

## Arquitectura de n capas

### **Programación por capas**

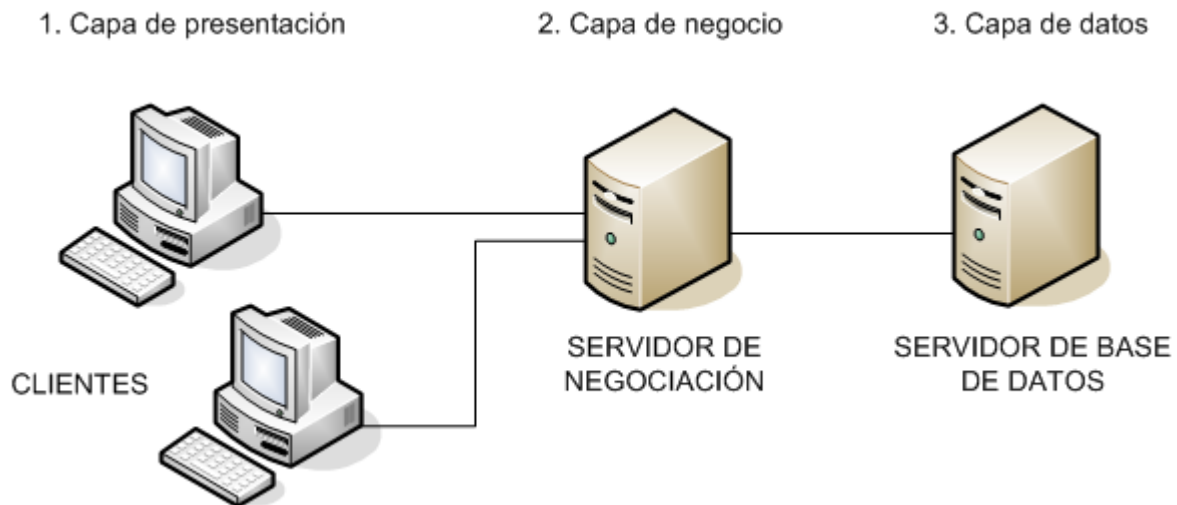
La programación por capas es una arquitectura cliente-servidor en el que el objetivo primordial es la separación de la lógica de negocios de la lógica de diseño; un ejemplo básico de esto consiste en separar la capa de datos de la capa de presentación al usuario.

La ventaja principal de este estilo es que el desarrollo se puede llevar a cabo en varios niveles y, en caso de que sobrevenga algún cambio, sólo se ataca al nivel requerido sin tener que revisar entre código mezclado. Un buen ejemplo de este método de programación sería el modelo de interconexión de sistemas abiertos.

Además, permite distribuir el trabajo de creación de una aplicación por niveles; de este modo, cada grupo de trabajo está totalmente abstraído del resto de niveles, de forma que basta con conocer la API que existe entre niveles.

En el diseño de sistemas informáticos actual se suelen usar las arquitecturas multinivel o Programación por capas. En dichas arquitecturas a cada nivel se le confía una misión simple, lo que permite el diseño de arquitecturas escalables (que pueden ampliarse con facilidad en caso de que las necesidades aumenten).

El diseño más utilizado actualmente es el diseño en tres niveles (o en tres capas).



### **Capas y niveles**

1. Capa de presentación: es la que ve el usuario (también se la denomina "capa de usuario"), presenta el sistema al usuario, le comunica la información y captura la información del usuario en un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). También es conocida como interfaz gráfica y debe tener la característica de ser "amigable" (entendible y fácil de usar) para el usuario. Esta capa se comunica únicamente con la capa de negocio.

2. Capa de negocio: es donde residen los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio) porque es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos almacenar o recuperar datos de él. También se consideran aquí los programas de aplicación.

3. Capa de datos: es donde residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

Todas estas capas pueden residir en un único computador, si bien lo más usual es que haya una multitud de computadoras en donde reside la capa de presentación (son los clientes de la arquitectura cliente/servidor). Las capas de negocio y de datos pueden residir en el mismo computador, y si el crecimiento de las necesidades lo aconseja se pueden separar en dos o más computadoras. Así, si el tamaño o complejidad de la base de datos aumenta, se puede separar en varias computadoras las cuales recibirán las peticiones del computador en que resida la capa de negocio. Si, por el contrario, fuese la complejidad en la capa de negocio lo que obligase a la separación, esta capa de negocio podría residir en uno o más computadores que realizarían solicitudes a una única base de datos. En sistemas muy complejos se llega a tener una serie de computadores sobre los cuales corre la capa de negocio, y otra serie de computadores sobre los cuales corre la base de datos.

### **Diferencia entre capas y niveles**

En una arquitectura de tres niveles, los términos "capas" y "niveles" no significan lo mismo ni son similares.

El término "capa" hace referencia a la forma como una solución es segmentada desde el punto de vista lógico.

Por ejemplo:

1. Presentación.
2. Lógica de Negocio.
3. Datos.

En cambio, el término "nivel" corresponde a la forma en que las capas lógicas se encuentran distribuidas de forma física.

Por ejemplo:

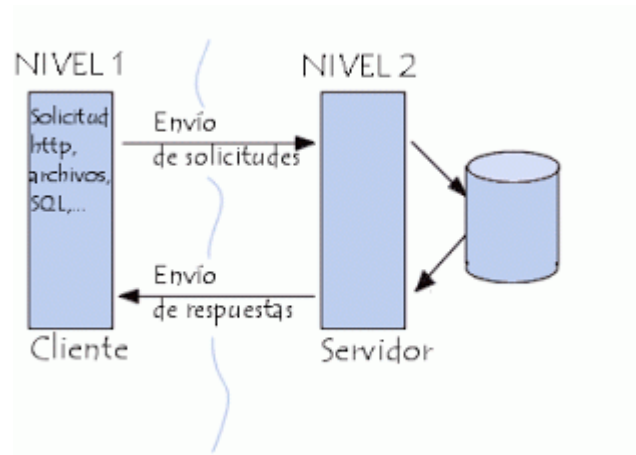
A. Una solución de tres capas (presentación, lógica del negocio, datos) que residen en una sola computadora (Presentación+lógica+datos). Se dice que la arquitectura de la solución es de *tres capas* y un nivel.

B. Una solución de tres capas (presentación, lógica del negocio, datos) que residen en dos

computadores (presentación+lógica por un lado; lógica+datos por el otro lado). Se dice que la arquitectura de la solución es de *tres capas y dos niveles*.

