

**INSTITUTO TECNOLÓGICO Y DE ESTUDIOS  
SUPERIORES DE MONTERREY**



**T E S I S**

**DEFINICION DE UNA ESTRATEGIA DE INTEGRACION  
DE APLICACIONES EMPRESARIALES PARA SISTEMAS  
HEREDADOS DENTRO DEL NIVEL DE DATOS**

**MAESTRIA EN ADMINISTRACION  
DE TECNOLOGIAS DE INFORMACION**

**POR  
LEOBARDO CORTES MANICA**

**NOVIEMBRE DE 2001**

**INSTITUTO TECNOLÓGICO Y DE ESTUDIOS SUPERIORES DE  
MONTERREY**



**TESIS**

**DEFINICIÓN DE UNA ESTRATEGIA DE INTEGRACIÓN DE  
APLICACIONES EMPRESARIALES PARA SISTEMAS  
HEREDADOS DENTRO DEL NIVEL DE DATOS**

**MAESTRÍA EN ADMINISTRACIÓN DE TECNOLOGÍAS DE  
INFORMACIÓN**

**POR**

**LEOBARDO CORTÉS MÁNICA**

**NOVIEMBRE DE 2001**

**DEFINICIÓN DE UNA ESTRATEGIA DE INTEGRACIÓN  
DE APLICACIONES EMPRESARIALES DE SISTEMAS  
HEREDADOS DENTRO DEL NIVEL DE DATOS**

**POR**

**LEOBARDO CORTÉS MÁNICA**

**TESIS**

**Presentada a la División de Graduados en Electrónica,  
Computación, Información y Comunicaciones.**

**Este trabajo es requisito parcial para obtener el título de**

**Maestro en Administración de Tecnologías de  
Información**

**INSTITUTO TECNOLÓGICO Y DE ESTUDIOS SUPERIORES  
DE MONTERREY**

**NOVIEMBRE DE 2001**

*A mis padres*

*José Leobardo y Concepción*

*Porque sacrificaron sus sueños personales para ofrecerme lo mejor del  
mundo: amor*

*Muchas gracias*

## AGRADECIMIENTOS

---

*A cada uno de mis amigos que han estado conmigo en las buenas y en las malas durante todo este tiempo.*

*A Elda, Adriana y David, quienes siempre tuvieron las palabras precisas para darle un toque profesional a este trabajo.*

*A mis padres, que siempre me apoyaron en todas las decisiones que tomé a lo largo de mi vida.*

*A mis hermanos, con quienes tuve la dicha de pasar muchos momentos divertidos.*

*A Dios, por que siempre me ha enseñado el mejor camino a seguir.*

Actualmente, las tecnologías de información han provocado un giro en la forma de operar de la mayoría de las empresas, de la misma manera que se ha dado un cambio en la sociedad. Los avances que surgen en materia de informática, telecomunicaciones y redes, han venido a ofrecer una nueva forma de implementar sistemas de información capaces de comunicarse con otros sistemas, ya sean dentro de la institución o incluso con sus proveedores y clientes.

Mucho se ha hablado de ERP (*Enterprise Resource Planning*), ASP (*Application Service Provider*), entre otras tecnologías que han surgido en la actualidad para automatizar los procesos organizacionales de muchas empresas. Sin embargo, la integración de los sistemas heredados, o como se les conoce comúnmente “*legacy systems*”, con los sistemas que se están desarrollando con nuevas tecnologías debe considerarse como una estrategia organizacional necesaria. El interés por desarrollar este estudio se debe a que muchas instituciones o empresas de nuestro país, no cuentan con los suficientes recursos para adquirir sistemas que involucren un total rediseño de la organización. Debido a esto, la unificación de los sistemas heredados dentro de la organización permitirá una comunicación más efectiva dentro y fuera de las áreas organizacionales, de tal manera se reducirán costos y el negocio contará con una base sólida para su incursión en negocios electrónicos.

Con base a esto, existe una metodología conocida como Integración de Aplicaciones Empresariales (*EAI*, por sus siglas en inglés); la cual consiste en ubicar cuatro etapas diferentes que abarcan, desde el nivel de datos hasta el nivel de aplicaciones; por medio de la cual es posible determinar qué tan integrados están los sistemas de información de una institución. Sin embargo, si entendemos como integración a la transportación y transformación de información entre dos o más aplicaciones, es necesario identificar desde el principio la manera como las instituciones manejan la información contenida en los sistemas, los formatos que emplean, así como si existe algún tipo de interacción entre

dichas aplicaciones. Basándose en esto, será posible encontrar una manera más acorde a la realidad organizacional para proponer una herramienta que sea útil para integrar los sistemas.

Finalmente, la integración de todos estos sistemas traerá como conclusión una sólida base tecnológica. Esto a su vez permitirá a la corporación una fácil transición hacia la organización electrónica, puesto que existirá una mejor relación entre cada una de sus aplicaciones y será más sencillo el soporte de las transacciones que se realicen a través de sus sistemas de información.

## TABLA DE CONTENIDO

---

---

Dedicatoria.....	v
Agradecimientos.....	vi
Resumen.....	vii
Tabla de Contenido.....	ix
Lista de Figuras.....	xii
Lista de Tablas.....	xiii
<b>Capítulo 1. Introducción.....</b>	<b>1</b>
1.1 Objetivo.....	5
1.2 Alcance del Estudio.....	5
1.3 Producto Final.....	6
1.4 Organización de la tesis.....	6
1.5 Metodología utilizada.....	7
<b>Capítulo 2. Revisión Bibliográfica: Conceptos básicos de EAI.....</b>	<b>9</b>
2.1 Sistemas heredados ( <i>Legacy Systems</i> ).....	9
2.2 <i>Enterprise Resource Planning (ERP)</i> .....	11
2.3 <i>Enterprise Application Integration (EAI)</i> .....	13
2.4 Clasificación de los diferente niveles de EAI.....	17
2.4.1 Nivel de datos.....	18
2.4.2 Nivel de interfaz de aplicación.....	18
2.4.3 Nivel de método.....	18
2.4.4 Nivel de interfaz de usuario.....	19
2.5 Características del nivel de datos.....	20
2.6 Características del nivel de interfaz de aplicación.....	21
2.7 Características del nivel de método.....	23
2.8 Características del nivel de interfaz de usuario.....	25
2.9 Implementación del nivel de datos de EAI.....	26
2.9.1 Identificación de los datos.....	27
2.9.2 Categorización de los datos.....	29
2.9.3 Clasificación del modelo de metadatos.....	30
<b>Capítulo 3. Revisión Bibliográfica: Metodologías para la EAI.....</b>	<b>32</b>
3.1 Tecnologías y estándares existentes para la Integración de Aplicaciones.....	32
3.1.1 Transferencia de archivos por lotes.....	32
3.1.2 Conectividad de Base de Datos Abierta ( <i>ODBC</i> ).....	32
3.1.3 Acceso a la base de datos por <i>middleware</i> .....	33
3.1.4 Transformación de datos.....	33
3.2 <i>Middleware</i> .....	34
3.2.1 Tipos de Comunicación.....	34
3.2.1.1 Comunicación asíncrona.....	34
3.2.1.2 Comunicación síncrona.....	36



3.2.2	Tipos de <i>middleware</i> .....	37
3.2.2.1	<i>Remote Procedure Call (RPC)</i> .....	38
3.2.2.2	<i>Message Oriented Middleware (MOM)</i> .....	39
3.2.2.3	<i>Middleware Orientado a las Bases de Datos</i> .....	40
3.2.2.4	Objetos Distribuidos.....	40
3.2.2.5	<i>Middleware Orientado a Transacciones</i> .....	41
	• Monitores de Procesamiento de Transacciones .....	41
	• Servidores de Aplicaciones.....	42
3.3	Evaluación del <i>middleware</i> .....	43
3.4	Integración a través de interfaces de datos.....	44
3.4.1	Tareas de planeación.....	45
3.4.2	Tareas de implementación.....	47
<b>Capítulo 4. Soluciones Comerciales y Casos de Estudio</b> .....		50
4.1	Soluciones comerciales por tipo de <i>middleware</i> .....	50
4.1.1	<i>RPC (Remote Procedure Call)</i> .....	50
4.1.2	<i>MOM (Message Oriented Middleware)</i> .....	51
4.1.3	<i>Middleware orientado a bases de datos</i> .....	52
4.1.4	<i>Middleware orientado a objetos distribuidos</i> .....	53
4.1.5	<i>Middleware orientado a transacciones</i> .....	54
4.2	Casos de aplicación de <i>middleware</i> .....	56
<b>Capítulo 5. Investigación de Campo</b> .....		59
5.1	Objetivo.....	59
5.2	Muestra seleccionada.....	59
5.3	Metodología de la Investigación.....	60
5.4	Perfil de las personas entrevistadas.....	61
5.5	Diseño de la entrevista.....	62
<b>Capítulo 6. Análisis de Resultados</b> .....		63
6.1	Información General.....	63
6.2	Resultados de la entrevistas.....	63
6.2.1	Sistemas de Información.....	64
6.2.2	Bases de datos.....	66
6.2.3	Tecnologías de acceso a los datos.....	66
6.2.4	<i>Middleware</i> empleado para conectar bases de datos.....	67
6.3	Departamento de Servicios de Información Administrativa.....	69
6.3.1	Bases de datos.....	69
6.3.2	Aplicaciones de las bases de datos.....	69
6.3.3	Tecnologías de acceso a los datos.....	70
6.3.4	<i>Middleware</i> empleado para conectar bases de datos.....	70

6.4 Departamento de Servicios de Información Escolar.....	71
6.4.1 Bases de datos.....	71
6.4.2 Aplicaciones de las bases de datos.....	71
6.4.3 Tecnologías de acceso a los datos.....	72
6.4.4 <i>Middleware</i> empleado para conectar bases de datos.....	73
6.5 Departamento de Ingeniería de Información.....	74
6.5.1 Bases de datos.....	74
6.5.2 Aplicaciones de las bases de datos.....	74
6.5.3 Tecnologías de acceso a los datos.....	74
6.5.4 <i>Middleware</i> empleado para conectar bases de datos.....	74
6.6 Departamento de Biblioteca Digital.....	75
6.6.1 Bases de datos.....	75
6.6.2 Aplicaciones de las bases de datos.....	75
6.6.3 Tecnologías de acceso a los datos.....	76
6.6.4 <i>Middleware</i> empleado para conectar bases de datos.....	76
6.7 Departamento de Servicios Auxiliares.....	77
6.7.1 Bases de datos.....	77
6.7.2 Aplicaciones de las bases de datos.....	77
6.7.3 Tecnologías de acceso a los datos.....	78
6.7.4 <i>Middleware</i> empleado para conectar bases de datos.....	78
<b>Capítulo 7 Definición de una estrategia de Integración de Aplicaciones Empresariales para sistemas heredados dentro del nivel de datos.....</b>	<b>80</b>
71 Situación actual de la organización.....	80
72 Desventajas actuales.....	81
73 Estrategia de implementación del nivel de datos.....	82
7.3.1 Identificación de los datos.....	83
7.3.2 Categorización de los datos.....	84
7.3.3 Elaboración del modelo de metadatos.....	84
7.4 Estrategia de selección del <i>middleware</i> .....	84
7.5 Implementación de la estrategia de EAI.....	91
<b>Capítulo 8. Conclusiones y Trabajos Futuros.....</b>	<b>97</b>
8.1 Conclusiones.....	97
8.2 Trabajos futuros.....	98
<b>Anexo I. Cuestionario de apoyo para las entrevistas.....</b>	<b>100</b>
<b>Anexo II. Resultados generales de las entrevistas.....</b>	<b>104</b>
<b>Anexo III. Tabla de apoyo para la selección de <i>middleware</i>.....</b>	<b>107</b>
<b>Anexo IV. Glosario.....</b>	<b>108</b>
<b>Referencias Bibliográficas.....</b>	<b>111</b>
<b>VITA.....</b>	<b>116</b>

## LISTA DE FIGURAS

---

Figura 2.1 Evolución de los sistemas heredados.....	10
Figura 2.2 Niveles de Integración.....	19
Figura 2.3 Arquitectura de aplicaciones en la actualidad.....	20
Figura 3.1 Tareas Funcionales de la Integración por Interfaces.....	47
Figura 6.1 Comunicación actual entre los servidores de aplicaciones de los sistemas satélites.....	69
Figura 6.2 Diagrama de carga de datos de la base de datos oficial a bases de datos alterna.....	80
Figura 6.3 Diagrama de conexión entre Banner® y Oracle®.....	80
Figura 7.1 Modelo punto a punto.....	88
Figura 7.2 Modelo Publicación/Suscripción (Pub/Sub) .....	89
Figura 7.3 Modelo de colas de mensajes.....	91
Figura 7.4 Diagrama de una infraestructura de EAI.....	93
Figura 7.5 Diagrama de la infraestructura actual de la organización.....	94
Figura 7.6 Diagrama propuesto de EAI para la organización.....	95

## LISTA DE TABLAS

---

Tabla 1.1 Niveles de evolución de las empresas en el Web.....	3
Tabla 4.1 <i>Middleware</i> de RPC.....	51
Tabla 4.2 <i>Middleware</i> de MOM.....	52
Tabla 4.3 <i>Middleware</i> orientado a bases de datos.....	53
Tabla 4.4 <i>Middleware</i> orientado a objetos distribuidos.....	54
Tabla 4.5 <i>Middleware</i> de monitor de procesamiento de aplicaciones.....	55
Tabla 4.6 <i>Middleware</i> de servidor de aplicaciones.....	56
Tabla 4.7 Caso de estudio de Chevron Corporation.....	57
Tabla 4.8 Caso de estudio de Whirlpool Corporation.....	58
Tabla 4.9 Caso de estudio de CBS SporstLine.....	58
Tabla 4.10 Caso de estudio de 3M Health Information Systems.....	59
Tabla 6.1 Módulos del sistema de información utilizados por cada departamento.....	66

# CAPITULO 1

## INTRODUCCIÓN

---

Sin duda, en los últimos años Internet se ha vuelto una herramienta tan común en muchas empresas e instituciones, que actualmente forma parte de su estrategia corporativa. Uno de los aspectos más interesantes de la aparición de esta tecnología es la profusión y alcance de las oportunidades y las amenazas que esto supone. Es en este aspecto donde se encuentra la ventaja competitiva de algunas industrias pero también el declive de la misma empresa (Bickerton, 2000)

Uno de los principales puntos en la planeación estratégica de muchas empresas radica en la promoción y venta de sus bienes o servicios a través de la red. Este concepto recibe el nombre de Comercio Electrónico. Una definición práctica (Kalakota, 1997) desde un punto de vista de los negocios, consiste en la aplicación de tecnología dedicada a la automatización de las transacciones de los negocios y los flujos de trabajo. Otra forma de ver el comercio electrónico es como un proceso de producción que transforma entradas digitales en salidas con valor agregado a través de un conjunto de intermediarios. Con respecto a los modelos, existe una gran variedad de clasificaciones del comercio electrónico; sin embargo una de las más aceptadas actualmente es la que distingue entre empresas (*Business*) y compradores (*Consumers*). De ahí se deriva una gran variedad de modelos como son B2C (*Business to Consumers*), B2B (*Business to Business*), C2B (*Consumers to Business*), C2C (*Consumers to Consumers*).

A pesar de lo que se ha venido mencionando sobre el gran avance que ha experimentado el comercio electrónico, existen ciertos problemas para la completa y total implementación de este rubro a un nivel internacional. La protección legal es uno de los mayores frenos para un desarrollo global, así mismo la privacidad, la seguridad y la estandarización técnica de los protocolos, son otras de las áreas que necesitan un poco más de avances. (Cashin, 1999a) De acuerdo a esta información, en 1997, el gobierno de los Estados Unidos publicó un conjunto de principios operativos con respecto al Comercio Electrónico. “Una guía para el Comercio Electrónico Global”, es un

documento autorizado por el Consejo de Seguridad Nacional/Consejo Económico Nacional. (White House, 1997)

Ahora bien, el comercio electrónico en Internet no debe ser visto como la única solución a los objetivos de aumento en las ventas de una empresa. La clave para el uso exitoso del Internet es verlo como una expansión de la empresa (Ranadivé, 1999). La mejor manera para llevar esto a cabo es crear una plataforma dinámica para el contenido y los servicios de la compañía más que un sitio Web. Una definición concreta de una corporación electrónica podría ser la que se menciona en la revista *Fortune* en el artículo “*The E-Corporation*”: “Una verdadera empresa electrónica no es aquella que usa exclusivamente Internet para acercarse a sus clientes y a su mercado; consiste en una combinación de equipos de cómputo, el Web y los programas complejos conocidos como software empresarial que hacen que cambie la manera en que opera la empresa” (Hamel, 1998).

Sin embargo, el desarrollo de una empresa de comercio electrónico dentro de una empresa no es un paso fácil y por lo general se requiere de bastante capital para llevar a cabo un proyecto de este tipo. Es necesario avanzar de acuerdo a las necesidades del negocio al igual que a la capacidad que éste tenga para reaccionar ante sus clientes. De acuerdo a esto, *The Gartner Group*, mencionado por Rational (2000), ha definido cuatro niveles de evolución del Web en las empresas, como se detallan en la Tabla 1.1

**Tabla 1.1 Niveles de evolución de las empresas en el Web [Fuente: Rational Software Corporation]**

Nivel	Proceso
Presencia Básica	La empresa despliega información sobre la compañía, sus productos y servicios en un sitio Web. Las páginas son estáticas y limitadas en número.
Prospección	La empresa añade más páginas informativas sobre la empresa y sus productos. Los visitantes pueden interactuar con el sitio para solicitar información adicional a través de formularios o búsquedas por palabras clave.
Integración del negocio	El sitio permite transacciones de comercio electrónico y capacidades de búsquedas inteligentes. Provee mejores servicios a los clientes incluyendo la creación y entrega de información, soporte y colaboración como son: conferencia en línea, creación de páginas web, entre otras.
Transformación del negocio	El sitio Web cambia cada aspecto de los procesos de la empresa y llega a ser parte integral del negocio. Los sitios Web, en este nivel de desarrollo, ofrecen un intercambio electrónico de documentos con los proveedores, servicios de integración con los vendedores y servicios de personalización de contenido para los visitantes.

Es este último punto donde se realiza la conversión total del negocio gracias al Web. Se puede decir que el sitio de la empresa ha pasado, de proporcionar información a ser un completo rediseñador de la estructura interna de la misma. De esta manera, la idea inicial de ver el sitio como una parte del negocio, viene a transformarlo en un ambiente global, para llegar al siguiente paso que se conoce como negocio electrónico, el cual desde luego tiene un alcance mucho mayor. Las diferencias entre el comercio y el negocio electrónico son muchas. Podemos encontrar que, mientras el comercio electrónico se refiere a una transacción electrónica a través de Internet o de un sistema de comercio dedicado, tipo EDI (*Electronic Data Interchange*), el negocio electrónico se refiere a los procesos complejos de la empresa y el flujo de información entre el cliente, la empresa y los proveedores, con el fin de llevar a cabo transacciones CTO (*configure-to-order*) o a través de las transacciones BTO (*build-to-order*). A través de un sistema de negocio electrónico es posible, además de tomar la orden de un pedido, permitirle al cliente una personalización de acuerdo a sus necesidades y ofrecer una fecha de entrega exacta (Adexa, 2000).

Sin embargo, mantener un negocio electrónico requiere de una planeación estratégica tal, que no es posible que se escape ningún detalle. Como resultado de esto, es necesario una integración de todas las áreas de la empresa, de igual forma se deben mantener relaciones muy estrechas tanto con los proveedores como con los clientes. Con respecto a la integración interna, es necesario contar con la suficiente tecnología que permita a la empresa obtener ganancias desde el primer momento en que se empiece a desarrollar el negocio electrónico. El advenimiento de los servidores de aplicaciones y el beneficio de las tecnologías cada vez más accesibles, eliminan los límites de desarrollo que anteriormente eran prohibitivos en todos los ambientes Web. Las instituciones pueden hacer uso de estas nuevas tecnologías para lograr un alcance mucho mayor del negocio (Peacock, 2000).

Actualmente muchas corporaciones implementan una gran variedad de aplicaciones a lo largo de la organización con el fin de soportar la mayoría de sus procesos. Más aún, estos procesos evolucionan rápidamente hacia otros que requieren de nuevos datos y métodos. Esto ha creado una demanda para integrar todas esas aplicaciones dentro de un proceso o modelo de datos unificado. Con este objetivo puesto en mente, se ha desarrollado una metodología denominada EAI que permite compartir tanto datos como procesos dentro de la institución. El interés de aplicar este modelo dentro de las instituciones tiene varios factores claves, como son la creciente competitividad existente entre las empresas así como la integración de aplicaciones para reducir costos de desarrollo lo cual permite compartir información dentro de la empresa o con las instituciones asociadas. (Linthicum, 2000a)

La falta de una aplicación adecuada significa que un gran número de sistemas de información existentes no se puedan coordinar para compartir información, esto trae como resultado una tremenda pérdida de ingresos, por lo cual la inclusión de EAI dentro de la mayoría de las empresas tiene el potencial de recuperar esos ingresos. De tal manera, EAI está diseñado para reducir los costos a las empresas. Ese tipo de ahorro proviene, ya sea de la automatización de procesos que actualmente se están completando a través de dispositivos mecánicos, o de la concepción de todos los sistemas empresariales como un único sistema virtual con acceso a todos los datos relevantes para



soporte a las decisiones. La razón de la integración de las aplicaciones empresariales se debe a que las arquitecturas se han planeado de forma equivocada durante las últimas dos décadas o más. Muchas compañías construyen sistemas basados en la tecnología "de moda", dejándose llevar por lo más popular, sin considerar la manera en que esos sistemas podrían compartir información en el futuro. (Linthicum, 2000b).

## **1.1 OBJETIVO**

El modelo de EAI (*Enterprise Application Integration*) puede ser dividido para su implementación en 4 niveles cada uno con sus respectivas fases y herramientas: datos, interfaz, métodos y usuarios. Estos niveles deben ser llevados a cabo paso a paso para lograr una correcta integración de los sistemas informáticos.

El objetivo de esta tesis se basa en dicha metodología, y consiste en analizar los sistemas de información dentro de una institución para determinar el estado en el que se encuentran los sistemas heredados o *legacy systems* y los sistemas desarrollados con nuevas tecnologías. Esto con el fin de proponer una estrategia basada en EAI que apoye la integración de dichas aplicaciones y permita llevar a cabo una transformación de la organización hacia un negocio electrónico.

## **1.2 ALCANCE DEL ESTUDIO**

El estudio pretende cubrir las funciones realizadas por los diferentes sistemas de información que empleen bases de datos dentro de una organización. Para ello, la tesis se limitará a los siguientes puntos:

- La investigación se basará exclusivamente en la metodología EAI dentro del nivel de datos sin realizar comparativas con otro tipo de metodologías ni penetrar en otros niveles de integración.

- El estudio no pretende enfocarse a procesos externos de las instituciones sino exclusivamente a sus procesos internos.
- El objetivo de la tesis es ofrecer una propuesta que sirva como una estrategia de apoyo para la integración de sistemas y no como una solución a los problemas que puedan existir en algún momento dado.

De esta manera, la investigación permitirá identificar con mayor precisión las áreas donde se requiera integración en el nivel de datos y que sirva como base para desarrollos posteriores.

### **1.3 PRODUCTO FINAL**

El resultado final esperado de esta investigación es el de ofrecer una estrategia de EAI para realizar una integración de los sistemas de una institución basada en los parámetros definidos por el modelo. Se espera que la aportación final sirva de referencia para la integración de aplicaciones en otras instituciones que sean similares en características a las de la unidad de estudio

### **1.4 ORGANIZACIÓN DE LA TESIS**

La presente tesis se divide en siete capítulos como se mencionan a continuación:

**Capítulo 1.** Introducción.- Trata el tema de la tesis y el producto final que se espera entregar como resultado del análisis llevado a cabo en la investigación de campo.

**Capítulo 2.** Revisión Bibliográfica: Conceptos básicos de EAI.- Incluye los aspectos fundamentales de la integración de aplicaciones empresariales y la clasificación de las diferentes maneras de integración.

**Capítulo 3. Revisión Bibliográfica: Metodologías para la EAI.-** Desarrolla los temas relacionados con las tecnologías actuales para integrar aplicaciones, el *middleware* necesario para integrar diferentes aplicaciones, así como algunas sugerencias para evaluar la adquisición de *middleware*.

**Capítulo 4. Soluciones comerciales y casos de estudio.-** Se desarrollan ejemplos reales y los diferentes productos de *middleware* existentes en el mercado.

**Capítulo 5. Investigación de campo.-** Este capítulo trata la metodología empleada para recopilar información en una organización con el fin de sugerir una estrategia de EAI.

**Capítulo 6. Análisis de resultados.-** En este capítulo se detalla la información recopilada en cada uno de los departamentos de la organización especificando la importancia y los elementos clave detectados en cada uno de ellos.

**Capítulo 7. Estrategia de Integración de Aplicaciones Empresariales para los sistemas heredados de una organización.-** Este capítulo es el más importante debido a que integra toda la información bibliografía con la información obtenida en la investigación de campo para sugerir una estrategia de EAI que permita a la organización un funcionamiento mas óptimo en cuanto a sus procesos internos.

**Capítulo 8. Conclusiones y Trabajos Futuros.-** En este capítulo se establece una conclusión sobre la base de la información sugerida además de establecer la pauta para desarrollos futuros dentro de la organización.

## **1.5 METODOLOGÍA UTILIZADA**

La presente tesis se llevará a cabo utilizando la metodología cualitativa, empleando la entrevista con el apoyo de un cuestionario como herramienta de recolección de datos. Para llevar a cabo esto, se entrevistará a las personas encargadas y administradores de los

departamentos de informática con el fin de obtener la mayor información posible en el manejo de los sistemas de información y la información contenida en la base de datos

A grandes rasgos, la información que se pretende recopilar servirá para identificar el uso que se hace de la información en las bases de datos, las aplicaciones empleadas para acceder a los datos y la manera en que se tienen integrados estos datos con otras aplicaciones al igual que con otras bases de datos.

Finalmente, es importante recalcar que la presente tesis propone una estrategia de integración de aplicaciones sobre un estudio realizado en una organización en particular involucrando áreas muy específicas de dicha institución. Dado esto, si se requiere emplear la metodología en otra organización, es primordial la adecuación de la estrategia propuesta con base en las necesidades identificadas por la empresa.

