pesar de su tamaño.

La tecnología cliente/servidor mejora la dirección estratégica de una compañía y da ventaja competitiva. Esta tecnología capacita a las organizaciones a desarrollar poderosos y flexibles sistemas de todos tipos que resuelvan necesidades específicas. Se pueden desarrollar sistemas cliente/servidor que soporte los alcances de un pequeño departamento con solo tres usuarios, o sistemas a nivel empresarial que soporten miles de usuarios a través de estados o países. Un sistema cliente/servidor puede mejorar las capacidades de acceso remoto, donde la gente que esta trabajando en campo tiene que acceder la información como si estuviera físicamente en la oficina. Además, los sistemas cliente/servidor pueden compartir información con otras aplicaciones así como integrarse con más tecnologías. Las empresas están desarrollando sistemas cliente/servidor para resolver una gran variedad de necesidades organizacionales.

PEMEX, no es una empresa que se encuentre al margen de esta tecnología. Constantemente evalúa los avances informáticos y aplica los que se adecuan a sus necesidades. PEMEX esta formada por varias dependencias, pero desafortunadamente no todas cuentan con el suficiente equipo para construir una arquitectura Cliente/Servidor. La Gerencia de Inspección y Mantenimiento (GIM) dentro de su infraestructura de computo dispone de los elementos necesarios para la operación de sistemas Cliente/Servidor. Por lo que el movimiento a Cliente/Servidor será benéfico, sin ninguna inversión costosa en adquisición de equipo y aprovechando al máximo las cualidades de esta tecnología.

1.10.3. DEFINICIÓN

Una *arquitectura* es un conjunto de modelos, reglas y definiciones que son usados como guías para construir un producto. Un *producto* es la implementación específica de una arquitectura. La arquitectura por sí misma, sin un producto que la implemente, no es muy útil. Por lo tanto un producto que implemente la arquitectura es lo que hace que la arquitectura sea utilizable.

Una arquitectura bien diseñada debe estar basada en varios productos y debe tener un tiempo de vida más allá de la vigencia de cada producto en específico. Para alcanzar estas metas, dicha arquitectura debe basarse en estándares de la industria y también debe incluir la introducción de nuevos estándares propios de la empresa. Los productos que implementen una arquitectura deben promover y soportar la estandarización.

Cliente/Servidor es la arquitectura que las compañías están adoptando para el manejo de su información. Cliente/Servidor (C/S) es una tecnología donde la funcionalidad de un sistema esta siendo llevada a cabo entre el cliente (front end) y el servidor de base de datos (back end), es decir, las aplicaciones del usuario corren en uno o más clientes mientras que el manejo de las bases de datos es controlado por los servidores (Figura 1.19).^[12]

Cliente/Servidor es la interacción de dos o más equipos interconectados mediante una red con diferentes procesos ejecutándose en cada una de las partes y colaborando en forma simultánea, aunque por lo general son radicalmente diferentes entre sí.

La arquitectura Cliente/Servidor es una extensión al procesamiento cooperativo. En este esquema se une el poder de los grandes *servidores de base de datos* de los sistemas multiusuarios con la funcionalidad y ambiente gráfico de las *computadoras personales*.

El usuario final usa la aplicación front end para solicitar información del servidor de base de datos. El servidor recibe la petición, la procesa y envía los resultados al cliente para sean desplegados. El procesamiento del sistema sostiene las tareas de acceso a la base de datos y la presentación de la información.

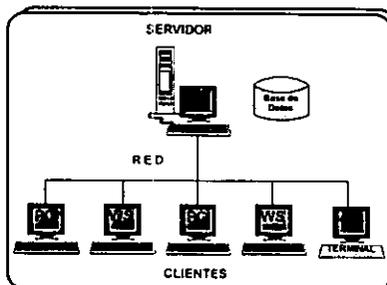


Figura (1.19) Arquitectura Cliente/Servidor

[12] Jenkins, Neil y otros - *Client/Server Unleashed*. - 1ª Edición - Editorial Sams - E.U.A. - 1996.

1.10.4. PROCESO COOPERATIVO

El modelo C/S implica el *proceso cooperativo* de los requerimientos solicitados por un cliente hacia el servidor, el cual procesa los requerimientos y regresa los resultados al cliente. El proceso cooperativo C/S es una forma especial de distribuir procesos.

Dentro de un ambiente LAN, las computadoras personales pueden estar ligadas entre sí para que puedan compartir recursos comunes como archivos, disco duro e impresoras. Dentro de la terminología LAN a esta clase de dispositivos compartidos son llamados *servidores* (Por ejemplo servidor de archivos, servidor de impresión). Este nombre es apropiado porque son usados para recibir solicitudes siendo dispositivos de servicio. Con esta clase de aprovechamiento, todo el acceso de las aplicaciones son ejecutadas en los clientes y solo ciertas funciones (Por ejemplo impresión)son distribuidas.

El *procesamiento del modelo C/S* es una extensión del proceso compartido y cooperativo. Dentro de este modelo la aplicación procesada esta dividida entre el *cliente* y el *servidor*. El proceso es actualmente iniciado por la solicitud de un servicio que hace el cliente y ambos cliente y servidor cooperar a ejecutar la aplicación exitosamente. El servidor de base de datos como Oracle, Sysbase, Informix son ejemplos de un ambiente de proceso cliente/servidor.

Una de las grandes ventajas del ambiente C/S es la compartición del *servidor de archivos* y el *servidor de base de datos*, por ejemplo, una PC (cliente) envía una solicitud de leer un archivo en el servidor, el servidor de archivos hace posible este proceso al cliente. En caso de un servidor de Base de Datos, la aplicación ejecutada en el cliente puede solicitar la consulta de determinados registros, para lo cual el servidor procesa solo los registros requeridos y los envía a la aplicación del cliente. Tanto el cliente como el servidor son recursos operativos que ejecutan la solicitud de un *query*.

En resumen la arquitectura C/S debe contar con:

- 1) Confiabilidad y robustez en las comunicaciones entre clientes y servidores.
- 2) Interacción entre el clientes y el servidor.
- 3) Proceso distribuido entre el cliente y el servidor.
- 4) Control por parte del servidor sobre los servicios o datos que los clientes puedan requerir.

1.11. COMPONENTES CLIENTE/SERVIDOR

A pesar de la separación lógica y física que existe entre el cliente y servidor, un sistema C/S coordina el trabajo de ambos componentes y eficientemente usa sus recursos disponibles para completar las tareas asignadas. La máquina cliente se comunica con el servidor a través de una red de área local (LAN - Local Area Network) o de cobertura amplia (WAN - Wide Area Network). Además, un sistema C/S puede involucrar más de un servidor. Cada servidor sostiene sobre un sistema C/S una función específica.

Como se anotó anteriormente, la arquitectura C/S esta compuesta por tres partes fundamentales: El front end (cliente), el back end (servidor) y la conectividad (red).

1.11.1. EL LADO CLIENTE

Es parte de la arquitectura C/S que esta enfocada a los usuarios finales. El cliente consiste en la aplicación front end y la computadora que un usuario final usa para acceder información a la base de datos. Cada vez más aplicaciones C/S tienen una interfaz gráfica de usuario (GUI - Graphic User Interfase) proporcionando mejor amigabilidad, mayor facilidad en su uso con solo apuntar y hacer click en los controles visuales como los botones, las cajas de listas y el desplegado de imágenes gráficas.

Atrás de la interfaz gráfica esta la programación que determina lo que se debe de hacer en respuesta a las acciones de los usuarios. En cada evento, por ejemplo hacer click en un botón, se encuentra alguna programación que invoca a una acción. Esta acción, puede ser que ejecute cálculos, habrá otra pantalla, envíe una solicitud a la base de datos para que valide la información capturada o despliegue información. Con el fin de garantizar, que la máquina cliente puede realizar todas estas tareas rápida y eficientemente, la computadora cliente debe cumplir con requerimientos de hardware como un rápido procesador y capacidad de memoria.

Dentro las aplicaciones front end que radican en el cliente; se podría hablar de lenguajes de programación, aplicaciones desarrolladas en 4 GL's u orientadas a objetos y reportadores; así como las herramientas para usuario final. Cualquiera de éstos debe cumplir cuando menos algunas de las características básicas como procesamiento local, independencia del back end, independencia de protocolos de comunicación, etc..

1.11.2. EL LADO SERVIDOR

El servidor o back end, consiste en el sistema manejador de base de datos (DBMS - Data Base Management System) y la computadora donde se encuentra éste. El DBMS almacena información dentro de la base de datos y es capaz de contar con múltiples clientes que lo accesen simultáneamente. Las aplicaciones front end pueden leer, modificar y borrar información de la base de datos a través de un Lenguaje Estructurado de Consulta (SQL - Structured Query Lenguaje), un lenguaje estándar de acceso de base de datos relacionales.

Para asegurar un rápido rendimiento, un DBMS debe soportar características avanzadas como almacenamiento de procedimientos y triggers. El DBMS también ejecuta numerosas funciones para dar mantenimiento a la base de datos, proporciona integridad a los datos, maneja procesos de transacciones, cumpliendo con las reglas de la organización. y soportando índices. También el servidor podría ser escalable para que pueda soportar un gran número de usuarios así como grandes volúmenes de información en el futuro. También debe cumplir con algunas características básicas como disponer de opciones para múltiples plataformas de hardware y sistemas operativos, permitir múltiples protocolos de comunicaciones, etc..

1.11.3. LA RED

El tercer componente del sistema C/S es la red. Las computadoras cliente se comunican con el servidor a través de la red. La red juega un importante papel en determinar el rendimiento del sistema C/S porque controla la rapidez en que las peticiones de los clientes son llevados al servidor, y la rapidez con que los resultados son enviados del servidor al cliente.

Dentro del entorno de un sistema C/S, uno de los objetivos del desarrollador es convivir eficientemente entre el cliente y el servidor para minimizar el tráfico de la red. La conectividad C/S debe proporcionar al front end un esquema sencillo que le permita conectarse transparentemente a los distintos DBMS y en las diferentes plataformas que soporten.

1.11.4. MIDDLEWARE

Durante el desarrollo de los sistemas C/S, es necesario ocultar los aspectos de la interacción entre el cliente y el servidor.

Middleware es el software diseñado para proporcionar conectividad dentro de una mezcla de plataformas de computo. Es un software de conectividad que soporta procesos distribuidos y es usado por los desarrolladores para la construcción de un sistema C/S. *Middleware* proporciona un alto nivel de protocolos de comunicación.^[13]

Middleware se ha convertido en un requerimiento importante para el éxito de un sistema C/S, proporcionando soluciones capaces de dar interoperatividad y portabilidad en las aplicaciones. La principal propuesta es el uso del *gateway*. Un *gateway* traduce peticiones y recibe resultados de un DBMS. Por ejemplo, una aplicación que se accesa a SQL Server puede también acceder datos de un DBMS DB2 a través de Micro Decisionware DB2 Gateway. Por lo regular, se necesita un *gateway* para cada tipo de DBMS que se requiera acceder.

Otra propuesta es la interfase común que es sencilla de programar y que es implantada mediante un *driver* por cada DBMS. El ODBC (Open Data Base Connectivity - Conectividad para acceder Bases de Datos) de Microsoft, es una interfaz estratégica para acceder datos en un ambiente distribuido con bases de datos relacionales o no relacionales.

SQL-NET DE ORACLE

La herramienta SQL-NET también funciona como un *middleware*, propio de Oracle y es una vía de acceso a la base de datos. Permite la compartición de datos entre bases de datos, aunque dichas bases de datos se encuentren en distintos tipos de servidores con distintos sistemas operativos y protocolos de comunicaciones. SQL-NET permite la comunicación

[13] Jenkins, Neil y otros.- *Client/Server Unleashed*.- 1ª Edición.- Editorial Sams.- E.U.A.- 1996.

especializada entre el front end y back end, proporcionando acceso, modificación y compartición de datos en la base de datos de Oracle.

El construir la plataforma C/S puede ser tan variada en sus componentes que no esta cerrada a un software o hardware específico. La compatibilidad es una propiedad que ofrece la arquitectura C/S solo hay que encontrar la combinación adecuada y adaptarla para cumplir eficientemente las necesidades de la organización.

1.12. ARQUITECTURA

Una *arquitectura* es poner definiciones y protocolos para construir un producto. Una *arquitectura de cómputo*, igual que la construcción, es un extenso plan para la completa forma de un sistema. Una *arquitectura Cliente/Servidor* abarca ambos, la arquitectura técnica y la de aplicación.

1.12.1. ARQUITECTURA TÉCNICA

La *arquitectura técnica* es la infraestructura sobre la cual la aplicación es construida. Es un detallado inventario de todos los elementos de hardware, software y la red que residen en los componentes cliente y servidor para todas las localizaciones donde está el sistema en operación. Debido a la gran variedad de mezclas de componentes para la infraestructura técnica, todos los miembros del equipo a cargo de un proyecto Cliente/Servidor deben probar y decidir oportunamente cual infraestructura es la más adecuada para que se cumplan los alcances del proyecto.

1.12.2. ARQUITECTURA DE APLICACIÓN

En la *arquitectura de la aplicación*, la tecnología Cliente/Servidor permite al diseñador y usuario final gran flexibilidad como para distribuir funciones y datos entre el cliente y servidor. Los alcances y la complejidad de la arquitectura de la aplicación es determinada por el numero de localizaciones, tamaño de las bases de datos, volumen de los datos, y demasiada concurrencia al acceso a los datos.

Una arquitectura de una aplicación específica como los componentes de la aplicación son distribuidos entre clientes y servidores. Estos componentes representan capas conceptuales que están alojadas en diferentes componentes físicos o de hardware. Esencialmente, tres capas necesitan ser distribuidas: *presentación, reglas de negocio y de lógica y manejo de datos* (Figura 1.20).

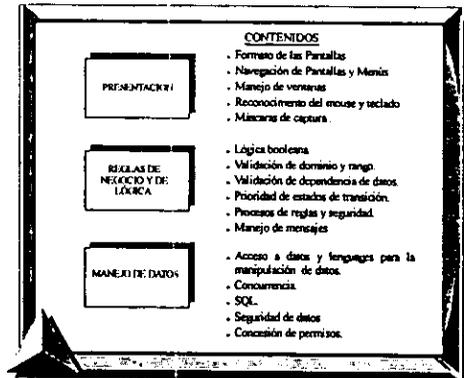


Figura (1.20) Capas de la Arquitectura de la Aplicación

La *capa de presentación* proporciona la interfaz de usuario para todo el sistema. Esto permite a los usuarios entrar y manipular datos, analizar la información y navegar por todas partes del sistema. La validación y verificación lógica es indispensable para la captura y selección de valores.

La *capa de reglas de negocio y lógica* procesa las políticas y restricciones que la aplicación es responsable de ejecutar. Esta incluyen: decisiones (¿Podemos presentar este curso el próximo martes?), políticas de ejecución (La aprobación de crédito es aceptada para cuentas que exceden los \$10,000 pesos), administración de decisiones (El límite de alumnos en una grupo de clase es de 20 estudiantes). Toda esta colección de reglas de negocio son atribuidas a los datos y los modelos.

La *capa de administración de datos* mantiene consistente y seguros los datos a través de aplicación de la integridad referencial y la seguridad de los datos. Las transacciones y las consultas a la base de datos son manejadas por esta capa e incluye las siguientes acciones: acceso, modificación, inserción, seguridad, respaldo, recuperación, restaurar, concesión de permisos

Cliente/Servidor permite implementar estas capas en una forma apropiada de localización física. Dando flexibilidad a los cambios físicos.

1.13. OPTAR POR CLIENTE/SERVIDOR

1.13.1. BENEFICIOS

Un sistema C/S proporciona tanto a la compañía como a los empleados numerosos beneficios. Dicho sistema ayuda a los usuarios hacer su trabajo mejor permitiendo dedicar su tiempo y energía en otras actividades como concluir negocios, atender a los clientes, realizar tareas administrativas. La tecnología C/S proporciona un acceso inmediato de la información para la toma de decisiones, facilita la comunicación y reduce tiempos, esfuerzo y costo para el realización de las tareas. A continuación se listan los principales beneficios de esta arquitectura:

Mejor Acceso a la Información.

Un sistema C/S es diseñado para dar a los usuarios un fácil acceso a toda la información que ellos necesitan obtener sobre el trabajo que están realizando. Con sólo oprimir un botón, la aplicación front end despliega información de las solicitudes de los usuarios. Esta información puede estar residida en diferentes bases de datos o aún estar en servidores físicamente separados. Para manipular la información los usuarios no necesitan registrarse en diferentes sistemas o depender de otra gente para obtener respuestas.

Incrementa la productividad.

Un sistema C/S incrementa la productividad de los usuarios proporcionándoles herramientas completas para que realicen sus tareas fácil y rápidamente. Por ejemplo, la pantalla de captura con controles gráficos y programación lógica soporta reglas de negocios, dando capacidad a los usuarios para introducir información rápida y con pocos errores, ya que automáticamente el sistema valida la información, ejecuta cálculos y reduce hasta el doble de tiempo y esfuerzo en la entrada de datos. Los sistemas C/S pueden integrarse con otras tecnologías así como correos electrónicos, documentos y groupware, por lo que la arquitectura C/S esta dirigida a la ganancia productiva.

Procesos Automáticos.

Un sistema C/S puede automatizar procesos de negocios de una empresa eliminando la labor manual mediante procedimientos completos que se ejecutan automáticamente con la finalidad de agilizar las actividades proporcionando resultados confiables y rápidos.

Poderosa capacidad de reportes.

Porque la información dentro de un sistema C/S es almacenada en una base de datos relacional, ésta puede ser fácilmente consultada por reportes específicos. Los programadores pueden crear rápidamente nuevos reportes usando SQL y los usuarios finales pueden también generar sus propios reportes a través de un reporteador sin tener que aprender SQL.

Mejor servicio al cliente.

Una compañía puede mejorar el servicio a sus clientes reduciendo tiempos de respuesta. Con los sistemas C/S se tiene un mejor control y manejo de la información, clave para proporcionar un mejor servicio.

Rápido Desarrollo de Aplicaciones.

Cada vez existen más herramientas de desarrollo para crear aplicaciones C/S tomándose ventaja con la programación orientada a objetos y el desarrollo de módulos de aplicación. Los objetos y códigos reutilizables igualmente aceleran el desarrollo de nuevos sistemas. Las herramientas para el diseño de la GUI facilitan a los programadores el crear pantallas visuales sin tener que programar su código. Las aplicaciones C/S pueden ser fácilmente modificadas en caso de existir cambios o alguna nueva regla de negocio. Además, las herramientas C/S pueden ser usadas

fácilmente para crear prototipos que ayudan al desarrollador para presentar el sistema a los usuarios y obtener comentarios inmediatos.

Reducción de costos y ahorro.

Un sistema C/S reduce costos de varias maneras, algunos pueden ser fácilmente cuantificados que otros. Muchas compañías han reemplazado su sistema mainframe con arquitectura C/S y se han ahorrado millones de dólares en los costosos mantenimientos. Otros se han beneficiado por el acceso de información y la reducción significativa de costos de papel, incluyendo su almacenamiento y distribución. Las personas cumplen sus tareas rápida y eficientemente con este tipo de sistemas, ahorrándose tiempo, esfuerzo y dinero.

Incremento de ganancias.

Un sistema C/S da fácil acceso a información crucial que juega un papel en identificar oportunidades para incrementar las utilidades y tomar decisiones correctas. Por ejemplo, identificar los tipos de productos que a los clientes les gusta, capacidad de detectar oportunidades y tendencias de ventas.

Arquitecturas abiertas.

C/S implica siempre sistemas abiertos. Un sistema abierto define un conjunto de normas y características de plataformas compatibles (tanto en el lado cliente como en el servidor), sobre los cuales se desempeñan los procesos, aprovechando características particulares de cada tecnología computacional.

Reducción de trabajo en el lado servidor.

El utilizar la CPU del lado cliente para tareas específicas evita que los equipos servidores se saturen. El ejemplo más simple sería validar un dato que está ingresando. Si bien la validación ocuparía unos cuantos ciclos de la CPU del servidor, en una operación que se repite millones de veces, el tiempo de proceso es significativo. Con esto se disminuye los accesos a la base de datos que consumen muchos recursos. Además se aprovecha las características específicas de los grandes servidores para funciones que le son propias, arquitecturas de entrada / salida mucho más amplias, CPU poderosas, subsistemas dedicados que agilizan operaciones particulares.

Interfaz Amigable.

En un ambiente gráfico, los usuarios se familiarizan mas rápido con la operación amigable que proporciona un sistema C/S, ya que por lo regular están acostumbrados a manejar el entorno de Windows y si el sistema posee las mismas bondades o hasta mejores, es más fácil su aceptación. Disminuyéndose el costo invertido en el entrenamiento del sistema.

1.13.2. MITOS

Desafortunadamente existen campañas publicitarias que hacen ver el desarrollo un sistema en C/S como algo trivial y muy fácil. Desarrollar e implantar exitosamente un sistema C/S es todo un reto que requiere de trabajo duro y sobre todo tiene un precio

monetario. Los dos grandes mitos que circulan entorno al desarrollo de la arquitectura C/S es que un sistema robusto puede desarrollarse de la noche a la mañana y que el implementarlo no es caro. Las personas deberían estar entusiastas acerca de la tecnología C/S pero también tener expectativas realistas.

Los sistemas C/S pueden ser desarrollados en cortos tiempos.

Realmente se pueden desarrollar sistemas C/S con pocas pantallas y con un mínimo de funcionalidad en relativamente cortos periodos, pero si se requiere un sistema robusto que soporte 50 usuarios, mejoren el tiempo de respuesta, soporten reglas de negocio, tengan objetos reutilizables y aseguren la integridad de los datos. No se puede acompletar esta tarea de la noche a la mañana. La aplicación front end debe ser diseñada cuidadosamente para que ésta sea amigable. Se tiene que identificar el mejor método para la convivencia entre el cliente y servidor de acuerdo a los requerimientos. El modelo de datos debe ser minuciosamente diseñado y ejecutado para crear la base de datos. El crear un sofisticado sistema C/S toma tiempo para terminarlo. Pero si el primer sistema C/S es desarrollado correctamente. entonces los esfuerzos por desarrollar futuros sistemas podrían terminarse mucho más rápido ya que se reutilizaría el código.

El movimiento a la tecnología C/S no es cara.

Algunas personas están bajo la impresión falsa de que el adoptar la tecnología C/S no es costosa. En realidad, la inversión financiera no es insignificante si es la primera vez que se quiere desarrollar un sistema C/S. Para empezar se tiene que adquirir nuevo software que incluye herramientas de desarrollo, el DBMS, productos para el modelado de datos, que necesitan ser comprados. Nuevo hardware que contemplan máquinas cliente y servidores que pueden actualizarse o comprarse y cursos de capacitación y entrenamiento.

1.13.3. ADOPTAR CLIENTE/SERVIDOR

Cada compañía ha introducido la arquitectura C/S de manera diferente, conforme a lo que tienen y a sus necesidades. Algunas empresas han reemplazado a los mainframes con esta tecnología, otras han cambiando sus aplicaciones residentes en servidores de archivos a C/S, otras más han integrado la arquitectura a sus sistemas actuales sin que sean sustituidos. Para lo cual existen varios tipos de adaptaciones C/S, las más comunes son:

Sustitución (downsizing).

Muchas empresas han examinado el papel que el mainframe está jugando en un futuro y han decidido remplazarlo para integrar la tecnología C/S. Las tres principales causas que convencieron a los directivos a realizar el cambio, son las siguientes:

- Incapacidad de soportar anuncios de queries.
- Incapacidad para modificar rápidamente sus programas.
- Alto costo de mantenimiento y soporte para su cuidado y ejecución.

No reemplazar pero si integrar.

Otras compañías que tienen mainframes no las han eliminado en favor a la tecnología C/S. Ellos han integrado las dos tecnologías tomando ventaja de ambas. Estas organizaciones continúan influenciadas con las aplicaciones del mainframe y también realizan procesos que son mejor ejecutados en los sistemas C/S. Por ejemplo, sistemas que son procesos batch y no requieren una interacción con el usuario, pudiendo estar escritas desde terminales, mientras que sistemas que requieren interacción con el usuario son desarrolladas sobre C/S. Algunas veces las dos tecnologías son justamente integradas para que el mainframe actúe como servidor de base de datos dentro de un sistema C/S

Modificación (Upsizing).

Upsizing es la tendencia a mudar las aplicaciones de PC a soluciones más seguras en ambiente C/S permitiendo que estas mismas sean migradas de manera que se aproveche el código de la aplicación y el conocimiento del desarrollador. Muchas pequeñas y grandes empresas que tienen aplicaciones de base de datos residentes sobre servidores de archivos están pasándolas a C/S debido a que han encontrado limitaciones con sus actuales sistemas ya que la organización ha crecido en tamaño y sus requerimientos han cambiado. Ellos han girado a C/S porque sus sistemas son incapaces de manejar un gran número de usuarios, sostener grandes volúmenes de información y presentar un rendimiento aceptable. Además de que estas compañías están enfocándose más a la integridad de datos, seguridad compartición de información, rentabilidad y escalabilidad., y sus sistemas existentes no cumplen todas estas características, por lo que se han actualizado a C/S.

1.13.4. HERRAMIENTAS DE DESARROLLO

Las herramientas de desarrollo tienen un alto desempeño en la creación de pantallas que permiten poner campos, texto fijo y elementos funcionales (como menús, botones, cajas de diálogo, etc) e incluso elementos decorativos como pueden ser bitmaps o manejo de distintas topografías. Por lo que rara vez se dirá un “lenguaje” a una herramienta, ya que mucho del software para C/S hace algo más que código visible, la mayoría tiene “algo” para pintar pantallas sin código visible.

También cuentan sin excepción con algún lenguaje de programación. Los lenguajes son, por regla general, de muy alto nivel de abstracción, de manera que en muchas ocasiones carecen de instrucciones de bajo nivel que permitan por ejemplo, abrir un puerto de comunicaciones. Esto implica que con frecuencia se deberá hacer uso de extensiones en un lenguaje tradicional. Además, en las herramientas es común que carezcan de rigor estructural, ya que permiten variables carentes de tipo, funciones o rutinas con más de un punto de entrada o salida. Esto no es malo, pero tienden a hacer confuso y consecuencia más costoso el mantenimiento.

Pero un gran beneficio del uso de herramientas es la rapidez y facilidad de desarrollo de aplicaciones ^[14]. La alternativa de los lenguajes tradicionales existe, pero en C se tiene la desventaja de la cantidad de código a escribir y el conocimiento especializado que debe tener el programador.

Cabe mencionar que casi todas las herramientas de desarrollo modernas están basadas en objetos y en consecuencia se benefician del encapsulamiento, herencia y polimorfismo, las características fundamentales de la tecnología de orientación a objetos.

En proyectos pequeños (entre 20 y 50 pantallas o de 10 y 30 tablas) normalmente se tendrá una solución más adecuada usando una herramienta para C/S pequeña. El territorio que se encuentra por arriba de esos límites es el ideal para C/S grande.

C/S cuenta con muchas herramientas de desarrollo. En orden alfabético se mencionarán las siguientes:

- Gupta SQL Windows (Actualmente Centura)
- Informix 4GL
- Developer 2000 Oracle
- PowerBuider de PowerSoft/Sysbase
- Uniface
- Visual Basic

1.14. ASPECTOS TECNICOS PARA INICIAR UN SISTEMA C/S

Al trabajar en un proyecto C/S, se deben considerar algunos aspectos técnicos, los más importantes son:

Escalabilidad.

Es importante que el sistema pueda ser escalable. Escalabilidad se refiere a la capacidad que tiene un sistema de soportar más usuarios, manejar grandes volúmenes de datos y sostener incrementos de carga de trabajo. Un sistema C/S debe ser capaz de poder manejar varios accesos y grandes demandas sin requerir muchas modificaciones, por ejemplo, la organización podría crecer o gente de otros departamentos podrían acceder la información del sistema o también se podría integrar el sistema C/S con otros sistemas y compartir recursos, como la base de datos. Si un sistema no es escalable, se debe reescribir código o comprar un software o hardware más poderoso. Es mejor anticipar el crecimiento y seleccionar el equipo que soporte las demandas del futuro.

Multiplataforma.

Porque muchas compañías tienen ambientes heterogéneos, un sistema C/S podría tener soporte para múltiples plataformas, especialmente si varios departamentos con diferente infraestructura de cómputo van a usar el sistema. Por ejemplo, un sistema

^[14] La aplicación, es en esencia, el conjunto de programas necesarios para que el usuario final maneje de manera transparente la información.

C/S puede ser utilizado por un departamento que use PCs y otro que use Macintoshes, u otro que soporte UNIX. Ciertamente los productos C/S pueden soportar múltiples plataformas y otros no. En este ejemplo, es necesario que la solución C/S soporte diferentes plataformas

Herramienta de desarrollo.

Hay una gran variedad de productos software para el desarrollo de aplicaciones front end y para manejar las bases de datos. El seleccionar el idóneo para el sistema C/S requiere las capacidades y características que posee el producto y evaluarlo y compararlo con otros productos.

Ambiente de desarrollo.

Un equipo de trabajo debe tener su propio servidor de base de datos. El grupo no debe desarrollar en una máquina donde los usuarios la utilizan para hacer su trabajo. El grupo de trabajo estaría constantemente reconfigurando, optimizando, probando, bajando la base de datos y todas esas tareas afectarían los sistemas que estén operando, disminuirían el rendimiento y crearían descontento a los usuarios. El contar con una red dedicada para el desarrollo es también benéfico, ya que se pueden hacer configuraciones y pruebas sin que se afecten a terceros.

Código reusable.

El principio de todo equipo de desarrollo es crear código reusable que pueda ser compartido. Un diseño inteligente es que mucho del código, objetos y funciones puedan ser reutilizadas por otros programadores en el actual proyecto, así como futuros proyectos. La aplicación de técnicas de desarrollo orientado a objetos y métodos modulares sobre la aplicación front end o en los procedimientos almacenados en la base de datos pueden significar reducción de tiempo en la programación y puede hacerse más fácilmente el mantenimiento a la aplicación.

Código reusable de otro sistema C/S.

El revisar otros sistemas C/S, se podría encontrar código o entidades que pueden ser reutilizadas para el nuevo sistema C/S. Una de las ventajas de la tecnología C/S es la habilidad para reusar el código existente, beneficio que siempre se debería aprovechar.

Transferencia de datos.

Si el sistema C/S va a obtener datos de otras fuentes, se tiene que examinar esos datos y definir planes para que puedan ser obtenidos. Por lo que es necesario considerar algunos de los obstáculos que se pudieran encontrar como el tipo de dato y la longitud del campo, palabras reservadas, volúmenes masivos de datos y programas que necesitan ser escritos para sostener la actual conversión y transferencia. El planear apropiadamente los resultados de la transferencia asegura que todos los datos dentro del sistema C/S son exactos y completos.

Seguridad.

Se deben de considerar dos tipos de seguridad para un sistema C/S. La primera medida de seguridad es la diseñada dentro del mismo sistema, así como lógicamente en la base de datos, asegurando que sólo personas autorizadas tengan el privilegio de acceder al sistema o tener alguna funcionalidad específica, o ver cierta información. Las medidas de seguridad pueden implantarse basadas en calificadores como la

posición de las personas, sus departamentos y entonces llevarlas a cabo a través de login IDs, passwords e encriptamiento. Estas medidas de seguridad garantizan que información confidencial y restringida no pueda ser comprometida. Otra tipo de seguridad es la que se aplica físicamente a la base de datos del servidor, el cual debe estar en una área segura y que sólo personal autorizado pueda entrar, evitándose accidentes o mal uso del servidor.

Arquitectura de la red.

La arquitectura de la red debe ser revisada para asegurarse de que sea capaz de soportar las demandas de un sistema C/S como carga tráfico en la red, acceso a más usuarios, transferencia o recepción de información. La red debe sostener estos incrementos y continuar con un rendimiento aceptable para todos los servicios.

Acceso Remoto.

Un sistema C/S debe estar especialmente diseñado para soportar accesos remotos tomando en consideración una infraestructura de red robusta y la planeación de procesos que aseguren que los usuarios remotos obtendrán fácil acceso y rápido tiempo de respuesta.

1.15. ARQUITECTURA C/S SEGUNDA GENERACION

La primera generación de la tecnología Cliente/Servidor fue una continuación de la tendencia hacia la descentralización y acentúa la naturaleza de la revolución de la PC, aunque represento desventajas significativas en términos de flexibilidad y gran durabilidad.

En Hurwitz Consulting Group creen que se están por testificar un cambio dramático en la computación C/S, cambio que se traducirá en el uso más productivo de los recursos de computo con el advenimiento de una nueva generación de computación C/S.

1.15.1. CARACTERÍSTICAS

Las características de la segunda generación son:

- Respalda el movimiento hacia una infraestructura basada en computación distribuida, la cual tiene una arquitectura que permite que los componentes de computo se coloquen en cualquier plataforma que sea más compatible con la estructura de una organización.
- Un desarrollador dividirá el código de la aplicación entre varios servidores diferentes, naciendo el concepto de “particionamiento de la aplicación o sistemas distribuidos”. Las razones de dividir la lógica de la aplicación son el rendimiento y la flexibilidad.
- En la mayoría de los sistemas C/S de la segunda generación tienen incluida una seguridad distribuida, garantizando protección en la administración del ambiente de acceso.
- Modelo de datos corporativos, componente importante en la evolución global de la computación. Cliente/Servidor segunda generación, proporciona un método que permite a una organización tener definiciones comunes de las entidades ya que los sistemas de información se distribuyen en toda la organización.

- Las herramientas de la primera generación les falta flexibilidad de desplazar la lógica entre las plataformas cliente y la del servidor. Por consiguiente, una aplicación diseñada para utilizarse en una plataforma Windows no podría modificarse de manera que se pusiera el código de la aplicación en el servidor. De igual forma una aplicación diseñada con una herramienta basada en el servidor era incapaz de mover una parte de código al cliente. Las herramientas que están surgiendo y que caen dentro de la computación C/S Segunda generación permiten desplazar su lógica en los clientes y puede colocarse en una variedad de plataformas servidor. Las herramientas que caen dentro de esta categoría incluyen a Informix New-Era, Dynasty, Forte y Seer Technology.
- Las plataformas pueden aceptar un número cada vez más grande de usuarios que ejecutan aplicaciones aún más complejas y manejen elevados volúmenes de transacciones crecientes. Su objetivo es proporcionar un alto grado de gradualidad.
- Manejan eventos simultáneamente y múltiples procedimientos sin que se reduzca el rendimiento. Se añade lógica cada vez mas compleja a la aplicación.

1.16. EJEMPLO: INFRAESTRUCTURA DE LA ARQUITECTURA C/S

PEMEX, en la mayoría de sus dependencias cuenta con la siguiente infraestructura de computo de sistemas en arquitectura Cliente/Servidor.

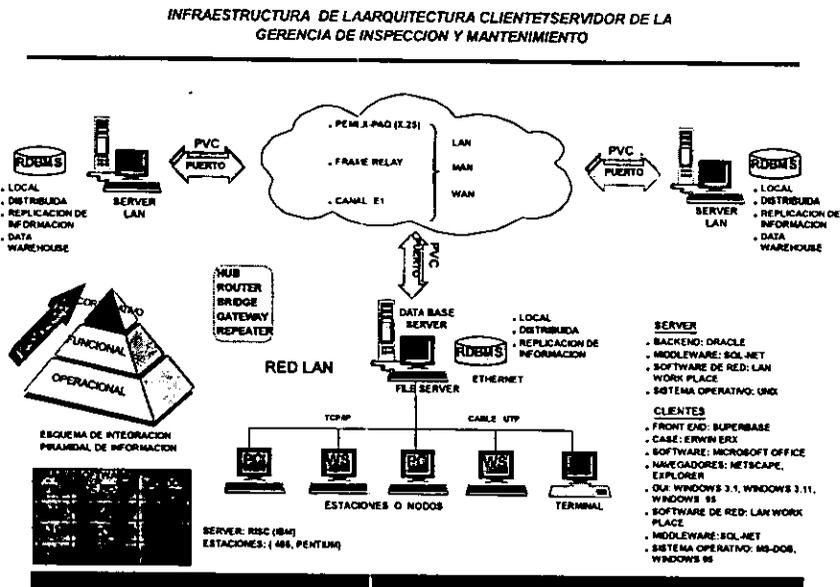


Figura (1.21) Esquema de la Plataforma C/S de la GIM

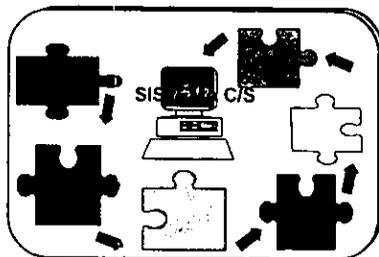
Para comprender mejor este esquema es recomendable entender primero la Figura 1.11 (Red de Cómputo de la GIM). Y ampliando la explicación tenemos que las estaciones de

trabajo ahora llamadas clientes, cuentan con la herramienta SUPERBASE como front end para la operación del sistema C/S, además del Erwin Logic como herramienta CASE para la elaboración del Diagrama Entidad - Relación.

La conexión con la base de datos, es a través de SQL-NET, donde el manejador de base de datos es Relacional, teniendo como back end a ORACLE, ubicado físicamente en el servidor, convirtiéndose en servidor de base de datos. La forma en que recibe y responde a las solicitudes de los clientes es por medio de SQL-NET. E igualmente cuenta con gateways para la conexión a diferentes bases de datos

CAPITULO II

METODOLOGIA PARA EL DESARROLLO DE SISTEMAS CLIENTE / SERVIDOR



2.1. INTRODUCCION

En los siguientes capítulos se estudian cada una de las actividades asociadas con el desarrollo de sistemas de información bajo la arquitectura cliente/servidor, estos conceptos y teorías que engloban el análisis y diseño de sistemas están ejemplificados con un Sistema Presupuestal llamado SISEFIN que fue desarrollado para PEMEX, así como comentarios y experiencias obtenidas durante la construcción del mismo.

Algunos datos son ficticios principalmente los nombres del personal de PEMEX y la información presupuestal que sea calificada como confidencial, aclarando que solo es una medida de seguridad sin afectar la calidad de la tesis.

Antes de iniciar con la metodología que se aplicó al sistema SISEFIN es necesario tener firmes los siguientes conceptos:

2.2. ANALISIS Y DISEÑO DE SISTEMAS

El análisis y diseño de sistemas se refiere al proceso de examinar la situación de una empresa con el propósito de mejorarla con métodos y procedimientos más adecuados.^[15]

El desarrollo de sistemas esta formado por dos grandes componentes:

Análisis de sistemas: Es el proceso de clasificación e interpretación de hechos, diagnóstico de problemas y empleo de la información para recomendar mejoras el procedimiento actual.

[15] Senn, James.- *Análisis y Diseño de Sistemas de Información.- Traduc. Urbina Medal Edmundo Gerardo y otros.- 2ª Edición.- Editorial McGRAW-HILL.- México.- 1994 - pp. 11.*

Diseño de sistemas: Es el proceso de planificar, reemplazar o complementar un sistema organizacional existente.

El análisis y diseño de los sistemas es un proceso en el que se aplican muchas herramientas, para mejorar los negocios mediante la implementación o el cambio de los sistemas de información.

El análisis y diseño de sistemas pretende estudiar sistemáticamente la operación de ingreso de datos, el flujo de los mismos y la salida de la información; todo ello dentro del contexto de una empresa en particular.

El análisis y diseño es más importante que el mismo desarrollo del sistema. Una analogía apropiada es cuando se realiza la construcción de un edificio habitacional, un arquitecto debe estudiar cuidadosamente el terreno y el medio ambiente donde va a edificar, imagínense que se decidiera construirlo sin hacer un previo análisis, simplemente con los planos de la fachada, quizá sea una arquitectura moderna y lo termine en poco tiempo pero que sucede en el interior, cuando se quiera poner la tubería o la luz y no existan las instalaciones adecuadas y se tenga que romper la pared o el piso para introducir el cableado o ductos necesarios, o se detecta que son cuartos húmedos en la noche y calurosos en el día porque la zona donde fue construido es de un clima tropical y el material de construcción no es el idóneo para estas condiciones y así se pueden describir otros problemas que generan la remodelación total del edificio o en casos críticos su demolición.

Lo mismo sucede con un sistema de información si no se hace un análisis y diseño bien estructurado las probabilidades de éxito serán nulas y se tendrá la necesidad de estarlo parchando o cambiando, y lo más triste no es que el sistema lo guarden en archivo muerto sino que el tiempo y esfuerzo de las personas involucradas haya sido infructuoso.

2.3. PERFIL DEL ANALISTA Y LIDER DE PROYECTO

Un grupo de trabajo está compuesto por varias personas que desempeñan diferentes roles, entre ellos están los analistas, gente responsable de la construcción de los cimientos de un sistema de información. Dependiendo de los alcances del proyecto, en un equipo de trabajo puede haber uno o más analistas que conjuntamente estudiarán las necesidades y requerimientos de los usuarios, proponiendo la mejor de opción para el manejo y aprovechamiento de la información.

El líder de proyecto también cumple con las mismas aptitudes que el analista pero con la particularidad de que él es responsable de monitorear el desarrollo del sistema hasta su liberación reportando periódicamente los avances del proyecto a los directivos y siendo el representante del equipo de trabajo.

2.3.1. CUALIDADES DEL ANALISTA DE SISTEMAS

➤ Es un **investigador** que estudia de forma sistemática los procesos de los negocios, los documenta y proporciona mejoras.

- Es un **solucionador de problemas**. El analista debe distinguir entre síntomas, problemas del usuario y causas. Con sus conocimientos y experiencia de la tecnología de las computadoras, el analista debe ayudar al usuario a explorar aplicaciones novedosas y más útiles de las computadoras.
- Debe ser un **buen interlocutor**, manteniendo una relación cordial con otra gente, durante largos periodos. El analista a menudo se encuentra entre usuarios, administradores, programadores, auditores, y otros diversos participantes, los cuales frecuentemente están en desacuerdo entre sí. Ante esta situación su labor principal es obtener un consenso y esto requiere del delicado arte de la diplomacia y la negociación.
- El analista de sistemas necesita siempre **prepararse** y estar al día sobre temas de computación para entender las capacidades de las computadoras y poder recomendar ampliamente la mejor solución a los usuarios y programadores. Esto es que el analista de proyecto es un **consultor** sobre ciertos tópicos de la informática y dispone de una experiencia profesional respecto al hardware y al software y a sus aplicaciones en la empresa.
- Debe ser **autodisciplinado** y **automotivado** como individuo.
- El analista es el protocolo entre las necesidades del usuario y la computadora.

Pero entre todas estas cualidades la más importante es el **saber relacionarse con los usuarios**, entender sus necesidades y convencerlos con las posibles sugerencias que proponga. El analista no debe rendirse, ni doblegarse ante el carácter de los usuarios, tampoco actuar de manera imponente e irrespetuoso. El analista de sistemas también toma el papel de psicólogo, sabiendo como manejar a la gente sin que estos se sientan manejados. Aunque a fin de cuentas, recordemos que el usuario es el que costea el sistema y usualmente tiene el derecho de rehusarse a pagar si no está conforme con el producto.

2.3.2. CUALIDADES DEL LÍDER DE PROYECTO DE SISTEMAS

- Cumple con las mismas aptitudes que el analista.
- Debe tener tacto para manejar un grupo de trabajo, dar las instrucciones de manera clara y sencilla con diplomacia y firmeza, crear una atmósfera de compañerismo y que todos estén enfocados a un mismo objetivo, que sientan al sistema como un reto y motivación personal.
- Es **administrador** y **coordinador** de actividades, además de **programador** de tiempos para el desarrollo del sistema.
- Es **comunicador** oportuno de los avances o problemas del proyecto a las personas directamente responsables como directivos, principales usuarios, coordinadores del proyecto.

Si un líder de proyecto cumple con estos requisitos los resultados son casi siempre, mayor calidad en el software y menor tiempo de desarrollo, lo cual beneficia a todos.

2.4. USUARIOS

El usuario es aquél (o aquéllos) para quien se construye el sistema. Es la persona a la que tendrá que entrevistar el líder y grupo de trabajo del proyecto, a menudo con gran detalle, a fin de conocer las características que deberá tener el nuevo sistema para poder tener éxito.

Los usuarios son la parte más importante y delicada que interviene en el desarrollo de un sistema. Un usuario es un cliente, él tiene la razón, pero qué sucede cuando el usuario está equivocado e insiste que el sistema cumpla con ciertos detalles, que a consideración del líder de proyecto no son los más apropiados. Para este tipo de situaciones, el líder de proyecto nunca debe comprometerse con algo que considere erróneo. La mayoría de estos usuarios solo quieren sacar el trabajo rápido y se niegan a realizar reingenierías o depuraciones de algunos procesos que afecten la forma acostumbrada en que trabajan, además de que no tienen visiones a futuro.

El líder de proyecto debe aprovechar las bondades de la tecnología computacional en la realización de tareas y si se puede minimizar o depurar los procesos tiene que hacerlo. Y la mejor manera de convencer a los usuarios es mostrar tangiblemente - ya sea con documentación, gráficas, diagramas o prototipos - el comportamiento y las consecuencias si se elige cierta opción y si están dispuestos a aceptar tal responsabilidad.

Actualmente los usuarios que trabajan en las empresas aprenden más acerca de computación acumulando experiencia al trabajar con aplicaciones que fueron desarrolladas para ellos o capacitándose en diversos aspectos en sistemas de información. Con lo anterior y sumando que existen herramientas más poderosas, fáciles y rápidas para desarrollar sistemas, la participación de los usuarios se vuelve cada vez más activa, surgiendo aplicaciones propias sin necesidad de un líder de proyecto o personas especializadas, pero con el riesgo de que estas no estén bien diseñadas, causando una serie de problemas cuando esta operando.

El hecho de que existan potentes lenguajes para el desarrollo de sistemas no garantiza que las aplicaciones generadas sean igualmente eficientes. La clave para que un sistema de información tenga éxito es el disponer con una adecuada metodología para la construcción del sistema, actividad que requiere de los conocimientos especializados y experiencia de una persona que invirtió años de estudio para su preparación.

En el Instituto Mexicano del Petróleo (IMP) su usuario principal es Petróleos Mexicanos (PEMEX). El IMP fue creado para dar servicio a PEMEX en todos sus ámbitos especialmente en actividades de investigación petrolera. Pero con la creciente demanda de informar sobre su producción y operaciones diarias a directivos y a la misma población mexicana, PEMEX recurrió a la tecnología computacional para facilitar y agilizar sus procesos de información, para lo cual el IMP proporciona sus servicios informáticos para satisfacer las necesidades de PEMEX.

Para PEMEX es indispensable que el IMP conozca sus operaciones administrativas, contables, petroleras, sus planes futuros, sus políticas, en pocas palabras a toda la organización. El saber todo esto da gran ventaja, porque el líder de proyecto detecta cuáles son los problemas y necesidades del usuario, y así propone sistemas adecuados para la empresa, acertando casi en un casi un 100% que sean autorizados los proyectos por PEMEX.

Con lo anterior, la función del área de informática de una organización debe ir más allá del mero desarrollo de sistemas de información, deberá estar al tanto de los últimos avances en las tecnologías de la información, de las tendencias en el campo de la administración y, lo que es más importante, deberá tener claros los objetivos y metas de la empresa y conocer a profundidad las diferentes áreas del negocio.

El sistema SISEFIN fue solicitado por una dependencia de PEMEX Exploración - Producción, llamada Gerencia de Inspección y Mantenimiento (GIM), la cual tiene oficinas en Villahermosa, Tabasco; Poza Rica, Veracruz y Cd. Del Carmen, Campeche. La GIM es el usuario del sistema SISEFIN y el que se encargará de proporcionar todos los elementos que conformen al sistema, además de realizar actividades de revisión, pruebas y operación del mismo.

2.5. CONCEPTOS DE SISTEMAS ORGANIZACIONALES

En tiempos actuales las empresas tienen la necesidad de ser más productivas y competitivas. Para lograr este objetivo, un elemento de suma importancia es la información.

2.5.1. LA NUEVA CONCEPTUALIZACIÓN DE LA EMPRESA

David R. Vicent presenta una nueva forma de mirar a la empresa. Los participantes de la empresa son los individuos, grupos y demás entidades que tienen el interés legal en el éxito de la empresa (inversionistas, empleados, legisladores, clientes, competidores, proveedores, etc.). Entre los participantes de la empresa deben existir un acuerdo con respecto a la relación bajo la cual se forma una organización que den como resultado una unidad de propósito.

Para que se dé la relación armoniosa entre los participantes empresariales, y la empresa pueda ser productiva, las empresas deben considerar como uno de sus activos importantes a la **información**, la cual debe ser administrada y usada eficientemente, y sobre ella debe basarse la entidad. Así, la información se convierte en un elemento más en el costo de un bien o servicio y le agrega valor y deja de ser un intangible.

Cuando se dice que la empresa debe basarse en la información, debe entenderse que todas sus actividades deben tenerla como sustento y debe propiciar que las actividades que realiza la empresa sean únicamente aquéllas que agregan valor. Más aún, debe permitir que todo el personal esté en posibilidades de tomar decisiones sobre su trabajo de manera prácticamente autónoma.

De lo anterior, se desprende que la empresa debe identificar sus procesos, conservar aquéllos que agregan valor y eliminar a los que no lo agregan y, por otra parte debe proveer de herramientas de tipo informático a su gente para la realización y coordinación de sus actividades, además de hacer uso de las tecnologías de la información para mantenerse informada y mantener la comunicación con los participantes de la empresa. De este modo, la informática y sus tecnologías propician un cambio en la estructura de la empresa a través del desarrollo y operación de sistemas de información que la orienta a ser más competitiva.

2.5.2. SISTEMAS DE INFORMACIÓN

Un sistema es un conjunto de componentes que interaccionan entre sí para lograr un objetivo común. Una organización es un sistema. Todo sistema organizacional depende, en mayor o menor medida, de una entidad abstracta denominada sistema de información.^[16]

Un sistema de información es el medio por el cual los datos fluyen de una persona o departamento hacia otros, puede ser desde la comunicación interna entre los diferentes componentes de la organización, líneas telefónicas, hasta sistemas de cómputo que generan reportes periódicos para varios usuarios.

Los sistemas de información proporcionan servicio a todos los demás sistemas de una organización y enlazan todos sus componentes en forma tal que éstos trabajen con eficiencia para alcanzar el mismo objetivo.^[17]

2.5.3. CARACTERÍSTICAS IMPORTANTES DE LOS SISTEMAS

La finalidad de un sistema es la razón de su existencia. Los sistemas que interactúan con su medio ambiente (reciben entradas y producen salidas) se denominan **sistemas abiertos**. En contraste, aquellos que no interactúan con su medio ambiente se conocen como **sistemas cerrados**.

El elemento de control está relacionado con la naturaleza de los sistemas, sean cerrados o abiertos. Los sistemas trabajan mejor cuando operan dentro de niveles de desempeño tolerables. Todos los sistemas tienen niveles aceptables de desempeño, denominados **estándares** y contra los que se comparan los niveles de desempeño actuales. La información proporcionada al comparar los resultados con los estándares junto con el proceso de reportar las diferencias a los elementos de control recibe el nombre de **retroalimentación** (Figura 2.1).^[18]

[16] [17] Senn, James.- *Análisis y Diseño de Sistemas de Información*.- Traduc. Urbina Medal Edmundo Gerardo y otros.- 2ª Edición.- Editorial McGRAW-HILL.- México.- 1994 - pp. 19,22.

[18] Senn, James.- *Análisis y Diseño de Sistemas de Información*.- Traduc. Urbina Medal Edmundo Gerardo y otros.- 2ª Edición.- Editorial McGRAW-HILL.- México.- 1994 - pp. 21,22.

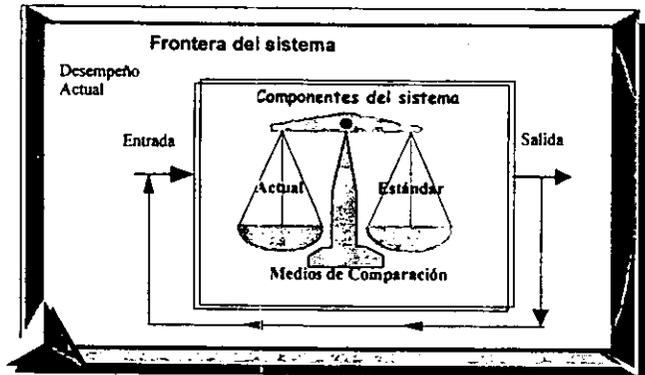


Figura (2.1) Modelo de Control

Para resumir, los sistemas emplean un modelo de control básico consistente en:

1. Un estándar para lograr un desempeño aceptable.
2. Un método para medir el desempeño actual.
3. Un medio para comparar el desempeño actual contra el estándar.
4. Un método de retroalimentación.

2.5.4. OBJETIVOS DE LOS SISTEMAS DE INFORMACIÓN

Las finalidades de los sistemas de información son procesar entradas, mantener archivos de datos relacionados con la organización y producir información, reportes y otras salidas. El conjunto particular de subsistemas utilizados - equipo específico, programas, archivos y procedimientos - es lo que denomina una **aplicación de sistemas de información** siendo el resultado de los requerimientos individuales de algún área de la organización.

2.5.5. PAPEL DE LOS SISTEMAS DE INFORMACIÓN

Un principio fundamental en el desarrollo de sistemas de información para las empresas es que las aplicaciones son una herramienta que debe tenerse para utilizar la tecnología de la información.

En consecuencia, los sistemas deben desarrollarse sobre la base de su propia capacidad para mejorar el desempeño de la organización. Los sistemas de información resultan eficaces si son desarrollados conforme a los objetivos, valores y metas de la organización.

Las solicitudes de proyectos siempre se deben preparar y evaluar de acuerdo con este principio.