### Arquitectura de 2 niveles

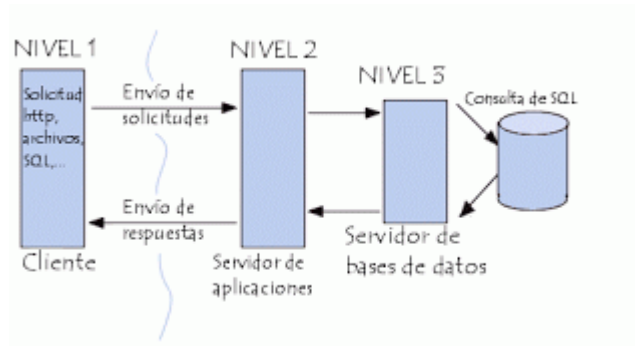
La arquitectura en 2 niveles se utiliza para describir los sistemas cliente/servidor en donde el cliente solicita recursos y el servidor responde directamente a la solicitud, con sus propios recursos. Esto significa que el servidor no requiere otra aplicación para proporcionar parte del servicio.



### Arquitectura en 3 niveles

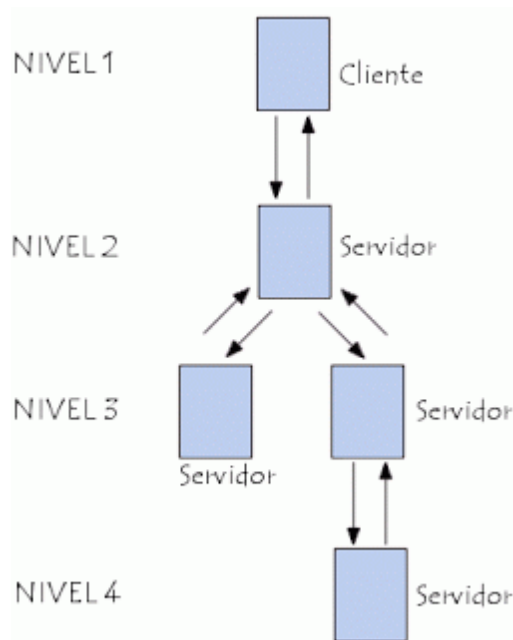
En la arquitectura en 3 niveles, existe un nivel intermediario. Esto significa que la arquitectura generalmente está compartida por: Un cliente, es decir, el equipo que solicita los recursos, equipado con una interfaz de usuario (generalmente un navegador Web) para la presentación. El servidor de aplicaciones (también denominado software intermedio), cuya tarea es proporcionar los recursos solicitados, pero que requiere de otro servidor para hacerlo. El servidor de datos, que proporciona al servidor de aplicaciones los datos que requiere.

Comparación entre ambos tipos de arquitecturas La arquitectura en 2 niveles es, por lo tanto, una arquitectura cliente/servidor en la que el servidor es polivalente, es decir, puede responder directamente a todas las solicitudes de recursos del cliente. Sin embargo, en la arquitectura en 3 niveles, las aplicaciones al nivel del servidor son descentralizadas de uno a otro, es decir, cada servidor se especializa en una determinada tarea, (por ejemplo: servidor web/servidor de bases de datos). La arquitectura en 3 niveles permite: Un mayor grado de flexibilidad Mayor seguridad, ya que la seguridad se puede definir independientemente para cada servicio y en cada nivel Mejor rendimiento, ya que las tareas se comparten entre servidores



### Arquitectura de niveles múltiples

En la arquitectura en 3 niveles, cada servidor (nivel 2 y 3) realiza una tarea especializada (un servicio). Por lo tanto, un servidor puede utilizar los servicios de otros servidores para proporcionar su propio servicio. Por consiguiente, la arquitectura en 3 niveles es potencialmente una arquitectura en N-niveles.



### Arquitectura de N-Capas y N-Niveles

**Lo que se conoce como arquitectura en capas es en realidad un estilo de programación donde el objetivo principal es separar los diferentes aspectos del desarrollo, tales como las cuestiones de presentación, lógica de negocio, mecanismos de almacenamiento, etc.**

Los que aprendieron a programar con Pascal, recuerdan que con ese lenguaje todo estaba en la misma porción de código.

A lo mejor un programador cuidadoso organizaba las cosas en "units", pero al final todo esta amontonado.



Figura 1

Vamos a dejar de lado los servicios que el Sistema Operativo facilita para el manejo de archivos, esto es imprescindible.

Muchos de nosotros debemos recordar que desde la aparición de los motores de base de datos existen dos "niveles" perfectamente definidos. Quiero resaltar el uso del término "nivel" y no el de "capa" porque no significan lo mismo. El término capa se utiliza para referenciar a las distintas "partes" en que una aplicación se divide desde un punto de vista lógico; mientras que "nivel" corresponde a la forma física en que se organiza una aplicación.

Cuando programaba en COBOL las cosas también se hacían como en Pascal, pero había instalaciones donde se podía utilizar una Base de Datos y codificar la aplicación con COBOL; de este modo ya teníamos una separación en niveles:

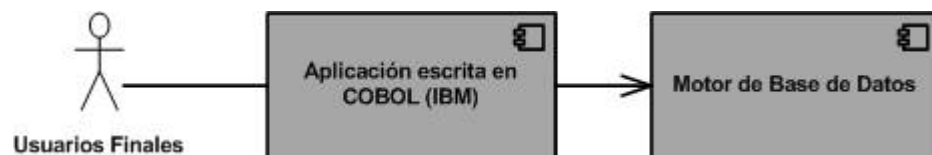


Figura 2

Se puede observar con total claridad el nivel de aplicación (seguramente en ella existe código de presentación y lógica de negocio) y el nivel de datos (donde está la o las bases de datos de la organización). El objetivo de este esquema fue y sigue siendo el de lograr un único repositorio de datos para la organización y múltiples aplicaciones que utilizan esos datos.

Ahora consideremos el ejemplo de un componente que permita grabar los programas de televisión emitidos. Este componente tiene un sistema de almacenamiento, dado que hace falta "guardar" las instrucciones sobre el día y hora se desea grabar el programa en particular; obviamente existe una porción de lógica de negocio, la que se refiere a los pasos necesarios para capturar el programa de televisión y grabarlo; finalmente hay una interfáz de usuario que permite a las personas ver y editar las instrucciones de grabación.

Podemos decir desde un punto de vista lógico que existen tres capas, y hasta que no veamos el código que escribió el desarrollador vamos a creer que es así.

Lo que no se puede negar es que hay un único nivel, todo esta empotrado en algún componente de hardware dentro del equipo.