### REVISIÓN BIBLIOGRÁFICA: CONCEPTOS BÁSICOS DE EAI

---

De acuerdo a la clasificación de los sistemas de cómputo definida por la Asociación de Maquinaria de Cómputo (ACM, por sus siglas en inglés), los sistemas de información pueden ser clasificados en: generales, modelos y principios, administración de bases de datos, almacenamiento y recuperación de información, aplicaciones de sistemas de información, interfaces y presentación de información, y sistemas misceláneos. En esta misma clasificación se encuentran otro tipo de sistemas denominados aplicaciones computacionales los cuales involucran: procesamiento de datos administrativos, ingeniería y ciencias físicas, ciencias médicas y de la vida, ciencias sociales y del comportamiento, arte y humanidades, ingeniería asistida por computadora, y otros sistemas (ACM, 1998). Sin embargo, todo estos sistemas pueden o no formar parte de un sistema mucho mayor por lo cual una clasificación más apta para del desarrollo de la tesis es clasificar los sistemas como integrados o no integrados.

En este capítulo se analizarán a detalle los sistemas no integrados, también denominados sistemas heredados o *legacy systems*, los sistemas integrados y la metodología para llevar a cabo la integración de aplicaciones empresariales. Dicha metodología puede ser vista desde varios niveles sobre los cuales se profundiza así como también se hace notar la diferencia entre cada uno con el fin de seleccionar el que mejor se apegue a las necesidades de integración de cualquier organización. Como conclusión, se describe el proceso para llevar a cabo una integración en el nivel de datos con una metodología concisa y bien estructurada.

#### 2.1 SISTEMAS HEREDADOS (*Legacy Systems*)

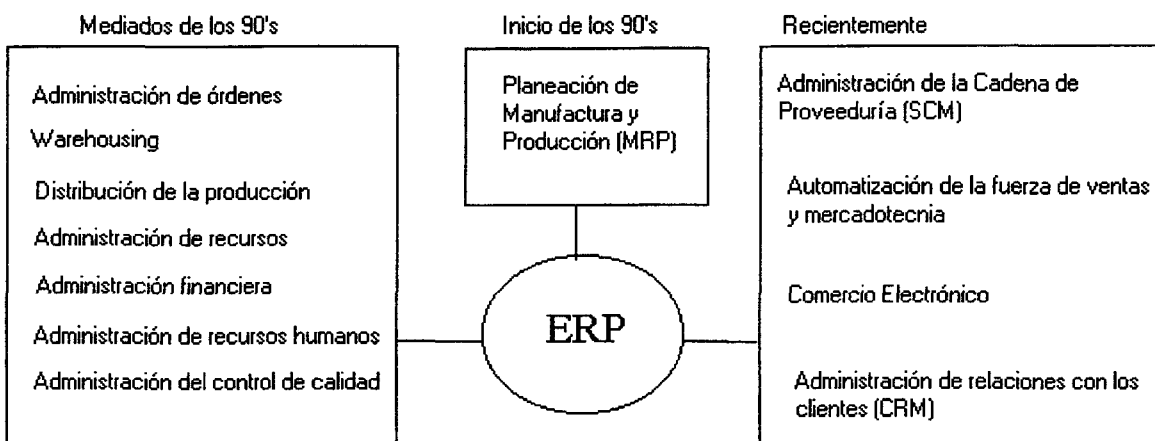
Rossmann comenta que los sistemas heredados forman parte del pasado de la informática sin embargo, estos sistemas no necesariamente se utilizaban en los *mainframes* como se cree en la actualidad. La definición técnica de un sistema heredado es cualquier sistema central de cómputo que es indispensable para mantener a una empresa operando

(Rossman, 2000). En este punto se puede pensar también que una empresa puede adquirir e instalar un nuevo sistema de cómputo y será, a pesar del grado de complejidad que tenga, un sistema heredado si no tiene una conexión con los demás sistemas existentes dentro de la empresa. Asimismo, estos sistemas incluyen tanto la base de datos como los procesos organizacionales en la misma aplicación.

**Se entiende como proceso organizacional a cualquier regla o lógica que existe dentro de una empresa y que tiene un efecto en cómo la información es procesada. (Linthicum, 2000)**

Un problema que surge con los sistemas heredados es que el Internet y muchas otras tecnologías actuales no eran importantes cuando fueron desarrollados. A menudo, sólo una pequeña parte de la funcionalidad de un sistema heredado dentro de la organización necesita ser incluido en una nueva aplicación. Desgraciadamente, la información que pueda emplearse con los demás sistemas puede no ser tan fácil de extraer y de reutilizar.

Hay que tomar en cuenta la integración de sistemas heredados con sistemas desarrollados en los últimos años debido a que existe el peligro que las organizaciones se encuentren construyendo sistemas heredados perpetuos. Según se muestra en la Figura 2.1, los sistemas heredados en la actualidad pueden ser integrados en un ERP.



**Figura 2.1: Evolución de los sistemas heredados. [Fuente: Zhang Chundi, 2000]**

Los problemas con este tipo de esquema de integración son mucho más complicados cuando el código de integración, como es el protocolo, la conversión de datos y la transmisión de mensajes entre aplicaciones, se encuentran incluidos dentro de la nueva aplicación. Con respecto a las desventajas existentes en los sistemas heredados, Mordacq (2000) comenta lo siguiente:

1. El costo de cambiar el sistema heredado es demasiado alto ya que se ha hecho una inversión de dinero, tiempo y entrenamiento.
2. La alta dificultad de vencer la resistencia al cambio de los usuarios que llevan tiempo empleando las aplicaciones.

A diferencia de este tipo de aplicaciones, últimamente se han venido implementado soluciones del tipo ERP con el objetivo de centralizar todos los procesos de la empresa en un sistema global. A continuación se mencionan las características de este tipo de aplicaciones.

## **2.2 ENTERPRISE RESOURCE PLANNING (ERP)**

La idea básica detrás de un ERP consiste en la optimización de todos los recursos disponibles y la integración de todos los procesos organizacionales. Chundi Zhang añade que, un ERP es un sistema de administración organizacional que integra todos los aspectos de la empresa incluyendo planeación, manufactura, ventas, mercadotecnia, finanzas, etc. Según Zhang, el término ERP fue originado a principios de los años 90 para el software que evolucionó de MRP (*Materials Requirement Planning*) y PPS (*Production Planning System*), teniendo sus orígenes en las industrias de manufactura. El aspecto primordial que debe prevalecer en un ERP es una base de datos y las herramientas de manipulación de datos que permiten almacenar, procesar y presentar la información de una manera apropiada cuando sea requerida por el usuario. (Zhang, 2000). Sin embargo, estas aplicaciones tenían como característica principal ser tecnologías de “*back-office*”, cuya función principal es mantener el control y la eficiencia

de los sistemas de manufactura de las industrias antes mencionadas; a diferencia de los actuales ERP que funcionan en su mayoría como “*front-office*” e incluso cuentan con sistemas de soporte al comercio electrónico. Utilizando un ERP es posible compartir información a través de la organización, asimismo se puede optimizar la comunicación y eliminar relaciones costosas o complicadas entre los diversos sistemas computacionales en diferentes áreas de la empresa.

La diferencia principal de los ERP con los sistemas tradicionales es que estos últimos tratan cada transacción por separado con funciones específicas, mientras que los ERP consideran cada transacción como parte de un proceso global vinculado a la organización.

Los ERP son, en la actualidad, motivo de una gran demanda por parte de la mayoría de las grandes empresas. Sin embargo, existen varias amenazas que pueden afectar la decisión de utilizar un sistema de este tipo. Primero, en algunos casos el costo de instalación, implementación, mantenimiento y entrenamiento es demasiado alto. Segundo, el tiempo de puesta en marcha para ofrecer servicio al mercado al que se dirige es largo. Tercero, puede tener problemas de actualización en el medio ambiente cambiante de las organizaciones, tal como el surgimiento de situaciones inesperadas o un cambio radical de estrategia operacional dentro de la organización. Una desventaja de este tipo de sistemas es que debido a su fuerte integración interna hace complicada para los vendedores del software la tarea de conectar módulos con otros alternos para aprovechar la tecnología existente (Mckie, 2001)

A pesar de las situaciones existentes en la implementación de un sistema ERP, estas aplicaciones son las más solicitadas por las empresas en cuestión de automatización de procesos. Connolly (1999) comenta que los beneficios que la gente busca en un ERP son la estandarización de procesos empresariales, la construcción de una base de datos concisa y eliminar las complejidades o gastos así como algunos problemas con los sistemas heredados tales como los del año 2000, entre otros que van surgiendo.



Sin embargo, para las organizaciones que cuentan con sistemas heredados y no tienen la capacidad de implementar un ERP dentro de su organización debido al alto costo que esto implica, es primordial que estudien la posibilidad de desarrollar una estrategia de integración de aplicaciones empresariales que les permita utilizar la infraestructura actual con el fin de ahorrar costos y en cierta forma mantener su información centralizada.

A continuación se define el concepto de EAI, así como los diferentes niveles en los que puede ser implementado dentro de los sistemas de información de una organización. Se definen las características de cada nivel y los requerimientos necesarios para la implementación del nivel de datos sobre el cual está centrado el objetivo de esta tesis.

### **2.3 *ENTERPRISE APPLICATION INTEGRATION (EAI)***

La Integración de Aplicaciones Empresariales (EAI, por sus siglas en inglés) se define como la compartición no restringida de datos y procesos organizacionales entre las aplicaciones o fuentes de datos dentro de la empresa (Linthicum, 2000). Con el mismo enfoque, The Butler Group define EAI como el requerimiento para integrar, dentro de nuevos procesos organizacionales, las funciones actuales o reglas de los sistemas así, como los componentes que lo forman y los conjuntos de datos que soportan todos estos procesos (Wilkes, 1999). EAI también puede ser definido como un conjunto de tecnologías que impulsan el intercambio de datos entre diferentes aplicaciones, bases de datos y procesos organizacionales dentro de una institución. (Kalafsky, 2000)

Basándonos en estas tres acepciones podemos encontrar que la integración de aplicaciones empresariales consiste en la necesidad de compartir los datos y procesos organizacionales tomando en cuenta no realizar demasiados cambios a las aplicaciones o a las estructuras de datos actuales. Aunque el enfoque inicial de EAI se encuentra en el nivel de datos a través de mover o replicar datos entre bases de datos, es posible ver que la evolución de EAI se viene dando claramente hacia la automatización de procesos organizacionales.

La mayoría de las organizaciones cuentan con una variedad de sistemas implementados para un proceso similar. Por ejemplo, no es muy frecuente que exista un sistema global que mantenga toda la información que se procesa en una organización. Es decir, puede existir toda una variedad de sistemas que recopilen información de diferentes procesos (producción, ventas, inventario, etc.) y se almacene en la base de datos de cada sistema. Ahora bien, una tarea de integración consiste en consolidar estas fuentes de datos en una sola vista. Sin embargo, esto puede parecer sencillo al querer reemplazar o emplear estos sistemas con un componente compartido, pero el esfuerzo para hacer esto puede ser considerablemente costoso y complejo como en el caso de los sistemas heredados o aplicaciones empaquetadas, como los ERP. Con respecto a estos sistemas, uno de los mayores retos de EAI es proveer una conexión entre ellos sobre todo en los sistemas heredados donde no es posible cambiar la configuración tan fácilmente (Linthicum, 2000)

Las aplicaciones empresariales que pueden ser integradas están basadas generalmente en diferentes tecnologías, siendo algunas de ellas desarrollos internos, mientras que otras aplicaciones pueden ser adquiridas por fuera. Butler Group categoriza estas fuentes en cinco principales:

- ❖ **Paquetes de “*back-office*”**: hace mención de los paquetes que automatizan la infraestructura común de la organización. Un ejemplo son los ERP o los sistemas de Recursos Humanos y Contabilidad Financiera.
- ❖ **Paquetes de “*front-office*”**: son paquetes empleados para automatizar los procesos organizacionales orientados hacia el consumidor. Un ejemplo son los Portales Electrónicos.
- ❖ **Sistema heredados**: se refiere a las aplicaciones desarrolladas o adquiridas por la empresa que en la actualidad pudieran considerarse obsoletas.
- ❖ **Nuevos desarrollos o actualizaciones**: son todas las nuevas aplicaciones o componentes desarrollados para la organización.
- ❖ **Servicios ofrecidos por empresas aliadas**: Son fuentes de información o procesos llevados a cabo por otras empresas, como es el caso de la autorización de crédito por parte de un banco.

El interés en EAI es conducido por unos cuantos factores claves. En primer lugar, la presión de un ambiente empresarial donde la administración de las tecnologías de información se está moviendo hacia aplicaciones con un ciclo de vida cada vez más corto. En lugar de crear los mismos procesos organizacionales y repositorios de datos una y otra vez, los administradores de tecnologías de información en las empresas están aprendiendo a reutilizar aplicaciones existentes y aplicaciones de servicios. Además, la integración de aplicaciones, con el fin de ahorrar costos en el desarrollo, ofrece una ventaja competitiva a las corporaciones que necesitan compartir aplicaciones, ya sea dentro de la misma empresa o con las empresas que mantengan alianzas estratégicas. Adicionalmente, los nuevos procesos organizacionales indican que esta integración debe ser realizada en tiempo real. Otra diferencia más radica en que la integración debe darse en doble sentido, es decir la información debe ser creada y actualizada en todos los sistemas integrados (Wilkes, 1999).

Cuando se empieza a pensar sobre la implementación de EAI dentro de una empresa, es necesario entender la suma y el contenido de los procesos organizacionales, así como los datos en la empresa. El departamento de informática también debe entender cómo estos procesos organizacionales se encuentran automatizados (o en algunos casos no automatizados) y la importancia de todos ellos. Esto podría llevar consigo una carga de trabajo significativa, por lo que muchas empresas contratan consultores a precios muy elevados para determinar y definir los procesos organizacionales relevantes de toda la compañía. Igualmente, muchas organizaciones buscan nuevas metodologías que los ayuden en este proceso. Sin duda, el primer paso para entender el verdadero valor dentro en la empresa al implementar una metodología de EAI consiste en comprender los procesos organizacionales con bastante detalle. Así, con esta información, será posible determinar qué nivel de integración debe llevarse a cabo para optimizar dichos procesos.

Para medir el valor de la implementación de EAI dentro de la empresa es necesario definir un conjunto de métricas o medidas de éxito para la organización. Se puede ver la medición de este valor como un proceso donde se refiere a observar el estado actual de la empresa y medirlo (Linthicum, 2000a). Para esto se pretende considerar el estado futuro deseado y la cantidad de esfuerzo que tomará para alcanzar ese estado. En el mundo de

EAI esto significa determinar el valor del esfuerzo de la integración y el impacto en las demás personas que trabajan en la empresa. Sin embargo, cuando se ha llegado a un conjunto de métricas para obtener el valor de EAI, estas mediciones deben ser ajustadas significativamente para que tomen en cuenta todos los factores que pueden surgir en un determinado tipo de empresa. Un ejemplo claro de esto puede ser visto en compartir información del sistema de inventarios de una empresa con el sistema de ventas; esto produce que el departamento funcione de manera más efectiva; pero mientras no haya una disminución de errores en el control de inventario, la integración tiene muy poco valor si se inclina hacia un solo lado exclusivamente.

En resumen, las organizaciones deben comprender a la perfección la información que manipulan así como los procesos organizacionales de la empresa y seleccionar cuál de ellos requiere el más alto nivel de prioridad para su integración. La falta de arquitecturas tecnológicas comunes es lo que hace la integración compleja en la mayoría de los casos. Esto no se refiere a la diferencia en los formatos de los datos, ni a las interfaces específicamente sino a la falta de la definición de un concepto organizacional sobre el cual debe encontrarse las diferentes fuentes para poder ser integradas.

La Integración de Aplicaciones Empresariales puede, en muchos casos, ser un proceso que modifique tanto la misma organización como los procesos organizacionales involucrados. Los proyectos de integración pueden obtener un buen retorno de la inversión a través del tiempo. EAI ofrece a las organizaciones una capa intermedia para al administración del flujo de datos entre ERP, aplicaciones heredadas y del usuario final, además de reducir costos futuros de actualización del software. (Gonsalves, 2001)

Con el fin de identificar dónde es necesario una reorganización, la mayoría de los autores identifican varios niveles de integración los cuales analizaremos con el fin de determinar el impacto que cada uno causa en la integración de aplicaciones.

## 2.4 CLASIFICACIÓN DE LOS DIFERENTES NIVELES DE EAI

Hurwitz Group identifica cinco categorías jerárquicas de integración. Estas divisiones son (mencionado por Ren, 2000):

- **Integración de Plataformas:** provee conectividad entre el hardware heterogéneo y diferentes sistemas operativos, además de ofrecer tecnología que permita una integración de plataformas, como el envío de mensajes y los ORBs (*Object Request Brokers*).
- **Integración de Datos:** este tipo de integración puede ser de dos tipos: el primero es el acceso a bases de datos heterogéneas a través de SQL; y el segundo consiste en la extracción, transformación, movimiento y carga de datos en la base de datos receptora.
- **Integración de Componentes:** permite una funcionalidad que puede ser fácilmente combinada con paquetes de ERP y aplicaciones heredadas de arquitectura cliente/servidor.
- **Integración de Aplicaciones:** ofrece una guía para una colección de tecnologías que ofrezcan integración en tiempo real. Esta guía incluye tecnologías de integración de plataformas a través de mensajes que ofrecen traducción de datos, transformación y reglas para el ruteo de datos.
- **Integración de Procesos Empresariales:** permite a los administradores la definición, monitoreo y cambios en los procesos empresariales a través de una interfaz de modelo gráfico. Este tipo de integración requiere de todos los anteriores.

Asimismo, Ruh (2001) ofrece otro modelo de EAI. Se clasifica de la siguiente manera:

- **Integración de la presentación:** en este modelo se busca la integración de diferentes componentes de software a través de una aplicación de interfaz de usuario. Este modelo permite integrar nuevo software a través de componentes de conexión contenidos en un software heredado.

- **Integración de datos:** permite la integración de software a través del acceso a los datos que son creados, administrados y almacenados por el software con el fin de ser reutilizados o sincronizados entre varias aplicaciones.
- **Integración funcional:** permite la integración de software con el fin de reutilizar o emplear la funcionalidad de aplicaciones nuevas o existentes. Este tipo de integración se realiza a través de interfaces de programación del software (API, por sus siglas en inglés).

Sin embargo, una de las divisiones más documentadas y que ofrece una mayor cobertura en la mayoría de los sistemas es la que ofrece Linthicum, la cual se puede clasificar en cuatro niveles o dimensiones generales (figura 2.2). Cada una de ellas tiene su propia metodología de integración la cual se mencionará más adelante. Los niveles de EAI son los siguientes (Linthicum, 2000):

#### **2.4.1 Nivel de datos**

Se refiere al nivel donde se involucran todos los procesos organizacionales al igual que las tecnologías para transportar datos entre varias aplicaciones. Un ejemplo práctico sería simplemente extraer la información de una base de datos, procesarla de acuerdo a las necesidades del usuario y actualizar la misma información en otra base de datos.

#### **2.4.2 Nivel de interfaz de aplicación**

Este nivel se refiere a la práctica de impulsar las interfaces ofrecidas por las aplicaciones heredadas con el fin de ser empleadas por otras aplicaciones. Una forma de llevar a cabo este aprovechamiento de las aplicaciones consiste en acceder, tanto a los procesos organizacionales, como a la información contenida en ellos.

#### **2.4.3 Nivel de método**

Consiste simplemente en compartir la lógica que pueda existir entre los procesos de la organización que se realizan a través de los sistemas. Por ejemplo, las aplicaciones empresariales pueden acceder a través de un método para actualizar la información de un cliente.

#### 2.4.4 Nivel de interfaz de usuario

En este nivel los desarrolladores son capaces de conectar aplicaciones usando las interfaces del usuario como punto de integración. Por ejemplo, las aplicaciones de *mainframe* que no proveen acceso a las bases de datos ni a los procesos organizacionales pueden ser accedidos a través de la interfaz de usuario.

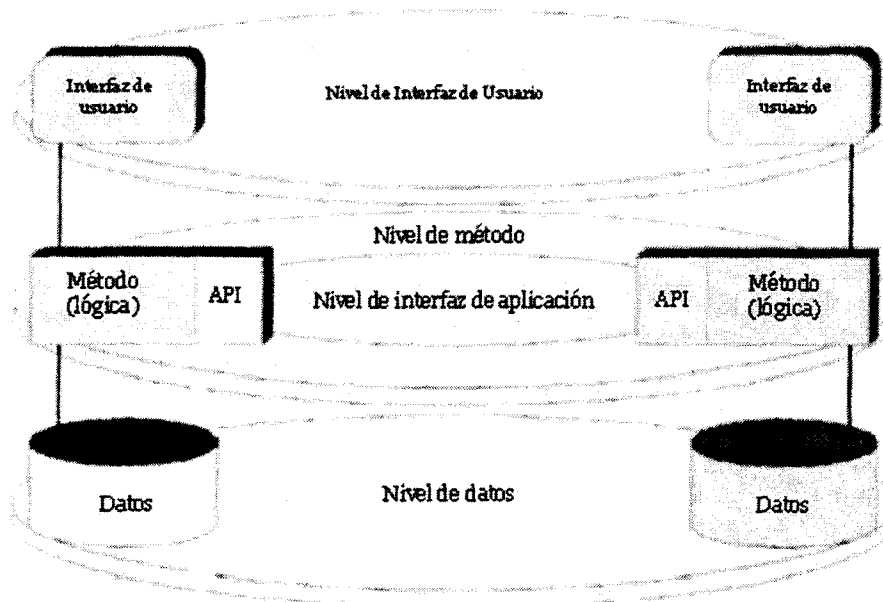


Figura 2.2. Niveles de Integración. [Fuente: Linthicum, 2000]

Cada nivel de la integración de aplicaciones empresariales cuenta con su propio esquema de integración. Cada uno puede aplicarse de forma independiente pero de la misma manera si se está buscando una integración completa, se puede partir desde la integración del nivel de datos y ascender al siguiente nivel una vez que se haya completado el anterior hasta concluir con el nivel de interfaz de usuario.

A continuación, se verán las características de cada nivel para identificar los requerimientos, tanto tecnológicos como organizacionales, que involucran cada uno de ellos en la integración de aplicaciones empresariales.

## 2.5 CARACTERÍSTICAS DEL NIVEL DE DATOS

El nivel de datos en EAI provee una ventaja en cuanto a facilidad de uso y velocidad de acceso. Esto se debe a que la lógica de la aplicación rara vez tiene que ser alterada, por lo cual no existe el riesgo o gasto de implementar nuevas versiones de la aplicación dentro de la empresa.

La principal diferencia entre el nivel de datos con los otros niveles consiste en que acceder a una base de datos es una tarea que puede ser llevada a cabo con pocos cambios a la lógica de la aplicación. En general, las aplicaciones empresariales no fueron diseñadas para ser compatibles con todos los sistemas y cumplir la metodología de EAI, sino que fueron creadas para trabajar independientemente. Como resultado de esto, una implementación exitosa del nivel de datos requiere descubrir la lógica de la aplicación y extraer o cargar los datos directamente a la base de datos a través de su interfaz original. Afortunadamente, la mayoría de las aplicaciones creadas en las últimas dos décadas separan la interfaz de usuario de la lógica de aplicación así como la información de la base de datos (figura 2.3), ofreciendo así una tarea relativamente sencilla.

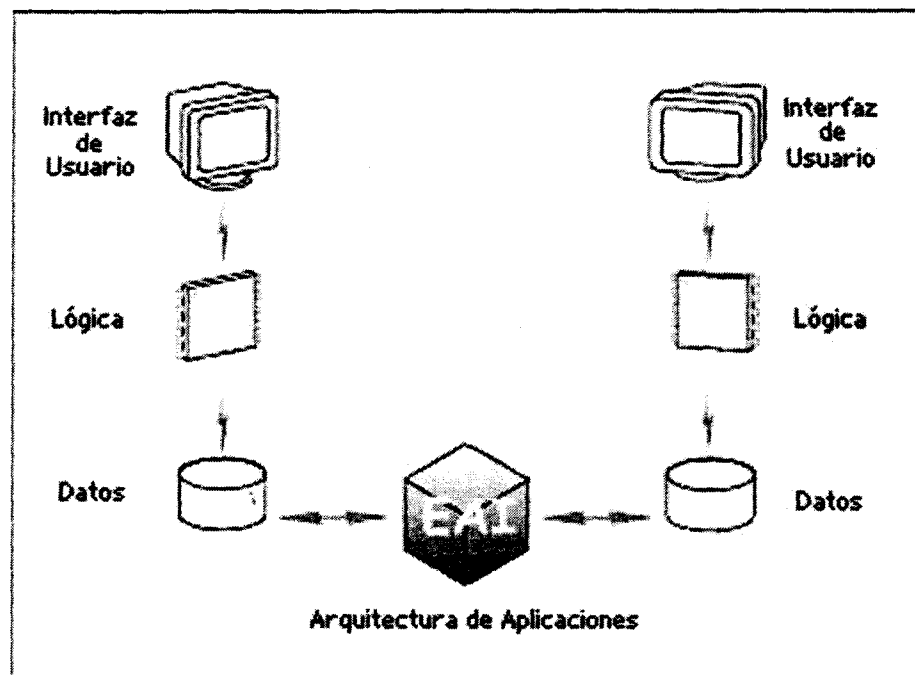


Figura 2.3: Arquitectura de aplicaciones en la actualidad Fuente: [Linthicum 2000]



A grandes rasgos la integración de aplicaciones empresariales dentro del nivel de datos supone una metodología mucho más fácil de implementar que las de los otros niveles. Muchas organizaciones usan esta técnica debido a que no requieren un cambio mayor en los sistemas de información lo que, de otra manera, los llevaría a modificar la infraestructura actual que soporta todas las tecnologías de información existentes.

## **2.6 CARACTERÍSTICAS DEL NIVEL DE INTERFAZ DE APLICACIÓN**

Las interfaces de aplicación se pueden definir como interfaces tomadas de una aplicación que son usadas por los desarrolladores para obtener acceso a varios niveles o servicios de la misma aplicación. Estas interfaces permiten, en determinados casos, acceso a los procesos organizacionales, otras veces a los datos; y en algunos casos a ambas partes de la aplicación. De esta manera, la información contenida en las aplicaciones puede ser compartida con otras aplicaciones.

Una manera para entender mejor el concepto de la interfaz de aplicación es el empleo de APIs (*Application Programming Interfaces*), las cuales son un tipo de mecanismo diseñado para conectar los recursos de una aplicación, sean éstos un servidor de aplicaciones, un *software* o una base de datos. Estos elementos permiten a los desarrolladores obtener servicios de los recursos para emplearlos ya sea en la misma aplicación o en alguna otra.

