



**UNIVERSIDAD NACIONAL AUTÓNOMA DE
MÉXICO**

**FACULTAD DE ESTUDIOS SUPERIORES
CUAUTITLÁN**

**DESARROLLO DE UN SISTEMA DE MANEJO DE
INFORMACIÓN PARA LA SIMULACIÓN Y
OPTIMIZACIÓN DE LOS PROCESOS DE
PRODUCCIÓN DE HIDROCARBUROS**

TESIS

QUE PARA OBTENER EL TÍTULO DE:

**LICENCIADA EN INFORMÁTICA
PRESENTA:**

CRISTINA CRUZ TOLEDO

**ASESOR: L.C. CARLOS PINEDA MUÑOZ
COASESOR: E. en I. ENRIQUE SOTO RAMOS**

CUAUTITLÁN IZCALLI, EDO. DE MÉX.

2004

**TESIS CON
FALLA DE ORIGEN**



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.

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



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

ASUNTO: VOTOS APROBATORIOS

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



SECRETARÍA DE EDUCACIÓN PÚBLICA
SECRETARÍA DE ESTADOS

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

ATN: Q. Ma. del Carmen García Mijares
Jefe del Departamento de Exámenes
Profesionales de la FES Cuautitlán

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

"Desarrollo de un Sistema de manejo de Información para la Simulación y Optimización de los Procesos de Producción de Hidrocarburos"

que presenta la pasante: Cristina Cruz Toledo
con número de cuenta: 9509165-5 para obtener el título de :
Licenciada en Informática

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

ATENTAMENTE
"POR MI RAZA HABLARA EL ESPIRITU"

Cuautitlán Izcalli, Méx. a 12 de Agosto de 2004

PRESIDENTE	<u>L.C. Carlos Pineda Muñoz</u>
VOCAL	<u>MCC. Araceli Nivon Zaghi</u>
SECRETARIO	<u>L.C. Magali Valdivia Velasco</u>
PRIMER SUPLENTE	<u>MAI. Manuel Jauregui Renault</u>
SEGUNDO SUPLENTE	<u>Ing. Antonio González Almaguer</u>

AGRADECIMIENTOS

A mis padres:

Por su apoyo e impulso para llegar hasta aquí.

A Alma:

Por su ayuda en el diseño y la realización de mi presentación.

Al IMP:

Por permitirme aprender y realizar mi estancia de trabajo de tesis.

Al GOIP:

Por recibirme con gusto como una compañera más, principalmente al área de Optimización por sus explicaciones y enseñanzas y por facilitarme el uso tanto del equipo como del software.

Al Ing. Enrique Soto:

Por asesorarme durante más de un año y apoyarme en la realización de mi trabajo, por su tiempo dedicado.

A ARA:

Por su ayuda, por darme ánimos para continuar cuando creía que no llegaría este día y por estar conmigo.

A mis compañeros becarios:

Por haberme hecho pasar un tiempo muy agradable y por su compañía, amistad, por escucharme.

Al Prof. Carlos Pineda:

Por su orientación en la realización del trabajo y su entusiasmo para que la carrera obtenga más titulados. Por su tiempo, y explicaciones en cuanto a trámites y situación laboral.

A la UNAM:

Por cobijarme todos estos años y formarme como profesionista.

A mis sinodales:

Por su tiempo dedicado a revisar mi trabajo y por sus observaciones y opiniones que me ayudaron a entender algunas cosas.

A todos mis profesores:

Por sus enseñanzas tanto académicas, como culturales y sociales.

DEDICATORIAS

A mis papás:

Por darme lo mejor: una profesión, por su cariño.

A mis hermanas:

Por crecer conmigo y por sus enseñanzas.

A Temin Pingûin:

Por su amor y por aguantarme todo este tiempo.

A mis amigas:

Lluvia, Dalila, Irene, Alma, Elisa, Amelia, Luz, Nancy. Por su amistad.

A mis amigos:

Goyito, Alejandro, Pablito, José Luis, Saúl, Enrique, Reynaldo. Por escucharme y estar ahí.

A mi tía Angélica y a mis primas:

A mi tía por su preocupación y apoyo. A mis primas por su compañía.

A las nenitas:

Por su alegría y compañía y por sus consejos, sus juegos, por estar conmigo.

A la Lic. Prisca y a Mari:

A la Lic. por su ejemplo para titularme y a ambas en agradecimiento de su apoyo en la realización de mi servicio social.

	Pág.
OBJETIVOS	vii
HIPÓTESIS	vii
PROBLEMÁTICA	vii
JUSTIFICACIÓN	viii
ALCANCES	viii
INTRODUCCIÓN	ix
CAPÍTULO 1. MARCO TEÓRICO GENERAL	1
1.1. Datos e Información	3
1.1.1. Datos	3
1.1.2. Información	4
1.1.3. Análisis de la diferencia entre datos e información	4
1.1.4. Datos e información desde un punto de vista informático	6
1.1.5. Fuentes de información	8
1.1.6. Cualidades de la información	8
1.1.7. Importancia de la información en las organizaciones	10
1.2. Sistemas de Información	11
1.2.1. Conceptos	11
1.2.2. Características	15
1.2.3. Clasificación	15
1.2.4. Componentes	18
1.2.5. Ciclo de vida del desarrollo de un sistema de información	21
1.3. Bases de Datos	26
1.3.1. Historia	26
1.3.2. Definición	27
1.3.3. Clasificación	31
1.3.4. Ventajas de las Bases de Datos	32
1.3.5. Organización de una base de datos	32
1.3.6. Modelos de Bases de datos	33
1.3.7. Sistemas de Gestión de Bases de Datos	35
1.4. Procesos de producción de hidrocarburos	41
1.4.1. El sistema de producción de hidrocarburos	41
1.5. Simulación de sistemas	42
1.5.1. Simulación frente a soluciones analíticas	42
1.5.2. Naturaleza de la simulación	43
1.5.3. Definición de la simulación en computadoras	44
1.5.4. Uso y limitaciones de la simulación	46
1.5.5. Aplicaciones de la simulación	47
1.5.6. Modelos para simulación	48
1.6. Optimización	49
1.6.1. Definición de problemas de optimización	49
1.6.2. Optimización de procesos	49
CAPÍTULO 2. MARCO TEÓRICO ESPECÍFICO Y FASE PRELIMINAR DEL PROYECTO	51
2.1. Estándares y normas que regulan la comunicación de datos entre diversas aplicaciones	52
2.1.1. DDE (Dynamic Data Exchange)	53

2.1.2.	CORBA (Common Object Request Broker Architecture)	55
2.1.3.	DLL (Dynamic-Link Library)	55
2.1.4.	OLE (Object Linking and Embedding)	56
2.1.5.	COM (Component Object Model)	57
2.1.6.	DCOM (Distributed COM)	60
2.1.7.	OCX (OLE Custom Controls)	62
2.1.8.	ActiveX	63
2.1.9.	OPC (OLE for Process Control)	63
2.1.10.	OLE DB (OLE Data Base)	65
2.1.11.	ODBC (Open Database Connectivity)	66
2.1.12.	DAO (Data Access Objects)	66
2.1.13.	ADO (ActiveX Data Objects)	67
2.1.14.	MDAC (Microsoft Data Access Components)	67
2.2.	Software para Simulación y Optimización de procesos de producción de hidrocarburos	68
2.2.1.	Simuladores de procesos	68
2.2.2.	Compañía Baker Jardine & Associates	76
2.2.3.	Compañía Simsci-Escor de Invensys plc	78
2.3.	Identificación de datos requeridos y generados por las herramientas de simulación	81
2.3.1.	PIPESIM 2000	81
2.3.2.	PRO/II	82
2.4.	Comunicación de datos entre los simuladores de proceso	83
2.4.1.	PIPESIM 2000	83
2.4.2.	PRO/II	85
2.5.	Análisis de Datos	92
2.6.	Alternativas existentes para la integración de los datos al sistema de información	93
2.6.1.	Métodos para la integración de los datos al sistema de información	93
2.7.	Propuesta de solución para la integración de los datos al sistema de información	97
CAPÍTULO 3. DESARROLLO DEL SISTEMA DE INFORMACIÓN PARA LA SIMULACIÓN DE PROCESOS DE PRODUCCIÓN DE HIDROCARBUROS		99
3.1.	Razones para integrar	100
3.1.1.	Integración de las aplicaciones en los sistemas de información	101
3.2.	Planificación y Análisis del sistema	101
3.2.1.	Planificación del sistema	101
3.2.2.	Análisis del Sistema	109
3.3.	Configuración de hardware y software a utilizar	111
3.3.1.	Hardware	111
3.3.2.	Software	112
3.4.	Modelos Conceptuales de Datos (Diseño Lógico)	114
3.5.	Diseño de la Base de Datos (Diseño Físico)	126
3.6.	Integración de información de proceso a la Base de Datos	128
3.7.	Integración de la información requerida y generada por los simuladores de proceso	130
3.8.	Instrucciones más importantes utilizadas en el Sistema de Información	156
3.9.	Puesta en marcha del sistema. Prueba	159
3.9.1.	Requerimientos de instalación del Sistema	159
3.9.2.	Implantación del sistema	159
CAPÍTULO 4. EVALUACIÓN DEL SISTEMA Y PROPUESTAS DE SOLUCIÓN PARA LÍNEAS DE DESARROLLO FUTURO		161
4.1.	Evaluación y análisis del sistema	162
4.1.1.	Soporte del sistema	163
4.2.	Resultados	163
4.3.	Propuestas de solución para líneas de desarrollo futuro	164

4.3.1. Adición de otras herramientas de simulación y optimización al sistema de información	164
4.3.2. Proyecto de Microsoft Access (aplicación cliente-servidor)	164
4.3.3. Sistema de Información en línea/tiempo real	166
4.3.4. Sistema Integral de Control y Manejo de Información	171
CONCLUSIONES	172
GLOSARIO	174
NOMENCLATURA	179
BIBLIOGRAFÍA	181
ANEXOS	184
Anexo A Diccionario de Datos	A-1
Anexo B Ejemplo de las tablas de la base de datos del Sistema de información	B-1
Anexo C Herramientas de simulación utilizadas en la aplicación	C-1

ÍNDICE DE FIGURAS

Figura	Pág.
1. Diferentes datos, que pueden ser simbólicos, icónicos, lingüísticos, lógicos, matemáticos, numéricos, entre otros.	3
2. Representación del procesamiento de datos en una computadora.	6
3. Representación de estados de un circuito electrónico: encendido y apagado.	7
4. Representación de un byte (formado por 8 bits).	7
5. Cada carácter está formado por una combinación de 8 bits.	7
6. Representación lógica de los caracteres almacenados en un campo	7
7. Visión del programador de un archivo del Departamento de Personal de una Compañía Ficticia.	8
8. Resumen de los atributos de la calidad de la información. Aquí se describen los atributos que deben estar presentes en los productos de información de alta calidad.	9
9. El sistema y su entorno.	13
10. Una empresa es un sistema organizacional donde los recursos económicos (entrada) se transforman mediante diversos procesos organizacionales (procesamiento) en bienes y servicios (salida). Los sistemas de información proporcionan información (retroalimentación) sobre las operaciones del sistema a la gerencia para la dirección y el mantenimiento del sistema (control), intercambian entradas y salidas con su entorno.	13
11. Modelo en pirámide de los sistemas de información informáticos.	14
12. Sistemas de Apoyo interrelacionados.	17
13. Sistemas de Información y Apoyo a la Gente.	18
14. Componentes de un Sistema de Información.	19
15. Ciclo de vida de un producto.	21
16. Ciclo de vida de un sistema de información.	22
17. Los ciclos de vida del desarrollo de sistemas clásico y moderno.	23
18. Modelo del proceso de las etapas del ciclo de vida del desarrollo de un sistema de información.	25
19. Jerarquía de organización de datos empleada por los sistemas de procesamiento basados en computadora. El nivel más alto de la jerarquía de organización de datos es la base de datos, que es un conjunto de elementos de datos relacionados entre sí.	27
20. Diferentes definiciones de base de datos.	29
21. Vista esquemática de datos almacenados. Los datos se almacenan físicamente en una disposición distinta a la de la perspectiva lógica. Todos los usuarios pueden tener acceso a los datos.	30
22. La estructura lógica del usuario o esquema externo de la visión que tiene de la base de datos cada usuario en particular; la estructura lógica global o esquema conceptual responde al enfoque del conjunto de la empresa y la estructura física o esquema interno es la forma en que se organizan los datos en el almacenamiento físico.	33
23. La representación de datos en una estructura jerárquica de datos se realiza con base en las ramificaciones.	34
24. Las estructuras en red permiten que la entidad cuente con numerosos subordinados o superiores, y las entidades se conectan por medio de enlaces comunes.	34
25. En una estructura de datos relacional los datos se almacenan en varias tablas.	34
26. Base de datos relacional.	35
27. Niveles de gestión en las organizaciones y obtención de información mediante grupos de aplicaciones no integradas (cada una con sus propios datos).	37
28. Niveles de gestión en las organizaciones u obtención de información mediante un sistema integrado a partir de una base común de datos).	38
29. Esquema básico de la Arquitectura Cliente-Servidor.	40
30. Proceso de producción de hidrocarburos.	41
31. El concepto de simulación.	43
32. Los servicios de aplicación OLE de alto nivel están contruidos en los fundamentos del Modelo de Objetos Componentes.	60

33. Componentes COM en el mismo proceso.	61
34. Componentes COM en diferentes procesos.	61
35. DCOM: Componentes COM en diferentes máquinas.	62
36. Esquema general de comunicación basada en el estándar OPC.	65
37. Clasificación de simuladores según su aplicación.	70
38. Suite de Desempeño en Línea.	71
39. Suite de Ingeniería de Procesos.	72
40. Suite de Optimización contra la corriente.	73
41. Acceso a las Bases de Datos.	86
42. Transferencia de Datos OLE.	88
43. Introducción de datos al sistema de información.	97
44. Plataforma estándar para comunicación de datos entre las diferentes aplicaciones (software de simulación).	98
45. Pirámide de optimización de procesos.	103
46. Pirámide de automatización.	104
47. Directrices Tecnológicas en la Pirámide de Automatización.	106
48. Organigrama del IMP.	107
49. Organigrama del GOIP.	108
50. Diagrama de flujo de datos de la problemática en el campo de la optimización de los procesos de producción.	110
51. Operación actual del proceso de simulación y modelado (operación fuera de línea).	110
52. Tabla: Características_de_Ductos.	119
53. Tabla: Características_de_Ductos.	128
54. Formulario: FR_Características_de_Ductos.	128
55. Tabla: Variables_de_Pozo.	129
56. Formulario: FR_Variabes_de_Pozo.	129
57. Tabla: Especificaciones_de_Tubería.	129
58. Formulario: FR_Especificaciones_de_Tubería.	130
59. Formato de las pantallas del Sistema de Información.	131
60. Diagrama de transición de estados para el Sistema de Información para la Simulación y Optimización de Procesos de Producción de Hidrocarburos.	132
61. Diagrama de transición de estados para la Ventana de Opciones_Lectura.	133
62. Diagrama de transición de estados para la Ventana de Leer_Modelo_PROII.	133
63. Diagrama de transición de estados para la Ventana de Opciones_Edición.	134
64. Módulo Principal.	136
65. Módulo: FR_Variabes_de_Pozo.	137
66. Módulo: FR_Características_de_Ductos.	137
67. Módulo: FR_Especificaciones_de_Tubería.	137
68. Módulo: Leer datos configurados de un modelo (pozo y red).	137
69. Módulo: Leer_Modelo_Pozo.	138
70. Módulo: FR_Leer_Pozo.	138
71. Módulo: Informe_Leer_Pozo.	138
72. Módulo: Leer_Modelo_Red.	139
73. Módulo: Leer_Modelo_PROII.	139
74. Módulo: FR_Leer_PROII.	139
75. Módulo: Informe_Leer_PROII.	140
76. Módulo: Editar datos de un modelo (pozo y red - procesos).	140
77. Licencia de PIPESIM.	141
78. Licencia de PRO/II.	142
79. Módulo: Editar_Modelo_Red_Pozo.	143
80. Módulo: Informe_Edición_Modelo_Pozo.	143
81. Módulo: Reporte_Pozo_Individual.	143
82. Módulo: Informe_Edición_Modelo_Red.	144

83. Módulo: Reporte_Red_Individual.	144
84. Módulo: Editar_Modelo_PRO/II.	144
85. Módulo: Informe_Flash_PROII.	145
86. Módulo: Reporte_Flash_PROII.	145
87. Pantalla de seguridad.	145
88. Pantalla Principal.	146
89. Formulario: FR_Variables_de_Pozo.	147
90. Formulario: FR_Características_de_Ductos.	148
91. Formulario: FR_Especificaciones_de_Tubería.	149
92. Formulario: Leer_Modelo_Pozo.	150
93. Informe: Informe_Leer_Pozo.	151
94. Formulario: Leer_Modelo_Red-Datos impresos.	152
95. Formulario: Leer_Modelo_PROII-Datos impresos.	153
96. Formulario: Edición_Modelo_Red_Pozo-Mostrar Resultados.	154
97. Formulario: Edición_Modelo_PROII-Mostrar Resultados.	155
98. Sistema de información conteniendo software de todos los procesos de producción de hidrocarburos.	164
99. Arquitectura de un proyecto de Access.	166
100. Un sistema en línea.	167
101. Un sistema en tiempo real.	168
102. Componentes de un sistema de optimización en línea/tiempo real.	169
103. Proceso dinámico en línea.	171
Diagramas Entidad-Relación del Sistema de Información.	115
Modelo Relacional del Sistema de Información.	121
Cuadro Explicativo de la estructura del Capítulo 2.	52
Cuadro Explicativo de la estructura del Capítulo 3.	100

OBJETIVOS

- General: Desarrollar un sistema de manejo de información que unifique información de proceso y que permita interactuar con herramientas de simulación.
- Particulares:
 - Definir al sistema de información e identificar sus características.
 - Identificar y revisar los estándares y normas que regulen la comunicación de datos entre diversas aplicaciones.
 - Definir la comunicación de datos de los simuladores de proceso involucrados.
 - Identificar los tipos de datos generados por los softwares de simulación.
 - Integrar los datos de proceso a la base de datos.
 - Integrar los datos requeridos y generados por los simuladores de proceso a la base de datos.
 - Evaluar el funcionamiento del sistema de información de acuerdo a las pruebas realizadas.

HIPÓTESIS

- ❖ El sistema de manejo de información almacena datos de proceso y permite la interacción de herramientas de simulación y optimización integrando los datos de entrada y salida de las mismas, con lo que se obtienen resultados mas rápidos y precisos, una toma de decisiones mas certera, mejora en la calidad de los productos y un ahorro sustancial en tiempo, energía y dinero. Además el sistema, al ser abierto, permite la integración de diferentes aplicaciones que complementen a las contenidas en el mismo.

PROBLEMÁTICA

La industria petrolera requiere para sus procesos de explotación y producción de hidrocarburos, la incorporación y desarrollo de estrategias que permitan modernizar su planta productiva, teniendo como premisa la optimización de los procesos que en ella se realizan; para competir en calidad y precio en los mercados internacionales.

El control de los procesos industriales realizados de forma manual incurren en muchos errores que debilitan la cantidad y calidad de los productos, así como en un incremento de los riesgos dentro de la planta productiva; esta falla puede corregirse con la adecuada evaluación, selección, asimilación y aplicación de tecnologías de optimización y automatización aplicables a los diferentes niveles de los procesos productivos.

Actualmente todas las aplicaciones de simulación que utiliza PEMEX Exploración y Producción (PEP) se ejecutan en forma aislada, como resultado se obtienen soluciones parciales. No existe un estándar de comunicación de datos entre herramientas de simulación que permita relacionar todas las aplicaciones y conciliación de datos a través de una plataforma o interfaz estándar.

Específicamente en el campo de optimización de procesos de producción, dentro de PEP existen diferentes problemas que no han permitido explotar todo el potencial de las herramientas de simulación, entre los cuales se encuentran:

-
- Ejecución aislada de las herramientas de simulación y soluciones parciales del proceso.
 - Información de proceso no actualizada.
 - Inexistencia de un estándar de comunicación de datos entre las herramientas de simulación que permita relacionar las aplicaciones existentes.
 - Inexistencia de una base de datos única que contenga la información requerida por las diversas herramientas de simulación.
 - Ingreso de datos a las herramientas de simulación en forma manual.

JUSTIFICACIÓN

Actualmente, PEP requiere de un sistema de información para mejorar los procesos de producción de hidrocarburos, obteniendo resultados más precisos gracias a la integración de aplicaciones (herramientas de simulación y optimización) y al almacenamiento de los datos de dichas herramientas en una misma base de datos. Con esto, se logrará facilitar el proceso de toma de decisiones y se establecerán parámetros bien definidos en la optimización de los procesos de producción.

Es altamente factible y necesario la ejecución de este sistema, en virtud de las diferentes aplicaciones de simulación utilizados por PEP en sus diversos grupos de trabajo y de la inexistencia de una estándar que permita ó facilite la manipulación de datos de manera transparente desde cualquiera de las aplicaciones. Así mismo permitirá a los niveles gerenciales la toma de decisiones en forma oportuna sobre los procesos de producción.

El contar con un sistema de información para los procesos de producción permitirá obtener las ventajas siguientes:

- Mayor certidumbre en los resultados de las simulaciones.
- Generación de información en tiempo real e histórico de las variables de los procesos de producción.
- Apoyo para la toma de decisiones tanto operativas como gerenciales.
- Control de los procesos de producción.
- Generación de base de datos, reportes, estadísticas.
- Información confiable.
- Ahorro en tiempo, energía y recursos financieros.

ALCANCES

- Desarrollar el sistema de información en trabajo conjunto con el usuario y de acuerdo a sus necesidades, para que con ello se mejoren los procesos y el sistema tenga éxito.
- Realizar la comunicación del software de simulación de pozos (PIPESIM), con el software de simulación de procesos (PRO/II) y ambos con la base de datos para que estas aplicaciones se relacionen y los resultado se obtengan con mayor rapidez.
- Realizar una base de datos en Microsoft Access que permita tener acceso a datos de proceso y de las herramientas de simulación Pro/ii y Pipesim tanto de entrada como de salida.
- Mantener en buen funcionamiento al sistema, para que de ahí se puedan realizar nuevos proyectos con base en el mismo, como la integración de otras aplicaciones y la optimización de procesos en línea.

INTRODUCCIÓN

Una gran parte de la energía motora disponible en este planeta proviene del petróleo y sus derivados. Se calcula que sólo en México, país al que la naturaleza dotó de una enorme riqueza petrolífera, entre el 90 y 95% de las necesidades energéticas son satisfechas por este aceite crudo. Pero, ¿qué es el petróleo?, es una sustancia combustible fósil que se encuentra en yacimientos subterráneos. Esta sustancia se compone principalmente de hidrocarburos (compuestos que contienen átomos de carbono e hidrógeno), además de azufre, nitrógeno y oxígeno. Los hidrocarburos se obtienen por medio de la refinación, gracias a la cual la sociedad recibe una amplia gama de productos comerciales como: energéticos (combustibles específicos para transportes, agricultura, industria, etc.), lubricantes, parafinas, asfaltos, materias primas para la industria petrolera básica, entre otros.

El petróleo líquido, como se produce en la cabeza del pozo, generalmente está asociado con gas natural y contaminado con agua y sólidos en suspensión. Del crudo deben eliminarse la mayor parte de esas impurezas antes de enviarse al mercado o a los centros de refinación.

El gas natural se separa del petróleo crudo en las Baterías de Separación, para después aplicar a cada fase, los procesos de tratamiento y acondicionamiento que se requieran para su distribución y venta, todos estos procesos son parte de la producción de hidrocarburos.

Para que la industria petrolera nacional siga siendo líder en el mercado, requiere para sus procesos de explotación y producción de hidrocarburos, la incorporación y desarrollo de estrategias que permitan modernizar su planta productiva, teniendo como premisa la optimización de los procesos que en ella se realizan.

Por ello en el Instituto Mexicano del Petróleo (IMP), por medio del Grupo de Optimización de Instalaciones de Producción (GOIP) busca mejorar la producción de hidrocarburos con ayuda de simuladores de procesos. El GOIP perteneciente a la Gerencia de Tecnología de Producción, desarrolla actividades de asesoría y asistencia técnica especializada para PEMEX Exploración y Producción (PEP), en la optimización y automatización integral de instalaciones de producción terrestres y marinas.

Para realizar la optimización primero se tiene que disponer de información de proceso (de campo *), la cual se complementa con información de escritorio, para luego hacer un análisis a través de distintas herramientas, es decir, se vinculan los datos con el software de simulación.

Actualmente existen algunos problemas en el campo de la optimización de procesos de producción de hidrocarburos que no han permitido explotar todo el potencial de las herramientas de simulación, entre los que se encuentran: la ejecución aislada de los simuladores de proceso, con lo que se obtienen soluciones parciales del proceso; inexistencia de un estándar de comunicación de datos entre las herramientas de simulación que permita relacionar las aplicaciones existentes; inexistencia de una base de datos única que contenga la información requerida y generada por las diversas herramientas de simulación; y el ingreso de datos al software en forma manual.

La realización de este trabajo abarca el desarrollo de un sistema de información para la simulación y optimización de procesos de producción de hidrocarburos, con el que se tratará de dar solución a todos los problemas existentes de la mejor manera posible con el fin de contribuir con el IMP en la optimización de estos procesos y lograr que Petróleos Mexicanos siga compitiendo en calidad y precio en los mercados internacionales.

La primera parte del trabajo está conformada por los dos primeros capítulos. El primero que corresponde al Marco Teórico, en donde se tratan temas como Datos, Información, Sistemas de Información, Bases de Datos, Producción de hidrocarburos, entre otros. Todo esto, con el fin de entender cómo se compone un sistema de

* La información de campo es la que se obtiene directamente de las instalaciones petroleras.

información y hacer un análisis para realizar todas las fases que llevan a la creación de un sistema eficiente, utilizando Microsoft Access.

En el segundo capítulo abarca dos partes, el Marco Teórico específico y la Fase Preliminar del Proyecto. En la primera parte se presentan las diferentes herramientas para simulación y optimización de procesos de producción de hidrocarburos, se comprende la función de las mismas y se conocen las que existen en el mercado en cuanto a producción de hidrocarburos se refiere. Además se identifican los estándares y normas que regulan la comunicación de datos entre diversas aplicaciones (específicamente de la empresa Microsoft). En la segunda parte se identifican los tipos de datos (entrada y salida) que arrojan las herramientas de simulación utilizadas en el IMP, para poder lograr una comunicación entre ellos, además de que se identifican y sugieren las alternativas de dicha comunicación de acuerdo al análisis de datos realizado.

En el capítulo tres, se desarrolla el sistema de información, tomando como base las fases del ciclo de vida del desarrollo de un sistema. Se inicia con el análisis de la empresa, para la cual se realiza el sistema, posteriormente se hace la configuración de hardware y software a utilizar. Se identifica la información, se diseña el Modelo de Datos, y se determinan las características de la Base de Datos que se necesita, para así integrar toda la información a dicha base. Además se presentan las interfaces de usuario y diversos diagramas para el entendimiento de la organización de los datos y el flujo de la información. Se termina con la puesta en marcha del sistema, realizando pruebas.

En el último capítulo se evalúa el sistema en funcionamiento y se analizan los resultados. Además se mencionan los proyectos que pueden surgir a partir de este trabajo de tesis, como la optimización en línea, un sistema de control integral, entre otras.

Al término de este trabajo se espera haber colaborado con una importante aportación para que la optimización de procesos de producción de hidrocarburos, ayude a mejorar la planta productiva en Petróleos Mexicanos y esta empresa mexicana sea el ejemplo para otras petroquímicas que quieran integrarse a esta gran competencia internacional.

CAPÍTULO 1

**MARCO TEÓRICO
GENERAL**

CAPÍTULO 1. MARCO TEÓRICO GENERAL

El procesamiento de los datos es esencial para la investigación, planificación y toma de decisiones de los gobiernos, los negocios y la educación, y aun para las actividades de entretenimiento. El pronóstico del tiempo, por ejemplo, el cual puede determinar los planes para el fin de semana o las vacaciones, se basa en el procesamiento de datos y comunicación de información precisa. En nuestra sociedad es vital para una organización o empresa proporcionar información correcta, oportuna, completa y coherente para apoyar la toma de decisiones y otras actividades gerenciales. Como resultado del crecimiento económico y avances tecnológicos, muchas organizaciones crecieron tanto en el tamaño como en la sofisticación de sus funciones administrativas.

La invención de las computadoras revolucionó los métodos tradicionales del procesamiento de los datos. A principio de los años 60, varias firmas comerciales empezaron a computarizar sus sistemas de información; los datos se guardaban en medios electrónicos en lugar de guardarlos en papel, y se usaban lenguajes de alto nivel para recuperar y manejar los datos en aditamentos de almacenamiento.

En los sistemas de información convencionales, las aplicaciones individuales se desarrollaban independientemente, y cada programa de aplicación procesaba sus propios archivos privados. Como resultado, algunas actividades se duplicaban y la información redundante se almacenaba para usarla en distintas operaciones. Al final de los años 60 surgió el sistema de bases de datos para superar los problemas asociados con los sistemas de información tradicionales (no hay que olvidar que las bases de datos constituyen una parte integrante y fundamental del sistema de información y tienen su razón de ser en la misma existencia de éste). Los archivos individuales se integraban en una sola base de datos para ser compartidos por todos los usuarios de una empresa. En vista de la centralización de los datos por medio de un sistema de base de datos, los requerimientos de todos los usuarios se podían coordinar de una manera efectiva para alcanzar la mejor utilidad general para la organización.

Aunado a esto surgen nuevas profesiones orientadas al tratamiento de la información con ayuda de dispositivos electrónicos como son: analistas de sistemas, administradores de bases de datos, programadores, entre otras. Esto hizo que las empresas empezaran a considerar con mucho cuidado las capacidades de sus sistemas de información cuando decidían ingresar o no en nuevos mercados o cuando planeaban la respuesta que darían a la competencia, ya que sin ayuda automatizada, las organizaciones tendrían que hacer un alto ante el volumen de trabajo que abrumaría a los administradores y empleados, por ello día con día crece la demanda por parte de las empresas en actualizar sus sistemas de información y hacerlos más eficientes.

A pesar de ello, es frecuente encontrar en las distintas organizaciones sistemas desintegrados o hasta la inexistencia de los mismos; este hecho se debe, en muchos casos, a una pobre planificación de los sistemas de información (debido a veces a la exigencia del usuario de pronto resultados), en otros, a la historia de adquisiciones y fusiones entre organizaciones y a la creación de negocios de la nada y sin conocimiento sobre cómo hacerlo. Sean cuales fueren los motivos de esta situación, el resultado es que se dispone de datos relacionados semánticamente entre sí, pero contenidos en sistemas que no permiten su tratamiento y estudio de una forma global, o de exceso de datos que no están almacenados.

La misión del proceso de integración de información en una organización consiste en identificar, analizar, almacenar y mantener las relaciones existentes entre sus datos, para que de esta forma puedan ser utilizados en mejorar las distintas actividades que se llevan a cabo en la organización (por ejemplo: toma de decisiones, procesos de producción, estimación de costos, etc.).

Para el caso de estudio de esta tesis, el proceso que es preciso mejorar y optimizar dentro del Instituto Mexicano del Petróleo (IMP), es el de producción de hidrocarburos, el cual permitirá incrementar la productividad de los procesos y la calidad de los productos, además de elevar los niveles de seguridad operacional y la protección al medio ambiente.

Para entender mejor el "Sistema de manejo de Información para la Simulación y Optimización de los Procesos de Producción de Hidrocarburos" desarrollado en este trabajo de tesis, es preciso introducir al lector, en este primer capítulo, analizando algunos aspectos relativos a conceptos y características de datos, información, sistema de información, bases de datos, simulación y optimización, entre otros; a fin de que en los capítulos siguientes no haya problema en entender la importancia de la aplicación realizada y su impacto en algunos proyectos de la organización (IMP).

1.1. Datos e información.

1.1.1. Datos

La palabra *dato* proviene del latín *datum*, forma del verbo *dare*, "dar", que significa "lo que es dado". *

Aunque el verbo *dar* existe desde los mismos orígenes del castellano, la forma culta *dato* no se registra hasta 1786, cuando Esteban de Terreros publica su *Diccionario castellano con las voces de ciencias y artes*, que efectivamente supuso el acta de nacimiento de muchas palabras técnicas.

En inglés se usaba el latinismo directo *datum*, plural *data*, y a finales del XIX ya tenía el sentido (como en las otras lenguas cultas) de "hechos numéricos reunidos para referencia o información", como recoge el *Oxford English Dictionary*. Pero con el desarrollo de herramientas automáticas, en la década de 1940, la palabra se especializa en el sentido de "Representación de una información de manera adecuada para su tratamiento por una computadora" (definición actual de la Academia).

Se refiere a "dato" cuando se toma a la información como materia. Los datos pueden ser numéricos o no numéricos (por ejemplo alfabéticos, o simbólicos). Es mediante la utilización de éstos como podemos representar conceptos, hechos o instrucciones. ²

Los datos son hechos brutos, no sumarizados y no analizados, considerados de forma aislada, imágenes o sonidos que pueden o no ser pertinentes o útiles para una tarea en particular. Son números, letras, símbolos, un "algo" aislado, sin tratar, sin procesar, sin significado. Los datos no son inmediatamente utilizables sin la relación con otros datos. En la Figura 1 podemos visualizar algunos ejemplos de datos.

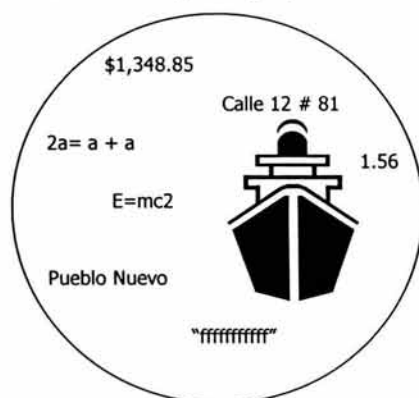


Figura 1. Diferentes datos, que pueden ser simbólicos, icónicos, lingüísticos, lógicos, matemáticos, numéricos, entre otros.

* Referencia 27. Tsai, Alice Y. H. México, 1990, 607 pp.

² Referencia Sitios de Internet 1. <http://www.dcc.uchile.cl/~rbaeza/libro/liendo.pdf>

Los datos son valores para diferentes atributos directamente relacionados al bien o entidad que es medido. Esto incluye descripciones, tipos, y cantidades de lo que está siendo medido, con frecuencia en el pasado, presente y futuro pronosticado.

Los datos pueden, y deberían interpretarse como la materia prima utilizada para producir información. En consecuencia, consideramos que los datos deben constituir uno de los pilares fundamentales de un sistema de información.

1.1.2. Información

La palabra información está compuesta precisamente por dos vocablos de origen latino que son: IN, que significa en; y FORMARE que significa dar forma, poner en forma. La conjunción más simple de estos dos vocablos forman el significado "dar forma en" algún lugar, en este caso en la mente humana (algunos autores mencionan que el término "informar" proviene del latín "*informare*" que quiere decir: "formar dentro"), que es el único elemento que puede formar en su interior la representación de los objetos o fenómenos cuando los percibe directamente; o bien, cuando los percibe por medio de los signos gráficos, iconográficos y fonéticos que los representan. Con este sencillo análisis se puede deducir que la palabra información, como acción, es el proceso por medio del cual se forman representaciones en la mente humana por medio de la información que percibe del medio que la rodea, por un lado; y por otro, gracias a la información como producto terminado, la mente humana puede adquirir conciencia (conocimiento) de su propia existencia y de los objetos y fenómenos que la envuelven.³

Cuando se habla formalmente de "información" se refiere al resultado de lograr, a partir del material con potencial informativo, relacionarlo o estructurarlo de forma tal que adquiera sentido. No es el simple material sino su "sentido". Cada acto informativo es único e individual. Para reconocer que se dio el acto informativo, el dato debe haber inducido una modificación del bagaje cognoscitivo del individuo. Los datos: "El hombre dejó su huella en la Luna", o "las ballenas se están extinguiendo" o "la población mundial está creciendo en forma alarmante" o "el hambre mata", no obstante sus enormes connotaciones no tienen valor informativo alguno para quien ya estaba enterado.⁴

La información es un conjunto de datos que se presentan de forma que es inteligible al receptor. Son datos que han sido organizados (en un sentido significativo, coherente), sistematizados y presentados, que tienen significado y alteran el entendimiento del receptor, datos cuya forma y contenido son apropiados para un uso particular, útiles para un propósito.

La información es una entidad tangible o intangible que reduce la incertidumbre acerca de un evento o estado. Información es cuando los datos son analizados y presentados en determinada forma, la cual apoya el proceso de toma de decisiones, esto quiere decir, que los datos correctos son reunidos, que las compilaciones repetitivas o tangenciales son evitadas y que las preguntas originalmente planteadas durante la ejecución de medidas están siendo respondidas.

La información está compuesta de números, estadísticas, descubrimientos científicos, tendencias de ventas, hechos, historias y/o imágenes que dicen algo que no se sabía, que confirman algo que se sospechaba, que contribuyen al entendimiento, que mueven a la acción.

1.1.3. Análisis de la diferencia entre datos e información

Se debe comprender en qué se diferencia el conocimiento de los datos y de la información. Los datos son una colección de hechos, entidades independientes sin evaluar. La información son los datos que han sido interpretados para su uso. La información es una abstracción de datos.

³ Referencia Sitios de Internet 2. <http://www.ambac.org.mx/publicaciones/V3N4art1.html>

⁴ Referencia Sitios de Internet 1. www.dcc.uchile.cl/~rbaeza/libro/liendo.pdf

Los datos son números en bruto y sin alterar, son hechos, estadísticas, es el precio de un producto, una parte de un número, un porcentaje de interés, una cantidad. Procesando estos datos de acuerdo a las reglas específicas se produce información. El precio de un producto multiplicado por la cantidad es el costo de una orden. Sin los datos en bruto el proceso de información no puede calcular el costo de la orden.

Los datos son usados para crear información. Por ejemplo, un pronóstico del tiempo puede reunir medidas de la presión del aire y temperatura en un período de tiempo determinado y entonces crear una gráfica de tendencia para mostrar cómo éstas varían en el tiempo. Las medidas de presión de aire y temperatura serían tomadas como datos, mientras que la gráfica de tendencia sería la información. Entonces, basado en estas tendencias el pronosticador puede predecir el futuro. Esto genera nuevos datos. Con estos datos el pronosticador puede producir un mapa del tiempo, el cual convierte las temperaturas en colores, y los datos de presión a isobaras. Éste ha convertido datos en información. La información debe tener valor, o en caso contrario sería un dato.⁵

La información es algo que incrementa nuestro conocimiento. Por ejemplo, la declaración: Ámbar es una estudiante, nos dice algo acerca de Ámbar (ella es una estudiante). Esto expande nuestro conocimiento (acerca de Ámbar). Si nosotros quisiéramos conservar datos acerca de Ámbar, nosotros podríamos almacenar: García, Ámbar, 264 40 13 09, La Garita, 11, los cuales nos dicen esencialmente nada acerca de Ámbar. Sin embargo, mirando los datos de arriba, nosotros obtenemos el sentimiento de que hay un patrón de cómo están estos almacenados. Esto nos da una definición básica para datos: los datos son una colección de símbolos no aleatorios.

Un dato es un conjunto discreto, de factores objetivos sobre un hecho real. Dentro de un contexto empresarial, el concepto de dato es definido como un registro de transacciones. Un dato no dice nada sobre el porqué de las cosas, y por sí mismo tiene poca o ninguna relevancia o propósito. Desde un punto de vista cuantitativo, las empresas evalúan la gestión de los datos en términos de costo, velocidad y capacidad. Todas las organizaciones necesitan datos y algunos sectores son totalmente dependientes de ellos. Bancos, compañías de seguros, agencias gubernamentales y la seguridad social son ejemplos obvios. En este tipo de organizaciones la buena gestión de los datos es esencial para su funcionamiento, ya que operan con millones de transacciones diarias. Pero en general, para la mayoría de las empresas tener muchos datos no siempre es bueno. Las organizaciones almacenan datos sin sentido. Realmente esta actitud no tiene sentido por dos razones. La primera es que demasiados datos hacen más complicado identificar aquellos que son relevantes. Segundo, y todavía más importante, es que los datos no tienen significado en sí mismos. Los datos describen únicamente una parte de lo que pasa en la realidad y no proporcionan juicios de valor o interpretaciones, y por lo tanto no son orientativos para la acción. La toma de decisiones se basará en datos, pero estos nunca dirán lo que hacer. Los datos no dicen nada acerca de lo que es importante o no. A pesar de todo, los datos son importantes para las organizaciones, ya que son la base para la creación de información.⁶

Pero, lo que para una persona es información, para otra puede ser un simple dato. Por ejemplo, pensemos en un informe que contenga una lista con todas las cuentas de clientes. Si los clientes llaman para saber cómo está su saldo actual, un administrativo puede responder a sus preguntas con una simple ojeada al informe. Para el administrativo, este informe es información. Pero supongamos que un directivo desea saber cuál es la cantidad total, en líquido, de cuentas impagadas. El directivo tendría que identificar las cuentas impagadas en el informe y sumar sus saldos. Así, el informe (tal como está) representa solo datos para el directivo. Por otra parte un informe que contuviera una lista de las cuentas impagadas y sus saldos sumados sería información para el directivo.

La información se describe como un mensaje, normalmente bajo la forma de un documento o algún tipo de comunicación audible o visible. Como cualquier mensaje, tiene un emisor y un receptor. La información es capaz de cambiar la forma en que el receptor percibe algo, es capaz de impactar sobre sus juicios de valor y comportamientos. Tiene que informar; son datos que marcan la diferencia. La palabra "informar" significa

⁵ Referencia Sitios de Internet 4. http://www.palgrave.com/skills4study/html/subject_areas/science/science-technology.htm

⁶ Referencia Sitios de Internet 3. http://www.gestiondelconocimiento.com/conceptos_diferenciaentredato.htm

originalmente "dar forma a " y la información es capaz de formar a la persona que la consigue, proporcionando ciertas diferencias en su interior o exterior. Por lo tanto, estrictamente hablando, es el receptor, y no el emisor, el que decide si el mensaje que ha recibido es realmente información, es decir, si realmente le informa. Un informe lleno de tablas inconexas, puede ser considerado información por el que lo escribe, pero a su vez puede ser juzgado como "ruido" por el que lo recibe.

A diferencia de los datos, la información tiene significado (relevancia y propósito). No sólo puede formar potencialmente al que la recibe, sino que esta organizada para algún propósito. Los datos se convierten en información cuando su creador les añade significado. Transformamos datos en información añadiéndoles valor en varios sentidos. Hay varios métodos:

- Contextualizando: sabemos para qué propósito se generaron los datos.
- Categorizando: conocemos las unidades de análisis de los componentes principales de los datos.
- Calculando: los datos pueden haber sido analizados matemática o estadísticamente.
- Corrigiendo: los errores se han eliminado de los datos.
- Condensando: los datos se han podido resumir de forma más concisa.

1.1.4. Datos e información desde un punto de vista informático

Para los informáticos hay una diferencia entre datos e información. Dato es la representación de algún hecho, concepto o entidad real, es la materia prima de la información. Información es el resultado del procesamiento de los datos.

Un sistema de procesamiento de la información involucra tres componentes: datos de entrada, procesador y datos de salida, como se muestra en la Figura 2. El procesador transforma los datos de entrada en datos de salida ejecutando instrucciones precisas y detalladas que se denominan programas.

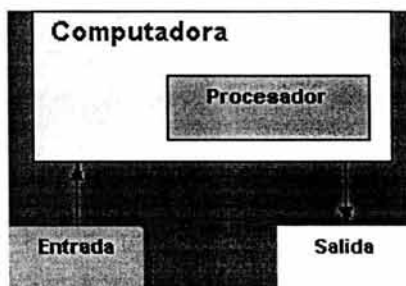


Figura 2. Representación del procesamiento de datos en una computadora.

El dato es simplemente una serie de caracteres, como un nombre, una dirección, o una cifra, escrita en forma análoga o digital, ya sea en un pedazo de papel, o en una unidad de almacenamiento magnético. La información son datos debidamente seleccionados, procesados y analizados, que pueden utilizarse para tomar decisiones, cuando se ha creado un Sistema de Información.

Es bueno recordar que un sistema electrónico está constituido por equipos (hardware) y programas (software). Son estos últimos, los que manipulan los datos para convertirlos en información. Las computadoras no son muy inteligentes, pero son muy rápidas para manipular los datos.

Las computadoras no pueden almacenar información. La información es algo que tiene un significado. Como las computadoras son máquinas, ellas no pueden entender el significado de una oración como "Erin está tocando la guitarra". En vez de esto, las computadoras, sólo almacenan secuencias de letras y números. Ellas no saben lo que estas oraciones significan. Las secuencias que la computadora almacena son llamadas datos. De hecho,

todo lo que hay dentro de una computadora son datos. Unos son datos que sirven para manipular otros datos (y constituyen los programas) y otros son datos inertes (aunque nos puedan resultar muy interesantes).

En una computadora digital los datos deben ser organizados antes de que estos puedan ser procesados efectivamente. Por esta razón, los ítems de los datos son colocados en los siguientes grupos:

- **Bit.** Los datos son representados por estados del circuito electrónico de la computadora encendido y apagado (ver Figura 3). Los símbolos que representa el encendido y apagado son los dígitos binarios 0 y 1. Cada 0 o 1 es llamado un bit. El bit es la unidad más pequeña de los datos que una computadora puede procesar.

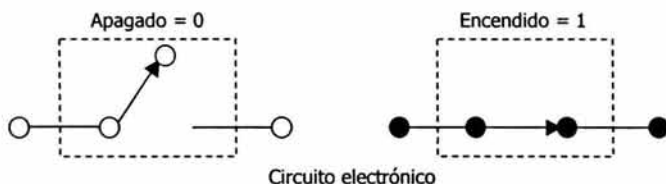


Figura 3. Representación de estados de un circuito electrónico: encendido y apagado.

- **Carácter.** Obviamente hay más de dos letras y números, pero la computadora sólo reconoce 0 y 1. Por lo tanto, las combinaciones de bits (0's y 1's) son usadas para representar caracteres – letras, dígitos, y símbolos especiales como %, #, o \$, como se muestra en la Figura 4. En el nombre Juan Cruz, los caracteres son las letras, J, u, a, n, C, r, u, z, y un espacio. Una combinación fija de bits adyacentes que representan un carácter es llamado byte. En nuestro ejemplo, ocho bits son usados para representar un carácter (ver Figura 5).

0	1	1	0	1	0	0	1
---	---	---	---	---	---	---	---

Figura 4. Representación de un byte (formado por 8 bits).

J	u	a	n		C	r	u	z
---	---	---	---	--	---	---	---	---

Figura 5. Cada carácter está formado por una combinación de 8 bits.

- **Campo.** Una universidad conserva datos específicos acerca de sus estudiantes, así como la dirección de su casa, su número de cuenta, carrera, cursos, y más. Cada una de estas categorías es llamada campo. Un campo es una colección de caracteres relacionados que expresan una unidad de información. Nótese que el tamaño de un campo depende de la información colocada en el campo. Por ejemplo, el campo para nombres necesitaría ser más grande que el campo para el número de cuenta (ver Figura 6).

Campo No. de Cuenta	9	5	0	9	1	6	5	-	5													
Campo Nombres	A	I	m	a	R	o	s	a	T	o	l	e	d	o	T	r	i	n	i	d	a	d

Figura 6. Representación lógica de los caracteres almacenados en un campo.

- **Registro.** Una colección de campos que se relacionan a una sola unidad es un registro. Un registro de un estudiante puede consistir en campos para el nombre del estudiante, número de cuenta, dirección, y carrera.
- **Archivo.** Una colección de registros del mismo tipo se llama archivo. La Figura 7 da el punto de vista del programador de lo que es un archivo. Cada columna representa un campo o un atributo, y cada renglón representa un registro consistente en cierto número de atributos que describen una entidad.

E003	A. RAMIREZ	CALLE LILAS # 502	CONTADOR
E002	T. FRANCO	AV. CORREOS # 103	GERENTE
E001	J. LOPEZ	EJE CENTRAL # 25	PROGRAMADOR
EMP-NO	EMP-NOMBRE	EMP-DIRECCION	PUESTO

Figura 7. Visión del programador de un archivo del Departamento de Personal de una Compañía ficticia. ⁷

En la misma figura, un registro en el archivo EMPLEADO describe al empleado por medio de ciertos atributos, esto es, EMP-NO, EMP-NOMBRE, EMP-DIRECCIÓN y PUESTO dispuestos como campos de datos.

1.1.5. Fuentes de información

Las fuentes de información pueden ser:

- Internas, que son las generadas por la propia organización.
- Externas, que son las generadas desde fuera de ella.
- Primaria, es la información que se obtiene por medio de una encuesta a realizar, es aquella información que se obtiene por primera vez para un tema específico.
- Secundaria, es aquella información que ya fue obtenida y se encuentra almacenada en algún sitio accesible.

1.1.6. Cualidades de la información ⁸

La *explosión de la información* - como se llama a veces a este enorme crecimiento de las necesidades de la información y a la mayor disponibilidad de este recurso - puede conducir, si no se ponen los medios para evitarlo, a una *polución informativa*. Fenómeno análogo al de la contaminación del aire, en el que la información, al perder sus cualidades, no puede cumplir sus objetivos, llegando incluso a ser más nociva que beneficiosa para sus destinatarios. Para evitar el peligro de esto, se debe exigir a la información un conjunto de cualidades que mantengan su valor comunicativo, ya que para hacer honor a su nombre debe ser capaz de *informar*, es decir, de aportar un conocimiento.

La información que es obsoleta, inexacta o difícil de entender no sería muy significativa, útil o valiosa para los usuarios finales. Las personas quieren información de alta calidad, es decir, productos de información cuyas características, atributos o cualidades ayuden a que ésta sea valiosa para ellas. Es de utilidad considerar la información como un elemento que posee las tres dimensiones de tiempo, contenido y forma. Esto se puede ver en la Figura 8.

⁷ Referencia 27. Tsai, Alice Y. H. México, 1990, 607 pp.

⁸ Referencia 18. O'Brien, James A. Colombia, 2001, 700 pp.



Figura 8. Resumen de los atributos de la calidad de la información. Aquí se describen los atributos que deben estar presentes en los productos de información de alta calidad.

Dimensión tiempo:

- **Oportunidad:** Se refiere al tiempo transcurrido desde el momento en que se produjo el hecho que originó el dato hasta el momento en el que la información se pone a disposición del usuario. La información debe suministrarse en el momento que sea necesaria. El valor de la información va disminuyendo con el transcurso del tiempo¹, e incluso, después de cierto momento, puede llegar a perder totalmente la relevancia que pudiera tener.
- **Actualidad:** La información debe ser reciente al momento de suministrarse.
- **Frecuencia:** La información debe suministrarse con la frecuencia que sea necesaria. La frecuencia con que se transmite o recibe información repercute en su valor. La información que aparece con excesiva frecuencia tiende a producir interferencia, ruido o distracción.
- **Período.** La información puede proporcionarse sobre períodos pasados, presentes y futuros.

Dimensión contenido:

- **Exactitud:** La información debe estar libre de errores (tanto de transmisión como de cálculo). Mide la aproximación de un número a lo que verdaderamente debería ser.
- **Pertinencia:** La información debe estar relacionada con las necesidades de información de un destinatario específico para una determinada situación.
- **Integridad:** Debe suministrarse toda la información que sea necesaria. Toda la información contenida en el sistema debe ser *coherente* en sí misma, además de consistente con las reglas semánticas propias del mundo real al que ha de representar lo más fielmente posible.
- **Brevidad:** Debe proporcionarse sólo la información que se necesite.
- **Alcance:** La información puede tener un alcance amplio o estrecho, o un enfoque interno o externo. Es el lugar de acción de la información, es el lugar por donde va a pasar esa información y quienes la van a utilizar. Los usuarios deben tener autoridad para determinar cuál es la información útil para determinada situación y en la forma en que ha de ser comunicada para su total aprovechamiento.
- **Desempeño:** La información puede revelar el desempeño, al medir las actividades logradas, el progreso alcanzado o los recursos acumulados.

Dimensión forma:

- **Claridad:** La información debe suministrarse en un formato que sea fácil de entender.

¹ En investigaciones históricas, por el contrario, la información gana con el transcurso del tiempo.

- Detalle: La información puede proporcionarse en un formato detallado o de resumen.
- Orden: La información puede ordenarse en una secuencia predeterminada.
- Presentación: La información puede presentarse en forma narrativa, numérica, gráfica, u otras formas.
- Medios: La información puede proporcionarse en la forma de documentos de papel impresos, presentaciones de video u otros medios.

Cuando se están haciendo los estudios que llevarán a la implantación de un sistema de información, es preciso tener muy en cuenta todos estos requisitos de información buscando el punto de equilibrio que permita alcanzar los objetivos del sistema.

1.1.7. Importancia de la información en las organizaciones.

La información en la empresa tiene tres características: puede ser utilizada simultáneamente, no se gasta con el uso, y sus trozos pueden ser combinados para generar más información. Los recursos que realmente cuentan son los que denomina invisibles, como por ejemplo, la capacidad para aprender, de asimilar y crear tecnología. Estos activos se nutren gracias a los flujos de información básicos: el de la información que entra en la empresa procedente de su entorno (la información ambiental), la información que fluye por la empresa (información interna), y la información que la empresa proyecta hacia el exterior (información corporativa).

La información externa que le llega al empleado y que le sirve para realizar su trabajo debe de ser de calidad. Por ello, la calidad de esta información tiene que asegurarse en el centro donde es recibida y procesada, en el centro de información de la empresa, que debe ser también el centro de operaciones del sistema de información y de aseguramiento de la calidad informativa. No hay que olvidar que un trabajo elaborado con calidad genera una información de calidad.

Las empresas dedican una parte importante de su tiempo y de sus recursos económicos y humanos a la obtención, proceso, aplicación y proyección de información. Por esta razón, la información interna juega un papel decisivo en la empresa y se convierte en su principal patrimonio. Debe ser clara, precisa y que se adapte a la formación y perfil de las personas a la que va dirigida. Tiene que ser rápida y estar disponible en el momento que se la necesite, y de una manera completa y armonizada con otras informaciones. Debe ser una información fácilmente accesible y con un coste de obtención menor que la utilidad que ofrece.

La información se obtiene, se procesa y se emplea de forma parecida a como se explotan los recursos tradicionales de las empresas (capital, energía, personas, etc.). Pero no se adquiere un recurso si no es con la idea de utilizarlo, no tiene sentido adquirir información si no es para aplicarla. Si no se considera la información como un recurso valioso y costoso, nadie se preocupará de manejarla adecuadamente, nadie se considerará responsable de su gestión.

Los empleados que forman parte del circuito informativo de su empresa resultan más productivos para la organización. Se debe informar a los trabajadores de los resultados de su empresa (bien la cotización en bolsa, costes empresariales de producción o de oferta de un servicio). Compartir información no es enviar la información, sino generarla. Se trata de que la información fluya y de que los empleados no tengan que esperar al informe periódico de las ganancias, sino que éste se publique mensualmente en su boletín o en su intranet.

Compartir información es lo que más cuesta en esta nueva cultura. Tan negativo es para la organización tener un empleado que realiza de manera rutinaria su trabajo, como tener un empleado que piense que nada le queda por aprender. En los dos casos, la información no fluye y sus tareas no crean información que interesen a otros departamentos, bien porque no da importancia a la información que le llega y no reconocerá su valor real y no la compartirá, o bien porque tenderá a acaparar la información, le sea de utilidad o no.

1.2. Sistemas de información.

Un sistema de información (para la mayoría de la gente) implica el uso de algún tipo de tecnología basada en computadora, y esa característica es ciertamente relevante para el tipo de sistemas que se verán más adelante. Sin embargo, los sistemas de información, en el sentido de un conjunto de reglas, procedimientos, facilidades, etc., que nos proporcionan información sobre lo que sea en lo que estemos interesados, han estado alrededor por siglos. El registro de productos de la cosecha en tablas de arcilla por las civilizaciones antiguas es un sistema de información. Las tres operaciones básicas de un sistema de información son: entrada, registro y recuperación, y el sistema de información de tabla de arcilla ciertamente exhibe todas estas características. Lo que ha cambiado es la tecnología, y por la magnitud del cambio, la utilidad, distribución y accesibilidad a los sistemas han cambiado radicalmente. Puede ser discutido que la mayoría de estos cambios ha ocurrido realmente en los últimos veinte años, más o menos. La tecnología continúa cambiando en una proporción cada vez mayor. Es esta tecnología cambiante lo que requiere que nos enfoquemos en la forma en que los sistemas están integrados, en lugar de estar principalmente conscientes con la tecnología que soportan los mismos. Por analogía, si nosotros fuéramos responsables de diseño y nuevos ambientes vivientes, nosotros estaríamos conscientes (entre otras cosas) de los modos de transporte y como estos son usados por la gente – nosotros dejaríamos el desarrollo de nuevas tecnologías a la ingeniería automotriz y gente por el estilo. Como usuarios de información, nosotros estamos al tanto de los tipos y características de los sistemas de información, en lugar de la tecnología de fondo. Sin embargo, es verdad también que una apreciación de la tecnología que sustenta los sistemas de información será útil para conocer como usarlos bien.

El objetivo principal de un sistema para el procesamiento de información es el de reunir, procesar, almacenar y presentar información a los niveles operativos, administrativos y al medio ambiente de la empresa. Para ello el sistema de información utiliza métodos de procesamiento de datos electrónicos, mecánicos o manuales, junto con personal y procedimientos de operación para después distribuir la información en todos los niveles de la organización. Desde que las computadoras son capaces de almacenar grandes cantidades de datos en una pequeña área, los sistemas de información basados en computadora son más útiles y populares. A continuación se habla un poco sobre el desarrollo de estos sistemas.

La evolución de los sistemas de información ha ido muy relacionada con el desarrollo de la tecnología de computadoras*. Durante la existencia de la primera y segunda generación de computadoras se desarrollaron los primeros sistemas de información a nivel operativo, cuyas funciones principales eran la reunión, tabulación y almacenamiento de grandes volúmenes de datos. Estos sistemas fueron llamados *sistemas batch*. La tercera generación de computadoras permitió la implementación de los sistemas de información para la administración cuyo énfasis se centraba en hacer análisis más extensivos de los datos para presentarlos a la gerencia y colaborar así en la toma de decisiones. Con la difusión de los sistemas en línea en tiempo real y de las computadoras personales de la cuarta generación, se empezaron a desarrollar sistemas más flexibles y fáciles de usar que permitieron a los gerentes a realizar ellos mismos sus labores de análisis de datos con el fin de generar información para la toma de decisiones de una manera rápida, ágil y con una interacción directa gerente - máquina. Estos sistemas de información conocidos como sistemas para el soporte de decisiones son objeto de mucho estudio e investigación. Con el avance vertiginoso de la tecnología y las comunicaciones, en la actualidad las empresas cuentan con herramientas útiles para su administración y creación de estrategias en un mercado altamente competitivo, donde la información se ha convertido en otro factor económico valioso.

1.2.1. Conceptos.

Sistema.

Nuestra sociedad está rodeada de sistemas. Por ejemplo, cualquier persona experimenta sensaciones físicas gracias a un complejo sistema nervioso formado por el cerebro, la médula espinal, los nervios y las células

* La tecnología de computadoras es conocida actualmente como tecnologías de información T.I. que es la integración de hardware, software, equipo o sistemas de comunicaciones, y otras tecnologías que apoyan las operaciones de negocio, mejoran la productividad, y ayudan a los administradores a tomar decisiones.

sensoriales especializadas que se encuentran debajo de la piel; estos elementos funcionan en conjunto para hacer que el sujeto experimente sensaciones de frío, calor, comezón, etc. Las personas se comunican con el lenguaje, que es un sistema muy desarrollado formado por palabras y símbolos que tienen significado para el que habla y para quienes lo escuchan. Asimismo, las personas viven en un sistema económico en el que se intercambian bienes y servicios por otros de valor comparable y en el que, al menos en teoría, los participantes obtienen un beneficio en el intercambio.⁹

Una organización es un sistema. Sus componentes – mercadotecnia, manufactura, ventas, investigación, contabilidad y personal – trabajan juntos para crear utilidades que beneficien tanto a los empleados como a los accionistas de la compañía. Cada uno de estos componentes es a su vez un sistema.

El concepto de sistema se aplica a los fenómenos más diversos y a veces sin demasiado rigor. Se trata de una noción difícil de precisar, posiblemente debido a la dosis de relatividad que este concepto lleva consigo.

El diccionario de la Real Academia Española define, en una segunda acepción, el vocablo *sistema* como "Conjunto de cosas que ordenadamente relacionadas entre sí contribuyen a determinado objeto".¹⁰

Para los fines de este trabajo se puede admitir como válida esta definición de la Real Academia, con las siguientes precisiones:

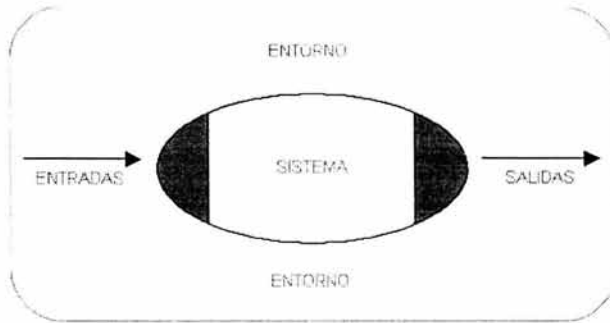
- El término "cosa", puede abarcar objetos físicos, actividades, formas de energía, seres vivientes, entes inanimados, conceptos, ideas, símbolos matemáticos, etc., sin que se exija que todos pertenezcan a una misma clase. Se puede llamar *elementos* a las cosas que integran el sistema.
- Los elementos tienen que estar relacionados entre sí con un *orden*, determinado por unas reglas que gozan de cierta estabilidad. Esta idea de relación ajustada a una normativa es fundamental.
- El sistema típico es *finalista*, es decir, los elementos están relacionados para contribuir a un determinado objetivo. Así, los sistemas debidos a los hombres como los sistemas de información, son creados para el cumplimiento de unos fines; razón por la que algunos autores se refieren a la dinamicidad de los sistemas, considerando que éstos se mueven, en sentido real o figurado, hacia la consecución de un objetivo global.
- La noción de sistema es *relativa*, ya que, a excepción del universo, en la parte más superior de la jerarquía, cualquier sistema es siempre un subsistema de otro sistema más amplio que lo engloba. Así, por ejemplo, la base de datos se puede considerar como un subsistema del sistema de información y éste, a su vez, es un subsistema de la organización.
- En el enfoque sistémico, el todo, es decir, el sistema, es más que la simple suma o agregación de las partes componentes: porque, en general, su objetivo es distinto y presenta nuevas propiedades o características que no son explicables a partir de las características de sus elementos considerados de forma aislada.

Dentro de este concepto de sistema, el sistema de información, como su nombre lo indica, será un sistema, y asimismo la base de datos es también un sistema.

Los sistemas están natural o artificialmente limitados, llamándose a todo lo que está situado fuera de sus límites el medio ambiente o entorno del sistema. De este entorno toma el sistema los elementos o materias primas que constituyen las entradas y al entorno se vierten los productos elaborados, que son las salidas. Véase Figura 9.

⁹ Referencia 23. Senn, James A. México, 1994, 942 pp.

¹⁰ Referencia 8. De Miguel Castaño, Adoración; Piattini Velthuis Mario. España, 1999, 515 pp.

Figura 9. El sistema y su entorno. ¹¹

Los conceptos de sistema y entorno se pueden comprender mejor aplicándolos a un caso real, por ejemplo: la empresa, la cual es un sistema organizacional y se rodea del entorno empresarial. Esto se puede observar en la Figura 10.

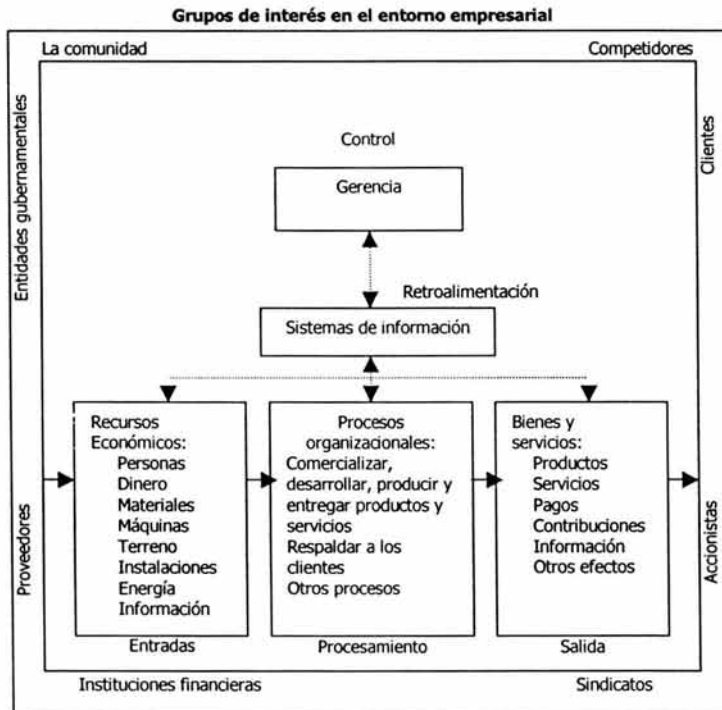


Figura 10. Una empresa es un sistema organizacional donde los recursos económicos (entrada) se transforma mediante diversos procesos organizacionales (procesamiento) en bienes y servicios (salida). Los sistemas de información proporcionan información (retroalimentación) sobre las operaciones del sistema a la gerencia para la dirección y el mantenimiento del sistema (control), intercambian entradas y salidas con su entorno. ¹²

¹¹ Referencia 8. De Miguel Castaño, Adoración; Piattini Velthius Mario. España, 1999, 515 pp.

¹² Referencia 18. O'Brien, James A. Colombia, 2001, 700 pp.

Los sistemas se pueden dividir en dos grandes grupos: los *naturales* y los *artificiales*. Entre estos últimos, que son debidos al hombre, se encuentran los sistemas de información.

Sistema de Información.

Siempre que se reúnen personas de una organización, ponen en práctica algún tipo de sistema para recoger, procesar e intercambiar información. Estos sistemas no requieren computadoras para funcionar. Sin embargo, el poder de la tecnología informática es el que hace posible la viabilidad de los sistemas de información modernos. Por ello, cuando empleamos el término *sistema de información*, en general queremos decir sistema de información basado en computadoras o sistema de información informático.

Todo sistema organizacional depende, en mayor o menor medida de esta entidad abstracta denominada *sistema de información*. Este sistema es el medio por el cual los datos fluyen de una persona o departamento hacia otros y puede ser cualquier cosa, desde la comunicación interna entre los diferentes integrantes de la organización y 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, todo esto a través del uso de T.I. (Ver Figura 11).

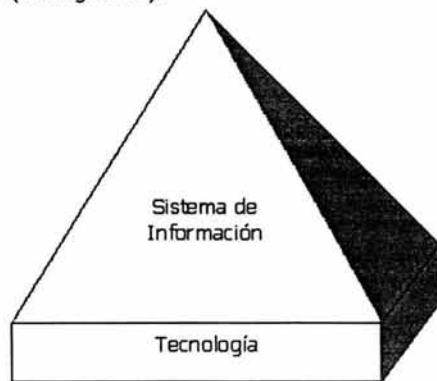


Figura 11. Modelo en pirámide de los sistemas de información informáticos.¹³

En un S.I. los componentes trabajan colectivamente para llevar a cabo entrada, procesamiento, salida, almacenamiento y acciones de control para convertir datos en información. Gracias a las funciones de estos sistemas se obtiene información que se utiliza tanto para una acción u operación, como para apoyar a los administradores y trabajadores a analizar y visualizar problemas complejos, para el control y visión de la empresa.

De acuerdo al concepto de sistema antes mencionado, se puede definir al sistema de información como un conjunto de elementos ordenadamente relacionados entre sí y que aporta a la organización la información necesaria para el cumplimiento de sus fines, para lo cual tendrá que recoger, procesar y almacenar datos, procedentes tanto de la misma organización como de fuentes externas, facilitando la recuperación, elaboración y presentación de los mismos. Uno de los instrumentos fundamentales para facilitar al S.I. el cumplimiento de estas funciones de recuperación, elaboración y presentación de los datos es la base de datos, de la cual se habla más adelante.

¹³ Referencia 29. Whitten, Jeffrey L.; Bentley, Loonie D.; Barlow, Victor M. España, 1996, 907 pp.

Los sistemas de información están formados por subsistemas que incluyen hardware, software, medios de almacenamiento de datos para archivos y bases de datos. El conjunto particular de subsistemas utilizados – equipo específico, programas, archivos y procedimientos – es lo que se denomina una *aplicación* de sistemas de información. De esta forma, los sistemas de información pueden tener varias aplicaciones.

1.2.2 Características.

Las características de un S.I. pueden agruparse en:¹⁴

- Tecnológicas, que afectan al rendimiento y seguridad del sistema, desde el punto de vista del equipo.
- Funcionales y semánticas, que se refieren a si el sistema hace lo que debe de una forma correcta (eficacia)¹ y si es capaz de adaptarse a requisitos cambiantes.
- Económicas, que poseen el énfasis en el coste del sistema y en la eficiencia con que responde a los objetivos.
- Sociales, que son las que tienen un impacto sobre el entorno social (interno o externo) en que se desenvuelve el sistema.

El S.I. puede ser comparado con un motor que impulsa la información, haciéndola circular por el organismo, distribuyéndola y aportándola a aquellas áreas donde es necesaria. Para realizar esta función es preciso que el sistema recoja previamente los datos allí donde son generados y los procese para convertirlos en información útil.

Entre el sistema de información y el organismo donde está inserto existe mutua y estrecha interrelación; en realidad, el S.I. no es otra cosa que un subsistema de los varios que integran la organización. Es imprescindible tener esto muy presente, ya que si no existe debida interacción y se produce un desfase entre ambos, el S.I. no podrá cumplir los objetivos para los que fue diseñado. La falta de adaptación entre el S.I. y el organismo es causa de fallo de muchos sistemas que prometían ser eficaces, estando demostrado que las causas de estos fracasos se encuentran más frecuentemente en los aspectos sociales y humanos que en el diseño tecnológico.

1.2.3. Clasificación.

Los sistemas de información pueden ser clasificados de acuerdo a:

- Estructura Organizacional
 - Área Funcional
 - Apoyo proporcionado (Support Provided)
 - Arquitectura de sistemas
 - Apoyo a actividades
- Clasificación por Estructura Organizacional
- Sistemas de información departamental. Frecuentemente, la organización usa algunos programas de aplicación en un área funcional o departamento.*
 - Sistemas de información empresarial. Mientras un sistema de información departamental usualmente está relacionado a un área funcional, la colección de todas las aplicaciones

¹⁴ Referencia 8. De Miguel Castaño, Adoración; Piattini Velthius Mario. España, 1999, 515 pp.

¹ La eficacia y la eficiencia son dos parámetros fundamentales para evaluar el comportamiento de un sistema. Llamamos eficacia al grado en que se cumplen los objetivos del sistema; es, en cierto modo, una medida referida a las relaciones externas del sistema con otro del que forma parte integrante. La eficiencia está más enfocada hacia operaciones y relaciones internas del sistema, y mide el grado de optimización en el uso de los recursos disponibles. Aunque son parámetros distintos, están muy interrelacionados, ya que ¿hasta qué punto la optimización de los recursos (eficiencia) no aumenta la eficacia?

* El sistema de información para la simulación y optimización de procesos de producción es departamental, ya que se desarrolla para el área de producción de hidrocarburos.

departamentales cuando se combinan con otras aplicaciones de funciones comprenden este sistema de información.

- Sistemas Interorganizacionales. Algunos sistemas de información conectan varias organizaciones.

- Clasificación por Área Funcional

Los principales sistemas de información funcionales son los siguientes:

- El sistema de información contable.
- El sistema de información financiero.
- El sistema de información manufacturero (operaciones / producción).⁹
- El sistema de información de mercadotecnia.
- El sistema de información de administración de recursos humanos.

- Clasificación por Apoyo proporcionado

Los principales tipos de sistemas bajo esta clasificación son:

- Sistemas de procesamiento de transacciones. Pueden almacenar, administrar y procesar grandes montos de información para transacciones de negocio de rutina. Ej. Sistema de cuentas de un usuario del banco.
- Sistemas de información de administración. Son sistemas de información computarizados para apoyar un amplio rango de funciones de negocio que hacen el procesamiento de datos. Proporcionan reportes periódicos para la planeación, control y toma de decisiones. Ej. Generar reportes de final de mes, mostrando número de nuevas cuentas, volumen de transacciones, etc.
- Sistemas de administración del conocimiento. Son algunas veces llamados sistemas expertos porque ellos simulan el razonamiento humano combinando una base de conocimiento y unas reglas de inferencia que determinan como es aplicado el conocimiento.
- Sistemas de automatización de oficina. Apoyan a las actividades de oficina, como procesamiento de textos, distribución electrónica de mensajes y correos, etc. En el futuro, es posible que la oficina se convierta en una red integrada de sistemas compatibles de cómputo/comunicación que enlace electrónicamente toda una organización.
- Sistemas de apoyo a la decisión. Apoyan a los tomadores de decisiones proporcionándoles información sobre demanda. Ej. Un sistema en línea que combina una hoja de cálculo con una base de datos para ayudar a los ejecutivos a elaborar un nuevo presupuesto.⁴
- Sistemas de planeación de recursos empresariales. Proporcionan acceso rentable a datos para usuarios y administradores por toda la compañía. Por ejemplo, una compañía que renta coches puede usar este sistema para pronosticar la demanda de clientes para renta de autos en cientos de localidades.
- Sistemas de productividad del usuario. Proporcionan empleos en todos los niveles organizacionales con una amplia colección de herramientas que pueden mejorar la calidad y desarrollo del empleo.
- Sistemas de Apoyo a Negocios. Proporcionan información de apoyo relacionada al empleo, a los usuarios de todos los niveles de la compañía. Una característica importante de estos es la capacidad de apoyo a decisiones para conducir a un análisis que – si (what-if). El apoyo a decisión ayuda a los usuarios a tomar decisiones creando un modelo de negocios y aplicando un conjunto de variables.
- Sistemas de apoyo inteligentes o Sistemas expertos. Capturan la experiencia y conocimientos de los tomadores de decisiones para interpretar información y para su uso en la solución de problemas. Ellos usan una base de conocimiento y reglas de inferencia que determinan cómo es usado el conocimiento para tomar decisiones. Muchos sistemas expertos usan un enfoque llamado lógica difusa (fuzzy logic) que permite que inferencias lógicas sean obtenidas de relaciones imprecisas. La

⁹ El sistema de información para los procesos de producción es un sistema manufacturero (producción).

⁴ El sistema de información para los procesos de producción de hidrocarburos es un sistema de apoyo a la decisión, ya que ayuda a los ejecutivos a modificar los valores en campo en cuanto a variables de proceso.

experiencia y conocimientos que obtuvieron estos sistemas los utilizan para servir como asistentes a los usuarios de un sistema de información. Ej. Un sistema que ofrece consejos a un administrador.

En la Figura 12 se muestra la relación que se puede dar entre los sistemas pertenecientes a esta clasificación.

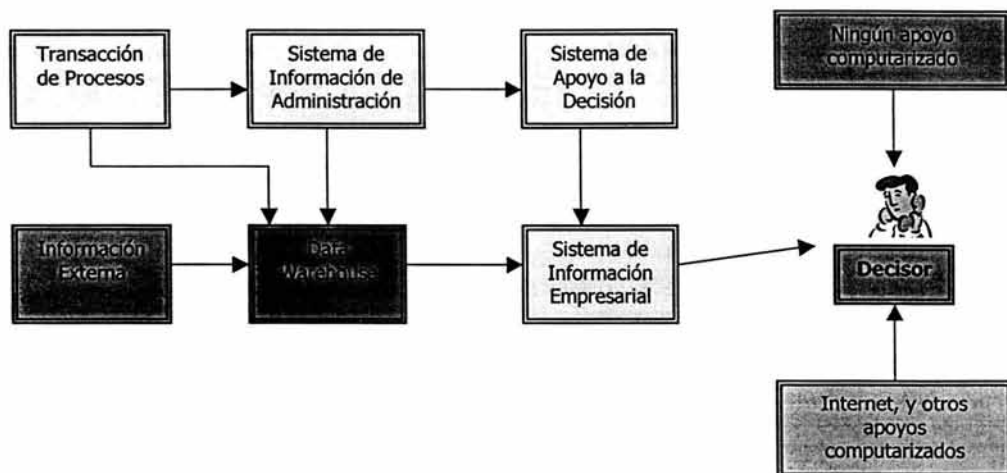


Figura 12. Sistemas de Apoyo interrelacionados.

- Clasificación por Arquitectura de Sistemas

Los sistemas de información pueden ser clasificados de acuerdo a tres tipos de arquitectura:

- Sistema basado en computadora central (mainframe).
- Una computadora personal autónoma.
- Un sistema de computación distribuido o conectado en red (existen algunas variaciones).

- Clasificación por apoyo a actividades

Otra importante forma para clasificar los sistemas de información es por la naturaleza de la actividad que ellos apoyan (ver Figura 13). Tal apoyo puede ser:

- Operacional. Operaciones diarias de una organización. Ejemplos de estos sistemas de apoyo a operaciones son: Sistemas de procesamiento de transacciones, Sistemas de automatización de oficinas.
- Directivo o de Gestión. Actividades de mandos medios, así como, planeación, organización y control a corto plazo. Ejemplos de sistemas de apoyo a la administración: Sistemas de información de administración, Sistemas de apoyo a la decisión, Sistemas de apoyo a Negocios.
- Estratégico. Decisiones que cambian significativamente la manera en que los negocios están siendo hechos. Ejemplos de sistemas de información estratégicos pueden ser: los sistemas de

procesamiento de transacciones, sistemas de información de administración, sistemas de apoyo a la decisión, etc. Son sistemas que ayudan a una organización a conocer sus objetivos estratégicos.

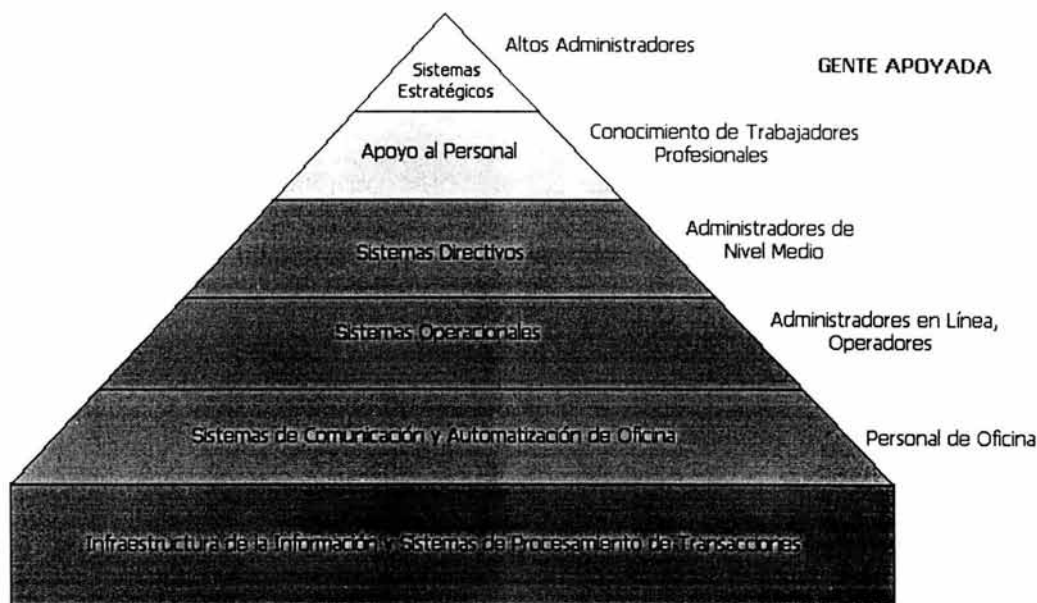


Figura 13. Sistemas de Información y Apoyo a la Gente.

La Infraestructura de la Información se compone principalmente de: Hardware de computadora, Software de desarrollo, Facilidades de redes y comunicaciones (incluyendo Internet e intranets), Bases de datos, Personal de administración de la información.