Es importante destacar que hace falta separar el código de presentación del código de lógica de negocio, porque el fabricante puede desarrollar este equipo para que muestre las instrucciones en un display del mismo equipo o utilizar el televisor; la cuestión es que el código de presentación tiene que poder intercambiarse, lo cuál es una de las ventajas del desarrollo en capas.

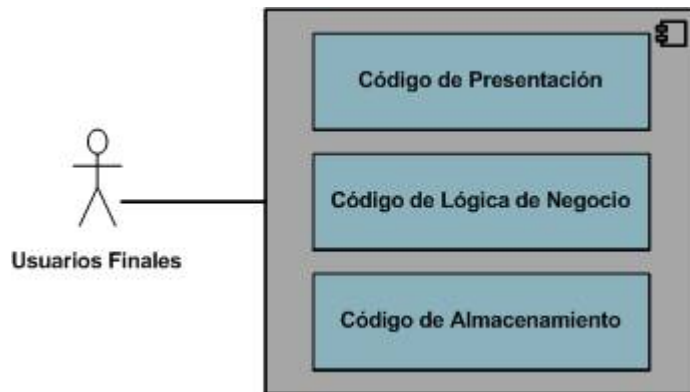


Figura 3

---

**La necesidad de contar con porciones de la aplicación que se puedan "intercambiar" sin tener que modificar el resto de la aplicación es lo que impulsa el desarrollo en capas; de este modo nos encontramos con el siguiente diagrama:**

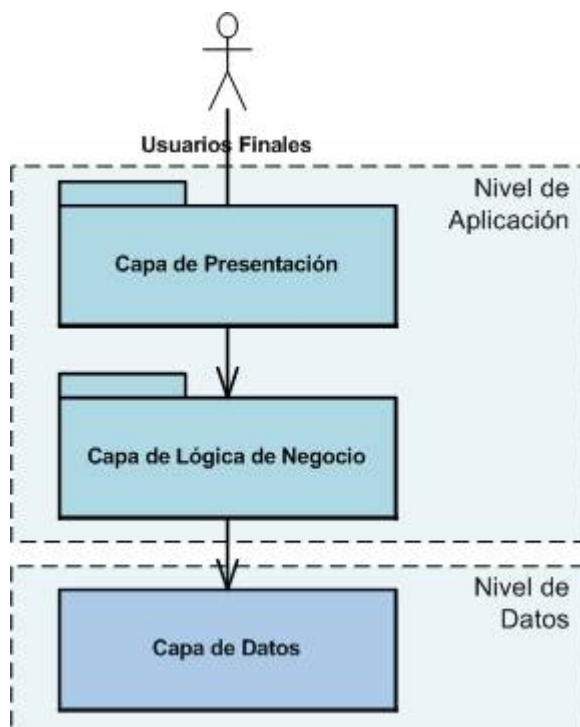


Figura 4

Ahora tenemos 2 niveles y en el primero de ellos diferenciamos 2 capas, de esta manera estamos diciendo que la capa de presentación interactúa con la capa de lógica de negocio; Desde la filosofía de arquitectura en capas, esto significa que **la capa de lógica de negocios presenta una "interfaz" para brindar servicios a la capa de presentación.**

Del mismo modo, en el diagrama estamos diciendo que existe otro nivel donde se encuentra una capa encargada de los datos. Esta capa no se muestra como un "paquete" o "ensamblado" dado que se trata (generalmente) de un motor de base de datos que puede o no ejecutarse en el mismo equipo. Indudablemente esta capa también presenta una "interfaz" para brindar sus servicios a la capa que está por encima.

Lo importante y que siempre debemos recordar es que las capas inferiores brindan servicios a las capas superiores (independientemente del nivel en que se encuentren).

**La clave del desarrollo en capas es que una capa solamente debe utilizar lo que la "interfaz" de la o las capas inferiores le brindan, de este modo se pueden intercambiar las capas respetando la "interfaz", que viene a ser como un "contrato**



entre capas"

Ahora escribiendo nuevo código podemos intercambiar la capa de presentación sin afectar el resto de la aplicación. El problema se presenta cuando queremos intercambiar la Capa / Nivel de Datos, esto significa que necesitamos utilizar otro motor de base de datos y resulta que los fabricantes de motores de bases de datos si bien respetan los estándares, incorporan características valiosas en sus productos. Una aplicación que pretenda utilizar la "potencia" o características particulares de un motor de base de datos, inevitablemente incorporará en la Capa de Lógica de Negocios algo de código que no es compatible con otros motores de base de datos. En consecuencia, cambiar la capa de datos significa corregir la capa de lógica de negocios.

Las buenas prácticas de diseño y desarrollo indican que se debe trabajar sobre el siguiente diagrama:

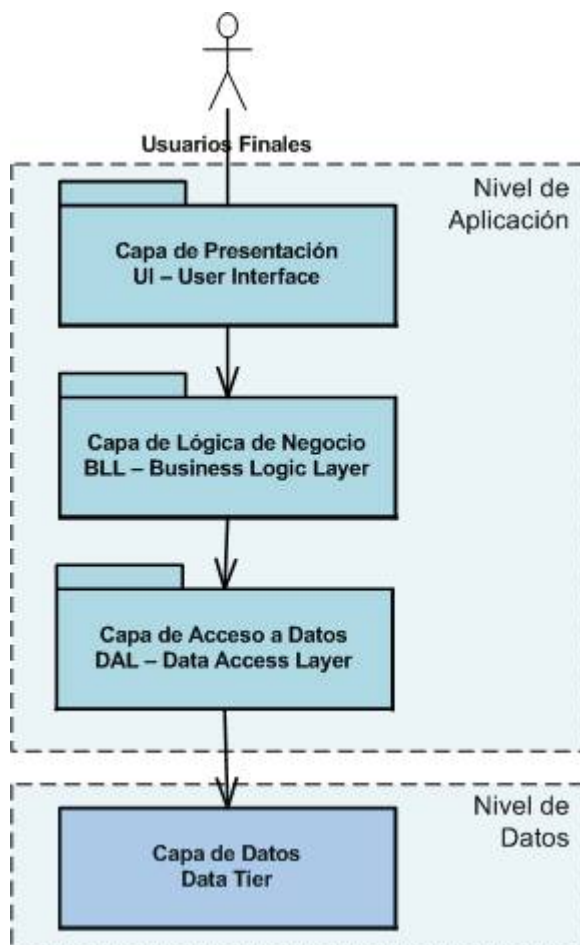


Figura 5

Ahora si tenemos las tan famosas **3 capas**, pero hay que hacer un par de aclaraciones para que nadie se confunda.