2.5.6. TIPOS DE SISTEMAS DE INFORMACIÓN

El líder de proyecto de sistemas desarrolla diferentes tipos de sistemas de información para satisfacer las diversas necesidades de una empresa. Entre las más importantes categorías están:

SISTEMAS PARA EL PROCESAMIENTO DE TRANSACCIONES	<p>Los sistemas de procesamiento de transacciones son aquellos sistemas de información computarizados que se desarrollan para procesar grandes volúmenes de información generada en las funciones administrativas, tales como la nómina o el control de inventarios. Con la finalidad de mejorar las actividades rutinarias que se realizan manualmente; sin embargo, el elemento humano sigue participando, al llevar a cabo la captura de la información requerida.</p> <p>Tales sistemas ejecutan periódicamente los programas de manera automática. Una vez preparados, escasamente se requiere el tomar decisiones. Los sistemas de procesamiento de transacciones brindan velocidad y exactitud; además se pueden programar para seguir rutinas sin ninguna variación.</p>
SISTEMA DE INFORMACION ADMINISTRATIVA	<p>Los sistemas de información administrativa ayudan a los directivos a tomar decisiones y resolver problemas. Los directivos recurren a los datos almacenados como consecuencia del procesamiento de las transacciones. A su vez, los analistas de sistemas desarrollan reportes bien estructurados que contienen la información necesaria para las decisiones o que indican el estado de las variables importantes.</p>
SISTEMA PARA SOPORTE DE DECISIONES	<p>Los sistemas para el soporte de decisiones ayudan a los directivos que deben tomar decisiones sobre situaciones particulares. Apoyan la toma de decisiones en circunstancias que no están bien estructuradas. Sin embargo, la decisión en sí, depende de la persona responsable de la misma. Estos sistemas se diseñan con la orientación hacia la persona o el grupo que los utilizará.</p>
SISTEMAS EXPERTOS	<p>Los sistemas expertos son en sí, un tipo muy especial de sistemas de información. Los sistemas expertos utilizan los enfoques del razonamiento de la inteligencia artificial para resolver problemas. Un sistema experto (también llamado sistema basado en el conocimiento) captura; y en efecto utiliza, el conocimiento de un experto, para la solución de un problema particular de la organización. Un sistema experto selecciona la mejor solución al problema.</p>

El líder de proyecto debe identificar el tipo de sistema que va a desarrollar para darle el tratamiento adecuado. Para el sistema SISEFIN es un sistema de información administrativo enfocado a dos tipos de usuarios: el operativo que se encarga de alimentar al sistema con información presupuestal para darle un mejor seguimiento y control; y el usuario

administrativo que tiene la responsabilidad de interpretar los reportes para dar soporte a sus decisiones.

2.6. FUNDAMENTOS DE LOS PROYECTOS

La demanda de nuevos proyectos, originada por los usuarios, es mucho mayor que la oferta de líder de proyectos de sistemas que puedan trabajar sobre éstos. Los líderes de proyectos de sistemas no trabajan sobre cualquier proyecto que ellos seleccionen sino más bien lo hacen sobre aquellos solicitados por los administradores, empleados e incluso por el propio personal de sistemas.

En PEMEX como en cualquier otra organización invierte dinero para el desarrollo de sistemas Cliente/Servidor, por esta razón es necesario justificar cada nuevo proyecto y evaluarlo para determinar si es factible a los objetivos y recursos de la entidad

La solicitud de proyecto, es la semilla del desarrollo de un nuevo proyecto.

2.6.1. CAUSAS PARA PROPONER PROYECTOS

Resolver un problema: Actividades, procesos o funciones que en la actualidad, o quizá en el futuro, no satisfacen los estándares de desempeño o las expectativas y para lo que es necesario emprender una acción que resuelva las dificultades. Este es el caso del sistema SISEFIN, solucionar el deficiente control del ejercicio presupuestal de la gerencia.

Aprovechar una oportunidad: Un cambio para ampliar o mejorar el rendimiento económico de la empresa y su competitividad. Aunque la GIM pertenezca a una empresa de servicio, el disponer de un sistema presupuestal para su control interno, mejorará la administración de sus recursos financieros.

Dar respuesta a directivos: Proporcionar información en respuesta a órdenes, solicitudes o mandatos originados por una autoridad legislativa o administrativa; llevar a cabo tareas de cierta manera, o también cambiar la información o tal vez el desempeño. Esta es la principal causa por la que se propuso del desarrollo del sistema SISEFIN, el desplegar información consolidada y detallada de los movimientos presupuestales de la gerencia, que ayude a la planeación y seguimiento del presupuesto y se detecten oportunamente posibles desviaciones .

Para iniciar proyectos, se busca ante todo que los sistemas de información mejoren:

La Capacidad	<ul style="list-style-type: none"> ✓ Capacidad de procesar transacciones con rapidez. ✓ Manejo eficiente del volumen creciente de transacciones. ✓ Recuperación más rápida de la información.
El Control	<ul style="list-style-type: none"> ✓ Mayor exactitud y mejora en la consistencia. ✓ Salvaguardar datos importantes y sensibles en una forma que sea accesible sólo al personal autorizado.

La Comunicación	<ul style="list-style-type: none"> ✓ Acelerar y coordinar el flujo de información y mensajes entre localidades remotas así como dentro de oficinas. ✓ Integración de áreas de la empresa.
Los Costos	<ul style="list-style-type: none"> ✓ Monitoreo de los costos dándoles seguimiento para determinar su evolución en relación con lo esperado. ✓ Reducción de costos al procesar los datos, con un costo menor del que es posible con otros métodos al mismo tiempo que se mantienen la exactitud y los niveles de desempeño.
Ventaja Competitiva	<ul style="list-style-type: none"> ✓ Modificar los servicios proporcionados incrementado su productividad y eficiencia. ✓ Disminuir costos y tiempos, siendo más competitivos dentro del mercado. ✓ Monitorear la forma que va operando la organización, detectando oportunamente posibles desviaciones.

2.6.2. FUENTES DE SOLICITUDES DE PROYECTO

Existen cinco fuentes primarias de solicitudes de proyectos. Los solicitantes dentro de la organización son los *empleados* que canalizan sus necesidades a través de su jefe inmediato, los *jefes de departamento*, los *altos ejecutivos* (gerentes, directores, consejos, vicepresidentes) y los *líderes de proyectos* de sistemas que buscan áreas donde deben desarrollarse proyectos y escriben la propuesta, o animan a un gerente para que éste permita la elaboración de la propuesta en su nombre. Por otra parte, es probable que *fuentes externas* (el gobierno) a la organización también soliciten proyectos de sistemas de información.

De acuerdo con el origen de la solicitud y el motivo para hacerla, los solicitantes buscan ya sea aplicaciones totalmente nuevas o algunos cambios en las ya existentes. Para del sistema SISEFIN su nacimiento fue a través de los usuarios presupuestales de la GIM, fuente que solicitó el desarrollo de un sistema que les ayudará al mejor manejo de las operaciones financieras, esta necesidad se dió a conocer al gerente para que evaluará la propuesta del proyecto junto con otros altos ejecutivos.

2.6.3. CONTRATACIÓN DE SERVICIOS INFORMÁTICOS EXTERNOS

La contratación de una empresa "experta" en Tecnología Informática es hoy en día una de las soluciones para el desarrollo de sistemas C/S. Los conceptos se resumen en un termino en inglés *Outsourcing* que significa obtener recursos externos, confiar en un tercero la responsabilidad de funciones informáticas, considerando como puntos esenciales que:

- La organización nunca dejará en manos de un proveedor de outsourcing los elementos estratégicos del negocio.

- El proveedor de outsourcing juega el papel de “socio” de la organización. Por lo cual su selección debe ser sumamente cuidadosa.

Los servicios que están pasando a ser contratados por un número creciente de organizaciones son los relacionados con los sistemas de información y las telecomunicaciones, partiendo de la base de la posibilidad de obtener sustanciales ahorros en los costos, que pueden ser canalizados al desarrollo de la “arquitectura cliente/servidor” y toda la infraestructura tecnológica que ésta demanda.

La idea subyacente en el outsourcing es que si la empresa ha de ser competitiva debe enfocar sus esfuerzos únicamente a aquellas actividades que le dan valor agregado, por tanto la integración de sus sistemas, incluidos los de información, debe ser realizada por un grupo externo a ella. En otras palabras, se sostiene que una empresa alimentaria, por ejemplo, tiene como objetivo producir alimentos, no desarrollar sistemas, así que la creación e integración de sistemas de computo no es una de las actividades del negocio.^[19]

A primera vista, este concepto parece muy lógico. Si una empresa ha de ser competitiva debe enfocar sus esfuerzos a aquellas actividades que pueden hacerla la mejor en su ramo. Sin embargo, algunos autores, consideran que no puede ser conveniente dejar en manos de un grupo externo la integración de los sistemas de una organización ya que puede ser una desventaja más que una ventaja, porque no existe garantía de que esa entidad esté compenetrada con las áreas del negocio y pueda comprenderlas a profundidad para integrar sus sistemas de modo que pueda lograr una ventaja competitiva.

Pero si se trata de empresas pequeñas o medianas que no poseen un área de desarrollo de sistemas, el outsourcing aplicado convenientemente puede dar buenos resultados. Si se trata de empresas grandes y se deja la integración de los sistemas a manos del área de informática de la empresa, el responsable de dicha área puede recurrir a la asesoría externa para realizar mejor sus funciones y lograr que, en efecto, los sistemas del negocio lo apoyen para lo largo de sus objetivos y el cumplimiento.

Las dependencias de PEMEX cuenta en la mayoría de los casos con suficiente capital para la adquisición y mantenimiento de tecnología de punta para que operen sistemas de información. Para estas áreas existen sectores informáticos que se encargan de atender las necesidades de información, las cuales son innumerables por ser uno de los organismos más grandes e importantes en México. Ante esta realidad, empresas privadas y el propio IMP ofrecen sus servicios para el desarrollo de sistemas y soporte técnico a PEMEX, por lo que Petróleos Mexicanos reparte sus tareas entre los organismos que cumplan con calidad sus requerimientos. La mayor parte de sus proyectos son dedicados al Instituto Mexicano del Petróleo, empresa con mas de 30 años de apoyo a PEMEX brindando atención a sus objetivos primordiales obteniendo resultados satisfactorios.

[19] Padua Diaz, Marcela.- *Transformational Outsourcing.- Revista Soluciones Avanzadas. Tecnologías de Información y Estrategias de Negocios.- Xview, S.A de C.V. - No. 28.- 15 de Diciembre de 1995.- México.- pp. 4 - 6.*

2.6.4. MANEJO DEL PROCESO DE SELECCIÓN Y REVISIÓN DE PROYECTOS

La decisión de aceptar o rechazar una solicitud puede tomarse de muchas formas diferentes y por distintos miembros de la organización. Los líderes de proyectos de sistemas no son los árbitros finales. Uno de los métodos más comunes para revisar y seleccionar proyectos para su desarrollo es por medio de un comité. En algunas organizaciones pueden tener comités que se encarguen de supervisar y revisar las propuestas de sistemas de información. Algunos de los comités son:

- **Comité directivo** está constituido por gerentes importantes de varios departamentos de la organización así como miembros del grupo de sistemas de información. Cualquier proyecto que sea enfocado para la GIM es revisado por el gerente junto con sus subgerentes, determinando cuales son factibles. Entre estos proyectos estuvo la solicitud del desarrollo del sistema SISEFIN.
- **Comité de sistemas de información** está integrado por gerentes y líderes de proyectos del departamento de sistemas de información.
- **Comité de grupos de usuarios** está formado por usuarios.
- **Otros métodos** son cuando los gerentes contratan compañías de sistemas independientes; dichas compañías manejan todo el análisis y el trabajo de diseño para los proyectos. Una desventaja de este enfoque es la posibilidad de que el departamento patrocine el desarrollo de un sistema mientras, al mismo tiempo, el grupo de sistemas de información o la alta gerencia ignoran que el proyecto se esté llevando a cabo. En muchos casos PEMEX se ve afectado por esta situación.

El comité incorpora los puntos de vista de los gerentes y empleados de toda la compañía. Este enfoque acepta que los sistemas de información son un recurso que sirve y beneficia a todos los miembros de la organización. Para esto el comité formulará las siguientes preguntas para las aplicaciones propuestas.

- El desarrollo de esta aplicación ¿traerá beneficios a largo plazo, capacidad, control, comunicación, costos o ventaja competitiva?
- ¿Cuál es el nivel de recursos financieros para esta función o real?
- ¿Existen suficientes datos para juzgar el desempeño actual con los sistemas existentes?

2.6.5. ADMINISTRACIÓN DEL PROCESO DE DESARROLLO DE SISTEMAS DE INFORMACIÓN

El manejo de sistemas de información es tan importante para los planes de las empresas, que en ocasiones se contratan gerentes con la única responsabilidad de administrar estos sistemas. Comprar la tecnología adecuada es la parte fácil; el reto es adecuar la tecnología a las necesidades de la organización. Alcanzar un alto grado de adecuación es un aspecto fundamental para el éxito de la compañía. Las aplicaciones seleccionadas deben ser aquellas que brinden los mayores beneficios para la compañía.

2.7. CICLO DE VIDA ESTRUCTURADO PARA EL DESARROLLO DE SISTEMAS CLIENTE/SERVIDOR

Con las metodologías de análisis, desarrollo y planeación de sistemas, empieza a conformarse la disciplina que actualmente conocemos como informática. En las metodologías de análisis se identifican los procesos de la organización en la que se hace uso de lo que actualmente se conoce como *reingeniería de procesos*.

El ciclo de desarrollo o de vida de los sistemas C/S es un enfoque por etapas de análisis, diseño, y construcción que postula el desarrollo de sistemas de información, adoptando una arquitectura Cliente/Servidor.

El método del ciclo de vida para desarrollo de sistemas Cliente/Servidor, es el conjunto de actividades que los líderes de proyectos llevan a cabo para desarrollar e implantar un sistema de información bajo una plataforma cliente/servidor.

La metodología que se aplicó en la mayoría de los sistemas Cliente/Servidor desarrollados por una área informática del IMP, fue la integración del análisis y diseño estructurado, ingeniería de software y la experiencia en la construcción de arquitecturas Cliente/Servidor. Actualmente esta metodología está obteniendo resultados satisfactorios.

El método del ciclo de vida para el desarrollo de sistemas Cliente/Servidor consta de:

1. Planeación del sistema (Propuesta Técnico - Económica).
2. Análisis Específico del sistema (Reingeniería de Procesos) y la plataforma C/S.
3. Diseño Preliminar del sistema Cliente/Servidor y su infraestructura.
4. Elaboración del Prototipo.
5. Diseño Final del sistema Cliente/Servidor y su infraestructura.
6. Construcción del sistema y la plataforma cliente/servidor.
7. Implantación
8. Operación del sistema Cliente/Servidor
9. Liberación

Para el desarrollo del SISEFIN se siguió la misma metodología sólo que aplicando también la reingeniería de software.

2.7.1. ESBOZO DE UNA METODOLOGÍA PARA LA REINGENIERIA DE SISTEMAS CLIENTE/SERVIDOR

Dada la complejidad del proceso de migración, se hace imprescindible el establecimiento de una metodología que facilite esta tarea y garantice la calidad del resultado final, reduciendo al mismo tiempo su costo.

A continuación se presentan una serie de fases que pueden servir como orientación hacia la creación de una metodología general para la migración de aplicaciones a nuevas plataformas.

1. Estudio de la vieja aplicación
2. Planeación del Sistema (Propuesta Técnico - Económica).
3. Análisis Específico del Sistema (Reingeniería de Procesos) y la plataforma cliente/servidor.
4. Diseño Preliminar del sistema C/S (Reingeniería de Software) y su infraestructura.
5. Elaboración del Prototipo.
6. Diseño Final del sistema Cliente/Servidor y su infraestructura.
7. Construcción del sistema y la plataforma cliente/servidor.
8. Implantación
9. Operación del sistema Cliente/Servidor
10. Liberación

Esta metodología es muy parecida a la anteriormente presentada, con la variante de que se parte con una previa aplicación y ésta requiere ser analizada para ver que se puede aprovechar y como se puede optimizar lo que ya se conoce y se aplica. La *reingeniería de software* requiere de la observación y los conocimientos del líder de proyecto tanto informáticos como del tema del que trata el sistema. Para poder visualizar las dimensiones de ambos de lo que se tiene con los que se quiere y ganar tiempo y sobretodo reducir esfuerzos.

En general, no existe ningún método correcto para desarrollar un sistema de información C/S, pero sí existen diferentes formas para producir el sistema correcto para una aplicación. En la comunidad empresarial existen muchas variaciones de los métodos aplicados a un sistema C/S. Algunos métodos tienen más éxito que otros y esto depende de cuándo se emplean, cómo se aplican y de los participantes en el proceso de desarrollo.

El indicador definitivo del éxito de un método de desarrollo en particular es aquel que se refiere a los resultados obtenidos y no a la "precisión" teórica del método.

ESTRUCTURA DE LA METODOLOGIA DEL SISTEMA "SISEFIN"

2.8. ANTECEDENTES DEL SISTEMA SISEFIN

Antes de presentar todo lo que fue el desarrollo el sistema SISEFIN, quisiera exponer algunos puntos importantes que influyeron en la parte de los alcances del sistema y sus objetivos.

El ambiente en el cual nace y se desenvuelve el sistema SISEFIN es totalmente financiero. Desde un principio los alcances del sistema abarcaban las necesidades

relacionadas tanto a nivel gerencia, como de la subdirección, considerándose un proyecto muy ambicioso en arquitectura cliente/servidor.

Ante este reto, el proyecto fue autorizado por la Gerencia de Inspección y Mantenimiento (GIM) PEMEX, dándose en marcha la etapa de análisis y diseño, la cual fue concluida. Posteriormente se diseñó y presentó el prototipo y con el paralelamente se fueron programando algunos módulos básicos del sistema. Ya teniendo un avance aproximado del 40% en su construcción, el sistema SISEFIN tomo un giro de 360 grados que afecto considerablemente su seguimiento normal. Por situaciones políticas y administrativas se requería urgentemente de un sistema que satisficiera solo las necesidades de la subdirección teniendo como base la misma plataforma computacional del SISEFIN pero diferente Front end (de SuperBase a Sql-Windows).

Con esta radical decisión, el sistema tubo que ser dividido en dos: uno que conservo el mismo nombre SISEFIN, cubriendo las necesidades gerenciales y otro llamado PRESST enfocado a las necesidades de la subdirección. Ante este cambio de planes, fue preciso aplicar la reingeniería de software, identificando los procesos que deberían ejecutar cada sistema sin duplicarse, haciendo una comparación de lo que se tenía, con respecto a los nuevos requerimientos.

Por cada sistema, se volvió a realizar otra vez un análisis y diseño. Pues como ya eran dos sistema aparentemente independientes en su desarrollo, se pretendía que fueran mas detallados, cada uno al nivel al que pertenece, pero guardando una estrecha relación en su lógica presupuestal.

Con este antecedente nace el nuevo SISEFIN, respetando su infraestructura cliente/servidor anteriormente propuesta, pero con un nuevo enfoque en el Control Interno Presupuestal desarrollado para la GIM.

Partiendo de esta explicación, la presente tesis expone la segunda fase por la cual atravesó el sistema, aplicando la metodología estructurada y considerando la reingeniería de procesos y de software. Actualmente el sistema SISEFIN esta operando con los requerimientos de esta nueva etapa.

Es lamentable, que cuando ya se esta en una etapa avanzada dentro del ciclo de vida de un sistema, se tenga que cambiar drásticamente los alcances del mismo debido a cuestiones políticas y/o administrativas. La primera fase del SISEFIN sufrió de estos factores, causando que el trabajo que ya se tenia fuera modificado casi en un 70%, requiriendo que se volvieran a realizar un nuevo análisis y diseño.

Este cambio no fue previsto por el analista porque desde un principio el sistema fue autorizado y enfocado hacia una gerencia. Pero al enterarse las otras dependencias sobre este proyecto, se interesaron, y pidieron que fuera general tomando los requerimientos sólo de las subdirecciones, además de que estaban dispuestos a financiarlo hasta su terminación. Dando como consecuencia que se volviera a rediseñar el sistema.

Con esta situación uno se plantea una serie de preguntas:

¿Por qué no se dejó el sistema SISEFIN tal como estaba y se volvió a hacer otro sistema que cumpliera con las necesidades de la subdirección?

Porque se duplicarían procesos, siendo que algunas de las necesidades de las subdirecciones las abarcaba el SISEFIN.

¿Por que no sólo se quitaron los módulos que fueran a estar duplicados?

En un principio se depuro, y se deshabilitaron ciertos módulos, pero como ya era más enfocado a los requerimientos de la GIM, los usuarios querían que el sistema cubriera a detalle los procesos, tanto que hasta la fecha el sistema que esta operando no se parece mucho al propuesto anteriormente.

¿Cuáles fueron las consecuencias que se provocaron por este cambio?

En primera se creo un descontento por parte del equipo de trabajo que estaba desarrollando inicialmente el sistema SISEFIN, pues detectaron que su trabajo no iba a tener los resultados esperados. Además mucho de lo que se tenía avanzado iba a modificarse y se tendría que volver a empezar desde el principio. Lo rescatable de la situación fue que se pudo aprovechar cierto código ya programado, se estandarizaron más reglas de programación y evitaron errores que se habían cometido en la primera fase. Igualmente se aplico y tomo experiencia de la Reingeniería de Software.

A continuación se presenta la metodología que se aplicó para el desarrollo del Sistema SISEFIN. Esta metodología resume de manera esquemática y sencilla las etapas más importantes para la construcción de un sistema Cliente/Servidor. Al mismo tiempo facilita la organización y control de las actividades de un proyecto. Sobre este último aspecto es conveniente recordar que un factor esencial para el éxito de un proyecto de desarrollo de sistemas C/S, es el control de todos los aspectos involucrados.

2.9. PLANEACION DEL SISTEMA

El propósito de planear y preparar es clave para el éxito de cualquier proyecto C/S. Es esencial que el equipo de trabajo conozca claramente los retos a los cuales se enfrentarán al iniciar el proyecto como son definir sus alcances y beneficios, entender los procesos de negocios, desarrollar el sistema y construir una sólida infraestructura para éste.

Los alcances de un proyecto C/S son usualmente definidos conforme a cada problema que debe ser resuelto o a un nuevo proceso que puede mejorar los negocios. Por ejemplo, una situación donde los representantes de ventas tiene que registrarse en tres diferentes sistemas para obtener información que ellos necesitan en la toma de decisiones, esto provoca un ineficiente uso de tiempo de los representantes. Los alcances del proyecto pueden ser desarrollar un sistema C/S donde los usuarios se registren a un sencillo sistema que se conecte a todos los datos que ellos necesitan y más aún que trabajen en un ambiente gráfico. Es importante asegurar que los alcances del proyecto sean realistas y manejables, teniendo la seguridad de que se tienen los esquemas y recursos necesarios para poner en marcha el proyecto

Antes de empezar con la propuesta de un sistema, debemos saber quién o quiénes solicitan que se haga el sistema, cuál es el objetivo principal que va a cubrir el sistema, quiénes los responsables para que se acepte y quién el coordinador que va a dar seguimiento hasta su terminación del proyecto. Para contestar estas preguntas, analizaremos los siguientes puntos:

2.10. OBJETIVOS GENERALES

Es importante identificar cuál es el objetivo principal que pretende cumplir el sistema. En ocasiones es difícil definirlo en sus inicios pero conforme el análisis preliminar avanza los objetivos se vuelven más claros.

Estos objetivos son los requerimientos que se pretenden cubrir con el sistema C/S. Existen los objetivos generales y los específicos que detallan particularmente las metas a lograrse.

El objetivo principal del sistema SISEFIN es Implantar un sistema que opere en arquitectura cliente/servidor para el Control Interno y Seguimiento Financiero de la Gerencia de Inspección y Mantenimiento.

2.11. DETERMINAR SI EXISTEN PROYECTOS SIMILARES

Es importante identificar algunos proyectos que hayan sido atendidos en el pasado y que tengan alguna similitud con los alcances del nuevo sistema. Esto es bueno porque se puede aprender de los errores de otros proyectos y no repetirlos. También se necesita determinar si los usuarios finales o dirigentes ya han participado en reuniones para discutir los detalles sobre anteriores proyectos.

Los usuarios de la GIM ya se han visto involucrados en muchas sesiones para determinar cuestiones financieras para un sistema, por lo que el equipo de trabajo del sistema SISEFIN tuvo que tomar medidas adecuadas en el momento que se iniciaban las juntas del nuevo proyecto para que la participación fuera activa, ya que la mayoría de las veces los usuarios ya están cansados de tratar con lo mismo.

2.12. DESIGNACION DE RESPONSABLES

En este aspecto debe de conjuntarse un grupo sólido para el seguimiento del sistema. Las personas responsables por parte del desarrollo del sistema tendrán que formar un equipo con los usuarios interesados. Y para que exista esta integración deben existir coordinadores que supervisen como se están desempeñando las actividades.

Los jefes de la División de Sistemas del IMP, acordaron entrevistas con la Gerencia de Inspección y Mantenimiento, PEMEX Exploración - Producción, para ofrecer servicios informáticos a esta dependencia.

El superintendente, Ing. Sánchez encargado del Area de Mantenimiento de la GIM solicitó la realización de un Sistema para el Control Interno del Ejercicio Presupuestal de la gerencia, anunciando mediante un oficio que requería por parte de IMP una Propuesta Técnico - Económica del Proyecto, la cual sería revisada por las demás áreas que componen la gerencia a nivel región y posteriormente al gerente para su autorización o desaprobación.

Directamente el IMP se coordinará con el Ing Sánchez, el cual será el usuario potencial que proporcionará la información necesaria y los requerimientos a cubrir, también hará las revisiones sobre los avances del sistema. A su vez el IMP tendrá un grupo de trabajo dirigido por el Mat. Vázquez, Ing. Ramos y Lic. Escobedo. Las personas anteriormente mencionadas serán las responsables del seguimiento del sistema SISEFIN hasta su terminación.

2.13. INTEGRACION DEL EQUIPO DE TRABAJO

El desarrollo de software es un proceso que requiere la intervención de diferentes actividades que pueden relacionarse por una o varias personas las cuales conforman un equipo o grupo de trabajo.

Un equipo de trabajo debe poseer estructura, experiencia, y talento necesario para entregar un sistema con calidad en funcionalidad y tiempo. Este equipo es responsable en diseñar, desarrollar e implantar el sistema C/S, así como cuidar su progreso y mantenimiento. El construir un equipo sólido es todo un reto, porque se requiere de conocimientos sobre Cliente/Servidor, programación orientado a objetos, metodologías de desarrollo de sistemas, reingeniería, administración de base de datos, etc.. Muchas compañías construyen exitosamente proyectos C/S a través de la combinación de un equipo de desarrollo de software con grupo de asesoría externa de sistemas de información.

ROLES Y RESPONSABILIDADES

A fin de especificar mejor la tarea de cada integrante de un equipo de trabajo se han definido roles. Un *rol* es un tipo de trabajo, un conjunto de actividades relacionadas. Un equipo de trabajo se conforma por varios roles que deben cubrirse y responsabilidades que deben ser aceptadas.

A continuación se listan una serie de roles donde las responsabilidades las puede cumplir una misma persona.

- El **líder de proyecto** es responsable de asegurarse de que el proyecto tenga éxito, supervisando desarrollo conduciendo las reuniones, y reportando los progresos a los dirigentes y usuarios finales.
- El **analista** determina los requerimientos funcionales para el sistema por medio de entrevistas a los usuarios finales para determinar sus necesidades y requerimientos. Esta

persona desarrolla el diseño funcional de especificaciones incluyendo las reglas de negocio y el plano de los procesos.

- El **arquitecto técnico** es el responsable de diseñar la arquitectura Cliente/Servidor y asegurar que todos los componentes estén apropiadamente definidos e implementados.
- El **diseñador** de la aplicación esboza la interfaz gráfica de usuario de la aplicación front end garantizando que sea fácil e intuitiva para usar.
- El **programador** de la aplicación front end hace el código de la aplicación y determina que objetos pueden compartirse para ser reutilizadas, igualmente programa las librerías.
- El **administrador de la base de datos** es el responsable de dar mantenimiento a la base de datos y al servidor, asegurando que pueda ser utilizado por los desarrolladores. Esta persona garantiza que el servidor este físicamente en óptima condición para un mejor rendimiento y que todas las entidades de las bases de datos apropiadamente creadas y disponibles.
- El **modelador de datos** es responsable de crear el modelo de datos para soportar el sistema C/S. Esta persona se encarga de que los datos estén eficientemente almacenados y fácilmente accesibles para el sistema.
- El **programador de la base de datos** es responsable de escribir los procedimientos almacenados y triggers (disparadores) así como todo el código SQL sea eficiente. Esta persona asegura que la aplicación front end pueda acceder fácilmente los datos y que todas su peticiones sean atendidas por la base de datos.
- El **usuario final** es el que comparte sus conocimientos con el analista y el que participa en las pruebas de evaluación.
- El **especialista en redes** se encarga de todo lo que respecta a la comunicación entre el cliente y el servidor, administra el tráfico de la red, protocolos y el tamaño de los paquetes. Esta persona garantiza que la red siempre este levantada y que pueda soportar la arquitectura de un sistema C/S.
- Los **representantes del soporte de hardware y software** son los responsables de que las computadoras usadas para el proyecto estén configuradas correctamente y que trabajen apropiadamente.
- Los **evaluadores** son los responsables de probar el sistema, checando que funcionalmente sea correcto para ser usado.
- El **documentador** es el que se encarga de elaborar la documentación para el usuario final sobre como usar el sistema C/S, así como la documentación técnica para que los programadores las usen.
- El **entrenador** es el responsable de desarrollar el material didáctico y capacita a los usuarios finales de como usar el sistema C/S.

Según los alcances y la complejidad del proyecto una persona puede desarrollar uno o varios roles y un rol puede ser cubierto por una o varias personas. Por ejemplo si un proyecto es de gran escala, con una inversión millonaria es necesario que se involucren varios individuos para llenar un solo rol, por ejemplo un proyecto puede requerir muchos

programadores para la aplicación front end que trabajen juntos para obtener el trabajo. Pero si se tiene un sistema con alcances cortos, un simple individuo puede sostener múltiples responsabilidades, por ejemplo una persona puede ser responsable de todo el análisis, diseño y programación del sistema C/S.

Las metas u objetivos deben ser razonables y con cierto reto como ingrediente. Si las metas fueran de fácil logro, carecerían de motivación para el equipo de trabajo, por otra parte, si las metas fueran demasiado difíciles y no pudieran cumplirse, el efecto sería igualmente negativo. Los líderes de proyecto de sistemas debe pensar en metas de productividad para procesos que muestren salidas tangibles.

Mientras que una meta se encuentra predeterminada, los medios para alcanzarla no lo están. Y en este caso los integrantes del grupo de trabajo cuentan con toda la libertad para hacer uso de su experiencia y habilidad.

2.14. REINGENIERIA: ESTUDIO DE LA VIEJA APLICACIÓN

Si se cuenta con un software, pero se requiere que este sea modificado o actualizado, soportando modernas plataformas, es necesario aplicar la **reingeniería de software**. Contar con software actualizado y eficiente favorece la competitividad de cualquier organización.

2.14.1. DEFINICIÓN

La reingeniería de software plantea la migración de las viejas aplicaciones que están funcionando de modo más o menos satisfactorio a nuevos sistemas, modificando solamente lo imprescindible para que el software se adapte a las nuevas necesidades de la empresa.^[20]

Las aplicaciones ya en funcionamiento han sido probadas por los propios usuarios, que han comprobado su utilidad, conocen con profundidad sus aciertos y sus carencias y las han ido adaptando con el tiempo a sus necesidades. Es posible aprovechar el trabajo empleado en el desarrollo de viejas aplicaciones para el desarrollo de otras nuevas que cubran las mismas viejas necesidades además de algunas nuevas que puedan haber surgido, con un código más eficiente y una interfaz más amigable, migrando normalmente a plataformas más modernas.

En reingeniería se analizará la aplicación existente, usando para ello la documentación de la misma, el código fuente y la experiencia del personal que la ha utilizado y que domina por completo su manejo.

La reingeniería de software constituye una alternativa a la creación de nuevas aplicaciones, muy beneficiosa en términos de costo y tiempo de desarrollo. Con la reingeniería está garantizado que los usuarios de la vieja aplicación no tendrán problema en adaptarse a la nueva aplicación porque el flujo de pantallas y trabajo es muy similar a la conocida vieja aplicación.

[20] *El liderazgo en la computación como eje de la Competitividad Internacional.- Memorias del Simposium Internacional de Computación, celebrado en el CENAC.- IPN.- Octubre de 1996 - pp. 130.*

2.14.2. BENEFICIOS

De modo general, los beneficios que se esperan conseguir con la reingeniería de software son:

- 1) Proporcionar a los usuarios finales una interfaz más amigable y fácil de usar.
- 2) Reducir tiempo y la dificultad del entrenamiento del usuario final proporcionando un flujo de pantallas al de la aplicación original con la que trabajaba actualmente.
- 3) Conservar las funcionalidades de una aplicación que ha demostrado ser de utilidad y que se ha adoptado a las necesidades de la empresa, añadiéndole las nuevas funcionalidades que se necesiten.
- 4) Permitir que los datos ya almacenados en la base de datos sigan siendo útiles para la nueva aplicación.

2.14.3. METODOLOGÍA

- **Comprender el funcionamiento general de la aplicación antigua:** Es básico recoger toda la documentación posible sobre el programa (código fuente, documentación de diseño, documentación de llamadas al sistema y rutinas externas,...). Para el analizar el código se debe tomar en consideración lo siguiente:
 - Examinar librerías existentes en la vieja aplicación.
 - Conocer la estructura, contenido y arquitectura de los programas existentes para poder determinar el código reutilizable o que debe ser reconstruido.
 - Identificar los estándares del entorno de programación y desarrollo, como: convenciones de nombre, organización de los programas, compilación del código fuente, etc.
 - En caso de que se requiera efectuar la migración de código se tendrá que estudiar cómo migrar cada proceso de la vieja aplicación al haberlos clasificado, y se reflejaría la decisión tomada sobre qué código viejo mantener y cuál reescribir.
- **Conocer la estructura de la Base de Datos:** Extraer la estructura de los datos en caso de que no exista un diagrama Entidad - Relación que representa el modelo conceptual de los mismos. Junto con este diagrama es imprescindible disponer de la información relativa a la estructura de las tablas de datos a las que accede la aplicación.
- **Extraer la estructura funcional:** Identificar la estructura de la aplicación y usarla para crear un diagrama de Flujo de Datos y de Pantallas.
- **Concentrarse en los módulos de la vieja aplicación:** Identificar que datos son los que se capturan, los que se eligen, los que se calculan, los que genera el sistema, además de como están estructuradas la secuencia de pantallas.
- **Resultados:** Al final del estudio se habrá obtenido una descripción exhaustiva del viejo producto y documentación del flujo de la aplicación (de procesos y pantallas). Para lo cual se habrá conseguido un conocimiento profundo de la antigua aplicación que servirá

de base para poder determinar con exactitud qué es lo que se pretende conseguir con la nueva, qué es lo que se debe de mantener, qué se debe mejorar, cuál debe ser la interacción con el usuario, etc.

La reingeniería de software aplicada al sistema SISEFIN se empezó a realizar de la siguiente manera:

Primeramente se organizaron dos grupos de trabajo para el desarrollo y seguimiento de los sistemas: SISEFIN (Sistema de Seguimiento Financiero para el Control Interno del Ejercicio Presupuestal de la Gerencia de Inspección y Mantenimiento - con nuevos alcances -) y el PRESST (Sistema Presupuestal de la Subdirección de Servicios Técnicos - sistema que nace de la división del SISEFIN). Ambos sistemas enfocados a los presupuestos, abarcaban diferentes necesidades. Para su desarrollo intervinieron las siguientes dependencias de PEMEX:

SISEFIN	PRESST
Gerencia de Inspección y Mantenimiento	Subdirección de Servicios Técnicos compuesta por: <ul style="list-style-type: none"> • Unidad de Apoyo Administrativo • Gerencia de Desarrollo Tecnológico • Gerencia de Ingeniería y Construcción • Gerencia de Inspección y Mantenimiento • Gerencia de Servicios Técnicos • Gerencia de Servicios a Pozos e Instalaciones

Cada gerencia contempla las tres regiones de PEMEX, Norte, Sur, Marina Noreste y Suroeste.

Debido a la prioridad que tenía el PRESST por ser un sistema presupuestal ejecutivo a nivel subdirección, tuvo que atenderse primero en su análisis y diseño. Paralelamente el grupo de trabajo del sistema SISEFIN estuvo realizando la reingeniería y checando los avances que iba teniendo el PRESST hasta que se concluyera el análisis y diseño del mismo y los usuarios de la GIM pudieran dedicar tiempo al seguimiento del SISEFIN.

El proceso de reingeniería consistió en cuatro etapas:

PRIMERA ETAPA: Esta etapa abarca el conocimiento de la vieja aplicación. El Líder de Proyecto, que recientemente se le había asignado el sistema, tenía la responsabilidad de prepararse en conocimientos financieros y presupuestales para entender los objetivos y alcances del anterior sistema. Las actividades que se realizaron a la vieja aplicación fueron las siguientes:

- ✓ Se revisó y entendió a detalle toda la documentación referente al sistema, entre las que se encuentran:
 - Documentación del análisis y diseño
 - Notas Informativas
 - Documentación del Prototipo
 - Código del sistema hasta donde se tenía desarrollado
- ✓ Igualmente se tuvieron reuniones de trabajo con las personas que intervinieron en su desarrollo, y especialmente con el líder de proyecto que estuvo a cargo de su seguimiento.
- ✓ También se hicieron entrevistas con la gente indirecta pero que se encontraba inmersa en los presupuestos de PEMEX.
- ✓ Paralelamente se revisó a detalle las tablas, a través del apoyo de una herramienta CASE conocida como Erwin ERX Logic, y se aplicó la **ingeniería en reversa**, proporcionando un panorama general de como estaba estructurada la base de datos y la relación que preveía entre las tablas.
- ✓ Se imprimieron las pantallas que tenía el sistema para observar detenidamente el flujo que existía de acuerdo a sus módulos.
- ✓ Se identificó el estado de desarrollo en que quedó el sistema, calificando el grado de avance en cuestión a programación por cada módulo.
- ✓ Se verificó el funcionamiento y actualización del prototipo con respecto al sistema.

SEGUNDA ETAPA: Esta etapa parte de la terminación del análisis y diseño del sistema PRESST, en donde el grupo de trabajo del SISEFIN ya tiene conocimiento de las necesidades que va a cubrir el PRESST y a partir de este momento se hace una comparación para descartar los procesos semejantes que pudiera tener el sistema SISEFIN. Las actividades que se desarrollaron fueron las siguientes:

- ✓ La estructura de la base de datos del PRESST a la del SISEFIN era muy idéntica con respecto a las tablas de los catálogos institucionales del presupuesto, por lo que se creó una liga con la base de datos del PRESST y se eliminaron físicamente estas tablas de la base de datos del SISEFIN.
- ✓ Se deshabilitaron los módulos que realizaban procesos similares a los que ejecutaba el PRESST y por consiguiente las tablas que se vieran involucradas fueron eliminadas.
- ✓ Se identificaron que procesos eran propios del SISEFIN, es decir que no fueran considerados por el sistema PRESST y de estos se determinó el grado de avance en su programación.

A partir de este momento la reingeniería se había efectuado en dos pasos: analizando la aplicación existente y comparándola con el sistema PRESST. Todo esto con el propósito de

conocer el viejo sistema e identificar procesos propios del SISEFIN sin que otro sistema los contemple, para que sobre la base de los nuevos requerimientos poder determinar qué es lo que se debe eliminar, mantener, o mejorar.

TERCERA ETAPA: Una vez concluidas las actividades del análisis y diseño del sistema PRESST, los usuarios de la GIM querían definir los nuevos requerimientos del SISEFIN para evaluar los costos y beneficios que se generarían en caso de que se continuare con el proyecto. PEMEX solicitó al IMP una propuesta Técnico - Económica del SISEFIN aplicando la reingeniería de software de lo que se tiene del SISEFIN con lo que se pretendía conseguir. Para dar inicio a esta propuesta el IMP llevo a cabo un análisis preliminar del sistema bajo estudio.

2.15. EVALUACION DE LA SOLICITUD DEL PROYECTO

La propuesta de proyecto presentada por los usuarios o líderes de proyecto ante el comité de selección de proyectos es un elemento crítico para emprender el estudio de sistemas. Aunque el formato de dicha solicitud cambia de una compañía a otra, existe un acuerdo general sobre la clase de información que debe contener.

En la propuesta, el solicitante identifica dónde necesita la asistencia y proporciona los detalles. Llevándose una investigación preliminar para obtener mayor información con respecto al problema o situación, por lo que el solicitante debe proporcionar el nombre de las personas a las cuales acudir para obtener la información.

Cuando se formula una solicitud se comienza con el análisis preliminar de la situación actual.

2.15.1. INVESTIGACIÓN O ANÁLISIS PRELIMINAR

La elección de una estrategia de desarrollo es un aspecto secundario; lo importante es determinar si la solicitud merece o no la inversión de recursos en un proyecto de sistemas de información. Siendo necesario que los líderes de proyecto se sistemas lleven a cabo una investigación o análisis preliminar de cómo se trabaja actualmente. El estudio se realiza bajo la dirección del coordinador o solicitante del proyecto.

La finalidad del análisis preliminar es evaluar la solicitud de proyecto. No es un estudio de diseño ni tampoco incluye la recolección de detalles para describir el sistema de la empresa. Más bien, es la reunión de información que permite a los miembros del comité evaluar los méritos de la solicitud de proyecto y emitir un juicio, con conocimiento de causa, con respecto a la factibilidad del proyecto propuesto.

Los líderes de proyecto que trabajan en el análisis preliminar deben satisfacer los siguientes objetivos:

Analizar el entorno en que se desenvuelven los usuarios y la información. ¿ Qué es lo que se tiene y cómo se hace ?

- Entendimiento de la situación en que se trabaja.
- Comprensión de los procesos y funciones de cada actividad.
- Identificación de la información generada, volumen, frecuencia, lugares donde se utiliza (áreas operativas y ejecutivas) y el flujo que tiene.
- Investigación de las soluciones actuales y métodos empleados.

Analizar las necesidades y requerimientos de los usuarios. ¿ Qué es lo que quiere y cómo hacerlo ?

- Especificación de las necesidades y requerimientos del usuario.
- Determinación de los objetivos y alcances del sistema.
- Evaluación de beneficios que traerá el implantar el sistema.
- Realización del estudio de factibilidad.
 - Operacional: Actitud de los usuarios frente a la propuesta del sistema.
 - Técnica: Evaluación de los recursos técnicos usuales frente al nuevo sistema.
 - Económica: Estudio del costo generado por el desarrollo del sistema (hardware y software) y ganancias a futuro.
- Especificación de la metodología de trabajo para el desarrollo del sistema.
- Formulación de propuestas de hardware y software.
- Definición del Programa de actividades.

Para responder a estos puntos es necesario aplicar metodologías para la recopilación de información que se presentan en el siguiente capítulo “Análisis de Sistemas C/S”.

El análisis preliminar que se solicita para hacer la propuesta, debe realizarse de manera rápida y competente para que la información que emita el estudio sea sólida y que se mantenga en alto el interés prevalente. Mientras más esmerado sea el estudio, brindará a largo plazo un mayor beneficio en tiempo y dinero, tanto a la empresa como a los líderes de proyecto de sistemas.

2.15.2. PRUEBA DE FACTIBILIDAD DEL PROYECTO

Las investigaciones preliminares examinan la factibilidad del proyecto, la posibilidad de que el sistema sea de utilidad para la organización. Se estudian tres pruebas de factibilidad: operacional, técnica y financiera.

A) FACTIBILIDAD OPERACIONAL

La factibilidad operativa del proyecto que ha sido solicitado dependen de los recursos humanos que participan durante la operación del proyecto. Esto se refiere al pronóstico de si, una vez instalado, el sistema llegará a funcionar o a usarse.

Si los usuarios están casados virtualmente con el sistema actual y no le ven problema alguno, y no acuden al analista para perfeccionarlo o reemplazarlo por otro nuevo sistema, es muy probable que la resistencia al cambio será enorme. En estas circunstancias, son muy pocas las oportunidades de que un sistema nuevo llegue a ser operativo.

En este punto, la determinación de una factibilidad operativa requiere de una imaginación creativa del analista de sistemas, así como de una hábil capacidad de persuasión, para hacerle saber al usuario cómo dentro de las posibles tecnologías habrá una que satisfaga sus necesidades. El analista debe saber escuchar lo que el usuario desea y lo que parece que llegará a utilizar.

La prueba de factibilidad formula las siguientes preguntas:

- ¿Trabjará el sistema cuando esté terminado e instalado?
- ¿Existen barreras importantes para la implantación?
- ¿Existe apoyo suficiente para el proyecto por parte de la administración? ¿y por parte de los usuarios?
- ¿Los métodos que actualmente se emplean en la empresa son aceptados por los usuarios?
- ¿Los usuarios han participado en la investigación preliminar?
- ¿El sistema propuesto causará perjuicios?
- ¿El sistema reducirá la productividad de otras áreas?

Aspectos que al inicio parecen tener poca importancia pueden convertirse en grandes problemas después de la implantación.

B) FACTIBILIDAD TECNICA

El analista debe indagar si los recursos técnicos actuales pueden actualizarse o complementarse de tal manera que satisfagan la necesidad considerada. Entre los aspectos técnicos que es común que aparezcan durante la etapa de factibilidad de la investigación, se incluyen los siguientes:

- ¿Existe o se puede adquirir la tecnología necesaria para realizar lo que se pide?
- ¿El equipo propuesto tiene la capacidad técnica para soportar todos los datos requeridos para usar el nuevo sistema?
- ¿El sistema propuesto ofrecerá respuestas adecuadas a las peticiones sin importar el número y ubicación de los usuarios?
- Si se desarrolla el sistema, ¿puede crecer con facilidad?
- ¿Existen garantías técnicas de exactitud, confiabilidad, facilidad de acceso y seguridad de los datos?

Para contestar estas preguntas el analista primero debe inventariar el equipo de cómputo existente para descubrir con qué dispone; luego, debe estimar cargas de trabajo presentes y futuras para el sistema. Con el estudio del equipo, el analista puede determinar si será recomendable un nuevo equipo de cómputo, modificaciones al actual o el actual sin cambios.

El analista debe trabajar en conjunto con los usuarios para evaluar y determinar el equipo que sea requerido para una plataforma cliente/servidor. Si se requiere cambiar o actualizar el hardware, el proceso será considerar los tipos disponibles de equipo que parecieran ajustarse a las necesidades proyectadas. El analista puede orientar a los usuarios y a la directiva acerca de las ventajas y desventajas sobre el equipo nuevo. El equipo y software que propone el analista es su respuesta a las necesidades de información del usuario. Para la determinación del hardware y software se considera:

- Tamaño y capacidad del hardware y software, sobre la base de los requerimientos.
- Características de flexibilidad, amigabilidad, compatibilidad y confiabilidad.
- Medición y evaluación del hardware y software.
- Soporte y mantenimiento por parte del proveedor.
- Análisis de los factores financieros
- Contrato, licencia y garantía.

C) FACTIBILIDAD ECONOMICA

Un sistema que puede ser desarrollado desde el punto de vista técnico y que, además, será utilizado si se llega a instalar, debe ser una buena inversión para la organización. Los beneficios financieros deben igualar o exceder los costos. Las cuestiones económicas y financieras formuladas por los analistas durante la investigación preliminar tienen el propósito de estimar lo siguiente:

- El costo de llevar a cabo el desarrollo del sistema.
- El costo del hardware y software para la aplicación que se está considerando.
- Beneficios en la forma de reducción de costos al operar el nuevo sistema.
- El costo si nada sucede (es decir si el proyecto no se lleva a cabo).

Para ser considerada como factible, la propuesta debe pasar todas las pruebas. Por ejemplo, un sistema de registro de personal que sea factible desde el punto de vista financiero y operacionalmente atractivo, no es factible si la tecnología necesaria para su desarrollo aún no existe.

2.15.3. REINGENIERIA: ANÁLISIS CON RESPECTO A LA ANTERIOR APLICACIÓN

En caso de que exista un sistema y se requiera mejorarlo o actualizarlo a las necesidades actuales, el primer paso a considerar es el estudiar la vieja aplicación (véase en temas

anteriores, punto 2.14) para que posteriormente pueda ser analizada con respecto a los requerimientos de la nueva aplicación y determinar lo que podemos aprovechar, ahorrándonos esfuerzo y tiempo.

La reingeniería entre ambas aplicaciones será también preliminar, pues el conocimiento que se tiene al momento de la nueva aplicación es solo lo suficiente para evaluar la solicitud, pero no para empezar con el diseño y construcción del mismo. La reingeniería en esta etapa es analizar los procesos u objetivos de ambos sistemas, detectando que proceso sigue igual y reutilizarlo o cuál necesita ser reconstruido. Estas observaciones influirán mucho en los costos y beneficios de la solicitud del proyecto, así como el tiempo que se invertirá en su desarrollo.

Con el análisis preliminar del sistema SISEFIN se obtuvo un panorama de cual eran los alcances del nuevo sistema y conforme a los conocimientos de la vieja a ésta, se pudo determinar lo que se podía aprovechar y lo que se descartaría.

2.15.4. EVALUACIÓN DE LA FACTIBILIDAD

Aunque los líderes de proyecto de sistemas proponen un sistema que satisfaga los requerimientos de manejo de información, la decisión para continuar con la propuesta del sistema se basará en el análisis de los costos y los beneficios y no en los requerimientos de la información.

Los costos y los beneficios pueden ser tanto de naturaleza tangible como intangible. Ambos deben tomarse en cuenta en las propuestas de los sistemas.

Los **beneficios tangibles** son las ventajas económicas cuantificables que obtiene la organización a través del uso del sistema de información. Ejemplo tangibles serían el incremento en la velocidad de proceso, contar con cierta información que de otra manera sería inaccesible; la obtención de información con mayor puntualidad que en el pasado; aprovechar el mayor poder de cálculo de las computadoras y reducir el tiempo requerido por los empleados para concluir una tarea específica.

Los **beneficios intangibles** incluyen: la mejora del proceso de toma de decisiones; el incremento de precisión; el llegar a ser más competitivo en los servicios al cliente; el mejoramiento de la imagen del negocio y el incremento de la satisfacción de los empleados al eliminar tareas de naturaleza tediosa.

Los **costos tangibles** son aquellos que pueden proyectar con precisión el analista y el personal de contabilidad. Dentro de estos costos se incluyen: el costo del equipo, el costo de los recursos de tiempo del analista de sistemas, el costo del tiempo de programación y de otros salarios de personal.

Los **costos intangibles** son el costo de perder una ubicación competitiva, perder puntos por no ser el primero en innovar, deterioro de la imagen de la compañía.

Aunque la propuesta considera varias alternativas, en sí solo recomendará una de ellas. Esto significa que se tendrá que hacer un análisis comparativo acerca de cuál opción tiene mayor sentido económico aún antes de que se redacte la propuesta. Este análisis deben incluirse para dar mayor soporte a la alternativa que finalmente se recomiende.

Las alternativas se deben comparar con base en los requerimientos de información que se identifiquen, los costos y beneficios de cada opción seleccionada. Es necesario mantener un criterio abierto y objetivo al comparar las alternativas para que los usuarios del sistema puedan decidir cuál se apeg a las exigencias de la empresa.

Con la evaluación, el analista podrá examinar las alternativas disponibles y finalmente ofrecer una recomendación bien fundamentada en la propuesta técnico - económica.

2.16. PROPUESTA TECNICO - ECONOMICA

La decisión de los proyectos la toma la directiva, en base a un documento que presenta el líder de proyecto llamado **Propuesta Técnico - Económica**. La propuesta del sistema es un resumen de lo que el líder ha aprendido acerca de la empresa y lo que esta requiere para mejorar su desempeño. En la propuesta se identifican y pronostican los costos y beneficios.

En esta propuesta se muestran los resultados obtenidos en el análisis preliminar. También esboza las opciones desarrolladas por los analistas y presenta sus recomendaciones a la organización para que determine si se continúa con el nuevo sistema.

La propuesta técnico - económica contiene los siguientes puntos:

- A) Carta de presentación
- B) Presentación inicial.
- C) Índice
- D) Resumen Ejecutivo
- E) Introducción.
- F) Descripción del estudio
- G) Resultado del estudio de las necesidades.
- H) Objetivos.
- I) Alcances y Beneficios
- J) Evaluación de factibilidad técnica y operativa.
- K) Propuestas de estrategias para satisfacer los requerimientos
- L) Recomendaciones.
- M) Metodología para el desarrollo del sistema
- N) Estudio de Factibilidad Económica
- O) Soporte Técnico, Garantía y Derechos de Autor
- P) Programa de actividades.

Q) Conclusiones

R) Anexos

A) CARTA DE PRESENTACION

La propuesta del sistema debe ir acompañada de una carta de presentación para el comité de evaluación. Debe incluir aquellos que participaron en el estudio y resumir los objetivos del mismo. La carta de presentación también puede llevar la fecha y el lugar para la presentación oral de la propuesta del sistema. La carta debe ser concisa ya que su función es poner al tanto a la dirección sobre la propuesta del sistema.

B) PRESENTACION INICIAL.

Se especifica esencialmente el nombre de la organización que hizo el análisis preliminar para la propuesta (puede incluirse los nombres de los miembros que realizaron el análisis preliminar), el nombre de la empresa que solicitó el proyecto, el nombre del sistema que se va a desarrollar y el título del presente documento y la fecha en que se presenta la propuesta. No importa el orden en que se coloquen estos datos, pero sí que la información contenida en la presentación sea muy clara.

C) INDICE

La clave para elaborar un índice de contenido, es prepararlo durante el desarrollo de la propuesta, cuando el analista ya tenga identificado los títulos y subtítulos. Mientras más conciso sea el índice llegará a ser más eficaz.

D) RESUMEN EJECUTIVO

Este resumen entera de quién, cuál, cuándo, dónde, qué y el cómo de la propuesta. También debe incluir las recomendaciones del analista del sistema y las acciones deseadas de la gerencia, ya que algunas personas sólo llegarán a leer tal resumen. El resumen ejecutivo es difícil de escribir, ya que su contenido debe ser conciso. Debe escribirse sólo cuando se haya completado la propuesta, pues en tal momento cuando el analista tendrá una visión completa y adecuada de lo que se propone.

E) INTRODUCCION

Se exponen de manera breve lo que refleja el contenido de la propuesta, se da las causas que propiciaron la solicitud del proyecto y el objetivo de la propuesta técnico - económica.

F) DESCRIPCION DEL ESTUDIO

Esta sección proporciona información acerca de todos los métodos utilizados en el estudio y quién o qué fue sujeto del estudio. Se incluye las fuentes de información utilizadas como la lista de personas entrevistadas y observadas y las formas que fueron analizadas.

G) RESULTADO DEL ESTUDIO DE LAS NECESIDADES.

Es un resumen de los problemas que presentan actualmente en la forma en que están llevando sus actividades.

H) OBJETIVOS.

Son los objetivos generales y particulares que cubrirá el sistema.

I) ALCANCES Y BENEFICIOS

Se definen los límites del sistema, los procesos que contemplará y las necesidades que cubrirá. También se definirán los beneficios cuando sea operado.

J) EVALUACIÓN DE FACTIBILIDAD TÉCNICA Y OPERATIVA.

Son los resultados obtenidos con respecto a la actitud de los usuarios y la evaluación de los recursos técnicos actuales.

K) PROPUESTAS DE ESTRATEGIAS PARA SATISFACER LOS REQUERIMIENTOS

Se especifican las características de las diferentes alternativas en plataformas computacionales que podrían implantarse para el desarrollo y operación del sistema, así como un panorama de que es lo que contemplará el sistema. El analista puede presentar dos o tres soluciones alternativas que se dirigen de manera directa a los problemas señalados anteriormente. Cada una de las alternativas se deben de analizar de manera separada.