Los sistemas de procesamiento de transacciones apoyan el monitoreo, colección, almacenamiento, procesamiento y diseminación de las transacciones básicas de negocios de la organización. Estos proporcionan la entrada de datos para muchas aplicaciones envolviendo a otros sistemas de apoyo. Los sistemas de procesamiento de transacciones son considerados importantes para el éxito de cualquier organización.

Estos sistemas coleccionan datos continuamente, frecuentemente por días o aún en "tiempo real".

1.2.4. Componentes.¹⁵

Un sistema de información está constituido por una serie de componentes que se resume en la Figura 14.

¹⁵ Referencia 29. Whitten, Jeffrey L.; Bentley, Loonie D.; Barlow, Victor M. España, 1996, 907 pp.

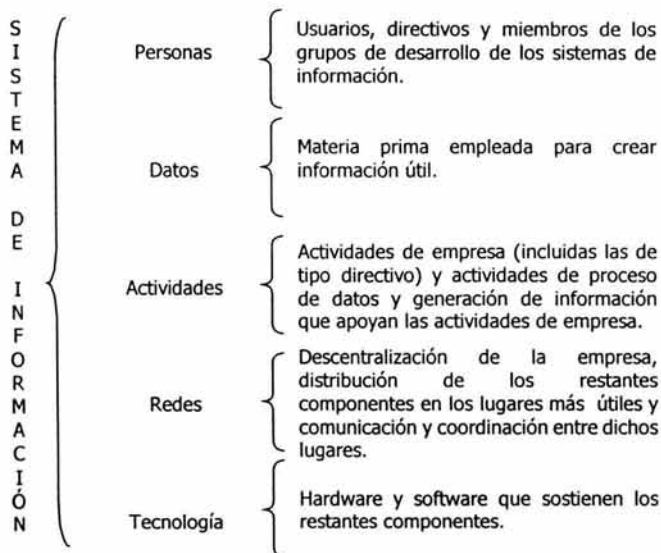


Figura 14. Componentes de un Sistema de Información

Personas. Son los individuos de la organización que interactúan con el sistema de información, es decir las personas que utilizan el sistema, tanto las que proporcionan los datos que el sistema necesita (secretarías, empleados de contaduría, capataces de planta, etc.), como las que reciben los resultados que el mismo produce (gerentes, administradores, contadores, empresarios comerciantes, productores). Y también las personas que diseñan y construyen el S.I. El propósito primario de un sistema de información es proporcionar información valiosa a los administradores, y usuarios dentro y fuera de la compañía.

- Los usuarios algunas veces llamados usuarios finales, incluyen empleados, clientes, vendedores y otras personas que interactúan con un sistema de información.

- Usuarios internos incluyen administradores, técnicos, representantes de ventas y oficiales corporativos.
- Los usuarios externos incluyen clientes quienes siguen la pista de sus órdenes en el sitio web de la compañía, proveedores que usan un sistema de clientes para planear su programa de manufactura y empleados que entran en la intranet de la compañía desde su casa para checar sus mensajes de correo electrónico.

Los diseñadores de sistemas traducen las necesidades y restricciones de empresa manifestadas por los usuarios a soluciones técnicas. Diseñan los archivos, las bases de datos, las entradas, las salidas, las pantallas, las redes y los programas informáticos requeridos por los usuarios de sistemas. Además integran las soluciones técnicas en el entorno de trabajo cotidiano de la empresa. Los constructores de sistemas fabrican sistemas de información multiusuario basados en las especificaciones de diseño obtenidas de los diseñadores. Algunas veces los usuarios finales, con ayuda de consultores, se encargan de construir sus propios sistemas de información.

Datos. Son las entradas del sistema, éstos datos son obtenidos de documentos, que se llaman fuentes de datos, que pueden ser externos a la empresa o ser producidos por otros sistemas de la empresa, por ejemplo: facturas, informes de ventas, (son documentos fuente de datos internos), en cambio las listas de precios enviadas por proveedores, (son fuentes de datos externas a la empresa).

Los datos se registran en formularios, se guardan en armarios de archivos o en libros y carpetas, se organizan en hojas de cálculo o se almacenan en archivos y bases de datos informáticos.

Las actividades (o procedimientos) que son las normas que determinan cómo se elabora la información, los procedimientos reflejan la forma de trabajar de nuestra empresa, qué se hace y con qué finalidad, cuándo y quién lo hace, dónde y cómo. Las actividades definen la función de un sistema de información.

Las actividades de una empresa son procesos cotidianos que sirven para apoyar sus cometidos, metas y objetivos. La mayor parte de las empresas se organiza en torno a actividades tales como la mercadotecnia, las ventas, las operaciones de almacén, las expediciones y las recepciones, la gestión de personal, la contabilidad y la producción.

Las actividades de los sistemas de información son procesos que apoyan las actividades de empresa por medio de el suministro de datos y el proceso de los mismos, y la mejora y simplificación de las actividades de la empresa. Algunas actividades pueden implantarse en forma de software. Otras han de ser realizadas por personas.

En esencia, las actividades son trabajos llevados a cabo para la empresa tanto por personas como por máquinas. Algunas actividades son repetitivas. Otras se producen con menor frecuencia, a veces casi nunca.

Las redes son estructuras de distribución de personas, datos, actividades y tecnología (los restantes componentes del S.I.) en los lugares adecuados de la empresa, y movimientos de datos entre dichos lugares.

El diseño de redes pretende suministrar procesos cooperativos entre los sistemas, las computadoras y las personas. Dicho diseño de redes ha experimentado un considerable auge en los últimos tiempos debido a que las empresas han incrementado su ritmo de trabajo, el ámbito de las mismas está en expansión, su complejidad es cada vez mayor y los datos se están duplicando sin cuidado a una velocidad alarmante.

Existen redes de empresa y redes de computadoras. Una red de empresa define los lugares detallados de trabajo, los recursos específicos disponibles en cada lugar y las necesidades de comunicaciones de empresa entre dichos lugares. Una red de empresa no requiere necesariamente una red informática. Una red informática es una disposición técnica que interconecta computadoras y periféricos de manera que puedan intercambiar datos y compartir recursos técnicos.

La tecnología incluye todo el hardware y software requerido para capturar, almacenar y gestionar los recursos de los datos, para apoyar las actividades de los sistemas de empresa y de información, y para interconectar la tecnología de datos y de procesos en diferentes lugares.

El hardware (la parte física) incluye las computadoras, los periféricos, las redes, las máquinas de fax, la telefonía, las impresoras inteligentes y otros tipos de dispositivos tecnológicos que apoyan el proceso de datos y las comunicaciones de empresa. El hardware es la capa física del sistema de información.

El software (o los programas), que a través de sus operaciones transformarán los datos en los resultados esperados. Los programas reflejan muchos de los procedimientos de la empresa.

El software consiste en software de sistema y software de aplicación.

- El software de sistema controla la computadora e incluye al sistema operativo, los controladores de dispositivos que comunican con el hardware, y utilidades que manejan tareas tales como convertir datos a diferentes formatos, protección de virus, crear copias de seguridad (backup), y sistemas de redes.
- El software de aplicación consiste en programas que apoyan a los usuarios y les permiten a las compañías llevar a cabo funciones de negocios.
 - Aplicaciones desarrolladas dentro de los negocios (In-house).
 - Paquetería de software desarrollado fuera de los negocios y vendidos a muchos otros negocios.

1.2.5. Ciclo de vida del desarrollo de un sistema de información

Uno de los enfoques de sistemas aporta la idea de ciclo de vida que es simple y lógica. Toda entidad (persona, familias, plantas, productos, máquinas, industrias, etc.) pasa a través de una serie de fases y el ciclo de vida es un medio conveniente para subdividir la duración de la vida en las diferentes etapas. Cada una de estas etapas tiene sus características propias y únicas.

El ciclo de vida del desarrollo de sistemas es un conjunto ordenado y sistemático de acciones cuyo fin es resolver los problemas de la empresa, por medio del desarrollo y el soporte de los sistemas de información resultantes. Es un proceso por el cual los analistas de sistemas, los ingenieros de software, los programadores y los usuarios finales elaboran sistemas de información y aplicaciones informáticas.

El concepto de ciclo de vida de un sistema de información es medular en las investigaciones de sistemas. Durante su desarrollo, cada sistema se mueve a través de varias fases de un ciclo de vida, después del cual sólo funciona por varios años con un mínimo mantenimiento. El sistema se deteriora gradualmente hasta el punto en que cesa de funcionar por completo y se comienza un nuevo ciclo de vida con el desarrollo de un nuevo sistema.

Un ejemplo de ciclo de vida es utilizado en la fluctuación de ventas de un producto desde su introducción hasta su salida del mercado. Esto se puede representar con una gráfica (ver Figura 15).



Figura 15. Ciclo de vida de un producto.

Una curva similar a ésta puede ser dibujada para un ciclo de vida de un sistema de información (ver Figura 16). El sistema puede ser una simple aplicación o un conjunto de aplicaciones de un sistema integrado de información para una organización completa. Por lo tanto todo S.I. tiene:

- **Nacimiento.** Durante esta fase, surge la idea de cómo puede ayudar un sistema, se descubre un problema o una oportunidad de desarrollar útilmente un nuevo sistema, y se lleva a cabo una cantidad limitada de investigación preliminar para ver si un proyecto de sistemas está garantizado.
- **Desarrollo.** Durante el desarrollo, las ideas se hacen realidad. Se diseña el nuevo sistema o una aplicación computarizada para satisfacer las necesidades. Asimismo se realiza la programación, e instalación de equipo.
- **Evolución - Operación.** Se instala el sistema y empieza a cubrir las necesidades de la empresa. El sistema está en operación continua.
- **Retroalimentación.** La retroalimentación parte del principio de que todos los elementos de una totalidad de un sistema deben comunicarse y ser funcionales entre sí para poder desarrollar interrelaciones coherentes. Sin comunicación y funcionalidad no hay orden y sin orden no hay totalidad. En esta etapa el sistema alcanza un alto desempeño y se modifica continuamente.

- Declive – Operación. Después de su máximo desempeño la efectividad de su costo declina gradualmente al cambiar su ambiente, al cambiar sus costos de operación o al convertirse en obsoleto el equipo. El sistema acumula muchas modificaciones.
- Desaparición o muerte. Se reconoce que el sistema no está funcionando satisfactoriamente y se reemplaza. De acuerdo al análisis realizado, se diseña un sistema de información a partir del existente o se crea uno completamente nuevo.



Figura 16. Ciclo de vida de un sistema de información.

Etapas del ciclo de vida del desarrollo de sistemas.¹⁶

En su mayoría los ciclos de vida del desarrollo de sistemas constan de fases. En su forma clásica más simple, los ciclos de vida constan de cuatro fases (ver Figura 17, parte izquierda): análisis de sistemas, diseño de sistemas, implantación de sistemas y soporte de sistemas. Modernas variantes han añadido a esto una nueva fase: la planificación de sistemas (ver figura 17, derecha).

¹⁶ Referencia 29. Whitten, Jeffrey L.; Bentley, Loonie D.; Barlow, Victor M. España, 1996, 907 pp.

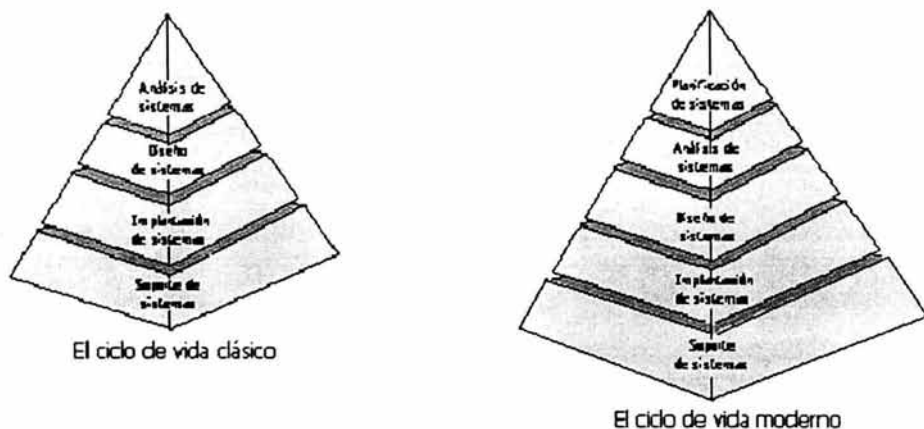


Figura 17. Los ciclos de vida del desarrollo de sistemas clásico y moderno. El modelo en pirámide puede cortarse en niveles para poner de relieve las fases clásicas y modernas del ciclo de vida del desarrollo de sistemas. La principal diferencia que existe entre ambos es que el ciclo de vida moderno está dirigido por la planificación de sistemas. Al descender en la pirámide, el interés se desplaza desde las cuestiones de empresa a las cuestiones tecnológicas.

Adviértase que el modelo en pirámide se ha cortado en varios niveles para ilustrar las fases. Cada nivel tiene más volumen que el anterior, al incluir su fase un mayor grado de detalle. Esos detalles describen los componentes del sistema de información: personas, datos, actividades, redes y tecnología. Conforme se avanza por las fases hacia abajo en el modelo, se acerca a la base tecnológica de la pirámide (ver Figura 11), lo que sugiere que las primeras preocupaciones deben ser las relacionadas con la empresa, y las últimas serán de índole más técnica.

Como puede verse, el ciclo de vida moderno consta de cinco funciones de alto nivel, que son:

1. **Planificación de sistemas:** Es el estudio continuado del entorno de un problema con el fin de determinar las posibilidades de solución que entraña. Su propósito es identificar y establecer las prioridades sobre aquellas aplicaciones de los sistemas de información cuyo desarrollo reporte máximos beneficios para la empresa considerada en su conjunto. Esta fase indica la relativa madurez del funcionamiento de los sistemas de información. Aquí se detecta un sistema de la empresa que necesita mejoras.

Sus entradas son las misiones de empresa y cualquier tipo de detalle o limitación de los sistemas existentes. Sus salidas clave o productos son los planes de sistemas de información y los proyectos planificados de desarrollo de aplicaciones. En consecuencia, los proyectos planificados giran hacia las fases restantes.

Se selecciona el sistema para ser desarrollado. Se hace una determinación preliminar de las necesidades de información, de los objetivos y restricciones del sistema y recursos requeridos. También se hace un estudio de costo - beneficio por lo cual a veces se le llama *Estudio de Factibilidad*.

2. **Análisis de sistemas:** Su propósito es analizar el problema o la situación de empresa de que se trate y, entonces, definir las necesidades de la empresa con respecto a la creación o el perfeccionamiento de un sistema de información. Las necesidades de empresa no implican obligatoriamente una solución de tipo informático.

El suceso que activa el análisis es bien el proyecto planificado de desarrollo de aplicaciones (procedente de la fase anterior) o un proyecto no planificado que responde a un problema. Otras posibles entradas son los detalles y limitaciones de los sistemas existentes y hechos y necesidades relacionados con la empresa. El producto clave obtenido es una relación de las necesidades de empresa que explica lo que precisan los usuarios, aunque no cómo se proyecta diseñar o implantar dichas necesidades.

Se analizan en detalle las *necesidades o requerimientos* de información de los usuarios, los flujos de datos a través del sistema y la relación del nuevo sistema dentro de la estructura y objetivos generales de la organización. El análisis debe producir las especificaciones funcionales del sistema o en otras palabras, *QUÉ* se va a hacer exactamente.

3. Diseño de sistemas: Es la evaluación de las diferentes soluciones alternativas, así como la especificación detallada de la solución final. Su propósito es diseñar una solución técnica, de tipo informático, que satisfaga las necesidades de empresa según han sido especificadas durante el análisis de sistemas.

El suceso que lo activa es la relación de necesidades de empresa. Otras entradas son las opiniones y recomendaciones relacionadas con el diseño expuestas por los usuarios de sistemas. El producto resultante principal del diseño de sistemas es una relación técnica de diseño. Este producto establece (o demuestra) cómo conseguirá el sistema de información satisfacer las necesidades de empresa de los usuarios.

En esta etapa se diseña o se define *CÓMO* se van a hacer reales las especificaciones funcionales definidas en el análisis. Aquí se definen los documentos, la estructura de la base de datos, la arquitectura del software, así como las necesidades detalladas de *hardware, software y personal* necesario para implantar y operar el sistema. También se codifican los programas que se definieron.

4. Implantación de sistemas: Su propósito es construir y/o ensamblar los componentes técnicos y poner en funcionamiento el sistema de información nuevo o mejorado.

En esta etapa se prueban los programas que se definieron en la etapa de diseño. Al final de las pruebas se reúnen todos los módulos y se hacen las pruebas integradas del sistema.

Una vez implantado, el nuevo sistema se dice que está en producción. El término producción se utiliza para describir un sistema que ha sido puesto en funcionamiento cotidiano. Se realiza la carga de archivos con datos reales, la capacitación, la preparación de las facilidades físicas para continuar con la actividad de más larga duración como es el mantenimiento y mejora del sistema. Sus salidas son la documentación y formación de usuarios finales necesaria para utilizar el sistema en producción.

5. Soporte de sistemas: El propósito del soporte de sistemas es sostener y mantener el sistema durante el resto de su vida útil (que puede durar semanas, meses o años). Es el mantenimiento continuado de un sistema después de que haya sido puesto en funcionamiento. Ello incluye el mantenimiento de programas y las mejoras al sistema.

La entrada a esta fase es el sistema de información en producción. También se activan diversas actividades de soporte a partir de los problemas del uso del sistema. Llegará un momento en que el sistema en producción será demasiado caro de mantener, o dejará de suministrar el apoyo adecuado a la empresa. Entonces, el ciclo de vida deberá cerrarse para volver a las fases de planificación o análisis de sistemas.

El ciclo de vida, como su nombre lo dice, debe ser visto como un proceso continuo, que vuelve a empezar una vez que la fase de soporte de sistemas se termina debido a circunstancias que hacen obsoleto o inadecuado el sistema. Normalmente el sistema es reemplazado por uno más eficiente, con mayores posibilidades y soportado por una tecnología más avanzada. Otro punto importante para considerar es que la mayoría de las etapas se

sobreponen en mayor o menor grado unas con otras durante el ciclo de vida. En efecto, las fases que hemos mencionado, no pueden considerarse como etapas aisladas; las actividades de análisis y diseño especialmente, se realizan en cierto grado en cada una de las etapas restantes del ciclo de vida (ver Figura 18).

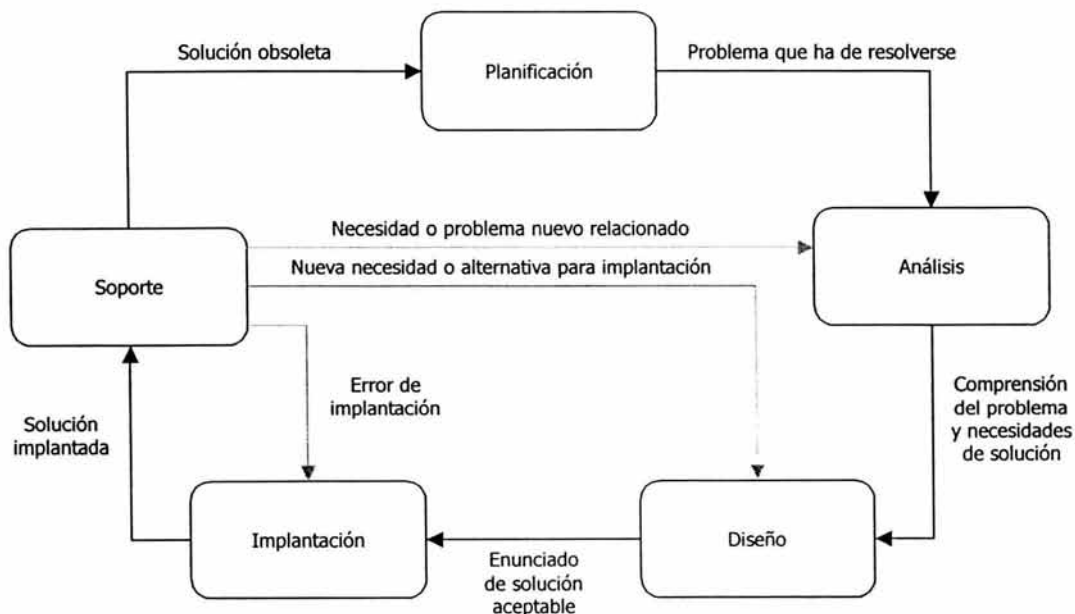


Figura 18. Modelo del proceso de las etapas del ciclo de vida del desarrollo de un sistema de información.

1.3. Bases de Datos

Las bases de datos, hoy en día, ocupan un lugar determinante en cualquier área del quehacer humano, comercial, y tecnológico. No sólo las personas que están involucradas en el área de Informática, sino todas las personas administrativas, técnicas y con mayor razón los profesionales de cualquier carrera, deben de tener los conocimientos necesarios para poder usar las bases de datos.

1.3.1. Historia

En los tiempos previos a las computadoras, el monto de información custodiada por un grupo de gente podía ser reunido en la sabiduría y las historias de sus miembros más viejos. En este mundo, los cuentistas, magos, y abuelos eran considerados grandes y honorados tesoros o minas por todo eso que era sabido. Aparentemente, y de acuerdo a los vastos datos arqueológicos, las fogatas eran usadas por los integrantes más jóvenes de la comunidad para tener acceso a la información almacenada en las mentes de los más ancianos.

Cuando llegó la agricultura, la sobreproducción de comida y los orígenes del comercio moderno, aparecieron los depósitos de arroz, trigo y maíz y los monarcas y emperadores desarrollaron la nueva economía. Había simplemente demasiados datos para ser manejados en las mentes de los más viejos. Y entonces, para almacenar toda la información nueva, la humanidad inventó la tecnología de escribir, y con la invención del alfabeto los datos empezaron a ser almacenados en depósitos voluminosos, llamados libros. Eventualmente los libros se propagaron con gran velocidad y pronto, comunidades enteras de libros migraron a las primeras "base de datos" reales, las bibliotecas.

A diferencia de las versiones previas de almacenes de datos (personas y libros), las bibliotecas revolucionaron, a pesar de que ellas eran increíblemente primitivas por supuesto. Específicamente, las bibliotecas introdujeron "estándares" por los cuales los datos podían ser almacenados y recuperados. Después de todo, sin estándares para lograr acceso a los datos, las bibliotecas serían interminables, un caos. Los libros, y los datos en éstos, tenían que ser accesibles rápidamente por cualquier persona para poder ser útiles. En efecto, la utilidad de una biblioteca, o de cualquier base de datos, es proporcional a su eficiencia en recuperación y almacenamiento. Este corolario dirigirá la evolución de las bases de datos sobre los próximos 2000 años hasta su estado actual. De este modo, las primeras bibliotecas definieron protocolos de clasificación y recuperación estandarizados. Y para el siguiente par de miles de años las bibliotecas crecieron, y crecieron junto con tecnologías de almacenamiento y recuperación como los ficheros o archiveros, las etiquetas de colores, etc.¹⁷

Todo esto hasta el surgimiento de la computadora. La computadora fue por supuesto, una tecnología intensamente revolucionaria. Casi instantáneamente, la computadora fue aplicada al antiguo problema de almacenamiento y recuperación de información. Después de todo, por la Segunda Guerra Mundial, la información estaba ya acumulándose en proporciones más allá del espacio disponible en las bibliotecas públicas. De este modo, los primeros intentos de almacenamiento y recuperación de la información siguieron las líneas tradicionales. El primer sistema estuvo basado en archivos discretos en una biblioteca virtual. En este sistema orientado a archivos, bastantes archivos serían almacenados en una computadora y se podría acceder a él por medio de un operador de computadoras. Los archivos de datos archivados fueron llamados "tablas" porque ellos parecían como tablas usadas en la conservación de archivos tradicionales. Las filas en la tabla fueron llamadas "registros" y las columnas fueron llamadas "campos" (Ver Figura 19).

¹⁷ Referencia Sitios de Internet 5. http://www.wdvl.com/Authoring/DB/Intro/what_is_database.html

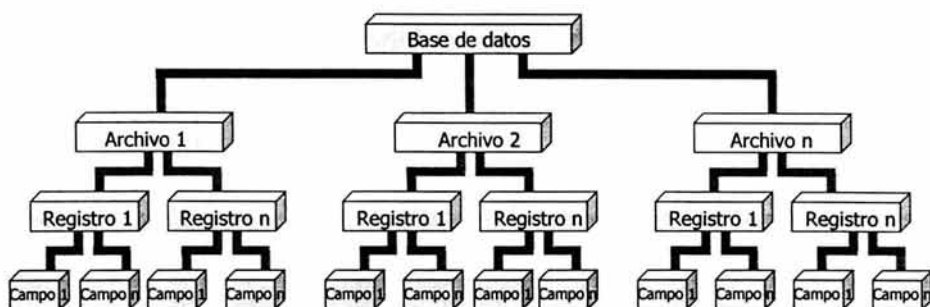


Figura 19. Jerarquía de organización de datos empleada por los sistemas de procesamiento basados en computadora. El nivel más alto de la jerarquía de organización de datos es la base de datos, que es un conjunto de elementos de datos relacionados entre sí.¹⁸

Las bases de datos no son un fenómeno nuevo. De hecho, Herman Hollerith mecanizó el almacenamiento del Censo de EE.UU. de 1890 en lo que de alguna forma se considera la primera base de datos significativa "computarizada". Hoy, la importancia e impacto de las bases de datos es incuestionable a medida que organizaciones gubernamentales, instituciones académicas, y entidades comerciales crean y mantienen importantes bases de datos que contienen toda clase de información desde documentos de texto en lenguaje natural, tablas estadísticas, datos financieros y objetos multimediales hasta datos de naturaleza técnica y científica. Muchas bases de datos están compuestas de *metadatos*, lo cual significa que los registros guardan "datos acerca de los datos" tales como información acerca del tamaño y carácter de otra base de datos en lugar de ser la fuente primaria de contenido tal como nombre y domicilio de una persona.¹⁹

Las tecnologías de bases de datos, incluyendo métodos de arquitectura y acceso, se están desarrollando rápidamente para mantenerse al día con esta demanda de mecanismos de administración de la información. Los diseñadores y administradores de bases de datos enfrentan muchos desafíos que reflejan la complejidad del floreciente entorno de la información. Las tecnologías de bases de datos deben manejar masivas cantidades de datos, extraer información útil desde estos almacenes, y tener la habilidad para reflejar las relaciones entre los datos mantenidos en diferentes bases de datos. Además de la arquitectura y sistema deben proveer integridad, recuperación, concurrencia, y seguridad.

Para responder a estos desafíos, los tres modelos fundamentales de bases de datos (jerárquico, red y relacional), han servido como la base para desarrollar modelos de datos más potentes y flexibles, tales como los modelos relacional extendido y el relacional de objetos. Un esquema de datos y una arquitectura bien definida aseguran almacenamiento de datos lógico y eficiente lo cual incrementa la capacidad de la base de datos y extiende las capacidades de los lenguajes de consulta y otros métodos de acceso. Además, la minería de datos crea información útil al identificar datos relacionados dentro de los vastos almacenamientos.

1.3.2. Definición.

La aparición de la expresión *base de datos* se produce a comienzos de los años setenta. Anteriormente, en el mundo de la informática se hablaba de archivos y conjuntos de datos. En 1963 tuvo lugar en Santa Mónica (EUA) un simposio en cuyo título se encontraba la expresión *Data Base*. Posteriormente, en 1967, el grupo de estandarización Codasyl decidió cambiar su primitiva denominación en la que no aparecía la expresión *bases de datos* por el de Data Base Task Group. Poco a poco el concepto y la expresión *base de datos* iba imponiéndose.

¹⁸ Referencia 22. Sanders, Donald H. México, 1990, 887 pp.

¹⁹ Referencia Sitios de Internet 6. <http://www.acm.org/crossroads/espanol/xrds7-3/intro73.html>

Las colecciones de bases de datos son denominadas, a veces, bancos de datos. Estas expresiones, *banco de datos* y *base de datos*, a veces se emplean como sinónimas y su alcance y diferenciación es conveniente precisar. En el campo de la documentación, la distinción entre ambos conceptos suele estar fundada en el tipo de información que se almacena, dándose en general el nombre de banco de datos a los sistemas que contienen información factual, es decir, datos numéricos o textuales que tienen para el usuario un interés en sí mismos; mientras que la denominación base de datos alude a los sistemas cuyo contenido es información referencial, o sea, referencias bibliográficas que nos permitirán posteriormente buscar el correspondiente documento, pero no el dato mismo.

Esta distinción no es unánimemente aceptada, y la interpretación que en el área informática se da a estas expresiones es diferente. Para algunos autores un banco de datos es un simple *almacén* en soporte informático, de los datos relativos a un determinado tema o materia. La información puede ser exhaustiva o simplemente masiva. Pero, el concepto de banco de datos, ha de reunir otras características necesarias para precisarlo y delimitarlo, las cuales provienen de su analogía con las entidades bancarias, de las que nació la expresión. Estas características son:

- comprende datos de *diversas fuentes*
- integrados en una *única organización*
- disponibles para múltiples *usuarios*

Se trata de un *enfoque organizativo* independiente de la tecnología utilizada (estructuración de los datos o sistema de gestión de los mismos). Al contrario de lo que ocurre con las bases de datos, cuya caracterización se apoya precisamente en *aspectos tecnológicos*.

Algunos autores opinan, sin embargo, que un banco de datos es un sistema de información que, reúna o no las anteriores características, está constituido por una base o depósito de datos, su sistema de gestión y la organización técnica y material que los soporta. En otros casos aparecen como sinónimos *banco* y *base de datos*.

El concepto de base de datos se ha definido de numerosas y diversas formas, una de ellas bastante completa es la siguiente, una base de datos es una colección o depósito de datos donde los datos están lógicamente relacionados entre sí, tienen una definición y una descripción comunes y están estructurados de una forma particular. Una base de datos es también un modelo del mundo real, y como tal debe poder servir para toda una gama de usos y aplicaciones.

Una base de datos debe estar organizada para proporcionar recuperación eficiente. Hay bases de datos físicas (papel / impresiones) y bases de datos electrónicas, ambas proporcionan información en una combinación de formatos: impresa, gráfica, audio, estadística, o combinaciones.

En resumen, se considera que un S.I. es un banco de datos cuando ofrece, de forma más o menos restringida, su acceso en el mercado de la información, bien directamente por la institución que lo ha producido o por medio de intermediarios; mientras que las bases de datos son S.I. cuyo objetivo fundamental es la utilización interna por parte de su creador * y que cumplen unos determinados requisitos tecnológicos en cuanto a su diseño lógico y a la organización de los datos. Por tanto, con la expresión *banco de datos* se refiere a una forma de organizar la *difusión* de la información, mientras que con *bases de datos* se hace alusión a una determinada *tecnología de estructuración* de los datos y, a una distinta filosofía en la concepción y diseño de un S.I. Desde luego, un banco de datos puede estar apoyado, y así ocurrirá por lo común, en la tecnología de bases de datos.

1.3.2.1. Bases de datos electrónicas

Actualmente, se acostumbra llamar base de datos a un grupo de datos que se encuentran en forma digital, y pueden ser procesados por medio de sistemas electrónicos de procesamiento de datos.

* Una base de datos puede convertirse con el transcurso del tiempo en un banco de datos de acceso público.

La expresión "base de datos" proviene del inglés *database* (que a veces se escribe separado, o con guión), y el diccionario de Oxford lo refleja como: "Una colección de datos estructurada, almacenada en una computadora, especialmente la que incorpora programas para hacerla accesible de distintas formas".

Son muy numerosas las definiciones de base de datos, y si se analizan detenidamente, se suele observar en casi todas ellas coincidencias en ciertos elementos; aunque también se detecta la falta de otros fundamentales, o al menos muy importantes, que son característicos de las bases de datos. En la Figura 20 se reproducen distintas definiciones de base de datos.

- "Colección de datos interrelacionados almacenados en conjunto sin redundancias perjudiciales o innecesarias"; su finalidad es servir a una aplicación o más, de la mejor manera posible; los datos se almacenan de modo que resulten independientes de los programas que los usan; se emplean métodos bien determinados para incluir nuevos datos y para modificar o extraer los datos almacenados." (*Martin, 1975*).
- "Conjunto de datos de la empresa memorizado en una computadora, que es utilizada por numerosas personas y cuya organización está regida por un modelo de datos." (*Flory, 1982*).
- "Conjunto estructurado de datos registrados sobre soportes accesibles por computadora para satisfacer simultáneamente a varios usuarios de forma selectiva y en tiempo oportuno." (*Delobel, 1982*).
- "Colección no redundante de datos que son compartidos por diferentes sistemas de aplicación." (*Deen, 1985*).
- Conjunto de ficheros maestros, organizados y administrados de una manera flexible de modo que los ficheros puedan ser fácilmente adaptados a nuevas tareas imprevisibles." (*Frank, 1988*).
- "Colección de datos interrelacionados." (*Eslmari y navathe, 1989*).

Figura 20. Diferentes definiciones de base de datos.²⁰

A continuación se estudiarán las características esenciales de una base de datos para apoyar en ellas el concepto de base de datos. En primer lugar, una base de datos es un *conjunto, colección o depósito de datos* almacenados en un soporte informático no volátil. Los datos están *interrelacionados y estructurados* de acuerdo con un modelo capaz de recoger el máximo contenido semántico. Dada la relevancia que tienen en el mundo real las interrelaciones entre los datos, es imprescindible que la base de datos sea capaz de almacenar estas interrelaciones. En el mundo real existen, además restricciones semánticas, a las que se está concediendo una importancia creciente que, en los sistemas actuales, tienden a almacenarse junto con los datos, al igual que ocurre con las interrelaciones. La base de datos describe y se manipula apoyándose en un modelo de datos.

La redundancia de los datos debe ser *controlada*, de forma que no existan duplicidades perjudiciales ni innecesarias, y que las redundancias físicas, convenientes muchas veces a fin de responder a objetivos de eficiencia, sean tratadas por el mismo sistema, de modo que no puedan producirse inconsistencias. Esto podría resumirse diciendo que en las bases de datos no debe existir *redundancia lógica*, aunque si se admite cierta *redundancia física* por motivos de eficiencia. Por tanto, un dato se actualizará *lógicamente* por el usuario de forma única, y el sistema se preocupará de cambiar físicamente todos aquellos campos en los que el dato estuviese repetido en caso de existir redundancia física; es lo que se denomina también *redundancia controlada* por el sistema.

Las bases de datos pretenden servir al conjunto de la organización, manejando los datos como otro recurso que viene a añadirse a los ya tradicionales. Por tanto, las bases de datos han de atender a *múltiples usuarios* y a *diferentes aplicaciones* en contraposición a los sistemas de ficheros, en los que cada fichero está diseñado para responder a las necesidades de una determinada aplicación.

²⁰ Referencia 8. De Miguel Castaño, Adoración; Piattini Velthius Mario. España, 1999, 515 pp.

Otro aspecto importante de las bases de datos es la *independencia*, tanto física como lógica, entre datos y tratamientos. Esta independencia, objetivo fundamental de las bases de datos, es una característica esencial que distingue las bases de datos de los ficheros.

La *definición* o *descripción* del conjunto de datos contenidos en la base (lo que se denomina *estructura* o *esquema* de la base de datos) deben ser *únicas* y estar *integradas* con los mismos datos. En los sistemas basados en ficheros, los datos se encuentran almacenados en ficheros, mientras su descripción está separada de los mismos, formando parte de los programas, para lo cual se precisa que los lenguajes faciliten medios para la descripción de los datos. Suele haber, además una documentación adicional, habitualmente en soporte de papel, en general insuficiente y desactualizada. Este tipo de organización da origen a infinidad de problemas, ya que a veces no se sabe cuál es la descripción de un determinado fichero, bien por pérdida de la misma, bien porque no se ha actualizado debidamente la correspondiente documentación y tampoco se conoce exactamente el programa que lo trataba. En las bases de datos, la descripción, y en algunos casos también una definición y documentación completas (metadatos), se almacena junto con los datos de modo que éstos están *autodocumentados*, y cualquier cambio que se produzca en dicha documentación se ha de reflejar y quedar recogido en el sistema, con todas las ventajas que de este hecho se derivan.

El concepto de base de datos ha ido cambiando y configurándose a lo largo del tiempo; en la actualidad la base de datos se puede definir como: "Colección o depósito de datos integrados, almacenados en soporte secundario y con redundancia controlada. Los datos, que han de ser compartidos por diferentes usuarios y aplicaciones, deben mantenerse independientes de ellos, y su definición (estructura de la base de datos) única y almacenada junto con los datos, se ha de apoyar en un modelo de datos, el cual ha de permitir captar las interrelaciones y restricciones existentes en el mundo real. Los procedimientos de actualización y recuperación, comunes y bien determinados, facilitará la seguridad del conjunto de los datos."

El contenido de una base se obtiene combinando datos de todas las diferentes fuentes en una organización, de tal manera que los datos estén disponibles para todos los usuarios, y los datos redundantes puedan eliminarse, o al menos minimizarse. En la Figura 21 se muestra la base de datos como un recipiente de datos a ser compartidos por varios programas. El usuario podrá recobrar datos de varias partes de la base ya que los archivos ahí almacenados, están conectados directa o indirectamente.

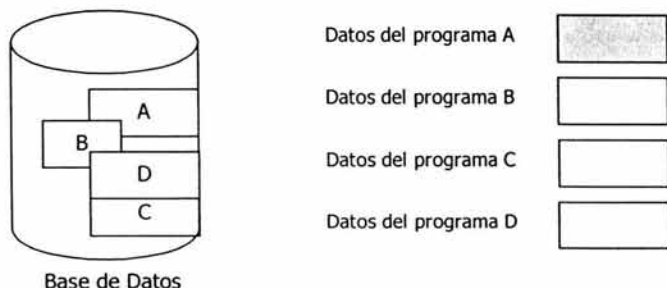


Figura 21. Vista esquemática de datos almacenados. Los datos se almacenan físicamente en una disposición distinta a la de la perspectiva lógica. Todos los usuarios pueden tener acceso a los datos.²¹

Los datos son persistentes y se encuentran almacenados en cada unidad de información de la base de datos, éstos se refieren a las características particulares de personas, cosas y eventos (conocidos comúnmente como entidades). Por ejemplo, una base de datos bibliográficos contendrá información sobre libros, reportes, artículos de revista, etcétera. Estos datos se pueden ordenar, buscar, agregar, editar e imprimir cuando se necesite. La base de datos es como el contenedor que sostiene los datos, y que es capaz de organizarlos y manipularlos.

²¹ Referencia 27. Tsai, Alice Y. H. México, 1990, 607 pp.

1.3.2.2. Ejemplos:

Algunos ejemplos de bases de datos electrónicas son:

- Sistemas de control de procesos, que se utilizan para verificar y controlar refinerías, procesos químicos, molinos y operaciones de maquinado.
- Sistemas de alta velocidad para adquisición de datos, que obtienen datos de telemetría a alta velocidad de satélites en órbita o las computadoras que capturan cantidades enormes de datos de experimentos de laboratorio.
- Un sistema de reservación de aerolíneas.
- Catálogo electrónico de libros y revistas.
- Sistemas de conmutación telefónica, que controlan la transmisión de voz y datos en miles de llamadas telefónicas, detectando los números marcados, condiciones de ocupado y todas las demás condiciones de una red telefónica típica.
- Archivos en el disco duro de una computadora.
- Registros de los pacientes en un hospital.

Otros ejemplos de bases de datos son listas de miembros de una organización, listas de clientes, catálogos de bibliotecas, archivos de tarjetas de negocios, e inventario de partes. La lista de bases de datos posiblemente útiles son, de hecho, infinitas.

1.3.3. Clasificación.

- Bases de datos Operacionales. Se refieren a las bases de datos que son usadas para operaciones diarias. Ellas son en las que quizá más comúnmente pensamos. Ellas proveen información detallada para ayudar a las operaciones de negocios. Ellas pueden tomar una variedad de nombre como bases de datos de producción o de transacciones. Ellas fueron el primer tipo de bases de datos que fue usado.
- Bases de datos Analíticas. Son las bases de datos que está almacenadas en datos organizacionales y son usados por los gerentes y usuarios para analizar tendencias de negocios dentro de la organización. Los datos almacenados en éstas pueden ser usados para Procesamiento Analítico en Línea (OLAP), Sistemas de apoyo a decisiones, o Sistemas de Información Empresarial, entre otros. El uso de estas bases de datos está limitado por lo costoso del almacenamiento secundario y la práctica de computación centralizada. La mayoría de las organizaciones tiene una computadora central (mainframe), y la mayoría de sus recursos fueron obtenidos para bases de datos operacionales.
- Data Warehouses. Son grandes bases de datos de multi – propósito que actúan como un depósito central para los datos extraídos de varias fuentes. Estos datos colocados tienden a ser masivos y requieren un tipo de técnica llamada data mining (“el proceso de extracción de conocimiento inédito, validación y despliegue de la información de grandes conjuntos de datos y con el objetivo de toma de decisiones cruciales para la empresa”) para analizarlos. Esto no era solo el advenimiento de grandes montos de almacenamiento secundario barato y velocidades de procesamiento muy rápidas (data mining depende de técnicas de procesamiento estadísticas) que se convierten en almacenes factibles.
- Bases de datos Externas. Hoy en día, hay una gran cantidad de datos disponibles en la internet. Lo importante para muchas organizaciones es el costo de la información (si no hoy en el futuro) y si este encuentra todas las necesidades individuales de la organización.
- Bases de datos Distribuidas. Las bases de datos que están dispersas cruzando diferentes localidades geográficas. Éstas llegan a ser viables debido a los avances en la tecnología de computación, el desarrollo de las redes de área local (LAN), la intranet, el Internet, y, por supuesto, los desarrollos en la infraestructura de telecomunicaciones. La idea es muy simple: “Los datos deben ser conservados en la localidad donde son más frecuentemente usados”.

1.3.4. Ventajas de las Bases de Datos

- a) Los Datos:
 - Independencia de éstos respecto de los tratamientos y viceversa.
 - Mejor disponibilidad de los mismos.
 - Mayor eficiencia en la recogida, codificación y entrada en el sistema.

- b) Los Resultados:
 - Mayor coherencia.
 - Mayor valor informativo.
 - Mejor y más normalizada documentación de la información.

- c) Los Usuarios:
 - Acceso más rápido y sencillo de los usuarios finales.
 - Más facilidades para compartir los datos por el conjunto de los usuarios.
 - Mayor flexibilidad para atender a demandas cambiantes.

1.3.5. Organización de una base de datos.²²

Una base de datos, a diferencia de un archivo, la comparten muchos usuarios. Y naturalmente cada usuario verá los datos de manera diferente. Nos referimos a la forma en que un usuario concibe y describe los datos desde una presentación de usuario. Sin embargo, el problema es que usuarios diferentes tienen enfoques diferentes. Estas presentaciones se examinan en el modelo lógico global de la base de datos, que eventualmente deberá desarrollarse. Finalmente, el modelo lógico de la base de datos debe transformarse en el correspondiente diseño físico de la base de datos. El diseño físico considera la forma del almacenamiento de los datos y de sus interrelaciones. La Figura 22 muestra cómo se relacionan el reporte de usuario y la presentación del usuario (esquema del usuario) a un modelo lógico (esquema conceptual) y a un diseño físico (esquema interno).

²² Referencia 13. Kendall y Kendall. México, 1991, 881 pp.

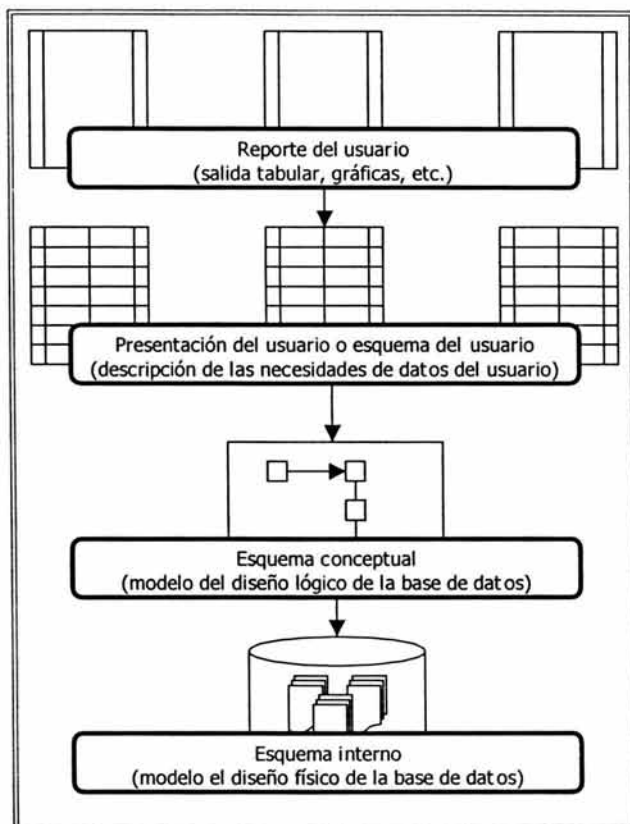


Figura 22. La estructura lógica del usuario o esquema externo es la visión que tiene de la base de datos cada usuario en particular; la estructura lógica global o esquema conceptual responde al enfoque del conjunto de la empresa y la estructura física o esquema interno es la forma en que se organizan los datos en el almacenamiento físico.

1.3.6. Modelos de Bases de datos.²³

Hay cuatro modelos de bases de datos con una estructuración lógica que se pueden usar cuando se desarrolla una base de datos:

- Jerárquico. El primer modelo operativo de base de datos fue el modelo jerárquico. En este modelo se tiene una entidad como piedra angular de toda la base de datos. A partir de esta entidad se pueden encontrar todos los datos.

Las estructuras jerárquicas en ocasiones se denominan árboles porque los subordinados conectados las entidades a las cuales pertenecen semejan las ramas de un árbol, aunque curiosamente dibujadas hacia abajo, tal como se muestra en la Figura 23.

²³ Referencia 13. Kendall y Kendall. *Análisis y Diseño de Sistemas*, Prentice Hall, México, 1991, 881 pp.

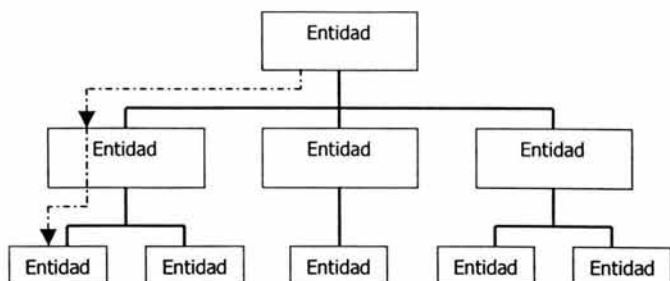


Figura 23. La recuperación de datos en una estructura jerárquica de datos se realiza con base en las ramificaciones.

- **Red.** Una estructura reticular permite que cualquier entidad cuente con cualquier número de subordinados o de superiores. En la Figura 24 se ilustra una estructura de red. Las entidades se conectan mediante el uso de enlaces de red, los cuales son datos comunes a ambas entidades conectadas.

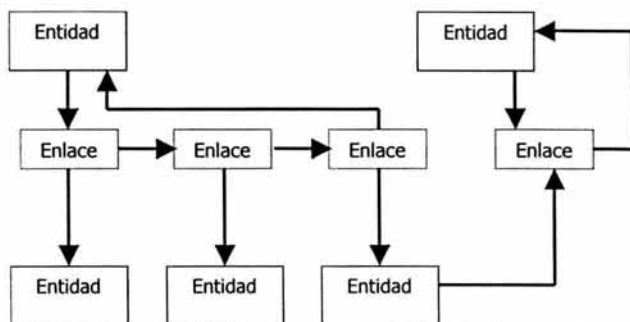


Figura 24. Las estructuras en red permiten que la entidad cuente con numerosos subordinados o superiores, y las entidades se conectan por medio de enlaces comunes.

- **Relacional.** Una estructura relacional consiste en una o más tablas bidimensionales, las cuales se refieren como relaciones. Los renglones de las tablas representan los registros y las columnas contienen los atributos (Ver Figura 25).

Tabla: Proveedores	
Id Proveedor	Nombre Empresa
1	IBM
2	DELL
3	HEWLET PARCKARD
4	COMPAQ

Tabla: Productos		
Nombre del Producto	Id Proveedor	Unidades en existencia
Computadora	2	56
Estación de trabajo	4	23
Estación de trabajo	3	14
Computadora	1	5

Figura 25. En una estructura de datos relacional, los datos se almacenan en varias tablas.

1.3.6.1. Base de datos relacional.

Una base de datos relacional es una herramienta para el mantenimiento y la manipulación de la información de una forma eficiente y efectiva, en el sentido de que los datos se protegen frente a pérdidas o daños accidentales, no utilizan más recursos (humanos o informáticos) de los necesarios y se pueden recuperar de diferentes formas dentro de unos límites de rendimiento aceptables. La base de datos en sí misma es la realización física de un modelo relacional, que es una forma de describir algunos aspectos del mundo real de acuerdo con una serie de reglas que fueron propuestas por el Dr. E. F. Codd, a finales de los años sesenta.

La ventaja principal del enfoque relacional está en la simplicidad de su representación en la estructura lógica de la base de datos y en la flexibilidad para establecer relaciones de datos por medio de campos de conexión. Todas las entidades en una base relacional están representadas como tablas separadas y no están colocadas en ninguna jerarquía fija. En esta base de datos, sólo se necesita definir la estructura de cada tabla, pero también las relaciones entre cada tabla, es decir, una tabla es un solo almacén de información relacionada. Por ejemplo, se podría rastrear toda la información acerca de los estudiantes en una escuela en una tabla de estudiantes. Entonces si se crearon tablas separadas conteniendo detalles sobre los maestros, clases y salones de clases, se podrían combinar las cuatro tablas en una base de datos programada (Ver Figura 26).

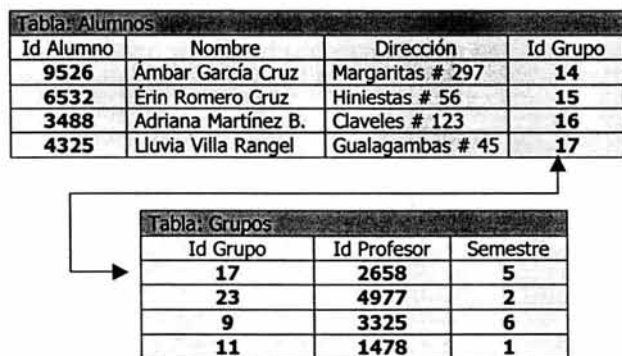


Figura 26. Base de datos relacional

En teoría, una base de datos relacional se puede crear desde cero, pero lo normal será utilizar los servicios de un sistema de gestión de bases de datos.

1.3.7. Sistemas de Gestión de Bases de Datos.²⁴

Las bases de datos pueden organizarse de muchas maneras, sin embargo, el aumento de éstas ha llevado a generar técnicas de organización, recomendaciones y hasta leyes para el uso adecuado y fidelidad de la información almacenada.

Para poder usar las bases de datos de la mejor manera debemos conocer una herramienta que nos permita almacenar y consultar información, estas herramientas son llamadas Sistemas de Gestión de Bases de Datos (SGBD), también llamados Sistemas Manejadores de Bases de Datos y Sistemas de Administración de Bases de Datos.

El antecedente de los manejadores de bases de datos es lo que se conoce como administrador de bases de datos, el cual era elaborado por un programador, lo que representaba una gran dependencia de los especialistas

²⁴ Referencia 7. De Miguel Castaño, Adoración; Piattini Velthius, Mario Gerardo. E.U.A., 1993, 989 pp.

en computación. Actualmente, no se requieren para manejar bases de datos conocimientos de programación, únicamente de los elementales para el uso de las computadoras.

Se entiende como sistema de gestión de bases de datos un conjunto de programas que permite el acceso a diferentes bases de datos para modificar, actualizar, imprimir, exportar e importar los mismos. Es una colección de numerosas rutinas de software interrelacionadas, cada una de las cuales es responsable de alguna tarea específica.

Las funciones principales de un SGBD son:²⁵

- i) crear y organizar la base de datos
- ii) establecer y mantener las trayectorias de acceso a la base de datos, de tal manera que los datos en cualquier parte de la base se puedan acceder rápidamente.
- iii) manejar los datos de acuerdo con las peticiones de los usuarios
- iv) mantener la integridad y seguridad de los datos
- v) registrar el uso de las bases de datos

El SGBD interpreta y procesa las peticiones del usuario para recobrar información de la base. Las preguntas a la base pueden tener distintas formas, pueden teclearse directamente desde la terminal, o codificarse como programas en lenguajes de alto nivel y presentarse para procesamiento interactivo o por lotes. En la mayoría de los casos, una petición de consulta tendrá que atravesar varias capas de software en el SGBD y en el sistema operativo antes de que pueda acceder la base de datos física. El SGBD responde a una pregunta llamando a los subprogramas apropiados, cada uno de los cuales realizará su función especial para interpretar la petición o localizar los datos deseados en la base y presentarlos en el orden solicitado. Así, el SGBD protege a los usuarios de la tediosa programación que tendrían que hacer para organizar el almacenamiento de los datos, o accederlos una vez almacenados.

Como ya se mencionó, una base de datos consiste en un grupo de archivos interrelacionados con distintos tipos de registros, y el SGBD permite a los usuarios el acceso a datos en cualquier parte de la base sin necesidad de conocer su organización dentro del dispositivo de almacenamiento.

1.3.7.1. El SGBD como interfaz entre el usuario y la base de datos.

En toda organización se suelen distinguir tres niveles de gestión: operacional, táctico y estratégico, de modo que el sistema de información estará integrado por tres subsistemas estructurados jerárquicamente y que se corresponden con cada uno de estos tres niveles. Ver figura 27.

La desconexión que muchas veces se produce entre estos tres subsistemas de información (los cuales constituyen a menudo sistemas aislados, o incluso grupos de aplicaciones sin relación alguna) aumenta el coste global de creación y mantenimiento del sistema de información y produce redundancias e incoherencias; es decir, impide una gestión racional de los datos, que debería estar encaminada a obtener el máximo rendimiento de los mismos, tal como se hace con cualquier otro recurso de la empresa.

²⁵ Referencia 27. Tsai, Alice Y. H. México, 1990, 607 pp.

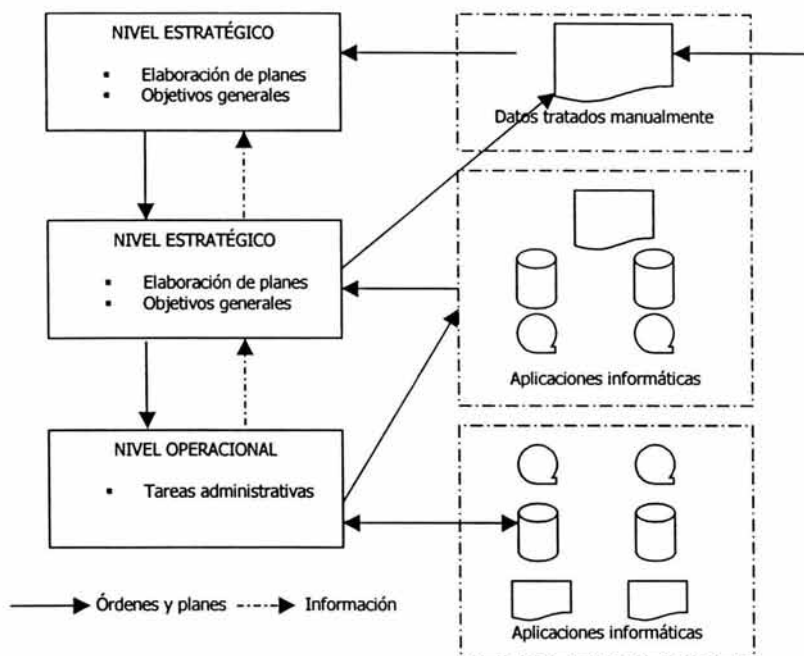


Figura 27. Niveles de gestión en las organizaciones y obtención de información mediante grupos de aplicaciones no integradas (cada una con sus propios datos).

La base de datos, como depósito único de datos para toda la organización, debe ser capaz de integrar los distintos subsistemas y aplicaciones atendiendo las necesidades de los usuarios en los tres niveles; el SGBD ha de suministrar la interfaz entre el conjunto de datos y dichos usuarios (Ver Figura 28).

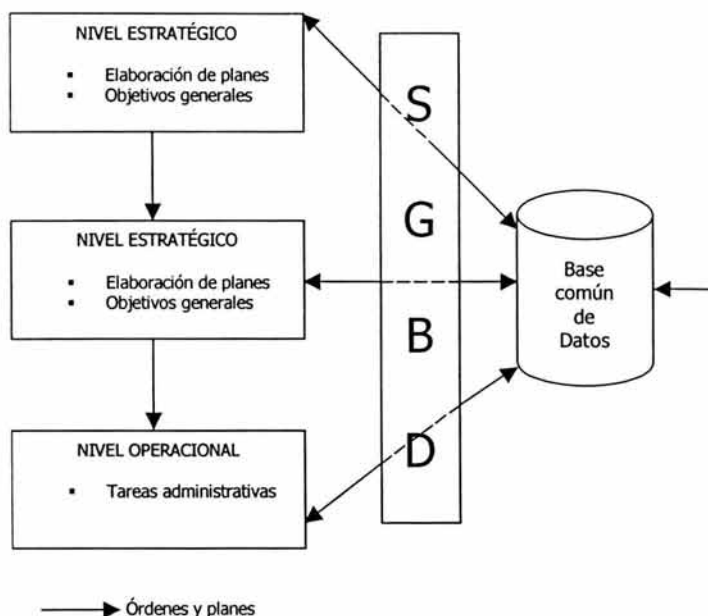


Figura 28. Niveles de gestión en las organizaciones u obtención de información mediante un sistema integrado (a partir de una base común de datos).

Además de estos usuarios de la base de datos que se podrían llamar finales, existen también, otros usuarios como son los analistas y programadores, cuyas exigencias son diferentes y a los que el SGBD ha de proporcionar también el correspondiente interfaz para que puedan acceder a la base. El administrador de la base de datos precisará asimismo que el SGBD le facilite sus tareas, procurándole un conjunto de medios que le permitan el desarrollo de sus funciones de forma eficaz.

En resumen, se puede decir que la finalidad del SGBD es establecer el interfaz necesario entre los diferentes tipos de usuarios y la base de datos.

Un SGBD de bases de datos da herramientas para:

- Diseñar la estructura de la base de datos.
- Crear formas de entrada de datos para que se pueda ingresar información a la base de datos.
- Validar los datos ingresados y checar inconsistencias.
- Ordenar y manipular los datos en la base de datos.
- Consultar la base de datos (qué es, preguntar algo acerca de los datos).
- Producir reportes flexibles, tanto en pantalla como en papel, que se haga fácil comprender la información almacenada en la base de datos.
- La mayoría de los programas de bases de datos avanzados han construido lenguajes de programación o macro, los cuales te permitirán automatizar muchas de sus funciones.

1.3.7.2. Microsoft Access

Microsoft Access es un programa para la gestión de información, es un sistema gestor de bases de datos relacionales. Permite diseñar las estructuras para almacenar la información y los medios para su introducción y explotación.

Microsoft Access revolucionó el mundo de las bases de datos con su facilidad para guardar y recuperar información. Los programas de bases de datos sencillos, como por ejemplo, el componente Base de datos de Microsoft Works, almacena la información en una única tabla, lo que se conoce como archivo plano (flat file). Con frecuencia, estas bases de datos sencillas se denominan bases de datos planas; en cambio los programas de base de datos más complejos, como Microsoft Access, pueden almacenar la información en varias tablas relacionadas, permitiendo crear bases de datos relacionales. Si la información de una base de datos relacional está organizada de forma correcta, podrá tratar estas tablas como una única área de almacenamiento y extraer de forma electrónica cualquier información de las diferentes tablas que pueda necesitar.²⁶

Este SGBD relacionales es una de las herramientas disponibles para la creación de aplicaciones de bases de datos en el escritorio. Access utiliza un potente lenguaje de programación, Visual Basic (el mismo lenguaje que se encuentra en otras aplicaciones de Microsoft Office y que se utiliza para desarrollar aplicaciones para Microsoft Windows en la Web). Utilizando técnicas de programación se puede aprovechar todo el potencial de Access de varias formas nuevas, creando rápidamente soluciones personalizadas. Aún más, se pueden crear interfaces de usuario a prueba de errores.

Microsoft Access contiene dos tipos de archivos, las bases de datos y los proyectos. La base de datos de Access es una colección de datos y objetos relacionados con una finalidad o tema en particular. Una base de datos de Microsoft Access contiene tablas. Una tabla es tan sólo uno de los tipos de objeto con los que se puede trabajar en Access. Otros objetos son: Consultas, Informes, Páginas de acceso a datos, Formularios, Macros y Módulos. De todos estos, sólo se utilizan las tablas para almacenar la información. Los demás objetos se utiliza para gestionar, tratar, analizar, recuperar, visualizar o publicar la información de las tablas. Es decir, para hacer que la información sea lo más accesible y útil posible. El motor de base de datos Microsoft Jet administra los datos de las tablas que residen en la base de datos. Los datos de tablas vinculadas pueden residir en otra base de datos de Access, en un origen de datos externo como, por ejemplo, dBASE o Microsoft Excel, o en un origen de datos ODBC como Microsoft SQL Server.

El Proyecto de Microsoft Access es un archivo de Access que se conecta a una base de datos de Microsoft SQL Server mediante la arquitectura de componentes OLE DB. Un proyecto de Access contiene únicamente objetos de base de datos basados en códigos o en HTML: formularios, informes, macros, módulos y accesos directos a páginas de acceso a datos. Un proyecto de Access no contiene datos ni objetos basados en definición de datos: tablas, vistas, diagramas de base de datos o procedimientos almacenados. Estos objetos de base de datos se almacenan en la base de datos de SQL Server. Los proyectos de Access se utilizan para crear aplicaciones cliente/servidor.

En el desarrollo de este trabajo se utilizará Microsoft Access.

1.3.7.3. Arquitectura Cliente-Servidor

El advenimiento de los sistemas de gestión de bases de datos relacionales fue, realmente, la tecnología clave que hizo posible la arquitectura informática cliente-servidor. El SGBD relacional servía como almacén centralizado para los datos de la organización y estaba diseñado para gestionar el acceso multiusuario a un conjunto compartido de datos. Toda la gestión de bloqueos y conexiones es llevada a cabo por el SGBD relacional, que también se encarga de la seguridad. El lenguaje SQL (Structured Query Language) fue creado como lenguaje universal de programación para solicitar datos específicos a un SGBD relacional.

La arquitectura cliente-servidor fue realmente una combinación de las mejores funciones del entorno basado en host y del entorno de red de área local basado en computadora personal. Esta arquitectura utiliza la potencia de la computadora para realizar la presentación de los datos, junto con el complicado procesamiento relativo a la lógica de negocio que añade valor a dichos datos. El SGBD relacional proporciona un área de almacenamiento centralizada para los datos, así como los servicios para gestionar el acceso compartido y concurrente a dichos

²⁶ Referencia 5. Callahan, Evan. España, 2002, 418 pp.

datos. La arquitectura cliente-servidor puede tomar muchas formas, dependiendo de cómo se elija separar los niveles de presentación, lógica de negocio y datos.

1.3.7.4. Microsoft SQL Server

Microsoft SQL Server es un servidor de base de datos relacionales que permite guardar información de forma estructurada y brinda rápido acceso a la misma. Este producto es utilizado por muchos sitios de Internet, además de aplicaciones empresariales, para que sus programas almacenen los datos que procesan.

SQL Server es una base de datos relacional destinada a aceptar aplicaciones con arquitectura Cliente-Servidor, en el que la base de datos reside en un computador central llamado servidor y cuya información es compartida por diversos usuarios que ejecutan las aplicaciones en sus computadoras locales o clientes. Dicha arquitectura propicia una mayor integridad de los datos, pues todos los usuarios trabajan con la misma información.²⁷

Por medio de las reglas del negocio, pueden imponerse controles aplicables a todos los usuarios, con relación a la información que se adiciona a la base de datos. La arquitectura Cliente-Servidor reduce considerablemente el tráfico en la red, ya que retorna al usuario sólo los datos solicitados.

Por ejemplo, si en una base de datos que tiene cien mil registros se hace una consulta cuyo resultado es de sólo tres registros, solamente estos tres registros serán enviados por la red a la máquina cliente. En un sistema tradicional, los cien mil registros viajarían por la red.

Las tareas de mantenimiento como realizar copias de seguridad y recuperación son de ejecución mucho más sencilla, puesto que los datos están en un lugar único. Véase en la Figura 29 un esquema básico del uso de la arquitectura Cliente-Servidor.

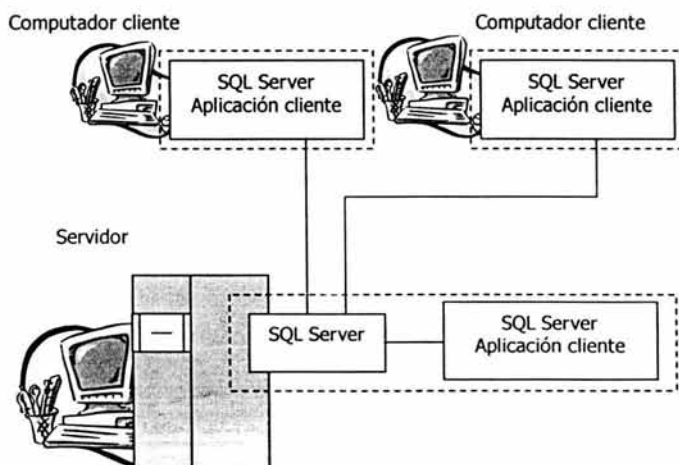


Figura 29. Esquema básico de la Arquitectura Cliente-Servidor.²⁸

Las necesidades y requisitos del cliente han dado lugar a innovaciones significativas en SQL Server, entre las que se incluyen la facilidad de uso, escalabilidad y fiabilidad, y almacenamiento de datos.

²⁷ Referencia 6. Coffman, Gayle. España, 1999, 909 pp.

²⁸ Referencia 19. Ramalho, José Antonio. Colombia, 2000, 580 pp.

1.4. Procesos de producción de hidrocarburos.

El petróleo líquido, como se produce en la cabeza del pozo, generalmente está asociado con gas natural y contaminado con agua y sólidos en suspensión. Del crudo deben eliminarse la mayor parte de esas impurezas antes de enviarse al mercado o a los centros de refinación.

El gas natural se separa de petróleo crudo en las Baterías de Separación, para después aplicar a cada fase, los procesos de tratamiento y acondicionamiento que se requieran para su distribución y venta. Todos los tratamientos del petróleo hasta antes de ser enviados a las refinerías son llamados procesos de producción de hidrocarburos.

1.4.1. El sistema de producción de hidrocarburos.

El sistema de producción de hidrocarburos está conformado por el yacimiento, el pozo de producción y las instalaciones de producción.

El yacimiento es la acumulación de aceite y/o gas en roca porosa tal como arenisca. Un yacimiento petrolero normalmente contiene tres fluidos (aceite, gas y agua) que se separan en secciones distintas debido a sus gravedades variantes. El gas siendo el más ligero ocupa la parte superior del yacimiento, el aceite la parte intermedia y el agua la parte inferior.

El pozo es un agujero perforado en la roca desde la superficie de un yacimiento a efecto de explorar o para extraer aceite o gas. El pozo y las instalaciones de producción representan la infraestructura construida por el hombre para la extracción, control, medición, tratamiento y transporte de los hidrocarburos extraídos de los yacimientos. La Figura 30 muestra las distintas etapas del proceso de producción de hidrocarburos.

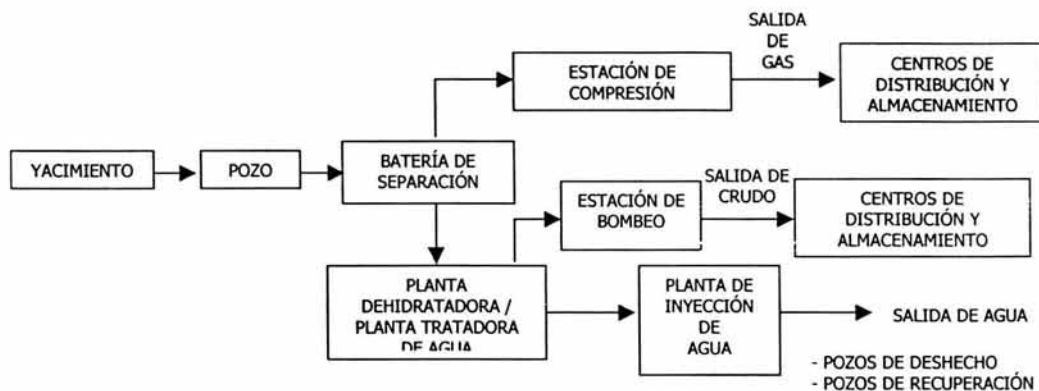


Figura 30. Proceso de producción de hidrocarburos.²⁹

Dentro de las instalaciones de producción que se encuentran en la superficie, podemos mencionar:

- Baterías de separación. En esta instalación se realiza la separación de la fase líquida y gaseosa del hidrocarburo.
- Planta deshidratadora. En este proceso se separa el agua contenida en el aceite, por medio de sistemas térmicos y químicos.

²⁹ Referencia 28. Villanueva Espinosa, Francisco Javier. México, D.F., Noviembre de 1998, 235 pp.

- Planta de inyección de agua. Esta planta se encarga de inyectar el agua recuperada por la planta deshidratadora al yacimiento.
- Estación de bombeo. Esta instalación se encarga de bombear el petróleo crudo hacia los centros de distribución y almacenamiento.
- Estación de compresión. En similitud a la estación de bombeo, la estación de compresión se encarga de comprimir el gas para desplazarlo hacia los centros de distribución y almacenamiento.
- Centros de distribución y almacenamiento. Son las instancias u organismos en donde se almacenan y desde los que se distribuyen los componentes separados de la mezcla de fluidos del pozo, como son, Pemex-Gas y Petroquímica Básica, Pemex-Refinación, entre otros.

La aplicación de un enfoque integral a los problemas u oportunidades permite comprender el real funcionamiento de los procesos de producción y así encontrarnos en mejor posición para decidir. Este principio es aplicable para cualquier tipo de organización. Si a este enfoque adicionalmente le incorporamos una herramienta de simulación, sus efectos se verán potenciados. La simulación es una interfaz desarrollada para computadora personal que permite al usuario interactuar y vivenciar un contexto específico y experimentar, sin riesgos, las consecuencias de sus decisiones. Las situaciones que pueden ser creadas son de la índole más variada y proporcionan a la persona o equipo que las utiliza importantes posibilidades de aprendizaje.

1.5. Simulación de sistemas.

La Simulación consiste básicamente en construir modelos informáticos que describen la parte esencial del comportamiento de un sistema de interés, así como en diseñar y realizar experimentos con el modelo y extraer conclusiones de sus resultados para apoyar la toma de decisiones. Típicamente, se utiliza en el análisis de sistemas tan complejos que no es posible su tratamiento analítico o mediante métodos del análisis numérico. Sus orígenes están en los trabajos de Student para aproximar la distribución que lleva su nombre, y los métodos que Von Neumann y Ulam introdujeron para resolver ecuaciones integrales. Desde entonces, la Simulación ha crecido como una metodología de experimentación fundamental en campos tan diversos como la Economía, la Estadística, la Informática o la Física, y con enormes aplicaciones industriales y comerciales, como los simuladores de vuelo, los juegos de simulación, o la predicción bursátil o meteorológica.³⁰

1.5.1. Simulación frente a soluciones analíticas

Inicialmente el uso de las ecuaciones y sistemas de ecuaciones diferenciales, han sido las herramientas más usadas para modelar sistemas. Para resolverlas, se han reducido muchas veces a sistemas lineales, siendo el álgebra vectorial (cálculo matricial, teoría de autovalores, etc..) , el análisis funcional y la teoría general de operadores herramientas útiles que con ciertas condiciones simplificadoras, se han mostrado eficaces en el estudio de la física, economía, biología, sociología, etc. Pero, ya en el siglo XIX este impresionante arsenal matemático, base de toda la ciencia clásica, empezó a mostrar su insuficiencia ante dos circunstancias que afectan al estudio de los sistemas : la incertidumbre y la imprecisión. Para la primera el Cálculo de Probabilidades y la Estadística con sus diversas ramas (Teoría de la Estimación y de los Procesos Aleatorios, los modelos de Previsión, el Análisis Multivariante, etc..) han dado solución a muchos problemas, pero su correcta aplicación está sujeta a fuertes restricciones teóricas y prácticas.

La cada vez mayor complejidad de los problemas y el estudio de sistemas abiertos hacen que la formalización matemática clásica sea insuficiente. Así han surgido otros artificios, como la caja negra, procedente de la teoría de regulación automática (cibernética y robótica), la topología de redes, la algorítmica y las técnicas de la matemática discreta, la programación matemática (lineal, cuadrática, dinámica) y todas aquellas técnicas que conforman la Investigación Operativa o Investigación de Operaciones (problemas de transporte, secuenciación óptima de actividades y tareas, la teoría de la decisión y de los juegos, etc..)

³⁰ Referencia 21. Ríos Insua, David; Ríos Insua, Sixto; Martín, Jacinto. España, 1997, 371 pp.

Es obvio señalar que el uso de estas herramientas se ha visto potenciado por el uso generalizado de la computadora. La computadora ha permitido también el tratamiento de problemas para los que la formalización matemática del sistema, tanto en sus elementos deterministas como en los aleatorios, es muy difícil o imposible. Esta dificultad se ha reducido en gran medida gracias al uso cada vez más general de las llamadas técnicas de simulación, las cuales permiten el modelado y estudio de muchos sistemas de estructura compleja, obteniendo soluciones numéricas aproximadas.

1.5.2. Naturaleza de la simulación.³¹

El concepto de simulación descansa sobre la realidad o los hechos (Ver Figura 31). Pocas personas comprenden totalmente cada uno de los aspectos de una situación compleja. Es por eso que se desarrollan teorías que pueden concentrar la atención en sólo una parte de un todo complejo. En algunas situaciones, esto permite construir modelos que ayuden a probar o representar una teoría. La simulación es el empleo de un modelo para tratar de identificar o reflejar el comportamiento de personas, procesos o sistemas reales.

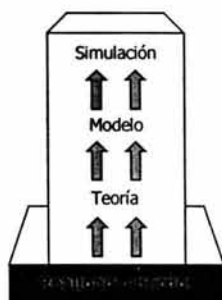


Figura 31. El concepto de simulación.³²

A partir del advenimiento de la computadora digital a principios de la década de 1950 se ha creado una gran cantidad de nuevos y poderosos instrumentos analíticos que han tenido un impacto profundo y duradero en gran número de disciplinas científicas. Una clase de procedimientos analíticos se refiere a las *técnicas de simulación en computadoras*.

Estas técnicas utilizan las computadoras para imitar, o *simular*, las operaciones de varios tipos de procesos del mundo real. El proceso de interés es comúnmente llamado un *sistema*, y para estudiarlo científicamente, con frecuencia se tiene que hacer un conjunto de suposiciones acerca de cómo trabaja. Estas suposiciones, las cuales usualmente toman la forma de relaciones lógicas o matemáticas, constituyen un *modelo* que es usado para tratar de obtener algún entendimiento de cómo es el comportamiento del sistema.

Si las relaciones que componen el modelo son lo suficientemente simples, podría ser posible usar métodos matemáticos (como álgebra, cálculo, o teoría de probabilidad) para obtener información *exacta* a preguntas de interés; esto es llamado una solución *analítica*. Sin embargo, la mayoría de los sistemas del mundo real son demasiado complejos para permitir que modelos realistas sean evaluados analíticamente, y estos modelos deben ser estudiados por medio de la simulación. En una *simulación* se usa una computadora para evaluar un modelo *numéricamente*, y los datos son reunidos para *estimar* las deseadas auténticas características del modelo. A pesar de esto, es sabido que para diseñar y llevar a cabo experimentos de simulación en computadoras se deben tener conocimientos mínimos de estadística matemática, teoría de la probabilidad, ecuaciones diferenciales, técnicas del diseño de experimentos y programación de computadoras.

Como un ejemplo del uso de simulación, considérese una compañía manufacturera que está contemplando construir una gran extensión en una de sus plantas pero no está segura si el potencial ganado en productividad

³¹ Referencia 14. Law, Averill M.; Kelton, W. David. USA, 2000, 760 pp.

³² Referencia 22. Sanders, Donald H. México, 1990, 887 pp.

justificaría el costo de la construcción. Ciertamente no sería un costo efectivo construir la extensión y entonces después quitarla si éste no trabaja bien. Sin embargo, un estudio de simulación cuidadoso podría aclarar un poco la pregunta al simular la operación de la planta como existe actualmente y ver qué pasaría si la planta fuera expandida. En este tipo de modelos tienen que incorporarse docenas de variables que complican la situación, como datos y suposiciones acerca del tamaño actual y potencial del mercado total y la parte actual y potencial de este mercado total que acapara la compañía.

Las áreas de aplicación de la simulación computadoras como un medio de análisis son numerosas y diversas e incluyen campos tales como: medicina, física, ingeniería, tecnología espacial, ciencias sociales, administración de empresas y economía, por mencionar sólo unos cuantos. Algunos problemas específicos para los que la simulación ha sido encontrada una herramienta útil y poderosa son:

- Análisis y diseño de sistemas manufactureros.
- Sistemas de evaluación de armas militares o sus requerimientos logísticos.
- Determinación de requerimientos de hardware o protocolos de redes de comunicación.
- Determinación de requerimientos de hardware y software para un sistema de cómputo.
- Diseño y operación de sistemas de transportación como aeropuertos, carreteras, puertos y transporte subterráneo (metro).
- Evaluación de diseños para organizaciones de servicio como restaurantes de comida rápida, hospitales, y oficinas postales.
- Reingeniería de procesos de negocios.
- Determinación de órdenes de pólizas para un sistema de inventarios.
- Análisis financiero o sistemas económicos.

A pesar de que la simulación es una de las técnicas más ampliamente usadas en investigación de operaciones, ciencia y administración, si no es que la más usada, ha habido algunos impedimentos aún con la amplia aceptación y utilidad de la simulación. Primeramente, que los modelos usados para estudiar sistemas a gran escala tienden a ser muy complejos, y escribir los programas de computadoras para ejecutarlos podría ser una tarea verdaderamente ardua. Esta tarea ha sido hecha más fácil en años recientes por el desarrollo de excelentes productos de software que automáticamente proporcionan muchas de las características necesarias para un programa de modelos de simulación. Un segundo problema sin la simulación de sistemas complejos es que es requerido algunas veces, una gran cantidad de tiempo de la computadora. Sin embargo, esta dificultad está llegando a ser menos severa conforme las computadoras van siendo más rápidas y más baratas. Finalmente, parece ser una impresión desafortunada que la simulación sea sólo un ejercicio de un programa de computadora, y uno muy complicado. Consecuentemente, muchos estudios de simulación han sido compuestos de construcciones de modelos heurísticos, código, y una sola corrida del programa para obtener la respuesta. Esta actitud descuida el importante asunto de cómo un modelo codificado propiamente debe ser usado para hacer inferencias acerca del sistema de interés, y ha llevado sin duda a conclusiones erróneas que son obtenidas de muchos estudios de simulación.

1.5.3. Definición de la simulación en computadoras.³³

El verbo simular se ha puesto recientemente en boga de numerosas disciplinas científicas, para describir el viejo arte de la construcción de modelos. Aunque la palabra simulación se aplica a diversas formas de construcción de modelos, como los de la pintura y escultura del Renacimiento, los modelos a escala de aviones supersónicos y los modelos en computadora de los procesos cognoscitivos, tiene ahora un significado específico en las ciencias físicas y en las del comportamiento. El uso moderno de esta palabra se remonta hasta a fines de 1940, cuando von Neumann y Ulam acuñaron el término "análisis de Monte Carlo" para aplicarlo a una técnica matemática que usaban entonces para resolver ciertos problemas de protección nuclear que eran, o demasiado costosos para resolverse experimentalmente o demasiado complicados para ser tratados analíticamente. El análisis de Monte Carlo involucra la solución de un problema matemático no probabilístico, mediante la simulación de un

³³ Referencia 17. Naylor, Thomas H.; Balintfy, Joseph L.; Burdick, Donald S.; Chu, Kong. México, 1982, 390 pp.

proceso estocástico cuyos momentos o distribuciones de probabilidad satisfacen las relaciones matemáticas del problema no probabilístico.