**La nueva capa, se denomina Capa de Acceso a Datos (o Capa de Persistencia) que no es lo mismo que Capa de Datos.**

**La capa de acceso a datos es una porción de código que justamente realiza el acceso a los datos. De esta manera cuando es necesario cambiar el motor de base de datos, solamente tendremos que corregir esa capa.**

**Mientras que la capa de datos (en el nivel de datos) es donde están los datos y se corresponde directamente con la definición de esquemas,**

***tablas, vistas,  
procedimientos  
almacenados y todo lo  
que se pueda o deba  
poner en un motor de  
base de datos.***

Los arquitectos están felices con este diagrama, sin embargo los desarrolladores tienen un problema. Resulta que los componentes de la capa de lógica de negocios necesitan referenciar a instancias de las "clases del dominio" (las que representan las entidades del negocio) y los componentes de la capa de acceso a datos también tienen que referenciarlas para poder "rellenar" tales instancias con los datos que obtienen de las capas inferiores.

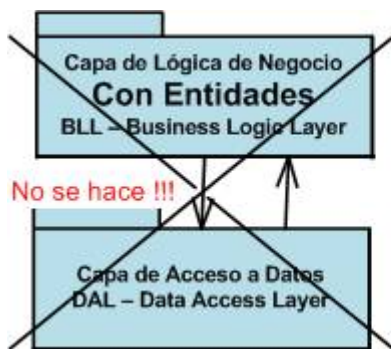


Figura 6 (a)

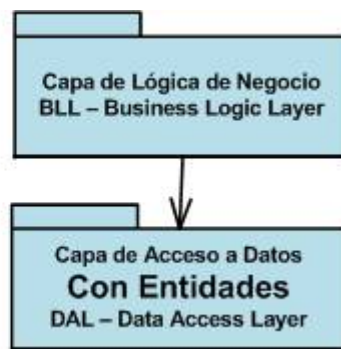


Figura 6 (b)

La porción de diagrama de la Figura 6 (a) muestra algo que no se debe hacer, porque los componentes no pueden caer en un ciclo de referencias recursivo, no podría compilarse la aplicación ¿cuál de los dos ensamblados debe resolverse antes para poder compilar el otro?

Una posible solución se presenta en la Figura 6 (b), donde se

muestra que la declaración de las "Entidades" se realiza en la capa de acceso a datos.

Bién esto también tiene un problema, la capa de acceso a datos surge porque no queremos "tocar" la lógica de negocio cuando cambiamos el nivel de datos, y con este esquema estamos incluyendo en la capa de acceso a datos uno de los aspectos más importantes, que es nuestra definición del dominio de la aplicación; los cambios en la capa de acceso a datos pueden impactar en las entidades.

Para pequeñas aplicaciones esto puede funcionar, incluso se puede poner ambas capas en un único ensamblado con diferentes

nombres de espacio (namespace) pero a la larga traerá problemas.

La solución que satisface a los arquitectos y a los desarrolladores, es la siguiente:

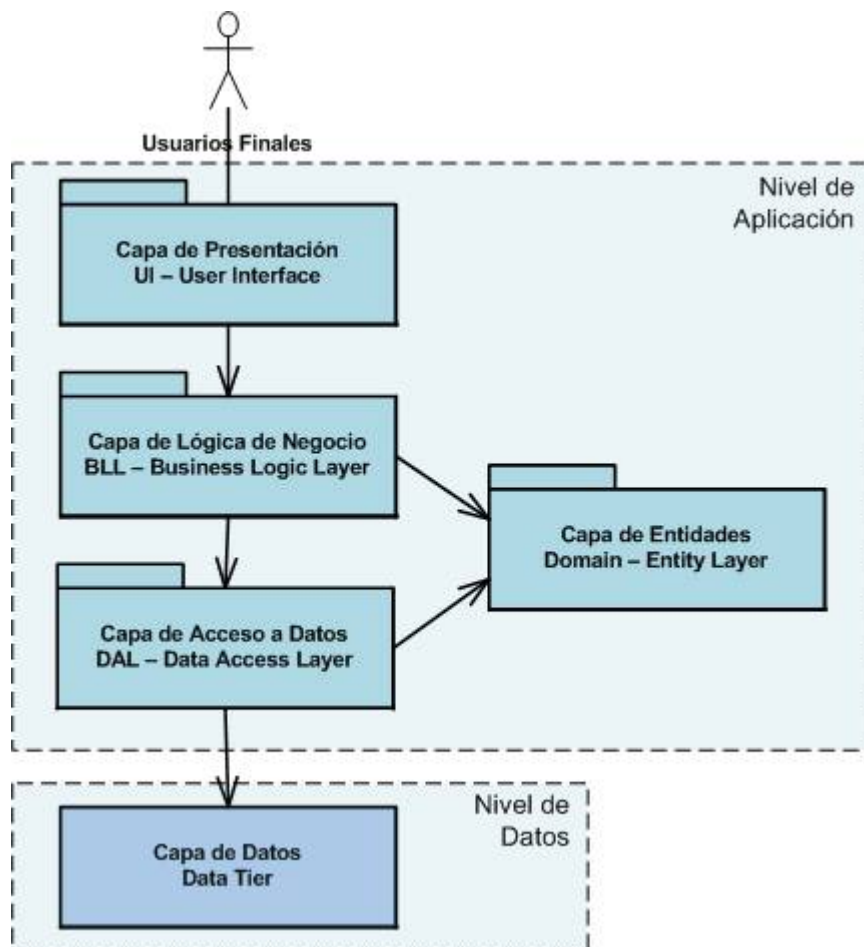


Figura 7

Ahora tenemos otra capa más, la capa de Entidades que corresponde al dominio de la aplicación.

En esta capa se encuentra la declaración de las entidades de la aplicación, de manera que se pueden referenciar desde otras capas sin entrar en ciclos recursivos de compilación.

Además este esquema permite una total independencia entre la lógica de negocios (Business Model) y las entidades (Domain

Model). Por supuesto que ambas partes están relacionadas por los casos de uso y otros requerimientos de la aplicación.

Por otro lado, este esquema facilita la incorporación, en la capa de acceso a datos, de componentes tipo ORM - Objet / Relational Mapping que permiten "mapear" (representar) objetos en un esquema relacional. Esto funciona bien dado que las bases de datos más utilizadas (por su mejor performance) son las bases de datos relacionales y no las orientadas a objetos (al menos por ahora).

---

El nivel de cooperación que presentan las organizaciones requiere que las aplicaciones de software de las distintas organizaciones interactúen unas con otras. El problema es que algunas de ellas no se ejecutan en la misma plataforma o están desarrolladas con marcos de trabajo diferentes. La solución fue presentada como SOA - Service Oriented Architecture, que brinda entre otras cosas una forma estandar de publicar y utilizar servicios, conocidos comunmente como servicios web (web services).