L) RECOMENDACIONES.

Una vez que el grupo de análisis de sistemas ha ponderado las alternativas, tendrá una opinión definida acerca de cuál solución sería la más adecuada. Se explican los motivos que apoyan las recomendaciones.

M) METODOLOGIA PARA EL DESARROLLO DEL SISTEMA

Son las etapas que se deben llevar a cabo para desarrollar satisfactoriamente un sistemas de información que cumpla con calidad las perspectivas de la organización.

N) ESTUDIO DE FACTIBILIDAD ECONÓMICA

Son los costos y beneficios financieros que se generarán con el desarrollo e implantación del sistema. Se estima el costo de la cada alternativa propuesta, el costo del equipo, el costo de mano de obra, determinando el número de personas participantes dentro del proyecto y el tiempo de duración del mismo.

O) SOPORTE TECNICO, GARANTÍA Y DERECHOS DE AUTOR

Se estipulan los términos en que se otorgará la garantía, los derechos de autor y el soporte técnico una vez que quede finiquitado el sistema.

P) PROGRAMA DE ACTIVIDADES.

Se muestra mediante un diagrama de barras el calendario de actividades programado por cada alternativa para el desarrollo del sistema, especificando el avance técnico y financiero que tiene cada periodo.

Q) CONCLUSIONES.

Las conclusiones exponen de manera resumida las razones y ventajas del proyecto C/S, haciendo hincapié en puntos importantes de las propuestas, y reflejando de manera entusiasta la confianza y profesionalismo del equipo de trabajo encargado a cumplir las expectativas del proyecto.

R) ANEXOS

Esta última parte de la propuesta de sistemas incluye cualquier información que el analista considere de interés para individuos específicos, pero sin ser indispensable para la comprensión de lo que se propone. Los anexos pueden incluir toda la correspondencia pertinente, gráficas, etc.

Es importante tomar en cuenta que el tamaño de la propuesta es directamente proporcional al tamaño de las modificaciones o del sistema que se propone. El estilo de redacción deber ser claro y comprensible. Es apropiado y recomendable, que el analista utilice ilustraciones, diagramas, tablas, figuras y gráficas. Y no necesariamente debe cubrir todos los puntos mencionados anteriormente.

Durante la propuesta técnico - económica del sistema SISEFIN, el coordinador del proyecto por parte de PEMEX, el Ing. Sánchez (GIM) visito las instalaciones del IMP (ubicado en el Distrito Federal) para empezar con el estudio del análisis preliminar, posteriormente el IMP también fue a las instalaciones de PEMEX (ubicadas en Villahermosa, Tabasco) para reunirse con los usuarios operativos y checar la infraestructura y el medio ambiente de trabajo. Todas estas actividades tuvieron un transcurso de 10 días para que al termino de 12 días ya se tenía concluida la propuesta Técnico - Económica.

En el anexo 1 se agrega lo que fue la propuesta técnico - económica del proyecto SISEFIN, en algunos datos se tuvo el cuidado de omitirlos por ser información confidencial.

2.16.1. PRESENTACIÓN DE LA PROPUESTA

Para la presentación oral de la propuesta es necesario saber quiénes van a conformar la audiencia. Si el auditorio es homogéneo, la tarea del ponente es más fácil, ya que sólo debe satisfacer unos cuantos puntos de vista. Aunque la mayoría de los líderes de proyecto que presentan una propuesta de sistemas se encuentra con un grupo heterogéneo.

En la presente tabla se determinan dos tipos de audiencias y cuales son los puntos que se deben recalcar en la presentación de la propuesta:

AUDIENCIAS DE EJECUTIVOS	<ul style="list-style-type: none"> • Se componen de la alta dirección y la gerencia. • Los ejecutivos pueden o no tener experiencia en el área bajo discusión. • Durante la presentación el analista no
-------------------------------------	--

<p>AUDIENCIAS DE EJECUTIVOS</p>	<p>invertirá tiempo en aspectos técnicos.</p> <ul style="list-style-type: none"> • La información costo/beneficio es pertinente e interesante en los ejecutivos. • El analista debe ser conciso y directo, porque el tiempo es un recurso muy valioso para el tomador de decisiones. • El tiempo para preguntas debe ser lo suficiente.
<p>AUDIENCIAS DE USUARIOS PRINCIPALES</p>	<ul style="list-style-type: none"> • Se componen de personas que están directamente relacionadas con el sistema bajo estudio. • Durante la presentación el analista discutirá los cambios que ocurrirán en su trabajo, así como los beneficios del sistema propuesto. • El análisis de costos y beneficios se expondrá pero no será el foco principal de la plática • Ayuda visual que permita concebir el uso del sistema

Para estas dos audiencias, también se expondrá de manera general los demás puntos de la propuesta escrita. Los ejemplos, las ilustraciones, referencias de testimonios, figuras, tablas y gráficas son apoyos visuales que ayudan a la comprensión y dan calidad a la presentación.

La presentación puede ser expuesta mediante láminas, acetatos, diapositivas, vídeo o por computadora. La presentación debe ser de lo más formal si en la reunión esta el director o consejo de administración. Si los usuarios, más que los ejecutivos integran la audiencia, tal vez sea más apropiado una presentación tipo taller. Asimismo se tiene que estimar el tiempo necesario para la presentación de forma tal que éste no sea excesivo.

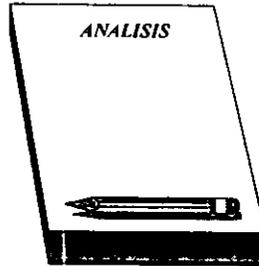
El analista cuando presenta una propuesta tiene el objetivo de dar a conocer el desarrollo del nuevo sistema y sobre todo convencer a la audiencia sobre la alternativa que recomienda. Para dar una buena presentación, el orador debe practicar lo que va a exponer, porque mientras más sepa de su tema más fácil será para el analista exponerlo. Igualmente deberá saber enfrentar cualquier contingencia durante su exposición, tal como asistentes críticos, sitios desfavorables, o tal vez sobreponerse a fallas de equipo.

2.16.2. MANEJO DE PROYECTOS NO FACTIBLES

No todos los proyectos que se someten a una evaluación y revisión son aceptados. Las solicitudes que no pasan las pruebas de factibilidad ya no reciben ninguna atención, a menos que se trabaje sobre ellas y sean presentadas otra vez ante el comité de selección como nuevas propuestas

CAPITULO III

ANÁLISIS DE SISTEMAS CLIENTE/SERVIDOR



3.1. ANÁLISIS ESTRUCTURADO

3.1.1. DEFINICIÓN

En el desarrollo del sistema SISEFIN en arquitectura Cliente/Servidor, el análisis se llevo a cabo de manera estructurada.

El análisis estructurado es un método para el análisis de sistemas, que conduce al desarrollo de especificaciones para sistemas nuevos o para efectuar modificaciones a los ya existentes. El análisis estructurado permite al analista conocer un sistema o proceso, en una forma lógica y manejable al mismo tiempo que proporciona la base para asegurar que no se omita ningún detalle pertinente.^[21]

El objetivo que persigue el análisis estructurado es **organizar las tareas** asociadas con la determinación de requerimientos para obtener la comprensión completa y exacta de una situación dada. En el análisis estructurado, la palabra **estructurar** significa:

- 1) Estructurar el proceso de determinación de los requerimientos comenzando con la documentación del sistema existente.
- 2) Intentar incluir todos los detalles relevantes que describen la situación actual.
- 3) Identificar los requerimientos e incluir las mejores soluciones y estrategias para las oportunidades de desarrollo de sistemas.

[21] [22] Senn, James.- *Análisis y Diseño de Sistemas de Información.- Traduc. Urbina Medal Edmundo Gerardo y otros.- 2ª Edición.- Editorial McGRAW-HILL.- México.- 1994 - pp. 176 - 177.*

3.1.2. COMPONENTES

El análisis estructurado hace uso de los siguientes componentes.^[22]

- **Símbolos gráficos:** Iconos para identificar y describir los componentes de un sistema.
- **Diccionario de datos:** Descripciones de todos los datos utilizados en el sistema.
- **Descripciones de procesos y procedimientos:** Declaraciones que permiten a los analistas describir actividades importantes que forman parte del sistema.
- **Reglas:** Estándares para describir y documentar el sistema en forma correcta y completa.

3.1.3. ANÁLISIS DE LA SITUACIÓN ACTUAL

Con la investigación preliminar realizada en la propuesta técnico - económica, se adquirió el conocimiento de la forma en que se trabaja actualmente. Pero este estudio sólo fue lo suficiente para evaluar la solicitud del proyecto, y ser autorizado por el comité responsable. La finalidad del análisis de la situación actual es retomar la investigación preliminar y estudiar más a detalle el sistema actual de la empresa. En caso de que no hubiera un análisis preliminar, el estudio partirá desde el principio, definiendo objetivos, alcances, reingeniería de software, análisis de la situación actual, etc.

En esta actividad se recopilará la información necesaria sobre el procedimiento actual, para que sirva como base a los esfuerzos de diseño de la nueva aplicación y permita evaluarlo para su ajuste a las nuevas capacidades.

La investigación describe su estructura organizacional del área (Organigrama), especifica paso a paso los procesos que intervienen en cada actividad (Modelo de Procesos Actual), esquematizando el flujo de información (Diagrama de Flujo de Datos Actual) e identificando controles, responsabilidades y actitudes para la realización de las tareas.

Para el estudio de la situación actual existen métodos de recopilación de información, los principales son:

RECOPIACION DE INFORMACION

Muestreo

El muestreo es el proceso por el cual se seleccionan de manera sistemática elementos representativos de una población. Cuando estos elementos se examinan a detalle, tal análisis revelará la información extensiva acerca de la población.^[23]

[22] Kendall, Kenneth y Kendall, Julie.- *Análisis y Diseño de Sistemas.* - Traduc. Hernández López Héctor y otros. - 1ª Edición.- Editorial Prentice-Hall Hispanoamericana.- México.- 1994 - pp. 103.

Existen numerosos motivos por los cuales un analista tiene que seleccionar muestras representativas de datos para examinar o personajes que entrevistar, interrogar u observar. Dentro de estos motivos están la recopilación ágil de datos, reducción de tiempos y costos, etc..

Técnicas para encontrar hechos

Los analistas de sistemas utilizan métodos específicos, con el objeto de reunir datos relacionados con los requerimientos. Entre éstos se incluyen la entrevista, el cuestionario, la revisión de documentación y la observación. En general, los analistas emplean estas técnicas para estar seguros de llevar a cabo una investigación amplia y exacta.

Entrevistas: Las entrevistas dan a los analistas oportunidades para reunir información de las personas debido a sus conocimientos del sistema que está bajo estudio. A menudo este método es la mejor fuente de información cualitativa (opiniones, políticas, descripciones subjetivas de actividades y problemas).

Este método puede ser de especial utilidad para reunir información de las personas que no se comunican por escrito en forma adecuada o que no disponen de tiempo para llenar cuestionarios. Las entrevistas permiten al analista descubrir áreas mal comprendidas e incluso indicadores de resistencia hacia el sistema propuesto.

Cuestionarios. El uso de cuestionario permite a los analistas reunir información proveniente de un grupo de personas. El empleo de formatos estandarizados para las preguntas puede proporcionar datos más confiables que otras técnicas; por otra parte, su amplia distribución asegura el anonimato de los encuestados, situación que pueden conducir a respuestas más honestas. Sin embargo, este método no permite al analista observar las expresiones o reacciones de los encuestados. Asimismo, la respuesta puede ser limitada ya que es posible que no tenga importancia para los encuestados llenar el cuestionario.

Revisión de documentación. Varios tipos de documentos y reportes pueden proporcionar al analista información valiosa con respecto a las organizaciones y sus operaciones. Al revisar los documentos, el analista examina la información asentada en ellos relacionada con el sistema y los usuarios, y sirve de base para comparar las operaciones actuales.

Observación. La observación permite al analista ganar información que no se puede obtener por otras técnicas. Por medio de la observación el analista obtiene información de primera mano sobre la forma en que se efectúan las actividades, la forma en que se manejan los documentos y se llevan a cabo los procesos y, por otro, si se siguen todos los pasos especificados. También el analista debe observar el comportamiento de las personas y el ambiente físico en que se desenvuelven.

El obtener información quizá no sea el aspecto más interesante de un estudio de sistema. Dar la consideración apropiada para desarrollar entrevistas, formular cuestionarios, enfocar la observación y conducir el estudio, es trabajo desafiante. Obtener información ofrece al analista la oportunidad de descubrir cómo trabajan las cosas en la realidad y cómo puede mejorarlas.

3.1.4. ANÁLISIS DE LOS REQUERIMIENTOS

El objetivo del análisis de sistemas C/S y de cualquier sistema de información es comprender situaciones, no resolver problemas. Por lo tanto, los buenos analistas hacen hincapié en la investigación y el cuestionamiento para conocer cómo opera el sistema e identificar los requerimientos que tienen los usuarios. Solo después de comprender en su totalidad el sistema, los analistas están en posición de analizarlo y generar recomendaciones para el diseño de sistemas.

DETERMINACION DE LOS REQUERIMIENTOS

La determinación de requerimientos es el estudio de un sistema para conocer cómo trabaja y dónde es necesario efectuar mejoras. Es el proceso por el cual los analistas obtienen el conocimiento relacionado con la organización y lo aplican para seleccionar la tecnología correcta para una aplicación en particular.

Un requerimiento es una característica que determina la forma para capturar o procesar datos, producir información, controlar una actividad de la empresa o brindar soporte a la gerencia.

ACTIVIDADES DE LA DETERMINACION DE REQUERIMIENTOS

Es útil ver la determinación de requerimientos a través de tres grandes actividades: anticipación, investigación y especificación de requerimientos.

ACTIVIDAD	DESCRIPCION
Anticipación de requerimientos	Es Prever las características que tendrá el nuevo sistema C/S con base en la experiencia que se tiene en un área o ambiente similar. Esto puede llevar al analista a investigar áreas y aspectos que de otra forma no sería tomados en cuenta. Tener las bases necesarias para saber qué preguntar o qué aspectos investigar puede ser beneficio sustancial para la organización. Por otra parte, si se introducen atajos al conducir la investigación, entonces es muy probable que la anticipación de requerimientos se convierta en un problema. Por tanto, siempre deben darse lineamientos para estructurar una investigación alrededor de cuestiones básicas con la finalidad de evitar consecuencias indeseables de la anticipación de requerimientos.
Investigación de requerimientos	Estudio y documentación de la situación actual utilizando para ello de varias herramientas y habilidades

ACTIVIDAD	DESCRIPCION
Especificación de requerimientos	Se analizan los datos obtenidos durante la recopilación, que describen la forma en que se opera actualmente, con el fin de determinar qué tan bueno es su desempeño, qué requerimientos se deben satisfacer y las estrategias para alcanzarlos, describiendo así las características del nuevo el sistema C/S.

REQUERIMIENTOS

En analista debe dar respuesta a las siguientes preguntas para desarrollar un perfil completo del sistema bajo investigación.

REQUERIMIENTOS	PREGUNTAS
Frecuencia y volumen del proceso	<p>¿Cuál es el volumen de actividades que se presentan?</p> <p>¿Con qué frecuencia ocurren las actividades?</p> <p>¿Ocurren las actividades de acuerdo con un ciclo?</p>
Identificación de controles	<p>¿Qué áreas necesitan un control específico?</p> <p>¿Cuáles son los métodos de control utilizados?</p> <p>¿Qué criterios se emplean para medir y evaluar el desempeño?</p> <p>¿Qué métodos se emplean para detectar lagunas en los controles?</p> <p>¿Se toman precauciones específicas de seguridad para protección contra una actividad impropia?</p>
Comprensión del proceso	<p>¿Cuál es la finalidad de esta actividad dentro de la empresa?</p> <p>¿Qué pasos se siguen para llevarla a cabo?</p> <p>¿Dónde se realizan estos pasos?</p> <p>¿Quiénes los realizan?</p> <p>¿Cuánto tiempo tardan en efectuarlos?</p> <p>¿Con cuánta frecuencia lo hacen?</p> <p>¿Quiénes emplean la información resultante?</p> <p>¿Qué procesos, pasos o funciones constituyen esta actividad?</p> <p>¿Qué es lo que da inicio a la actividad? ¿Con qué propósito?</p> <p>¿Cuánto tiempo tarda cada actividad? ¿Qué factores intervienen en la duración de la actividad?</p> <p>¿Qué retrasos ocurren o pueden ocurrir?</p> <p>¿Cómo interactúan los elementos entre sí?</p> <p>¿Cuál es el costo de operación de la forma en que están operando actualmente?</p> <p>¿Se satisfacen los objetivos de la gerencia?</p>

REQUERIMIENTOS	PREGUNTAS
Identificación de datos empleados e información generada	<p>¿Qué datos entran y cuál es su origen?</p> <p>¿En que forma se reciben los datos? y ¿En qué forma son almacenados?</p> <p>¿Quiénes utilizan la información generada? ¿Con qué finalidad la utilizan?</p> <p>¿Qué es lo que no se utiliza?</p> <p>¿Qué datos faltan con mayor frecuencia?</p> <p>¿Qué diagramas u otros datos se utilizan?</p> <p>¿Cómo están codificados o abreviados los datos o actividades?</p> <p>¿Qué información se utiliza para tomar decisiones?</p> <p>¿Qué datos se originan en fuentes externas a la organización?</p> <p>¿Cómo se deben procesar los datos para producir la información necesaria?</p> <p>¿Cómo debe presentarse la información?</p>
Otros	<p>¿Quiénes son las personas clave en el sistema? ¿Por qué son importantes?</p> <p>¿Qué obstáculos o influencias de tipo político afectan la eficiencia del sistema?</p>

Cuando los analistas estudian sistemas para un departamento también deben evaluar las implicaciones para las demás áreas que se vean involucradas con el sistema bajo investigación. Los analistas deben tener conocimiento de todo requerimiento en cualquier otra parte de la organización que dependa sobre los procesos que de deseen automatizar.

3.2. PANORAMA DE LA REINGENIERIA DE PROCESOS

La filosofía organizacional que busca que las empresas hagan un alto, y cambien su forma de operar para así poder resistir y competir en el entorno actual, se le conoce como: *Reingeniería de Procesos de Negocios*.

Las empresas utilizan estrategias o métodos (por ejemplo: automatización, simplificación o racionalización de los procesos, entre otros) para mejorar su desempeño, pero a veces no cumplen con los requerimientos impuestos actualmente por los mercados; siendo necesario romper con los viejos paradigmas sobre los cuales se trabaja, cambiar los principios, el fondo y la forma de las corporaciones. Es decir, *reinventar a las empresas*.

Hoy en día, la mayoría de las empresas se encuentran construidas sobre principios que resultan obsoletos porque no consideran el constante cambio, la incertidumbre y la globalización de los mercados en los cuales se desenvuelven. Adicionalmente no contemplan un punto vital: “ El uso adecuado de las tecnología de información para obtener ventajas competitivas “.

La reingeniería surge como una filosofía formal a finales de la década de los ochenta cuando Dr. Michael Hammer y James Champy hicieron un análisis de varias compañías que habían mejorado de manera singular su rendimiento, como resultado de cambiar radicalmente la manera en la que operaban. En este estudio se descubrió que muchas actividades realizadas por los empleados poco o nada tenían que ver con los objetivos de la empresa.

Estas compañías no cambiaron el giro de su negocio, simplemente modificaron total o parcialmente sus procesos operativos y estratégicos, al mismo tiempo desarrollaron técnicas que les permitieron seguir adelante en un entorno cada vez más cambiante y competitivo. Se olvidaron de aquellas técnicas funcionales que bloqueaban la visibilidad necesaria para captar en forma más clara la esencia del negocio, y se retornó la importancia de los procesos, modificando el carácter y participación de los elementos de la empresa.

3.2.1. DEFINICIÓN

Existen muchas corrientes del pensamiento que brindan conceptos de innovación, creatividad y estrategia a la Reingeniería. Sin embargo, la siguiente definición es una de las propuestas en los últimos años.

Es la revisión fundamental y el rediseño del sistema organizacional total (procesos de negocios, estructuras organizacionales, sistemas administrativos y de medición, cultura empresarial y el uso de tecnologías) con el objeto de lograr mejoras espectaculares en medidas críticas de desempeño como son los costos, calidad, servicio y rapidez.^[24]

En la anterior definición nos muestra que en realidad la reingeniería es el rediseño, replanteamiento de todos los parámetros y fuerzas que convergen en una organización.

La reingeniería no es cuestión de hacer mejoras marginales o incrementales sino de dar saltos gigantescos en rendimiento. Si una empresa tiene costos altos en un 10%, si su calidad es el 10% muy baja, esa compañía no necesita reingeniería. Con métodos convencionales, estableciendo programas incrementales de calidad se puede sacar a una empresa de un retraso del 10%. Se debe apelar la reingeniería únicamente cuando existan empresas que se encuentren en grandes dificultades y no tienen más remedio; las compañías que todavía no se encuentran en dificultades, pero cuya administración tiene la previsión de detectar que se avecinan problemas. o las compañías que no tienen dificultades visibles ni ahora ni en el horizonte pero su administración tiene aspiraciones y energía.

[24] Hammer, Michael y Champy, James. - *Reingeniería*. - Traduc. Nannetti Cárdenaz José. - Editorial Norma. - Colombia. - 1994.

3.2.2. PROCESOS DE NEGOCIOS

En las organizaciones tradicionales operan con el modelo de división y especialización del trabajo propuesto por Adam Smith en el siglo XVII. Este principio se basa en que la productividad se logra a través de la división del trabajo, en la cual se piensa que a medida que los trabajadores estén más especializados en una tarea, el tiempo que utilizarán para desempeñarla será cada vez menor, logrando con esta división producir mayor cantidad de artículos en el menor tiempo posible.

Esta manera de operar genera que se tenga una serie de actividades fragmentadas, en donde el control y supervisión se vuelven indispensables. Por lo general las tareas de supervisión y control no agregan ningún valor al servicio o producto, adicionalmente demandan una gran cantidad de recursos, ya que se vuelven el "pegamento" por medio del cual se unen los pedazos fragmentados artificialmente. Con esta división, en la mayoría de los casos, se pierde flexibilidad, teniendo estructuras muy rígidas y esto se refleja en la imposibilidad de las empresas para responder rápidamente a los cambios.

En la Reingeniería de Procesos, uno de los puntos clave es ver en las empresas los procesos que dan valor agregado. Un *proceso* es una serie de actividades vinculadas que agregan valor y crean un producto que sea más útil y efectivo para el receptor. Un *proceso de negocios* es un conjunto de actividades que reciben uno o más insumos y crea un producto de valor. Es un conjunto estructurado y medible de actividades, diseñado para producir un resultado.

En general un proceso:

- 1) Involucra un conjunto de actividades, realizadas en tiempos y lugares específicos, con un principio, un fin y con entradas y salidas bien definidas.
- 2) Implica un énfasis en el cómo se hace el trabajo.

La reingeniería indica que las tareas deben organizarse en proceso, es decir conjuntos de tareas o actividades bien definidas en las cuales exista un responsable y se trabaje en equipo con el menor número posible de personas, con lo que se busca eliminar la fragmentación y reducir el control generado por la excesiva división del trabajo, haciendo uso de controles solamente hasta donde se justifica económicamente.

Por lo regular los proyectos para elevar la calidad y para el cambio organizacional han sido mucho más utilizados. Sin embargo, cuando no se ha considerado la mejora del proceso, dichos proyectos, han sido menos eficaces de lo que hubieran podido ser.

La reingeniería busca avances decisivos, no mejorando los procesos existentes sino descartándolos por completo o cambiándolos por otros enteramente nuevos. La reingeniería es volver a empezar, con una hoja de papel en mano. La reingeniería es un nuevo comienzo.

Adicionalmente la reingeniería indica que se debe procurar en lo posible, que las actividades de un proceso, así como los procesos, se realicen en forma paralela. Esto generará una significativa reducción en los tiempos, y por lo tanto en la respuesta a los cambios de la empresa.

Para visualizar y aplicar la reingeniería a las operaciones. Se enfatiza en la elaboración de los diagramas de la actividad y de relaciones formando lo que se conoce como *modelo de proceso*.

El *Modelo de Proceso* puede definirse como una representación de la operación de la compañía o de una parte específica. Por lo común, es una descripción gráfica de la estructura y actividades de la operación que muestra las relaciones entre las etapas de trabajo y su secuencia. En conjunto, estas representaciones reflejan el flujo de trabajo.

3.2.3. CULTURA ORGANIZACIONAL

En la mayoría de las veces cuando se intentan muchos cambios, pero prevalecen las mismas condiciones, con frecuencia se debe a la relación con un problema cultural. Al rediseñar los procesos de una empresa es lógico que se presentan estos conflictos, como son:

Incertidumbre acerca de las causas y efectos del cambio. Los empleados suelen rendirse al cambio por preocuparles la manera en que su trabajo y sus vidas se verán afectados por él. Aún cuando sientan una gran insatisfacción con su trabajo actual, les preocupa que las cosas empeoren al rediseñarse los procesos.

Renuencia a perder los beneficios actuales. Para algunos individuos el costo del cambio es mayor que las ventajas del mismo. Por ejemplo, pérdida de poder, prestigio, salario, reconocimiento, calidad del trabajo y otras. La *pérdida de poder* es una de las principales fuentes de resistencia de cambio. Antes de rediseñar los procesos de una organización existen grupos de poder claramente identificados que tienen influencia en la toma de decisiones. Después de la reingeniería, el poder que se encontraba en unas cuantas manos se dispersa entre los miembros de la organización, esto genera que las personas que gozaban de éste se resistan a perderlo.

Existen diferentes técnicas que se pueden aplicar para contrarrestar la resistencia al cambio, entre las que se encuentran:

- Campañas de educación audiovisual y comunicación oportuna.
- Participación de los directamente afectados en el rediseño de los procesos.
- Facilitar el cambio y dar apoyo a quienes interviene en el cambio.
- Negociación con las personas que se oponen al cambio para lograr su aceptación.

La reingeniería busca que los empleados trabajen en equipos, asuman sus propias responsabilidades y sean capaces de tomar decisiones, es decir, tener empresas con un alto grado de poder de decidir y actuar.

Pero la tarea no es sencilla, se debe capacitar a los empleados, cambiar la manera de recompensar el trabajo orientándose no sólo a las recompensas monetarias individuales, sino también a las recompensas psicológicas y de trabajo en equipo, dar los empleados su lugar y que se sientan parte de la empresa.

3.2.4. TECNOLOGÍA DE INFORMACIÓN

Estudios especializados revelan que en la década de los ochenta las empresas americanas invirtieron un trillón de dólares en tecnologías de información, sin embargo los resultados obtenidos no fueron los esperados. Y la razón fue que se buscó automatizar los procesos, por medio del uso de la tecnología, pero no innovarlos. En este contexto se podría considerar a la reingeniería de procesos como un enfoque que utiliza a la tecnología como: "Un elemento estratégico para la modificación, rediseño o innovación de los procesos"

La informática debe ser parte de cualquier esfuerzo de reingeniería, es uno de los habilitadores fundamentales en el cambio. Para aplicar la reingeniería se requiere que el pensamiento tradicionalmente deductivo se convierta a pensamiento inductivo. No se debe confundir el uso de la informática con una simple automatización, es necesario explotar sus capacidades para alcanzar metas completamente nuevas.

La mayoría de los ejecutivos y los gerentes saben pensar en forma deductiva. Es decir, saben definir muy bien un problema y luego buscar y evaluar sus diversas soluciones. Pero para aplicar la informática a la reingeniería de negocios es necesario pensar en forma inductiva: La capacidad de reconocer primero una solución poderosa y enseguida buscar los problemas que ella podría resolver, problemas que la compañía probablemente ni sabe que existen.

El error fundamental que muchas compañías cometen al pensar en tecnología es verla a través del lente de sus procesos existentes. Se preguntan: ¿Cómo podemos usar estas nuevas capacidades tecnológicas para realizar o dinamizar o mejorar lo que ya estamos haciendo? por el contrario, deberían preguntarse: ¿Cómo podemos aprovechar la tecnología para hacer cosas que no estamos haciendo?. La reingeniería, a diferencia de la automatización es innovación. Es explotar las más nuevas capacidades de la tecnología para alcanzar metas enteramente nuevas. Uno de los aspectos más difíciles de la reingeniería es reconocer las nuevas capacidades no familiares de la tecnología en lugar de la familiares.

La orientación deductiva, no sólo desconocerá lo realmente importante de la tecnología, sino también se puede caer en la selección y aplicación de tecnologías que no den valor agregado a la organización. Los potenciales de la tecnología deben ser explotados para cambiar los procesos de una compañía y hacer que ésta se adelante a la competencia, pero no de manera aislada o temporal, sino permanentemente, es decir, aprender a reconocer e

incorporar en la organización los nuevos cambios tecnológicos. El desarrollo y explotación de la tecnología es una ventaja estratégica y competitiva de cualquier empresa.

El esfuerzo debe ser permanente porque lleva tiempo desarrollar e implementar una tecnología, se debe estudiar, entender su significado, conceptualizar sus usos potenciales y planificar su ejecución. Si bien esta orientación a la tecnología es importante, no hay que tomarla como lo esencial o como el elemento único de la reingeniería de procesos.

A pesar del papel destacado de la informática en la reingeniería, ya debe estar bien claro que reingeniería no es lo mismo que automatización. Tampoco se debe confundir la reingeniería de procesos con la llamada reingeniería de software, que significa reconstruir sistema obsoletos de información con tecnologías más modernas.

La tecnología es el habilitador fundamental para el cambio, y las organizaciones deben enfocarse en ésta para poder mejorar.

En resumen, la tecnología permitirá rediseñar los procesos actuales, mediante nuevas y más modernas técnicas, es decir, la tecnología puede ser un elemento clave durante el replantamiento de los procesos, ya que ésta puede mostrar nuevas y claras posibilidades de mejorar procesos mediante un enfoque inductivo.

3.2.5. HERRAMIENTAS PARA DOCUMENTAR PROCESOS Y DECISIONES

Una herramienta es cualquier dispositivo, objeto u operación utilizada para ejecutar una tarea específica. Las herramientas ayudan al analista a integrar los datos recopilados.

Cuando se analizan procesos y decisiones el primer paso es identificar condiciones y acciones, conceptos comunes a todas las actividades.

Condiciones: Son los posibles estados de una entidad (persona, lugar, objeto o evento). Las condiciones cambian y por esto el analista se refiere a ellas como variables de decisión.

Por ejemplo una factura, si es autorizada o no, realiza un proceso diferente, cada condición cumple con cierto seguimiento y constituye una variable de decisión por tener dos o más alternativas.

Cuando se conocen todas las posibles condiciones, el siguiente paso del analista es determinar que hacer cuando se presentan algunas de éstas.

Acciones: Son las opciones; que comprenden pasos, actividades o procedimientos, que puede elegir una persona cuando se enfrentan a un conjunto de condiciones.

Como ayuda para comprender y adaptar condiciones y acciones, se emplean: diagramas de flujo de datos, dibujos, arboles de decisión y el español estructurado.

Diagrama de flujo de datos y de procesos: Herramientas que esquematizan los procesos involucrados para el desarrollo del nuevo sistema. Considerando que primero se debe identificar las condiciones que se presenten en un proceso y las acciones que se deben tomar cuando este sucede.

Dibujos: Otra herramienta es con el apoyo de dibujos en donde se hace una secuencia de todos los pasos en el orden en que se llevan a cabo todos los procesos.

Arboles de decisión: Son representaciones gráficas y secuenciales de las variables de decisión que indican qué condiciones considerar en primera instancia, cuáles en segunda y así sucesivamente. La raíz del árbol de decisión es el punto donde se inicia el análisis de una situación específica, las ramas indican la secuencia de decisiones que conducen a la acción apropiada que es necesario emprender.

Los arboles son eficaces cuando es necesario describir problemas con más de una condición. Pero si existen varias combinaciones de condiciones puede crecer y tener un tamaño considerable, constituyendo un riesgo para el analista, de no determinar qué políticas o estrategias de la empresa son la guía para la toma de decisiones específicas. Cuando aparecen estos problemas, entonces es momento de considerar otras herramientas.

Español Estructurado: El español estructurado es un recomendable método para evitar los problemas de ambigüedad del lenguaje al establecer condiciones y acciones, tanto en procedimientos como en decisiones. Este método utiliza declaraciones para describir los procesos.

Las especificaciones en español estructurada requieren que el analista primero identifique las condiciones que se presentan en un proceso y las decisiones que se deben tomar cuando esto sucede, junto con las acciones correspondientes. Sin embargo, este método también le permite hacer una lista de todos los pasos en el orden en que se llevan a cabo. Por otra parte, el resultado puede ser revisado con facilidad por otras personas para detectar y corregir tanto errores como equivocaciones.

Las declaraciones pueden ser en forma *secuencial* donde se emplean varias instrucciones en secuencia para describir un proceso, de *decisión* que incluyen secuencias de acciones dentro de estructuras de decisión que sirven para identificar condiciones, y de *interacción* las cuales son instrucciones de actividades rutinarias de operación, y que se repiten mientras existen ciertas condiciones o hasta que éstas se presentan.

3.2.6. CONCLUSIONES

La reingeniería no es más que una propuesta que por sus características puede o no funcionar en las empresas. Se debe tener en cuenta que cada organización es como un pequeño ecosistema, con características, cultura y patrones propios. La reingeniería y el uso adecuado de las tecnologías pueden dar enormes beneficios a las empresas, pero la manera en la que éstos penetran en cada empresa es diferente; no se trata de recetas o de copiar

patrones extranjeros, los empresarios deben analizar cómo estas filosofías o formas de trabajar que ya han tenido éxito en otras organizaciones pueden funcionar en la propia. La clave del éxito está en el conocimiento y en la habilidad

Sabemos que durante la etapa de análisis y diseño de sistemas, es cuando el analista se esta involucrando con el conocimiento de las operaciones y actividades de la organización una vez conocidas debe poner especial atención en determinar si realmente los procesos son los adecuados y eficientes o hay la necesidad de aplicar una reingeniería de procesos. A continuación se ejemplifica un caso que se puede presentar en cualquier empresa.

Supongamos que una empresa solicita un sistema que lleve el control de la gente que sale de comisión. El líder de proyecto a cargo, hace un estudio preliminar de cómo realizan las actividades, percatándose que el sistema no es grande y requiere de simples operaciones, por lo que pronostica tres meses para su desarrollo. Pero al estudiar el flujo de información, identificó que la mayor parte de los datos que se capturan en el sistema requieren la autorización y la firma de una serie de personas previas, obstaculizando uno de los objetivos importantes del sistema: agilizar la solicitud de comisiones para que el trabajador reciba oportunamente sus viáticos.

Ante esta problemática de burocracia y cuellos de botella de información, el líder de proyecto concertó una reunión con la gente operativa y administrativa para exponer que aunque el sistema ayuda a tener un control del seguimiento de las comisiones y da rapidez en la contabilidad de los gastos que generaron éstas, si no se depuran los procesos por los cuales pasa la información para su revisión y autorización, las ventajas por automatizar estas actividades serán mínimas, siendo quizá más un costo que un beneficio. La mayoría de los asistentes estaba de acuerdo que se requería modificar la manera de llevar a cabo los procesos, y entonces el líder de proyecto empezó a hablar de la Reingeniería de Procesos.

Por lo regular no es tan sencillo convencer a los usuarios de que es necesario aplicar la reingeniería de procesos cuando existen problemas en la manera que llevan a cabo su trabajo; y aunque pudieran estar de acuerdo sobre innovar sus procesos, las empresas a veces suelen ser tan grandes que los procesos son institucionales y si se requiere cambiarlos se necesita toda una revolución hasta llegar a afectar otras organizaciones y al grado de modificar la misma administración del gobierno de un país. Por lo que algunas veces es imposible cambiarlos.

La reingeniería no promete curas milagrosas. No ofrece ningún arreglo rápido, sencillo e indoloro: antes bien implica trabajo difícil. Exige que los que manejan las compañías y los que trabajan en ellas modifiquen su modo de pensar, no menos que lo que hacen. Se requiere que las compañías cambien sus viejas prácticas por otras enteramente nuevas. Hacer esto no es fácil y trivial se requiere de personal con experiencia en el tema.

Las principales corporaciones, en casi todas las industrias, ya han empezado a rediseñar sus procesos, y a medida que muchas otras llevan también los suyos a un nivel más alto de rendimiento, la opción de reingeniería se convierte en una necesidad competitiva.

El mundo de la revolución industrial esta cediendo el campo a una economía global, a poderosas tecnologías informáticas y a un cambio inexorable. Se levantan el telón de la edad de la Reingeniería. Los que respondan a su llamada escribirán las nuevas reglas de los negocios. Todo lo que se necesita es voluntad de triunfar y valor para empezar.

Nota: Lo anterior expuesto sobre reingeniería solo es algo superficial, realmente se tiene que estudiar y aplicar toda una metodología para llevarla a la práctica y obtener resultados exitosos.

La mayor parte del sistema SISEFIN cumple con los estándares presupuestales y procedimientos oficiales ya establecidos por PEMEX, los cuales no pueden, ni deben ser cambiados a menos que sea institucional. Se depuraron algunas operaciones para agilizar el desempeño presupuestal a nivel gerencia y sobre todo se definieron otras a seguir, siendo que actualmente no todas las gerencias lo aplican. Por ejemplo, en la mayoría de las gerencias el control del presupuesto lo llevan a nivel proyecto, pero se detectó que la descripción del proyecto abarca conceptos muy generales que para la administración financiera institucional son apropiados, pero para una gerencia en particular, varias actividades que generan un gasto necesitan ser especificadas como proyectos particulares y darles un seguimiento sin mezclarlas con otras. Para este buen control se determino manejar subproyectos.

Aunque no se aplicó la reingeniería de procesos en el sistema SISEFIN, siempre se buscó que los procedimientos fueran los adecuados y óptimos para su automatización y así obtener un mejor control del presupuesto interno y brindar rápidos resultados al elaborar los informes solicitados por el departamento de Finanzas.

3.3. ANÁLISIS DE FLUJO DE DATOS

3.3.1. DEFINICIÓN

Los analistas desean conocer las respuestas a cuatro preguntas específicas: ¿qué procesos integran el sistema?, ¿qué datos emplea cada proceso?, ¿qué datos son almacenados? y ¿qué datos ingresan y abandonan el sistema?. De lo anterior es claro que se da gran importancia al análisis de los datos.

Los datos son la guía de las actividades de la empresa. Seguir el flujo de datos por todos los procesos de la empresa, les dice mucho a los analistas sobre cómo se alcanzan los objetivos de la organización.

El *análisis de flujo de datos* estudia el empleo de los datos en cada actividad. Documenta los hallazgos con diagramas de flujo de datos que muestran en forma gráfica la relación entre procesos y datos, y en los diccionarios de datos que describen de manera formal los datos del sistema y los sitios donde son utilizados.^[25]

[25] Senn, James.- *Análisis y Diseño de Sistemas de Información.*- Traduc. Urbina Medal Edmundo Gerardo y otros.- 2ª Edición.- Editorial McGRAW-HILL.- México.- 1994 - pp. 177.

3.3.2. ESTRATEGIA DE FLUJO DE DATOS

El análisis de flujo de datos examina el empleo de los datos para llevar a cabo procesos específicos de la empresa. El análisis puede pensarse de tal manera que se estudien actividades del sistema desde el punto de vista de los datos: dónde se originan, cómo se utilizan o cambian, hacia dónde van, incluyendo las paradas a lo largo del camino que siguen desde su origen hasta su destino.

La estrategia de flujo de datos muestra el empleo de éstos en forma gráfica. Las herramientas para el flujo de datos ayudan a ilustrar los componentes esenciales de un sistema junto con sus interacciones.

El análisis de flujo de datos utiliza las siguientes herramientas:

1. Diagrama de flujo de datos
2. Diccionario de datos
3. Diagrama de estructura de datos o modelo lógico
4. Diagrama de flujo estructurado o cartas estructuradas

3.3.3. DIAGRAMA DE FLUJO DE DATOS

Es una herramienta gráfica que se emplea para describir y analizar el movimiento de datos a través de un sistema. Los diagramas de flujo de datos señalan las actividades paralelas, se concentran en el *movimiento de datos* a través del sistema, no en los dispositivos o el equipo. Los analistas identifican y describen datos, desde el inicio hasta el final del proceso. Es muy importante determinar cuándo entran los datos al área de aplicación y cuándo salen de ésta.

Los analistas pueden trabajar con los usuarios y lograr que participen en el estudio de los diagramas de flujo de datos. Los usuarios pueden hacer sugerencias para modificar los diagramas con la finalidad de describir la actividad con mayor exactitud. Si los problemas no son detectados en las primeras fases del proceso de desarrollo, entonces será difícil corregirlos cuando aparezcan más adelante. Evitar los errores desde el inicio puede prevenir una posible falla del sistema.

DIAGRAMAS LOGICOS Y FISICOS DE FLUJO DE DATOS

Los diagramas de flujo de datos son de dos tipos ^[26]:

➤ **Diagramas físicos de flujo de datos:**

Proporcionan un panorama de la situación actual. Muestra las tareas que lleva a cabo y cómo Las características físicas incluyen:

[26] Senn, James.- *Análisis y Diseño de Sistemas de Información.- Traduc. Urbina Medal Edmundo Gerardo y otros.- 2ª Edición.- Editorial McGRAW-HILL.- México.- 1994 - pp. 190.*

- Nombres de personas
- Nombres o números de formatos y documentos
- Nombres de departamentos
- Equipo y dispositivos
- Gabinetes de archivo
- Ubicaciones
- Nombres de procedimientos

Estos diagrama identifican a las personas y lo que hacen, qué documentos y formas inician las diferentes actividades y el equipo empleado para el procesamiento. Asimismo, identifican el movimiento de personas, documentos e información entre departamentos y diversas localidades.

➤ **Diagramas lógicos de flujo de datos:**

Proporcionan un panorama de flujo de datos entre los procesos. Describen datos, procesos y eventos en forma diferente. Son más abstractos. El analista tiene que conocer el trabajo que debe hacerse más que lo que hacen las personas actualmente. Al centrarse en los elementos de fondo, aspectos lógicos y no físicos, es cuando el analista comprende la estructura del sistema. Sólo entonces el analista de proyecto puede desarrollar un entendimiento completo y sentar las bases para diseñar el sistema correcto.

3.3.4. DICCIONARIO DE DATOS

Los diccionarios de datos son un componente importante del análisis estructurado. Contiene las características lógicas de los sitios donde se almacenan los datos del sistema, incluyendo nombre, descripción, alias, contenidos y organización. Sirve como punto de partida para identificar los requerimientos de las bases de datos durante el diseño del sistema.

Un *diccionario de datos* es un catálogo de los elementos en un sistema. En un diccionario de datos se encuentra la lista de todos los elementos que forman parte del flujo de datos en todo el sistema. El diccionario guarda los detalles y descripciones de todos estos elementos.

El diccionario se desarrolla durante el análisis de flujo de datos y auxilia a los analistas que participan en la determinación de los requerimientos del sistema. Sin embargo, también se emplea su contenido durante el diseño.

IMPORTANCIA DEL DICCIONARIO

Los analistas utilizan los diccionarios de datos por cinco razones importantes.^[27]

1. Manejar los detalles en sistemas grandes.
2. Comunicar un significado común para todos los elementos del sistema.
3. Documentar las características del sistema.
4. Facilitar el análisis de los detalles con la finalidad de evaluar las características y determinar dónde efectuar cambios en el sistema.
5. Localizar errores y omisiones en el sistema.

CONSIDERACIONES DE UN REGISTRO DEL DICCIONARIO

El diccionario contiene dos tipos de descripciones: elementos de datos y estructuras de datos. Los elementos de datos se agrupan para formar una estructura de datos.

Elemento dato: El nivel más importante de datos es el elemento dato (campo, dato o parte elemental). Los elementos dato son los bloques básicos para todos los demás datos del sistema. Por sí mismos no con llevan suficiente significado. Por ejemplo, el significado de FECHA que en relación con la factura, es la fecha en que se expidió la factura

Estructuras de datos: Una estructura de datos es un grupo de datos elementales que están relacionados con otros y que en conjunto describen un componente del sistema. Por ejemplo la estructura de datos FACTURA está definida por un grupo de datos elementales que incluye la fecha, el vendedor, la dirección de éste y detalles relacionados con los productos de la factura.

DESCRIPCION DE LOS ELEMENTOS DATO

Cada entrada en el diccionario de datos consiste de un conjunto de detalles que describen los datos utilizados o producidos por el sistema. Cada uno está identificado con un nombre, descripción, alias y longitud.

Nombre de los datos: Para distinguir un dato de otro, los analistas les asignan nombres que sean significativos. Los nombres se emplean para hacer referencia a cada elemento durante todo el proceso de desarrollo de sistemas. Algunas organizaciones imponen estándares para el desarrollo de nombres de datos. Un estándar común es que no deben contener espacios en blanco.

[27] Senn, James.- *Análisis y Diseño de Sistemas de Información.*- Traduc. Urbina Medal Edmundo Gerardo y otros.- 2ª Edición.- Editorial McGRAW-HILL.- México.- 1994 - pp. 211.

Descripción de los datos: La descripción de un dato indica de manera breve lo que éste representa en el sistema. Las descripciones de datos deben escribirse con la suposición de que la persona que las leerá no sabe nada con respecto al sistema.

Longitud: Identifica el número de espacios necesarios para cada dato.

Alias: Con frecuencia el mismo dato recibe varios nombres, mismos que dependen de quién haga uso del dato. Estos nombres se denominan alias. Un diccionario de datos significativo debe incluir todos los alias.

Valores de los datos: En algunos procesos sólo son permitidos valores muy específicos para los datos. Por ejemplo, la letra C es Contabilidad, la P personal y la V ventas. Si los valores de los datos están restringidos a un intervalo específico, esto debe verse reflejado en la correspondiente entrada del diccionario de datos. Todos estos detalles serán de utilidad a los analistas más adelante, cuando diseñen los controles del sistema.

Los diccionarios de datos son un camino para estructurar los datos del sistema.

3.3.5. ESTRUCTURA DE DATOS (MODELO LÓGICO)

Son herramientas básicas que muestran los requerimientos lógicos de las estructuras de datos de una aplicación. Este diagrama es una descripción de los datos asociados con las entidades mostrando las relaciones que existen entre ellas. Su finalidad es comunicar los requerimientos de datos a un diseñador o administrador de la base de datos.

NOTACION

Una notación común se usa al preparar los diagramas de estructuras de datos. Las entidades se representan mediante rectángulos, con el nombre de la entidad en la parte de arriba y una lista de atributos (datos o campos) que describan la entidad. Cada entidad se puede identificar mediante un atributo de llave, el cual, por convención es el primer campo mencionado.

3.3.6. DIAGRAMA DE FLUJO ESTRUCTURADO O CARTAS ESTRUCTURADAS

Los diagramas de flujo estructurado son herramientas gráficas que muestra la estructura jerárquica del sistema y de cada uno de los componentes y la interacción y flujo de información entre los módulos o subsistemas de la aplicación y en su caso sistemas externos con los que debe tener comunicación. Estas diagramas fuerzan al diseñador a estructurar software modular y descendente, es decir, separado en componentes manejables.

3.4. HERRAMIENTAS ASISTIDAS POR COMPUTADORA

3.4.1. DEFINICIÓN

Las herramientas automatizadas nos permiten ser más productivos. En el ámbito de la informática, las herramientas mejoran la calidad y la magnitud de la interacción entre los usuarios y los responsables del proceso de desarrollo. Las herramientas automatizadas ayudan a hacer el mejor uso del tiempo invertido en el desarrollo de un proyecto.

Una *herramienta CASE* (Computer-Aided Software Engineering - Ingeniería de Software Asistido por Computadora) son herramientas poderosas para el análisis y diseño de sistemas. Una *herramienta I-CASE* (CASE - Integral) son las herramientas que cubren todos los aspectos del desarrollo de software y están integradas.^[28]

Una herramienta es cualquier dispositivo que cuando se emplea en forma apropiada, mejora el desempeño de una tarea. El objetivo de las herramientas CASE es automatizar los aspectos clave de todo el proceso de desarrollo, desde el principio hasta el final

El análisis y diseño interactúa con herramientas CASE por medio de diagramas. Los diagramas son usados para tener una mejor visión y una clara idea de la relación que guardan los datos y como están conformados. Anteriormente el analista en sistemas dibujaba sus diagramas manualmente, ocasionalmente, cometía errores, inconsistencias, omisiones, además que su diseño era lento. Las herramientas CASE dan precisión en la diagramación. Una buena herramienta CASE emplea diferentes tipos de diagramas los cuales son precisos y pueden ser chequeados por la computadora. Además ayuda a la construcción física de la base de datos, mediante el modelo de datos lógico y desarrolla procesos a través del modelo de procesos.

Las herramientas integrales proporcionan un ambiente que automatiza tareas clave a lo largo de todo el proceso de desarrollo. Si bien estas herramientas incluyen facilidades para manejar aspectos de análisis y desarrollo, también facilitan el diseño, administración y mantenimiento del código. Asimismo brindan un ambiente eficiente para crear, almacenar, manipular, administrar y documentar sistemas, por lo que abarcan todo el ciclo de vida de la aplicación.

Una característica de una herramienta I-CASE es que ésta genera código, automatizando la preparación del sistema. La firme integración de herramientas sobre el análisis, el diseño y el código generador proporciona alta productividad más que las herramientas donde no se pueden acoplar. Algunas herramientas I-CASE son PowerBuilder, Developer 2000 de Oracle, UNIFACE.

Como se verá más adelante el modelo lógico de la base de datos del sistema SISEFIN fue desarrollado a través de una herramienta CASE llamada Erwin Logic Works (Figura 3.1). Desafortunadamente el IMP no cuenta con una herramienta CASE para el análisis o

[28] Pressman, Roger. - *Ingeniería de Software, Un enfoque práctico*. - Traduc. Troya José María y Hernández Yarex Luis. - 7ª Edición. - Editorial McGRAW-HILL. - México. - 1990.

para generar código, aunque recientemente Erwin esta demostrando una herramienta para el diseño de modelo de procesos, basada en la metodología de Jourdon, misma que se esta probando en el Instituto.

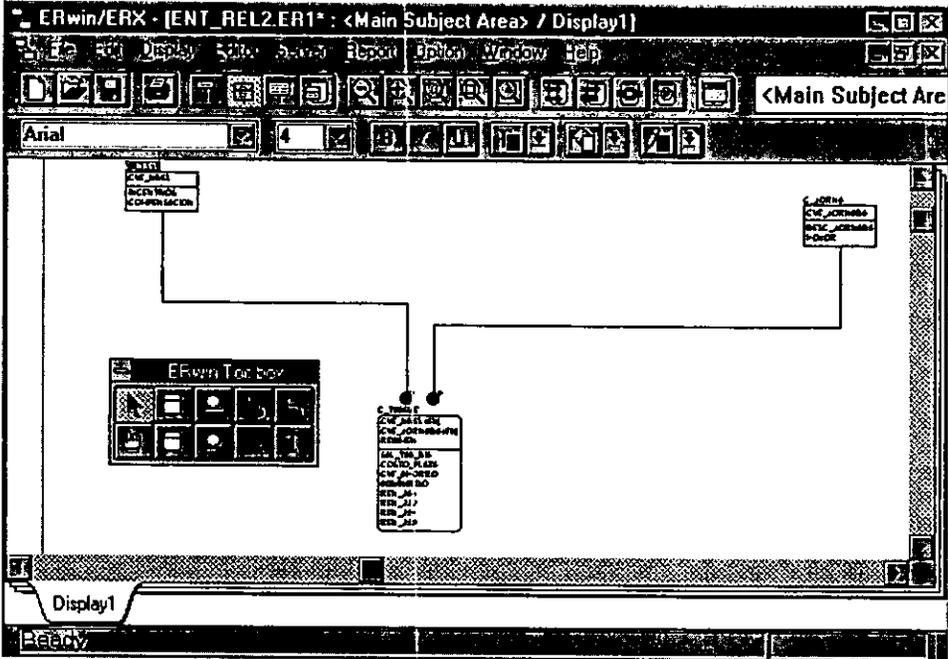


Figura (3.1) Pantalla principal de Erwin Logic Works

3.4.2. VENTAJAS

Mejora en la productividad: Las herramientas CASE cuando se utilizan adecuadamente, aumentan la productividad y eficiencia del equipo de trabajo (analistas, diseñadores, programadores, etc.) al disminuir la cantidad de tiempo necesaria para documentar, analizar y desarrollar sistemas de información.

Mejora en la eficacia: Si la productividad significa realizar la tarea correcta, la eficiencia significa hacer esta tarea en forma correcta. Las herramientas CASE pueden sugerir la mejor forma para abordar una tarea. Identificando los requerimientos del usuario, trasladarlos en una forma comprensible y comunicarlos a todas las partes interesadas, así como iniciar con rapidez la codificación de programas.

Mejora en la calidad del sistema de información: Los usuarios de los sistemas de información desean calidad en el sistema con un tiempo razonable y con las herramientas CASE se logra esto.

3.4.3. FUTUROS DESARROLLOS CON CASE

En el futuro las herramientas CASE llegarán a ser más prevaletientes en el desarrollo. Los vendedores de CASE están tratando de que sus productos asistan el desarrollo de lenguajes populares como Java, Delphi y Centura. Otras están intentando hacer multisoporte de lenguajes y que estas trabajen con diversas bases de datos, así como diversas tecnologías. Todo esto con la finalidad de que los desarrolladores de un ambiente Cliente/Servidor mejoren aún más el análisis, diseño y programación.

Las herramientas CASE nunca convertirán a un analista malo en uno bueno, el analista debe saber cómo formular las preguntas correctas y desarrollar su habilidad para interactuar en forma efectiva con las demás personas. Aunque actualmente las herramientas CASE están en uso y ofrecen muchos beneficios, el obtener y comprender los requerimientos son tareas todavía realizadas por los seres humanos, y lo más probable es que se continúan de tal forma.

3.5. ANALISIS DE LA INFRAESTRUCTURA CLIENTE/SERVIDOR

Para implementar una arquitectura Cliente/Servidor se deberá especificar los elementos de hardware, software y recursos humanos que participarán en la misma. Siendo necesario realizar un estudio de requerimientos para detectar los elementos que se integrarán en la diseño de la arquitectura.

3.5.1. EVALUACIÓN DE LA INFRAESTRUCTURA ACTUAL

El grupo de trabajo del proyecto debe conocer la infraestructura actual, los requerimientos en cuanto arquitectura técnica del sistema de información. Además de identificar el lugar físico donde las actividades son ejecutadas, la ubicación de los usuarios y así poder especificar la localización del hardware para la nueva aplicación.