Actualmente, muchas de las aplicaciones empaquetadas como son los ERP ofrecen una interfaz abierta para su acceso y posteriormente su integración con otras aplicaciones.

Las interfaces ofrecidas por algunas aplicaciones proporcionan tres tipos de servicios: **servicios organizacionales, servicios de datos y compartición de objetos** (Linthicum, 2000).

Los **servicios organizacionales** se caracterizan por incluir interfaces programables de la lógica de la aplicación que permitan acceder a los datos o a la aplicación misma. Por ejemplo, si algún usuario desea usar la interfaz para actualizar la base de datos de los clientes pero desde otra aplicación, es posible llamar a este servicio y transferir dicha información a la base de datos.

Los **servicios de datos** son rutas directas a la base de datos ya sea de manera física o lógica. En general, las interfaces utilizadas por las aplicaciones empaquetadas para acceder a los datos son empleadas para la extracción de los mismos.

La **compartición de datos** se refieren al encapsulamiento, tanto de datos como de servicios organizacionales que actúan sobre la información contenida en la base de datos. La ventaja de este tipo de servicio se basa en el hecho de que no existe ningún problema referente a la integridad de los datos debido que no pueden ser accedidos sin invocar a algún método que los llame desde la aplicación misma.

Muchos de estos accesos son hacia los datos, la lógica del programa y los objetos. Sin embargo, muchas aplicaciones de este tipo no ofrecen una interfaz abierta por lo que es necesario emplear otro nivel de integración como es el de datos o la interfaz de usuario por decir un ejemplo.

Asimismo, las interfaces de usuario pueden clasificarse por su acceso en **interfaces de servicio completo**, las cuales proveen acceso al nivel de servicios organizacionales u operacionales, nivel de servicios de datos y el nivel de objetos; **interfaces de nivel limitado**, las cuales ofrecen acceso a sólo un nivel ya sea el de datos, objetos o servicios operacionales, además de que los servicios que proporcionan en ese nivel son limitados; e **interfaces controladas**, que proveen tan sólo un mínimo de características y funcionalidades de los niveles de datos o la lógica de la aplicación y están limitadas por decisiones económicas del vendedor o de mercadotecnia.

Este nivel de integración requiere un poco más de una base tecnológica pues es necesario un conocimiento más a fondo de la lógica de programación de los sistemas de información empleados en cada organización. Tanto esta metodología como la del nivel de método requieren de un análisis detallado de todos los sistemas de la organización con el fin de identificar sobre qué sistemas es más factible llevar a cabo una integración de aplicaciones.

## 2.7 CARACTERÍSTICAS DEL NIVEL DE MÉTODO

Este nivel se caracteriza por permitir a la empresa integrarse a través de la compartición de lógicas operacionales comunes o métodos. Esto es posible definiendo los métodos que pueden ser compartidos o integrados por algunas de las aplicaciones u ofreciendo la infraestructura necesaria para compartirlos, ya sea instalándolos en un servidor central o accediendo a ellos como objetos, obteniendo con esto un conjunto de procesos que pueden ser reutilizados en otras aplicaciones, el cual es el objetivo central de este nivel.

A diferencia de los otros niveles, éste requiere que la mayoría, si no es que todas, las aplicaciones empresariales sean cambiadas para obtener una ventaja significativa.

El problema existente en muchas aplicaciones empresariales es que contienen un gran número de procesos organizacionales o lógica del negocio, los cuales se encuentran conectados a través de varias tecnologías; por ejemplo cada proceso puede tener un método implementado en un lenguaje de programación o tecnología diferente uno del otro lo cual hace difícil la comunicación entre ellos.

Para llevar a cabo una correcta implementación del nivel de método es necesario comprender todos los procesos existentes dentro de una empresa, para lo cual, Linthicum los clasifica en cuatro tipos: **reglas, lógica, datos y objetos**.

Las **reglas** se refieren a condiciones que deben ser cumplidas debido a que afectan la identificación, procesamiento y manipulación de los datos. Estas reglas

pueden ser compartidas y accedidas por varias de las aplicaciones utilizadas en la empresa a través de la implementación de este nivel de EAI.

La **lógica** se define como la secuencia de instrucciones en un programa y se puede dividir en tres partes: **procesamiento secuencial**, es el conjunto de pasos de manera consecutiva; **selección**, es la toma de decisiones dentro de un programa comparando dos conjuntos de datos; **iteración** es la repetición de un conjunto de pasos.

Los **datos** son empleados para la compartición de información entre las aplicaciones empresariales. Debido a que los métodos actúan en los datos, es importante entender cómo se comparte la información para implementar un nivel de método exitoso.

Los **objetos** consisten en el encapsulamiento de datos utilizados por los métodos que actúan sobre la información contenida en la base de datos.

Una herramienta muy empleada en este nivel son los “*frameworks*”, los cuales son subsistemas de software, localizados de manera centralizada y accedidos por muchas aplicaciones. Proveen la infraestructura necesaria para compartir métodos u objetos entre las aplicaciones, de tal manera que permiten la construcción de una biblioteca de objetos para ser utilizada. El uso de estos “*frameworks*” es primordial para impulsar la integración de aplicaciones, obteniendo como beneficio principal la reutilización del código existente, éste puede darse dentro en las aplicaciones de la empresa bien con las empresas aliadas en cuanto a ciertos procesos como logística o distribución.

Como su nombre lo indica, el nivel de método involucra la reutilización del código de los sistemas de información existentes en la organización por lo que es importante contar con toda la documentación de los sistemas de información en caso de que se hayan realizado desarrollos internos o por otras compañías. Puede decirse que este nivel es uno de lo más complicados puesto que interviene un conocimiento total del funcionamiento de las aplicaciones que se desean integrar.

## 2.8 CARACTERÍSTICAS DEL NIVEL DE INTERFAZ DE USUARIO

El nivel de interfaz de usuario tiene la ventaja de que no requiere ningún tipo de cambio a la fuente de los datos o a la aplicación en sí. Se caracteriza por el empleo de técnicas o tecnologías que pueden ser usadas para acceder, o colocar, la información en una aplicación. Para implementar este nivel es primordial comprender la aplicación en su totalidad; es decir, el esquema de almacenamiento de los datos, la lógica de la aplicación y cómo la información es presentada en la interfaz de usuario. De igual manera, es necesario interpretar la razón de ser de los datos en la aplicación para comprender cómo la información es representada en la pantalla. A diferencia de los otros niveles, la información que se presenta en este nivel puede no existir físicamente en una base de datos sino que pueden ser datos, calculados por la lógica de la aplicación por lo cual es evidente la comprensión total de esta lógica.

Entendida la lógica del programa y el esquema de la base de datos, es necesario crear un catálogo de la información que aparece en las interfaces de usuario. Esto se lleva a cabo tomando una imagen de cada pantalla y categorizando cada dato que aparezca en ella. Además, es necesario mapear las posiciones de la información desplegada en las pantallas; para ello se requiere un proceso denominado mapeo de pantallas que consiste en determinar la localización de cada dato en la pantalla, el número de posiciones que ocupa y cualquier otra información relevante.

Una vez llevado a cabo este paso, es necesaria la extracción de los datos, la cual está basada en dos técnicas: **estática** y **dinámica**. (Linthicum, 2000)

- La extracción **estática** se refiere a que, sin importar qué tecnología se emplee, la información está siendo accedida de un punto estático en la pantalla; por lo que no es necesario buscar en la pantalla o emplear una lógica condicional para encontrar la información. La ventaja de esta técnica es su simplicidad sin embargo, si existe un cambio en la interfaz

puede darse el caso que los datos cambien de lugar y no se extraiga ningún tipo de información o en el peor de los casos, se extraiga información errónea. Se recomienda que esta técnica sea empleada en aplicaciones que nunca, o muy rara vez, cambian.

- La extracción **dinámica** consiste en buscar información en la pantalla usando una lógica condicional, es decir, tomando un punto como referencia y extrayendo todos los datos que concuerden con ese objetivo. La ventaja de esta técnica consiste en la habilidad de adaptarse automáticamente a los cambios dentro de la aplicación y reaccionar a los cambios en la estructura de la pantalla.

Como último punto es necesario considerar que, para aplicar el nivel de interfaz de usuario, es necesario un conocimiento pleno de todas las pantallas, vistas u información que la aplicación despliegue al usuario final. De esta manera, el análisis que se lleve a cabo se acercará a una integración mucho más efectiva de la información, contendrá menos errores o problemas de actualización y será mucho más fácil detectar desde el principio las necesidades de la información a integrar.

A continuación se estudiará una metodología sobre la integración de aplicaciones usando el nivel de datos. A diferencia de los otros niveles que involucran altos costos o riesgos mucho mayores además de una complejidad de la lógica del sistema de información, el nivel de datos es mucho más factible de emplear en la mayoría de los casos.

## **2.9 IMPLEMENTACIÓN DEL NIVEL DE DATOS DE EAI**

La integración de aplicaciones empresariales al nivel de datos es empleada por aquellas instituciones donde una simple compartición de datos es el obstáculo a vencer. Este nivel es el que tiene el riesgo más bajo de error y tiene como ventajas el costo, debido a que no se requiere programación en varios casos, las pruebas son mínimas y el mecanismo de

transporte de datos entre aplicaciones es muy accesible en cuanto a precio (Kalafsky, 2000).

La implementación del nivel de datos se lleva a cabo identificando el lugar donde se encuentran los datos, reuniendo información sobre éstos y aplicando la lógica o los procesos organizacionales actuales para determinar qué datos fluyen, hacia dónde y por qué.

Existen tres pasos básicos que deben seguirse antes de implementar el nivel de datos de EAI (Linthicum, 2000). Estos pasos son:

1. la identificación de los datos;
2. la categorización de los datos y,
3. la elaboración de un modelo de metadatos empresarial que será usado como guía para integrar los diferentes repositorios que existan dentro de la empresa.

Hummingbird (2000) define que los metadatos consiste en información descriptiva sobre el contenido, calidad, condición y característica de los datos, por lo cual podemos entender metadatos como los datos que definen la manera como se encuentra relacionada la información en una base de datos, su formato y su estructura.

### **2.9.1 Identificación de los datos**

Como primer punto, es necesario crear una lista de sistemas candidatos con el fin de poder determinar qué bases de datos son empleadas por estos sistemas. Enseguida, es necesario saber dónde se encuentran localizadas, quién las posee, así como cualquier información relevante como marca, modelo o revisiones de la misma base de datos. Asimismo, cualquier tecnología en el mercado que sea capaz de realizar ingeniería inversa tanto física como lógica a las bases de datos servirá para identificar la información contenida en la estructura; sin embargo, estas herramientas no son capaces de determinar cómo es usada la información a través de la aplicación. Adicionalmente, más información puede ser seleccionada examinando los diccionarios de datos

relacionados a los repositorios de datos analizados. De acuerdo a Mattila (2001), estos diccionarios pueden contener información como:

- definiciones de elementos de datos;
- definiciones de tablas;
- esquema de la base de datos,
- modelo entidad-relación y,
- modelo de seguridad de la base de datos.

Cuando se analizan las bases de datos para aplicar EAI, pueden aparecer ciertos aspectos relacionados con la seguridad. En este punto, es importante comprender las reglas y regulaciones que fueron aplicadas para la construcción de la base de datos, ya que la mayoría del software empleado para comunicar las bases de datos de las aplicaciones no toma en cuenta estas normas y, como resultado, puede existir un peligro de daño a la integridad de la base de datos.

Otro factor a tomar en cuenta, cuando se identifican los datos para su integración a través de EAI, se refiere a la latencia de los datos, es decir la característica que determina qué tan actualizada debe encontrarse la información. De esta manera, los diseñadores de la arquitectura de integración podrán determinar cuándo la información debe de ser copiada o transferida a otra aplicación empresarial.

Basándose en esto es preciso definir el significado de latencia de los datos. De acuerdo al diccionario en línea Whatis, latencia es un sinónimo de retraso y es usado para definir cuanto tiempo toma un paquete de datos de moverse desde un punto específico a otro (whatis.com, 2001). Existen tres tipos de latencia de los datos, éstos son:

**Tiempo real:** se refiere a la información que es colocada en la base de datos tal y como se va generando, es decir sin ningún retraso ni proceso de actualización. Un ejemplo claro de este tipo de datos es el monitoreo de precios de acciones, información que está disponible a cualquier usuario o aplicación que la requiera para su procesamiento.



**Tiempo cercano:** se refiere a la información que es actualizada por intervalos pero no de manera instantánea. Un ejemplo práctico sería la información de temperatura de las ciudades que se publican a través de Internet.

**Sin tiempo:** se refiere a la información que es actualizada una vez exclusivamente. Podemos mencionar como ejemplo los números de cuenta de los clientes.

Finalmente, un componente vital en la identificación es el formato de los datos. Esta característica permite determinar cuánta información se encuentra estructurada, incluyendo las propiedades de los datos dentro de la misma estructura, esto es: la longitud, el tipo de datos, nombre del elemento, y qué tipo de información almacena.

### **2.9.2 Categorización de los datos**

En la metodología de EAI, la categorización de datos se define como el proceso de recopilación de información de metadatos y otros datos a través de las aplicaciones empresariales. Una vez que ha sido llevado a cabo este proceso, es posible crear un catálogo de todos los datos que existan en la organización. Así, este catálogo se convierte en la base para crear el modelo de metadatos de la empresa, que es en sí el fundamento del nivel de datos de EAI. Una regla importante para la creación de un catálogo es mientras más información, mucho mejor. Una característica primordial es que el catálogo sea de gran tamaño pues en él se encontrará una gran cantidad de elementos de los datos así como los propios metadatos. La ventaja de tener un catálogo amplio radica en poder abarcar la mayoría de los datos que se manejan dentro de la organización y facilitar los desarrollos futuros que requieran acceso a dichos datos.

Una vez que toda la información sobre los datos de la empresa se encuentra en el catálogo de datos, es necesario enfocarse en la realización del modelo empresarial de metadatos.

De acuerdo a las definiciones empleadas en el sitio especializado en el manejo de base de datos searchDatabase, la diferencia entre el catálogo de datos y el modelo de metadatos radica en que el catálogo sólo define la estructura de los datos, mientras que el modelo

además de la estructura, también determinan cómo interactuarán los datos en las aplicaciones empresariales (searchDatabase, 2001).

### **2.9.3 Clasificación del modelo de metadatos**

De acuerdo a Linthicum (2000), el modelo de metadatos empleado puede ser dividido en dos componentes: **lógico y físico**.

El **modelo lógico** se refiere al proceso de crear una arquitectura para todos los repositorios de datos que son independientes de un modelo físico de una base de datos, herramientas de desarrollo o un manejador de base de datos. El componente principal del modelo lógico es un diagrama Entidad-Relación.

El **modelo físico** se construye empleando el modelo lógico y el catálogo de datos con el fin que puedan ser utilizados en una base de datos física. En el contexto de EAI esto significa crear una estructura de base de datos que sirva para localizar todas las demás bases de datos dentro de la empresa.

Finalmente, antes de que el modelo lógico o físico se encuentre listo, es necesario normalizarlo para que funcione de manera óptima. El objetivo de este paso consiste en lograr examinar todas las bases de datos dentro de la empresa como una sola base de datos y hacerla lo más eficiente posible a través de los procesos organizacionales. La vista de la base de datos obtenida se encontrará completamente integrada y mucho más fácil de administrar y escalar.

Como se ha visto en este capítulo, la integración de los sistemas es una tarea que necesita un despliegue efectivo de recursos, si se desea llevar a cabo una implementación exitosa. Sin duda, se puede aplicar cada nivel o metodología de manera independiente pero, si se desean mantener los sistemas completamente integrados, una buena opción consiste en comenzar desde el nivel más básico (nivel de datos), pasando por el nivel de método y de interfaz de aplicación que tienen mayor nivel de complejidad relacionado con los

aspectos técnicos de los sistemas de información, hasta concluir con el nivel de interfaz de usuario que se encuentra relacionado con el usuario final.

Sin embargo, existen muchos casos en donde las organizaciones cuentan con sistemas de información sin ningún tipo de documentación sobre el código o no tienen manera alguna de identificar los componentes funcionales de la aplicación ni siquiera los datos. Moragenthal (2000) comenta que la falta de un modelo de metadatos estándar consiste en un obstáculo para la integración de aplicaciones. La sugerencia más óptima consiste en llevar a cabo una integración en el nivel de la base de datos o a través de la interfaz de usuario, dependiendo la facilidad que ofrezca el sistema para cada nivel. De estos dos, el nivel de datos ofrece una ventaja, debido a que no necesita conocimiento alguno sobre la lógica del programa, sino que es necesario llevar a cabo actualizaciones o modificaciones directamente a la base de datos para contar con información actualizada a través de un nuevo tipo de software denominado *middleware*, el cual analizaremos con mayor detalle en el capítulo siguiente.

La integración de aplicaciones es posible gracias a una gran variedad de tecnologías o metodologías que parten desde una transferencia de datos hasta todo un proceso complejo de transferencia de permisos, sincronización de mensajes o solicitud de peticiones de acceso. A últimas fechas las empresas han incrementado su presupuesto en rubros como *middleware*. The Gartner Group (mencionado por Rowe, 2000) comenta que aproximadamente un 40% del presupuesto de informática se emplea para sistemas de integración.

Con toda la avalancha de sistemas de información dispersos en las organizaciones, se ha venido dado un auge de la utilización de software intermedio (denominado *middleware*) que permite llevar a cabo diferentes tipo de conexión entre los sistemas o las bases de datos. Existe una gran variedad de este tipo de software con características muy variadas las cuales se definen a lo largo del capítulo siguiente.

### 3.1 TECNOLOGÍAS Y ESTÁNDARES EXISTENTES PARA LA - INTEGRACIÓN DE APLICACIONES

Existe una gran variedad de tecnologías para acceder a los datos dentro de las bases de datos organizacionales. Es posible realizar, por alguno de estos métodos la integración de la información. Cada método tiene un grado de especialización y complejidad diferente de los demás por lo que es necesario la realización de un estudio de las bases de datos para decidir cual es la mejor a utilizar. A continuación se mencionan las tecnologías más empleadas para la adquisición de datos (Ruh, 2001)

#### 3.1.1 Transferencia de archivos por lotes

Se refiere a las herramientas que permiten mover archivos entre sistemas o aplicaciones de alguna manera predeterminada. Esta técnica es empleada en los grandes equipos de cómputo o *mainframes*. En realidad, lo que se realiza con esto es un movimiento de datos mas que una integración de los mismos.

#### 3.1.2 Conectividad de Base de Datos Abierta (ODBC)

Puede definirse como una interfaz estándar de programación que permite abstraer información de bases de datos relacionales heterogéneas. La interfaz puede ser usada en otras formas de almacenamiento de datos siempre y cuando soporte la estructura u operaciones de la base de datos.

Antes de seguir describiendo las tecnologías empleadas, es necesario definir el concepto de “*middleware*” al ser la columna vertebral que opera entre las aplicaciones y transfiere datos sistemáticamente entre ellas mismas.

*Middleware* se refiere a todo el software que provee los servicios que permite la comunicación entre las aplicaciones (Ruh, 2001). Linthicum define *middleware* como:

“software que facilita la comunicación entre dos aplicaciones” (Linthicum, 2000). Por su parte, el diccionario en línea *whatis.com* define *middleware* como un término general para cualquier programa que se encarga de pegar o servir como intermediario entre dos aplicaciones existentes y separadas (*whatis.com*, 2001)

**Con esta información podemos definir que *middleware* es toda herramienta de software que permite comunicar aplicaciones entre sí empleando la funcionalidad existente de las mismas.**

### **3.1.3 Acceso a la base de datos por *middleware***

Es un tipo de software que provee conectividad a las bases de datos, especialmente las distribuidas, a través de la creación de conexiones. Provee un ambiente para administrar las peticiones enviadas a la base de datos y el retorno de los resultados. Sus funciones principales son el intercambio de consultas, la administración de resultados, la conectividad a las bases de datos, la creación de conexiones, entre otras tareas.

El *middleware* orientado a datos emplea interfaces programables de aplicación (APIs por sus siglas en inglés) para mostrar la información que se encuentra en las tablas. La mayoría del *middleware* de acceso a datos está basado en un modelo síncrono de petición / respuesta tal como se verá en la siguiente sección.. (Network Computing, 1995)

### **3.1.4 Transformación de datos**

Esta herramienta generalmente se emplea para complementar el *middleware*. Provee la habilidad de convertir los datos desde el formato de la base de datos fuente hacia el formato de la base de datos destino ya que a menudo las definiciones, estructuras y esquemas de los datos empleados por las aplicaciones son diferentes.

Dependiendo de los sistemas de información existentes en cada organización, se debe elegir la tecnología que cumpla con las necesidades específicas de los mismos. Sin embargo, lo más importante es contar con un plan estratégico de integración antes de decidir que tecnología emplear. A continuación se definen los tipos de *middleware* y los mecanismos de comunicación que utilizan cada uno ellos.

## 3.2 **MIDDLEWARE**

### 3.2.1 **Tipos de Comunicación**

Tanto Ruh, Linthicum e International System Group coinciden en que existen dos mecanismos de comunicación del *middleware*: **comunicación síncrona y comunicación asíncrona.**

#### 3.2.1.1 **Comunicación asíncrona**

La comunicación **asíncrona** se refiere al tipo de comunicación empleada por el *middleware* que transporta información entre dos o más aplicaciones de un modo asíncrono, es decir, son capaces de desconectarse de las aplicaciones fuente u objetivo, y además estas aplicaciones no son dependientes de otras aplicaciones con las que se encuentre conectadas para llevar a cabo el procesamiento. En pocas palabras, el proceso requerido para llevar esto a cabo consiste en que la aplicación pone un mensaje en una cola y prosigue con el procesamiento, esperando que lleguen las respuestas posteriormente desde cualquier otra aplicación.

Una ventaja que se puede obtener de este tipo de comunicación es que no bloquea a la aplicación en espera de una señal para continuar con el procesamiento. Asimismo, las mismas aplicaciones pueden seguir operando en un ambiente donde, tanto las redes como los servidores, pueden no estar siempre disponibles. Sin embargo, una desventaja consiste en que las aplicaciones no se encuentran comunicadas en tiempo real por lo que puede darse el caso que alguna no reciba la información que la otra le envía y existan retrasos en el procesamiento de la información.

Entre los más populares, se pueden definir tres tipos de comunicación asíncrona:

- Pase de mensajes,
- Publicación/suscripción, y
- *Broadcast*

**Pase de mensajes:** esta forma de comunicación consiste en enviar una petición desde una aplicación mensajera hacia una aplicación receptora. Una vez realizado el envío del mensaje, la aplicación mensajera continúa con el procesamiento. De tal manera que el receptor recibe el mensaje y procesa la información que recibe. Se emplea cuando la información necesita ser transmitida pero no se requiere una respuesta.

**Publicación/suscripción:** la comunicación de este tipo se caracteriza por el envío por parte de la aplicación mensajera de información la cual puede ser compartida en un destino, ya sea en una lista de publicación o una de suscripción. Esta lista redistribuye la información actualizada a cualquier aplicación interesada en ella. Este tipo de comunicación permite que cada aplicación sea notificada de los eventos de su interés a través de la definición de información, estructuras de datos o tipos de información que esté interesado en recibir. El desarrollador de la aplicación receptora define el interés de cada una de ellas. Esta comunicación se emplea en procesos donde la petición es en realidad, la notificación que un evento ha ocurrido.

***Broadcast:*** en este tipo de comunicación una aplicación mensajera envía la información a todos los receptores dentro de una red, de tal manera que cada uno de ellos determina si la información es de su interés de acuerdo a su contenido para aceptarlas y continuar con su procesamiento.

A continuación, se analizará el otro tipo de comunicación que pueden emplear el *middleware* para permitir la transferencia de información entre aplicaciones.

### 3.2.1.2 Comunicación síncrona

La comunicación **síncrona** se refiere a la comunicación empleada por el *middleware* que se encuentra fuertemente acoplado con las aplicaciones, por lo que estas aplicaciones dependen del *middleware* para el procesamiento de una o más funciones enviadas a una aplicación remota. Como consecuencia de esto, las aplicaciones que hacen la petición deben esperar que la aplicación remota responda. (Ruh, 2001)

Este tipo de comunicación es mucho menos confiable que la anterior dado que cada aplicación debe esperar una respuesta de la aplicación remota a la que hace la petición, la cual puede ser detenida por problemas en la red o del servidor remoto, además del amplio consumo de ancho de banda dado que deben realizarse varias llamadas a través de la red a las aplicaciones remotas para soportar este tipo de comunicación.

Entre los tres tipos más populares de comunicación síncrona tenemos:

- Petición/respuesta
- Unidireccional
- Polling

**Petición/respuesta:** en este tipo de comunicación, una aplicación hace una petición a una aplicación receptora y espera una contestación antes de continuar con el procesamiento. El empleo de este tipo de comunicación está destinado a situaciones en las que la respuesta contiene información necesaria para continuar con el procesamiento de la aplicación que envió la petición.

**Unidireccional:** es el tipo de comunicación síncrona más fácil y consiste en que una aplicación envía una petición hacia otra aplicación receptora esperando a que esta última envíe de vuelta un mensaje de recepción del mensaje (*acknowledgment*) para que pueda continuar con el procesamiento. Este tipo de comunicación se emplea cuando la aplicación mensajera debe sincronizar su procesamiento con la aplicación receptora. Una desventaja consiste en que no existe otro tipo de información para la aplicación mensajera que el mensaje de recepción (*acknowledgment*). Otra desventaja es el conjunto



de las implicaciones de desempeño que provocan mantener detenida a la aplicación mensajera hasta que reciba el mensaje de regreso.

**Polling:** este tipo de comunicación es utilizado cuando la aplicación mensajera envía una petición a una aplicación receptora pero en vez de bloquearse, continua su procesamiento. Enseguida, a través de intervalos de tiempo, revisa si existe una respuesta por parte de la aplicación receptora, en caso negativo continúa procesando pero mantiene el “*polling*” en los intervalos definidos y, al recibir respuesta, procesa la información y detiene el chequeo de mensajes. Este tipo de comunicación se emplea en situaciones cuando la aplicación mensajera necesita una respuesta, sin embargo es capaz de continuar procesando mientras espera la misma. Este tipo de comunicación síncrona es mucho más efectiva que las dos anteriores pero mucho más compleja de desarrollar.

El tipo de comunicación del *middleware* es un punto importante a considerar en la selección del mismo, pues involucra aspectos del desempeño de los sistemas de información así como de las bases de datos e infraestructura computacional. De la misma manera, es importante definir el tipo de *middleware* a utilizar y que se encuentre relacionado con la comunicación seleccionada.