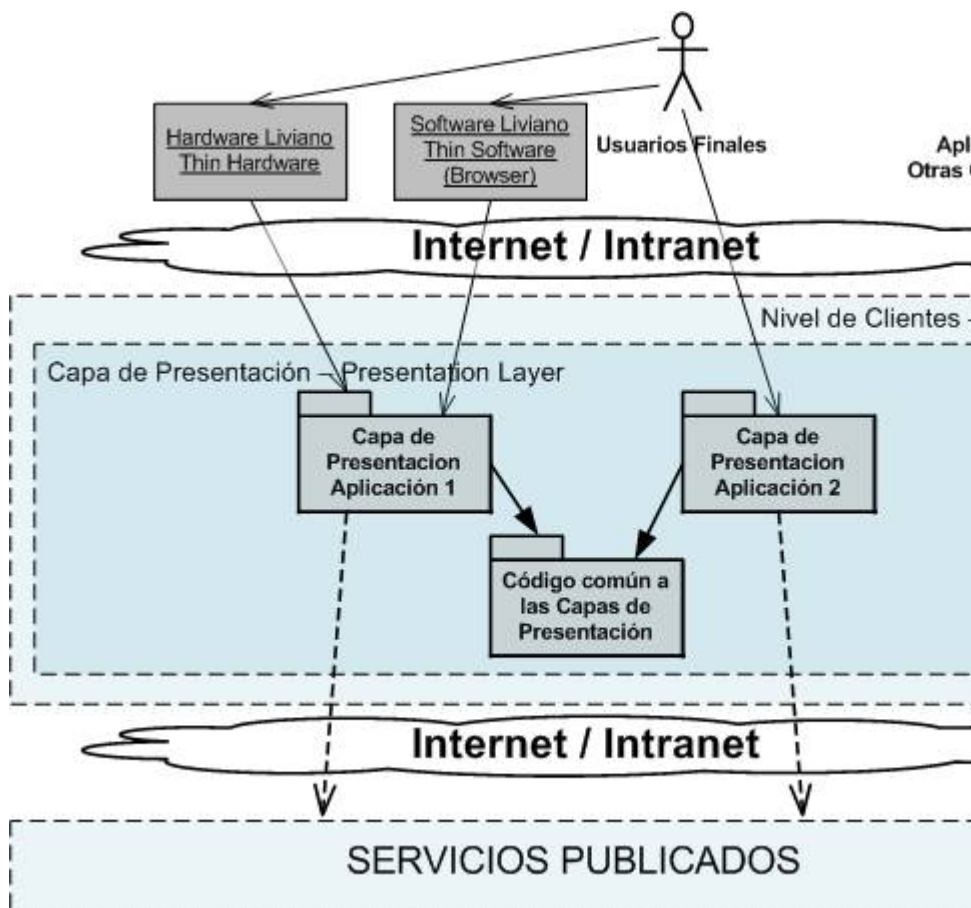


Figura 8

La Figura 8 muestra como los usuarios finales, mediante la utilización de hardware o software liviano, pueden acceder a lo que se denomina el nivel de clientes o aplicaciones que básicamente se constituyen de la capa de presentación y consumen los servicios publicados por la misma organización. Este tipo de aplicaciones son en general sitios o portales en la web.

Pueden implementar se soluciones del tipo cliente - servidor en donde el usuario final accede de manera directa a una aplicación de escritorio que consume los servicios publicados al igual que las aplicaciones para hardware y/o software liviano.

Aplicaciones de otras organizaciones también pueden utilizar los servicios publicados por una organización en particular, obviamente esto necesita de acuerdos comerciales y credenciales para

autenticar y  
autorizar a  
quienes  
consumen  
los  
servicios.

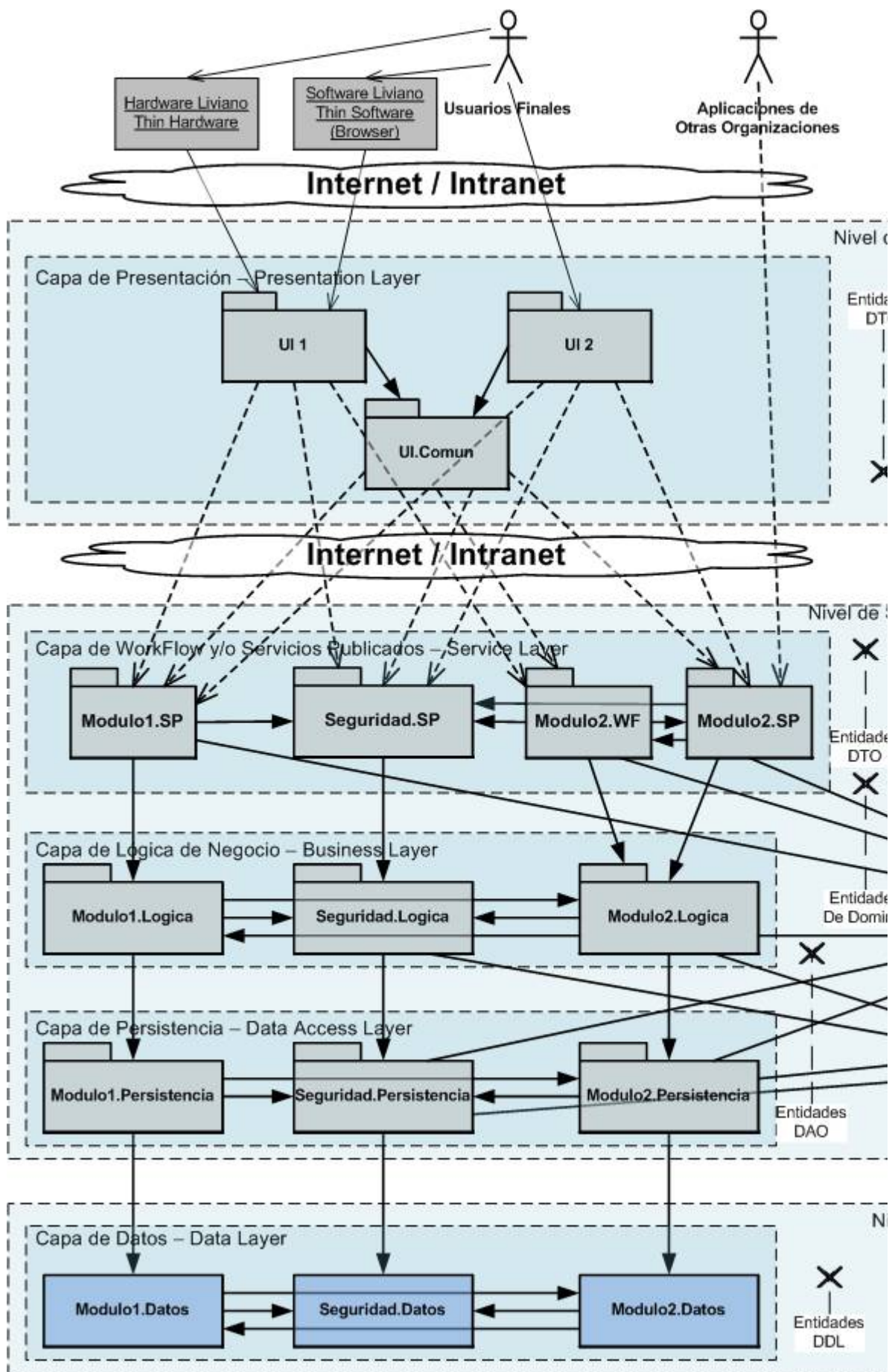
**La pregunta ahora es: ¿Los servicios publicados forman parte de la capa de lógica de negocio?**