Arquitectura Técnica Actual:

Se determina el tamaño, complejidad y potencia de los equipos que se están utilizando en la actualidad y que están sujetos a su utilización por el nuevo sistema. Igualmente se describe cada una de las aplicaciones que están corriendo en las CPU del área.

Instalaciones Físicas:

Se describe y evalúa las instalaciones físicas actuales para saber todos los espacios disponibles, dónde son ejecutadas las actividades y dónde esta localizado el hardware para la aplicación. También se considera la ubicación de los usuarios quienes accesan la aplicación.

3.5.2. NECESIDADES DE INFRAESTRUCTURA

En función a lo encontrado en la fase de Análisis y tomando en cuenta las conclusiones de la actividad anterior, se establecerá la propuesta de necesidades de infraestructura como base para la toma de decisiones.

Este estudio abarca la identificación de las necesidades y requerimientos de la infraestructura en base a los alcances del sistema C/S.

Configuración de Arquitectura Técnica Actual:

Tiene como propósito determinar los elementos comunes y particulares de la arquitectura técnica, estableciendo tamaño, complejidad y potencia de los equipos que soportan el nuevo sistema y su ambiente de desarrollo y pruebas.

Configuración de Instalaciones Físicas:

Tiene como propósito determinar el espacio, distribución y preparaciones especiales de las instalaciones físicas, tomando en cuenta las instalaciones físicas actuales.

3.5.3. ESTABLECIMIENTO DE LA ARQUITECTURA C/S

A partir de este estudio se podrá determinar los detalles de la arquitectura para la operación del sistema, como son los elementos de hardware y software que serán usados entre los componentes cliente, servidor y la red. También se definirán todas las ubicaciones donde la aplicación será instalada.

Como se menciona en la propuesta técnico económica, la GIM y todas sus regiones (Norte, Sur, Marina Noroeste y Marina Suroeste) cuentan con los componentes para la construcción y operación del sistema C/S. Pero en la mayoría de las veces las organizaciones no tienen estas condiciones y si desean operar un sistema con las bondades que ofrece una arquitectura C/S es imprescindible hacer un estudio detallado para la construcción de la infraestructura de acuerdo a las necesidades y requerimientos de la organización.

Cuando se trata de sistemas pensados para uso exclusivo en una red punto a punto, a nivel departamental o de pequeña empresa, se está hablando de C/S pequeño. Pero por otra parte, también se puede hablar de sistemas altamente complejos, con múltiples entornos de operación y desarrollo, en donde varios sistemas operativos coexisten en una misma red compartiendo o distribuyendo amplios volúmenes de información, muchas veces de fuentes heterogéneas, entre cientos de equipos.

Dentro del análisis de la arquitectura C/S se determinan sus alcances con respecto al sistema, evitando un dimensionamiento inadecuado de un sistema C/S, que casi resulta en enormes costos, con esfuerzos de desarrollo desperdiciados y en un repudio a cualquier cambio tecnológico mayor.

3.6. EJEMPLO: ANALISIS DEL SISTEMA SISEFIN

En el análisis del sistema SISEFIN, los procesos financieros que intervienen contemplan aspectos técnicos e información clasificada de PEMEX. Y con fines prácticos de la tesis, se extraerá los puntos principales que conforman el análisis del sistema, exponiendo las necesidades, requerimientos y procesos de manera entendible.

ANALISIS DE LA SITUACION ACTUAL

En este documento se esboza los datos obtenidos durante la investigación de las necesidades del proyecto SISEFIN. La combinación de las entrevistas, observación, recopilación de documentos y revisión de expedientes, produjo gran cantidad de detalles relacionados con los procesos financieros que lleva a cabo la Gerencia de Inspección y Mantenimiento (GIM), identificando los problemas y requerimientos para que el seguimiento presupuestal sea más eficiente.

DESCRIPCION DE LOS PROCEDIMIENTOS ACTUALES PARA LLEVAR A CABO EL PROCESO DEL EJERCICIO PRESUPUESTAL

Conforme a las indicaciones del Coordinador del Proyecto por parte de la GIM. El proceso para llevar el control financiero del ejercicio presupuestal de la gerencia esta compuesto por los siguientes procedimientos.

Cada año todas las dependencias de PEMEX elaboran un presupuesto solicitado. Este presupuesto es un plan esencialmente numérico que se anticipa a las operaciones financieras que se pretenden llevar a cabo en el período de un año. La estimación es realizada en base a la información estadística que posea incluyendo datos históricos, con lo cual se hacen armados presupuestales, que se calendarizan mensualmente. La **Familia Presupuestal** es primordial en la elaboración del presupuesto, ya que mediante la construcción de ésta se puede saber a qué programa y proyecto pertenece determinado gasto, así como también su guión y concepto de origen.

El proceso para que el presupuesto solicitado sea autorizado es el siguiente.

SOLICITUD PRESUPUESTAL

La **Secretaría de Hacienda y Crédito Público (SHCP)** a través del área de Finanzas de PEMEX, solicita a las dependencias que integran esta organización, el **Presupuesto Anual** el cual nace a partir del **Presupuesto Solicitado** o **Anteproyecto** que se obtiene con la elaboración del programa anual de gastos estimados por cada gerencia. En este caso la GIM tiene el siguiente organigrama (Figura 3.2) compuesto por:

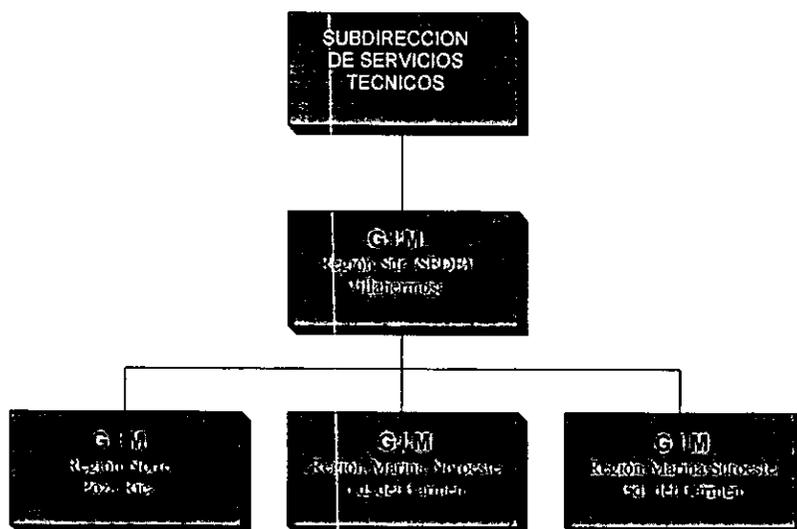


Figura (3.2) Organigrama

La **Gerencia de Inspección y Mantenimiento** en Sede, se encarga de integrar, revisar, jerarquizar y autorizar la información de Regiones, en caso de que existan modificaciones en el presupuesto solicitado se regresa al subgerente de región.

La información es enviada a **Evaluación de la Subdirección de Servicios Técnicos**, en donde se integra lo de las Subgerencias, para que después se envíe el anteproyecto a **Planificación y Finanzas** (si existe alguna modificación sobre la información, se regresa a la Gerencia correspondiente).

Después el anteproyecto es mandado a la **Dirección General de PEP** (PEMEX Exploración Producción) en donde también se revisa y se autoriza, una vez aceptado por la Dirección, se envía a **Finanzas PEP** para que posteriormente lo acepte **Finanzas Corporativo** que se encarga de integrar la cifras autorizadas, y consolidar el documento definitivo del **Anteproyecto de PEMEX**, y lo apruebe el **Director General de PEMEX**. Finalmente se envía a la **Secretaría de Hacienda y Crédito Público (S.H.C.P.)** después de la autorización de la S.H.C.P. el anteproyecto, ahora llamado **Presupuesto Autorizado** es devuelto a los interesados, de la misma como fue solicitado pero ahora de manera descendente.

Descripción Gráfica:

PROCEDIMIENTO ACTUAL DE LA SITUACION PRESUPUESTAL

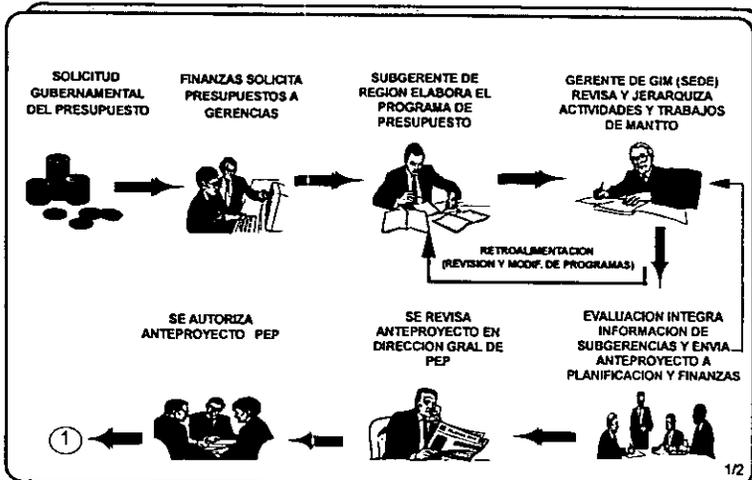


Figura (3.3) Esquematzación del proceso a través de dibujos. Lamina 1

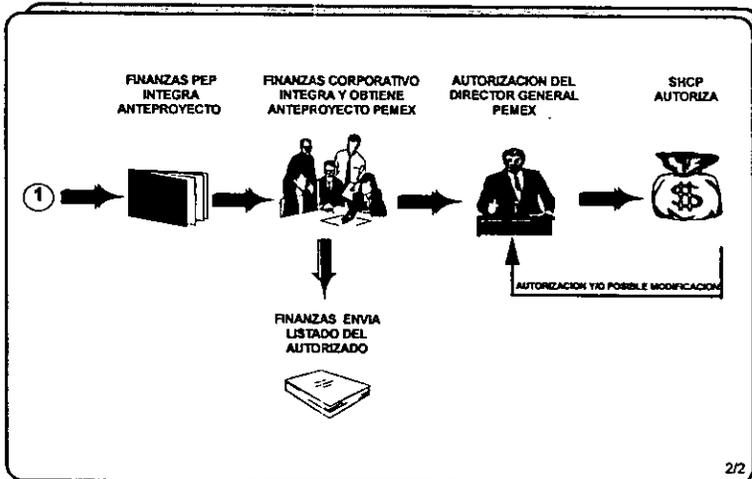


Figura (3.4) Esquematzación del proceso a través de dibujos. Lamina 2

PROCESO FINANCIERO DE LA GERENCIA

Una vez que la gerencia tiene autorizado su presupuesto, realiza la comparación entre el presupuesto solicitado y el autorizado, por lo general existen variaciones siendo necesario hacer ajustes. Con el definitivo plan presupuestal, la gerencia tiene la responsabilidad de

vigilar su cumplimiento, llevando un control de sus gastos y justificando ante **Finanzas PEP** todos sus movimientos financieros.

El Control del Ejercicio Presupuestal de la GIM tiene como principal objetivo dar seguimiento a los procesos financieros, y detectar oportunamente desviaciones con el programa estimado, así como, cualquier cargo incorrecto por parte de Finanzas PEP.

Los principales procesos financieros de la GIM son:

- Las operaciones financieras de la gerencia empieza con la requisición, haciendo un pedido con lo cual se adquiere un **compromiso**, que es cualquier documento que significa un gasto. El compromiso se registra con un folio autorizado y se envía a ventanilla única ó a Finanzas PEP. Este proceso se hace manualmente
- Cuando llega la factura se registra bajo el respaldo de una **familia presupuestal** y un **CAP** (Clave de Autorización Presupuestal) y hasta que sea pagada pasa a ser devengada. Los usuarios utilizan la herramienta de Excel para llevar un control de todas sus facturas.
- Tanto el compromiso como la facturación es respaldado por un **CAP** que es proporcionado por finanzas, pero muchas veces esta clave tarda en ser recibida y el registro del movimiento se retrasa.
- Al final de cada mes o cuando el gerente lo solicite se hace un reporte de los gastos de la gerencia y como han ido afectando el presupuesto autorizado. Los reportes son realizados en el momento haciendo cálculos manuales y la forma de presentarlos es por consolidados, sin mostrar a detalle los movimientos que intervinieron en el presupuesto de la gerencia.
- Por política institucional cada mes, la dependencia de Finanzas-PEP envía reportes mensuales sobre los movimientos presupuestales de la GIM, y aveces existen desviaciones o cargos incorrectos que la gerencia no detecta oportunamente.

En general todas sus operaciones las hacen en forma manual y los resultados los registran en hojas de calculo con la herramienta Excel. Además no se tiene un control adecuado de cómo las facturas, compromisos y CAPs afectan el presupuesto de la gerencia, incluyendo que tampoco se tiene un seguimiento a detalles de estos movimientos.

ANÁLISIS DE SUS REQUERIMIENTOS

Conforme al estudio de la situación actual se detectaron los siguientes requerimientos:

- Llevar el control interno de los presupuestos a nivel subproyecto, dando seguimiento a los compromisos y facturas a través de la clave de autorización presupuestal y la asignación de una familia presupuestal.

- Determinar qué facturas están afectando determinado proyecto y cuál es su monto acumulado por mes, y si coincide con lo estimado.
- Especificar las facturas que están respaldadas por una clave de autorización presupuestal (CAP) o por una familia presupuestal (subproyecto). Además de conocer cuáles ya se han devengaron.
- Saber los compromisos que se generaron y cómo esta afectando el presupuesto.
- Controlar el movimiento de las Claves de autorización Presupuestal a través del manejo de CAPs externos e internos
- Crear un ambiente donde la información pueda ser compartida y cualquier transacción o movimiento ejecutado sea detectado inmediatamente por todos los usuarios.
- Obtener confiable y oportunamente consultas, reportes y gráficas del ejercicio presupuestal.

La frecuencia y volumen de información por la gerencia es constante con un promedio de seis facturas diarias al igual que sus compromisos. Lo métodos de control aplicados son simplemente hacer cumplir el presupuesto autorizado conforme a lo estimado y llevar a cabo las reglas y políticas del presupuesto en base a lo dictado por PEMEX. Las personas que intervienen en el proceso son analistas que se encargan de dar seguimiento a todas las actividades del movimiento presupuestal de la gerencia, y el encargado de autorizar y tomar decisiones sobre ejercicio presupuestal es el gerente de la GIM.

ANALISIS DEL MODELO DEL SISTEMA

El propósito del SISEFIN es llevar el Control Interno de la Gerencia de Inspección y Mantenimiento. Para ello requiere la información de los catálogos institucionales y la captura de sus documentos (Figura 3.5) para que posteriormente se haga una comparación de su control interno con lo que les envía finanzas.

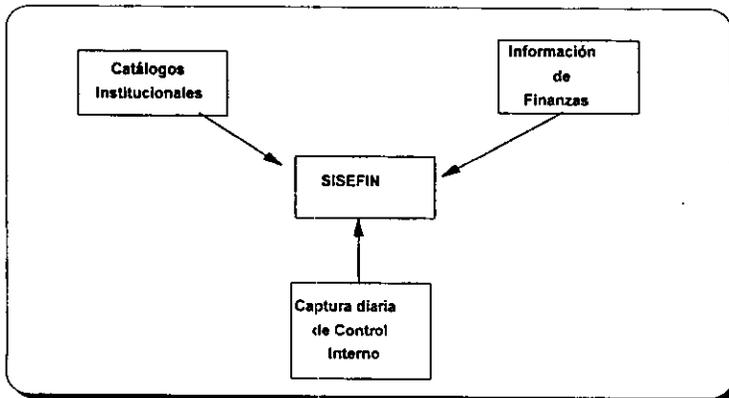


Figura (3.5) Esquema Funcional del Sistema de Seguimiento Financiero

DIAGRAMAS DE FLUJO DE DATOS

El flujo de información y las transformaciones que se aplican a los datos conforme se mueven de la entrada a la salida del sistema, se muestran en los siguientes diagramas de flujo.

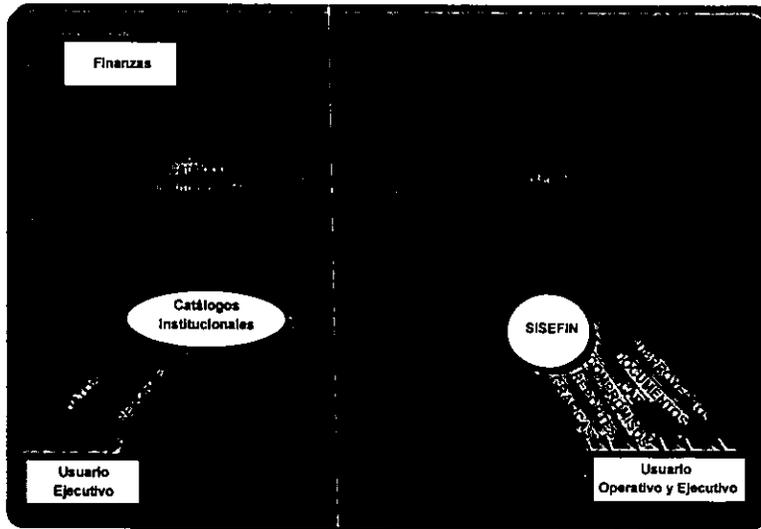


Figura (3.6) Diagrama de Flujo de Datos

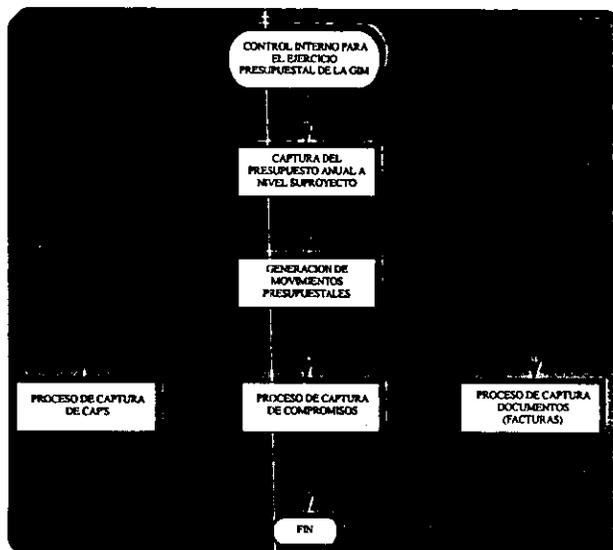


Figura (3.7) Diagrama del Control del Ejercicio Presupuestal de la GIM

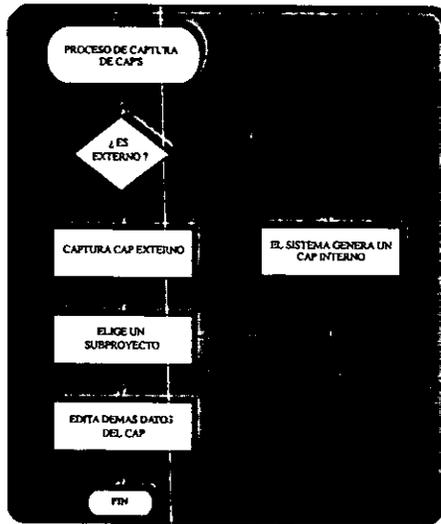


Figura (3.8) Diagrama de Proceso de captura de CAP's

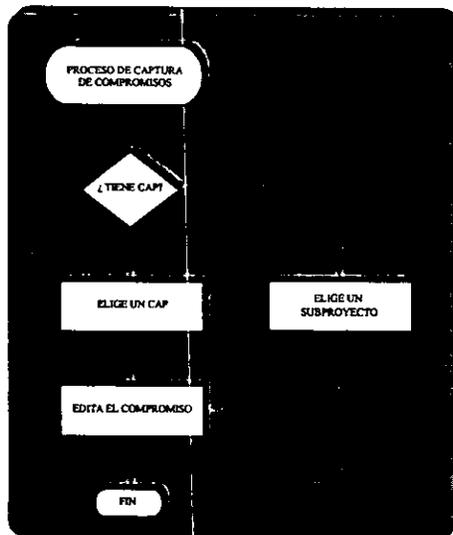


Figura (3.9) Diagrama de Proceso de captura de Compromisos

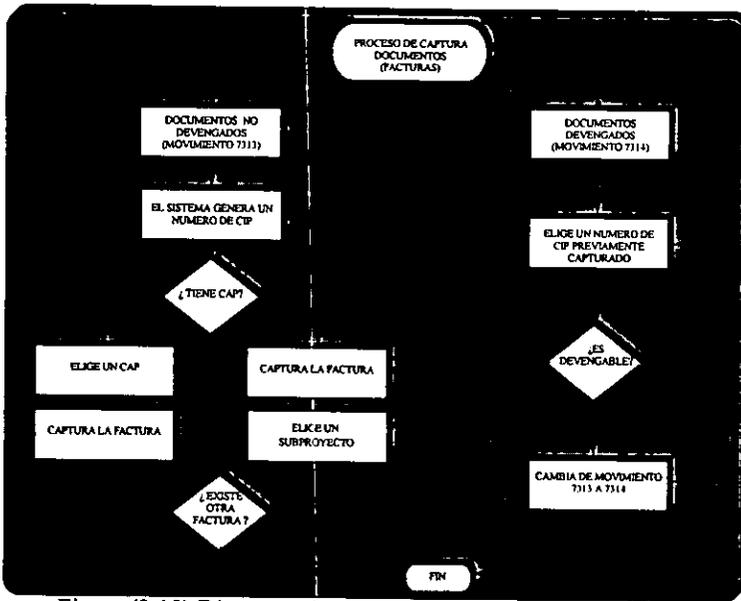


Figura (3.10) Diagrama de Proceso de captura de Documentos

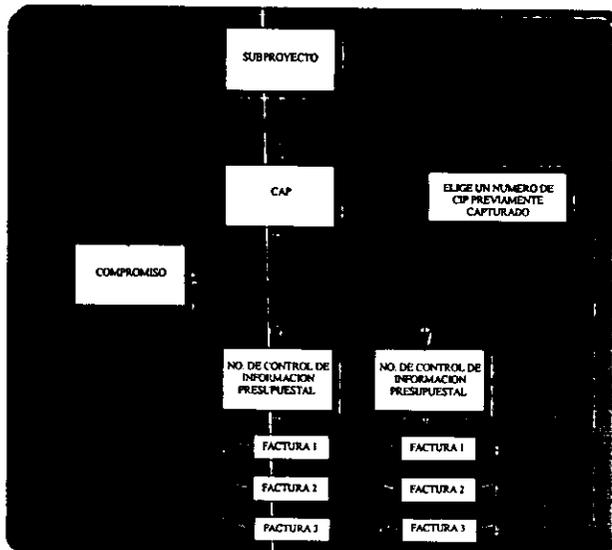


Figura (3.11) Diagrama representativo del Flujo de Procesos

MODELO DE PROCESOS

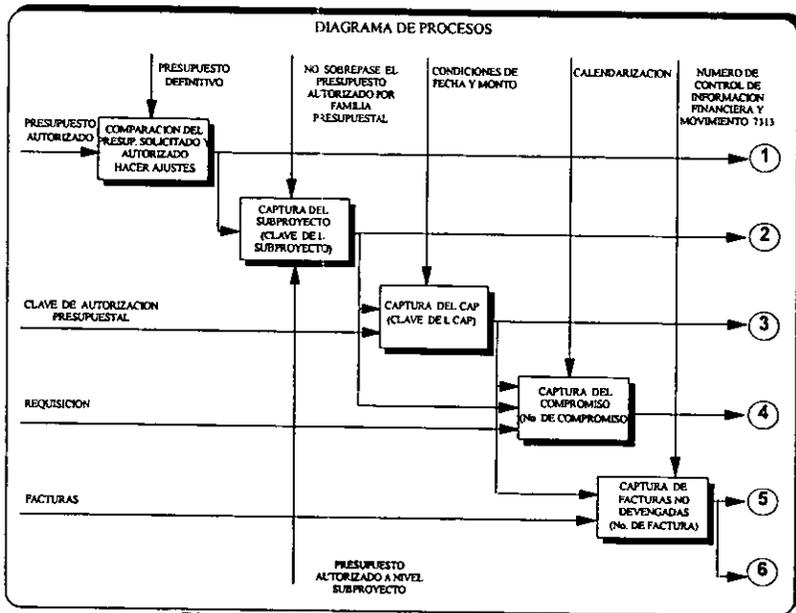


Figura (3.12) Modelo de Procesos Parte I

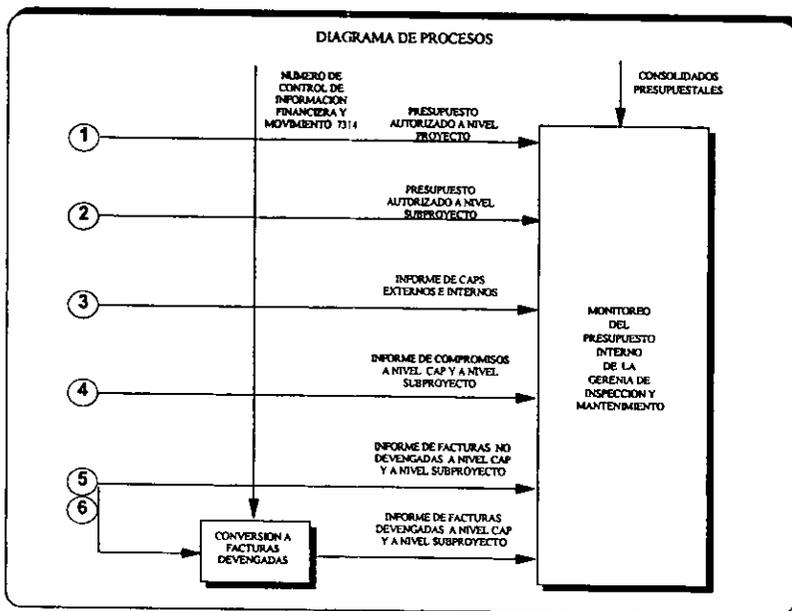


Figura (3.13) Modelo de Procesos Parte II

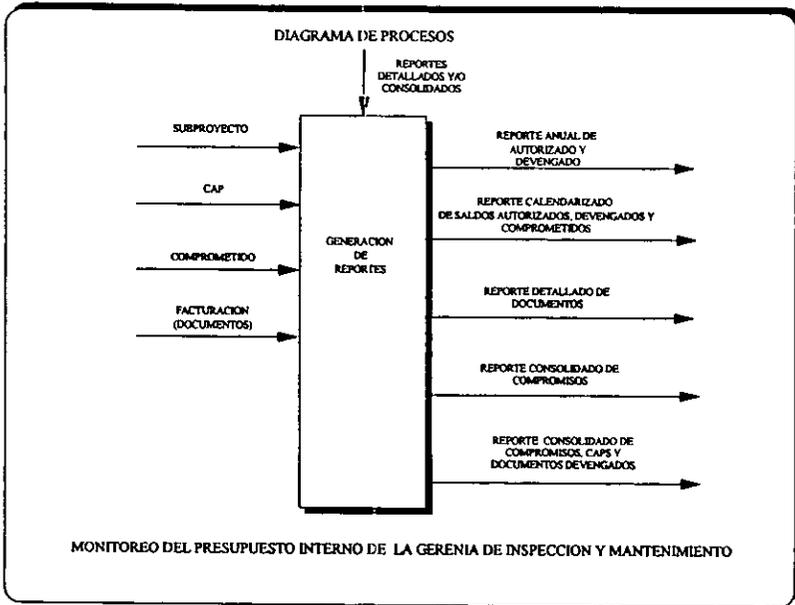


Figura (3.14) Modelo de Procesos Parte III

DIAGRAMAS DE NAVEGACION

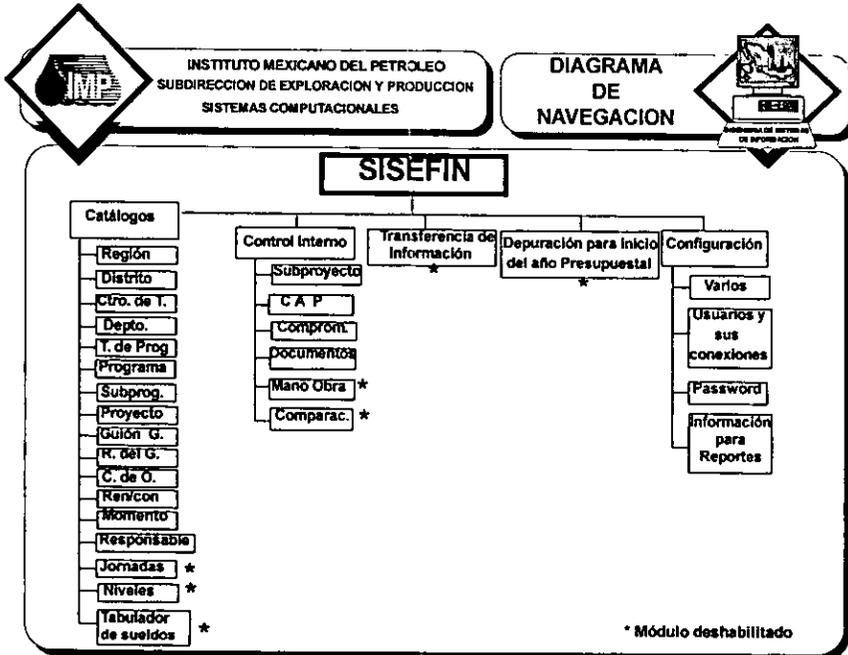


Figura (3.15) Diagrama de Navegación

DICCIONARIO DE DATOS

NOMBRE	TIPO	LONG	DESCRIPCION
ADEFAS	Carácter	1	Adeudo de Ejercicios Fiscales de Años Anteriores
CAP	Carácter	11	Clave de Autorización Presupuestal
COMPANIA	Carácter	40	Nombre del proveedor
CVE_CENTRO	Carácter	3	Clave del centro de trabajo
CVE_CONCEP	Carácter	8	Clave del concepto de origen
CVE_CTRADO	Carácter	3	Clave del centro de trabajo afectado
CVE_CTRDOR	Carácter	3	Clave del centro de trabajo afectador
CVE_DEPTO	Carácter	5	Clave del departamento
CVE_DPTADO	Carácter	5	Clave del departamento afectado
CVE_DPTDOR	Carácter	5	Clave del departamento afectador
CVE_FAC	Carácter	10	Número de la Factura
CVE_FICHA	Carácter	6	Clave del responsable que captura el documento
CVE_GUION	Carácter	2	Clave del guión del gasto
CVE_PROY	Carácter	8	Clave del proyecto
CVE_PROYOC	Carácter	8	Clave del proyecto de obra capitalizable
CVE_REGION	Carácter	1	Clave de la región
CVE_RENGLON	Carácter	3	Clave del renglón del gasto
CVE_SPROY	Carácter	8	Clave del Subproyecto
CVE_SUBP	Carácter	2	Clave del subproyecto
FECHA_CAP	Fecha	dd/mm/yyy y	Fecha de captura de un documento
FECHA_DEV	Fecha	dd/mm/yyy y	Fecha de devengado de un documento
FECHA_DOC	Fecha	dd/mm/yyy y	Fecha del documento
FOL_ADEF	Carácter	10	Número de folio de ADEFAS
JUSTIF	Carácter	100	Justificación del monto de la factura
MON_IVA	Numérico	11	Monto del IVA
MONTO	Numérico	11	Monto total de la factura
NO_CIF	Carácter	4	Número de Control de Información Financiera
NO_CIP	Carácter	7	Número de control Interno para un documento
etc.	etc.	etc.	etc.

Figura (3.16) Parte del diccionario de datos del sistema SISEFIN

ANÁLISIS DE LA INFRAESTRUCTURA CLIENTE/SERVIDOR

INSTALACIONES FÍSICAS ACTUALES

Ubicación y Tamaño: El servidor se encuentra físicamente ubicado en el sexto piso del Edificio Pirámide de PEMEX, y el lugar destinado es un cubículo acondicionado donde el acceso es restringido.

Accesos: El control de acceso a la red es mediante una contraseña al prender el equipo y posteriormente hay que proporcionar otra contraseña al entrar a la red.

Seguridad: *Física:* La seguridad es la adecuada, ya que el equipo tiene accesos controlados y se encuentra en un lugar cerrado.

Lógica: El equipo cuenta con un control de acceso mediante una contraseña por usuario al entrar a la red y al prender el equipo.

Instalaciones: La Gerencia de Inspección y Mantenimiento cuenta con las siguientes instalaciones:

Es una Red de Area Local (LAN) con topología Ethernet, compuesta en la parte del servidor de archivos de un RISC de IBM, con sistema operativo UNIX y en la parte de los nodos intervienen computadoras personales, estaciones de trabajo y terminales, aceptando cualquier marca, excepto Macintosh, con procesadores 486 y Pentium, y en memoria RAM 8 MB mínimo.

El protocolo por el cual se comunican todos los equipos es por medio de TCP/IP, a través del cable UTP. En general el cableado esta diseñado de forma estructurada, existiendo un concentrador conocido como HUB. El servidor tiene salida a la red PEMEX-PAQ a través del protocolo X.25, o por medio de Canal E1, creándose la comunicación con otras redes locales.

Con respecto al software, las estaciones de trabajo manejan como sistema operativo a MS-DOS o Windows 95 y como Interfaz Gráfica de Usuario a Windows en sus diferentes versiones, además de diverso software para trabajos de oficina. El software de red para estos equipos (nodos y servidor) es LAN WORK PLACE. En el servidor su software es compartido, principalmente almacenan herramientas como gráficos, procesadores de palabras, manejadores de imágenes y otros, que requieren de considerable espacio en disco duro y recursos de mayor rendimiento, razón principal para que estén en el servidor. El servidor también posee un manejador de base de datos relacional (RDBMS) llamado ORACLE Ver. 7.01 aunque por su escaso uso sólo un equipo está configurado para trabajar en un ambiente Cliente/Servidor.

Con lo anterior el servidor actúa como administrador de archivos, base de datos y también de impresión. La mayoría del equipo se conecta a un UPS (Ininterrumpido Suministro de Poder - Uninterruptable Power Supply) para garantizar la continuidad de la corriente eléctrica.

Para las demás regiones las condiciones de infraestructura son similares a la de la GIM de Villahermosa, siendo importante considerar los posibles factores ante la reestructuración de PEMEX, que pudieran afectar el ambiente técnico y operacional de la infraestructura, se tenga que configurar el equipo junto con los administradores de redes y bases de datos y crear el ambiente idóneo para la el sistema SISEFIN.

INSTALACIONES FISICAS REQUERIDAS

A continuación se especifican las configuraciones mínimas de hardware y software necesarias para la instalación y operación del sistema SISEFIN.

Se requiere que a nivel Región se cuente con el siguiente equipo:

- Una red de computadoras y de servicios de transmisión de datos, en cada lugar donde se instale.
- Servidor y computadoras Personales Compatibles configuradas como clientes y administrador de base de datos respectivamente.

REQUERIMIENTOS DE HARDWARE:

Servidor

- 1.- Suficiente espacio en disco duro.
- 2.- Memoria en RAM.
- 3.- Medios de almacenamiento.
- 4.- Canales de comunicación locales y remotos.

Cliente

- 1.- Espacio en disco duro para la instalación y operación del sistema.
- 2.- Memoria en RAM.
- 3.- Impresora esclava o en red.
- 4.- Monitor gráfico.
- 5.- Procesador.

REQUERIMIENTOS DE SOFTWARE:

Servidor

- 1.- Sistema Operativo.
- 2.- Software de red.
- 3.- Middleware.
- 4.- Sistema Manejador de Base de Datos Relacionales (RDBMS)

Cliente

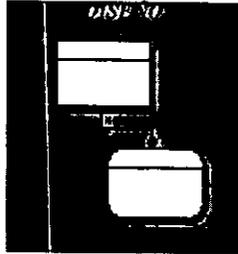
- 1.- Sistema Operativo.
- 2.- Software de red.
- 3.- Interfaz de Usuario Gráfico (GUI)
- 4.- Middleware.
- 5.- Front End.

ALCANCES

En el sistema SISEFIN son cinco personas por área las que van a operar y trabajar simultáneamente el sistema. Para la primera versión tendrá cobertura por red local que identifica solo una gerencia o subgerencia, posteriormente integrará toda la información con otras gerencias y subgerencias. La dimensión de este sistema C/S es pequeña pero con la certeza de que en un futuro sus aspiraciones serán las de crecer.

CAPITULO IV

DISEÑO DE SISTEMAS CLIENTE / SERVIDOR



4.1. DISEÑO PRELIMINAR

4.1.1. DEFINICIÓN

El diseño determina el éxito del sistema. Por lo que el *diseño preliminar* de sistemas es la selección y especificación de las características del sistema de información y la plataforma de computo en que operará, con la finalidad de elaborar del prototipo. Prácticamente el diseño preliminar y el prototipo se van haciendo al paralelo. Cuando se tiene terminado el prototipo del sistema C/S se muestra a los usuarios para su revisión y una vez autorizado, lo que fue el diseño preliminar se convierte en el diseño final anexando y considerando los puntos de vista y observaciones hechas al prototipo durante la presentación.

El manejo de proceso de diseño significa tomar los pasos necesarios para que el esfuerzo de construcción tanto del sistema como de la arquitectura C/S, avance en forma apropiada y produzca los resultados esperados.

Los eventos de los negocios son la fuerza manejadora para definir a detalle las aplicaciones Cliente/Servidor. Para una eficaz y eficiente aplicación se requiere diseñar múltiples modelos, elaborar un prototipo interactivo y tener conocimiento de la infraestructura técnica.

El diseño para el sistema SISEFIN se baso en un diseño estructurado, considerando que va a ser desarrollado en modo gráfico y con bases de datos relacionales, puntos característicos de una arquitectura Cliente/Servidor. Y conforme se fue avanzando en el diseño se aplico la reingeniería de software para descartar y/o aprovechar algunos diseños de la vieja aplicación.

DISEÑO DE LA ARQUITECTURA CLIENTE/SERVIDOR

Una compañía que se mueve de un sistema centralizado a un nuevo ambiente Cliente/Servidor tiene que pasar por numerosos cambios. Actualmente existe documentación que especifican pasos y consideraciones para que el movimiento a cliente/servidor sea exitoso. Pero lo más importante es que éste se dirija a los objetivos de la organización y a los alcances del desarrollo del nuevo sistema.

4.1.2. APLICACIÓN DEL USO DE INFORMACIÓN

La especificación de características de cómo la aplicación será usada debe ser de conocimiento para el equipo del proyecto y determinar que arquitectura o combinación de arquitecturas es la más eficiente. Estas características incluyen el número de usuarios, su localización, la prioridad de los eventos de procesos, la frecuencia de cada proceso, su respuesta en tiempo requerida para los procesos, datos necesarios, volumen de transacciones y consultas, accesos a los datos, además de la concurrencia de acceso a los mismos datos. Opciones técnicas se han establecido en base a los requerimientos físicos de la aplicación.

Los sistemas C/S por su naturaleza son sistemas de trabajo en grupo en forma distribuida que usan equipo de escritorio y el más robusto reside en los servidores (back end). El software sobre estos sistemas es relativamente nuevo, complejo, altamente configurable y capaz de mejorar diversas aplicaciones. Los sistemas Cliente/Servidor son encontrados dentro de todas las áreas de los negocios, incluyendo la medicina, manufacturación, ventas, servicios al cliente, ciencia o actividades de recreación (juegos como el Quake). En todos estos sistemas hay un inevitable impacto de hardware y software.

En el desarrollo de una aplicación Cliente/Servidor deben combinarse los mejores atributos entre la máquina cliente y el servidor. La interfaz basada en el cliente debe ser fácil de usar, incrementando la productividad del usuario final y facilitando el manejo eficiente de la información. El servidor debe reforzar la integridad y seguridad de los datos, promover las reglas de negocio, interoperar con otras plataformas y explotar la infraestructura. Y el acceso para el servidor desde el cliente debe ser transparente para los usuarios. Para construir toda esta arquitectura es necesario llevar a cabo un proceso de planificación y selección de acuerdo a sus componentes.

4.1.3. REDES PARA EL CLIENTE/SERVIDOR

El problema con las redes en la mayoría de las organizaciones es que se instalaron sin diseñarse, y con el constante crecimiento de éstas, el costo de mantenimiento se incremento, creándose obstáculos para su modificación. Por lo que para construir es necesario llevar un proceso de planificación, considerando los siguientes puntos básicos:

1. Selección de topología.
2. Selección de protocolo y direcciones IP.
3. Selección de los componentes del cableado.
4. Selección de tipo de Red.
5. Selección del Sistema Operativo de Red.
6. Selección del equipo físico (servidor y estaciones de trabajo).
7. Ubicación y especificaciones eléctricas.
8. Instalación del equipo físico y el sistema operativo de red.
9. Configuración del sistema y carga de aplicaciones.
10. Creación del entorno de usuario.
11. Administración de la red.

4.1.4. HARDWARE PARA EL CLIENTE/SERVIDOR

SELECCION DE MAQUINAS CLIENTE

Las maquinas clientes requerirán ser modificadas dependiendo de la aplicación Cliente/Servidor. Una aplicación que necesita velocidad de procesamientos matemáticos debe de tener un rápido coprocesador matemático. Una aplicación que necesita ejecutar grandes volúmenes de datos gráficos requiere un adaptador de gráficos y un gran monitor.

Para operar sistemas Cliente/Servidor los equipos debe cumplir con ciertas características mínimas como: tener procesador 486 o PENTIUM I,II, 16 MB en RAM o superior, un disco duro de 1 GB de capacidad y 16, 32 o 64 bits de adaptador gráfico.

SELECCION DE SERVIDORES

La función de un servidor de base de datos es mejorar la ingeniería de los datos para proporcionar rápidas respuestas a las solicitudes hechas por las maquinas cliente. Para elegir un servidor de base de datos se deben considerar varios aspectos que mejoren el rendimiento de los datos. El servidor de base de datos es la más valiosa posesión de una empresa porque es el que controla la información de la organización. Por lo que el elegir el tipo de servidor requiere de conocimientos de computación, ingeniería y electrónica, además de tener presente los objetivos y alcances de la empresa.

Ya que existen diversos fabricantes dedicados a la venta de servidores, tales como Digital, IBM, Hewlett Packard, Compaq, etc. La tarea de seleccionar un servidor es complicada en el sentido de la cantidad de información que es necesario recopilar para determinar si alguna elección realizada satisface las necesidades. Algunos de los factores a considerar para la selección de un servidor son:

- Precio/Desempeño
- Compatibilidad
- Relación Óptima Costo/Desempeño
- Crecimiento
- Flexibilidad
- Productividad
- Soporte Técnico
- Interoperabilidad
- Capacidad de Interconexión con otros sistemas

Cada empresa dará mayor peso a determinados factores, lo cual estará en función de sus recursos económicos, expectativas de funcionalidad, desempeño y crecimiento.

4.1.5. SOFTWARE PARA EL CLIENTE/SERVIDOR

SELECCION DE SISTEMAS OPERATIVOS

La selección de sistemas operativos para el servidor depende en parte de la elección del RDBMS y de las necesidades de la organización, así como de sus posibilidades económicas. En el mercado existen opciones como UNIX, Windows NT y OS/2 cada uno cumpliendo con los conceptos de interconectividad con características propias.

En cuestión al cliente, la plataforma de desarrollo básica en nuestros días es el ambiente Windows, y el sistema operativo MS-DOS 6.x, Windows 3.x y Windows 95 permiten manejar éste ambiente.

SELECCION DEL FRONT END

Se deben buscar ciertas características en el front end que satisfagan las exigencias de los usuarios, por ejemplo:

- Diseño de Base de Datos.
- Capacidad de Programación.
- Diseño de Archivos.
- Diseño de formas de Presentación.
- Enlaces entre formas y datos.
- Reportes.
- Comunicación con el RDBMS.

En el mercado se encuentra el siguiente software: Sql Windows, Centura, Visual Basic, Visual C, Paradox, Power Builder, Delphi, Superbase.

SELECCION DEL RDBMS

Un software especializado como lo es el RDBMS debe soportar el acceso a los datos en diferentes plataformas, incluyendo mainframes, minis y microcomputadoras, así como la comunicación con otros RDBMS iguales o distintos a él. Para ello, debe cubrir las siguientes características:

- *Interoperabilidad*: Posibilidad de integrar herramientas de software de otros fabricantes.
- *RDBMS inteligente*: Incluya integridad en la base de datos y sea altamente eficiente.
- *Alto desempeño*: Satisfaga las expectativas del consumidor en la relación costo-beneficio.
- *Tecnología Probada*: La tecnología de software incorporada deberá certificar su correcto funcionamiento.
- *Escalabilidad*: Evitar la obsolescencia por medio de la actualización de los componentes del sistema.

SELECCION DEL MIDDLEWARE

El software de comunicación de acceso a la base de datos es importante determinarlo. De acuerdo con el front end y el RDBMS los proveedores ofrecen varias alternativas de conexión, algunas son propias del RDBMS, otras como ODBC son más abiertos. En este caso se deberá hacer un estudio de cuál middleware proporciona mayor rapidez de acceso a la base de datos en condiciones de mayor concurrencia y transferencia de información con otras bases de datos.

4.1.6. UBICACIÓN DEL EQUIPO Y ESPECIFICACIONES ELÉCTRICAS

Para brindar un adecuado esquema de funcionamiento, la instalación eléctrica del local deberá ser diseñada para tal efecto. De lo anterior la distribución de control reviste gran importancia para la seguridad de los equipos, así mismo las condiciones del local deberán estar adecuadas para el trabajo del personal con equipos de cómputo, esto es la utilización de sistemas de ventilación y el control de luz interior.

4.1.7. COSTOS

El aspecto de obtención de recursos financieros en la construcción de una arquitectura C/S es muy importante, siendo necesario realizar un estudio de mercado con la finalidad de obtener un servicio adecuado, así como garantías de actualización, etc., que prolonguen la duración de hardware y software, y con ello la inversión.

4.1.8. INTEGRACIÓN DE PLANES DE TRABAJO

Este reporte incorpora los planes de trabajo de Arquitectura Técnica e Instalaciones Físicas, para que en coordinación con los proveedores se establezcan planes que permitan obtener resultados parciales de la instalación. También se diseña el ambiente de configuración, operación y administración de la arquitectura Cliente/Servidor. Algunas de las actividades que dan soporte al proyecto y promueven la efectiva utilización de recursos: gente, software y hardware son:

ACTIVIDAD DE INFRAESTRUCTURA	DEFINICION
Afinación a la Red	Define como el rendimiento de la red será probado, monitoreado y ajustado a los perfiles para el soporte de los recursos necesarios.
Seguridad	Define los permisos a accesos a interfaces, archivos de datos y ejecución de procedimientos.
Entrenamiento	Define programas de capacitación para la administración de la infraestructura.
Conversión y migración del plan	Define cómo algunos datos de conversión serán ejecutados.
Pruebas	Define cómo y cuándo la aplicación debe ser probada. Provee de un plan para describir escenarios de prueba, scripts y criterios. Detallan objetivos para probar la interfaz de usuario. Define los procedimientos y roles para quienes conducirán las pruebas.
Control/Configuración del manejo	Define como los cambios serán sostenidos. Detalla la estrategia de migración para el movimiento de los componentes del software. Define las plataformas de software y hardware necesarias para configurar el manejo. Define mecanismos de control y monitoreo.
Afinación de la administración de la base de datos	Quién, dónde y cómo se ejecuta el diseño físico de la base de datos, la instalación y su configuración.
Ayuda de escritorio y soporte	Define que soporte técnico será proporcionado a los usuarios. Determina la disponibilidad, los costos, y localización del soporte.

DISEÑO DE SISTEMAS CLIENTE/SERVIDOR

4.1.9. CRITERIOS DE DISEÑO

- ✓ **Tener Capacidad:** La *capacidad* es la habilidad que tiene el sistema actual formado por personas, equipo, espacio y procedimientos, para alcanzar las metas y objetivos básicos del sistema. Un sistema debe tener capacidad para manejar las transacciones y cumplir

con los requerimientos de procesamiento y satisfacer los niveles de rendimiento esperados.

- ✓ **Tener Control:** El *control* permite que el sistema detecte y notifique los casos donde las actividades no se realizaron. También es esencial el manejo de eventos no anticipados. Un sistema bien diseñado debe tener la capacidad de notificarlos sin interrumpir las demás actividades. Muchos sistemas hacen hincapié en el control de las tareas, notificando cuándo no se realizan en forma apropiada. El control sobre el contenido se enfoca hacia los datos. Los datos incorrectos conducen hacia información poco confiable.

Desde el punto de vista del diseño de sistemas C/S, las siguientes estrategias se encuentran entre las que deben considerarse:

- Diseño para *evitar* fallos en el control.
- Diseño para *detectar y notificar* problemas de control.
- Diseño para *detectar y corregir* fallos en los controles.

- ✓ **Accesibilidad de la Información:**

- *Facilitar el acceso a la información:* Los sistemas se pueden diseñar para que respondan a mandatos y dependan de opciones preestablecidas que se ajusten a las situaciones que se presentan con mayor frecuencia
- *Eliminar la necesidad de procesamiento:* Los detalles utilizados con mayor frecuencia pueden almacenarse en una forma que no requiera procesamiento o manejo. En este caso mejoran tanto la disponibilidad como la accesibilidad.
- *Cambiar el método de presentación:* Se pueden seleccionar el empleo de otros formatos, incluyendo gráficas, colores e incluso técnicas de animación.

- ✓ **Sencillez:** La elegancia de la sencillez es un atributo de los sistemas C/S mejor diseñados. Los procedimientos que contienen un número muy grande de tareas, pasos o actividades, a menudo dan como resultado un rendimiento inaceptable o incompleto. Para reducir la complejidad se deben considerar tres estrategias:

- *Simplificación:* La simplificación puede obtenerse al eliminar los pasos que no son necesarios o el registro de información que no se utiliza. El uso de estándares preestablecidos simplifica los procesos. Los casos que no se ajustan a lo establecido por los estándares pueden manejarse como excepciones.
- *División:* Dividir un proceso complejo en tareas separadas disminuye con frecuencia la complejidad.

➤ *Cambios en la secuencia:* Cambiar el orden en el que ocurre un proceso puede disminuir la complejidad aparente.

✓ *Cumplimiento Total de los Requerimientos:* A menudo el analista del sistema sugiere procedimientos para entrada, procesamientos, generación de reportes y de control con el objeto de guiar las operaciones o la toma de decisiones. El analista tiene la responsabilidad de sugerir dónde serán de utilidad los procedimientos. La administración decide si acepta o rechaza dichas recomendaciones. Tanto el analista como el diseñador toman la decisión al aplicar el conocimiento que tiene sobre las capacidades de la computadora y combinarlo con la comprensión de los requerimientos de sistemas.

4.1.10. OBJETIVOS DE DISEÑO

ESPECIFICACION DE LOS ELEMENTOS LOGICOS DEL DISEÑO

El diseño de sistemas C/S tiene dos etapas: el diseño lógico y la construcción física del sistema.

Cuando se formula el *diseño lógico*, escribe las especificaciones detalladas del nuevo sistema, es decir aquellas que describen sus características: salidas, entradas, archivos y bases de datos, y los procedimientos, todo en una forma que satisfaga los requerimientos del proyecto.

El diseño lógico de un sistema de información, incluyen definiciones de reportes y pantallas de presentación, especifica los formatos de entrada y la distribución de la salida en pantalla para todas las transacciones y archivos que son necesarios para dar mantenimiento a los datos. Las especificaciones de procedimientos describen los métodos utilizados para ingresar datos en el sistema, copiar archivos y detectar problemas, si éstos se presentaran.

El *diseño físico* es la siguiente actividad después del diseño lógico, produce el software, los archivos y un sistema que funciona. Las especificaciones de diseño indican a los programadores lo que el sistema debe hacer y generar el primer producto de la aplicación llamado *prototipo*, que será evaluado por los usuarios y si es aceptado se continuará hasta terminar e implantar el nuevo sistema C/S.

El diseño físico, está formado por instrucciones de programa, escritas en una herramienta de desarrollo de sistemas. Para el sistema SISEFIN se ocupo la herramienta Superbase. El lider de proyecto especifica los algoritmos necesarios. Durante la construcción física, los programadores escriben las instrucciones necesarias del sistema y producir los resultados.

APOYO PARA LAS ACTIVIDADES DE LA EMPRESA

Un objetivo fundamental en el diseño de un sistema C/S es asegurar que éste brinde apoyo a la actividad de la empresa para la que esta siendo desarrollado. El diseño tiene que ajustarse a lo que requiere la empresa, aunque a veces se tengan que modificar los procesos actuales, por medio de la reingeniería de procesos.

ASEGURAR QUE LAS CARACTERISTICAS DEL SISTEMA CUMPLA CON LOS REQUERIMIENTOS DEL USUARIO

Se dice que un sistema de información satisface las necesidades de los usuarios si: ^[29]

- Realiza en forma apropiada los procedimientos correctos.
- Presenta información e instrucciones en una forma aceptable y efectiva.
- Produce resultados exactos.
- Proporciona una interfase y métodos de interacción aceptables.
- Es percibido por los usuarios como un sistema confiable.

Los sistemas de información son una herramienta importante para la empresa. Por tanto, es esencial adecuar el sistema a la organización. Los líderes de proyecto más eficientes son los que comprenden las razones del éxito de la organización y crean sistemas de información que vayan al paralelo con estos propósitos.

PROPORCIONAR UN SISTEMA QUE SEA FACIL DE UTILIZAR

Para lograr un sistema amigable se procura diseñar con ingeniería dirigida hacia las personas y, en consecuencia, incluyen características ergonómicas.

Ingeniería humana: En general, el diseñador de la interfase debe procurar formular el diseño en forma que incorpore características del sistema que sean fáciles de comprender y utilizar. Entre algunas consideraciones es de que el sistema tenga suficiente flexibilidad para adaptarse a las necesidades de cada usuario en una forma que parezca natural y evite fallas o procedimientos inapropiados que generen perjuicios o complicaciones para los usuarios.

Diseño ergonómico: En el contexto de los sistemas de información, la ergonomía estudia los factores físicos que afectan el rendimiento, la comodidad y la satisfacción de los usuarios directos. Se deben considerar factores ergonómicos cuando se escogen los colores para la presentación de información, la ubicación de las teclas de mandatos, o los métodos de interacción con el sistema.

^[29] Senn, James.- *Análisis y Diseño de Sistemas de Información.- Traduc. Urbina Medal Edmundo Gerardo y otros.- 2ª Edición.- Editorial McGRAW-HILL.- México.- 1994 - pp. 382.*

PROPORCIONAR ESPECIFICACIONES DETALLADAS PARA EL DESARROLLO DE SOFTWARE

El diseño de sistemas incluye la formulación de las especificaciones de software. Estas especificaciones establecen las funciones de entrada, salida y procesamiento así como los algoritmos necesarios para efectuarlas. Los módulos de software, se enfocan sobre lo que cada función realiza, se especifican los procedimientos necesarios para llevar a cabo dichas funciones. Y las recomendaciones de infraestructura para que el sistema tenga un mayor rendimiento.