Con el advenimiento de la computadora de gran velocidad, a principios de 1950, como ya se mencionó, la simulación tomó otro significado aún, ya que surgió la posibilidad de experimentar con modelos matemáticos (que describen algún sistema de interés) en una computadora. Por vez primera, los sociólogos encontraron que podrían realizar, como los físicos, experimentos controlados de laboratorio; sin embargo, usaron para esto computadoras electrónicas en lugar de dispositivos como el reactor nuclear. Al crearse la simulación en computadoras, surgieron incontables aplicaciones y con ello, un número mayor de problemas teóricos y prácticos.

Desafortunadamente, no ha habido un acuerdo, entre quienes usan el término *simulación*, respecto a su significado exacto. Para los que prefieren una definición de simulación estrictamente formal, la propuesta por C. West Churchman puede resultar satisfactoria, ya que admite las ambigüedades e inconsistencias inherentes al uso actual de la palabra y define la simulación como sigue:

"*x simula a y*" si y sólo si: (a), *x* y *y* son sistemas formales; (b), *y* se considera como el sistema real; (c), *x* se toma como una aproximación del sistema real; (d), las reglas de validez en *x* no están exentas de error.

Si bien la definición de Shubik no es tan precisa, parece la más apropiada para los propósitos de este (trabajo), porque es típica entre las definiciones más populares:

Simulación de un sistema (o un organismo) es la operación de un modelo (simulador), el cual es una representación del sistema. Este modelo puede sujetarse a manipulaciones que serían imposibles de realizar, demasiado costosas o imprácticas. La operación de un modelo puede estudiarse y con ello, inferirse las propiedades concernientes al comportamiento del sistema o subsistema real.

La simulación es, esencialmente, una técnica que enseña a construir el modelo de una situación real aunada a la realización de experimentos con el modelo. Esta definición es sumamente amplia y puede muy bien comprender situaciones aparentemente no relacionadas entre sí, como los simuladores de vuelo, juegos militares, juegos de gerencia, modelos físicos de ríos, modelos econométricos, diversos dispositivos eléctricos analógicos y pruebas de aeroplanos en túneles aerodinámicos. No obstante, en este (trabajo) nos interesará una definición mucho más restringida.

La definición de simulación se restringirá solamente a los experimentos con modelos lógicos o matemáticos; por consiguiente, todos los modelos físicos, verbales, iconográficos y analógicos, necesariamente serán excluidos. Por la omisión de tales modelos, sin embargo, no debe pensarse que son menos importantes que los modelos matemáticos, pues no es el caso. El hecho es que el principal interés de este trabajo radica en los experimentos de simulación que pueden realizarse en una computadora digital.

Definición:

Simulación, es una técnica numérica para conducir experimentos en una computadora digital, los cuales requieren ciertos tipos de modelos lógicos y matemáticos, que describen el comportamiento de un negocio o un sistema económico (o algún componente de ellos, en este caso un proceso) en períodos extensos de tiempo real.

A pesar de su utilidad la simulación no puede considerarse como una panacea capaz de resolver todo tipo de situaciones, aún contando con la ayuda de los lenguajes especializados para la simulación, o de los avances que han representado los entornos de software específico para simulación, con simuladores visuales, etc. La realización de un estudio de simulación puede comportar un esfuerzo y un consumo de recursos no despreciable en cualquiera de sus fases : definición del problema, recogida de información, construcción del modelo y programación del mismo, realización de los experimentos de simulación en computador, etc. Sistemas complejos

pueden conducir a programas largos y complejos que requieran cantidades importantes de recursos computacionales.

Sin embargo la simulación, por sus características, y por los desarrollos computacionales que se han conseguido en los últimos años, sigue presentando una serie de ventajas que no solo la convierten en el procedimiento más adecuado en muchos casos, sino que hacen que sea la única alternativa tecnológica en muchos otros. Esto resulta especialmente obvio en aquellos casos en los que las características del sistema que se pretenden estudiar hacen inviable, por razones físicas o de coste, la experimentación directa sobre el sistema. El mundo de la producción industrial, del tráfico, la aeronáutica, la industria del automóvil, etc. son claros ejemplos de esta situación. Incluso en aquellos casos en los que es posible la experimentación directa la simulación puede ofrecer ventajas tales como un coste inferior, tiempo, repeticiones y seguridad.

1.5.4. Uso y limitaciones de la simulación

La clave del éxito en un proyecto de simulación es muchas veces el saber cuándo podemos aplicar simulación para conseguir los objetivos que nos hemos marcado en el momento de plantearse el estudio de un sistema.

La simulación es una técnica de resolución de problemas de forma experimental y su uso debe aplicarse en los siguientes supuestos :

1. No existe una formulación matemática completa del problema, o no se han desarrollado aun los métodos analíticos para resolver el modelo matemático.
2. Existen los métodos analíticos, pero las hipótesis simplificadoras, necesarias para su aplicación, desvirtúan las soluciones obtenidas y su interpretación.
3. Los métodos analíticos existen, y en teoría están disponibles, pero los procedimientos numéricos son tan arduos y complejos que la simulación constituye un método más sencillo para obtener una solución.
4. Es deseable observar una historia simulada del proceso dentro de un horizonte temporal dado para poder estimar ciertos parámetros.
5. La simulación constituye la mejor alternativa por la dificultad de realizar experiencias en el contexto real.

Se citan algunas otras razones que hay que tomar en cuenta para utilizar la simulación en computadoras: ³⁴

- La simulación hace posible estudiar y experimentar con las complejas interacciones que ocurren en el interior de un sistema dado, ya sea una empresa, industria, economía o un subsistema de cualquiera de ellas.
- A través de la simulación se pueden estudiar los efectos de ciertos cambios informativos, de organización y ambientales, en la operación de un sistema, al hacer alteraciones en su modelo y observar los efectos de éstas en el comportamiento del sistema.
- La observación detallada del sistema que se está simulando, conduce a un mejor entendimiento del mismo y proporciona sugerencias para mejorarlo, que de otro modo no podrían obtenerse. Hay, en realidad, una buena evidencia de que los seres humanos tienen una gran capacidad para entender el funcionamiento de los sistemas complicados y pueden encontrar reglas de decisión casi óptimas, procedimientos de operación, etc., si tienen bastante experiencia con el sistema y éste es lo suficientemente estable.
- La experiencia que se adquiere al diseñar un modelo de simulación en una computadora, puede ser más valiosa que la simulación en sí misma. El conocimiento que se obtiene al diseñar un estudio de simulación sugiere, frecuentemente, cambios en el sistema en cuestión. Los efectos de estos cambios puede probarse, entonces, a través de la simulación, antes de implantarlos en el sistema real.
- La simulación de sistemas complejos puede producir un valioso y profundo conocimiento acerca de cuáles variables son más importantes que otras en el sistema y cómo ellos obran entre sí.

³⁴ Referencia 17. Naylor, Thomas H.; Balintfy, Joseph L.; Burdick, Donald S.; Chu, Kong. México, 1982, 390 pp.

- La simulación puede emplearse para experimentar con situaciones nuevas acerca de las cuales tenemos muy poca o ninguna información, con el objeto de estar preparados para alguna eventualidad.
- La simulación puede servir como una prueba de *preservicio* para ensayar nuevas políticas y reglas de decisión en la operación de un sistema, antes de tomar el riesgo de experimentar con el sistema real.
- Las simulaciones son valiosas algunas veces, ya que proporcionan una forma conveniente de dividir un sistema complicado en subsistemas, cualquiera de los cuales puede ser modelado por un analista o un equipo de expertos en esa área.
- La simulación permite estudiar los sistemas dinámicos, ya sea en tiempo real, tiempo comprimido o tiempo expandido.
- Cuando se presentan nuevos componentes de un sistema, la simulación puede emplearse para ayudar a descubrir los obstáculos y otros problemas que resulten de la operación del sistema.
- La simulación convierte a los especialistas en técnicos generales. Se obliga a los analistas a hacer una apreciación y a entender todos los aspectos del sistema, con el resultado de que las conclusiones serán menos susceptibles a la parcialidad por inclinaciones particulares y menos susceptibles de volverse impracticables dentro de la configuración del sistema.

Como limitaciones en la aplicación de la simulación en la resolución de problemas se pueden citar :

- La simulación es costosa en horas de desarrollo y de computador.
- Suele ser muy difícil la validación del modelo y de los resultados.
- La recuperación, análisis e interpretación de los resultados suele requerir personal con conocimientos estadísticos.
- La aceptación de los resultados requiere un elevado conocimiento del modelo empleado, por lo que es difícil su aceptación por personas no involucradas en el grupo que ha realizado la simulación.

1.5.5. Aplicaciones de la simulación

Los modelos necesarios para la realización de los experimentos de simulación no se utilizan exclusivamente para predecir el comportamiento de sistemas reales, sino que pueden ser empleados en otro tipo de tareas.

Algunas de estas tareas son las siguientes :

- Diagnóstico . El modelo se emplea como representación profunda del sistema, sobre el que es posible determinar las causas que generan una desviación respecto a un comportamiento teórico. En este tipo de aplicaciones es donde los modelos funcionales son especialmente importantes, dado que modelan directamente las funciones del sistema.
- Control basado en modelos. El modelo se emplea para determinar las posibles acciones a realizar sobre el sistema que conducirían al mismo a una determinada situación. Los modelos causales son especialmente importantes para ello, dado que representan los mecanismos de propagación de efectos en el sistema modelado.
- Optimización. El modelo se emplea para determinar situaciones del proceso en las que se logra una mejoría del rendimiento.
- Enseñanza. El modelo se utiliza para que una persona estudie el comportamiento del sistema al que modela. Este tipo de sistemas se han empleado en multitud de circunstancias.

En los negocios, por ejemplo, los gerentes pueden evaluar proyectos o estrategias propuestas mediante la construcción de modelos teóricos. En seguida pueden determinar lo que sucede con esos modelos cuando se presentan ciertas condiciones o cuando se prueban ciertas suposiciones. Así pues, la simulación, dentro de los sistemas de apoyo a decisiones, es un enfoque de prueba y error para resolver problemas, el cual puede ser muy útil en la planificación.

1.5.6. Modelos para simulación.³⁵

Un modelo es una abstracción de la realidad que captura la esencia funcional del sistema, con el detalle suficiente como para que pueda utilizarse en la investigación y la experimentación en lugar del sistema real, con menos riesgo, tiempo y coste.

Un modelo es un conjunto formado por otros dos conjuntos :

- Un conjunto de variables.
- Un conjunto de relaciones entre las variables del modelo.

Las variables del modelo representan habitualmente magnitudes físicas del sistema que se modela; las relaciones describen su comportamiento ante una cierta clase de situaciones.

Es posible realizar distintos tipos de clasificaciones de modelos. Por ejemplo según la naturaleza del sistema o del uso que se vaya a dar al modelo. En este caso se presenta un sistema de clasificación que resulta apropiado para los modelos matemáticos, que son los usados en las simulaciones en las que se basa el desarrollo de este trabajo.

- Modelos determinísticos. En los modelos determinísticos no se permite que las variables sean al azar, en tanto que se suponen relaciones exactas para las características de operación en lugar de funciones de densidad de probabilidad. Este tipo de modelos requieren poco procesamiento en computadoras y con frecuencia es posible resolverlos analíticamente, por medio de la utilización de técnicas de cálculo de máximos y mínimos.
- Modelos estocásticos. Aquellos modelos en los que por lo menos una de las características de operación está dada por una función de probabilidad, se denominan modelos estocásticos. La suficiencia de las técnicas analíticas para solucionar modelos estocásticos, se encuentra bastante restringida debido a que estos modelos son considerablemente más complejos que los modelos determinísticos. Por esta razón, la simulación es un método mucho más atractivo para analizar y resolver los modelos estocásticos. Este tipo de modelos también tienen interés desde el punto de vista de la generación de muestras de datos al azar, que se emplean, en las etapas de observación o prueba, de la investigación científica.
- Modelos estáticos o estacionarios. Son aquellos modelos que no toman en cuenta, explícitamente, a la variable tiempo. Aquí el tiempo ha sido abstraído del modelo, es decir, en el sistema no ocurre cambio alguno y el transcurso del tiempo no se considera en absoluto.
- Modelos dinámicos. Los modelos matemáticos que tratan de las interacciones que varían con el tiempo, se denominan modelos dinámicos. Son los sistemas en los que dado un conjunto de condiciones iniciales, se tiene la posibilidad de predecir el comportamiento del sistema en períodos subsiguientes (de tiempo).

En las simulaciones realizadas como base para el desarrollo de este trabajo de tesis, se utilizan los modelos estáticos y dinámicos.

La simulación de sistemas por computadora esta basada en una generalización del concepto de experimentación del método científico, según el cual en lugar de realizar los experimentos sobre el sistema real, se realizan sobre un modelo (estático o dinámico) que lo representa, de manera que si el modelo es una representación válida del sistema entonces los resultados de la experimentación con el modelo pueden transferirse al propio sistema.

Por supuesto, los modelos de simulación que se producen serán solamente tan buenos como los datos y las suposiciones que las personas alimenten a la computadora.

³⁵ Referencia 17. Naylor, Thomas H.; Balintfy, Joseph L.; Burdick, Donald S.; Chu, Kong. México, 1982, 390 pp.

1.6. Optimización.

La simulación de procesos tuvo su desarrollo de manera colateral con el auge de las computadoras. Con el advenimiento de equipo con mayor capacidad de almacenamiento y más rápido fue posible realizar en menor tiempo las largas rutinas de cálculo que se utilizan para simular los procesos. Su aplicación al entendimiento y mejoramiento de los procesos ha producido beneficios tangibles a la industria. De manera paralela, la instrumentación y automatización de los procesos han redundado en operaciones más seguras y confiables con mayor control de calidad en los productos y disminución de los costos de operación y mantenimiento.

1.6.1. Definición de problemas de optimización.³⁶

Un problema de optimización comienza con un conjunto de parámetros o variables independientes, y con frecuencia incluye condiciones o restricciones que definen valores aceptables de las variables. Tales restricciones son llamadas los preceptos del problema. El otro componente esencial de un problema de optimización es una sola medida de bondad, llamada la *función objetivo*, la cual depende en alguna forma de las variables. La solución de un problema de optimización es un conjunto de valores permitidos de las variables para las cuales la función objetivo asume un valor "óptimo". En términos matemáticos, la optimización usualmente envuelve maximización o minimización; por ejemplo, se puede desear maximizar los beneficios o minimizar el peso.

Los problemas en todas las áreas de matemáticas, ciencia aplicada, ingeniería, economía, medicina, y estadística pueden ser puestos en términos de optimización. En particular, los modelos matemáticos son con frecuencia desarrollados para analizar y entender el fenómeno complejo. La optimización es usada en este contexto para determinar la forma y las características del modelo que corresponde más cercanamente a la realidad. Además, la mayoría de los procedimientos de toma de decisiones envuelven una solución explícita de un problema de optimización a ser la "mejor" opción.

1.6.2. Optimización de procesos.

La optimización es el proceso de búsqueda de la mejor solución posible de un conjunto de alternativas de un problema dado, es decir, es la búsqueda del valor óptimo operacional de una planta. Es la condición potencial de un sistema, que dada la forma en que se relacionan sus variables (funcionalidad), existe un valor extremo para la variable de interés, de tal manera que es posible llevar al sistema a un estado tal que maximiza o minimiza dicha variable, por ejemplo, el margen de ganancia en la producción de un producto, en este caso el producto son los hidrocarburos.

La teoría matemática de la optimización es actualmente un área madura. Es de amplia aplicación en el análisis en muchas áreas de la ingeniería. Sin embargo, es con el desarrollo de la tecnología y con el mayor grado de automatización en los procesos que se vuelve factible tomar datos directamente del proceso y llevar a cabo la optimización en línea/tiempo real de la operación de los mismos.

La optimización en línea/tiempo real es el proceso de búsqueda de la mejor condición de funcionamiento de un sistema, servicio o producto contemplando la reducción de costos de operación y tiempo de ejecución. El término en línea implica un contacto continuo con el proceso a través del intercambio de información, en tanto que un sistema en tiempo real debe estar en contacto con el sistema físico, proporcionar la respuesta adecuada cumpliendo con las restricciones de tiempo que la dinámica del sistema físico demande. Los sistemas en línea no interactúan directamente con el mundo físico. La optimización en línea/tiempo real se refiere a que en un sistema de optimización, no todas las tareas involucradas cumplen con las restricciones para tiempo real.

³⁶ Referencia 10. Gil, Philip E.; Murray, Walter; Wright, Margeret H. Great Britain, 1995, 401 pp.

Para que un proceso reporte beneficios con la aplicación de esta tecnología, se requiere mantener altos factores de servicio, tener cambios en su ritmo de producción debido a cambios en gastos de alimentación, en la composición, demanda o suministro de energía, ser procesos complicados con muchas variables, con diseños altamente integrados y gran número de restricciones de operación y, finalmente que pueda ser desarrollado un modelo matemático representativo del proceso.

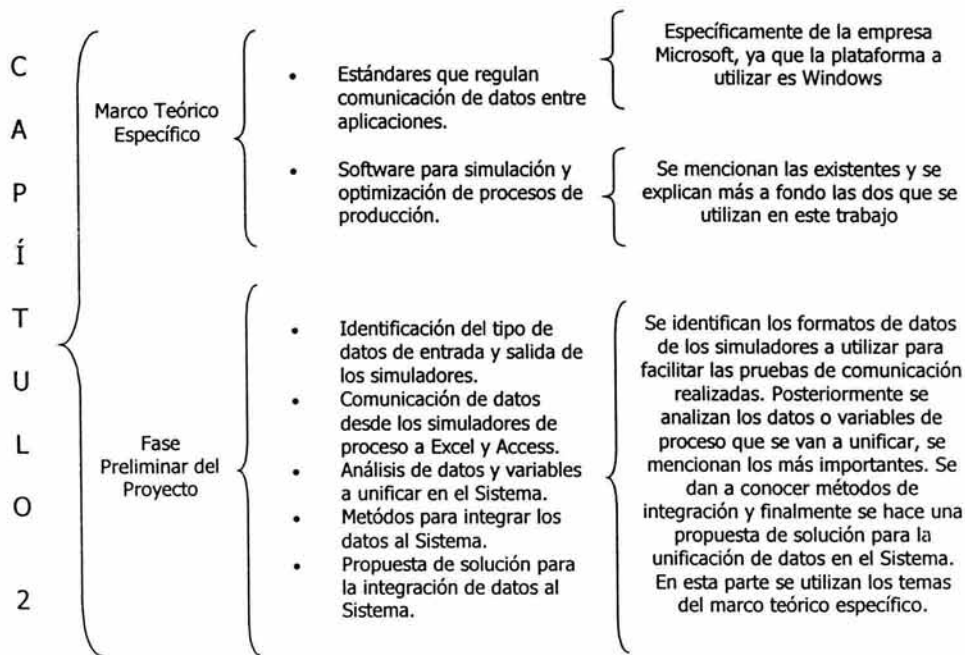
CAPÍTULO 2

MARCO TEÓRICO ESPECÍFICO Y FASE PRELIMINAR DEL PROYECTO

CAPÍTULO 2. MARCO TEÓRICO ESPECÍFICO Y FASE PRELIMINAR DEL PROYECTO

En este capítulo se da a conocer el marco teórico específico relacionado a los estándares de comunicación entre aplicaciones y a las herramientas de simulación y optimización de procesos de producción. Posteriormente se presenta la fase preliminar del proyecto, la cual se divide en cinco partes, en donde se realizaron pruebas y se determinó la manera en la que se integrarían los datos al sistema de información. Estas pruebas sirven para determinar la viabilidad técnica del funcionamiento y comportamiento de la interfaz desarrollada con el software de simulación.

Para un mejor entendimiento en el siguiente cuadro sinóptico se muestra cómo está conformado este capítulo y en qué consiste cada tema.



2.1. Estándares y normas que regulan la comunicación de datos entre diversas aplicaciones.³⁷

Para realizar la transferencia de datos entre aplicaciones se tienen que seguir ciertas normas establecidas ya sea por una institución reconocida o por la misma empresa que desarrolló el software. En este trabajo se utilizarán herramientas desarrolladas bajo Windows, por lo tanto el sistema de información se desarrollará en la plataforma Windows 2000. Por ello se presentan los estándares de comunicación principalmente de la empresa Microsoft Corporation.

³⁷ Referencia Sitios de Internet 9. <http://www.microsoft.com/com/wpaper/default.asp>

2.1.1. DDE. Intercambio Dinámico de Datos (Dynamic Data Exchange).³⁸

Microsoft Windows proporciona varios métodos para transferencia de datos entre aplicaciones. Un método es el uso del protocolo Intercambio Dinámico de Datos (Dynamic Data Exchange) o DDE. Muchas aplicaciones usan DDE, incluyendo Microsoft Excel, Word y Visual Basic. DDE está basada en el sistema de mensajería construido en Windows. DDE es un sistema de comunicación entre procesos (Inter-Process Communication) o IPC construido en el ambiente Windows y el cual permite que dos programas Windows compartan datos o envíen comandos entre sí, es decir, se comuniquen e inicien una "conversación" DDE. El protocolo DDE consiste en un conjunto de mensajes y directrices. Este envía mensajes entre aplicaciones que comparten datos y utiliza memoria compartida como una área de intercambio de datos común. Las aplicaciones pueden usar el protocolo DDE para transferir datos una sola vez y para intercambios continuos en los que las aplicaciones envíen actualizaciones a otras conforme nuevos datos lleguen a estar disponibles. Por ejemplo, cuando se cambia una forma en un programa de bases de datos o un dato en un programa de hoja de cálculo, se puede hacer que estas formas o datos también cambien en cualquier lugar en donde otro programa pueda estarlos usando.

DDE puede ser pensado como una relación de comunicación directa entre dos programas de aplicación, de hecho DDE utiliza un modelo cliente-servidor. En la mayoría de los casos, un programa está proporcionando algunos de los datos (tanto texto como gráficos) a otra aplicación. La aplicación que es la fuente de los datos que pueden ser útiles a otros programas Windows, o la que los proporciona es llamada "el servidor" y la aplicación que está recibiendo los datos, o la que los solicita es llamada "el cliente". Por ejemplo, DDE hace posible insertar un cuadro de hoja de cálculo en un documento creado con un procesador de palabras. Cada vez que los datos de la hoja de cálculo cambien, el cuadro en el documento cambia correspondientemente. Aunque el mecanismo DDE es usado aún por muchas aplicaciones, este está siendo suplantado por OLE, el cual provee mayor control sobre los datos compartidos.

2.1.1.1. Usos para el intercambio dinámico de datos de Windows.

DDE es el más apropiado para intercambios de datos que no requieren interacción continua con el usuario. Usualmente, una aplicación proporciona un método para que el usuario establezca el vínculo entre los datos de intercambio de aplicaciones, sin embargo, una vez que el vínculo es establecido, las aplicaciones intercambian datos sin más intervención del usuario.

DDE puede ser usado para implementar un amplio rango de características de aplicación, por ejemplo:

- Vincular datos en tiempo real, como instrumentos científicos o control de procesos.
- Crear documentos compuestos (compound), como un documento de procesamiento de palabras que incluya un cuadro o tabla producido por una aplicación gráfica. Utilizando DDE, el cuadro cambiará cuando los datos fuente sean cambiados, mientras el resto del documento permanecerá igual.
- Llevar a cabo consultas entre aplicaciones, como una consulta de una hoja de cálculo a una base de datos sobre deudas de cuentas pasadas.

2.1.1.2. DDEML. Librería de administración de intercambio de datos dinámicos. (Dynamic Data Exchange Management Library).³⁹

Windows también soporta la Biblioteca de Administración de Intercambio Dinámico de Datos (Dynamic Data Exchange Management Library) o DDEML. La DDEML es una biblioteca de vínculo dinámico (Dynamic-Link Library) o DLL que las aplicaciones pueden usar para compartir datos. La DDEML proporciona una interfaz de funciones y mensajes que simplifica la tarea de agregar capacidad DDE a una aplicación. En vez de enviar, mandar por correo, y procesar mensajes DDE directamente, una aplicación usa las funciones proporcionadas por

³⁸ Referencia Sitios de Internet 10. <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/winui/WinUI/WindowsUserInterface/DataExchange/DynamicDataExchange/AboutDynamicDataExchange.asp>

³⁹ Referencia Sitios de Internet 7. <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/winui/WinUI/WindowsUserInterface/DataExchange/DynamicDataExchangeManagementLibrary/AbouttheDDEML/BasicConcepts.asp>

el DDEML para administrar conversaciones DDE. Una conversación DDE es la interacción entre aplicaciones cliente – servidor.

La DDEML también proporciona la facilidad de administrar las cadenas y los datos que las aplicaciones DDE comparten. En vez de utilizar apuntadores para objetos de memoria compartida, las aplicaciones DDE crean e intercambian conjuntos de cadenas, que identifican cadenas, y conjuntos de datos, que identifican objetos de memoria. La DDEML también hace posible para una aplicación de servidor registrar los nombres de servicio que este soporta. Los nombres son transmitidos a otras aplicaciones en el sistema, el cual puede utilizar los nombres para conectarse al servidor. Además, la DDEML asegura compatibilidad entre aplicaciones DDE forzándolas a implementar el protocolo DDE de una manera constante.

Las aplicaciones existentes que utilizan el protocolo DDE basado en mensajes son completamente compatibles con aquellos que utilizan la DDEML. Esto es, una aplicación que utiliza DDE basado en mensajes puede establecer conversaciones y llevar a cabo transacciones con aplicaciones que usan la DDEML. Por muchas ventajas de la DDEML, las nuevas aplicaciones deben usarla en lugar de los mensajes DDE. Para usar los elementos de la interfaz de programación de aplicaciones (API) * de la DDEML, se debe incluir el archivo cabeza DDEML en los archivos fuente, vincularlo con la biblioteca DDEML, y asegurarse de que la biblioteca de vínculo dinámico DDEML está en la ruta de búsqueda del sistema.

A continuación se mencionan algunas características de la DDEML:

- Interacción Cliente – Servidor. El intercambio dinámico de datos siempre ocurre entre una aplicación cliente y una aplicación servidor, como ya se mencionó. La aplicación cliente DDE inicia el intercambio estableciendo una conversación con el servidor para enviar transacciones al servidor. Una transacción es una solicitud de datos o servicios. La aplicación servidor DDE responde a transacciones proporcionando datos o servicios al cliente. Por ejemplo, una aplicación gráfica debe contener una representación gráfica de barras de las utilidades trimestrales de una empresa, pero los datos para la gráfica de barras deben estar contenidos en una aplicación de hoja de cálculo. Para obtener las figuras de las utilidades más recientes, la aplicación de gráficas (el cliente) podría establecer una conversación con la aplicación de hoja de cálculo (el servidor). La aplicación gráfica podría entonces enviar una transacción a la aplicación de hoja de cálculo, solicitando las figuras de las utilidades más recientes.
- Conversación. Cuando dos aplicaciones están participando en DDE se dice que están entablando una conversación DDE. Una conversación DDE toma lugar entre dos ventanas, una para cada uno de las aplicaciones participantes. Una ventana puede ser la ventana principal de la aplicación; una ventana asociada con un documento específico, como en una aplicación interfaz de documentos múltiples o MDI (multiple-document interface); o una ventana invisible de quien el único propósito es procesar mensajes DDE.

Debido a que una conversación DDE es identificada por un par de manejadores (handles) a las ventanas que toman parte en la conversación, ninguna ventana debe estar participando en más de una conversación con otra ventana. Tanto la aplicación cliente como la aplicación servidor deben proporcionar una ventana diferente para cada una de sus conversaciones con una aplicación cliente o servidor en particular. Una aplicación puede asegurar que un par de ventanas cliente y servidor nunca estén involucradas en más de una conversación creando una ventana escondida para cada conversación.

* Funciones, mensajes, estructuras de datos, tipos de datos e instrucciones que se pueden utilizar para crear aplicaciones que se ejecuten bajo Microsoft Windows 95 o posterior.

Un servidor puede tener muchos clientes al mismo tiempo, y un cliente puede solicitar datos de múltiples servidores. Una aplicación puede también ser ambos, un cliente y un servidor, por lo que puede entablar varias conversaciones al mismo tiempo, actuando como el cliente en algunas y como el servidor en otras. Tanto el cliente como el servidor pueden terminar la conversación en cualquier momento.

- Administración de la conversación. Una conversación entre un cliente y un servidor está siempre establecida en la petición del cliente. Cuando una conversación está establecida, cada integrante recibe un manejador que identifica la conversación y utilizan este manejador en otras funciones de Biblioteca de Administración de intercambio dinámico de datos (DDEML) para enviar transacciones y administrar la conversación. Un cliente puede solicitar una conversación con un solo servidor, o puede solicitar múltiples conversaciones con uno o más servidores.
- Administración de datos. Debido a que DDE utiliza objetos de memoria para pasar datos de una aplicación a otra, la librería DDE (DDEML) proporciona un conjunto de funciones que las aplicaciones DDE pueden usar para crear y administrar objetos DDE.

2.1.2. CORBA (Common Object Request Broker Architecture).

CORBA es un producto promovido por empresas como Sun, Netscape e IBM que permite que partes de aplicaciones, llamadas objetos, se comuniquen con otras, sin importar en qué lenguaje de programación están escritas o en cuál sistema operativo están corriendo. CORBA es independiente de plataforma, es decir, con CORBA se pueden escribir aplicaciones compatibles con cualquier sistema.

El modelo de comunicación de CORBA utiliza *Proxies*, que hacen creer tanto al servidor como al cliente que el otro (cliente y servidor respectivamente) forma parte del mismo proceso. Al *proxy* del cliente se le denomina *Stub*, y al *proxy* del servidor *Skeleton*.

El verdadero intercambio de datos entre aplicaciones y computadoras se realiza a través del ORB (Object Request Broker). Éste desempeña la función de regular la comunicación entre las aplicaciones repartidas. No obstante, para encontrar el servidor apropiado sobre una petición del cliente, utiliza una utilidad especial *Smart Agent*. *Smart Agent* se debe ejecutar en una computadora de la red local.

El desarrollo de objetos CORBA se basa en la declaración e implementación de interfaces. Generalmente los servidores CORBA se inician manualmente. No obstante también es posible hacer que el servidor arranque automáticamente cuando un cliente intenta acceder a él, para hacer esto se deberán hacer los preparativos en CORBA para el arranque dinámico del servidor y habrá que modificar el código del cliente.

2.1.3. DLL (Dynamic-Link Library)

Una biblioteca de vínculo dinámico (DLL) es un archivo ejecutable que actúa como una biblioteca de funciones compartidas. El vínculo dinámico proporciona una forma para que un proceso llame a una función que no es parte de su código ejecutable. El código ejecutable para la función es localizado en una DLL, la cual contiene una o más funciones que son compiladas, vinculadas, y almacenadas separadamente de los procesos que las usan. Las DLLs también facilitan el compartimiento de datos y recursos. Múltiples aplicaciones pueden acceder simultáneamente a los contenidos de una sola copia de una DLL en memoria.

La DLL o archivo de código, contiene funciones que pueden ser llamadas desde otro código ejecutable (tanto una aplicación u otra DLL). Los programadores usan DLLs para reusar código y repartir distintos trabajos. A diferencia de un archivo ejecutable (EXE), una DLL no puede ser corrida directamente. Las DLLs deben ser llamadas desde otro código que está ejecutándose en ese momento. Las DLLs y EXEs aplican solo en ambientes Windows.

El sistema operativo mapea las DLLs en el espacio de dirección del proceso que se está llamando, esto puede ser cuando el proceso está empezando o mientras está corriendo. Ellas no necesitan ser compiladas con el programa principal. El conjunto de tales archivos (o la DLL) es un tanto comparable a las rutinas de biblioteca proporcionadas con lenguajes de programación tales como C y C++. Típicamente, una DLL proporciona una o más funciones particulares y un programa tiene acceso a las funciones creando un vínculo tanto estático como dinámico a la DLL. Un vínculo estático queda constante durante la ejecución del programa, mientras que un vínculo dinámico es creado por el programa cuando se necesita.

La vinculación dinámica difiere de la vinculación estática en que la dinámica permite a un módulo ejecutable (tanto un archivo .DLL como uno .EXE) incluir sólo la información necesaria en tiempo de ejecución para localizar el código ejecutable para una función DLL. En la vinculación estática, el vinculador obtiene todas las funciones referenciadas desde ésta biblioteca y las coloca con el código del usuario en el ejecutable. Utilizando DLLs en vez de bibliotecas de vínculo estático hace al tamaño del archivo ejecutable más pequeño. Si algunas aplicaciones utilizan la misma DLL, este puede ser un gran ahorro en espacio y memoria de disco.

Las DLLs también pueden contener únicamente datos. Los archivos DLL usualmente terminan con la extensión .dll, .exe, .drv, o .fon. Una DLL puede ser usada por varias aplicaciones al mismo tiempo. Algunas DLLs son proporcionadas con el sistema operativo Windows y disponibles para cualquier aplicación Windows. Otras DLLs son escritas para una aplicación en particular y son cargadas con la aplicación y de este modo podrían ser usadas para integración de una aplicación hecha en casa dentro de un sistema que soporta DLLs.

2.1.4. OLE. (Object Linking and Embedding)

La historia de OLE es la que concierne con la creación y administración de documentos compuestos, pero OLE es mucho más que eso.

OLE es una colección de poderosas tecnologías orientadas a objetos, de nivel sistema. OLE solía ser un acrónimo para incrustación y vinculación de objetos, una tecnología usada para desarrollar aplicaciones que soportan documentos compuestos (compound documents). Pero OLE ha progresado más allá de documentos compuestos y el término "OLE" ahora abarca un amplio conjunto de tecnologías. OLE es ahora un conjunto de sistemas orientados a objetos "interfaces y servicios" proporcionando una estructura estándar para construir componentes de software reutilizables e integrados. OLE es una arquitectura extensible y abierta construida en la base de COM.

OLE y las tecnologías que caen bajo la sombrilla de OLE (almacenamiento estructurado y persistencia de objetos, nombres inteligentes o "monikers", transferencia de datos uniformes, OLE Drag and Drop, OLE Automation, OLE Documents) se refieren a la integración entre componentes funcionales de todos tipos, donde sea que ellas puedan estar – en el sistema o dentro de aplicaciones, dentro de DLLs en procesos o en archivos .EXE fuera de procesos, y aún en módulos distribuidos a través de una red. Estos componentes vienen en muchas formas y medidas, y las interfaces proporcionadas en esos componentes varían también enormemente. En algunos casos, un componente particular está implementado por varios tipos de aplicaciones. Los componentes que son primariamente usuarios de interfaces implementadas en otros componentes son llamados "clientes" o "contenedores", dependiendo de lo que ellos hacen con esas interfaces. Los módulos que implementan componentes con interfaces son llamados "servidores".

La base para toda la integración es el Modelo de Objetos Componentes (COM). OLE utiliza COM como la tubería de bajo nivel que permite comunicación transparente entre componentes a través de un estándar binario de interfaces. Todos los componentes basados en COM son llamados *objetos componentes* porque ellos soportan nociones fundamentales de reutilización, encapsulación, y polimorfismo.*

* Por conveniencia, el uso de la palabra *objeto* por sí misma en estas tecnologías significa "objeto componente".

OLE por sí mismo proporciona la comunicación para trabajar con interfaces, define las reglas para su uso, e implementa algunos componentes como un estándar. OLE también proporciona funciones API que facilitan la interacción entre varios conjuntos de interfaces en diferentes componentes.

Como COM evoluciona, OLE se beneficiará automáticamente y será extendido a servicios de sistema de soportes diversos, desde acceso a bases de datos a servicios de mensajería, administración de sistemas y más.

2.1.5. COM. Modelo de Objetos Componentes (Component Object Model).

COM de Microsoft es una plataforma independiente, distribuida, y un sistema orientado a objetos para crear componentes de software binario que puedan interactuar. COM es la tecnología base para OLE de Microsoft (documentos compuestos), ActiveX (componentes para Internet), así como otros. Para entender COM (y por tanto todas las tecnologías basadas en COM), es crucial entender que éste no es un lenguaje orientado a objetos, sino un estándar. COM tampoco especifica cómo debe estar compuesta una aplicación (el lenguaje, la estructura, y los detalles de implementación se dejan al programador de la aplicación). En lugar de esto, COM especifica un modelo de objeto y los requerimientos de programación que permiten a objetos COM (también llamados componentes COM, o algunas veces simplemente objetos) interactuar con otros objetos. Estos objetos pueden estar dentro de un solo proceso, en otros procesos, y pueden aún estar en máquinas remotas. Pueden también haber sido escritos en otros lenguajes, y pueden ser completamente distintos en su estructura, lo cual es porque COM está asociado a un estándar binario – un estándar que se aplica después de que un programa ha sido traducido a código de máquina binario.

El único requerimiento en cuanto a lenguaje para COM es que el código sea generado en un lenguaje que pueda crear estructuras de apuntadores y, ya sea explícitamente o implícitamente, llamar a funciones a través de apuntadores. Los lenguajes orientados a objetos, tales como Microsoft Visual C++ y Smalltalk proporcionan mecanismos de programación que simplifican la implementación de objetos COM, pero lenguajes tales como C, Pascal, Ada, Java, y aún ambientes de programación Basic pueden crear y utilizar objetos COM.

COM (Component Object Model) es una especificación desarrollada por Microsoft con el cometido de facilitar la comunicación entre dos aplicaciones. COM se compone de dos partes:⁴⁰

- Un estándar binario que se limita a definir las reglas de comunicación entre aplicaciones (básicamente este estándar consta de una serie de interfaces que deben implementar los programas COM). Esta norma no está ligada a ningún lenguaje de programación, el lenguaje únicamente debe permitir la llamada de funciones a través de punteros. Para que dos aplicaciones puedan comunicarse entre sí, su código binario debe ser compatible con COM. Es irrelevante en qué lenguaje está escrito el código fuente de cada aplicación.
- La implementación de una serie de servicios compatibles COM integrados en el sistema operativo Windows (*Ole32.dll* y *OleAuto32.dll*).

2.1.5.1. ¿Qué hace COM?

En 1993 Microsoft introdujo COM como un mecanismo básico para la comunicación entre procesos bajo Windows.

OLE 2.0 fue la primera tecnología basada en COM para facilitar la comunicación entre procesos (las técnicas anteriores de intercomunicación DDE y OLE 1.0 todavía no se basaban en COM). OLE 2.0 permitía:

- Arrastrar y Colocar.
- Transferencias directas entre aplicaciones.
- Incrustar y vincular objetos (Object Linking and Embedding).

⁴⁰ Referencia 15. Louis, Dirk. España, 2000, 856 pp.

- Automatización.

Vincular e Incrustar. A través de este mecanismo aquellas aplicaciones que procesan archivos pueden integrar en sus archivos objetos de otras aplicaciones; bien sea tomando para sí y guardando los datos del objeto (Incrustar), o bien guardando un enlace al archivo del objeto (Vincular). Los procesadores de texto son un ejemplo típico, permiten al usuario incluir y guardar en sus documentos otros objetos junto al texto puro (gráficos, videos, etc.). En lugar de disponer de la funcionalidad para generar y editar esos objetos, la aplicación de texto llama a través de COM a otros programas especializados en esos objetos. Para ello es imprescindible que esas aplicaciones estén registradas bajo Windows como servidores OLE, y que el procesador de texto pueda funcionar como contenedor de los objetos de servidor OLE.

Wordpad es un ejemplo de contenedor OLE. A través de la opción *Insertar/Objeto* se pueden incluir objetos de todos los servidores OLE registrados en el sistema.

Los documentos en los que además de los datos nativos también se pueden almacenar objetos incrustados o enlaces a objetos vinculados, se denominan *documentos combinados*.

Para editar un objeto incrustado (o vinculado) el contenedor OLE llama al servidor OLE correspondiente. Este se inicia como una aplicación autónoma o bien se ejecuta directamente en el contenedor OLE, donde integra su menú y barras de herramientas dentro del marco del contenedor OLE.

Automatización. Se habla de automatización cuando una aplicación pone parte de su funcionalidad al servicio de otras en forma de objeto COM. Las aplicaciones que exportan parte de sus funciones de esta forma se denominan servidores de automatización y los programas que aprovechan esa funcionalidad (es decir, ejecutan métodos de los objetos COM), se llaman controladores o clientes de automatización.

2.1.5.2. Objetos y componentes COM.

COM define la naturaleza esencial de un objeto COM. En general, un objeto de software está compuesto de un conjunto de datos y las funciones que manipulan los datos. Un objeto COM es en el que se accede a un dato de objeto, y es realizado exclusivamente a través de uno o más conjuntos de funciones relacionadas. Estos conjuntos de funciones son llamados *interfaces*, y las funciones de una interfaz son llamadas métodos. Además, la única manera que requiere COM para obtener acceso a los métodos de una interfaz es a través de un apuntador a la interfaz. Aparte de especificar el estándar básico de objeto binario, COM define ciertas interfaces básicas que proporcionan funciones comunes a todas las tecnologías basadas en COM, y proporciona un número pequeño de funciones API que requieren todos los componentes. COM también define cómo trabajan juntos los objetos sobre un ambiente distribuido y tienen agregadas características de seguridad para ayudar a proporcionar integridad a componentes y sistemas.

La palabra *objeto* tiende a significar algo diferente para cada persona. Para clarificar, en COM, un objeto es alguna pieza de código compilado que proporciona algún servicio al resto del sistema. Para evitar confusión, es probablemente mejor referirse a un objeto COM como un "objeto componente" o simplemente un "componente". Esto evita confundir objetos COM con código fuente, objetos de programación orientada a objetos (OOP) tal como esos definidos en C++.

Los Objetos Componentes usualmente tienen algo asociado a los datos, pero a diferencia de los objetos C++, un objeto componente dado nunca tendrá acceso directo a otro objeto componente en su totalidad. En vez de esto, los objetos componentes siempre acceden a otros objetos componentes a través de apuntadores de interfaz. Esto es una característica de arquitectura primaria del Modelo de Objetos Componentes, porque esta permite a COM conservar la encapsulación de datos y procesamiento completamente, un requisito fundamental de un verdadero software de componentes estándar.

El Modelo de Objetos Componentes define algunos conceptos fundamentales que proporcionan argumentos estructurales del modelo. Estos incluyen:

- Un estándar binario para llamadas a funciones entre componentes.
- Una provisión para agrupar funciones en interfaces.
- Una interfaz base proporcionando:
- Una manera para que los componentes descubran dinámicamente las interfaces implementadas por otros componentes.
- Un conteo de referencia para permitir a los componentes rastrear su propio tiempo de vida y borrarse a ellos mismos cuando sea apropiado.
 - Un mecanismo para identificar únicamente componentes y sus interfaces.
 - Un "cargador de componentes" para establecer interacciones de componentes y adicionalmente en los casos de atravesar-proceso (cross-process) y atravesar-red (cross-network) ayudar a manejar las interacciones entre los componentes.

2.1.5.3. ¿Cómo funciona COM?

COM se basa en un sencillo modelo cliente-servidor.

- Por un lado está el servidor, que proporciona el código binario en forma de uno o varios objetos COM. Se debe entender el concepto "Objeto" tanto a nivel general y común a todo lenguaje como también en su sentido literal.
- Por otro lado está el cliente, el que utiliza los métodos de los objetos COM.

COM se encarga de la comunicación entre cliente y servidor, donde en función de la localización del servidor se diferencian tres modelos.

- In-Process-Server. Si el servidor se encuentra en el mismo proceso que el cliente, éste puede llamar directamente el objeto COM a través de su interfaz COM. (*In-Process-Server* son DLLs, por ejemplo controles ActiveX).
- Out-of-Process-Server. Si el servidor se encuentra en un proceso propio pero en la misma computadora que el cliente, COM inserta en la comunicación un *Stub* y un servidor *Proxy*. El *Stub* se ejecuta en el proceso del servidor y representa allí al cliente. El servidor *Proxy* por su parte se ejecuta en el proceso del cliente representando allí al objeto COM. Al cliente, por tanto, se le da a entender que el objeto COM requerido se encuentra en su proceso, mientras que el servidor y el objeto COM creen que la solicitud procede de su mismo proceso. COM realiza la auténtica comunicación entre procesos, también llamada *Marshalling*.
- Remote-Server. Si el servidor se encuentra en una computadora distinto al del cliente la comunicación también discurre a través de un servidor *Proxy* y un *Stub*, pero en este caso el intermediario es DCOM.

Para que COM pueda arreglar la comunicación, los objetos COM deben cumplir determinados requisitos, es decir, deben corresponder al estándar binario especificado en COM.

2.1.5.4. Servicios de aplicación OLE

COM es la arquitectura fundamental que formula las bases para los servicios de software de más alto nivel, como aquellos proporcionados por OLE. Los servicios OLE alcanzan varios aspectos de software de componentes, incluyendo documentos compuestos, controles personalizables, transferencia de datos, y otras interacciones de software (Ver Figura 32).

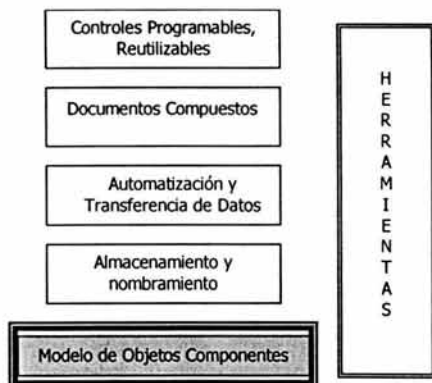


Figura 32. Los servicios de aplicación OLE de alto nivel están construidos en los fundamentos del Modelo de Objetos Componentes.

Estos servicios proporcionan diferente funcionalidad con claridad para el usuario; sin embargo ellos comparten un requisito fundamental para un mecanismo que permite a componentes de software binario, proporcionado por diferentes vendedores de software, conectarse y comunicarse entre sí de una manera bien definida. Este mecanismo es proporcionado por COM, una arquitectura de software de componentes que:

- Define un estándar binario para interoperabilidad de componentes.
- Es independiente del lenguaje de programación.
- Es suministrado en múltiples plataformas (Microsoft Windows, Microsoft Windows NT, Apple Macintosh, UNIX).
- Está provisto para evolución robusta de sistemas y aplicaciones basados en componentes.
- Es extensible.

Además COM proporciona mecanismos para lo siguiente:

- Comunicaciones entre componentes, aún a través de proceso y límites de red.
- Administración de memoria compartida entre componentes.
- Reporte de estado y error.
- Carga dinámica de componentes.

Es importante notar que COM es una arquitectura general para software de componentes. Mientras Microsoft está aplicando COM para direccionar áreas específicas como controles, documentos compuestos, automatización, transferencia de datos, almacenamiento y nombrando, entre otras, cualquier desarrollador puede tomar ventaja de la estructura y fundamentos que COM proporciona.

2.1.6. DCOM. Modelo de Objetos Componentes Distribuido.

DCOM (Distributed COM) es un desarrollo de COM para redes. DCOM es una extensión del COM para soportar objetos distribuidos a través de una red. Gracias a DCOM una aplicación puede ejecutar el código de un servidor DCOM instalado en cualquier computadora de la red.

DCOM (Distributed Component Object Model) es un protocolo que permite a componentes de software comunicarse directamente sobre una red de una manera eficiente, segura y fiable. Previamente llamado "Red OLE" (Network OLE), DCOM está diseñado para uso a través de múltiples transportes de red, incluyendo protocolos de Internet como HTTP * (Hypertext Transfer Protocol). DCOM está basado en la DCE-RPC * (The

* HTTP es un conjunto de reglas para transferencia de archivos (texto, imágenes, gráficos, sonido, video, y otros archivos multimedia) en la World Wide Web. Tan pronto como un usuario de la Red, utiliza su navegador el usuario está utilizando indirectamente http.

Distributed Computing Environment – DCE - Remote Procedure Call – RPC -) de la Fundación de Software Abierto (Open Software Foundation's DCE-RPC) y trabaja tanto con applets de Java como con componentes ActiveX a través del uso del COM.

Algunas características de DCOM son:

- DCOM está basado en la tecnología de componentes más ampliamente usada hoy en día. DCOM es simplemente "COM con un cable más largo".
- DCOM es la mejor tecnología de red para extender aplicaciones de componentes a través de Internet. Por ser una tecnología ActiveX, DCOM trabaja nativamente con tecnologías Internet como TCP/IP, Java, y HTTP, permitiendo a las aplicaciones de negocios trabajar a través de la Web.
- DCOM es una tecnología abierta que corre en múltiples plataformas.

Microsoft está licenciando abiertamente la tecnología DCOM a otras compañías de software para correr en la mayoría de los sistemas operativos, incluyendo implementaciones múltiples de sistemas basados en UNIX.

2.1.6.1. La Arquitectura DCOM

DCOM es una extensión de COM (Component Object Model). COM define cómo los componentes y sus clientes interactúan. Esta interacción está definida tal que el cliente y el componente puedan conectarse sin la necesidad de algún componente de sistema intermediario. El cliente llama métodos en el componente sin algo innecesario. La Figura 33 ilustra esto en la notación del Modelo de Objetos de Componentes.



Figura 33. Componentes COM en el mismo proceso.

En los sistemas operativos de hoy, los procesos están protegidos entre sí. Un cliente que necesita comunicarse con un componente en otro proceso no puede llamar al componente directamente, sino que tiene que usar algo de comunicación entre procesos proporcionada por el sistema operativo. COM proporciona esta comunicación de una manera completamente transparente: éste intercepta las llamadas del cliente y las reenvía al componente en el otro proceso. La Figura 34 ilustra cómo las bibliotecas en tiempo de ejecución (run-time) COM/DCOM proporcionan el vínculo entre el cliente y el componente.

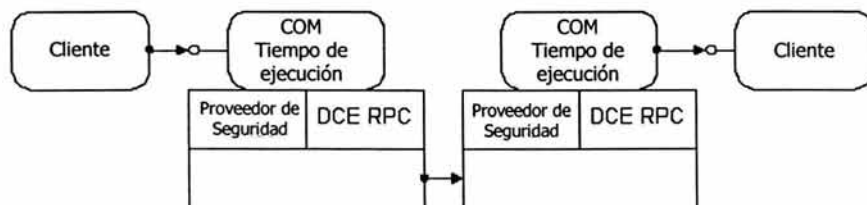


Figura 34. Componentes COM en diferentes procesos.

Cuando el cliente y el componente residen en diferentes máquinas, DCOM simplemente reemplaza la comunicación local entre proceso con un protocolo de red. Ni el cliente ni el componente están enterados de que el cable que los conecta ha llegado a ser un poco más largo.

* DCE-RPC es un protocolo a nivel aplicación de Grupo Abierto (OpenGroup), que permite a las aplicaciones hacer llamadas remotas a procedimientos. Este es el protocolo fundamental para DCOM.

La Figura 35 muestra la arquitectura completa DCOM: La run-time COM proporciona servicios orientados a objetos para clientes y componentes y usa RPC y el proveedor de seguridad para generar paquetes de red estándar que forman el protocolo de cable estándar DCOM.

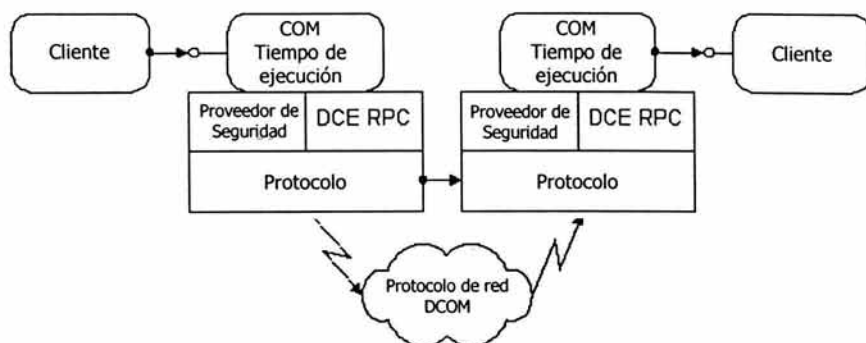


Figura 35. DCOM: Componentes COM en diferentes máquinas.

2.1.6.2. COM+

COM+. COM+ es la última mejora de la norma COM, distribuida por primera vez junto con Windows 2000. COM+ tiene una versión integrada de *Microsoft Transaction Server* (MTS), proporciona servicios adicionales y libera al programador de muchas tareas básicas que surgen en la programación COM.

2.1.7. OCX. OLE Custom Controls

El desarrollo de COM ha sido constante a lo largo de los años. Paralelamente, Microsoft ha ido acuñando nuevos nombres para comercializar su tecnología COM.

La automatización representa una estupenda posibilidad para "aprovechar" el código de otras aplicaciones de una forma legal. El inconveniente es que el cliente depende siempre de la presencia de un servidor, que normalmente procede de otro fabricante y que en raras ocasiones se distribuye junto con la aplicación cliente. Lo interesante sería que se pudiera integrar el servidor de automatización como un componente binario en la aplicación. Este deseo condujo a los controles OCX.

OCX es la abreviación para un control de uso OLE, un módulo de programa independiente que puede ser accedido por otros programas en un ambiente Windows. Los controles OCX terminan con una extensión .ocx. Los controles OCX representan la segunda generación de arquitectura de control Microsoft, siendo la primera los controles VBX escritos en Visual Basic.

Los controles OCX son servidores COM que no se pueden ejecutar de forma autónoma sino que necesitan y se integran en aplicaciones cliente. Los controles OCX son perfectamente comparables a los controles Windows: tras su registro amplían la funcionalidad del sistema operativo y cualquier aplicación los puede instanciar y utilizar. Los programas que utilizan controles OCX se suelen distribuir junto con el código binario del control. La rutina de instalación del programa suele encargarse de registrar el control OCX en el sistema.

Mientras que al principio Microsoft no concedió excesiva importancia al desarrollo de Internet, en los últimos años su tecnología COM ha apostado por la red bajo el lema de ActiveX. La especificación para controles OCX amplió con informaciones de tipo complementarias que permiten la distribución de los controles a través de Internet, como parte integral del páginas web, por ejemplo. Desde entonces se habla de controles ActiveX en

lugar de controles OCX. Sin embargo, ActiveX es compatible con los controles OCX, lo que significa que los contenedores ActiveX, tal como Internet Explorer de Microsoft, pueden ejecutar componentes OCX.

2.1.8. ActiveX ⁴¹

ActiveX es un conjunto de tecnologías que permite a componentes de software interactuar con otros en un ambiente de red. Estas tecnologías fueron construidas en el COM (Component Object Model) que permite a los componentes de software, sin importar el lenguaje en el que estén desarrollados, trabajar juntos en un ambiente de red. Aunque las tecnologías ActiveX son usadas primariamente para desarrollar contenido de páginas Web interactivo, ellas también pueden ser usadas en aplicaciones de escritorio y otros programas.

Estas tecnologías están definidas con bastante flexibilidad y son desarrolladas por Microsoft. ActiveX es una consecuencia de otras dos tecnologías Microsoft llamadas OLE y COM. Por su nombre, ActiveX puede ser confundido porque éste aplica a un conjunto completo de tecnologías basadas en COM. La mayoría de la gente, sin embargo, piensa únicamente en controles ActiveX, los cuales representan una forma específica de implementar tecnologías ActiveX.

2.1.8.1. ActiveX Controls

Un control ActiveX es un elemento de una interfaz de usuario creado utilizando tecnología ActiveX. Los controles ActiveX son pequeños, rápidos y poderosos, y hacen fácil la integración y la reutilización de componentes de software.

Los controles ActiveX están entre los muchos tipos de componentes que utilizan tecnologías COM para proporcionar interoperabilidad con otros tipos de componentes y servicios COM. Los controles ActiveX son la tercera versión de los controles OLE (OCX), proporcionando un número de mejoras diseñadas específicamente para facilitar la distribución de componentes sobre redes y proporcionar integración de controles en navegadores Web.

Un control ActiveX es esencialmente un simple "objeto OLE" o más específicamente, "Objeto COM (Componente Object Model)", el cual es auto-registrable.

Los controles ActiveX han llegado a ser la arquitectura básica para desarrollar componentes de software programables para su uso en una variedad de diferentes contenedores, extendiéndose desde herramientas de desarrollo de software hasta herramientas de productividad para el usuario final. Para que un control opere bien en una variedad de contenedores, el control debe ser capaz de asumir alguna funcionalidad de nivel mínimo que sea confiable en todos los contenedores.

Un control ActiveX puede ser descargado automáticamente y ejecutado por un navegador Web. ActiveX no es un lenguaje de programación, es más bien un conjunto de reglas de cómo las aplicaciones deben compartir información. Los programadores pueden desarrollar controles ActiveX en una variedad de lenguajes, incluyendo C, C++, Visual Basic, y Java. Los controles ActiveX tienen acceso total al sistema operativo Windows. Esto les da mucho más poder, pero con este poder viene un cierto riesgo de que se dañe software en la computadora. Para controlar este riesgo, Microsoft desarrolló un sistema de registro para que los navegadores puedan identificar y autenticar un control ActiveX antes de descargarlo.

2.1.9. OPC. (OLE for Process Control)

OLE para Control de Procesos es un software emergente estándar diseñado para proporcionar aplicaciones de automatización de fábricas con fácil acceso a datos de plantas industriales a través de una robusta infraestructura de comunicación de alta velocidad. OPC es un modelo cliente-servidor donde el servidor OPC

⁴¹ Referencia Sitios de Internet 11. <http://www.microsoft.com/com/tech/ActiveX.asp>

proporciona una interfaz estándar para los objetos OPC, permitiendo a aplicaciones cliente controlar dispositivos y administrar datos de dispositivos de manera genérica. Las aplicaciones OPC habilitadas pueden comunicarse con cualquier servidor OPC, si el servidor representa un sensor, válvula o un controlador lógico programable. Las tecnologías OLE/COM y DCOM son usadas para realizar el modelo cliente-servidor.

OPC al ser un estándar abierto que proporciona métodos consistentes para acceder datos de campo de los dispositivos de planta. Estos métodos mantienen el mismo tipo y fuente de datos; por lo tanto, el usuario se encuentra libre para elegir el software y el hardware que cumpla con sus necesidades primarias, sin tener que considerar la disponibilidad de los drivers (controladores) propietarios. OPC está construido usando tecnología OLE de Microsoft. Las especificaciones OPC hasta la fecha son las siguientes:

- Acceso a Datos (Data Access, DA)
- Alarmas y Eventos (Alarms and Events, AE)
- Batch (Proceso por lotes)
- Intercambio de Datos (Data Exchange DE)
- Acceso a Datos Históricos (Historical Data Access HDA)
- Seguridad (Security)
- Acceso a Datos XML (XML-Data Access XML-DA)

De las especificaciones OPC, la más utilizada es OPC DA, existe una necesidad creciente de integrar la información generada en los procesos de producción con sistemas de información que faciliten y mejoren la toma de decisiones a nivel gerencial.

La funcionalidad de OPC está basada en componentes de software que se apegan al estándar COM (Component Object Model) y se implementan en OLE (Object Linking and Embedding) de Microsoft, aunque existen versiones que funcionan sobre plataformas distintas a los sistemas operativos de Microsoft.

Las especificaciones de OPC son las definiciones de las interfaces de programación orientada a objetos que se deben implementar para establecer comunicación entre clientes y servidores. COM es un modelo de interacción entre objetos componentes y está basado en un esquema cliente-servidor, como se mencionó anteriormente. De forma general una comunicación vía OPC puede verse como en la siguiente figura, cada bloque de la figura representa unidades funcionales con diversos propósitos, programas de simulación, sistemas de control de procesos, sistemas de información, todos estos integrados a través de servidores OPC.

Dentro de las unidades OPC existen servidores y una unidad de integración. El propósito de la Figura 36 es integrar la información de un sistema de control, que implica comunicación con HMI's (Human Machine Interface) e incluso PLC's (Programmable Logic Controller) hacia programas de simulación y sistemas de información. Algunos simuladores pueden comunicarse con otras aplicaciones a través de DDE (Data Dynamic Exchange) de Microsoft y/o ODBC (Open Data Base Connectivity), algunos servidores OPC pueden comunicarse con esos sistemas y extraer información con la especificación OPC DA. La interacción entre los sistemas es bidireccional de ahí que los resultados de simulación y optimización puedan regresar al control de procesos para mejorar su producción. La Figura 36 muestra la integración de la información a través de OPC.

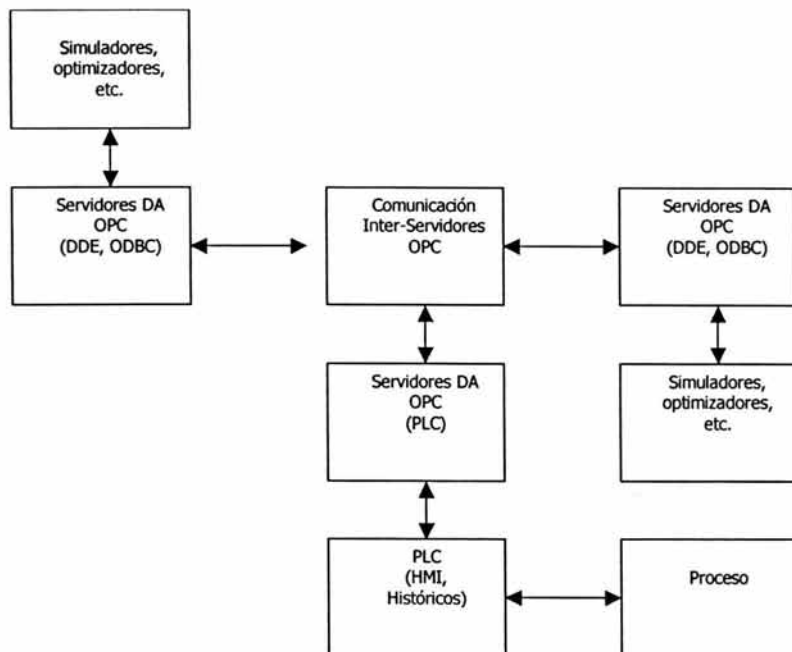


Figura 36. Esquema general de comunicación basada en el estándar OPC.

Bajo la idea anterior se plantea la integración de información que demuestra las posibilidades de los esquemas de interconectividad basada en OPC. Una ventaja de utilizar estándares como OPC es la posibilidad de comunicarse con sistemas de control para su configuración, de manera que es posible realizar pruebas en sistemas de producción reales.

2.1.10. OLE DB

Microsoft OLE DB es una arquitectura de base de datos de componentes que implanta un acceso eficiente a red e Internet para llegar a muchos tipos de orígenes de datos, entre los que se incluyen datos relacionales, archivos de correo, archivos sin formato y hojas de cálculo.

OLE DB es un conjunto de interfaces que muestran datos desde una variedad de fuentes relacionales y no relacionales utilizando el Modelo de Objetos Componentes (COM). Las interfaces OLE DB proporcionan aplicaciones con acceso uniforme a datos almacenados en diversas fuentes de información. Estas interfaces soportan la cantidad de funcionalidad de un SGBD (Sistema de Gestión de Bases de Datos) apropiada al almacenamiento de datos, permitiendo al almacén de datos compartir los mismos.

OLE DB consta de un modelo programático que consiste en proveedores de datos, los cuales, contienen y muestran los datos; consumidores de datos, los que utilizan los datos; y componentes de servicio, los que procesan y transportan los datos (tales como procesadores de consulta). Además, OLE DB incluye un puente a ODBC que permite soporte continuo para la amplia gama de controladores de bases de datos relacionales ODBC.

Por ejemplo, la arquitectura OLE DB se puede utilizar para conectar un proyecto de Microsoft Access a una base de datos de Microsoft SQL Server.

2.1.11. ODBC (Open Database Connectivity)

Existen muchos conceptos erróneos acerca de ODBC en el mundo de la computación. Para el usuario final, es un icono en el Panel de Control de Microsoft Windows. Para el programador de aplicaciones, es una biblioteca que contiene rutinas de acceso a datos. Para muchos otros, es la respuesta a todos los problemas de acceso a bases de datos que se hayan imaginado.

ODBC es un protocolo estándar para servidores de bases de datos desarrollado por Microsoft Corporation. Esta conectividad proporciona un lenguaje común para diferentes aplicaciones para obtener acceso a bases de datos en una red. ODBC, al ser formulado, fijó un grupo de protocolos para manejar las bases de datos de manera uniforme. La especificación cubría temas como maneras uniformes de comunicarse con el servidor, iniciar sesión y realizar consultas. ODBC se convirtió en una capa de software que, desde el punto de vista del usuario, proporcionaba una forma estándar de comunicarse con una base de datos y, desde el punto de vista del sistema operativo, permitía que todos los comercializadores de bases de datos tradujeran la información que el usuario proporcionaba para cumplir con los requisitos de la base de datos. Así, ODBC fue el primer golpe de la industria de cómputo para proporcionar acceso universal a bases de datos.⁴²

ODBC es una interfaz de programación de aplicación (API) aceptada ampliamente para acceso a bases de datos. Esta está basada en las especificaciones CLI (Call-Level Interface) de X/Open e ISO/IEC (ISO. International Organization for Standardization/IEC. International Electrotechnical Commission) para APIs de bases de datos y usos de SQL (Structured Query Language) como su lenguaje de acceso a bases de datos.

Este estándar está diseñado para interoperabilidad máxima – que es, la capacidad de que una aplicación acceda a diferentes sistemas de gestión de bases de datos con el mismo código fuente. Las aplicaciones de bases de datos llaman funciones en la interfaz ODBC, la cual está implementada en los módulos específicos de bases de datos llamados *drivers* (controladores). El uso de drivers aísla aplicaciones de llamadas específicas de base de datos en la misma forma en que los drivers de impresión aíslan los programas de procesamiento de palabras de los comandos específicos de impresión. Debido a que los drivers son cargados en tiempo de ejecución, el usuario solo tiene que agregar un nuevo driver para acceder a un nuevo SGBD; no es necesario recompilar o revincular la aplicación.

La meta de ODBC es hacer posible el acceso a cualquier dato de cualquier aplicación, sin importar cual sistema de administración de bases de datos (SGBD) está conteniendo el dato. ODBC administra esto insertando una media capa, llamada controlador de base de datos, entre una aplicación y el SGBD. El propósito de esta capa es convertir las consultas de datos de aplicación en comandos que el SGBD entienda. Para este trabajo, tanto la aplicación como el SGBD deben ser ODBC-compliant (sumiso), que es, la aplicación debe ser capaz de emitir comandos ODBC y el SGBD debe ser capaz de responderlos.

Es importante entender que ODBC está diseñado para exponer las capacidades de la base de datos, no para complementarlas. Por tanto, los escritores de aplicaciones no deben esperar que usando ODBC una simple base de datos de repente se transforme en un motor de base de datos relacional completamente caracterizado.

2.1.12. DAO

La primera interfaz de Microsoft para los programadores de bases de datos se llamó objetos de acceso a datos (DAO; Data Access Objects), y se popularizó en asociación con la escritura de aplicaciones para recuperar datos guardados en Access de Microsoft. Y esta popularidad se debió a su facilidad de uso. Por tanto, casi todos los entornos de desarrollo empezaron a soportarlos. DAO permite agregar una referencia a un proyecto que define la interfaz del objeto.⁴³

⁴² Referencia 12. Houlette, Forrest. México, 2003, 422 pp.

⁴³ Referencia 12. Obra citada.

DAO permite utilizar un lenguaje de programación para acceder y manipular datos en bases de datos locales o remotas, y administrar bases de datos, sus objetos y su estructura.

Los Objetos de Acceso a Datos proporcionan una estructura para utilizar código para crear y manipular bases de datos. DAO proporciona un conjunto jerárquico de objetos que utiliza el motor de bases de datos Microsoft Jet para acceder datos y estructuras de base de datos en:

- Bases de datos Microsoft Jet (.MDB)
- Fuentes de datos ODBC, utilizando un controlador ODBC.
- Bases de datos ISAM (Indexed Sequential Access Method) instalables, como dBASE, Paradox y Microsoft FoxPro, que el motor de base de datos puede leer directamente.

2.1.13. ADO. (ActiveX Data Objects)

Los objetos de datos de ActiveX (ADO) son la nueva estrategia de Microsoft para acceso a datos, diseñados para acompañar su método OLE DB para el acceso universal a datos. Los objetos ADO proporcionan una jerarquía de objetos diferente para acceso a bases de datos. La estructura de los objetos está mucho menos orientada a Access de Microsoft, que es el programa de bases de datos y el motor central de la base de datos que hizo crecer a DAO. Por tanto, a muchos les parece que ADO es más intuitivo para la interacción con bases de datos que DAO. Mientras Microsoft se concentra en el lenguaje de marcación extensible (XML, Extensible Markup Language) como medio para acceder bases de datos a través de firewalls de Internet, ADO soporta el conjunto de características orientadas al futuro para Microsoft como empresa.⁴⁴

ADO puede ser usada para tener acceso a muchos tipos diferentes de datos, incluyendo páginas Web, hojas de cálculo, y otros tipos de documentos. Dentro de algunas aplicaciones ADO es comúnmente usado en conjunto con controladores ODBC para conectar a bases de datos y a otras fuentes de datos disponibles a través de los controladores ODBC.

Los Objetos de datos ActiveX (ADO), son objetos de programación que representan la estructura de la base de datos y los datos que contiene la misma. En el Editor de Visual Basic, se pueden utilizar objetos ADO y un componente adicional de ADO denominado ADOX (Microsoft ADO Extensions for DDL and Security) para crear o cambiar tablas y consultas, proteger la base de datos o tener acceso a datos de orígenes de datos externos. También se pueden utilizar objetos ADO en código para manipular datos almacenados en la base de datos. ADO y ADOX admiten el acceso a cualquier origen de datos que tenga un controlador ODBC o un proveedor OLE DB creados para dicho origen de datos.

ADO proporciona un modelo de programación común para cualquier fuente de datos OLE DB; esto es esencialmente una colección de objetos que exponen los atributos y métodos usados para comunicarse con una fuente de datos. ADO utiliza los proveedores OLE DB generales para acceder a características únicas de fuentes de datos específicas, este también utiliza proveedores OLE DB nativos incluyendo un proveedor OLE DB específico que proporciona acceso a controladores ODBC (drivers). Diseñado para reemplazar las necesidades de todos los otros métodos de acceso a datos de alto nivel, ADO puede acceder a bases de datos ya sea jerárquicas, con método de acceso secuencial indexado (ISAM) o relacionales, o cualquier tipo de fuentes de datos tan grande como un controlador ODBC-compliant.

2.1.14. MDAC

Los Componentes de Acceso a Datos de Microsoft (MDAC) son las tecnologías clave que permiten el Acceso a Datos Universal. Las aplicaciones cliente-servidor que transportan datos utilizadas sobre la Web o una LAN pueden utilizar estos componentes para integrar fácilmente la información de una variedad de fuentes, tanto relacionales (SQL) como no relacionales. Estos componentes incluyen Microsoft ActiveX Data Objects (ADO), OLE DB, Y ODBC.

⁴⁴ Referencia 12. Houlette, Forrest. México, 2003, 422 pp.

2.2. Software para Simulación y Optimización de procesos de producción de hidrocarburos.

Las aplicaciones a utilizar en este trabajo son específicamente para simulación y optimización, por lo tanto se explica brevemente la función de las mismas en general y su clasificación según su aplicación en los procesos petroleros. Finalmente se da a conocer más a fondo información tanto de las dos compañías que desarrollaron los softwares a integrar como las características de los mismos.

2.2.1. Simuladores de procesos.

En la década de los años ochenta con el advenimiento de los computadores personales y el desarrollo de mejores correlaciones de flujo multifásico que permitían estimar con más precisión las pérdidas de energía a lo largo de las tuberías, se fomentó el desarrollo de simuladores del proceso de producción de tal forma que en cuestión de pocos minutos se podía determinar la capacidad de producción de un pozo. Estos simuladores fueron adaptados para realizar diversos análisis de sensibilidad para establecer la influencia de las variables sobre la capacidad de producción del sistema. Esta técnica recibió el nombre de "Análisis Nodal", registrada por la Macco-Schlumberger, y otros la llamaron "Análisis del Sistema de Producción". Básicamente lo que se realiza es un balance de energía entre la capacidad de aporte de fluidos del Yacimiento (Inflow) y la habilidad del Pozo conjuntamente con sus facilidades de superficie (Outflow) para extraer y transportar los fluidos desde el fondo del pozo hasta la estación de flujo en la superficie. El balance de energía puede realizarse en cualquier punto de la trayectoria del fluido que se denomina "Nodo" de allí el nombre de "Análisis Nodal".

La masiva utilización de la informática en la enseñanza y en el entorno industrial, la sorprendente y revolucionaria evolución de las computadoras personales en cuanto a tamaño, coste, velocidad, softwares, etc. han ayudado sin lugar a dudas a que la simulación digital o simulación por computadora sea hoy en día la herramienta más utilizada para realizar experimentos de simulación de sistemas. Un programa de simulación de computadora se puede definir como una secuencia de instrucciones que el usuario define para resolver un problema que puede estar plasmado en unas ecuaciones que describen a un sistema que previamente hemos modelado mediante dichas ecuaciones.

La construcción de un modelo de simulación ha pasado, de ser una labor reservada a especialistas en programación, de difícil y costosa realización, basada en procesos batch y en una interpretación en general elaborada a partir del procesado de tediosos listados, a ser un ejercicio estructurado alrededor de la utilización de entornos cada vez mas amables y flexibles que permiten aprovechar la característica mas destacable de la simulación: la posibilidad de estudiar la evolución dinámica de los sistemas a lo largo del tiempo.

Hoy en día al ingeniero se le abren un amplio abanico de posibilidades para resolver estos problemas y para programas estas operaciones necesarias para realizar la simulación. El abanico corresponde a los distintos lenguajes que podemos utilizar para traducir nuestros modelos en una computadora y posteriormente resolverlos para obtener la simulación del comportamiento del sistema modelado. Podemos utilizar lenguajes de programación general, lenguajes específicos para simulación o paquetes de software de simulación especialmente preparados para la misma.

A la hora de elegir una herramienta u otra hay que tener en cuenta primeramente la velocidad de ejecución de los programas y la utilización de recursos necesaria (memoria, coprocesadores, etc.).

2.2.1.1. Optimización del sistema.

Una de las principales aplicaciones de los simuladores del proceso de producción es optimizar el sistema, lo cual consiste en eliminar o minimizar las restricciones al flujo tanto en la oferta (Inflow) como en la demanda (Outflow), para ello es necesario la realización de múltiples balances con diferentes valores de las variables más importantes que intervienen en el proceso, para luego, cuantificar el impacto que dicha variable tiene sobre la

capacidad de producción del sistema. La técnica puede usarse para optimizar la completación de pozos que aun no han sido perforados, o en pozos que actualmente producen quizás en forma ineficiente.

Para este análisis de sensibilidad la selección de la posición del nodo es importante, ya que a pesar de que la misma no modifica, obviamente, la capacidad de producción del sistema, si interviene tanto en el tiempo de ejecución del simulador como en la visualización gráfica de los resultados.

2.2.1.2. Clasificación de los simuladores comerciales.

La simulación de procesos ya no es sólo para expertos. En la última década su uso se ha estado incrementando y hay desarrollos recientes en simuladores que están haciendo que éstos sean más flexibles para su aplicación general en ingeniería química.

El primer software para modelado y simulación de procesos fue desarrollado en compañías químicas y universidades. Alrededor de 1980, el software comercial llegó a ser usado más normalmente. Hoy en día, está disponible una amplia variedad de software dirigido a ingeniería de procesos. Entre el software de simulación y modelado de procesos usado más comúnmente en la industria química están los sistemas de Aspen Technology Inc. (AspenTech), Cambridge, Mass.; Simulation Sciences Inc. (Simsci), Brea, Calif.; Hyprotech, Calgary, Alberta; y Chemstations Inc., Houston.⁴⁵

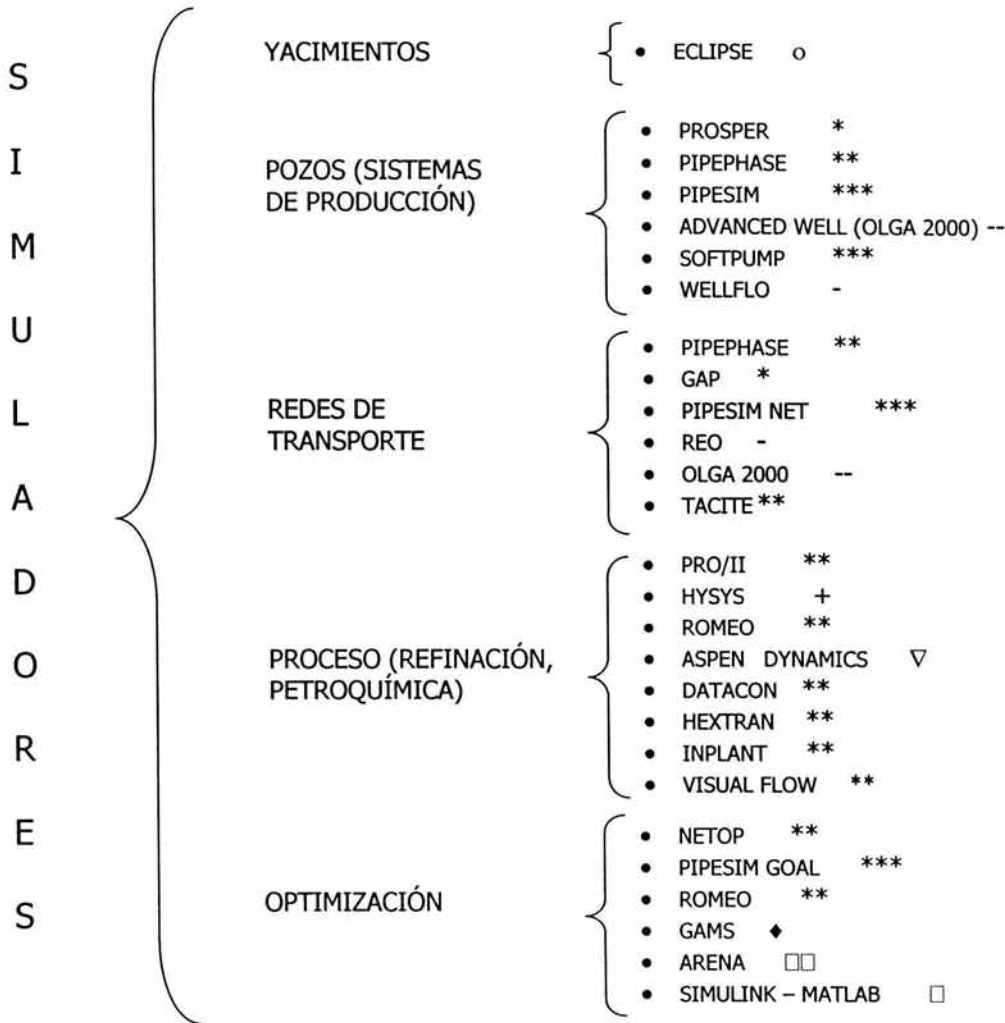
La simulación de procesos depende de un modelo consistente en expresiones matemáticas que describen el comportamiento de cada uno de los procesos químicos y operaciones físicas formando los pasos de una hoja de flujo de procesos. Las propiedades termodinámicas y físicas de los componentes en las corrientes de flujo son también importantes para una simulación y están típicamente contenidos en los mismos programas de software.

La hoja de flujo del proceso proporciona información acerca del proceso para el modelo – por ejemplo, la configuración de las unidades de operación junto con condiciones de operación y composiciones de las corrientes de alimentación y proporciones de flujo. La simulación entonces predice el funcionamiento de la planta computando el flujo y las propiedades de todas las corrientes de producto e intermedias, el desarrollo de cada unidad en el proceso, y los costos de operación y capital de la planta. De esta manera, los ingenieros, pueden, por ejemplo, investigar alternativas de diseño cambiando las condiciones de la simulación.

Los simuladores comerciales tienen aplicaciones en un amplio rango de áreas. Pero los que interesan para el desarrollo de este trabajo son los de simulación de procesos de producción.

En la Figura 37 se muestra la clasificación de paquetes de software, específicamente diseñados para simulación relacionada con los procesos de producción de hidrocarburos, de acuerdo a la aplicación que se les da en PEMEX. Todos ellos se pueden encontrar en el mercado:

⁴⁵ Referencia Sitios de Internet 15. <http://pubs.acs.org/hotartcd/cenear/950327/art10.html>



COMPAÑÍAS:

- * Petroleum Experts
- ** SIMSCI – INVENSYS
- *** Baker Jardine – Schlumberger
- + HYPROTECH
- EPS (Edimburg Petroleum Services)
- ▽ ASPENTECH

- SCAND POWER
- MATHWORKS
- Rockwell Automation
- GeoQuest (Schlumberger Oilfield Services)
- ◆ GAMS Development Corporation

Figura 37. Clasificación de simuladores según su aplicación.