### 3.2.2 Tipos de *middleware*

El *middleware*, en sus diferentes versiones, se puede clasificar en: RPC (*Remote Procedure Call*), MOM (*Message Oriented Middleware*), Objetos Distribuidos, *middleware* orientado a la base de datos y *middleware* orientado a transacciones (monitores de procesamiento de transacciones y servidores de aplicaciones).

A continuación, se describe cada uno de estos diferentes tipos de *middleware*.

### 3.2.2.1 *Remote Procedure Call (RPC)*

Se refiere al tipo de *middleware* que está basado en la noción de un desarrollo distribuido de aplicaciones que se integra en el nivel de método o procedimiento. Tiene la posibilidad de crear llamadas de procedimiento a través de una red. (Ruh, 2001)

Los RPC son unos de los mecanismos más remotos de construcción de aplicaciones distribuidas. Por naturaleza, los RPC necesitan una conexión síncrona y persistente, es decir, un proceso que un cliente enviar al servidor es bloqueado hasta que se tenga una respuesta para dicho proceso por lo cual tienen la habilidad de crear llamadas a procedimientos a través de la red. (International, 1997)

Este tipo de *middleware* es el más antiguo que existe, sin embargo son los más fáciles de comprender y emplear. La funcionalidad consiste en proveer a los desarrolladores la facilidad de invocar una función dentro de un programa y lograr que esa misma función se ejecute dentro de otro programa en un equipo remoto. Debido a esto, los RPC son síncronos, pues se necesita que uno reciba la orden para que el otro pueda funcionar.

A primera vista, la ventaja que ofrecen los RPC es la simplicidad del mecanismo y la facilidad de programación. Sin embargo, existe la desventaja de un requerimiento mayor de poder de procesamiento. De la misma manera, es casi imposible la escalabilidad de este tipo de tecnología a no ser que se combinen con otro *middleware* como monitores de transacciones o *middleware* de envío sincronizado de mensajes.

Actualmente, este tipo de *middleware* es usado en muchas organizaciones pero, si se planea una integración futura, no se sugiere como una alternativa viable dado el actual crecimiento de las nuevas tecnologías orientadas en objetos, debido a que esta tecnología es procedimental. Esto trae una dificultad al aplicar el concepto de RPC en el desarrollo de nuevo software y en la integración del mismo. (Linthicum, 2000)

### 3.2.2.2 *Message Oriented Middleware (MOM)*

MOM es un tipo de *middleware* que emplea mensajes como un método de integración, además de ofrecer la habilidad de integrar diversas aplicaciones a través del uso de mensajes. Permiten la creación, manipulación, almacenamiento y comunicación de mensajes para mover información de un punto a otro.

Este tipo de *middleware* fue creado para cubrir las deficiencias de los RPC a través del uso de los mensajes antes mencionados, pues con esta característica no se requiere una integración directa con las aplicaciones. Asimismo, los mensajes pueden ser enviados a través de la red sin esperar una respuesta instantánea de la aplicación que recibe el mensaje, por lo que es posible continuar con el funcionamiento de la aplicación que envió el mensaje. De este modo se dice que MOM es una aplicación asíncrona. El empleo de este paradigma es más conveniente para los desarrolladores ya que no bloquea la aplicación desde el punto de vista de procesamiento, sin embargo el modelo es un poco más complejo que el modelo síncrono. El *middleware MOM* intercambia datos de una manera confiable y segura entre aplicaciones en plataformas y redes diferentes (McIntyre, 2000)

La ventaja de los mensajes puede verse en que son pequeñas unidades de información y son fáciles de manipular. Estos mensajes tienen una estructura y contenido, en pocas palabras pueden describirse como bases de datos de un registro que se mueven entre las aplicaciones a través de mecanismos de envío de mensajes. Existen dos modelos soportados por MOM: punto-a-punto y *message queuing (MQ)*, aunque a últimas fechas se ha estado desarrollando el modelo de publicación/suscripción (*pub/sub*). (Haase, 2001) El más empleado de estos dos modelos es el segundo (MQ), ya que permite a cada programa realizar todos sus procesos sin interrupción alguna al poner los mensajes en una cola de espera. Otra ventaja que se puede apreciar es que una aplicación puede enviar el mensaje remotamente a otras aplicaciones sin esperar que se encuentren corriendo o funcionando.

Como desventaja de esta tecnología, Shoffner (2000) comenta que los desarrolladores que no están familiarizados con el uso de mensajes tienen que aprender a emplear el uso

correcto de ellos, así como los mensajes empleados son asíncronos por naturaleza, no pueden crear aplicaciones que bloqueen el servidor para llevar a cabo transacciones.

### **3.2.2.3 Middleware Orientado a las Bases de Datos**

*Middleware* orientado a las bases de datos es un tipo de *middleware* basado en el acceso a datos distribuidos ya sea en archivos o bases de datos (Ruh, 2001). Permite la integración en el nivel de datos así como la realización de consultas o movimiento de información a través de una red. Los desarrolladores emplean este tipo de *middleware* como un mecanismo para extraer información de bases de datos, sean locales o remotas.

Los primeros desarrollos de este tipo de tecnología proveían un mecanismo propietario para acceder a los datos; a este tipo de tecnología se le denominaba *middleware* nativo de base de datos. La principal desventaja es que sólo podía comunicarse con un tipo de datos, sin embargo permitía un desempeño más efectivo en el acceso a los datos.

Sin embargo, en los últimos años los mecanismos empleados con el acceso a datos se realiza a través de estándares como ODBC (*Open Database Connectivity*) o JDBC (*Java Database Connectivity*). Este tipo de *middleware*, denominado Interfaz de Nivel de Llamada (CLIs, por sus siglas en inglés) ofrece una solución de integración dentro del nivel de datos ya que permite acceder directamente a diferentes fuentes de datos o a las bases de datos sin profundizar en la funcionalidad o las interfaces de presentación de una aplicación.

### **3.2.2.4 Objetos Distribuidos**

La tecnología de objetos distribuidos es un tipo de *middleware* que extiende el concepto de tecnología orientada a objetos a procesamiento distribuido (Ruh, 2001) El objetivo de esta tecnología es la creación de interfaces para las aplicaciones que permitan que el software se vea como un objeto. En general, se divide en dos categorías: los que soportan el estándar CORBA (*Common Request Object Broker Architecture*) y los que soportan la tecnología OLE/COM de Microsoft (International, 1997)

El funcionamiento de esta tecnología según Robinson (1999) consiste en que el cliente envía una petición. Invocando a un método en un servidor de objetos, el *middleware* objetos distribuidos intercepta la petición y busca el objeto en la misma máquina o a través de la red. De la misma manera, este *middleware* se encarga de devolver la respuesta al cliente además de proveer interoperabilidad entre aplicaciones en diferentes equipos de cómputo dentro de una red que permita interconectar múltiples sistemas.

La ventaja de este tipo de tecnologías es la reutilización de los componentes en toda la infraestructura existente y permite la creación de sistemas basados en estos mismos componentes.

Una desventaja podría ser la complejidad comparada con la tecnología MOM dado que requiere un mayor nivel de acoplamiento entre las aplicaciones. A pesar de este problema, la integración de ambas tecnologías permite resolver el más amplio conjunto de problemas en cuanto a EAI se refiere.

### **3.2.2.5 *Middleware* Orientado a Transacciones**

El objetivo de este tipo de *middleware* es coordinar el movimiento de información y la compartición de procedimientos entre diferentes recursos. Esta tecnología ofrece un excelente mecanismo para compartir los procedimientos o métodos pero no es tan efectivo para la compartición de información. Los dos tipos de *middleware* orientado a transacciones más empleados en el mercado actualmente son: monitores de procesamiento de transacciones y servidores de aplicaciones.

- **Monitores de Procesamiento de Transacciones**

Un monitor de procesamiento de transacciones es un tipo de *middleware* que preserva la integridad de una transacción (Ruh, 2001). Este tipo de tecnología provee un espacio destinado a la lógica de la aplicación así como un mecanismo para facilitar la comunicación entre dos o más aplicaciones. Se basa en la premisa de una transacción, la cual la podemos definir como una unidad de trabajo que cuenta siempre con un inicio de transacción y un final de la misma.

El objetivo de mantener la lógica de la aplicación dentro de una transacción consiste en permitir que ésta se complete exitosamente o realice un *rollback* en caso de alguna falla. Además, soporta funciones como detección de fallas, autoreinicio, bitácora de errores y replicación.

La ventaja de esta tecnología consiste en la capacidad de dividir una aplicación en pequeñas porciones y posteriormente invocar a éstas mismas, ya sea por otro usuario o por algún otro sistema conectado. Dado su tamaño, son fáciles de manipular y procesar. Asimismo, proveen conectores que les permiten acoplarse a recursos como bases de datos, aplicaciones, entre otros recursos. Otra característica de los monitores de procesamiento de transacciones es la capacidad de rutear, sincronizar y organizar el envío de mensaje por lo que permite la asignación de prioridades a cada mensaje.

Este tipo de tecnologías son mejores para transacciones síncronas en tiempo real, tales como manejo de inversiones o aplicaciones bancarias (Sánchez, 2001)

- **Servidores de Aplicaciones**

En la actualidad, uno de los usos de los servidores de aplicaciones consiste en el procesamiento de transacciones de aplicaciones basadas en Web, empleando lenguajes orientados a objetos como Java. Asimismo, este tipo de tecnología ofrece conexión a los recursos del *back-end* como son bases de datos, aplicaciones de ERP e incluso aplicaciones tradicionales de *mainframes*. Adicionalmente, los servidores de aplicaciones ofrecen mecanismos para el desarrollo de interfaces de usuario.

Esta tecnología es comúnmente aplicada en el procesamiento de transacciones asíncronas entre sistemas acoplados que pueden no estar disponibles en algún momento determinado, lo cual cubre la mayoría de las aplicaciones en Web. (Sánchez, 2001)

Cada uno de estos tipos de *middleware* tiene sus propias ventajas o desventajas a considerar al momento de llevar a cabo la selección del software a utilizar. Uno de los aspectos más importantes en la selección del software consiste en la capacidad de acceder a los datos, sin embargo existen varios otros atributos que deben ser tomados en cuenta

para seleccionar el *middleware* que cumpla con las expectativas de la organización. De tal manera, la evaluación del software a elegir forma parte importante de la estrategia de integración de aplicaciones empresariales.

### 3.3 EVALUACIÓN DEL *MIDDLEWARE*

La selección entre las distintas alternativas de *middleware* requiere de un vasto conocimiento en diversas áreas, debido a que las aplicaciones existentes en las organizaciones cuentan con una gran variedad de características que las hacen diferentes unas de otras. Debido a esto, pueden existir diferencias sutiles pero críticas en las aplicaciones de *middleware* (Mcyntire, 2000). Estas aplicaciones pueden ser:

- Comunicación de aplicación a aplicación,
- Comunicación de aplicación a la Base de Datos,
- Servicios de Base de Datos a Base de Datos, o
- Redes Externas.

Cuando se está evaluando una aplicación de *middleware*, las áreas más importantes a considerar son:

- Criterios primarios (disponibilidad, desempeño, administración y escalabilidad)  
Estos factores pueden ser medidos por el número de:
  - Usuarios (concurrentes y totales)
  - Archivos y bases de datos.
  - Programas
- Tipo de aplicaciones a las que se conectarán (proceso por lotes, tiempo real, etc.)
- Variedad de protocolos, sistemas operativos, etc
- Arquitecturas (dos partes, tres capas, etc.)
- Volumen de datos por mover (actual y futuro)
- Requerimientos de tiempo para mover los datos

- Ruteo interno vs. externo
- Complejidad de las transformaciones
- Capacidad técnica del staff interno
- Presupuesto
- Tiempo de implementación

Mientras se evalúan los diversos paquetes de EAI para su adquisición, es necesario observar que contengan una buena cantidad de adaptadores integrados que soporten las aplicaciones, formatos de datos y protocolos de los sistemas de información de la organización actual así como los que se piensen adquirir en un futuro. Asimismo es importante identificar herramientas de desarrollo que permitan la creación de adaptadores personalizados para el manejo de datos. (Craig, 2000)

Una buena técnica para la selección de *middleware* consiste en identificar el área que más peso tiene de acuerdo a la estrategia de integración y llevar a cabo una matriz del área contra el peso específico (ANEXO II) para tener así un esquema mucho más fácil de identificar en la evaluación.

### **3.4 INTEGRACIÓN A TRAVÉS DE INTERFACES DE DATOS**

Se puede definir la integración a través de interfaces de datos al proceso de mover datos directamente entre las bases de datos de las aplicaciones. (Taviz, 2000b). Debido a que emplea un lenguaje de manipulación de datos para transformar y validar los datos para adecuarse a las reglas de la base de datos receptora, no es necesaria la complejidad existente en la reprogramación de la aplicación.

Sin duda, uno de los problemas existentes en pasar información directamente a una base de datos es la posible corrupción de los mismos o incompatibilidad con la lógica de la aplicación. Debido a esto, un requisito clave e indispensable consiste en que las herramientas empleadas, para llevar a cabo este proceso, contengan reglas definidas y probadas para la transformación y validación, antes de la carga de datos en la base de



datos receptora. Así, existirá un mecanismo que permita asegurar que los datos sean compatibles con la aplicación antes de vaciarlos en la base de datos receptora.

Una ventaja de este tipo de integración, a diferencia del *middleware* transaccional, consiste en la habilidad de simplificar las tareas de integración, así como crear y mantener los datos compartidos de una manera mucho más fácil. La integración a través de interfaces de datos elimina la complejidad existente en el desarrollo de código, al igual que comprender la lógica o las estructuras de datos de las aplicaciones. (Taviz, 2000a)

Para llevar a cabo la integración de datos mediante la metodología de interfaz de datos, independientemente del tipo de herramienta que se emplee, es primordial definir un equipo de trabajo con las habilidades necesarias, realizar una agenda y comprometer el proyecto dentro de la infraestructura general de la empresa.

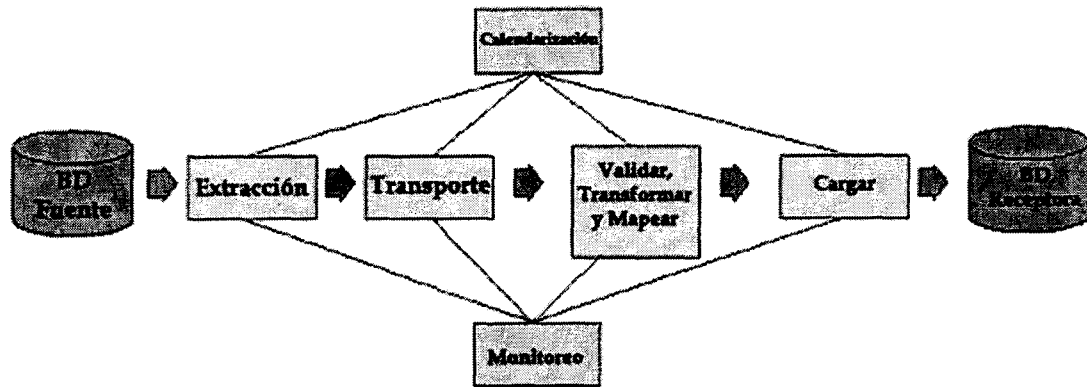
El equipo encargado de la integración enfrenta la ejecución de varias tareas, las cuales pueden dividirse en dos categorías: **planeación (analítica) e implementación (programación)**.

#### **3.4.1 Tareas de Planeación**

Estas tareas requieren que el equipo tenga una estrategia de integración antes de realizar el trabajo de implementación. Taviz (2000a) enumera algunas acciones que permiten una correcta decisión en la planeación:

- Definir las relaciones entre los datos de las aplicaciones heredadas y las tablas de las nuevas bases de datos.
- Identificar columnas de las tablas de las bases de datos asignadas por dependencias jerárquicas.
- Identificar columnas que requieran constantemente valores por default.
- Identificar columnas que requieran manipulación de datos como concatenación o multiplicación por otros valores.
- Identificar columnas derivadas de vistas o secuencias.

- Validar columnas con información contenida en otras tablas, con el fin de mantener integridad referencial durante la integración.
- Mapear cada dato en su correspondiente columna.



**Figura 3.1: Tareas Funcionales de la Integración por Interfaces. Fuente: Taviz Technology, 2000b**

La Figura 3.1 define la integración de datos a través de una interfaz por medio de los siguientes procesos automatizados:

- Extracción de datos de la base de datos fuente.
- Transporte de los datos a un programa de mapeo y transformación, que puede encontrarse en la aplicación fuente, receptora o algún sistema intermedio.
- Transformación, mapeo y validación de los datos.
- Carga de los datos transformados en la base de datos receptora.

Adicionalmente, existen varias tareas que sirven como soporte para aplicaciones de misión crítica:

- Calendarización de la integración de los datos para que, a través de un intervalo definido de tiempo o alguna acción específica, se carguen los datos en la base de datos receptora bajo circunstancias correctas.
- Monitoreo de los resultados permitiendo que, si la integración de los datos no se realiza adecuadamente, se notifique al usuario y se tomen acciones automáticas

definidas por el desarrollador que permitan recuperar o volver al estado original las transacciones de datos.

- Control para permitir que múltiples tareas puedan ejecutarse simultáneamente.

En sí, estas tareas están más relacionadas con la administración del proyecto antes de llevar a cabo la operación. Es vital que la estrategia definida cubra todos los aspectos posibles para evitar fallas o problemas no contemplados al momento de llevar a cabo la implementación.

### **3.4.2 Tareas de Implementación**

Debido a la complejidad existente de transformar y mapear grandes cantidades de datos heredados de una variedad de fuentes, la implementación puede ser complicada y consumir gran cantidad de tiempo (Taviz, 2000b). Las tareas referentes a la implementación están identificadas de acuerdo a las decisiones hechas en la fase de planeación y se enfocan en los siguientes aspectos:

- Creación de programas de extracción o descarga de datos del sistema heredado.
- Creación de archivos de texto o separados por coma de los datos heredados.
- Definición y creación de tablas de interfaz.
- Llenado de información de las tablas de interfaz.
- Definir las entidades a integrar y los valores por default.
- Implementar la transformación de datos en las tablas de interfaz.
- Copiar las tablas de interfaz a las tablas de producción.
- Confirmar la carga de las tablas en la aplicación.
- Validar la integración de entidades.
- Repetir el proceso para redefinir las validaciones y transformaciones.

Si se llevan a cabo estos pasos antes de realizar el verdadero proceso de integración, el equipo puede anticipar los obstáculos y aplicar los recursos o herramientas apropiados con el fin de mantenerse a tono con el proyecto.

Finalmente, la integración de datos debe ser una parte de un plan general de implementación. Un componente clave es la evaluación y selección de un producto o herramienta para la integración de datos. Para ello es necesario observar ciertas características:

- Un conjunto completo de funciones de transformación de datos.
- Buena integración con la base de datos receptora.
- Soporte para todas las fases de la integración de datos, por ejemplo, vistas gráficas, modificación de la base de datos receptora, transformación, mapeo, carga y consulta. Es preferible que todas estas características se encuentren completamente integradas.
- Facilidad de uso y modificación a través de ciclos iterativos.
- Habilidad de probar mapeos y transformaciones antes de enviar información a la base de datos receptora.
- Habilidad de probar, de manera interactiva, las reglas de transformación y validación antes de enviar información a la base de datos receptora.
- Facilidad de extraer datos de las aplicaciones fuentes sin la necesidad de realizar ingeniería inversa a las aplicaciones.
- Evitar el acoplamiento de datos de varias tablas de aplicaciones fuentes separadas en una o más tablas de la aplicación receptora.
- Rutear, volver a poner en cola de manera automática, o suspender la transformación de datos en casos críticos o de emergencia.
- Recuperar los estados previos de la base de datos en caso de una falla del sistema.

Las tecnologías revisadas en el capítulo para la obtención de datos así como el *middleware* a utilizar para conectar los sistemas de información forman parte importante de la estrategia de integración de aplicaciones empresariales. Un buen análisis de los sistemas de información existentes en la organización, las bases de datos que utilizan así como una buena identificación del *middleware* o tecnología que pueda ser utilizada permitirá desarrollar un esquema mucho más completa de la integración.

A continuación se analizará el caso de estudio con el fin de identificar los sistemas de información de una organización, las bases de datos que emplean y en dado caso algún tipo de *middleware* que utilicen. La información obtenida servirá para tener una idea global de la situación de los sistemas de la organización, identificar las áreas de oportunidad y plantear una solución o estrategia referente a la integración de los mismos basándose en la información antes mencionada.

## SOLUCIONES COMERCIALES Y CASOS DE ESTUDIO

**4.1 SOLUCIONES COMERCIALES POR TIPO DE *MIDDLEWARE***

En el mercado, existe toda una gran variedad de *middleware* a elegir dependiendo de las necesidades de la empresa. A continuación se mencionan algunos ejemplos clasificándolos por el tipo de *middleware*:

**4.1.1 RPC (*Remote Procedure Call*)**

Como menciona Ruh (2000), los RPC están basados en la noción de un desarrollo distribuido de aplicaciones. Por ello, la mayoría del software existente son plataformas de desarrollo. Entre ellas, el sitio Middleware.org (2000) y Poujade (1997), mencionan los siguientes RPC con sus características particulares:

**Tabla 4.1. *Middleware* de RPC**

<b>Producto</b>	<b>Vendedor</b>	<b>Características</b>
JaRPC®	NC Laboratories ( <a href="http://www.nc-labs.com">http://www.nc-labs.com</a> )	Ofrece herramientas para desarrollar RPC para clientes y servidores en Java
PowerRPC®	Netbula ( <a href="http://www.netbula.com">http://www.netbula.com</a> )	Consiste en un compilador de RPC más una librería de funciones compatibles con C/C++
NobleNet®	RogueWave ( <a href="http://www.roguewave.com">http://www.roguewave.com</a> )	Ofrece un paquete integrado de herramientas cliente/servidor de RPC. Disponible para todas las estructuras de datos y las interfaces programables de aplicación (API, por sus siglas en inglés)
Distinct RPC® for Java	Distinct ( <a href="http://www.distinct.com">http://www.distinct.com</a> )	Conjunto de aplicaciones basado en Java para el desarrollo de aplicaciones cliente/servidor

La característica principal de los RPC consiste en que es necesario el desarrollo de la aplicación por parte de un programador que tenga conocimiento de los sistemas de información de la organización así como de las bases de datos que se empleen en la misma. Es por eso que este tipo de *middleware* es usado de forma más común en el nivel de método.

#### 4.1.2 MOM (*Message Oriented Middleware*)

De acuerdo a Vondrak (1997), el *middleware* orientado a mensajes es empleado para aplicaciones basadas en eventos es decir, cuando ocurre un evento el cliente comunica al servidor que se necesita llevar a cabo una acción.

La tabla 4.2 muestra algunos ejemplos de mensaje orientado a mensajes.

**Tabla 4.2 Middleware de MOM**

<b>Producto</b>	<b>Vendedor</b>	<b>Características</b>
MessageQ®	BEA Systems ( <a href="http://www.beasys.com">http://www.beasys.com</a> )	Permite comunicar a las aplicaciones existentes empleando la tecnología de colas de mensajes (MQ).
MQSeries®	IBM ( <a href="http://www.ibm.com">http://www.ibm.com</a> )	Permite a las aplicaciones intercambiar información a través de veinticinco diferentes sistemas operativos.
MSMQ®	Microsoft ( <a href="http://www.microsoft.com">http://www.microsoft.com</a> )	Ofrece la capacidad de comunicar las aplicaciones con otras aplicaciones a través de mensajes.
SmartSockets®	Talarian ( <a href="http://www.talarian.com">http://www.talarian.com</a> )	Consiste en un paquete de desarrollo que permite a las aplicaciones comunicarse entre sí a través de diferentes sistemas operativos.
TIB/RendezVous®	TIBCO ( <a href="http://www.tibco.com">http://www.tibco.com</a> )	Se trata de un sistema de transacciones basado en mensajes que emplea el modelo de publicación / suscripción.

Este tipo de *middleware* permite escoger entre tres tipos diferentes de aplicación: publicación/suscripción, colas de mensajes y un modelo punto a punto (Haase, 2001)

#### 4.1.3 *Middleware* orientado a bases de datos

El *middleware* orientado a las bases de datos permite la integración en el nivel de datos, así como la realización de consultas o movimiento de información a través de una red. Según Haas (2001), los sistemas de *middleware* de bases de datos ofrecen a los usuarios la habilidad de combinar datos de múltiples fuentes en una consulta sencilla con el fin de proveer extensibilidad y el encapsulamiento de dicha información, evitando así la creación de un almacén de datos o *datawarehouse*.

Tabla 4.3 *Middleware* orientado a bases de datos

Producto	Vendedor	Características
Direct Connect®	Sybase ( <a href="http://www.sybase.com">http://www.sybase.com</a> )	Permite el acceso a las bases de datos para su integración de un modo transparente. De igual manera puede ser usado en un <i>datawarehouse</i> y aplicaciones basadas en Internet.
DiscoveryLink®	IBM ( <a href="http://www.ibm.com">http://www.ibm.com</a> )	Provee a los usuarios la capacidad de generar consultas complejas a diferentes fuentes de datos.
Enterprise Gateway®	Informix Software ( <a href="http://www.informix.com">http://www.informix.com</a> )	Consiste en una herramienta que permite la migración de información de una base de datos a través de llamadas a un controlador de ODBC.
Simba Solutions®	Simba ( <a href="http://www.simba.com">http://www.simba.com</a> )	Provee un completo sistema de acceso a los datos incluyendo el desarrollo y la integración para las pruebas y operaciones finales.

El objetivo de este tipo de *middleware* es lograr la conexión a los datos independientemente de la plataforma o localización de los mismos.



#### 4.1.4 *Middleware* orientado a objetos distribuidos

De acuerdo a Ruh (2000), el funcionamiento de esta tecnología consiste en que una aplicación puede invocar a un objeto sin conocer su localización o dónde se encuentre almacenado. Esta capacidad permite a los componentes ser movidos, reemplazados o replicados sin afectar a los demás componentes.