AJUSTARSE A LOS ESTANDARES DE DISEÑO

Las especificaciones de diseño se establecen dentro del marco fijado por los estándares. Los siguientes son ejemplos de áreas incluidas en el diseño de estándares:

- *Estándares para datos*: Lineamientos para asignar nombre a los datos y especificar su longitud y tipo.
- *Estándares estructurales*: Lineamientos sobre cómo estructurar el sistema. Políticas para dividir el software en módulos, para la codificación estructurada u orientada a objetos y la relación existente entre los componentes del sistema.
- *Estándares de documentación*: Descripciones de las características del diseño de sistemas, de la relación entre componentes y de las características de operación que pueden ser revisadas para conocer los detalles de la aplicación.

Con el objeto de garantizar que la aplicación cumpla con los estándares, muchas organizaciones cuentan con un grupo de control de calidad que tiene la responsabilidad de revisar todas las especificaciones de diseño de sistemas de información, así como el propio sistema una vez terminado.

4.1.11. CARACTERÍSTICAS QUE SE DEBEN DISEÑAR

Las especificaciones de diseño describen las características del sistema, los componentes o elementos del sistema C/S y la forma en que éstos aparecerán ante los usuarios.

Los líderes de proyecto y/o analistas deben diseñar los siguientes elementos: ^[30]

- *Flujos de datos*: Movimientos de datos hacia, alrededor y desde el sistema.
- *Almacenes de datos*: Conjuntos temporales o permanentes de datos.
- *Procesos*: Actividades para aceptar, manejar y suministrar datos e información.

^[30] Senn, James.- *Análisis y Diseño de Sistemas de Información*.- Traduc. Urbina Medal Edmundo Gerardo y otros.- 2ª Edición.- Editorial McGRAW-HILL.- México.- 1994 - pp. 385,386.

- *Procedimientos*: Métodos y rutinas para utilizar el sistema de información y lograr con ello los resultados esperados.
- *Controles*: Estándares y lineamientos para determinar si las actividades están ocurriendo en la forma anticipada o aceptada. Asimismo, debe especificar las acciones que tienen que emprenderse cuando ocurren problemas o se presentan circunstancias inesperadas.
- *Funciones del personal*: Las responsabilidades de todas las personas que tienen que ver con el nuevo sistema, incluyendo los usuarios, operadores de computadora y personal de apoyo. Abarca todo el espectro de componentes del sistema, incluso desde la entrada de datos hasta la distribución de salidas o resultados.

4.1.12. REGLAS DE NEGOCIO

Las reglas de negocio son las restricciones, pruebas o condiciones que describen al negocio. Son la descripción de las políticas de trabajo dentro de los alcances contemplados por el sistema. Estas se expresan en lenguaje natural y pueden ser encontradas escritas en documentación, examinando los procesos de la empresa o en otras formas tales como gráficas, organigramas y tablas.

Cada modelo puede representar múltiples reglas de negocio. El modelo de procesos considerará las reglas, así como fórmulas para derivar la información y validación lógica. El modelo de datos o diagrama entidad - relación contemplará las reglas para la definición de entidades, relaciones y derivación de atributos, y dependencias. Igualmente las reglas son definidas durante el desarrollo del prototipo.

Eventualmente las reglas se convertirán en el código de la aplicación. Estas reglas reforzarán la integridad referencial (mantenimiento de relaciones entre entidades) pudiendo tomar la forma de código (así como triggers en la base de datos), o pueden estar sostenidos el DBMS (por ejemplo, declarar la integridad referencial).

4.1.13. PARTICIPACIÓN DE LOS USUARIOS

El diseño de un sistema de información no sólo es responsabilidad de analistas y diseñadores. La participación de los usuarios proporciona al analista una retroalimentación que es importante a medida que avanza el diseño. Existen tres razones para que los usuarios participen en el diseño de un sistema. La primera de ellas es que el sistema pertenece a los usuarios, no a los líderes de proyecto. La segunda razón para que el usuario participe es la detección de omisiones y de características innecesarias o difíciles de emplear. La tercera razón es que cualquier error tanto en los procedimientos, debe detectarse lo más pronto posible. Descubrir errores una vez que el sistema está implantado provoca dudas por parte de los usuarios, aun después de corregirlos.

Se puede lograr una buena participación de los usuarios durante el diseño al pedirles que revisen los prototipos, formatos de entrada y los reportes. Los usuarios también son de gran ayuda para los métodos de prueba, el empleo de menús y en la evaluación de la secuencia de mandatos. Tener un punto de vista fresco de un usuario que no ha utilizado la interfase con anterioridad puede llevar a cambios productivos en las especificaciones, con lo que se evitan dificultades futuras.

Cuando existe un número muy grande de usuarios para un sistema que abarca a toda la organización, se deben seleccionar a usuarios en potencia que tengan bastante experiencia para que participen en el diseño del sistema.

4.2. DISEÑO DE SALIDAS DEL SISTEMA

El término salida, se refiere a los resultados e información generados por el sistema. En la realidad, muchos usuarios no operan el sistema de información y tampoco ingresan datos en él, pero utilizan la salida generada por el sistema. Cuando se diseña la salida, se debe realizar lo siguiente: ^[1]

- Determinar qué información presentar.
- Decidir si la información será presentada en forma visual, verbal o impresa y seleccionar el medio de salida.
- Disponer la presentación de la información en un formato aceptable.
- Decidir cómo distribuir la salida entre los posibles destinatarios.

4.2.1. IDENTIFICACIÓN DE LAS NECESIDADES DE SALIDA DEL SISTEMA

El término *salida* se utiliza para denotar cualquier información producida por un sistema C/S, ya sea impresa o en una pantalla (un reporte, un documento o un mensaje). Cuando los analistas de sistemas diseñan la salida, ellos identifican la salida específica que es necesaria para satisfacer los requerimientos de información. No se puede clasificar como "buena" una salida estéticamente atractiva o que haga uso de una nueva tecnología a menos que satisfaga las necesidades de la organización y de sus usuarios.

En el desarrollo de prototipos de aplicaciones, a menudo se desarrollan pantallas de salida para permitir que los usuarios reaccionen tanto al contenido como a la forma de la salida. El objetivo principal durante el diseño de la salida de la computadora es la información que será presentada a las personas.

El verdadero reto en el diseño de la salida de computadora no es cuánta información proporcionar sino cuál es el mínimo necesario para poner a disponibilidad información

[1] Senn, James.- *Análisis y Diseño de Sistemas de Información.*- Traduc. Urbina Medal Edmundo Gerardo y otros.- 2ª Edición.- Editorial McGRAW-HILL.- México.- 1994 - pp. 386.

importante. El desarrollo de cualquier diseño para la salida debe dirigirse hacia aquello que los usuarios necesitan saber.

4.2.2. PRESENTACIÓN DE LA INFORMACIÓN

La forma en que se presenta la información determinará si la salida es clara y comprensible, si los detalles son convincentes y si la toma de decisiones se efectúa con mayor rapidez y exactitud. La información puede ser presentada se acuerdo con su propósito, en:

- Formato Tabular.
- Formato Gráfico.
- Uso de iconos.
- Presentaciones en color.

4.2.3. DISEÑO DE SALIDA IMPRESA

El diseñador debe buscar el empleo de sólo aquellas salidas impresas que son absolutamente necesarias. Los tamaños y estilos adicionales en la tipografía, más el empleo de tinta de color, logotipos y otros símbolos especiales asociados con la compañía, mejoran la apariencia del informe, así como el mensaje que éste envía a los lectores (Figura 4.1).

 <p style="text-align: center;">SUBDIRECCIÓN GERENCIA</p> <p style="text-align: center;">REGION: REGION DISTRITO: DISTRITO</p>	<p style="text-align: center;">SISTEMA DE SEGUIMIENTO FINANCIERO</p> <p style="text-align: center;">Subproyecto</p> <p>Fecha: 23/07/97 Hora: 5:29 am</p> <p>Ordenado por: CLAVE DE SUBPROYECTO</p> <p style="text-align: right;">Página: 1</p>																		
<p>AGRUPADO POR: CLAVE DEL PROGRAMA</p>																			
<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th style="width: 10%;">SECCION</th> <th style="width: 10%;">CONCEPTO</th> <th style="width: 10%;">SUBPROYECTO</th> <th style="width: 10%;">REACTIVACION</th> <th style="width: 10%;">PROGRAMA</th> <th style="width: 10%;">PROYECTO</th> </tr> </thead> <tbody> <tr> <td>EFESO</td> <td>FEERESB</td> <td>MALESD</td> <td>AREIL</td> <td>ALAYO</td> <td>JIRMO</td> </tr> <tr> <td>JULIO</td> <td>AGOSTO</td> <td>SEPTIEMBRE</td> <td>OCTUBRE</td> <td>NOVIEMBRE</td> <td>DICIEMBRE</td> </tr> </tbody> </table>		SECCION	CONCEPTO	SUBPROYECTO	REACTIVACION	PROGRAMA	PROYECTO	EFESO	FEERESB	MALESD	AREIL	ALAYO	JIRMO	JULIO	AGOSTO	SEPTIEMBRE	OCTUBRE	NOVIEMBRE	DICIEMBRE
SECCION	CONCEPTO	SUBPROYECTO	REACTIVACION	PROGRAMA	PROYECTO														
EFESO	FEERESB	MALESD	AREIL	ALAYO	JIRMO														
JULIO	AGOSTO	SEPTIEMBRE	OCTUBRE	NOVIEMBRE	DICIEMBRE														

Figura (4.1) Salida en Impresa

4.2.4. DISEÑO DE SALIDA EN PANTALLA

El diseño de la pantalla (Figura 4.2) comienza con la verificación de las características de presentación visual. Entre éstas se incluyen las siguientes:

- Dimensiones físicas de la pantalla.
- Número de colores disponibles.
- Métodos de realce (subrayado, negritas, recuadros, parpadeo, diferentes intensidades).

Sistema de Seguimiento Financiero " SISEFIN "

Datos:

PEMEX

PARTIDAS MOMENTO: 7313 NO. PARTIDA: 1

No. DE CIP: 5G97001 FECH. DEV: 1/1/97 COMPAÑIA: _____
 No. DE FACT: _____ FECH. DOCTO: 23/07/97

SUBPROYECTO	_____	Unidad Contable	_____
PROGRAMA	_____	Ctro. de Trabajo Afectador	_____
SUBPROGRAMA	_____	Departamento Afectador	_____
PROYECTO	_____	Ctro. de Trabajo Afectado	_____
PROY-OBRA	_____	Departamento Afectado	_____
GUIÓN	_____		
REGLÓN	_____		
CONCEPTO	_____		

JUSTIFICACION: _____

I.V.A. _____ %

MONTO CAP: \$ _____ MONTO \$ _____ IVA \$ _____ TOTAL \$ _____

Figura (4.2) Salida en Pantalla

La forma en que se estructura el área física de un monitor, así como los métodos particulares para destacar y señalar datos pueden mejorar la posibilidad de lectura de la información mostrada. La facilidad de lectura debe ser uno de los objetivos que sirva de guía en el diseño de la salida de los sistemas.

El diseño de salida del sistema SISEFIN se basó en el formato de salida de la anterior aplicación, en cuestión al aspecto visual, aunque la distribución tendió a cambiar a excepción de algunos catálogos. Además se tuvo el cuidado de que todas las pantallas cumplieran con el estándar de tamaño y fuente de letras.

4.3. DISEÑO DE ENTRADAS Y CONTROLES DEL SISTEMA

4.3.1. DISEÑO DE ENTRADAS

Las especificaciones de *entrada* describen la manera en que los datos ingresarán al sistema para su procesamiento. Las características de diseño de la entrada pueden producir resultados a partir de datos exactos.

El diseño de la entrada es el enlace que une al sistema de información con sus usuarios.

4.3.2. OBJETIVOS DEL DISEÑO DE ENTRADAS

Los objetivos del diseño de la entrada consisten en el desarrollo de procedimientos para la preparación de datos, la realización de los pasos necesarios para poner los datos en una forma utilizable para su procesamiento. Disminuir los requerimientos de datos puede reducir los costos y acelerar todo el proceso desde la captura de datos hasta que los resultados llegan a manos de los usuarios.

Los diseñadores de sistemas deciden los siguientes detalles del diseño de entradas:

1. Qué datos ingresan al sistema.
2. Qué medios utilizar.
3. La forma en que deben disponer o codificar los datos.
4. El diálogo que servirá de guía a los usuarios para dar entrada a los datos.
5. Validación necesaria de datos y transacciones para detectar errores.
6. Métodos para llevar a cabo la validación de las entradas y los pasos a seguir cuando se presentan errores.

La disposición de mensajes y comentarios en las conversaciones, así como la ubicación de los datos, encabezados y títulos sobre las pantallas o documentos fuentes, también forman parte del diseño de entradas.

4.3.3. DISEÑO DE CONTROLES

Los controles de entrada proporcionan medios para asegurar que sólo los usuarios autorizados tengan acceso al sistema, garantizar que las transacciones sean aceptables, validar los datos para comprobar su exactitud y determinar si se han omitido datos que son necesarios.

VALIDACION DE LA ENTRADA

Los diseños de las entradas tienen como finalidad reducir las posibilidades de cometer errores o equivocaciones durante la entrada de datos. Estos deben detectarse durante la captura y corregirse antes de guardar los datos o procesarlos. El término general dado a los métodos cuya finalidad es detectar errores en la entrada es *validación de entradas*^[32] Figura (4.3).

[32] Senn, James.- *Análisis y Diseño de Sistemas de Información.- Traduc. Urbina Medal Edmundo Gerardo y otros.- 2ª Edición.- Editorial McGRAW-HILL.- México.- 1994 - pp. 497.*

The screenshot shows the SISEFIN application window with the title 'Sistema de Seguimiento Financiero SISEFIN'. The main area contains the FEMEX logo and the text 'DOCUMENTOS NO DEVENGADOS'. Below this, there are input fields for 'NO. DE CIP: 5G97001', 'NO. DE CIF: []', 'FECHA DE CAPTURA: 23/07/97', and 'FECHA DEVENGADO: 7/7'. A modal dialog box is open with the title 'MOMEN SISEFIN' and the message: 'EXISTE UN CAMPO NULO Y NO PUEDE QUEDAR VACIO POR FAVOR, VERIFIQUE'. An 'Aceptar' button is visible in the dialog. Below the dialog, there are several input fields labeled 'CLAVE', 'JUSTIFICACION', 'FICHA', and 'NOMBRE', along with a 'RESPONSABLE' field and a 'CAP' button.

Figura (4.3) Validación de Entrada

Validación de transacciones:

Lo primero es identificar todas las transacciones que no son válidas. La transacción debe ser aceptable para el sistema antes de que pueda ser procesada por él.

Prueba de secuencia: Mientras no dé el dato correcto no podrá proseguir con la captura o el procesamiento. Estas son las pruebas de completes, otra forma de validar la transacción.

Verificación de los datos de la transacción:

Aun las transacciones válidas pueden contener datos que no lo son. Por consiguiente, se debe especificar métodos para validar los datos cuando se desarrollen los procedimientos de entrada.

- *Pruebas de existencia campos no Nulo.*
- *Pruebas de límites y rangos.*
- *Procesamiento no duplicado.*
- *Corrección automática:*

En ocasiones puede haber excepciones, permitiendo que el dato esté fuera de la restricción y sea guardado, siendo necesario que el líder de proyecto recalque que es responsabilidad por parte del usuario que omita el mensaje.

4.4. DISEÑO DE DIALOGO

4.4.1. DEFINICIÓN

El diálogo conduce a la interacción entre el sistema y el usuario (Figura 4.4). El diálogo correcto puede hacer que la frontera entre usuario y sistema parezca inexistente.

Un *diálogo* es la forma en la que el usuario interactúa con el sistema de cómputo. Sus características determinan lo "amigable" del sistema, influyen en la decisión de una persona de usar el sistema.

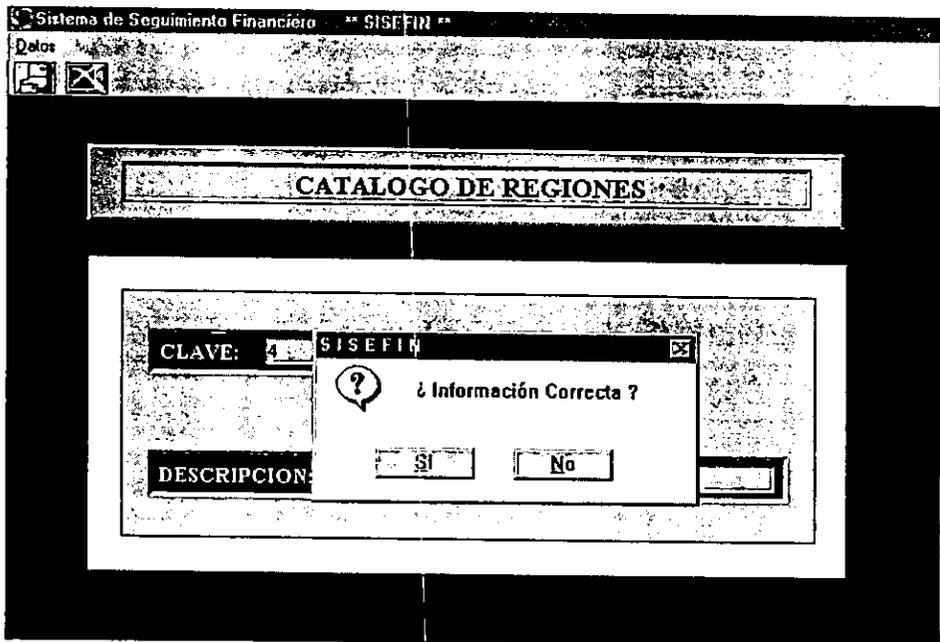


Figura (4.4) Diálogo

4.4.2. RECEPCIÓN DE MENSAJES

Una parte importante del diseño es la comunicación de mensajes entre el sistema C/S y el usuario. Las personas desean saber cuándo iniciar o tomar acciones, el estado de ciertos eventos y actividades, y cuándo se ha terminado una tarea. Por supuesto, también necesitan saber cuándo ocurren errores e interrupciones de procesamiento.

Por ejemplo si una pantalla se pone en blanco durante el procesamiento de la transacción, debe asegurarse de que aparezca el mensaje en el cual informe al usuario que la computadora está trabajando. Para el procesamiento que se tarde periodos largos, se debe hacer que la pantalla dé al usuario un mensaje en el que se dice aproximadamente cuánto tiempo se tardará y dando una opción para cancelar la petición. El usuario puede decidir correr ese trabajo en otro momento cuando el sistema no este tan ocupado. Otro método para periodos cortos de procesamiento es cambiar el apuntador del mouse y representa un mensaje gráfico de que se esta ejecutando un proceso.

4.4.3. DECISIONES EN EL DISEÑO DE DIÁLOGO

Estas decisiones, determinan la naturaleza del diálogo:

1. *Estrategia general del diálogo*: Opciones de menús y barra de iconos. Son una lista de funciones disponibles en el sistema de modo que el usuario pueda elegir entre ellas.
2. *Diálogo de entrada de datos*: Formatos de entrada de datos, respetando un orden lógico y que corresponda con el orden natural en el que la gente esté acostumbrada a la información o como viene en el documento fuente.
3. *Mensajes y comentarios*: Los mensajes son la forma de comunicación del sistema con los usuarios cuyo objetivo es informar. Los mensajes tienen alguna de las siguientes finalidades: indicar el estado del procesamiento o transacción (Mensajes de estado), indicar que se ha detectado un error (Mensajes de error), solicitar que el usuario elija una acción (Mensajes de solicitud de acciones), verificar que una acción elegida sea correcta (Mensajes de verificación de acciones). Un mensaje debe ser conciso, suficiente y comprensible.
4. *Navegación del usuario*: Manejo adecuado en el movimiento entre formas, cajas de dialogo, menús, u otras pantallas para pasar a una o retroceder a otra.
5. *Asignación de teclas*: Es usual que se programen teclas de propósito especial para algunas funciones
6. *Sistemas de ayuda*: Tienen como objetivo instruir al usuario acerca del sistema. Las funciones de ayuda resuelven problemas que el usuario tiene en cierto momento al operar el sistema.

4.5. DISEÑO DE PROCEDIMIENTOS

4.5.1. DEFINICIÓN

Los *procedimientos* especifican qué tareas deben efectuarse al utilizar el sistema C/S y quiénes son los responsables de llevarlas a cabo.

4.5.2. TIPOS DE PROCEDIMIENTOS

Los procedimientos deben formularse por escrito y formar parte de la documentación del sistema. Entre los procedimientos importantes se encuentran

- ✓ *Procedimientos para entrada de datos.*
- ✓ *Procedimientos durante la ejecución.*
- ✓ *Procedimientos para el manejo de errores.*
- ✓ *Procedimientos de seguridad y respaldo.*

4.6. DISEÑO DE ESPECIFICACIONES PARA PROGRAMAS

4.6.1. DEFINICIÓN

Las especificaciones para programas son la estructuración de software en módulos que permita su prueba y validación.

En el sistema SISEFIN, se retomó el código de la vieja aplicación en cuanto al proceso de altas, bajas y cambios, y se convirtieron en procedimientos generales, creándose lo que se conoce como librerías, utilizables para cada uno de los módulos del sistema.

Independientemente de las herramientas utilizadas para escribir el software, el tener una buena estructuración de programación, asegura que las modificaciones futuras se puedan realizar en forma eficiente y fácil.

4.7. DISEÑO DE BASE DE DATOS

4.7.1. DEFINICIÓN

El diseño de la base de datos incluye decisiones con respecto a la naturaleza y contenido de la propia base de datos. Entre las decisiones que se toman durante el diseño, se encuentran las siguientes:

- Los datos que deben incluirse en cada tabla.
- La longitud de cada registro, con base en las características de los datos que contiene.

- Las propiedades de cada tabla, como llaves primarias, campos únicos, requeridos y no requeridos, restricciones por campo o tabla, privilegios, etc.
- Las relaciones entre las tablas, determinando llaves foráneas.

Algunos nuevos sistemas C/S requieren la utilización de otras bases de datos, y por ende de otras tablas que están diseñadas y construidas para aplicaciones ya existentes, haciendo sólo referencia a éstas para la extracción de datos, tal es el caso del sistema SISEFIN, que en la parte de catálogos requiere la información de otra base de datos (Figura 4.6).

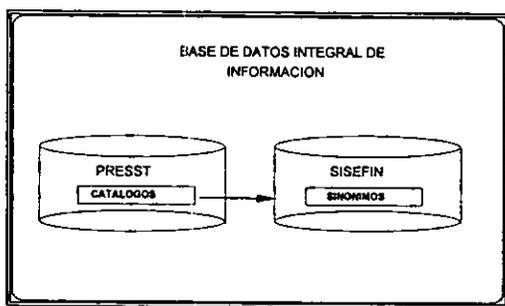


Figura (4.5) Relación de dos bases de datos

El diseño de las bases de datos se divide en el modelo lógico de la base de datos y en su diseño físico.

4.7.2. MODELO LÓGICO DE DATOS

El modelo lógico es abstracto, esto significa que es independiente de la base de datos o del hardware. Para el diseño de este modelo intervienen los siguientes conceptos:

NOMBRES CONVENCIONALES

El propósito de establecer nombres convencionales es crear un punto de referencia consistente a la comunidad de usuarios y grupo de trabajo del proyecto. Los nombres de todo tipo de objetos deben ser simples y significativos. Evitar el uso de acrónimos a menos que la mayoría este familiarizada.

Es importante crear una consistencia en los nombres de los atributos. El crear una lista de todas las palabras y estandarizaciones de abreviaciones debe estar disponible para todos los que integran el sistema.

ENTIDADES

Una entidad es un objeto o cosa de relevancia, y es usualmente un sujeto. Uno de los primeros pasos en el modelo lógico de la base de datos es identificar las entidades que se necesitan para el soporte de los datos requeridos del sistema.

Un **supertipo** es una entidad que contiene otras entidades llamadas subtipos. Los supertipos contiene atributos que son comunes, y los subtipos contienen atributos que son específicos. Por ejemplo, se define una entidad cliente como un supertipo con algunos atributos básicos. Entonces dentro de la entidad cliente se define, la entidad pedidos como subtipo, conteniendo atributos que soportan y describen los pedidos que hizo cada cliente.

ATRIBUTOS

Un atributo describe la entidad o mejora la información acerca de la entidad. Después de que las entidades sean definido, se identificará todos los atributos de cada entidad. Se especifica el tipo de datos, su longitud máxima y si es o no un valor requerido. Las columnas de llaves foráneas no son atributos, sólo son derivaciones de datos representadas dentro del modelo de datos lógico como relaciones.

DEPENDENCIA

Las relaciones entre entidades se describen mediante su *dependencia* uno de la otra, al igual que por el alcance de la relación. Existen dos tipos de dependencia entre entidades.

En la primera, la *dependencia existencial*, una entidad no puede existir a menos que la otra esté presente; el que exista la segunda depende de la existencia de la primera. Al eliminar los registros de una entidad en una base de datos puede ocurrir que se eliminen los registros de otra si existe una dependencia existencial.

En el otro tipo de dependencia, la *dependencia de identificación*, una entidad no puede identificarse de manera única con sus propios atributos. La identificación es posible sólo mediante las relaciones de una entidad con otras. Para identificar una identidad, se deben conocer las otras. La *dependencia no identificada*, una entidad puede identificarse de manera única con sus propios atributos y sólo hace referencia a los atributos de la otra entidad para relacionar datos.

IDENTIFICADORES UNICOS (UID - UNIQUE IDENTIFIERS)

Cada entidad debe ser identificada por uno o más de estos atributos.

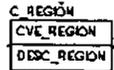
NORMALIZACIÓN

Durante el proceso del modelo lógico, es importante entender la teoría relacional y su aplicación práctica. La normalización es el proceso de simplificar la relación entre los campos de un registro. La normalización se lleva a cabo por cuatro razones:

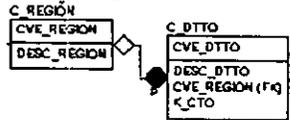
- Estructurar los datos de forma que se puedan representar las relaciones pertinentes entre los datos.
- Permitir el acceso sencillo de los datos en respuesta a las solicitudes de consultas y reportes.
- Simplificar el mantenimiento de los datos actualizándolos, insertándolos y borrándolos.
- Reducir la necesidad de reestructurar o reorganizar los datos cuando surjan nuevas aplicaciones.

La normalización cumple con tres formas:

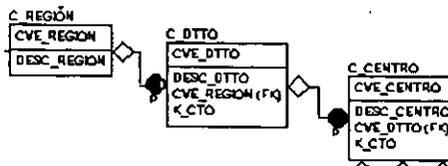
Primera Forma Normal: Todas las entidades tiene una llave primaria y se quitan todos los grupos de repetición. Esto es que las entidades deben tener un identificador único para uno o mas atributos, el cual no deber ser nulo. Cuando más de un identificador único existe, uno de estos debe nominarse como llave primaria. Además de que el mismo atributo no puede existir dentro de una misma entidad más que una sola vez.



Segunda Forma Normal: Todos los atributos dependen de una llave. Los atributos asociados con la entidad deben ser directamente dependientes de una llave primaria para que su valor tenga significado.



Tercera Forma Normal: Los atributos de una entidad alcanzan una dependencia transitoria con otros atributos. Los atributos asociados con una entidad pueden tener dependencia sobre otros atributos y asociarse transitoriamente con otra entidad.



Los atributos de la entidad Centro de Trabajo dependen de los atributos de la entidad Distrito mediante su llave primaria CVE_DTTO, y los atributos de la entidad Distrito dependen de los atributos de la entidad Región mediante su llave primaria CVE_REGION, por lo tanto existe una dependencia transitoria entre los atributos de Región con los de Centro.

Un modelo de datos que no esta normalizado es diferente de un modelo desnormalizado. El modelo de datos **no normalizado** no tiene, ni cumple con normalización, apenas va ha ser normalizado, mientras que el modelo de datos **desnormalizado** fue originalmente normalizado y es proceso de deshacer algunos procesos de normalización, algunas veces por razones de rendimiento y facilidad de uso.

INTEGRIDAD

La integridad dentro de un sistema relacional significa que los datos deben ser consistentes y confiables. Hay cuatro tipos de integridad encontrados en un sistema relacional.

- **Entidad:** Un sistema tiene integridad en su entidad cuando las entidades no contiene nulos en su llave primaria.
- **Referencia:** La integridad referencial existe cuando cada llave foránea tiene referencia con una llave primaria.
- **Columna:** La integridad por columna significa que los valores dentro de las columnas conforman el tipo de datos y un formato definido.
- **Reglas de Negocio:** La integridad de reglas de negocio significa que algunas o todas las reglas definidas por el usuario se cumplen por el sistema.

RELACIONES

Las relaciones describen como una entidad es asociada con otra entidad. Las relaciones entre entidades también son llamadas *cardinalidad*. Existen cinco tipos de relaciones: ^[33]

Relaciones de Una a Una: Es raro que dos tablas compartan una clave primaria exactamente igual. En caso de ocurrir, suele ser por motivos de rendimiento o de seguridad. Por ejemplo Oracle recomienda que, cuando se utilice un tipo de datos LONG en un tabla, se almacene en una tabla aparte, relacionada con la primera, con la finalidad de mejorar el rendimiento de búsqueda.

Relaciones de Uno a Muchos: Es la más habitual y significa que un registro de una tabla está relacionado con varios registros de otra. Por ejemplo, una entidad OFICINA puede tener uno o mas representantes de ventas de la entidad REP_VENTAS. Y también es posible que exista una oficina que no tenga asignados representantes de ventas.

Relaciones de muchas a muchas: También puede ser posible que varias filas de una tabla estén relacionadas con varias filas de otra tabla. Por ejemplo, varias empresas pueden tener varios representantes de ventas y viceversa.

Relaciones mutuamente exclusivas: Se representa cuando una entidad tiene relación con una entidad u otra, pero nunca con ambas entidades.

Relaciones recursivas: Es cuando una entidad tiene relación a sí misma. Es ideal para identificar padre e hijos de una forma jerárquica.

^[33] Jenkins, Neil y otros.- *Client/Server Unleashed*.- 1ª Edición.- Editorial Sams.- E.U.A.- 1996 - pp. 319 - 320.

DIAGRAMAS ENTIDAD - RELACION

El uso de diagramas entidad relación es común para definir las relaciones y puede ser creado usando una herramienta CASE. En el diseño de la base de datos del SISEFIN se tuvo el apoyo de una herramienta CASE llamada ERwin Logic Works, donde se definieron las entidades, campos, tipo de datos, llaves primarias, índices, relaciones de cardinalidad, definición de procedimientos y disparadores (triggers). Igualmente para la reingeniería de software, se explotó esta herramienta CASE para la ejecución de ingeniería en reversa de la base de datos de la vieja aplicación, descartando y adecuando las anteriores entidades y construyendo las nuevas.

4.7.3. DISEÑO FÍSICO DE LA BASE DE DATOS

Después de que un modelo lógico esta completo, el diseño físico de la base de datos puede empezar. El *diseño físico* es la transición de un modelo lógico abstracto a la implementación física en la base de datos. En este caso es Oracle Versión 7.14, el manejador de la base de datos donde el sistema SISEFIN hará las transacciones y almacenará sus datos. La siguiente información son conceptos que se manejaron en la realización del físico de la base de datos del sistema SISEFIN.

TABLAS

Las tablas están definidas por las entidades identificadas dentro del modelo lógico de datos. Las tablas son el mecanismo de almacenamiento de los datos dentro de una base de datos Oracle. Las tablas propiedad del usuario SYS se llaman tablas del diccionario de datos (data dictionary) y proporciona un catálogo del sistema que la base de datos utiliza para gestionarse a sí misma. Para identificar todas las tablas y sinónimos se debe consultar el modelo de datos. Es necesario documentar la definición de las tablas existentes, así como cada una de las nuevas tablas.^[34]

TIPOS DE DATOS

Antes de definir las columnas de las tablas, es necesario considerar los tipos de datos que son soportados por la base de datos. En Oracle 7 los tipos de datos disponibles son:

TIPO DE DATOS	DESCRIPCION
CHAR	Campo de longitud fija, con una longitud máxima de 255 caracteres.
VARCHAR	En la actualidad, idéntico a VARCHAR2, aunque su funcionalidad puede cambiar en versiones futuras.

^[34] La mayor parte de la información técnica contenida en el punto 4.7.3. de la tesis, fue consultada en la siguiente bibliografía: *Loney, Kevin. ORACLE Manual del Administrador. - Traduc. Vallejo Pinto José Angel. - 1ª Edición. - Editorial McGRAW-HILL. - México. - 1996.*

TIPO DE DATOS	DESCRIPCION
VARCHAR2	Campo de longitud variable, con una longitud máxima de 2,000 caracteres.
DATE	Campo con una longitud de siete bytes que se utiliza para almacenar todas las fechas. La hora se almacena como parte de la fecha.
NUMBER	Columna numérica de longitud variable. Los valores permitidos son cero y los números positivos y negativos desde 1.0^{E-130} hasta $9.99..^{E125}$.
LONG	Campo de longitud variable, con una longitud máxima de 3 Gb.
RAW	Campo de longitud variable que se utiliza para datos binarios, con una longitud máxima de 2,000 caracteres.
LONG RAW	Campo de longitud variable que se utiliza para datos binarios, con una longitud máxima de 2 Gb.

COLUMNAS

Las tablas constan de un conjunto fijo de columnas, y a partir del modelo lógico de datos, se define las columnas de las tablas donde el nombre de los atributos son también los nombres de las columnas.

Cada columna tiene un nombre y unas características específicas. Las características de una columna se componen de dos partes: su tipo de datos y su longitud. Para las columnas que han sido creadas de relaciones, se usa la columna de la llave primaria de la tabla que ha sido referenciada.

Las tablas se relacionan entre ellas mediante las columnas que tienen en común. Puede utilizarse la base de datos para imponer estas relaciones por medio de la *integridad referencial* (referencial integrity). En Oracle 7, la integridad referencial se impone al nivel de la base de datos por medio de las *restricciones de las tablas*.

RESTRICCIONES

Una de las mas poderosas características de una base de datos es su capacidad de soportar las restricciones (CONSTRAINTS) sobre una tabla. La propiedad de identificar, definir y mantener la declaración de restricciones es esencial para el éxito de un proyecto.

Nombres de restricciones - Definir estándares de nombres convencionales para las restricciones es importante. El nombre de la restricción esta limitado a 30 caracteres.

Tipos de restricciones - Se definen cuatro tipos de identificación de restricciones:

- Identificadores que son llaves primarias se definen como Constraints Primary Key.
- Identificadores que no son llaves primarias pero son únicas se definen como Constraints Unique.
- Valores requeridos se definen como Constraints Not Null.
- Valores por omisión se definen como Constraints Default.
- Restricción de Valores se definen como Constraints Check.
- Relaciones se definen como Constraints Reference.

RESTRICCIONES	DEFINICION
<i>Llave primaria y única</i>	Los atributos que son identificadores únicos y nominados como llaves primarias se les asigna la restricción PRIMARY KEY. La llave primaria de la tabla es la columna o conjunto de columnas que hacen que cada fila de dicha tabla sea exclusiva. Una columna se identificada como llave primaria es NOT NULL (No Nula).
<i>Únicos</i>	Los atributos que son identificadores únicos pero no son nominados como llaves primarias se les asigna la restricción UNIQUE (exclusivo).
<i>No Nulos</i>	Los atributos que requieren de valores se les asigna la restricción NOT NULL, lo que significa que todas las filas almacenadas en esa tabla deben tener un valor en dicha columna; no puede dejarse en blanco (NULL).
<i>Omisión</i>	Los atributos que requieren de valores predeterminados se les asigna la restricción DEFAULT.
<i>Comprobación de Rango y Condición</i>	Los atributos que tienen restringido ciertos valores se les asigna la restricción CHECK. Esta restricción sirve para asegurar que los valores de una columna en concreto cumplan cierto criterio.
<i>Referencia</i>	Las relaciones entre tablas se definen como restricciones FOREIGN KEY (llave foránea) y se utilizan para reforzar la integridad referencial (referencial integrity). Una llave foránea de una tabla hace referencia a una llave primaria definida con anterioridad en cualquier otro lugar de la base de datos.

Para cada restricción de referencia se debe elegir el modo de procesamiento en cascada. Existen cinco:

1. *Restricción en cascada*: No se permitirá borrar al padre mientras existan hijos que dependan de él.
2. *Borrado en cascada*: Todos los renglones que dependan de la referencia serán borrados cuando el padre sea borrado.

3. *Nulo en cascada*: El valor de la columna Foreign Key tendrá el valor Nulo cuando el padre sea borrado.
4. *Omisión en cascada*: El valor de la columna Foreign Key tendrá el valor predeterminado cuando el padre sea borrado.
5. *Modificación en cascada*: Cuando el valor del padre sea cambiado, los correspondientes hijos automáticamente serán modificados.

La primera y segunda opción son declarativas, mientras que las tres últimas requieren de procedimientos adicionales y triggers (disparadores).

VISTAS

Durante el proceso de diseño físico, se pueden definir vistas. Una vista (VIEW) tiene el mismo aspecto que una tabla con columnas y se consulta de la misma forma que ésta. Conceptualmente, puede pensarse en una vista como una máscara que cubre una o más de las tablas subyacentes, así las vistas no almacenan físicamente los datos. Las vistas no pueden indexarse ya que no hay datos físicos directamente asociados a ellas. La definición de una vista se almacena en el diccionario de datos.

PROCEDIMIENTOS

Un procedimiento (PROCEDURE) es un bloque de sentencias PL/SQL (Procedure Language/SQL) al que pueden llamar las aplicaciones. Los procedimientos permiten almacenar dentro de la base de datos aplicaciones que se utilicen con frecuencia. Los procedimientos no devuelven ningún valor al programa que los llama. Los procedimientos almacenados pueden ayudar a reforzar la seguridad de los datos, lo que se consigue no concediendo a los usuarios el acceso directo a las tablas de una aplicación.

FUNCIONES

Las funciones (FUNCTIONS), lo mismo que los procedimientos, son bloques de código que se almacenan en la base de datos. A diferencia de éstos, las funciones pueden devolver valores al programa que las llama.

PAQUETES

Los paquetes (PACKAGES) sirven para organizar los procedimientos y las funciones en agrupamientos lógicos; sus definiciones se almacenan en el diccionario de datos. Los paquetes son muy útiles en las tareas administrativas necesarias para gestionar los procedimientos y las funciones.

DISPARADORES

Los disparadores (TRIGGERS) son procedimientos que se ejecutan cuando se produce un evento de base de datos específico. Puede utilizarse para aumentar la integridad referencial, conseguir una seguridad adicional o mejorar las opciones de auditoría disponibles.

ESPACIOS DE TABLAS

Un espacio de tablas (TABLESPACES) es una división lógica de una base de datos. Cada base de datos tiene al menos un espacio de tablas llamado *espacio de tablas SYSTEM*. Cada espacio de tablas consta de uno o más archivos, llamados *archivos de datos*, en un disco. Los archivos de datos tienen un tamaño fijo en el momento de su creación; cuando se necesita más espacio hay que añadir archivos nuevos. Para crear nuevos espacios de tablas hace falta crear nuevos archivos de datos.

Así, la división de los objetos de base de datos entre varios espacios de tablas permite que dichos objetos se almacenen de forma física en archivos de datos separados, que pueden situarse en discos distintos. Esta es una herramienta importante de planificación.

SEGMENTOS, EXTENSIONES Y BLOQUES

Los *segmentos (segments)* son la contrapartida física a los objetos lógicos de base de datos que almacenan los datos. Como un segmento es una entidad física, debe estar asignado a un espacio de tablas de la base de datos y por lo tanto encontrarse en uno de los archivos de datos de dicho espacio de tablas. Un segmento consta de secciones llamadas *extensiones (extens)* (conjunto contiguos de bloques de Oracle). Cuando las extensiones existentes en un segmento ya no pueden contener datos nuevos, el segmento consigue otra extensión. Este proceso de extensión continua hasta agotar el espacio libre disponible en los archivos de datos del segmento.

SEGMENTOS DE ROLLBACK

Para mantener la consistencia de lectura entre varios usuarios de la base de datos y poder anular las transacciones. Oracle dispone de un mecanismo de reconstrucción de una *imagen anterior* de los datos para transacciones no confirmadas. Oracle utiliza *segmentos de rollback* (anulación) dentro de la base de datos para llevar a cabo esta tarea. Los segmentos de rollback crecen hasta hacerse tan grandes como las transacciones que admiten.

INDICES

En una base de datos relacional, la situación física de una fila es irrelevante (a menos, por supuesto, que la base de datos tenga que encontrarla). Para que sea posible encontrar los datos, cada fila de cada tabla se etiqueta con un ROWID (Identificador de Fila). Este ROWID le indica a la base de datos el lugar exacto en que se encuentra la fila.

Un índice es una estructura de la base de datos que le permite al servidor localizar rápidamente una fila de una tabla. Los índices se utilizan para mejorar el rendimiento y asegurar la exclusividad de una columna.

Los índices se crean automáticamente siempre que se especifique una cláusula de restricción UNIQUE y PRIMARY KEY.

SINONIMOS

Para realizar una identificación completa de un objeto de base de datos (como una tabla o una vista) en una base de datos distribuida Oracle, es necesario especificar el nombre de la máquina anfitriona, el nombre del servidor, el propietario del objeto y el nombre del objeto. Los desarrolladores pueden crear sinónimos que apunten al objeto adecuado para ocultar este proceso a los usuarios, que sólo tienen que conocer el nombre del sinónimo.

La definición de sinónimos hace referencia a tablas de otro usuario sin crear su estructura. Los sinónimos públicos los comparten todos los usuarios de una base de datos concreta, mientras que los sinónimos privados pertenecen a propietarios de cuentas de bases de datos individuales. Siempre que se haga referencia a este sinónimo apuntará a la tabla adecuada. Los sinónimos permiten proporcionar punteros a tablas, vistas, procedimientos, funciones, paquetes y secuencias. Puede apuntar a objetos de la base de datos local o de bases de datos remotas.

USUARIOS

Una cuenta de usuario no es una estructura física de la base de datos, pero sí tiene importantes relaciones con los objetos de la base de datos: los usuarios son propietarios de los objetos de la base de datos. El usuario SYS es propietario de las *tablas del diccionario de datos* (*data dictionary tables*); en ellas se almacena información sobre el resto de las estructuras de la base de datos. El usuario SYSTEM es propietario de las vistas que permiten acceder a estas tablas del diccionario de datos, para que las utilicen el resto de los usuarios de la base de datos.

Los objetos que se crean en la base de datos como apoyo a las aplicaciones, se crean bajo cuentas de usuario. Las cuentas de la base de datos pueden conectarse a una cuenta del sistema operativo, lo que permite al usuario acceder a la base de datos desde el sistema operativo, sin tener que introducir contraseñas tanto para el sistema operativo como para la

base de datos. Pueden entonces acceder a los objetos de su propiedad o a aquéllos a los que se les ha concedido acceso.

ESQUEMAS

El conjunto de objetos que posee una cuenta de usuario se denomina esquema (schema) del usuario. En Oracle, es posible crear usuarios que no tengan la capacidad de acceder a la base de datos. Dichas cuentas de usuario proporcionan un esquema que puede utilizarse para mantener un conjunto de objetos de base de datos separados de otros esquemas de los usuarios.

CAPACIDADES DE SEGURIDAD

Las cuentas de la base de datos pueden protegerse mediante una contraseña. Esta protección es independiente de la protección mediante contraseña del sistema operativo. También se pueden crear cuentas con capacidad de acceso automático, lo que permite a los usuarios que han accedido a una cuenta del anfitrión acceder a una cuenta de base de datos. Disponer de una cuenta o de privilegios en una base de datos no le concede al usuario una cuenta o privilegios en otras base de datos.

PRIVILEGIOS Y FUNCIONES

Para que una cuenta pueda acceder a un objeto propiedad de otra cuenta tiene que habersele concedido primero el *privilegio* (PRIVILEGE) de acceso. Los privilegios son los tipos de acceso que cada usuario requiere según su status y responsabilidades.

Se pueden conceder o asignar privilegios específicos y formar roles para proporcionarlos a usuarios adecuados. De esta manera, el hecho de añadir nuevos usuarios a las aplicaciones se simplifica en gran medida, ya que basta con conceder o retirar roles al usuario.

Los *privilegios de sistema de la base de datos* permiten ejecutar un conjunto concreto de ordenes. Oracle, ofrece tres roles CONNECT (Conexión), RESOURCE (Recursos) y DBA (Administrador de base de datos). También pueden crearse roles a nivel sistema a partir de un conjunto de privilegios como CREATE TABLE (Crear Tabla), CREATE INDEX (Crear índice), SELECT ANY TABLE (Seleccionar cualquier tabla), GRANT ANY PRIVILEGE (Conceder cualquier privilegio), etc.

Los *privilegios de objetos de la base de datos* permiten realizar cierta operación sobre varios objetos, como son los privilegios INSERT (Insertar), SELECT (Seleccionar), UPDATE (Actualizar), DELETE (Borrar), EXECUTE (Ejecutar), ALTER (Alterar), REFERENCES (Referencias), INDEX (Índice), etc.

Los privilegios pueden concederse a usuarios individuales o a públicos (PUBLIC), que concede los privilegios a todos los usuarios de la base de datos. También los usuarios pueden conceder a otros usuarios la posibilidad de realizar concesiones sobre sus objetos.

4.8. PROTOTIPOS

4.8.1. DEFINICIÓN

Una manera efectiva para asegurar que las necesidades de los usuarios serán satisfechas es recalcar la identificación de requerimientos del sistema, más que las actividades de diseño.

El desarrollo de prototipos de aplicación proporciona un camino para adquirir información que describa los requerimientos de la aplicación y su evaluación con base en el empleo de un sistema que trabaja. El prototipo puede ser el vehículo de análisis o diseño.

El término *prototipo* se refiere a un modelo que funciona para una aplicación de sistemas de información. El prototipo no contiene todas las características o lleva a cabo la totalidad de las funciones necesarias del sistema final. Más bien incluye elementos suficientes para permitir a las personas utilizar el sistema propuesto para determinar qué les gusta, qué no les gusta e identificar aquellas características que deben cambiarse o añadirse.^[35]

El proceso de desarrollo y empleo de un prototipo tiene cinco características:

- El prototipo es una aplicación que funciona.
- La finalidad del prototipo es probar varias suposiciones formuladas por analistas y usuarios con respecto a las características requeridas del sistema.
- Los prototipos se crean con rapidez.
- Los prototipos evolucionan a través de un proceso iterativo.
- Los prototipos tienen un costo bajo de desarrollo.

La finalidad del desarrollo de los prototipos es incrementar el nivel de productividad en el desarrollo de sistemas.

4.8.2. USOS

El desarrollo de prototipos de aplicación tiene dos usos principales.

1. Es un medio eficaz para aclarar los requerimientos de los usuarios.

[35] Senn, James.- *Análisis y Diseño de Sistemas de Información*.- Traduc. Urbina Medal Edmundo Gerardo y otros.- 2ª Edición.- Editorial McGRAW-HILL.- México.- 1994 - pp. 243 - 246.

2. Sirve para verificar la factibilidad del diseño de un sistema. Crear un prototipo y evaluar el diseño por medio de su uso, mostrará la factibilidad del diseño o sugerirá la necesidad de encontrar otras opciones.

4.8.3. RAZONES

Las razones para el uso de prototipos son resultado directo de la necesidad de diseñar y desarrollar sistemas de información con rapidez, eficiencia y eficacia.

- ✓ ***Aumento en la productividad:*** La productividad, cuando se aplica al desarrollo de sistemas, significa llevar a cabo las actividades en la forma más eficiente, obteniendo el mayor impacto con la mejor utilización de los recursos. Para ello, la elaboración de prototipos minimiza errores de diseño, minimiza el tiempo que se pierde debido al desarrollo incorrecto, garantiza que los esfuerzos realizados por el equipo de trabajo sean fructíferos, garantiza que los usuarios reciban la aplicación que necesitan y garantiza que no tendrá que volverse a hacer el trabajo de desarrollo.
- ✓ ***Redesarrollo planificado:*** El prototipo de una aplicación está diseñado para ser modificado y esto trae beneficios tanto a los usuarios como a la organización. Los usuarios pueden cambiar de opinión con respecto a los requerimientos e incluso se les invita a que lo hagan cuando evalúan el prototipo, entonces no existirán sorpresas cuando el sistema sea implantado. Los usuarios pueden señalar con mayor facilidad las características que les agradan o desagradan e indicar caminos más cortos en un sistema existente y que funciona, que identificarlos en una descripción gráfica o por escrito del sistema propuesto.
- ✓ ***Entusiasmo de los usuarios con respecto a los prototipos:*** La disponibilidad de un sistema real (no de una especificación abstracta como documentos o ejemplos) fomenta la participación entusiasta de los usuarios para revisar y evaluar las características propuestas del sistema.

4.8.4. CARACTERÍSTICAS

- Los datos proporcionados por el usuario durante el desarrollo dan información valiosa sobre el diseño.
- La experiencia ganada a través del empleo del prototipo, evita sorpresas desagradables en la fase de implantación.
- Se esperan sugerencias, que serán bien recibidas, para efectuar cambios en las especificaciones y modificaciones en las características del sistema.

- Se concede mayor importancia a la velocidad de desarrollo y no a la eficiencia en el funcionamiento del prototipo, se ahorra tiempo y se disminuye el retraso en el desarrollo de sistemas de información.
- Se recalca la determinación de los requerimientos correctos.
- Los costos de desarrollo son bajos, siempre y cuando se utilicen las Herramientas apropiadas.
- La eficiencia en el desarrollo para completar la aplicación, mejora como resultado de la recepción inmediata de las reacciones de los usuarios.
- La participación de personal clave fomenta la comunicación durante el desarrollo.
- Los usuarios ganan experiencia durante el desarrollo, lo que facilita el entrenamiento previo en el uso del sistema.

4.8.5. ETAPAS DEL MÉTODO DE PROTOTIPOS

El desarrollo de un prototipo para una aplicación se lleva a cabo en una forma ordenada, sin importar las herramientas utilizadas.

IDENTIFICACIÓN DE REQUERIMIENTOS CONOCIDOS

Antes de crear el prototipo, los analistas y usuarios deben trabajar juntos para identificar los requerimientos conocidos que tienen que satisfacerse. Para hacerlo determinan los fines para los que servirá el sistema y el alcance de sus capacidades establecidos en el análisis de sistema C/S.

DESARROLLO DE UN MODELO DE TRABAJO

La construcción de un prototipo es un proceso iterativo de desarrollo. Para comenzar la primera iteración, usuarios y analistas identifican de manera conjunta los datos que son necesarios para el sistema y especifican la salida que debe producir la aplicación, basándose en el diseño preliminar del sistema C/S.

La construcción del prototipo inicial está a cargo del analista y diseñador de sistemas que para este fin emplean cualquiera de las herramientas. La rapidez con la que se genera el sistema es esencial para que no se pierda el estado de ánimo sobre el proyecto y los usuarios puedan comenzar a evaluar la aplicación a la mayor brevedad posible.

En el desarrollo de un prototipo se preparan los siguientes componentes:

- El Front End para la conversación entre el usuario y el sistema.
- Pantallas y formatos para la entrada de datos.

- Módulos esenciales.
- Salida del sistema.

REVISION DEL PROTOTIPO

Es responsabilidad del usuario trabajar con el prototipo y evaluar sus características para determinar los cambios o mejoras que sean necesarios así como la eliminación de características inadecuadas o innecesarias. Y el equipo de trabajo del proyecto será responsable de realizar las modificaciones.

REPETICION DEL PROCESO LAS VECES QUE SEA NECESARIO

El proceso antes descrito se repite varias veces hasta que usuarios y analistas están de acuerdo en que el sistema ha evolucionado lo suficiente como para incluir todas las características necesarias o cuando ya es evidente que no se obtendrá mayor beneficio con una iteración adicional.

USO DEL PROTOTIPO

Cuando el prototipo está terminado, el siguiente paso es tomar la decisión sobre cómo proceder. Existen caminos a seguir después de evaluar la información obtenida con el desarrollo y uso del prototipo: descartar el prototipo y abandonar el proyecto de aplicación, implantar el prototipo, continuar con el desarrollo de la aplicación o comenzar con otro prototipo

4.8.6. PROPÓSITO

El desarrollo de prototipos para aplicaciones no debe interpretarse como la estrategia a seleccionar cuando el analista no sabe por dónde comenzar un proyecto. Si el analista no comprende los requerimientos generales de los usuarios o no puede articular detalles esenciales de diseño, el prototipo no será de gran ayuda.

El analista primero debe identificar los requerimientos generales o las características esenciales del diseño. Un prototipo de aplicación permite a usuarios y personal encargado de su desarrollo, ganar información adicional con respecto a la forma en que será utilizado el sistema, las características que serán necesarias y otras consideraciones. La falta de conocimiento no sugiere la necesidad de construir un prototipo, más bien es un indicador de la necesidad que se tiene de investigar para poder conocer los requerimientos del usuario.

Los prototipos de pantallas también proporcionan una manera para obtener las reacciones de los usuarios hacia la cantidad de información presentada sobre la pantalla de

visualización. Tal vez el usuario decida que un diseño en particular es muy denso ya que existen demasiados detalles sobre la pantalla.

En el sistema SISEFIN el prototipo fue presentado a los usuarios de PEMEX.

4.9. DISEÑO FINAL

4.9.1. DEFINICIÓN

Conforme a las observaciones que se generaron en la presentación del prototipo se depura el diseño preliminar creándose el diseño definitivo. Las actividades del diseño final son principalmente revisar y actualizar el diseño preliminar y preparar la documentación del diseño para iniciar la construcción, unificando esfuerzos y se avance en forma apropiada en base a lo especificado.

4.9.2. CARPETA DE DESCRIPCIÓN DEL DISEÑO DEL SISTEMA

Cuando el diseño de un sistema de información está completo, las especificaciones son documentadas en una forma que esboza las características de la aplicación. Los líderes de proyecto de sistemas denominan a estas especificaciones, información liberada o carpeta de diseño. Contiene todos los detalles que deben incluirse en el software de computadora, conjuntos de datos y procedimientos que comprenden el sistema de información. Además de un plan de desarrollo que indican los tiempos necesarios para el desarrollo de las actividades; analistas de sistemas, programadores y personal necesario para el desarrollo.

La carpeta de diseño es recomendable para asegurar que las especificaciones son completas y que el personal de sistemas de información estará familiarizado con la naturaleza y formato de la información.