A continuación se da una breve explicación de cada uno de éstos, de acuerdo a la compañía que los desarrolló y que ahora distribuye.

COMPañÍA: SimSci-Esscor es parte de la familia Invensys plc, una compañía de control y automatización. SimSci-Esscor soporta a más de 750 compañías clientes en más de 70 países.⁴⁶

- Suministro de software comercial de simulación para las industrias química, petrolera y petroquímica.

Suite OPS (On-Line Performance Suite). OPS entrega la última generación de Control de Procesos Avanzado (APC) y aplicaciones de Optimización en Línea para ayudar a los usuarios a obtener el máximo desarrollo desde sus unidades de operación. Ver Figura 38.

- ROMeo:** Es una herramienta de análisis operacional completamente integrada, un ambiente de modelaje avanzado para optimización de procesos dentro y fuera de línea (Optimization Off-Line y On-Line). ROMeo es un ambiente de modelado avanzado diseñado para ayudar a maximizar la productividad de refinerías, petroquímicas, y procesos químicos. ROMeo permite llevar a cabo los beneficios de la optimización de procesos en toda la planta y es la única solución de optimización de procesos completo que integra simulación rigurosa, conciliación de datos, detección de grandes errores, optimización económica y monitoreo del desempeño en un ambiente de un solo usuario.
- Connoisseur:** Es una extensa suite de software APC que controla simultáneamente algunas variables de proceso para alcanzar sus valores óptimos vía el Sistema de Control Distribuido (DCS) de la planta. Connoisseur provee un integrado e inigualable conjunto de facilidades de ingeniería, permitiendo obtener nuevas revelaciones del proceso y optimizar estrategias de control predictivo para intensificar la capacidad del proceso. Connoisseur se conecta directamente al proceso o puede estar fijo dentro del equipo de control existente. Una vez conectado, desarrolla una serie de pruebas para coleccionar dinámicamente datos, los cuales toman lugar en operaciones de proceso.
- ARPM:** Es una solución completa para monitorear el desarrollo del equipo y los procesos usando modelos de simulación rigurosos y datos de planta en tiempo real para extraer procesos válidos e información del desempeño del equipo. ARPM emplea técnicas de simulación junto con técnicas de conciliación de datos para proveer datos de operación de la planta que sean consistentes, confiables y que estén a tiempo.

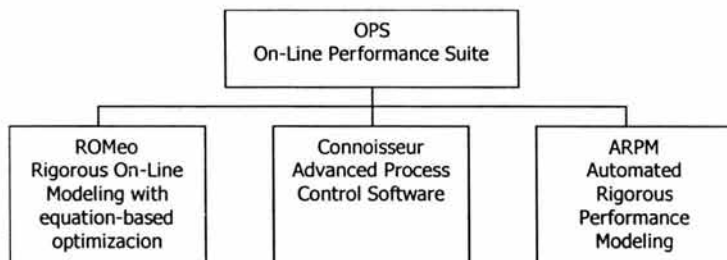


Figura 38. Suite de Desempeño en Línea.

Suite PES (Process Engineering Suite). Es un conjunto consistente de herramientas técnicamente superiores para análisis operacional y diseño de ingeniería de procesos. PES mejora la productividad en el ciclo de vida de la planta y puede ser usado para producción de gas y aceite, refinería, petroquímicas, químicas, farmacéuticas y modelado de plantas de polímeros. Ver Figura 39.

- Pro/II:** Simulador general para modelado y optimización de procesos químicos, petroquímicos, de refinación y petróleo en estado estacionario. Este programa de simulación de procesos desempeña rigurosos balances de masa y energía para un alto rango de procesos químicos. Desde la separación

⁴⁶ Referencia Sitios de Internet 25. <http://www.simsci.com/corporate/>

gas-aceite hasta la destilación reactiva, Pro/ii ofrece la solución de simulación disponible más extensa y fácil de usar.

- b) Hextran: Simulador para diseño y optimización de intercambiadores de calor y redes de intercambio térmico. Hextran es un poderoso simulador en estado estacionario que proporciona una vista exacta y confiable de los sistemas de transferencia de calor. Éste combina tecnología de simulación de transferencia de calor probada con los más grandes y más respetados bancos de datos de propiedades físicas en la industria. Hextran simula rigurosamente los sistemas más altamente integrados. Se puede monitorear y examinar el desempeño de las redes de intercambiadores de calor desde una perspectiva global o ver cada intercambiador individualmente.
- c) Datacon: Sistema de reconciliación de datos para detección de errores de medición. Datacon convierte datos de proceso en tiempo real a información de negocios consistente y confiable. Datacon usa técnicas de sonido para conciliar o ajustar fluido, temperatura y medidas de composición para satisfacer balances de energía y material alrededor de cada unidad en un proceso de planta. Existen aplicaciones de interfaces entre Datacon y sistemas DCS (Sistemas de Control Distribuido), históricos de datos y Bases de Datos Centralizadas.
- d) Inplant: Simulación de flujo de fluidos multifásico para redes de tubería interplanta. Un simulador riguroso en estado estacionario para análisis y diseño de sistemas de redes de tuberías en planta. Los ingenieros pueden valorar y analizar rápidamente la seguridad de los sistemas de tubería de la planta. Inplant permite el diseño de nuevos sistemas de tubería o la renovación de una gran variedad de sistemas existentes.
- e) Visual Flow: Es una herramienta de diseño, documentación y modelaje de sistemas de seguridad y redes de alivio de presión. Visual Flow proporciona respuestas rápidas, confiables y exactas a los problemas más difíciles, desde la valoración de complejos sistemas de ayuda hasta sistemas de agua de enfriamiento altamente ciclados. Visual Flow provee un ambiente fácil de usar para simulación rigurosa en estado estacionario en cuanto a diseño, valoración y análisis.

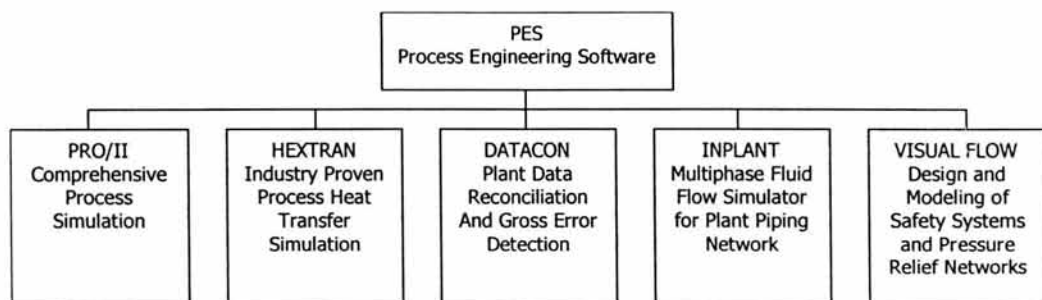


Figura 39. Suite de Ingeniería de Procesos.

Suite UOS (Upstream Optimization Suite). Es un conjunto integrado de herramientas de apoyo a la decisión, técnicamente superior, diseñado para proporcionar facilidad en la administración de bienes desde la profundidad hasta la superficie. UOS une las necesidades de la industria de producción de gas y aceite, estándares de software actuales y expertos técnicos de SIMSCI para entregar las mejores soluciones de su clase, a problemas del mundo real. Ver Figura 40.

- a) Pipephase: Simulador de flujo de fluidos, multifásico, en estado estacionario para modelar rigurosamente sistemas de producción y transporte de petróleo y gas. Pipephase combina un algoritmo de solución con métodos de producción modernas y software de técnicas de análisis para crear una herramienta de planeación y diseño de campo de aceite robusta y eficiente.

- b) **Tacite**: Simulador de flujo de fluidos multifásico para modelaje de condiciones transitorias en tuberías y sistemas de producción. Tacite, usa una rigurosa metodología de rastreo composicional de multicomponentes, fenómenos tales como la acumulación de componentes pesados en los puntos bajos de la tubería también pueden ser predecidos.
- c) **NetOpt**: Poderosa herramienta de software para la optimización de sistemas de producción de petróleo y gas. Contempla los métodos de producción por Gas Lift y ESP (Bombeo Neumático y Bombeo Electrocentrifugo). NetOpt permite la definición de una estrategia de optimización para un alto rango de operaciones de campo de gas y aceite.

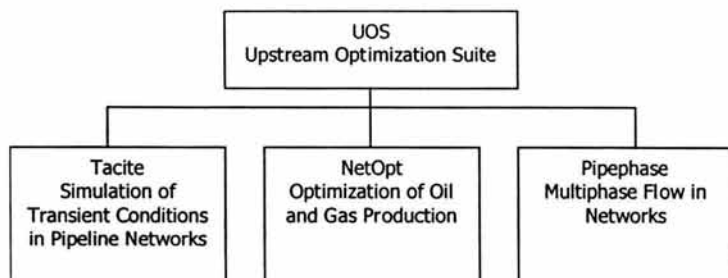


Figura 40. Suite de Optimización contra la corriente.

COMPAÑÍA: Baker Jardine que pertenece a Schlumberger Oilfield Services, una compañía líder que suministra un alto rango de soluciones y servicios de tecnología para la industria internacional de gas y aceite.⁴⁷

- Modelado de Sistemas de Producción y Productos de Software de Optimización.

La Suite de PIPESIM comprende el siguiente software de simulación:

- Pipesim**: Simulador general de flujo multifásico para líneas de transporte de flujo multifásico y líneas de producción de hidrocarburos. Comprende modelos de producción por métodos: Análisis Nodal, Gas Lift y ESP (Bombeo Electrocentrifugo) y equipo de superficie.
- Pipesim Goal**: Es una herramienta de optimización para diseño y operaciones de producción por Gas Lift y ESP.
- Pipesim Net**: Simulador general de redes de transporte de flujo multifásico. Contiene todas las características del Pipesim más el proceso de recolección, distribución y modelado de redes cerradas.
- Goal-Online**: Es un sistema integrado cuya función es enlazar Pipesim Net y Pipesim Goal para operaciones de transferencia de datos con el sistema SCADA (System Control and Data Acquisition).
- Pipesim Connect**: Conecta la Suite de Pipesim a las bases de datos para actualización rápida de los datos y actualización de los modelos con pruebas de pozos.

COMPAÑÍA: PETEX Petroleum Experts. Desarrollo de Software de Ingeniería. Petroleum Experts proporciona las mejores soluciones de ingeniería para la industria de gas y aceite. Esto es logrado a través de una gran inversión en investigación y desarrollo.⁴⁸

- Modelado de campo integrado y optimización de la producción.

- Prosper**: Production and Systems Performance Analysis Software. Software de simulación que permite asistir al ingeniero de producción o de yacimientos para el análisis de la hidráulica de tubería de producción y transporte, así como de temperaturas con exactitud y velocidad. Ejecutando separadamente la simulación de cada componente del sistema de producción del pozo, y permitiendo al usuario verificar cada subsistema del modelo ejecutando el "matching", Prosper asegura que los cálculos

⁴⁷ Referencia Sitios de Internet 12. <http://www.bakerjardine.com>

⁴⁸ Referencia Sitios de Internet 24. <http://www.petex.com/>

de las predicciones son los más exactos posibles. Una vez que el sistema es ajustado (entonado) a las condiciones reales de operación, Prosper puede utilizarse en predicciones de presiones del yacimiento basado en los datos de producción de superficie.

- b) Gap: Gap Allocation Program. Este software de simulación fue desarrollado como un programa de asignación de volúmenes de gas de inyección (Gas Lift). Se ha convertido en una herramienta que permite hacer un análisis completo de sistemas de recolección, asignación y producción de hidrocarburos. Modela tanto los sistemas de producción de flujo multifásico como de inyección (en pozos productores de aceite, gas, condensados y agua). El control de los volúmenes de producción se realiza a través de estranguladores, frecuencias de operación en sistemas ESP o asignación de Gas Lift para maximizar la producción de aceite de acuerdo a ciertas restricciones de proceso.

COMPAÑÍA: HYPROTECH. Hyprotech es el suministrador líder de servicios y software de simulación y modelado para las industrias de procesamiento por lotes y continuo, incluyendo separación de aire, química, procesamiento de gas, petroquímica, farmacéutica y refinería.⁴⁹

- Proporciona soluciones (servicios y software) de simulación y modelado para mejorar significativamente, la productividad, eficiencia y creatividad.

a) HYSYS: Simulador general para modelado y optimización de procesos químicos. Con HYSYS, los ingenieros necesitan desarrollar únicamente un solo modelo de proceso que puede ser usado desde el diseño conceptual a través de aplicaciones en línea para mejorar los diseños, optimizar la producción y mejorar la toma de decisiones. HYSYS es el único ambiente de simulación integrado disponible comercialmente diseñado para una completa personalización por parte del usuario.

COMPAÑÍA: EPS (Edimburg Petroleum Services). EPS provee lo mejor en su clase en cuanto a software, servicios y capacitación para la ingeniería petrolera.⁵⁰

- Modernizar y Automatizar las tareas de ingeniería de producción asociadas con la administración de bienes o activos.

- a) WellFlo. WellFlo es un software altamente usado por compañías de producción de gas y aceite para mejorar el funcionamiento en campo y de los pozos. WellFlo está diseñado para los ingenieros petroleros con un nivel de sofisticación, velocidad y exactitud. Al mismo tiempo, el programa proporciona la facilidad de utilizarlo asegurado un rápido aprendizaje y un alto nivel de productividad – un factor importante hoy en día con una gran demanda por parte de la industria de gas y aceite. Las condiciones de operación son optimizadas para todo el campo incluyendo la red de las líneas de flujo de la superficie.
- b) ReO. ReO es un software probado para simulación y optimización de sistemas de producción de hidrocarburos. En la producción de gas y aceite aguas arriba (upstream) son usadas varias tecnologías para direccionar diferentes aspectos de modelado y optimización de la producción. Estos enfoques y la tecnología de software desarrollada para servir a los mismos tiene un número de aplicaciones, por ejemplo la simulación de un yacimiento proporciona una vida útil del campo. ReO optimiza el área de operaciones entre el yacimiento y las instalaciones, para auxiliar en el diseño de nuevas capacidades de producción y para optimizar los sistemas de producción en tiempo real. ReO integra complejos cálculos de ingeniería, límites prácticos y valores económicos para determinar la configuración óptima.

COMPAÑÍA: ASPENTECH. Aspen Technology, Inc. es un proveedor líder en software y servicios para las industrias de procesos. Conduciendo mejoras a través de procesos complejos que están situados en el corazón

⁴⁹ Referencia Sitios de Internet 16. <http://www.hyprotech.com/corp/default.asp>

⁵⁰ Referencia Sitios de Internet 18. http://www.epsedin.co.uk/EPS_about/aboutEPS_home.htm

de las operaciones de las compañías de procesos. AspenTech ayuda a los usuarios a lograr la excelencia operacional, permitiéndoles incrementar su productividad y optimizar el funcionamiento de sus negocios.⁵¹

- Ayuda a las compañías a diseñar y mejorar las plantas y procesos, maximizando los beneficios a través de la vida operacional.

a) Aspen Dynamics®: Simulador general para modelado y optimización de procesos químicos, petroquímicos, de refinación y petróleo en estado transitorio. Aspen Dynamics entrega los beneficios del modelado dinámico a las industrias petroquímicas y químicas, así como a las organizaciones de ingeniería y operación de las plantas. Aspen Dynamics puede reproducir lo dinámico de las operaciones de planta, incrementando la operabilidad, productividad y seguridad.

COMPAÑÍA: SCAND POWER. Scandpower tiene 30 años de experiencia proveyendo servicios de consultoría al mercado internacional. Scandpower proporciona servicios internacionales dentro de la "Administración de Riesgos" y la "Tecnología de Información".⁵²

- Soluciones orientadas a la administración y a la técnica protegiendo la vida y salud de la gente, el ambiente externo contra la contaminación e incrementando los bienes y la productividad.

a) OLGA 2000. Simulador de comportamiento de flujo multifásico para redes de transporte (tuberías y sistemas de producción) en condiciones de flujo en estado transitorio. OLGA 2000 es la suite de software líder en el mundo para simulaciones de flujo multifásico transitorio. La mayoría de compañías tanto de aceite como de ingeniería son usuarias de este software. A través de la cooperación extensa de institutos de investigación y compañías de aceite, la suite de software está desarrollándose continuamente de acuerdo a la demanda de la industria. OLGA 2000 tiene una paquetería extensa de módulos especializados. Estos módulos son desarrollados a través de una larga investigación y datos medidos en campo, para ser capaces de simular los retos demandados relacionados a transporte multifásico.

b) Advanced Well Module. Olga 2000 contiene un módulo para mejorar las simulaciones de tres fases de flujo dentro de un pozo perforado. El módulo de pozo está diseñado para aplicaciones de flujo de pozos en donde las propiedades del yacimiento y las relaciones de afluencia jugarán un papel importante cuando se modelen los escenarios. Esta herramienta será el primer simulador de tres fases transitorias de pozo perforado (wellbore) disponible en la industria.

COMPAÑÍA: MATHWORKS. Desde su fundación en 1984, MathWorks ha llegado a ser el proveedor global líder en software de diseño basado en modelos y computación técnica.⁵³

- Producción de software de diseño de modelos y computación técnica para ingenieros, científicos, matemáticos e investigadores.

a) Simulink - MATLAB. Matlab es usado para desarrollar cálculos matemáticos, analizar y visualizar datos, y escribir nuevos programas de software. Simulink®, es utilizado para modelado y simulación de sistemas dinámicos complejos. Simulink es una herramienta interactiva para modelado, simulación y análisis dinámico, de sistemas de múltiples dominios. Ésta te permite describir exactamente, simular, evaluar y refinar el comportamiento de un sistema a través de estándares y librerías de bloques. Simulink se integra con MATLAB, proporcionando acceso a una gran variedad de herramientas de diseño y análisis. Estos beneficios hacen a Simulink la herramienta de elección para diseño de sistemas de control, diseño de sistemas de procesamiento de señales, diseño de sistemas de comunicaciones y otras aplicaciones de simulación.

⁵¹ Referencia Sitios de Internet 17. <http://www.aspentech.com/corporate/index.cfm>

⁵² Referencia Sitios de Internet 19. <http://www.scandpower.no/?ItemId=1102>

⁵³ Referencia Sitios de Internet 21. <http://www.mathworks.com/company/>

COMPAÑÍA: Rockwell Automation. Rockwell Automation es la compañía líder en automatización industrial enfocada a ser el mayor proveedor global de soluciones de información y control. La compañía es también proveedor líder de tecnologías y aplicaciones de administración.⁵⁴

- Ayuda a reducir costos, conservar recursos, mejorar la productividad y reducir el tiempo para el mercadeo de buenos materiales y productos.

a) Arena®. La familia de productos de Arena apoya a las organizaciones en la toma de decisiones con ayuda de herramientas de modelado y simulación de procesos.

COMPAÑÍA: GeoQuest (Schlumberger Oilfield Services). Schlumberger Oilfield Services es el proveedor líder más grande del mundo en servicios de exploración y producción, además de soluciones y tecnología para la industria petrolera internacional. Schlumberger Oilfield Services ofrece una suite de servicios y soluciones en tiempo real que permiten a los usuarios a traducir datos adquiridos a información útil y transformar esta información en conocimiento como una base para mejorar la toma de decisiones en cualquier momento y en cualquier lugar.⁵⁵

- Schlumberger Information Solutions (SIS), una unidad de operación de Schlumberger, diseñada para proporcionar un conjunto de soluciones completas y capacidades que apoyan a los procesos operacionales de las industrias de gas y aceite.

- SIS, a través de la división GeoQuest Software, está entregando una suite de herramientas de software para mejorar el entendimiento de los yacimientos complejos y proporcionar acceso oportuno a los datos necesitados para la toma de decisiones en exploración y producción.

a) ECLIPSE. Software de Simulación de Yacimientos. La suite de simuladores ECLIPSE son herramientas que ayudan a administrar rápida y eficientemente las simulaciones de yacimientos.

COMPAÑÍA: GAMS Development Corporation.⁵⁶

- Diseño de modelos de problemas de optimización lineales, no lineales y mezclados.

a) GAMS. El Sistema de Modelado Algebraico General (GAMS) es un sistema de modelado de alto nivel para problemas de programación matemática. Este consiste en un lenguaje compilador y un solucionador estable de alto desempeño. GAMS, al ser complejo, está hecho a la medida para aplicaciones de modelado a gran escala, y permite construir grandes modelos que pueden ser adaptados rápidamente a nuevas situaciones. GAMS es un software para modelado y solución de problemas de optimización.

2.2.2. Compañía: Baker Jardine & Associates

Baker Jardine Petroleum Engineering & Software

BJA es un proveedor de herramientas de software, consultoría en tecnologías de información y soluciones integradas enfocadas a incrementar la producción de gas y aceite.

Baker Jardine desarrolla Software de Simulación de Ingeniería Petrolera para el diseño y administración de la operación de sistemas de producción de gas y aceite multifásico; yacimiento, pozos, tuberías e instalaciones de producción. Baker Jardine ofrece tecnología única y servicios a quienes están envueltos en estas áreas y

⁵⁴ Referencia Sitios de Internet 22. http://www.rockwellautomation.com/about_us/index.html

⁵⁵ Referencia Sitios de Internet 15. <http://www.sis.slb.com/>

⁵⁶ Referencia Sitios de Internet 23. <http://www.gams.com/>

también tiene capacidad de consultoría multidisciplinaria para soportar la implementación y aplicación de sus tecnologías.

Características:

- Ofrece servicios a la industria petrolera y de gas desde 1986.
- Oficinas principales: Londres (Principal), Houston, México, Venezuela y Calgary.
- Ahora parte de Schlumberger(SIS) desde Abril del 2001
 - Brinda beneficios a los clientes en:
 - Desarrollo de productos.
 - Soporte
 - Consultoría
 - Entrenamiento

Schlumberger (SIS) provee herramientas para simulación de red, optimización de la producción y capacidades de administración de sistemas para mejorar cada etapa de proceso de ingeniería de producción, y ofrece a la industria tecnología avanzada y soluciones de producción integrada desde una sola fuente.

Schlumberger Oilfield Services, por medio de GeoQuest, una unidad de operación perteneciente a este corporativo (la cual adquirió a BJA) abastece sistemas de software integrado y soluciones de administración de datos para ayudar a las compañías a optimizar el valor de sus yacimientos de gas y aceite. Schlumberger Oilfield Services, proporciona tecnologías servicios y a la industria petrolera internacional, con oficinas operativas, lugares de servicio, e instalaciones de desarrollo e investigación alrededor del mundo.

2.2.2.1. Software: PIPESIM 2000

PIPESIM: Modelos de Sistemas de Producción de Gas y Aceite Integrados (Integrated Oil & Gas Production Systems Models).

PIPESIM 2000 es la paquetería de instalaciones e ingeniería petrolera para diseño, operación y optimización.

PIPESIM 2000 proporciona un estado estacionario (steady-state), simulación de flujo multifásico para Sistemas de Producción de Gas y Aceite. Los módulos individuales PIPESIM son usados para una amplia variedad de análisis incluyendo: modelado de pozos, optimización artificial lift, modelado de instalaciones de proceso y tuberías, y planeación de campo.

PIPESIM 2000 es soportado en las siguientes plataformas: Win95, Windows 98, Windows NT y Windows 2000.

PIPESIM 2000 puede estar asegurado en dos diferentes formas, con una llave física de acceso (dongle), la cual debe ser instalada antes de que el software sea instalado; o utilizando el Servidor Administrador de Licencias (Sentinel License Manager Server).

PIPESIM 2000 consta de los siguientes módulos:

- Análisis de Tuberías e Instalaciones de Superficie (Pipeline & Facilities Analysis)
- Análisis de Comportamiento de Pozos (Well Performance Analysis)
- Análisis de Redes de Tuberías (Network Analysis)
- Optimización de Producción (Production Optimization -GOAL-)
- Planificación de Campos de Producción (Field Planning -FPT-)
- Modelado de Pozos Horizontales y Multilaterales (Multi-lateral -HoSim-)

Beneficios de PIPESIM 2000

- Incremento potencial de la Producción del Campo manteniendo la misma infraestructura.

- Reducción de los costos operativos.
 - Mejoramiento de los resultados de las simulaciones de yacimiento al considerar en forma integral las restricciones de los pozos y la de las instalaciones de superficies.
 - Mejoras en la planificación de la producción de campos de gas y petróleo.
 - Maximiza el retorno de la inversión en diseño de campos nuevos.
 - Incremento en la productividad de los ingenieros (reducción del número de horas hombre).
- Característica Open Link COM.

Muchas paqueterías de software son autónomas (stand alone) y permiten muy poco al usuario en cuanto a interacción y personalización. Con el número cada vez mayor de simulaciones los usuarios llegaron a ser familiares con los lenguajes de programación tales como Visual Basic, es importante que las paqueterías líderes en la industria ofrezcan un ambiente de simulación abierto.

Open Link ha sido desarrollado para dar a los usuarios de PIPESIM mas apertura y flexibilidad en el uso del software. Open Link permite que funciones de PIPESIM sean accedidas vía cualquier aplicación COM (MS Excel, Visual C++).

Por ejemplo, el Open Link ha sido usado recientemente para permitir que hojas de cálculo de Excel sean vinculadas con un modelo de sistema de producción. En este caso el open link es usado para pasar datos de entrada desde Excel a un modelo y entonces recuperar datos y desplegar datos de salida.

La funcionalidad Open Link es proporcionada vía un número de módulos que pueden ser usados para acceder a la funcionalidad de PIPESIM 2000.

Net32COM.DLL incluye funciones de interfaz para permitir al usuario cargar modelos PIPESIM 2000 tanto de red (network) como de una sola rama (sigle branch), consultarlos (configuración de equipo, inyección gas lift, etc.), fijar y recuperar propiedades físicas, correr simulaciones y guardar cualquier modificación.

GLDesign.DLL y GLDiagn.DLL son los módulos de Gas Lift Design y Gas Lift Diagnostics y pueden ser usados para tener acceso a las capacidades de diagnósticos y diseño construidas en PIPESIM 2000.

PNSReader.DLL y Wellcurve.DLL son usadas para recuperar resultados de las simulaciones.

2.2.3. Compañía SimSci-Esscor de Invensys plc.



SimSci-Esscor. Líder Mundial en Soluciones de Simulación y Optimización.

SimSci-Esscor es parte de la familia Invensys, una compañía de control y automatización. SimSci-Esscor soporta a más de 750 compañías clientes en más de 70 países.

Invensys plc es un líder global en administración de energía y tecnología de producción. El grupo ayuda a los usuarios a mejorar su desempeño y productividad utilizando tecnologías y servicios innovadores y un profundo entendimiento de sus industrias y aplicaciones. Invensys ha apalancado la solidez, experiencia y tecnología de su familia de unidades de negocio incluyendo APV, Avantis, Eurotherm, Foxboro, SimSci-Esscor, Tricoex y Wonderware, enfocadas a proporcionar lo mejor en su clase, en cuanto a software de simulación y optimización y servicios relacionados a la refinería, gas y aceite, papel y pulpa, generación de poder e industrias de procesos químicos.

SimSci-Esscor™ es un líder en el desarrollo y mercadeo de software y sistemas de simulación de procesos industriales y para una variedad de industrias, incluyendo exploración de gas y aceite, refinería de petróleo, manufactura química y petroquímica, generación eléctrica de poder, minería, pulpa y papel, e ingeniería y construcción. Los productos de SimSci-Esscor permiten a los clientes minimizar los requerimientos de capital, optimizar el funcionamiento, y maximizar el regreso de las inversiones produciendo bienes.

SimSci-Esscor ha servido a las industrias de proceso alrededor de 35 años, con soluciones estratégicas que agregan valores medibles en sus operaciones.

Los productos de SimSci-Esscor están diseñados para incrementar el rendimiento reduciendo costos de inversión de capital, además ayuda en la toma de decisiones. SimSci-Esscor tiene oficinas en Brasil, Alemania, Japón, Singapur, Emiratos Árabes Unidos, Reino Unido, Estados Unidos, y Venezuela y proporciona soporte y servicios a más de 650 usuarios en alrededor de 65 países.

2.2.3.1. PRO/II

PRO/II. Simulación de Procesos de Refinería (Simulation of Refinery Processes).

Pro/ii, como se vió anteriormente, pertenece a la Suite de Ingeniería de Procesos (PES), la cual fácilmente se comunica con otras aplicaciones comúnmente usadas por ingenierías de procesos gracias a una interfaz, y mejora la productividad en el ciclo de vida de la planta.

Pro/ii, Simulación de Procesos Extensos, (Comprehensive Process Simulation) es un programa de simulación de procesos que desempeña rigurosos balances de masa y energía para un alto rango de procesos químicos. Desde separación gas-aceite a la destilación reactiva, Pro/ii ofrece la solución de simulación más extensa y fácil de usar. Las industrias servidas son: procesamiento de gas y aceite, refinerías, petroquímicas, químicas, polímeros, químicas finas y farmacéuticas, e ingeniería y construcción.

La simulación se aplica en: diseño de nuevos procesos, evaluación alterna de configuraciones de la planta, modernización y renovación de plantas existentes, valoración y documentación de acuerdo con reglas ambientales, eliminar cuellos de botella de procesos de la planta y en monitoreo, optimización y mejora del rendimiento y productos de la planta.

La interfaz de usuario gráfica (GUI) Provisión (un ambiente de soporte basado en Microsoft Windows de 32-bits) del Pro/ii, proporciona una completa interactividad, en ambiente basado en Windows, que es ideal para construir y modificar modelos de Pro/ii tanto simples como complejos. PROvision incorpora muchos estándares Microsoft, como OLE Automation (Automatización OLE), permitiendo al usuario transferir rápidamente gráficos y datos de proceso a otras aplicaciones Windows, además se pueden revisar resultados de la simulación por medio de reportes, gráficas y tablas.

Pro/ii no sólo es fácil de usar, sino que también proporcionar la abertura necesaria para transferir exitosamente datos importantes a otros programas de ingeniería. Los grupos de trabajo de ingeniería pueden mejorar significativamente la productividad usando esta característica. El Pro/ii Automation Layer (componente de Pro/ii), proporciona acceso a prácticamente cualquier grupo o parte de datos en una base de datos de simulación. Se pueden enviar datos fácilmente entre Pro/ii y Microsoft Excel u otras aplicaciones OLE-compliant a través de la interfaz de servidor. Esta poderosa capacidad soporta prácticamente cualquier proceso de flujo de trabajo (workflow) de ingeniería.

Pro/ii proporciona un ambiente donde los ingenieros pasan menos tiempo simulando problemas de proceso y más tiempo implementando mejoras en el proceso.

Seguridad.

La seguridad de Pro/ii incluye dos tipos de llaves: un periférico semiconductor, el cual es la licencia física que se conecta al CPU para que el software se ejecute, o una llave virtual (no física), que es una clave de acceso que proporciona el proveedor con menos duración, con la cual también se puede ejecutar el software.

Beneficios.

- Bajos costos de operación y capital.
- Diseño de la planta mejorado.
- Mayor productividad de la planta y más calidad de productos.
- Modelos de planta más flexibles.
- Ingeniería de procesos más productiva
- Aprendizaje de las curvas mucho más sencillo para modelos de planta complejos.
- Reducción del capital o costos de ingeniería de cinco a quince por ciento.

Para realizar un modelo confiable en Pro/ii se tienen que seguir los siguientes pasos:

- Dibujar el diagrama o la hoja de flujo (flowsheet) definiendo las corrientes en cada unidad de operación.
 - Definir los componentes del proceso.
 - Seleccionar el Método de Cálculo Termodinámico.
 - Definir la alimentación de las corrientes (temperatura, presión, etc.).
 - Proporcionar las condiciones del proceso.
 - Correr la simulación.
- Pro/ii OLE Automation Layer.

Pro/ii OLE Automation Layer permite la comunicación de Pro/ii con otras aplicaciones, ofreciendo la capacidad para transferencia de datos bidireccional. Para realizar dicha comunicación el usuario debe tener conocimientos de Pro/ii y Visual Basic. Pro/ii Automation Layer proporciona las herramientas para recibir y enviar información entre diversas aplicaciones, usando un mecanismo de transferencia de datos.

Pro/ii con Provisión tiene otros métodos disponibles para intercambiar información, además de OLE Automation Layer, y son:

- Portapapeles de Windows. Esta herramienta estándar de Windows puede ser usada para transferir datos entre aplicaciones. Aunque éste ahorrará reescribir, no determina el formato de los datos y puede ocuparse demasiado tiempo para grandes cantidades. En Pro/ii, el portapapeles puede ser usado para transferir tablas de propiedades de corrientes, secciones en los archivos de salida o la paleta PFD (Process Flow Diagram).
- PDTs (Pro/ii Data Transfer System). Son unas herramientas de programación FORTRAN, las cuales permiten acceso a datos de simulaciones realizadas. La aplicación resultante puede ser leída únicamente desde la base de datos de la simulación y puede ser hecha sólo para escribir a otras aplicaciones con gran dificultad.
- Plantillas Excel. Estas plantillas estándar instaladas ofrecen acceso limitado a Pro/ii OLE Automation Layer. Aunque un gran paso es que ahora permite que la salida pueda ser completamente configurada, éstas son sólo implementaciones de OLE Automation Layer. Sin embargo, nuevas aplicaciones pueden ser hechas usando OLE y agregándolas a la instalación de Pro/ii.

- OLE Automation Layer. Esta tecnología permite acceso completo a datos de entrada y salida para una simulación Pro/ii. Además, ofrece muchas de las funciones de cálculo de Pro/ii y puede ser usado de manera parecida en la mayoría de las aplicaciones Windows.

Pro/ii OLE Automation Layer incluye una Biblioteca de Clases. Esta biblioteca es una DLL que puede ser accedida a través de muchas de las aplicaciones utilizando OLE Automation Layer. Utilizando bastantes funciones del mismo OLE Automation combinadas con algunas rutinas comunes de Windows, se elimina mucho código requerido en la configuración de una sesión OLE. Esto hace que la configuración OLE sea mucho más fácil.

2.3. Identificación de datos requeridos y generados por las herramientas de simulación.

Se requiere conocer los formatos de los datos tanto de entrada como de salida de las herramientas de simulación para identificar las el tratamiento de los mismos por parte del software, y también se analizan las etapas en dónde se tiene que definir una manera para la transferencia de datos. Esta información ayuda también a saber cuáles son las extensiones de los archivos con datos de entrada y cuáles son los que se crean con datos de salida, para que la recuperación de los mismos a la base de datos sea correcta.

2.3.1. PIPESIM 2000

PIPESIM usa lo siguiente para almacenar datos:

- ASCII files
- Binary files
- Microsoft Access Database

Datos de entrada. (*.BPS, *.BPN, *.PGW, *.FPT, *.HSM)

Contiene todos los datos que son necesarios para correr un modelo. Este incluye datos para; unidades, composición de fluidos, datos de sistema, etc. El equipo de soporte requiere de estos archivos cuando son hechas consultas de este tipo.

Datos de salida. (*.OUT, *.SUM)

Contiene datos de salida del programa en diferentes formatos.

Archivos de transferencia (*.PLT, *.PLC, *.PWH, *.PBT, *.TNT, *.PST)

Archivos que transfieren datos de un módulo de PIPESIM 2000 a otro.

Tabla PVT (*.PVT)

Un archivo que contiene una sola composición de flujo y una tabla de propiedades del fluido para un conjunto dado de valores de presión y temperatura. Este archivo puede (si se requiere) ser creado por una paquetería comercial PVT, por ejemplo: Multiflash, Hysys, PVTsim, EQUI90, etc., o vía el módulo composicional en PIPESIM 2000.

Archivos de Base de Datos (*.MDB)

El archivo de Base de Datos Microsoft Access que contiene:

- Datos de fluido Black Oil.
- Curvas de desempeño ESP.
- Curvas de compresor y bomba definidas por el usuario.

Archivo de unidades (*.UMF)

Archivos de unidades usados para almacenar conjuntos de unidades definidas por el usuario. Estos archivos pueden ser pasados de un usuario a otro.

2.3.2. PRO/II

Pro/ii utiliza los siguientes formatos de datos:

Archivos de Bases de Datos.

Se pueden extraer una variedad de datos de los resultados desde la base de datos de la simulación y exportarlos a archivos de texto planos. La mayoría de los datos son exportados a un solo archivo. En una computadora personal, el nombre del archivo que recibe los datos incluye típicamente el sufijo ".P2I". También se pueden generar o escribir archivos con el sufijo ".ASC", se crea un archivo ".ASC" en donde se escribe un conjunto completo de datos de resultados a ese archivo.

Datos de entrada. (*.INP, *.PR1)

Contiene todos los datos que son necesarios para correr un modelo. Este incluye datos para unidades de operación, datos de corrientes, composición de fluidos, datos de sistema, etc.

Simulación. (*.INP, *.PR1, *.PRZ)

Ejecutar corridas de simulaciones de entradas de archivos con extensión (*.inp) y/o archivos de simulación Pro/ii (*.pr1).

El archivo con extensión .prz se genera después de la primera corrida de la simulación en base al archivo con extensión .pr1. En caso de que se modifique el modelo y se vuelva a correr la simulación el archivo .prz desaparece y se crea uno nuevo.

Datos de salida. (*.OUT, *.PLT)

Resultados de la ejecución de una simulación en archivo en formato ASCII .OUT.

Generar el reporte de salida de una simulación que ha sido resuelta. El reporte de salida será almacenado en un archivo con una extensión .OUT.

Generar y desplegar esquemas (plots) de gráficos de resultados de cierta unidad de operación. Estos cuadros gráficos serán desplegados mediante un visor contenido en Pro/ii, pero éste también soporta Microsoft Excel o "GnuPlot" para desplegarlos.

Si se está utilizando el visor de esquemas interno, los cuadros de gráficos generados puede ser guardados en un archivo con una extensión .PLT.

Archivos de Datos (*.ADJ, *.RAW, *.TDF, *.DAT).

Los datos de una planta medidos en varias formas son transferidos a Pro/ii a través de archivos de texto. Los archivos de datos reconciliados puede tener extensiones .ADJ, mientras que los archivos no reconciliados pueden tener extensiones .RAW, .TDF, o .DAT. Los archivos de datos pueden ser creados utilizando un editor de textos, o éste puede ser generado por el software del sistema de control de la planta.

Importar datos desde archivos (*.adj, *.raw, *.tdf, *.dat)

Se pueden importar archivos de datos reconciliados (*.adj), o archivos de datos no reconciliados (*.raw). Los datos en los archivos puede estar caracterizados por el sufijo utilizado en el nombre del archivo (*.adj, *.tdf, *.dat, o *.raw).

Exportar datos. (*.adj, *.tdf, *.dat, *.raw)

Exportar datos desde una simulación actual. Pro/ii pregunta el nombre del archivo y se escoge uno con las extensiones (*.adj, *.tdf, *.dat, or *.raw). La extensión que se escoja ayudará a caracterizar los datos exportados.

Exportar datos desde archivos de Pro/ii a otros archivos.

Exportar datos de gráficas y simulaciones a otros archivos desde la simulación actual abierta. Las opciones disponibles que permiten exportar son las siguientes:

- Datos de una simulación Pro/ii a un archivo ASCII.
- Datos de una tabla de propiedades de corriente a un archivo de hoja de cálculo compatible.
- Hoja de flujo o una página seleccionada del dibujo de la hoja de flujo al portapapeles (clipboard), a un archivo de Autocad con extensión .DXF, o a un archivo de correo (postscript).
- Datos "etiqueta" a un archivo de datos (con sufijo ".raw").

2.4. Comunicación de datos entre los simuladores de proceso.

2.4.1. PIPESIM 2000

La comunicación, en este caso, consiste en ver al software como una caja negra (cuyo funcionamiento interno desconocemos) y solo utilizar los datos de entrada y de salida. Primero se obtuvieron los datos de salida con ayuda de las DLLs de Open Link, esto se realizó tanto con Excel como con Access utilizando VBA (Visual Basic para aplicaciones) de la siguiente manera:

- Se agregó la librería PnsReader 1.0 Type Library (PNSREADER.DLL), la cual incluye interfaces compatibles que pueden ser usadas para leer datos de salida de archivos de salida (.PNS) generados por el PIPESIM y los motores de PIPESIM-NET.

La interfaz soportada por esta librería es IPNSComPtr, para modelos de red como para modelos de una sola rama.

Esta interfaz consta de varios métodos, de los cuales se ocuparon los siguientes:

➤ Métodos.

- ReadPnsFile. Este método operacional lee un archivo de salida (.PNS). Parámetros: nombre del archivo, ruta completa del archivo .pns.
- GetNodeCount. Este método obtiene el número total de nodos, uniones + pozos + fuentes + sumideros. Salida: entero (int), iNodeCount.
- GetNodeName. Este método obtiene el nombre del índice de un nodo en un arreglo. Parámetros: entero(int), iNode – índice en el arreglo. Salida: nombre del elemento.
- GetNodeType. Obtiene el tipo de nodo en la red. Parámetros: entero(int), iNode –índice en el arreglo. Salida: nombre del tipo de nodo, ya sea fuente, sumidero, pozo o unión.
- GetNodeIndex. Obtiene el índice del nodo. Parámetros: cadena(string), nombre del nodo.

- **GetNodeVariableValue.** Obtiene el valor de la variable de cualquier tipo (temperatura, presión, etc.), para el elemento del índice en el arreglo. Parámetros: entero(int), iNode – índice del elemento en el arreglo. Salida: doble(double), valor requerido.
- **GetBranchCount.** Obtiene el número total de ramas (excluyendo pozos, los cuales el motor trata como rama + fuente/sumidero). Salida: entero(int), numero de ramas.
- **GetBranchName.** Obtiene el nombre de el elemento que indica el índice en el arreglo. Parámetros: entero (int), iNode – índice en el arreglo. Salida: cadena (string), nombre del elemento.
- **GetBranchInVariableValue.** Obtiene el valor de la variable de cualquier tipo (temperatura, presión, etc.), para el elemento del índice en el arreglo. Parámetros: entero(int), iNode – índice del elemento en el arreglo. Salida: doble(double), valor requerido.
- **GetBranchOutVariableValue.** Obtiene el valor de la variable de cualquier tipo (temperatura, presión, etc.), para el elemento del índice en el arreglo. Parámetros: entero(int), iNode – índice del elemento en el arreglo. Salida: doble(double), valor requerido.

- Con estas interfaces se realizó la conexión de los archivos de salida de PIPESIM 2000 tanto con Excel como con Access.

- En Access se realizó una conexión a la base de datos por medio del proveedor Microsoft Jet OLEDB 4.0. y un objeto ADODB.

Para la introducción de datos al software se realizó lo siguiente:

Se agregó la librería Net32COM 1.0 Type Library (Net32COM.DLL), la cual incluye interfaces para fijar y recuperar propiedades físicas, realizar consultas, correr el modelo, entre otras que ya se mencionaron.

Las interfaces soportadas por esta librería son InetModel (para modelos y operaciones de red) y ISingleBranchModel (para modelos y operaciones de una sola rama).

Esta interfaz consta de varios métodos y operaciones, de los cuales se ocuparon los siguientes:

➤ Métodos.

- **GetLastError.** Regresa el último mensaje de error producido por la interfaz. Esta función puede ser llamada inmediatamente después de un método que regresa un valor falso.
- **GetNameList.** Regresa un arreglo con los nombres de os objetos del tipo dado (Object Type). El número de objetos encontrados es regresado en una variable long. Parámetros: tipo de Objeto. Salidas: Variant, el arreglo de nombres y largo (long) numero de objetos.
- **SetBoundaryTemperature.** Fija el valor dado en las unidades dadas como condición límite de temperatura para el objeto dado. Parámetros: Nombre del objeto, y un valor doble.
- **SetBoundaryPressure.** Fija el valor dado e las unidades dadas como condición límite de presión para el objeto dado. Parámetros: Nombre del objeto, y un valor doble.
- **SetBoundaryFluidrate.** Fija el valor dado e las unidades dadas como condición límite de proporción de fluido (fluidrate) para el objeto dado. El tipo de fluido puede ser cualquiera de estos: 0 = proporción de líquido (liquid rate), 1 = proporción de gas (gas rate), 2 = proporción de masa (mass rate). Parámetros: Nombre del objeto, y un valor doble.

➤ Operaciones.

- **OpenModel.** Abre el modelo de red dado. El nombre del archivo debe ser un modelo de red PIPESIM 2000 válido (extensión .bpn). Esta función debe ser llamada al principio, cuando se utiliza esta interfaz. Si esta función es exitosa se regresa un OK, de otra manera se regresa un FALSE, en este caso, el método GetLastError, debe ser llamado para obtener una descripción del error encontrado. Parámetros: nombre del archivo (string).
- **SaveModel.** Guarda el modelo actualmente abierto en un archivo dado por el nombre de la ruta. Parámetros: Ruta del archivo.

2.4.2. PRO/II

Para entender mejor como se llevó a cabo la comunicación entre este simulador y la base de datos de Access, se hablará de los objetos de Pro/ii. La estructura de la jerarquía de objetos de Pro/ii consta de tres niveles:

- **Server Object (Objeto Servidor).** En el primer nivel, este objeto se puede comparar con el inicio de cualquier aplicación, aquí el driver OLE Automation (P2OLEDBS.EXE) es iniciado. El objeto servidor es inicializado para el archivo esquema correspondiente (diccionario para todos los objetos de la simulación) y la configuración Pro/ii.
- **File Object (Objeto Archivo).** El objeto archivo puede ser comparado con una simulación abierta en Pro/ii. Como con Archivo, Nuevo o Abrir, el usuario puede abrir archivos de simulación existentes o crear nuevos utilizando la funcionalidad en el objeto servidor.
- **Data Object (Objeto Datos).** Activar un objeto datos es similar a dar doble click sobre un objeto de simulación para obtener acceso a la ventana de datos de entrada. Los objetos de datos nuevos pueden ser creados así como borrados en una aplicación OLE.

La Figura 41 ofrece un resumen de la arquitectura completa OLE, la cual consta de los siguientes bloques:

- El *OLE Automation Driver* (Controlador de Automatización OLE) es el programa que permite acceso a la funcionalidad del objeto Pro/ii desde una aplicación OLE (P2OLEDBS.EXE).
- El *Database Server* (Servidor de Base de Datos) es la interfaz que administra el acceso a archivos de datos de simulación y a objetos en estos archivos (DBSDLL.DLL).
- El *Schema File* (Archivo Esquema) es un archivo binario, el cual proporciona el diccionario de todas las clases y atributos de los objetos que pueden ser accedidos (SCHEMA.SDF).
- El *Pro/ii Binding* (Atador Pro/ii) es la capa que traduce las genéricas llamadas OLE a sus implementaciones Pro/ii específicas. El archivo P2DBLB.DLL implementa la capa atadora de Pro/ii que consta de las dos partes siguientes:
 - Las *Database Acces Functions* (Funciones de Acceso a la Base de Datos) y las *Pro/ii Utilities* (Utilidades de Pro/ii) son las dos partes de la capa atadora. Las funciones de acceso a la base de datos manipulan los objetos de datos de Pro/ii y leen y escriben datos desde estos objetos. Las utilidades de Pro/ii proporcionan cálculos útiles (como cálculo de propiedades de corriente, separación de corriente, etc.) las cuales pueden ser llamadas desde una aplicación OLE.

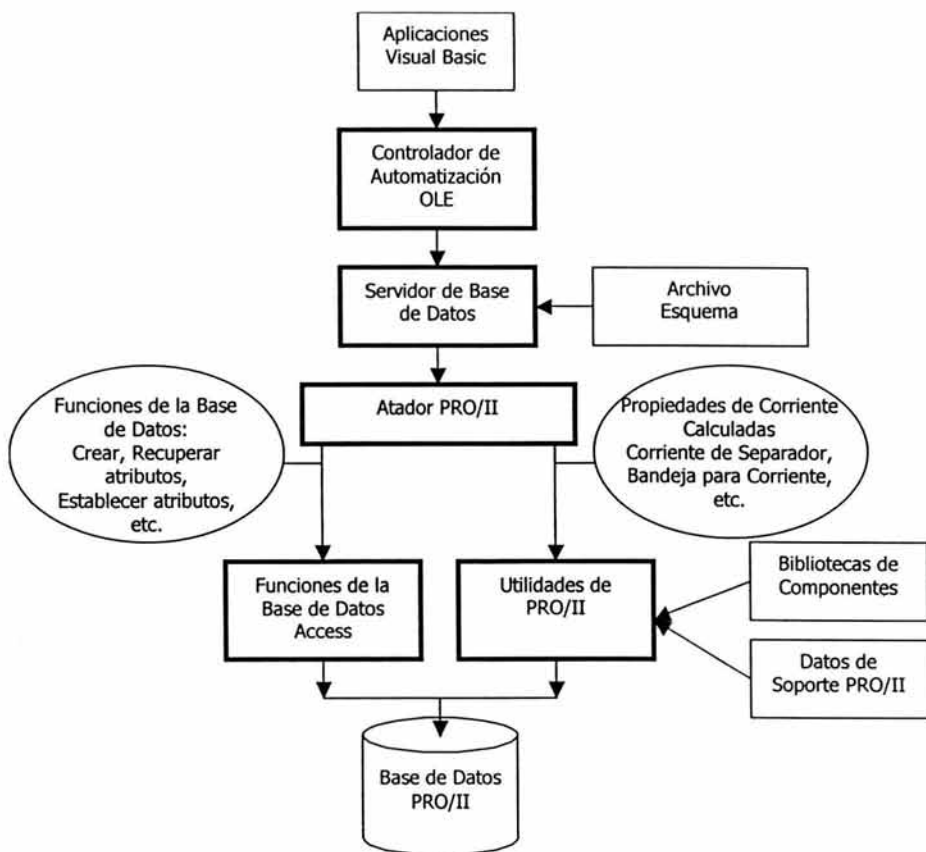


Figura 41. Acceso a las Bases de Datos.

Hay tres formas que se pueden utilizar para realizar la comunicación entre Pro/ii con otra aplicación, y son: Utilizando DDE y dos maneras de Transferencia de datos OLE, ésta se puede utilizar tanto con comandos de Visual Basic como CreateObject como con la Biblioteca de Clases de Pro/ii. A continuación se hace una breve descripción de cada una de ellas.

2.4.2.1. Utilizando DDE.

Para una transferencia de datos OLE que se hace con la instalación de Pro/ii (por ejemplo abrir Excel desde Pro/ii), se requiere utilizar DDE. Para esto, se crea un canal DDE que será creado entre las dos aplicaciones. A través de este canal, está disponible la siguiente información:

- Localización del archivo PROII.INI
- La ruta completa de la base de datos de la simulación permanente.
- La ruta completa de la base de datos de la simulación temporal.
- Los objetos que el usuario ha seleccionado en Pro/ii.

Como seguridad y para minimizar las oportunidades de errores, Pro/ii (y la mayoría de los softwares que trabajan con bases de datos), nunca abre la base de datos original de la simulación directamente. En vez de

esto, una copia idéntica con un nombre aleatorio es creada y utilizada. Cuando la simulación es guardada, las bases de datos temporales son simplemente copiadas sobre las originales.

Los tres pasos que están envueltos para establecer un vínculo DDE y transferir la información mencionada arriba son: creación del canal, requerimiento de información, liberar el canal.

El primer paso para establecer una sesión DDE, es pedir un canal exclusivo entre, en este caso Access y Pro/ii. La función a usar es DDEInitiate. Se debe enviar con éste, el nombre de la aplicación, el vínculo debe ser establecido con el programa llamado "PVision" y el tema (algo en la aplicación para el cual el canal es abierto – usualmente un documento de esa aplicación). Si el comando es exitoso, éste regresa el número del canal creado.

- DDEInitiate ("PVision", "PFDInfo") ----- Obtener el canal.

Después la información puede ser pasada a través del canal utilizando el comando DDERequest. Esta función requiere el numero del canal (puede haber mas de un canal abierto) y el ítem que es requerido.

- DDERequest (Channel, Item) ----- para "INIFile" y "TempDbFile"

El término del canal DDE consta de dos partes. Primero, DDEExecute es usado para correr la función "IAmDone" en el lado de Pro/ii. Este manda a la aplicación de regreso a su control. Para entonces descartar el canal ahora obsoleto, el comando DDETerminate es utilizado, con el numero de canal como un parámetro.

- DDEExecute (Channel, Message)
- DDETerminate (Channel)

2.4.2.2. Transferencia de datos OLE.

Para tener acceso a los componentes individuales de una simulación, una sesión OLE debe establecer la jerarquía completa de los objetos desde el Objeto Servidor, a el Objeto Archivo para entonces obtener el acceso al Objeto Datos (Ver Figura 42).

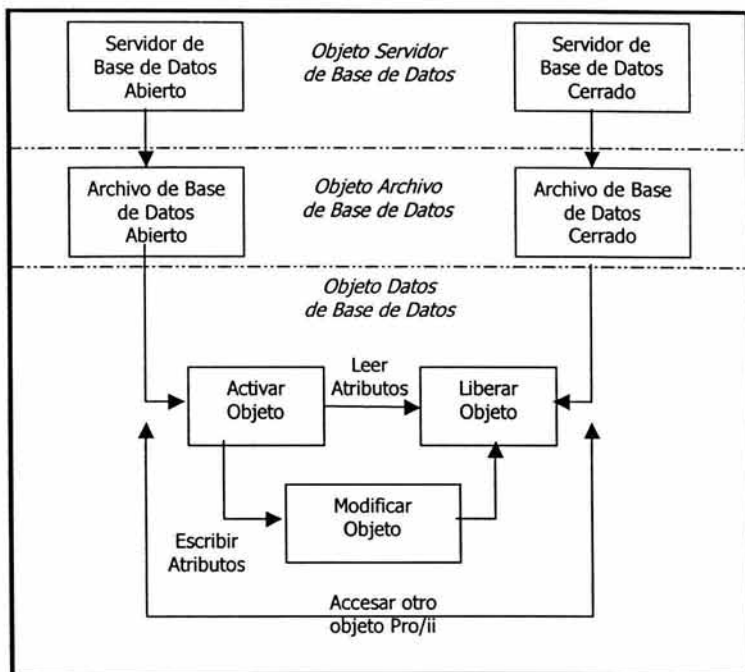


Figura 42. Transferencia de Datos OLE.

Se tiene que crear un Objeto Servidor al iniciar la configuración. Esto es hecho utilizando la función de Visual Basic CreateObject. Esta función creará y regresará una interfaz a un objeto OLE. Los parámetros que deben ser proporcionados con esta función son: el nombre de la aplicación (SimSciDbs) y el tipo de objeto que debe ser creado esa aplicación (Database). Una vez que el objeto ha sido creado, el método Initialize (inicializar) del Servidor debe ser utilizado para indicar donde están localizados los archivos Esquema y la configuración (como una ruta completa a estos archivos).

Para obtener una programación correcta, el nombre de la variable que recibe (AP2Server) debe ser del tipo 'Object'.

La creación de este objeto puede ser comparada con dar doble click a un icono de Pro/ii en Windows.

Ejemplo de comandos:

```
Set AP2Server = CreateObject ("SimSciDbs.Database")
AP2Server.Initialize (CFGFilePath, SDFFilePath)
```

Una vez que el Objeto Servidor ha sido creado (Pro/ii abierto), el siguiente paso es abrir una base de datos de la simulación actual (activar el Objeto Archivo). El Objeto Servidor que acaba de ser creado tiene el método a utilizar: OpenDatabase (abrir base de datos). El parámetro que debe ser proporcionado con este método es la ruta completa de la base de datos de la simulación. Para crear una nueva base de datos, el método NewDatabase debe utilizarse.

Ejemplo:

```
Set AP2File= AP2Server.OpenDatabase (ADbsFile)
```

Una vez que el archivo de la simulación está abierto, es posible activar los componentes individuales de la simulación usando el método perteneciente al Objeto Archivo, `ActivateObject`. Debido a que la base de datos de la simulación contendrá un gran número de objetos de simulación (separadores, bombas, componentes, métodos termodinámicos), el usuario tiene que proporcionar no sólo el tipo del componente que debe ser activado (clase), sino también el nombre del componente (debe haber más de un separador en una simulación).

Ejemplo:

```
Set AFlash= AP2File.ActivateObject (ObjectType, ObjectName)
```

Y aquí también los objetos deben ser del tipo 'Object'.

Ahora que la jerarquía de objetos está establecida completamente, es posible iniciar la interacción con la base de datos. Para esto, una variedad de métodos pertenecientes al Objeto Datos está disponible. Los métodos para recuperar y escribir la información que está disponible para todos los objetos, y ciertos objetos como las corrientes y los componentes, tienen otros métodos asociados con ellos (corrientes de separación, componentes y cálculos de corrientes).

Los métodos más comúnmente usados probablemente serán los de escribir y recuperar información. El método `GetAttribute` (obtener propiedad) requiere al menos un nombre de atributo asociado con el objeto. Por ejemplo, para recuperar la temperatura de una corriente, una vez que el Objeto Datos de la corriente ha sido creado, el siguiente comando debe ser usado:

```
dTemperature= AStream.GetAttribute ("Temperature")
```

2.4.2.3. Biblioteca de Clases Pro/ii

La Biblioteca de Pro/ii, aparece con el nombre de: PRO/II Class Library Version 5.61.

Después de que se agregó la biblioteca, el usuario tendrá acceso a tres tipos de variables nuevas, correspondientes a los tres objetos en la jerarquía de objetos OLE. Se debe saber que estos objetos tienen que ser creados con la palabra clave "New", para asegurar que una nueva instancia está siendo invocada.

```
Dim P2Server As New CP2Server
```

Uno de los otros beneficios de la Biblioteca de Clases de Pro/ii es que el usuario no tiene que especificar la eliminación de estos objetos (lo cual puede conducir a errores severos). Cuando las variables de estos objetos ya no se necesitan, los objetos correspondientes son puestos automáticamente a 'Nothing' (Nada). Sin embargo, si hay una razón específica para eliminar un objeto en cualquier momento, esto puede ser logrado utilizando el comando 'Set'.

Las aplicaciones tendrán ahora una configuración simplificada. No es requerido recuperar la localización de la configuración de los Archivos Esquema. Esto es realizado dentro de la biblioteca de clases interrogando a Windows Registry.

Ejemplo:

```
If P2Server. Initialize Then  
P2File = P2Server.OpenDatabase ("C:\Temp\MyFile.pr1")
```

```

varStreamNames = P2File.GetObjectNames ("Stream")
For i=0 to Ubound (varStreamNames)
    P2Stream = P2File.ActivateObject ("Stream", i)
    varComposition = P2Stream.GetAttribute ("TotalComposition")
    Set P2Stream = Nothing
Next i
End If

```

2.4.2.4. Comunicación.

Para realizar la comunicación de Pro/ii con Access, se utilizó la Transferencia de Datos OLE vista anteriormente, con ayuda de la Biblioteca de Clases del simulador.

Primero se agregó en las Referencias de Visual Basic, la biblioteca mencionada anteriormente, llamada PRO/II Class Library Version 5.61.

Después se crearon los tres objetos de acuerdo a la jerarquía de objetos OLE:

- Objeto Servidor. Este objeto es utilizado para iniciar y administrar la interacción con el Servidor OLE.
- Objeto Archivo. Este objeto encapsula las operaciones en una sola base de datos Pro/ii. Solo una base de datos de una simulación debe ser accedida en cualquier momento utilizando un solo Objeto Servidor.
- Objeto Datos. Los objetos individuales de datos en Pro/ii (separadores, corrientes métodos termodinámicos, etc.) son tratados como instancias de un Objeto Datos.

Para la transferencia de los datos (enviar y recibir) se utilizaron diversas propiedades y métodos de Pro/ii OLE Automation Layer, al cual permite acceso completo a las bases de datos de la simulación para leer y escribir. Cada Objeto contiene varias propiedades y métodos, pero solo se mencionaran los que se usaron de cada uno de ellos.

- Objeto Servidor. Este objeto debe ser creado e inicializado antes de que cualquier base de datos sea creada o abierta. De este objeto se utilizó lo siguiente:

- Propiedades:
 - ClearMsg. Elimina todos los errores, advertencias y mensajes desde el búfer interno.
 - MsgCount. Regresa el número de errores, advertencias y mensajes desde el búfer interno.
 - Status. Regresa el estado interno del Servidor. Un estado cero, significa que ningún error fue encontrado. Cuando esta propiedad es consultada, el estado se reinicia a cero. Esto es requerido para que cada llamada subsecuente al objeto servidor sea ejecutada correctamente.
- Métodos:
 - GenerateReport. Genera el reporte estándar en formato de texto de la base de datos proporcionada de Pro/ii. La base de datos debe haber sido ejecutada antes de llamar a este método. También, el acceso al servidor de la base de datos debe haber sido cerrado, llamando al método 'Unload', antes de que el método de generar reporte pueda ser llamado.
 - GetClassNames. Regresa el nombre de la clase del índice proporcionado.
 - Initialize. Inicializa el Servidor OLE abriendo los archivos de configuración y esquema, también construye estructuras de datos internas. Este método debe ser llamado antes de que cualquier otro método sea invocado. Para reiniciar (cerrar) el servidor de base de datos, se debe establecer la variable objeto a 'Nothing'.
 - MsgText. Regresa el texto de un error, advertencia o mensaje del búfer interno.

- NewDatabase. Crea y abre una base de datos nueva de Pro/ii. Para cerrar (reiniciar), se debe establecer la variable objeto a 'Nothing'. Se puede abrir una base de datos en cualquier momento, pero solo debe ser una. Si se desea crear o abrir otra base de datos utilizando el mismo objeto servidor, se tiene que cerrar primero la base de datos que está abierta actualmente.
- OpenDatabase. Abre una base de datos de Pro/ii existente. Para cerrar (reiniciar) la base de datos, la variable objeto debe ser establecida a 'Nothing'. Solo una base de datos puede ser abierta. Si se desea crear o abrir otra base de datos utilizando el mismo objeto servidor, se tiene que cerrar primero la base de datos que está abierta actualmente.
- Unload. Cierra el acceso al servidor de base de datos de Pro/ii. Sin embargo, el objeto servidor sigue activo. Este método necesita ser llamado antes de ejecutar el método RunCalcs o el método GenerateReport. Este método cierra el acceso a los archivos de configuración y esquema y reinicia las estructuras de datos internas. Si se quiere abrir la base de datos después de realizar los cálculos o generar el reporte, el método Initialize necesita ser llamado otra vez para abrir los archivos de configuración y esquema así como para reconstruir las estructuras de datos internas.

- Objeto Archivo.

- Propiedades:
 - Name. Regresa el nombre del objeto.
- Métodos:
 - ActivateObject. Recupera un Objeto Datos desde el disco a la memoria. Para liberar el objeto activado, se debe establecer la variable objeto a 'Nothing'.
 - CalculateStreamProps. Calcula un conjunto completo de propiedades para la corriente especificada.
 - CheckData. Realizar chequeos necesarios para la consistencia de los datos y construye la estructura del cálculo requerida para ejecutar la simulación. Este paso debe ser hecho antes de ejecutar la simulación, la cual se realiza con el método RunCalcs.
 - ConvertValue. Convierte un valor de punto flotante desde una unidad de medida a otra.
 - CreateObject. Crea y activa un nuevo Objeto Datos de la clase especificada. Para liberar este objeto, se debe establecer la variable a 'Nothing'.
 - DeleteObject. Borra un Objeto Datos de la clase especificada desde la base de datos (en el disco). Este proceso no puede ser deshecho.
 - GetObjectCount. Regresa el número de Objetos de Datos de la clase especificada.
 - GetObjectNames. Regresa el nombre de un objeto de acuerdo al índice especificado.
 - GetUOMString. Regresa la versión texto de una unidad de medida.

- Objeto Datos.

- Propiedades:
 - Name. Regresa el nombre del objeto.
- Métodos:
 - GetAttribute. Obtiene un valor escalar o un elemento de un arreglo.
 - GetAttributeCount. Regresa el número de atributos asociados con el objeto datos.
 - PutAttribute. Establece un valor escalar o un elemento de un arreglo.
 - UnitCount. Recupera el número de elementos de un arreglo en un atributo.

Todas estas propiedades y métodos son los que se utilizaron para tener acceso a las bases de datos de los objetos del Modelo de Simulación, se recuperaron datos desde el modelo, relacionados con los atributos y con

los resultados de los cálculos, y se transfirieron a la base de datos, también se ingresaron datos de entrada al Modelo de Simulación y al mismo tiempo se guardaron en la base de datos.

2.5. Análisis de Datos.

La información es un producto de vital interés tanto para los directivos como para los usuarios finales. Habrá de evaluarse la capacidad de cada sistema de información para producir información útil relativa a los problemas y las oportunidades. Mejorar la información no quiere decir generar grandes volúmenes de información. Por el contrario, la sobrecarga de información es uno de los principales problemas de las empresas, que puede percibirse fácilmente viendo las enormes pilas de papel que emergen de las salidas informáticas.

El análisis de datos se refiere a la separación de una masa de datos en sus partes componentes y al estudio y manipulación de esas partes con el propósito de lograr el resultado que se desea. Por supuesto, es preciso que se tenga acceso a los datos requeridos antes de poder realizar un análisis, y también se debe tener un concepto claro de los resultados que se necesita obtener.

Con el análisis de datos, se detectaron situaciones en la organización, en las que se requieren mejoras en la información, entre las que se incluyen:

- Falta de información en lo que concierne a decisiones o situaciones actuales.
- Falta de información relevante en lo que concierne a decisiones o situaciones actuales.
- Información que no se expresa en forma útil para su gestión.
- Falta de la información en el momento oportuno.
- Exceso de información.
- Información inexacta.

Todos estos sucesos surgieron debido a la inexistencia de un sistema de información. Además de las situaciones mencionadas también se identificaron los siguientes problemas:

- Redundancia de datos. Los mismos datos han sido capturados y/o almacenados en varios lugares. La redundancia de datos consume un valioso espacio de almacenamiento y crea problemas de integridad de datos.

- Inflexibilidad de datos. Los datos que si se capturaron y almacenaron no estaban bien organizados de modo que resulta muy difícil obtener ciertos informes o consultas.

Al reunir y analizar los datos que se utilizan en los procesos se encontraron los siguientes:

- Parámetros de diseño de aparejos de producción en pozos (fluyentes y/o artificiales).
- Parámetros de diseño de cabezales y ductos.
- Parámetros de diseño de instalaciones de producción.
- Parámetros operativos de pozos (fluyentes, artificiales).
- Parámetros operativos de ductos (oleogasductos, oleoductos, gasoductos, gasolinoductos, acueductos).
- Parámetros operativos de instalaciones de producción (separación, endulzamiento de gas, manejo de condensados).
- Gastos producidos de hidrocarburos (aceite, gas, condensados, agua).
- Propiedades de los fluidos producidos (composicionales de mezclas, caracterizaciones de aceite, cromatografías de gas, viscosidades, densidades, temperaturas, etc).
- Ubicaciones geográficas (coordenadas).

Todos ellos, se unificarán en el sistema de información para simulación y optimización de procesos de producción de hidrocarburos a desarrollar. Los datos más importantes que se utilizan en los procesos y a los que se les almacenará en mayor cantidad que los demás son: presión, temperatura, nivel y flujo.⁵⁷

Presión. Una de las variables que se presenta en la mayoría de los procesos y que debe ser lo mejor controlada posible es la presión, por el alto nivel de riesgo que se presenta cuando la variable está fuera de control.

Temperatura. La medida de temperatura constituye una de las mediciones más comunes y más importantes que se efectúan en los procesos industriales. La temperatura es la energía media de las moléculas de un cuerpo o, como coloquialmente se dice, es el grado de calor o frío medido en una escala definida.

Las escalas comúnmente usadas para medir esta variable son:

- a) Escala Celcius.
- b) Escala Kelvin.
- c) Escala Rankine.
- d) Escala Fahrenheit.

Nivel. En la industria de la medición de nivel es muy importante, tanto desde el punto de vista del funcionamiento correcto del proceso, así como de la consideración del balance adecuado de materias primas o de productos finales.

Flujo. La variable de proceso más medida en la industria es el flujo. La medición y el control de esta variable son muy importantes tanto para el buen funcionamiento de los diversos procesos industriales como para la consideración del balance de materias primas o de productos finales.

Todas las variables se tienen que manejar correctamente dentro del sistema de información, para poder solucionar las situaciones y problemas encontrados en algún proceso.

La importancia de la administración correcta de los datos es la optimización, es decir, obtener los datos o valores óptimos, en cuanto a funcionamiento, de cada unidad de operación desde los simuladores y almacenarlos en el sistema, y así los tomadores de decisiones busquen la mejor solución, después de un arduo análisis en las bases de datos, para la realización eficaz y eficiente de un proceso y aplicarlo a la realidad. Con todo esto se logra también una reducción en costos y en tiempo de ejecución.

2.6. Alternativas existentes para la integración de los datos al sistema de información.

Ya habiendo analizado los datos más importantes que utilizan las aplicaciones de simulación se pueden escoger algunos métodos de integración de los mismos al Sistema de Información. A continuación se presentan algunos, para después plantear una propuesta de solución.

2.6.1. Métodos para la integración de los datos al sistema de información.⁵⁸

La perforación de las tarjetas era el caballo de batalla en el mundo del procesamiento de datos, pero ahora la superan métodos más eficientes y más precisos. La perforación era tediosa y los datos en las tarjetas tenían que capturarse dos veces: una vez para su perforación y la segunda para su verificación. Además, si las tarjetas se introducían en otro orden, la tarea de cómputo o bien los datos carecían de sentido.

⁵⁷ Referencia 28. Villanueva Espinosa, Francisco Javier. México, D.F., Noviembre de 1998, 235 pp.

⁵⁸ Referencia 13. Kendall y Kendall. México, 1991, 881 pp.

En la actualidad se dispone de distintos métodos, todos eficientes para la entrada de datos, y la elección entre ellos queda determinada por diversos factores que incluyen la necesidad de velocidad, la precisión y la capacitación del operador; el costo del método de captura (ya sea intensivo en materiales o en mano de obra) y de los métodos de uso actual dentro de la organización.

Las decisiones aplicadas suelen ser sencillas (por ejemplo, trabajar en modo *batch* o en modo *on-line*). En la actualidad, debemos considerar también alternativas más modernas, como el *batch* remoto, la entrada de datos sin teclado, la introducción de datos mediante lápices, las interfaces gráficas de usuario, el intercambio electrónico de datos, el intercambio de imágenes y documentos, etc.

- Tecla-almacenamiento. Los procesos de almacenar directo desde el teclado, incluyen teclas-a-cinta, teclas-a-disco y teclas-a-disco-flexible, tienen mejoras sobre la perforación de tarjetas en la captura de datos. Los tres métodos anteriores requieren del teclado para la entrada de datos, y por ello, toman tal derivación común en su nombre.
 - Teclas-a-cinta es el método más antiguo de los tres y es una manera de introducir de manera eficiente grandes cantidades de datos. Desde un teclado, un operador introduce datos a una cinta magnética. Esto evitaba el problema del mezclado o pérdida de las tarjetas en papel, existentes en el método antiguo de entrada de datos de perforación de tarjetas. Además, los sistemas de registros de teclas-a-cintas pueden programarse para verificar las entradas, la búsqueda de material previamente almacenado y obtener productividad en el manejo de datos. Las cintas pueden transportarse al centro de procesamiento o transmitirse en línea.
 - De manera similar al proceso de teclas-a-cintas es la entrada de datos de teclas-a-disco, la cual utiliza disco en lugar de cintas como medio de almacenamiento. Además, los operadores utilizan un monitor para definir las instrucciones acerca de su trabajo. El proceso teclas-a-disco también permite el transporte de discos concluidos a los centros de procesamiento o se pueden diseñar sistemas que transmitan en línea.
 - Teclas-a-disco-flexible. Su facilidad de actualización y el intercambio de archivos en los discos flexibles, junto con la facilidad para transportarlos hacen una excelente elección para pequeñas tareas. Sin embargo, su almacenamiento limitado los hace poco prácticos para tareas de mayor volumen. En la actualidad ya existen los discos compactos en los que se puede realizar igualmente la entrada de datos y tienen mayor capacidad de almacenamiento.

Este tipo de entrada de datos requiere de una microcomputadora y de software para permitir las funciones de edición y de control; de una pantalla que proporcione las instrucciones al operador; de manejadoras de discos y de discos flexibles compatibles con el sistema. La entrada de datos puede realizarse de manera independiente sin el uso de una macrocomputadora o puede realizarse en línea si se cuenta con el equipo, acelerando el proceso.

- Formas sensibles a marcas. Las formas sensibles a marcas permiten la captura de datos a través de un digitalizador que verifica el sitio de las marcas que el lápiz de plomo hizo sobre las formas especiales. Un uso común es la toma de respuestas de cuestionarios de encuestas. El personal capturista necesita muy poca capacitación.
- Formas perforadas. Las formas perforadas son una alternativa rápida y conveniente para capturar e introducir una cantidad reducida de datos. Las formas perforadas se utilizan para propósitos de votación, cuando los votos se deben de contar de manera rápida.
- Cintas de datos. Las cintas o bandas de datos codifican textos, gráficas y datos en pequeños puntos rectangulares sobre una cinta angosta de papel. Esta cinta, la lee la microcomputadora mediante un detector infrarrojo. Esto es altamente compacto y una manera muy confiable de capturar los datos; las cintas de datos no son tan susceptibles a daños ambientales (como podría ser derramamiento de café), como ocurre con los medios magnéticos.

A diferencia de un disco flexible, las cintas de datos pueden ser usadas para publicarse fácilmente en libros y en revistas a un costo razonable, reduciendo en gran medida el trabajo de transferir del teclado al disco lo que en un principio el programa de computadora requeriría capturar a partir de un listado

impreso. También es posible que el usuario genere su propia tira de datos a través de un software especial.

- Entradas en modo batch. En el proceso por lotes (batch), las transacciones se acumulan en lotes para su tratamiento periódico. Las entradas por lotes se procesan confrontándolas con los archivos o bases de datos maestros. Los archivos o bases de datos de transacciones pueden también crearse o actualizarse a partir de transacciones.

Al contrario de lo que suele pensarse, las aplicaciones de tipo batch no están obsoletas. Las necesidades de algunas aplicaciones conducen de forma natural hacia el tratamiento por lotes. Tal vez porque las entradas llegan en lotes naturales (por ejemplo, por correo).

- Entradas en modo on-line. Los sistemas en línea (on-line) ofrecen la posibilidad de mantener un diálogo o conversación entre el usuario y la computadora. A menudo es más aconsejable procesar las transacciones y las consultas de empresa en el momento en que se producen. Ello hace posible identificar y corregir los errores con más rapidez, y procesar antes las transacciones, ya que los sistemas en línea eliminan la necesidad de preparar archivos de datos por lotes. Más aún, los métodos en línea permiten una mayor participación de las personas en la toma de decisiones, incluso cuando los datos llegan por lotes de forma natural.

La necesidad de obtener tiempos de respuesta menores en la mayoría de las aplicaciones y el deseo de la interacción humana en el proceso han llevado a buscar alternativas on – line en el desarrollo de sistemas. Otros factores que han contribuido a ello ha sido la disponibilidad de computadoras más rápidas, la mayor capacidad para dar servicio a un mayor número de usuarios a la vez y la mejora en la calidad del software de control y de los desarrollos on – line.

Las aplicaciones en clientes-servidores son, sencillamente, una nueva forma de los procesos on – line. La edición de las entradas se realiza en las computadoras clientes en un modo on – line. Las transacciones de entrada se transmiten en línea a varias computadoras para su proceso.

- Batch remoto. El tratamiento por lotes remoto (batch remoto) combina los mejores aspectos del tratamiento de entradas en modos batch y on – line. Computadoras distribuidas en línea gestionan la introducción y la edición de datos. Las transacciones editadas se recogen en un archivo batch para su ulterior transmisión a las computadoras centrales que procesan dicho archivo por lotes.

El tratamiento por lotes remoto difícilmente puede considerarse como una alternativa novedosa, pero las computadoras personales han hecho de esta opción algo cada vez más atractivo. Las PCs ofrecen una notable potencia de captura y edición de datos en línea y a bajo costo, así como un modo de almacenamiento adecuado para los lotes resultantes. Las PC con interfaces gráficas de usuario (por ejemplo, Windows) ofrecen medios más sencillos de introducción de datos y asistencia al usuario.

El tratamiento por lotes remoto basado en computadoras personales debería dar un nuevo impulso a la tecnología de computadoras portátiles y agendas electrónicas. Estas computadoras, que pesan desde kilo y medio a apenas cuatro kilogramos, pueden utilizarse para recoger lotes en cualquier parte, incluidas aplicaciones locales, domésticas, universitarias, etc.

- Introducción de datos sin teclado. Los errores mecanográficos han sido siempre una importante fuente de errores en la introducción de datos informáticos. Cualquier tecnología capaz de reducir o eliminar estos errores mecanográficos debería tenerse en cuenta a la hora de diseñar un sistema.

En los sistemas batch, los errores mecanográficos pueden eliminarse mediante tecnologías de lectura óptica de caracteres (OCR), reconocimiento de caracteres en tinta magnética (MICR) y lectura óptica de marcas (OMR). Ambas opciones viables para el diseño de entradas de datos.

Los auténticos avances en la introducción de datos sin teclado han llegado de la mano de los sistemas on – line. Los sistemas de códigos de barras (similares a los sistemas universales de códigos de productos que son habituales en los supermercados y las tiendas de alimentación) están ampliamente disponibles en muchas aplicaciones modernas. Los códigos de barras pueden considerarse como "metacódigos" o códigos que codifican a otros códigos. Aparecen como una serie de bandas angostas y anchas en una etiqueta, la cual codifica números o letras. Estos símbolos, a su vez, tienen acceso a los datos del producto almacenados en la memoria de la computadora. Un haz de luz de un digitalizador pasa sobre las bandas de las etiquetas para confirmar o registrar los datos del producto que se evalúa.

La introducción de datos sin teclado debería considerarse como una alternativa especialmente válida para aquellos sistemas basados en un alto volumen de transacciones que tengan que ser rediseñados.

- Introducción con lápices. La informática basada en el uso de lápices ópticos y electrónicos está experimentando una rápida evolución. Conforme los sistemas operativos basados en lápices (como Pen Windows, de Microsoft) tengan un uso más extendido y pueda disponerse de herramientas para la construcción de aplicaciones basadas en estos lápices, es de esperar que aparezcan más diseños de sistemas que apliquen esta tecnología. Algunas empresas han hecho ya uso de la misma para la captura remota de datos.
- Interfaces gráficas de usuario. Las interfaces gráficas de usuario adquirieron gran notoriedad por el éxito de Apple Macintosh y Microsoft Windows. Mientras que el éxito comercial llegó de la mano de aplicaciones como los procesadores de texto y las hojas de cálculo, la popularidad de la interfaz fue atrayendo hacia ella a numerosas aplicaciones.
Existen técnicas para crear auténticas interfaces gráficas de usuario basadas en PC que funcionen en aplicaciones de computadoras centrales a través de un proceso cooperativo. Y lo que es más importante, la tecnología de interfaces gráficas de usuario se ha convertido en la interfaz de usuario idónea para las aplicaciones de entornos clientes-servidores.
Las interfaces gráficas de usuario no hacen mejores automáticamente a las aplicaciones. Interfaces de este tipo mal diseñadas pueden echar a perder las ventajas mencionadas de las interfaces de usuario. Por suerte, los estándares de interfaces gráficas de usuario han experimentado una evolución que les permite llevar a los diseñadores de sistemas a crear interfaces coherentes. Las interfaces gráficas de usuario adecuadamente diseñadas simplifican la introducción, limitan la labor mecanográfica requerida y ofrecen opciones útiles e interesantes de diseño de formatos.
- Intercambio electrónico de datos. Las empresas que operan en muchos lugares distintos y aquellas que buscan mayor eficacia en el intercambio de transacciones con sus proveedores y/o clientes utilizan a menudo el intercambio electrónico de datos (IED). El IED es el flujo electrónico de transacciones de empresa entre los clientes y los proveedores. Con el IED, una empresa puede eliminar su dependencia de los documentos en papel y del correo. Las ventajas competitivas obtenidas por la reducción del tiempo de respuesta son evidentes. Es de esperar que, hacia el final del decenio, las grandes empresas realicen las transacciones con sus clientes y proveedores a través del intercambio electrónico de datos.
- Intercambio de imágenes y documentos. Otra tecnología en auge de gestión de entradas es la basada en el intercambio de imágenes y documentos. Similar a la del intercambio electrónico de datos con la salvedad de que lo que se transmite y se recibe son imágenes reales de formularios y datos, esta tecnología resulta de especial utilidad en aplicaciones en que se requieran imágenes o gráficos de los formularios. Otras aplicaciones de imágenes combinan datos con dibujos o gráficos.