En algunos casos, como por ejemplo recuperar la información de un inmueble podemos decir que el servicio en sí mismo no presenta lógica alguna.

En otros casos, como por ejemplo pagar la factura de un servicio público desde la cuenta bancaria del usuario final sí presenta un nivel de lógica que no tiene que ver con la lógica del negocio (sea de la institución pública o del banco). Se trata de la lógica de la transacción que involucra dos organizaciones diferentes (aplicaciones diferentes), consecuentemente debe existir un mecanismo que permita confirmar la realización de todos los "pasos" que cada una de las organizaciones requiere; esto normalmente se conoce como Workflow.

De esta manera surge otra capa más, la capa de Servicios Publicados que pueden o no incluir componentes de Workflow pero inevitablemente deben estar por arriba de la capa de lógica de negocios. La Figura 9 muestra el diagrama completo incluso con el componente esencial de la aplicación: el Esquema de Seguridad, así como la línea de vida de las entidades en relación a las distintas capas.





## Desarrollos de n-capas

El modelo n-tier (n-capas) de informática distribuida ha emergido como la arquitectura predominante para la construcción de aplicaciones multiplataforma en la mayor parte de las empresas pertenecientes a Fortune 1000. Este cambio radical en los modelos de computación, desde los sistemas monolíticos basados en mainframe y los tradicionales sistemas cliente-servidor, hacia sistemas distribuidos multiplataforma altamente modulables, representa simplemente la punta del iceberg de lo que está por llegar en el mundo del desarrollo de aplicaciones, tal y como se pone de manifiesto en las últimas tendencias de las grandes empresas de tecnología, como Sun con su estrategia Sun Tone, o Microsoft con DotNET (.Net).

### Ventajas del modelo

- Desarrollos paralelos (en cada capa)
- Aplicaciones más robustas debido al encapsulamiento
- Mantenimiento y soporte más sencillo (es más sencillo cambiar un componente que modificar una aplicación monolítica)
- Mayor flexibilidad (se pueden añadir nuevos módulos para dotar al sistema de nueva funcionalidad)
- Alta escalabilidad . La principal ventaja de una aplicación distribuida bien diseñada es su buen escalado, es decir, que puede manejar muchas peticiones con el mismo rendimiento simplemente añadiendo más hardware. El crecimiento es casi lineal y no es necesario añadir más código para conseguir esta escalabilidad.

Como tecnología, las arquitecturas de n-capas proporcionan una gran cantidad de beneficios para las empresas que necesitan soluciones flexibles y fiables para resolver complejos problemas inmersos en cambios constantes.

En Kynetia, como empresa de desarrollo especializada en plataformas distribuidas, podemos ayudarle identificando las tecnologías disponibles que mejor se adapten a su negocio y que mayores ventajas le puedan proporcionar. Trabajamos con todos los estándares del mercado relacionados con la computación distribuida, incluyendo CORBA, DNA (DCOM/COM+), EJB, XML, Java, Servidores de Aplicaciones, etc. Estas y otras tecnologías relacionadas con las arquitecturas en n-capas y las comunicaciones tendrán profundos efectos en el futuro, tanto en los negocios como en el mundo en el que vivimos.

La revolución del conocimiento es un evento global al que todas las compañías han de prestar mucha atención de cara a un futuro inmediato. Según vallamos avanzando en el nuevo milenio, las empresas, los países y las personas a lo largo de todo el mundo van a incrementar el desarrollo de su riqueza por medio de una comunicación global y una cooperación extensible a todos los niveles.

El crecimiento de la ciencia y la tecnología, combinado con el florecimiento de Internet y las economías basadas en mercados globales abiertos, contribuirán de forma conjunta a que la economía del Siglo XXI nos conduzca a una prosperidad sin precedentes.

Algo obvio en nuestros días es que el futuro de la informática no es el ordenador de escritorio, sino Internet. La arquitectura emergente basada en los estándares Internet es "Navegador/Red", significando que desde ahora, el foco estará en la informática basada en Red. Serán aplicaciones que únicamente existan en las redes y que estarán disponibles para cualquiera, en cualquier lugar y en cualquier momento.

Todas las aplicaciones basadas en n-capas permitirán trabajar con clientes ligeros, tal como navegadores de Internet, WebTV, Teléfonos Inteligentes, PDAs (Personal Digital Assistants o Asistentes Personales Digitales) y muchos otros dispositivos preparados para conectarse a Internet.

De este modo, las arquitecturas de n-capas se están posicionando rápidamente como la piedra angular de los desarrollos de aplicaciones empresariales y las compañías están adoptando esta estrategia a una velocidad de vértigo como mecanismo de posicionamiento en la economía emergente que tiene su base en la red (lo que se ha venido a denominar "Nueva Economía").

Actualmente, la Red (Internet, intranets y extranets) es el ordenador o, como diría Sun Microsystems, el ordenador es la Red. Este paradigma está creando un cambio fundamental en los modelos de computación que, a su vez, proporciona desafíos y oportunidades como nunca antes había se habían producido.

Las arquitecturas basadas en n-capas permiten a los componentes de negocio correr en una LAN, WAN o Internet. Esto significa que cualquiera con un ordenador y conexión a la Red (por ejemplo, un miembro del equipo de fuerza de ventas) posee toda la funcionalidad que tendría si se encontrase delante de su sistema de escritorio.