4.10. EJEMPLO: DISEÑO DEL SISTEMA SISEFIN

DISEÑO DE LA ARQUITECTURA CLIENTE/SERVIDOR

INSTALACIONES FÍSICAS:

De acuerdo a la infraestructura de la GIM el diseño de las instalaciones físicas es la siguiente:

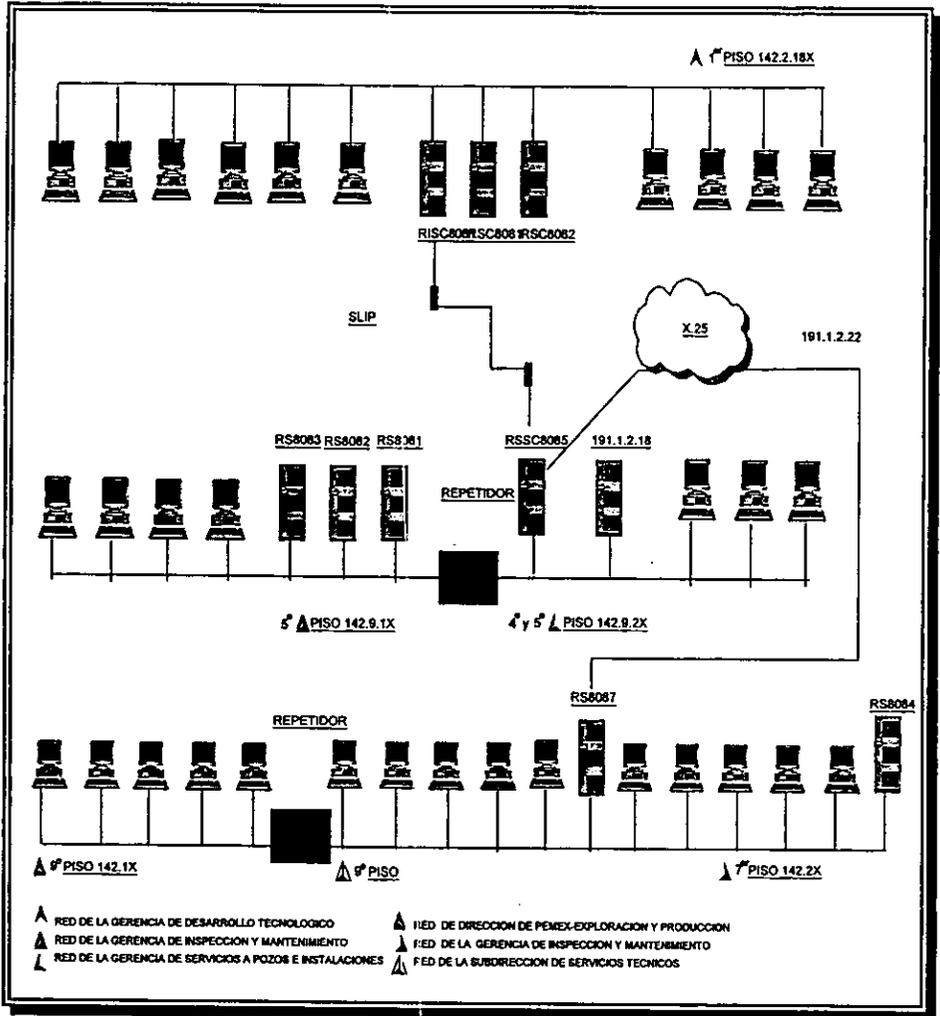


Figura (4.6) Diseño Físico de las Instalaciones de la GIM

INFRAESTRUCTURA REQUERIDA PARA LA ARQUITECTURA CLIENTE/SERVIDOR:

La infraestructura requerida para la operación del sistema SISEFIN en ambiente Cliente/Servidor es:

SERVIDOR

Hardware	Software
32 Mb de RAM	Oracle v7.1x (Back End)
File System 10 MB (ambiente lógico y físico)	SQL *PLUS
Unidad de cinta de 1/4 de pulgada ó unidad de cinta de 8 mm para respaldo.	SQLDBA
Tarjeta Ethernet	SQL*NET TCP/IP
Canales de comunicación remota (PEMEX-PAQ)	Sistema operativo UNIX
	Software de Red (LAN WORK PLACE o SYNTAX)

PC Cliente

Hardware	Software
Procesador 80386DX/33 o superior	Sistema Operativo de Red (LAN WORK PLACE o SYNTAX)
8 MB de RAM o superior	SQL*NET TCP/IP
Espacio de 10 MB en disco duro para el sistema	MicroSoft Windows v3.x o 95
Tarjeta de red Ethernet	Runtime SUPERBASE v. 2.01 (Front End)
Monitor Super VGA color.	EXCEL v 4
Impresora esclava o en red	Sistema Operativo MSDOS v 6.x O Windows 95

Y a partir de estos requerimientos técnicos el esquema de la plataforma computacional de la GIM se transformará de la siguiente manera:

INFRAESTRUCTURA DE LA ARQUITECTURA CLIENTE/SERVIDOR DE LA GERENCIA DE INSPECCION Y MANTENIMIENTO

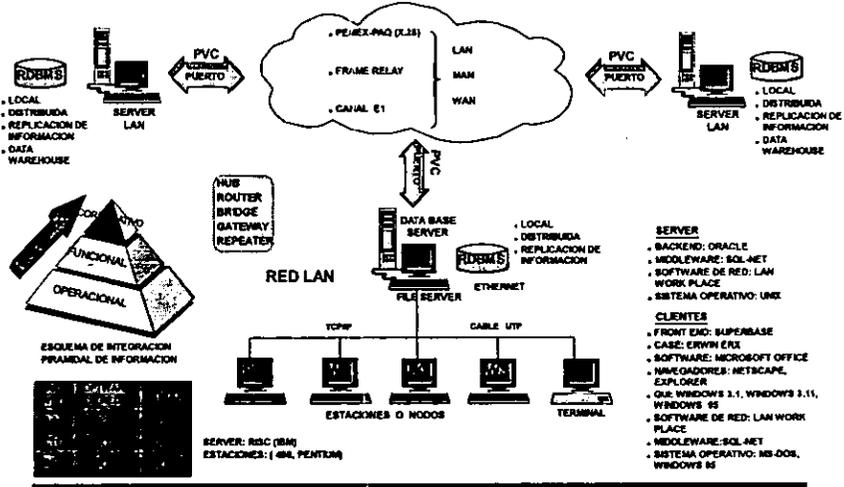


Figura (4.7) Infraestructura de la Arquitectura Cliente/Servidor de la GIM

CAPACITACION DE LOS RECURSOS HUMANOS

Para la capacitación de las personas encargadas de la administración de la arquitectura Cliente /Servidor se recomiendan los siguientes cursos:

- ↳ La Tecnología Cliente/Servidor
- ↳ Administración de bases de datos relacionales.
- ↳ Replicación entre bases de datos.
- ↳ Administración de Redes con bases de datos.
- ↳ Base de Datos ORACLE.
- ↳ Sistema Operativo UNIX nivel avanzado.
- ↳ Redes Ethernet y LAN

PROTOTIPO

El prototipo mostrado cubrió las expectativas del usuario en cuanto la información que almacena y se consulta. Las pantallas impresas también fueron vistas y aprobadas.

DISEÑO DEL SISTEMA SISEFIN

ESTANDARES

En la documentación del diseño del sistema SISEFIN se establecieron estándares como son:

REPORTE DE ESTANDARES EXISTENTES APLICABLES AL PROYECTO.

A. Sistemas Operativos

Se proponen los siguientes Sistemas Operativos:

1. Cliente: MS-DOS Versión 6.2
Windows 3.1
Windows 3.11
Windows 95

2. Servidor: Unix

B. Sistemas Operativos

Se deberán desarrollar con la herramienta SUPERBASE y como manejador de base de datos ORACLE.

REPORTE DE NUEVOS ESTANDARES PARA EL PROYECTO.

Estándares propuestos para el ambiente de desarrollo.

(1) Entidad-Relación

Nomenclaturas, etc.

(2) Nombres de tablas y campos

Todos los nombres de tablas y campos se escriben en singular. Etc.

(3) Atributos y entidades

Si el nombre que se adecue consta de más de una palabra, se podrán abreviar y deberán separarse con un guión bajo “_”.Etc.

(4) Campos y Tablas

Se recomienda que el nombre asignado a los atributos y las entidades, sea el mismo que se asigne a los campos y tablas de la base de datos. Etc.

(5) Palabras y abreviaturas

Si se utiliza una palabra o abreviación en tablas o variables siempre debe escribirse igual para la declaración de otras. Etc.

(6) Catálogos

Los nombres de los campos que denoten claves deberán tener el prefijo "Cve". Etc.

(7) Programación

Estándares para: Tipos de datos, Constantes, Variables y Parámetros, Funciones, Procedimientos, Módulos, Comandos SQL, etc.

Estándares propuestos para el ambiente diseño de pantallas.

A. Generales

- (1) Tipo, tamaño y color de letra.
- (2) Colores de objetos.
- (3) Reloj de Arena.
- (4) Etc.

B. Objetos

- (1) Formas y Diálogos.
- (2) Pushbutton, Radio Button, Check Box, Background Text, Data Field, Combo Box, List Box.
- (3) Etc.

C. Tipos de Mensajes

- (1) Mensajes de Información.
- (2) Mensajes de Proceso.
- (3) Mensajes de Confirmación.
- (4) Mensajes de Error.
- (5) Etc.

D. Seguridad y Ayuda

(1) Seguridad.

(2) Ayuda.

PROCEDIMIENTOS

Procedimientos para entrada de datos.

Para el cálculo del número de CIP se toman las siguientes consideraciones:

1. Tener número de la clave del centro de trabajo.
2. De la fecha de captura tomar el mes y asignarle una letra.

A- Ene	E- May	I- Sep
B- Feb	F- Jun	J- Oct
C- Mar	G- Jul	K- Nov
D- Abr	H- Ago	L- Dic

3. Tomar los 2 últimos dígitos del año.
4. Tomar 3 números consecutivos

Ejemplo: $\frac{\cdot\#}{1} \frac{M}{2} \frac{\#\#}{3} \frac{\###}{4}$

Procedimientos durante la ejecución.

Modulo de Subproyecto

- El Programa, Subprograma, Proyecto, Proy. Obra, Subproyecto, Guión, Renglón y Concepto son botones que al oprimirlos despliegan información, la cual esta condicionada.
- Si se marca con una cruz la opción de DOLAR, se hace la conversión a moneda nacional, abriendo el archivo de Configuración de varios (CONF_VAR) y se toma el tipo de cambio que tiene dólar para multiplicarlo con cada uno de los montos mensuales y el monto autorizado.
- El campo MONTO es la suma de todos los montos mensuales y la variable DIFERENCIA visualiza la resta del monto autorizado y el monto total.

Procedimientos de seguridad y respaldo.

Para poder usar el sistema es necesario proporcionar una cuenta de usuario y una contraseña. El sistema a desarrollar deberá verificar los privilegios de los usuarios en cuanto a los accesos que tengan a los objetos de cada aplicación.

El administrador de la base de datos será el responsable directo de la actividad de respaldo. Los respaldos pueden ser a disco o cinta. Se recomiendan respaldos diarios de la base de datos y tres respaldos al día del registro de transacciones.

Procedimientos de validación.

- Avisar si existe registro duplicado y que NO permita salvar el registro.
- Avisar cuando existan campos que no deben estar vacíos y que NO permita salvar el registro.
- Avisar cuando rompa con la integridad referencial y que NO permita borrar o modificar el registro.

PSEUDO CODIGO**PROCESO DE CAPTURA**

Para dar de alta un registro se tiene que seleccionar el icono correspondiente o en la barra de menú seleccionar Insertar, automáticamente el sistema desplegará la pantalla para iniciar la captura. El usuario en algunas pantallas tendrá la opción de teclear la información que se pide o simplemente oprimirá un botón para seleccionar datos.

Al oprimir el **ICONO DE SALVAR** o dar el último **ENTER** en la captura, (♣1) el sistema preguntará si la información esta correcta

SI: Salva el registro (♣2) y pregunta si desea capturar otro registro.

SI: Continúa con el proceso de captura.

NO: Sale del proceso de captura

NO: Continúa con el mismo registro de captura.

Al oprimir el **ICONO DE CANCELAR** o dar un **ESC** no salva el registro y sale del proceso de captura (♣3).

- (♣1) Salva el registro en SuperBase dando un STORE FORM.
- (♣2) Salva el registro en Oracle.
- (♣3) No salva el registro en Superbase, ni en Oracle.

Consideraciones:

Al capturar, que presente los mensajes de ayuda en la barra de STATUS para cada uno de los campos.

Todos los campos que sean tipo CHAR que se salven y aparezcan en mayúsculas

MODULO DE CAP

1. La FECHA DE CAPTURA debe traer por omisión la fecha de hoy, pero debe permitir modificarla.
2. El AUTORIZADO ANUAL lo trae de la tabla SUBPROY, que de acuerdo al subproyecto que eligió se toma MON_AUT.
3. AUTORIZADO AL MES:
 - Tomar el mes de la fecha de captura y asignarlo a MES_CAP.
 - Abrir la tabla de SUBPROY y posicionarse en el subproyecto que eligió en el CAP.
 - Sumar los montos de Enero hasta MES_CAP.
 - El resultado es el Autorizado del Mes.
4. ACUMULADO: (Es la suma de todos los montos del CAP que pertenecen a un mismo suproyecto)
 - Pasar a la tabla de SUBPROY y posicionarse en el subproyecto que eligió en el CAP.
 - Tomar el campo ACUMES_CAP y sumar el MONTO_CAP y el resultado asignarlo en ACUMES_CAP.

DICCIONARIO DE TABLAS

Catálogos	Descripción
C_REGION	Catálogo de regiones.
C_DTTO	Catálogo de distritos.
C_CENTRO	Catálogo de centros de trabajo.
C_DEPTO	Catálogo de los departamentos.
C_TPROG	Catálogo de tipos de programa.
C_PROG	Catálogo de programas.
C_SUBPRO G	Catálogo de subprogramas.
C_PROY	Catálogo de proyectos de inversión.
C_PROYOC	Catálogo de proyectos de obra capitalizable.
C_GUION	Catálogo de guiones del gasto.
C_RENGLO	Catálogo de renglón del gasto.
C_ORIGEN	Catálogo conceptos de origen.
C_RENCO	Contiene el catálogo de la relación de renglón de gasto / concepto de origen.
C_MOMEN T	Catálogo de momentos contables.
C_NIVEL	Catálogo de niveles.
C_JORNA	Catálogo de jornada.

Archivos	Descripción
C_TABSUE	Catálogo de tabulador de sueldos.
C_RESPON	Catálogo de responsables.
SUBPROY	Archivo del subproyecto.
CAP	Archivo de la Clave de Autorización Presupuestal.
COMPROME	Archivo del comprometido.
DOCUMENT	Archivo de documentos.
DOC_DET	Archivo de detalle de documentos.
CONF_VAR	Archivo de configuración de varios.
CONEXION	Archivo de conexión.
INF_REPO	Archivo de información para reportes.
PERMISO	Archivo de información del password.

ESTRUCTURA DE DATOS

CATALOGOS

CATALOGO DE RENGLON DEL GASTO

C_RENGLO	TIP Y LONG	CARACTERISTICAS	VALIDACION		
CVE_RENGLON	VARCHAR2(3)	Llave Primaria	Unica	NOT NULL	
DESC_RENGLON	VARCHAR2(75)				
CVE_GUION	VARCHAR2(2)	Llave Foránea (C_GUION)	Unica		

CATALOGO DE CONCEPTOS DE ORIGEN

C_ORIGEN	TIP Y LONG	CARACTERISTICAS	VALIDACION		
CVE_CONCEP	VARCHAR2(8)	Llave Primaria	Unica	NOT NULL	
DESC_CONCEP	VARCHAR2(100)				

CATALOGO DE LA RELACION DE RENGLON DEL GASTO/CONCEPTO DE ORIGEN

C_RENCON	TIP Y LONG	CARACTERISTICAS	VALIDACION		
CVE_RENGLON	VARCHAR2(3)	Llave Foránea (C_RENGLO)	Unica	NOT NULL	
CVE_CONCEP	VARCHAR2(8)	Llave Foránea (C_ORIGEN)	Unica		
REN_CON	VARCHAR2(11)	Llave Primaria	Unica		

TABLAS GENERALES

ARCHIVO DEL CAP

CAP	TIP Y LONG	CARACTERISTICAS	VALIDACION		
CAP	VARCHAR2(11)	Llave Primaria	Unica	NOT NULL	
CVE_SPROY	VARCHAR2(8)	Llave Foránea (C_SUBPROY)		NOT NULL	
CVE_CTRDOR	VARCHAR2(3)	Llave Foránea (C_CENTRO)		NOT NULL	
CVE_DPTDOR	VARCHAR2(5)	Llave Foránea (C_DEPTO)		NOT NULL	
CVE_CTRADO	VARCHAR2(3)	Llave Foránea (C_CENTRO)		NOT NULL	
CVE_DTADO	VARCHAR2(5)	Llave Foránea (C_DEPTO)		NOT NULL	
UNID_CONT	VARCHAR2(3)			NOT NULL	
FECHA_CAP	DATE			NOT NULL	
MOMENTO	VARCHAR2(4)			NOT NULL	
MONEDA	VARCHAR2(1)			NOT NULL	Check (MONE DA in (N',D'))
MONTO	NUMBER(11,2)			NOT NULL	
JUSTIFICA	VARCHAR2(100)			NULL	
TIP_CAP	VARCHAR2(1)			NOT NULL	Check (TIP_CA P in (I',E))

DIAGRAMA ENTIDAD - RELACION

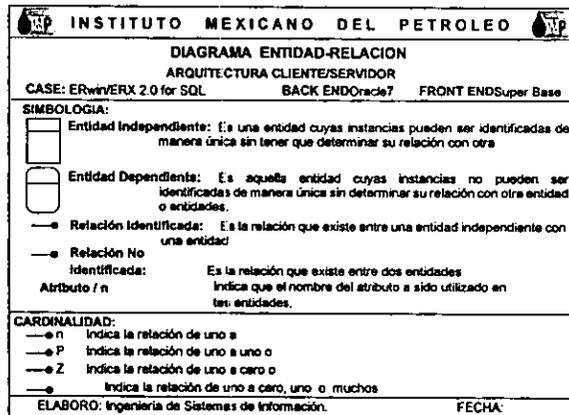
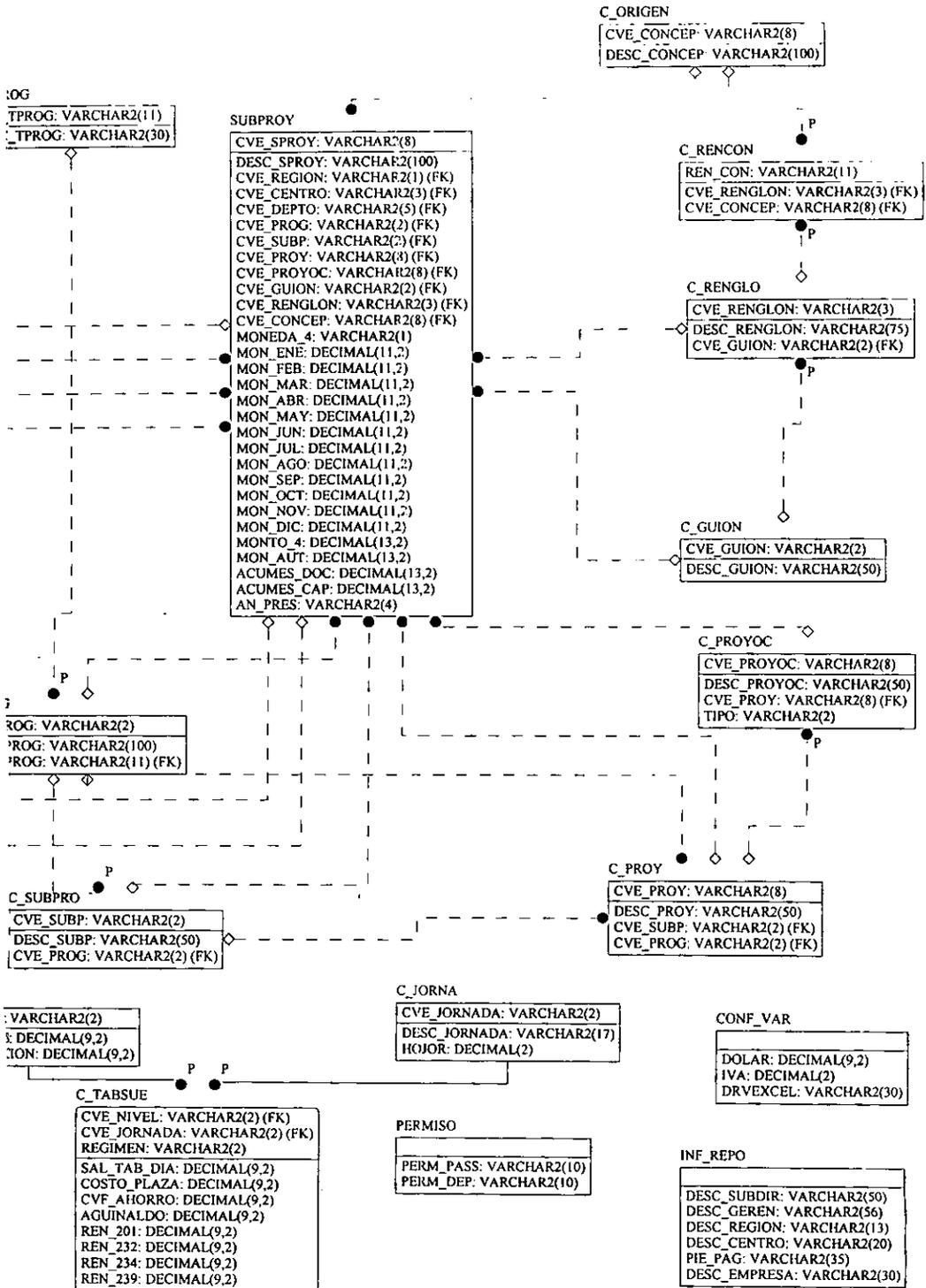


Figura (4.8) Simbología del Diagrama Entidad - Relación



SCRIPTS

```

/* ***** */
/*  CREACION DE LOS USUARIOS DE SISEFIN  */
/* ***** */

/* Dentro del esquema de seguridad se toman en cuenta dos tipos de usuarios para */
/* proteger la información del sistema SISEFIN. De acuerdo a cada usuario se */
/* asignan privilegios de insertar, borrar, modificar y consultar la información */
/* de la Base de Datos. */

/* Los siguientes nombres de usuarios y su definición de passwords son una pro- */
/* puesta pero pueden ser cambiados a consideración del administrador de la base */
/* de datos donde van a ser creados. */

/* El primer usuario a crear es SISEFIN y tiene una clave de acceso con el mismo */
/* nombre. */

create user SISEFIN identified by SISEFIN;

/* Se asignan privilegios de conexión y recursos, esto significa que puede */
/* insertar, borrar, modificar y consultar la información de la Base de Datos del */
/* SISEFIN, a excepción de los catálogos institucionales de PEMEX donde su acceso */
/* será en modo de consulta. */

grant connect,resource to SISEFIN;

/* Se direcciona al espacio de trabajo (tablespace) de SIGIM y al espacio tempo-*/
/* ral (temporary tablespace) de TEMP asumiendo que SIGIM y TEMP ya existen, por */
/* lo cual el usuario SISEFIN toma por omisión los parámetros de almacenamiento */
/* que están definidos en SIGIM y TEMP. */

alter user SISEFIN identified by SISEFIN
default tablespace SIGIM temporary tablespace TEMP;

/* En caso de que se desee borrar el usuario SISEFIN se ejecuta la siguiente */
/* instrucción. */

/* drop user SISEFIN cascade; */

/* ***** */

/* Conexión a el usuario SISEFIN y password SISEFIN. */

connect SISEFIN/SISEFIN;

etc.

/* ***** */
/*  CONCESION DE PRIVILEGIOS DEL USUARIO CATALOGO AL USUARIO SISEFIN  */
/*      SOBRE LOS CATALOGOS INSTITUCIONALES DE PEMEX. */
/* ***** */

/* Conexión con el usuario CATALOGO y password CATALOGO. */

connect CATALOGO/CATALOGO;

/* El privilegio de insertar, borrar, modificar, consultar y0 referencia son */

```

```

/* otorgados a el usuario SISEFIN sobre los catálogos institucionales de PEMEX. */
/* Con la opción de que el receptor de los privilegios se los asigne al usuario */
/* SISEFINI. */

```

```

grant INSERT,DELETE,UPDATE,SELECT,REFERENCES on C_REGION to SISEFIN with grant option;
grant INSERT,DELETE,UPDATE,SELECT,REFERENCES on C_DTTO to SISEFIN with grant option;
etc.

```

```

/* ..... */
/*          CREACION DE SINONIMOS Y TABLAS DEL USUARIO SISEFIN          */
/* ..... */

```

```

/* ..... */
/*          SINONIMOS          */
/* ..... */

```

```

/* Los catálogos institucionales de PEMEX son creados como sinónimos, es decir */
/* cada sinónimo incluye la tabla entera de cada catálogo del usuario CATALOGO */

```

```

/* Antes de crear cada sinónimo, tiene también la opción de borrarlo siempre y */
/* cuando exista. */

```

```

connect SISEFIN/SISEFIN;

```

```

/* 1) CATALOGO DE REGION */;

```

```

drop synonym C_REGION;
create synonym C_REGION for CATALOGO.C_REGION;

```

```

/* 2) CATALOGO DE DISTRITOS */;

```

```

drop synonym C_DTTO;
create synonym C_DTTO for CATALOGO.C_DTTO;

```

```

etc.

```

```

/* ..... */
/*          TABLAS          */
/* ..... */

```

```

/* Las siguientes tablas tienen especificado: campo(s) que conforman su llave */
/* primaria, referencias con otras tablas, especificación de nulos y declara- */
/* ción de validaciones. Además de que por cada tabla de determinan sus .....*/
/* parámetros de almacenamiento, */

```

```

/* Estas tablas se dividen en catálogos y archivos pero se crean de igual manera.*/

```

```

/* Antes de crear cada tabla, tiene también la opción de borrarla siempre y */
/* cuando exista. */

```

```

/*          CATALOGOS PROPIOS DEL SISEFIN          */

```

```

/* 14) CATALOGO DE MOMENTOS CONTABLES */;

```

```

drop table C_MOMENT ;
create table C_MOMENT (
    CVE_MOMENT VARCHAR2(4) primary key NOT NULL,
    DESC_MOMENT VARCHAR2(20) NULL )

```

```

pctused      90
pctfree      5
storage (
  initial     1 K
  next        1 K
  pctincrease 0
  minextents  1
  maxextents  1 )
tablespace SIGIM ;

```

etc.

```
/*      TABLAS PROPIAS DEL SISEFIN      */
```

```
/* 2) ARCHIVO DE CAP */;
```

```

drop table CAP;
create table CAP (
  CAP          VARCHAR2(11) primary key NOT NULL,
  CVE_SPROY    VARCHAR2(8)  references subproy NOT NULL,
  CVE_DPTDOR   VARCHAR2(3)  references c_centro NOT NULL,
  CVE_DPTDOR   VARCHAR2(5)  references c_depto NOT NULL,
  CVE_CTRADO   VARCHAR2(3)  NOT NULL,
  CVE_DPTADO   VARCHAR2(5)  references c_depto NOT NULL,
  UNID_CONT    VARCHAR2(3)  NOT NULL,
  FECHA_CAP    DATE          NOT NULL,
  MOMENTO      VARCHAR2(4)  NOT NULL,
  MONEDA       VARCHAR2(1)  NOT NULL check(MONEDA in('N','D')),
  MONTO        NUMBER(11,2) NOT NULL,
  JUSTIFICA    VARCHAR2(100) NULL,
  TIP_CAP      VARCHAR2(1)  NOT NULL check(TIP_CAP in('I','E')) )
pctused      90
pctfree      5
storage (
  initial     156 K
  next        156 K
  pctincrease 0
  minextents  1
  maxextents  3 )
tablespace SIGIM ;

```

```

ALTER TABLE CAP
ADD ( FOREIGN KEY (CVE_CTRADO)
      REFERENCES C_CENTRO );

```

```

ALTER TABLE CAP
ADD ( FOREIGN KEY (UNID_CONT)
      REFERENCES C_CENTRO );

```

```

ALTER TABLE CAP
ADD ( FOREIGN KEY (MOMENTO)
      REFERENCES C_MOMENT );

```

```
/* 4) ARCHIVO DE DOCUMENTOS */;
```

```

drop table DOCUMENT ;
create table DOCUMENT (
  NO_CIP      VARCHAR2(7) primary key NOT NULL,
  NO_CIF      NUMBER(4)  NULL,

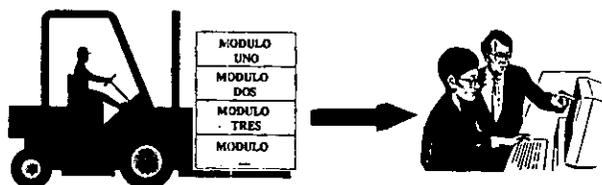
```

```
FECHA_CAP DATE NOT NULL,  
FECHA_DEV DATE NOT NULL,  
MOMENTO_AC VARCHAR2(4) references c_moment NOT NULL,  
CAP VARCHAR2(11) references cap NULL,  
MONEDA VARCHAR2(1) NOT NULL check(MONEDA in('N','D')),  
TOTAL NUMBER(11,2) NULL,  
TOT_IVA NUMBER(11,2) NULL,  
CVE_FICHA VARCHAR2(6) references c_respon NOT NULL))  
pctused 90  
pctfree 5  
storage (  
  initial 80 K  
  next 80 K  
  pctincrease 0  
  minextents 1  
  maxextents 3 )  
tablespace SIGIM ;
```

etc.

CAPITULO V

CONSTRUCCION E IMPLANTACION DE SISTEMAS CLIENTE / SERVIDOR



5.1. CONSTRUCCION DE LA INFRAESTRUCTURA

Para la construcción de la arquitectura Cliente/Servidor se deben considerar los siguientes puntos:

5.1.1. INSTALACIÓN DE LA RED

Primeramente es necesario realizar la distribución física de los equipos, con la finalidad de establecer las distancias entre ellos y así poder cuantificar las longitudes de cable requeridas. La instalación de la Red requiere de: Instalación del cableado e Instalación de la Tarjeta de Red.

Finalizado lo anterior se procederá a probar cada tarjeta previniendo detectar fallas de la tarjeta, así como su conexión con el cable de la red.

5.1.2. CONFIGURACIÓN DEL SERVIDOR

Por lo regular los servidores ya tienen cargado un sistema operativo, por lo que sólo es necesario realizar labor de dar de alta a los dispositivos, así como realizar pruebas de ambiente, como:

- Crear filesystems, usuarios, passwords
- Levantar la interfaz (configuración de la tarjeta red)
- Proporcionar las direcciones IP para los elementos de la red.
- Configurar los dispositivos periféricos del servidor.
- Dar de Alta a los dispositivos.

Al terminar lo anterior se verifica con alguna PC que ya tenga cargado el software de red los siguientes servicios, de la computadora al servidor.

- Ping
- TELNET
- FTP

5.1.3. INSTALACIÓN Y CONFIGURACIÓN DEL SISTEMA OPERATIVO DE RED

Tanto en el servidor como en el cliente se tiene que cargar el sistema operativo de red para establecer la comunicación entre los equipos. Cualquier software de red proporciona una instalación y configuración guiada en la aplicación como en su documentación. Ya levantados los servicios de la red en los equipos se realizan pruebas de accesos a los discos duros compartidos e impresoras instaladas. Una de las pruebas es utilizar el comando Ping, con el cual se detectará si existe conectividad entre los diferentes dispositivos.

5.1.4. INSTALACIÓN DEL SOFTWARE DE APLICACIÓN

En el servidor se instala el sistema manejador de base de datos actividad realizada por el proveedor del RDBMS o una persona con amplios conocimientos en administración de bases de datos, y sobre todo en la información o proyectos de sistemas que se van ha manejar dentro de la organización. Posteriormente del levantamiento de la base de datos se hace la instalación del middleware en el cliente para la interacción con la base de datos y la instalación del front end el cual debe tener las librerías necesarias para la conexión con el manejador de la base de datos

5.2. PANORAMA DE LA INGENIERIA DE SOFTWARE

La calidad de un sistema depende de su diseño, desarrollo, prueba e implantación. Una debilidad en cualquiera de éstas áreas pondría seriamente en peligro la calidad y, por lo tanto, el valor del sistema para los usuarios. Actualmente existe un área que se enfoca al estudio del aseguramiento de la calidad del software, la **Ingeniería de Software**.

5.2.1. HISTORIA

El patrón general de actuación industrial en materia de desarrollo de Software sigue el de los calendarios incumplidos, los sobregiros presupuestales y los problemas de calidad que han provocado pérdidas cuantiosas. Esta condición de la industria es lo que generalmente se conoce como la *crisis del software*.

El objetivo primordial de la ingeniería de software, fundada en el seno de una conferencia de la OTAN celebrada en Roma en 1968, es responder a estos desafíos de productividad y calidad aplicando disciplinas de ingeniería al desarrollo de software. La entonces naciente programación estructurada, obra del profesor Dijkstra, se constituía como condición de posibilidad de la nueva disciplina de ingeniería.

Durante la década de 1970 se realizaron importantes avances. Se refinaron los conceptos de la programación estructurada, se crearon diversas metodología estructuradas efectivas

para el análisis y diseño (entre las que alcanzaron gran popularidad las metodologías de Edward Yourdon y de Michel Jackson).

Durante la década de 1980, continuo el avance en métodos formales, se desarrollaron las métricas de software, las herramientas CASE y el empleo de prototipos y se introdujeron la reingeniería y un nuevo paradigma en la practica del desarrollo de software el *paradigma de los objetos*. La esencia de la *programación orientada a objetos* es el ocultamiento o encapsulamiento de estado interno de las entidades y la especificación de sus propiedades interactivas por medio de una interfaz de operaciones. Toda esta tecnología ha venido a incrementar la productividad y la calidad del software.

5.2.2. DEFINICIÓN

El *software* se define como programas de computadora, procedimientos, documentación y datos pertinentes a la operación de un sistema de computo; la *ingeniería* es la aplicación sistemática del conocimiento científico a la creación y construcción de soluciones costo-efectivas a problemas prácticos al servicio de la especie humana.

Por ende, se puede definir la *ingeniería de software* como la rama de la ingeniería que aplica los principios de la ciencia de la computación y las matemáticas para lograr soluciones costo-efectivas a los problemas del desarrollo de software.

La ingeniería de software es un conjunto coherente de actividades o pasos relacionados entre sí, con el objeto de crear y evolucionar productos software.

5.2.3. METAS

Las metas de la ingeniería de software son elaborar consistentemente productos correctos, utilizables y costo-efectivos. Los productos que cumplan con estas metas de calidad serán mas fáciles de desarrollar y de adaptarse a necesidades no anticipadas.

Corrección. Un producto de software es correcto en la medida que realiza las funciones para las que fue construido de modo apropiado, consiste y predecible. Mientras que la calidad de código y las pruebas son parte de esta meta, en un sentido más amplio la corrección incluye la sensibilidad y tolerancia a fallas. Estas cualidades pueden lograrse mas prácticamente si se tiene cuidado en el diseño y la construcción, más que en las pruebas y reparaciones subsecuentes.

Utilizabilidad. Un producto de software es utilizable en la medida que su estructura básica, su construcción y su documentación son utilizables por la audiencia a la que esta destinada.

Costo-Efectividad. Un producto de software es costo-efectivo en la medida que sus costos de desarrollo y operación satisfacen las necesidades de los usuarios. El costo-efectividad se expande para incluir tanto la reparabilidad como la modificabilidad.

Proceso. El proceso de la ingeniería de software concierne a las actividades de requerimientos, diseño, implementación, validación y soporte indispensables para obtener y sostener un producto de software que cumpla tanto con sus requerimientos como con las metas de la buena ingeniería.

5.2.4. PRINCIPIOS

Generalmente se acepta que en última instancia la causa de la crisis del software reside en la *complejidad*. Esta complejidad de los sistemas de software frecuentemente excede la capacidad intelectual humana. Las metodologías de software son los instrumentos más poderosos para reducir la complejidad cuando promueven los principios de la ingeniería de software.

Diseño del Sistema. El *primer principio* de la ingeniería de software establece que en el proceso de diseño de un sistema se debe asegurar que los productos resultantes satisfagan las verdaderas necesidades de los usuarios.

Diseño del Software. Las consideraciones importantes en el diseño del software.

- La *modularidad* define la separación del sistema en componentes manejables con un mínimo efecto sobre los demás módulos del software.
- La *abstracción* identifica las propiedades esenciales comunes a entidades superficialmente diferentes y hace inaccesible la información no necesaria.
- La *localización* permite que el diseño del software refleje las entidades y relaciones a las que concierne. Las metodologías orientadas a objetos amplifican la modificabilidad por la alta localización de los efectos de los cambios.
- La *uniformidad* garantiza la consistencia de software.
- La *conformidad* asegura que la información necesaria para verificar la corrección ha sido establecida explícitamente.

El *segundo principio* de la ingeniería de software establece que el empleo de métodos efectivos de diseño facilita grandemente el desarrollo de software que cumpla con las metas de la ingeniería de software.

Herramientas de Software. Una metodología está formada por una base conceptual, un conjunto de métodos y herramientas. Las herramientas incrementan la competencia profesional y reducen o eliminan fuentes de error. Por ello, el *tercer principio* de la ingeniería de software establece que la calidad y economía de un proyecto de ingeniería de software depende directamente de la calidad y efectividad de las herramientas provistas a los ingenieros de software.

Administración del Software. La administración del software concierne a la utilización efectiva de los recursos disponibles tanto para producir software bien diseñado como para mejorar la capacidad de las organizaciones dedicadas a la ingeniería de software.

Dada la importancia de la administración, el *cuarto principio* de la ingeniería de software establece que solamente habrá ingeniería de software efectiva cuando haya administración de software efectiva.

5.2.5. ASEGURAMIENTO DE LA CALIDAD DEL SOFTWARE

El aseguramiento de la calidad a través de la ingeniería de software, permite la "garantía de calidad" de los productos derivados de un proyecto, en base a métodos y técnicas de la ingeniería de software.

El *aseguramiento de la calidad* es la revisión de los productos y documentación relacionada con el software para verificar su cobertura, corrección, confiabilidad y facilidad de mantenimiento. Y, por supuesto, incluye la garantía de que un sistema cumple las especificaciones y los requerimientos para su uso y desempeño deseados.

Anteriormente eran dos las ideas subyacentes al aseguramiento de la calidad. La *primera* consistía en que el usuario del sistema de información es el elemento más importante para establecer y evaluar la calidad. El *segundo* residía en corregir problemas cuando éstos se encuentran en sus etapas iniciales que esperar a que el problema se exprese mediante quejas de los usuarios o la aparición de crisis.

Ahora para garantizar altos niveles de un producto final (el código), es imperativo también hacerlo para cada uno de los productos intermediarios como los documentos de requerimientos, de diseño, construcción e implantación. Así mismo, se verificará la conformidad o apego con los estándares de métodos y técnicas adoptados en los diferentes pasos del proceso.

La *Garantía de Calidad de Software (GCS)* es una actividad de protección que se aplica a lo largo del proceso software [36]. Son acciones que se requieren para asegurar la calidad del software. El aspecto más importante de la GCS es comprobar que el sistema realice las actividades para las que fue pensado y que además satisfaga las necesidades del cliente/usuario.

El proceso de GCS está enmarcado en la realización de las siguientes actividades:

- Realización de revisiones técnicas formales durante cada paso del proceso del software.
- Planeación y realización de una estrategia de Prueba de Software.
- Control de Cambios.
- Mecanismos de Medición.
- Registro y realización de informes.

[36] *El liderazgo en la computación como eje de la Competitividad Internacional.- Memorias del Simposium Internacional de Computación, celebrado en el CENAC - IPN - Octubre de 1996 - pp. 198.*

5.2.6. MÉTRICAS DE SOFTWARE

Las métricas de software son las herramientas que usa la Ingeniería de Software para monitorear el desarrollo de los proyectos. Las métricas de software son indispensables para la administración de las tareas de desarrollo de software, y el mejoramiento de procesos y productos. Las mediciones indican el grado de calidad y los costos asociados. Las métricas de software no sólo miden tiempos y costos, sino productividad, efectividad y calidad. El uso de métricas se comienza a llevar a cabo desde las primeras etapas del ciclo de vida del software hasta el mantenimiento.

En proyectos pequeños es posible mantenerse al tanto de su estado simplemente observándolo directamente y conversando directamente con los programadores. Esto es prácticamente imposible para los líderes de proyectos de software de mayor magnitud, pues no hay suficiente tiempo para conversar con todos los miembros del equipo en forma continua o para integrar los datos obtenidos y conseguir una perspectiva precisa del avance. Las métricas permiten aumentar el nivel de confianza del futuro éxito de un proyecto y dan la base para predicciones en proyectos nuevos.

Las prácticas adecuadas de medición mejoraran el control del proyecto y aumentan la calidad. Sin mediciones detalladas, es necesario adivinar el estado del proyecto y las metas. Las mediciones precisas de software proporcionan una base cuantitativa para apoyar decisiones críticas. Los líderes de proyecto pueden usar los datos para reasignar recursos en forma efectiva, darse cuenta de cuando es tiempo de iniciar la siguiente actividad o fase de desarrollo y enfocar recursos en áreas que necesitan más ayuda.

Sin estimaciones, los miembros del proyecto no pueden predecir cuándo lo van a terminar. Y un plan de medidas no es suficiente para evitar que un proyecto se retrase o rebase el presupuesto, pero el plan es una herramienta esencial que ayuda a los administradores a monitorear la salud de un proyecto y permitir la ejecución de acciones correctivas que ayuden a minimizar los efectos en costos y calendarios.

Solamente con el uso efectivo de mediciones, los ingenieros de software pueden estimar hasta dónde ha llegado el proyecto, cuánto falta para llegar a un objetivo (cuánto más va a costar) e incluso definir cuál es el objetivo a alcanzar.

Un ejemplo de métricas de software es el generar semanalmente un reporte del avance del proyecto. Dicho avance debe estar basado en el plan de trabajo indicando claramente cualquier retraso o problema. Todo retraso debe tener especificado el responsable y la duración, además de estar firmado por el Líder del Proyecto y el Coordinador del Proyecto por parte del usuario. Todo retraso que no pueda ser recuperado debe causar una Reprogramación.

5.2.7. AUDITORIA

La auditoría es otra forma de asegurar la calidad de la información que contiene el sistema. La auditoría implica contar con un experto que no se encuentre involucrado en la

puesta en operación y en el uso del sistema, para examinar la información, con el fin de establecer su relevancia.

Los auditores participan en la confiabilidad del sistema, estudiando los controles utilizados por el sistema para asegurar que éste realice lo que supuestamente debe de hacer. También examinan la operación de los controles de seguridad. Cualquier hallazgo lo comunican al líder de proyecto y a la organización, con el fin de mejorar la utilidad de la información generada por el sistema.

5.3. CONSTRUCCION DEL SOFTWARE

5.3.1. PRINCIPIOS

Los siguientes principios deben guiar la construcción del software.^[37]

PRINCIPIO	DESCRIPCION	OBJETIVOS
MODULARIDAD Y FRAGMENTACIÓN	Diseño de un sistema como una jerarquía de módulos	Diseñar la estructura en forma descendente con módulos que realicen funciones específicas.
ACOPLAMIENTO	La fuerza de las relaciones entre módulos	Maximizar la independencia entre los módulos.
COHESIÓN (INTEGRACIÓN)	La fuerza de las relaciones dentro de un módulo.	Maximizar la cohesión; los elementos altamente relacionados deben estar en el mismo módulo.
EXTENSIÓN DE CONTROL	Número de módulos subordinados controlados por el módulo que hace la llamada.	Limitar la extensión de control de 5 a 7 módulos.
TAMAÑO	Número de instrucciones que componen a un módulo.	Limitar el tamaño de forma que la función de todo el módulo se centre en un sólo propósito.
USO COMPARTIDO	Uso de un módulo por otros módulos	Evitar la duplicación, permitiendo que los módulos sean llamados por otros que necesitan la función de cada uno.

El compartir módulos minimiza la cantidad de software que debe diseñarse y escribirse. También minimiza el número de cambios que hay que hacer durante el mantenimiento del sistema. Al tener un módulo compartido se reduce la probabilidad de error.

[37] Senn, James.- *Análisis y Diseño de Sistemas de Información.*- Traduc. Urbina Medal Edmundo Gerardo y otros.- 2ª Edición.- Editorial McGRAW-HILL.- México.- 1994 - pp. 774.

Muchos sistemas establecen los módulos de biblioteca - procedimientos predefinidos - que se incluyen en la biblioteca de programas del sistema. La rutina se invoca rápidamente mediante un solo comando o llamada.

5.3.2. CONFIABILIDAD

Se dice que un sistema tiene confiabilidad si no produce fallas costosas o peligrosas al usarse de una manera razonable, es decir, de tal forma que el usuario típico espera que sea normal. Los analistas deben garantizar que el sistema sea confiable cuando se instala y que la confiabilidad se puede mantener después de la implantación.

Hay dos niveles de confiabilidad. El *primero* es que el sistema cumpla con los requerimientos correctos. El *segundo* nivel tiene que ver con los resultados reales que el sistema entrega al usuario.

Un *error* aparece cuando el sistema no produce los resultados esperados. Cuando se detecta que hay una diferencia entre la salida "especificada" y la salida "obtenida" al ejecutar el programa. Una *falla* es la aparición de un error del software, con un cierto peso dado por su seriedad.

Existen tres enfoques para la confiabilidad.

ENFOQUES DE CONFIABILIDAD	DESCRIPCION
PREVENCIÓN DE ERRORES	Previene la ocurrencia de errores en el software.
DETENCIÓN Y CORRECCIÓN DE ERRORES	Reconoce los errores cuando se les halla y corrige el error o el defecto del error de tal forma que el sistema no falla
TOLERANCIA A ERRORES	Reconoce los errores cuando aparecen, pero permite que el sistema siga trabajando con una ejecución degradada o aplicando reglas que instruyan al sistema para que continúe el proceso.

Las causas de error pueden ser:

- No obtener los requerimientos correctos,
- No entender correctamente los requerimientos y
- No traducir los requerimientos de una forma clara y entendible, de manera que los programadores los implanten correctamente.

Todas estas causas de error es importante detectarlos antes de la implantación.

5.3.3. CONSTRUCCIÓN DE LA BASE DE DATOS

La Construcción de la Base de Datos es la actividad de crear las bases de datos físicas para el nuevo sistema. La creación estará en función de los Scripts, los cuales serán ejecutados a través de un usuario en la base de datos. Al terminar se documentará: los resultados de la generación y revisión del proceso de creación de la base de datos y tablas, la descripción de la base de datos, y la descripción de las tablas y atributos.

En caso de existir discrepancias entre la base de datos y el diseño se elaborará un reporte de verificación de la estructura de la base de datos.

5.3.4. CONSTRUCCIÓN DE LA APLICACIÓN

La Construcción de la Aplicación es el desarrollo e integración de los módulos que componen el sistema. El programador deberá hacer uso de su lógica y capacidad analítica para hacer que el sistema realice lo esperado. El manejo de librerías, clases y procedimientos recursivos evitan que el código crezca y sobre todo dará la ventaja de que sea reutilizable.

Cada módulo terminado será probado por el programador y otra persona (ya sea otro programador o el líder de proyecto) para verificar su buen funcionamiento. Todos los programadores deben estar en constante comunicación dando a conocer entre el grupo sus clases, procedimientos generales o descubrimientos de funciones sobre la herramienta que están utilizando. El tener una buena comunicación también evitará que surjan problemas en la integración de los módulos.

Bibliotecas de código reutilizable

Por lo general los módulos fueron desarrollados para emplearse en otros sistemas y si los módulos son reutilizables pueden convertirse en una herramienta muy valiosa para el desarrollo de futuros prototipos que funcionen y sistemas terminados rápidamente con un costo relativamente bajo. Si es necesario se pueden recuperar y modificar fácilmente varios módulos de código reutilizable.

5.3.5. ELABORACIÓN DE LOS DISCOS DE INSTALACIÓN

La elaboración de los discos de instalación es parte de la construcción del software una vez que el producto está terminado en su totalidad. Existen varias herramientas que facilitan la generación de discos de instalación como. Visual Basic, Visual C, SDK, etc.

La propiedad de estos discos es que se obtiene una copia compactada de la aplicación de manera que al instalarla en una máquina cliente e insertar el primer disco de una serie de ellos, el instalador empieza a verificar el ambiente físico de la PC, checando espacio en disco, directorios existentes, configuraciones e informando al usuario de cada verificación, dándole la oportunidad de especificar parámetros de instalación o cancelarla en cualquier momento. Cada disco es insertado de manera sucesiva según lo vaya pidiendo el instalador.

Si el instalador detecta que los recursos de la maquina son insuficientes automáticamente cancelará el proceso.

El uso de los discos ayuda a los participantes de la instalación del sistema C/S y principalmente al usuario a realizar rápida y confiablemente la copia del sistema en los equipos. Aunque el instalador actúa de forma amigable con el usuario es necesario elaborar una guía de instalación especificando a detalle el proceso.

Para el sistema SISEFIN los discos de instalación fueron construidos a través dos herramientas Visual C y SDK generándose un total de cuatro discos.

5.4. PRUEBA DEL SOFTWARE

5.4.1. DEFINICIÓN

La prueba del sistema es un proceso para demostrar que no hay errores en un programa. Representa una revisión final de las características funcionales y operativas del código.

El enfoque más útil y práctico, es que la *prueba* es el proceso de ejecutar un programa con la intención explícita de hallar errores, es decir, hacer que el programa falle. El examinador, que puede ser el líder de proyecto, el programador, o especialista entrenado en la prueba de software, está tratando realmente de hacer que el programa falle. Así, una prueba exitosa es aquella que encuentra un error.^[38]

La evaluación se debe llevar a todo lo largo del desarrollo del sistema, no sólo al final. Los analistas saben que un programa de prueba efectivo no garantiza la confiabilidad del sistema. La confiabilidad es asunto del diseño.

Los usuarios son los clientes y determinan si un sistema es bueno o no. Cualquier intento de un líder de proyecto por convencer a los usuarios de lo sabio o correcto de un cierto enfoque no sirve de nada si los usuarios no piensan que el sistema funciona de manera razonable. Si los usuarios no están convencidos que el sistema hace lo correcto, no es un sistema confiable.

5.4.2. OBJETIVOS

- La prueba es un proceso de ejecución de un programa con la intención de descubrir un error.
- Un buen caso de prueba es aquel que tiene alta probabilidad de mostrar un error no descubierto hasta entonces.
- Una prueba tiene éxito si descubre un error no descubierto hasta entonces.

[38] Senn, James.- *Análisis y Diseño de Sistemas de Información.- Traduc. Urbina Medal Edmundo Gerardo y otros.- 2ª Edición.- Editorial McGRAW-HILL.- México.- 1994 - pp. 793.*

La principal función de una prueba de software es proporcionar una configuración de prueba (plan y procedimientos de prueba, casos de prueba) que ayude a determinar que el sistema no contiene determinados errores, es decir, la prueba no puede asegurar la ausencia de defectos; sino más bien la existencia de éstos. Las pruebas de software se han desarrollado con el fin de proporcionar un nivel de confianza adecuado de que el software es correcto.

5.4.3. VERIFICACIÓN, VALIDACIÓN Y CERTIFICACIÓN

Como la prueba, la *verificación* tiene la intención de hallar errores. Se lleva a cabo ejecutando un programa en un ambiente simulado. La *validación* se refiere al proceso del uso del software en un ambiente no simulado para hallar sus errores. La *retroalimentación* de la fase de validación generalmente produce cambios en el software para resolver los errores y fallas que se descubren.

La verificación es una manera de escudriñar la programación y el desarrollo global, resaltar los problemas del sistema y permitir que el programador responsable de la sección, realice cambios correspondientes. La inspección puede realizarse cada vez que concluya una porción de codificación.

La *certificación* del software es una garantía por escrito de lo correcto de un sistema.

5.4.4. ESTRATEGIAS DE PRUEBA

La estrategia de trabajo integra las técnicas de prueba en una serie de pasos bien planificados que dan como resultado una construcción correcta del software. Los pasos previstos son:

- La planificación de la prueba.
- El diseño de casos de prueba.
- La ejecución de la prueba.
- La evaluación de los datos resultantes.

Un *caso de prueba* es un conjunto de datos que el sistema procesará como entrada normal con la intención de encontrar errores. Sin embargo, los datos se crean para determinar si el sistema los procesará correctamente. La prueba la lleva a cabo el que desarrolla el software y para proyectos grandes un grupo de prueba.

Hay dos estrategias generales para la prueba de software.

Prueba de Código: La estrategia de prueba de código examina la lógica del programa permitiendo módulos más confiables. Para seguir este método de prueba, el programador desarrolla casos de prueba que produzcan la ejecución de cada instrucción del sistema, es decir, se prueba cada ruta del programa. Una *ruta* es una combinación específica de condiciones manejadas por el programa.

A primera vista, la prueba de código parece ser un método ideal para probar el software. Sin embargo, es casi imposible hacer una prueba exhaustiva de esta naturaleza. Y, aunque la prueba de código se pueda llevar a cabo en su totalidad, no es una garantía en contra de las fallas del software.

Prueba de Especificación o Integridad de Información: Para llevar a cabo esta prueba se examina las especificaciones que señalan lo que el programa debe hacer y cómo lo debe llevar a cabo bajo diferentes condiciones. Después se desarrollan casos de prueba para cada condición o combinación de condiciones y se mandan para su procesamiento para determinar si el sistema cumple con las salidas correctas y realiza sus funciones adecuadamente en base a los requerimientos especificados. La prueba de especificación es la estrategia más eficiente, ya que se centra en la forma que se espera se use el software.

5.4.5. MANEJO DE LAS PRACTICAS DE PRUEBA

Independientemente de cual estrategia se aplique, hay ciertas prácticas preferidas para garantizar que la prueba sea útil.

PRUEBAS PARCIALES

En las pruebas parciales, se prueba los programas que conforman a un sistema..

Prueba de Unidad: Las pruebas parciales se centran primero en los módulos independientes entre sí, para localizar los errores. Esto permite al que realice la prueba detectar errores en el código y lógica contenidos dentro de ese único módulo. En el módulo se prueba:

- La interfaz.
- Las estructuras de datos locales.
- Las condiciones límite.
- Las opciones.
- Los caminos independientes.
- Los caminos de manejo de errores.

Prueba de Integración: Son las pruebas parciales que se dirigen a la construcción de la arquitectura del software. Las pruebas de integración se pueden llevar a cabo en forma *ascendente*, comenzando con los módulos más pequeños y de nivel inferior y continuando de uno en uno. Para ejecutar un módulo inferior se crea un programa corto que ejecute el módulo y proporcione los datos necesarios. Otra forma de prueba es la *descendente* se empieza con los módulos de nivel superior. Sin embargo, puesto que no se encuentra con las rutinas de nivel inferior (ya que no se están probando), se escriben fragmentos de programas que son llamados por el módulo de nivel superior regresando un mensaje de que ocurrió la interacción apropiada.

