

**UNIVERSIDAD NACIONAL DE INGENIERÍA**  
**FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA**



**INTERCONEXIÓN MULTIPLATAFORMA MEDIANTE UN  
“MIDDLEWARE” DE COMUNICACIONES**

**INFORME DE SUFICIENCIA**

PARA OPTAR EL TÍTULO PROFESIONAL DE:

**INGENIERO ELECTRÓNICO**

**PRESENTADO POR:**

**MÁXIMO ALBERTO DÍAZ LÓPEZ**

**PROMOCIÓN  
1977-II  
LIMA-PERÚ  
2006**

**INTERCONEXIÓN MULTIPLATAFORMA MEDIANTE UN “MIDDLEWARE”  
DE COMUNICACIONES**

*Dedico este trabajo a:  
Mis padres, ejemplo de laboriosidad,  
Y mi pareja, por su apoyo y aliento.*

## SUMARIO

El presente trabajo busca hacer de dominio público que cualquier empresa interesada en interconectar sus procesos de negocios con los de sus clientes y proveedores, sólo necesita dedicarse al desarrollo de sus aplicaciones de negocios, si delega la complejidad del manejo de las comunicaciones a un “middleware” de comunicaciones, el que servirá de adaptador entre la plataforma de comunicación de la empresa y las de sus clientes y proveedores.

En el capítulo I se da una introducción a la comunicación cliente/servidor, que posibilita el intercambio de información entre sistemas heterogéneos.

El capítulo II describe cómo surge la necesidad de negocio de una compañía de seguros de interconectarse en línea con sus clínicas afiliadas (proveedores).

El capítulo III detalla los pasos seguidos para la selección del proveedor de un “middleware” de comunicaciones.

El capítulo IV describe el “middleware” seleccionado.

El capítulo V muestra una consulta de asegurado con sus coberturas.

El capítulo VI es un esbozo de los servicios Web, lo que será el estado futuro de la tecnología en interoperabilidad.

## ÍNDICE

<b>PRÓLOGO</b>	1
<b>CAPÍTULO I</b>	
<b>INTRODUCCIÓN A LA COMUNICACIÓN CLIENTE/SERVIDOR</b>	2
1.1. El modelo cliente/servidor	2
1.1.1. Algunas definiciones	2
1.1.2. Estilos de comunicación cliente/servidor	2
1.1.3. Distribución de funcionalidades	3
1.2. Clientes y servidores	5
1.2.1. Roles y funciones del cliente	5
1.2.2. Funciones del servidor	7
1.2.3. Cambio en el paradigma de procesamiento	8
1.2.4. Hoy en día el computador es la red	8
1.2.5. Algunos conceptos asociados con las redes	8
1.2.6. Sistemas cliente/servidor orientados al objeto	8
1.2.7. Quién es cliente y quién es servidor	9
1.3. API's para el modelo cliente/servidor	9
1.3.1. Integración y acceso	9
1.3.2. Clasificación de las API's	9
1.4. "Middleware" en cliente/servidor	11
1.4.1. Productos de acceso remoto a bases de datos	11
1.4.2. Productos Gateway	11
1.4.3. Productos Remote Procedure Call – RPC	11
1.4.4. Productos Object Request Broker – ORB	12
1.4.5. Productos de encolamiento de mensajes	12
1.4.6. Productos de comunicación interprocesos	12
1.4.7. Productos de acceso remoto a archivos	12
1.4.8. Productos de servicio de ruteo de mensajes	12
1.4.9. Productos de servicios de directorio	13

1.4.10. Productos de servicios conversacionales	13
1.4.11. Productos de servicios de tiempos	13
1.4.12. Productos de servicios de terminales	13
1.4.13. Productos de administración de redes	13
1.4.14. Productos de seguridad de redes	13
1.5. Implantación de la comunicación cliente/servidor	14
1.5.1. La gerencia estratégica	14
1.5.2. La perspectiva gerencial	16
1.5.3. La arquitectura de la comunicación cliente/servidor	18
1.5.4. Reposicionamiento de la empresa para la comunicación Cliente/servidor	19
<b>CAPÍTULO II</b>	
<b>PROYECTO CONSULTA EN LÍNEA DE ASEGURADOS</b>	23
2.1. Antecedentes del proyecto	23
2.2. Estado tecnológico de mercado asegurador y clínicas	24
2.3. Toma de decisión sobre ejecución del proyecto	25
2.4. Formación de grupo de trabajo del proyecto	25
2.5. Inicio de labor de grupo de trabajo	25
<b>CAPÍTULO III</b>	
<b>SELECCIÓN DEL PROVEEDOR DE “MIDDLEWARE” DE COMUNICACIONES</b>	27
3.1. Búsqueda del proveedor	27
3.2. Criterios utilizados en selección de proveedor	27
3.2.1. Cartera de clientes	27
3.2.2. Proyectos desarrollados	27
3.2.3. Propuesta técnica	27
3.2.4. Propuesta económica	27
3.3. Propuestas	28
3.3.1. Omnisciencia Consultores	28
3.3.2. Novatronic	33
3.4. Selección de proveedor	34
<b>CAPÍTULO IV</b>	
<b>DESCRIPCIÓN DEL “MIDDLEWARE” SELECCIONADO: TEN</b>	37

4.1. Conceptos generales	37
4.2. Arquitectura del sistema	38
4.3. Queue Manager	38
4.4. Gestor de eventos	39
4.5. Transporte	39
4.6. Administración	39
<b>CAPÍTULO V</b>	
<b>EJEMPLO DE UNA CONSULTA</b>	41
5.1. Tipos de transacción	41
5.2. Esquema funcional del TEN	41
5.3. Procesamiento de una transacción	43
5.4. Ejemplo de transacciones	44
<b>CAPÍTULO VI</b>	
<b>SERVICIOS WEB</b>	46
6.1. Antecedentes	46
6.2. Definición	46
<b>CONCLUSIONES</b>	49
<b>ANEXO A: DEFINICIONES DE CLIENTE/SERVIDOR</b>	51
<b>ANEXO B: TRANSACCIONES DEL TEN CLIENTE</b>	53
<b>ANEXO C: PARÁMETROS DEL TEN CLIENTE</b>	59
<b>ANEXO D: GUÍA DE PROGRAMACIÓN DEL TEN CLIENTE</b>	62
<b>ANEXO E: ACRÓNIMOS</b>	71
<b>BIBLIOGRAFÍA</b>	74

## PRÓLOGO

El presente informe tiene como propósito hacer de dominio público que cualquier empresa interesada en interconectar sus procesos de negocios con los de sus clientes y proveedores, sólo necesita dedicarse al desarrollo de sus aplicaciones de negocios, si delega la complejidad del manejo de las comunicaciones a un “middleware”, el que servirá de adaptador entre la plataforma de comunicación de la empresa y las de sus clientes y proveedores. Es decir si subcontrata el desarrollo de esta función a un proveedor especializado en la integración de sistemas multiplataforma.

El informe inicia con una introducción a la comunicación cliente/servidor, que proporciona las bases para la interconexión entre sistemas heterogéneos, donde un componente pide servicios de otro.

Se presenta el caso particular de una compañía de seguros con la necesidad de interconectarse con sus clínicas afiliadas en Lima, provincias y el extranjero, a fin de proporcionarles datos de los asegurados y sus correspondientes coberturas.

Se dan a conocer en primer lugar los criterios de selección del proveedor del “middleware”, luego la descripción del “middleware” escogido y finalmente se muestran las consultas de asegurados, que es lo que quería lograr.

Ahora bien muchos de los “middleware”, son software hechos con estándares propios, que los convierten en una tecnología propietaria, es decir sólo se podrían interconectar empresas que adquieran el mismo “middleware”. Por ello se presenta lo que será el futuro estado de la tecnología en interoperabilidad: los Servicios Web (Web Services), que permiten el intercambio de datos entre aplicaciones y están basados en estándares abiertos de Internet : XML como formato de datos , los protocolos de Internet para transporte y alguno de los siguientes estándares

- SOAP
- WSDL
- UDDI



# CAPÍTULO I

## INTRODUCCIÓN A LA COMUNICACIÓN CLIENTE/SERVIDOR

### 1.1. El modelo cliente/servidor.

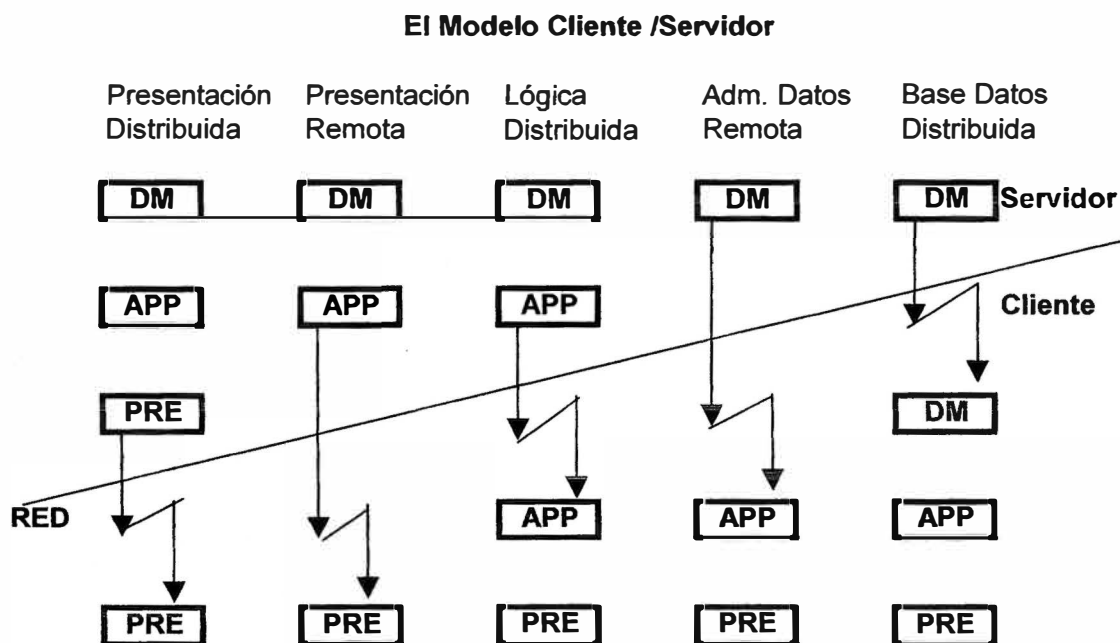
#### 1.1.1. Algunas definiciones.

Ver anexo A Definiciones de Cliente/Servidor.

#### 1.1.2. Estilos de comunicación cliente/servidor.

- Presentación distribuida.
- Presentación remota.
- Lógica distribuida.
- Administración de datos remota.
- Base de datos distribuida.

Estos estilos se muestran en figura 1.1:



**Figura 1.1 Estilos de comunicación cliente/servidor**

### **1.1.3. Distribución de funcionalidades.**

#### **a) Modelo de presentación distribuida.**

- Se distribuye la interfaz entre el cliente y la plataforma servidora.
- La aplicación y los datos están ambos en el servidor.
- Similar a la arquitectura tradicional de un host (servidor) y terminales.
- El PC se aprovecha sólo para mejorar la interfaz gráfica para el usuario.

#### **Ventajas:**

- Revitaliza los sistemas antiguos.
- Bajo costo de desarrollo.
- No hay cambios en los sistemas existentes.

#### **Desventajas:**

- El sistema sigue en el host, no se ha bajado.
- No se aprovecha la GUI y/o la LAN.
- La interfaz del usuario se mantiene en muchas plataformas.

#### **b) Modelo de presentación remota.**

- La interfaz para el usuario está completamente en el cliente.
- La aplicación y los datos están en el servidor. Ej: X/Windows cliente contra UNIX en el servidor.

#### **Ventajas:**

- La interfaz del usuario aprovecha bien la GUI y la LAN.
- La aplicación aprovecha el host.
- Adecuado para algunos tipos de aplicaciones de apoyo a la toma de decisiones.

#### **Desventajas:**

- Las aplicaciones pueden ser complejas de desarrollar.
- Los programas de la aplicación siguen en el host.
- El alto volumen de tráfico en la red puede hacer difícil la operación de aplicaciones muy pesadas.

#### **c) Modelo de lógica distribuida.**

- La interfaz está en el cliente.
- La base de datos está en el servidor.
- La lógica de la aplicación está distribuida entre el cliente y el servidor.

**Ventajas:**

- Arquitectura más corriente que puede manejar todo tipo de aplicaciones.
- Los programas del sistema pueden distribuirse al nodo más apropiado.
- Puede utilizarse con sistemas existentes.

**Desventajas:**

- Es difícil diseñar como distribuir la aplicación.
- Difícil prueba y mantención si los programas del cliente servidor están en distintos lenguajes.
- No son manejados hoy por los GUI 4GL.

**d) Modelo de administración de datos remota.**

- En el cliente residen tanto la interfaz como los procesos de la aplicación.
- La base de datos está en el servidor.
- Es lo que comúnmente imaginamos como aplicación cliente/servidor.

**Ventajas:**

- Configuración típica de las herramientas GUI 4GL.
- Muy adecuada para las aplicaciones de apoyo a las decisiones del usuario final.
- Fácil de desarrollar ya que los programas de aplicación no están distribuidos.
- Se han descargado los programas del host.

**Desventajas:**

- No maneja aplicaciones pesadas, eficientemente.
- La totalidad de los datos viaja por la red, ya que no hay procesamiento que realice el host.

**e) Modelo de base de datos distribuida.**

- La interfaz, los procesos de la aplicación y parte de los datos de la base de datos están en el cliente.
- El resto de los datos está en el servidor.

**Ventajas:**

- Configuración soportada por herramientas GUI 4GL.
- Adecuada para las aplicaciones de apoyo al usuario final.
- Apoya acceso a datos almacenados en ambientes heterogéneos.
- Ubicación de los datos es transparente para la aplicación.

**Desventajas:**

- No maneja aplicaciones grandes, eficientemente.
- El acceso a la base de datos distribuida es dependiente del proveedor del software administrador de bases de datos.

**1.2. Clientes y servidores.**

Un **cliente** es un computador que requiere información ó un servicio de otro computador.

Clientes son en general PC's que son utilizados por empleados para desempeñar algún trabajo en la organización.

Un **servidor** es un computador que provee información ó un servicio a un cliente.

Servidores pueden ser otros PC's, o máquinas dedicadas, diseñadas para interactuar con los PC's clientes, permitiendo a los clientes interfacear con hardware/software especializado, tales como bases de datos, módems, impresoras, o mainframes.

**1.2.1. Roles y funciones del cliente.**

- Una de las principales es la de presentación.
- En algunos casos tienen algo de lógica de procesamiento.
- La interacción con el usuario es a través de la lógica de la presentación.
- La tradicional función de presentación consistía en el despliegue secuencial de caracteres recibidos desde la aplicación, en un tipo de letra (font) fijo en la pantalla.
- La evolución de las capacidades gráficas permite a los diseñadores de software crear interfaces de aplicaciones intuitivas, que utilizan todas las capacidades gráficas a través del despliegue de la imaginación, aprovechando todas las capacidades de audio y video.
- El usuario final sólo apunta y da un click.

**a) Administración de la presentación.****Por qué adoptar un estándar GUI.**

- Afecta a desarrolladores y usuario final.
- GUI incompatibles : Macintosh, MS Windows, OS/2 Presentation Manager.

**Requerimientos generales para una GUI estándar.**

- Portabilidad.

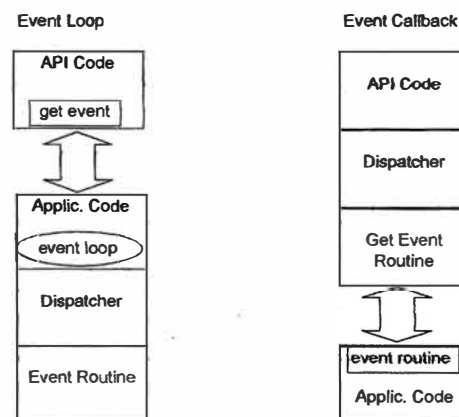
- Compromiso con estándares (X/Window System sirve como un modelo a NIST, ANSI, IEEE estándares y especificaciones X/Open).
- Herramientas de desarrollo.
- Flexibilidad (acomodarse hacia el futuro).
- Internacionalización.
- Independencia de plataforma.