CORBA es uno de los estándares más aceptados para el desarrollo de este tipo de *middleware*, pues permite a las aplicaciones comunicarse sin importar la localización, plataforma o ambiente computacional (Jaben-Eilon, 2000)

La tabla 4.4 muestra algunos ejemplos de este tipo de *middleware*:

Tabla 4.4 *Middleware* orientado a objetos distribuidos

Producto	Vendedor	Características
ObjectBroker®	BEA Systems ( <a href="http://www.beasys.com">http://www.beasys.com</a> )	Simplifica la creación de soluciones basadas en objetos distribuidos integrando las aplicaciones existentes así como el desarrollo de nuevas aplicaciones.
Orbix®	Iona ( <a href="http://www.iona.com">http://www.iona.com</a> )	Provee una plataforma para el despliegue de aplicaciones altamente distribuidas e integradas.
CORBAplus®	Expersoft	Permite el desarrollo y la administración de las aplicaciones de objetos distribuidos ofreciendo servicios de alto nivel y herramientas gráficas.
SOM/DSOM®	IBM ( <a href="http://www.ibm.com">http://www.ibm.com</a> )	Es utilizado para proveer mecanismos orientados a objetos con el fin de ser empleados en una variedad de lenguajes de programación.
Voyager®	ObjectSpace ( <a href="http://www.objectspace.com">http://www.objectspace.com</a> )	Consiste en un agente basado en Java con la capacidad de crear aplicaciones sofisticadas de red usando técnicas tradicionales de programación.

Con el fin de emplear este tipo de *middleware*, es necesario contar con conocimientos específicos del estándar de CORBA para poder aplicarlo en los sistemas donde se desee implementar.

#### 4.1.5 *Middleware* orientado a transacciones

Este tipo de *middleware* se puede dividir en dos: monitores de procesamientos de transacciones y servidores de aplicación.

##### **Monitores de procesamiento de transacciones**

Sadoski (1997) comenta que este tipo de tecnología controla las transacciones con las aplicaciones y realiza cálculos de los procesos organizacionales y actualiza las bases de datos. Provee servicios de aplicación a una gran cantidad de clientes en un ambiente distribuido de cliente/servidor.

En la siguiente tabla se mencionan algunos ejemplos:

**Tabla 4.5 *Middleware* de monitor de procesamiento de aplicaciones**

<b>Producto</b>	<b>Vendedor</b>	<b>Características</b>
CICS®	IBM ( <a href="http://www.ibm.com">http://www.ibm.com</a> )	Es un software de propósito general que es empleado en la mayoría de los casos para transacciones en línea.
Tuxedo®	BEA ( <a href="http://www.beasys.com">http://www.beasys.com</a> )	Provee un esquema para el desarrollo de aplicaciones cliente/servidor en ambientes heterogéneos y distribuidos.
Open/OLTP®	Unisys ( <a href="http://www.unisys.com">http://www.unisys.com</a> )	Permite a que las aplicaciones participen en transacciones cliente/servidor basadas en el estándar de procesamiento de transacciones distribuidas.

## Servidores de aplicación

Schussel (1995) describe que la arquitectura de los servidores de aplicaciones destinan la parte principal de una aplicación a un servidor para su ejecución en vez de colocarlo en la interfaz de usuario. El servidor en sí comparte la lógica del programa, los cálculos de la aplicación y un motor de búsqueda de datos.

A continuación se describen algunos ejemplos de este tipo de *middleware*

Tabla 4.6 *Middleware* de servidor de aplicaciones

Producto	Vendedor	Características
iPlanet Application Server®	iPlanet ( <a href="http://www.iplanet.com">http://www.iplanet.com</a> )	Provee una plataforma robusta basada en Java para el desarrollo, despliegue y la administración de servicios de aplicación.
WebLogic®	BEA ( <a href="http://www.beasys.com">http://www.beasys.com</a> )	Ofrece una solución completa para el despliegue de aplicaciones adaptables y personalizadas a través del uso de estándares como Java y CORBA.
WebSphere®	IBM ( <a href="http://www.ibm.com">http://www.ibm.com</a> )	Proporciona las herramientas necesarias para integrar y administrar aplicaciones de <i>e-business</i> soportando presentaciones dinámicas en Web hasta el procesamiento de transacciones sofisticadas.

En conclusión, cada uno de estos tipos de *middleware* puede ser usado de acuerdo a las necesidades de la organización y los sistemas de información que se deseen integrar. Enseguida se analizarán algunos casos de aplicación dependiendo del *middleware* empleado.

## 4.2 CASOS DE APLICACIÓN DE *MIDDLEWARE*

La gran variedad de *middleware* existente en el mercado ha permitido que varias empresas optimicen sus procesos internos a través del uso de software específico. A continuación se muestran algunos casos reales de aplicación empleando diferentes tipos de tecnologías.

Tabla 4.7 Caso de estudio de Chevron Corporation Fuente: [Talarian, 2000b]

<b>Compañía:</b>	Chevron Corporation ( <a href="http://www.chevron.com">http://www.chevron.com</a> )
<b>Situación:</b>	Conectar 3000 computadoras y navegadores de web a un sistema de información tanto geográfica como sísmica y un historial de perforaciones. Dicha información se encontraba almacenada en diferentes bases de datos dentro de una gran variedad de tecnologías de servidores.
<b>Tecnología Usada:</b>	<i>RPC, Object Request Brokers, CORBA y como producto Orbix®</i>
<b>Solución Implementada:</b>	Chevron Company ha desarrollado una serie de servidores de objetos de datos que abstraen los formatos propietarios de las fuentes de datos empleadas por las diferentes bases de datos de la organización. Estos servidores emplean Orbix® y su función consiste en ofrecer un despliegue de la información a lo largo del mundo de mas de 600 estaciones de trabajo UNIX y Windows. Aunado a esto, Chevron se encuentra construyendo un catálogo para administrar las diversas fuentes de datos de toda la empresa. Este esquema de integración le permitirá a la organización definir un modelo estándar en el cual puedan ser escritas las aplicaciones a futuro.

**Tabla 4.8 Caso de estudio de Whirlpool Corporation Fuente: [IBM, 2000]**

<b>Compañía:</b>	Whirlpool Corporation ( <a href="http://www.whirlpool.com">http://www.whirlpool.com</a> )
<b>Situación:</b>	Los métodos de procesamiento de información de Whirlpool eran ineficientes y costosos en cuanto a tiempo y capital. El 10% del ingreso total de la compañía generado por los vendedores no permitía mantener conexiones dedicadas de sistema a sistema.
<b>Tecnología usada:</b>	<i>Application Servers y como producto WebSphere® Application Server</i>
<b>Solución Implementada:</b>	A través del portal, llamado Whirlpool Web World, las empresas asociadas y vendedores seleccionan los artículos que desean ordenar seleccionando la clave y cantidad requerida. Igualmente, pueden conectarse para monitorear el estatus de su pedido. El proceso de ordenamiento ha reducido el costo por orden a menos de cinco dólares por transacción, es decir, al menos un 80% del costo total del pedido.

**Tabla 4.9 Caso de estudio de CBS SportsLine Fuente: [TIBCO, 2000]**

<b>Compañía:</b>	CBS Sportsline ( <a href="http://cbs.sportsline.com/">http://cbs.sportsline.com/</a> )
<b>Situación:</b>	Los sitios de web de CBS SportsLine requerían un sistema de <i>back-end</i> dinámico que sea efectivo, flexible, escalable y confiable las 24 horas para administrar la información que contenían.
<b>Tecnología usada:</b>	<i>Middleware orientado a mensajes y como producto TIB/RendezVous®</i>
<b>Solución Implementada:</b>	La empresa comenzó a emplear TIB/ActiveEnterprise® como un esquema de publicación para desplegar información deportiva de una fuente de datos externa. Posteriormente desarrollaron un proceso automático para convertir el contenido, generando páginas de HTML nuevas cada 30 o 40 segundos. El componente clave consistió en el software de envío de mensajes TIB/RendezVous® que permiten a los desarrolladores construir y desplegar aplicaciones distribuidas más fácil y más rápido empleando la tecnología de publicación/suscripción.

**Tabla 4.10 Caso de estudio de 3M Health Information Systems Fuente: [Talarian, 2000b]**

<b>Compañía:</b>	3M Health Information Systems ( <a href="http://www.3m.com/market/healthcare/hhs">http://www.3m.com/market/healthcare/hhs</a> )
<b>Situación:</b>	Extender las plataformas de hardware con el fin de ofrecer a los hospitales acceso instantáneo y confiable a información crítica de los pacientes.
<b>Tecnología usada:</b>	<i>Monitores de Procesamiento de Transacciones y como producto BEA Tuxedo®</i>
<b>Solución Implementada:</b>	La solución consistió en seleccionar una arquitectura de tres etapas empleando Oracle® como la base de datos, BEA Tuxedo® como el monitor de transacciones y un cliente basado en objetos de ActiveX. El sistema recaba información de una variedad de sistemas existentes y los integra a la base de datos. La información contenida en dicha base puede ser vista en cualquier terminal de trabajo de 3M o por cualquier cliente personalizado. Además el sistema se encarga de monitorear los datos para avisar a las personas responsables cualquier problema o inconsistencia cuando éste suceda.

Como conclusión, cada *middleware* ofrece soluciones específicas de acuerdo a las necesidades de la empresa. Se puede decir que no existe una solución única para la integración de aplicaciones empresariales, por lo cual es necesario un análisis completo de los sistemas de información con el fin de detectar la mejor opción para su implementación.

## **5.1 OBJETIVO**

El objetivo de la investigación de campo consiste en identificar las áreas organizacionales de una institución educativa privada que empleen sistemas de información con bases de datos pudiendo estar integrados o que se manejen en forma independiente con el fin de proponer una estrategia de integración de aplicaciones.

## **5.2 MUESTRA SELECCIONADA**

Con el fin de llevar a cabo la investigación de campo se definió como estudio de caso a una institución educativa que cuente con una variedad de departamentos que manejen sistemas de información y bases de datos que manipulen una alta cantidad de información. Dentro de dicha institución existe un gran número de unidades organizacionales por lo que sólo se seleccionaron aquéllas que contaban con sistemas de información y bases de datos que pudieran o no encontrarse integrados.

Una característica es que la información que se maneja debe, en ciertos aspectos, estar relacionada de cierta manera para identificar alguna conexión existente entre sus bases de datos con el fin de sugerir una estrategia de integración de aplicaciones.

De los departamentos involucrados en la Dirección de Informática, se entrevistaron a las personas quienes tenían algo que ver directamente con los sistemas de información y las bases de datos. En total, se entrevistaron a un total de 10 personas, las cuales se mencionan a continuación divididas por departamento:

### **Departamento de Servicios de Información Escolar**

Ing. César de Luna

Ing. Luis Zamora

Lic. Alvaro Espinosa

### **Departamento de Servicios de Información Administrativa**

Ing. Fernando Cota

### **Departamento de Ingeniería de Información**

Ing. Alfonso Treviño

Ing. Juan Rivera

### **Departamento de Biblioteca Digital**

Ing. Luis Cosmes

Ing. Rosa Gómez

### **Departamento de Servicios Auxiliares**

Ing. Marcela Grijalva

Ing. Roberto de la Rosa

## **5.3 METODOLOGÍA DE LA INVESTIGACIÓN**

La presente tesis se llevó a cabo utilizando la metodología cualitativa, empleando para ello una herramienta específica: la entrevista. A continuación se describe el panorama de las entrevistas que se llevaron a cabo para sustentar este trabajo.

### **Entrevistas.**

La información recopilada sirvió para identificar el uso que hacen de la información contenida en las bases de datos, las aplicaciones empleadas para acceder a los datos y la manera en que tienen integrados los datos con otras aplicaciones al igual que con otras bases de datos. Es muy importante recalcar que la entrevista realizada trata ciertos



aspectos técnicos relacionados con la base de datos así como las aplicaciones utilizadas en los departamentos y la tecnología empleada por lo que los resultados obtenidos serán analizados con la información que se haya recopilado en la revisión bibliográfica para contar con una base sólida para definir la estrategia a utilizar.

Con respecto a la información que se pretende recopilar de las entrevistas, se enlistan algunas características importantes referentes a la integración:

1. Los departamentos que utilicen sistemas de información y bases de datos.
2. Las aplicaciones y bases de datos que se usan en cada departamento.
3. La cantidad de sistemas heredados que tiene cada departamento.
4. Las tecnologías empleadas para conectar las bases de datos con las aplicaciones.
5. El conocimiento que se tiene de *middleware* para conectar las aplicaciones y bases de datos.

Con base a esta información, será posible la clasificación de los sistemas de información así como la identificación de las bases de datos que se encuentran integradas o no.

#### **5.4 PERFIL DE LAS PERSONAS ENTREVISTADAS**

Las entrevistas se llevaron a cabo con los encargados, personal técnico y usuarios del departamento de informática de la institución que manejaran sistemas de información y bases de datos. En general, estas personas conocen el manejo que se le da a las bases de datos y a las aplicaciones que acceden a ellas debido a que son directores de departamentos, subdirectores o personal de staff que lleva a cabo las operaciones diarias en dichos sistemas.

## 5.5 DISEÑO DE LA ENTREVISTA

Como se comentaba al principio, las preguntas de la entrevista estaban relacionadas para obtener información relativa a los siguientes aspectos:

- ❖ Sistemas de información
- ❖ Bases de datos
- ❖ Tecnologías de acceso a los datos
- ❖ *Middleware* empleado para conectar bases de datos

Cada una de estas áreas es fundamental para identificar las aplicaciones heredadas dentro de la organización y de esta manera plantear una estrategia de integración de aplicaciones global.

La guía completa del cuestionario en el cual se sustentaron las entrevistas realizadas durante la investigación de campo se encuentra en el Anexo I.

## **6.1 INFORMACIÓN GENERAL**

De acuerdo a las entrevistas realizadas con las personas de cada departamento, la información recopilada se puede clasificar en cuatro grandes secciones: sistemas de información, bases de datos, tecnologías de acceso a los datos y *middleware* empleado para conectar bases de datos. Cada una de estas cuatro secciones representa un aspecto específico en la selección de la estrategia de integración de aplicaciones.

Un punto importante a recalcar es que la tesis se llevó a cabo sobre un estudio de caso de una organización en particular por lo que, si se desea aplicar en otro caso semejante, es necesario el análisis de cada una de estas secciones con el fin de identificar las áreas donde pueda aplicarse el mismo nivel de integración.

## **6.2 RESULTADO DE LAS ENTREVISTAS**

En las entrevistas, se observó que el sistema utilizado en toda la institución es el BANNER® desarrollado por SCT (*Systems and Computer Technology*) (SCT, 2000). Los módulos comerciales con los que cuenta este sistema son: Recursos Humanos, Finanzas, Escolar, Cuentas por Cobrar y Ayudas Financiera. Actualmente, la institución cuenta con todos los módulos excepto con el de Ayudas Financieras. Aunado a estos módulos, se ha desarrollado internamente un componente el cual tiene la función de ser usado para la administración legal. El sistema, a través de cada uno de sus módulos, emplea la base de datos ORACLE® como base de datos oficial.

La decisión a un nivel directivo es la de mantener la información de la organización en una base de datos así como utilizar una aplicación estándar que acceda a esta base de

datos, sin embargo, se detectaron ciertos casos donde se empleaban sistemas heredados con otras bases de datos como se podrán ver en los resultados de las entrevistas.

Uno de los resultados más relevantes obtenidos en las entrevistas es la identificación de la inexistencia de un plan definido para llevar a cabo una integración de todos los datos que se manejan, tanto en la base de datos oficial, como en las otras bases de datos existentes. Se identificó también, en varios casos, que la información empleada por un departamento no puede ser usada directamente en el servidor donde se encuentra la base de datos, debido a que existe el peligro de sobrecargar el servidor de producción y disminuir el desempeño. Para solucionar este problema, en la mayoría de los departamentos, realizan un vaciado de los datos a otro servidor con la misma base de datos (ORACLE®) u otra base de datos alterna (INFORMIX®), posteriormente se realiza el procesamiento de los datos y finalmente se regresa la información a la base de datos original.

Otra característica que se ha observado es que la mayoría de los departamentos publican información a través de web usando diferentes tecnologías. Esto es un factor importante pues generalmente emplean el lenguaje (PL-SQL) que trae consigo el manejador de la base de datos, pero, de acuerdo a los comentarios de los entrevistados, si en un futuro existen nuevos desarrollos, sería mejor que se usara un lenguaje o una plataforma de desarrollo más general, ya que esto daría una mayor flexibilidad para soportar cualquier avance en caso de que la actual tecnología se volviera obsoleta.

A continuación se detalla cada una de las secciones con la información recopilada de las entrevistas. Basándose en estos datos y apoyados por la revisión bibliográfica será posible detectar las necesidades y la manera más óptima de realizar una integración de aplicaciones.

### **6.2.1 Sistemas de Información**

De acuerdo a la información recopilada, se pudo identificar al BANNER® como el sistema de información utilizado por todos los departamentos para la operación dentro de la organización. La organización cuenta con cinco módulos de este sistema: Escolar, Recursos Humanos, Finanzas, Cuentas por Cobrar y Ayudas Financieras o Becas. El

único departamento donde no se detectó este software fue la Biblioteca Digital, la cual emplea un sistema de información propietario (INNOPAC®) con una base de datos exclusiva el cual será manejado como una entidad aparte en el desarrollo de la estrategia de integración.

Cada uno de estos módulos es utilizado por un departamento como se menciona a continuación en la Tabla 6.1.

**Tabla 6.1: Módulos del sistema de información utilizados por cada departamento**

<b>Departamento</b>	<b>Módulo utilizado</b>
Servicios de Información Escolar	Escolar
Servicios de Información Administrativa	Recursos Humanos y Finanzas
Servicios Auxiliares	Cuentas por Cobrar
Biblioteca Digital	No emplea ningún módulo actualmente

Cada departamento tiene sus propios procesos y reglas referentes a la manera de operar de cada uno, a pesar de compartir el mismo sistema de información. En algunos de estos departamentos se han realizado desarrollos sobre la misma aplicación para aprovechar su funcionalidad y, en algunos casos, complementar los servicios ofrecidos por un módulo que no contemplaba toda la funcionalidad requerida.

Además del BANNER®, se han desarrollado en los departamentos de Servicios de Información Escolar y de Servicios Auxiliares, sistemas denominados satélites que se encargan de realizar procesos requeridos para la operación de cada departamento. En general, estos sistemas han sido desarrollados empleando varias tecnologías relacionadas como Java, CGI, C++, Javascript, etc. La característica principal de estos sistemas es que son ejecutados en equipos de cómputo diferentes donde se encuentra la base de datos oficial, con la finalidad de llevar a cabo el procesamiento de los datos y no afectar el desempeño.

### **6.2.2 Bases de datos**

Con el fin de mantener la información que genera cada departamento en una sola base de datos para facilitar los procesos organizacionales, se ha establecido ORACLE® como la base de datos oficial. Debido a la gran cantidad de información contenida, se han definido dominios o áreas restringidas de la base de datos con esquemas y usuarios propios para cada uno de los departamentos arriba mencionados, con el objetivo de mantener separada la operación de cada uno de ellos.

El sistema BANNER® puede ser visto como un ERP pues, apoyándonos en la literatura, podemos observar que “el aspecto primordial que debe prevalecer en un ERP es una base de datos y las herramientas de manipulación de datos que permiten almacenar, procesar y presentar la información de una manera apropiada cuando sea requerida por el usuario”. (Zhang, 2000)

Existe un departamento que se encarga de administrar la información general de todos los registros contenida en la base de datos. A este departamento se le conoce como Ingeniería de Información o Núcleo. Su función consiste en darle mantenimiento a la base de datos, crear respaldos y permitir un funcionamiento óptimo de la misma.

Como se comentaba en la sección de Sistemas de Información, los sistemas denominados satélites emplean una base de datos alterna para llevar a cabo su procesamiento. Esta base de datos puede ser INFORMIX® u ORACLE® la cual es almacenada en un servidor diferente para evitar disminuir el desempeño del equipo de cómputo donde reside la base de datos oficial.

### **6.2.3 Tecnologías de acceso a los datos**

Los módulos del BANNER® se comunican internamente con la base de datos ORACLE® usando el SQLNET que es una tecnología propietaria de la misma base de datos. Asimismo, las aplicaciones desarrolladas por cada departamento sobre BANNER® hacen uso de esta tecnología de conexión. Sin embargo hay muchas aplicaciones desarrolladas para ser utilizadas a través de web. El acceso a la base de datos empleando

este tipo de aplicaciones se realiza a través de PL/SQL, lenguaje propietario de la base de datos ORACLE®.

A diferencia de los sistemas que acceden a la base de datos oficial, los sistemas satélites utilizan tecnologías como Java o CGI para acceder a las bases de datos alternas que utilizan para el procesamiento de la información. Una de las características de estos sistemas es que nunca entran en contacto con la base de datos oficial pero la información que manejan proviene de ella. La carga de los datos a estas bases de datos alternas se lleva a cabo a través de un proceso “*batch*”. Este proceso consiste en la generación de un archivo de texto con un formato específico con información de la base de datos oficial. El archivo es cargado entonces en la base de datos alterna, donde se procesa la información y posteriormente es devuelto de la misma manera a la base de datos oficial llevando a cabo las validaciones correspondientes para evitar pérdidas en la integridad de la información.

#### **6.2.4 *Middleware* empleado para conectar bases de datos**

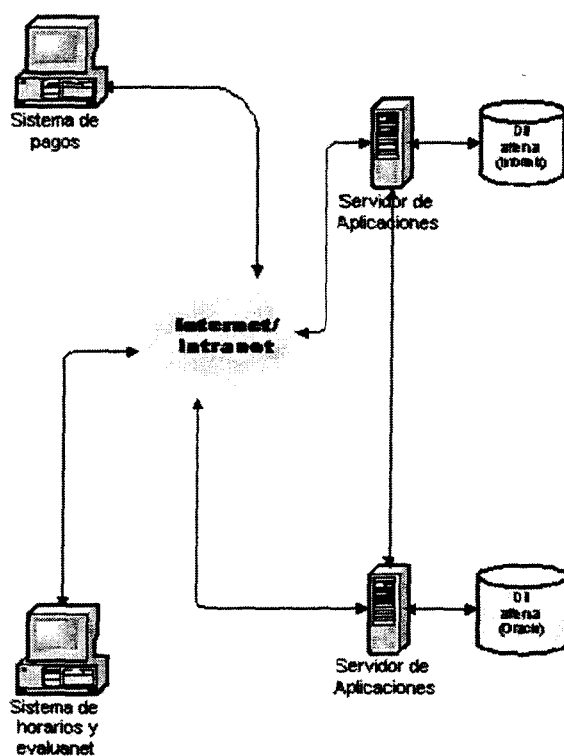
De los departamentos analizados no se detectó un *middleware* específico basado en software para conectar las bases de datos debido a dos razones: la primera es que la mayor parte de la información de todos los departamentos es mantenida en una sola base de datos (ORACLE®) y la segunda es porque la actualización de información en las bases de datos alternas la realizan a través de procesos *batch* como se comentaba en la sección anterior.

Sin embargo, de acuerdo a lo que comenta Linthicum (2000) en la literatura referente a *middleware* puede decirse que los sistemas satélites del departamento de Escolar y Cuentas por Cobrar, emplean un tipo de *middleware* conocido como servidor de aplicaciones debido a que estos equipos son empleados para procesar transacciones de aplicaciones basadas en Web, empleando lenguajes orientados a objetos como Java.

En breves palabras, el funcionamiento de este servidor de aplicaciones consiste en esperar peticiones de las aplicaciones basadas en web. Estas aplicaciones se encargan de enviar mensajes en formato de cadena o *string* al servidor el cual se conecta a la base de

datos para solicitar la información requerida. Cada petición hecha a la base de datos activa un procedimiento almacenado en la misma para devolver al usuario la información que solicita.

Por ejemplo, la funcionalidad principal de dicho sistema se lleva a cabo durante el proceso de inscripciones, cuando el alumno registra su clave de usuario y su contraseña a través de la interfaz de usuario del sistema de horarios del departamento de Escolar. Esta interfaz envía una petición al servidor de aplicaciones el cual solicita información referente si el alumno ya realizó el pago. Esto se hace en la base de datos alterna en INFORMIX® del departamento de Servicios Auxiliares a través del servidor de aplicaciones. Una vez llevada a cabo esta petición, se prosigue con la operación normal del sistema. La figura 6.1 explica detalladamente esta conexión entre ambos servidores de aplicaciones.



**Figura 6.1 Comunicación actual entre los servidores de aplicaciones de los sistemas satélites.**



## **6.3 DEPARTAMENTO DE SERVICIOS DE INFORMACIÓN ADMINISTRATIVA**

Este departamento cuenta con 44 personas y se encarga de administrar la información de las finanzas y del personal que labora en la organización. Dicha información es llenada por los departamentos administrativos y financieros de la institución como son: Tesorería, Contraloría y Contabilidad, entre otros. Como se mencionó al principio, el sistema de información empleado en el departamento es BANNER®. Los módulos de este sistema empleados por este departamento son los de Recursos Humanos y Finanzas. De este departamento se entrevistó en total a una persona: el director del área: Ing. Fernando Cota.

### ***6.3.1 Bases de datos***

Con respecto a las bases de datos, este departamento almacena la información en ORACLE®. Sin embargo, cierta información del área de contabilidad es mantenida en la base de datos INFORMIX® debido a que la Secretaría de Hacienda y Crédito Público les obliga a tener respaldos de cuando menos diez años atrás de información histórica en caso de se requiera alguna revisión. Al momento de la entrevista ya se encuentran almacenando la información que se va generando desde hace dos años aproximadamente en la base de datos oficial.

En este punto se detectó un área de oportunidad ya que la solución implementada en la actualidad es poner un atributo (campo) dentro de la base de datos ORACLE® donde muestre en qué posición de la base de datos INFORMIX® se encuentra la información que se desea revisar. De esta manera, el proceso de examinar la información en la base de datos INFORMIX® es manual, al tener que apuntar el número de referencia del registro deseado.

### ***6.3.2 Aplicaciones de las bases de datos***

No existe una aplicación propietaria para la base de datos donde se mantiene la información pero podría decirse que BANNER® es la aplicación principal de la base de

datos pues los usuarios de los diferentes departamentos administrativos lo emplean para manejar altas, consultas y actualizaciones.

La información que maneja este departamento también se accedida a través de web con el fin de hacer consultas o en algunos casos modificar ciertos datos pero sólo para algunas personas con permisos específicos como directores de área. Un ejemplo que se mencionó en la entrevista con respecto a la modificación de información a través de web es la que concierne a los presupuestos de sueldos que llevan a cabo exclusivamente los directores de departamento.

### **6.3.3 Tecnologías de acceso a los datos**

En cuanto a las tecnologías empleadas para acceder a los datos, BANNER® accesa a ORACLE® a través de un controlador (*driver*, por su nombre en inglés) que se conecta a la base de datos para extraer la información. Este proceso se lleva a cabo de manera natural de acuerdo a la implementación original del sistema a través del SQLNET. Con respecto a los accesos por web, la tecnología usada es PL/SQL el cual es un lenguaje propietario de la base de datos para desplegar información en formato HTML que le permite ser visualizada en cualquier navegador de Internet.