Es fundamental asegurar la calidad de los datos que entran al sistema de información para asegurar una salida de calidad. La calidad de los datos que se introducen puede mejorarse, apegándose a tres objetivos principales de la captura de datos, los cuales son: la codificación eficaz, la captura de datos efectiva y eficiente, y la validación de los datos. Se debe suponer que los errores en el proceso de captura ocurrirán y se debe trabajar con los usuarios para diseñar pruebas de validación que prevengan que los datos equivocados se procesen y se almacenen.

Las transacciones de las entradas deben verificarse para asegurar que la transacción solicitada sea aceptable, autorizada y correcta. La entrada de datos puede validarse a través de la incorporación en el software de diversas pruebas de verificación para datos que pudieran estar equivocados, longitud de los datos, rango y razonabilidad de los datos y valores no válidos. La entrada de datos puede compararse con los datos almacenados, con el fin de validarlos. Una vez que se han capturado, puede verificarse y corregirse de manera automática mediante el uso de dígitos de verificación.

2.7. Propuesta de solución para la integración de los datos al sistema de información.

Gracias al análisis de datos se encontró que la mayoría de los datos son numéricos, algunos de texto, y otros son imágenes. Algunos de éstos datos se encuentran impresos en papel, otros en archivos de computadora y los otros serán los que se produzcan como resultados de las simulaciones.

En la Figura 43 se observa la manera en la que los datos podrían ser introducidos al sistema de información, en este caso a la base de datos.

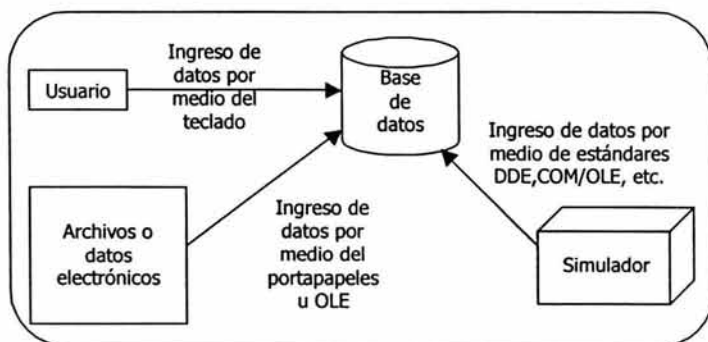


Figura 43. Introducción de datos al sistema de información.

Por un lado se encuentra el usuario que ingresará los datos impresos por medio del teclado a las tablas en la base de datos, esto debido a que no se cuenta con infraestructura para realizarlo de otra manera.

Los archivos electrónicos se pueden integrar por medio del portapapeles de Windows o a través de OLE, gracias a la compatibilidad de los mismos con la base de datos que se realizará en Access.

Y por último, los datos que resulten de los procesos de simulación pueden ser guardados directamente en la base de datos, esto es, el sistema contendrá un módulo que indique, que después de realizada la simulación, se guarden los datos en una tabla específica, con ayuda de los estándares que se puedan utilizar como OLE, DDE, etc.

Después de integrar la información a la base de datos, tanto la de proceso como la requerida y generada por los simuladores, se tiene que crear la interfaz para la comunicación de datos entre las aplicaciones y éstas con la base de datos. Para que la interfaz sea estándar, debe ser realizada utilizando lenguajes y estándares compatibles con Microsoft, ya que todos los simuladores a utilizar están hechos bajo esta plataforma y se comunican con la misma.

En la Figura 44 se observa cómo quedaría la aplicación realizada con los simuladores interactuando entre sí por medio de una interfaz o interfaces.

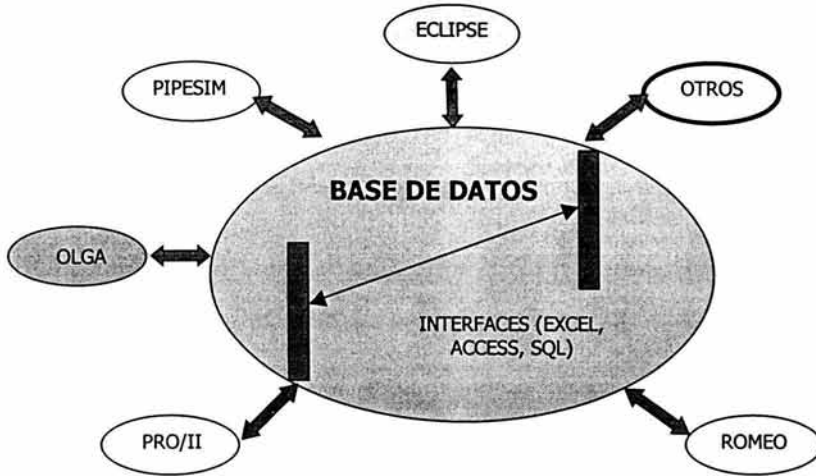


Figura 44. Plataforma estándar para comunicación de datos entre las diferentes aplicaciones (software de simulación).

Las herramientas de simulación utilizadas en cada uno de los procesos de producción, como son ECLIPSE, PIPESIM, OLGA, PRO/II Y ROMEO, reciben datos de la base de datos como entradas para realizar la simulación y/o optimización, posteriormente, ya realizada la simulación envían datos de salida a la base de datos. La interfaz se encarga de organizar los datos y de determinar las variables necesarias a ser enviadas desde un simulador a otro y desde la base de datos a un simulador. Se obtendrán como resultados valores óptimos para la realización del proceso y una buena toma de decisiones gracias al análisis de los mismos.

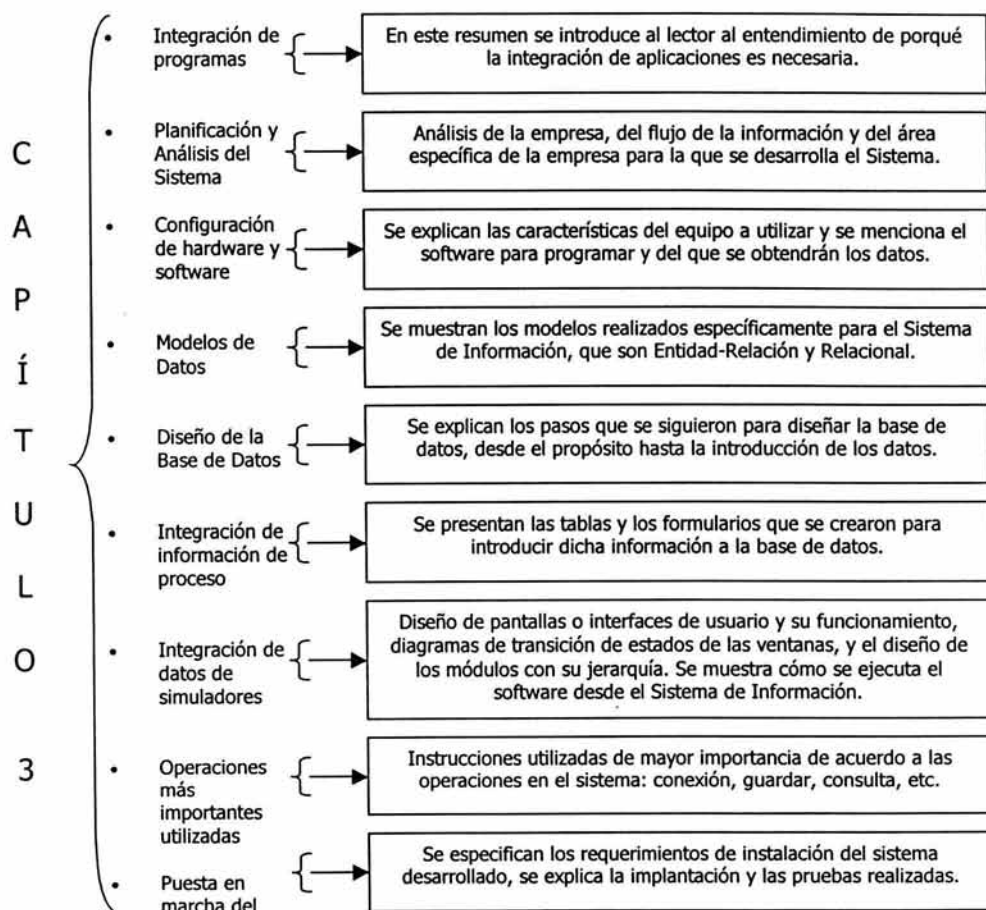
CAPÍTULO 3

DESARROLLO DEL SISTEMA DE INFORMACIÓN PARA LA SIMULACIÓN DE PROCESOS DE PRODUCCIÓN DE HIROCARBUROS

CAPÍTULO 3. DESARROLLO DEL SISTEMA DE INFORMACIÓN PARA LA SIMULACIÓN DE PROCESOS DE PRODUCCIÓN DE HIDROCARBUROS.

El problema principal a atacar en el trabajo de esta tesis es la integración entre aplicaciones de diferentes empresas proveedoras, para tener ahorros tanto en tiempo y recursos.

En este capítulo se abarca todo el desarrollo del Sistema de Información, en el siguiente cuadro se muestra la composición del mismo.



3.1. Razones para integrar programas *

Algunas personas se dan cuenta de que un programa les permite realizar todo lo que desean hacer con la computadora, pero otros necesitan correr varios programas con regularidad, tal necesidad puede causar varias frustraciones. Suponga que un usuario está escribiendo un informe con un paquete de procesamiento de

* Se utilizarán indistintamente las palabras: programas y aplicaciones.

palabras y decide incluir algunos datos de hoja electrónica en el informe. Si se emplean programas de proceso de texto y hoja electrónica de cálculo separados, es posible que el usuario tenga que grabar el documento, suspender el programa de proceso de textos, cambiar discos, cargar el programa y los datos de la hoja electrónica de cálculo y recuperar los datos por medio de un conjunto de comandos que no tienen parecido alguno con los que se utilizaban en el programa de procesamiento de palabra. Una vez obtenida la información de la hoja electrónica, el usuario quizá tenga que imprimirla, recortarla y pegarla sobre el informe, tendría que hacer antes todo esto porque con frecuencia es difícil, si no imposible, hacer que los programas separados de función única compartan datos. La integración de programas reduce la frustración. De manera más específica, la integración de programas:

- Facilita que los usuarios pasen de una aplicación a otra, muchas veces mientras se trabaja simultáneamente con ambas aplicaciones en el mismo procesador.
- Les permite transferir datos libremente entre estas aplicaciones.
- Proporciona una interfaz común que hace a los programas más fáciles de aprender y utilizar porque es posible realizar tareas similares mediante comandos similares.

3.1.1. Integración de las aplicaciones en los sistemas de información

En términos históricos, las aplicaciones y funciones han vivido en su mayoría un desarrollo aislado unas de otras. Ello ha conducido a lo que algunos especialistas han dado en llamar islas de información. La integración de sistemas pretende conectar dichas islas entre sí del mejor modo posible, de manera que puedan compartir los datos y las informaciones comunes.

La integración de sistemas es el proceso de resolución de problemas cuyo fin es construir aplicaciones de sistemas de empresa y de información que trabajen entre sí de forma cooperativa, compartiendo los datos, la información y otros recursos disponibles.

En ninguna parte es más evidente esta tendencia que en la industria de fabricación. Los fabricantes han acelerado la automatización de los procesos complejos de producción mediante el uso de robots y controladores. Los sistemas de información para fabricación han sido desarrollados, la mayor parte de las veces, de forma relativamente aislada con respecto a tales procesos de automatización de las fábricas. Sin embargo, en las empresas más avanzadas, los sistemas de información se han ido integrando en los sistemas de fabricación a través de un concepto denominado fabricación integrada por computadora (Computer Integrated Manufacturing, o CIM).

El sistema de información a desarrollar, es precisamente de la industria petrolera, en el área de producción de hidrocarburos. Por lo tanto, con esta pequeña introducción se ve la importancia de unir en una sola base de datos los datos de varias herramientas de simulación.

3.2. Planificación y Análisis del sistema.

El sistema de manejo de información se desarrolló de acuerdo a las fases del ciclo de vida mencionadas en el Capítulo I. La descripción de las etapas se realizará en todo este capítulo.

3.2.1. Planificación del sistema

La planificación del sistema consta de tres fases: estudiar el cometido de la empresa, analizar el flujo de información y analizar un área de empresa.

1. Estudiar el cometido de la empresa.⁵⁹

- Nombre de la empresa: Instituto Mexicano del Petróleo (IMP).
- Ubicación: El Instituto Mexicano del Petróleo sede se ubicada en *Eje Central Lázaro Cárdenas No. 152 Col. San Bartolo Atepehuacan, Delegación Gustavo A Madero*, al norte de la Ciudad de México. Cuenta con 33 edificios, en donde se realizan principalmente actividades de investigación, innovación y desarrollo de tecnologías para el apoyo a PEMEX y a la industria nacional.
- Giro.

El Instituto Mexicano del Petróleo (IMP) ha sido desde su creación, una importante plataforma para la investigación científica y el desarrollo tecnológico al servicio de las industrias petrolera, petroquímica básica, petroquímica derivada y química. Hoy es, además, una institución moderna y competitiva que se propone asegurar el fortalecimiento de la investigación y el desarrollo tecnológico, con programas y proyectos de investigación de punta; mantener una sana capacidad de autofinanciamiento; orientar sus esfuerzos hacia soluciones con servicios integrados a plena satisfacción de Petróleos Mexicanos, su cliente principal, y fortalecer sus competencias institucionales.

- Historia.

Surgimiento. El Instituto Mexicano del Petróleo (IMP) -organismo público descentralizado del Gobierno Federal, sectorizado en la Secretaría de Energía- se creó el 23 de agosto de 1965 como consecuencia de la transformación industrial del país y de la necesidad de incrementar la tecnología relacionada con el desarrollo de las industrias petrolera, petroquímica básica, petroquímica derivada y química.

El presidente Gustavo Díaz Ordaz aprobó el decreto que se publicaría en el Diario Oficial, en el cual se establecen como objetivos fundamentales: crear programas de investigación científica básica y aplicada; formar investigadores; desarrollar tecnologías aplicables a la técnica petrolera, y capacitar personal en todos los niveles.

Inicio de Actividades. El Instituto Mexicano del Petróleo inició sus actividades con trescientos empleados y cuatro edificios, para labores de investigación y administrativas. Fue creado para generar tecnología petrolera propia y así reducir los altos gastos que existían por concepto de importación de la misma.

El IMP nació por iniciativa del entonces director general de PEMEX, Jesús Reyes Heróles, quien reconoció que la planeación y el desarrollo de la industria petrolera deberían ser congruentes con las necesidades de una economía mixta y planteó al presidente Gustavo Díaz Ordaz la urgencia de fomentar la investigación petrolera y formar recursos humanos que impulsaran el desarrollo de tecnología propia.

El gobierno federal decidió crear un "organismo descentralizado de interés público y preponderantemente científico, técnico, educativo y cultural, con personalidad jurídica y patrimonio propios, cuya función será buscar la independencia científica y tecnológica en el área petrolera".

- Misión.
 - Transformamos el conocimiento en realidades industriales innovadoras.

⁵⁹ Referencia Sitios de Internet 14. <http://www.imp.mx>

- Visión.
 - Somos una Institución de excelencia enfocada a la Industria Petrolera, cuyo negocio es la innovación orientada al cliente y su capital el conocimiento; por lo que estamos centrados en la investigación y el desarrollo tecnológico para generar soluciones de alto valor.
 - Comercializamos productos de alto contenido tecnológico con calidad, oportunidad y precios competitivos, y generamos valor a nuestros clientes de lo que resulta una amplia solvencia financiera.
 - Nuestro modelo de atención al cliente ha permitido que PEMEX y el IMP sean socios estratégicos y tecnológicos; asimismo, ha resultado en una creciente participación en otros mercados.
 - La excelencia de nuestra operación es un parámetro de referencia en la industria petrolera; contamos con personal experto, comprometido y bien recompensado.
 - Nos constituimos como líderes en la administración del conocimiento; formamos líderes y especialistas abocados a la innovación tecnológica.
- Slogan. *Somos una comunidad de aprendizaje al servicio de la Industria Petrolera.*

2. Análisis del flujo de información.

En el Capítulo 2 se habló de la importancia de la optimización de procesos, la cual es parte esencial en la competencia de producción de hidrocarburos. En la Figura 45 vemos la pirámide de optimización de procesos de producción.

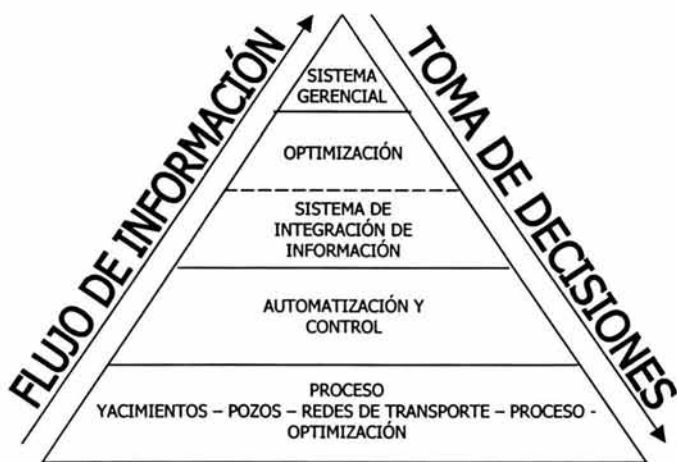


Figura 45. Pirámide de optimización de procesos.

El flujo de información inicia desde los datos obtenidos en los procesos (de yacimientos, pozos, redes de transporte etc.), para de ahí continuar hacia la automatización y el control, es decir los PLC's o DCS's, que son la infraestructura y los sistemas que controlan y monitorean el manejo de las variables de proceso en los instrumentos. Posteriormente la información se tiene que almacenar en el sistema que integre la información, de ahí hay un análisis y se obtienen los valores óptimos, finalizando con el Sistema Gerencial, que es en donde se toman las decisiones relacionadas al movimiento de algunos de los valores en cada uno de los instrumentos para la mejora de los procesos de producción.

Para entender mejor cómo se da el recorrido de los datos desde, es importante conocer un poco sobre automatización, ya que ésta es básica para lograr la optimización de los productos.

La automatización de procesos industriales, sin duda alguna, constituye una herramienta que permite mejorar la productividad de los procesos, y la calidad de los productos entre otros aspectos. Los sistemas que proveen dicha automatización dentro de la planta son aquellos elementos necesarios para la toma de decisiones, que en algunos casos pueden hacerse de manera automática a través de sistemas informáticos.

A este procedimiento, que va desde, la recopilación de información de campo, pasando por el manejo de información, hasta la toma de decisiones se le conoce como "*Pirámide de Automatización*", el cual se explica a continuación.

Pirámide de Automatización.

A través de la "*Pirámide de Automatización*". Se puede visualizar los niveles en que se ubican los instrumentos, equipos de control y sistemas de visualización que permiten alcanzar dicha automatización, así como la forma en la que fluye la información entre ellos, para garantizar la integración real de las diferentes áreas (operativas, mantenimiento, comercialización, etc.), pertenecientes a una planta productiva.

En la actualidad las instalaciones de producción, manejo y distribución, de gas y crudo, requieren asegurar una mayor eficiencia de sus recursos para mejorar la productividad y el control de calidad de los hidrocarburos. Lo anterior se puede lograr mediante la implantación de sistemas de automatización integrales que permitan el intercambio de información desde el piso de la planta (proceso) pasando a través de los diferentes niveles de la pirámide, hasta llegar al nivel más alto, que es la toma de decisiones la cual se muestra en la Figura 46.

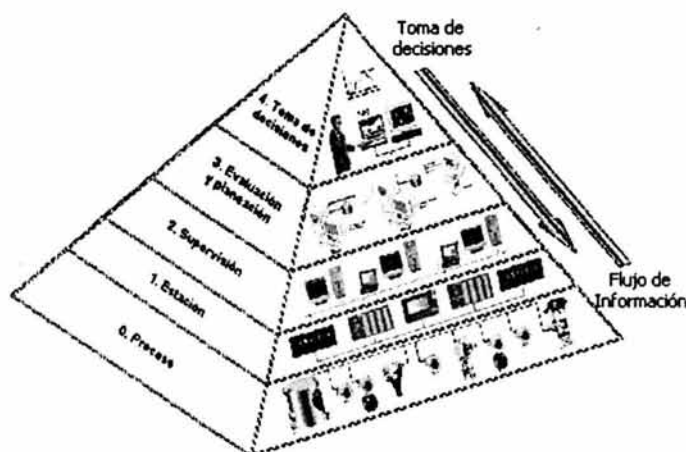


Figura 46. Pirámide de Automatización

Proceso (Nivel 0)

En este nivel se encuentran los elementos de campo o elementos primarios de control, como son: transmisores, registradores de presión o flujo, detectores de fuego, gas, etc. De todos estos elementos se obtiene información acerca del proceso. Por otro lado tenemos los elementos finales de control, que ejecutan acciones para mitigar o disminuir las consecuencias de una falla, dichos elementos van desde válvulas de control, válvulas de corte, bombas hasta sistemas de supresión de fuego.

Estación (Nivel 1)

En este nivel se encuentran los equipos de control "inteligentes" como lo son los Controladores Direccionables o Controladores Lógicos Programables (PLC's). Es en este nivel donde se lleva a cabo la primera estrategia de control, responsable de procesar la información recogida del primer nivel de la pirámide, como primer paso básico y local.

Supervisión (Nivel 2)

En este nivel se encuentran los equipos de computo, que manejan herramientas de software, para facilitar la recopilación e integración de toda la información procesada por los niveles 0 y 1, además desempeña el papel de interfaz con el operador, también, aspectos funcionales como: manejo de información en gráficos de tendencia en tiempo real e históricos, reporte de alarmas y eventos, control estadístico de procesos, importación-exportación de los archivos, etc. En este nivel se empieza a tener interacción con los operadores.

Evaluación y Planeación (Nivel 3)

Este nivel es responsable del control de la producción basado en la información obtenida del proceso que es estudiada y analizada, de manera que el proceso pueda ser llevado a su nivel óptimo de operación. Auxiliados por las herramientas de software adecuadas para el proceso específico y asesores especializados en el proceso. El uso de todas estas herramientas está encaminado al mejoramiento de la calidad en el desarrollo de procesos.

Toma de Decisiones (Nivel 4)

El último nivel de la pirámide, requiere de algunas herramientas del nivel 3. El equipo de computo esencialmente para la toma de decisiones que deberá tomar la información procesada del nivel 3, tal como son los datos históricos en tiempo real del proceso, para que los mandos superiores obtengan un resumen de por los menos tres tipos de información; datos de operación, de diagnóstico, de mantenimiento y de seguridad.

Estos datos son obtenidos de la base global del sistema. La cual cuenta con un sistema de manejo de información que concentra los datos generados en cada uno de los niveles de la pirámide de automatización con lo cual se tienen los elementos para que los mandos a nivel gerencial tomen las decisiones pertinentes.

En la Figura 47 se muestran los elementos principales a considerar en cada uno de los niveles de la pirámide de automatización.

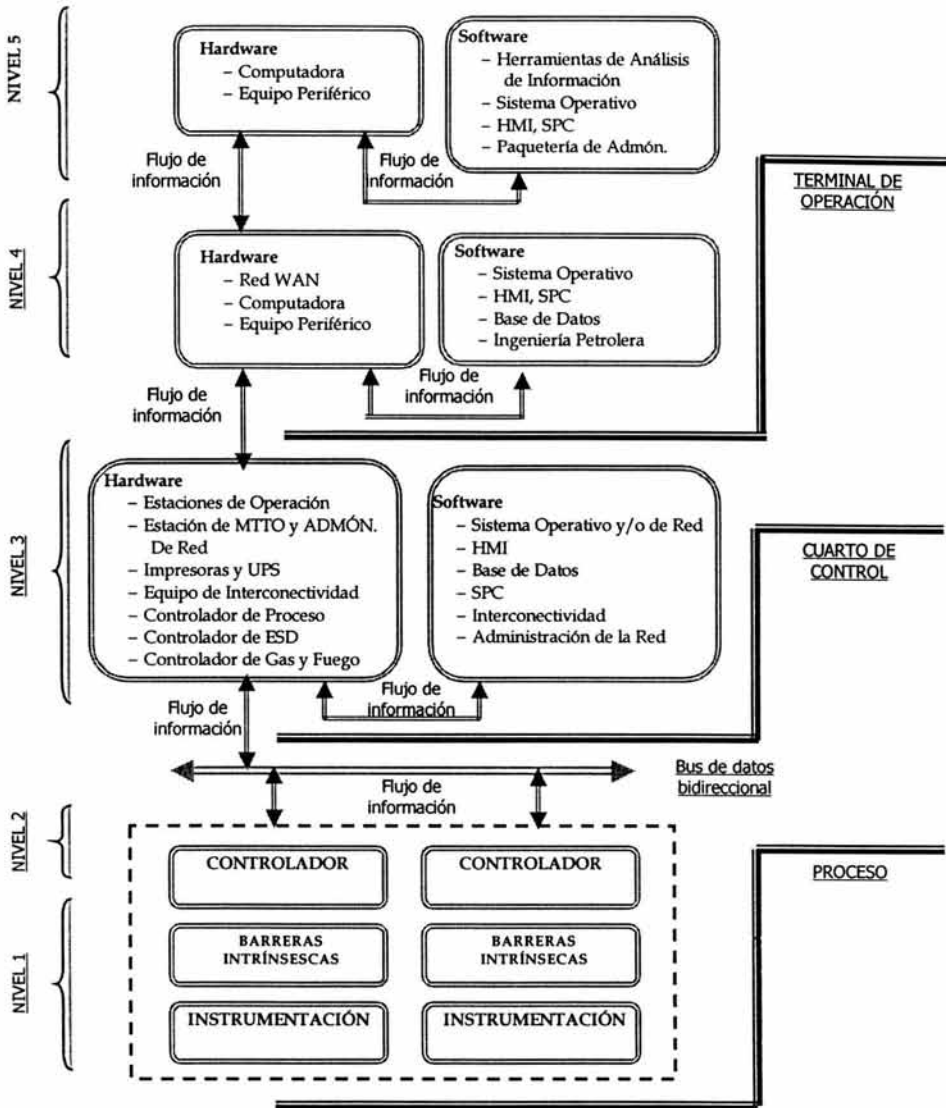


Figura 47. Directrices Tecnológicas en la Pirámide de Automatización

La estructura comúnmente usada para realizar un proyecto de automatización mostrada aquí, brinda un panorama general, sobre como fluye la información a través de los diferentes niveles de la pirámide de automatización, esto da una orientación de qué elementos y factores intervienen en la implementación de sistemas de automatización en instalaciones petroleras, que en determinado momento pueden afectar la eficiencia del sistema.

3. Análisis de una área de empresa.

En la Figura 48 se muestra la Estructura Organizacional del IMP. Dentro de ella, el área a analizar y para la cual se realizará el Sistema de Información se encuentra en la parte de Direcciones Ejecutivas, específicamente en la Dirección Ejecutiva de Exploración y Producción. Ésta área es denominada Grupo de Optimización de Instalaciones de Producción.



Figura 48. Organigrama del IMP

Grupo de Optimización de Instalaciones de Producción (GOIP).

El Grupo de Optimización de Instalaciones de Producción de la Gerencia de Tecnología de Producción, desarrolla actividades de asesoría y asistencia técnica especializada para PEMEX Exploración y Producción (PEP), en la optimización y automatización integral de instalaciones de producción terrestres y marinas.

Para el cumplimiento eficaz y eficiente de sus actividades, este grupo mantiene un programa permanente de capacitación especializada en México y en el extranjero, así como, la búsqueda continua de tecnologías en las áreas de optimización y automatización en el ámbito internacional, aplicables a los procesos de extracción, producción, manejo y transporte de hidrocarburos de las instalaciones de PEP, así como a los sistemas de seguridad que permiten elevar la seguridad de las mismas y la protección al medio ambiente.

GOIP, está conformado por especialistas que por experiencia, calidad, alcance y contenido de sus trabajos, cuentan con el reconocimiento del personal de los diferentes activos de la Región Marina Noreste, Región Marina Suroeste y Región Sur de PEMEX Exploración y Producción, además del personal especialista de compañías líderes en el ramo de optimización y automatización de instalaciones de producción.

Misión: "La misión del grupo es coadyuvar a la mejor continua de los procesos productivos de PEMEX Exploración-Producción a través de la investigación, desarrollo y asimilación de tecnologías en optimización, automatización; ofreciendo servicios de consultoría para su implantación".

Para el cumplimiento de la misión del GOIP, y con el objetivo de ofrecer asesorías en las diferentes áreas que integran un proyecto de optimización, GOIP cuenta con cuatro áreas de especialización que trabajan en una organización matricial, en la que personal de las diferentes especialidades participa de forma concurrente con la finalidad de ofrecer trabajos de alto contenido tecnológico y con un gran beneficio para su principal cliente PEMEX (Ver Figura 49).

El abanico de servicios de asesoría especializada que el grupo proporciona en optimización y automatización de instalaciones de producción tanto terrestres como marinas (plataformas), cubre cuatro áreas de trabajo: Instrumentación, Sistemas de Control, Sistemas Instrumentados de Seguridad y Optimización de la operación. Detallándose a las funciones de cada área de especialidad:

El Área de Instrumentación proporciona servicios de asesoría, definición de especificaciones técnicas, asimilación tecnológica; búsqueda y evaluación de nuevas tecnologías en el campo de la Instrumentación.

En particular, cubre áreas de normatividad en instrumentación, protocolos de comunicación a nivel instrumentos, métodos de seguridad aplicados a la instrumentación y mantiene actualizados los conocimientos de instrumentación para sistemas de control, seguridad y los procesos.

El Área de Sistemas de Control proporciona servicios de asesoría, definición de especificaciones técnicas, asimilación tecnológica y búsqueda y evaluación de nuevas tecnologías en equipos de control (PLC, SCADA, etc., y control basado en PC).

En particular, cubre áreas de normatividad, integración de equipos de control, protocolos, redes, tipos de entradas y salidas, bases de datos, software complementario, interfaz humano-máquina, conexión a nivel gerencial y mantiene actualizados los conocimientos en equipos de control.

El Área de Sistemas de Seguridad proporciona servicios de asesoría, definición de especificaciones técnicas, asimilación tecnológica y búsqueda y evaluación de nuevas tecnologías en sistemas de seguridad. En particular, cubre áreas de normatividad en seguridad, integración de equipos de seguridad, gas y fuego, redes contra incendio, interfaz humano-máquina, sistemas de paro de emergencia, protocolos de comunicaciones, conexión a nivel gerencial y mantiene actualizados los conocimientos en materia de seguridad.

El Área de Optimización de Procesos en Línea/Tiempo Real, busca proporcionar servicios de investigación y desarrollo en tecnología aplicada al control automático de procesos, así como la búsqueda y evaluación de nuevas tecnologías en control avanzado aplicables al mejoramiento de la operación de las instalaciones de PEMEX Exploración-Producción.

Ésta área aprovecha los conocimientos y la experiencia de las áreas restantes para lograr la función planteada. En particular cubre aspectos del modelado matemático de los procesos de producción y su validación. Sus propósitos son la adecuación de modelos dinámicos para simulación de algoritmos de control, adaptación de éstos algoritmos de uso corriente, aplicación de técnicas de optimización de procesos (en línea/tiempo real) y además actualización sostenida de los conocimientos del grupo en materia de procesos, control y optimización de los mismos.

Ésta área interactúa con el resto de las áreas en la definición de las áreas de capacitación, de manera que la experiencia de cada área pueda ser aprovechada para el fortalecimiento de las habilidades del GOIP en conjunto.

Grupo de Optimización de Instalaciones de Producción

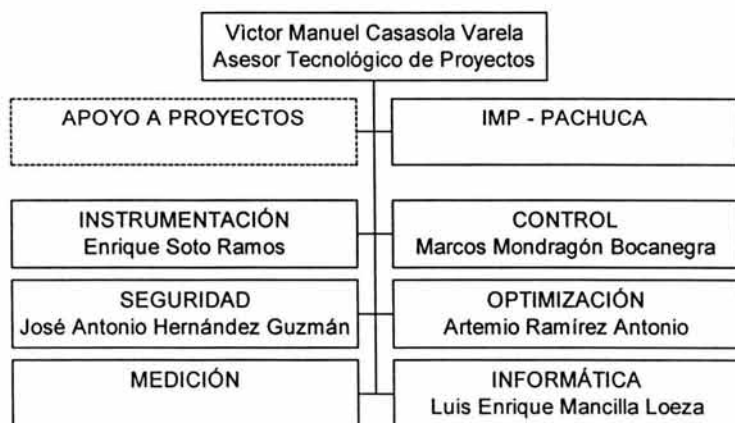


Figura 49. Organigrama del GOIP.

3.2.2. Análisis del Sistema.

El análisis de sistemas consta de tres fases: estudio de la viabilidad del proyecto, estudio y análisis del sistema actual y establecimiento de prioridades entre las necesidades de usuario.

1. Estudio de la viabilidad del proyecto.

En el caso del sistema de manejo de información para la simulación y optimización de procesos de producción de hidrocarburos no se requiere esta evaluación. Ya que este sistema procede de una planificación formal que fue justificada por el equipo de planificación y dirección del GOIP.

Por lo tanto aquí se pasó directamente a la fase de estudio y análisis.

2. Estudio y análisis del sistema actual.

El caso del sistema de manejo de información, es especial, ya que no existía un sistema a estudiar. Esto es debido a que el objetivo del proyecto es crear una aplicación que integre distintas herramientas de simulación en una misma base de datos. En este caso se tuvo que investigar la existencia de sistemas parecidos existentes en el mercado o ya desarrollados por otras empresas.

Se encontraron varios sistemas de información que contenían pozos y sus características, y otros Sistemas de Información de Gas y Aceite. En PEMEX se encontró uno que es el más parecido al sistema que se necesita en donde se integra información desde el Sistema de Control Distribuido (DCS) a una base de datos en tiempo real, pero no hay una integración de aplicaciones. La búsqueda fue realizada en Internet debido a la necesidad de información actualizada. Algunos de estos sistemas son los siguientes:

- Sistema de Información de Pozos (WIS). En Pensilvania existe una base de datos de gas y aceite llamada: Sistema de Información de Pozos (Wells Information System WIS). La base de datos WIS almacena y organiza los detalles asociados con alrededor de 133,000 pozos perforados, así como más de 36,000 no perforados, cancelados o expirados. Esta paquetería de base de datos relacional permite entrada, almacenamiento, acceso y análisis de datos a través de cualquier número de campos de datos relacionados.⁶⁰
- Sistema de Información de Gas y Aceite Colorado. (Colorado Oil and Gas Information System COGIS).⁶¹
- Sistema de Información de Gas y Aceite Integral (Comprehensive Oil and Gas Information System COGIS). Este sistema contiene todo tipo de información acerca de aceite crudo, petróleo y productos de petróleo y gas natural. El sistema fue desarrollado por la Oficina de Administración de Información de Energía del gobierno de los Estados Unidos.⁶²
- El Sistema de Información de Gas y Aceite en Línea (Online Oil and Gas Information System Montana) permite al público usuario investigar acerca de datos de gas y aceite desde las bases de datos. Éste incluye pozos básicos, producción, disposición, inyección e información de inspección. También permite al usuario exportar los resultados de la consulta.⁶³
- Sistema de Control de Información (SampleManager). Este proyecto elaborado para Pemex por AspenTech, está basado en la integración de información desde el Sistema de Control Distribuido (DCS) a una base de datos en tiempo real, PI, la cual tiene la función de almacenar información histórica. PI enviará la información a la base de datos relacional Oracle, la cual almacena y comparte la información de aplicación principal de todos los productos que maneja el Sistema de Información de Planta de Pemex.⁶⁴

⁶⁰ Referencia Sitios de Internet 26. <http://www.dcnr.state.pa.us/topogeo/wis/wis.htm>

⁶¹ Referencia Sitios de Internet 27. <http://oil-gas.state.co.us/>

⁶² Referencia Sitios de Internet 28. <http://www.aldea.com/guides/ag/b400.html>

⁶³ Referencia Sitios de Internet 29. <http://bqgc.dnrc.state.mt.us/ldpIntro.htm>

⁶⁴ Referencia Sitios de Internet 30. http://www.thermo.com/eThermo/CMA/PDFs/Articles/articulosFile_10363.pdf

A pesar de la inexistencia de un sistema de información se realizó un análisis de la situación que existía. En el desarrollo del sistema se utilizaron técnicas como entrevistas, reuniones, investigación en manuales, observación, pruebas con los simuladores, etc. Todas ellas, con el fin de conocer un poco sobre la problemática y el manejo de los distintos simuladores de manera aislada, además de entender un poco conceptos relacionados con la industria petrolera.

Se realizaron dos diagramas para dar una idea de los procesos seguidos a debido a la inexistencia de un sistema de información. Estos se observan en las Figuras 50 y 51.

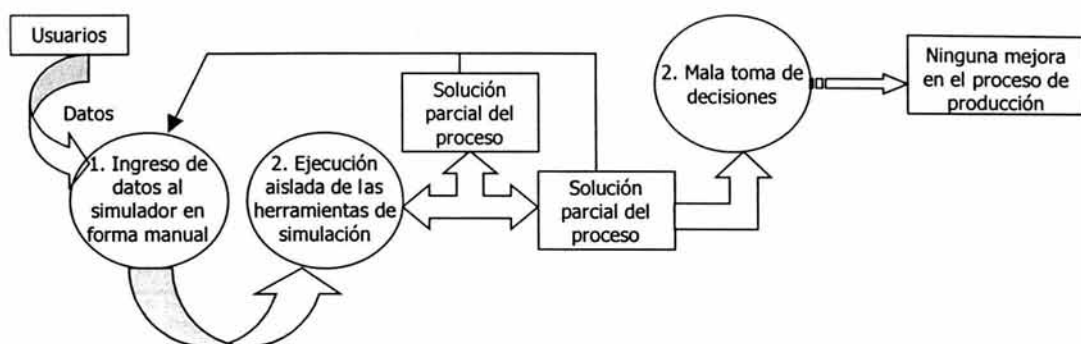


Figura 50. Diagrama de flujo de datos de la problemática en el campo de la optimización de los procesos de producción.

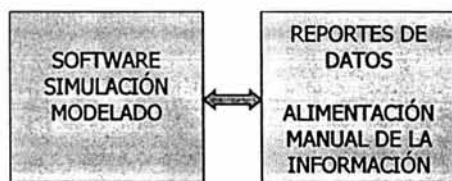


Figura 51. Operación actual del proceso de simulación y modelado (operación fuera de línea).

Se encontró que todas las aplicaciones de simulación que utiliza PEMEX Exploración y Producción se ejecutan en forma aislada, como resultado se obtienen soluciones parciales. Se utilizan diversas herramientas de simulación e ingreso de datos en forma manual.

No existe un estándar de comunicación de datos entre herramientas de simulación que permita relacionar todas las aplicaciones y conciliación de datos a través de una plataforma o interfaz estándar.

Después de entender la situación actual se formularon los objetivos, los cuales se mencionaron al inicio del trabajo. Estos objetivos plantean lo que se espera alcanzar al final del desarrollo del sistema de información.

3. Establecimiento de prioridades entre las necesidades del usuario.

Se requiere de la creación de un Sistema de Información que almacene en una misma base de datos información de proceso y de los simuladores, y que la mismo tiempo interactúe con los mismos, para así evitar desperdicio de tiempo y recursos.

Es altamente factible y necesario la ejecución de este proyecto, en virtud de las diferentes aplicaciones de simulación utilizados por PEP en sus diversos grupos de trabajo y de la inexistencia de una normatividad que permita ó facilite la manipulación de información de manera transparente desde cualquiera de las aplicaciones. Así mismo permitirá a los niveles gerenciales la toma de decisiones en forma oportuna sobre los procesos de producción.

3.3. Configuración de hardware y software a utilizar.

3.3.1. Hardware.

Características del Hardware:

- Estación de trabajo LAPTOP Dell Precision.
- Dell Computer Corporation
- Dell Precision M50
- Mobile Intel® Pentium®
- Procesador: Pentium 4 – M CPU 2.00 GHz
- AT/AT COMPATIBLE
- 1,047,992 de RAM
- Memoria RAM: Memoria RAM de 1 GB DDR, tipo SDRAM (2x512MB)
- Velocidad del bus frontal: 400 MHZ
- Unidad de disco flexible: Unidad de disco flexible de 3.5" de capacidad 1.44 MB integrado.
- Unidad de CD-ROM: Unidad combo DVD/CD-RW de 8X para DVD, 24X escritura de CD, 8X para reescritura de CD y 24X de lectura de CD.
- Capacidad de disco duro: 60 GB ultra ATA 100
- Gráficos: Controlador de video integrado de 64 MB y soporte 2D/3D para resolución de 1024x768 a 16 millones de colores.
- Pantalla: 15 Pulgadas TFT tipo LCD, UXGA matriz activa
- Puertos: Con los siguientes puertos de comunicaciones:
 - 2 puertos seriales DB-9
 - 1 puerto minidin para mouse teclado
 - 2 puertos USB
 - 1 puerto IEEE 1394
 - 1 puerto s-video infrarrojo
 - 1 puerto para estación de expansión o replicador de puertos, entrada y salida de audio.
- Audio: Controlador de audio compatible con sound blaster.
- Ranuras totales: 2 ranuras PCMCIA tipo II o I tipo III
- Controlador de red: Puerto Ethernet de 10/100 base TX integrado a la tarjeta madre con características Autosense, conector RJ-45 wake on lan y PXE.
- Fax modem: 56 KBPS V.92 integrado a la tarjeta madre.
- Teclado: 88 teclas idioma español latinoamericano
- Mouse: Óptico de 2 botones con scroll coneCtor USB.
- Fuente de poder: Adaptador de corriente de AC de 90 Watts.
- Batería: De Ion Litio.

Características del Sistema:

- Fabricante del Sistema: DELL Computer Corporation
- Modelo del Sistema: Precision M50
- Tipo de Sistema: Equipo basado en X86
- Procesador: X86 Family 15 Model 2 Stepping 7 GenuineIntel 1996 Mhz.
- Versión del BIOS: Phoenix ROM BIOS PLUS Versión 1.10 A12
- Cantidad total de memoria física: 1,047,992 KB
- Memoria virtual total: 3,568,480 KB

3.3.2. Software.

3.3.2.1. Microsoft Windows 2000 Professional.

Microsoft (R) Windows.
Versión 5.0 (Build 2195: Service Pack 2)
Copyright (C) 1981-1999 Microsoft Corp.

3.3.2.2. Microsoft Access 2000.

Microsoft Access 2000 (9.0.2720)
Copyright (C) 1981-1999 Microsoft Corp.

3.3.2.3. Microsoft Visual Basic para Aplicaciones 6.0 (VBA).⁶⁵

Hace más de una década, Bill Gates propuso un lenguaje universal para macros para aplicaciones de escritorio. Microsoft Visual Basic para aplicaciones (VBA= *Visual Basic for Applications*) es la realización de ese sueño y más aún. El VBA de Microsoft Access 2002 es común a todos los componentes de Microsoft Office y a multitud de paquetes de software de otras marcas. Su sintaxis también es consistente con el lenguaje de programación independiente Visual Basic. La ubicuidad de VBA permite a los desarrolladores utilizar un único lenguaje de programación en multitud de contextos, simplemente aprendiendo un nuevo modelo de objeto para un contexto determinado. VBA es como el pegamento para las aplicaciones de Access. VBA lo une todo y le da forma a la aplicación.

Microsoft proporciona Visual Basic para Aplicaciones (VBA) en Access 2002 como una arma poderosa de desarrollo para automatizar una base de datos. Microsoft incorporó VBA en este producto para conseguir que Access 2002 fuera un sistema poderoso y versátil de gestión de bases de datos para los usuarios programadores en la actualidad.

Access 2002 conserva el editor de Visual Basic (VBE= *Visual Basic Editor*) que Microsoft ofreció por primera vez a los desarrolladores de Access con Access 2000. Este entorno integrado de desarrollo relaciona la interfaz de programación de Access con otros componentes de Office y con la versión independiente de Visual Basic.

Los programadores necesitan el poder adicional y la capacidad de abordar los errores que VBA proporciona. VBA permite que los programadores controlen la interfaz del usuario y manipulen los acontecimientos para crear una solución de base de datos que sea funcional, efectiva y fácil de usar.

3.3.2.4. PIPESIM (Modelos de Sistemas de Producción de Gas y Aceite Integrados).

Requerimientos mínimos de Hardware y Sistema:

- PC IBM Compatible con un procesador de 200 MHz o mayor.
- Disco Duro.
- Al menos 100 MB o espacio libre en disco duro.
- Un drive CD-ROM.
- Pantalla VGA.
- Un mouse.
- 16 MB en RAM.

⁶⁵ Referencia 9. Dobson, Rick. España, 2002, 997 pp.

- Microsoft Windows 95 o mayor.
- El Sistema de la PC debe ser puesto en la fecha actual. La seguridad del sistema utiliza la fecha actual e la computadora.

3.3.2.5. PRO/II (Simulación de Procesos de Refinería).

Requerimientos mínimos de Hardware y Sistema:

PC/LAN: Intel Pentium 120Mhz o CPU compatible.

- 32 Mb de RAM.
- 75 Mb espacio en disco.
- Monitor y tarjeta de gráficos SVGA.
- Drive CD-Rom
- Windows 95/98/NT
- Mouse compatible

3.3.2.6. Lenguaje de Consulta Estructurado (SQL).

La palabra SQL está formada con las iniciales de las inglesas *Structured Query Language*, o en español, Lenguaje de Consulta Estructurado. El nombre no refleja todas las posibilidades del lenguaje, pues éste no sólo permite consultas sino también actualizaciones de datos y otras operaciones.⁶⁶

El Lenguaje de Consulta Estructurado SQL (*Structured Query Language*), es un estándar informático corrientemente utilizado para definir, modificar, y gestionar datos y controlar cómo se realizan cambios en la base de datos usando tablas, índices, claves, filas y columnas para almacenar la información. SQL fue desarrollado a partir de las ideas del Dr. Edgar F. Codd, de IBM, que colaboró en el desarrollo del modelo relacional cuando trabajaba en los laboratorios de investigación de IBM en los años 70.

SQL permite a los usuarios acceder a los datos en sistemas de administración de bases de datos, como son Oracle, Sybase, Informix, Microsoft SQL Server, Access, y otros, permitiendo describir los datos que el usuario desee ver. SQL también permite a los usuarios definir los datos en una base de datos, y manipular los mismos. El SQL es un lenguaje que permite expresar operaciones diversas, por ejemplo aritméticas, combinatorias y lógicas.

En los últimos años han ido apareciendo en el mercado múltiples productos de bases de datos basados en el SQL, tanto para micro y minicomputadoras como para grandes sistemas, haciendo de este lenguaje un vehículo común ampliamente utilizado en entornos muy diversos de máquinas y sistemas operativos. Esta tendencia hacia su uso generalizado en la industria se ha visto consolidada al ser adoptado por el Instituto Americano de Normas (ANSI, *American National Standards Institute*), que ha desarrollado y publicado unas especificaciones para este lenguaje, aceptadas también posteriormente (1987) por la organización ISO (*International Standards Organization*).

Una de las características más importantes del SQL frente a los lenguajes tradicionales no relacionales de bases de datos es que sus sentencias permiten manejar conjuntos de registros, en vez de un solo registro cada vez. Otra es que, al tener una firme base teórica, posee una gran capacidad expresiva aunque su estructura es muy simple. Todo ello dota al lenguaje de una gran potencia que permite expresar con una sola sentencia consultas complejas, que tradicionalmente podrían requerir uno o más programas para su formulación, lo que redundaría en una alta productividad en la codificación y prueba de programas.

Por otra parte, esta flexibilidad y potencia expresiva unidas a la capacidad de usarlo interactivamente para realizar consultas no planificadas, es decir, no previamente incluidas en programas, abre a los usuarios finales la posibilidad de acceder directamente a los datos.

⁶⁶ Referencia 4. Benavides Abajo, J.; Olaizola Bartolomé, J. M.; Rivero Cornello, E. España, 1998, 383 pp.

Las peticiones sobre los datos se expresan en SQL mediante sentencias, que deben escribirse de acuerdo con las reglas sintácticas y semánticas de este lenguaje. Estas instrucciones pueden utilizarse de distintas maneras:

- Estas sentencias pueden escribirse directamente en la pantalla de una terminal interactiva, en la cual también se recibe el resultado de la petición expresada en ellas.⁶⁸
- También pueden escribirse las sentencias de SQL embebidas en programas, incorporándose así su capacidad expresiva a la lógica y funciones de éstos. Naturalmente el autor de los programas deberá conocer, además del SQL, el lenguaje de programación correspondiente, que puede ser cualquiera de los más conocidos, como Visual Basic, C++, Delphi, etc. Esta forma de uso requiere por tanto conocimientos informáticos que suelen estar más al alcance de programadores profesionales que de usuarios finales.

3.4. Modelos Conceptuales de Datos (Diseño Lógico).

La realización de un modelo es necesario para organizar los datos a fin de atender a las necesidades de cada proceso; posteriormente se intenta que respondan a los requisitos de un conjunto de procesos y, por último, se busca una interpretación de la realidad con el objetivo de conseguir captar la semántica del mundo real.

Un modelo de datos es una representación conceptual de la estructura de los datos que es requerida por una base de datos. Las estructuras de los datos incluyen objetos de datos, las asociaciones entre los objetos datos y las reglas que gobiernan las operaciones entre los objetos. Como su nombre lo dice, el modelo de datos se enfoca en los datos que son requeridos y cómo éstos deben estar organizados además de qué operaciones serán realizadas con los datos. Para entender mejor, el modelo de datos es equivalente a unos planos de construcción de un arquitecto.

Un modelo de datos es independiente del hardware o software, y trata de representar los datos como una base de datos los vería, el modelo de datos se enfoca en representar los datos, como ya se mencionó, de la manera en que el usuario los ve en el "mundo real". Esto sirve como un puente entre los conceptos de los eventos y procesos de la realidad y la representación física de éstos en una base de datos.

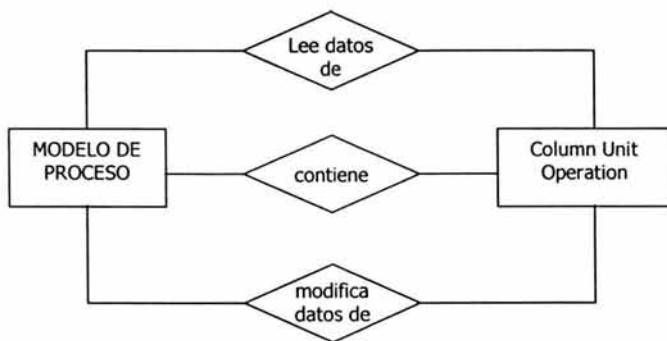
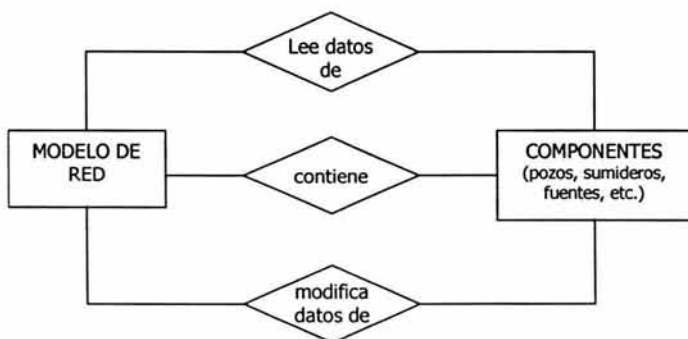
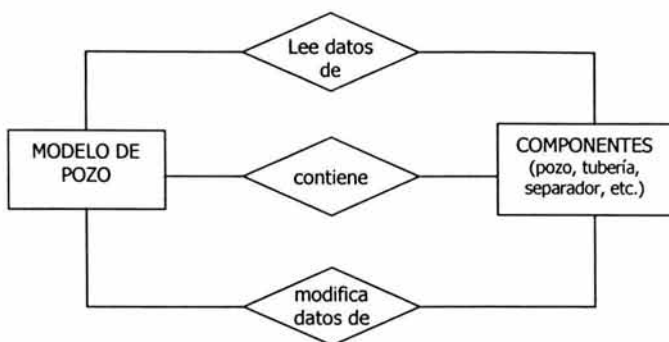
El diseño lógico traduce los modelos conceptuales en el modelo de datos de un SGBD. Éste diseño adopta una vista lógica de los datos que describe los elementos de datos y relaciones de la base de datos.

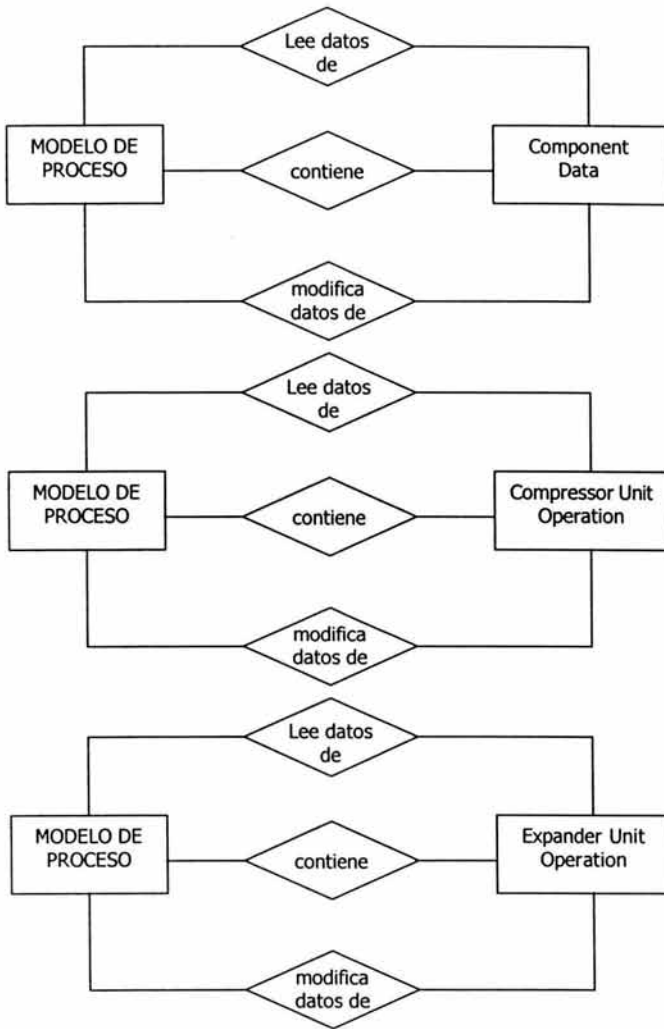
Para el Sistema de Información para la Simulación y Optimización de Procesos de Producción de Hidrocarburos se realizaron dos modelos de datos, estos son: modelo entidad-relación y modelo relacional.

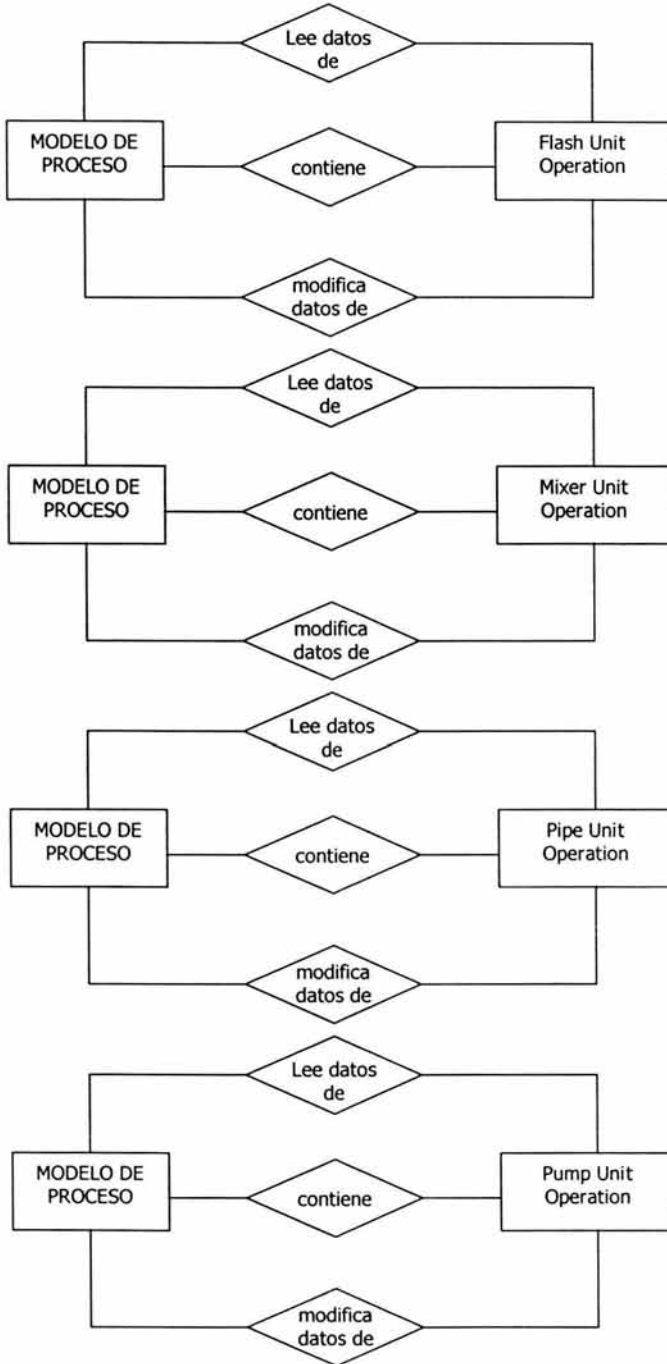
El Modelo Entidad-Relación es modelo de red que describe con un alto nivel de abstracción la distribución de datos almacenados en un sistema. Este modelo de datos se utiliza para examinar las relaciones de datos complejas independientemente del proceso que se llevará a cabo. Además este modelo es el que se muestra a los usuarios ejecutivos, quienes podrían no estar interesados en los detalles funcionales cotidianos del sistema.

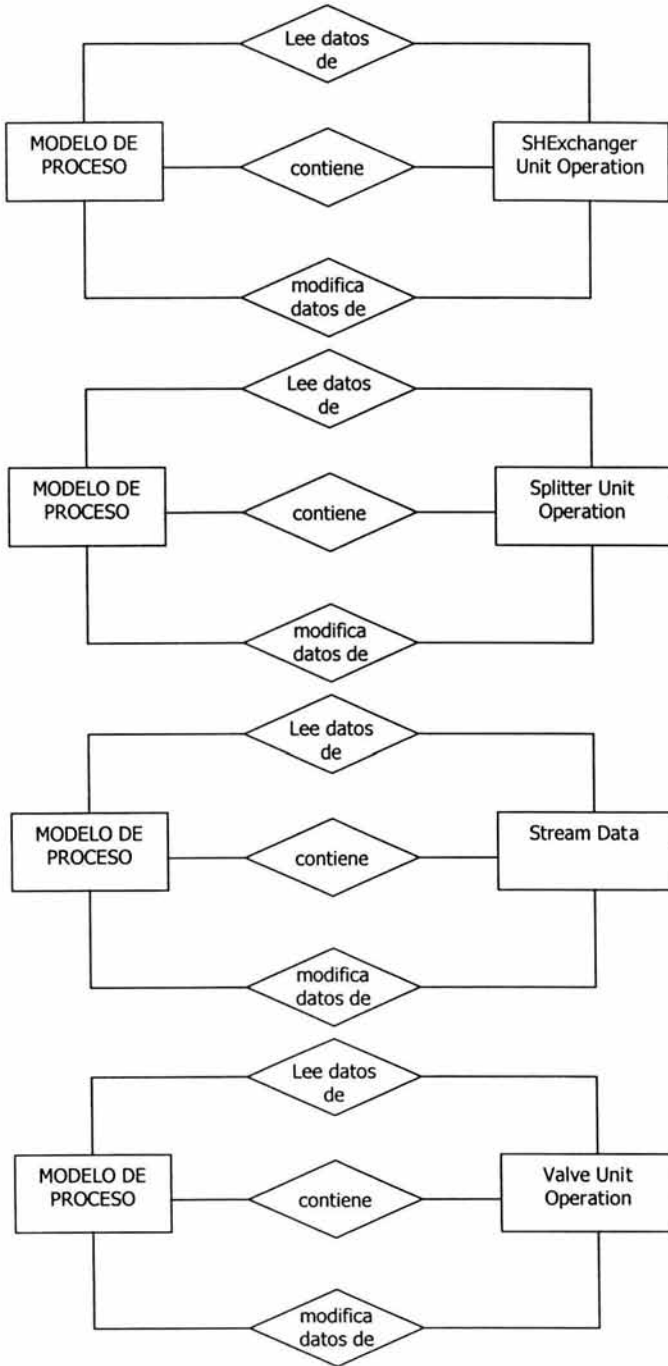
A continuación se muestran los diagramas entidad-relación del Sistema de Información llevado a cabo en este trabajo de tesis.

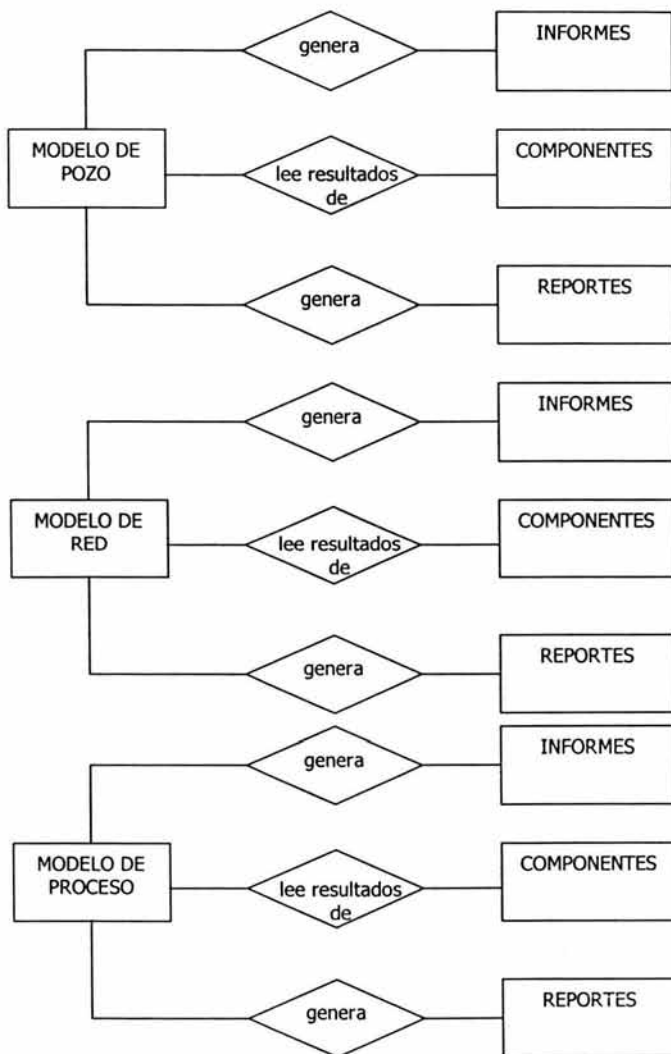
⁶⁸ Referencia 4. Benavides Abajo, J.; Olaizola Bartolomé, J. M.; Rivero Cornelio, E. España, 1998, 383 pp.











En el Modelo de Datos Relacional del Sistema de Información de Procesos de Producción, todas las entidades están representadas como tablas separadas y no están colocadas en ninguna jerarquía fija como es el caso de los árboles (Ver Figura 52) °.

Tramo	Gasto	DI	Longitud	Presion_Entrada	Presion_Salida
Rebombeo - Bos Bocas	5000	24	5.6	25.3	18.2

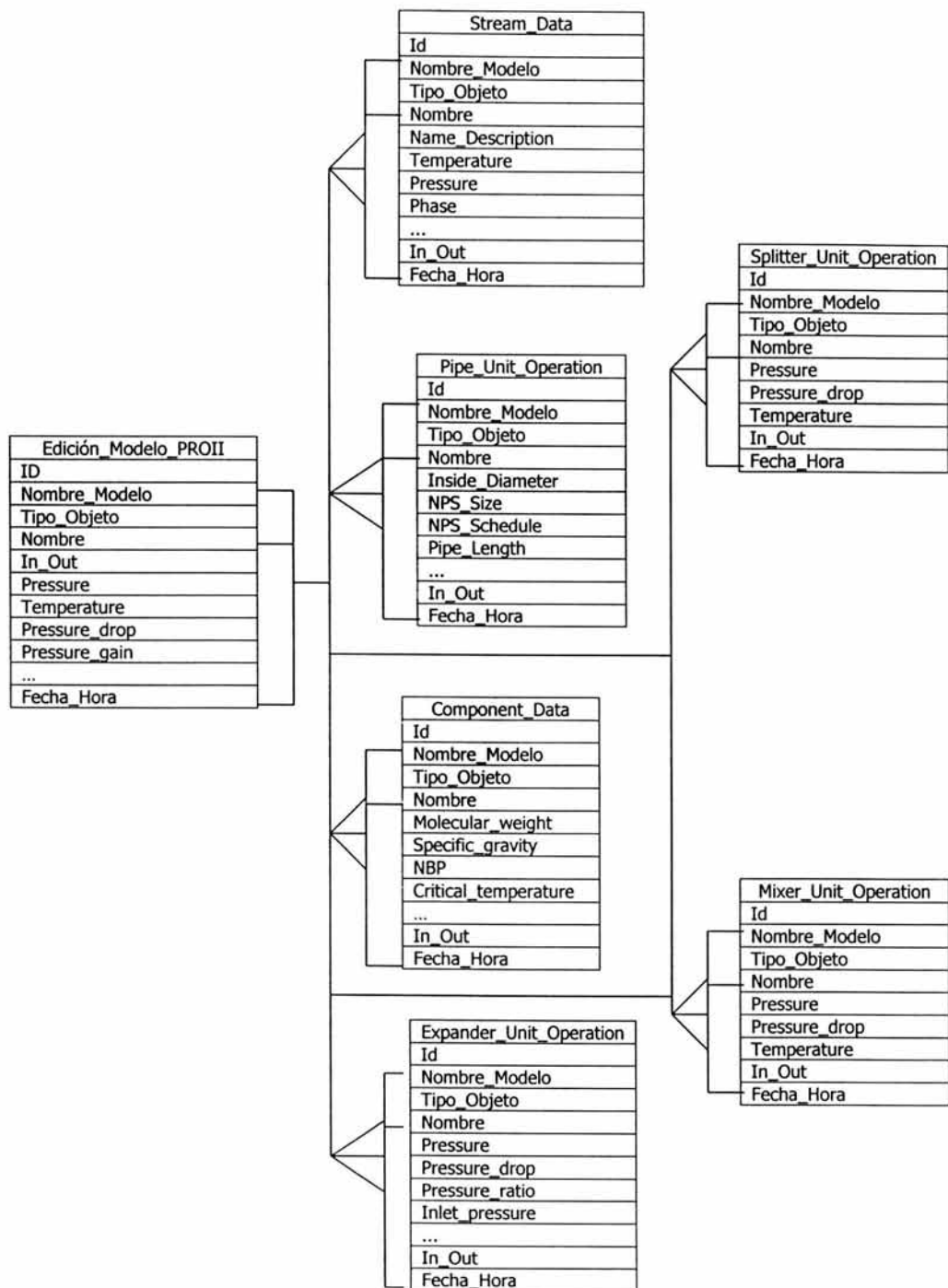
Figura 52. Tabla: Caracteristicas_de_Ductos.

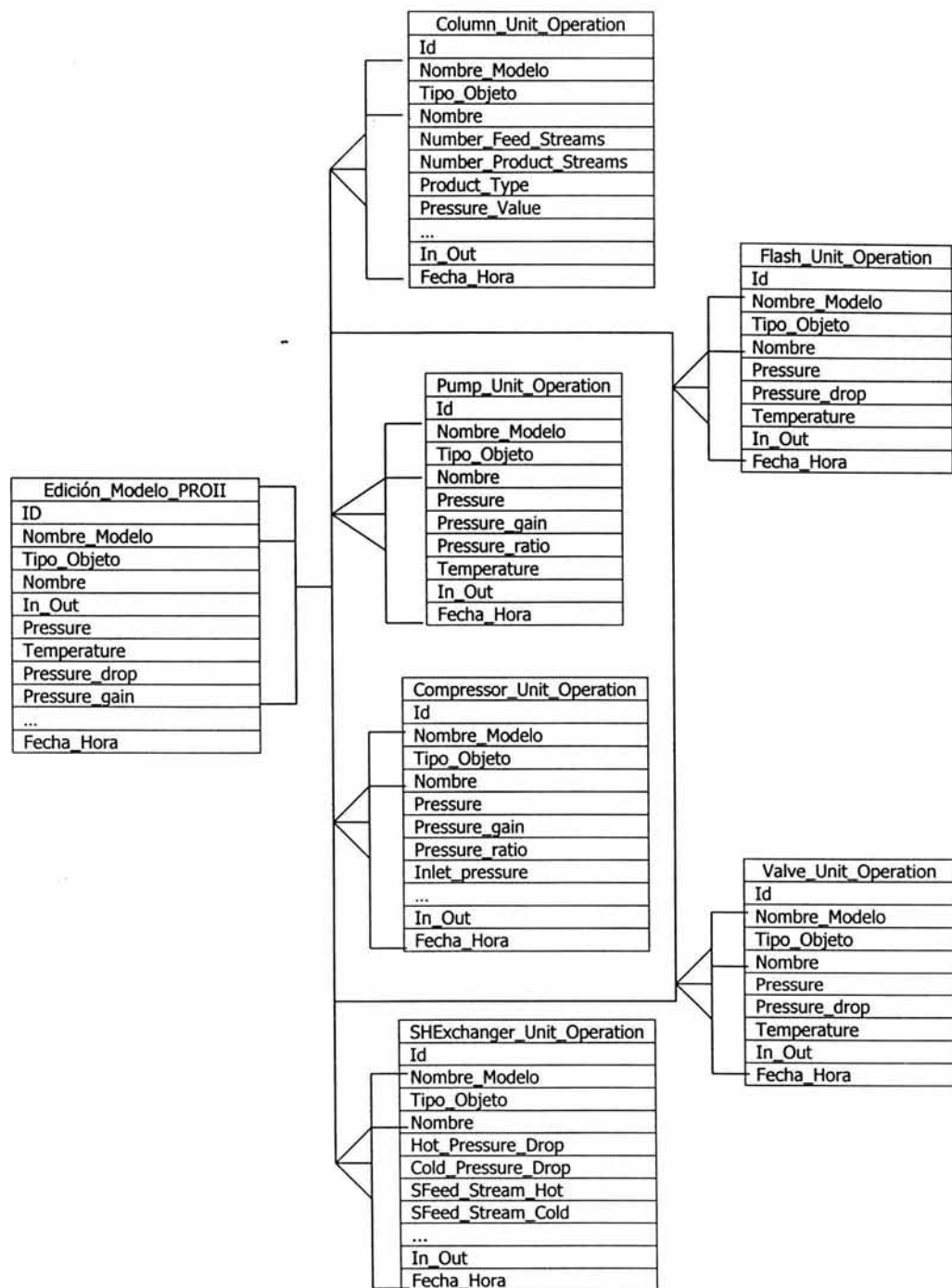
° Ver ANEXO A.

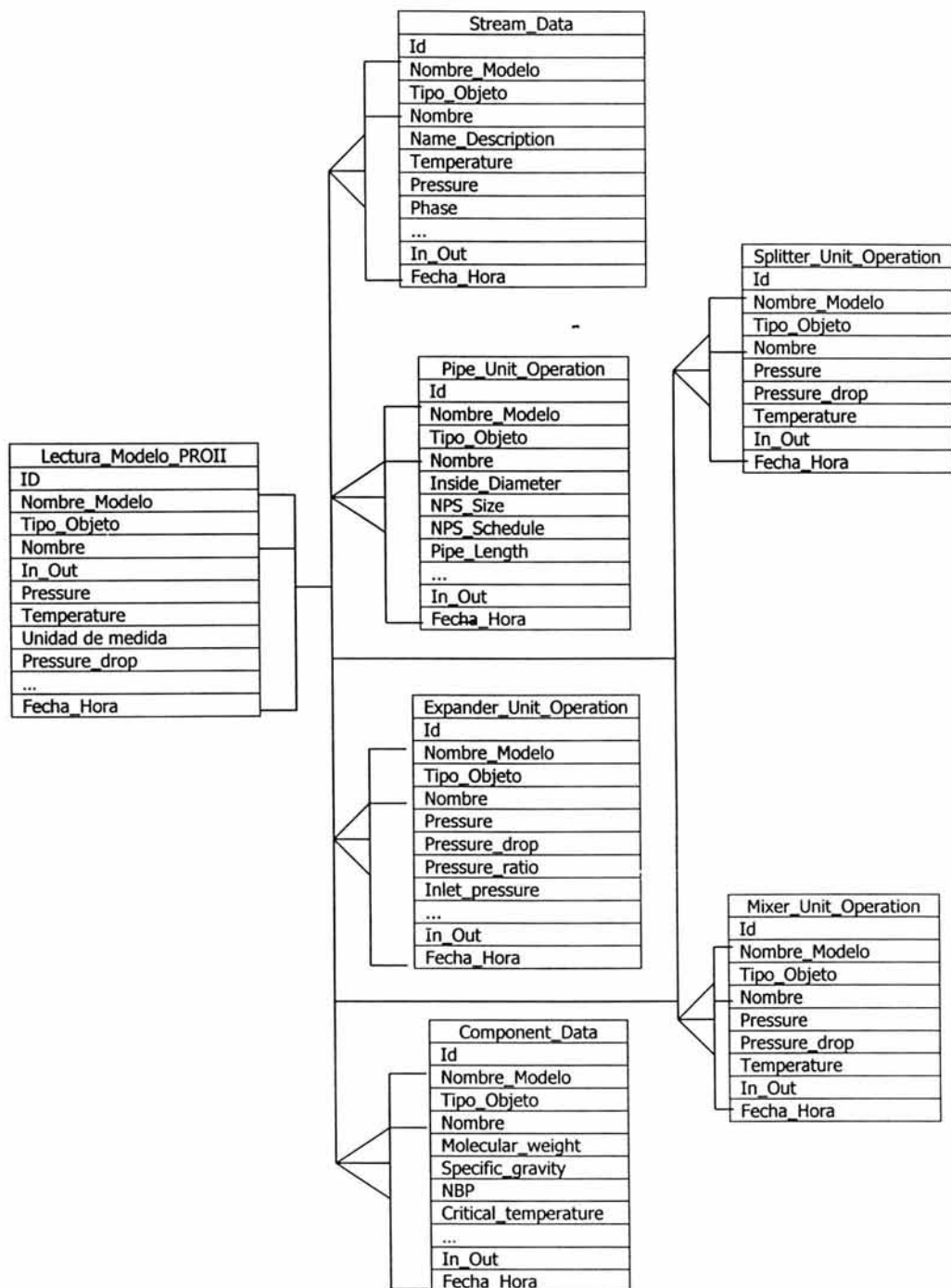
Se observan los componentes del modelo relacional como son:

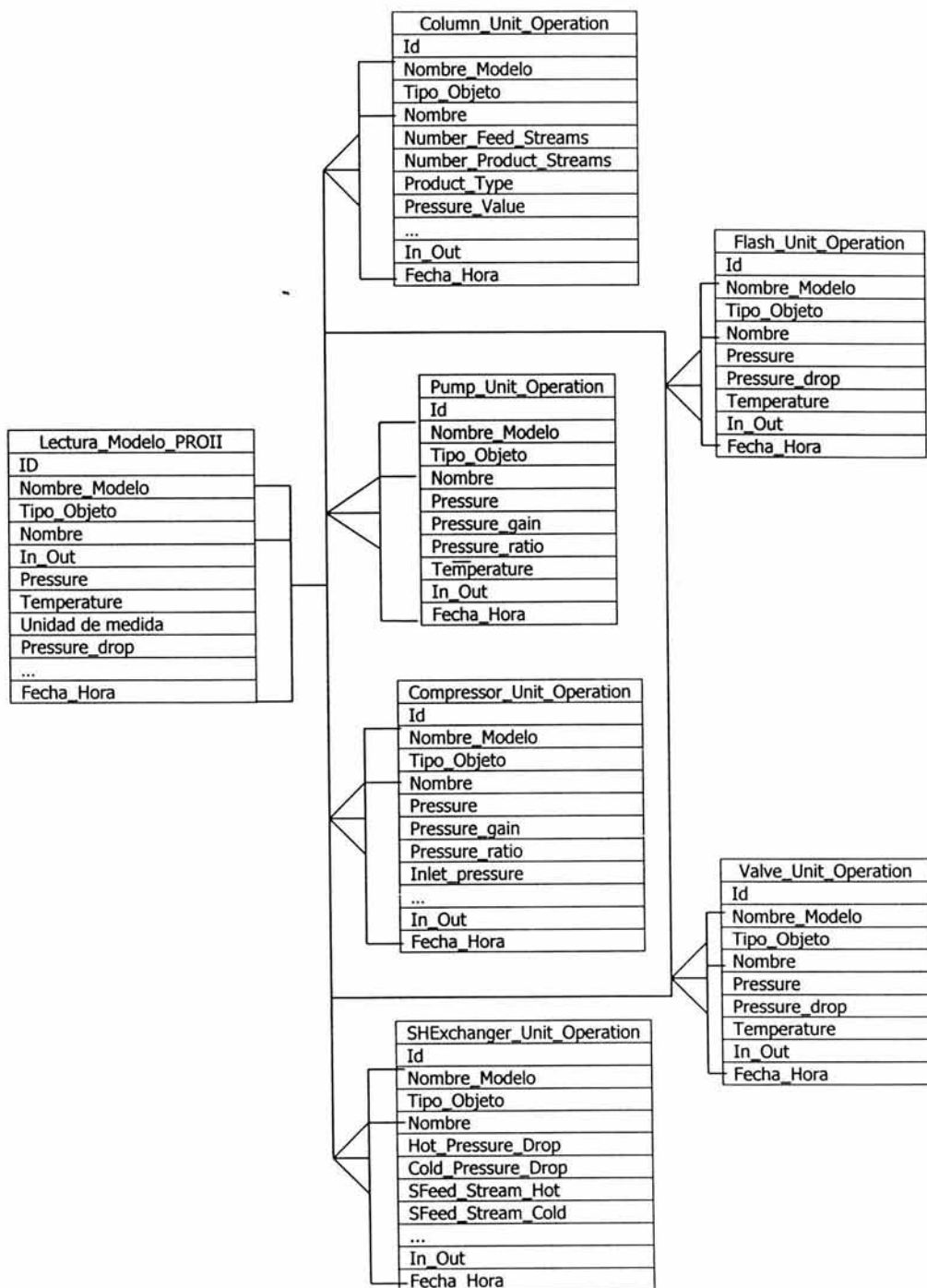
- a) Relación: una tabla también se llama **relación**. Una relación es un archivo conceptual que consta de ocurrencias con la misma composición de campo. Generalmente, una relación se implanta como un archivo almacenado.
- b) Atributo: cada columna en una relación representa un **atributo**, el cual es equivalente a un campo. Una entidad está descrita por uno o más atributos.
- c) Dominio: el **dominio** de un atributo es el conjunto de todos los valores posibles de entre los cuales el atributo puede tomar su valor.
- d) Tupla: los renglones de la tabla se llaman **tuplas**. Cada una representa la ocurrencia de un registro. Una relación es un conjunto de tuplas. Si una relación tiene columnas, se dice que es de grado n y consta de un conjunto de n -tuplas. Una relación binaria tiene grado 2 y una relación n tiene grado n .
- e) Clave principal y clave aspirante: un campo de datos se llama aspirante si el contenido del campo identifica de manera única cada tupla de la relación. Una relación puede tener más de una clave aspirante; en este caso una de éstas se puede designar como clave principal. Una clave principal puede constar de una combinación de más de un atributo si es que ninguno de estos identifica de manera única a cada registro en la relación.
- f) Clave externa: en terminología relacional, un campo de conexión se llama "clave externa" porque es un campo "externo" a la relación principal. Una clave externa en el caso de un registro hijo, contiene el valor clave-principal de su registro padre. Por lo tanto, el registro hijo identifica a su padre vía la clave externa.