#### b) Características de las GUI.

- Presentan información en áreas rectangulares (ventanas).
- Las ventanas contienen objetos que pueden ser seleccionados con un click del mouse.
- GUI's avanzadas permiten eliminar completamente la necesidad de dar comandos escritos, permitiendo al usuario seleccionar utilizando el mouse o las teclas de función.
- Cambia el concepto de la programación tradicional con la interacción constante del usuario con la GUI, programación a través de eventos.
- La lógica de eventos principalmente es a través de: mouse, teclado, menú, eventos de la ventana, eventos de activación y desactivación, eventos de inicialización y término.

#### c) Distribución de los eventos.

Existen varios modelos de procesamiento de distribución de eventos: event loop y event callback, mostrados en figura 1.2.



**Figura 1.2 Modelos de procesamiento de distribución de eventos**

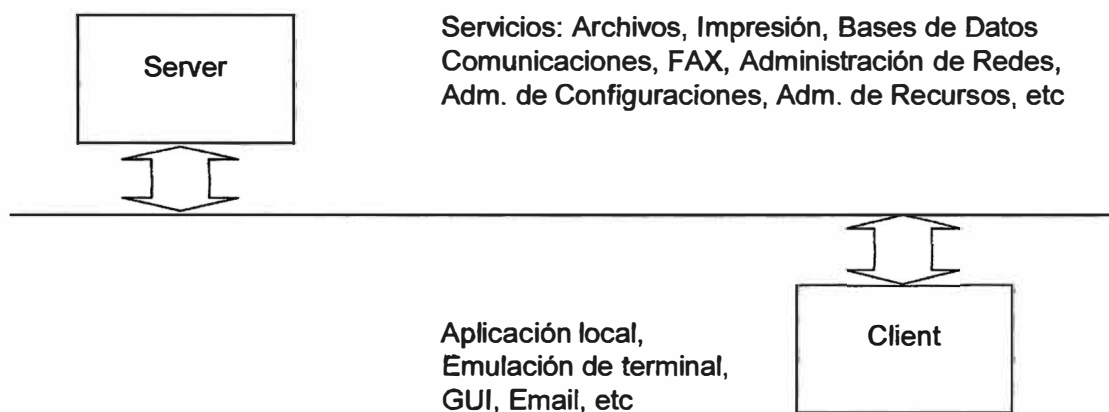
#### d) Implementaciones GUI.

- OSF/Motif
- Open Look
- Macintosh
- MS Windows
- Presentation Manager
- NeXT
- Modo carácter y bloques

#### 1.2.2. Funciones del servidor.

- La especialización del servidor está reflejada en la funcionalidad requerida y el diseño de servidores de bases de datos.
- Capacidad de almacenamiento de grandes cantidades de datos.
- Extensión de sus funciones hacia comunicaciones, fax, administración de librerías, correo electrónico.
- Las funciones de un servidor están en gran medida definidas por los requerimientos de los clientes/usuarios.
- Así, si un servidor está inhabilitado para contestar un requerimiento de un cliente, entonces no puede participar en la interacción cooperativa cliente/servidor.

Se muestra esquema de funciones del servidor y cliente en figura 1.3.



**Figura 1.3 Esquema de funciones del servidor y cliente**

### **Requerimientos al servidor.**

- Soporte multiusuario.
- Escalabilidad
- Performance.
- Almacenamiento.
- Multimedia
- Funcionamiento en redes.

### **1.2.3. Cambio en el paradigma de procesamiento.**

- Por lotes (Batch).
- Interactivo o en Línea.
- En computador personal aislado (stand alone).
- En computador personal en red.

### **1.2.4. Hoy en día el computador es la red.**

- Potencia computacional y disponibilidad de datos corresponden a todos los servidores que están visibles a un cliente.
- No importan las capacidades del computador personal o estación de trabajo, sino las tareas que va a desempeñar como cliente.
- Tendencia a globalización de redes: ej. Internet.

### **1.2.5. Algunos conceptos asociados con las redes.**

- **“Internetworking”**. Es la búsqueda de conectividad entre redes a través de hardware, software, elementos de comunicación utilizando módems, routers, bridges y gateways, así como servicios de transmisión de datos dedicados o conmutados.
- **Interoperabilidad**. Es la capacidad de las aplicaciones de una red (de computadores) para poder comunicarse y transmitirse datos entre procesos unos a otros.
- **Redes**. Simples (LAN) , complejas : múltiples servidores, WAN : redes de redes.

### **1.2.6. Sistemas cliente/servidor orientados al objeto.**

Aplicaciones que obtienen su funcionalidad a partir de objetos que residen en bibliotecas de clases de objetos.

### 1.2.7. Quién es cliente y quién es servidor.

Un servidor puede además ser cliente, por ejemplo:

- Si consulta a los clientes que le están pidiendo datos sobre el tamaño de la pantalla para formatearle los datos.
- Si un servidor de datos departamental, en otra instancia requiere extraer datos de la base de datos corporativa.
- Un servidor de una LAN, que es gateway de ella en una WAN, puede ser cliente de otros equipos en la WAN.
- Una máquina se comporta como cliente o como servidor en instantes distintos.

## 1.3. API's para el modelo cliente/servidor.

### 1.3.1. Integración y acceso.

- Una aplicación necesita apoyo del sistema operativo, de otras aplicaciones y de bases de datos. Para ello necesita saber cuándo solicitudes de servicios o información serán enviadas a una aplicación externa.
- **Una Application Programming Interface (API)** es una interfaz programable de una aplicación en la cual se escriben los llamados que serán transmitidos a funciones de aplicaciones externas.
- Algunas aplicaciones tienen API's propios, otros hacen alianzas para que su software converse con otros software.
- Algunos sistemas operativos definen algunas API's para que las aplicaciones que corren bajo éste puedan utilizarlas para comunicarse fácilmente.
- La tendencia de los fabricantes de software es ir hacia API's abiertos. Un API abierto es un API documentado, estándar, que cualquiera puede usar si quiere comunicarse con cualquier programa que use ese mismo API.

### 1.3.2. Clasificación de las API's.

#### a) API's para GUI.

- Para los modelos de base de datos remota y de aplicación distribuida, un conjunto de API's engancha la interfaz del usuario y la aplicación.
- El API para el GUI hace posible al usuario ejecutar la aplicación desde dentro de un particular ambiente gráfico o de administración de ventanas.



**b) API's para acceso a datos.**

- Facilitan la interacción entre aplicaciones y bases de datos remotas (modelo más común).
- En el pasado las aplicaciones tendían a acceder a los datos en bases de datos remotas mediante un enganche específico de programación. Mientras que en el enfoque cliente/servidor el enganche entre aplicación y la base de datos debe ser transparente.
- Para facilitar este enganche automático y transparente aparece el software “middleware”.
- El lenguaje SQL permite el acceso remoto a bases de datos y es el API más conocido.
- Para adquirir datos, una aplicación traduce su consulta a comandos SQL, los que transmite (vía “middleware”) al SABD (Sistema Administrador de Bases de Datos ).
- El SABD traduce los comandos SQL que recibe en el lenguaje de manipulación interno de la BD, hace lo que sea necesario para armar los datos pedidos, y los envía de vuelta a la aplicación.

**c) RPC's (Remote Procedure Call)**

- En el modelo de aplicación distribuida se asume que la aplicación está dividida, y que una parte (o parte de una) debe interactuar con otra parte.
- Hay varias maneras de pasar llamados a funciones entre aplicaciones pero los RPC's, es el enfoque más comúnmente utilizado.
- El uso más simple es invocar código procedural que está almacenado en una BD, o en otra máquina (ej. en el “mainframe”).
- Un RPC es un tipo especial de API que permite a un programador especificar una frontera (interfaz) y una llamado a función (function call).
- Toda vez que un programa hace contacto con la interfaz, un llamado a función es iniciado, y subsecuentemente datos son devueltos a esa interfase.
- En su mínima expresión, un RPC es una especificación de interfaz, un lenguaje y protocolos en los cuales es posible escribir llamados a funciones.

**d) ORB's (Object Request Brokers)**

- El Object Management Group (OMG) ha desarrollado un estándar para ORB's llamado Common Object Request Broker Architecture (CORBA).

- CORBA está compuesta de dos diferentes enfoques para un ORB: estático y dinámico.
- La versión estática de CORBA permite al programador tomar ventaja del OOL (Object Oriented Language) cuando está creando interfaces para la red. El desarrollador crea clases para representar los distintos tipos de interfaces y llamados a función, y luego simplemente especializa las clases para proveer el código para interfaces y funciones específicas, luego de lo cual la ORB estática debe compilarse.
- La versión dinámica de CORBA nunca se compila completamente. Cuando se le envía un mensaje, el OOL que está ejecutando sobre el RPC determina dinámicamente que interfaces están disponibles. Usando un ORB dinámico se puede agregar nuevo hardware a un sistema sin necesidad de recompilar el ORB.

#### **1.4. “Middleware” en cliente/servidor.**

El “middleware” es un software diseñado para conectar (o comunicar) aplicaciones a bases de datos o para facilitar la operación o administración de una red y que está en el medio, entre la interfaz gráfica del usuario y las aplicaciones en el cliente y las aplicaciones o sistemas de administración de bases de datos en el servidor.

Existen los siguientes tipos de software “middleware”:

##### **1.4.1. Productos de acceso remoto a bases de datos.**

- DRDA de IBM
- RDA de ISO/ANSI
- ODBC de Microsoft
- IDAPI de Borland
- RDA de SQL Access Group

##### **1.4.2. Productos Gateway.**

Productos de acceso remoto a bases de datos para conexión server/”mainframe” corporativo).

- SQL Bridge
- SQL Network

##### **1.4.3. Productos Remote Procedure Call - RPC**

**Escritos en lenguaje procedural.**

- RPC de OSF/DCE

- ONC de Sun
- RPC de TI
- NCS RPC de HP
- RPC Tool de NewWise

#### **1.4.4. Productos Object Request Broker – ORB**

Escritos en lenguaje orientado al objeto.

- DOMF de HP y Sun
- ACA de Digital Equipment Corp.
- DOMS de Hyperdesk

#### **1.4.5. Productos de encolamiento de mensajes.**

- Pipes de Peerlogic
- Tooltalk Message de SunSoft
- DECMessageQ de DEC
- Communication Integrator de Covia Technologies

#### **1.4.6. Productos de comunicación interprocesos.**

- UNIX Sockets
- HP Sockets
- VAX Sockets

#### **1.4.7. Productos de acceso remoto a archivos.**

- DFS de OSF/DCE
- RFS de AT&T

#### **1.4.8. Productos de servicio de ruteo de mensajes.**

- MAPI
- CMC de XAPIA
- Mailbus de DEC
- SNADS de IBM

**1.4.9. Productos de servicios de directorio.**

- Global Naming de OSF/DCE
- Yellow Pages de Sun

**1.4.10. Productos de servicios conversacionales.**

- TP de OSI
- APPC de IBM

**1.4.11. Productos de servicios de tiempo.**

- Time Service de OSF/DCE
- Netware TSS

**1.4.12. Productos de servicio de terminales.**

- Telnet
- VTxxx Emulation
- 32xx Emulation
- 52xx Emulation

**1.4.13. Productos de administración de redes.**

- OSF/DME
- Open-View
- NetView
- DECmmc
- SunNet Manager

**1.4.14. Productos de seguridad de redes.**

- Kerberos de OSF/DCE
- RSA Public Key Encryption
- Digital Signature Standard (DSS)

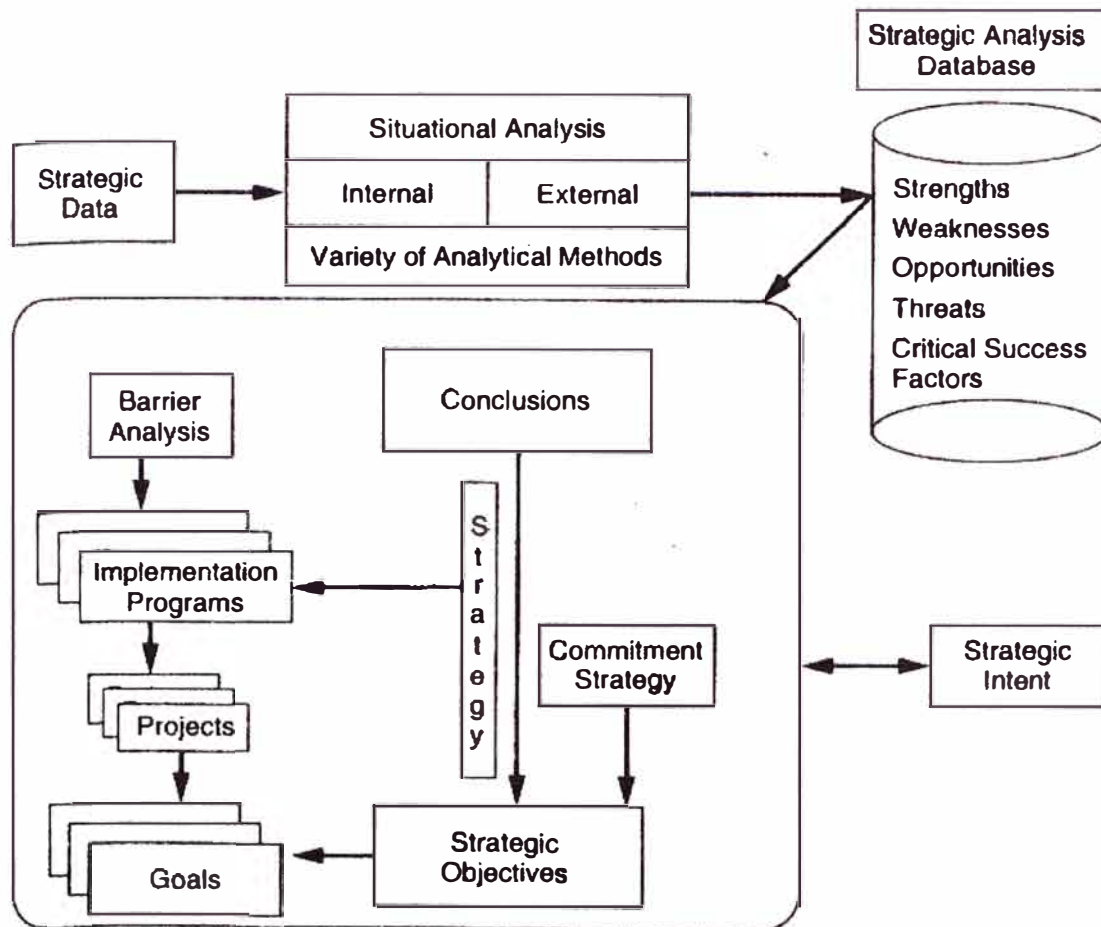
### **1.5. Implantación de la comunicación cliente/servidor.**

El propósito de esta sección es proveer una evaluación gerencial de las ventajas competitivas de negocios que pueden ser acumuladas al construir la arquitectura de los sistemas de información de la corporación sobre una infraestructura de comunicación cliente/servidor. La tesis es que la comunicación cliente/servidor representa una nueva, diferente y ventajosa capacidad que permite nuevas y mejores soluciones (más baratas, más rápidas y/o características/funciones mejoradas) a ser aplicadas a las prácticas de negocio de la empresa. La comunicación cliente/servidor posibilita que los activos de las Tecnologías de la Información T/I (voz, conectividad de datos, proceso y datos) sean unificados y conectados por vez primera. Si se reacciona a esto con una respuesta estratégica apropiada, esto tiene tremendas implicancias en términos de mejorar la posición competitiva de la empresa.

La comunicación cliente/servidor ofrece tal oportunidad de ventaja competitiva que necesita ser gerenciada e implantada como una iniciativa estratégica de negocios – no simplemente otra introducción de nueva tecnología.