### **6.3.4 Middleware empleado para conectar bases de datos**

En esta sección se puede decir que la única base de datos fuera de la oficial es la de contabilidad (INFORMIX®) pero no cuentan con ninguna tecnología para compartirla sino sólo una referencia en ORACLE® como se mencionaba en la sección de base de datos. Al mantener toda la información relevante dentro de una misma base de datos, no hacen uso de un *middleware* específico. Sin embargo, para resolver la situación anterior de conectar las dos bases de datos (INFORMIX® y ORACLE®) sería necesario algún software de este tipo que permita un medio automatizado de recuperación de la información pero que no afecte el desempeño del equipo de cómputo donde reside la base de datos oficial.

## **6.4 DEPARTAMENTO DE SERVICIOS DE INFORMACIÓN ESCOLAR**

Este departamento cuenta con 50 personas aproximadamente y se encarga de administrar la información de todos los alumnos y profesores de la institución así como las encuestas y el roster institucional. Dicha información es introducida por los departamentos encargados de admisión de alumnos como son Escolar y Admisiones. La aplicación usada en el departamento para acceder a los datos es BANNER®. El módulo del sistema BANNER® empleado por este departamento es el de Escolar.

De este departamento se entrevistó en total a tres personas: el director del área (Ing. César de Luna), el encargado de procesos de inscripciones (Lic. Alvaro Espinosa) y el encargado de apoyo a escolar (Ing. Luis Zamora).

### **6.4.1 Bases de datos**

Con respecto a las bases de datos, este departamento almacena la información en ORACLE®. Hasta hace unos pocos meses mantenían una base de datos en INFORMIX® pero está descontinuada y se encuentra ahora en desuso. Toda la información de los alumnos, desde su registro en la institución hasta su salida, es administrada por este departamento.

De acuerdo a la información proporcionada por el encargado de procesos de inscripciones, el departamento hace uso de una base de datos que se encuentra en el departamento de Cuentas x Cobrar que está en INFORMIX® para detectar, durante el proceso de admisiones, si el alumno ya efectuó el pago para que pueda proceder a inscribirse. Este proceso se explicará más adelante en la sección de *middleware*.

### **6.4.2 Aplicaciones de las bases de datos**

Al igual que en el departamento de servicios de información administrativa, no existe una aplicación propietaria de la base de datos sino que a través del BANNER® los usuarios acceden a la base de datos para actualizar la información. Este departamento hace uso de aplicaciones basadas en web para desplegar, modificar o actualizar la información de la

base de datos. Entre las aplicaciones se encuentran la consulta de calificaciones, localización de alumnos, consultas de grupos, captura y consulta de calificaciones, entre otras aplicaciones.

La mayoría de las aplicaciones pueden ser accedidas por los alumnos o personas que tengan los permisos específicos para observar los datos por lo que se necesitan aplicaciones y equipos de cómputo que soporten gran cantidad de accesos a la base de datos.

#### ***6.4.3 Tecnologías de acceso a los datos***

En cuanto a las tecnologías empleadas para acceder a los datos, la mayoría de las aplicaciones basadas en web acceden a la base de datos a través del lenguaje propietario (PL/SQL). Además cuentan con una aplicación desarrollada en Java (inscripciones) que permite a los alumnos realizar su inscripción a través de Web. Esta aplicación se comunica con la de trámite de pagos (del departamento de servicios auxiliares). Todas las aplicaciones en web emplean información de la base de datos en ORACLE®.

Con respecto al uso de tecnologías empleadas para actualizar información en las bases de datos, el departamento cuenta con dos procedimientos de actualización de información los cuales son:

##### **❖ Proceso de Inscripciones**

Este proceso se lleva a cabo creando una réplica de la base de datos de ORACLE® y copiándola físicamente a otra computadora. En este equipo es donde se procesa toda la información relacionada con las inscripciones con el fin de evitar que no afecte el desempeño del equipo que tiene la base de datos original. Finalmente, la información es devuelta a la base de datos para actualizarse.

#### ❖ Proceso de Encuestas

Las encuestas se realizan de manera similar que el proceso de inscripciones con la diferencia que la información no es devuelta al ORACLE® sino que se utiliza para procesar la información y generar reportes.

La figura 6.2 muestra la manera como se lleva a cabo la carga de información de la base de datos oficial a la base de datos alterna.

#### ***6.4.4 Middleware empleado para conectar bases de datos***

En la entrevista realizada con el encargado de procesos de inscripciones, comentó que existe un proceso donde se utiliza un servidor de aplicaciones el cual se utiliza como intermediario en el proceso de inscripciones. Su función principal es conectar la información contenida en ORACLE® del sistema de inscripción académica con la información de pagos que está en INFORMIX®.

Cuando el alumno se inscribe remotamente, se hace una petición al servidor de aplicaciones que envía un mensaje tipo “string” (que es una clave del alumno) a la aplicación de trámite de pagos la cual emplea la base de datos INFORMIX® para confirmar si el pago existe. En caso afirmativo, se regresa una respuesta que puede ser un CERO (0) al servidor de aplicaciones y se permite el proceso de inscripción del alumno de manera remota. En caso de error, se regresa un número diferente de CERO que indica el error encontrado en la base de datos INFORMIX® por ejemplo: no existe el alumno, no ha pagado o no se realiza la conexión. En este último punto, la inscripción se realiza pero se queda pendiente la revisión del pago una vez que se reactive la conexión con el servidor.

## **6.5 DEPARTAMENTO DE INGENIERÍA DE INFORMACIÓN**

El funcionamiento de este departamento consiste en mantener una homogeneidad de la información contenida en la base de datos de todo el departamento de informática, controlando los accesos y manteniendo respaldos de la información existente en la base de datos oficial. Este departamento (denominado el Núcleo) se encarga de administrar la base de datos donde se tiene información general de los usuarios.

En este departamento se entrevistó al director del área: Ing. Alfonso Treviño y al encargado de soporte: Ing. Juan Rivera

### ***5.5.1 Bases de datos***

Toda la información de este departamento es almacenada en la misma base de datos oficial en ORACLE® que todos los demás departamentos usan para acceder a la información.

### ***5.5.2 Aplicaciones de las bases de datos***

Al ser una entidad de soporte a los procesos, no existe una aplicación propietaria diseñada por el departamento para acceder a la base de datos, a excepción del BANNER® que es usado a lo largo de la institución.

### ***5.5.3 Tecnologías de acceso a los datos***

Como en la mayoría de los departamentos, la tecnología para acceder a las bases de datos a través de web está basada en el lenguaje PL/SQL propietario del ORACLE®, y a través del BANNER® directamente con el SQLNET. La figura 6.3 muestra la manera como se conectan los diferentes usuarios al sistema BANNER®.

### ***5.5.4 Middleware empleado para conectar bases de datos***

Este departamento no ocupa ningún *middleware* especializado para conectarse a otras bases de datos ya que no tienen aplicaciones externas.

## **6.6 DEPARTAMENTO DE BIBLIOTECA DIGITAL**

Este departamento tiene como función proveer acceso a toda la información de la biblioteca digital a los alumnos, profesores o personas que trabajen dentro de la institución.

En este departamento se entrevistó al administrador del sistema INNOPAC® : Ing. Rosa Gómez y al encargado de soporte técnico del departamento: Ing. Luis Cosmes

### **6.6.1 Bases de datos**

La base de datos no es de ningún formato conocido pues fue diseñada por una empresa externa (*Innovate Interfaces Inc.*) que se especializa en desarrollar sistemas para bibliotecas. La información que se almacena en esa base de datos son datos bibliográficos, información de usuarios, órdenes de compra, cursos de reserva, etc.

Según comentaban las personas entrevistadas, la siguiente versión del software soportará ORACLE® como base de datos operacional por lo que se está planeando llevar a cabo esa migración en un futuro muy próximo.

### **6.6.2 Aplicaciones de las bases de datos**

La aplicación (INNOPAC®) es desarrollada por la empresa que diseñó las base de datos por lo que tiene acceso exclusivo y se puede decir que es una aplicación propietaria. Los módulos del sistema con los que cuenta este departamento son: adquisiciones, catalogación, control de autoridad, circulación, reserva manual y publicaciones periódicas. Cada uno de estos módulos desempeña un papel específico dentro del departamento pero aún así existen varios módulos que no han sido adquiridos por el departamento los cuales son reservaciones (*material booking*), módulo de encuadernación, órdenes electrónicas (*electronic order*) y reservas electrónicas (*electronic reserve*).

Con el fin de solventar estas necesidades o algunas otras que han surgido a lo largo de la utilización del sistema, se han desarrollado ciertas aplicaciones usando un lenguaje de programación conocido como EXPECT que se encarga de ejecutar procesos automáticos

como por ejemplo la actualización de información. Dichos procesos son ejecutados desde un servidor que es administrado en este departamento.

Esta base de datos es usada exclusivamente por este departamento y la información no es compartida ni accedida por ningún otro departamento.

### ***6.6.3 Tecnologías de acceso a los datos***

Las tecnologías de acceso se hacen a través de conexiones internas del INNOPAC® por lo que las actualizaciones se realizan empleando la misma aplicación.

Con respecto al uso de tecnologías externas empleadas para actualizar información en las bases de datos tanto de usuarios como de libros, el procedimiento a realizar es a través de un proceso *batch* donde se toma un listado de todos los usuarios de un archivo de texto con un formato específico que es enviado por la base de datos oficial ORACLE®. Se coloca este archivo en el servidor del departamento y usando un procedimiento definido por la aplicación, se procederá a cargar la información a la base de datos.

La carga de información de usuarios se divide de dos maneras: empleados y alumnos. Cada uno de ellos tiene un tiempo específico de carga al sistema dependiendo de las fechas de cierre de registros en la base de datos oficial, así como en las fechas límites importantes como graduaciones o fines de contratos.

### ***6.6.7 Middleware empleado para conectar bases de datos***

Este no ocupa ningún *middleware* especializado para conectarse a otras bases de datos pues manejan una base de datos exclusiva como se comentaba anteriormente.



## **6.7 DEPARTAMENTO DE SERVICIOS AUXILIARES**

La función de este departamento consiste en administrar los sistemas de información que soportan todos los procesos organizacionales relacionados con las finanzas y contabilidad como son las cajas de ingreso, facturación, becas, cuentas por cobrar, así como el trámite de pago de servicios. Comparten información con los departamentos de escolar, tesorería, becas y el departamento de ingeniería de información para llevar a cabo la mayoría de sus procesos organizacionales. Los módulos del sistema BANNER® empleado por este departamento es el de Cuentas por Cobrar y el de Ayudas Financieras (Becas).

### ***6.7.1 Bases de datos***

La información que generan los procesos de este departamento es almacenada en la base de datos ORACLE®. Al ser ésta la base de datos oficial, la actualización de la información se realiza en la misma instancia de la base de datos para ser utilizada por cualquier departamento que lo necesite.

Sin embargo, el proceso de trámite de pagos se encarga de almacenar y procesar la información en otra base de datos (INFORMIX®) debido a que, si estos procesos se realizan en el equipo de cómputo donde reside la base de datos oficial, puede afectar el desempeño de la máquina. Una vez llevado a cabo el proceso de los datos, se regresa la información actualizada a la instancia de ORACLE® que se encuentra en producción.

### ***6.7.2 Aplicaciones de las bases de datos***

Al igual que los demás departamentos, la aplicación que accede directamente a modificar la información contenida en la base de datos es BANNER®. Se puede decir que ésta es la aplicación propietaria y oficial de toda la organización. Sobre esta aplicación, el departamento ha desarrollado módulos para ser usados en los departamentos de Tesorería y Becas. Cuentan también con desarrollos en Java en la mayoría de los casos para realizar consultas por web de los estados de cuenta, solicitud de becas y trámite de pagos.

### **6.7.3 Tecnologías de acceso a los datos**

El acceso de la base de datos con BANNER® se realiza de manera natural al encontrarse la aplicación con la base de datos integradas desde la implementación del sistema de información. La tecnología para acceder a las bases de datos a través de web está basada en el lenguaje PL/SQL propietario del ORACLE®. La característica principal de este lenguaje es que es muy sencillo de usar y aprovecha toda la funcionalidad del ORACLE®.

Con respecto al uso de tecnologías empleadas para actualizar información en las bases de datos, el departamento cuenta con un procedimiento de actualización de información de las bases de datos que llevan a cabo a través de un proceso “*batch*”, el cual se explica a continuación:

- ❖ El proceso de carga de información del trámite de pagos consiste en la generación de información de la base de datos de ORACLE® en un archivo con formato específico, el cual es puesto en un servidor para posteriormente ser utilizado por la aplicación de trámite de pagos que almacenará la información en la base de datos alterna INFORMIX® con el objetivo de llevar a cabo el procesamiento en otro equipo de cómputo. Posteriormente se actualizará la información en la base de datos oficial con ORACLE® con este mismo proceso de carga del archivo con formato específico al servidor dicha base de datos.

### **6.7.4 Middleware empleado para conectar bases de datos**

Se detectó en este departamento la utilización de *middleware* a través de un servidor de aplicaciones para conectar la aplicación del trámite de pagos con la aplicación de inscripción académica.

El funcionamiento de este sistema puede ser visto gráficamente en la figura 5.1. Brevemente, la aplicación funciona de tal manera que el alumno accede a la aplicación de trámite de pagos para llevar a cabo su pago, el sistema se encarga de comunicarse con la base de datos alterna para almacenar la información del usuario junto con su registro de pago. Posteriormente, en conjunto con el sistema de inscripción académica, el usuario

solicita su horario y el servidor de aplicaciones de este sistema se conecta con el servidor de aplicaciones del sistema de pagos para requerir la información de pago del alumno. La información es recuperada de la base de datos INFORMIX® de la base de datos del sistema de trámite de pagos y se envía a la interfaz del usuario donde se procede a llevar a cabo su inscripción.

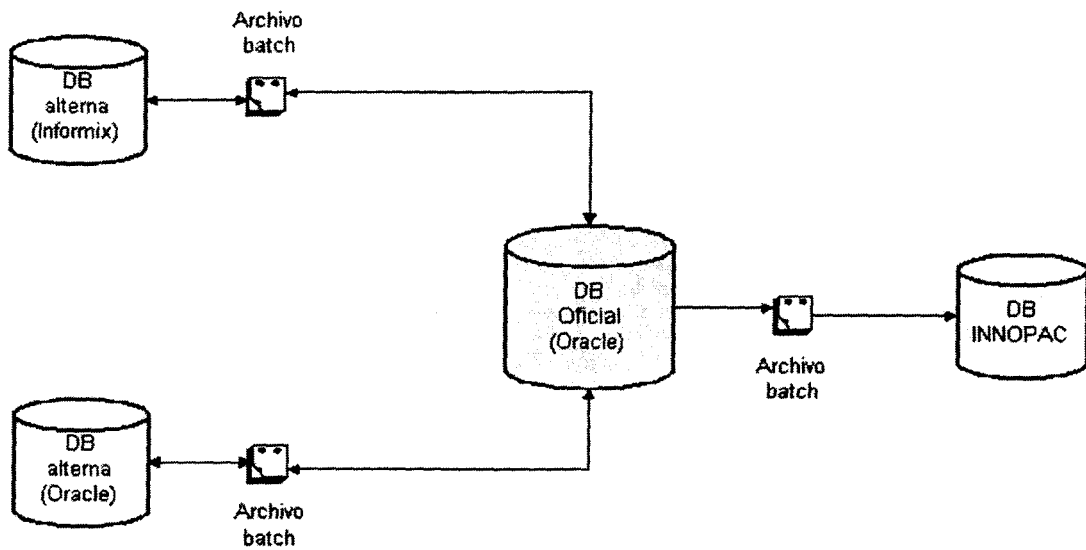


Figura 6.2 Diagrama de carga de datos de la base de datos oficial a bases de datos alterna

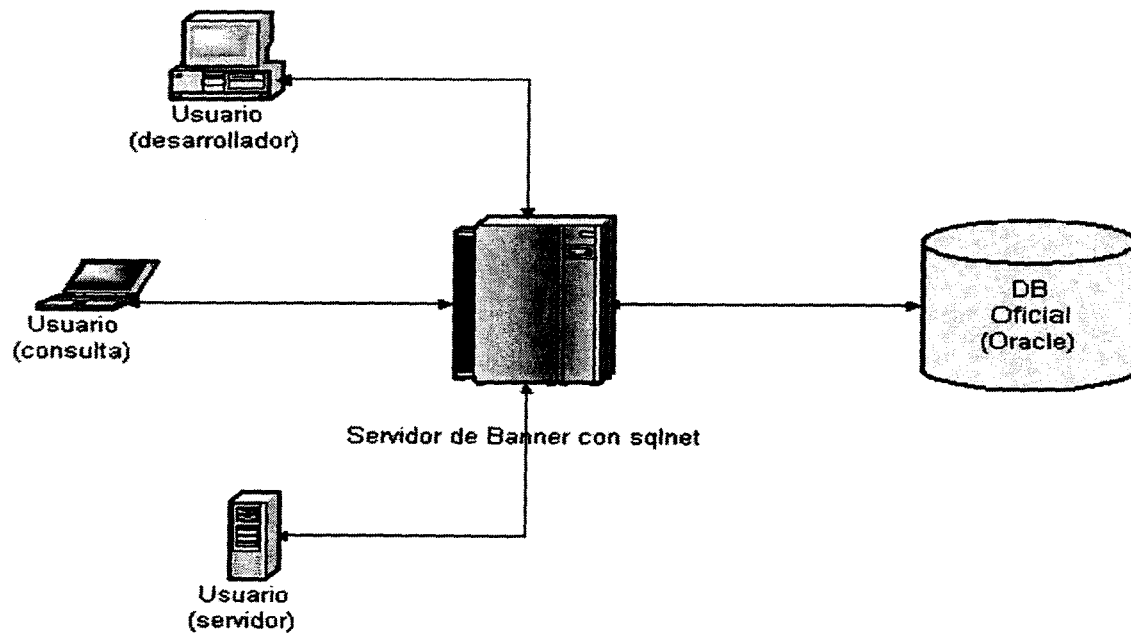


Figura 6.3 Diagrama de conexión entre BANNER® y ORACLE®

### **Definición de una Estrategia de Integración de Aplicaciones Empresariales de sistemas heredados dentro del nivel de datos**

---

La organización estudiada es una universidad privada que maneja toda la información de los alumnos como de los empleados en un sistema de información integral conocido como BANNER®. De la misma manera, toda la información de la organización es almacenada en la base de datos ORACLE® . Sin embargo, también existen ciertos sistemas que no forman parte del sistema oficial por lo que es en este punto donde se prestó un mayor interés a la hora de llevar a cabo las entrevistas para promover una estrategia efectiva.

Dentro de la organización, existe un departamento de informática que se encarga de llevar a cabo todos los procesos relacionados con la operación y mantenimiento tanto de la base de datos como del sistema de información. Este departamento se divide en varias secciones dentro de las cuales se llevaron a cabo las entrevistas a personas que manejaran estas aplicaciones.

A continuación se muestra la situación actual de la empresa, desventajas actuales y la estrategia propuesta de EAI para la organización.

#### **7.1 SITUACIÓN ACTUAL DE LA ORGANIZACIÓN**

La estrategia de mantener centralizada la información de la organización en una sola base de datos ORACLE® y a su vez permitiendo compartir dicha información a través de un sistema de información modular como el BANNER®, ha ofrecido en cierto modo, desde su implementación, un funcionamiento automatizado de los procesos organizacionales. De esta manera, se podría pensar que todos los departamentos se encuentran completamente integrados pero, de acuerdo a la información obtenida en las entrevistas, existen algunas aplicaciones que son parte importante de la operación de varios

departamentos que se realizan con sistemas de información y bases de datos alternas empleando en ocasiones tecnologías obsoletas de acceso a los datos como pudieran ser los procesos *batch*. Podrían llamarse obsoletas ya que, a pesar de ser funcionales, no ofrecen toda la capacidad de una aplicación que permita un acceso rápido o en tiempo real a la información, sin necesidad de generar archivos para el intercambio de información.

Sin embargo, uno de los aspectos más interesantes a considerar en el desarrollo de esta tesis, es el relacionado con la implementación de una estrategia de EAI para llevar a cabo la integración de dichas aplicaciones en el nivel de datos. Al momento de la entrevista, los departamentos de Escolar y Cuentas por Cobrar empleaban un servidor de aplicaciones como intermediario para conectar la información del sistema satélite de un departamento con el sistema satélite del otro con el fin de intercambiar información entre ambos. Sin embargo, detrás de estos sistemas satélites existen toda una variedad de desarrollos que soportan el funcionamiento y el mantenimiento de los servidores de aplicaciones en conjunto con las interfaces de usuario y los sistemas satélites.

Con respecto al departamento de Biblioteca Digital, el sistema que emplean está desarrollado de una manera cerrada por una empresa externa y accede a una base de datos propietaria. Sin embargo, en las entrevistas se obtuvo la información que la empresa que desarrolló el sistema permitirá implementar compatibilidad con la base de datos ORACLE®, por lo que se ha planeado llevar a cabo la migración de la información a esta base de datos próximamente. Así, la integración de datos en este departamento puede llevarse a cabo mucho más fácil al tener una base de datos alterna similar a la base de datos oficial.

## **7.2 DESVENTAJAS ACTUALES**

La mayor desventaja que se puede ver en continuar con la utilización de este tipo de servicio reside en que, posiblemente, los cambios futuros que ocurran, tanto en la base de datos como en el sistema de información oficial, provocarán la obsolescencia de los

sistemas satélites convirtiéndolos en un sistema heredado. De acuerdo a lo visto en el tema de sistemas heredados (capítulo 2), una solución consistiría en la recodificación de la aplicación involucrada o, en un caso extremo, la elaboración de otra aplicación completamente diferente que soporte los cambios llevados a cabo. En palabras textuales de David Linthicum: “La demanda de las empresas actualmente es compartir datos y procesos sin tener que llevar a cabo cambios radicales en las aplicaciones o las estructuras de datos. Empleando un método de EAI que pueda cumplir con esta característica permitirá que la integración con las aplicaciones y las bases de datos pueda ser funcional y redituable” (Linthicum, 1999)

Otro gran aspecto a considerar es la consistencia de la información en ambas bases de datos: la oficial y las alternas. Existen procedimientos de validación que son utilizados al momento de transferir la información a la base de datos oficial pero, debido a la gran cantidad de información que se procesa, estas validaciones pueden tomar cierto tiempo ocasionando quizás una disminución innecesaria en el desempeño del equipo de cómputo.

Una última desventaja se puede observar en la utilización de archivos *batch* para la carga de información. Sin duda, estos archivos resuelven el problema en la actualidad pero no es posible llevar a cabo actualizaciones de la información en tiempo real que pueden, en un futuro próximo convertirse en información crítica para la organización. De la misma manera, la información contenida en estos archivos puede encontrarse corrupta ocasionando con esto la inclusión de información inservible dentro de las bases de datos.

### **7.3 ESTRATEGIA DE IMPLEMENTACIÓN DEL NIVEL DE DATOS**

Retomando la idea original de EAI obtenida como conclusión en el capítulo 2, tenemos que se puede definir la Integración de Aplicaciones Empresariales como la necesidad de compartir los datos y procesos organizacionales, tomando en cuenta no llevar a cabo demasiados cambios a las aplicaciones o a las estructuras de datos actuales. Basándonos en esta definición podemos observar que, en el caso de la organización estudiada, su esquema de integración puede ser resuelto directamente en el acceso y transformación de

los datos enviados desde la base de datos oficial a la base de datos alterna. De esta manera, se evitará adentrarse en la lógica de programación del sistema de información disminuyendo la complejidad de la estrategia propuesta.

Sin embargo, de acuerdo a Linthicum, antes de la implementación del *middleware* seleccionado es primordial llevar a cabo tres pasos básicos que deben seguirse: (Linthicum, 2000)

1. la identificación de los datos;
2. la categorización de los datos, y
3. la elaboración de un modelo de metadatos empresarial que será usado como guía para integrar los diferentes repositorios que existan dentro de la empresa.

### **7.3.1 Identificación de los datos**

El primer punto consiste en identificar los sistemas candidatos con el fin de determinar qué bases de datos son empleadas. Dentro de la organización, estos sistemas son el sistema oficial de BANNER® con la base de datos oficial en ORACLE® , los sistemas satélites de inscripción masiva y encuestas de Escolar con su base de datos alterna en ORACLE®, el sistema satélite de trámite de pagos con su base de datos en INFORMIX® y la base de datos del departamento de biblioteca digital con el sistema INNOPAC® y su base de datos propietaria. Las bases de datos alternas tienen permisos para usuarios de cada departamento y la seguridad de estas bases de datos se encuentra implementada a través de cuentas de acceso y contraseñas. Actualmente la información contenida en las bases de datos alternas no es crítica por lo que puede considerarse un estatus de información en tiempo cercano ya que se actualiza en un determinado intervalo de tiempo como son temporada de inscripciones, tardías o fechas definidas por cada departamento. Los formatos de los datos son similares a los datos contenidos en la base de datos oficial por lo que no existe una diferencia de estructura en estas bases de datos alternas, excepto que son mantenidas en diferentes manejadores.

### **7.3.2 Categorización de los datos**

El segundo punto va más allá de las bases de datos y consiste en crear un catálogo de datos, tanto de la información almacenada en INFORMIX® como en ORACLE® . El catálogo servirá para definir la estructura de los datos dentro de la organización o, si se quiere dividir, para identificar la razón de ser de la información que maneja cada departamento en sus diferentes bases de datos.

Posteriormente, este catálogo servirá para la creación de un modelo de metadatos de toda la organización que es en sí el fundamento del nivel de datos de EAI.

### **7.3.3 Elaboración del modelo de metadatos**

El modelo de metadatos sirve para determinar cómo interactuarán los datos en la integración total de las aplicaciones y, a su vez, será utilizado como el directorio maestro para la solución de EAI, tanto para el nivel de datos, como para los niveles posteriores si se desean implementar.

Finalmente, este modelo de metadatos puede emplearse para crear un modelo físico o lógico de todas las bases de datos de la organización, con el objetivo de contar con un esquema general para visualizar toda la información que maneja cada departamento y facilitar integraciones posteriores.

## **7.4 ESTRATEGIA DE SELECCIÓN DEL *MIDDLEWARE***

Con el objetivo de plantear una estrategia para la selección de *middleware* es necesario enfocarse en la correcta selección de un software que sustituya los procesos *batch* empleados actualmente en los departamentos de Escolar y Servicios Auxiliares así como en el departamento de Biblioteca Digital para la carga de información de la base de datos oficial a la base de datos alterna.