Realmente, los componentes distribuidos de una arquitectura de n-capas es una tecnología esencial para crear la siguiente generación de aplicaciones e-business, aplicaciones que son altamente escalables, fiables y que proporcionan un alto rendimiento y una integración sin fisuras con los sistemas de back-end heredados.

### **Next-Generation Technology**

Las arquitecturas empresariales de n-capas se están convirtiendo en la nueva base para el desarrollo de aplicaciones de misión crítica y ofrecen la única arquitectura funcional para la siguiente generación de soluciones informáticas distribuidas basadas en Internet. Los sistemas distribuidos de n-capas proporcionan un conjunto de avances tecnológicos sin precedentes, como pooling de conexiones, multiplexado de conexiones, balanceo de carga dinámico y rendimientos excelentes en hardware trabajando en cluster. Las aplicaciones industriales basadas en n-capas pueden ayudar a las compañías a desarrollar un nuevo núcleo de habilidades en prácticamente todo, desde la gestión del conocimiento hasta los sistemas relacionados con comercio electrónico.

La estrategia de desarrollo de aplicaciones para el siglo XXI

A diferencia de lo que se pudiera pensar, el desarrollo en n-capas no es un producto o un estándar, es un concepto estratégico que ayuda a la construcción y despliegue lógico de un sistema distribuido. Los sistemas de n-capas subdivididos ayudan a facilitar el desarrollo rápido de aplicaciones y su posterior despliegue, con beneficios incrementales fruto de los esfuerzos del desarrollo en paralelo coordinado y del outsourcing inteligente, resultando un enorme decremento del tiempo de desarrollo y de sus costes. Muchas de las aplicaciones de

e-business que se utilizan actualmente simplemente utilizan un navegador de Internet como cliente ligero que implementa una interfaz universal. Una arquitectura basada en clientes ligeros desplaza la capa de presentación de la aplicación en el lado del cliente, mientras que la lógica de negocio y los datos residen en el middleware y los servidores de back-end. El diseño para clientes ligeros minimiza los problemas de despliegue de las aplicaciones, mientras que maximiza la accesibilidad a la misma desde una amplia variedad de plataformas heterogéneas. Los frameworks basados en n-capas se crean para obtener las ventajas de los estándares abiertos de la industria que permiten a las aplicaciones resultantes operar en entornos distribuidos multiplataforma.

Utilizando estos potentes estándares abiertos se permite a los Integradores de Sistemas asegurar la efectividad y consistencia de la comunicación y mensajería a todos los niveles en aplicaciones cruzadas Interdepartamentales e Interempresariales. El diseño de aplicaciones basado en n-capas considera a la red como un pool de servicios distribuidos, un concepto mucho más ambicioso que el simple acceso de un cliente a un servidor. La separación de la presentación, lógica de negocio y datos es realizada en un número indefinido de capas lógicas, permitiendo a cada capa ser desarrollada, mejorada, gestionada y desplegada de forma independiente. Esta es precisamente la base para el modelo de informática de red en n-capas. Las plataformas multicapa funcionan consistentemente a lo largo de un variado conjunto de hardware, permitiendo escalar las operaciones del negocio desde un simple portátil, hasta un datacenter, desde el dispositivo más simple hasta el más complejo de los mainframes.

## **La evolución-revolución**

### **La evolución**

Las arquitecturas basadas en n-capas son el siguiente paso lógico en un proceso de evolución, el cuál, está basado en las arquitecturas convencionales cliente-servidor (2 y 3 capas) más la convergencia de dos tecnologías tremendamente potentes:

- Desarrollo de aplicaciones basadas en componentes - relacionado directamente con la Programación Orientada a Objetos (Lenguajes y Técnicas)
- Internet - primer ejemplo de un sistema complejo de n-capas cliente-servidor.

Los sistemas de n-capas utilizan técnicas de desarrollo basadas en componentes combinados con los estándares abiertos de Internet, para crear aplicaciones multiplataforma muy potentes con bajos costes, fáciles de mantener y con gran efectividad. Lo que realmente es nuevo en el modelo de n-capas es la posibilidad de distribuir objetos independientes sobre el número de capas que sean necesarias y enlazarlas dinámicamente, cuando sea necesario, para proporcionar una flexibilidad ilimitada a la aplicación.

### **La revolución**

#### **Arquitectura en n-capas: Un sistema adoptivo**

N-Tier forma parte también de un revolucionario proceso, actualmente en desarrollo, basado en la aplicación de estas nuevas tecnologías (componentes y estándares de Internet). Estas tecnologías son los bloques para crear Software de Negocio y Sistemas de Información adaptables que ayuden a las empresas a integrar todos sus sistemas de Tecnologías de la

Información, así como las inversiones realizadas en éstos, mientras que obtienen una ventaja clara en el uso de Internet.

Las empresas exitosas del futuro serán aquellas que se adapten mejor a un mundo conectado. Los framework de n-capas utilizan herramientas basadas en Internet que proporcionan a los clientes la adopción de las últimas y más potentes tecnologías que proporcionarán claros avances competitivos. Las empresas hoy en día (no importa dónde estén, qué tamaño tengan o en qué industria se encuentren) deben ser capaces de implementar las últimas prácticas de negocio, ventas y estrategias de distribución, procesos de fabricación, logística de la cadena de suministro, etc. Por eso, los sistemas basados en n-capas ayudan rápidamente a cambiar los negocios para experimentar la compartición sin restricciones de datos a lo largo de aplicaciones o fuentes de datos en la empresa, incluyendo Enterprise Resource Planning (ERP), aplicaciones hechas a medida, empaquetadas, heredadas o bases de datos.

### **Desarrollo de aplicaciones basado en componentes**

El surgimiento de la tecnología de componentes distribuidos es la clave de las arquitecturas de n-capas. Estos sistemas de computación utilizan un número variable de componentes individuales que se comunican entre ellos utilizando estándares predefinidos y frameworks de comunicación como:

- CORBA - (Common Object Request Broker Architecture) del Object Management Group (OMG)
- DNA - (Distributed interNet Architecture) de Microsoft (incluye COM/DCOM y COM+ además de MTS, MSMQ, etc.)
- EJB - (Enterprise Java Beans) de Sun Microsystems
- XML - (eXtensible Markup Language) del World Wide Web Consortium (W3C)

Estas y otras tecnologías en rápida evolución proporcionan la infraestructura necesaria y la fontanería relacionada que permite a las compañías operar en un entorno complejo, multiplataforma y con capacidades de computación distribuida, tanto interna como externamente según se requiera en cada caso.