#### **1.5.1. La gerencia estratégica.**

La gerencia estratégica se refiere a la disciplina por la cual objetivos y estrategias son desarrollados para guiar la dirección a largo plazo de la empresa. Estratégico implica que el ítem es de consecuencias mayores, que su logro es imperativo y tendrá un impacto duradero. Si es estratégico, es crítico para el bienestar de la compañía y a menudo requerirá muchos cambios a través de toda la organización. Los temas estratégicos son manejados por el directorio de la empresa. El ejercicio de planeamiento estratégico provee un medio para inyectar pensamiento estratégico como pensamiento diario a todo el equipo gerencial. La figura 1.4 provee una vista del proceso de planeamiento estratégico, el que puede ser entendido como sigue:



**Figura 1.4 Proceso de Planeamiento Estratégico**

- Los datos son recolectados por área estratégica (competencia, comercialización, recursos humanos, productos, gerencia de sistemas, etc.) para análisis interno y externo. Métodos analíticos tales como análisis SWOT, significando Fortalezas, Debilidades, Oportunidades y Riesgos), Factores Críticos de Éxito y análisis Five Forces (Cinco Fuerzas) son usados para interpretar los datos y desarrollar comprensión de la situación actual y su evolución. Este entero proceso de recolección de datos y análisis es llamado análisis situacional y resulta en la creación de una base de datos que sirve como un repositorio de hechos para tomar decisiones.
- Se llegan a conclusiones sobre temas que requieren una iniciativa o respuesta estratégica.
- Los objetivos son definidos. Ellos identifican donde la empresa necesita estar. Son específicos, medibles y fechados.
- Las estrategias son desarrolladas como acciones a propósito y coherentes para obtener los objetivos.

- Las estrategias son hechas operativas a través de programas de implantación que son dirigidos para ejecutar proyectos. Los programas de implantación logran metas interinas que son hitos en la ruta para alcanzar los objetivos.

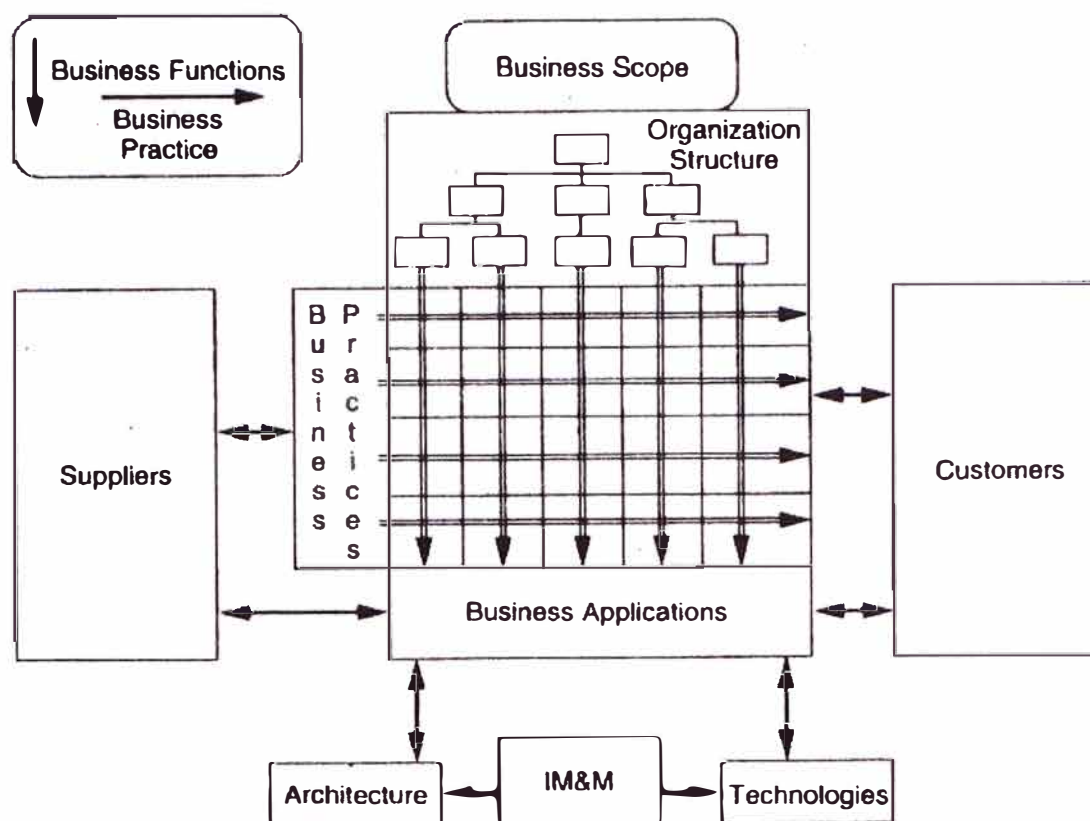
### 1.5.2. La perspectiva gerencial.

La figura 1.5 ilustra la estructura de la empresa y la relación de IM&M (Information Movement and Management) a esa estructura.

**Nota.- IM&M es también conocida como Tecnologías de la Información T/I.**

- Un negocio tiene un alcance que define los principales atributos y límites del negocio. Un alcance de negocio típico es ilustrado en la tabla 1.1.
- La empresa se organiza a sí misma en unidades funcionales para realizar su misión.
- Una unidad organizacional ejecuta una o más funciones. La empresa entrega productos/servicios y opera ejecutando prácticas de negocio que son colecciones horizontales de funciones. Las prácticas trabajan para el logro de los objetivos internos de la organización así como interactúan con las entidades externas (clientes y proveedores). Las prácticas de negocio son descompuestas en capacidades que representan requerimientos perdurables de la práctica.
- Las capacidades de las prácticas de negocios son automatizadas como aplicaciones de negocios, las que utilizan T/I apropiadas para acumular ventajas en su ejecución. El beneficio puede ser económico, oportunidad, capacidad, volumen, nivel de servicio, confiabilidad, receptividad, etc.

La conclusión de la perspectiva gerencial de la relación entre la empresa y las T/I es que el propósito de las T/I es posibilitar las prácticas de negocio. Las soluciones de T/I deben ser enlazadas a una práctica de negocio con el propósito de obtener ventaja. Las prácticas de negocio están primero que las T/I.



**Figura 1.5** Perspectiva gerencial de la empresa y las T/I ó IM&M

**Tabla 1.1** Modelo de Alcances de Negocios

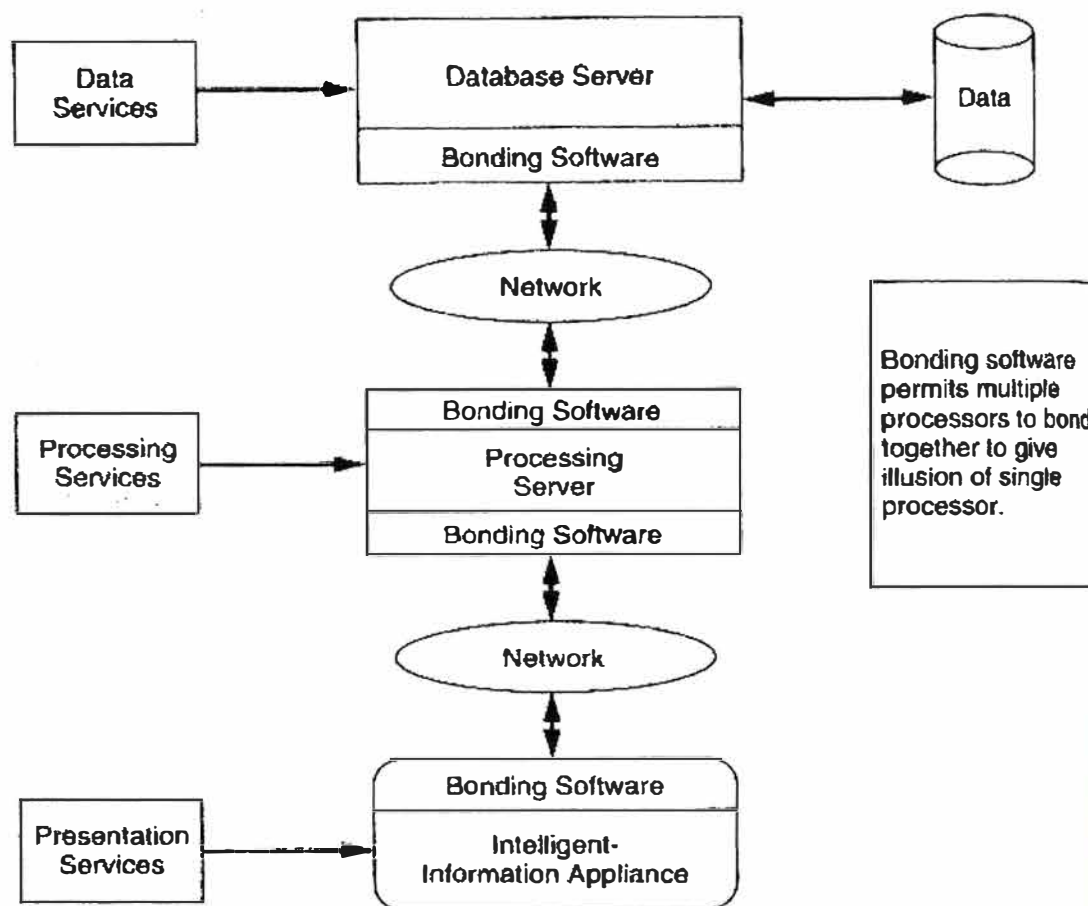
Atributo de Alcance de Negocio	Definición de Atributo
Misión	El propósito de la empresa.
Valores	Aquello en lo que la empresa cree.
Clientes/mercados	A quiénes vendemos?
Geografía	Dónde vendemos?
Productos/servicios	Qué vendemos?
Propósito estratégico	Cuál es la ambición a largo plazo de nuestras acciones?
Fuerza impulsora	Cuáles son los determinantes primarios de los productos y servicios que vendemos?
Ventaja competitiva sostenible	Qué de nosotros atrae a nuestros clientes y disuade a los competidores?



### 1.5.3. La arquitectura de la comunicación cliente/servidor.

Esta arquitectura ilustrada en la figura 1.6, es utilizada para permitir que programas en dos procesadores cooperen transparentemente para completar una transacción como una sola tarea unificada.

La tabla 1.2 reseña los atributos de arquitectura que hacen a una arquitectura generalmente mejor o más poderosa. La suma de todos los atributos es la maniobrabilidad.



**Figura 1.6 Arquitectura de comunicación cliente/servidor**

Sólo la arquitectura de comunicación cliente/servidor cumple los requerimientos de atributos de una arquitectura poderosa, en contraste con las otras arquitecturas como:

- Proceso cooperativo.
- Intercambio de mensajes.
- Computación personal.
- Tiempo compartido.
- "Host" OLTP (On Line Transaction Processing).
- "Host" batch (por lotes).

**Tabla 1.2 Atributos de una arquitectura poderosa**

<b>Atributo</b>	<b>Definición</b>
Mantenibilidad	La facilidad de dar mantenimiento a la arquitectura.
Modularidad	La capacidad de agregar, modificar y remover piezas de la arquitectura.
Escalabilidad	La capacidad de escalar la arquitectura por las dimensiones de volumen de transacción, volumen de almacenamiento, usuarios concurrentes, etc.
Adaptabilidad	La facilidad de cambio.
Portabilidad	La capacidad de mover aplicaciones a través de la arquitectura.
Estándares abiertos	El cumplimiento de la arquitectura con los estándares abiertos.
Autonomía	La capacidad de cada parte para funcionar independientemente y como parte del todo.
Flexibilidad	La capacidad para crecer y contraerse
Accesibilidad de datos	La capacidad de acceder datos local y remotamente.
Interoperabilidad	La capacidad de trabajar cooperativamente entre múltiples procesadores heterogéneos.
Conectividad de artefactos	La capacidad de agregar una amplia variedad de artefactos de información
Maniobrabilidad	La suma de todos los atributos que posiciona a la empresa para actuar y reaccionar a la dinámica del mercado.

#### **1.5.4. Reposicionamiento de la empresa para la comunicación cliente/servidor.**

Se analizan en esta sección los temas implicados en el reposicionamiento de una organización para pasar de una posición anterior a la comunicación cliente/servidor hacia la posición donde se logra la implantación de ella. La discusión es particionada como sigue:

### **a) El proveedor.**

Hay tres temas de amoldamiento estratégico en lo que concierne a la comunicación cliente/servidor y proveedores:

- Out-sourcing (Tercerización). La práctica de comprar todo o parte de servicios de T/I a un proveedor externo. Se recomienda tercerizar sólo las funciones de los servicios públicos.
- Proveedor socio de negocios. Comprometerse en una relación preferente de largo plazo con un proveedor. Se recomienda el socio Deming (definido por el Dr. W.E. Deming).
- Táctica de un solo proveedor. Comprar soluciones cliente/servidor a un solo proveedor. Se recomienda sólo como un paso táctico al comenzar el proceso de aprendizaje.

### **b) Tecnología.**

Los temas a ser cubiertos son:

- Injertar SQL. El prospecto de acelerar la migración hacia la comunicación cliente/servidor injertando un lenguaje de manipulación de datos SQL en las bases de datos jerárquicas y de redes. Se diagnostica poca probabilidad de éxito.
- Rol del “mainframe”. El “mainframe” migra gradualmente de su rol de manejador de datos, ambiente de desarrollo de programas y de ejecución, y manejador de la presentación al rol de servidor de recursos (servidor de datos, servidor de procesos por lotes y administrador de la red de datos) de la corporación.
- Grupos de tecnología avanzada. Enlazan la tecnología a los requerimientos de capacidades del negocio a través del uso de modelos.

### **c) Estructura de la organización.**

Se analiza cómo la organización de las T/I debe reorganizarse para mejor soportar el cambio de paradigma hacia la comunicación cliente/servidor. Se recomienda:

- Organizar por función opuesto a por tecnología. La organización de T/I debe ser coordinada para planear, diseñar, distribuir operar, etc. el proyecto completo.
- Posicionar las unidades como clientes y servidores unos de otros. Esto proveerá interfases limpias entre funciones horizontales y permitirá juntar funciones como se requieran en los grupos de tarea.

**d) Recursos humanos.**

De vital importancia para el reposicionamiento del profesional en T/I son los temas de:

- Habilidades. A las tradicionales habilidades en procesamiento, comunicaciones, manejo de datos, desarrollo de aplicaciones se deben agregar ahora manejo de procesos, T/I aplicada e integración de sistemas.
- Valores/conductas. Hacer reingeniería de los sistemas de recursos humanos para reforzar conductas positivas impulsadas por valores.

**e) Sistemas de información.**

Se sugieren las siguientes acciones para migrar gradualmente al ambiente cliente/servidor:

- Desarrollar una arquitectura de información.
- Implantar la administración de datos.
- Hacer desarrollo por capas: presentación, procesamiento y datos.
- Tender un puente hacia las bases de datos heredadas. Mover presentación y proceso fuera del “host”.
- Hacer la arquitectura de soporte de decisiones primero.
- Revalorizar las aplicaciones heredadas.
- Construir servidores OLTP ( On Line Transaction Processing) y OSS ( Operation Support Systems) y luego enlazarlos con DDBMS (Distributed Data Base Management System).

**f) Sistemas gerenciales.**

Se recomienda lo siguiente:

- Hacer planeamiento estratégico.
- Insistir en la calidad.
- Analizar competidores.
- Aplicar T/I a las funciones de T/I.

**g) Ventaja competitiva sostenible.**

Ciertas iniciativas estratégicas trascienden la justificación del costo. Hay tres inversiones en infraestructura que tienen este atributo:

- Red punto a punto de alto ancho de banda. La comunicación cliente/servidor causará una explosión en el tráfico de la red punto a punto, que puede ser anticipada.
- Sistema para manejo de configuración. Es un requerimiento de capacidad para el cambio de paradigma.
- Operaciones de extremo a extremo, administración y capacidad de mantenimiento.