A continuación se muestran las relaciones de las tablas del Sistema de Información de Procesos de Producción de Hidrocarburos.









Edición_Modelo_Pozo_Pipesim2000
Id
nom_Modelo
nom_Objeto
tipo_Objeto
Tipo_Operación
Pressure(psia)
Temperature(C)
Fecha_Hora
Clave principal: Id

Reporte_Modelo_Pozo_Pipesim2000
Numero
nom_Modelo
Nombre_Nodo
Tipo_Nodo
Pressure
Temperature
LiquidRate
GasRate
MassRate
LGR
WaterCut
PressureResidual
RateResidual
LGR
MaxPres
CompPower
Choke_DP
Clave principal: Numero

Edición_Modelo_Red_Pipesim2000
Id
Nom_Modelo
Nom_Objeto
tipo_Objeto
Pressure(psia)
Temperature(C)
LiquidRate(STB/D)
GasRate(mmscf/d)
MassRate(lb/s)
Fecha_Hora
Clave principal: Id

Reporte_Modelo_Red_Pipesim2000
Numero
nom_Modelo
Nombre_Nodo
Tipo_Nodo
Pressure
Temperature
LiquidRate
GasRate
MassRate
LGR
WaterCut
PressureResidual
LGR
MaxPres
CompPower
Choke_DP
Clave principal: Numero

3.5. Diseño de la Base de Datos (Diseño Físico).

El diseño físico determina las estructuras de almacenamiento de datos y los métodos de acceso. Éste diseño adopta una vista física de los datos (que también recibe el nombre de vista interna) que describe la manera como los datos se organizarán, almacenarán físicamente y se accederán en los dispositivos de almacenamiento del sistema.

Para el diseño de la base de datos se siguieron los siguientes pasos:

1. Determinar el propósito de la base de datos.

Se realizaron reuniones con los usuarios o la gente que utilizará la base de datos para determinar la función de la base de datos y la forma de su utilización. Se determinó que la base de datos debía proporcionar datos de las redes de los pozos, de los pozos de manera individual, características de ductos, especificaciones de tuberías, características y variables de los componentes de los procesos (separador, compresor, tubería, intercambiador de calor, etc.), entre otras.

También se necesita que la base de datos almacene datos de simulaciones, tanto de entradas, como de salidas. Para esto hay que determinar los métodos y librerías a utilizar para realizar la simulación desde la aplicación, sin necesidad de abrir el software de simulación.

De la determinación de los datos a proporcionar se fueron obteniendo los campos a utilizar en cada una de las tablas. Un punto importante aquí es que se necesitan datos históricos, por lo tanto se incluirá en la mayoría de las tablas un campo dedicado a la fecha y hora del suceso.

También se obtuvo información sobre los informes y reportes que se deben producir, que en este caso, son casi de todas las tablas, ya sea de manera conjunta de los registros o de manera individual. El informe se obtendría de las entradas de los datos o del proceso y los reportes de las salidas, después de realizada una simulación.

2. Determinar las tablas que se necesitan.

En el caso del Sistema de Información de Procesos de Producción, las tablas se van diseñando de acuerdo con la información obtenida por parte de los usuarios. Primeramente se organiza la información, para determinar de qué tipo es y en qué parte de la aplicación se requiere. Después se determinan los campos, tratando de que la información no esté duplicada en una misma tabla ni entre tablas. Además se requiere que la información de una sola entidad esté contenida en una sola tabla. ▼

3. Determinar los campos que se necesitan (en las tablas).

La mayoría de las tablas de la base de datos son fáciles de diseñar, ya que los datos a contener son diferentes de las demás, tanto de la información de proceso, como de la de las entradas y salidas de las herramientas de simulación.

En cada tabla se crearon campos pertenecientes a dicha entidad y en algunos casos, es bastante la información que se necesita obtener que quedan algunas tablas un poco grandes. ▼

4. Identificar los campos con valores únicos en cada registro.

En todas las tablas se establece un campo dedicado a identificador o a número, éste campo se refiere en algunas tablas ya sea a la clasificación que se hace de las entidades o a un número continuo, pero en todas está incluido para identificar únicamente a un registro individual en cada tabla.

* Ver ANEXO B.

* Ver ANEXO B.

5. Determinar las relaciones entre las tablas.

Esta parte ya fue mencionada en la parte de: Modelo de Datos Relacional, dentro del desarrollo de este trabajo.

6. Refinar el diseño.

Al estudiar el diseño realizado, se encontró que casi no había campos duplicados, ya que la información suele ser única, debido al campo de Fecha-Hora, ya que es muy raro que se repita.

7. Introducir los datos y crear otros objetos de bases de datos.

Los datos de los datos de proceso se introdujeron de manera manual, y por medios electrónicos, como se menciona en la propuesta de solución, en el capítulo anterior.

Para la introducción de datos de las simulaciones se tuvo un poco de más trabajo, ya que se tenía que comunicar primeramente al software de simulación a la base, para de ahí introducir los datos por medio de formularios y posteriormente correr la simulación. Las salidas se guardan nuevamente. *

En esta fase es donde se empezaron a crear otros objetos de base de datos como formularios, módulos, informes, entre otros.

* Ver puntos 3.6 y 3.7 para la descripción detallada de la integración de datos al sistema de información.

3.6. Integración de información de proceso a la Base de Datos.

La información de proceso, se dividió en fracciones más pequeñas, es decir, en datos, para introducirla a los campos de las tablas diseñadas.

Se diseñaron tres tablas que son las que guardan los datos de proceso, éstas tablas son:

1. Características_de_Ductos. Esta tabla incluye datos sobre el tramo recorrido por el ducto, el gasto, la longitud del ducto, la presión de entrada y la presión de salida.

Los primeros datos se capturaron directamente en la tabla en la Ventana Hoja de Datos de Microsoft Access, esto se puede ver en la Figura 53:

Tramo	Gasto	DI	Longitud	Presion_Entrada	Presion_Salida
Rebombeo - Bos Bocas	5000	24	5.6	25.3	18.2

Figura 53. Tabla: Características_de_Ductos

Y se creó un formulario de entrada de datos para capturar los datos subsecuentes de una manera más fácil. El formulario se muestra en la Figura 54:

Microsoft Access - [FR_CARACTERÍSTICAS DE DUCTOS] (TABLA)

INSTITUTO MEXICANO DEL PETRÓLEO
DIRECCIÓN EJECUTIVA DE EXPLORACIÓN Y PRODUCCIÓN
LABORATORIO DE INVESTIGACIÓN APLICADA EN INSTRUMENTACIÓN Y CONTROL
SISTEMA DE INFORMACIÓN PARA LA SIMULACIÓN Y OPTIMIZACIÓN DE PROCESOS DE PRODUCCIÓN DE HIDROCARBUROS

Tramo:

Gasto: Bariles por día

DI: Pulgadas

Longitud: Metros

Presion_Entrada: PSIG

Presion_Salida: PSIG

Registros: 1

Figura 54. Formulario: FR_Características_de_Ductos

2. Variables_de_Pozo. Esta tabla incluye datos de las características del pozo, como campo y plataforma, nombre del pozo, gastos, presiones, temperaturas, diámetros, fecha, y tres campos en donde se guarda como objeto OLE, el estado mecánico, el plano de localización y la sección estructural de dicho pozo.

Aquí también se capturaron los datos como se muestra en la Figura 55:

Id	Campo	Plataforma	Pozo	Fecha	Gasto_Aceite	Gasto_Gas_Inyeccion	Rga
1	Cantarell	Akal-O	C-16	31/05/1997	3507	2	70.2
2	Cantarell	Akal-O	C-16	13/07/1997	3408	2.5	133.2
3	Cantarell	Akal-O	C-16	22/05/1999	2588	2.56	185.85

Figura 55. Tabla: Variables_de_Pozo

Y se creó el formulario para la captura de los datos, como se ve en la Figura 56:

Figura 56. Formulario: FR_Variables_de_Pozo

3. Especificaciones_de_Tubería. Esta tabla incluye datos sobre la tubería como diámetro interno, tipo de ducto, si flota o no en el agua, espesor del ducto, peso, área de flujo y presión.

La tabla de las características de tubería se encuentra en la Figura 57:

DI	Tipo	F	Espesor_Pulgadas	Peso_Lbs	Peso_Ton	Area_Flujo	Presion_Permissible
2.5		Sí	0.065	1.604	4.234	0.02749	1379

Figura 57. Tabla: Especificaciones_de_Ttuberia

Así como el formulario de entrada de datos se puede observar en la Figura 58:

INSTITUTO MEXICANO DEL PETRÓLEO DIRECCIÓN EJECUTIVA DE EXPLORACIÓN Y PRODUCCIÓN LABORATORIO DE INVESTIGACIÓN APLICADA EN INSTRUMENTACIÓN Y CONTROL SISTEMA DE INFORMACIÓN PARA LA SIMULACIÓN Y OPTIMIZACIÓN DE PROCESOS DE PRODUCCIÓN DE HIDROCARBUROS	
DI	DI
Tipo	Tipo
Flot en Agua	<input checked="" type="checkbox"/>
Espesor Pulgadas	Espesor Pulga
Peso Lbs	Peso Lbs
Peso Ton	Peso Ton
Area de Flujo	Area Flujo
Presión Permiso	Presion Permis

Figura 58. Formulario: FR_Especificaciones_de_Tuberia

3.7. Integración de la información requerida y generada por los simuladores de proceso.

Para la realización de la integración de datos requeridos y generados por los simuladores de procesos, se desarrolló el Sistema de Información, de manera completa, es decir, el diseño de interfaces de usuario y el diseño de programas, junto con la codificación del mismo.

Se llevaron a cabo las alternativas propuestas en el Capítulo 2, para la integración de los datos al Sistema de Información.

El diseño de pantallas fue uno sólo para todas las que componen el Sistema de Información y se realizó dividiendo las mismas en zonas, ya que de esta manera es más fácil leer la información y los mensajes que se presentan durante la ejecución de la aplicación y el usuario aprende a utilizarla con mayor rapidez. La Figura 59 muestra el formato de las pantallas del Sistema de información, el cual se presenta para que se le facilite al usuario la ubicación de los componentes de las mismas:

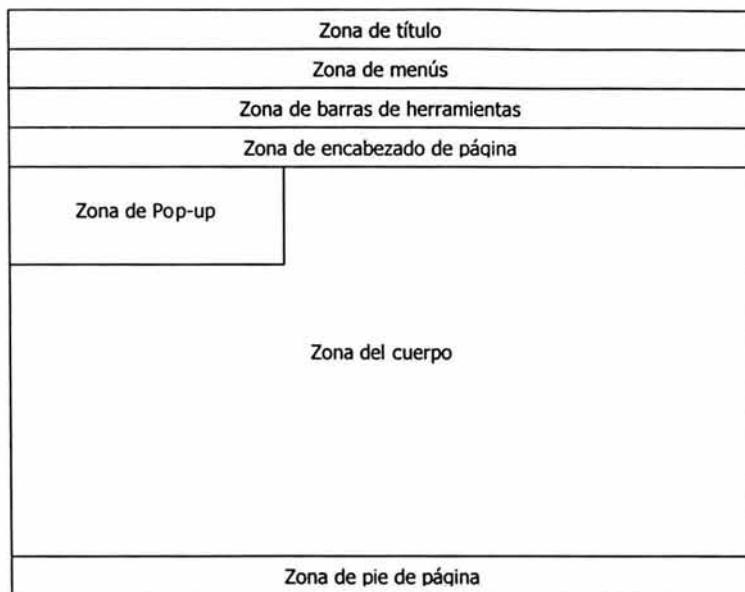


Figura 59. Formato de las pantallas del Sistema de Información.

Definiciones de zona:

- Zona de título. La zona de títulos identifica la pantalla desde el punto de vista del usuario. Opcionalmente, puede identificar el archivo o registro actualmente en uso. En un entorno de ventanas, como el del Sistema de Información realizado en este trabajo, la zona de título por lo general es una barra con nombre situada en la parte superior de la ventana.
- Zona de menús. La zona de menús identifica el área de pantalla donde pueden encontrarse las opciones de menú. En un entorno de ventanas, el menú superior se sitúa en una barra inmediatamente debajo de la zona de título. Los submenús se observan como menús desplegados cuando se selecciona una opción en la barra superior.
- Zona de barras de herramientas. Las barras de herramientas contienen imágenes o iconos que suministran un medio más rápido para ejecutar secuencias de órdenes de menús. Las barras de herramientas se sitúan debajo de la zona de menús.
- Zona de encabezado de página. La zona de encabezado de página contiene información importante acerca del Sistema y la Institución en que fue desarrollado, así como los escudos de las instituciones que hicieron posible la realización del mismo.
- Zona de pop-up. La zona de Pop-up es normalmente invisible. Se trata de un panel que aparece de repente como respuesta a una selección de menús, una solicitud de ayuda ampliada o un caso de error. En los entornos con ventanas, las zonas de pop-up suelen recibir el nombre de cuadros de diálogo.
- Zona del cuerpo. La zona del cuerpo es, por lo general, la de mayor tamaño en la pantalla o en la ventana. Es en esta zona donde el usuario introduce los datos de empresa o recibe la documentación de empresa. En el caso de este Sistema de Información, es donde se visualizan y modifican los datos de proceso y de los simuladores.
- Zona de pie de página. La zona de pie de página contiene datos de la elaboración del Sistema de Información.

Diagramas de transición de estados.

Se realizaron diagramas de transición de estados de las ventanas contenidas en la aplicación. El objetivo de este tipo de diagramas es describir la secuencia y las variantes de las pantallas que pueden sucederse cuando el usuario del sistema se sienta ante la computadora. Puede verse como un mapa de carreteras. Cada pantalla sería semejante a una ciudad. No todas las carreteras pasan por todas las ciudades. En las siguientes figuras se observan estos diagramas. Los rectángulos representan pantallas de visualización, las flechas indican el orden en que se producen dichas pantallas. Se dibujó un rótulo cerca de cada flecha, el cual indica las acciones que desencadenan los flujos de control hacia y desde una pantalla dada, estas acciones son diferentes.

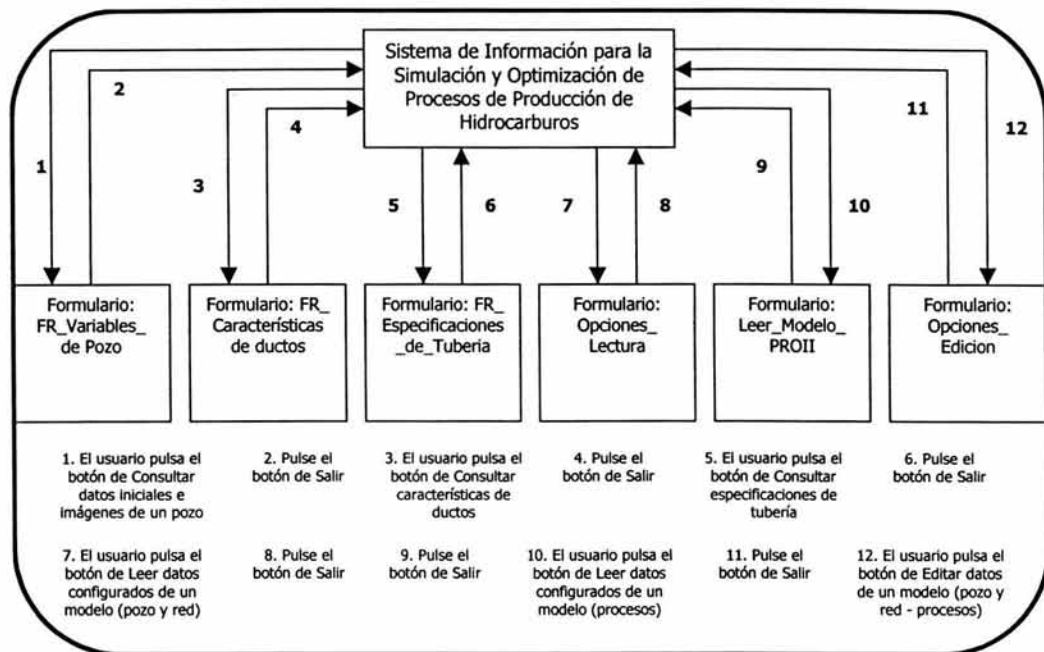


Figura 60. Diagrama de transición de estados para el Sistema de Información para la Simulación y Optimización de Procesos de Producción de Hidrocarburos.

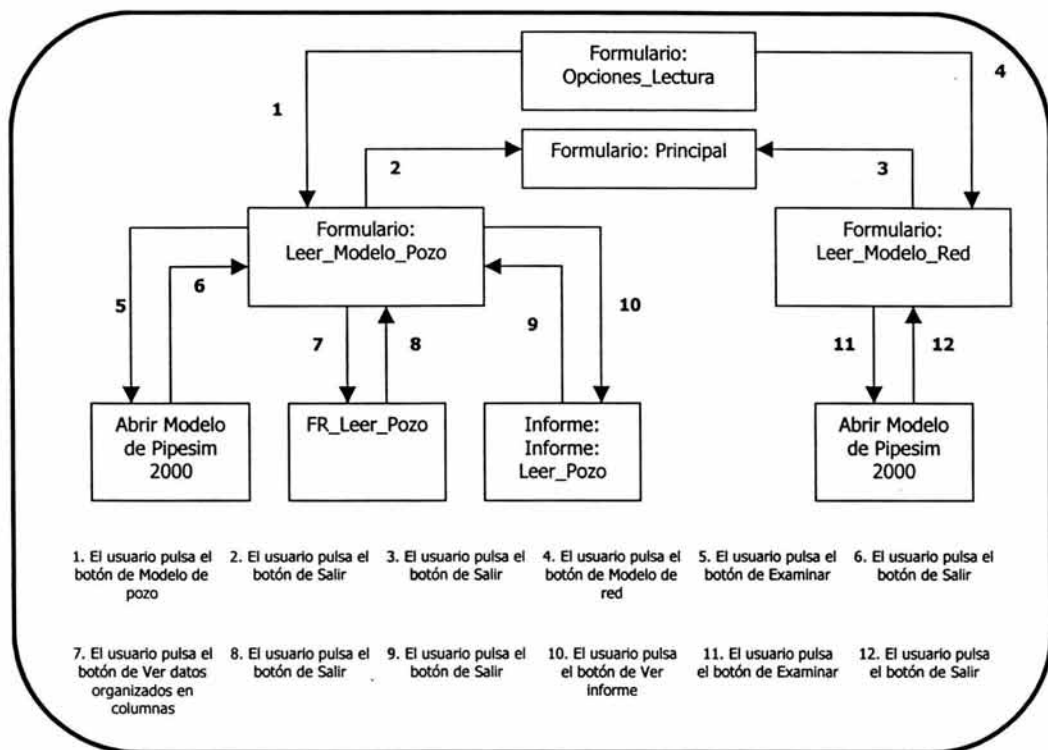


Figura 61. Diagrama de transición de estados para la Ventana de Opciones_Lectura.

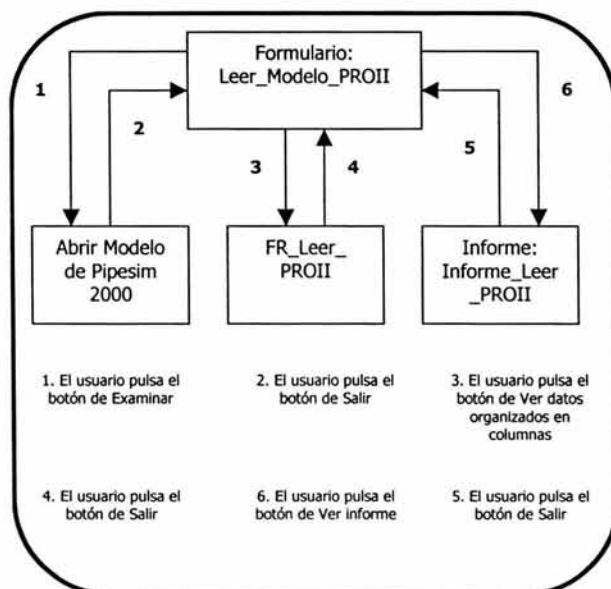
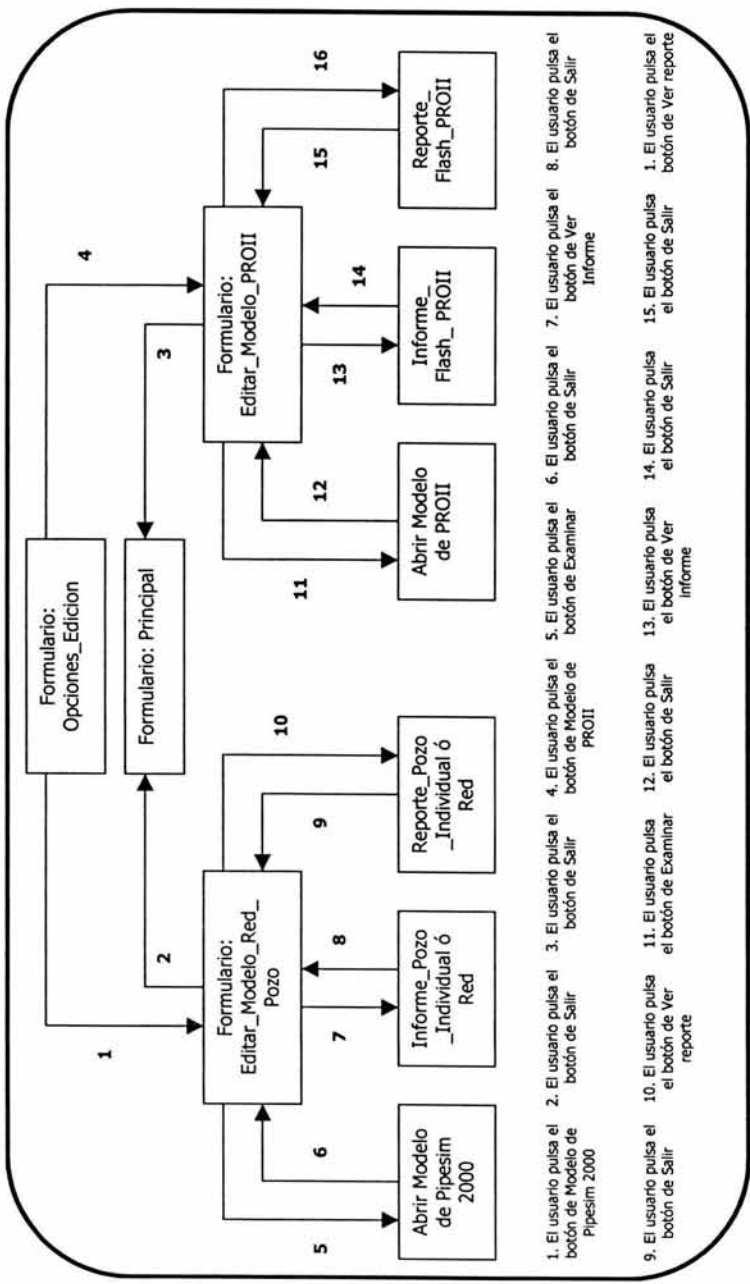


Figura 62. Diagrama de transición de estados para la Ventana de Leer_Modelo_PROII.



- 1. El usuario pulsa el botón de Salir Pipesim 2000
- 2. El usuario pulsa el botón de Salir Pipesim 2000
- 3. El usuario pulsa el botón de Salir Pipesim 2000
- 4. El usuario pulsa el botón de Salir Pipesim 2000
- 5. El usuario pulsa el botón de Salir Pipesim 2000
- 6. El usuario pulsa el botón de Salir Pipesim 2000
- 7. El usuario pulsa el botón de Salir Pipesim 2000
- 8. El usuario pulsa el botón de Salir Pipesim 2000
- 9. El usuario pulsa el botón de Salir Pipesim 2000
- 10. El usuario pulsa el botón de Salir Pipesim 2000
- 11. El usuario pulsa el botón de Salir Pipesim 2000
- 12. El usuario pulsa el botón de Salir Pipesim 2000
- 13. El usuario pulsa el botón de Salir Pipesim 2000
- 14. El usuario pulsa el botón de Salir Pipesim 2000
- 15. El usuario pulsa el botón de Salir Pipesim 2000
- 16. El usuario pulsa el botón de Salir Pipesim 2000

Figura 63. Diagrama de transición de estados para la Ventana de Opciones_Edicion.

Falta página

N° 135

A continuación se muestran los diagramas de los programas realizados para el mejor entendimiento del usuario, acerca de la descomposición de los programas en módulos (subprogramas o subrutinas). Se debe recordar que un módulo es cualquier programa parte de la aplicación, que contiene un grupo de instrucciones ejecutables con un único punto de entrada y un único punto de salida.

Los diagramas siguientes están hechos de forma jerárquica, es decir, a partir del módulo principal y hacia abajo, dependiendo de su ejecución. Un módulo se representa por un cuadro.

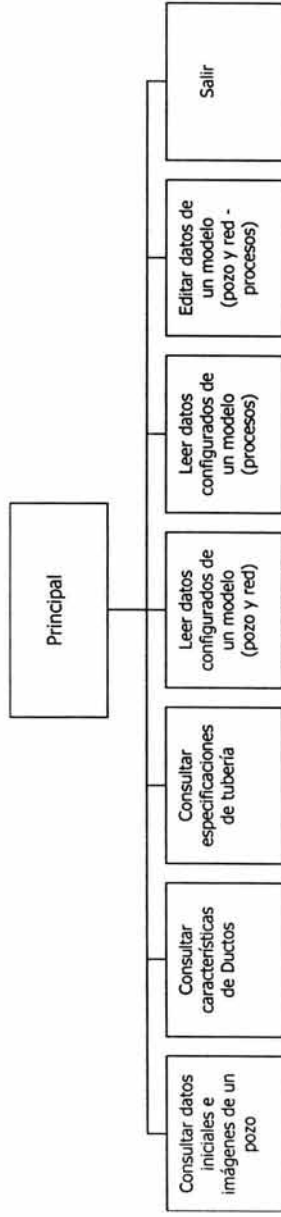


Figura 64. Módulo Principal.

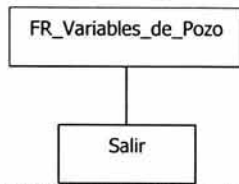


Figura 65. Módulo: FR_Variables_de_Pozo.



Figura 66. Módulo: FR_Caracteristicas_de_Ductos.



Figura 67. Módulo: FR_Especificaciones_de_Tuberia.



Figura 68. Módulo: Leer datos configurados de un modelo (pozo y red).

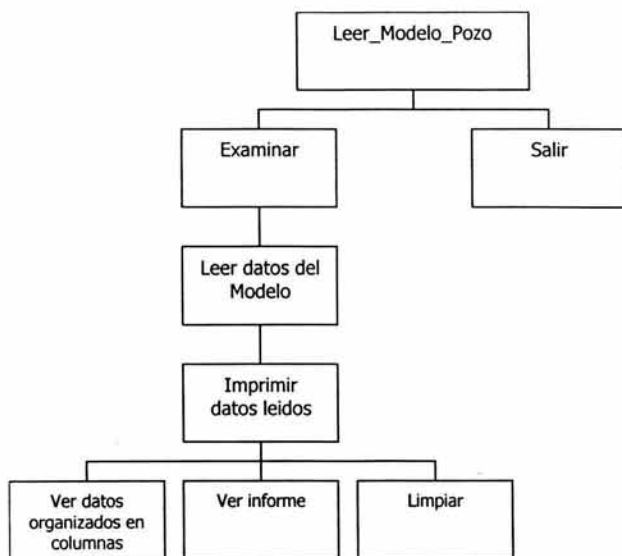


Figura 69. Módulo: Leer_Modelo_Pozo

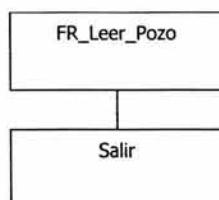


Figura 70. Módulo: FR_Leer_Pozo.

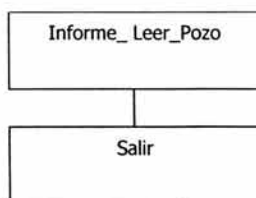


Figura 71. Módulo: Informe_Leer_Pozo.

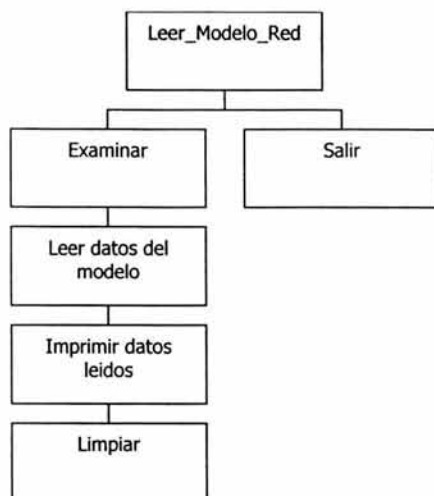


Figura 72. Módulo: Leer_Modelo_Red

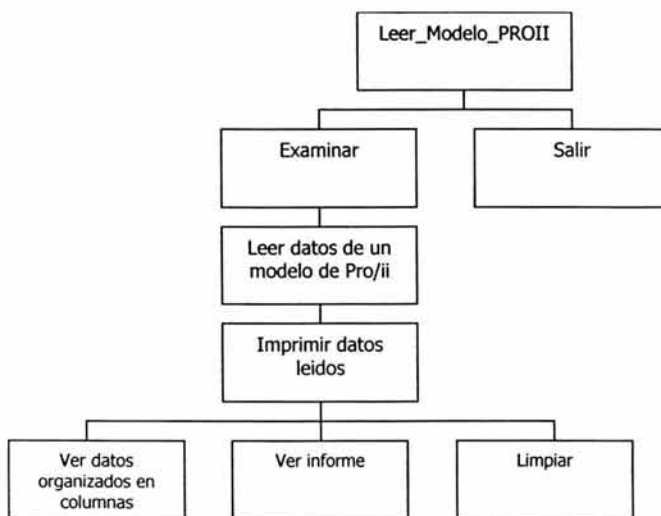


Figura 73. Módulo: Leer_Modelo_PROII.

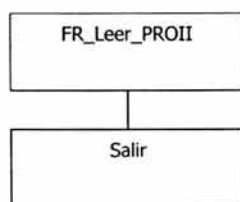


Figura 74. Módulo: FR_Leer_PROII.

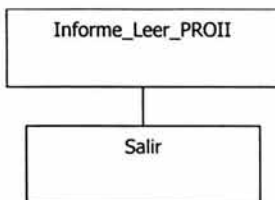


Figura 75. Módulo: Informe_Leer_PROII.

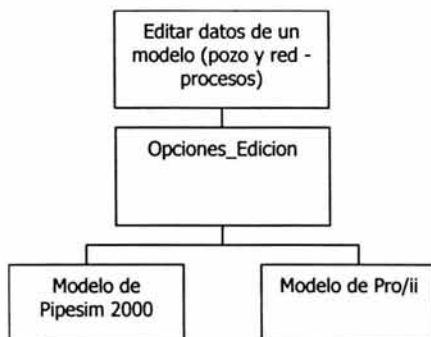


Figura 76. Módulo: Editar datos de un modelo (pozo y red - procesos).

Para poder realizar el proceso de integración de los datos generados por los simuladores de proceso, se requiere que se ejecute la simulación del modelo, de acuerdo a la herramienta de simulación correspondiente, para así obtener los datos de salida o los resultados de la misma. En el Anexo C se pueden observar las pantallas que muestran ejemplos de cada una de las herramientas (Pipesim y Pro/ii) con modelos abiertos.

Para hacer más eficiente la realización de la simulación, uno de los requerimientos del usuario es que la aplicación realice la ejecución de las simulaciones sin necesidad de abrir el software. Para la realización de esto, se investigaron los respectivos métodos en las librerías tanto de Pipesim como de Pro/ii y se analizó la forma de utilizarlos. A continuación se mencionan dichos métodos.

Pipesim 2000.

En la biblioteca Net32COM 1.0 Type Library (Net32COM.DLL), que se agregó en las Referencias del Editor de Visual Basic, se incluyen las interfaces para correr los modelos de pozo y de red, además de las que se mencionaron en el Capítulo 2.

Las interfaces soportadas por esta librería son InetModel (para modelos y operaciones de red) y ISingleBranchModel (para modelos y operaciones de una sola rama).

Los métodos agregados en la aplicación para la ejecución de los modelos son los que se utilizan para realizar Operaciones, y de acuerdo a la interfaz, son:

- ISingleBranchModel Interface
 - o RunSingleBranchModel. Ejecuta el modelo de pozo actualmente abierto llamando al programa motor de cálculo de Pipesim 2000.
 - o GetIsModelRunning. Devuelve el valor verdadero (-1) si el proceso de simulación está activo.

- INetModel Interface

- o Run Network. Ejecuta el modelo de red actualmente abierto llamando al programa motor de cálculo de Pipesim 2000.
- o GetIsModelRunning. Devuelve el valor verdadero (-1) si el proceso de simulación está activo.

Se debe recordar el poner la llave del software antes de realizar la simulación (Ver figura 77).



Figura 77. Licencia de PIPESIM

Pro/ii.

En la biblioteca Pro/ii Class Library Versión 5.61., encontramos el método de ejecución de modelos en el objeto servidor.

- Objeto Servidor. Este objeto debe ser creado e inicializado antes de que cualquier base de datos sea creada o abierta. El método agregado para la corrida de los modelos es:

- RunCalcs. Ejecuta los cálculos de la simulación de Pro/ii y regresa el estado de la solución (realizada la simulación o no). Para llamar a este método, se deben de seguir los siguientes tres pasos en secuencia:
 - El método CheckData debe ser ejecutado en la base de datos, en la cual se van a realizar los cálculos.
 - La base de datos debe ser cerrada, configurando el objeto archivo a 'Nothing'.
 - El servidor de acceso a la base de datos debe estar cerrado, llamando al método 'Unload'.

En la Figura 78 se muestra la llave de Pro/ii que se tiene que conectar.



Figura 78. Licencia de PRO/II.

Estos métodos se utilizaron en módulos independientes, dependiendo del software, para realizar la simulación de los modelos correspondientes a cada uno de ellos desde una ventana del Sistema de Información para la Simulación y Optimización de Procesos de Producción de Hidrocarburos. Esto es parte de la comunicación realizada entre las herramientas de simulación y la base de datos.

Los siguientes diagramas muestran los módulos creados para la edición y ejecución de los modelos.

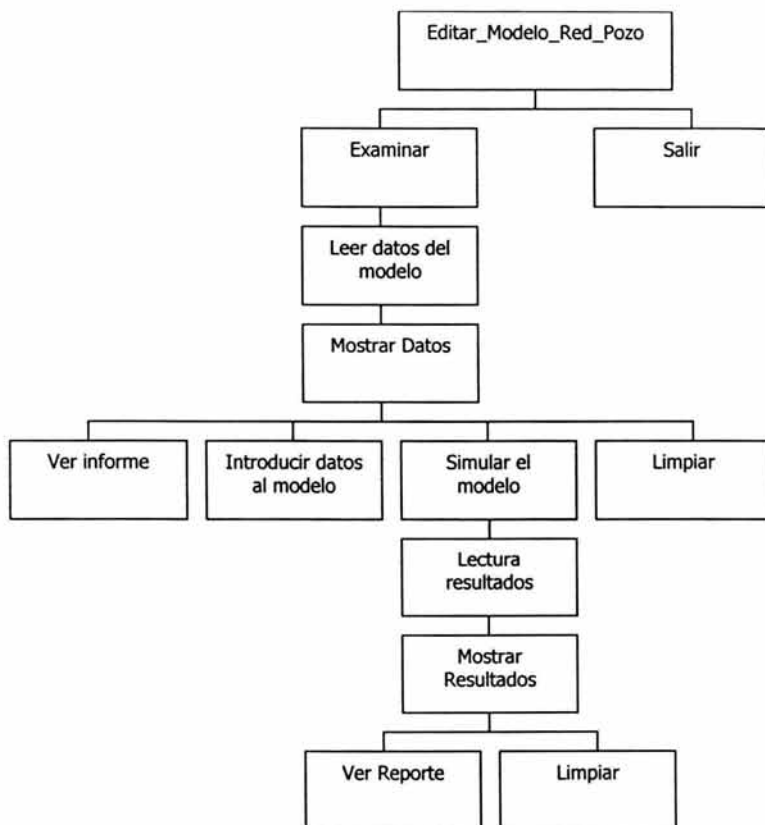


Figura 79. Módulo: Editar_Modelo_Red_Pozo.

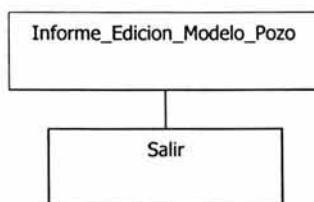


Figura 80. Módulo: Informe_Edicion_Modelo_Pozo.

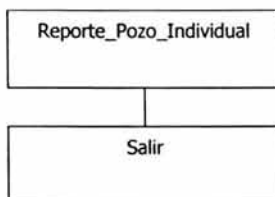


Figura 81. Módulo: Reporte_Pozo_Individual

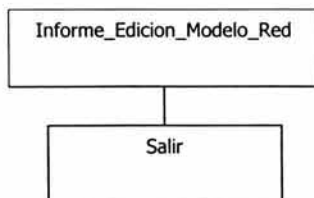


Figura 82. Módulo: Informe_Edicion_Modelo_Red.

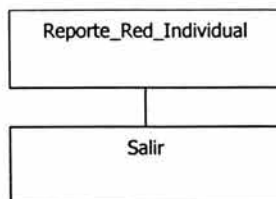


Figura 83. Módulo: Reporte_Red_Individual.

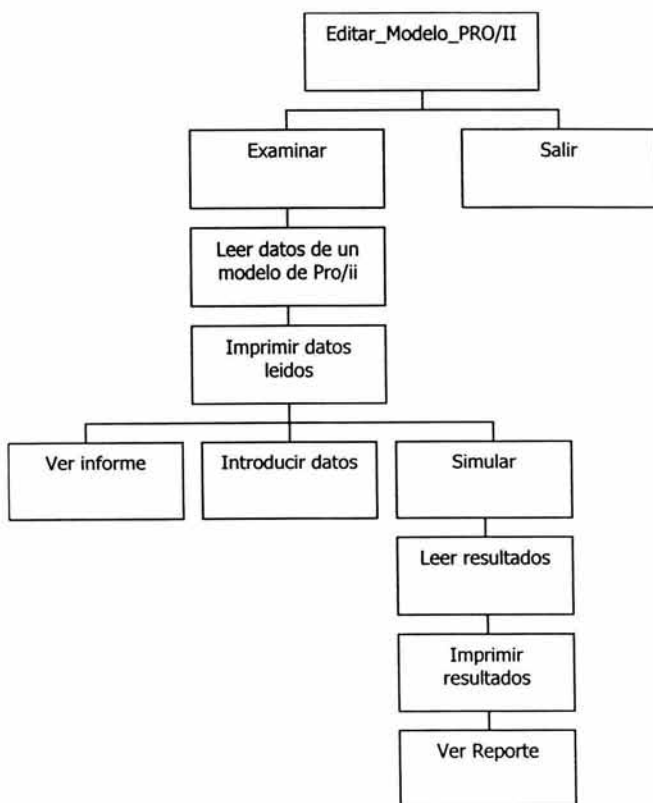


Figura 84. Módulo: Editar_Modelo_PRO/II.

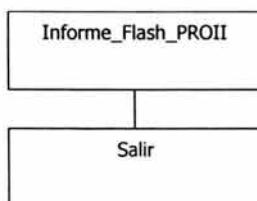


Figura 85. Módulo: Informe_Flash_PROII

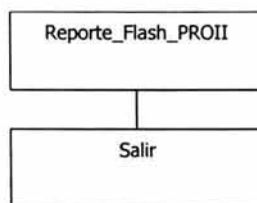


Figura 86. Módulo: Reporte_Flash_PROII.

Para comprender mejor la secuencia de las pantallas y la función de las mismas, a continuación se muestran y describen las principales. *

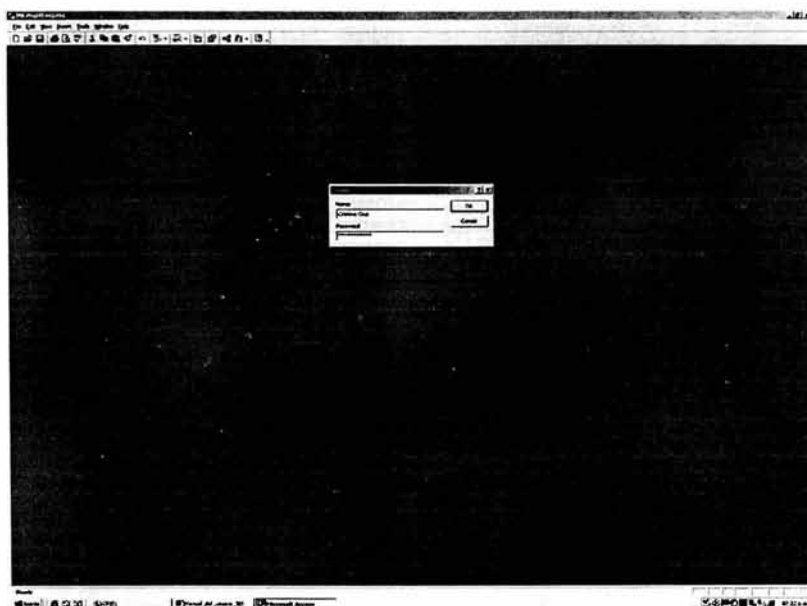


Figura 87. Pantalla de seguridad.

* Para más información se puede consultar el Manual del Usuario en la opción de ayuda del Sistema de Información para la Simulación y Optimización de Procesos de Producción de Hidrocarburos.

Los tipos de usuario determinarán el tipo de acceso y derechos que se le concedan a un usuario dentro del sistema. Se consideran dos tipos de usuario:

- Administradores. Tendrán acceso a todo el sistema. Serán responsables de administrar, leer el diseño, modificar el diseño, leer datos, actualizar datos, insertar datos, eliminar datos.
- Usuarios. Tendrá derecho a abrir y ejecutar los módulos y sólo podrá acceder a la información para consultar datos.

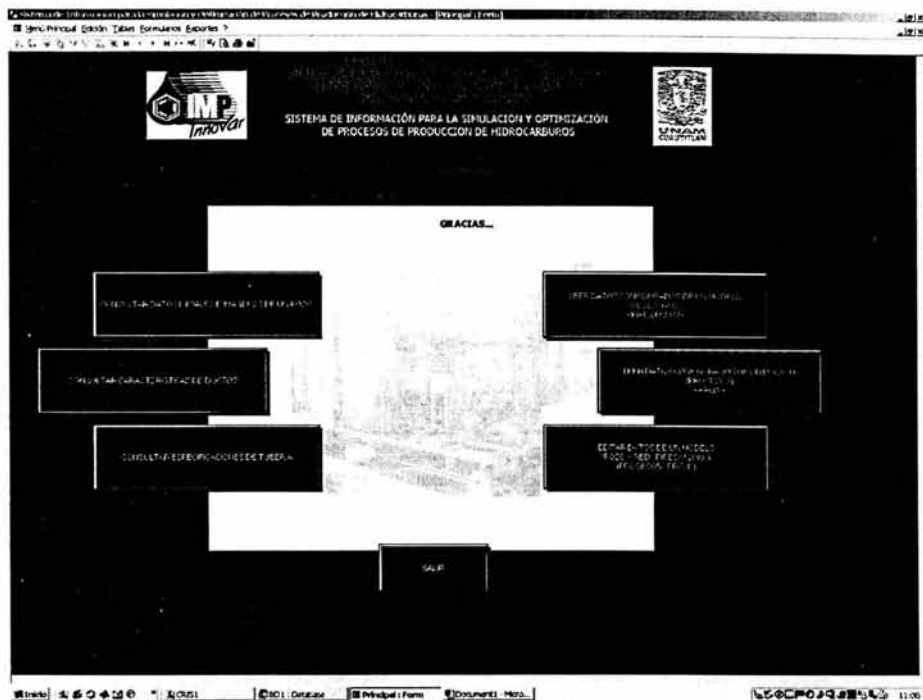


Figura 88. Pantalla Principal.

Esta pantalla muestra seis botones con las opciones para utilizar todo el sistema, además contiene un menú y una barra de herramientas, desde las cuales se accede a la información de manera más específica como a las tablas y a los informes y reportes.

Los botones de la izquierda permiten acceder a los módulos de captura de la información de proceso y los de la izquierda permiten editar, simular e imprimir reportes de los modelos creados en las herramientas de simulación.

Este es un formulario de entrada de datos de los variables de un pozo. Puede agregar un pozo después del último registro.

Variable	Valor
ID	
Nombre	Canal
Ubicación	W-0
Fecha	01/01/1997
Capacidad	3007
Capacidad	2
Pico	30.2
Presión	5.3
Distribución	2
Presión	38.24
Presión	36.94
Presión	1.7
Temperatura	64

EdoMec.gif Plano.gif

Secc_Est.gif

Salir

Figura 89. Formulario: FR_Variables_de_Pozo.

Este formulario muestra todos los registros de los pozos y sus características, y tiene la opción de modificar y agregar nuevos registros directamente en los cuadros de texto. También se pueden observar las imágenes del Estado Mecánico del pozo, la ubicación del pozo (plano) y la Sección Estructural, dando clic en los cuadros de imagen.

Este formulario es de entrada de datos de las Características de Ductos. Puede agregar registros al final

Tramo	<input type="text" value="Rebombeo a B"/>	
Gasto	<input type="text" value="5000"/>	Bariles por día
DI	<input type="text" value="24"/>	Pulgadas
Longitud	<input type="text" value="5.6"/>	Metros
Presion_Entrada	<input type="text" value="25.3"/>	PSIG
Presion_Salida	<input type="text" value="18.2"/>	PSIG

Figura 90. Formulario: FR_Características_de_Ductos.

Este formulario permite consultar datos de los registros guardados en la tabla con el mismo nombre, estos datos acerca de los ductos, también se pueden modificar. El formulario permite agregar nuevos registros a la tabla, en los cuadros de texto.

Este formulario es de entrada de datos de las Especificaciones de Tuberías. Puede agregar registros al final

DI	2.245
Tipo	5
Flota en Agua	<input type="checkbox"/>
Espesor/Pulgadas	0.065
Peso/Lbs	1.604
Peso/Ton	4.234
Area de Flujo	0.02749
Presion Permisible	1379

Salir

Figura 91. Formulario: FR_Especificaciones_de_Tubería.

Este formulario es igual que los dos anteriores, permite visualizar los registros de la tabla y permite modificar y agregar nuevos datos referentes a las especificaciones de las tuberías.



Figura 92. Formulario: Leer_Modelo_Pozo.

En este formulario podemos examinar todas las carpetas y archivos contenidos en la estación de trabajo para localizar los que tengan extensión .bps (esta extensión indica que el archivo es Single Branch, es decir de un solo pozo) y abrirlo. Al abrirlo se oprime el botón Leer datos del modelo y enseguida se abre el software Pipesim para llevar a cabo la lectura, después se debe oprimir el botón Imprimir datos leídos y aparece en el cuadro de la derecha los datos, en forma de hoja de datos.

Informe de Información para la Simulación y Optimización de Procesos de Producción de Hidrocarburos [Informe Leer Pozo - Reporte]

Inicio | Principal | Edición | Datos | Consultas | Reportes

Monitor del Estado: **O2 Vial Demopolo**

Temperatura	24	Presión	1.0
Nivel	1.0	Velocidad	0.0
...

Inicio | Manual del Usuario | Principal Form | Leer_Modo_Pozo | Informe_Leer_Po... | 13:14

Figura 93. Informe: Informe_Leer_Pozo.

Hay botones para ver los informes que contienen datos de entrada y también los reportes, que son muy parecidos a los informes, pero que contienen datos de salida.

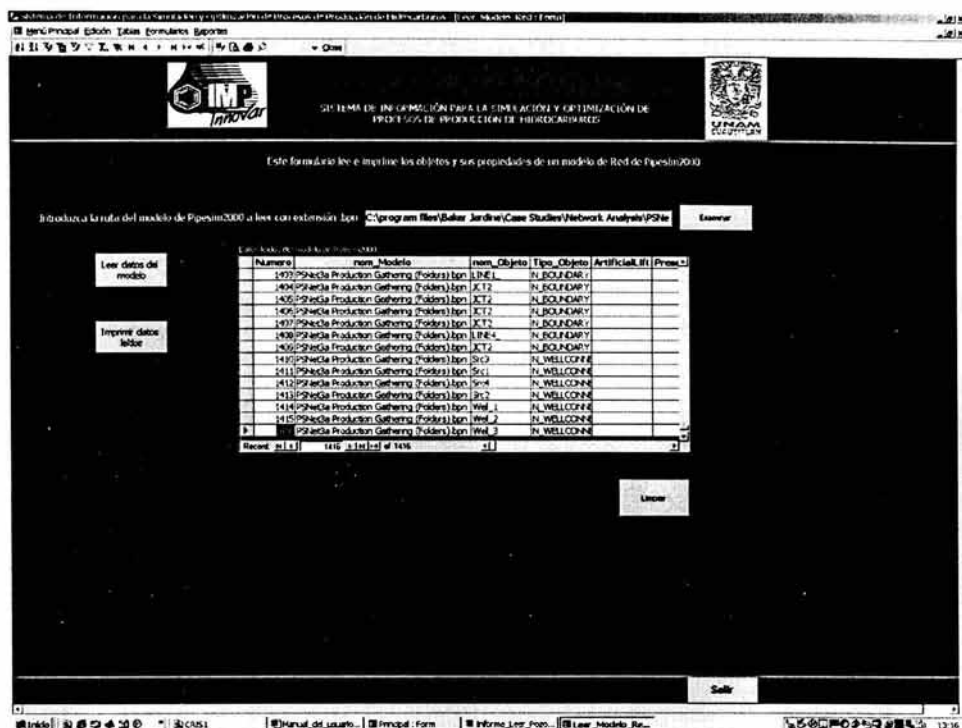


Figura 94. Formulario: Leer_Modelo_Red – Datos impresos.

En este formulario podemos examinar todas las carpetas y archivos contenidos en la estación de trabajo para localizar los que tengan extensión .bpn (esta extensión indica que el archivo es Newtork File, es decir de una red de pozos) y abrirlo. Al abrirlo se oprime el botón Leer datos del modelo y enseguida se abre el software Pipesim para llevar a cabo la lectura, después se debe oprimir el botón Imprimir datos leídos y aparece en el cuadro de la derecha los datos, en forma de hoja de datos.

Este formulario lee e imprime los componentes y propiedades de un Modelo de Pro

Introduzca la ruta del modelo de Pro. II a leer:

Leer datos de un modelo de PROII

Imprimir datos leídos

ID	Nombre_Modelo	Tipo_Objeto	Nombre	In_Out	Pressure	Temperature	Pressure_drop	Pressure_gain
5200	A1.pr.1	Flash	P1	in			0	0
5201	A1.pr.1	Flash	C1V	in		195.2722		0
5202	A1.pr.1	Flash	C1V	in		292.15		0
5203	A1.pr.1	Flash	M11	in		297.0388	0	0
5204	A1.pr.1	Flash	P2	in			0	0
5205	A1.pr.1	Mixer	M12	in			0	0
5206	A1.pr.1	Splitter	S11	in	4709.2245		0	0
5207	A1.pr.1	Compressor	C1C1	in	1723.6099		0	0
5208	A1.pr.1	Compressor	C1C2	in			0	0
5209	A1.pr.1	Simple Heat Exchanger	H1C1	in			0	0
5210	A1.pr.1	Simple Heat Exchanger	H1C2	in			0	0
5211	A1.pr.1	Simple Heat Exchanger	H1C3	in			0	0
5212	A1.pr.1	Simple Heat Exchanger	H1C4	in			0	0

Ver datos organizados en columnas

Figura 95. Formulario: Leer_Modelo_PROII – Datos impresos.

En este formulario podemos examinar todas las carpetas y archivos contenidos en la estación de trabajo para localizar los que tengan extensión .pr1 ó .prz (esta extensión indica que el archivo es de un modelo de procesos) y abrirlo. Al abrirlo se oprime el botón Leer datos del modelo y enseguida se abre el software Pro/ii para llevar a cabo la lectura, después se debe oprimir el botón Imprimir datos leídos y aparece en el cuadro de la derecha los datos, en forma de hoja de datos.

Este formulario permite editar los objetos y sus propiedades de un Modelo de Red de Pipesim 2000, entre la simulación e imprimir resultados.

Introduzca el nombre del Modelo con expresión RegEx:

Leer datos del modelo

Nombre del Modelo: Presiones(psc) Fecha_Hora:
 Nombre del Objeto: Temperaturas(°C)
 Tipo de Objeto: LiquidRate(STB/D)
 GasRate(mmscfd)
 MassFlow(lbs)

Record: 1 of 1 | 7 of 11 | 1 of 8

Numero	Objeto	Propiedad	Valor	Unidad	Presión	Temperatura	LiquidRate	GasRate	MassFlow
179	Pipesim Production Gathering # Lindero-Lps	Orif	0.02	inH	2000	130	39.6	13.5279208	13.946
180	Pipesim Production Gathering # Lindero-Lps	Orif	0.02	inH	2000.0	130.0	39.600	13.5279208	13.946
181	Pipesim Production Gathering # Lindero-Lps	Orif	0.02	inH	2000.0	130.0	39.600	13.5279208	13.946
182	Pipesim Production Gathering # Lindero-Lps	Orif	0.02	inH	2000.0	130.0	39.600	13.5279208	13.946
183	Pipesim Production Gathering # Lindero-Lps	Orif	0.02	inH	2000.0	130.0	39.600	13.5279208	13.946
184	Pipesim Production Gathering # Lindero-Lps	Orif	0.02	inH	2000.0	130.0	39.600	13.5279208	13.946
185	Pipesim Production Gathering # Lindero-Lps	Orif	0.02	inH	2000.0	130.0	39.600	13.5279208	13.946
186	Pipesim Production Gathering # Lindero-Lps	Orif	0.02	inH	2000.0	130.0	39.600	13.5279208	13.946
187	Pipesim Production Gathering # Lindero-Lps	Orif	0.02	inH	2000.0	130.0	39.600	13.5279208	13.946
188	Pipesim Production Gathering # Lindero-Lps	Orif	0.02	inH	2000.0	130.0	39.600	13.5279208	13.946
189	Pipesim Production Gathering # Lindero-Lps	Orif	0.02	inH	2000.0	130.0	39.600	13.5279208	13.946
190	Pipesim Production Gathering # Lindero-Lps	Orif	0.02	inH	2000.0	130.0	39.600	13.5279208	13.946
191	Pipesim Production Gathering # Lindero-Lps	Orif	0.02	inH	2000.0	130.0	39.600	13.5279208	13.946

Figura 96. Formulario: Edición_Modelo_Red_Pozo – Mostrar Resultados.

Este formulario permite editar los datos de entrada de un modelo de pozo o de red de Pipesim, además de correr la simulación y leer e imprimir los resultados.

Se oprime el botón Leer datos del modelo y enseguida se abre el software Pipesim para llevar a cabo la lectura, después se debe oprimir el botón Mostrar datos y aparecen en el cuadro de la derecha los datos, en forma de hoja de datos

Aparecen los botones Introducir datos y Simular. Para poder introducir los datos se tienen que modificar en el formulario de impresión de datos, arriba de estos botones, estos cambios deben respetar las reglas del software. Hecho esto, se oprime el botón Introducir datos y se introducen los datos al modelo, y al mismo tiempo se guardan los cambios en la base de datos. Y aparece un mensaje que indica que se guardaron los datos. Posteriormente se puede oprimir el botón Simular y se realiza la simulación, a menos que haya errores.

Si se oprime el botón Lectura Resultados, se leen los archivos de texto del software Pipesim, después se oprime el botón Mostrar Resultados y aparece el cuadro de la derecha en forma de Hoja de Datos, en donde se pueden observar los resultados, si se quiere un Reporte de los mismos, se oprime el botón Ver Reporte.

The screenshot shows a software window titled 'Edición_Modelo_PROII'. At the top, there are logos for 'IMP Innovar' and 'UNAM'. Below the logos, there is a table with the following data:

Nombre	Objeto	Nombre	Fecha	Hora
Simple Heat Exchanger	CO02		26/07/2003	13:37:54
Simple Heat Exchanger	H01		26/07/2003	13:37:54
Simple Heat Exchanger	CO01		26/07/2003	13:37:54
Simple Heat Exchanger	H07		26/07/2003	13:37:54
Stream Data	1		26/07/2003	13:37:54

Below this table, there are several input fields for component details, such as 'Nombre_Modelo', 'Tipo_Objeto', 'Nombre', 'Hot_Pressure_Drop', 'Cold_Pressure_Drop', 'SFeed_Stream_Hot', 'SFeed_Stream_Cold', 'EFeed_Stream_Hot', 'EFeed_Stream_Cold', 'SProduct_Stream_Hot', 'EProduct_Stream_Hot', 'SProduct_Stream_Cold', 'EProduct_Stream_Cold', 'In_Out', and 'Fecha_Hora'. There are also buttons for 'Leer datos de un modelo de PROII', 'Imprimir datos leídos', 'Ver Informe', 'Introducir Datos', and 'Simular'.

At the bottom, there is a table with the following data:

Nombre_Proceso	Tipo_Objeto	Nombre	Unidad	Temp_In	Temp_Out	Pres_In	Pres_Out	Qp	Numero_Instalacion
Al p1	Simple Heat Exch	H02	0						2
Al p1	Simple Heat Exch	CO02	0						2
Al p1	Simple Heat Exch	H01	0						2
Al p1	Simple Heat Exch	CO01	0						2
Al p1	Simple Heat Exch	H07	0						2

The interface also includes a status bar at the bottom with information like 'Manual de usuario', 'BD1: Database', 'Principal: Form', and 'C:\Programas\IMP\...'.

Figura 97. Formulario: Edición_Modelo_PROII – Mostrar Resultados.

Este formulario permite editar los datos de entrada de un modelo de procesos de Pro/ii, además de correr la simulación y leer e imprimir los resultados.

Se oprime el botón Leer datos de un modelo de Pro/ii y enseguida se abre el software Pro/ii para llevar a cabo la lectura, después se debe oprimir el botón Imprimir datos leídos y aparecen en el cuadro de la derecha los datos. Se llena un cuadro con todos los componentes del modelo incluyendo: Tipo de Objeto, el Nombre y la Fecha-Hora en que fueron leídos. Al dar click en uno de ellos aparece un pequeño cuadro con las características específicas de ese componente.

Aparecen los botones Introducir datos y Simular. Para poder introducir los datos se tienen que modificar en el cuadro del componente que aparece al dar clic en alguna componente del cuadro en donde aparecen los datos leídos (Figura 38), precisamente arriba de estos botones; estos cambios deben respetar las reglas del software. Hecho esto, se oprime el botón Introducir datos y se introducen los datos al modelo, y al mismo tiempo se guardan los cambios en la base de datos. Y aparece un mensaje que indica que se guardaron los datos. Es importante saber que se pueden realizar cambios a una o todas las características de uno o todos los componentes, esto para poder ver cambios en los resultados.

Posteriormente se puede oprimir el botón Simular, y se ejecuta la simulación.

Si se oprime el botón Lectura Resultados, se leen los archivos de texto del software Pipesim, después se oprime el botón Imprimir Resultados y aparece el cuadro de la derecha en forma de Hoja de Datos, en donde se pueden observar los resultados, si se quiere un Reporte de los mismos, se oprime el botón Ver Reporte.

3.8. Instrucciones más importantes utilizadas en el Sistema de Información.

Conexión:

Para conectarnos a la Base de datos creamos un objeto ADO de tipo conexión, para ello usamos Server.CreateObject, una vez que tenemos el objeto, le indicamos el driver que tiene que usar, en este caso el de Access y en donde se encuentra la base de datos.

```
Sub OpenMyDB()
Dim cnn1 As New Connection
Dim rst1 As Recordset

'Crea la conexión
cnn1.Open "Provider=Microsoft.Jet.OLEDB.4.0;" & "Data Source=C:\Documents and Settings\IMP\Mis
documentos\CRIS\BD1.mdb;"

'Crea la referencia al conjunto de registros y establece sus propiedades
Set rst1 = New ADODB.Recordset
rst1.CursorType = adOpenKeyset
rst1.LockType = adLockOptimistic

'Abre el conjunto de registros
rst1.Open "Variables", cnn1
rst1.MoveFirst
rst1.AddNew
rst1.Fields("Temperature").value = 34
rst1.Update
```

Desconexión:

Para cerrar la conexión con la base de datos usaremos el método Close y seguidamente como ya no usaremos el objeto lo destruimos asignándole nothing.

```
'Limpia objetos
rst1.Close
cnn1.Close
Set rst1 = Nothing
Set cnn1 = Nothing
```

Guardar:

```
rst1.AddNew
rst1.Fields("nom_Objeto").value = nom(i)
rst1.Fields("Tipo_Objeto") = "N_SOURCE"
rst1.Fields("nom_Modelo") = nombreachivo
rst1.Fields("Fecha_Hora") = Now

nom1 = rst1.Fields("nom_Objeto").value
RedObj.GetBoundaryProperties nom1, pres, temp, fluido, 0
If (pres <> -7777) Then
    rst1.Fields("Pressure(psia)").value = pres
    rst2.Fields("Pressure(psia)").value = pres
End If
If (temp <> -7777) Then
    rst1.Fields("Temperature(C)").value = temp
    rst2.Fields("Temperature(C)").value = temp
```

```

End If
If (fluido <> -7777) Then
  rst1.Fields("LiquidRate(STB/D)").value = fluido
  rst2.Fields("LiquidRate(STB/D)").value = fluido
End If
RedObj.GetBoundaryProperties nom1, pres, temp, fluido, 1
If (fluido <> -7777) Then
  rst1.Fields("GasRate(mmscf/d)").value = fluido
  rst2.Fields("GasRate(mmscf/d)").value = fluido
End If
RedObj.GetBoundaryProperties nom1, pres, temp, fluido, 2
If (fluido <> -7777) Then
  rst1.Fields("MassRate(lb/s)").value = fluido
  rst2.Fields("MassRate(lb/s)").value = fluido
End If

```

Borrado:

```

rst2.Delete adAffectCurrent
rst2.Update
rst2.Close
Set rst2 = Nothing

```

Consulta:

- SELECT [Tipo_Objeto], [Nombre], [Fecha_Hora] FROM Edicion_Modelo_PROII WHERE Fecha_Hora='09/01/2004 13:21:19';
- SELECT [Nombre_Modelo], [Tipo_Objeto], [Nombre], [Temperature], [Pressure], [Pressure_drop] FROM Flash_Unit_Operation WHERE Fecha_Hora='09/01/2004 13:22:47';

Ejemplo:

```

'Recupera datos de la tabla en la que se acaban de guardar los valores y los muestra en el ListBox
Lista.RowSourceType = "Tabla/Consulta"
inst = "SELECT Numero, nom_Modelo, Nombre_Nodo, Tipo_Nodo, Pressure, Temperature, LiquidRate, GasRate, MassRate, GLR, WaterCut, PressureResidual, RateResidual, LGR, MaxPres, CompPower, Choke_DP FROM Temporal_Reporte ORDER BY Numero"
Lista.RowSource = inst
Lista.Requery

- inst = "SELECT Tipo_Objeto, Nombre, Fecha_Hora FROM Edicion_Modelo_PROII WHERE Fecha_Hora= "" & horario2 """
- inst = "SELECT Nombre_Modelo, Tipo_Objeto, Nombre, Temperature, Pressure, Pressure_drop FROM Flash_Unit_Operation WHERE Fecha_Hora= "" & horario_flash & """
- inst = "SELECT Nombre_Modelo, Tipo_Objeto, Nombre, Temperature, Pressure, Pressure_drop FROM Valve_Unit_Operation WHERE Fecha_Hora= "" & horario_valve & """
- inst = "SELECT Nombre_Modelo, Tipo_Objeto, Nombre, Temperature, Pressure, Pressure_drop FROM Mixer_Unit_Operation WHERE Fecha_Hora= "" & horario_mixer & """
- inst = "SELECT Nombre_Modelo, Tipo_Objeto, Nombre, Temperature, Pressure, Pressure_gain, Pressure_ratio FROM Pump_Unit_Operation WHERE Fecha_Hora= "" & horario_pump & """
- inst = "SELECT Nombre_Modelo, Tipo_Objeto, Nombre, Temperature, Pressure, Pressure_drop FROM Splitter_Unit_Operation WHERE Fecha_Hora= "" & horario_split & """
- inst = "SELECT Nombre_Modelo, Tipo_Objeto, Nombre, Flow_efficiency, Inlet_Temperature, Outlet_Temperature, Inlet_Pressure, Outlet_Pressure FROM Pipe_Unit_Operation WHERE Fecha_Hora= "" & horario_pipe & """

```

- inst = "SELECT Nombre_Modelo, Tipo_Objeto, Nombre, Outlet_Temperature, Outlet_Pressure, Pressure_drop, Pressure_ratio FROM Expander_Unit_Operation WHERE Fecha_Hora= '' & horario_expan & ''"
- inst = "SELECT Nombre_Modelo, Tipo_Objeto, Nombre, Temperature, Pressure FROM Compressor_Unit_Operation WHERE Fecha_Hora= '' & horario_compr & ''"
- inst = "SELECT Nombre_Modelo, Tipo_Objeto, Nombre, Utility_side_flag, Utility_temp_In, Utility_temp_Out, Utility_saturation_pressure, Utility_flowrate, Utility_Cp, Number_Shell_Passes, Number_Tube_Passes FROM SHExchanger_Unit_Operation WHERE Fecha_Hora= '' & horario_shex & ''"
- inst = "SELECT Nombre_Modelo, Tipo_Objeto, Nombre, Specified_Value, Calculated_Value, Absolute_Tolerance, Relative_Tolerance, Primary_Units_Measure, Primary_Basis, Primary_Wet_Dry, Reference_Units_Measure, Reference_Basis, Reference_Wet_Dry, Number_Pumparounds, Temperature, Liquid_fraction_basis, Molar_Rates, Pressures, Temperatures, Liquid_fractions, Temperature_reference FROM Column_Unit_Operation WHERE Fecha_Hora= '' & horario_colu & ''"
- inst = "SELECT Nombre_Modelo, Tipo_Objeto, Nombre, Component_Mole_Fractions, Vapor_Phase_Comp, Liquid_Phase_Comp, Solid_Phase_Comp, Temperature, Pressure, Mole_flow_rate, Mole_fraction_vapor, Mole_fraction_liquid, Mole_fraction_water, Mole_fraction_inerts, Weight_fraction_solids, Molar_enthalpy, Molar_entropy, Molecular_weight, Tc, Pc, Vc, Cp_Cv_Ratio, Fecha_Hora FROM Stream_Data WHERE Fecha_Hora= '' & horario_stream & ''OR Id> '' & contadorstr & ''"
- inst = "SELECT Nombre_Modelo, Tipo_Objeto, Nombre, Component_Type, Phase, Molecular_weight, Specific_gravity, NBP, Normal_melting_point, Acentric_factor, Critical_temperature, Critical_pressure, Critical_volume, Crit_Compress_Factor, Liquid_Molar_Volume, Carbon_number FROM Component_Data WHERE Fecha_Hora= '' & horario_comp & ''"
- inst = "SELECT Numero, nom_Modelo, Nombre_Nodo, Tipo_Nodo, Pressure, Temperature, LiquidRate, GasRate, MassRate, GLR, WaterCut, PressureResidual, RateResidual, LGR, MaxPres, CompPower, Choke_DP FROM Temporal_Reporte ORDER BY Numero"
- inst = "SELECT Numero, nom_Modelo, Nombre_Nodo, Tipo_Nodo, Pressure, Temperature, LiquidRate, GasRate, MassRate, GLR, WaterCut, PressureResidual, RateResidual, LGR, MaxPres, CompPower, Choke_DP FROM Temporal_Reporte_Pozo ORDER BY Numero, Nombre_Nodo"
- inst = "SELECT Id, Pressure, Temperature, LiquidRate, GasRate, MassRate, GLR, WaterCut FROM Variables_Lectura_Pipesim2000 ORDER BY Id"
- inst = "SELECT Numero, Nombre, TipoNodo, Pressure, Temperature, LiquidRate, GasRate, MassRate, GLR, WaterCut FROM Reporte ORDER BY Numero"
- inst = "SELECT Tipo_Objeto, Nombre FROM Edicion_Modelo_PROII" 'WHERE Fecha_Hora =' & fech & ''"
- inst = "SELECT Tipo_Objeto, Nombre, Fecha_Hora FROM Edicion_Modelo_PROII WHERE Fecha_Hora= '' & horario & ''"
- inst = "SELECT Tipo_Objeto, Nombre, Fecha_Hora FROM Edicion_Modelo_PROII WHERE Fecha_Hora= '' & horario2 & ''"
- inst = "SELECT Numero, nom_Modelo, Nombre_Nodo, Tipo_Nodo, Pressure, Temperature, LiquidRate, GasRate, MassRate, GLR, WaterCut, PressureResidual, RateResidual, LGR, MaxPres, CompPower, Choke_DP FROM Temporal_Reporte ORDER BY Numero"
- inst = "SELECT Numero, nom_Modelo, Nombre_Nodo, Tipo_Nodo, Pressure, Temperature, LiquidRate, GasRate, MassRate, GLR, WaterCut, PressureResidual, RateResidual, LGR, MaxPres, CompPower, Choke_DP FROM Temporal_Reporte_Pozo ORDER BY Numero, Nombre_Nodo"
- inst = "SELECT Id, Pressure, Temperature, LiquidRate, GasRate, MassRate, GLR, WaterCut FROM Variables_Lectura_Pipesim2000 ORDER BY Id"
- inst = "SELECT Numero, Nombre, TipoNodo, Pressure, Temperature, LiquidRate, GasRate, MassRate, GLR, WaterCut FROM Reporte ORDER BY Numero"

3.9. Puesta en marcha del sistema. Prueba.

3.9.1. Requerimientos de instalación del Sistema.

Hay que asegurarse de que la computadora donde se vaya a instalar el Sistema de Información para la Simulación y Optimización de Procesos de Producción de Hidrocarburos cubra todos los requerimientos que a continuación se listan, pues de ello depende que se presenten o no problemas durante la ejecución del mismo.

Requerimientos de hardware:

- Computadora personal o estación de trabajo que contenga lo siguiente:
- Procesador: Pentium 4 – M CPU 2.00 GHz o mayor.
- Al menos 1,047,992 (1 GB DDR) de RAM
- Velocidad del bus frontal: 400 MHz
- Unidad de disco flexible: Unidad de disco flexible de 3.5" de capacidad 1.44 MB integrado.
- Unidad de CD-ROM.
- Disco duro con capacidad de al menos 10 GB.
- Gráficos: Controlador de video integrado de 64 MB y soporte 2D/3D para resolución de 1024x768 a 16 millones de colores.
- Pantalla: al menos de 15 Pulgadas TFT tipo LCD, UXGA matriz activa.
- Resolución de pantalla mínima: 1280 x 1024 pixeles.
- Controlador de red.
- Fax modem.
- Un Teclado.
- Un Mouse.

Requerimientos de software:

- Microsoft Windows 2000 Professional. Microsoft (R) Windows. Versión 5.0 (Build 2195: Service Pack 2).. Copyright (C) 1981-1999 Microsoft Corp.
- Microsoft Access 2000. Microsoft Access 2000 (9.0.2720). Copyright (C) 1981-1999 Microsoft Corp.
- Microsoft Visual Basic para Aplicaciones 6.0 (VBA).
- PIPESIM (Modelos de Sistemas de Producción de Gas y Aceite Integrados).
- PRO/II (Simulación de Procesos de Refinería).
- El Sistema de la PC debe ser puesto en la fecha actual. La seguridad de las herramientas de simulación utiliza la fecha actual de la computadora.

3.9.2. Implantación del sistema.

La implantación del sistema consiste en llevar dicho sistema a producción, es decir, a explotación diaria.

En el caso del sistema de información de procesos de producción se construyó una aplicación nueva en todos los aspectos, ya que no existía un sistema.

Para poder implantarlo correctamente se requiere realizar pruebas de todo el sistema de información, por separado.

Para evitar errores al implantar el sistema de información, se crearon tablas de prueba para detectar los errores existentes y evitarlos en la creación de las tablas originales. Además se tenía que probar también la comunicación entre las aplicaciones y de éstas con la base de datos.

Los formularios o programas de la aplicación se estuvieron corriendo varias veces con ayuda de los usuarios para descubrir errores o fallas, así como encontrar otros requerimientos por parte de ellos de dichos formularios.

Cabe mencionar que también se hicieron pruebas con las herramientas de simulación y optimización de manera independiente, ya que se necesita conocer su funcionamiento para determinar en qué casos el error es del propio simulador y no del sistema de información. Es decir, se trabajó con modelos de simulaciones incluidos con el software y se cambiaron e introdujeron datos y se veían los tipos de reportes y el formato de los mismos que la propia herramienta proporciona, entre otras cosas.

Ya que se realizaron las pruebas por separado al sistema, ahora si se instala el software necesario y el sistema de información en la computadora o estación de trabajo en donde se vaya a utilizar. Posteriormente se realizan pruebas a la aplicación completa corriendo, es decir, al sistema de información ejecutable. Esta es la fase de producción del sistema.

Al mismo tiempo se entrega documentación y se informa a los usuarios del nuevo sistema sobre las características de su funcionamiento.

En este caso se explicó la forma de utilizar el sistema y su utilidad, y también se entregó un Manual en donde se explica qué hace y que se requiere en cada una de los formularios del sistema de información, así como las características de los reportes que se devuelven del mismo.

CAPÍTULO 4

EVALUACIÓN DEL SISTEMA Y PROPUESTAS A FUTURO

CAPÍTULO 4. EVALUACIÓN DEL SISTEMA Y PROPUESTAS DE SOLUCIÓN PARA LÍNEAS DE DESARROLLO FUTURO.

4.1 Evaluación y análisis del sistema.

Después de haber llevado a cabo la implantación y pruebas del sistema de información de procesos de producción de hidrocarburos se realiza la Evaluación y análisis del sistema. Esta revisión cumplió con los objetivos siguientes:

- Evaluar el sistema de información operativo que se ha desarrollado.
- Evaluar los procedimientos de desarrollo de sistemas para determinar en qué medida podría haberse mejorado el proyecto.

Para cumplirlos se habló con la gente que ayudó en la realización de las pruebas y se pidieron opiniones de los resultados, así como las opiniones sobre los procesos. Aquí es donde se anotan las mejoras que se le pudieran hacer al sistema para su buen mantenimiento.

- Se requieren nuevas tablas, ya que es demasiada la información que contienen, así como también relaciones entre las mismas. Es decir, mejorar la base de datos relacional.
- Poner más indicaciones para el usuario, ya que a veces no utiliza algunos objetos o provoca errores porque la base de datos no es cerrada como se debiera al terminar de utilizar un formulario, sin salir completamente del sistema.
- Mejorar la presentación de los formularios, es decir el diseño de las interfaces de usuario, ya que a veces, de acuerdo a la resolución de la pantalla (si es que se llega a instalar en diferentes máquinas, lo cual también se probó) no se ven completos.

También se examinaron los costos y las necesidades del proyecto, como este sistema es parte de un proyecto dentro de la institución (IMP) entra en el estudio de factibilidad del mismo, pero después de un análisis se encontró que se realizó con menos presupuesto del que se tenía contemplado, en cuanto a las necesidades del proyecto, no se integraron todas las herramientas de simulación que se tenían planeadas desde un origen, debido a que no estaban disponibles durante el desarrollo del sistema de información, pero lo que se realizó es suficiente para poder utilizar el sistema con beneficios para la organización.

Como herramienta de medición de satisfacción al cliente, se elaboró el siguiente cuestionario:

- ¿Cumple el nuevo sistema de información las metas y los objetivos identificados y perfeccionados en las primeras fases del proyecto?
- ¿Da el sistema un soporte adecuado al proceso de transacciones, la elaboración de informes de gestión y las necesidades de ayuda a la toma de decisiones en la empresa?
- ¿Se han hecho realidad los beneficios previstos?
- ¿Qué opinan los usuarios del nuevo sistema? ¿Cómo sería posible mejorar las relaciones con los usuarios en proyectos futuros?
- ¿Debería realizarse de inmediato alguna de las mejoras propuestas al sistema? Manejo de prioridades.
- ¿Son adecuados los controles internos?

En base a las respuestas se tiene que el Sistema de Información para la Simulación y Optimización de Procesos de Producción de Hidrocarburos, si cumple con los objetivos identificados al inicio de éste, además de que si produce informes y reportes de los datos que se requieren y si ayuda en la toma de decisiones de manera más rápida.

El proceso de revisión es una sesión de preguntas y respuestas en torno a los beneficios a obtener en la realización de proyectos futuros. A lo largo de esta evaluación se identificaron necesidades de mejora en el sistema de información de procesos de producción. Estas necesidades son la base para el soporte de sistemas.

4.1.1. Soporte del sistema.

El soporte del sistema es el mantenimiento permanente del mismo después de que ha sido puesto en operación diaria. Ello incluye tanto el mantenimiento de los programas como las mejoras que pudieran añadirse al sistema y adaptarlo a nuevas necesidades (reingeniería), así como asesoría a los usuarios del mismo.

Estas fases se describen a continuación:

- Corrección de errores. Se encontraron errores relacionados a fallos en la comunicación del software con la base de datos, y otros a defectos en el diseño del sistema, así también otros por mal uso del sistema de información. Se emprendieron acciones de corrección. Estas son:
 - Realización de cambios predecibles en los programas para corregir errores que se cometieron durante el diseño y la implantación del sistema.
 - Se trató de evitar que dichas correcciones afecten a otros aspectos en la función del sistema, ya que se generarían otros errores.
- Recuperación del sistema: superar los fallos generales del sistema. En este caso no ha pasado un fallo al sistema, o que sea abortado. De cualquier manera se conocen los riesgos de esto y se sabe que se tienen que hacer respaldos de las tablas y la base de datos para evitar pérdidas. La ventaja que se tiene es que en Microsoft Access hay asistentes para la solución de este tipo de problemas.
- Asistencia al usuario final. Para la realización de esto, se observó a los usuarios en la utilización del sistema, se realizaron pláticas con los mismos para ver su grado de satisfacción, y se les ayudó en los requerimientos de uso y errores simples.
- Mejoras y reingeniería del sistema. Las mejoras se realizaron al mismo sistema de información y la reingeniería consiste en agregar nuevos módulos a la aplicación, que hasta ahora no se ha requerido, pero si se requiere será de manera sencilla escribir los nuevos programas, aunque se requiere volver a las fases de diseño del sistema y base de datos.

4.2 Resultados.

El resultado es un Sistema de Información para la Simulación y Optimización de Procesos de Producción de Hidrocarburos, el cual realiza la interacción de dos herramientas de simulación (PIPESIM y PRO/II). Además contiene una base de datos única, la cual almacena datos de proceso, así como datos requeridos y generados por el software.

El sistema de información es una herramienta necesaria en la producción de hidrocarburos, ya que el tiempo que antes se invertía en la obtención de las variables y la realización de la optimización era mas tardado porque el software de simulación se ejecutaba de manera aislada. Con este sistema se obtienen beneficios como: ahorro en tiempo, energía y dinero.

4.3 Propuestas de solución para líneas de desarrollo futuro.

A continuación se presentan algunos proyectos que se pueden llevar a cabo a partir de este trabajo de tesis.

4.3.1. Adición de otras herramientas de simulación y optimización al sistema de información.

Debido a problemas que se tuvieron con la adquisición del software de simulación, sólo se pudieron comunicar dos de ellos. Pero el sistema está abierto para poder conectar otras herramientas de simulación y optimización que se vayan requiriendo en la utilización de este, o para mejorarlo.