### **Software Adaptable. Creado para ser flexible**

El desarrollo de aplicaciones en n-capas es un proceso iterativo de división del problema en piezas manejables denominadas componentes. Estos componentes, o "Componentes de Negocio - Business Objects" son "modelos software" basados típicamente en la "vista" de un objeto real, evento o proceso de negocio. Los componentes software individuales pueden formar parte y adaptarse tanto de estructuras independientes como de sistemas colaborativos.

El diseño de aplicaciones en n-capas es ideal para la creación de sistemas adaptables, donde cada componente puede ser utilizado y reutilizado en nuevas combinaciones para satisfacer requisitos de negocio dinámicos. Esto permite a los desarrolladores y a las nuevas aplicaciones reutilizar componentes existentes que modelan lógica de negocio sobradamente probada. En un entorno tremendamente cambiante como el actual, utilizar aplicaciones basadas en diseños de n-capas posibilitan a las empresas ser más ágiles y adaptables en proporcionar valor a sus clientes. Los sistemas basados en n-casas tienen el potencial de reducir drásticamente tanto el time-to-market para las nuevas aplicaciones de negocio, como

el coste total de mantenimiento, adaptando estos complejos y caros sistemas a las siempre cambiantes necesidades empresariales.

### **Una transformación global**

En esta era de comunicaciones instantáneas a lo largo de todo el mundo, los sistemas informáticos y las nuevas tecnologías son mucho más que simples herramientas que soportan procesos de negocio existentes. Los ordenadores y las redes sobre las que trabajan los primeros han pasado a ser facilitadores clave para ayudar a las organizaciones a transformarse continuamente en un mundo muy dinámico.

La conectividad entre los usuarios por medio de redes locales, empresariales y globales está cambiando la forma en la que las organizaciones operan en todos los sectores, donde los trabajadores hacen su trabajo e incluso el tipo de trabajo que realizan.

Los sistemas basados en n-capas posibilitan un desplazamiento estratégico en el uso de Internet como el "Ordenador Global". Esta revolución global no sólo está cambiando la dirección de la informática empresarial, sino que también está cambiando la naturaliza de cómo hacemos negocios. Por ahora, las empresas más progresistas han reconocido lo inevitable y tienen estrategias claras para abrazar Internet, más allá de proporcionar un navegador y una conexión a la Red a sus empleados. Utilizando la potencia de la información de Internet, se puede crear, mejorar y mantener relaciones con todas las partes de las que depende un negocio para alcanzar el éxito.

### **Sistemas de Negocio Integrados**

Uno de los mayores cambios que está afrontando hoy en día el e-business es la integración con todas las inversiones realizadas en software hasta el momento. Los framework de n-capas ayudan a proveer una solución integral enlazando los procesos de negocio críticos a la vez que se conservan las inversiones existentes de la empresa en tecnologías de la información y recursos relacionados.

Un entorno de n-capas puede ayudar a crear enlaces dinámicos entre marketing y ventas de e-business y el resto de la infraestructura de negocio establecida en la empresa, un paso crítico en la implementación de una estrategia efectiva de e-business. Asimismo, los sistemas de n-capas proporcionan a las organizaciones la posibilidad de integración completa y automática con todos los procesos de negocio esenciales para realizar e-business. Las soluciones integradas de n-capas se pueden extender a lo largo de las aplicaciones de empresa para permitir transacciones y procesos de negocio cruzados entre compañías, mientras que proporcionan un servicio de alto valor a sus clientes.

### **N-Capas significa e-business**

Como se ha podido ver hasta este momento, n-capas no es una tecnología, sino una estrategia de uso de las tecnologías para crear un negocio a la vez que se obtiene todo el potencial de éste inherente a Internet.

La informática basada en n-capas no se refiere solamente al despliegue de clientes ligeros de bajo coste conectados a servidores de aplicaciones muy flexibles con balanceo de carga e integrados con bases de datos distribuidas existentes a lo largo de diferentes plataformas y localizaciones. En realidad tiene que ver con la aplicación de las tecnologías relacionadas con desarrollos en n-capas para mejorar el conocimiento de los negocios y proveer un servicio de

valor mediante la aplicación de esta avanzada tecnología como una solución para envolver oportunidades del mundo real. Para tener éxito en el futuro, una compañía debe hacer más que establecer simples canales de comercio electrónico. Las organizaciones de mañana tienen que reconfigurarse a sí mismas de forma continua, tanto interna como externamente, como un negocio electrónico camaleónico, creando rápidamente relaciones y maximizando el rendimiento de una "empresa extendida". Los sistemas de información de negocio serán cada vez más adaptables, permitiendo la modernización regular de las estrategias de e-business y los modelos de negocio.

### **Convergencia y nueva economía**

"Estamos viviendo un periodo histórico de cambios tecnológicos, desarrollado sobre la aplicación de tecnologías de información y comunicación. Este proceso es más diferente y rápido que cualquiera de los que hayamos visto hasta el momento. Tiene un gran potencial para la creación de riqueza, niveles de vida más altos y mejores servicios". (Alvin Toffler, 1980)

Internet y la World Wide Web está generando la mayor revolución en tecnologías de comunicación e información desde el desarrollo del ordenador y, quizá, desde la invención del teléfono. La economía del Siglo XXI se convertirá cada vez más en una economía de información basada en billones de transacciones electrónicas realizadas cada día. La convergencia del ordenador y las tecnologías relacionadas con la comunicación está teniendo profundos efectos en los negocios en todo el mundo. La aplicación de tecnologías de n-capas a las necesidades reales de negocio facultará a éstos para obtener ventaja real de los constantes cambios.

### **Ha llegado la hora**

La población de Internet a lo largo del mundo se ha estimado en más de 200 millones de usuarios en el año 2000, de acuerdo con el estudio realizado por IDC, prediciendo que la cantidad de usuarios conectados a la Red ascenderá a más de mil millones al final de la década actual.

En un futuro muy cercano, una gran variedad de dispositivos electrónicos con amplias capacidades estarán listos para operar con aplicaciones desarrolladas para la Web, asegurando la máxima velocidad y la simplicidad para los usuarios. El incremento de popularidad de dispositivos NetTV con interfaces muy amigables, aplicaciones, dispositivos de todo tipo conectados a Internet y los nada caros dispositivos de mano como los PDA y los teléfonos inteligentes promete acelerar el incremento de usuarios de forma exponencial.

Mediante la adopción a día de hoy de arquitecturas de aplicaciones basadas en n-capas se permitirá la integración, escalabilidad, enlace o reingeniería de los sistemas existentes para adaptarse continuamente a los constantes cambios en las necesidades de negocio y convertirse en una tarea mucho más manejable en el futuro.