#### **h) Resumen de reposicionamiento.**

En esta sección se han analizado los temas principales implicados al reposicionar la empresa en respuesta al cambio de paradigma. Es claro que la comunicación cliente/servidor es mucho más que un problema de transferencia de tecnología: requiere una implantación estratégica para reposicionar todas las áreas estratégicas impactadas en armonía con el nuevo paradigma.

## **CAPÍTULO II**

### **PROYECTO CONSULTA EN LÍNEA DE ASEGURADOS**

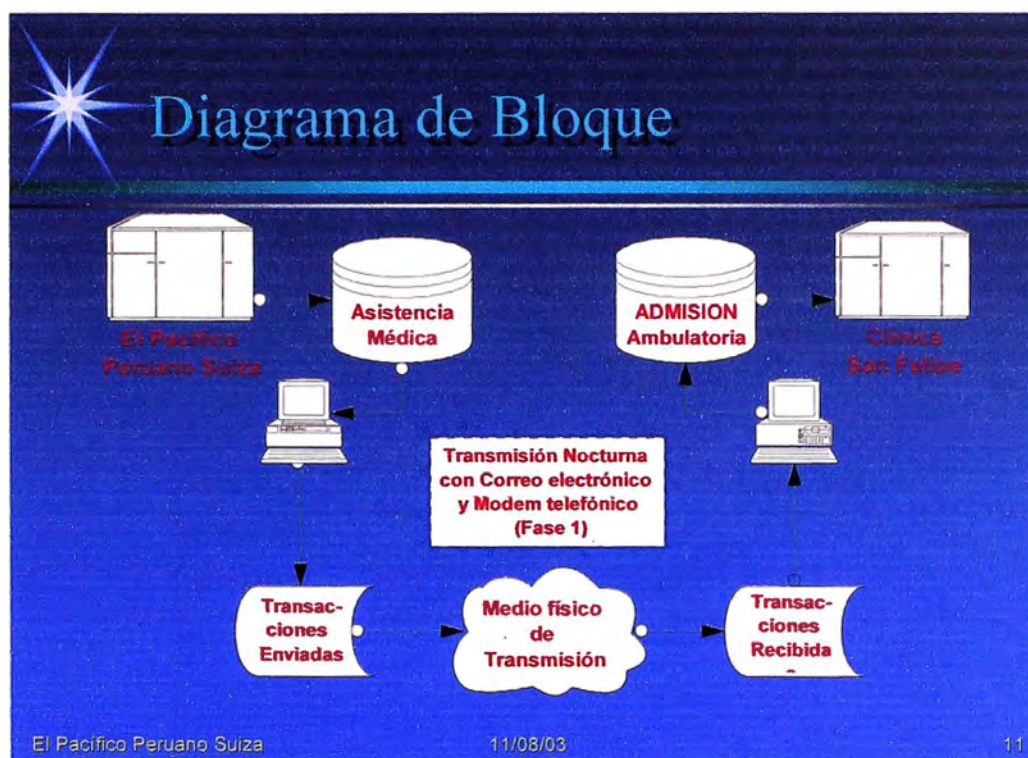
#### **2.1. Antecedentes del proyecto.**

La compañía de seguros El Pacífico Peruano Suiza, en adelante PPS, inició el año 1996 el envío de información de pólizas, planes y asegurados a varias clínicas afiliadas vía correo electrónico en la forma de archivos adjuntos. Este envío era una vez al día por la noche.

Al recibir el operador de la clínica el correo electrónico, extraía los archivos adjuntos y los alimentaba al sistema de admisión de la clínica, que actualizaba los maestros de pólizas, planes y asegurados de PPS en la clínica afiliada. Con esta información actualizada, el departamento de admisión de la clínica afiliada, podía verificar si una persona era asegurada de PPS, si estaba habilitado para atención y qué valores de deducible y coaseguros le correspondían.

Esta modalidad de intercambio de información fue un avance, respecto a la tradicional, la cual era el envío de pólizas y planes impresos a las clínicas afiliadas. No se actualizaba la información de asegurados, a los cuales sólo se les exigía la presentación del carnet emitido por PPS para su atención, pudiendo estar el asegurado inhabilitado, sin saberlo la clínica afiliada por no contar con la información.

El sistema de envío de información vía correo electrónico, cuyo diagrama de bloque se muestra en la figura 2.1, sirvió por varios años. Empero adolecía de dos desventajas, una era que la información estaba atrasada un día y la segunda era que en algunos casos la información de pólizas, planes y asegurados que poseían las clínicas afiliadas no era igual que la de PPS, debido a varias causas: no se ejecutó proceso de actualización en la clínica, sistema de actualización de la clínica tenía errores, etc.



**Figura 2.1 Envío de transacciones de pólizas, planes y asegurados a clínicas.**

## 2.2. Estado tecnológico de mercado asegurador y clínicas.

Al año 2000, la tecnología de comunicaciones había avanzado y más importante aún era que estaba siendo adquirida y usada por las compañías de seguros y las clínicas para interconectar sus procesos de negocios.

Entre las tecnologías de comunicaciones adquiridas podemos mencionar:

- Enlace dedicado de Internet.
- ISDN.
- ADSL.
- Enlace dedicado de Radio Módem.

Asimismo al año 2000, 4 compañías de seguros: Generali, La Positiva, Fénix y Novasalud, contaban cada una por separado con una solución que daba acceso en línea a la información de sus asegurados a 14 clínicas, detalladas en Tabla 2.1.

<b>Clínica</b>	<b>Tipo de enlace</b>
Ricardo Palma	Enlace dedicado de Internet
San Felipe	Enlace dedicado de Internet
San Pablo	Enlace dedicado de Internet
San Borja	Enlace dedicado de Internet
Stella Maris	ISDN
Angloamericana	ADSL
Javier Prado	Enlace dedicado de Radio Módem
Vesalio	Enlace dedicado de Radio Módem
Santa Teresa	Enlace dedicado de Radio Módem
Tezza	Enlace dedicado de Radio Módem
Mater Dei	Línea conmutada
Montefiori	Línea conmutada
Good Hope	Línea conmutada
Cruz Verde	Línea conmutada

**Tabla 2.1 Clínicas Interconectadas**

### **2.3. Toma de decisión sobre ejecución del proyecto.**

Para superar las 2 desventajas mencionadas del envío de información a las clínicas vía correo electrónico, y dado que ya la tecnología de comunicaciones había evolucionado y sobre todo adquirida por algunas clínicas, a mediados del año 2000 PPS decidió implantar un Sistema de Consulta en Línea de Asegurados, que facultara a las clínicas afiliadas a acceder en línea a las bases de datos de pólizas, planes y asegurados de PPS.

### **2.4. Formación de grupo de trabajo del proyecto.**

PPS asignó 2 especialistas para el proyecto Consulta en Línea de Asegurados: un Analista de Sistemas, especializado en Sistemas de Salud, como responsable del proyecto y un Analista de Comunicaciones, como asistente técnico.

### **2.5. Inicio de labor de grupo de trabajo.**

El grupo de trabajo inició su labor identificando primero los retos inherentes al proyecto:

- Interconexión de plataformas distintas. PPS contaba con una plataforma IBM AS/400, mientras que las clínicas contaban con una variedad de plataformas: Unix, AIX, Windows NT, etc.



- Al año 2000 en PPS el desarrollo de aplicaciones para Organizaciones Externas (clínicas, agentes de seguros, etc.) se había orientado sobre la plataforma Web. Sin embargo, la mayoría de clínicas tenía terminales tontos en sus módulos de Admisión, que no cuentan con capacidad gráfica ni disponibilidad de un navegador para desplegar una solución web.

Al descartarse una solución Web para interconectar procesos de negocio de PPS con las clínicas, se **estableció que la solución** para el proyecto Consulta en Línea de Asegurados **debía ser una solución cliente/servidor.**

Para el reto de interconectar plataformas distintas, PPS no contaba con personal técnico con experiencia en el tema y ante el costo en tiempo y dinero de desarrollar ese conocimiento internamente en la empresa, **se decidió que una mejor alternativa era tercerizar la solución, encargándosela a un proveedor de soluciones de comunicación en línea,** que existiera en el mercado.

## **CAPÍTULO III**

### **SELECCIÓN DEL PROVEEDOR DE “MIDDLEWARE” DE COMUNICACIONES**

#### **3.1. Búsqueda de proveedor.**

Decidida la implantación de una solución de comunicación en línea entre la compañía de seguros El Pacífico Peruano Suiza y sus clínicas afiliadas (mediados del 2000), se procedió a la búsqueda en el mercado peruano de proveedores de soluciones de comunicación en línea, contactándose a 2 compañías: Omnisciencia Consultores y Novatronic.

#### **3.2. Criterios utilizados en selección de proveedor.**

Se utilizaron los siguientes criterios:

##### **3.2.1. Cartera de clientes.**

Se requirió una lista de clientes con los que hubiera trabajado cada compañía, para poderles consultar sobre la satisfacción del trabajo desarrollado, si fuese necesario.

##### **3.2.2. Proyectos desarrollados.**

Se requirió una lista de los proyectos desarrollados por cada compañía, en los que hubiera de por medio una solución de comunicación en línea.

##### **3.2.3. Propuesta técnica.**

Se pidió a las 2 compañías que presentaran una propuesta técnica para la interconexión de El Pacífico Peruano Suiza con 15 clínicas afiliadas, que incluyera:

- Descripción de la solución técnica.
- Requerimientos de hardware y software.
- Estrategia de Implantación.

##### **3.2.4. Propuesta económica.**

Se pidió a las 2 compañías que presentaran sus condiciones económicas.

### **3.3. Propuestas.**

#### **3.3.1. Omnisciencia Consultores.**

Omnisciencia ofrece la integración de los Sistemas locales de las Clínicas para la atención de asegurados con los procesos de El Pacífico Peruano Suiza para la consulta del estado del asegurado y sus condiciones de atención médica, a través de su producto “middleware” TEN, el cual está basado en la transmisión y conversión de mensajes entre computadores. Para una eficiente administración de recursos, tanto en el servidor de comunicaciones como en los clientes intermedios se utilizan extensivamente instancias de procesos- las cuales se generan por demanda- y colas de mensajes entre los diferentes componentes.

El sistema ha sido diseñado bajo una arquitectura cliente/servidor y utiliza una modalidad de acceso a los datos en forma transaccional. Gracias a este concepto transaccional, el sistema trabaja en función a mensajes entre los diferentes módulos, los mismos que pueden interactuar en forma local y/o remota. Esto permite que los clientes y los aplicativos conectados al sistema puedan ser escritos en diferentes lenguajes de programación: Visual Basic, Visual Foxpro, Visual C, Visual Java, Clipper, Foxpro, Cobol, C, Pascal, RPG, etc., y facilitar de esta forma la adición de nuevos servicios a los sistemas existentes.

El sistema permite el uso de prácticamente cualquier host o equipo multiusuario como equipo intermedio entre las estaciones cliente finales y el servidor de comunicaciones, como por ejemplo AS/400, UNIX, Windows NT, IBM /390, etc.

#### **a) Módulos de administración y control.**

- **Módulo de comunicaciones en el servidor.**

Realiza la conexión con las estaciones remotas a través de los equipos de comunicaciones, verificando la integridad de los datos enviados o recibidos para su corrección.

- **Módulo de comunicaciones en la estación remota.**

Es el que interactúa con el módulo de comunicaciones del servidor principal a través del equipo de comunicación, para la transferencia de señales de control o datos, manejando los posibles errores en la línea de enlace.

- **Módulo de configuración de ambiente de comunicaciones.**

Permite definir los parámetros de comunicaciones, tales como direcciones IP de servidores, clientes, administración de fallas, protocolos a usar, interfaces de red, etc.

- **Módulo de monitoreo y control.**

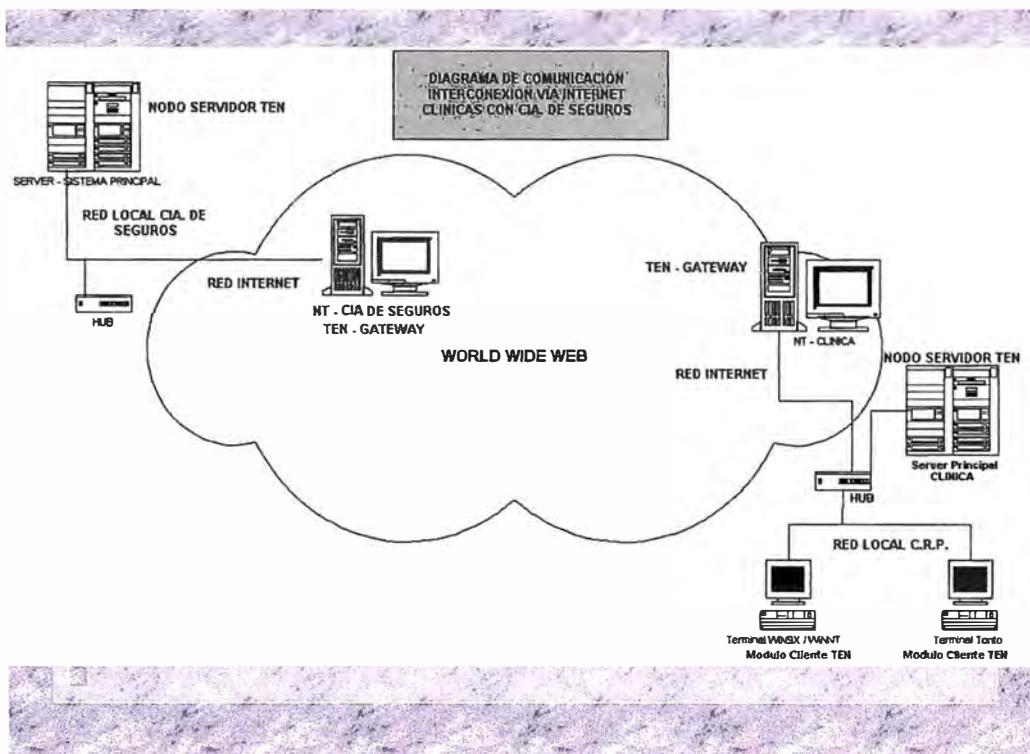
Permite monitorear la actividad del sistema y verificar el estado normal de sus componentes, como por ejemplo usuarios conectados, transacciones realizadas, errores detectados, etc.

- **Módulo de auditoría.**

Permite almacenar información para auditoría de eventos (log) donde se registrará la fecha y hora en que se enviaron y recibieron las transacciones y quién las realizó y el estado de todas las operaciones que se realizaron.

**b) Diagrama de comunicación.**

En la figura 3.1 se muestra el diagrama de comunicación de las clínicas con la compañía de seguros.



**Figura 3.1 Interconexión vía Internet de Clínicas con Cía de Seguros**

**c) Requerimientos mínimos de hardware y software.**

**Nodo Servidor (Compañía de Seguros)**

- Servidor de comunicaciones.  
Equipo de alta capacidad transaccional: Pentium III 500 Mhz (\*)  
128 MB de memoria RAM (recomendado)  
Tarjeta de red de alta velocidad: 10/100 Mbps  
400 MB de espacio libre en disco duro SCSI –2 / Ultra Wide SCSI (\*\*)  
Software: Windows NT Server v 4.0
- Servidor SNA.  
Equipo de alta capacidad transaccional: Pentium III 500 Mhz (\*)  
128 MB de memoria RAM (recomendado)  
Tarjeta de red de alta velocidad: 10/100 Mbps  
400 MB de espacio libre en disco duro SCSI –2 / Ultra Wide SCSI (\*\*)  
Software: Windows NT Server v 4.0  
SNA Server v 3.0
- Servidor Gateway a Internet.  
Pentium de 300 Mhz (\*)  
64 MB de memoria RAM  
150 MB de espacio libre en disco duro (\*\*)  
Software: Windows NT server v 4.0  
IP válida en Internet para acceso desde al WWW.  
Este equipo es necesario sólo si se desea aislar el TEN Server de Internet

(\*) Equipo fue adquirido en octubre de 2000, según valores dados por proveedor.

(\*\*) Espacio requerido sólo para instalación de software indicado por proveedor.