En la Figura 98 se muestra el ejemplo de la comunicación de nuevo software con la base de datos, de acuerdo a la clasificación del mismo de acuerdo a su aplicación:

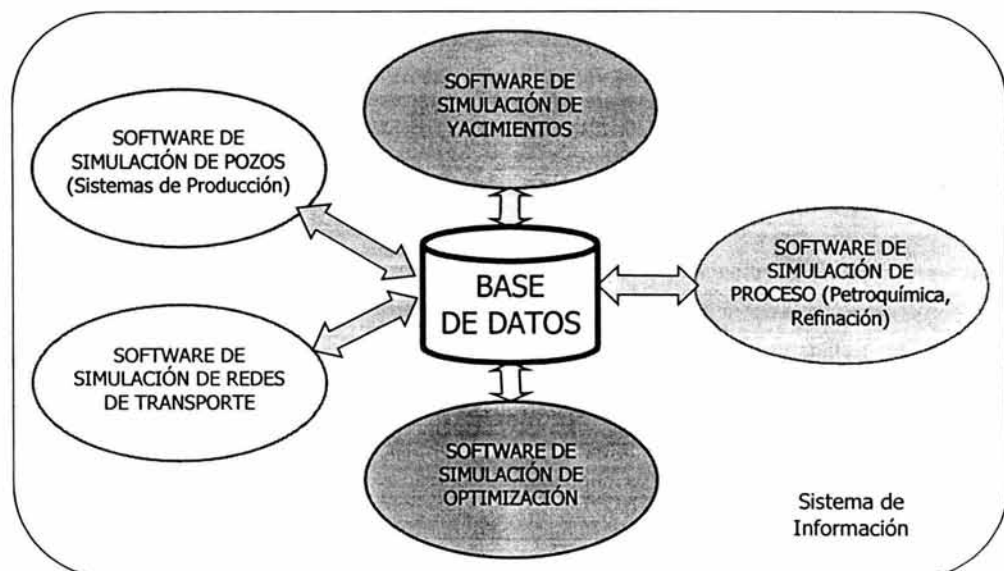


Figura 98. Sistema de información conteniendo software de todos los procesos de producción de hidrocarburos.

4.3.2. Proyecto de Microsoft Access (aplicación cliente-servidor).⁶⁹

4.3.2.1. Qué es un proyecto de Microsoft Access

Un proyecto de Microsoft Access (.ADP) es un nuevo tipo de archivo de Access que proporciona un acceso eficaz en modo nativo a una base de datos de Microsoft SQL Server mediante la arquitectura de componentes OLE DB. Con un proyecto de Access se puede crear una aplicación cliente-servidor con la misma facilidad que una aplicación servidor de archivos. Esta aplicación cliente-servidor puede ser una solución tradicional basada en formularios e informes, una solución basada en Web mediante el uso de páginas de acceso a datos o una combinación de ambas.

⁶⁹ Referencia 9. Dobson, Rick. España, 2002, 997 pp.

Un proyecto de Access recibe el nombre de proyecto porque contiene únicamente objetos de base de datos basados en códigos o en HTML: formularios, informes, páginas de acceso a datos, macros y módulos. Estos son los objetos de base de datos que se utilizan para crear una aplicación. A diferencia de una base de datos de Microsoft Access, un proyecto de Access no contiene datos ni objetos basados en definición de datos: tablas, vistas, diagramas de base de datos o procedimientos almacenados (que también pueden contener código de aplicación). En su lugar, estos objetos se almacenan en la base de datos de SQL Server.

4.3.2.2. Trabajar con un proyecto de Access

El trabajo con un proyecto de Access es muy parecido al trabajo con una base de datos de Access. El proceso de crear formularios, informes, páginas de acceso a datos, macros y módulos es prácticamente igual al que se utiliza para crear una base de datos de Access.

Después de conectarse a una base de datos de SQL Server, se pueden ver, crear, modificar y eliminar tablas, vistas, procedimientos almacenados y diagramas de base de datos mediante la utilización de las Herramientas de diseño de Microsoft SQL Server, que son las mismas que se utilizan en Microsoft Visual Interdev y Microsoft Visual Basic. La interfaz de usuario para trabajar con objetos de estas bases de datos es diferente de la de los objetos equivalentes en una base de datos de Access; no obstante, es igual de sencilla para el usuario. Un proyecto de Access también contiene muchos de los asistentes que se utilizan en una base de datos de Access como, por ejemplo, el Asistente para formularios, el Asistente para informes y el Asistente para páginas, que se pueden utilizar para crear de forma rápida un prototipo o una aplicación sencilla.

4.3.2.3. Cuándo convertir una base de datos de Microsoft Access a Microsoft SQL Server

Con el transcurso del tiempo, la mayoría de las aplicaciones de base de datos crecen, se vuelven más complejas y necesitan ser compatibles para más usuarios. En algún momento de la existencia de la aplicación de base de datos de Microsoft Access, conviene considerar la posibilidad de una conversión a Microsoft SQL Server para optimizar el rendimiento, escalabilidad, seguridad, confiabilidad, disponibilidad y capacidad de recuperación de la base de datos y de la aplicación.

Antes de realizar la conversión, se tiene que realizar la normalización de las tablas, la cual es el proceso de transformación de las complejas presentaciones de usuarios y de los almacenamientos de datos en conjuntos estables de estructuras de datos de menor tamaño. Además de ser más sencillas, tales estructuras son más estables. Las estructuras de datos normalizados son más fáciles de mantener. *

El proceso de convertir a SQL Server consiste en transferir alguno o todos los objetos de base de datos de una base de datos de Microsoft Access (.MDB) a una base de datos nueva o existente de Microsoft SQL Server o a un proyecto de Microsoft Access (.ADP) nuevo.

El Asistente para convertir a SQL Server convierte una base de datos de Access en una base de datos nueva o existente de Microsoft SQL Server 6.5 y 7.0 o en un proyecto de Access nuevo, mediante la conversión de los datos y definiciones de datos y la transferencia de objetos de base de datos. Hay tres formas de utilizar el Asistente para convertir a SQL Server:

- Convertir sólo datos o definiciones de datos de una base de datos de Access a una base de datos de SQL Server.
- Crear una base de datos cliente de Access como una base de datos servidor de SQL Server, para poder crear una aplicación cliente-servidor.

* Las tres primeras formas normales estaban contempladas en la formulación original de la teoría relacional de Codd, y, en la mayoría de los casos, es todo lo que se ha de tener en cuenta. Pero actualmente existen otras dos formas normales (Boyce/Codd, cuarta y quinta) que se desarrollaron para manejar casos especiales.

- Convertir todos los objetos de una base de datos de Access a un proyecto de Access para poder crear una aplicación cliente-servidor.

4.3.2.4. Arquitectura de un proyecto de Access

El proyecto de Access representa el lado cliente de una aplicación cliente-servidor. La base de datos SQL Server a la que se conecta el proyecto de Access representa el lado servidor de la aplicación cliente-servidor. Una conexión OLE DB vincula las partes cliente y servidor de esta aplicación, y sirve como conducto a través del cual el proyecto de Access obtiene el acceso a los datos (Ver Figura 99).

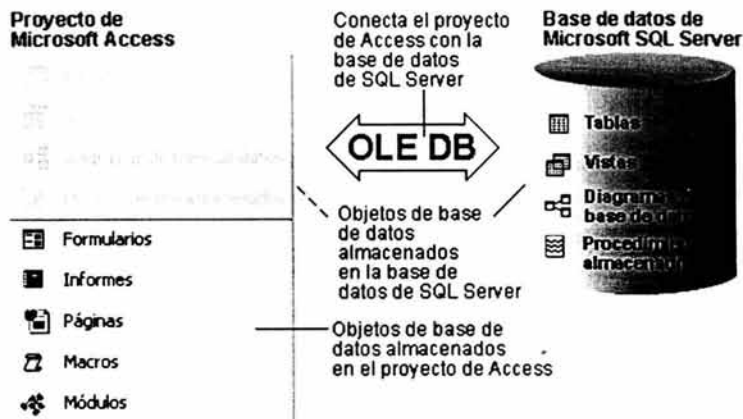


Figura 99. Arquitectura de un proyecto de Access.

4.3.3. Sistema de Información en línea/tiempo real.

4.3.3.1. Sistemas en línea.⁷⁰

Un sistema en línea es aquel que acepta material de entrada directamente del área donde se creó. También es el sistema en el que el material de salida, o el resultado de la computación, se devuelve directamente a donde es requerido.

Esto usualmente significa que el sistema computacional tiene una arquitectura de hardware parecida a la de la Figura 100.

⁷⁰ Referencia 30. Yourdon, Edward. México, 1993, 735 pp.

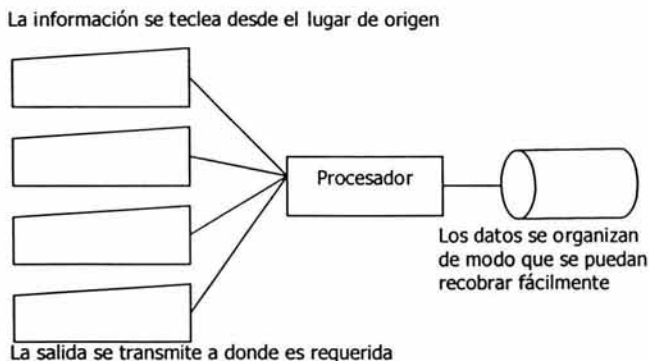


Figura 100. Un sistema en línea.

Una característica común de los sistemas en línea es que entran datos a la computadora o se les recibe de ella en forma *remota*. Es decir, los usuarios del sistema computacional normalmente interactúan con la computadora desde terminales * que pueden estar localizadas a cientos de kilómetros de la computadora misma.

Otra característica de un sistema en línea es que los datos almacenados, es decir, sus archivos o su base de datos, usualmente se organizan de tal manera que los componentes individuales de información (tales como un registro individual de reservación aérea o un expediente individual personal) puedan ser recuperadas modificados o ambas cosas 1) rápidamente y 2) sin tener necesariamente que efectuar accesos a otros componentes de información del sistema. Esto contrasta enormemente con los sistemas fuera de línea o en lotes (batch), que eran más comunes en las décadas de los 60 y 70. En un sistema computacional por lotes, la información suele recuperarse de una manera secuencial, lo cual significa que el sistema computacional lee todos los registros de la base de datos, procesando y actualizando aquellos para los cuales haya actividad. La diferencia entre un sistema computacional por lotes y uno en línea es análoga a la diferencia entre encontrar una selección musical específica en un cassette o en un disco; lo primero involucra el acceso secuencial a través de todas las pistas, mientras que lo segundo permite el acceso aleatorio a cualquiera de las pistas sin tener que escuchar las demás.

La decisión de convertir o no un sistema ordinario en un sistema en línea es, una decisión de puesta en práctica, es decir, no es algo que debiera ser determinado por el analista de sistemas sino más bien por las personas que ponen en práctica el sistema. Sin embargo, dado que decidir tal cosas tiene un impacto tan obvio en el usuario (la presencia o ausencia de terminales e computadora, etc.), es una decisión de puesta en práctica en la cual habitualmente los usuarios querrán participar.

4.3.3.2. Sistemas de tiempo real.⁷¹

Un sistema de tiempo real es considerado por muchos como una variante de un sistema en línea; de hecho, muchos usan ambos términos indistintamente. Sin embargo, es importante distinguirlos.

Un sistema computacional de tiempo real puede definirse como aquél que controla un ambiente recibiendo datos, procesándolos y devolviéndolos con la suficiente rapidez como para influir en dicho ambiente en ese momento.

* Actualmente la palabra "terminal" se usa de manera tan común en la sociedad que en realidad no requiere definirse. Sin embargo se deben mencionar sus sinónimos: "pantalla", "estación de trabajo", "teclado", etc.

⁷¹ Referencia 30. Yourdon, Edward. México, 1993, 735 pp.

La expresión "con la suficiente rapidez" está, desde luego, sujeta a muchas interpretaciones. Ciertamente, existen muchos sistemas en línea (sistemas bancarios, de reservaciones aéreas y sistemas de bolsa) que se espera reaccionen en uno o dos segundos a un mensaje tecleado en la terminal. Sin embargo, en la mayoría de los sistemas de tiempo real, la computadora debe de reaccionar en *milisegundos* y a veces en *microsegundos* a los estímulos que recibe. Esto es característico de los siguientes tipos de sistemas:

- *Sistemas de control de procesos*: los sistemas computacionales que se utilizan para verificar y controlar refinerías, procesos químicos, molinos y operaciones de maquinado.
- *Sistemas de cajeros automáticos*: las "máquinas de efectivo" que muchos de nosotros usamos para hacer retiros sencillos del banco.
- *Sistemas de alta velocidad para adquisición de datos*: los sistemas computacionales que obtienen datos de telemetría a alta velocidad de satélites en órbita o las computadoras que capturan cantidades enormes de datos de experimentos de laboratorio.
- *Sistemas de guía de proyectiles*: los sistemas computacionales que deben rastrear la trayectoria de un proyectil y hacer ajustes continuos a la orientación y empuje de los propulsores.
- *Sistemas de conmutación telefónica*: sistemas computacionales que controlan la transmisión de voz y datos en miles de llamadas telefónicas, detectando los números marcados, condiciones de ocupado y todas las demás condiciones de una red telefónica típica.
- *Sistemas de vigilancia de pacientes*: sistemas computacionales que detectan los "signos vitales" de diversos pacientes (por ejemplo, temperatura y pulso) y que son capaces ya sea de ajustar el medicamento administrado o de hacer sonar la alarma si los signos vitales se mantienen fuera de ciertos límites predeterminados.

Además de la velocidad, existe otra característica que diferencia a los sistemas de tiempo real de los sistemas en línea: estos últimos suelen interactuar con las personas, mientras que los sistemas de tiempo real usualmente interactúan tanto con personas como con un *ambiente* que en generalmente es autónomo y a menudo hostil. De hecho, la principal preocupación del analista de sistemas en tiempo real es que, si la computadora no responde con la suficiente rapidez, el ambiente pudiera quedar fuera de control, los datos de entrada pudieran perderse sin remedio o un proyectil pudiera salirse de su trayectoria tanto que ya no fuera posible recuperarlo, o bien que un proceso de manufactura pudiera explotar*. En cambio, un sistema en línea que no responda con la suficientemente rapidez en general no hará más que volver impacientes y gruñones a los usuarios. Si tienen que esperar más de tres segundos la respuesta de un sistema en línea, las personas pueden "explotar" en sentido figurado, pero no en sentido literal. Esto se ilustra en la Figura 101.

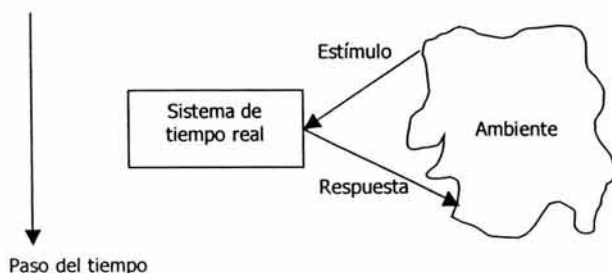


Figura 101. Un sistema de tiempo real.

* Uno de los ejemplos más interesantes en una situación de tiempo real es el de un equipo de trabajo cuya misión era unir una pequeña computadora a una bomba nuclear. Cuando se detonara la bomba (como parte de un programa de pruebas), la computadora dispondría tan sólo de unos cuantos microsegundos para capturar los tantos datos como fuera posible y transmitirlos a un sistema de computadoras remoto, antes de que se desintegraran el software y el hardware por la explosión. Esa sí que es una real exigencia del procesamiento en tiempo real.

Dada la preocupación con la respuesta instantánea a las entradas del sistema, un analista que trabaja en sistemas de tiempo real generalmente estará muy preocupado por el *comportamiento dependiente del tiempo* que el sistema tenga.

Desde un punto de vista de su puesta en práctica, los sistemas de tiempo real (así como algunos sistemas en línea) se caracterizan por lo siguiente:

- Simultáneamente se lleva a cabo el proceso de muchas actividades.
- Se asignan prioridades diferentes a diferentes procesos: algunos requieren servicio inmediato mientras que otros pueden aplazarse por periodos razonables.
- Se interrumpe una tarea antes de concluir, para comenzar otra de mayor prioridad.
- Existe gran comunicación entre tareas, especialmente dado que muchas tratan diferentes aspectos de un proceso general, como el control de un proceso de manufactura.
- Existe acceso simultáneo a datos comunes, tanto en memoria como en el almacenamiento secundario, por lo cual se requiere de un elaborado proceso de sincronización y "semáforos" para asegurar que los datos comunes no se corrompan.
- Existe un uso y asignación dinámicos de memoria RAM en el sistema computacional, dado que a menudo resulta poco económico (aún cuando la memoria sea barata) asignar suficiente memoria fija para manejar situaciones pico de alto volumen.

4.3.3.3. Sistema de Optimización en Línea/Tiempo Real.

Los componentes que debe integrar un sistema de optimización en línea/tiempo real de procesos de producción se presentan en la Figura 102.

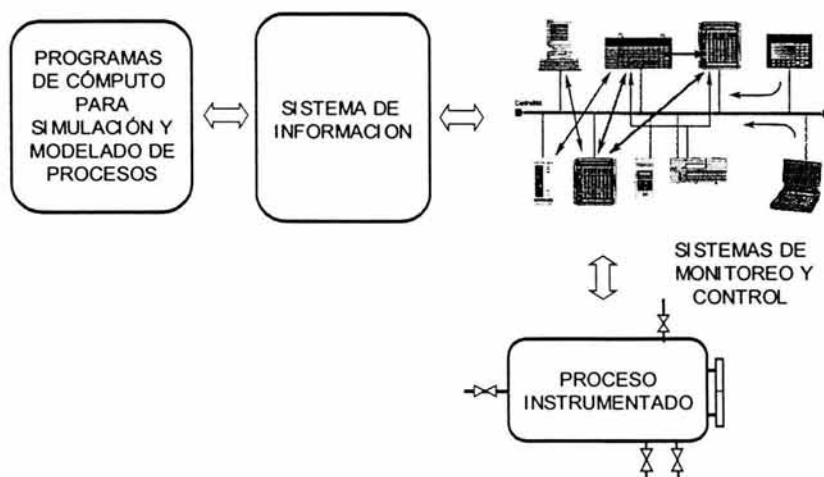


Figura 102. Componentes de un sistema de optimización en línea/tiempo real.

Requerimientos para la optimización en línea/tiempo real de los procesos de producción:

Un sistema de Optimización en Línea/Tiempo real debe comprender la siguiente infraestructura operativa.

a) Instrumentación.

La instrumentación debe ser:

- Específica para cada proceso que satisfaga los requerimientos del sistema de optimización en línea/tiempo real y que permita realizar el ajuste de los modelos.
- Redundante, para la reconciliación de datos en caso de ser necesario.

b) Sistemas de Monitoreo y Control.

- Los sistemas de monitoreo deben medir las variables definidas por los requerimientos del sistema de optimización en línea/tiempo real para el ajuste continuo del modelo.

Los sistemas de control deben:

- Poseer lazos de control definidos por los requerimientos del sistema de optimización en línea/tiempo real.
- Tener la capacidad de llevar al proceso al punto de operación determinado por el sistema de optimización en línea/tiempo real.

c) Sistema de manejo de información.

Los sistemas de manejo de información deben:

- Integrar la información de la base de datos del sistema de monitoreo y control del proceso con los programas de simulación y optimización.
- Las bases de datos deben ser accedidas mediante protocolos estandarizados.

d) Programas de cómputo para simulación y optimización en línea/tiempo real para los procesos de producción deben.

- Trabajar con protocolos de comunicación estandarizados.

En relación con la parte de simulación los programas de cómputo deben:

- Reproducir el comportamiento actual del problema físico bajo las condiciones de operación.
- Permitir el ajuste del modelo de forma automática.
- Tener conectividad con el programa de optimización, de manera que este último lo utilice dentro de su algoritmo de optimización.

En relación con la parte de optimización los programas de cómputo deben:

- Realizar un proceso de eliminación errores en los datos de campo erróneos.
- Realizar la reconciliación de datos.

- b). Detectar el estado estacionario.
- c) Permitir que el usuario programe las tareas a realizarse en tiempo real.
- d.) Realizar la búsqueda del punto óptimo de operación, mediante un algoritmo de optimización robusto.

4.3.4. Sistema Integral de Control y Manejo de Información.

El Sistema de Información para la Simulación y Optimización de Procesos de Producción de Hidrocarburos, forma parte modular de un Sistema de Control, denominado: Sistema de Control y Manejo de Información aplicable a la simulación de procesos de producción de hidrocarburos.

Este sistema de control funcional realiza el manejo y manipulación de la información adquirida del proceso, para realizar la optimización de procesos de producción de hidrocarburos en tiempo real por medio de software de simulación.

Al término de la implementación de algunos de los proyectos mencionados se tendría un Sistema de Información como el que se observa en la Figura 103.

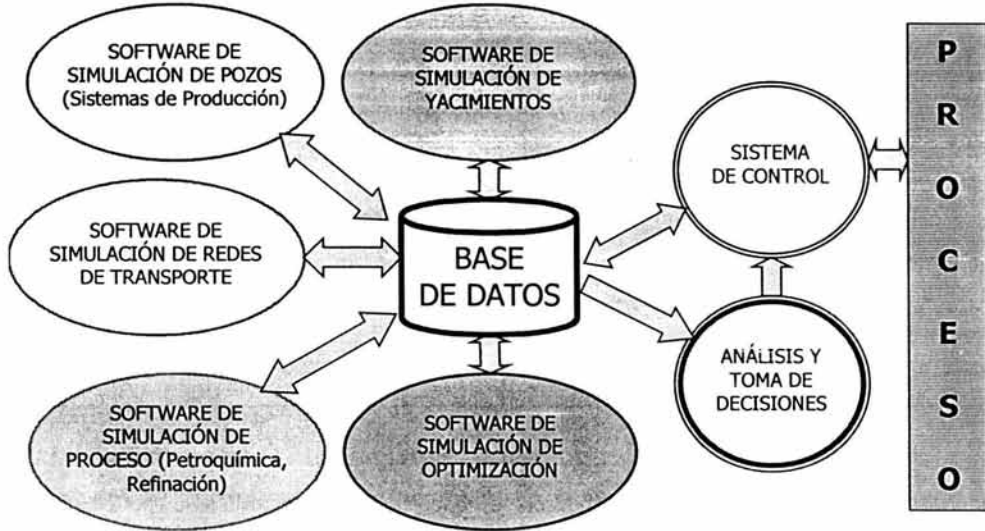


Figura 103. Proceso dinámico en línea.

CONCLUSIONES

La importancia de los sistemas de información se ha incrementado en México en los últimos años debido a los avances de la tecnología. Gracias a las herramientas que día con día se van perfeccionando, el desarrollo de sistemas está llegando a ser menos complejo y los ejecutivos se van dando cuenta de la necesidad cada vez mayor de la existencia de un sistema de información en la organización, ya que con éstos se obtienen grandes beneficios y se está a la altura de la competencia.

Por ello el Grupo de Optimización de Instalaciones de Producción perteneciente a la Dirección Ejecutiva de Exploración y Producción del IMP propuso un proyecto importante a PEMEX Exploración y Producción denominado "Asistencia Técnica para el Desarrollo de un Sistema Informático de Optimización Integral para los Procesos de Producción", con clave F. 33769, el cual se está llevando a cabo a largo plazo.

Este trabajo de tesis está aportando la estructura conceptual y directivas para el desarrollo del proyecto, en relación con la comunicación entre aplicaciones de diferentes compañías y a la creación de una base de datos única en donde se almacene la información requerida y generada por las herramientas de simulación.

Durante el desarrollo del Sistema de Información para la Simulación y Optimización de Procesos de Producción de Hidrocarburos se identificaron varias situaciones como:

- Falta de organización de la información.
- Inexistencia de un sistema de información de toma de decisiones en todas las áreas.
- Realización de simulaciones y optimizaciones de forma apartada y aislada.
- Falta de organización y actualización de la información.
- Obtención de resultados retrasados.

Estas situaciones sumadas a los problemas planteados al inicio de este trabajo, son los que se trataron de resolver desarrollando un sistema de información.

De acuerdo a lo observado y a las experiencias obtenidas en el tiempo de estancia en el IMP, elaborando el sistema se concluye que:

- El trabajo en equipo reduce enormemente el tiempo invertido en el desarrollo de un cualquier sistema de información.
- El trabajo conjunto de los desarrolladores con los usuarios es muy importante durante el análisis y diseño del sistema, así como en el mantenimiento, ya que finalmente el usuario es el que más conoce sus necesidades.
- La necesidad urgente de una aplicación y la falta de recursos para contrataciones de gente experta hace que ingenieros y administradores intenten desarrollarlas por sí mismos, lo cual no es muy recomendable, ya que pueden surgir problemas durante la puesta en marcha de la misma y el surgimiento de múltiples errores por falta de conocimiento del tema.
- Para los procesos de producción de hidrocarburos, es necesario disponer de información de proceso (es decir, de campo) de manera oportuna, complementándola con información de escritorio, para su análisis posterior a través de distintas herramientas y la obtención de resultados a tiempo, para así aplicar estas variables óptimas directamente al proceso y de manera rápida.

En relación a la hipótesis y objetivos planteados, así como al análisis de resultados y evaluación del sistema:

- Se cumplieron todos los objetivos planteados y la hipótesis se verificó, ya que todo se llevó a cabo, aunque invirtiendo un poco más de tiempo, y el sistema queda funcionando.
- Debido a la gran competencia existente en el mercado en cuanto a herramientas de simulación y optimización, la mayoría se desarrollan bajo plataforma Windows, lo que hizo más fácil la comunicación entre las herramientas aún siendo de diferentes empresas.

- Gracias al uso de estándares de comunicación el sistema queda abierto para poder agregar otras herramientas en el futuro y hacerlo más eficiente.
- La integración de los datos sistema se realizó de acuerdo a la propuesta planteada, aunque fue un poco tardado debido a la falta de organización de los mismos.
- Los errores encontrados durante las pruebas, surgieron debido a problemas en cuanto a las licencias del software y a problemas en cuanto al uso del equipo, ya que surgían necesidades de la Laptop y al cargarlo a otra computadora, la versión del software era diferentes y en otro idioma, además de que la resolución de los monitores cambiaba, por ello se recomienda seguir los requerimientos de instalación mencionados en el Capítulo 3. Todo esto sirvió como experiencia para tomar en cuenta otros factores durante la realización de un sistema de información.
- El software Microsoft Access y VBA está muy limitado, pero se utilizó debido a que eran las licencias de las herramientas requeridas que estaban disponibles al inicio del desarrollo del sistema.
- El sistema, finalmente cumple con los beneficios plantados, como ahorro de tiempo, energía y recursos. Además de que sirve como base para la planeación de nuevos sistemas de información, no solo para la toma de decisiones, sino de otros tipos, los cuales son muy necesarios en la organización.

En cuanto a mi aprendizaje personal, quiero comentar que una como alumna no se da cuenta de la diferencia que existe entre la teoría y la práctica. En el IMP, me percaté de la importancia que tiene la planeación desde las propuestas de los proyectos, ya que para que el instituto los acepte se requiere pasar por varias pruebas, presentaciones y trámites, no es sólo el hecho de saber que se necesita desarrollar un proyecto, sino justificarlo de la mejor manera posible.

Me agradó también el hecho de que no se utilice software pirata, ya que los alumnos estamos acostumbrados a ellos por falta de recursos financieros, pero en el IMP todas las aplicaciones tienen licencia, por las auditorías constantes que le hacen a la organización. Esto es parte de las normas establecidas en el instituto, donde también entra todo el protocolo que hay que seguir para cada uno de los trámites, lo cual deja ver la excelente administración, organización y orden que existe ahí.

Quiero hablar también sobre la realización de la tesis. Aprendí que para realizar un buen trabajo no sólo se requiere conocer el tema, sino también el entusiasmo para desarrollarla y determinación para concluirla satisfactoriamente.

Finalmente, y más relacionado al análisis del sistema de información para la simulación y optimización de procesos de producción de hidrocarburos, se observa que actualmente la tendencia que sigue la incorporación de nuevas tecnologías aplicadas a la producción de hidrocarburos es la Optimización en línea * de todo el proceso (explotación, producción y manejo de hidrocarburos), es decir, integrar y conectar en un mismo sistema a las herramientas de simulación con toda la infraestructura de monitoreo y control de parámetros del proceso, de manera que las condiciones de operación determinadas en las aplicaciones de simulación sean actualizadas en tiempo real por el sistema hacia todo el proceso, esto permite mejorar sustancialmente el rendimiento de los mismos al ejecutarse los cambios de manera automática.

La optimización y automatización de los procesos industriales constituye una herramienta que permite incrementar la productividad y la calidad de los productos, ahorro de energía, aumentar el grado de seguridad de los procesos, entre otros aspectos.

La modernización de las instalaciones de producción con sistemas de información automatizados con aplicación directa hacia el área de optimización, tiene como consecuencia el ahorro sustancial de energía y recursos de consumo, sobre todo cuando las operaciones de optimización y automatización son ejecutadas de manera integral. El contar con un plan integral para la organización de la información de todo el proceso de producción de hidrocarburos requerido por cada una de las herramientas de simulación, permitirá establecer una línea directa en la estrategia de modernización, automatización y optimización de proceso, así mismo, busca contribuir directa y continuamente a lograr el máximo valor técnico y económico de la Región y/o Activos.

* Ver el punto 4.3.3.3. perteneciente a las líneas abiertas de investigación.

GLOSARIO

Aceite: Líquido oleaginoso que se encuentra formado en la naturaleza, como el petróleo, o que se obtiene por destilación de ciertos minerales bituminosos o de la hulla, el lignito y la turba. Sustancia grasa, líquida a temperatura ordinaria, de mayor o menor viscosidad, no miscible con agua y de menor densidad que ella. Líquido viscoso que, por lo general, se deriva del petróleo, pero que también se puede obtener de animales y vegetales. Los aceites de hidrocarburos son productos viscosos de baja volatilidad que se utilizan ampliamente como lubricantes de maquinarias, para la elaboración de textiles y en la producción de medicamentos.

Agua: Líquido incoloro formado por la combinación de un volumen de oxígeno y dos de hidrógeno, inodora, insípida, en pequeña cantidad incolora y verdosa o azulada en grandes cantidades. Es el componente más abundante de la superficie terrestre y, más o menos puro, forma la lluvia, las fuentes, los ríos y los mares. Es parte constituyente de todos los organismos vivos. Aparece en compuestos naturales y como agua de cristalización en muchos cristales.

Aire: Gas que rodea la atmósfera terrestre. Fluido en estado gaseoso. Elemento vital de los seres humanos, animales y plantas. Se compone de nitrógeno, oxígeno, dióxido de carbono, vapor de agua, gases inertes, etc.

API (interfaz de programación de aplicaciones de Windows): Funciones, mensajes, estructuras de datos, tipos de datos e instrucciones que se pueden utilizar para crear aplicaciones que se ejecuten bajo Microsoft Windows 95 o posterior. Las partes de la interfaz API que se utilizan mayoritariamente son los elementos de código incluidos para llamar a funciones de la API desde Windows. Incluyen declaraciones de procedimientos (para las funciones de Windows), definiciones de tipos definidos por el usuario (para estructuras de datos pasadas a dichas funciones) y declaraciones de constantes (para valores pasados a, y devueltos desde, dichas funciones).

Apuntador: Elemento de información de un registro que contiene la dirección de localización de otro registro relacionado lógicamente con el primero.

Archivo: Conjunto de registros relacionados entre sí que se tratan como unidad.

Crudo: Dícese del mineral viscoso que una vez refinado proporciona el petróleo, el asfalto y otros productos.

Datos: Hechos; materia prima de la información.

Datawarehouse: Una colección estratégica de todo tipo de datos (combinación de diferentes bases de datos), la cual está específicamente modelada para apoyar los procesos de análisis y toma de decisiones en todos los niveles de una empresa. Es una gran base de datos multidimensional que no realiza transacciones.

DDE: Dynamic Data Exchange (DDE). Método proporcionado por Microsoft Windows para transferir datos entre aplicaciones compatibles. Este método es un protocolo que contiene un conjunto de mensajes y guías. Estos mensajes son enviados entre aplicaciones que comparten datos y usan memoria compartida para intercambiar datos entre estas aplicaciones. El protocolo DDE es usado por aplicaciones para transferir datos en un tiempo o continuos.

Estabilidad: Es la propiedad de un sistema de alcanzar un estado de equilibrio en un intervalo de tiempo.

Estado dinámico: El estado dinámico es la condición en la cuál las variables de un proceso varían en función del tiempo, normalmente son de carácter transitorio entre dos puntos de equilibrio para un proceso estable.

Estado estacionario o estable: El estado estacionario es la condición en la cuál las variables del proceso no cambian en función del tiempo, es una condición de equilibrio. Se considera un estado ideal de operación.

Fluido: Sustancia cuyas moléculas tienen entre sí poca o ninguna coherencia, capaz de desplazarse bajo mínima presión. Cuerpo que se puede mover en condiciones normales.

Gas: Fluido, como el aire, que tiende a expandirse indefinidamente y que se caracteriza por su pequeña densidad. Con esta expresión se denomina el gas natural, gas inflamable, gas tóxico o corrosivo.

Gas "Lift": El usado en el levantamiento artificial. Método para llevar a la superficie el petróleo o el agua mediante la inyección de gas en un pozo productor.

Hidrocarburo: Término general para el compuesto orgánico que contiene solamente carbono e hidrógeno en su molécula. A medida que aumenta el peso molecular el compuesto tiende a volverse un fluido pesado. Los de bajo peso molecular normalmente están en estado gaseoso; los de peso molecular alto son líquidos y los extremadamente pesados se convierten en sólidos.

HMI: Interfaz Humano Máquina. Es la zona de acción en donde el usuario se comunica con la computadora, el lugar en donde existe un diálogo entre ellos. El humano puede teclear en su computadora lo que necesita del sistema, y puede obtener todas las respuestas que el sistema pueda dar, además la computadora recibe también todas las respuestas que pudiera dar el humano ante las respuestas de la misma.

Industria del petróleo: Conjunto de operaciones materiales ejecutadas para la obtención, transformación o transporte del petróleo. Instalación destinada a tales operaciones. Suma o conjunto de las industrias de petróleo de todo un país o parte de él.

Interfaz: Superficie de contacto. Zona de comunicación o acción de un sistema sobre otro. Frontera compartida (por ejemplo, la frontera entre dos sistemas o dispositivos). Comúnmente conocida como "interfase" (del inglés interface). Área que limita o separa dos caras o superficies. Elemento de enlace.

Líquido: Dícese de todo cuerpo cuyas moléculas tienen tan poca cohesión que se adaptan a la forma de la cavidad que las contiene y tienden siempre a ponerse a nivel, como el agua, el vino, el azogue, etc. Sustancia en un estado de la materia intermedio entre los estados sólido y gaseoso.

Modelo: Es la representación matemática de un fenómeno o proceso real que reproduce de manera aproximada las condiciones de funcionamiento, y que permite predecir su comportamiento a diferentes condiciones de operación.

Molécula: En los fluidos, cada una de las partículas que se mueven con independencia de las restantes; y en los sólidos, agrupación de átomos ligados entre sí más fuertemente que con el resto de la masa. La partícula más pequeña que mantiene las propiedades químicas específicas de una sustancia.

OPC: OLE for Process Control (OPC). Es un nuevo diseño de tecnología que proporciona un puente entre las aplicaciones basadas en Windows y procesos de control de hardware. Este es un estándar abierto que permite un método consistente para acceder datos de campo de los dispositivos de planta. Este método mantiene el mismo tipo y fuente de datos. Por lo tanto, el usuario se encuentra libre para elegir el software y el hardware que cumplan con sus necesidades primarias, sin tener que considerar la disponibilidad de los drivers propietarios.

Optimalidad (punto o estado óptimo): Es la condición potencial de un sistema que dada la forma en que se relacionan sus variables (Funcionalidad), existe un valor externo para la variable de interés, de tal manera que es posible llevar el sistema a un estado tal que maximiza o minimiza dicha variable, por ejemplo, el margen de ganancia en la producción de un producto.

Optimización de un proceso: Procedimiento mediante el cual se establecen las mejores condiciones de operación para alcanzar un valor máximo, o un valor mínimo, de una o más variables.

Optimización en línea/tiempo real: Proceso de búsqueda de la mejor condición de funcionamiento de un sistema, servicio o producto contemplando la reducción de costos de operación y tiempo de ejecución. El término en línea implica un contacto continuo con el proceso a través del intercambio de información, en tanto que un sistema en tiempo real debe estar en contacto con el sistema físico, proporcionar la respuesta adecuada cumpliendo con las restricciones de tiempo que la dinámica del sistema demande. Los sistemas en línea no interactúan directamente con el mundo físico. La optimización en línea/tiempo real se refiere que en un sistema de optimización, no todas las tareas involucradas cumplen con las restricciones para tiempo real.

Petróleo: Mezcla de hidrocarburos producidos de manera natural por transformación de la materia orgánica y que se almacena en el subsuelo. Líquido oleaginoso e inflamable, que se extrae de lechos geológicos, continentales o marítimos. El hombre ha aprendido a sacarlo a la superficie, fraccionarlo y usarlo como materia energética. Mediante diversas operaciones de destilación y refinación se obtienen de él diversos productos para fines energéticos e industriales (gasolina, nafta, queroseno, gasóleo, etc.).

Petróleo crudo: Porción inmóvil de hidrocarburos que existe en la fase líquida en los yacimientos naturales y permanece líquida en condiciones atmosféricas de presión y temperatura. El petróleo crudo puede contener pequeñas cantidades de azufre, nitrógeno, oxígeno y cenizas. Los hidrocarburos presentes en el petróleo crudo son de tres tipos: parafínicos, nafténicos y aromáticos. El petróleo varía considerablemente en color y apariencia desde lechoso, marrón, verde, amarillo, hasta negro. A pesar de la variedad de colores con que suele encontrarse el petróleo, la gente tiene la tendencia a considerarlo negro.

Petroquímica: Conjunto de operaciones unitarias físicas y químicas requerido para transformar los hidrocarburos en productos de determinadas especificaciones, cuyas composiciones son polímeros o copolímeros.

Planta: Conjunto de instalaciones de producción en una determinada localización geográfica, que normalmente comparten almacenamiento, servicios y similares. Fábrica central de energía, instalación industrial; en algunos países hispanoamericanos, especialmente referido a una central eléctrica.

Plataforma: Combinación de sistema operativo y hardware.

PLC: Es un equipo electrónico programable que permite almacenar una secuencia de ordenes (programa) en su interior y ejecutarlo de forma cíclica con el fin de realizar una tarea. El PLC está diseñado para trabajar en ambientes industriales, ejecutar su programa de forma indefinida y es menos propenso a fallos o "cuelgues" que una computadora convencional. Además, su programación está más orientada al ámbito industrial.

Pozo: Hueco que se perfora en la superficie de la tierra con el fin de producir petróleo, gas, vapor o agua.

Pozo de Petróleo: Hueco que se hace para extraer petróleo de los yacimientos y producirlo, llevando el fluido hasta la superficie, donde se almacena.

Presión: Parámetro clave de diseño. Fuerza por unidad de superficie que ejerce un cuerpo (líquido o un gas) perpendicularmente a esa superficie.

Proceso: Conjunto de las fases sucesivas de un fenómeno natural o de una operación.

Producción: Acto o modo de producir. Productos que se sacan del subsuelo a través de los pozos de gas y/o petróleo. Lo que se obtiene de las plantas o de los procesos con fines rentables o con beneficios económicos.

Punto de operación o Setpoint (Punto de ajuste o de consigna): Es el conjunto de valores que toman las variables de un proceso que definen un estado de operación determinado. Valor o nivel en el cual se desea controlar la variable de un proceso.

Punto óptimo de operación: Donde se alcanza el funcionamiento eficiente de la instalación, equipo o planta, a costa de optimizar cada uno de los parámetros que intervienen en el proceso.

Reconciliación de datos: La reconciliación de datos es el mejoramiento de la calidad de las mediciones en un proceso. Puesto que los datos de proceso adquiridos están sujetos a errores aleatorios debido al ruido en las mediciones y errores debido a fallas en los equipos por descalibración, la reconciliación de datos puede retornar las mediciones a una forma más consistente.

Red Ethernet: Una Red Ethernet es un tipo particular de cableado de red más un grupo de especificaciones de señalización que cubren las capas 1 (Física) y 2 (Enlace) del modelo OSI. Es una red de banda base, o sea que provee un único canal de comunicación sobre el medio físico (cable), de forma que solo puede usarlo un dispositivo a la vez. Ethernet, como red de medio compartido, usa un mecanismo de acceso basado en la contención. Para ello usa un protocolo llamado CSMA/CD: Carrier Sense Multiple Access with Collision Detection.

Separación: Proceso que permite aislar los diversos componentes (crudo, agua y gas), con el fin de optimizar el procesamiento y la comercialización de algunos de ellos (crudo y gas). Fraccionamiento de cada uno de los componentes analizados.

Separador: Dispositivo mecánico usado en la separación de líquidos (agua, hidrocarburos, glicol, aminas, etc.) de los gases. Que separa dos o más fases. El término aplica con mayor frecuencia a los separadores de hidrocarburos. El más común es el bifásico horizontal y vertical, diseñado para separar el líquido (agua y petróleo) del gas natural. Siguen luego en complejidad los separadores trifásicos, capaces de entregar tres fases (agua, petróleo y gas), cada una de ellas independiente de las otras y el tetrafásico o trifásico con espuma que suele dejar un espacio disponible para manejar el espumaje.

Simulación: Es el proceso de reproducir las condiciones de operación de un sistema contemplando la relación entre los elementos que lo integran de manera que permita visualizar su comportamiento a diferentes condiciones (escenarios).

- Simulación es el proceso de diseñar un modelo de un sistema real y llevar a cabo experiencias con el mismo con la finalidad de comprender el comportamiento del sistema o de evaluar nuevas estrategias - dentro de los límites impuestos por un criterio o conjunto de ellos -, para el funcionamiento del sistema.

Simulador: Programa de computación que automatiza los cálculos para el diseño de equipos y estudio del proceso. Sirve para entrenar a los ingenieros y técnicos en los cálculos implícitos en múltiples diseños, incluyendo los separadores. Se organiza para efectuar los cálculos de un modo directo. Con él se pueden emular los diseños de los fabricantes de equipos. Representa eventos del mundo real, bien sea a través de modelos o de complicados procesos de animación. La simulación es altamente interactiva e involucra al usuario como elemento esencial en las situaciones de la vida real.

Simulador Comercial: Se refiere al "software" que se consigue en el mercado para simular procesos.

Simular: Reproducir, en condiciones de prueba, fenómenos de posible ocurrencia, sin necesidad de efectuar experimentos reales.

Sintonización: Proceso de ajuste de los parámetros que definen el comportamiento de una variable para estabilizarla dentro de los límites de operación y mantener el óptimo funcionamiento del proceso.

Sistema de información: Es un conjunto integrado de programas y equipos de cómputo cuyo propósito fundamental es obtener y proporcionar información con un objetivo definido.

Tag: Las variables de proceso se representan por *etiquetas* o *tags* que permiten la definición de cada variable en cuanto a su naturaleza continua (analógica) o discreta (binaria), la asociación de un nombre, el rango de valores a tomar, unidades de ingeniería y otras propiedades de utilidad para la monitorización como son el dispositivo de adquisición, alarmas, su registro, etc. La organización de todas las variables (adquiridas e internas) se hace en los entornos de monitorización en bases de datos.

TCP/IP: Es un protocolo DARPA que proporciona transmisión fiable de paquetes de datos sobre redes. El nombre TCP/IP proviene de dos protocolos importantes de la familia, el Transmission Control Protocol (TCP) y el Internet Protocol (IP). Todos juntos llegan a ser más de 100 protocolos diferentes definidos en este conjunto. El TCP/IP es la base del Internet que sirve para enlazar computadoras que utilizan diferentes sistemas operativos, incluyendo PC, minicomputadoras y computadoras centrales sobre redes de área local y área extensa.

Temperatura: Medida de la cantidad de calor que tiene la materia. Parámetro clave de diseño que, conjuntamente con el volumen y la presión, define el comportamiento de los fluidos.

Tiempo real: Término que describe los sistemas de proceso por computadora en línea, que reciben y procesan los datos con la suficiente rapidez para producir salidas que controlan, dirigen o afectan el resultado de una actividad o proceso.

Variable: Parámetro que interactúa para conocer cómo se afecta el comportamiento de un determinado hecho o fenómeno cuando cambian sus valores. Que cambia. Aplica a las condiciones cambiantes de un parámetro o hecho específico. Por ejemplo, condiciones variables de temperatura y presión, cuando estos dos parámetros cambian.

Variable de operación: Parámetro que afecta el comportamiento de la planta o del sistema.

Variable de proceso: Puede ser cualquier propiedad física cambiante del mismo. Esta puede ser medida por la observación de los cambios resultantes en una serie de variables físicas.

Viscosidad: Resistencia al deslizamiento ofrecida por los líquidos a causa de su cohesión y adhesión. Es la facilidad que tiene un fluido para fluir cuando se le aplica una fuerza externa.

Yacimiento: Con respecto al petróleo o al gas natural, se refiere a una acumulación de hidrocarburos en el subsuelo que puede estar en estado gaseoso, líquido o estratificado en diferentes fases. Normalmente se encuentra confinado en rocas porosas y permeables. Se le clasifica como yacimientos de gas natural, de petróleo y de condensados.

NOMENCLATURA

ADO: Objetos de Datos ActiveX (ActiveX Data Objects).

API: Interfaz de Programación de Aplicaciones (Application Programming Interface).

ASCII: Código Estándar Estadounidense para el Intercambio de Información (American National Standard Code for Information Interchange).

CIM: Fabricación Integrada por Computadora (Computer Integrated Manufacturing).

CLI: (Call-Level Interface).

COM: Modelo de Objetos Componentes (Component Object Model).

CORBA: Arquitectura de Recuperación de Objetos Comunes (Common Object Request Broker Architecture).

DAO: Objetos de Acceso a Datos (Data Access Objects).

DCE-RPC: Ambiente de Computación Distribuida y Llamadas a Procedimientos Remotos (The Distributed Computing Environment – DCE - Remote Procedure Call – RPC).

DCOM: Modelo de Objetos Componentes Distribuido (Distributed COM).

DCS: Sistemas de Control Distribuido.

DDE: Intercambio Dinámico de Datos (Dynamic Data Exchange).

DDEML: Librería de administración de intercambio de datos dinámicos. (Dynamic Data Exchange Management Library).

DLL: Biblioteca de Vínculos Dinámicos (Dynamic-Link Library).

EXE: Archivo Ejecutable.

GOIP: Grupo de Optimización de Instalaciones de Producción.

GUI: Interfaz de Usuario Gráfica.

HTTP: Protocolo de Transferencia de Hipertexto (Hypertext Transfer Protocol).

IED: Intercambio Electrónico de Datos.

IMP: Instituto Mexicano del Petróleo.

IPC: Comunicación entre Procesos (Inter-Process Communication).

ISAM: Método de Acceso Secuencial Indexado (Indexed Sequential Access Method).

ISO/IEC: ISO. International Organization for Standardization/IEC. International Electrotechnical Commission.

LAN: Red de Área Local (Local Area Network).

MDAC: Componentes de Acceso a Datos de Microsoft.

OCX: OLE Custom Controls.

ODBC: Conectividad de Bases de Datos Abierta (Open Database Connectivity).

OLAP: Procesamiento Analítico en Línea (On-Line Analytic Processing).

OLE: Incrustación y Vinculación de Objetos (Object Linking and Embedding).

OLE DB: Incrustación y Vinculación de Objetos de Bases de Datos.

OOP: Programación Orientada a Objetos (Object Oriented Programming).

OPC: OLE para Control de Procesos (OLE for Process Control).

PEMEX: Petróleos Mexicanos.

PEP: PEMEX Exploración y Producción.

PIPESIM: Modelos de Sistemas de Producción de Gas y Aceite Integrados (Integrated Oil & Gas Production Systems Models).

PLC: Controlador Lógico Programable (Programmable Logic Controller).

PRO/II: Simulación de Procesos de Refinería (Simulation of Refinery Processes).

SGBD: Sistema de Gestión de Bases de Datos.

SQL: Lenguaje de Consulta Estructurado (Structured Query Language).

VBA: Visual Basic para Aplicaciones.

XML: Extensible Markup Language.

BIBLIOGRAFÍA

1. Baker Jardine. PIPESIM 2000 – Open Link. Reference Manual, © Copyright 2000 Baker Jardine & Associates Limited. All Rights Reserved. United Kingdom, 27 pp.
2. Baker Jardine. PIPESIM 2000. PNSReaderCOM, © Copyright 2000 - 2001 Baker Jardine. All rights reserved. England, 12 pp.
3. Baker Jardine. PIPESIM 2000. User Guide, © Copyright 2000-01 Baker Jardine. All rights reserved. England, 219 pp.
4. Benavides Abajo, J.; Olaizola Bartolomé, J. M.; Rivero Cornelio, E. SQL para usuarios y programadores, Paraninfo, 4ª. Edición, España, 1998, 383 pp.
5. Callahan, Evan. Programación con Microsoft Access 2002. Macros y Visual Basic para aplicaciones, McGraw Hill, España, 2002, 418 pp.
6. Coffman, Gayle. SQL Server 7. Manual de Referencia, Osborne McGraw Hill, España, 1999, 909 pp.
7. De Miguel Castaño, Adoración; Piattini Velthius, Mario Gerardo. Concepción y Diseño de Bases de Datos. Del Modelo E/R al Modelo Relacional, Addison-Wesley Iberoamericana, S.A. E.U.A., 1993, 989 pp.
8. De Miguel Castaño, Adoración; Piattini Velthius Mario. Fundamentos y Modelos de Bases de Datos, Alfaomega Grupo Editor, Segunda Edición, España, 1999, 515 pp.
9. Dobson, Rick. Programación Avanzada con Microsoft Access Versión 2002, McGraw Hill, España, 2002, 997 pp.
10. Gil, Philip E.; Murray, Walter; Wright, Margeret H. Practical Optimization, Academic Press, Great Britain, 1995, 401 pp.
11. González, Alfons. SQL Server. Programación y administración, Alfaomega Ra-ma, México, 1999, 368 pp.
12. Houlette, Forrest. Fundamentos de SQL. Conocimientos esenciales para programadores, McGraw Hill, México, 2003, 422 pp.
13. Kendall y Kendall. Análisis y Diseño de Sistemas, Prentice Hall, México, 1991, 881 pp.
14. Law, Averill M.; Kelton, W. David. Simulation Modeling and Analysis, McGraw Hill, Tercera edición, USA, 2000, 760 pp.
15. Louis, Dirk. Delphi 5, Alfaomega – Marcombo, España, 2000, 856 pp.
16. Martínez, Marcías J.; Velásquez de Bermúdez, Belén. Diccionario de Gas Natural, Segunda Edición, ICONSA (Ingenieros Consultores y Asociados, C.A.), Venezuela, 2003, 383 pp.
17. Naylor, Thomas H.; Balintfy, Joseph L.; Burdick, Donald S.; Chu, Kong. Técnicas de simulación en computadoras, Quinta reimpresión, Limusa, México, 1982, 390 pp.
18. O'Brien, James A. Sistemas de Información Gerencial, Cuarta Edición, Irwin McGraw – Hill, Colombia, 2001, 700 pp.

19. Ramalho, José Antonio. SQL Server 7. Iniciación y Referencia, Mc-Graw Hill, Colombia, 2000, 580 pp.
20. Riordan, Rebeca M. Diseño de bases de datos relacionales con Access y SQL Server, McGraw Hill, España, 2000, 290 pp.
21. Ríos Insua, David; Ríos Insua, Sixto; Martín, Jacinto. Simulación. Métodos y Aplicaciones, RA-MA, España, 1997, 371 pp.
22. Sanders, Donald H. Informática. Presente y Futuro, McGraw Hill, Tercera Edición, México, 1990, 887 pp.
23. Senn, James A. Análisis y Diseño de Sistemas de Información, 2da. Edición, McGraw-Hill, México, 1994, 942 pp.
24. Simulation Sciences. Introduction to PRO/II OLE Automation Layer, Copyright © 1998 Simulation Sciences Inc. All Rights Reserved. Revision 4.0. USA, 62 pp.
25. Simulation Sciences. PROvision Database Server OLE Automation Reference. Pro/ii OLE Automation Reference, Copyright © 1999 Simulation Sciences Inc. All Rights Reserved, USA, 47 pp.
26. Simulation Sciences. Pro/ii Object Reference Manual, Copyright © 1998 Simulation Sciences Inc. All Rights Reserved, USA, 59 pp.
27. Tsai, Alice Y. H. Sistemas de bases de datos: Administración y uso, Prentice-Hall Hispanoamericana, S.A., México, 1990, 607 pp.
28. Villanueva Espinosa, Francisco Javier. Protocolos de Comunicación aplicados a la Instrumentación de Campo y su Impacto en los Procesos de Producción de Hidrocarburos, Tesis Individual para obtener el Título de: Ingeniero en Control y Automatización. Instituto Politécnico Nacional. Escuela Superior de Ingeniería Mecánica y Eléctrica. Unidad profesional. Adolfo López Mateos. México, D.F., Noviembre de 1998, 235 pp.
29. Whitten, Jeffrey L.; Bentley, Loonie D.; Barlow, Victor M. Análisis y Diseño de Sistemas de Información, Tercera edición. Edit. Irwin, España, 1996, 907 pp.
30. Yourdon, Edward. Análisis Estructurado Moderno, Prentice Hall, México, 1993, 735 pp.

SITIOS DE INTERNET CONSULTADOS:

1. www.dcc.uchile.cl/~rbaeza/libro/liendo.pdf
2. <http://www.ambac.org.mx/publicaciones/V3N4art1.html>
3. http://www.gestiondelconocimiento.com/conceptos_diferenciaentredato.htm
4. http://www.palgrave.com/skills4study/html/subject_areas/science/science-technology.htm
5. http://www.wdvl.com/Authoring/DB/Intro/what_is_database.html
6. <http://www.acm.org/crossroads/espanol/xrds7-3/intro73.html>
7. <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/winui/WinUI/WindowsUserInterface/DataExchange/DynamicDataExchangeManagementLibrary/AbouttheDDEML/BasicConcepts.asp>
8. http://itco.web.cern.ch/itco/Projects-Services/JCOP/SubProjects/TechnologySurvey/report/phase2/phase2_tasks.pdf
9. <http://www.microsoft.com/com/wpaper/default.asp>

10. <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/winui/WinUI/WindowsUserInterface/DataExchange/DynamicDataExchange/AboutDynamicDataExchange.asp>
11. <http://www.microsoft.com/com/tech/ActiveX.asp>
12. <http://www.bakerjardine.com>
13. <http://pubs.acs.org/hotartcl/cenear/950327/art10.html>
14. <http://www.imp.mx>
15. <http://www.sis.slb.com/>
16. <http://www.hyprotech.com/corp/default.asp>
17. <http://www.aspentech.com/corporate/index.cfm>
18. http://www.epsedin.co.uk/EPS_about/aboutEPS_home.htm
19. <http://www.scandpower.no/?ItemId=1102>
20. http://www.olga2000.com/main_software.html
21. <http://www.mathworks.com/company/>
22. http://www.rockwellautomation.com/about_us/index.html
23. <http://www.gams.com/>
24. <http://www.petex.com/>
25. <http://www.simsci.com/corporate/>
26. <http://www.dcnr.state.pa.us/topogeo/wis/wis.htm>
27. <http://oil-gas.state.co.us/>
28. <http://www.aldea.com/guides/ag/b400.html>
29. <http://bogc.dnrc.state.mt.us/jdpIntro.htm>
30. http://www.thermo.com/eThermo/CMA/PDFs/Articles/articlesFile_10363.pdf

ANEXO A

ANEXO A

BASE DE DATOS: BD1.mdb

Tablas pertenecientes a la base de datos:

1. Caracteristicas_de_Ductos
2. Column_Unit_Operation
3. Component_Data
4. Compressor_Unit_Operation
5. Edicion_Modelo_Pozo_Pipesim2000
6. Edicion_Modelo_PROII
7. Edicion_Modelo_Red_Pipesim2000
8. Especificaciones_de_Tuberia
9. Expander_Unit_Operation
10. Flash_Unit_Operation
11. Lectura_Modelo_Pozo_Pipesim2000
12. Lectura_Modelo_PROII
13. Lectura_Modelo_Red_Pipesim2000
14. Mixer_Unit_Operation
15. Pipe_Unit_Operation
16. Pump_Unit_Operation
17. Reporte_Modelo_Pozo_Pipesim2000
18. Reporte_Modelo_Red_Pipesim2000
19. SHExchanger_Unit_Operation
20. Splitter_Unit_Operation
21. Stream_Data
22. Valve_Unit_Operation
23. Variables_de_Pozo

Diccionario de datos:

1. Caracteristicas_de_Ductos

NOMBRE DEL CAMPO	TIPO DE DATO / LONGITUD	DESCRIPCION	MÓDULO QUE LA UTILIZA
Tramo	Texto (50)	Nombre del tramo del ducto.	Características de Ductos.
Gasto	Númérico (Doble)	Gasto en barriles por día.	Características de Ductos.
DI	Númérico (Doble)	Diámetro interno en pulgadas.	Características de Ductos.
Longitud	Númérico (Doble)	Longitud en metros.	Características de Ductos.
Presion_Entrada	Númérico (Doble)	Presión de entrada en psig.	Características de Ductos.
Presion_Salida	Númérico (Doble)	Presión de salida en psig.	Características de Ductos.

Clave principal: Gasto

2. Column_Unit_Operation

NOMBRE DEL CAMPO	TIPO DE DATO / LONGITUD	DESCRIPCION	MÓDULO QUE LA UTILIZA
Id	Autonumérico	Clave de identificación.	Editar Modelo PROII.
Nombre_Modelo	Texto (50)	Nombre del modelo abierto.	Editar Modelo PROII.
Tipo_Objeto	Texto (50)	Tipo del objeto consultado.	Editar Modelo PROII.
Nombre	Texto (50)	Nombre del objeto consultado.	Editar Modelo PROII.
Number_Feed_Streams	Númérico (Entero largo)	Número de corrientes de	Editar Modelo PROII.

		alimentación.	
Number_Product_Streams	Númérico (Entero largo)	Número de corrientes de producto.	Editar Modelo PROII.
Product_Type	Texto (50)	Tipo de producto.	Editar Modelo PROII.
Pressure_Value	Texto (50)	Valor de la presión.	Editar Modelo PROII.
Pressure_Top	Númérico (Doble)	Presión más alta.	Editar Modelo PROII.
Column_Pressure_Drop	Númérico (Doble)	Presión de caída en la columna.	Editar Modelo PROII.
PerTray_Pressure_Drop	Númérico (Doble)	Presión de caída por bandeja.	Editar Modelo PROII.
Specified_Value	Númérico (Doble)	Valor especificado.	Editar Modelo PROII.
Calculated_Value	Númérico (Doble)	Valor calculado.	Editar Modelo PROII.
Absolute_Tolerance	Númérico (Doble)	Tolerancia absoluta.	Editar Modelo PROII.
Relative_Tolerance	Númérico (Doble)	Tolerancia relativa.	Editar Modelo PROII.
Primary_Units_Measure	Texto (50)	Medida de unidades primarias.	Editar Modelo PROII.
Primary_Basis	Númérico (Entero largo)	Base de la sección primaria.	Editar Modelo PROII.
Primary_Wet_Dry	Númérico (Entero largo)	¿Mojado o seco en la sección primaria.?	Editar Modelo PROII.
Reference_Units_Measure	Texto (50)	Medida de unidades de referencia.	Editar Modelo PROII.
Reference_Basis	Númérico (Entero largo)	Base en la sección referencia.	Editar Modelo PROII.
Reference_Wet_Dry	Númérico (Entero largo)	¿Mojado o seco en la sección de referencia.?	Editar Modelo PROII.
Number_Pumparounds	Texto (50)	Número de bombas alrededor.	Editar Modelo PROII.
Temperature	Texto (50)	Temperatura de la bomba.	Editar Modelo PROII.
Liquid_fraction_basis	Texto (50)	Base de la fracción líquida.	Editar Modelo PROII.
Molar_Rates	Texto (50)	Gastos molares de la bomba.	Editar Modelo PROII.
Pressures	Texto (50)	Presiones de la bomba.	Editar Modelo PROII.
Temperatures	Texto (50)	Temperaturas de la bomba.	Editar Modelo PROII.
Liquid_fractions	Texto (50)	Fraciones líquidas.	Editar Modelo PROII.
Temperature_reference	Texto (50)	Referencia de temperatura.	Editar Modelo PROII.
In_Out	Texto (50)	Si es dato de entrada o de salida.	Editar Modelo PROII.
Fecha_Hora	Texto (50)	Fecha y hora de la captura.	Editar Modelo PROII.

3. Component_Data

NOMBRE DEL CAMPO	TIPO DE DATO / LONGITUD	DESCRIPCIÓN	MÓDULO QUE LA UTILIZA
Id	Autonumérico	Clave de identificación.	Editar Modelo PROII
Nombre_Modelo	Texto (50)	Nombre del modelo abierto.	Editar Modelo PROII
Tipo_Objeto	Texto (50)	Tipo del objeto consultado.	Editar Modelo PROII
Nombre	Texto (50)	Nombre del objeto consultado.	Editar Modelo PROII
Molecular_weight	Texto (50)	Peso molecular.	Editar Modelo PROII
Specific_gravity	Texto (50)	Gravedad específica.	Editar Modelo PROII
NBP	Texto (50)	Punto normal en el que hierve.	Editar Modelo PROII
Critical_temperature	Texto (50)	Temperatura crítica.	Editar Modelo PROII
Critical_pressure	Texto (50)	Presión crítica.	Editar Modelo PROII
Critical_volume	Texto (50)	Volumen crítico.	Editar Modelo PROII
Crit_Compress_Factor	Texto (50)	Factor de compresión crítico.	Editar Modelo PROII
Component_Type	Texto (50)	Tipo de componente.	Editar Modelo PROII
Phase	Texto (50)	Fase.	Editar Modelo PROII
Normal_melting_point	Texto (50)	Punto normal de fundición.	Editar Modelo PROII
Acentric_factor	Texto (50)	Factor acéntrico.	Editar Modelo PROII
Liquid_Molar_Volume	Texto (50)	Volumen líquido molar.	Editar Modelo PROII
Carbon_number	Texto (50)	Número de carbonos.	Editar Modelo PROII
In_Out	Texto (50)	Si es dato de entrada o de salida.	Editar Modelo PROII
Fecha_Hora	Texto (50)	Fecha y hora de la captura.	Editar Modelo PROII

4. Compressor_Unit_Operation

NOMBRE DEL CAMPO	TIPO DE DATO / LONGITUD	DESCRIPCION	MÓDULO QUE LA UTILIZA
Id	Autonumérico	Clave de Identificación.	Editar Modelo PROII
Nombre_Modelo	Texto (50)	Nombre del modelo abierto.	Editar Modelo PROII
Tipo_Objeto	Texto (50)	Tipo del objeto consultado.	Editar Modelo PROII
Nombre	Texto (50)	Nombre del objeto consultado.	Editar Modelo PROII
Pressure	Númérico (Doble)	Presión.	Editar Modelo PROII
Pressure_gain	Númérico (Doble)	Presión adquirida.	Editar Modelo PROII
Pressure_ratio	Númérico (Doble)	Presión de proporción.	Editar Modelo PROII
Inlet_pressure	Númérico (Doble)	Presión de entrada.	Editar Modelo PROII
Design_pressure	Númérico (Doble)	Presión de diseño.	Editar Modelo PROII
Outlet_Temperature	Númérico (Doble)	Presión de salida.	Editar Modelo PROII
Estimated_Outlet_Temp	Númérico (Doble)	Temperatura de salida estimada.	Editar Modelo PROII
Temperature	Númérico (Doble)	Temperatura.	Editar Modelo PROII
In_Out	Texto (50)	Si es dato de entrada o de salida.	Editar Modelo PROII
Fecha_Hora	Texto (50)	Fecha y hora de la captura.	Editar Modelo PROII

5. Edicion_Modelo_Pozo_Pipesim2000

NOMBRE DEL CAMPO	TIPO DE DATO / LONGITUD	DESCRIPCION	MÓDULO QUE LA UTILIZA
Id	Autonumérico	Clave de identificación.	Editar Modelo Red – Pozo.
nom_Modelo	Texto (50)	Nombre del modelo abierto.	Editar Modelo Red – Pozo.
nom_Objeto	Texto (50)	Nombre del objeto consultado.	Editar Modelo Red – Pozo.
tipo_Objeto	Texto (50)	Tipo del objeto consultado.	Editar Modelo Red – Pozo.
Tipo_Operación	Texto (50)	Tipo de operación a realizar.	Editar Modelo Red – Pozo.
Pressure(psia)	Númérico (Doble)	Presión en psia.	Editar Modelo Red – Pozo.
Temperature(C)	Númérico (Doble)	Temperatura en grados C.	Editar Modelo Red – Pozo.
Fecha_Hora	Fecha/Hora	Fecha y hora de la captura.	Editar Modelo Red – Pozo.
Clave principal: nom_Modelo-nom_Objeto-Fecha_Hora			

6. Edicion_Modelo_PROII

NOMBRE DEL CAMPO	TIPO DE DATO / LONGITUD	DESCRIPCIÓN	MÓDULO QUE LA UTILIZA
ID	Autonumérico	Clave de identificación.	Editar_Modelo_PROII.
Nombre_Modelo	Texto (50)	Nombre del modelo abierto.	Editar_Modelo_PROII.
Tipo_Objeto	Texto (50)	Tipo del objeto consultado.	Editar_Modelo_PROII.
Nombre	Texto (50)	Nombre del objeto consultado.	Editar_Modelo_PROII.
In_Out	Texto (50)	Si el dato es de entrada o de salida.	Editar_Modelo_PROII.
Pressure	Númérico (Doble)	Presión.	Editar_Modelo_PROII.
Temperature	Númérico (Doble)	Temperatura.	Editar_Modelo_PROII.
Pressure_drop	Númérico (Doble)	Presión de caída.	Editar_Modelo_PROII.
Pressure_gain	Númérico (Doble)	Presión adquirida.	Editar_Modelo_PROII.
Pressure_ratio	Númérico (Doble)	Presión de proporción.	Editar_Modelo_PROII.
Inlet_pressure	Númérico (Doble)	Presión de entrada.	Editar_Modelo_PROII.
Design_pressure	Númérico (Doble)	Presión de diseño.	Editar_Modelo_PROII.
Estimated_Outlet_Temp	Númérico (Doble)	Temperatura de salida estimada.	Editar_Modelo_PROII.
Outlet_Temperature	Númérico (Doble)	Temperatura de salida.	Editar_Modelo_PROII.
Outlet_Pressure	Númérico (Doble)	Presión de salida.	Editar_Modelo_PROII.
Inside_Diameter	Númérico (Doble)	Diámetro interno.	Editar_Modelo_PROII.
NPS_Size	Númérico (Doble)	Tamaño nominal de la tubería.	Editar_Modelo_PROII.
NPS_Schedule	Númérico (Entero largo)	Programa nominal de la tubería.	Editar_Modelo_PROII.
Pipe_Length	Númérico (Doble)	Longitud de la tubería.	Editar_Modelo_PROII.

Num_Segments	Númérico (Doble)	Número de segmentos.	Editar_Modelo_PROII.
Elevation_gain	Númérico (Doble)	Elevación adquirida.	Editar_Modelo_PROII.
Flow_efficiency	Númérico (Doble)	Eficiencia de flujo.	Editar_Modelo_PROII.
Hot_Pressure_Drop	Númérico (Doble)	Presión de caída caliente.	Editar_Modelo_PROII.
Cold_Pressure_Drop	Númérico (Doble)	Presión de caída fría.	Editar_Modelo_PROII.
SFeed_Stream_Hot	Númérico (Entero largo)	Corriente de alimentación caliente de inicio.	Editar_Modelo_PROII.
SFeed_Stream_Cold	Númérico (Entero largo)	Corriente de alimentación fría de inicio.	Editar_Modelo_PROII.
EFeed_Stream_Hot	Númérico (Entero largo)	Corriente de alimentación caliente final.	Editar_Modelo_PROII.
EFeed_Stream_Cold	Númérico (Entero largo)	Corriente de alimentación fría final.	Editar_Modelo_PROII.
SProduct_Stream_Hot	Númérico (Entero largo)	Corriente de producto caliente de inicio.	Editar_Modelo_PROII.
SProduct_Stream_Cold	Númérico (Entero largo)	Corriente de producto fría de inicio.	Editar_Modelo_PROII.
EProduct_Stream_Hot	Númérico (Entero largo)	Corriente de producto caliente final.	Editar_Modelo_PROII.
EProduct_Stream_Cold	Númérico (Entero largo)	Corriente de producto fría final.	Editar_Modelo_PROII.
Number_Feed_Streams	Númérico (Entero largo)	Número de corrientes de alimentación.	Editar_Modelo_PROII.
Number_Product_Streams	Númérico (Entero largo)	Número de productos de alimentación.	Editar_Modelo_PROII.
Product_Type	Texto (50)	Tipo de producto.	Editar_Modelo_PROII.
Pressure_Value	Texto (50)	Valor de la presión.	Editar_Modelo_PROII.
Pressure_Top	Númérico (Doble)	Presión alta.	Editar_Modelo_PROII.
Column_Pressure_Drop	Númérico (Doble)	Presión de caída en la columna.	Editar_Modelo_PROII.
PerTray_Pressure_Drop	Númérico (Doble)	Presión de caída por bandeja.	Editar_Modelo_PROII.
Name_Description	Texto (50)	Descripción del nombre.	Editar_Modelo_PROII.
Phase	Númérico (Entero largo)	Fase.	Editar_Modelo_PROII.
Liq_Fraction_Flag	Númérico (Entero largo)	Bandera de la fracción líquida.	Editar_Modelo_PROII.
Liq_FracBasis_Flag	Númérico (Entero largo)	Bandera de la base de la fracción líquida.	Editar_Modelo_PROII.
Liq_Fraction	Númérico (Doble)	Fracción líquida.	Editar_Modelo_PROII.
Rate	Númérico (Doble)	Gasto.	Editar_Modelo_PROII.
Molecular_weight	Númérico (Doble)	Peso molecular.	Editar_Modelo_PROII.
Specific_gravity	Númérico (Doble)	Gravedad específica.	Editar_Modelo_PROII.
NBP	Númérico (Doble)	Punto normal en que hierve.	Editar_Modelo_PROII.
Critical_temperature	Númérico (Doble)	Temperatura crítica.	Editar_Modelo_PROII.
Critical_pressure	Númérico (Doble)	Presión crítica.	Editar_Modelo_PROII.
Critical_volume	Númérico (Doble)	Volumen crítico.	Editar_Modelo_PROII.
Crit_Compress_Factor	Númérico (Doble)	Factor de compresión crítico.	Editar_Modelo_PROII.
Fecha_Hora	Texto (50)	Fecha y hora de la captura.	Editar_Modelo_PROII.
Clave principal: Nombre_Modelo-Nombre-Fecha_Hora			

7. Edicion_Modelo_Red_Pipesim2000

NOMBRE DEL CAMPO	TIPO DE DATO / LONGITUD	DESCRIPCION	MÓDULO QUE LA UTILIZA
Id	Autonumérico	Clave de identificación.	Editar Modelo Red - Pozo.
Nom_Modelo	Texto (50)	Nombre del modelo abierto.	Editar Modelo Red
Nom_Objeto	Texto (50)	Nombre del objeto consultado.	Editar Modelo Red
tipo_Objeto	Texto (50)	Tipo del objeto consultado.	Editar Modelo Red
Pressure(psia)	Númérico (Doble)	Presión en psia.	Editar Modelo Red
Temperature(C)	Númérico (Doble)	Temperatura en grados C.	Editar Modelo Red
LiquidRate(STB/D)	Númérico (Doble)	Gasto de líquido en STB/D.	Editar Modelo Red

GasRate(mmscf/d)	N Numérico (Doble)	Gasto de gas en mmscf/d.	Editar Modelo Red
MassRate(lb/s)	N Numérico (Doble)	Gasto de masa en lb/s.	Editar Modelo Red
Fecha_Hora	Fecha/Hora	Fecha y hora de la captura.	Editar Modelo Red
Clave principal: Nom_Modelo-Nom_Objeto-Fecha_Hora			

8. Especificaciones_de_Tuberia

NOMBRE DEL CAMPO	TIPO DE DATO / LONGITUD	DESCRIPCION	MÓDULO QUE LA UTILIZA
DI	N Numérico (Doble)	Diámetro Interno en Pulgadas.	Especificaciones de Tubería.
Tipo	T Texto (50)	Tipo de tubería.	Especificaciones de Tubería.
F	S/No	¿Flota en el agua? Si ó No.	Especificaciones de Tubería.
Espesor_Pulgadas	N Numérico (Doble)	Espesor de la tubería.	Especificaciones de Tubería.
Peso_Lbs	N Numérico (Doble)	Peso de la tubería en Libras/Pie.	Especificaciones de Tubería.
Peso_Ton	N Numérico (Doble)	Peso en Tonelada/Milla.	Especificaciones de Tubería.
Area_Flujo	N Numérico (Doble)	Área del flujo en (Pie) ² .	Especificaciones de Tubería.
Presion_Permissible	N Numérico (Doble)	Presión Permissible en la tubería.	Especificaciones de Tubería.
Clave principal: DI			

9. Expander_Unit_Operation

NOMBRE DEL CAMPO	TIPO DE DATO / LONGITUD	DESCRIPCION	MÓDULO QUE LA UTILIZA
Id	A Autonumérico	Clave de identificación.	Editar Modelo PROII.
Nombre_Modelo	T Texto (50)	Nombre del modelo abierto.	Editar Modelo PROII.
Tipo_Objeto	T Texto (50)	Tipo del objeto consultado.	Editar Modelo PROII.
Nombre	T Texto (50)	Nombre del objeto consultado.	Editar Modelo PROII.
Pressure	N Numérico (Doble)	Presión.	Editar Modelo PROII.
Pressure_drop	N Numérico (Doble)	Presión de caída.	Editar Modelo PROII.
Pressure_ratio	N Numérico (Doble)	Presión de proporción.	Editar Modelo PROII.
Inlet_pressure	N Numérico (Doble)	Presión de entrada.	Editar Modelo PROII.
Design_pressure	N Numérico (Doble)	Presión de diseño.	Editar Modelo PROII.
Estimated_Outlet_Temp	N Numérico (Doble)	Temperatura de salida estimada.	Editar Modelo PROII.
Outlet_Temperature	N Numérico (Doble)	Temperatura de salida.	Editar Modelo PROII.
Outlet_Pressure	N Numérico (Doble)	Presión de salida.	Editar Modelo PROII.
In_Out	T Texto (50)	Si el dato es de entrada o de salida.	Editar Modelo PROII.
Fecha_Hora	T Texto (50)	Fecha y hora de la captura.	Editar Modelo PROII.

10. Flash_Unit_Operation

NOMBRE DEL CAMPO	TIPO DE DATO / LONGITUD	DESCRIPCION	MÓDULO QUE LA UTILIZA
Id	A Autonumérico	Clave de identificación.	Editar Modelo PROII.
Nombre_Modelo	T Texto (50)	Nombre del modelo abierto.	Editar Modelo PROII.
Tipo_Objeto	T Texto (50)	Tipo del objeto consultado.	Editar Modelo PROII.
Nombre	T Texto (50)	Nombre del objeto consultado.	Editar Modelo PROII.
Pressure	N Numérico (Doble)	Presión.	Editar Modelo PROII.
Pressure_drop	N Numérico (Doble)	Presión de caída.	Editar Modelo PROII.
Temperature	N Numérico (Doble)	Temperatura.	Editar Modelo PROII.
In_Out	T Texto (50)	Si el dato es de entrada o de salida.	Editar Modelo PROII.
Fecha_Hora	T Texto (50)	Fecha y hora de la captura.	Editar Modelo PROII.

11. Lectura_Modelo_Pozo_Pipesim2000

NOMBRE DEL CAMPO	TIPO DE DATO / LONGITUD	DESCRIPCION	MÓDULO QUE LA UTILIZA
Numero	Autonumérico	Clave de identificación.	Leer Modelo Pozo.
nom_Modelo	Texto (80)	Nombre del modelo abierto.	Leer Modelo Pozo.
Upstream_Most_Object	Texto (50)	Objeto con mayor corriente arriba.	Leer Modelo Pozo.
nom_Objeto	Texto (50)	Nombre del objeto consultado.	Leer Modelo Pozo.
Tipo_Objeto	Texto (50)	Tipo del objeto consultado.	Leer Modelo Pozo.
Tubing_ArtificialLift	Texto (50)	Tubería Artificial Lift.	Leer Modelo Pozo.
Pressure(psia)	Numérico (Doble)	Presión en psia.	Leer Modelo Pozo.
Temperature(C)	Numérico (Doble)	Temperatura en C.	Leer Modelo Pozo.
LiquidRate(STB/D)	Numérico (Doble)	Gasto de Líquido.	Leer Modelo Pozo.
GasRate(mmscf/d)	Numérico (Doble)	Gasto de Gas.	Leer Modelo Pozo.
MassRate(lb/s)	Numérico (Doble)	Gasto de Masa.	Leer Modelo Pozo.
Choke_BeanSize	Numérico (Doble)	Tamaño del estrangulador.	Leer Modelo Pozo.
Tubing_TubModel	Texto (50)	Modelo de la tubería (tubería).	Leer Modelo Pozo.
Tubing_SurfaceTVD	Texto (50)	Superficie de la tubería.	Leer Modelo Pozo.
Tubing_SurfaceTemperature	Numérico (Doble)	Temperatura de la superficie.	Leer Modelo Pozo.
Tubing_KickoffTVD	Texto (50)	TVD inicial de la tubería.	Leer Modelo Pozo.
Tubing_KickoffAngle	Texto (50)	Ángulo inicial de la tubería.	Leer Modelo Pozo.
Tubing_PerfsTVD	Texto (50)	TVD adquirido de la tubería.	Leer Modelo Pozo.
Tubing_PerfsMD	Texto (50)	Profundidad adquirida.	Leer Modelo Pozo.
Tubing_PerfsTemp	Numérico (Doble)	Temperatura adquirida.	Leer Modelo Pozo.
Separator_MeasureDepth	Numérico (Doble)	Profundidad del separador.	Leer Modelo Pozo.
Separator_Efficiency	Numérico (Doble)	Eficiencia del separador.	Leer Modelo Pozo.
Separator_Type	Texto (50)	Tipo de separador.	Leer Modelo Pozo.
Tubing_Choke_MD	Numérico (Doble)	Profundidad de estrangulador.	Leer Modelo Pozo.
Tubing_Choke_BS	Numérico (Doble)	Tamaño estrangulador (tubo).	Leer Modelo Pozo.
Tubing_Pump_MD	Numérico (Doble)	Profundidad de la bomba.	Leer Modelo Pozo.
Tubing_Pump_ESPSt	Texto (50)	ESPSt de la bomba (tubería).	Leer Modelo Pozo.
Tubing_Pump_ESPSP	Texto (50)	ESPSP de la bomba (tubería).	Leer Modelo Pozo.
Tubing_Injector_MD	Numérico (Doble)	Profundidad Injector (tubería).	Leer Modelo Pozo.
Tubing_Injector_GLR	Numérico (Doble)	GLR del Injector (tubería).	Leer Modelo Pozo.
Tubing_Injector_GLST	Numérico (Doble)	GLST del Injector (tubería).	Leer Modelo Pozo.
Tubing_Injector_GLSG	Texto (50)	GLSG del Injector (tubería).	Leer Modelo Pozo.
Tubing_Injector_GLPD	Numérico (Doble)	GLPD del Injector (tubería).	Leer Modelo Pozo.
Tubing_Injector_GLSP	Numérico (Doble)	GLSP del Injector (tubería).	Leer Modelo Pozo.
Tubing_Separator_MD	Numérico (Doble)	Profundidad separador (tubo).	Leer Modelo Pozo.
Tubing_Separator_Eff	Numérico (Doble)	Eficiencia separador (tubería).	Leer Modelo Pozo.
Tubing_Separator_Type	Texto (50)	Tipo de separador (tubería).	Leer Modelo Pozo.
Tubing_SSSV_MD	Numérico (Doble)	Profundidad SSSV (tubería).	Leer Modelo Pozo.
Tubing_SSSV_BID	Texto (50)	BID SSSV de la tubería.	Leer Modelo Pozo.
Tubing_GLValve_MD	Numérico (Doble)	Profundidad de GLValve (tubo)	Leer Modelo Pozo.
Tubing_GLValve_VPS	Numérico (Doble)	VPS de GLValve (tubería).	Leer Modelo Pozo.
Tubing_GLValve_VCH	Numérico (Doble)	VCH de GLValve (tubería).	Leer Modelo Pozo.
Tubing_GLValve_VP	Texto (50)	VP de GLValve (tubería).	Leer Modelo Pozo.
Tubing_GLValve_VAP	Texto (50)	VAP de GLValve (tubería).	Leer Modelo Pozo.
Tubing_GLValve_VAB	Texto (50)	VAB de GLValve (tubería).	Leer Modelo Pozo.
Tubing_GLValve_VC	Texto (50)	VC de GLValve (tubería).	Leer Modelo Pozo.
Tubing_Tubing_MD	Numérico (Doble)	MD de la tubería (tubería).	Leer Modelo Pozo.
Tubing_Tubing_ID	Numérico (Doble)	ID de la tubería (tubería).	Leer Modelo Pozo.
Tubing_Tubing_OD	Numérico (Doble)	OD de la tubería (tubería).	Leer Modelo Pozo.

