

# Arquitectura de redes

## Introducción

Conviene tener presente que el intercambio de información entre ordenadores tiene toda una serie de implicaciones, entre las que se encuentran las siguientes:

- Aspectos eléctricos de los medios de transmisiones como (cables, conectores, señales, )
- Forma de agrupar la información en los paquetes (mensajes)
- Errores que hay que corregir
- Compartición de un único medio de transmisión
- Distinción entre unos ordenadores y otros
- Tipo de información a transmitir
- Manejo de información confidencial, etc.

Atacar todos aspectos de una manera global no es viable. Demasiadas cosas y demasiado diferentes entre sí. Es preferible tratarlas una a una y de forma aislada.

Para ello se desarrollaron modelos de comunicaciones que está formados por “entidades”: en cada “entidad” se lleva a cabo una tarea y la colaboración de todas las entidades proporciona la comunicación entre los equipos y usuarios.

¿Qué son esas entidades? ¿Cómo se organizan? ¿Tiene algo que ver esas entidades con el concepto de capa o nivel? ¿Qué relación existe entre nivel y protocolo?

Así mismo, durante las últimas dos décadas ha habido un enorme crecimiento en la cantidad y tamaño de las redes. Muchas de ellas sin embargo, se desarrollaron utilizando implementaciones de hardware y software diferentes. Como resultado, muchas de las redes eran incompatibles y se volvió muy difícil para las redes que utilizaban especificaciones distintas poder comunicarse entre sí. ¿Qué ocurrió a partir de los 80 con la aparición de Ethernet, UNIX y TCP/IP? ¿Quiénes son Ethernet, UNIX y TCP/IP?

El **software de red** es el conjunto de programas encargado de:

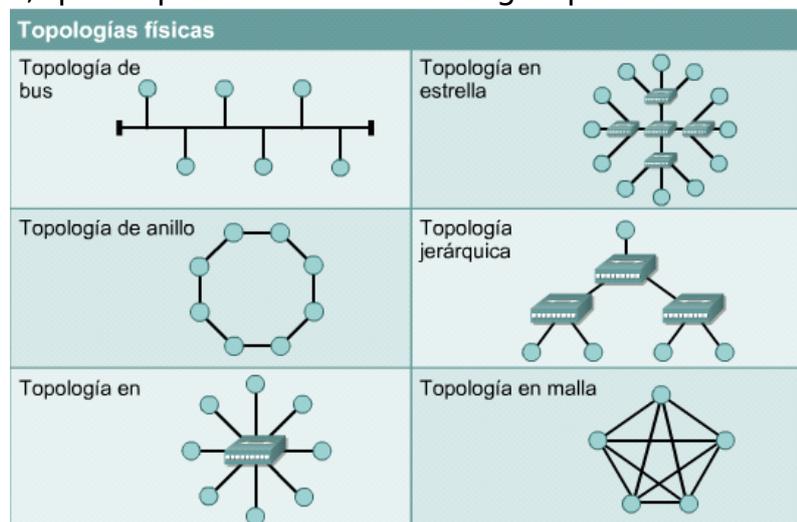
- Gestionar la red
- Controlar su uso
- Realizar detección y corrección de errores, etc.



Al igual que un sistema operativo realiza una gestión eficiente de los recursos de una máquina de cara a su utilización por los usuarios y las aplicaciones, el software de red realiza esta misma tarea de cara a los recursos físicos de la red (hardware de red).

## Características de la arquitectura de red

La arquitectura de una red viene definida por tres características fundamentales, que dependen de la tecnología que se utilice en su



construcción:

- **Topología:** la topología de una red es la organización de su cableado, ya que define la configuración básica de la interconexión de estaciones y, en algunos casos, el camino de una transmisión de datos sobre el cable.
- **Método de acceso a la red:** todas las redes que poseen un medio compartido para transmitir la información necesitan ponerse de acuerdo a la hora de enviar información, ya que no pueden hacerlo a la vez. En este caso, si dos estaciones transmiten a la vez en la misma frecuencia, la señal recogida en los receptores será una mezcla de las dos. Para las redes que no posean un medio compartido, el método de acceso al cable es trivial y no es necesario llevar a cabo ningún control para transmitir.

- **Protocolos de comunicaciones:** Son las normas y procedimientos utilizados en una red para realizar la comunicación. Esas normas tienen en cuenta el método utilizado para corregir errores, establecer una comunicación, velocidad de transmisión, tipo de información, formato de los mensajes, etc.