De la misma manera, este *middleware* servirá para reemplazar el servidor de aplicaciones que valida los accesos entre los sistemas satélites de los departamentos de Escolar y



Servicios Auxiliares, con el fin de contar con una solución que sea mucho más confiable y con mayor disponibilidad.

Entre la gran variedad de *middleware* existente en el mercado y atendiendo las necesidades exclusivas de la organización estudiada, se llegó a la conclusión que el *middleware* basado en mensajes es la solución más óptima para la organización. Las siguientes razones que llevaron a esta decisión se encuentran apoyadas en documentos de varios autores como son Linthicum, Ruh, Talarian Corporation, entre otros; enfatizando también el rol que cumpliría en la estrategia de EAI dentro de la organización estudiada:

1. La característica principal de este tipo de *middleware* es que es asíncrono por lo cual se garantiza que el mensaje llegue a su destino a pesar de que éste no se encuentre disponible en algún momento de la operación.
  - a. Con base a esto podemos sustituir los procesos *batch* y los mensajes que envía el servidor de aplicaciones por un *middleware* de este tipo contando así con una mayor seguridad en la carga de datos o envío de mensajes aún cuando un servidor no se encuentre disponible.
2. Una ventaja importante radica en la flexibilidad del envío de los mensajes que puede ser adaptado a cualquier aplicación usando un protocolo estándar como TCP/IP.
  - a. Se puede implementar desde cualquier equipo de cómputo empleando cualquier tipo de aplicación que permita una conexión con este tipo de protocolo, facilitando así un desarrollo futuro con estándares definidos. Basándose en este protocolo, se pueden sustituir los mensajes que emplea el servidor de aplicaciones para comunicarse con los sistemas satélites y la base de datos, con información generada por el *middleware* con el fin de ser usado de una manera directa entre la aplicación del usuario y la base de datos alterna para el envío de información.
3. Tiene la capacidad de integrar aplicaciones nuevas, viejas, heredadas, centralizadas o distribuidas. Asimismo, es capaz de unir diferentes plataformas y

aplicaciones de desarrollo, conectándolas a través de una variedad de mecanismos para rutear la información entre ellos.

- a. Este aspecto es sin duda el más importante, pues ofrece soporte para todas las plataformas existentes en la organización y la capacidad de integrar los sistemas satélites que fueron desarrollados con una variedad de lenguajes de programación como Java, C++, CGI, etc.
4. Cuentan con herramientas de transformación de datos que permiten cambiar la manera como se presenta la información para cada aplicación.
    - a. En un futuro es posible que se pretenda emplear la información de los sistemas satélites en otras aplicaciones evitando con esto la necesidad de un desarrollo adicional para el despliegue de la información.
  5. La capacidad de integrar muchos sistemas al mismo tiempo sin la necesidad de otras conexiones o software adicional.
    - a. Este es otro aspecto importante a considerar pues permite la integración de todos los sistemas satélites actuales más la carga de información de la base de datos. Además, cualquier otro desarrollo posterior puede ser integrado a los sistemas actuales si la información es transferida a través del *middleware* seleccionado.

Tanto Linthicum, Ruh, Haase y la empresa Talarian coinciden que, en el mercado actual existen tres tipos de *middleware* orientado a mensajes (MOM, por sus siglas en inglés) los cuales se pueden clasificar en: publicación/suscripción (*pub/sub*), mensajes por cola (*Message Queueing*) y punto a punto (*point-to-point*).

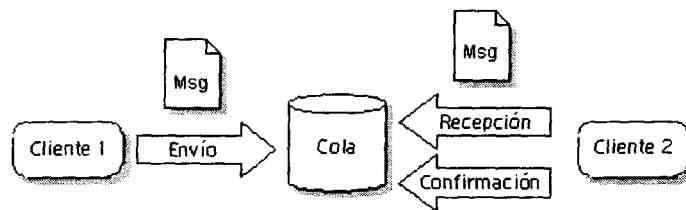
#### **Punto a punto (*point-to-point*)**

Las aplicaciones de mensajes **punto-a-punto** se encuentran relacionadas con los conceptos de colas de mensajes, emisores y receptores (figura 7.1). Cada mensaje es dirigido a una cola específica donde son almacenados y el cliente obtiene el mensaje de esta cola. (Haase, 2001)

La cola (*queue*) retiene todos los mensajes enviados hasta que éstos expiran o son requeridos por la aplicación receptora o cliente.

Los mensajes **punto-a-punto** tienen las siguientes características:

- ❖ Cada mensaje tiene solo un receptor
- ❖ No existen dependencias de tiempo entre el receptor y el emisor del mensaje. Esto es, el receptor puede ir a buscar el mensaje en cualquier momento aún y cuando a pesar que el emisor todavía no lo haya enviado.
- ❖ El receptor emite una señal de respuesta (*acknowledgement*) una vez que el mensaje ha sido procesado.



**Figura 7.1 Modelo Punto a punto. [Fuente: Kim Haase 2001]**

### **Publicación/Suscripción (*Pub/Sub*)**

El *middleware* de **publicación/suscripción** está especializado en integrar aplicaciones distribuidas. Los componentes básicos de este tipo de herramienta son los mensajes, un servicio de publicación y las aplicaciones suscriptoras (figura 7.2)

El servicio de publicación es donde los mensajes son enviados y ruteados de manera efectiva a las aplicaciones suscriptoras. Un servicio de publicación es similar a una transmisora de radio, donde los mensajes son publicados o difundidos a todas las aplicaciones que se han registrado para cierto tipo de mensajes (Talarian, 2000a)

Características básicas del *middleware* de publicación/suscripción.

- ❖ Cada mensaje puede tener múltiples receptores.
- ❖ Existe una dependencia de tiempo entre las aplicaciones publicistas y las suscriptoras debido a que el cliente que se suscribe puede recibir exclusivamente los mensajes que han sido publicados después de que se ha hecho la suscripción, y el suscriptor debe encontrarse activo para recibir dichos mensajes.

Ventajas:

- ❖ Comunicación asíncrona oportuna
- ❖ Mayor flexibilidad en el envío de mensajes

Desventajas:

- ❖ Soporte de transacciones no muy completo
- ❖ Complejidad en su implementación.
- ❖ Necesidad de una alta infraestructura de red

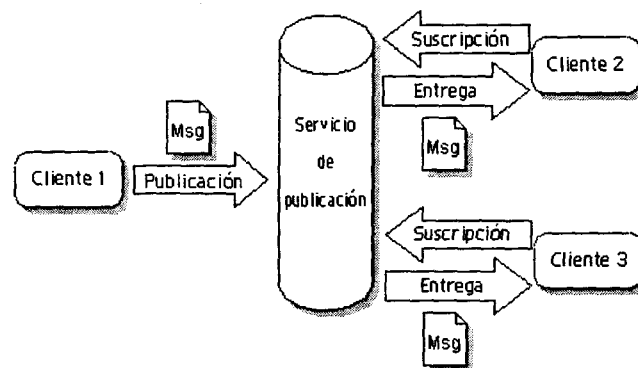


Figura 7.2 Modelo Publicación/Suscripción (Pub/Sub) [Fuente: Kim Haase 2001]

### **Cola de mensajes (*Message Queueing*)**

Este tipo de *middleware* puede ser usado en aplicaciones distribuidas. Los componentes básicos de un modelo de MQ son un mensaje y una cola (*queue*). Un mensaje es un segmento de información que es enviado entre dos aplicaciones y la cola es un contenedor que puede mantener y distribuir estos mensajes. Las aplicaciones que usan MQ se comunican asincrónicamente gracias a la cola (*queue*) donde residen los mensajes hasta que son recogidos por la aplicación receptora. (figura 7.3)

El *middleware* MQ provee medios confiables para que las operaciones de las aplicaciones se lleven a cabo sin ningún problema. Las aplicaciones que se comunican vía un MQ pueden no encontrarse relacionadas entre sí, es decir, una puede seguir trabajando mientras que la otra se encuentra ocupada o incluso desconectada de la red. La cola administra los mensajes para que no se pierdan, se confunda su entrega o se genere un mensaje duplicado, hasta que la aplicación receptora esté lista para recuperar dichos mensajes. (Talarian, 2000a)

Algunos ejemplos de aplicación de este tipo de *middleware* son:

- ❖ Aplicaciones de web que deben comunicarse con uno o más sistemas heredados para procesar una orden.
- ❖ Transacciones en los sistemas de *back-end* donde la pérdida de conexión o servicio no debe causar una pérdida en la transacción.

Ventajas:

- ❖ Necesidad de medios altamente confiables de comunicación sobre redes no confiables, como Internet.
- ❖ Necesidad de comunicación asíncrona.
- ❖ Necesidad de integrar aplicaciones emergentes o heredadas.

Desventajas:

- ❖ Administración y configuración un poco compleja.
- ❖ La comunicación asíncrona puede ser demasiado lenta si la cola está sobrecargada.

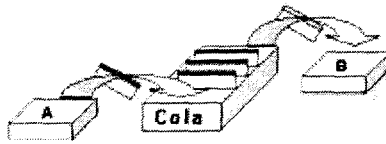


Figura 7.3 Modelo de colas de mensajes [Fuente: Talarian 2000a]

Analizando detenidamente cada uno de los diferentes tipos de MOM, el que cumple con todas las características necesarias para la integración de las aplicaciones de los sistemas satélites y la carga de datos de la base de datos oficial a la base de datos alterna es el *middleware* de colas de mensajes (MQ).

Las razones por las que se tomó esta decisión son las siguientes:

1. Existe actualmente la infraestructura capaz de mantener una cola de mensajes que los almacene para ser enviados a los sistemas satélites o enviar información para ser cargada en las bases de datos alternas.
2. Este tipo de *middleware* permite mantener todo el conjunto de aplicaciones funcionando el mayor tiempo posible en los períodos críticos de operación a pesar de que exista una falla en algunos de los sistemas así como en la red misma.
3. La facilidad de integrar el *middleware* con otros sistemas reduce el costo de recodificar las aplicaciones existentes así como también la adquisición de infraestructura o software más reciente que lleve a cabo las mismas funciones.

A continuación se describe la implementación global tanto del modelo de metadatos como del *middleware* seleccionado con el fin de ofrecer la estrategia de integración de aplicaciones en el nivel de datos.

## 7.5 IMPLEMENTACIÓN DE LA ESTRATEGIA DE EAI

NC.Focus define el uso de *middleware* orientado a mensajes como una capa del EAI que provee una herramienta para transformar el formato de un documento de una aplicación en un formato adecuado para ser usado por la otra aplicación. Esta capa depende mucho de la existencia de un repositorio de reglas o metadatos que describan el esquema de los tipos de mensajes o documentos, dónde deberían ser ruteadas dentro de la infraestructura de EAI, y las reglas para procesar esta información. (NC.Focus, 1999)

Apoyándonos en el punto 7.3 que trata sobre la implementación del nivel de datos, emplearemos el modelo de metadatos que la organización defina para implementar las reglas y el manejo de la información en las bases de datos actuales dentro de la organización. Posteriormente, se empleará el *middleware* seleccionado para sustituir la conexión existente entre los sistemas satelitales y las bases de datos así como la transferencia de información de los procesos *batch* actuales.

La decisión de implementar un *middleware* orientando a mensajes con soporte a colas de mensajes cubre varios aspectos que ya han sido explicados en el punto anterior pero sin duda uno de los más importantes y que viene a redondear la estrategia de EAI consiste en el costo que involucra la implementación de un software de este tipo.

En primer lugar, es posible elegir entre una gran variedad de este tipo de *middleware* desde el software especializado en un sola base de datos así como el que permite conectarse a varias bases de datos y soporte varios protocolos o pueda usarse en diferentes plataformas.

Este tipo de software puede ser reutilizado en otras aplicaciones siempre y cuando se analice detalladamente la situación de las mismas y pueda acoplarse a los requerimientos que se le piden para interactuar con las bases de datos y las aplicaciones a las que se desee conectarlas.

Finalmente, la integración de aplicaciones al nivel de datos a través de un *middleware* es la base para los niveles siguientes de integración por lo que la correcta selección traería

como consecuencia costos menores en la implementación de una estrategia completa de EAI en toda la organización.

La figura 7.4 presenta un ejemplo global de la integración de aplicaciones en todos sus niveles. Como se puede apreciar, existe una variedad de sistemas, servicios y niveles de implementación. Dentro de la capa de los servicios de **Ruteo e Intermediación (Routing and Brokering)** se implementará el *middleware* orientado a mensajes a través de colas que permitirá conectar los sistemas satelitales con la información de las bases de datos alternas.

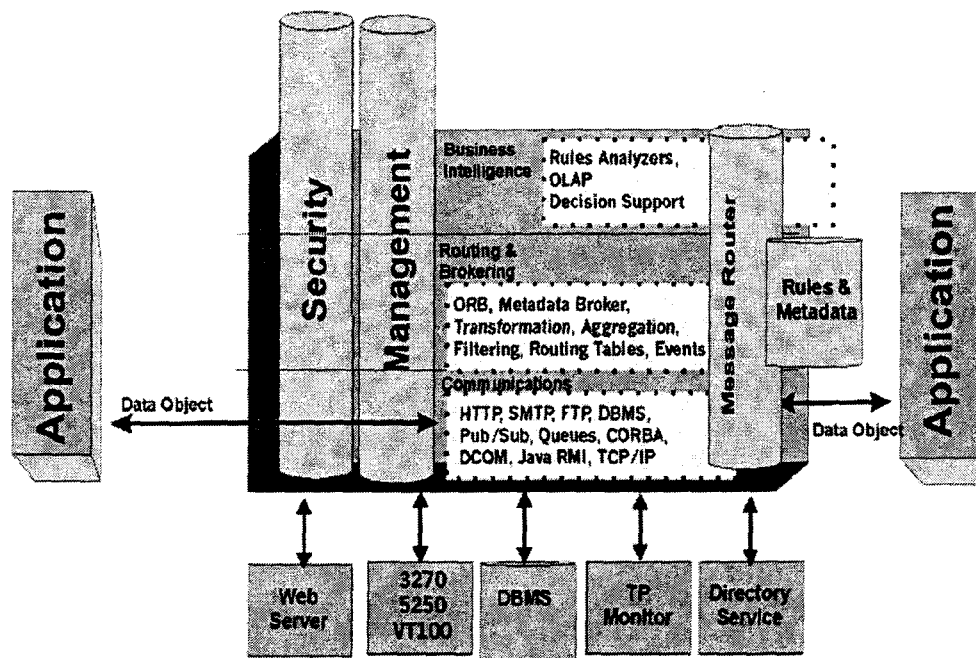


Figura 7.4 Diagrama de una infraestructura de EAI. [Fuente: NC.Focus, 1999]

Basándose en la información recopilada de los diferentes departamentos de informática, se llevó a cabo un diagrama representativo de los sistemas heredados existentes en esta dirección así como el *middleware* que se emplea. La figura 7.5 ejemplifica el estado actual de los sistemas y la infraestructura computacional empleada para llevar a cabo las operaciones de dichos departamentos.



En este diagrama se detalla la situación actual de los sistemas satélites, la carga de información a las bases de datos alternas por parte de los departamentos de Escolar y de Servicios Auxiliares, así como la aplicación empleada por el departamento de Biblioteca Digital y la manera como se lleva a cabo la carga de información de la base de datos oficial a la base de datos de dicho sistema.

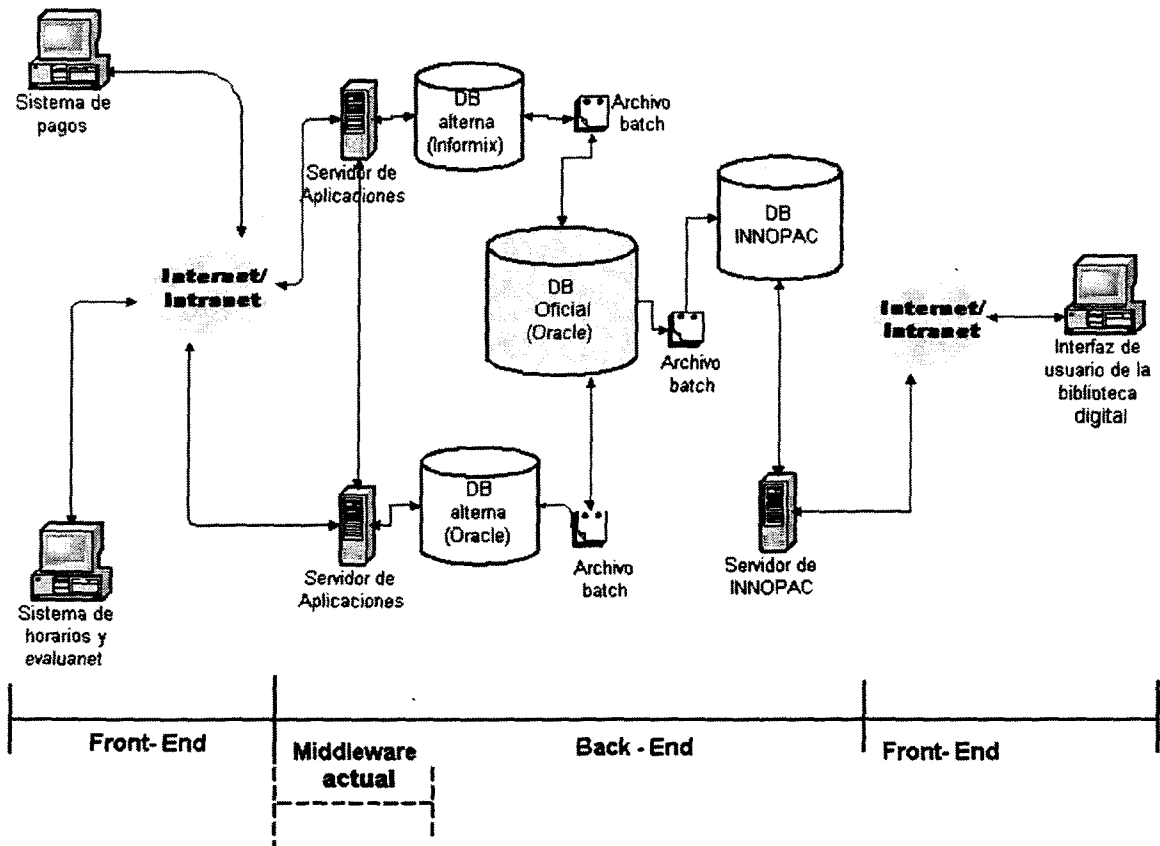


Figura 7.5 Diagrama de la infraestructura actual de la organización.

Una vez que se cuenta con el modelo de metadatos de la organización y se ha seleccionado el *middleware* orientado a mensajes a través del empleo de colas de mensajes, es preciso definir las aplicaciones que se verán afectadas para llevar a cabo la integración de aplicaciones.

La figura 7.6 representa la estrategia propuesta empleando el *middleware* orientado a mensajes del tipo de mensaje de colas (MQ) sustituyendo los archivos *batch* para la carga de información a las bases de datos alternas de los departamentos de Escolar y Servicios

Auxiliares así como la Biblioteca Digital. De la misma manera, este *middleware* sustituirá el servidor de aplicaciones empleado para la comunicación de los sistemas satélites con el fin de ofrecer un servicio más eficiente en cuanto a tiempo de conexión a la red.

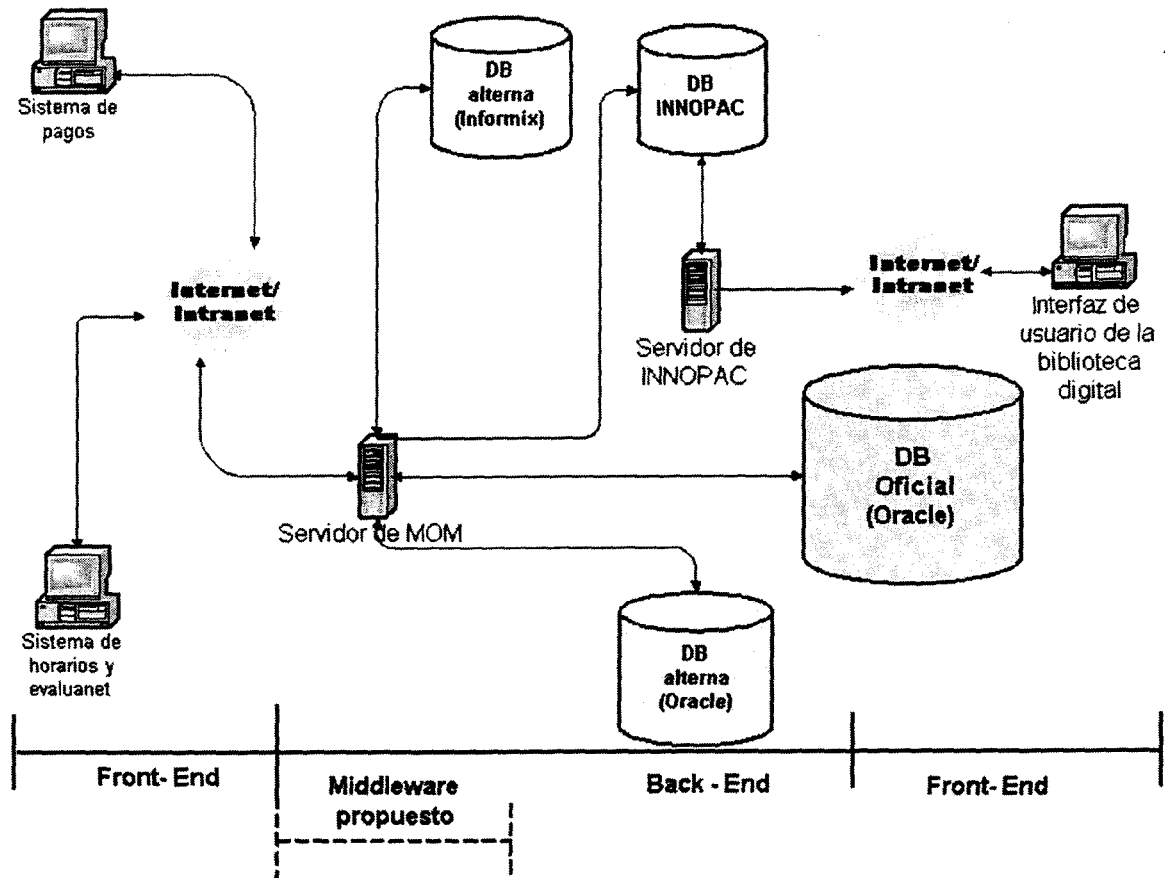


Figura 7.6 Diagrama propuesto de EAI para la organización.

De los dos servidores de aplicaciones existentes, uno será utilizado para mantener en producción el *middleware* mientras que el otro puede ser liberado de su funcionalidad para ser usado como equipo de pruebas o como segundo respaldo en caso de que el primero llegue a fallar por alguna causa.

La carga de información de la base de datos oficial hacia las bases de datos alterna ya no será realizada a través de un proceso *batch* sino que toda la información ahora será controlada por el servidor del *middleware* evitando la creación de archivos de este tipo que puedan contener errores en algún momento de la generación o la carga de los mismos

a las bases de datos alternas. Asimismo, el envío de los datos directamente a la base de datos alterna ofrece información actualizada y en tiempo real que permita reducir los errores en la información almacenada en las base de datos alternas.

Finalmente, con respecto al departamento de Biblioteca Digital, y de acuerdo a la información recopilada en las entrevistas, la base de datos de su sistema INNOPAC® será cambiada a ORACLE® por lo que el proceso de carga de información podrá llevarse a cabo de manera semejante con el *middleware*, empleando la actualización de los datos en tiempo real a través de *triggers* o eventos que permitan detectar cambios en la base de datos oficial para poder ser actualizados al momento en todas las bases de datos alternas conectadas a través del *middleware* seleccionado.

En general, los puntos que cubre el empleo de la metodología de EAI dentro de la organización son:

1. Los servidores de aplicación solucionan el problema de comunicación entre los sistemas satelitales pero si en algún momento alguno de ellos se desconecta o se encuentra fuera de servicio se perderá la comunicación y por consiguiente toda la funcionalidad de dichos sistemas.
2. La carga de información a través de procesos *batch* es sustituida por el *middleware* que permitirá una actualización en tiempo real de la información contenida en las bases de datos alternas.
3. Se definirá un modelo de metadatos de todas las bases de datos empleadas en la organización para poder detectar la relación existente entre ellas y poder llevar a cabo una integración posterior con mayor facilidad.
4. Las aplicaciones que sean utilizadas en otros departamentos pueden ser fácilmente integradas si se define la información que debe recibir de la base de datos oficial e incluirla en el servidor que mantiene el *middleware* seleccionado.

La estrategia propuesta viene a cubrir algunos aspectos que son ciertamente básicos en una estrategia de integración de aplicaciones como son el uso de un modelo de metadatos organizacional, definición de un *middleware* estándar para la transferencia de información y la reestructuración de los sistemas empleados para el manejo de información interna. Cualquier infraestructura actual con la ayuda de una correcta estrategia de EAI permite integrar las aplicaciones que la conforman siempre y cuando se lleve a cabo un análisis profundo y detallado de la situación de los sistemas de información que la conformen.

La correcta definición así como la implementación de una integración de datos es el primer paso que debe tomarse dentro de una estrategia de EAI si desea llevarse a cabo una integración de aplicaciones empresariales global que comprenda los cuatro niveles. El nivel de datos es sin lugar a dudas el escalón más importante pues permite desarrollar los demás niveles con una base sólida tomando en cuenta las herramientas definidas mente como es el modelo de metadatos empresarial y la selección de un *middleware* que soporte las aplicaciones existentes así como los futuros desarrollos.

La estrategia de la implementación de nivel de datos definida como objetivo de esta tesis puede ser empleada como base para cualquier organización que desee llevar a cabo una integración similar tomando en cuenta sus sistemas de información y las necesidades de crecimiento futuro para decidir la mejor alternativa en cuanto a la selección de sus herramientas de integración de información.

## CAPITULO 8

# CONCLUSIONES Y TRABAJOS FUTUROS

---

En este capítulo se presentan las conclusiones a las que se llegaron con la realización de este trabajo de investigación. Se presentan también recomendaciones sobre el desarrollo de trabajos futuros basados en la metodología de la Integración de Aplicaciones.

### 8.1 CONCLUSIONES

Actualmente, el surgimiento de diferentes sistemas de manejo de información empresarial como son los ERP han venido a subsanar las deficiencias de aplicaciones aisladas pero, a pesar de esto todavía persiste el problema de poder acoplar estas “islas” a los sistemas actuales para compartir al menos la información que se ha venido utilizando para la operación de la organización. Es en este punto donde recae la utilización de una metodología de EAI que permita ensamblar tanto los sistemas heredados como aquéllos que se vayan adquiriendo para mantener a la empresa operando sin problemas.