Tubing_Tubing_WT	Númerico (Doble)	WT de la tubería (tubería).	Leer Modelo Pozo.
Tubing_Tubing_R	Númerico (Doble)	R de la tubería (tubería).	Leer Modelo Pozo.
Tubing_Tubing_AT	Númerico (Doble)	AT de la tubería (tubería).	Leer Modelo Pozo.
Tubing_Tubing_Flowtype	Texto (50)	Tipo de flujo en tubo (tubería).	Leer Modelo Pozo.
PintComp_IPRTYPE	Texto (50)	Tipo de IPR.	Leer Modelo Pozo.
PintComp_FluidType	Texto (50)	Tipo de Fluido.	Leer Modelo Pozo.
Vert_WellPi_LiqPi	Númerico (Doble)	Líquido Pi en Vert_WellPi.	Leer Modelo Pozo.
Vert_WellPi_GasPi	Númerico (Doble)	Gas Pi en Vert_WellPi.	Leer Modelo Pozo.
Vert_BackPressure_C	Númerico (Doble)	C en Vert_BackPressure.	Leer Modelo Pozo.
Vert_BackPressure_N	Númerico (Doble)	N en Vert_BackPressure.	Leer Modelo Pozo.
Vert_Fetkovitch_AOFP	Númerico (Doble)	AOFP en Vert_Fetkovitch.	Leer Modelo Pozo.
Vert_Fetkovitch_N	Númerico (Doble)	N en Vert_Fetkovitch.	Leer Modelo Pozo.
Vert_Jones_LiqA	Númerico (Doble)	Líquido A en Vert_Jones.	Leer Modelo Pozo.
Vert_Jones_LiqB	Númerico (Doble)	Líquido B en Vert_Jones.	Leer Modelo Pozo.
Vert_Jones_GasA	Númerico (Doble)	Gas A en Vert_Jones.	Leer Modelo Pozo.
Vert_Jones_GasB	Númerico (Doble)	Gas B en Vert_Jones.	Leer Modelo Pozo.
Vert_Forchheimer_GasA	Númerico (Doble)	Gas A en Vert_Forchheimer.	Leer Modelo Pozo.
Vert_Forchheimer_GasB	Númerico (Doble)	Gas B en Vert_Forchheimer.	Leer Modelo Pozo.
Vert_Vogel_AOFP	Númerico (Doble)	AOFP en Vert_Vogel.	Leer Modelo Pozo.
Vert_Vogel_N	Númerico (Doble)	N en Vert_Vogel.	Leer Modelo Pozo.
Vert_Pseudoss_Permeability	Númerico (Doble)	Permeabilidad en Vert_Pseud.	Leer Modelo Pozo.
Vert_Pseudoss_Thickness	Númerico (Doble)	Espesor en Vert_Pseudos.	Leer Modelo Pozo.
Vert_Pseudoss_DrainRadius	Númerico (Doble)	Radio desagüe en Vert_Pseud.	Leer Modelo Pozo.
Vert_Pseudoss_Skin	Númerico (Doble)	Superficie en Vert_Pseudos.	Leer Modelo Pozo.
Vert_Pseudoss_WellboreId	Númerico (Doble)	WellboreId en Vert_Pseudos.	Leer Modelo Pozo.
Vert_Pseudoss_DamageId	Númerico (Doble)	Id Dañado en Vert_Pseudos.	Leer Modelo Pozo.
Vert_Pseudoss_DamagePerm	Númerico (Doble)	Perm Dañado en Vert_Pseudo.	Leer Modelo Pozo.
Vert_Pseudoss_CompactedId	Númerico (Doble)	Id Compacto en Vert_Pseudos.	Leer Modelo Pozo.
Vert_Pseudoss_CompactedPerm	Númerico (Doble)	Perm.Compacta en Vert_Pseu.	Leer Modelo Pozo.
Vert_Pseudoss_CompInterval	Númerico (Doble)	Intervalo Compacto en Vert P.	Leer Modelo Pozo.
Vert_Pseudoss_CompVertPerm	Númerico (Doble)	Perm. Vert. en Vert_Pseudos.	Leer Modelo Pozo.
Vert_Pseudoss_PerfId	Númerico (Doble)	Id Perf. en Vert_Pseudos.	Leer Modelo Pozo.
Vert_Pseudoss_PerfLength	Númerico (Doble)	Longitud Perf. en Vert_Pseudo.	Leer Modelo Pozo.
Vert_Pseudoss_PerfShotDensity	Númerico (Doble)	Densidad en Vert_Pseudos.	Leer Modelo Pozo.
Vert_Pseudoss_GravelSieve	Númerico (Doble)	Cernedor en Vert_Pseudos.	Leer Modelo Pozo.
Vert_Pseudoss_GravelPerm	Númerico (Doble)	PermGravel en Vert_Pseudos.	Leer Modelo Pozo.
Vert_Pseudoss_GravelTunnel	Númerico (Doble)	Tunel en Vert_Pseudos.	Leer Modelo Pozo.
Vert_Pseudoss_CompOptions	Númerico (Doble)	Opción de composición en Vert_Pseudos.	Leer Modelo Pozo.
PintHorComp_IPROption	Texto (50)	Opción de IPR en Pintor.	Leer Modelo Pozo.
Hor_Dist_LiqPi	Númerico (Doble)	Líquido Pi en Horizontal Dist.	Leer Modelo Pozo.
Hor_Dist_GasPi	Númerico (Doble)	Gas Pi en Horizontal Dist.	Leer Modelo Pozo.
Hor_Joshi_Rextn	Númerico (Doble)	Rextn en Horizontal JOSHI.	Leer Modelo Pozo.
Hor_Joshi_Thickness	Númerico (Doble)	Espesor en Horizontal JOSHI.	Leer Modelo Pozo.
Hor_Joshi_Permeabilityx	Númerico (Doble)	Permeabilidad X en JOSHI.	Leer Modelo Pozo.
Hor_Joshi_Permeabilityy	Númerico (Doble)	Permeabilidad Y en JOSHI.	Leer Modelo Pozo.
Hor_Joshi_Permeabilityz	Númerico (Doble)	Permeabilidad Z en JOSHI.	Leer Modelo Pozo.
Hor_Joshi_Eccent	Númerico (Doble)	Excento en Horizontal JOSHI.	Leer Modelo Pozo.
Hor_Joshi_Length	Númerico (Doble)	Longitud en Horizontal JOSHI.	Leer Modelo Pozo.
Hor_Joshi_WellBoreRadius	Númerico (Doble)	Radio del pozo en JOSHI.	Leer Modelo Pozo.
Hor_Joshi_Skin	Númerico (Doble)	Superficie en JOSHI.	Leer Modelo Pozo.
Hor_Joshi_Viscosity	Númerico (Doble)	Viscosidad en JOSHI.	Leer Modelo Pozo.
Hor_Joshi_OFVT	Númerico (Doble)	CFVT en Horizontal JOSHI.	Leer Modelo Pozo.
Hor_Babu_Xdim	Númerico (Doble)	Dimensión X en BABU.	Leer Modelo Pozo.
Hor_Babu_Ydim	Númerico (Doble)	Dimensión Y en BABU.	Leer Modelo Pozo.
Hor_Babu_Thickness	Númerico (Doble)	Espesor en Horizontal BABU.	Leer Modelo Pozo.
Hor_Babu_Permeabilityx	Númerico (Doble)	Permeabilidad X en BABU.	Leer Modelo Pozo.

Hor_Babu_Permeabilityy	Númérico (Doble)	Permeabilidad Y en BABU.	Leer Modelo Pozo.
Hor_Babu_Permeabilityz	Númérico (Doble)	Permeabilidad Z en BABU.	Leer Modelo Pozo.
Hor_Babu_XWell	Númérico (Doble)	X del pozo en Horizontal BABU.	Leer Modelo Pozo.
Hor_Babu_YWell	Númérico (Doble)	Y del pozo en Horizontal BABU.	Leer Modelo Pozo.
Hor_Babu_ZWell	Númérico (Doble)	Z del pozo en Horizontal BABU.	Leer Modelo Pozo.
Hor_Babu_Length	Númérico (Doble)	Longitud en Horizontal BABU.	Leer Modelo Pozo.
Hor_Babu_WellBoreRadius	Númérico (Doble)	Radio de pozo Horizontal BABU	Leer Modelo Pozo.
Hor_Babu_Babu_Skin	Númérico (Doble)	Superficie en Horizontal BABU.	Leer Modelo Pozo.
Hor_Babu_Viscosity	Númérico (Doble)	Viscosidad en Horizontal BABU.	Leer Modelo Pozo.
Hor_Babu_OFVF	Númérico (Doble)	CFVF en tipo Horizontal BABU.	Leer Modelo Pozo.
Fecha_Hora_Lectura	Fecha/Hora	Fecha y hora de la lectura.	Leer Modelo Pozo.
Clave principal: Numero.			

12. Lectura_Modelo_PROII

NOMBRE DEL CAMPO	TIPO DE DATO / LONGITUD	DESCRIPCIÓN	MÓDULO QUE LA UTILIZA
ID	Autonumérico	Clave de identificación.	Leer Modelo PROII.
Nombre_Modelo	Texto (50)	Nombre del modelo abierto.	Leer Modelo PROII.
Tipo_Objeto	Texto (50)	Tipo del objeto consultado.	Leer Modelo PROII.
Nombre	Texto (50)	Nombre del objeto consultado.	Leer Modelo PROII.
In_Out	Texto (50)	Si el dato es de entrada o de salida.	Leer Modelo PROII.
Pressure	Númérico (Doble)	Presión.	Leer Modelo PROII.
Temperature	Númérico (Doble)	Temperatura.	Leer Modelo PROII.
Unidad de medida	Texto (50)	Presión de caída.	Leer Modelo PROII.
Pressure_drop	Númérico (Doble)	Presión adquirida.	Leer Modelo PROII.
Pressure_ratio	Númérico (Doble)	Presión de proporción.	Leer Modelo PROII.
Inlet_pressure	Númérico (Doble)	Presión de entrada.	Leer Modelo PROII.
Design_pressure	Númérico (Doble)	Presión de diseño.	Leer Modelo PROII.
Estimated_Outlet_Temp	Númérico (Doble)	Temperatura de salida estimada.	Leer Modelo PROII.
Outlet_Temperature	Númérico (Doble)	Temperatura de salida.	Leer Modelo PROII.
Outlet_Pressure	Númérico (Doble)	Presión de salida.	Leer Modelo PROII.
Inside_Diameter	Númérico (Doble)	Diámetro interno.	Leer Modelo PROII.
NPS_Size	Númérico (Doble)	Tamaño nominal de la tubería.	Leer Modelo PROII.
NPS_Schedule	Númérico (Entero largo)	Programa nominal de la tubería.	Leer Modelo PROII.
Pipe_Length	Númérico (Doble)	Longitud de la tubería.	Leer Modelo PROII.
Num_Segments	Númérico (Doble)	Número de segmentos.	Leer Modelo PROII.
Elevation_gain	Númérico (Doble)	Elevación adquirida.	Leer Modelo PROII.
Flow_efficiency	Númérico (Doble)	Eficiencia de flujo.	Leer Modelo PROII.
Hot_Pressure_Drop	Númérico (Doble)	Presión de caída caliente.	Leer Modelo PROII.
Cold_Pressure_Drop	Númérico (Doble)	Presión de caída fría.	Leer Modelo PROII.
SFeed_Stream_Hot	Númérico (Entero largo)	Corriente de alimentación caliente de inicio.	Leer Modelo PROII.
SFeed_Stream_Cold	Númérico (Entero largo)	Corriente de alimentación fría de inicio.	Leer Modelo PROII.
EFeed_Stream_Hot	Númérico (Entero largo)	Corriente de alimentación caliente final.	Leer Modelo PROII.
EFeed_Stream_Cold	Númérico (Entero largo)	Corriente de alimentación fría final.	Leer Modelo PROII.
SProduct_Stream_Hot	Númérico (Entero largo)	Corriente de producto caliente de inicio.	Leer Modelo PROII.
SProduct_Stream_Cold	Númérico (Entero largo)	Corriente de producto fría de inicio.	Leer Modelo PROII.
EProduct_Stream_Hot	Númérico (Entero largo)	Corriente de producto caliente final.	Leer Modelo PROII.
EProduct_Stream_Cold	Númérico (Entero largo)	Corriente de producto fría final.	Leer Modelo PROII.

Number_Feed_Streams	Númérico (Entero largo)	Número de corrientes de alimentación.	Leer Modelo PROII.
Number_Product_Streams	Númérico (Entero largo)	Número de productos de alimentación.	Leer Modelo PROII.
Product_Type	Texto (50)	Tipo de producto.	Leer Modelo PROII.
Pressure_Value	Texto (50)	Valor de la presión.	Leer Modelo PROII.
Pressure_Top	Númérico (Doble)	Presión alta.	Leer Modelo PROII.
Column_Pressure_Drop	Númérico (Doble)	Presión de caída en la columna.	Leer Modelo PROII.
PerTray_Pressure_Drop	Númérico (Doble)	Presión de caída por bandeja.	Leer Modelo PROII.
Name_Description	Texto (50)	Descripción del nombre.	Leer Modelo PROII.
Phase	Númérico (Entero largo)	Fase.	Leer Modelo PROII.
Liq_Fraction_Flag	Númérico (Entero largo)	Bandera de la fracción líquida.	Leer Modelo PROII.
Liq_FracBasis_Flag	Númérico (Entero largo)	Bandera de la base de la fracción líquida.	Leer Modelo PROII.
Liq_Fraction	Númérico (Doble)	Fracción líquida.	Leer Modelo PROII.
Rate	Númérico (Doble)	Gasto.	Leer Modelo PROII.
Molecular_weight	Númérico (Doble)	Peso molecular.	Leer Modelo PROII.
Specific_gravity	Númérico (Doble)	Gravedad específica.	Leer Modelo PROII.
NBP	Númérico (Doble)	Punto normal en que hierve.	Leer Modelo PROII.
Critical_temperature	Númérico (Doble)	Temperatura crítica.	Leer Modelo PROII.
Critical_pressure	Númérico (Doble)	Presión crítica.	Leer Modelo PROII.
Critical_volume	Númérico (Doble)	Volumen crítico.	Leer Modelo PROII.
Crit_Compress_Factor	Númérico (Doble)	Factor de compresión crítico.	Leer Modelo PROII.
Fecha_Hora	Fecha/Hora	Fecha y hora de la captura.	Leer Modelo PROII.
Clave principal: Nombre_Modelo-Nombre-Fecha_Hora			

13. Lectura_Modelo_Red_Pipesim2000

NOMBRE DEL CAMPO	TIPO DE DATO / LONGITUD	DESCRIPCION	MÓDULO QUE LA UTILIZA
Numero	Autonumérico	Clave de identificación.	Leer Modelo Red.
nom_Modelo	Texto (80)	Nombre del modelo abierto.	Leer Modelo Red.
nom_Objeto	Texto (50)	Nombre del objeto consultado.	Leer Modelo Red.
Tipo_Objeto	Texto (50)	Tipo del objeto consultado.	Leer Modelo Red.
ArtificialLift	Texto (50)	Tiene ArtificialLift?	Leer Modelo Red.
Pressure(psia)	Númérico (Doble)	Presión en psia.	Leer Modelo Red.
Temperature(C)	Númérico (Doble)	Temperatura en C.	Leer Modelo Red.
LiquidRate(STB/D)	Númérico (Doble)	Gasto de líquido.	Leer Modelo Red.
GasRate(mmscf/d)	Númérico (Doble)	Gasto de Gas.	Leer Modelo Red.
MassRate(lb/s)	Númérico (Doble)	Gasto de masa.	Leer Modelo Red.
Branch_Choke_BeanSize	Númérico (Doble)	Tamaño del estrangulador. (Una Rama).	Leer Modelo Red.
Branch_Tubing_TubModel	Texto (50)	Modelo de la tubería (tubería). (Una Rama).	Leer Modelo Red.
Branch_Tubing_SurfaceTVD	Texto (50)	Superficie de la tubería. (Una Rama).	Leer Modelo Red.
Branch_Tubing_SurfaceTemperature	Númérico (Doble)	Temperatura de la superficie. (Una Rama).	Leer Modelo Red.
Branch_Tubing_KickoffTVD	Texto (50)	TVD inicial de la tubería. (Una Rama).	Leer Modelo Red.
Branch_Tubing_KickoffAngle	Texto (50)	Ángulo inicial de la tubería. (Una Rama).	Leer Modelo Red.
Branch_Tubing_PerfsTVD	Texto (50)	TVD adquirido de la tubería. (Una Rama).	Leer Modelo Red.
Branch_Tubing_PerfsMD	Texto (50)	Profundidad adquirida. (Una Rama).	Leer Modelo Red.

		Rama).	
Branch_Tubing_PerfsTemp	Numérico (Doble)	Temperatura adquirida. (Una Rama).	Leer Modelo Red.
Branch_Separator_MeasuredDepth	Numérico (Doble)	Profundidad del separador. (Una Rama).	Leer Modelo Red.
Branch_Separator_Efficiency	Numérico (Doble)	Eficiencia del separador. (Una Rama).	Leer Modelo Red.
Branch_Separator_Type	Texto (50)	Tipo de separador. (Una Rama).	Leer Modelo Red.
Branch_Tubing_Choke_MD	Numérico (Doble)	Profundidad de estrangulador. (Una Rama).	Leer Modelo Red.
Branch_Tubing_Choke_BS	Numérico (Doble)	Tamaño estrangulador (tubo). (Una Rama).	Leer Modelo Red.
Branch_Tubing_Pump_MD	Numérico (Doble)	Profundidad de la bomba. (Una Rama).	Leer Modelo Red.
Branch_Tubing_Pump_ESPSt	Texto (50)	ESPSt de la bomba (tubería). (Una Rama).	Leer Modelo Red.
Branch_Tubing_Pump_ESPSp	Texto (50)	ESPSp de la bomba (tubería). (Una Rama).	Leer Modelo Red.
Branch_Tubing_Injector_MD	Numérico (Doble)	Profundidad Injector (tubería). (Una Rama).	Leer Modelo Red.
Branch_Tubing_Injector_GLR	Numérico (Doble)	GLR del Injector (tubería). (Una Rama).	Leer Modelo Red.
Branch_Tubing_Injector_GLST	Numérico (Doble)	GLST del Injector (tubería). (Una Rama).	Leer Modelo Red.
Branch_Tubing_Injector_GLSG	Texto (50)	GLSG del Injector (tubería). (Una Rama).	Leer Modelo Red.
Branch_Tubing_Injector_GLPD	Numérico (Doble)	GLPD del Injector (tubería). (Una Rama).	Leer Modelo Red.
Branch_Tubing_Injector_GLSP	Numérico (Doble)	GLSP del Injector (tubería). (Una Rama).	Leer Modelo Red.
Branch_Tubing_Separator_MD	Numérico (Doble)	Profundidad separador (tubo). (Una Rama).	Leer Modelo Red.
Branch_Tubing_Separator_Eff	Numérico (Doble)	Eficiencia separador (tubería). (Una Rama).	Leer Modelo Red.
Branch_Tubing_Separator_Type	Texto (50)	Tipo de separador (tubería). (Una Rama).	Leer Modelo Red.
Branch_Tubing_SSSV_MD	Numérico (Doble)	Profundidad SSSV (tubería). (Una Rama).	Leer Modelo Red.
Branch_Tubing_SSSV_BID	Texto (50)	BID SSSV de la tubería. (Una Rama).	Leer Modelo Red.
Branch_Tubing_GLValve_MD	Numérico (Doble)	Profundidad de GLValve (tubo). (Una Rama).	Leer Modelo Red.
Branch_Tubing_GLValve_VPS	Numérico (Doble)	VPS de GLValve (tubería). (Una Rama).	Leer Modelo Red.
Branch_Tubing_GLValve_VCH	Numérico (Doble)	VCH de GLValve (tubería). (Una Rama).	Leer Modelo Red.
Branch_Tubing_GLValve_VP	Texto (50)	VP de GLValve (tubería). (Una Rama).	Leer Modelo Red.
Branch_Tubing_GLValve_VAP	Texto (50)	VAP de GLValve (tubería). (Una Rama).	Leer Modelo Red.
Branch_Tubing_GLValve_VAB	Texto (50)	VAB de GLValve (tubería). (Una Rama).	Leer Modelo Red.
Branch_Tubing_GLValve_VC	Texto (50)	VC de GLValve (tubería). (Una Rama).	Leer Modelo Red.
Branch_Tubing_Tubing_MD	Numérico (Doble)	MD de la tubería (tubería). (Una Rama).	Leer Modelo Red.
Branch_Tubing_Tubing_ID	Numérico (Doble)	ID de la tubería (tubería).	Leer Modelo Red.

		(Una Rama).	
Branch_Tubing_Tubing_OD	Numérico (Doble)	OD de la tubería (tubería). (Una Rama).	Leer Modelo Red.
Branch_Tubing_Tubing_WT	Numérico (Doble)	WT de la tubería (tubería). (Una Rama).	Leer Modelo Red.
Branch_Tubing_Tubing_R	Numérico (Doble)	R de la tubería (tubería). (Una Rama).	Leer Modelo Red.
Branch_Tubing_Tubing_AT	Numérico (Doble)	AT de la tubería (tubería). (Una Rama).	Leer Modelo Red.
Branch_Tubing_Tubing_Flowtype	Texto (50)	Tipo de flujo en tubo (tubería). (Una Rama).	Leer Modelo Red.
ProdWell_Choke_BeanSize	Numérico (Doble)	Tamaño del estrangulador. (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_TubModel	Texto (50)	Modelo de la tubería (tubería). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_SurfaceTVD	Texto (50)	Superficie de la tubería. (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_SurfaceTemperature	Numérico (Doble)	Temperatura de la superficie. (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_KickoffTVD	Texto (50)	TVD inicial de la tubería. (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_KickoffAngle	Texto (50)	Ángulo inicial de la tubería. (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_PerfsTVD	Texto (50)	TVD adquirido de la tubería. (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_PerfsMD	Texto (50)	Profundidad adquirida. (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_PerfsTemp	Numérico (Doble)	Temperatura adquirida. (Pozo de Producción).	Leer Modelo Red.
ProdWell_Separator_MeasuredDepth	Numérico (Doble)	Profundidad del separador. (Pozo de Producción).	Leer Modelo Red.
ProdWell_Separator_Efficiency	Numérico (Doble)	Eficiencia del separador. (Pozo de Producción).	Leer Modelo Red.
ProdWell_Separator_Type	Texto (50)	Tipo de separador. (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_Choke_MD	Numérico (Doble)	Profundidad de estrangulador. (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_Choke_BS	Numérico (Doble)	Tamaño estrangulador (tubo). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_Pump_MD	Numérico (Doble)	Profundidad de la bomba. (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_Pump_ESPSt	Texto (50)	ESPSt de la bomba (tubería). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_Pump_ESPSp	Texto (50)	ESPSp de la bomba (tubería). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_Injector_MD	Numérico (Doble)	Profundidad Injector (tubería). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_Injector_GLR	Numérico (Doble)	GLR del Injector (tubería). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_Injector_GLST	Numérico (Doble)	GLST del Injector (tubería). (Pozo de Producción).	Leer Modelo Red.

ProdWell_Tubing_Injector_GLSG	Texto (50)	GLSG del Inyector (tubería). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_Injector_GLPD	Númérico (Doble)	GLPD del Inyector (tubería). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_Injector_GLSP	Númérico (Doble)	GLSP del Inyector (tubería). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_Separator_MD	Númérico (Doble)	Profundidad separador (tubo). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_Separator_Eff	Númérico (Doble)	Eficiencia separador (tubería). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_Separator_Type	Texto (50)	Tipo de separador (tubería). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_SSSV_MD	Númérico (Doble)	Profundidad SSSV (tubería). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_SSSV_BID	Texto (50)	BID SSSV de la tubería. (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_GLValve_MD	Númérico (Doble)	Profundidad de GLValve (tubo). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_GLValve_VPS	Númérico (Doble)	VPS de GLValve (tubería). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_GLValve_VCH	Númérico (Doble)	VCH de GLValve (tubería). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_GLValve_VP	Texto (50)	VP de GLValve (tubería). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_GLValve_VAP	Texto (50)	VAP de GLValve (tubería). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_GLValve_VAB	Texto (50)	VAB de GLValve (tubería). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_GLValve_VC	Texto (50)	VC de GLValve (tubería). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_Tubing_MD	Númérico (Doble)	MD de la tubería (tubería). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_Tubing_ID	Númérico (Doble)	ID de la tubería (tubería). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_Tubing_OD	Númérico (Doble)	OD de la tubería (tubería). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_Tubing_WT	Númérico (Doble)	WT de la tubería (tubería). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_Tubing_R	Númérico (Doble)	R de la tubería (tubería). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_Tubing_AT	Númérico (Doble)	AT de la tubería (tubería). (Pozo de Producción).	Leer Modelo Red.
ProdWell_Tubing_Tubing_Flowtype	Texto (50)	Tipo de flujo en tubo (tubería). (Pozo de Producción).	Leer Modelo Red.
InjWell_Choke_BeanSize	Númérico (Doble)	Tamaño del estrangulador. (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_TubModel	Texto (50)	Modelo de la tubería (tubería). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_SurfaceTVD	Texto (50)	Superficie de la tubería. (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_SurfaceTemperature	Númérico (Doble)	Temperatura de la superficie. (Pozo de inyección).	Leer Modelo Red.

InjWell_Tubing_KickoffTVD	Texto (50)	TVD inicial de la tubería. (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_KickoffAngle	Texto (50)	Ángulo inicial de la tubería. (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_PerfsTVD	Texto (50)	TVD adquirido de la tubería. (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_PerfsMD	Texto (50)	Profundidad adquirida. (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_PerfsTemp	Numérico (Doble)	Temperatura adquirida. (Pozo de inyección).	Leer Modelo Red.
InjWell_Separator_MeasuredDepth	Numérico (Doble)	Profundidad del separador. (Pozo de inyección).	Leer Modelo Red.
InjWell_Separator_Efficiency	Numérico (Doble)	Eficiencia del separador. (Pozo de inyección).	Leer Modelo Red.
InjWell_Separator_Type	Texto (50)	Tipo de separador. (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_Choke_MD	Numérico (Doble)	Profundidad de estrangulador. (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_Choke_BS	Numérico (Doble)	Tamaño estrangulador (tubo). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_Pump_MD	Numérico (Doble)	Profundidad de la bomba. (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_Pump_ESPSt	Texto (50)	ESPSt de la bomba (tubería). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_Pump_ESPSP	Texto (50)	ESPSP de la bomba (tubería). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_Injector_MD	Numérico (Doble)	Profundidad Inyector (tubería). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_Injector_GLR	Numérico (Doble)	GLR del Inyector (tubería). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_Injector_GLST	Numérico (Doble)	GLST del Inyector (tubería). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_Injector_GLSG	Texto (50)	GLSG del Inyector (tubería). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_Injector_GLPD	Numérico (Doble)	GLPD del Inyector (tubería). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_Injector_GLSP	Numérico (Doble)	GLSP del Inyector (tubería). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_Separator_MD	Numérico (Doble)	Profundidad separador (tubo). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_Separator_Eff	Numérico (Doble)	Eficiencia separador (tubería). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_Separator_Type	Texto (50)	Tipo de separador (tubería). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_SSSV_MD	Numérico (Doble)	Profundidad SSSV (tubería). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_SSSV_BID	Texto (50)	BID SSSV de la tubería. (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_GLValve_MD	Numérico (Doble)	Profundidad de GLValve (tubo). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_GLValve_VPS	Numérico (Doble)	VPS de GLValve (tubería). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_GLValve_VCH	Numérico (Doble)	VCH de GLValve (tubería).	Leer Modelo Red.

		(Pozo de inyección).	
InjWell_Tubing_GLValve_VP	Texto (50)	VP de GLValve (tubería). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_GLValve_VAP	Texto (50)	VAP de GLValve (tubería). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_GLValve_VAB	Texto (50)	VAB de GLValve (tubería). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_GLValve_VC	Texto (50)	VC de GLValve (tubería). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_Tubing_MD	Numérico (Doble)	MD de la tubería (tubería). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_Tubing_ID	Numérico (Doble)	ID de la tubería (tubería). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_Tubing_OD	Numérico (Doble)	OD de la tubería (tubería). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_Tubing_WT	Numérico (Doble)	WT de la tubería (tubería). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_Tubing_R	Numérico (Doble)	R de la tubería (tubería). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_Tubing_AT	Numérico (Doble)	AT de la tubería (tubería). (Pozo de inyección).	Leer Modelo Red.
InjWell_Tubing_Tubing_Flowtype	Texto (50)	Tipo de flujo en tubo (tubería). (Pozo de inyección).	Leer Modelo Red.
PintComp_IPRType	Texto (50)	Tipo de IPR.	Leer Modelo Red.
PintComp_FluidType	Texto (50)	Tipo de Fluido.	Leer Modelo Red.
PintHorComp_IPROption	Texto (50)	Opción de IPR.	Leer Modelo Red.
Fecha_Hora_Lectura	Fecha/Hora	Fecha y hora de la lectura.	Leer Modelo Red.
Clave principal: Numero			

14. Mixer_Unit_Operation

NOMBRE DEL CAMPO	TIPO DE DATO / LONGITUD	DESCRIPCION	MÓDULO QUE LA UTILIZA
Id	Autonumérico	Clave de identificación.	Editar Modelo PROII.
Nombre_Modelo	Texto (50)	Nombre del modelo abierto.	Editar Modelo PROII.
Tipo_Objeto	Texto (50)	Tipo del objeto consultado.	Editar Modelo PROII.
Nombre	Texto (50)	Nombre del objeto consultado.	Editar Modelo PROII.
Pressure	Numérico (Doble)	Presión.	Editar Modelo PROII.
Pressure_drop	Numérico (Doble)	Presión de caída.	Editar Modelo PROII.
Temperature	Numérico (Doble)	Temperatura.	Editar Modelo PROII.
In_Out	Texto (50)	Si el dato es de entrada o salida.	Editar Modelo PROII.
Fecha_Hora	Texto (50)	Fecha y hora de la captura.	Editar Modelo PROII.

15. Pipe_Unit_Operation

NOMBRE DEL CAMPO	TIPO DE DATO / LONGITUD	DESCRIPCION	MÓDULO QUE LA UTILIZA
Id	Autonumérico	Clave de identificación.	Editar Modelo PROII.
Nombre_Modelo	Texto (50)	Nombre del modelo abierto.	Editar Modelo PROII.
Tipo_Objeto	Texto (50)	Tipo del objeto consultado.	Editar Modelo PROII.
Nombre	Texto (50)	Nombre del objeto consultado.	Editar Modelo PROII.
Inside_Diameter	Numérico (Doble)	Diámetro interno.	Editar Modelo PROII.
NPS_Size	Numérico (Doble)	Tamaño nominal de la tubería.	Editar Modelo PROII.
NPS_Schedule	Numérico (Entero largo)	Programa nominal de la tubería.	Editar Modelo PROII.
Pipe_Length	Numérico (Doble)	Longitud de la tubería.	Editar Modelo PROII.
Num_Segments	Numérico (Doble)	Número de segmentos.	Editar Modelo PROII.
Elevation_gain	Numérico (Doble)	Elevación adquirida.	Editar Modelo PROII.

Flow_efficiency	Númérico (Doble)	Eficiencia de flujo.	Editar Modelo PROII.
Inlet_Temperature	Númérico (Doble)	Temperatura de entrada.	Editar Modelo PROII.
Outlet_Temperature	Númérico (Doble)	Temperatura de salida.	Editar Modelo PROII.
Inlet_Pressure	Númérico (Doble)	Presión de entrada.	Editar Modelo PROII.
Outlet_Pressure	Númérico (Doble)	Presión de salida.	Editar Modelo PROII.
In_Out	Texto (50)	Si el dato es de entrada o de salida.	Editar Modelo PROII.
Fecha_Hora	Texto (50)	Fecha y hora de la captura.	Editar Modelo PROII.

16. Pump_Unit_Operation

NOMBRE DEL CAMPO	TIPO DE DATO / LONGITUD	DESCRIPCION	MÓDULO QUE LA UTILIZA
Id	Autonumérico	Clave de identificación.	Editar Modelo PROII.
Nombre_Modelo	Texto (50)	Nombre del Modelo abierto.	Editar Modelo PROII.
Tipo_Objeto	Texto (50)	Tipo del objeto consultado.	Editar Modelo PROII.
Nombre	Texto (50)	Nombre del objeto consultado.	Editar Modelo PROII.
Pressure	Númérico (Doble)	Presión.	Editar Modelo PROII.
Pressure_gain	Númérico (Doble)	Presión de caída.	Editar Modelo PROII.
Pressure_ratio	Númérico (Doble)	Presión de proporción.	Editar Modelo PROII.
Temperature	Númérico (Doble)	Temperatura.	Editar Modelo PROII.
In_Out	Texto (50)	Si el dato es de entrada o salida.	Editar Modelo PROII.
Fecha_Hora	Texto (50)	Fecha y hora de la captura.	Editar Modelo PROII.

17. Reporte_Modelo_Pozo_Pipesim2000

NOMBRE DEL CAMPO	TIPO DE DATO / LONGITUD	DESCRIPCION	MÓDULO QUE LA UTILIZA
Numero	Autonumérico	Clave de identificación.	Editar Modelo Red – Pozo.
nom_Modelo	Texto (50)	Nombre del Modelo abierto.	Editar Modelo Red – Pozo.
Nombre_Nodo	Texto (50)	Nombre del nodo consultado.	Editar Modelo Red – Pozo.
Tipo_Nodo	Texto (50)	Tipo del nodo consultado.	Editar Modelo Red – Pozo.
Pressure	Númérico (Doble)	Presión.	Editar Modelo Red – Pozo.
Temperature	Númérico (Doble)	Temperatura.	Editar Modelo Red – Pozo.
LiquidRate	Númérico (Doble)	Gasto de Líquido.	Editar Modelo Red – Pozo.
GasRate	Númérico (Doble)	Gasto de gas.	Editar Modelo Red – Pozo.
MassRate	Númérico (Doble)	Gasto de masa.	Editar Modelo Red – Pozo.
GLR	Númérico (Doble)	GLR.	Editar Modelo Red – Pozo.
WaterCut	Númérico (Doble)	Corte de Agua.	Editar Modelo Red – Pozo.
PressureResidual	Númérico (Doble)	Presión Residual.	Editar Modelo Red – Pozo.
RateResidual	Númérico (Doble)	Gasto Residual.	Editar Modelo Red – Pozo.
LGR	Númérico (Doble)	LGR.	Editar Modelo Red – Pozo.
MaxPres	Númérico (Doble)	Presión Máxima.	Editar Modelo Red – Pozo.
CompPower	Númérico (Doble)	Poder Comp.	Editar Modelo Red – Pozo.
Choke_DP	Númérico (Doble)	DP del estrangulador.	Editar Modelo Red – Pozo.
Clave principal: Numero			

18. Reporte_Modelo_Red_Pipesim2000

NOMBRE DEL CAMPO	TIPO DE DATO / LONGITUD	DESCRIPCION	MÓDULO QUE LA UTILIZA
Numero	Autonumérico	Clave de identificación.	Editar Modelo Red – Pozo.
nom_Modelo	Texto (50)	Nombre del Modelo abierto.	Editar Modelo Red – Pozo.
Nombre_Nodo	Texto (50)	Nombre del nodo consultado.	Editar Modelo Red – Pozo.
Tipo_Nodo	Texto (50)	Tipo del nodo consultado.	Editar Modelo Red – Pozo.
Pressure	Númérico (Doble)	Presión.	Editar Modelo Red – Pozo.
Temperature	Númérico (Doble)	Temperatura.	Editar Modelo Red – Pozo.
LiquidRate	Númérico (Doble)	Gasto de Líquido.	Editar Modelo Red – Pozo.
GasRate	Númérico (Doble)	Gasto de gas.	Editar Modelo Red – Pozo.

MassRate	Númérico (Doble)	Gasto de masa.	Editar Modelo Red – Pozo.
GLR	Númérico (Doble)	GLR.	Editar Modelo Red – Pozo.
WaterCut	Númérico (Doble)	Corte de Agua.	Editar Modelo Red – Pozo.
PressureResidual	Númérico (Doble)	Presión Residual.	Editar Modelo Red – Pozo.
LGR	Númérico (Doble)	LGR.	Editar Modelo Red – Pozo.
MaxPres	Númérico (Doble)	Presión Máxima.	Editar Modelo Red – Pozo.
CompPower	Númérico (Doble)	Poder Comp.	Editar Modelo Red – Pozo.
Choke_DP	Númérico (Doble)	DP del estrangulador.	Editar Modelo Red – Pozo.
Clave principal: Numero			

19. SHExchanger_Unit_Operation

NOMBRE DEL CAMPO	TIPO DE DATO / LONGITUD	DESCRIPCION	MÓDULO QUE LA UTILIZA
Id	Autonumérico	Clave de identificación.	Editar Modelo PROII.
Nombre_Modelo	Texto (50)	Nombre del modelo abierto.	Editar Modelo PROII.
Tipo_Objeto	Texto (50)	Tipo del objeto consultado.	Editar Modelo PROII.
Nombre	Texto (50)	Nombre del objeto consultado.	Editar Modelo PROII.
Hot_Pressure_Drop	Númérico (Doble)	Presión de caída caliente.	Editar Modelo PROII.
Cold_Pressure_Drop	Númérico (Doble)	Presión de caída fría.	Editar Modelo PROII.
SFeed_Stream_Hot	Númérico (Entero largo)	Corriente de alimentación de inicio caliente.	Editar Modelo PROII.
SFeed_Stream_Cold	Númérico (Entero largo)	Corriente de alimentación de inicio fría.	Editar Modelo PROII.
EFeed_Stream_Hot	Númérico (Entero largo)	Corriente de alimentación final caliente.	Editar Modelo PROII.
EFeed_Stream_Cold	Númérico (Entero largo)	Corriente de alimentación final fría.	Editar Modelo PROII.
SProduct_Stream_Hot	Númérico (Entero largo)	Corriente de producto de inicio caliente.	Editar Modelo PROII.
SProduct_Stream_Cold	Númérico (Entero largo)	Corriente de producto de inicio fría.	Editar Modelo PROII.
EProduct_Stream_Hot	Númérico (Entero largo)	Corriente de producto final caliente.	Editar Modelo PROII.
EProduct_Stream_Cold	Númérico (Entero largo)	Corriente de producto final fría.	Editar Modelo PROII.
Utility_side_flag	Númérico (Entero largo)	Utilidad de la bandera.	Editar Modelo PROII.
Utility_temp_In	Númérico (Doble)	Utilidad de la temperatura de entrada.	Editar Modelo PROII.
Utility_temp_Out	Númérico (Doble)	Utilidad de la temperatura de salida.	Editar Modelo PROII.
Utility_saturation_pressure	Númérico (Doble)	Utilidad de la presión de saturación.	Editar Modelo PROII.
Utility_flowrate	Númérico (Doble)	Utilidad del gasto de flujo.	Editar Modelo PROII.
Utility_Cp	Númérico (Doble)	Utilidad Cp.	Editar Modelo PROII.
Number_Shell_Passes	Númérico (Entero largo)	Número de Fases del casco.	Editar Modelo PROII.
Number_Tube_Passes	Númérico (Doble)	Numero de fases del tubo.	Editar Modelo PROII.
In_Out	Texto (50)	Si el dato es de entrada o de salida.	Editar Modelo PROII.
Fecha_Hora	Texto (50)	Fecha y hora de la captura.	Editar Modelo PROII.

20. Splitter_Unit_Operation

NOMBRE DEL CAMPO	TIPO DE DATO / LONGITUD	DESCRIPCION	MÓDULO QUE LA UTILIZA
Id	Autonumérico	Clave de identificación.	Editar Modelo PROII.
Nombre_Modelo	Texto (50)	Nombre del modelo abierto.	Editar Modelo PROII.
Tipo_Objeto	Texto (50)	Tipo del objeto consultado.	Editar Modelo PROII.

Nombre	Texto (50)	Nombre del objeto consultado.	Editar Modelo PROII.
Pressure	N Numérico (Doble)	Presión.	Editar Modelo PROII.
Pressure_drop	N Numérico (Doble)	Presión de caída.	Editar Modelo PROII.
Temperature	N Numérico (Doble)	Temperatura.	Editar Modelo PROII.
In_Out	Texto (50)	Si el dato es de entrada o de salida.	Editar Modelo PROII.
Fecha_Hora	Texto (50)	Fecha y hora de la captura.	Editar Modelo PROII.

21. Stream_Data

NOMBRE DEL CAMPO	TIPO DE DATO / LONGITUD	DESCRIPCION	MÓDULO QUE LA UTILIZA
Id	Autonumérico	Clave de identificación.	Editar Modelo PROII.
Nombre_Modelo	Texto (50)	Nombre del modelo abierto.	Editar Modelo PROII.
Tipo_Objeto	Texto (50)	Tipo del objeto consultado.	Editar Modelo PROII.
Nombre	Texto (50)	Nombre del objeto consultado.	Editar Modelo PROII.
Name_Description	Texto (50)	Descripción del nombre.	Editar Modelo PROII.
Temperature	N Numérico (Doble)	Temperatura.	Editar Modelo PROII.
Pressure	N Numérico (Doble)	Presión.	Editar Modelo PROII.
Phase	N Numérico (Entero largo)	Fase.	Editar Modelo PROII.
Liq_Fraction_Flag	N Numérico (Entero largo)	Bandera de Fracción de líquido.	Editar Modelo PROII.
Liq_FracBasis_Flag	N Numérico (Entero largo)	Bandera base de fracción de líquido.	Editar Modelo PROII.
Liq_Fraction	N Numérico (Doble)	Fracción de líquido.	Editar Modelo PROII.
Rate	N Numérico (Doble)	Gasto.	Editar Modelo PROII.
Component_Mole_Fractions	Texto (50)	Fracciones de componentes.	Editar Modelo PROII.
Vapor_Phase_Comp	Texto (50)	Composición de la fase vapor.	Editar Modelo PROII.
Liquid_Phase_Comp	Texto (50)	Composición de la fase líquida.	Editar Modelo PROII.
Solid_Phase_Comp	Texto (50)	Composición de la fase sólida.	Editar Modelo PROII.
Mole_flow_rate	N Numérico (Doble)	Gasto de Flujo molar.	Editar Modelo PROII.
Mole_fraction_vapor	N Numérico (Doble)	Fracción de vapor molar.	Editar Modelo PROII.
Mole_fraction_liquid	N Numérico (Doble)	Fracción líquida molar.	Editar Modelo PROII.
Mole_fraction_water	N Numérico (Doble)	Fracción de agua molar.	Editar Modelo PROII.
Mole_fraction_inerts	N Numérico (Doble)	Fracción inerte molar.	Editar Modelo PROII.
Weight_fraction_solids	N Numérico (Doble)	Fracción sólida de peso.	Editar Modelo PROII.
Molar_enthalpy	N Numérico (Doble)	Entalpía molar.	Editar Modelo PROII.
Molar_entropy	N Numérico (Doble)	Entropía molar.	Editar Modelo PROII.
Molecular_weight	N Numérico (Doble)	Peso molecular.	Editar Modelo PROII.
Tc	N Numérico (Doble)	Temperatura crítica.	Editar Modelo PROII.
Pc	N Numérico (Doble)	Presión crítica.	Editar Modelo PROII.
Vc	N Numérico (Doble)	Volumen crítico.	Editar Modelo PROII.
Cp_Cv_Ratio	N Numérico (Doble)	Proporción Cp Cv.	Editar Modelo PROII.
In_Out	Texto (50)	Si el dato es de entrada o de salida.	Editar Modelo PROII.
Fecha_Hora	Texto (50)	Fecha y hora de la captura.	Editar Modelo PROII.

22. Valve_Unit_Operation

NOMBRE DEL CAMPO	TIPO DE DATO / LONGITUD	DESCRIPCION	MÓDULO QUE LA UTILIZA
Id	Autonumérico	Clave de identificación.	Editar Modelo PROII.
Nombre_Modelo	Texto (50)	Nombre del modelo abierto.	Editar Modelo PROII.
Tipo_Objeto	Texto (50)	Tipo del objeto consultado.	Editar Modelo PROII.
Nombre	Texto (50)	Nombre del objeto consultado.	Editar Modelo PROII.
Pressure	Númérico (Doble)	Presión.	Editar Modelo PROII.
Pressure_drop	Númérico (Doble)	Presión de caída.	Editar Modelo PROII.
Temperature	Númérico (Doble)	Temperatura.	Editar Modelo PROII.
In_Out	Texto (50)	Si el dato es de entrada o salida.	Editar Modelo PROII.
Fecha_Hora	Texto (50)	Fecha y hora de la captura.	Editar Modelo PROII.

23. Variables_de_Pozo

NOMBRE DEL CAMPO	TIPO DE DATO / LONGITUD	DESCRIPCION	MÓDULO QUE LA UTILIZA
Id	Númérico	Clave de identificación	Variables de pozo.
Campo	Texto (255)	Nombre del Campo.	Variables de pozo.
Plataforma	Texto (255)	Nombre de la plataforma.	Variables de pozo.
Pozo	Texto (50)	Nombre del pozo.	Variables de pozo.
Fecha	Fecha/Hora	Fecha de exploración.	Variables de pozo.
Gasto_Aceite	Númérico (Doble)	Gasto de Aceite Qo(bpd).	Variables de pozo.
Gasto_Gas_Inyeccion	Númérico (Doble)	Gasto Iny. Qgi(mmcpd).	Variables de pozo.
Rga	Númérico (Doble)	Rga en (m ³ /m ³).	Variables de pozo.
Presion_Cabeza_Pozo	Númérico (Doble)	Presión en (kg/cm ²).	Variables de pozo.
Diametro_Estrangulador	Númérico (Doble)	Diámetro en (Pg).	Variables de pozo.
Presion_Fondo_Estatica	Númérico (Doble)	Presión en (kg/cm ²).	Variables de pozo.
Presion_Fondo_Fluyendo	Númérico (Doble)	Presión en (kg/cm ²).	Variables de pozo.
Delta_P	Númérico (Doble)	Delta P en (kg/cm ²).	Variables de pozo.
Temperatura_Cabeza_Pozo_Fluyendo	Númérico (Doble)	Temperatura en (C).	Variables de pozo.
Estado_Mecanico	Objeto OLE	Imagen del estado.	Variables de pozo.
Plano_Localizacion	Objeto OLE	Imagen del plano.	Variables de pozo.
Seccion_Estructural	Objeto OLE	Imagen de la sección	Variables de pozo.
Clave principal: Pozo-Fecha			



ANEXO B

Id	nom_Modelo	nom_Objeto	tipo_Objeto	Tipo_Operacion
3	INYGAS.bps	VWell_1	Pintvertcomp	
4	INYGAS.bps	VWell_1	Pintvertcomp	
5	PS09 Gas Lift Performance.bps	VWell_1	Pintvertcomp	
6	PS17 Bottom Hole Nodal Analysis.bps	VERTICAL COMPLETIO	Pintvertcomp	
7	PS08 System Analysis.bps	VWell_1	Pintvertcomp	
8	ENRIQUE KUMAZAKU10.bps	VertWell_1	Pintvertcomp	
9	PS11 ESP Lift.bps	Well_1	Pintvertcomp	
10	Oil Well DesignA.bps	VertWell_1	Pintvertcomp	
11	PS06 Reservoir Tables.bps	VWell_1	Pintvertcomp	
12	PS16 Gas Lift Design.bps	VertWell_1	Pintvertcomp	
13	Oil Well DesignA.bps	VertWell_1	Pintvertcomp	
14	PS11 ESP Lift.bps	Well_1	Pintvertcomp	
15	PS07 Injection System.bps	VWell_1	Pintvertcomp	
16	Gas Lift Performance.bps	VWell_1	Pintvertcomp	
17	PS17 Bottom Hole Nodal Analysis.bps	VERTICAL COMPLETIO	Pintvertcomp	
18	PS14 Multiphase Booster.bps	VertWell_1	Pintvertcomp	
19	Oil Well DesignA.bps	VertWell_1	Pintvertcomp	
20	PS02 Pressure and Temperature Profil	VWell_1	Pintvertcomp	
21	Gas Lift Performance.bps	VWell_1	Pintvertcomp	
22	Gas Lift Performance.bps	VWell_1	Pintvertcomp	
23	PS07 Injection System.bps	VWell_1	Pintvertcomp	
24	INYGAS.bps	VWell_1	Pintvertcomp	
25	PS09 Gas Lift Performance.bps	VWell_1	Pintvertcomp	
26	INYGAS.bps	VWell_1	Pintvertcomp	
27	PS01 Nodal Analysis.bps	Well_2	Pintvertcomp	
28	PS11 ESP Lift.bps	Well_1	Pintvertcomp	
29	Gas Lift Performance.bps	VWell_1	Pintvertcomp	
30	Oil Well DesignA.bps	VertWell_1	Pintvertcomp	
31	INYGAS.bps	VWell_1	Pintvertcomp	
32	PS04 Horizontal Well Performance.bps	HWell_1	Pinthorcomp	
33	PS11 ESP Lift.bps	Well_1	Pintvertcomp	
34	PS01 Nodal Analysis.bps	Well_2	Pintvertcomp	
35	Oil Well Design.bps	VertWell_1	Pintvertcomp	

Pressure(psia)	Temperature(C)	Fecha_Hora
9000	700	2001 12:46:12 p.m.
9000	700	2001 12:46:53 p.m.
4000	250	2001 12:47:59 p.m.
4000	250	2001 12:53:50 p.m.
4269	210	2001 01:18:48 p.m.
160	111	2001 01:20:47 p.m.
4269	210	2001 02:49:44 p.m.
2500	180	2001 02:57:23 p.m.
4000	250	2001 09:13:45 a.m.
3800	275	2001 09:15:22 a.m.
2500	180	2001 09:16:10 a.m.
4269	220	2001 10:10:56 a.m.
3800	190	2001 10:21:00 a.m.
4000	250	2001 10:28:51 a.m.
4000	250	2001 10:35:39 a.m.
1958	112.1	2001 10:41:20 a.m.
2500	180	2001 10:45:41 a.m.
4269	210	2001 10:49:42 a.m.
4000	250	2001 10:50:01 a.m.
4000	250	2001 10:51:43 a.m.
3800	190	2001 11:26:29 a.m.
9000	700	2001 11:39:29 a.m.
4000	250	2001 11:56:44 a.m.
9000	700	2001 11:56:59 a.m.
4269	210	2001 12:16:15 p.m.
4269	220	2001 01:04:34 p.m.
4000	250	2001 01:12:50 p.m.
2500	180	2001 01:17:38 p.m.
9000	700	2001 01:18:30 p.m.
2400	115	2001 01:19:44 p.m.
4269	220	2001 01:20:21 p.m.
4269	210	2001 01:22:53 p.m.
5000	220	2001 01:23:58 p.m.

Id	Nombre_Modelo	Tipo_Objeto	Nombre	Pressure	Pressure_drop	Temperature	In_Out
1	A1.prz	Flash	C2V			195.3722	In
2	A1.prz	Flash	C3V			233.15	In
3	A1.prz	Flash	HXT1		0	297.0389	In
4	A1.prz	Flash	F2	1723.69			In
5	G3.prz	Flash	F1				In
6	A1.prz	Flash	F1		0		In
7	G3.prz	Flash	F1				In
8	A1.prz	Flash	F1		0		In
9	A1.prz	Flash	C2V			195.3722	In
10	A1.prz	Flash	C3V			233.15	In
11	A1.prz	Flash	HXT1		0	297.0389	In
12	A1.prz	Flash	F2	1723.69			In
13	G3.prz	Flash	F1				In
14	A1.prz	Flash	F1		0		In
15	A1.prz	Flash	C2V			195.3722	In
16	A1.prz	Flash	C3V			233.15	In
17	A1.prz	Flash	HXT1		0	297.0389	In
18	A1.prz	Flash	F2	1723.69			In
19	G3.prz	Flash	F1				In
20	G3.prz	Flash	F1				In
21	G3.prz	Flash	F1				In
22	G3.prz	Flash	F1				In
23	A1.prz	Flash	F1		0		In
24	A1.prz	Flash	C2V			195.3722	In
25	A1.prz	Flash	C3V			233.15	In
26	A1.prz	Flash	HXT1		0	297.0389	In
27	A1.prz	Flash	F2	1723.69			In
28	G3.prz	Flash	F1				In
29	G3.prz	Flash	F1				In
30	G3.prz	Flash	F1				In
31	G3.prz	Flash	F1				In
32	G3.prz	Flash	F1				In
33	G3.prz	Flash	F1				In
34	G3.prz	Flash	F1				In
35	G3.prz	Flash	F1				In
36	G3.prz	Flash	F1				In
37	G3.prz	Flash	F1				In
38	G3.prz	Flash	F1				In
39	G3.prz	Flash	F1				In
40	G3.prz	Flash	F1				In
41	G3.prz	Flash	F1				In
42	G3.prz	Flash	F1				In
43	A1.prz	Flash	F1		0		In
44	A1.prz	Flash	C2V			195.3722	In
45	A1.prz	Flash	C3V			233.15	In
46	A1.prz	Flash	HXT1		0	297.0389	In
47	A1.prz	Flash	F2	1723.69			In

Fecha_Hora
16/12/2003 11:23:40
16/12/2003 11:23:40
16/12/2003 11:23:40
16/12/2003 11:23:40
16/12/2003 11:24:40
16/12/2003 11:23:40
16/12/2003 11:41:31
16/12/2003 11:43:44
16/12/2003 11:43:44
16/12/2003 11:43:44
16/12/2003 11:43:44
16/12/2003 11:43:44
16/12/2003 11:53:46
16/12/2003 12:13:23
16/12/2003 12:13:23
16/12/2003 12:13:23
16/12/2003 12:13:23
16/12/2003 12:13:23
16/12/2003 12:14:35
16/12/2003 12:23:32
16/12/2003 12:24:01
16/12/2003 12:24:27
16/12/2003 12:29:19
16/12/2003 12:29:19
16/12/2003 12:29:19
16/12/2003 12:29:19
16/12/2003 12:29:19
16/12/2003 12:31:20
16/12/2003 13:00:44
16/12/2003 13:15:50
16/12/2003 13:19:35
16/12/2003 13:19:50
16/12/2003 13:40:09
16/12/2003 15:41:57
16/12/2003 15:54:54
16/12/2003 15:57:10
16/12/2003 15:58:06
16/12/2003 15:58:27
16/12/2003 16:01:27
16/12/2003 16:04:30
16/12/2003 16:12:58
16/12/2003 16:14:34
16/12/2003 16:19:02
16/12/2003 16:19:02
16/12/2003 16:19:02
16/12/2003 16:19:02
16/12/2003 16:19:02

Id	Nombre_Modelo	Tipo_Objeto	Nombre	Pressure	Pressure_drop	Temperature	In_Out
48	A1.prz	Flash	F1		0		In
49	A1.prz	Flash	C2V			195.3722	In
50	A1.prz	Flash	C3V			233.15	In
51	A1.prz	Flash	HXT1		0	297.0389	In
52	A1.prz	Flash	F2	1723.69			In
53	G3.prz	Flash	F1				In
54	A1.prz	Flash	F1		0		In
55	A1.prz	Flash	C2V			195.3722	In
56	A1.prz	Flash	C3V			233.15	In
57	A1.prz	Flash	HXT1		0	297.0389	In
58	A1.prz	Flash	F2	1723.69			In
59	A1.prz	Flash	F1		0		In
60	A1.prz	Flash	C2V			195.3722	In
61	A1.prz	Flash	C3V			233.15	In
62	A1.prz	Flash	HXT1		0	297.0389	In
63	A1.prz	Flash	F2	1723.69			In
64	G3.prz	Flash	F1				In
65	G3.prz	Flash	F1				In
66	G3.prz	Flash	F1				In
67	G3.prz	Flash	F1				In
68	G3.prz	Flash	F1				In
69	G3.prz	Flash	F1				In
70	G3.prz	Flash	F1				In
71	A1.prz	Flash	F1		0		In
72	A1.prz	Flash	C2V			195.3722	In
73	A1.prz	Flash	C3V			233.15	In
74	A1.prz	Flash	HXT1		0	297.0389	In
75	A1.prz	Flash	F2	1723.69			In
76	A1.prz	Flash	F1		0		In
77	A1.prz	Flash	C2V			195.3722	In
78	A1.prz	Flash	C3V			233.15	In
79	A1.prz	Flash	HXT1		0	297.0389	In
80	A1.prz	Flash	F2	1723.69			In
81	A1.prz	Flash	F1		0		In
82	A1.prz	Flash	C2V			195.3722	In
83	A1.prz	Flash	C3V			233.15	In
84	A1.prz	Flash	HXT1		0	297.0389	In
85	A1.prz	Flash	F2	1723.69			In
86	flash.prz	Flash	F1	500		303.15	In
87	flash.prz	Flash	F1	500		303.15	In
88	G3.prz	Flash	F1				In
89	A1.prz	Flash	F1		0		In
90	A1.prz	Flash	C2V			195.3722	In
91	A1.prz	Flash	C3V			233.15	In
92	A1.prz	Flash	HXT1		0	297.0389	In
93	A1.prz	Flash	F2	1723.69			In
94	G3.prz	Flash	F1	60			In

Fecha Hora
16/12/2003 16:19:15
16/12/2003 16:19:15
16/12/2003 16:19:15
16/12/2003 16:19:15
16/12/2003 16:19:15
16/12/2003 16:19:59
16/12/2003 16:20:13
16/12/2003 16:20:13
16/12/2003 16:20:13
16/12/2003 16:20:13
16/12/2003 16:20:13
16/12/2003 16:20:29
16/12/2003 16:20:29
16/12/2003 16:20:29
16/12/2003 16:20:29
16/12/2003 16:20:29
16/12/2003 16:20:53
16/12/2003 16:22:15
16/12/2003 16:22:29
17/12/2003 9:59:58
17/12/2003 10:04:37
17/12/2003 10:11:03
17/12/2003 10:45:03
17/12/2003 10:45:29
17/12/2003 10:45:29
17/12/2003 10:45:29
17/12/2003 10:45:29
17/12/2003 10:45:29
17/12/2003 10:45:44
17/12/2003 10:45:44
17/12/2003 10:45:44
17/12/2003 10:45:44
17/12/2003 10:45:44
17/12/2003 10:45:44
17/12/2003 10:47:12
17/12/2003 10:47:12
17/12/2003 10:47:12
17/12/2003 10:47:12
17/12/2003 10:47:56
17/12/2003 10:48:39
17/12/2003 10:56:13
17/12/2003 12:53:58
17/12/2003 12:53:58
17/12/2003 12:53:58
17/12/2003 12:53:58
17/12/2003 12:53:58
17/12/2003 15:27:27

Id	Nombre Modelo	Tipo Objeto	Nombre	Name Description	Temperature
1	G3.prz	Stream Data	1	PROPANE	333.15
2	G3.prz	Stream Data	2		
3	G3.prz	Stream Data	1	PROPANE	333.15
4	G3.prz	Stream Data	2		
5	G3.prz	Stream Data	1	PROPANE	333.15
6	G3.prz	Stream Data	2		
7	G3.prz	Stream Data	1	PROPANE	333.15
8	G3.prz	Stream Data	2		
9	G3.prz	Stream Data	1	PROPANE	333.15
10	G3.prz	Stream Data	2		
11	G3.prz	Stream Data	1	PROPANE	333.15
12	G3.prz	Stream Data	2		
13	G3.prz	Stream Data	1	PROPANE	333.15
14	G3.prz	Stream Data	2		
15	G3.pr1	Stream Data	1	PROPANE	333.15
16	G3.pr1	Stream Data	2		
17	G3.pr1	Stream Data	1		
18	G3.pr1	Stream Data	1	PROPANE	333.15
19	G3.pr1	Stream Data	2		
20	G3.pr1	Stream Data	1		
21	G3.pr1	Stream Data	1	PROPANE	333.15
22	G3.pr1	Stream Data	2		
23	G3.pr1	Stream Data	1		
24	G3.pr1	Stream Data	1	PROPANE	333.15
25	G3.pr1	Stream Data	2		
26	G3.pr1	Stream Data	1	PROPANE	333.15
27	G3.pr1	Stream Data	2		
28	G3.pr1	Stream Data	1		333.15
29	G3.pr1	Stream Data	1	PROPANE	333.15
30	G3.pr1	Stream Data	2		
31	G3.pr1	Stream Data	1		333.15
32	G3.pr1	Stream Data	1		333.15
33	G3.pr1	Stream Data	1		333.15
34	A1.prz	Stream Data	5		197.0389
35	A1.prz	Stream Data	10		297.0389
36	A1.prz	Stream Data	18		244.2611
37	G3.prz	Stream Data	1	PROPANE	333.15
38	G3.prz	Stream Data	2		
39	G3.prz	Stream Data	1	PROPANE	333.15
40	G3.prz	Stream Data	2		
41	G3.prz	Stream Data	1	PROPANE	333.15
42	G3.prz	Stream Data	2		
43	G3.prz	Stream Data	1	PROPANE	333.15
44	G3.prz	Stream Data	2		
45	G3.prz	Stream Data	1	PROPANE	333.15
46	G3.prz	Stream Data	2		
47	G3.prz	Stream Data	1	PROPANE	333.15

Pressure	Phase	Liq_Fraction_Flag	Liq_FracBasis	Liq_Fraction	Rate
2171.8494	0	1	1		2.52
	0	1	1		
2171.8494	0	1	1		2.52
	0	1	1		
2171.8494	0	1	1		2.52
	0	1	1		
2171.8494	0	1	1		2.52
	0	1	1		
2171.8494	0	1	1		2.52
	0	1	1		
2171.8494	0	1	1		2.52
	0	1	1		
2171.8494	0	0	0	0	2.52
	0	1	1		
2171.8494	0	0	0	0	2.52
	0	1	1		
2171.8494	0	0	0	0	2.52
	0	1	1		
2171.8494	0	0	0	0	2.52
	0	1	1		
2171.8494	0	0	0	0	2.52
	0	1	1		
444.843					
2171.8494	0	0	0	0	2.52
	0	1	1		
444.843					
444.843					
444.843					
4619.4893	0	1	1		
1689.2162	0	1	1		
4274.751	0	1	1		
2171.8494	0	1	1		2.52
	0	1	1		
2171.8494	0	1	1		2.52
	0	1	1		
2171.8494	0	1	1		2.52
	0	1	1		
2171.8494	0	1	1		2.52
	0	1	1		
2171.8494	0	1	1		2.52
	0	1	1		
2171.8494	0	1	1		2.52

Tc	Pc	Vc	Cp_Cv Ratio	In Out	Fecha Hora
				In	16/12/2003 13:19:3
				In	16/12/2003 13:19:3
				In	16/12/2003 13:19:5
				In	16/12/2003 13:19:5
				In	16/12/2003 13:40:0
				In	16/12/2003 13:40:0
				In	16/12/2003 15:41:5
				In	16/12/2003 15:41:5
				In	16/12/2003 15:54:5
				In	16/12/2003 15:54:5
				In	16/12/2003 15:57:1
				In	16/12/2003 15:57:1
				In	16/12/2003 15:58:0
				In	16/12/2003 15:58:0
0	0	0	0	In	08/01/2004 11:06:3
0	0	0	0	In	08/01/2004 11:06:3
0	0	0	0	Out	08/01/2004 11:07:0
0	0	0	0	In	08/01/2004 11:19:1
0	0	0	0	In	08/01/2004 11:19:1
0	0	0	0	Out	08/01/2004 11:19:4
0	0	0	0	In	08/01/2004 11:56:1
0	0	0	0	In	08/01/2004 11:56:1
0	0	0	0	Out	08/01/2004 11:56:5
0	0	0	0	In	08/01/2004 12:20:5
0	0	0	0	In	08/01/2004 12:20:5
0	0	0	0	In	08/01/2004 12:21:0
0	0	0	0	In	08/01/2004 12:21:0
0	0	0	0	Out	08/01/2004 12:21:3
0	0	0	0	In	08/01/2004 12:24:0
0	0	0	0	In	08/01/2004 12:24:0
0	0	0	0	Out	08/01/2004 12:24:2
0	0	0	0	Out	08/01/2004 12:26:3
0	0	0	0	Out	08/01/2004 12:27:2
				In	16/12/2003 16:20:2
				In	16/12/2003 16:20:2
				In	16/12/2003 16:20:2
				In	16/12/2003 16:20:5
				In	16/12/2003 16:20:5
				In	16/12/2003 16:22:1
				In	16/12/2003 16:22:1
				In	16/12/2003 16:22:2
				In	16/12/2003 16:22:2
				In	17/12/2003 9:59:58
				In	17/12/2003 9:59:58
				In	17/12/2003 10:04:3
				In	17/12/2003 10:04:3
				In	17/12/2003 10:11:0

Numero	nom_Modelo	Upstream_Most_Object	nom_Objeto	Tipo_Objeto
1	WELL_12.bps		WELL 1	Tubing
2	WELL_12.bps		N1	Node
3	WELL_12.bps		FLOWLINE 1	Flowline
4	WELL_12.bps	VERTICAL COMPLETION 1	N2	Node
5	WELL_12.bps		VERTICAL COMPLETION 1	Pintvertcomp
6	PS17 Bottom Hole		NAPnt_1	Napoint
7	PS17 Bottom Hole		TUBING 1	Tubing
8	PS17 Bottom Hole		N1	Node
9	PS17 Bottom Hole		FLOWLINE 1	Flowline
10	PS17 Bottom Hole		N2	Node
11	PS17 Bottom Hole		RISER 1	Riser
12	PS17 Bottom Hole	VERTICAL COMPLETION 2	N3	Node
13	PS04 Horizontal		HWell_1	Pinthorcomp
14	PS04 Horizontal		Tubing	Tubing
15	PS04 Horizontal	HWell_1	Outlet	Node
16	PS07 Injection Sy		Src_1	Pintsource
17	PS07 Injection Sy		Riser_1	Riser
18	PS07 Injection Sy		N1	Node
19	PS07 Injection Sy		Flowl_1	Flowline
20	PS07 Injection Sy		N2	Node
21	PS07 Injection Sy		Tub_1	Tubing
22	PS07 Injection Sy	Src_1	VWell_1	Pintvertcomp
23	PS15 Wax Deposi		Source_1	Pintsource
24	PS07 Injection Sy		N1	Node
25	PS07 Injection Sy		Flowl_1	Flowline
26	PS07 Injection Sy		N2	Node
27	PS07 Injection Sy		Tub_1	Tubing
28	PS07 Injection Sy	Src_1	VWell_1	Pintvertcomp
29	Gas Lift Performa		VWell_1	Pintvertcomp
30	Gas Lift Performa		Tub_1	Tubing
31	Gas Lift Performa		N3	Node
32	PS15 Wax Deposi		Flowline_1	Flowline
33	PS15 Wax Deposi		N1	Node
34	PS15 Wax Deposi		Riser_1	Riser
35	PS15 Wax Deposi	Source_1	N2	Node
36	Gas Lift Performa		Flowl_1	Flowline
37	Gas Lift Performa		N4	Node
38	Gas Lift Performa		Riser_1	Riser
39	Gas Lift Performa	VWell_1	S1	Boundary Node
40	WELL_12.bps		VERTICAL COMPLETION 1	Pintvertcomp
41	WELL_12.bps		WELL 1	Tubing
42	WELL_12.bps		N1	Node
43	WELL_12.bps		FLOWLINE 1	Flowline
44	WELL_12.bps	VERTICAL COMPLETION 1	N2	Node
45	PS15 Wax Deposi		Source_1	Pintsource
46	PS15 Wax Deposi		Flowline_1	Flowline
47	PS15 Wax Deposi		N1	Node
48	PS06 Reservoir T		VWell_1	Pintvertcomp
49	PS06 Reservoir T		Tub_1	Tubing
50	PS06 Reservoir T		N3	Node
51	PS06 Reservoir T		Flowl_1	Flowline
52	PS06 Reservoir T		N4	Node
53	PS06 Reservoir T		Riser_1	Riser

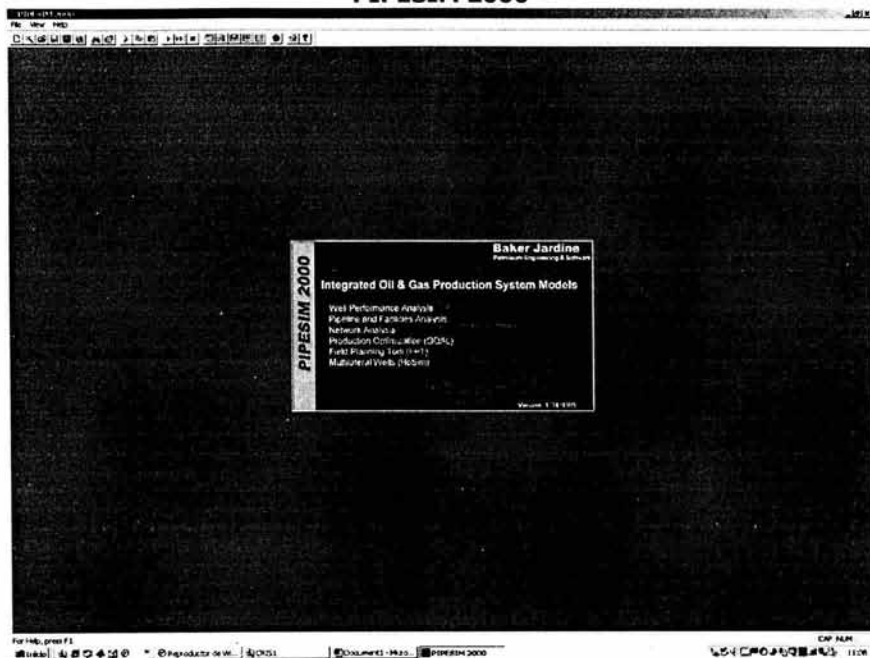
Tubing_ArtificialLift	Pressure(psia)	Temperature(C)	LiquidRate(STB/D)	GasRate(mmscf/d)
Gas lift injection	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	1600	190	0	0
	0	0	0	0
Gas lift injection	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	2400	115	0	0
No artificial lift	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
No artificial lift	0	0	0	0
	3800	190	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
No artificial lift	0	0	0	0
	3800	190	0	0
	4000	250	0	0
Gas lift injection	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	1600	190	0	0
Gas lift injection	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	4000	250	0	0
Gas lift injection	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0
	0	0	0	0

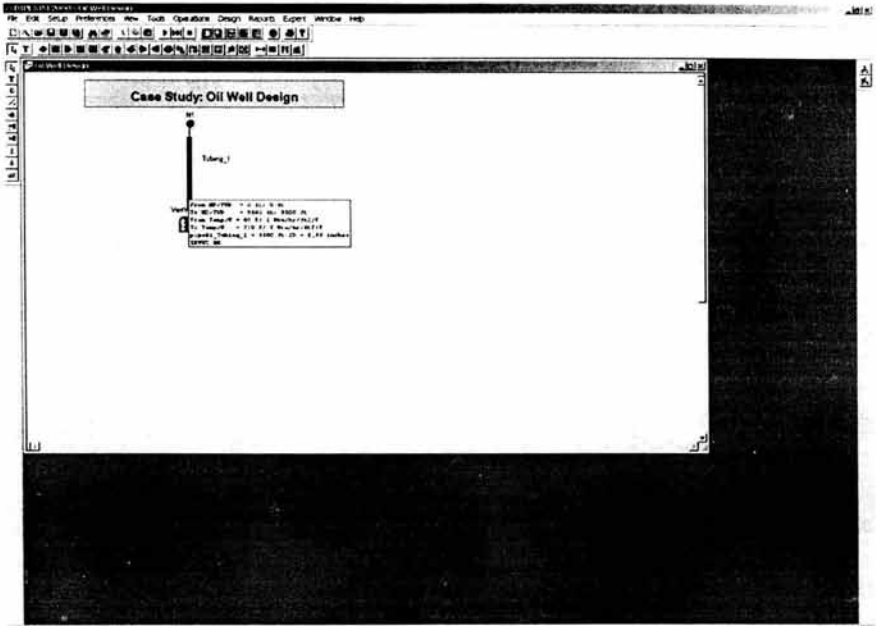
Tubing_SurfaceTemperature	Tubing KickoffTVD	Tubing KickoffAngle
00		0
0		
0		
0		
0		
0		
33		3
0		
0		
0		
0		
0		
-777-7777		-7777
0		
0		
0		
0		
0		
00		0
0		
0		
0		
0		
00		0
0		
0		
0		
0		
0		
0		
0		
0		
00		0
0		
0		
0		
0		
0		
0		
00		0
0		
0		
0		
0		
0		
00		0
0		
0		
0		
0		

ANEXO C

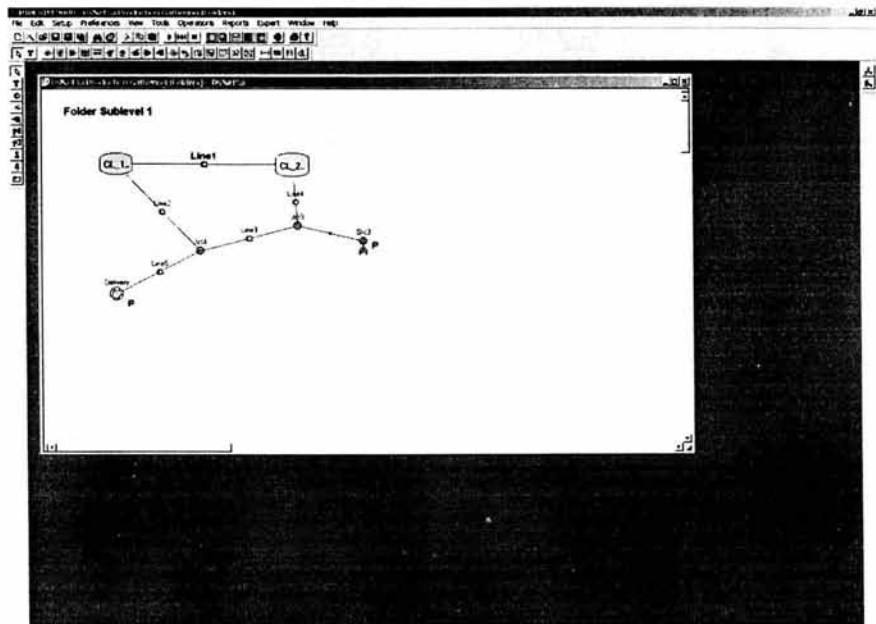
HERRAMIENTAS DE SIMULACIÓN UTILIZADAS EN LA APLICACIÓN

PIPESIM 2000

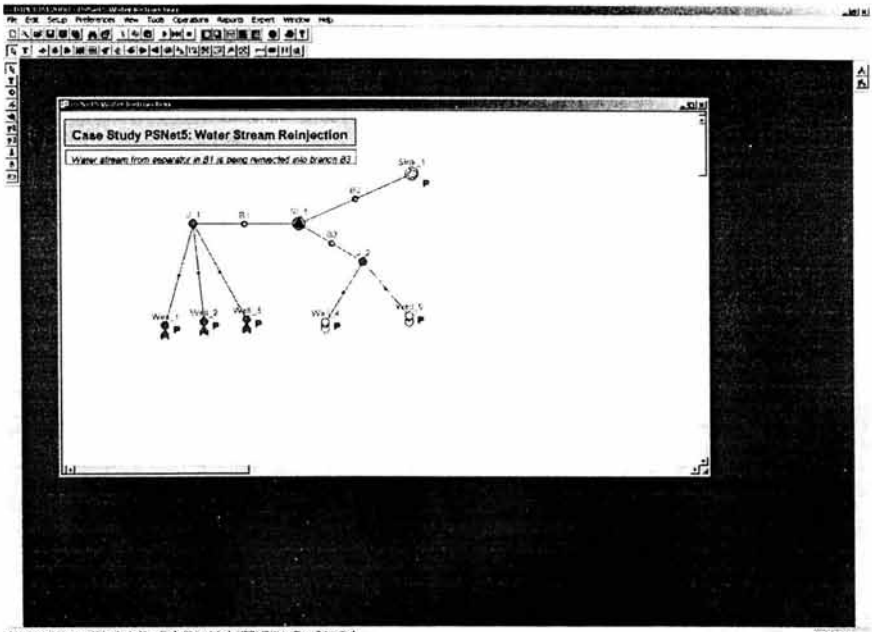




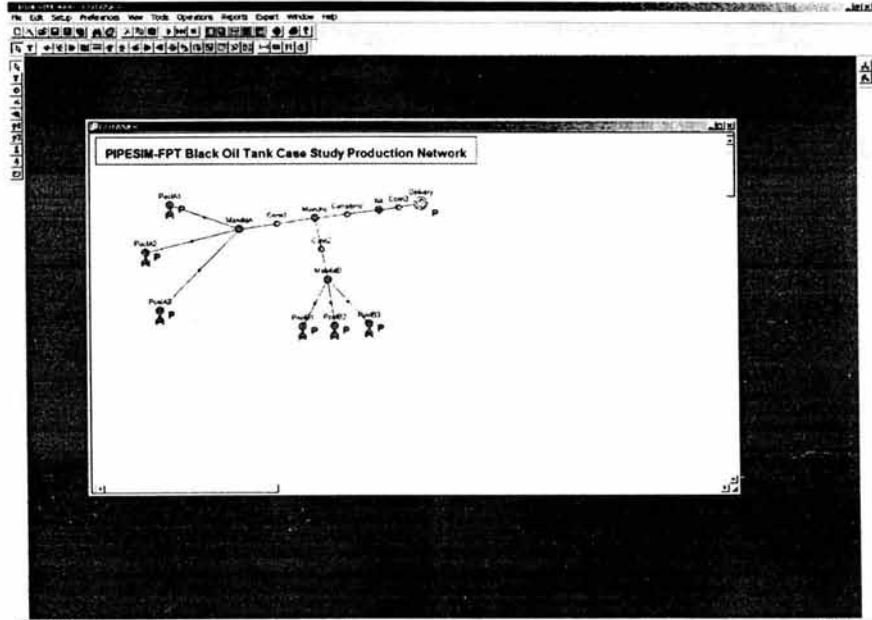
For help, press F1 | @ Reproducir de W | @ CASI | @ Documents - HERR | @ PAPERM 2000 - ... | CAP 14.M | 11/04



Cartolina de programador de la Case Study Network Analysis para la Production Offshoring (Folder) | @ Reproducir de W | @ CASI | @ Documents - HERR | @ PAPERM 2000 - ... | CAP 14.M | 11/04



C:\Program Files\Aspentech\Aspen\PIPESIM\Case Studies\PSNet5\Water Stream Reinjection
 C:\Program Files\Aspentech\Aspen\PIPESIM\Case Studies\PSNet5\Water Stream Reinjection
 C:\Program Files\Aspentech\Aspen\PIPESIM\Case Studies\PSNet5\Water Stream Reinjection
 C:\Program Files\Aspentech\Aspen\PIPESIM\Case Studies\PSNet5\Water Stream Reinjection



C:\Program Files\Aspentech\Aspen\PIPESIM\Case Studies\FPT\Black Oil Tank Case Study Production Network
 C:\Program Files\Aspentech\Aspen\PIPESIM\Case Studies\FPT\Black Oil Tank Case Study Production Network
 C:\Program Files\Aspentech\Aspen\PIPESIM\Case Studies\FPT\Black Oil Tank Case Study Production Network
 C:\Program Files\Aspentech\Aspen\PIPESIM\Case Studies\FPT\Black Oil Tank Case Study Production Network

PRO/II 5.61