- Enlace para comunicación.  
Alguno de los siguientes:  
Línea dedicada a Internet.  
Enlace para red privada de radiomódem.  
Enlaces dedicados Interlan / Digired o similar.
- Programas en servidor de aplicaciones (AS/400) de la compañía de seguros que resuelvan :

La consultas por código de asegurado.

La consulta por nombre de asegurado.

La consulta de beneficios normales.

La consulta de beneficios adicionales.

La consulta de exclusiones.

La consulta de observaciones.

La consulta de condiciones especiales.

### **Nodo Cliente (Clínicas)**

- Conexión remota.

Cualquiera de las opciones siguientes:

Línea dedicada a Internet.

Acceso por línea conmutada (ISDN / Infovía con cuenta de acceso a Internet)

Enlace dedicado por radiomódem.

Enlace punto a punto (Digired o similar).

- Software:

Cualquiera de los siguientes:

Windows 95

Windows NT

SCO UNIX

AS/400

AIX

UNIX SUN Solaris

UNIX Digital

UNIX DG

### **d) Estrategia de implantación.**

Para la implantación del proyecto se requerirá la ejecución de lo siguiente:

Desarrollo de las transacciones y programas necesarios para poder efectuar las diversas consultas o servicios que brindará la Consulta en Línea de Asegurados.

Para su ejecución será necesaria la participación en le proyecto de un analista de la compañía de seguros a tiempo parcial (10 horas semanales) por un lapso de 20 días útiles.

Los fuentes de los programas en RPG para la interacción con el AS/400 serán entregados para su revisión y aprobación antes de la puesta en producción.

**e) Costos del proyecto.**

Se muestran en la tabla 3.1.

**Tabla 3.1 Costos del Proyecto**

<b>Costos del Proyecto de Desarrollo de Interconexión con Clínicas</b>	<b>USD</b>
<b>OMNISCENCIA</b>	
<b>1. Costo de Sistema TEN</b>	<b>8000</b>
a) Licencia de uso para servidor de comunicaciones	
b) Licencia para usuarios locales ilimitados	
c) Licencias para 8 clínicas	
<b>2. Implantación en PPS y Clínicas</b>	<b>2200</b>
Desarrollo de los programas del lado de la Interfase en PPS y Clínicas	
<b>3. Licencia por 7 clínicas adicionales</b>	<b>1540</b>
(Licencia por clínica adicional USD 220)	
<b>Total Omnisciencia</b>	<b>11740</b>
<b>HARDWARE Y SOFTWARE</b>	
<b>1. Servidor de Comunicaciones + Windows NT V 4.0</b>	<b>2400</b>
<b>2. Servidor SNA + Windows NT v 4.0 + SNA Server V3.0</b>	<b>3000</b>
<b>3. Servidor Gateway a Internet + Windows NT v 4.0</b>	<b>2400</b>
<b>Total Hardware+Software</b>	<b>7800</b>
<b>Total Proyecto de Desarrollo</b>	<b>19540</b>

### **3.3.2. Novatronic.**

La compañía Novatronic ofrece la interconexión de El Pacífico Peruano Suiza con las clínicas afiliadas, a través de su producto SIX Cliente/Servidor, el cual brinda facilidades transaccionales entre múltiples plataformas y protocolos de comunicación.

Six es un software “middleware” que provee un ambiente que facilita el desarrollo de aplicaciones en esquema Cliente/Servidor. SIX permite que su software aplicativo pueda interactuar independientemente de los protocolos de comunicación de datos (TCP/IP, X.25, SNA, Asíncrono) en múltiples ambientes de desarrollo (“mainframe” IBM, AS/400, UNIX, NT, Windows, VMS y otros).

SIX está diseñado para simplificar la complejidad del manejo de protocolos, los cuales son accedidos dinámicamente de acuerdo a las características y configuración de la red de computación de su institución. SIX permite una fácil integración de sus sistemas actuales (legacy systems) con las modernas redes de servidores, por lo que es una herramienta ideal para la migración hacia la tecnología Cliente/Servidor.

SIX está compuesto de dos módulos principales: el Módulo Cliente y el Módulo Servidor.

El Módulo Cliente permite que un aplicativo escrito en lenguajes de cuarta generación o lenguajes de entorno gráfico, puedan enviar mensajes de requerimiento de información (request) a uno o múltiples aplicativos residentes en módulos Servidores o en computadores conectados al servidor a través de gateways (puertas de enlace), a fin de que éstos atiendan dicho requerimiento enviando un mensaje de respuesta (response).

El Módulo servidor administra los servicios brindados por los procesos aplicativos, así como los mecanismos de control de acceso y uso de estos servicios por parte de los clientes. Este módulo permite la definición del ambiente de comunicaciones y los protocolos que se tendrán disponibles.

La arquitectura del SIX, considera una interfase estándar que independiza a los aplicativos de las plataformas de hardware y software empleados, permitiendo al usuario la utilización del protocolo de comunicaciones, gestor de base de datos, lenguaje de aplicaciones e inclusive el sistema operativo, que más se ajuste a sus necesidades.

SIX le permite distribuir sus aplicaciones de acuerdo a sus necesidades, por ejemplo en un mismo servidor usted puede crear múltiples procesos para atender un mismo servicio, más aún usted puede decidir tener otro servidor aplicativo en un segundo (hardware) servidor físico.



Todas estas características, hacen del SIX una solución ideal a los nuevos requerimientos de los usuarios:

- Herramienta que facilite el desarrollo de aplicaciones Cliente/Servidor.
- Plataforma de integración de los diferentes equipamientos de computación y de redes con que cuentan las instituciones.
- Implementar aplicaciones distribuidas.
- Aislar el desarrollo aplicativo de la complejidad del manejo de las comunicaciones.

#### **a) Cliente SIX**

El Cliente SIX consiste de una serie de funciones:

- Establecimiento de sesiones con los servidores locales o remotos.
- Direccionamiento a una determinada aplicación dentro de un servidor.
- Envío de requerimientos a una determinada aplicación dentro de un servidor.
- Recepción de respuestas enviadas por los servidores.
- Cierre de sesión.

Estas funciones cuentan con una interfase estándar que puede ser invocada por programas escritos en diversos lenguajes como:

- COBOL, RPG
- C/C++
- Visual Basic, Visual C++
- Delphi, PowerBuilder

En ambientes DOS/Windows, UNIX, Netware, Windows NT, Windows 95, MVS, AS/400, OPEN MVS, etc.

#### **b) Servidor SIX**

El Servidor SIX es el corazón del sistema, y está compuesto por un conjunto de módulos que realizan las siguientes funciones:

- Recepción de los requerimientos de las estaciones Clientes (Request), provenientes de la red de comunicaciones.
- Control de acceso de los usuarios a los diferentes servidores aplicativos.
- Ruteo y entrega de los requerimientos a los aplicativos.
- Manejo de las respuestas y su envío a los clientes.

- Administración del ambiente de comunicaciones y protocolos.
- Administración y monitoreo del sistema.
- Registro de eventos (Logging).
- Otros utilitarios y funciones especializadas.

Los usuarios desarrollan sus servidores aplicativos, utilizando las facilidades del Servidor SIX, mediante una interfase estándar similar ala de los Clientes.

Al igual que los Clientes, los servidores aplicativos, pueden estar escritos en diversos lenguajes. Se provee una interfase para cada tipo de ambiente. Asimismo los servidores aplicativos pueden acceder a diversas bases de datos como DB2, Informix, Oracle, Sybase y otros.

El Servidor SIX está disponible para plataformas SCO UNIX, AIX (RS/6000), DG/UX, UNIX SVR4 y otras similares.

### **c) Gateways**

Si bien el sistema de comunicaciones se basa en un ambiente UNIX, los módulos servidores no necesariamente tienen que residir en este ambiente. La facilidad de gateways, permite que el módulo servidor pueda residir en cualquier otro ambiente tal como: mainframe IBM, AS/400, otro UNIX, NT, etc. Encargándose el SIX de rutear los mensajes (transacciones y servicios) a los computadores donde se encuentran los módulos aplicativos servidores.

Esta facilidad amplía las aplicaciones del SIX, permitiéndole competir en el campo de los conmutadores de transacciones y ampliando asimismo su capacidad de integración.

### **d) Protocolos.**

Los protocolos de transporte actualmente soportados, son los siguientes:

- Familia TCP/IP
- X.25
- SNA
- Asíncrono

### e) Aplicaciones.

La flexibilidad del SIX permite al usuario, una gran gama de aplicaciones, entre las cuales podemos mencionar:

- Desarrollo de aplicaciones Cliente/Servidor.
- Establecer redes de servicio remoto para clientes y proveedores.
- Establecer sistemas de transferencia electrónica de fondos que brinden servicios de cajeros automáticos, puntos de venta, respuesta por voz, entre otros.

### 3.4. Selección de proveedor

Para la selección del proveedor se trabajó un sistema de puntaje, asignando pesos a cada criterio de evaluación, dando mayor peso a los de las propuestas técnica y económica. Se muestra la matriz de evaluación de proveedores en la tabla 3.2.

De acuerdo a este sistema, se eligió al proveedor Omnisciencia.

**Tabla 3.2 Matriz de evaluación de proveedores**

Criterio de Evaluación	Peso	Omnisciencia		Novatronic	
		Evaluación (1-10)	Puntaje	Evaluación (1-10)	Puntaje
Cartera de Clientes	5	8	40	8	40
Proyectos Desarrollados	5	10	50	8	40
Propuesta Técnica	10	8	80	8	80
Propuesta Económica	10	6	60	0	0
			<b>230</b>		<b>160</b>

Nota.- El proveedor Novatronic, si bien estaba calificado técnicamente, finalmente no presentó su propuesta económica.

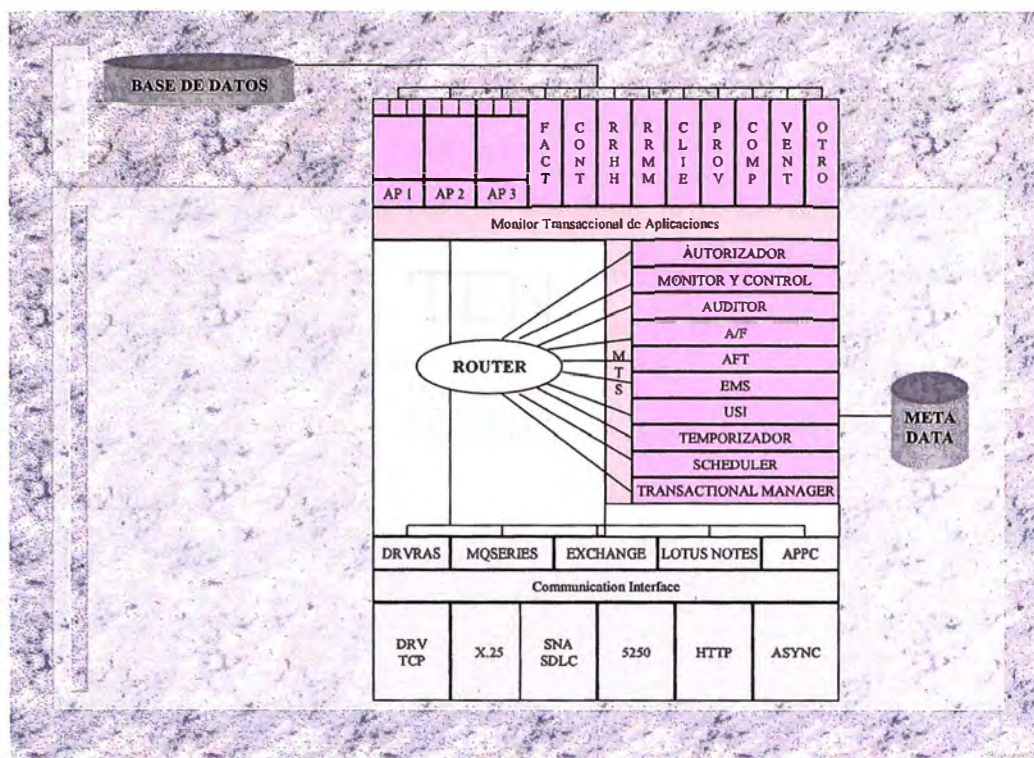
## **CAPÍTULO IV**

### **DESCRIPCIÓN DEL “MIDDLEWARE” SELECCIONADO: TEN**

#### **4.1. Conceptos generales.**

- El TEN es un “middleware”, es decir es un conjunto de módulos (software) que se incluyen entre el sistema operativo, las aplicaciones y los niveles de transporte de red, cuyo objetivo es facilitar algunos aspectos del proceso cooperativo.
- El TEN específicamente es un “middleware” orientado a mensajes, Messaging & Queuing M&Q (Mensajería y Encolamiento), que permite la comunicación entre programas que operan en el mismo sistema o en sistemas diferentes y heterogéneos, por medio de mensajes.
- Si se trata de programas operando en diferentes sistemas, la comunicación se realiza sin necesidad de tener establecida una conexión lógica y privada entre ellos. Esta comunicación se establece a través de una cola.
- El sistema de colas se interpone entre los procesos que intercambian datos. Recibe los datos de un proceso y los encamina al proceso destinatario.
- Permite activar los procesos a los que van destinados los mensajes.
- Si el proceso destinatario no está activo, acumula los mensajes según éstos van llegando a la cola, y los reenvía cuando el proceso vuelve a estar en condiciones normales de funcionamiento.
- Los objetos de TEN M&Q son: Queue Manager, colas, clases, listas de distribución y mensajes.
- Proporciona las API's que enmascaran la topología de la red y permiten a los programadores ignorar los problemas de la red. Son los puntos de acceso a los servicios que el Queue Manager proporciona a las aplicaciones.
- Soporta los tipos de comunicaciones siguientes: síncrona, asíncrona y envíos paralelos.

En figura 4.1 se muestra el esquema del TEN.



**Figura 4.1 Esquema del TEN**

#### 4.2. Arquitectura del sistema.

- TEN opera en un modelo cliente/servidor.
- La arquitectura en niveles permite que soporte un gran número de plataformas y protocolos de red.
- El núcleo del software es común en todas las plataformas soportadas.
- Este núcleo es el Queue Manager o Monitor.

#### 4.3. Queue Manager.

- El Queue Manager o Monitor, proporciona todos los servicios necesarios a las aplicaciones clientes, para intercambiar mensajes y ficheros
- El Queue Manager maneja una o más colas en la máquina local.
- Las colas pueden guardar cualquier clase de datos: ficheros o mensajes.
- El Queue Manager y cada una de sus colas, tienen nombres asociados. Las aplicaciones usan estos nombres para acceder al TEN.
- El Queue Manager tiene la responsabilidad de hacer llegar un fichero o mensaje a la cola destino, ya sea local o remota.

#### 4.4. Gestor de eventos.

- Cada cola puede manejar eventos de forma que se arranquen procesos a la llegada de un fichero o mensaje a dicha cola.
- En algunas plataformas (MVS, Windows NT), las colas pueden manejar eventos que arranquen procesos, además con las siguientes condiciones: a una fecha determinada, a la terminación de otro evento o combinación de las anteriores.

#### 4.5. Transporte.

El transporte de TEN puede utilizar diferentes protocolos de comunicaciones:

- DRV / TCP
- X.25
- SNA / SDLC
- 5250
- HTTP
- ASYNC

Características:

- Restart (reinicio) automático
- Sesiones paralelas.
- Transferencia de datos síncrona y asíncrona entre colas.
- Garantiza la transferencia de mensajes y ficheros.
- Garantiza la no pérdida ni duplicación de datos.
- Proporciona dos tipos de ACK (Acknowledgement - Reconocimiento).
  - A la recepción de los datos en el Queue Manager del sistema remoto.
  - Cuando los datos son consultados en el sistema remoto.
- Compresión de datos.
- Encriptación de datos.
- Autenticación de datos.
- Reconocimiento recíproco.

#### 4.6. Administración.

- En cada nodo TEN existe un fichero de Log (registro) donde quedan registradas las incidencias detectadas por los procesos de TEN.