A lo largo de la tesis se han analizado todos los aspectos necesarios para llevar a cabo una correcta integración de aplicaciones empresariales. A pesar de que existen ciertas diferencias por parte de los autores en los niveles de integración, la mayoría de ellos coincide que el primer paso para realizar una integración global es afinar y estructurar la información contenida en las bases de datos de la organización.

El nivel de datos viene a ser el primer paso que debe realizarse si se desea llevar a cabo una integración global de toda la organización. El factor primordial en este nivel consiste en la correcta selección y aplicación de una estrategia que vaya desde la identificación de los sistemas empleados dentro de la organización, su funcionalidad y las bases de datos que son empleadas; después es necesario la definición de un modelo de metadatos que permita identificar la información contenida en todas las bases de datos de la organización, su empleo y aquellas personas que tiene contacto con dicha información; y

para concluir, la evaluación, selección e implementación del *middleware* que cumpla con los requerimientos de las aplicaciones así como las bases de datos de la organización.

En fin, la integración de aplicaciones en todos sus niveles es un trabajo que requiere mucho más que conocimientos técnicos de las aplicaciones y las bases de datos existentes dentro de una organización. Es primordial que exista un involucramiento total de todos los departamentos comenzando por la alta dirección y a su vez que comprenda la magnitud del proyecto que se desea realizar para aportar los suficientes recursos que permitan llegar a una conclusión exitosa.

## **8.2 TRABAJOS FUTUROS**

El objetivo de la tesis es presentar una estrategia de integración de aplicaciones exclusivamente para el nivel de datos con el fin de acoplar los sistemas con las bases de datos de la organización estudiada. Como consecuencia, quedan abiertos ciertos puntos los cuales se mencionan a continuación:

- ❖ Basándose en este estudio, existe la posibilidad de continuar con la realización de un análisis posterior para llevar a cabo la integración de aplicaciones a otros niveles como son los de aplicación, interfaces de programación o interfaces de usuario.
- ❖ La definición de una guía para el desarrollo o adquisición de aplicaciones que cumplan con el modelo de metadatos definido por la organización con el fin de acoplar los sistemas con las bases de datos de la organización.
- ❖ Tomando en cuenta la cantidad de sistemas heredados con los que cuente la organización, se puede analizar la creación de un área (dependiente del departamento de informática) que tenga como tarea la evaluación de calidad de los sistemas que se vayan generando o adquiriendo y que se encargue de llevar a

cabo pruebas constantes del funcionamiento de las aplicaciones para evitar la creación de futuros sistemas heredados.

- ❖ Un aspecto que puede ser llevado a cabo para redondear el análisis realizado es la implementación de un programa en toda la organización que recopile sugerencias de manera periódica de los usuarios de los sistemas de información, con el objetivo de aprovechar al máximo la integración llevada a cabo y modificarla según sean las necesidades de los interesados.

Existen muchas áreas de oportunidad para la mejora continua de los sistemas de información de toda una organización que va desde lo más básico, pasando por aspectos técnicos hasta la completa redefinición de sus sistemas. La mayoría de las organizaciones no cuenta con un área destinada a evaluar las aplicaciones actuales y el impacto que éstas tienen hacia los usuarios finales, por ello es primordial llevar a cabo un análisis que permita identificar los puntos donde se necesite llevar cabo una estrategia de optimización.

Para concluir, es importante tener en cuenta que esta tesis trató sobre uno de estos aspectos de mejora continua que es la integración de aplicaciones y puede ser tomado como punto de partida para muchas otras organizaciones que deseen comenzar a establecer una infraestructura robusta de tecnologías de información.

CUESTIONARIO DE APOYO PARA LAS ENTREVISTAS

---

---

1. ¿Cuáles son las bases de datos que se manejan en su departamento?

\_\_\_\_\_ Oracle                      \_\_\_\_\_ Informix                      \_\_\_\_\_ SQL Server

\_\_\_\_\_ MySQL

\_\_\_\_\_ Otro (especifique): \_\_\_\_\_

2. ¿Qué tipo de información se almacena en estas bases de datos?

3.- ¿Existe una aplicación propietaria para cada base de datos?

Sí \_\_\_\_\_                      No \_\_\_\_\_

4. - ¿Cuáles son estas aplicaciones propietarias?

5.- ¿La información contenida en las bases de datos puede ser accedida (altas, bajas, cambios, consultas) por otros sistemas de información además de la aplicación propietaria?

Sí \_\_\_\_\_                      No \_\_\_\_\_



6.- ¿Qué aplicaciones no propietarias acceden a estas bases de datos?

7.- ¿Qué tecnologías usan estas aplicaciones externas para acceder a las bases de datos?

_____ ODBC	_____ Java	_____ XML
_____ Scripts	_____ PL/SQL	

\_\_\_\_\_ Otro (especifique):

8.- ¿La información contenida en la base de datos es usada en otros departamentos?

Sí \_\_\_\_\_ No \_\_\_\_\_

9.- ¿Qué tecnologías o aplicaciones se utilizan para compartir información entre las bases de datos?

_____ ODBC	_____ Procesos "Batch"	_____ Middleware
------------	------------------------	------------------

\_\_\_\_\_ Otro (especifique):

*Middleware* se refiere al software que facilita la comunicación entre dos aplicaciones o bases de datos. A continuación se tratarán preguntas relacionadas al *middleware* empleado en el departamento para conectar los sistemas o las bases de datos:

10.- ¿Conoce los siguientes tipos de *middleware*:

\_\_\_\_\_ Remote Procedure Call (RPC)

\_\_\_\_\_ *Middleware* Orientado a Mensajes (MOM)

\_\_\_\_\_ Objetos distribuidos

\_\_\_\_\_ *Middleware* Orientado a bases de datos

\_\_\_\_\_ *Middleware* Orientado a transacciones

\_\_\_\_\_ Application Servers

\_\_\_\_\_ Otros (especifique):

11.- De los siguientes tipos de *middleware*, ¿cuáles emplean en sus sistemas?

\_\_\_\_\_ Remote Procedure Call (RPC)

\_\_\_\_\_ *Middleware* Orientado a Mensajes (MOM)

\_\_\_\_\_ Objetos distribuidos

\_\_\_\_\_ *Middleware* Orientado a bases de datos

\_\_\_\_ *Middleware* Orientado a transacciones

\_\_\_\_ Application Servers

\_\_\_\_ Otros (especifique):

## RESULTADOS GENERALES DE LAS ENTREVISTAS

---

El anexo que se presenta a continuación muestra un concentrado de las respuestas obtenidas de acuerdo a las entrevistas llevadas a cabo con el cuestionario anterior.

1. *¿Cuáles son las bases de datos que se manejan en su departamento?*

8 Oracle                      6 Informix                      \_\_\_\_\_ SQL Server

\_\_\_\_\_ MySQL

2 Otro (especifique): Innovate Interfaces Inc.

2. *¿Qué tipo de información se almacena en estas bases de datos?*

Informix: Algunos datos de contabilidad y pagos.

Oracle: Información de personal, alumnos, profesores, datos de encuestas, horarios.

Innovate Interfaces Inc: datos bibliográficos, usuarios, órdenes de compra, cursos de reserva.

3.- *¿Existe una aplicación propietaria para cada base de datos?*

Sí 10                      No \_\_\_\_\_

4. - *¿Cuáles son estas aplicaciones propietarias?*

BANNER: 8

INNOPAC: 2

5.- *¿La información contenida en las bases de datos puede ser accedida (altas, bajas, cambios, consultas) por otros sistemas de información además de la aplicación propietaria?*

Sí 10                      No \_\_\_\_\_

6.- ¿Qué aplicaciones no propietarias acceden a estas bases de datos?

Navegadores de web (browsers) y aplicaciones desarrolladas internamente.

7.- ¿Qué tecnologías usan estas aplicaciones externas para acceder a las bases de datos?

<u>1</u> ODBC	<u>5</u> Java	<u>        </u> XML
<u>2</u> Scripts	<u>8</u> PL/SQL	

2 Otro (especifique): herramientas internas del INNOPAC, HTML

8.- ¿La información contenida en la base de datos es usada en otros departamentos?

Sí 9 No 1

9.- ¿Qué tecnologías o aplicaciones se utilizan para compartir información entre las bases de datos?

<u>        </u> ODBC	<u>8</u> Procesos "Batch"	<u>2</u> Middleware
----------------------	---------------------------	---------------------

         Otro (especifique):

*Middleware* se refiere al software que facilita la comunicación entre dos aplicaciones o bases de datos. A continuación se tratarán preguntas relacionadas al *middleware* empleado en el departamento para conectar los sistemas o las bases de datos:

10.- ¿Conoce los siguientes tipos de *middleware*:

         Remote Procedure Call (RPC)

         Middleware Orientado a Mensajes (MOM)

\_\_\_\_\_ Objetos distribuidos

  2   Middleware Orientado a bases de datos

\_\_\_\_\_ Middleware Orientado a transacciones

  2   Application Servers

\_\_\_\_\_ Otros (especifique):

*11.- De los siguientes tipos de middleware, ¿cuáles emplean en sus sistemas?*

\_\_\_\_\_ Remote Procedure Call (RPC)

\_\_\_\_\_ Middleware Orientado a Mensajes (MOM)

\_\_\_\_\_ Objetos distribuidos

\_\_\_\_\_ Middleware Orientado a bases de datos

\_\_\_\_\_ Middleware Orientado a transacciones

  2   Application Servers

\_\_\_\_\_ Otros (especifique):

### TABLA DE APOYO PARA LA SELECCIÓN DE MIDDLEWARE

El presente anexo se propone como una lista de aspectos a medir para la selección de un *middleware*. Los valores pueden ir desde el 1 que significa un nivel de importancia baja hasta 5 que indica un aspecto crítico en la selección de la tecnología.

Aspectos a Medir	Middleware 1	Middleware 2	.....	Middleware n	Total por aspecto
Criterios Primarios					
❖ Disponibilidad (24x7)					
❖ Desempeño					
❖ Administración					
❖ Escalabilidad					
Conexión a varias aplicaciones (procesos <i>batch</i> , tiempo real, web)					
Soporte de diferentes protocolos de red (TCP/IP, IPX/SPX, AppleTalk)					
Soporte de diferentes sistemas operativos (UNIX, WINDOWS, LINUX, AIX)					
Soporte de diferentes arquitecturas (cliente-servidor, cliente-servidor de aplicación- servidor de BD)					
Soporte de mover altos volúmenes de datos entre bases de datos					
Tiempo de implementación					
Complejidad de uso e instalación					
Costo (adquisición, mantenimiento, etc.)					
<b>Total por <i>middleware</i></b>					

## ANEXO IV GLOSARIO

---

**EAI (*Enterprise Application Integration*):** se refiere a automatizar la forma de compartir los datos y procesos organizacionales tomando en cuenta no realizar demasiados cambios a las aplicaciones o a las estructuras de datos actuales

**Metadata:** se refiere a los datos que definen la manera como se encuentra relacionada la información en una base de datos, su formato y su estructura.

**ERP (*Enterprise Resource Planning*):** es un sistema de administración organizacional que integra todos los aspectos de la empresa incluyendo planeación, manufactura, ventas, mercadotecnia, finanzas, entre otros.

**Legacy System (sistema heredado):** cualquier sistema central de cómputo que es indispensable para mantener a una empresa operando

**B2B (*business to business*):** tipo de venta de productos, servicios o información a través de Internet donde las empresas le venden directamente a otras empresas sin necesidad de intermediarios.

**B2C (*business to consumer*):** tipo de venta de productos, servicios o información a través de Internet donde las empresas le venden a los consumidores tal y como sucede en la venta tradicional.

**C2B (*consumer to business*):** tipo de venta de productos, servicios o información a través de Internet donde se reúnen una cantidad de consumidores para solicitar productos o servicios a las empresas.



**C2C (consumer to consumer):** tipo de venta de productos, servicios o información a través de Internet donde los consumidores venden sus artículos a otros consumidores. Este método se asemeja al tradicional modelo de las subastas presenciales.

**Front-office:** sistemas empleados por los usuarios finales para el manejo de la información de una empresa.

**Back-office:** sistemas que son usados para soportar las operaciones llevadas a cabo por los usuarios de los sistemas de front office.

**ODBC (Open Database Connectivity):** Conectividad de Base de Datos Abierta, se refiere a un estándar utilizado para acceder a diferentes bases de datos.

**Proceso batch:** se refiere a los trabajos que son ejecutados en los equipos de cómputo sin necesidad de la interacción del usuario y son ejecutados en intervalos de tiempo definidos.

**Mainframes:** computadora de grandes dimensiones que tiene como objetivo servir a una alta cantidad de usuarios de manera simultánea.

**API (Application Programming Interface):** consiste en un método definido por una aplicación o un sistema operativo a través del cual una aplicación se puede comunicar con otra aplicación o sistema operativo

**Middleware:** se refiere a toda herramienta de software que permite comunicar aplicaciones entre sí empleando la funcionalidad existente de las mismas.

**RPC (Remote Procedure Call):** es un protocolo que una aplicación emplea para solicitar un servicio a un programa localizado remotamente en otro equipo de cómputo.

**MOM (*Message Oriented Middleware*):** tipo de *middleware* que emplea mensajes como un método de integración, además de ofrecer la habilidad de integrar diversas aplicaciones a través del uso de mensajes

***Message Queueing*:** Método a través del cual los procesos o aplicaciones pueden intercambiar o transferir datos empleando una cola de mensajes que permita el envío y recepción de mensajes.

## REFERENCIAS BIBLIOGRÁFICAS

---

About.com. (2001). *What is ERP?*. [WWW] URL: <http://about.com> (Accesado Agosto 2001)

ACM. (1998). *Association for Computing Machinery (ACM)*. [WWW] The ACM Computing URL: Classification System [1998 Version] URL: <http://www.acm.com/class/overview.html> (Accesado Octubre 2001)

Adexa. (2000). *What's your e-Business Strategy for 2000 and Beyond?*. [WWW] Adexa Inc. URL: <http://adexa.com> (Accesado Febrero 2001)

Bickerton, Pauline et al. (2000). *Ciberestrategia*. Prentice Hall

Cashin, Jerry. (1999a). *E-commerce Success – Building a Global Business Architecture*. Computer Technology Research Corp. USA

Cashin, Jerry. (1999b). *Web Commerce – Developing and Implementing Effective Business Solutions*. Computer Technology Research Corp. USA

Craig, Robert. (2000). *Enterprise Application Integration*. [WWW] Noviembre 22, 2000. Vol 5 No. 19. URL: <http://proquest.umi.com/pqdweb?Did=000000064931096&Font=3&Deli=1&Mtd=1&Idx=11&Sid=1&RQT=309> (Accesado Junio 2001)

Connolly, James. (1999) *ERP: Corporate Cleanup*. [WWW] Computerworld. Vol. 33 No. 9 Páginas: 74-78 URL: <http://proquest.umi.com/pqdweb?TS=1005612993&Did=000000039370210&Mtd=1&Fmt=3&RQT=431> (Accesado Noviembre 2001)

Gilpin, Mike. (1999a). *How to Select an Enterprise Application Integration Solution*. [WWW] Giga Information Group URL: [http://eai.ebizq.net/shared/white\\_papers.jsp?ID=giga\\_select.pdf](http://eai.ebizq.net/shared/white_papers.jsp?ID=giga_select.pdf) (Accesado Noviembre 2000)

Gilpin, Mike. (1999b). *Internet Application Integration: A New Market Emerges to Support Cross-Enterprise E-Business*. [WWW] Giga Information Group URL: [http://eai.ebizq.net/shared/white\\_papers.jsp?ID=giga\\_iai.pdf](http://eai.ebizq.net/shared/white_papers.jsp?ID=giga_iai.pdf) (Accesado Noviembre 2000)

Gonsalves, Antone. (2001). *Value of EAI grows as integration needs expand*. [WWW] InformationWeek, Mayo 28, 2001. URL: <http://proquest.umi.com/pqdweb?Did=000000073423143&Fmt=4&Deli=1&Mtd=1&Idx=9&Sid=7&RQT=309> (Accesado Agosto 2001)

Haas, L.M., Rice J.E., Schwarz, P.M., Swope, W.C. et al. (2001) *DiscoveryLink: A system for integrated access to life sciences data sources* IBM Systems Journal Volumen 40, Issue 2 Páginas 489-511 Armonk

Haase, Kim, 2001 *Java Message Service Tutorial* Sun Microsystems. [WWW] Agosto 28, 2001 URL: [http://java.sun.com/products/jms/tutorial/1\\_3-fcs/doc/jms\\_tutorialTOC.html](http://java.sun.com/products/jms/tutorial/1_3-fcs/doc/jms_tutorialTOC.html) (Accesado Octubre 2001)

Hamel, Gary & Sampler, Jeff (1998). The E-Corporation. Fortune Magazine 138 (Diciembre 7, 1998)

Hummingbird. (2001) *Metadata Strategy: The Road Ahead* [WWW] Hummingbird Communications LTD URL: <http://www.dw-institute.com/vendors/hummingbird/metadata.pdf> Page 6 (Accesado June 2001)

IBM. (2000) *Whirlpool's B2B trading portal cuts per-order costs significantly*. [WWW] IBM corporation URL: <http://www2.software.ibm.com/casestudies/swcs.nsf/customername/684CEC59CD9E0CAE00256938001CBDBB> (Accesado Noviembre 2001)

International. (1997) *Middleware White Paper* [WWW] International Systems Group Inc. URL : <http://www.openvms.compaq.com/openvms/whitepapers/middleware/isgmiddleware.pdf> (Accesado Noviembre 2001)

Jaben-Eilon, Jan. (2000). *Understanding CORBA*. Banl Technology News. Volumen 13 Issue 8 Paginas 52-56 New York

Kalafsky, Mark J. (2000). *Legacy reuse strategies for EAI*. [WWW] Unisys World. Vol. 21 No. 10. Austin Octubre 2000 URL: <http://proquest.umi.com/pqdweb?Did=000000062658513&Fmt=4&Deli=1&Mtd=1&Idx=1&Sid=2&RQT=309> (Accesado Agosto 2001)

Kalakota, Ravi & Whinston, Andrew B. (1997). Electronic Commerce – A Manager's guide. Addison-Wesley

Linthicum, David S. (1999). *Moving forward with EAI*. Component Strategies. New York. Addison-Wesley. Vol. 2 No. 1, Paginas 44-50

Linthicum, David. (2000a). *How to Free your Information*. [WWW] SAGA Software Group. URL: [http://eai.ebizq.net/enterprise\\_integration/linthicum\\_4.html](http://eai.ebizq.net/enterprise_integration/linthicum_4.html) (Accesado Agosto 2000)

Linthicum, David. (2000b). La integración de aplicaciones empresariales toma fuerza en el mercado. Revista RED. Abril de 2000.

Linthicum, David S. (2000). Enterprise Application Integration. Addison-Wesley.

Mattila, Sakari. (2001) *Data dictionary –what should be in it?* [WWW] University of Canberra URL: <http://www.canberra.edu.au/~sam/whp/datadict.html> (Accesado Noviembre 2001)

- Mckie, Stewart. (2000). *The smoking gun*. Intelligent Enterprise, Noviembre 10, 2000. Vol 3 No. 17 URL: [http:// http://proquest.umi.com/pqdweb?TS=999219311&RQT=309&CC=1&Dtp=1\\$Did=000000063104137&Mtd=1&Fmt=3](http://http://proquest.umi.com/pqdweb?TS=999219311&RQT=309&CC=1&Dtp=1$Did=000000063104137&Mtd=1&Fmt=3) (Accesado Agosto 2001)
- McIntyre, Hal. (2000). *Middleware White Paper*. [WWW] Agosto 1, 2000. Securities Operations Forum. <http://www.soforum.com/library/middleware.shtml> (Accesado November 2001)
- Middleware.org. (2000). *Middleware Resource Center* [WWW] URL [http:// http://www.middleware.org/](http://http://www.middleware.org/) (Accesado November 2001)
- Moragenthal, JP. (2000) *Metadata: A Key to EAI*. [WWW] EAI Journal. URL: <http://www.eaijournal.com/ArticlePrint.asp?ArticleID=225> (Accesado Noviembre 2000)
- Mordacq, Michele. (2001) *Reasons and Options for Migrating Legacy Devices to a TCP/IP WAN*. [WWW] SSI Embedded Systems Programming. URL: <http://www.ssinc.com/lanpanther/whitpapr.html> (Accesado Noviembre 2001)
- NC.Focus. (1999). Lowering the total cost of EAI. NC.Focus Corporation.
- Network Computing. (1995). *A Typology of Middleware*. [WWW] Network Computing. URL: <http://www.nwc.com/netdesign/cdmwtypo.htm#dbms> (Accesado Noviembre 2001)
- Peacock, Robert. (2000). *Enabling Technologies for E-Business*. IEEE Computer Society. URL: <http://www.ieee.org> (Accesado Noviembre 2001)
- Poujade, Rod. (1997) *Distributed Computer Protocols* [WWW] McGill University, Montreal, Canada September 1997, Page 3-4 URL <http://www.is.mcgill.ca/e-Business/techrep2.pdf> (Accesado Noviembre 2001)
- Ranadivé, Vivek & McNealy, Scott. (1999). The Power of Now: How Winning Companies Sense and Respond to Change Using Real-Time Technology. Mc-Graw Hill.
- Ren, Frances. (2000). *The Marketplace of Enterprise Application Integration (EAI)*. [WWW] URL: <http://www.public.asu.edu/~mbfr2047/eai.html> (Accesado Febrero 2001)
- Rational. (2000). *Managing Content and Code for e-Business*. [WWW] Rational Software Corporation URL: <http://www.rational.com> (Accesado Noviembre 2000)
- Robinson, Brian. (1999). *The ORB slowly rises*. [WWW] Federal Computer Week URL: [http://www.fcw.com/fcw/articles/1999/FCW\\_050399\\_438.asp](http://www.fcw.com/fcw/articles/1999/FCW_050399_438.asp) (Accesado Noviembre 2001)
- Rossmann, Brian & Lach, Stephen. (2000). Legacy Systems, to extend or not to extend, that is the question. Mainframe Computing, Dec 1, 2000

Rowe, Terry. (2000). *Making the Business Case for Middleware* [WWW] EAI Journal. URL: <http://www.eaijournal.com/ArticlePrint.asp?ArticleID=295> (Accesado Noviembre, 2000)

Ruh, William et al. (2001). *Enterprise Application Integration: A Wiley Tech Brief*. Wiley Computer Publishing.

Sadoski, Darleen. (1997) *Message Oriented Middleware* [WWW] Software Technology review. Carnegie Mellon Software Engineering Institute URL <http://www.sei.cmu.edu/str/descriptions/momt.html> (Accesado Noviembre 2001)

Sánchez, Eric & Fenner, Joe. (2001). *EAI users go with the flow*. [WWW] InformationWeek. Marzo 26, 2001. Página: 75-80 URL: <http://proquest.umi.com/pqdweb?Did=000000070116057&Fmt=4&Deli=1&Mtd=1&Idx=7&Sid=3&RQT=309> (Accesado Agosto 2001)

SCT. (2000). *Systems and Computer Technology Corporation: Banner, Eudcation, Manufacturing and Software*. [WWW]. SCT. URL: <http://www.sct.com/index.htm> (Accesado Noviembre 2001)

Schussel, George. (1996) *Client/Server Past, rpesent and Future* [WWW] URL: <http://news.dci.com/geos/dbsejava.htm> (Accesado Noviembre 2001)

SearchDatabase.com. (2001) *The Database Specific Serach Engine*. [WWW] URL: <http://searchdatabase.techtarget.com/> (Accesado Junio 2001)

Seybold, Patricia. (1998). *Customers.com*. Patricia Seybold Group. USA

Shoffner, Michael. (2000). *Write your own MOM!*. [WWW] JavaWorld. URL: [http://www.javaworld.com/javaworld/jw-05-1998/jw-05-step\\_p.html](http://www.javaworld.com/javaworld/jw-05-1998/jw-05-step_p.html) (Accesado Noviembre 2001)

Talarian. (2000a). *Mastering middleware* [WWW] Talarian Corporation,. URL: <http://www.talarian.com> (Accesado Octubre 2001)

Talarian. (2000b). *Case Studies* Talarian Corportaion [WWW] URL: [http://www.talarian.com/industry/mmw/mmw\\_casestudies.htm](http://www.talarian.com/industry/mmw/mmw_casestudies.htm) (Accesado Noviembre 2001)

Taviz. (2000a). *The Hidden Costs of Enterprise Application Integration*. [WWW] Taviz Technology. Agosto 29, 2000. URL: <http://www.taviz.com/solutions/papers.html> (Accesado Junio 2001)

Taviz. (2000b). *Data Interfacing for Enterprise Applications*. [WWW] Taviz Technology, Agosto 31, 2000. (Accesado Junio 2001)

Tibco. (2000). *TIBCO Customer Profiles: CBS SportsLine* [WWW] TIBCO Software URL: <http://www.tibco.com/customers/cbs.html> (Accesado Noviembre 2001)

Ulrich, William. (2000). *The essential information integration strategy*. [WWW] Tactical Strategy Group, Inc. URL: [http://eai.ebizq.net/str/ulrich\\_1.html](http://eai.ebizq.net/str/ulrich_1.html) (Accesado Noviembre 2000)

Vondrak, Cory. (1997) *Message Oriented Middleware* [WWW] Software Technology review. Carnegie Mellon Software Engineering Institute URL <http://www.sei.cmu.edu/str/descriptions/momt.html> (Accesado Noviembre 2001)

Whatis.com. (2001). [WWW] The IT-specific encyclopaedia. URL: <http://whatis.com> (Accesado Noviembre 2001)

White house. (1997). *A framework for Global E-Commerce*. The White House

Wilkes, Lawrence. (1999). *Application Integration*. [WWW] Butler Group Limited. Mayo 1999. URL: <http://www.butlergroup.com> (Accesado Marzo 2001)

Yee, Andre. (2000). *Demystifying Business Process Integration*. [WWW] SAGA software  
URL: [http://eai.ebizq.net/enterprise\\_integration/yee\\_4.html](http://eai.ebizq.net/enterprise_integration/yee_4.html) (Accesado Agosto, 2000)

Zhang, Chundi. *Enterprise Resource Planning (ERP)*. [WWW]  
URL : <http://theweb.badm.sc.edu/701sstu/zhang/assign11.htm> (Accesado, Marzo 2001)

Centro de Información-Biblioteca



3000200605511