La *prueba combinada* conjunta las pruebas descendentes con las pruebas ascendentes.

PRUEBA DE SISTEMAS

La prueba de sistemas no prueba el software en sí, sino busca las discrepancias entre el sistema y su objetivo original, especificaciones y documentación del sistema. La prueba de sistemas también debe verificar que los parámetros de almacenamiento son adecuados.

PRUEBA DE VALIDACION

Se validan los requerimientos establecidos comparándolos con el sistema que ha sido construido. Los métodos pueden ser repaso de la configuración, y pruebas alfa y beta.

Se elige un conjunto de instalaciones usuarias que ponen a trabajar el sistema en un ambiente real. Estas instalaciones de *prueba beta* usan el sistema en las actividades cotidianas; procesan transacciones en directo y producen salidas normales del sistema. El sistema está a prueba en toda la extensión de la palabra, excepto que los usuarios están advertidos de que están usando un sistema que puede fallar. En el curso de la validación del sistema, puede ocurrir una falla y el software será modificado.

PRUEBAS ESPECIALES DE SISTEMAS

Existen otras pruebas en una categoría especial ya que no se centran en el funcionamiento normal del sistema. Hay seis pruebas básicas.

TIPO DE PRUEBA	DESCRIPCION
Prueba de Carga Máxima o Resistencia	Determinar si el sistema manejará el volumen de actividades que ocurran cuando el sistema esté en el punto alto de su demanda de procesamiento.
Prueba de Almacenamiento	Determinar la capacidad del sistema para almacenar datos de transacciones en un disco u otros archivos.
Prueba de Rendimiento	Determinar cuánto tiempo se lleva recibir una respuesta a una consulta, hacer una copia de respaldo o procesar los datos de una transacción.
Prueba de Recuperación	Determinar la capacidad del usuario para recuperar datos o restablecer el sistema después de una falla.
Prueba de Procedimientos	Determinar la claridad de la documentación en los aspectos de operación y uso de un sistema, haciendo que los usuarios lleve a cabo exactamente lo que el manual pide.
Prueba de Factores Humanos	Determinar como utilizarán los usuarios el sistema al procesar datos o preparar informes. Observar sus reacciones ante las respuestas del sistema.
Prueba de Seguridad	Determina la confiabilidad del esquema de seguridad del al ser accesado por varios perfiles de usuarios.

Todas estas pruebas al sistema deben llevarse a cabo antes de la implantación. Entre "mejor" sea el criterio que determina si el conjunto de pruebas es "adecuado" más es la confianza de que el programa sea correcto.

5.4.6. DISEÑO DE DATOS DE PRUEBA

Hay dos fuentes muy diferentes de datos de prueba, *reales* y *artificiales*.

Uso de datos de pruebas reales: Los datos de prueba reales son aquellos que se extraen de los archivos de la organización. Después de que un sistema está terminado parcialmente, es frecuente que los programadores o líderes de proyecto pidan a los usuarios que introduzcan un conjunto de datos de sus actividades normales. Aunque los datos reales mostrarán cómo funciona el sistema para los requerimientos típicos de procesamiento, suponiendo que los datos reales introducidos son en realidad típicos, generalmente dichos datos no probarán todas las combinaciones o formatos que puedan entrar al sistema. El sesgo hacia los valores típicos no proporciona entonces una verdadera prueba del sistema y de hecho ignora los casos más probables que causan la falla del mismo.

Uso de datos artificiales: Los datos de prueba artificiales se crean solamente con fines de prueba, ya que se pueden generar para probar todas las combinaciones de formatos y valores. Los programas de prueba más efectivos usan datos de prueba artificiales generados por personas distintas de las que escribieron los programas. Frecuentemente, un equipo independientemente formula un plan de prueba, usando las especificaciones del sistema.

5.4.7. REVISIONES POR PARTE DEL USUARIO

Es importante que durante la construcción, un usuario con capacidad de sugerir y autorizar este informado sobre los avances del proyecto haciéndolo participe en las pruebas y revisión de los módulos que se vayan terminando. El usuario se encargará de determinar si el sistema cumple con las salidas correctas y realiza sus funciones adecuadamente en base a los requerimientos especificados.

Por lo regular, el usuario al estar operando con el sistema, puede encontrar ciertos detalles que tienen que cambiarse en cuanto a los requerimientos. El líder de proyecto evaluará las observaciones del usuario y si estos cambios no repercuten considerablemente en la programación o en la estructura de la base de datos podrán ser realizados, de lo contrario se expondrá al usuario las razones por las cuales no es conveniente efectuar las modificaciones tomando como principal justificación el retraso del proyecto en cuanto a tiempos programados y proponiéndole que en la etapa de mantenimiento del sistema se considerará el cambio. Del usuario dependerá decidir si realmente necesita que se realice ese requerimiento autorizando que se desface el calendario de actividades del proyecto o que se retome el cambio durante el mantenimiento del sistema.

5.4.8. CORRECCIONES DE LA APLICACIÓN

Esta actividad tiene como propósito ajustar los módulos de la aplicación que de acuerdo a las pruebas realizadas no cumplan con los requerimientos o presenten errores de lógica. Así mismo se documentarán las pruebas que se realizaron sobre la sistema y se reportarán los resultados relevantes de las pruebas aplicadas, destacando los errores y las adecuaciones que se hayan realizado al sistema.

5.5. DOCUMENTACION

5.5.1. TÉCNICAS DE DOCUMENTACIÓN

Para que los programas se documenten de manera adecuada se requieren de técnicas que auxilien a distribuir el esfuerzo de programación en módulos manejables. El software y los procedimientos deben quedar documentados de manera tal, que sean codificados en un formato de fácil acceso. La documentación permite a los usuarios, programadores y analistas "ver" el sistema, su software y los procedimientos, sin necesidad de una interacción directa.

Son pocas las oportunidades de que la gente que concibió e instaló el sistema original sea la misma que le dará mantenimiento. De manera consistente, una documentación bien estructurada reducirá el número requerido de horas para que se conozca el sistema, antes de poder darle mantenimiento.

5.5.2. ELABORACIÓN DE MANUALES

Los manuales son documentos de referencia para el manejo del sistema. Y como tal, necesita organizarse de una manera lógica, con un pensamiento cuidadoso acorde con las circunstancias en las cuales se utilizará.

Los manuales son documentos que tienen como propósito incorporar las referencias técnicas que puede necesitar el usuario u operador para identificar y resolver situaciones sobre la operación y uso del sistema.

Los manuales pueden contener pasos para realizar diferentes transacciones; instrucciones de cómo resolver problemas de operación y qué hacer si algo no funciona. Es esencial que los manuales se consideren como documentos actuales, más que históricos; para que los manuales sean útiles deberán actualizarse.

De acuerdo con su propósito hay tres tipos de manuales:

TIPOS DE MANUALES	DEFINICION
MANUAL TÉCNICO	Este manual contiene información necesaria para la operación, revisión y adecuación del sistema.

TIPOS DE MANUALES	DEFINICIÓN
MANUAL DE USUARIO	Este manual permitirá al usuario obtener la información producida por el sistema y comprender las funciones que realiza.
MANUAL DE ENTRENAMIENTO	Este manual contiene el material de apoyo didáctico que plantea la introducción de un nuevo usuario al uso del sistema. Asimismo se integrará un plan de desarrollo en función a los conocimientos necesarios para operar y utilizar el sistema.

5.5.3. EVALUACIÓN DE LA DOCUMENTACIÓN

Lo peor que puede ocurrir cuando se implanta un nuevo sistema es que el software no funcione. Sin embargo, algo casi tan malo es que en la documentación los usuarios lean una cosa y que el sistema haga otra. En ese momento, tanto la aplicación como el líder de proyecto pierden credibilidad.

La documentación que va a dar a las manos del usuario final, ya sea en forma de manuales, sistemas de ayuda o diagramas sencillos, debe coincidir con la forma en que funcione la aplicación. Los líderes de proyecto tienen que asegurarse de que ponen la atención suficiente para mantener este aspecto de credibilidad, evaluando la utilidad y comprensión de la documentación del sistema.

5.6. IMPLANTACION

5.6.1. DEFINICIÓN

El nuevo sistema puede ser totalmente nuevo y reemplazar al sistema que hay, ya sea manual o automatizado; o bien ser una modificación importante de un sistema existente. En cualquier caso, la adecuada implantación es esencial para lograr un sistema confiable y que cumpla con las necesidades de la organización. Una implantación exitosa no garantiza el mejoramiento de la organización que use el nuevo sistema (eso es una cuestión de diseño), pero su instalación inadecuada lo impediría.

Se denomina implantación al proceso que asegura la operatividad del sistema y que permite al usuario obtener beneficios por su operación^[39]. Los tres aspectos de la implantación son: *capacitación de personal, procedimientos de conversión y la revisión después de la implantación.*

Antes de iniciar la implantación se requiere de elaborar y presentar al usuario un plan que incluye las actividades que deben ocurrir al implantar el sistema nuevo y ponerlo en

[39] Kendall, Kenneth y Kendall, Julie.- *Análisis y Diseño de Sistemas*. - Traduc. Hernández López Héctor y otros.- 1ª Edición.- Editorial Prentice-Hall Hispanoamericana.- México.- 1994 - pp. 791

operación. En este programa se indica cuando debe llevarse a cabo cada actividad. Para ello se identifica los siguientes puntos:

- **Programa de Visitas** a los lugares se va a instalar el sistema C/S, se identifican responsables de cada actividad y funciones, y determinando las fechas para la implantación.
- **Programa de Instalación** en la creación del ambiente idóneo para la operación del sistema Cliente/Servidor.
- **Programa de Capacitación** a los usuarios del sistema C/S.
- **Programa de Operación** del Sistema C/S
- **Liberación**

5.7. INSTALACION

5.7.1. DEFINICIÓN

La instalación del Sistema Cliente/Servidor es la creación de su ambiente físico y lógico para su operación.

La instalación comienza con la verificación de la infraestructura y adecuación para que funcione como una plataforma Cliente/Servidor, y posteriormente se realiza la instalación del sistema C/S.

5.7.2. MÉTODOS DE CONVERSIÓN

La conversión es el proceso de cambiar el sistema anterior al nuevo y existen cuatro métodos o estrategias para llevar a cabo una conversión de sistemas, muchas situaciones dictan el uso de un método sobre los demás, aún cuando los otros puedan ser más benéficos. En general, la conversión de sistemas debe realizarse lo más rápidamente posible. Los periodos de conversión largos aumentan la posible frustración y la dificultad de la tarea para las personas implicadas, incluyendo tanto a los analistas como a los usuarios.

Sistemas paralelos.

El método más seguro para convertir un sistema anterior a uno nuevo es el de utilizar ambos sistemas en *paralelo*. Como este enfoque, los usuarios siguen operando el sistema anterior de la forma acostumbrada, pero también comienzan a usar el sistema nuevo y en caso de surgir problemas, como errores en el procesamiento o incapacidad de manejar ciertos tipos de transacciones en el nuevo sistema, la organización pueda regresar al sistema anterior sin pérdida de tiempo, ingresos o servicios.

Las desventajas del enfoque son que los costos del sistema se duplican, ya que existen dos sistemas. En algunos casos es necesario contratar personal temporal para operar ambos sistemas en paralelo. En segundo lugar, el hecho de que los usuarios sepan que es posible regresar a las formas antiguas puede ser una desventaja si existe una resistencia potencial al cambio o si los usuarios prefieren el sistema anterior. En otras palabras, el sistema nuevo puede no tener un juicio justo.

Conversión directa

El método de *conversión directa* transforma el sistema anterior al nuevo de manera rápida. No hay actividades paralelas, el sistema nuevo reemplaza completamente al anterior, de manera que los usuarios no cuentan con los métodos anteriores. Psicológicamente, obliga a todos los usuarios a hacer trabajar el sistema nuevo, pues ellos no tienen otro sistema al cual recurrir.

La ventaja de no contar con un sistema de respaldo puede convertirse en una desventaja si surgen problemas serios con el sistema nuevo. En algunos casos, las organizaciones llegan a parar las operaciones cuando se presentan los problemas para que las dificultades pueden ser corregidas invirtiendo tiempo e incrementándose los costos. Y si los usuarios saben que un sistema se detiene por dificultades, ellos no tendrán toda la confianza de que el sistema será confiable, aún cuando el líder de proyecto diga que se han corregido los problemas.

La conversión aplicada al sistema SISEFIN fue de manera directa, por tratarse del primer sistema automatizado para este tipo de procesos. La conversión se dirigió adecuadamente y los riesgos fueron nulos, por lo que las probabilidades de éxito se incrementaron.

Enfoque piloto

En este método se implanta una versión de trabajo del sistema en una parte de la organización, como una sola área de trabajo o un departamento. Los usuarios de esta área usualmente saben que están probando un nuevo sistema y que se pueden hacer cambios para manejar el sistema. Cuando el sistema se ha probado en su totalidad, se instala en toda la organización, ya sea completamente y de una sola vez (método de la conversión directa) o en forma gradual (método por etapas).

Este enfoque tiene la ventaja de proporcionar una sólida base de prueba antes de la implantación total. Sin embargo, si no se conduce bien la implantación, los usuarios pueden llegar a pensar que el sistema sigue teniendo problemas y que no es posible confiar en él.

Conversión por Etapas.

El método por etapas se usa cuando no es posible instalar de golpe un nuevo sistema en toda la organización. Algunos usuarios comenzarán a tomar ventajas del sistema nuevo antes que otros. Las ventajas incluyen permitir que los usuarios se involucren con el nuevo sistema de manera gradual. Las desventajas es que los largos periodos de conversión por

etapas crean dificultades para los líderes de proyecto, independientemente de si la conversión marcha bien o no.

Conversión Distribuida.

Esto se refiere a una situación en la cual se contemplan muchas instalaciones del mismo sistema. Una conversión total se realiza en uno de los sitios (o con cualquiera de las anteriores conversiones). Cuando esta conversión se concluye con éxito, se realizan las otras conversiones en otros sitios.

Una ventaja de la conversión distribuida es que los problemas pueden detectarse y manejarse, más que inducirse a todos los demás sitios. Una desventaja es que aún cuando una conversión tenga éxito, cada sitio tendrá sus propias peculiaridades que deberán resolverse.

5.7.3. ELABORACIÓN DE UN PROGRAMA DE INSTALACIÓN

En el Programa de Instalación se especifican las acciones a seguir para la creación del ambiente físico y lógico del sistema C/S. Las actividades principales que se tienen que cubrir durante la instalación de un sistema son:

- a) Identificación y verificación de Hardware, Software y Humanware de cada uno de los lugares de Instalación.
- b) Verificación o Construcción de la arquitectura Cliente/Servidor.
 - Verificación o Instalación de la Red.
 - Verificación o Configuración del Servidor.
 - Verificación o Instalación del Sistema Operativo de Red.
 - Verificación o Instalación del Software de Aplicación.
- c) Instalación del Sistema C/S.
 - En el servidor.
 - Verificación y/o creación de espacio en disco duro y memoria RAM.
 - Creación de la estructura de datos del sistema C/S en el RDBMS.
 - Creación de usuarios y privilegios del sistema C/S.
 - Verificación de los canales de comunicación.
 - Pruebas de conectividad local y remota.
 - En el Cliente.
 - Verificación y/o creación de espacio en disco duro y memoria RAM.
 - Generación del ambiente de operación del sistema en el front end.
 - Actualización del archivo HOSTS.
 - Verificación o configuración del middleware.

Verificación o configuración del software de red.

Carga de la aplicación.

Verificación de enlace en la aplicación C/S.

Pruebas de conectividad locales y remotas.

d) Pruebas y Ajustes locales e integrales del sistema C/S.

5.8. CAPACITACION.

5.8.1. DEFINICIÓN

La capacitación es el proceso educativo a los usuarios sobre la utilización del nuevo sistema. Aquellos que estén asociados con el sistema o afectados por el mismo deben conocer con detalle cuales serán sus papeles, como pueden usar el sistema y que hará o no hará el sistema. Tanto los operadores como los usuarios del sistema necesitan capacitación.

[40]

Los buenos gerentes ven la capacitación como una valiosa inversión. Los gerentes eficientes saben que ningún sistema será eficiente a menos que su personal lo use sabia y adecuadamente, para lo cual se necesita una buena capacitación. Con la capacitación, los gerentes y los empleados pueden ir mas allá del dominio del nuevo sistema y aprender mas sobre su negocio.

5.8.2. CAPACITACIÓN DE OPERADORES DE SISTEMAS

Muchos sistemas dependen del personal del centro de computo, el cual es responsable de mantener el equipo funcionando, así como de proporcionar el servicio de apoyo necesario. Su capacitación debe asegurar que pueden manejar todas las operaciones posibles, tanto rutinarias como extraordinarias.

En la construcción de una arquitectura Cliente/Servidor, la capacitación de los operadores debe incluir aspectos tan básicos como instalación, configuración, operación y mantenimiento de la infraestructura C/S como del sistema C/S. También debe capacitarse a los operadores acerca de los desperfectos más comunes, como reconocerlos y que pasos llevar a cabo cuando ocurran. Como parte de su capacitación, se debe dar a los operadores una lista de formas de resolver los problemas y que identifique los posibles problemas y solución, así como los nombres y números telefónicos de las personas a quien buscar cuando surjan problemas inesperados.

La capacitación también necesita la familiarización con los procedimientos de ejecución del nuevo sistema. Estos procedimientos permiten a los operadores familiarizarse con las acciones que deben realizar como copiar archivos o hacer funcionar sistemas de

[40] Senn, James.- *Análisis y Diseño de Sistemas de Información.* - Traduc. Urbino Medal Edmundo Gerardo y otros. - 2ª Edición. - Editorial McGRAW-HILL. - México. - 1994 - pp. 819.

información y saber cuando deben ocurrir dichas acciones. Además, sabrán cuanto tiempo le llevara a las aplicaciones correr bajo condiciones normales. Esta información es importante, tanto para que los usuarios puedan planear su trabajo como para identificar los sistemas que corran más rápido o más lento de lo esperado (una señal que usualmente indica problemas con la corrida).

5.8.3. CAPACITACIÓN DE USUARIOS

En estos casos, primero se debe capacitar a los usuarios sobre como operar el equipo, especialmente para los nuevos usuarios que no están familiarizados con las computadoras o con la nueva tecnología C/S.

La capacitación de los usuarios también debe incluir la identificación de los problemas, determinando si el problema que surge es causado por el equipo o el software o por algo que ellos hicieron al usar el sistema. Al incluir una guía de identificación de problemas en la documentación de los sistemas, se obtendrá una referencia útil por mucho tiempo después de terminar el periodo de capacitación.

La mayor parte de la capacitación del usuario tiene que ver con la operación del sistema en sí. La capacitación enfatiza los métodos a seguir en la captura de datos, la edición de datos, la formulación de consultas y el borrado de registros de datos.

En resumen, hay dos aspectos de la capacitación de los usuarios: la familiarización con el sistema de procesamiento en sí (es decir, el equipo usado para la captura y procesamiento de datos) y la capacitación en el uso de la aplicación (es decir, el software que acepta los datos, los procesa y produce los resultados). La debilidad de cualquier aspecto de la capacitación trae la posibilidad de llegar a situaciones embarazosas que producirían en el usuario frustración, errores, o ambos. Una buena documentación, aunque esencial, no reemplaza la capacitación. No hay sustituto para la operación directa del sistema mientras que se aprende su uso.

5.8.4. MÉTODOS DE CAPACITACIÓN

La capacitación de operadores y usuarios se puede obtener de distintas formas. Las actividades de capacitación pueden llevarse a cabo en las instalaciones del proveedor, en locales rentados (como hoteles o campus de universidades) o en las instalaciones de la empresa. Los métodos y contenido de la capacitación varían a menudo, dependiendo de la fuente y del lugar de la capacitación.

Capacitación por el proveedor: A menudo, la mejor fuente de capacitación para un equipo es el proveedor de este. Los cursos impartidos, cubren todos los aspectos del uso de la infraestructura y el sistema C/S. Si surgen preguntas, estas se pueden contestar enseguida. Es recomendable mandar a usuarios a cursos fuera de la empresa para obtener una capacitación más amplia sobre la construcción y administración de la arquitectura C/S, los beneficios de compartir preguntas, problemas y experiencias de otras compañías son muy

importantes. Los cursos cortos a menudo implican tiempo y costos adicionales para viajar a otras ciudades.

Capacitación "en casa": La ventaja de ofrecer capacitación para el sistema en el lugar de la empresa es que la instrucción se puede unir a la organización enfocándose a procedimientos especiales usados por ella. Además de que los usuarios no necesitan viajar. También hay desventajas. El simple hecho de que los empleados estén en su propio medio es una distracción, ya que las llamadas telefónicas y las emergencias pueden interrumpir las sesiones de capacitación.

Durante la capacitación, el personal de sistemas debe mantenerse atento a los comentarios hechos por los usuarios y a los problemas que éstos pueden hallar. Aun con las pruebas, es posible que surjan durante la capacitación manejos incorrectos del teclado al introducir los datos, transacciones inesperadas o formas no usuales de preparar transacciones. El instructor debe asegurarse de involucrar al personal de sistemas cuando se hallen errores en el diseño, a la vez que tiene que ayudar a los usuarios que son renuentes a cambiar sus viejas formas por los nuevos métodos necesarios para usar el sistema.

5.8.5. MANUALES

La capacitación se puede ofrecer por medio de material instruccional. Los manuales de capacitación son aceptables para la familiarización, pero las experiencias de usar realmente el equipo, hacer y corregir errores y hallar situaciones inesperadas son la mejor y más duradera forma de aprender.

Los manuales de capacitación generalmente tienen dos enfoques. Algunos hacen que el usuario trabaje distintas actividades paso por paso en el orden adecuado. El otro enfoque común consiste en crear un caso de estudio que incluya todas las situaciones que se hallan frecuentemente, las cuales pueden ser manejadas por el sistema y que el usuario debería conocer.

5.8.6. PLAN DE CAPACITACIÓN

En este plan se estiman los tiempos requeridos en la capacitación, la estrategia para concentrar al personal y el equipo necesario para la práctica y operación del sistema.

5.9. RESPALDO Y RECUPERACION DE DATOS

5.9.1. CAUSAS POTENCIALES DE LA PERDIDA DE DATOS

Debido a que los sistemas de información C/S no son inmunes a problemas que puedan conducir a la pérdida accidental de los datos, los líderes de proyecto están interesados en establecer procedimientos para el respaldo y recuperación de la información durante toda la vida del sistema.

Algunas de las razones para la necesidad de los respaldos son el resultado del uso normal del sistema, pero otras no, incluso bajo los mejores calendarios de mantenimiento preventivo, los componentes pueden fallar y provocar la pérdida de datos. Los problemas que se pueden presentar son originados por:

- Errores en los discos
- Borrado y modificado accidental o malintencionado de los datos.
- Virus destructivos
- Desastres naturales
- Robo
- Fallas súbitas en la energía eléctrica

Los datos son el componente más importante de un sistema de información. Por lo que se debe diseñar procedimientos de respaldo efectivos para proteger a la organización contra los datos perdidos o dañados.

5.9.2. MÉTODOS DE RESPALDO

Respaldo una base de datos es hacer una copia de sus tablas de sistema, objetos definidos por el usuario y sus datos. Los archivos de respaldo duplican el conjunto original de datos. Existen varios métodos efectivos para mantener copias de respaldo.

Vaciado de una base de datos

El vaciado de base de datos es el proceso de duplicar la base de datos para crear otra copia de él. El vaciado produce una copia adicional a la vez que deja la versión original sin alterarla. Si la copia actual de la base de datos se daña o borra, se puede procesar una copia y actualizarla con las transacciones acumuladas desde que se realizó el último vaciado.

Copias imágenes del registro

En los sistemas muy grandes, el vaciado regular de los datos no es posible. Un método alternativo de respaldo descansa en la presencia de copias los archivos de transacciones. Si se encuentra que ciertos registros se cambiaron en forma inadecuada, las transacciones se pueden ejecutar hacia atrás para restaurar en un punto donde se sepa que eran correctas. Este procesamiento se llama *retorno*. Análogamente, en caso necesario, las copias de respaldo de los registros se pueden actualizar reprocesándolos con las copias de las transacciones. Este proceso se llama *traer hacia adelante a los datos*.

Las bases de datos y sus transacciones son respaldadas en dispositivos de respaldo, pudiendo ser estos discos o cintas, se pueden guardar varias bases de datos en un mismo respaldo.

5.9.3. CONSIDERACIONES DE RESPALDO

Los líderes de proyecto son los responsables de diseñar un plan sólido de respaldo de datos antes de que una aplicación se ponga a prueba o en operación. Se deberá probar el plan de respaldo antes de liberar el sistema para comprobar su eficiencia. Y tener presente que las copias de respaldo se necesitarán tarde o temprano para mantener al sistema funcionando.

La frecuencia y tipos de respaldos generalmente dependerán de dos factores:

1. La cantidad aceptable de trabajo que se puede perder y
2. El volumen de transacciones de la base de datos.

Se recomienda respaldar las bases de datos semanalmente y las transacciones diariamente para la mayoría de las aplicaciones típicas.

5.10. MONITOREO DEL SISTEMA

5.10.1. DEFINICIÓN

Después de implantar el sistema C/S se hace un *monitoreo* del mismo que es la revisión para determinar que tan bien esta funcionando el sistema, como ha sido aceptado y cuales ajustes con necesarios. El monitoreo es conducido igualmente por los usuarios y el equipo del proyecto; y consiste en operar el sistema con información y situaciones reales, generar reportes y gráficas, y realizar pruebas especiales.

Este proceso también es importante para recabar información para el mantenimiento del sistema C/S. Puesto que ningún sistema es en realidad totalmente completo, el sistema permanecerá mientras no se requieran cambios debido a desarrollos internos (Actividades de la empresa) o bien, situaciones externas (nuevos requisitos legales, estándares de la industria). El monitoreo después de la implantación es la primera fuente de información de los requisitos de mantenimiento.

5.10.2. PROPÓSITOS DEL MONITOREO

Los propósitos fundamentales durante el monitoreo es determinar:

- Si el sistema cumplió su objetivo. Los líderes de proyecto desean saber si el nivel de desempeño de los usuarios ha mejorado y si el sistema esta produciendo el resultado deseado. Si ninguna de estas expectativas sucede, es oportuno preguntarse si el sistema se puede considerar exitoso.
- La calidad de la salida del sistema. El interés surgido durante el análisis y el diseño respecto a la precisión de la información, lo oportuno de su presentación, su cobertura y lo apropiado del formato siguen siendo indicadores de la calidad del sistema. Los

reportes de difícil interpretación o que no tengan suficiente información para ser útiles deberán ser rediseñados.

- La facilidad en el uso y la tendencia hacia los errores en la captura de datos.
- La confianza del usuario que es por lo general un indicador de la calidad del sistema.
- Los costos y beneficios económicos que se deben comparar Al impacto del sistema.

En general, los sistemas que son fáciles de usar y que dan distintos beneficios son aceptados y bien recibidos por la gente. Sin embargo, aunque se este usando un sistema que de como resultado un mejor desempeño, si el sistema en sí es difícil de usar, no es poco común que se evite el uso de ese sistema.

5.11. LIBERACION

5.11.1. ACTA DE ACEPTACIÓN

La *Liberación del Sistema* es la entrega oficial del software y su documentación final. Para ello se elabora un *Acta de Aceptación* que contendrá los detalles que justifican la aceptación de la aplicación y sus componentes, estableciendo el seguimiento de lo estipulado en la metodología y políticas relativas; cuales fueron los productos desarrollados; cuáles no fueron desarrollados y porqué; algunos controles de seguimiento estadístico; elementos para la revisión del proyecto terminado; y el cumplimiento total de lo establecido en el plan de trabajo del proyecto.

También se elabora el documento de *Relación de los Productos de la Aplicación* presentando un índice de contenidos que relacione los productos (Planes, módulos, bases, manuales, discos de instalación, guías, etc.; incorporando una breve descripción de cada uno de los productos) que componen la aplicación y que fueron entregados a los usuarios.

Después de la fecha de la firma del presente documento, será planeada y presupuestada de manera independiente y sometida a aprobación por parte del usuario para poder llevarse a cabo el *mantenimiento evolutivo del sistema*

5.12. MANTENIMIENTO

5.12.1. DEFINICIÓN

El mantenimiento es un aspecto más del desarrollo de sistemas de información. Sin embargo, efectuar cambios y ajustes no necesariamente indica la corrección de errores o la ocurrencia de problemas. Entre los cambios más frecuentes solicitados por los usuarios finales se encuentra la adición de información al formato de un reporte. Quizá sea necesario corregir algún descuido que ocurrió durante el proceso de desarrollo. Quizá sea necesario

añadir características para la detección de errores con la finalidad de evitar que los usuarios del sistema emprendan por equivocación una acción no deseada.

Todas estas situaciones son realidades del mantenimiento de aplicaciones. Cuando se presentan, sin embargo, son un buen indicador de que el sistema se está utilizando, de que tiene una función útil y de que los usuarios no lo están "archivando".

5.12.2. PUNTOS DE MANTENIMIENTO

Cuando se instalan los sistemas, generalmente se usan por periodos largos. Debido al uso de un sistema después de que se ha implantado por completo, se deben tomar precauciones para garantizar que la necesidad del mantenimiento se controle mediante el *diseño* y la *prueba*.

Una vez que los sistemas se instalan, la necesidad de depurar y corregir errores o las fallas en caso de emergencias es relativamente baja. Por lo que el mantenimiento de los sistemas implica en un alto porcentaje, adaptaciones de versiones anteriores del software debido a que las organizaciones están en un estado de flujo constante. La mayor cantidad de trabajo de mantenimiento es para cumplir con peticiones de los usuarios, mejorar la documentación o recodificar los componentes del sistema para una mayor eficiencia.

5.12.3. TIPOS DE MANTENIMIENTO

Existen tres tipos de mantenimiento el *Correctivo*, *Adaptativo* y *Perfectivo*.

CATEGORIA	ACTIVIDAD
CORRECTIVO	Ajustes de emergencia, depuración rutinaria.
ADAPTATIVO	Inclusión de cambios a los datos y archivos, así como al hardware y software del sistema.
PERFECTIVO	Mejoras solicitadas por los usuarios, mejoras en la documentación, recodificación para mejorar la eficiencia computacional.

5.12.4. REUTILIZACIÓN DE SOFTWARE

El mantenimiento del software (mejoramiento y reparación) continua siendo desproporcionadamente costoso con respecto al costo esperado de los cambios requeridos y la calidad del software resultante. Diversos estudios han demostrado que la mayor parte de las tareas de mantenimiento están relacionadas con mejoras requeridas por los usuarios o adaptaciones para un ambiente que cambia, más que tareas relacionadas con la corrección de errores.

El alto costo de desarrollo de software determina que los componentes de software y productos como los requerimientos, diseños y planes de prueba deben ser reutilizados siempre que sea posible. La reutilización puede hacer que el software sea desarrollado mas rápidamente, más económicamente y con mejor confiabilidad.

El *diseño orientado a objetos*, que modela el software como un conjunto de objetos, el comportamiento de dichos objetos y las relaciones entre ellos, es una de las estrategias que se usan con más éxito hoy para el diseño de programas que necesitan evolucionar continuamente. Muchas casas productoras de software están escogiendo las técnicas de orientación a objetos para el análisis, diseño y programación de sus sistemas nuevos.

Estos cambios se hacen con la esperanza de producir software de mayor calidad basados en que los métodos de orientación a objetos proporcionan una mejor abstracción de los datos y mejor "ocultamiento de la información", permiten concurrencia más directamente y responden mejor a los cambios del mundo real modelado por el software.

Muchos otros sistemas han tolerado modificaciones mayores y menores a lo largo de los años, pero su documentación no ha sido actualizada de manera paralela.

5.13. EJEMPLO: CONSTRUCCION E IMPLANTACION DEL SISTEMA SISEFIN

CONSTRUCCION DE LA INFRAESTRUCTURA

Para el desarrollo del sistema SISEFIN, el IMP dispone de la infraestructura de cómputo y comunicaciones necesaria para el desarrollo de sistemas Cliente /Servidor. El marco de la arquitectura C/S se construyó en base a los estándares institucionales, es decir, adecuado a las plataformas de desarrollo y operación de PEMEX. Esta situación procede de la vinculación Operativa - Administrativa entre el IMP y Petróleos Mexicanos.

La construcción de la infraestructura se aplico cuando se realizó la implantación del sistema SISEFIN en las regiones (Poza Rica, Villahermosa y Cd. del Carmen) de PEMEX. Teniendo la ventaja de que en la mayor parte de las dependencias que están en las regiones cuentan con una red operando y el DBMS instalado y en algunos casos funcionando para otras aplicaciones. Haciendo solo labor de verificación de la plataforma de cómputo y la instalación del sistema.

CONSTRUCCION DEL SOFTWARE

La construcción de la base de datos del sistema SISEFIN se hizo mediante la ejecución satisfactoria del script en el RDBMS, en este caso Oracle. Para el desarrollo de la aplicación se consideraron los principios de la Ingeniería de Software y en base a un plan de avance (Figura 5.1) se llevo el control de tiempos y costos de por cada módulo del sistema. Se realizaron visitas programadas con los usuarios para la revisión de procesos terminados, en algunos casos se tuvo la necesidad de realizar ciertos cambios en la programación y hasta en la estructura de la base de datos, modificaciones que no repercutieron considerablemente en el plan de trabajo.

INSTITUTO MEXICANO DEL PETROLEO
SUBDIRECCION GENERAL DE INVESTIGACION Y DESARROLLO TECNOLOGICO EN EXPLORACION Y PRODUCCION

PROYECTO **CCC-0237 " Sistema de Seguimiento Financiero de la Operación de Inspección y Mantenimiento."**

SUBDIRECCION	ACTIVIDAD	RESPONSABLE PEMEX:		FECHA INICIO					FECHA FIN					
		NOMBRE	CARGO	1	2	3	4	5	6	7	8	9	10	
1	Instalación de la Base de Datos													
	Programa y desarrollo del diagrama Entidad-Relación													
2	Elaboración del flujo de datos													
	Definición del flujo de información													
3	Modelado de Entidades													
	Programación para lenguaje PL/SQL													
4	Control Interno													
	Desarrollo													
5	Caja													
	Comprobantes													
6	Comprobantes													
	Comprobantes para inicio del año presupuestal													
7	Comprobantes													
	Comprobantes para													
8	El nuevo sistema													
	Pruebas finales													
9	Instalación de la OMS en base													
	Definición de la documentación													
10	Manual de Usuario													
	Manual Técnico													

PROGRAMADO NOMBRE FECHA INICIO
 REAL CARGO FECHA FIN
 PRESUPUESTO MILES % NOMBRE FECHA INICIO
 \$ PRESUPUESTO MILES NOMBRE FECHA INICIO

Figura (5.1) Plan de Avance de Construcción del Sistema

Al final de la construcción del sistema SISEFIN se actualizó la documentación correspondiente conforme a los cambios surgidos en las pruebas hechas por el usuario y se elaboraron los discos de instalación con su respectiva guía.

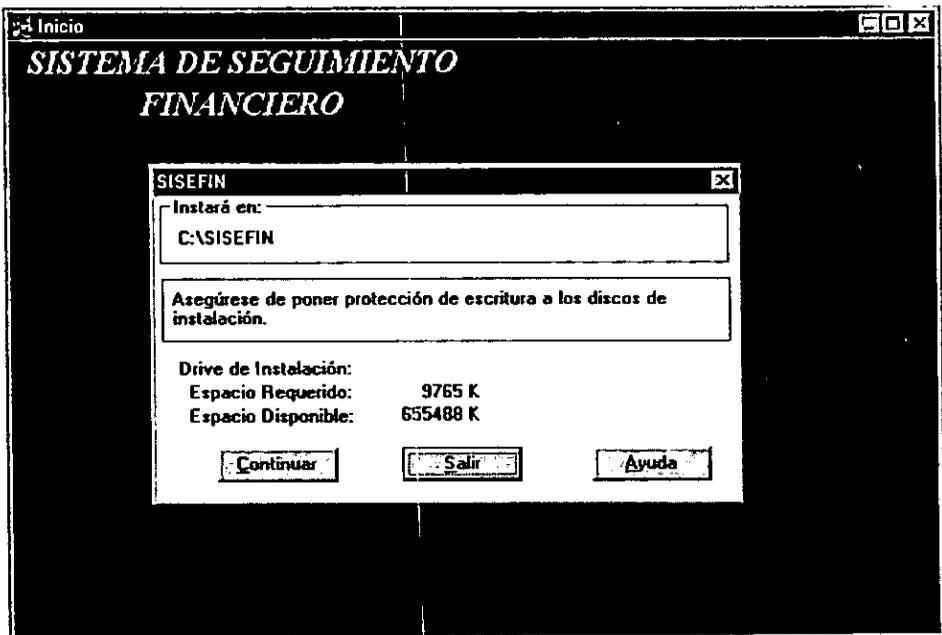


Figura (5.2) Pantalla del Instalador del sistema SISEFIN

PRUEBAS DEL SOFTWARE

Existen dos clases de pruebas que se aplicaron al sistema SISEFIN, las pruebas del módulo (procedimiento) y pruebas de integridad de información; las primeras tienen como objetivo detectar errores de lógica en la programación, permitiendo módulos más confiables; en la segunda instancia, son pruebas encaminadas a evaluar si el sistema cumple con las salidas correctas y realiza sus funciones adecuadamente, como base para las pruebas de aceptación de los productos del sistema por parte del usuario.

Los datos de prueba fueron proporcionados por documentación del usuario, otros fueron inventados por los programadores para probar todas las combinaciones o casos del sistema.

DOCUMENTACION

Se desarrollaron los manuales técnico y de usuario, y guías de instalación y procedimientos de operación y control, cada documento fue elaborado y revisado por un analista y el Líder de Proyecto, y posteriormente por el Coordinador de Proyecto del IMP y por PEMEX.

IMPLANTACION DEL SISTEMA SISEFIN

PROGRAMA DE ACTIVIDADES PROPUESTO POR PERSONAL DEL IMP PARA LA IMPLANTACION DEL SISTEMA DE SEGUIMIENTO FINANCIERO "SISEFIN"

1.- Presentación del sistema y programa de implantación al grupo de trabajo PEMEX-IMP.

- a) Presentación del Sistema.
- b) Definición de responsables y sus funciones.
- c) Presentación del programa de implantación.
- d) Identificación de la infraestructura de cómputo.

2.- Elaboración del programa de visitas a regiones

- a) El personal del IMP formará tres grupos de trabajo distribuidos de la siguiente manera:

GRUPO	REGION	SUBGERENCIA
1	NORTE	DUCTOS INSTALACIONES
2	SUR	DUCTOS INSTALACIONES

GRUPO	REGION	SUBGERENCIA
3	MARINA SURESTE	DUCTOS INSTALACIONES
	MARINA NORESTE	CANTAREL EKBALAM

b) Determinación de fechas para la implantación del sistema.

3.- Integración y formación de Grupos de Trabajo PEMEX-IMP.

4.- Instalación del Sistema.

- a) Identificación y verificación de Hardware, Software y Humanware de cada uno de los lugares de Instalación.
- b) Instalación en el server.
- b.1) Verificación y/o creación de espacio en disco duro y memoria RAM.
 - b.2) Creación de la estructura del sistema en el RDBMS ORACLE.
 - b.3) Creación de usuarios y privilegios del sistema.
 - b.4) Verificación de los canales de comunicación.
 - b.5) Pruebas de conectividad local y remota.
- c) Instalación en el cliente.
- c.1) Verificación y/o creación de espacio en disco duro y memoria RAM.
 - c.2) Generación del ambiente de operación del sistema en el front end SUPERBASE.
 - c.3) Actualización del archivo HOSTS.
 - c.4) Verificación y/o configuración del driver SQL NET TCP/IP.
 - c.5) Verificación y/o configuración del software de red (SYNTAX, LANWORK PLACE, WINDOWS FOR WORK GROUPS).
 - c.6) Carga de la aplicación.
 - c.7) Verificación de enlace en la aplicación CLIENTE/SERVIDOR.
 - c.8) Pruebas de conectividad locales y remotas.
- d) Pruebas locales e integrales del sistema.

5.- Capacitación en la operación del sistema.

- a) Programa de Capacitación.
- b) Estrategia. (Concentrar a los usuarios en su región correspondiente).
- c) Captura de información.

- d) Consulta.
- e) Generación de reportes.
- f) Respaldos de información.
- g) Cortes de información.
- h) Instalación del sistema.

6.- Monitoreo del sistema.

- a) Captura de información real.
- b) Consulta de información.
- c) Generación de reportes.

7.- Apoyo y supervisión del sistema.

- a) Visitas programadas a cada región.
- b) Asesoría en la operación del sistema.
- c) Asesoría en la transferencia de información.

8.-Liberación.

- a) Entrega de documentación final (usuario y técnico).
- b) Entrega de discos, guías, etc.
- c) Entrega oficial del sistema al coordinador del proyecto por parte de PEMEX.
- d) Negociación del mantenimiento evolutivo del sistema.

**ACTIVIDADES A REALIZAR PARA EL PROYECTO CCD-0237
IMPLANTACIÓN DEL SISTEMA DE SEGUIMIENTO FINANCIERO "SISEFIN"**

La implantación del sistema se realizará en dos fases, las cuales se describen a continuación:

FASE 1) Instalación del sistema en regiones:

Preparación del ambiente en el servidor para la carga del sistema. Esta actividad la debe realizar el personal de PEMEX encargado de la administración de la red (ver anexo de requerimientos informáticos)

Carga del sistema en las PC's de la red local.
Pruebas de comunicación local entre las PC's y el servidor.
Prueba del sistema instalado a nivel de la red local.

FASE 2) Capacitación a usuarios en la operación del sistema:

Captura de información.

Consulta de datos del sistema

Generación de Reportes.

Contexto general de la metodología para determinar el Control Interno Presupuestal.

ESTRATEGIA DE IMPLANTACIÓN

1. Para el cumplimiento de la implantación del sistema, el personal del IMP visitará cada una de las regiones (NORTE, SUR, MARINA SURESTE, MARINA NOROESTE).
2. Durante su estancia en cada región, el personal del IMP se distribuirá de la siguiente manera:

Una persona en la cabecera de la región que se encargará de monitorear la comunicación y la operatividad del sistema.

Dos grupos de trabajo formados por dos personas que se encargarán de instalar el sistema en cada región.

3. Al menos una computadora personal configurada de acuerdo al diagrama que se anexa al documento, así como el software necesario y conectada al servidor.
4. En cuanto al equipo requerido para la capacitación es necesario contar como mínimo con una computadora conectada en red y un datashow con el software necesario, conectadas al servidor y con el sistema instalado.

PLAN DE CAPACITACIÓN

1. Para la operación del sistema, se propone concentrar al personal de la región con su correspondiente cabecera para que el IMP imparta la capacitación en el manejo del sistema.
2. Una vez concluida la capacitación, el personal de PEMEX procederá a capturar la información correspondiente al Sistema (Subproyectos, Cap's, Compromisos y Documentos, etc.), con el fin de que estos cuenten con información real y actualizada de la región para acordar los siguientes compromisos.

COMPROMISOS

1. Cuando se cuente con información capturada en las Regiones se procederá a realizar una prueba integral y obtención de reportes ejecutivos a nivel Región.

2. Cada una de las actividades señaladas en este documento se realizarán en las regiones: NORTE, SUR, MARINA SUROESTE Y MARINA NORESTE. El tiempo estimado para cada región es la siguiente:

REGIÓN SUR : __ días hábiles

REGIÓN NORTE : __ días hábiles

MARINA SURESTE : __ días hábiles

MARINA NORESTE: __ días hábiles

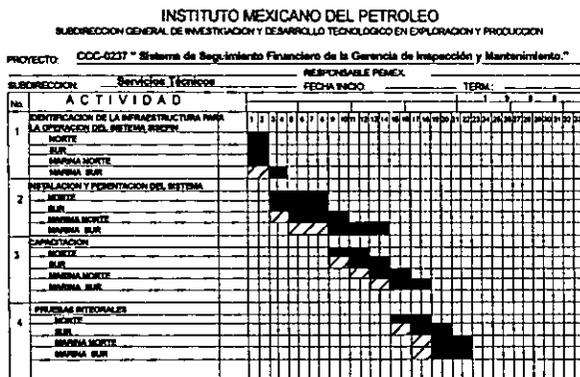


Figura (5.3) Programa de tiempos para Implantación del sistema SISEFIN

ATENTAMENTE

DE ACUERDO

NOTAS:

- 1 El no contar con los elementos antes mencionados **provocará un defasamiento en el programa de Trabajo para la Implantación.**
- 2 Para el buen desempeño de las actividades de implantación del sistema, se solicita el apoyo de PEMEX para el transporte de personal del IMP para desplazarse entre las diferentes Regiones.
- 3 Se distribuirá al personal del IMP en todas las regiones para lograr alcanzar los objetivos que se tienen planeados en el menor tiempo posible.

GUIA DE INSTALACION DEL SISTEMA DE SEGUIMIENTO FINANCIERO "SISEFIN"

I) IDENTIFICACION DE LA ESTRUCTURA DE COMPUTO

EN EL CLIENTE.

1. Sistema Operativo

Versión Es necesario tener la versión 5 o superior

Número de licencias

2. Espacio libre en disco duro

Espacio mínimo requerido para la instalación de el sistema:

Aplicación	15 MB
SUPERBASE	6 MB
EXCEL	20 MB

3. Memoria RAM

La memoria RAM de la máquina debe ser como mínimo de 8 MB, para que pueda existir una buena respuesta del sistema

4. Procesador

Es recomendable que el procesador sea de 386 o superior

5. Tipo de monitor

Para que las pantallas del sistema no se distorsionen es recomendable tener un monitor VGA.

6. Middleware: SQL NET TCP/IP de ORACLE

7. Software de red (LANWORK PLACE, SYNTAX, WINDOWS for groups, WINDOWS NT, LAN MANAGER)

8. Archivos de control (host, services, config.sys, autoexec.bat, win.ini, oracle.ini, config.ora)

9. Software – Front end instalado :

Windows 3.1

Superbase 2.0

Excel 4

EN EL SERVER.

1. *Sistema Operativo* (UNIX).
2. *Espacio en disco* (df).
3. *Archivos de control* (/etc>services,/etc>oratab,/etc>hosts)
4. *Manejador de Base de Datos* (RDBMS Oracle7).
5. *Middleware*: SQL NET TCP/IP.
6. *Usuarios de la base de datos* PRESST.
7. *Rutas de comunicación*

II) *INSTALACION DEL SISTEMA*

EN EL CLIENTE

1. Si no esta instalado Software de red.
Instalar LANWORK PLACE de acuerdo a los sig. pasos:
2. Si no esta instalado SQL-NET.
Instalar de la siguiente manera:
3. Verificar que haya conexión.
4. Si no esta instalado SUPERBASE.
Instalar de la siguiente manera:
5. Si no esta instalado EXCEL.
Instalar de la siguiente manera:
6. Ejecutar el instalador.

NOTA: En caso de ser necesario configurar los archivos de control.

EN EL SERVIDOR

1. Si no existe el tablespace SIGIM
Crear el Tablespace.

2. Si no existe el usuario PRESST
Correr el script de PRESST
3. Si no existe el usuario SISEFIN y el usuario SISEFIN1
Crear los usuarios con sus respectivos privilegios
4. Ejecutar el script del sistema
5. Pruebas locales y remotas

RESULTADOS DE LA IMPLANTACION

Al final de la implantación se redactó un informe con los resultados obtenidos en la instalación, capacitación, pruebas y monitoreo del sistema en cada una de las dependencias. También se anexaron las propuestas y acuerdos por parte de los usuarios hacia el sistema, información valiosa para el mantenimiento del sistema.

En general la implantación fue satisfactoria a excepción de que en la región Marina Noreste se pospuso la implantación del sistema por motivos de movilización de equipo y personal.

LIBERACION

La liberación del sistema se llevo a cabo en la instalaciones de PEMEX en Villahermosa. Tabasco. Estando presente el encargado del grupo de transición de la GIM, el jefe del departamento de Sistemas del IMP, el Coordinador del Proyecto por parte de PEMEX e IMP, el líder de Proyecto y tres testigos (usuarios potenciales del sistema) para la entrega oficial del sistema SISEFIN. El documento firmado fue el siguiente:

PETROLEOS MEXICANOS
SUBDIRECCION DE SERVICIOS TECNICOS
GERENCIA DE INSPECCION Y MANTENIMIENTO

OFICINAS GENERALES **VILLAHERMOSA, TABASCO**

En Villahermosa, Tabasco, siendo las ____ del día __ de ____ de 199_, se reunieron en las oficinas de la Gerencia de Inspección y Mantenimiento PEMEX Exploración y Producción, el C. Ing. _____ ficha 00001, Encargado Grupo de Transición de la GIM , con domicilio Boulevard Ruiz Cortinez No. 1202 6o piso Villahermosa Tabasco; el Mat. _____ clave 0008, Jefe de la Línea de Negocios en Sistemas Expertos

IMP, el Ing _____, clave 00007, Jefe de División de Sistemas de Información y Coordinador el proyecto, y la Lic. _____, clave 00004, Líder del Proyecto, los tres últimos con domicilio en el Eje Central Lázaro Cárdenas No. 152 colonia San Bartolo Atepehuacan en México D.F.

I.- SITUACION PROGRAMATICA

Uno de los objetivos principales de la Gerencia de Inspección y Mantenimiento es lograr satisfacer de manera confiable, veraz y oportuna todos sus requerimientos del Control Interno Presupuestal en materia de egresos de operación e inversión a nivel Gerencia en Región Sur.

II.- SISTEMA DE SEGUIMIENTO FINANCIERO.- PROYECTO CCD-0237

Se recibe del Instituto Mexicano del Petróleo el proyecto CCD-0237 denominado Sistema de Seguimiento Financiero para el Control Interno Presupuestal, con la siguiente documentación y acciones:

- Manual de Usuario del SISEFIN para la cuatro regiones de PEMEX. (5 juegos)
- Manual Técnico del Sistema SISEFIN.
- Programas fuentes de software del sistema SISEFIN.
- Relación de equipos de cómputo en donde el sistema se encuentra instalado.
- Documento resumen de procedimientos de respaldo .- subir y bajar información.
- Puesta en operación del sistema SISEFIN en las instalaciones de la Gerencia de Inspección y Mantenimiento.

III.- COMENTARIOS FINALES

La Gerencia de Inspección y Mantenimiento, recibe en este acto la totalidad del Sistema de Seguimiento Financiero en ambiente de red en condiciones de continuar su operación, documentación de manuales de usuario y técnico, programas fuentes, procedimientos de respaldos y funcionando el sistema en las instalaciones de la GIM; con el compromiso de que el Instituto Mexicano del Petróleo proporcionará durante dos meses, mantenimiento y asesoría de acuerdo a solicitudes que indiquen en tal efecto.

El Ing. _____ y el Mat. _____ de PEMEX Exploración-Producción e Instituto Mexicano del Petróleo respectivamente, manifiestan haber proporcionado sin omisión alguna todos los datos y elementos necesarios para la formulación de la presente acta.

Los anexos que se mencionan en este documento, forman parte integrante del mismo y se firman en todas sus fojas para su identificación y efectos legales a que haya lugar.

Previa lectura del acta y no habiendo mas que hacer constar, se dan por terminada las presentes actuaciones siendo las 18:00 hrs. del propio día, firmando para la constancia en todas sus fojas al margen y alcance los que en ella intervinieron.

RECIBE

ENTREGA

TESTIGOS

MANTENIMIENTO

Las negociaciones de mantenimiento se retomaran una vez iniciada la estabilización organizacional de PEMEX, quedando pendiente la reunión con los usuarios principales del sistema SISEFIN de las cuatro regiones para el planteamiento de las actividades de mantenimiento y el inicio de la segunda versión de la aplicación.

CONCLUSIONES

La arquitectura Cliente/Servidor se ha convertido en una norma para todo tipo de organizaciones hasta para la industria de la computación. Sus ventajas han elevado el grado de productividad y mejoramiento en el acceso y manejo de la información, ofreciendo un ambiente distribuido con sistemas abiertos e interfaces amigables para la adecuada toma de decisiones. Un sistema Cliente/Servidor reduce significativamente tiempo, esfuerzo y dinero recursos importantes para incrementar las utilidades.

El adoptar esta tecnología requiere de una especializada metodología y sobre todo una revolución (en muchos de los casos) de la forma en que se manejan actualmente los procesos dentro de una empresa, aplicando lo que se conoce como reingeniería de procesos, obteniéndose así los máximos beneficios al operar los sistemas Cliente/Servidor.

Con el sistema SISEFIN se logro integrar los componentes de computo (maquina cliente y servidor), comunicaciones y bases de datos bajo un estricto proceso de administración y control de recursos. Desarrollando el sistema bajo conceptos de métricas de software, metodologías de análisis y diseño, herramientas CASE, y programación estructurada y por eventos. Uno de los factores importantes para el éxito de un sistema C/S es que los miembros que integran el equipo de trabajo tengan el mismo nivel de conocimientos sobre este ambiente y se conjugue la participación, apoyo, comunicación y responsabilidad de cada una de las partes, incluyendo a los usuarios y coordinadores. Este aspecto no se cumplió favorablemente en el desarrollo del sistema SISEFIN, hubo cierta incompatibilidad de conocimientos en el equipo y la comunicación entre los sectores organizacionales donde se instaló el sistema fue escasa, habiendo problemas de aceptación y afectando notablemente los tiempos programados del sistema.

En general, el sistema SISEFIN abarcó la metodología expuesta en la Tesis, empezándose a vislumbrar beneficios en el control del ejercicio presupuestal para la GIM. Pero lamentablemente, la reestructuración organizacional de PEMEX obstaculizó el interés por parte de varios usuarios para operar el sistema, debido a que tenían otras preocupaciones como el asegurar su situación laboral dentro de la empresa, reubicandose en otras dependencias. Este factor externo es uno de los problemas que en ocasiones no se tiene control en detectarse oportunamente y sin embargo afecta la vida de un sistema.

Durante el desarrollo de la Tesis y del sistema SISEFIN me surgieron diferentes inquietudes sobre ciertas cuestiones cómo cuál es el futuro de la tecnología Cliente/Servidor y de las venideras, con sus enormes avances, que sin pensarlo de un momento a otro lo que creíamos novedoso se vuelve obsoleto; o por qué muchas veces los empresarios prefieren comprar aplicaciones ya hechas, en la mayoría por extranjeros, en vez de desarrollarlas nosotros mismos; y con toda esta evolución y crecimiento acelerado de la computación cuál es el verdadero concepto de la informática. Para lo cuál concluyó lo siguiente según mi criterio y lo que he investigado:

EL FUTURO DE LA ARQUITECTURA CLIENTE/SERVIDOR

Hoy en día se ofrecen mejoras para el ambiente Cliente/Servidor y directamente hacia los sistemas que han sido construidos bajo esta plataforma. Asegurando una permanencia como infraestructura de cómputo.