- Los procesos de Transporte y Drivers permiten activar los ficheros de Trace.

## **CAPÍTULO V**

### **EJEMPLO DE UNA CONSULTA**

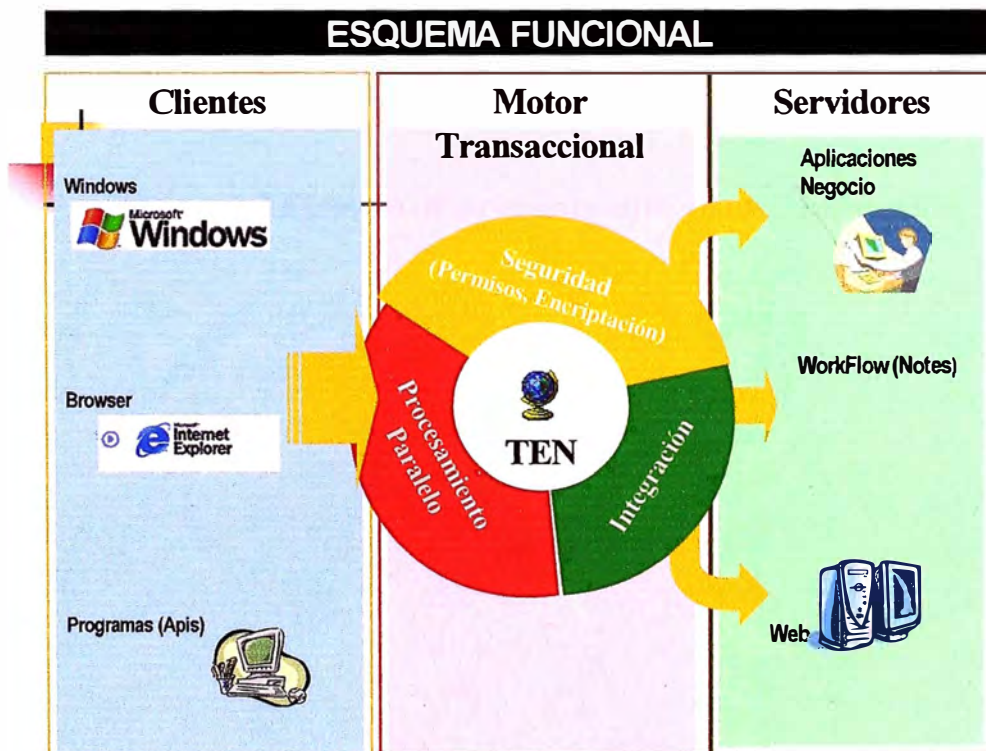
#### **5.1. Tipos de transacción**

El aplicativo Consulta en Línea de Asegurados ejecuta una serie de transacciones, como la búsqueda por código de asegurado, la búsqueda por nombre de asegurado ó el listado de beneficios. Ver anexo B: Transacciones del TEN Cliente para una lista completa de las transacciones y su diseño.

#### **5.2. Esquema funcional del TEN.**

Para poder ejecutar estas transacciones el aplicativo, recurre a las API's que proporciona el TEN. Los programadores usan estas API's para acceder a los servicios de un sistema remoto sin conocer el hardware ni el protocolo de red que se está utilizando. Ver anexo C: Parámetros del TEN Cliente y anexo D: Guía de Programación del TEN Cliente, donde se muestran las API's proporcionadas con el TEN. En figura 5.1 se muestra el esquema funcional del TEN.

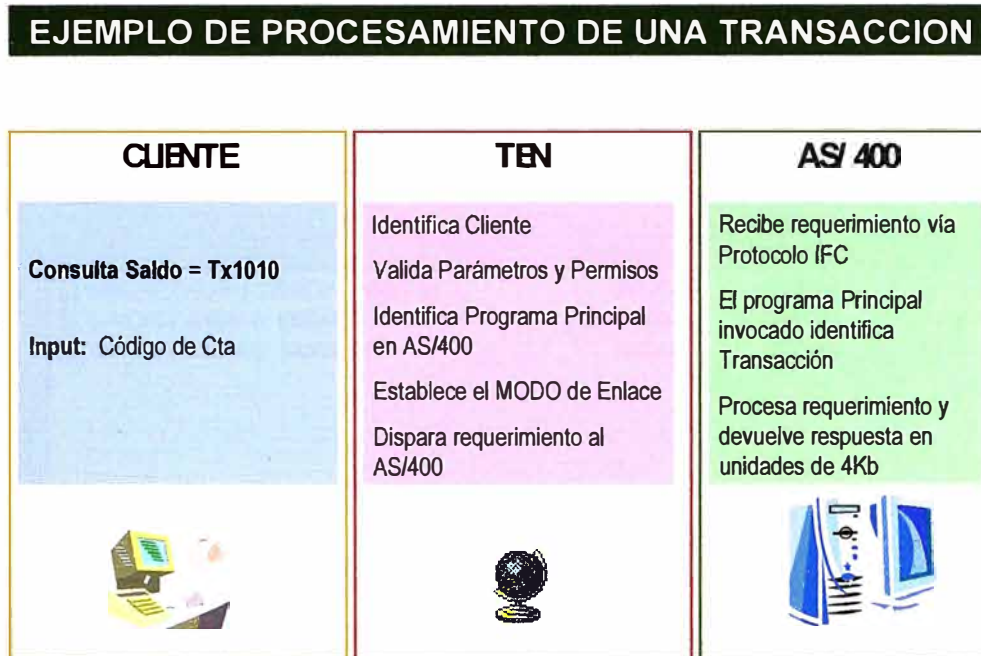




**Figura 5.1 Esquema funcional del TEN**

### 5.3. Procesamiento de una Transacción.

Se grafica el procesamiento de una transacción en figura 5.2.



**Figura 5.2 Ejemplo de procesamiento de una transacción**

#### 5.4. Ejemplo de transacciones.

Se muestra una transacción de búsqueda por nombre en la figura 5.3 y una transacción de búsqueda por código en la figura 5.4.

Nombre	Póliza	Certificado	Parentesco
SANCHEZ ALAYO LISSETH VANESSA	0000079	2003257	TITULAR
SANCHEZ ANGULO EDGARD	0000446	0000345	TITULAR
SANCHEZ ANGULO SANTOS E	0000446	0000161	TITULAR

Nro de Registros Encontrados : 0

Ten Cliente - Pacifico Peruano Suiza / Clínica Anglo Americana - Conectado...

Configurar... Servidor Ftp Salir Acerca de...

Busqueda por Código | Busqueda por Nombre | Consulta Log

Poliza: 0000079 Certificado: 2003257 Cod.Parent: T00

Nombre del Paciente: SANCHEZ ALAYO LISSETH VANESSA Plan: F

Nombre del Titular: SANCHEZ ALAYO LISSETH VANESSA Estado: VIGENTE

Fecha de Nacimiento: 05/08/1980 Parentesco: TITULAR Moneda: SOLES

Sexo: F Fecha de ingreso: 01/04/2003 Fecha Ult. Chequeo:

Número de Póliza: 0000079 Certificado: 2003257 Fecha Inicio Latenc.: 00/00/0000

Contratante: BANCO DE CREDITO DEL PERU Fecha Fin Latencia: 00/00/0000

Fin de vigencia de la cobertura del paciente: 30/06/2004 Documento:

Tipo Deducible	Importe	%Coaseguro	Interv.Días	Benef.Inicial	%C
AMBULATORIO POR CADA CONSULTA	30.00	30.00	0	10,000.00	
CREDITO HOSPITALARIO-IMPORTE POR INT...	125.00	0.00		0.00	
CHEQUEO MEDICO SIN DEDUCIBLE	0.00	0.00			

## **CAPÍTULO VI SERVICIOS WEB**

### **6.1. Antecedentes.**

En capítulos anteriores hemos visto una solución de una interconexión multiplataforma utilizando un “middleware” de comunicaciones. Sin embargo ésta es una solución particular que al usar API’s no abiertos, deviene en una tecnología propietaria que habilita la interconexión sólo para la empresa adquirente del “middleware” y sus clientes y proveedores, creándose un circuito cerrado entre ellas.

Al igual que el ejemplo precedente, en los últimos años, han estado evolucionando las aplicaciones de software de sistemas únicos en constelaciones que finalmente abarcan múltiples dispositivos, servicios y organizaciones. Sin embargo, la habilidad de interconectar estas soluciones sigue siendo un reto frustrante. Entre empresas y organizaciones, facilitar una comunicación transparente entre diferentes tecnologías y plataformas ha sido demasiado difícil, costoso y ha requerido de demasiado tiempo.

La necesidad de resolver este problema fundamental de la integración que ha existido desde el inicio de la computación, ha requerido de la implantación de estándares que proporcionan la base para lograr el objetivo de arquitectura de la interoperabilidad.

### **6.2. Definición.**

Los servicios Web XML son módulos de software construidos e integrados utilizando XML, que se pueden publicar, ubicar e invocar desde cualquier lugar de la Web o desde el interior de una red local basada en los estándares abiertos de Internet, y permiten que aplicaciones, servicios y dispositivos trabajen juntos de manera inherente. Esta interoperabilidad transparente del software, previamente inalcanzable debido a la naturaleza propietaria de las arquitecturas tradicionales de los componentes, se permite a través de los estándares abiertos de Internet que comprenden los servicios Web XML:

- **XML y HTTP** – en el centro de los servicios Web XML están XML y HTTP.

XML es el formato universal para los datos estructurados. Se trata de un conjunto de reglas que gobiernan el diseño de formatos de texto para los datos, de modo que se producen archivos sencillos de generar y leer, sin ambigüedad e independientes de la plataforma.

HTTP actúa como el protocolo de comunicaciones para pasar información con base en XML de manera transparente y segura sobre Internet y a través de servidores de seguridad corporativos.

- **SOAP** – El dialecto de servicios Web para intercambio, SOAP describe el contenido de los paquetes XML que se transmiten por Internet.
- **WSDL** – WSDL describe las capacidades de alto nivel de un servicio Web para que las aplicaciones del cliente puedan consumir fácilmente su funcionalidad.
- **UDDI** – Conforme continúan a proliferar los servicios Web XML, se vuelve un requerimiento un directorio completo de estos servicios. UDDI es el registro de servicios Web XML disponibles de manera global.

Lo que hace únicos y revolucionarios a los servicios Web XML es su simplicidad – un modelo con base en estándares que permite la federación de componentes con ejecución en cualquier lenguaje y sobre cualquier plataforma, permitiendo a las empresas integrar software e información tanto dentro de su propia infraestructura de informática como más allá de sus paredes empresariales. Este enfoque directo e intuitivo para la integración está resonando a través de todas las empresas en el mundo.

En figura 6.1 se muestra los tipos de conexión brindados por Internet y en figura 6.2 se muestra cómo los Web Services son una evolución natural de la tecnología Internet.



Figura 6.1 Tipos de conexión brindadas por tecnologías Internet

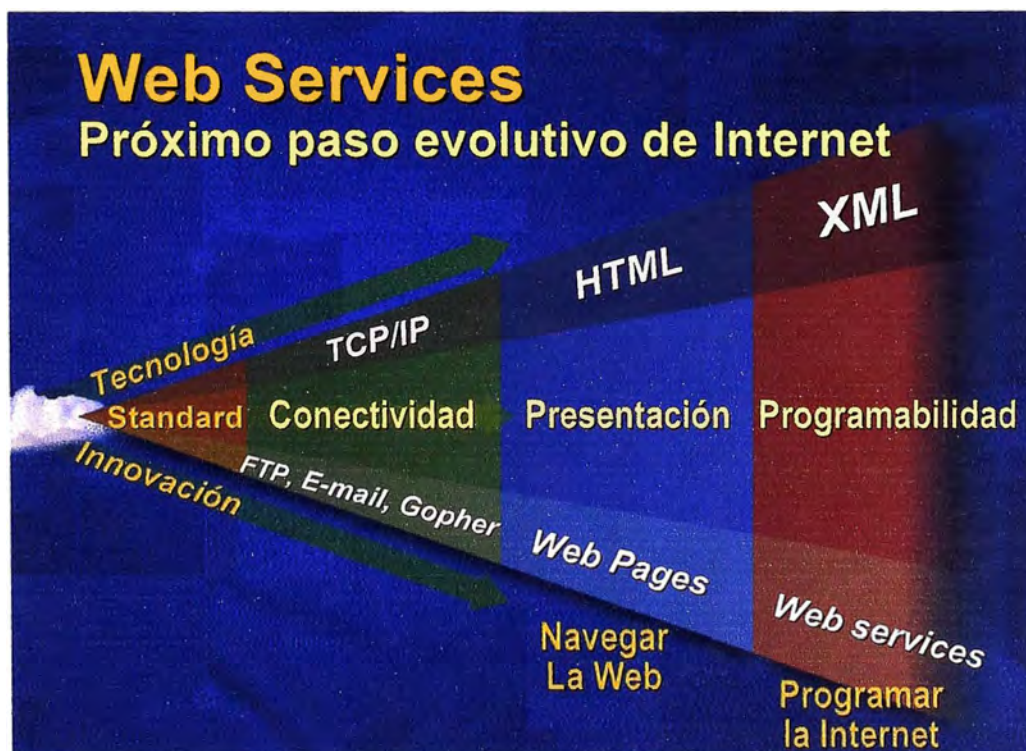


Figura 6.2 Pasos evolutivos de Internet

## CONCLUSIONES

1. Si bien en la actualidad hay la tendencia de interconectar los procesos de negocios de las empresas con los de sus clientes y proveedores mediante la plataforma Web, por la facilidad y economía que brindan los estándares abierto de Internet, para el proyecto Consulta en Línea de Asegurados de Pacífico Peruano Suiza, se estableció **una solución de arquitectura cliente/servidor** debido a que la mayoría de los terminales de las clínicas en los cuales se instalaría el aplicativo de consulta eran terminales tontos, que no cuentan con capacidad gráfica ni disponibilidad de un navegador para desplegar una solución web. Lo anterior ilustra el principio que lo importante no es el uso de la tecnología más nueva, sino el utilizar la tecnología apropiada para resolver la necesidad de negocio de la empresa.
2. Al no disponer la empresa Pacífico Peruano Suiza de personal técnico con experiencia en la interconexión entre distintas plataformas de comunicación y ante el costo en tiempo y dinero de desarrollar ese conocimiento internamente en la empresa, **se decidió que una mejor alternativa era tercerizar la solución, encargándosela a un proveedor de “middleware” de comunicaciones.** Las plataformas de comunicación soportadas por el “middleware” son: AS/400, UNIX, AIX, Windows NT, IBM /390. El proveedor se compromete a incluir la funcionalidad para cualquier otra plataforma nueva que surgiera por un costo adicional.
3. Al delegar la complejidad del manejo de las comunicaciones a un “middleware” de comunicaciones provisto por un tercero, la empresa pudo concentrar los recursos de su personal en el desarrollo de las aplicaciones de seguros de salud, que es el giro de negocio de la empresa. Esto ilustra otro principio de administración, que recomienda tercerizar los procesos de negocio que no forman parte del núcleo de negocio de la empresa.



4. El “middleware” de comunicaciones adquirido es un “middleware” orientado a mensajes (MOM), el cual es típicamente implantado como un producto propietario (lo que significa que es incompatible con otras implantaciones MOM de otros proveedores), por lo que la empresa dependerá del proveedor para soporte de mantenimiento y futuras mejoras, lo que podría tener un efecto negativo en la flexibilidad y mantenibilidad del sistema implantado. Sin embargo, evaluado contra el costo de desarrollar internamente en la empresa una solución de “middleware” única, este potencial efecto negativo puede ser visto como aceptable.
5. Los servicios Web XML introducen un nuevo paradigma en el proceso de creación de complejas aplicaciones distribuidas. Se presentan como el siguiente paso en la evolución de Internet. Se proyecta en el horizonte tecnológico la consolidación de los servicios Web en el mediano plazo.