## Niveles de protocolos

Existen diferentes **niveles de protocolos** según el contexto en el que se aplique:

- Los protocolos de **alto nivel** definen cómo se comunican las aplicaciones (programas de ordenador)
- Los protocolos de **bajo nivel** definen cómo se transmiten las señales por el cable.
- Entre los protocolos de alto y bajo nivel, hay **protocolos intermedios** que realizan otras funciones, como establecer y mantener sesiones de comunicaciones y controlar las transmisiones para detectar errores.

Observe que los protocolos de bajo nivel son específicos del tipo de cableado utilizado para la red.

## Ejemplo de protocolo



Por ejemplo, podemos hacer la analogía con el sistema telefónico. En este caso, los servicios proporcionados pueden ser transmisión de voz, transmisión de datos, llamada en espera, llamada a tres, etc. Así mismo, el protocolo para establecer una comunicación (transmisión de voz) debe seguir estrictamente los siguientes pasos:

1. Descolgar el teléfono.
2. Comprobar si hay línea.
3. Si no hay, colgar y volver al paso 1.
4. Marcar número del otro usuario.

5. Esperar tono.
6. Si el tono es comunicando, colgar y volver al paso 1.
7. Si da mas de 6 tonos y no contesta, ir al paso 9.
8. Hablar cuando el otro usuario conteste.
9. Colgar.

Si no se siguen las reglas del protocolo estrictamente, la comunicación no se realizará en condiciones. Resulta absurdo que el usuario comience a hablar antes de tiempo porque la otra persona no oiría la conversación; así mismo si cuelga de forma precipitada, también se perderá una parte de la conversación.

Este es un ejemplo de protocolo al que estamos habituados. En comunicación de datos, los protocolos empleados son mas complejos porque deben ser capaces de corregir errores; en el caso de una comunicación normal, si el usuario no entiende, sólo tiene que decir ¿cómo dices? Sin embargo para ambos casos, la idea de base es la misma.

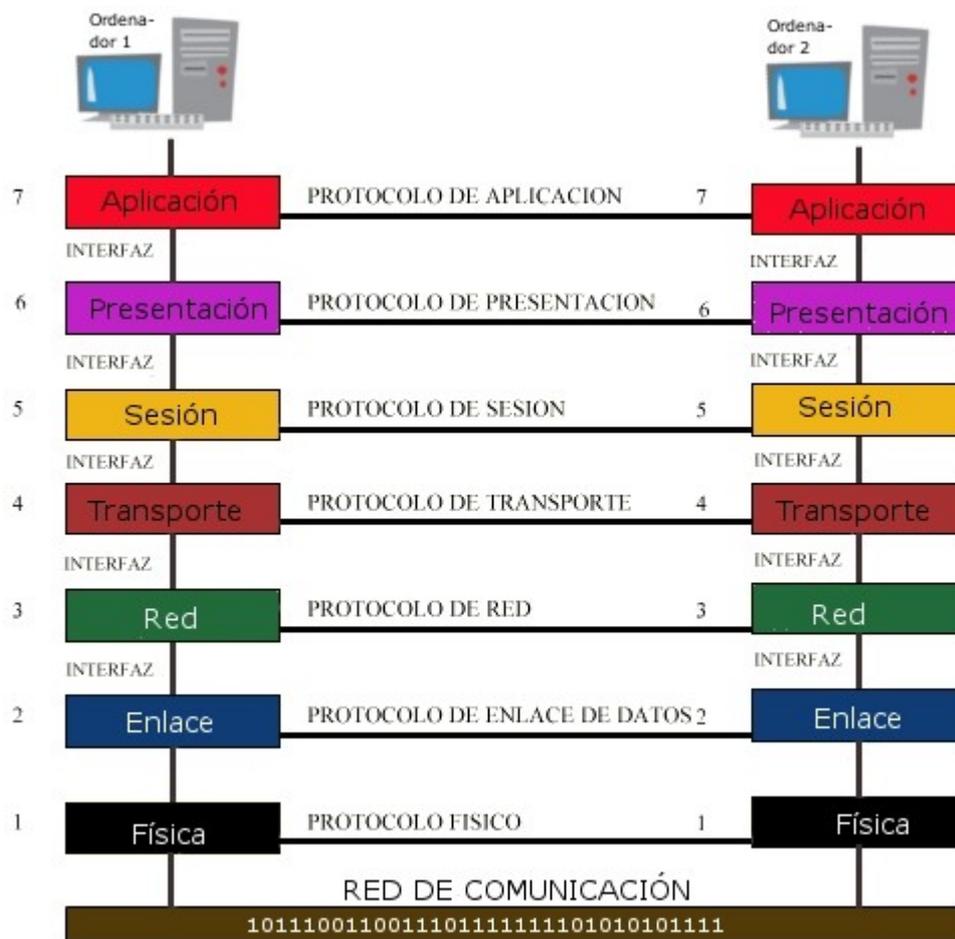
## **Características de las arquitecturas por niveles**