MEJORAS SOBRE EL CLIENTE

La robustez del cliente es un punto clave para la operación de sistemas C/S y continuará mejorando su capacidad y seguridad. Surgiendo cambios en su mismo hardware, sistemas operativos y aplicaciones.

Por ejemplo, en el hardware del cliente la arquitectura será cada vez más amigable con el advenimiento de tecnologías como Plug and Play (Conecta y usa) y la integración de audio, fax, módem, y correo de voz. Las características básicas como procesador, GUI y memoria serán más poderosas. Igualmente se incrementará las capacidades para la integración de software de red.

Con el gusto de Windows '95, Windows NT, UNIX y OS/2, los sistemas operativos de los clientes pueden llegar a ser cada vez más seguros. Estos han tomado ventajas tecnológicas como multitareas, seguridad y comunicaciones, normalmente encontradas en sistemas operativos de gran escala. Y los sistemas operativos ofrecerán máxima compatibilidad con los anteriores drivers, aplicaciones y viejo equipo. La evolución de los sistemas operativos seguirán siempre obteniendo beneficios al ambiente C/S.

También se empezará a ver la completa integración de soporte de Internet e Intranet por los sistemas operativos. La navegación y consulta de Internet llegará a ser más fácil y rápida. Y la activación y reconocimiento de voz permitirá tener un mayor control de la computadora en vez del tradicional teclado y ratón (mouse).

El futuro de las herramientas de desarrollo para C/S es un tanto incierto. Pero algo si es seguro, la programación de ahora no es la misma a la que fue en la década pasada. Las herramientas de desarrollo se están moviendo a la orientación por objetos, accediendo a base de datos relacionales y a base de datos por objetos. Esta clase de acceso permitirá construir bases de información sobre objetos que incluyen datos tipo texto, numérico, imagen y sonido. Por lo que cada vez se mejora la independencia de la base de datos por medio de la división de la aplicación y presentación en el cliente y la base de datos en el servidor.

El tiempo de desarrollo será invertido en dirigir y unir proyectos RAD (Rapid Application Development – Desarrollo de rápidas aplicaciones), rápidamente ensamblando objetos e interactuando con los usuarios.

MEJORAS AL SERVIDOR

La base de datos comercial se moverá a la base por objetos. Los sistemas de información correrán sobre una plataforma distribuida basada en manejadores de base de datos orientada a objetos.(ODBMSs). Las organizaciones requieren de diferentes tipos de datos, eficaz rendimiento e integración de esos tipos de datos para la estandarización de acuerdo a las herramientas de programación orientada a objetos. Los sistemas basados en objetos están empezando a aparecer, sosteniendo complejos tipos de datos y transacciones, particularmente a través de diferentes plataformas y ambientes.

MEJORAS SOBRE LA RED

La necesidad de disponer de más ancho de banda es esencial para aplicaciones con multimedia. Diversas compañías han empezado a crear redes para atender estas necesidades presentando grandes tecnologías de transmisión como Fast Ethernet como estándar para las LANs y ATM (Asynchtonous Transfer Mode - Modo de Transferencia Asincrona) para las WANs.

Las redes llegarán a ser tan grandes, que cada vez más usuarios serán conectados desde lugares remotos. Además se incrementará el número de usuarios con computadoras móviles que harán uso de redes inalámbricas. Con el modelo ISDN (Integrated Services Digital Network – Red Digital de Servicios Integrados) representa la red del futuro que incluye la integración de voz, datos y vídeo viajando desde un tipo de red a otra, apropiado para las videoconferencias.

El futuro de la arquitectura cliente/servidor es emocionante y cambiante siempre con el objetivo de evolucionar y mejorar. El modelo Cliente Servidor puede ayudar a cualquier organización o negocio a lograr sus objetivos mediante el avance de la tecnología.

BIENVENIDO INTERNET, INTRANET Y EXTRANET

Actualmente el apogeo de la red más grande del mundo, Internet, esta haciendo presencia en diferentes sectores tanto financieros, educativos, comerciales, industriales, gubernamentales, etc, y más aún disponible para cualquier tipo de persona.

Internet es la interconexión de miles de redes (a través de líneas telefónicas, fibras ópticas y satélites) ubicadas en todo el planeta, que se encuentran interconectadas mediante el protocolo TCP/IP, y que juntas ponen a disposición del público una enorme cantidad de información y servicios.

En Internet se puede encontrar todo tipo de informaciones, contenidos y presentaciones en pantalla (en la mayoría con características de multimedia) mediante el sistema conocido como World Wide Web (WWW), que son diversos sitios web (servidores Web permanentes localizados en diferentes partes del planeta) en donde existen páginas web (home page) con formato HTML (HyperText Markup Language) para ser consultadas por todo el mundo.

La infraestructura es simple, se requiere de una computadora, módem, línea telefónica, software para navegar por la Web (navegadores –browsers- como Netscape, Explorer, gopher,...) y para enviar y recibir correo electrónico, y el servicio de conexión por parte de una empresa para obtener una dirección en Internet.

Actualmente, el servicio más usado de Internet es el correo electrónico (e-mail), un medio seguro, económico y confiable para intercambiar mensajes a cualquier persona de cualquier parte del mundo. Existen también otros servicios atractivos como entablar conversaciones, adquirir productos o servicios de empresas, videoconferencias, y mucho más.

La red Internet ha sido aprovechada por organizaciones para integrar su información y poder accederla desde cualquier lugar y ser presentada al usuario final utilizando solo una interfaz amigable, bajo este principio nace las Intranets y Extranets. Ambas se valen de los estándares empleados en Internet y el WWW. Sin embargo, una **Intranet** es la que no permite el acceso a ningún usuario externo. Y por el contrario, una **Extranet** cuenta con salida al exterior para los empleados y permite que ciertos usuarios tengan acceso a la información, utilizando Internet. Los grandes comerciantes impulsan esta tecnología porque se ha convertido en un escaparate de alcance mundial para los millones de clientes potenciales, usuarios de Internet, los que sin moverse de sus casas u oficinas conoce, buscan y solicitan productos y servicios muy diversos.

La forma de restringir el acceso y el nivel de interacción con la información es realizado a través de sistemas de software o hardware llamados **Firewalls**.

Las Intranets y Extranets no reemplazan o promueven la desaparición de la arquitectura Cliente/Servidor, por el contrario, es la evolución del mismo, presentando y trabajando bajo el mismo concepto, permitir que cada computadora “haga lo que hace mejor”. El futuro de esta tecnología es muy prospero, marcando la forma de trabajar del siguiente milenio.

EL USO DE LA TECNOLOGIA

En ocasiones las Tecnologías de Información (TI) se utilizan para mecanizar los procesos de control actuales (por ejemplo, control de inventarios, control de costos, nómina, etc.), lo cual si bien puede hacer más eficientes estos procesos, no implica que la empresa se vuelva más competitiva. Es más en ocasiones los mecanismos de estos procesos hace que la empresa se vuelve más rígida y menos innovadora, lo cual, a la larga, puede perjudicar su competitividad.

El buen uso de la TI estará orientada a modificar las actividades; no solamente a mecanizar el registro de transacciones. Así, las TI pueden utilizarse, entre otras cosas, para apoyar la toma de decisiones, para obtener nueva información y realizar simulaciones, para transformar el contenido físico de una actividad o bien, para modificar el alcance competitivo.

La TI es una herramienta que nos permite romper con reglas obsoletas como se hacen negocios y para aprovecharla al máximo es necesario hacer un análisis sobre la forma en que se trabaja actualmente y determinar si se requiere de aplicar la reingeniería de procesos que es el replantamiento fundamental y un rediseño radical de los procesos del negocio para obtener mejoras en indicadores críticos de productividad como costo, calidad, servicio y tiempo.

Algunas organizaciones han entendido la importancia que tienen las TI y han obtenido beneficios importantes de ellas. En nuestro país, las TI juegan un papel fundamental en el proceso de modernización en la cual estamos inmersos. Por ello es necesario hacer un buen uso de las TI, aprovechando las múltiples oportunidades que ofrecen para mejorar la manera en la que operan en las organizaciones, para crear nuevos negocios o para reinventar los procesos de trabajo.

DESARROLLO DE NUESTRAS PROPIAS APLICACIONES

Muchas empresas mexicanas gastan millones de pesos comprando productos extranjeros, teniendo tal vez, una mejor garantía de contar con sistemas que realmente incrementen la producción. La idea es aceptable. Pero la preocupación de muchos de los egresados de la carrera es que sentimos que los empresarios mexicanos conservan una mala imagen de nuestras capacidades como desarrolladores.

Sabemos que no todos estamos preparados para realizar grandes productos, pero hay muchos desarrolladores capaces de resolver desde problemas triviales hasta los más avanzados. Por lo que el problema no es de incapacidad intelectual sino de incapacidad de trabajar en equipo, colaborando unos con otros.

El fomentar el espíritu de grupo es la mejor manera de resolver los problemas. Los grupos de trabajo deben ser motivados, y encauzados por intereses comunes en un tema, aunque se pertenezca a agrupaciones u organizaciones diferentes. Igualmente se debe valorar los productos desarrollados de manera individual que busquen el beneficio de la sociedad.

Para alcanzar un nivel de confianza adecuado que nos permita vernos como buenos creadores, se podría organizar concursos de software más formativos, a nivel estatal o nacional en las cuales se vaya ganando experiencia y madurando las capacidades creadoras, e ir adquiriendo la suficiente confianza y preparación para penetrar los productos al mercado.

Si los sectores empresariales, instituciones académicas y organismos públicos, mediante un mecanismo, manifestaran sus necesidades de automatización no resueltas dando oportunidad de ser solucionadas por grupos de trabajo nacionales, se fomentaría una verdadera industria informática mexicana representativa y reconocida, fortaleciéndose la economía del país.

REUTILIZACION DE COMPONENTES DE SOFTWARE

La verdadera importancia y éxito de los objetos es la reutilización de componentes de software. Construir un sistema será ensamblar componentes de software ya hechos y probados. Existen cientos de miles de objetos (de clases de objetos) que pueden utilizarse y ensamblarse para construir soluciones computacionales más rápidamente y más fáciles de mantener, y por lo mismo más adaptables a los cambios que sufren las organizaciones.

Por lo que es importante formar gente en la teoría de objetos con visión de integradores y no con la visión del programador tradicional que con cualquier pretexto programa sus clases, incluso las clases ya hechas y probadas, y lo hace por inercia, por educación, por desconfianza o desconocimiento. La gente integradora podrá construir aplicaciones de valor estratégico (productividad) para las organizaciones reutilizando componentes de software.

Por otro lado, se debe entender que las bibliotecas existentes no pueden cubrir todo y entonces será necesario, ahora sí, ponerse a programar las clases faltantes, pero para eso estarán los ingenieros en objetos, los especialistas que diseñarán objetos (clases) particulares y específicos que harán falta.

Hay que aprovechar las ventajas que nos ofrece la orientación a objetos, que curiosamente no es algo nuevo, hay que empezar a usar las mejores herramientas y ambientes que existen, y avanzar más rápidamente de lo que podríamos avanzar con mentalidad convencional.

DEFINICION DE INFORMATICA

Para la definición de informática se proponen distintas interpretaciones. Primero, el de aquellos que quieren ver a la informática reconocida como una disciplina científica, con su propia metodología y sus reglas de validación. Esta óptica, implícitamente conlleva la opción por un juego cerrado donde los actores que no cumplen ciertas normas metodológicas o principios teóricos quedan fuera.

En segundo lugar están quienes ven a la informática como un gran mercado en expansión donde pueden volcar sus intereses emprendedores ofreciendo una amplia gama de productos y servicios. Su punto de vista asume que son participantes de un juego abierto con todos los ingredientes de la competencia.

Como tercero, aparece la visión ingenieril que asocia la informática con la integración, propia de los especialistas preocupados por generar soluciones aquí y ahora. Hoy por hoy, es el eslabón más dinámico y que viene marcando la pauta y el ritmo de avance desde hace varios años, dando prioridad a las tecnologías con las que están comprometidos.

Otros ven a la informática como espacio muy relevante para la economía, la vida social, política y cultural de nuestra sociedad y se dan a la tarea de velar por los intereses públicos y de la nación.

Cada definición va de acuerdo al ejercicio de su práctica, es decir, a campos de acción bien determinados. Diversos intereses están presentes tras las distintas interpretaciones de la informática, por lo que se debe de clarificar en qué foro o entendimiento de la informática estamos ubicados.

Desde mi punto de vista como desarrolladora de sistemas de información, la informática deriva hacia una disciplina que atiende a problemas de los hombres en el marco de restricciones que impone la tecnología disponible. La informática es aquella disciplina que se hace cargo de resolver los problemas de diseño, implantación, explotación y mantenimiento de los sistemas de información que registren y distribuyan el conjunto de las distintas actividades humanas, así como el capacitar a los usuarios para saber trabajar y crear valor agregado en este nuevo mundo simbólico, donde la coordinación hace el producto, al que ha dado origen lo que hoy conocemos como el desarrollo informático y de telecomunicaciones. La función de la Informática es que debe ser capaz de responder a las necesidades dinámicas del hombre en forma rápida, flexible y con bajo costo.

LA INFORMATICA COMO HERRAMIENTA COMPETITIVA

La informática es clave para el futuro de las empresas, por lo que es necesario transformar a la información es una herramienta competitiva. Para ello, las empresas deben comprometer a sus áreas operacionales con la innovación informática.

En el **área operativa** tiene que estar comprometida, a poner los nuevos sistemas de trabajo y a lograr los beneficios anunciados. Sin este compromiso no es posible que los sistemas informáticos por sí solos produzcan beneficios.

El **grupo técnico** interno tiene que estar comprometido, a soportar los sistemas, tanto desde el punto de vista de mantener su operación fluida, como de entrenar y ayudar a sus usuarios. Esto es independiente de que el software sea adquirido a terceros o desarrollado internamente.

La **alta gerencia** debe de estar comprometida a proveer los recursos para realizar adecuadamente el proyecto. A cambio, debe pedir a la respectiva área operativa compromisos de beneficios medibles para el negocio que se obtendrán con los nuevos sistemas.

La introducción de la tecnología a partir de esta visión se estará modernizando la organización y no sólo automatizando día a día. Los esfuerzos conjuntos de la alta gerencia y los equipos informáticos podrán observar hacia donde se mueve el negocio y luego emplear en conjunto todas las herramientas que ofrece el mercado informático para dar los saltos que hagan de la compra de tecnología una compra de oportunidades y no una compra de problemas.

BIBLIOGRAFIA

LIBROS

- Senn, James.- Análisis y Diseño de Sistemas de Información.- Traduc. Urbina Medal Edmundo Gerardo y otros.- 2ª Edición.- Editorial McGRAW-HILL.- México.- 1994.
- Kendall, Kenneth y Kendall, Julie.- Análisis y Diseño de Sistemas.- Traduc. Hernández López Héctor y otros.- 1ª Edición.- Editorial Prentice-Hall Hispanoamericana.- México.- 1994.
- Gonzáles, Cristobal del Río.- El Presupuesto.- 2ª Edición .- 8ª reimp.- Editorial ECASA.- México.- 1991.
- Loney, Kevin.- ORACLE Manual del Administrador.- Traduc. Vallejo Pinto José Angel.- 1ª Edición.- Editorial McGRAW-HILL.- México.- 1996.
- Vyles, Black.- Redes de área local, Protocolos, Normas e Interfaces.- 1ª Edición.- Editorial Macrobite.- México.- 1990.
- Jenkins, Neil y otros.- Client/Server Unleashed.- 1ª Edición.- Editorial Sams.- E.U.A.- 1996.
- Enterprise Series: LAN Applications
- Hammer, Michael y Champy, James.- Reingeniería.- Traduc. Nannetti Cárdenaz José.- Editorial Norma.- Colombia.- 1994.
- Morris, Daniel y Brandon, Joel.- Reingeniería.- Traduc. Avila de Barón Cecilia.- Colombia.- 1994.
- Pressman, Roger.- Ingeniería de Software, Un enfoque práctico.- Traduc. Troya José María y Hernández Yanéz Luis.- 2ª Edición.- Editorial McGRAW-HILL.- México.- 1990.

TESIS

- Molina Felix, Luis Carlos.- Análisis del Diseño e Instalación de la Red de Area Local para Computadoras en el Instituto Mexicano del Petróleo.- Tesis profesional para optar por el grado de Ingeniero en Comunicaciones y Electrónica.- Escuela Superior de Ingeniería Mecánica y Eléctrica.- IPN.- México.- 1993.
- Basto García, Carlos Humberto.- Diseño de una Arquitectura Cliente / Servidor para el departamento de Sistemas de Información del I.M.P..- Tesis profesional para optar por el grado de Ingeniero en Comunicaciones y Electrónica.- Escuela Superior de Ingeniería Mecánica y Eléctrica.- IPN.- México.- 1995.

MEMORIAS

- El liderazgo en la computación como eje de la Competitividad Internacional.- Memorias del Simposium Internacional de Computación, celebrado en el CENAC. – IPN. - Octubre de 1996.

REVISTAS

PERSONAL COMPUTING

- Oropesa Talavera, Enrique.- Los recursos humanos para redes y el outsourcing.- Revista Personal Computing.- Sayrols, S.A de C.V.- No. 88.- Septiembre de 1995.- México.- pp. 58 a 59.
- Entrevista a Kahn, Philippe.- El upsizing va para arriba.- Entrevistado por Martínez Paulina.- Revista Personal Computing.- Sayrols, S.A de C.V.- No. 72.- Mayo de 1994.- México.- pp. 58 a 60.
- Oropesa Talavera, Enrique.- El sistema de Cableado Estructurado.- Revista Personal Computing.- Sayrols, S.A de C.V.- No. 72.- Mayo de 1994.- México.- pp. 76 a 78.

RED

- El ABC de las Redes Locales.- Revista Red.- Edición Especial.- Editorial Novelco.- México.- 1991.
- Guerrero Chavez, Gustavo.- Reingeniería: ideas para que su empresa se coloque a la cabeza del mercado.- Revista Red.- Red, S.A de C.V.- No. 39.- 1994.- México.- pp. 2 a 6.
- Mayo Guzmán, Laura.- Topologías de red, tendencias en la selección de redes.- Revista Red.- Red, S.A de C.V.- No. 39.- 1994.- México.- pp. 20 a 25.
- Villarreal, Francisco.- Windows NT: Deseos y realidades.- Revista Red.- Red, S.A de C.V.- No. 39.- 1994.- México.- pp. 36 a 43.
- George, Juan Carlos.- ¿Cómo funciona Token Ring?.- Revista Red.- Red, S.A de C.V.- No. 32.- 1993.- México.- pp. 12 a 15.
- Esquivel Sánchez, Jesús.- Las Redes UNIX.- Revista Red.- Red, S.A de C.V.- No. 32.- 1993.- México.- pp. 34 a 35.
- De Vega, Tony.- Evolución para interoperar: ruteo de comunicación múltiple.- Revista Red.- Red, S.A de C.V.- No. 33.- 1993.- México.- pp. 10 a 16 .
- Aguilar, Sergio.- El mantenimiento de redes: la mejor protección de su inversión.- Revista Red.- Red, S.A de C.V.- No. 33.- 1993.- México.- pp. 17 a 18.
- De Vega, Tony.- Un nuevo rol de UNIX en redes corporativas.- Revista Red.- Red, S.A de C.V.- No. 33.- 1993.- México.- pp. 34 a 40.

SOLUCIONES AVANZADAS

- Cárdenas García, Sergio R.- Una Panorámica de la Ingeniería de Software.- Revista Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios.- Xview, S.A de C.V.- No. 8.- Marzo a Abril de 1994.- México.- pp. 4 a 6.
- Entrevista a Ghezzi Carlos y Pezze Mauro.- Opiniones sobre Ingeniería de Software.- Entrevistados por Vizcaino Sahagún Carlos.- Revista Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios.- Xview, S.A de C.V.- No. 8.- Marzo a Abril de 1994.- México.- pp. 34 a 35.
- Menchaca, Rolando.- Hacia las Redes de Transmisión de Datos de Alta Velocidad.- Revista Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios.- Xview, S.A de C.V.- No. 8.- Marzo a Abril de 1994.- México.- pp. 36 a 40.

- Pérez Almanza, Enrique- Ingeniería de Procesos.- Revista Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios.- Xview, S.A de C.V.- No. 21.- 15 Mayo de 1995.- México.- pp. 30 a 33.
- Lemaître, Christian- Informática Avanzada y Reingeniería de Procesos.- Revista Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios.- Xview, S.A de C.V.- No. 21.- 15 Mayo de 1995.- México.- pp. 34 a 35.
- Cota Aguilar, Manuel- Ingeniería de Software.- Revista Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios.- Xview, S.A de C.V.- No. 11.- Julio de 1994.- México.- pp. 5 a 13.
- Palomino Haddas, Carlos- Base de Datos para PC: Análisis Comparativo.- Revista Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios.- Xview, S.A de C.V.- No. 9.- Mayo de 1994.- México.- pp. 34 a 37.
- Del Cueto y Legaspi, Ricardo- Windows NT.- Revista Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios.- Xview, S.A de C.V.- No. 14.- Octubre de 1994.- México.- pp. 4 a 8.
- Coyote Estrada, Hugo César- Sistemas Operativos. Una Breve Introducción.- Revista Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios.- Xview, S.A de C.V.- No. 14.- Octubre de 1994.- México.- pp. 43 a 45.
- Tavera Parra, J. Fernando.- Perspectivas de Frame Relay y ATM en Redes Públicas.- Revista Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios.- Xview, S.A de C.V.- No. 27.- 15 de Noviembre de 1995.- México.- pp. 20.
- Entrevista a Katrib Mora, Miguel.- Orientación a Objetos. Una oportunidad para la empresa y la academia.- Entrevistado por Ibarguengoitia Guadalupe.- Revista Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios.- Xview, S.A de C.V.- No. 22.- 15 de Junio de 1995.- México.- pp. 4 a 8.
- Calderón Macías, Marcos.- Una Visión General de los Sistemas Manejadores de Bases de Datos Orientadas a Objetos.- Revista Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios.- Xview, S.A de C.V.- No. 22.- 15 de Junio de 1995.- México.- pp. 37 a 43.
- Cárdenas García, Sergio R.- Medición e Ingeniería de Software.- Revista Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios.- Xview, S.A de C.V.- No. 22.- 15 de Junio de 1995.- México.- pp. 44 a 48.

- Padua Díaz, Marcela.- Transformational Outsourcing.- Revista Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios.- Xview, S.A de C.V.- No. 28.- 15 de Diciembre de 1995.- México.- pp. 4 a 6.
- Ríos Herrera, Alfonso.- Ambientes Distribuidos.- Revista Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios.- Xview, S.A de C.V.- No. 28.- 15 de Diciembre de 1995.- México.- pp. 20 a 23.
- Hernández, Rafael.- Administración de Redes de Telecomunicaciones.- Revista Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios.- Xview, S.A de C.V.- No. 18.- Febrero de 1995.- México.- pp. 24 a 27.
- Mejía Olvera, Marcelo y Incera Diéguez, José.- Ethernet en Evolución.- Revista Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios.- Xview, S.A de C.V.- No. 18.- Febrero de 1995.- México.- pp. 48 a 55.
- Roa Villarinos, Rafael.- Outsourcing: Cada quien a lo suyo.- Revista Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios.- Xview, S.A de C.V.- No. 13.- Septiembre de 1994.- México.- pp. 5 a 7.
- Gonzáles Lozano, Marisol.- Una Panorámica de la Reingeniería de Procesos.- Revista Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios.- Xview, S.A de C.V.- No. 30.- 15 de Febrero de 1996.- México.- pp. 31 a 39.
- González Sustaña, Jorge y Sandoval Jiménez, Patricia.- Nuevas Reglas de Oro en el Diseño de Interfaces Humano - Computadora.- Revista Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios.- Xview, S.A de C.V.- No. 10.- Junio de 1994.- México.- pp. 10 a 14.
- Soto Sumuano, Leonardo y Sánchez Sandoval, Alfonso.- Red X.25.- Revista Soluciones Avanzadas, Tecnologías de Información y Estrategias de Negocios.- Xview, S.A de C.V.- No. 7.- Enero-Febrero de 1994.- México.- pp. 16 a 19.

BYTE

- Halfhill, Tom.- Unix Vs Windows NT.- Revista Byte.- McGraw-Hill/Interamericana de México, S.A de C.V.- No. 100.- Mayo de 1996.- México.- pp. 6 a 20.
- Prado Angeles, Ernesto.- Ergonomía del Software.- Revista PC/TIPS Byte.- McGraw-Hill/Interamericana de México, S.A de C.V.- No. 53.- 15 de Junio de 1992.- México.- pp. 34 a 37.

- Ceiko, Joe y Ceiko, Jackie.- ¡Adiós! A los mitos ODBMS.- Revista Byte.- McGraw-Hill/Interamericana de México, S.A de C.V.- No. 117.- Octubre 1997.- México.- pp. 30 a 36.
- Zinno, Todd.- El Objeto es administrar datos.- Revista Byte.- McGraw-Hill/Interamericana de México, S.A de C.V.- No. 117.- Octubre 1997.- México.- pp. 20 a 26.

DR. DOBB'S

- Carone, Timothy.- Middleware and Three-Tier Client/Server Development.- Revista Dr. Dobb's Journal.- Miller Freeman Inc.- No. 253.- Junio de 1992.- E.U.A.- pp. 16 a 22.
- Bist, Gary.- Optimizing Client/Server Database Applications.- Revista Dr. Dobb's Journal.- Miller Freeman Inc.- No. 253.- Noviembre de 1996.- E.U.A.- pp. 24 a 35.
- Herold, Phill y Merrill Carla.- Interactive Design Methodology.- Revista Dr. Dobb's Journal.- Miller Freeman Inc.- No. 231.- Junio de 1995.- E.U.A.- pp. 40 a 47.
- Client/Server: Costs Vs Benefits.- Revista Datamation.- 15 de Junio de 1995.- E.U.A.- pp. 7 a 9.
- Wittenburg, Tim.- Data Models, CASE Tools, and Client/Server Development.- Revista Dr. Dobb's Journal.- Miller Freeman Inc.- No. 236.- November de 1995.- E.U.A.- pp. 92 a 96 y 99.

DBMS

- Shimberg, David.- Database Methodology. Following a Client/Server.- Revista DBMS Database & Client/Server Solutions.- Miller Freeman Inc.- No. 5.- Mayo de 1995.- E.U.A.- pp. 48 a 58 y 70 a 71.
- Tillmann, George.- Logical Data Modeling: The Keeper of the Vision.- Revista DBMS Database & Client/Server Solutions.- Miller Freeman Inc.- No. 5.- Mayo de 1995.- E.U.A.- pp. 60 a 68.
- Atre, Shaku.- Client/Server Culture Shock.- Revista DBMS Database & Client/Server Solutions.- Miller Freeman Inc.- No. 5.- Mayo de 1995.- E.U.A.- pp. 87 a 91.

MANUALES

- Manual de Usuario del Sistema de Seguimiento Financiero SISEFIN (Exploración – Producción PEMEX). Elaboró: Instituto Mexicano del Petróleo. – 1996. – México.
- Manual de Técnico del Sistema de Seguimiento Financiero SISEFIN (Exploración – Producción PEMEX). Elaboró: Instituto Mexicano del Petróleo. – 1996. – México.
- Manual de Instalación y Configuración del Sistema de Seguimiento Financiero SISEFIN (Exploración – Producción PEMEX). Elaboró: Instituto Mexicano del Petróleo. – 1996. – México.

OTROS

- Documento de Análisis y Diseño del Sistema de Seguimiento Financiero SISEFIN (Exploración – Producción PEMEX). Elaboró: Instituto Mexicano del Petróleo. – 1994. – México.
- Conectividad y Arquitectura Cliente / Servidor.- Glosario de Términos.- SHL SYSTEM HOUSE.- Edición 1995.

DIRECCIONES DE INTERNET

- <http://www.w3.org/hypertext/WWW/protocols/HTTP/HTTP2.html>
- <http://www.w3.org/hypertext/WWW/Clients.html>
- <http://www.oracle.com>

ANEXO A

INSTITUTO MEXICANO DEL PETROLEO

DIVISION DE SISTEMAS COMPUTACIONALES

PROPUESTA TECNICO - ECONOMICA

SISTEMA DE SEGUIMIENTO FINANCIERO DE LA GERENCIA DE INSPECCION Y MANTENIMIENTO

- SISEFIN -

SISTEMA DE SEGUIMIENTO FINANCIERO DE LA GERENCIA DE INSPECCION Y MANTENIMIENTO

PROPUESTA TECNICO - ECONOMICA

CONTENIDO

- 1.- INTRODUCCION
- 2.- DESCRIPCION DEL ESTUDIO
- 3.- RESULTADO DEL ESTUDIO DE LAS NECESIDADES
- 4.- OBJETIVO
- 5.- ALCANCES Y BENEFICIOS
- 6.- EVALUACION DE FACTIBILIDAD TECNICA Y OPERATIVA.
- 7.- PROPUESTAS PARA SATISFACER LOS REQUERIMIENTOS
- 8.- RECOMENDACIONES
- 9.- METODOLOGIA DE TRABAJO
- 10.- EVALUACION DE FACTIBILIDAD ECONOMICA
- 11.- DERECHOS DE AUTOR, GARANTIA Y SOPORTE TECNICO
- 12.- PROGRAMA DE ACTIVIDADES
- 13.- CONCLUSIONES

1.- INTRODUCCION

Petróleos Mexicanos la empresa más importante del país está experimentando una nueva era, sustentada en la concurrencia tecnológica entre las ciencias de la computación, las telecomunicaciones y las diversas áreas de la Ingeniería, teniendo como punto de convergencia a la informática; cuya notable evolución en cada una de estas tecnologías ha modificado progresivamente el ámbito que nos rodea.

En la actualidad las empresas líderes como PEMEX - Exploración Producción, requiere contar en forma oportuna y confiable de la información que apoye su toma de decisiones, para mejorar sus índices de productividad. En este caso la información presupuestal es de vital importancia porque influye en la economía nacional.

El Sistema Institucional de Control del Ejercicio Presupuestal SICEP, instrumenta los mecanismos y procedimientos esenciales para respaldar adecuadamente la gestión financiera del Control del Presupuesto, en materia de egresos de operación e inversión.

A nivel Región, se requiere llevar un control interno del seguimiento del presupuesto. Por esta razón se han desarrollado sistemas informáticos internos que proporcionan información como apoyo a dicho control. La limitación de estos sistemas reside en estar fuera de los alcances de una red, por lo que la transferencia de información es por medio de diskettes, lo que ocasiona que la información no esté actualizada en el momento que se requiere.

El Sistema Presupuestal de la Subdirección de Servicios Técnicos PRESST es otro sistema que cumple con el Control del Ejercicio Presupuestal pero a nivel subdirección. Por tal motivo la Gerencia de Inspección y Mantenimiento (GIM) requiere un sistema que lleve el Control Interno de su presupuesto.

El desarrollo de este sistema estará a cargo del Instituto Mexicano del Petróleo, y llevará el nombre de "Sistema de Seguimiento Financiero de la Gerencia de Inspección y Mantenimiento" (SISEFIN), el cual operará en ambiente de red bajo una arquitectura Cliente/Servidor.

En siguiente documento expone los objetivos y alcances del sistema SISEFIN, así como los resultados del estudio de factibilidad que se llevo a cabo para determinar los costos y beneficios que generará el desarrollo e implantación del sistema. El IMP también especificará su plan de trabajo y la más conveniente y adecuada infraestructura de computo para que operé el sistema.

El responsable de la propuesta es el Mat. Vázquez, teléfono 9 99 99 99, quien atenderá cualquier información adicional que se necesite.

2.- DESCRIPCION DEL ESTUDIO

Para la elaboración de la propuesta se recurrió a técnicas para encontrar y reunir datos relacionados con los requerimientos. Entre éstos se incluyó la entrevista, la revisión de registros y la observación.

Las fuentes de información analizadas fueron:

- Reporte de subproyectos generado mensualmente.
- Lista de Cap's proporcionados por Finanzas.
-

Entre las personas entrevistadas estuvieron :

NOMBRES	PUESTO
• Ing.	
• Lic.	
• Sr.	

El estudio se realizó en las instalaciones de PEMEX, en un periodo de 12/12/12 al 12/12/12. Las personas que intervinieron por parte del IMP para la elaboración de la propuesta fueron las siguientes:

NOMBRES	PUESTO
• Ing.	
• Lic.	
• Lic.	

3.- RESULTADO DEL ESTUDIO DE LAS NECESIDADES

A continuación se muestra un resumen de los problemas que actualmente se presentan en la forma que realizan sus actividades.

- Se operan sistemas alternos, los cuales están en ambiente stand alone. La transferencia de información se hace por medio de diskettes, ocasionando que esta no se encuentre actualizada en el momento que se requiere, e incluso la pérdida de la misma.
- Para la programación del presupuesto autorizado se generan a través de hojas de cálculo Excel, propiciando un control inadecuado en el seguimiento financiero.
- Se requiere tener una calculadora para hacer las estimaciones sobre las erogaciones mensuales por cada renglón del gasto y que concuerden con lo autorizado.
- En el control de las facturas, el proceso es manual.
- La información no esta compartida en red.
-

4.- OBJETIVO

GENERAL:

Implantar un sistema en ambiente de red de computadoras para el Control Interno del Seguimiento Financiero de la Gerencia de Inspección y Mantenimiento

PARTICULARES:

Llevar el control interno de los presupuestos a nivel subproyecto, dando seguimiento a los compromisos y facturas a través de la clave de autorización presupuestal y la asignación de una familia presupuestal.

5.- ALCANCES Y BENEFICIOS

ALCANCES

Desarrollar e implantar un sistema que:

- Opere en ambiente de red de computadoras acorde a la plataforma informática de la Subdirección de Servicios Técnicos.
- Genere consultas, reportes y gráficas.
- Contemple el seguimiento de las facturas de la gerencia a través de la Clave de Autorización Presupuestal y la asignación de una familia presupuestal a nivel subproyecto.
- Controle el gasto presupuestal de acuerdo a programado mensualmente.

BENEFICIOS

- Proporcionar un ambiente de operación amigable para el usuario (ambiente gráfico, manejo del mouse, etc.)
- Obtener el Control Interno del Ejercicio Presupuestal en forma confiable y oportuna.
- Realizar comparaciones de los resultados obtenidos del control interno contra la información obtenida de Finanzas PEP, con el objeto de detectar oportunamente desviaciones del presupuesto como cargos incorrectos, sobregiros presupuestales y cargos no autorizados.

6.- ESTUDIO DE FACTIBILIDAD OPERATIVA Y TÉCNICA

Para llevar a cabo el control interno del ejercicio presupuestal se utilizan alternamente sistemas informáticos, que proporcionan una parte del apoyo a dicho control. La limitación de estos sistemas reside en estar fuera de los alcances de una red, por lo que la transferencia de información es por medio de diskettes, causando que la información no se encuentre actualizada en el momento que se requiere.

Ante esta problemática el personal está de acuerdo en tener un sistema que les ayude con sus tareas, aunque ya han tenido experiencias con otros sistemas presupuestales que les han ofrecido pero que ninguno cumple con sus necesidades primordiales como gerencia, ellos no quisieran que fuera lo mismo. Por lo que se les explicó que el sistema SISEFIN prácticamente lo iban a diseñar ellos, cumpliendo con sus requerimientos actuales, además que el sistema se enfocaría a usuarios presupuesteros de una gerencia y no de subdirección u otros niveles organizacionales.

Los usuarios se mostraron entusiastas pues sabían que el sistema se les estaba desarrollando para ellos y no otro sistema ya terminado que se les iba a implantar.

Se considera la actitud de los usuarios buena y favorable para el desarrollo del sistema.

En cuanto a la infraestructura de cómputo, la GIM y en todas sus regiones (Norte, Sur, Marina Noroeste y Marina Suroeste) cuentan con una red Ethernet con servidores RISC y como manejador de base de datos a Oracle con sistema operativo UNIX, las computadoras son 386, 486 y Pentium. Para el desarrollo del sistema SISEFIN se requiere la compra de la herramienta SUPERBASE el cual tiene un costo de \$

Se califica a la infraestructura satisfactoria para el desarrollo del sistema.

7.-PROPUESTA PARA SATISFACER LOS REQUERIMIENTOS

Actualmente, para llevar el control interno del ejercicio presupuestal de la GIM se operan sistemas alternos, los cuales están en ambiente stand alone. La transferencia de información se hace por medio de diskettes, ocasionando que esta no se encuentre actualizada en el momento que se requiere, e incluso la pérdida de la misma. Por lo anterior se propone un sistema en arquitectura Cliente/Servidor en el cual se pueda concentrar la información en un servidor y los usuarios puedan accederla sabiendo que esta actualizada.

En el proceso actual del presupuesto autorizado para su programación y estimación, se generan a través de hojas de cálculo Excel. A veces la información se llega a perder y por lo mismo es imposible llevar un control adecuado, además de que se requiere tener una calculadora para hacer las estimaciones sobre las erogaciones mensuales por cada renglón del gasto y que concuerden con lo autorizado. El sistema que se propone llevará el control interno a nivel subproyecto (Nivel a detalle de la familia presupuestal que no en todas las áreas se aplica) permitiendo tener un mejor seguimiento de los gastos generados en la gerencia. Y también se tendrá un fácil manejo en el planteamiento de estimaciones.

En el control de las facturas, actualmente el proceso es manual, se reciben y se archivan bajo un proyecto, pero no se tiene un seguimiento sobre qué facturas están afectando determinado subproyecto y cuál es su monto acumulado por mes, y si coincide son lo estimado. Además de que no se sabe exactamente si cada factura esta respaldada por una clave de autorización presupuestal (CAP). El sistema SISEFIN contemplará un módulo de Facturación las cuales serán capturadas bajo un subproyecto, indicando el dinero disponible de acuerdo al mes y también se pedirá cada factura tenga un CAP o en su defecto un subproyecto, para respaldar la erogación. Igualmente se tendrá un control de que facturas ya se devengaron y los compromisos que se han generado.

Puesto que no existe un sistema que lleve a cabo estos procesos, los reportes mandados por Finanzas sobre los presupuestos de la gerencia, tenían que ser aceptados y estar conforme con ellos, porque si existía desviaciones o cargos incorrectos, la gerencia no podía justificarlos o no se detectaban oportunamente, El sistema SISEFIN generará reportes y gráficas que además de ayudar a tomar decisiones más adecuadas sobre el control del presupuesto de la gerencia, proporcionará la base para comparar los resultados con los de Finanzas y detectar posibles desviaciones.

8.-RECOMENDACIONES

Conforme al estudio de la infraestructura con que dispone actualmente la gerencia, no tenemos problema es recomendar una plataforma de computo basada en la tecnología Cliente/Servidor. Y ajustándonos a los estándares institucionales de PEMEX sobre hardware y software, se propone el siguiente equipo:

REQUERIMIENTOS DE HARDWARE:

Servidor

- 1.- 10 MB de espacio en disco duro (ambiente lógico y físico).
- 2.- 32 MB de memoria RAM.
- 3.- Medio de almacenamiento (CD, unidad de cinta de 1/4 de pulgada, unidad de cinta de 8mm, etc).
- 4.- Canales de comunicación locales y remotos para integrar información de los distintos niveles jerárquicos de PEMEX (PEMEX-PAQ, FRAME RELAY, etc).

Cliente

- 1.- 20 MB de espacio en disco duro para la instalación y operación del sistema.
- 2.- 8 MB de memoria RAM o superior.
- 3.- Impresora esclava o en red.
- 4.- Monitor VGA.
- 5.- Procesador 80386 DX/33 o superior.

REQUERIMIENTOS DE SOFTWARE:

Servidor

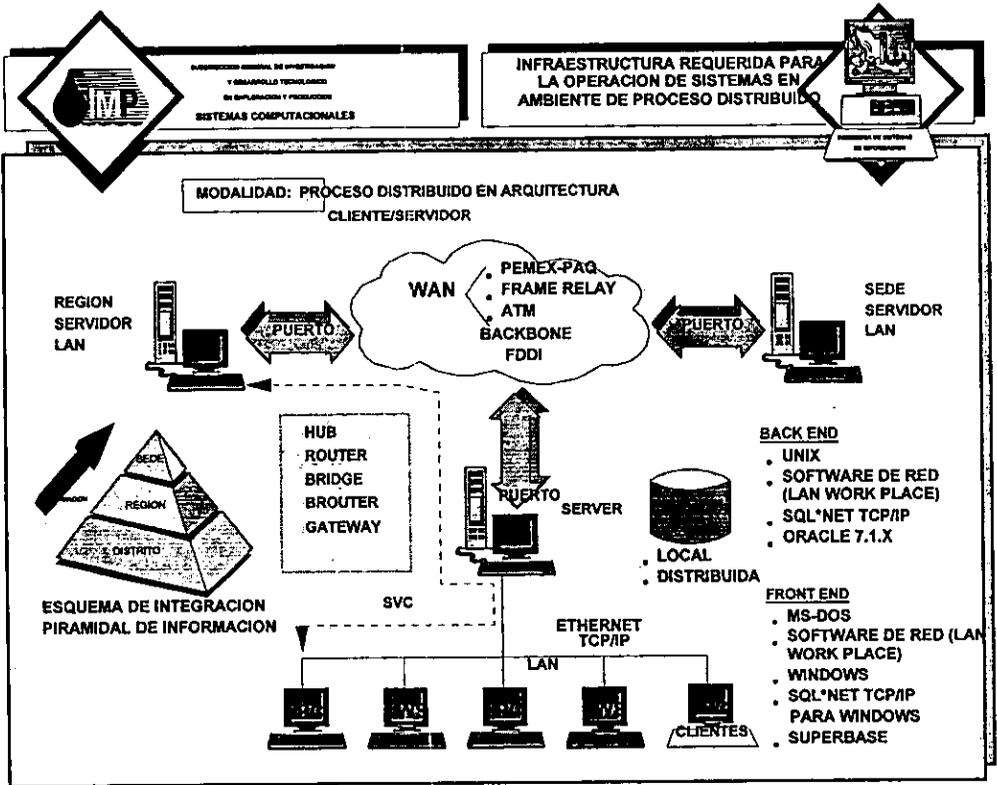
- 1.- Sistema Operativo UNIX.
- 2.- Software de red (LAN WORK PLACE o SYNTAX).
- 3.- SQL *NET TCP/IP.
- 4.- ORACLE v 7.X. (Back End).

Cliente

- 1.- Sistema Operativo MS-DOS v 6.x o Windows 95.
- 2.- Software de red (LAN WORK PLACE o SYNTAX).
- 3.- Microsoft WINDOWS v 3.x o 95
- 4.- SQL *NET TCP/IP.
- 5.- Runtime SUPERBASE v.2.01 (Front End).

Nota: El Front end propuesto es el Superbase ver. 2.,01 con la finalidad de aprovechar código reutilizable escrito en la fase anterior del sistema.

INFRAESTRUCTURA DE OPERACION



9.- METODOLOGIA DE TRABAJO

Es necesario hacer hincapié que esta es la segunda etapa (no versión) del sistema SISEFIN, anteriormente el sistema tenía otros alcances, que por cuestiones políticas y administrativas tuvieron que volverse a definir y lo que ya se tenía desarrollado como aplicación fue estudio para realizar la reingeniería conforme a los nuevos requerimientos del sistema SISEFIN

La metodología de trabajo que se propone para el desarrollo del sistema es :

1. Integración del Grupo de Trabajo PEMEX - IMP
2. Análisis Específico (Reingeniería de Procesos).
3. Diseño Preliminar (Reingeniería de Software).
4. Elaboración del Prototipo.
5. Diseño Final.
6. Construcción del sistema.
7. Implantación
8. Operación del sistema
9. Liberación

1) INTEGRACION DEL GRUPO DE TRABAJO PEMEX - IMP

- Designación de responsables.
- Especificación de objetivos.

2) ANALISIS ESPECIFICO (REINGENIERIA DE PROCESOS)

- Análisis de las necesidades del usuario
- Análisis de los requerimientos de información
- Análisis detallado de los procesos para la generación de la información
- Definición de los módulos del sistema
- Determinación de funciones y procesos para cada módulo
- Definición del ambiente de conectividad y la arquitectura cliente / servidor

3) DISEÑO PRELIMINAR (REINGENIERIA DE SOFTWARE)

- Definición de entidades de datos
- Estudio del código de programación del anterior sistema
- Diagrama de flujo de datos preliminares
- Esquematación de procesos relevantes
- Desarrollo del diccionario de datos preliminar (elementos de información)
- Definición preliminar del diagrama entidad - relación

- Diseño preliminar de entradas y salidas del sistema
- Diseño de la conectividad local y remota
- Diseño de los canales de comunicación local y remota

4) ELABORACION DE UN PROTOTIPO

- Construcción
- Presentación al usuario para el Vo. Bo.
- Ajustes
- Aprobación del prototipo por los usuarios

5) DISEÑO

- Revisión del diseño preliminar
- Especificaciones técnicas y estándares de programación
- Diagrama de flujo de datos
- Especificación de procesos relevantes.
- Diagrama Entidad - Relación
- Estructura de la Base de Datos
- Diccionario de datos
- Diseño de módulos de captura
- Diseño de módulos de procesamiento
- Diseño de módulo de reportes
- Diseño de módulo de gráficas
- Diseño de los canales de comunicación
- Diseño de la arquitectura cliente / servidor

6) CONSTRUCCION

- Ejecución de la estructura de la Base de Datos.
- Programación del sistema
- Pruebas y ajustes
- Construcción de Arquitectura cliente / servidor
- Visitas programadas con los usuarios para la revisión de los avances.
- Desarrollo del Instalador del Sistema.
- Elaboración de los manuales técnico y usuario.

7) IMPLANTACION

INSTALACION

- Elaboración del programa de instalación
- Identificación y verificación de Hardware, Software y Humanware de cada uno de los lugares de Instalación.
- Instalación en el cliente y en el servidor
- Selección del tipo de instalación
- Pruebas y ajustes locales e integrales del sistema

CAPACITACION

- Elaboración del programa de capacitación
- Material necesario
- Manual técnico y de usuario

8) OPERACION DEL SISTEMA

SUPERVICION DEL SISTEMA

- Operación del sistema con información real
- Generación de reportes y gráficas
- Pruebas especiales de sistemas

ASESORIA EN LA OPERACION DEL SISTEMA

- Visitas Programadas

9) LIBERACION

ENTREGA DE DOCUMENTACION FINAL

- Manual Técnico
- Manual de Usuario

ENTREGA DEL SOFTWARE

- Discos de instalación
- Guías de instalación

ENTREGA OFICIAL DEL SISTEMA

- Elaboración del acta oficial de entrega al coordinador del proyecto
- Negociación del mantenimiento evolutivo del sistema

10.- EVALUACION DE FACTIBILIDAD ECONOMICA**RECURSOS HUMANOS**

El personal que intervendrá en el desarrollo del sistema es el siguiente:

Lic.- Licenciado en Informática.

Líder de proyecto.

- Especialista con amplia experiencia en desarrollo de sistemas en arquitectura CLIENTE/SERVIDOR a través de UNIX/ORACLE-SYBASE/SQL WINDOWS-SUPERBASE.

•

Ing.- Ing. en Computación

Analista Programador.

- Experiencia en el desarrollo de sistemas de cobertura regional en CLIENTE/SERVIDOR a través de UNIX/ORACLE/SQL WINDOWS.

•

COSTO DE LA HERRAMIENTA DE DESARROLLO SUPERBASE : \$**TIEMPO Y COSTO PARA EL DESARROLLO DEL SISTEMA**

Duración en meses : Mes1 a Mes2 de Año

Duración en horas: 00:00 horas
Lunes - Viernes 00:00 - 00:00
00:00 - 00:00

COSTO POR MANO DE OBRA.-

1 Líder de proyecto	100%	Nivel C-..	\$ 0.00/hr. *	00:00 hrs =	\$ 0.00
2 Analistas programadores	100%	Nivel C-..	\$ 0.00/hr. *	00:00 hrs =	0.00
2 Analista programador Jr.	100%	Nivel C-..	\$ 0.00/hr. *	00:00 hrs =	0.00

SUBTOTAL **00 :00 HRS.** **\$0.00**

VIÁTICOS ESTIMADOS

Viajes estimados \$ 0.00

COSTO TOTAL DEL SISTEMA : \$ 0.00

11.- DERECHOS DE AUTOR

- El Instituto Mexicano del Petróleo se compromete a no negociar con alguna otra empresa el sistema que se propone desarrollar. Podrá ser presentado ante otras dependencias solo para efectos de curricula y previa autorización escrita de la Gerencia de Administración y Finanzas de la Región Norte.
- El IMP se compromete a no difundir el sistema en eventos nacionales ni extranjeros, tales como congresos o seminarios.
- El código que se utilice en el sistema es propiedad del IMP y podrá ser reutilizado para el desarrollo de otros sistemas, sin perjuicio de la Gerencia de Planificación.

GARANTIA Y SOPORTE TECNICO

- A la entrega oficial del sistema, se elaborará un acta para su recepción por parte de la Subgerencia de Informática de la Gerencia de Planificación.
- El sistema tiene garantía contra vicios ocultos, únicamente en software construido por el IMP y considerando únicamente los alcances descritos en las especificaciones de diseño. No se aplica en caso de existir anomalías por intervenciones a los programas de cómputo, efectuadas por personal ajeno al grupo de trabajo o por problemas en la red o comunicaciones. **La vigencia de la garantía será de 12 meses, sin costo alguno para la Gerencia.**
- Durante 12 meses, el IMP proporcionará soporte técnico a través de vía telefónica o red de PEMEX PAQ, en la operación del sistema o si se justifica en Poza Rica.
- En caso de tener que aplicarse la garantía en sitio, es decir, Poza Rica, esta se atenderá en un tiempo máximo de 48 hrs. Para tal efecto, se efectuará un diagnóstico de la falla; la Gerencia se compromete asignar a un responsable que conozca los conceptos y operación del sistema, para atender a tiempo completo al personal del IMP. Si el diagnóstico muestra que no hay falla en el software del sistema elaborado por el IMP, o es debido a deficiencias en su operación, red o comunicaciones, los gastos que generen serán a cargo de la Gerencia.
- Para atender algunos requerimientos adicionales, el Área de Especialidad en Computación Aplicada del IMP, negociará con la Subgerencia de Informática su tiempo y costo.

12.- PROGRAMA DE ACTIVIDADES

INSTITUTO MEXICANO DEL PETROLEO
SUBDIRECCION GENERAL DE INVESTIGACION Y DESARROLLO TECNOLÓGICO EN EXPLORACION Y PRODUCCION

PROYECTO: **COC-0237 "Sistema de Seguimiento Financiero de la Gerencia de Inspección y Mantenimiento."**

SUBDIRECCION: **Servicios Técnicos** RESPONSABLE POMEK: **Ing.**
FECHA INICIO: **1996** TERM: **1996**

No.	ACTIVIDAD	SEMANAS																														
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
1	ANÁLISIS DE RECURSOS																															
	ANÁLISIS DE LAS NECESIDADES																															
	ANÁLISIS DE LOS REQUERIMIENTOS																															
	ANÁLISIS DE NÚMERO DE LOS PROCESOS																															
	DEFINICIÓN DE LOS MODULOS DEL SBT DE TERCER NIVEL PARA CADA MOD.																															
2	DISEÑO PRELIMINAR																															
	DEFINICIÓN DE ENTREGAS DE DATOS																															
	PROGRAMACIÓN DE DATOS																															
	ELABORACIÓN DE PROGRAMA DE DATOS																															
	DEFINICIÓN PRELIMINAR DEL DIAGRAMA DE																															
3	PROTOTIPO																															
	CONSTRUCCIÓN DEL PROTOTIPO																															
	PRESENTACIÓN AL USUARIO																															
	AJUSTES AL PROTOTIPO																															
	APROBACIÓN DEL PROTOTIPO																															
4	DISEÑO RNA																															
	DISEÑO DE MODULOS DE CAPTURA																															
	DISEÑO DE MODULOS DE PROCESAMIENTO																															
	DISEÑO DE CANALES DE COMUNICACIÓN																															
	DISEÑO DE LA ARQUITECTURA OS																															

PROGRAMADO	AVANCE TÉCNICO	PRG. ACUM. M.																														
	%	RE. ACUM. M.																														
REAL	AVANCE FINANCIERO	PRG. ACUM. M.																														
	M. LIBRES	RE. ACUM. M.																														

PRESUPUESTO MILES \$

NOTA.- En el diagrama de barras que muestra el calendario de actividades, se indica lo siguiente :

- Actividad, No. de personas/actividad, lugar, porcentaje de la actividad en el sistema, semanas/actividad.
- Avance técnico(%), programado acumulado semanal, avance técnico real de la semana y avance real acumulado hasta esa semana.
- Avance financiero, programado acumulado semanal, avance financiero real de la semana y avance financiero real acumulado hasta esa esa semana.
- El IMP aplicará la facturación mensualmente previa comprobación del avance técnico.

13.- CONCLUSIONES

La propuesta para el desarrollo de un sistema que controle el seguimiento financiero es enteramente factible en todos los aspectos tanto técnicos, financieros y operativos. En cuestión a la factibilidad técnica la construcción de la arquitectura Cliente/Servidor tiene ventajas considerables que se muestran en el documento, entre las principales están rendimiento en el procesamiento de datos, fácil acceso de la información en cualquier punto físico donde se encuentre el sistema, confiabilidad y seguridad en el manejo de la información y sobre todo fácil operación del sistema proporcionando un ambiente amigable. Además existe la gran ventaja, de que la gerencia cumple con todos las condiciones para crear la plataforma Cliente/Servidor, sólo es necesario configurar los componentes. Esto significa que el costo de hardware y software es mínima para que se desarrolle el sistema SISEFIN. Tan solo se debe considerar la compra de la herramienta de desarrollo llamada SUPERBASE que en el mercado tiene un precio de ...

Con respecto a los usuarios están entusiasmados por operar un sistema que les ayude al control financiero de la gerencia y sobre todo que les proporcione informes y consultas de compromisos y erogaciones presupuestales y los movimientos que va teniendo el flujo de dinero de la gerencia en el transcurso de su periodo presupuestal.

Por lo que nuestra parte concierne se tiene la suficiente experiencia en la construcción de sistemas Cliente/Servidor, consiguiendo un prestigio en la calidad de nuestros productos, y sobre todo siendo competitivos y proporcionando el más aceptable costo en el desarrollo de sistemas Cliente/Servidor, cotizado conforme a las capacidades de la Gerencia y a los beneficios futuros al operar el sistema.

Agradeceremos la participación de la gerencia en la elaboración de la propuesta. Y estaremos al pendiente de su respuesta.