## **ANEXO A**

## **DEFINICIONES DE CLIENTE/SERVIDOR**

- “Cualquier combinación de sistemas que pueden cooperar para dar a los usuarios toda la información que ellos necesitan, sin que tengan que preocuparse de dónde reside.”
- “Una arquitectura de procesamiento cooperativo donde una componente pide servicios de otra.”
- “Es un sistema de procesamiento distribuido.”
- “El término cliente/servidor es originalmente aplicado a la arquitectura de software que describe el procesamiento entre dos programas: una aplicación y un servicio soportante.”
- “El modelo de computación cliente/servidor representa una instancia específica del procesamiento cooperativo distribuido, donde la relación entre clientes y servidores es la relación entre componentes de hardware y software (la arquitectura).”

## **ANEXO B**

TABLA B.1 TRANSACCIONES DEL TEN CLIENTE

Trama de envío - Consulta por Código - 1010	Longitud	Posición	
Poliza	7	1	
Certificado	7	8	
Código Parentesco	3	15	
Código Clínica	6	18	000011
<b>TOTAL</b>	<b>23</b>		

Buscar en Tabla de Relación Poliza - Clínica-Categoría  
Luego con Categoría usar la Tabla de Deducibles  
correspondiente

Trama Recibida	Longitud	Posición	
Resultado Consulta	1	1	0=OK 1=Error
Código de Error	3	2	Código de Mensaje 099=No existe
Apellido Paterno del Paciente	30	5	RODRIGUEZ
Apellido Materno del Paciente	30	35	AMES
Nombre del Paciente	30	65	ALEIDA C.
Apellido Paterno del Titular	30	95	RODRIGUEZ
Apellido Materno del Titular	30	125	AMES
Nombre del Titular	30	155	GENARO A.
Sexo Paciente	1	185	F
Fecha de Nacimiento	8	186	19770817
Tipo Documento	2	194	AAAAMDD
Número de Documento	10	196	Tipo Documento. 01=RUC 02=LE/DNI 03=NIT (Nro. Ident. Tributaria)
Fecha Ingreso	8	206	Número de Documento
Fecha Salida	8	214	Fecha de Ingreso del Asegurado FMT= AAAA MM DD
Fecha Inicio de Latencia	8	222	Fecha de Salida del Asegurado FMT= AAAA MM DD
Fecha Fin Latencia	8	230	Fecha Inicio de Latencia del Asegurado FMT= AAAA MM DD
Chequeo	1	238	blanco =Sin Chequeo Médico X=Con Chequeo Médico
Fecha Último chequeo	8	239	
Flag Beneficios Adicionales	1	247	blanco =Sin Benef. Adic. Variables X=Con Benef. Adic. variables
Nro Beneficios Adic.	2	248	# de Beneficios Adicionales variables
Exclusion	1	250	blanco =Sin Exclusiones X=Con Exclusiones
Observac o preexisten	1	251	blanco =Sin Observaciones X=Con Observaciones
Flag Condiciones Especiales Certificado	1	252	blanco =Sin Cond. Especiales X=Con Condiciones Especiales
Nro condiciones Especiales	2	253	# de Condiciones Especiales de l certificado
Poliza	7	255	Número de la Póliza
Certificado	7	262	Número del certificado
Código Parentesco	3	269	Código de Parentesco: T00 - titular C00-cónyuge P00-padre M00-madre H01 hijo-1, H02...H99 hijo-99
Plan	3	272	Código de Plan del certificado
Contratante	40	275	Razon Social o Nombre del Cliente
Fecha Inicio de Vigencia	8	315	Fin Inicio de la Póliza FMT= AAAA MM DD
Fecha Fin Vigencia	8	323	Fecha Fin de Vigencia de la Póliza FMT= AAAA MM DD
Moneda	2	331	0 Moneda de la póliza 01=Dólares 99 =Soles
Estado	1	333	Código de Estado del Asegurado. blanco=Vigente A=asegurado Anulado / cesado H = Inhabilitado por falta de pago X = Beneficio Máximo copado , no se atiende L= Latencia
Tipo Deducible Ambulatorio	1	334	Tipo de deducible para crédito ambulatorio. C- por cada Consulta D- por cada Diagnóstico P- por persona cada XX días E- Beneficio excludido
Importe	11	335	Importe del deducible a cobrar (moneda de la póliza)
Intervalo Días	3	346	Número de días a transcurrir para cobrar otro deducible ( TIPDEA=P )
Porcentaje Coaseguro	5	349	Porcentaje coaseguro del asegurado por crédito ambulatorio.
			Tipo de deducible para crédito hospitalario.

Tipo Deducible Hospitalario	1	354	D- 1 día de cuarto M- 1 día de cuarto - importe	I - Importe por internamiento E- Beneficio excluido
Importe	11	355		Importe del deducible (crédito hospít.) a cobrar cuando el TIPDEH es I ó M (moneda de la póliza)
Porcentaje Coaseguro	5	366		Porcentaje de Coaseguro de Beneficio Inicial
Beneficio Inicial	11	371		Importe de Beneficio Inicial
Porcentaje Coaseguro Exceso	5	382		Porcentaje de Beneficio en Exceso
Importe Beneficio en exceso	11	387		Importe de Beneficio en Exceso
Tipo Deducible Pre-Post-natal	1	398		Tipo de deducible por Pre-post-natal S- sin deducible C- por cada consulta E- Beneficio excluido X- Beneficio copado
Importe deducible	11	399		Importe deducible por pre-post-natal a cobrar.
Coaseguro	5	410		Porcent. coaseg. por pre-post-natal
Tipo Deducible Aborto	1	415		Tipo de deducible por Aborto/amenaza de aborto S- sin deducible C- por cada consulta E- Beneficio excluido X- Beneficio copado
Importe	11	416		Importe deducible por aborto/amenaza a cobrar.
Coaseguro	5	427		Porcent. de coaseg. por aborto/amenaza
Tipo Deducible Medicion vista	1	432		Tipo de deducible por Medición de vista S- sin deducible C- por cada consulta E- Beneficio excluido X- Beneficio copado
Importe	11	433		Importe deducible por medición-vista a cobrar.
Coaseguro	5	444		Porcent. coaseg. por medición de vista
Tipo Deducible Odontológico	1	449		Tipo de deducible por Odontológico S- sin deducible C- por cada consulta E- Beneficio excluido X- Beneficio copado
Importe	11	450		Importe deducible por Odontológico a cobrar.
Coaseguro	5	461		Porcent. coaseg. por Odontológico
Tipo Deducible Evaluacion Sicológica	1	466		Tipo de deducible por Evaluación Sicológica S- sin deducible C- por cada consulta E- Beneficio excluido X- Beneficio copado
Importe	11	467		Importe deducible por Evaluación Sicológica a cobrar.
Coaseguro	5	478		Porcent. coaseg. por Evaluación Sicológica
Tipo Deducible Gastos Oncológicos.	1	483		Tipo de deducible por Gastos Oncológicos S- sin deducible C- por cada consulta E- Beneficio excluido X- Beneficio copado
Importe	11	484		Importe deducible por Gastos Oncológicos a cobrar.
Coaseguro	5	495		Porcent. coaseg. por Gastos Oncológicos
Tipo Deducible Control niño	1	500		Tipo de deducible por Control de Niño Sano S- sin deducible C- por cada consulta E- Beneficio excluido X- Beneficio copado
Importe	11	501		Importe deducible por Control de Niño Sano.
Coaseguro	5	512		Porcent. coaseg. por Control de Niño Sano
Tipo Deducible Chequeo Médico	1	517		Tipo de deducible por Chequeo Médico S- sin deducible C- por cada consulta E- Beneficio excluido X- Beneficio copado
Importe	11	518		Importe deducible por Chequeo Médico
Coaseguro	5	529		Porcent. coaseg. por Chequeo Médico
Flag Condiciones Especiales Poliza	1	534		Activa Tx 1049
Cant Condiciones	2	535		
Porc Coaseguro Gastos Oncologicos	5	537		
Flag Exclusiones - Cl. Exclusivas	1	542		
RUC Contratante	11	543		
Tipo Poliza	1	554		

Este campo solo es usado para consulta de EPS e Invoca la Tx1040

**TOTAL 554**

**Trama de envio - Consulta por Poliza - 1011**

	Longitud	Posicion	
Posicion	1	8	El primer pedido va como posicion 0
Poliza	7	1	
Certificado	7	8	
Codigo Clinica	6	15	000011
<b>TOTAL</b>	<b>21</b>		

Buscar en Tabla de Relacion Poliza - Clinica-Categoria  
Luego con Categoria usar la Tabla de Deducibles correspondiente

**Trama Recibida**

	Bytes	Posicion
Resultado Consulta	1	1
Codigo de Error	3	2
Apellido Paterno+Materno+Nombres	45	5
Poliza	7	50
certificado	7	57
Codigo Parentesco	3	64

0=OK 1=Error  
Codigo de Mensaje 099=No existe

62 Bytes Por # Reg Devueltos

**TOTAL 5 + 61 \* # Reg**

Se envian Max. 15 registros - Forzar a ingresar datos para no recibir mas de 15

**Trama de envio - Consulta por Nombre - 1012**

	Bytes	Posicion	
Posicion	1	7	El primer pedido va como posicion 0
CodClinica	6	1	000011
Apellido Paterno	30	8	
Apellido Materno	30	38	
Nombres	30	68	

**TOTAL 97**

**Trama Recibida**

	Bytes	Posicion
Resultado Consulta	1	1
Codigo de Error	3	2
Mas Registros	1	5
Apellido Paterno+Materno+Nombres	45	6
Poliza	7	51
certificado	7	58
Codigo Parentesco	3	65

0=OK 1=Error  
Codigo de Mensaje 099=No existe  
0= No hay mas ; 1=Hay mas Registros

62 Bytes Por # Reg Devueltos

**TOTAL 5 + 62 \* # Reg**

Max. 15 Por lote

**Trama de envio - ConsultaOtros Beneficios - 1035**

	Longitud	Posicion	
Poliza	7	1	
Plan	3	8	
Inicio Vigencia	6	11	
Codigo Clinica	6	17	000011
<b>TOTAL</b>	<b>22</b>		

AAMMDD  
Buscar en Tabla de Relacion Poliza - Clinica Categoria  
Luego con Categoria usar la Tabla de Deducibles correspondiente

**Trama Recibida**

Longitud Posicion

Resultado Consulta	1	1
Codigo de Error	3	2
Código Beneficio	3	5
Tipo Deducible	1	8
Importe	11	9
Coaseguro	5	20

0=OK 1=Error

Código de Mensaje 099=No existe

Código de Beneficio ( descripción enviada aparte)
Tipo de deducible del Beneficio
S- sin deducible C- por cada consulta E- Beneficio excluido X- Beneficio copado
Importe del deducible para el beneficio
Porcentaje de Coaseguro de Beneficio Inicial

**TOTAL 24 20\*n**

**Trama de envío - Consulta Exclusiones - 1040**

	Longitud	Posicion
Poliza	7	1
Certificado	7	8
Código Parentesco	3	15
Código Clínica	6	18
Plan	3	24

**TOTAL 26**

**Trama Recibida**

	Longitud	Posicion
Resultado Consulta	1	1
Código de Error	3	2
IXTEXC1	55	5
IXTEXC2	55	60
IXTEXC3	55	115
IXTEXC4	55	170

**TOTAL 224**

0=OK 1=Error

Código de Mensaje 099=No existe

Texto 1 de la cláusula de exclusión del Asegurado

Texto 2 de la cláusula de exclusión del Asegurado

Texto 3 de la cláusula de exclusión del Asegurado

Texto 4 de la cláusula de exclusión del Asegurado

**Trama de envío - Consulta Observaciones - 1045**

	Longitud	Posicion
Poliza	7	1
Certificado	7	8
Código Parentesco	3	15

**TOTAL 17**

**Trama Recibida**

	Longitud	Posicion
Resultado Consulta	1	1
Código de Error	3	2
IXTEXC1	70	5
IXTEXC2	70	75
IXTEXC3	70	145
IXTEXC4	70	215

**TOTAL 284**

0=OK 1=Error

Código de Mensaje 099=No existe

Texto 1 de la cláusula de observaciones / preexistencias del Asegurado

Texto 2 de la cláusula de observaciones / preexistencias del Asegurado

Texto 3 de la cláusula de observaciones / preexistencias del Asegurado

Texto 4 de la cláusula de observaciones / preexistencias del Asegurado

**Trama de envío - Con.Cond Especiales Certif-1048**

	Longitud	Posicion
Poliza	7	1
Certificado	7	8

**TOTAL 14**

**Trama Recibida**

Longitud Posicion





Resultado Consulta	1	1	0=OK 1=Error
Codigo de Error	3	2	Codigo de Mensaje 099=No existe
Texto	55	5	0
<b>TOTAL</b>	<b>59</b>	<b>55 * n</b>	0

<b>Trama de envio - Con.Cond Especiales Poliza -1049</b>		
	Longitud	Posicion
Poliza	7	1
Código del Plan	3	8
Día Inicio de Vigencia del Plan	2	11
Mes Inicio de Vigencia del Plan	2	13
Año Inicio de Vigencia del Plan	2	15
Cod Clínica	6	17
<b>TOTAL</b>	<b>22</b>	

<b>Trama Recibida</b>		
	Longitud	Posicion
Resultado Consulta	1	1
Codigo de Error	3	2
Descripcion	55	5
<b>TOTAL</b>	<b>59</b>	<b>55 * n</b>

0=OK 1=Error  
Codigo de Mensaje 099=No existe  
0

Descripcion  
Texto variable de Condiciones Especiales del Certificado

0=OK 1=Error  
Codigo de Mensaje 099=No existe  
0

Descripcion  
Texto variable de Condiciones Especiales del Certificado

## **ANEXO C**

**TABLA C.1 PARÁMETROS DEL TEN CLIENTE**

**Aplicativo**

	<b>Aplicativo</b>
Pacífico Peruano Suiza	000096
Pacífico EPS	000097

**Cod. Transacción - Tipo de Transacción**

	<b>Cod.Tx</b>	<b>Tipo Tx</b>
Búsqueda por Código	1010	0420
Búsqueda por Grupo familiar	1011	0420
Búsqueda por Nombre	1012	0420
Listado de Beneficios	1035	0420
Listado de Exclusiones	1040	0420
Listado de Observaciones	1045	0420
Listado de Condiciones Especiales Certf.	1048	0420
Listado de Condiciones Especiales Póliza	1049	0420

**Cabecera de la Trama**

<b>Campo</b>	<b>Pos</b>	<b>Long</b>	<b>Valor</b>
Tipo de Mensaje	1	4	Determina la funcionalidad del mensaje
Tipo de Transacción	5	4	Determina la acción de cada transacción por tipo de mensaje.
Aplicativo	9	6	Identifica a la aplicación o programa que administra la base de datos correspondiente.
Cod.Transacción	15	4	Indica el requerimiento específico de la transacción. Tiene una única equivalencia en la aplicación que la atiende.
Fecha Actual	19	6	Fecha de transmisión del mensaje que contiene la transacción.
Hora Actual	25	6	Hora de transmisión del mensaje que contiene la transacción.
Número Secuencia	31	3	Número secuencial generado por cada terminal para identificar exclusivamente al mensaje enviado y su transacción asociada.
Identificador de Terminal	34	4	Identificador asociado al cliente que realiza una transacción en el sistema.
Código de Terminal	38	3	Código asignado por el Sistema a la computadora por donde un cliente se conecta a este.
Código TEN	41	6	Identificador del Servidor TEN, al cual será enviada la transacción.
Código Respuesta	47	3	Es el código de respuesta a la transacción enviada por el cliente al sistema. Un valor diferente a cero indica que la transacción ha sido rechazada, en caso contrario, la transacción ha sido realizada exitosamente.
Longitud de Data	50	3	Indica el número de bytes de data que se envían en el mensaje.
BitMap	53	16	Indica los campos del diccionario de datos correspondientes a la aplicación requerida.
Data	69	956	Es la información que viaja dentro del mensaje. El tamaño máximo que se puede consignar es de 956 bytes.
		1024	

**Errores de Comunicación**

	<b>Error</b>
Si Pos 47 Long 3 de la Cabecera de la Trama = 999	Tiempo de Espera Excedido
Si Pos 47 Long 3 de la Cabecera de la Trama = 100	Servidor No Activo
Si Pos 47 Long 3 de la Cabecera de la Trama <> 999 y <> 100	Verificar datos de configuración o de su línea de comunicación

**Errores del Aplicativo**

	<b>Error</b>
Si Cuerpo de Trama de Envío = Cuerpo de Trama de Recepción	Aplicativo no se Encuentra Registrado

**Errores en la Data**

	<b>Error</b>
<b>Tx1010</b>	
Si Pos 1 Long 1 + Pos 2 Long 3 del Cuerpo de la Trama = "1099"	Registro No Existe
Si Pos 1 Long 1 del Cuerpo de la Trama <> número o Pos 2 Long 3 del Cuerpo de la Trama <> número	Error al Enviar
Si Pos 1 Long 1 del Cuerpo de la Trama = "1"	El resultado de la consulta no es el esperado
<b>Tx1011/Tx1012</b>	
Si Pos 1 Long 1 + Pos 2 Long 3 del Cuerpo de la Trama = "1099"	Registro No Existe
Si Pos 1 Long 1 + Pos 2 Long 3 + Pos 5 Long 1 del Cuerpo de la Trama = "00000" Y Pos 6 Long 5 = " "	Registro No Existe
Si Pos 1 Long 1 del Cuerpo de la Trama <> número o Pos 2 Long 3 del Cuerpo de la Trama <> número	Error al Enviar
Si Pos 1 Long 1 del Cuerpo de la Trama = "1"	El resultado de la consulta no es el esperado
Si Pos 5 Long 1 = "1"	Por favor Ingrese mas Informacion, se encontraron demasiados registros (Es una advertencia no un error)
<b>Tx1035/Tx1040</b>	
Si Pos 1 Long 1 del Cuerpo de la Trama = "1"	Registro No Existe
<b>Tx1045/Tx1048/Tx1049</b>	
Si Pos 1 Long 1 + Pos 2 Long 3 del Cuerpo de la Trama = "1099"	Registro No Existe

## **ANEXO D**

# GUÍA DE PROGRAMACIÓN DEL TEN CLIENTE

## 1. API CLIENTE AS/400

### Funciones del API Cliente Transaccional

#### Send

Definición de Parámetros:

```
Declare Function SendFile Lib "dllcom.dll" (ByVal  
ArchivoLocal As String, ByVal ArchivoRemoto As String,  
ByVal Hostname As String, ByVal FtpPort As Integer) As  
Integer
```

Código Ejemplo

```
SendFile("c:\temp.txt","temp.txt",localhost, 4001)
```

#### GetFile

Definición de Parámetros:

```
Declare Function GetFile Lib "dllcom.dll" (ByVal  
ArchivoRemoto As String, ByVal ArchivoLocal As String,  
ByVal Hostname As String, ByVal FtpPort As Integer) As  
Integer
```

Código Ejemplo

```
GetFile("temp.txt","c:\temp.txt",localhost, 4001)
```

## 2. API CLIENTE WINDOWS/UNIX

### Administración:

Estructura de directorio y archivos generados.

## Directorios generados

Principal o de la Aplicación ( Program Files/...)  
Log / Tramas

## Archivos por directorio

Principal : Ejecutable y dlcom.dll  
Estructura del archivo log: HHMMSS + 1024  
(Ver Tabla 1)

## Configuración de la conexión con el TEN Server

Función EnviarServidor

Definición de Parámetros:

```
Declare Function EnviarServidor Lib "dlcom.dll"
(ByVal Hostname As String, ByVal Puerto As Integer)
As Integer
```

## **Funciones Generales:**

Initcomunicación : Establecer Conexión

```
Declare Function InitComunicacion Lib "dlcom.dll" ()
As Integer
```

GetBufferRcv : Capturar Respuesta del TEN Server

```
Declare Sub GetBufferRecv Lib "dlcom.dll" (ByVal
cadena As String)
```

SetBufferSend : Enviar consulta al Servidor TEN

```
Declare Sub SetBufferSend Lib "dlcom.dll" (ByVal
cadena As String)
```

## **Funciones API cliente FTP(file transfer)**

### **Send**

Definición de Parametros : Declare Function SendFile  
 Lib "dllcom.dll" (ByVal ArchivoLocal As String, ByVal  
 ArchivoRemoto As String, ByVal Hostname As String,  
 ByVal FtpPort As Integer) As Integer.

### **Código Ejemplo**

```
SendFile("c:\temp.txt","temp.txt",localhost, 4001)
```

### **GetFile**

Declare Function GetFile Lib "dllcom.dll" (ByVal  
 ArchivoRemoto As String, ByVal ArchivoLocal As  
 String, ByVal Hostname As String, ByVal FtpPort As  
 Integer) As Integer

### **Código Ejemplo**

```
GetFile("temp.txt","c:\temp.txt",localhost, 4001)
```

## **3. CÓDIGO DE EJEMPLO RUTINAS EN VISUALBASIC**

### **Paso a Paso.**

### **Aplicación de ejemplo " Enviar Trama "**

#### **Declaración de Funciones :**

```
Declare Function InitComunicacion Lib "dllcom.dll" () As Integer  

  Declare Function TerminarComunicacion Lib "dllcom.dll" () As  

  Integer
```



```
Declare Sub SetBufferSend Lib "dllcom.dll" (ByVal cadena As String)
```

```
Declare Function EnviarServidor Lib "dllcom.dll" (ByVal Hostname As String, ByVal Puerto As Integer) As Integer
```

```
Declare Sub GetBufferRecv Lib "dllcom.dll" (ByVal cadena As String)
```

### **CODIGO PARA ENVIAR CONSULTA AL TEN**

```
Private Sub CmdEnviar_Click()
```

```
    Dim Error As Long
```

### **Validación si existe dato en Trama:**

```
    If Trim(m_DataToSend) = "" Then
```

```
        MsgBox "Ingrese una Trama", vbInformation
```

```
        m_DataToSend.SetFocus
```

```
        Exit Sub
```

```
    End If
```

### **Captura datos para configurar conexion:**

```
    Screen.MousePointer = 11
```

```
    EscribirIni "Puerto", txtPuerto
```

```
' variable que almacena el puerto que usa el servidor Ten
```

```
    EscribirIni "Servidor", txtServidor
```

```
' variable que almacena el nombre del Servidor Ten
```

```
    EscribirIni "Transaccion", txtTransaccion
```

```
' variable que almacena el codigo de transaccion
```

```
    EscribirIni "Aplicativo", txtAplicativo
```

```
' variable que almacena el codigo de aplicativo
```

### **Descripcion General**

- 1- Se llama a la función de inicialización InitComunicacion, esta función debe ser llamada solo una vez al iniciar el programa
- 2- Se genera una Trama de 1024 bytes los 68 primeros bytes es la cabecera y 956 es data
- 3- se utiliza la funcion SetBufferSend para asignar la trama a la Dllcom
- 4- se utiliza la funcion EnviarServidor para enviar la trama al servidor Ten
- 5- se utiliza la funcion GetBufferRecv para recibir la trama del Servidor Ten

### **Funcion que Genera La Cabecera Para La Trama Del Ten (ver detalle al final)**

GenerarCabecera

'se asigna a la variable Trama la cabecera mas la data a enviar

Trama = m\_Cabecera.Text + m\_DataToSend.Text

' se asigna la trama a enviar a la dllcom

SetBufferSend (Trama)

' se envia la trama al servidor TenServer con Funcion

EnviarServidor

Error = EnviarServidor(txtServidor.Text, CInt(txtPuerto.Text))

' Validacion de error en comunicacion

If Error = 0 Then ' ocurrio error

```

' Se recibe la trama de la dllcom
  GetBufferRecv (Trama)

      If Mid(Trama, 47, 3) = "999" Then
          MsgBox "Time Out", vbOKOnly + vbInformation,
"TenCliente"
      Else
          MsgBox "Error al Enviar", vbOKOnly + vbInformation,
"TenCliente"
      End If
  Else
      ' se recibe la trama del servidor Ten
      GetBufferRecv (Trama)
      m_DataRecv = Trama
  End If
  Screen.MousePointer = 0
End Sub

```

### **Funcion Complementaria para Generar Cabecera Trama:**

```

Private Sub GenerarCabecera()

  Dim sCodMensaje As String
  Dim sTipoTrans As String
  Dim sTerminal As String
  Dim sNroTerminal As String
  Dim sClinica As String
  Dim sError As String
  Dim sLenData As String
  Dim sBitMap As String

```

```

sCodMensaje = "0300"
sTipoTrans = "0420"
sTerminal = "PC01"
sNroTerminal = "001"
sClinica = "SERV01"
sError = "000"
sLenData = LPad(CStr(Len(RTrim(m_DataToSend.Text))), 3, "0")
sBitMap = "000000000000000000"

m_Cabecera.Text = sCodMensaje & sTipoTrans &
txtAplicativo.Text & txtTransaccion.Text & _
                Format(Date, "mmddyy") & Format(Time,
"hhmmss") & "000" & _
                sTerminal & sNroTerminal & sClinica + sError +
sLenData + sBitMap
End Sub

```

```

sTramaSend = txtInput1.Text
sCadena = sTramaSend

```

```

Call SetBufferToSend(sTramaSend, "0420", "1011")
Error = EnviarServidor(m_NombreHost, CInt(m_Puerto))

```

```

If Error = 0 Then

```

```

    Call GetBufferRecv(sTramaRecv)

```

```

        If Mid(sTramaRecv, 47, 3) = "999" Then

```

```

            MsgBox "Time Out", vbOKOnly + vbInformation,
                "TenCliente"

```

```

        Else

```

```

    MsgBox "Error al Enviar", vbOKOnly + vbInformation,
    "TenCliente"
End If

```

```

Else
    Call GetBufferRecv(sTramaRecv)
    MsgBox sTramaRecv
End if

```

#### **4. FUNCIONES DE ENCRIPCIÓN DE ARCHIVOS**

Declaracion de Funciones

```

Declare Sub EncryptFile Lib "dllcom.dll" (ByVal ArchivoOrigen As String,
ByVal ArchivoDestino As String)

```

```

Declare Sub DesEncryptFile Lib "dllcom.dll" (ByVal ArchivoOrigen As
String, ByVal ArchivoDestino As String)

```

##### **Uso de la función:**

' la funcion EncriptaFile genera un nuevo archivo encriptado con el nombre del segundo parametro

```

EncryptFile("c:\prueba.txt","c:\Enc.txt");

```

' la funcion DesEncriptaFile genera un nuevo archivo desencriptado con el nombre del segundo parametro

```

DesEncryptFile("c:\Enc.txt",c:\prueba1.txt);

```

## **ANEXO E**

## ACRÓNIMOS

4GL	Fourth Generation Language. Lenguaje de Programación de 4ta Generación.
ACK	Acknowledgement. Reconocimiento.
ADSL	Asymmetric Digital Subscriber Line. Línea Digital Asimétrica de Suscriptor.
ANSI	American National Standard Institute. Instituto Estadounidense de Estándares Nacionales.
API	Application Programming Interface. Interfaz de Programación de Aplicaciones.
APP	Application. Aplicación.
AS/400	Advanced System /400. Sistema Avanzado/400.
AT&T	American Telephone & Telegraphs.
BD	Base de Datos.
CORBA	Common Object Request Broker Architecture. Arquitectura Común de Agente de Solicitud de Objetos.
DCE	Distributed Computing Environment. Ambiente de Computación Distribuido.
DEC	Digital Equipment Corporation.
DM	Data Management. Administración de los Datos.
DDBMS	Distributed Data Base Management System. Sistema Administrador de Bases de Datos Distribuidas.
GUI	Graphical user Interface. Interfaz Gráfica de Usuario.
HP	Hewlett Packard.
HTTP	Hyper Text Transfer Protocol. Protocolo de Transferencia de Hipertexto.
IBM	International Business Machines.
IEEE	Institute of Electrical and Electronic Engineers. Instituto de Ingenieros Electricistas y Electrónicos.
IM&M	Information Movement and Management. Movimiento y Gestión de la Información.
IP	Internet Protocol. Protocolo de Internet.
ISDN	Integrated Service Digital Network. Red Digital de Servicios Integrados.
ISO	International Standards Organization. Organización Internacional de Estándares.
KB	KiloByte. 1024 bytes.
LAN	Local Area Network. Red de Área Local.
M&Q	Message and Queueing. Mensajería y Encolamiento.
MB	MegaByte. 1'048,576 bytes.
MS	Microsoft.
MOM	Message Oriented Middleware. Middleware Orientado a Mensajes.
ODBC	Open Data Base Connectivity. Conectividad Abierta de Bases de Datos.
OLTP	On Line Transaction Processing. Procesamiento de Transacciones en Línea.
OMG	Object Management Group. Grupo de Gestión de Objetos.

OOL	Object Oriented Language. Lenguaje Orientado al Objeto.
ORB	Object Request Broker. Agente de Solicitud de Objetos.
OS	Operating System. Sistema Operativo.
OSF	Open Software Foundation. Fundación para el Software Abierto.
OSI	Open Systems Interconnection. Interconexión de Sistemas Abiertos.
OSS	Operations Support Systems. Sistemas de Soporte de Operaciones.
PC	Personal Computer. Computador Personal.
PPS	Pacífico Peruano Suiza Cía de Seguros.
PRE	Presentation. Presentación.
RAM	Random Access Memory. Memoria de Acceso Aleatorio.
RPC	Remote Procedure Call. Llamada a Procedimiento Remoto.
RPG	Report Program Generator. Programa Generador de Reportes.
SABD	Sistema Administrador de Base de Datos.
SCSI	Small Computer System Interface. Interfaz Pequeña para Sistemas de Computación.
SNA	System Network Architecture. Arquitectura de Red de Sistemas.
SOAP	Simple Object access Protocol. Protocolo Simple de Acceso a Objeto.
SQL	Structured Query Language. Lenguaje Estructurado de Consulta.
SWOT	Strengths, Weaknesses, Opportunities and Threats. Fortalezas, Debilidades, Oportunidades y Riesgos.
TCP	Transmission Control Protocol. Protocolo de Control de Transmisión.
TI	Texas Instruments.
TEN	Transactional Environment for Networking. Ambiente Transaccional para Interconexión.
T/I	Tecnologías de la Información
UDDI	Universal Description, Discovery and Integration. Descripción, Descubrimiento e Integración Universal.
WAN	Wide Area Network. Red de Área Ancha.
WSDL	Web Services Description Language. Lenguaje de Descripción de Servicios en Web.
WWW	World Wide Web. Sistema Mundial de Hipertexto.
X.25	Estándar internacional para redes de conmutación de paquetes.
XML	Extensible Markup Language. Lenguaje de Marcación Ampliado.



## BIBLIOGRAFÍA

1. Centro de Informática e Ingeniería de Sistemas en Administración, “Introducción a la Arquitectura Cliente/Servidor”, Centro de Informática e Ingeniería de Sistemas en Administración – Santiago, Chile, 1997.
2. Omnisciencia Consultores, “Propuesta Técnico-Económica para la Instalación del TEN”, Omnisciencia Consultores, 2000.
3. Novatronic, “Brochure de la Empresa”, Novatronic, 2000.
4. Mike Bray, “Middleware”, Carnegie Mellon University – Software Engineering Institute, 2000, en [http://www.sei.cmu.edu/str/descriptions/middleware\\_body.html](http://www.sei.cmu.edu/str/descriptions/middleware_body.html)
5. Cory Vondrak, “Message-Oriented Middleware”, Carnegie Mellon University – Software Engineering Institute, 2000, en [http://www.sei.cmu.edu/str/descriptions/momt\\_body.html](http://www.sei.cmu.edu/str/descriptions/momt_body.html)
6. Zoran Zaev, “Servicios Web XML”, Ediciones Anaya Multimedia – Madrid, España, 2002.
7. Microsoft Corporation, “Microsoft .NET Framework, Microsoft Corporation, 2002.
8. Bernard H. Boar, “Implementing Client/Server Computing”, McGraw-Hill, Inc., 1993.