Como se ha podido comprobar en el apartado anterior, el diseño de un sistema de comunicación requiere de la resolución de muchos y complejos problemas.

Por este motivo, las redes se organizan en capas o niveles para reducir la complejidad de su diseño.

	Esta técnica se ha heredado de la metodología de programación consistente en dividir el problema en subproblemas más sencillos de tratar y en la programación modular (“ Divide y vencerás”).
--	---

Cada una de estas capas o subniveles (equivalente a un módulo) se construye sobre su predecesor (es decir, utiliza; los servicios o funciones diseñados en él) y cada nivel es responsable de ofrecer servicios a niveles superiores.



Dentro de cada nivel de la arquitectura coexisten diferentes servicios. En una jerarquía de protocolos se siguen las siguientes reglas:

- Cada nivel dispone de un **conjunto de servicios**.
- Los servicios están definidos mediante **protocolos** estándares.
- Cada nivel **se comunica** solamente con el nivel inmediato superior y con el inmediato inferior.
- Cada uno de los niveles inferiores **proporciona servicios** a su nivel superior.

Cuando se comunican dos ordenadores que utilizan la misma arquitectura de red, los protocolos que se encuentran al mismo nivel de la jerarquía deben coordinar el proceso de comunicación.

Por ejemplo, el nivel 2 de un equipo (transmitiendo, por ejemplo) coordina sus actividades con el nivel 2 del otro extremo (que se encargaría de recibir). Esto quiere decir que ambos deben ponerse de acuerdo y utilizar las mismas reglas de transmisión (es decir, el mismo protocolo).

En general, el **nivel n** de una máquina se comunica de forma indirecta con el **nivel n** homónimo de la otra máquina. Como se ha mencionado anteriormente, las reglas y convenciones usadas en esa comunicación se conocen como protocolo de nivel n. A los elementos activos de cada capa se les llama entidades o procesos y son éstos los que se comunican mediante el uso del protocolo. Al grupo formado por las entidades o procesos en máquinas diferentes que están al mismo nivel se llaman entidades o procesos pares.

# Encapsulación de los datos

El modelo de arquitectura por niveles necesita información adicional para que los procesos pares puedan comunicarse a un determinado nivel.

Estos datos adicionales dependen del protocolo utilizado y sólo se conoce su verdadero significado a ese nivel; normalmente, los niveles inferiores los tratan como si fuera información o datos a transmitir.

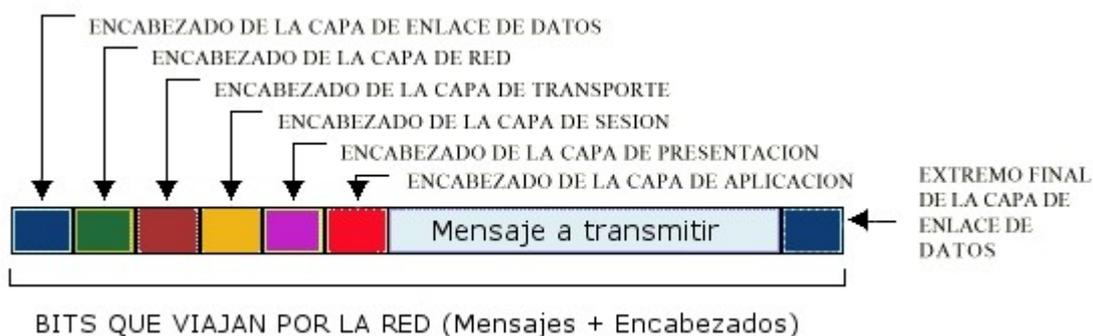
A ese añadido se le llama generalmente **cabecera** o **información de control** y suele ir al principio y/o al final del mensaje. Al proceso de añadir esta cabecera comúnmente se le conoce como **encapsulación**.

La forma que adopta una sección de datos en cualquier capa se denomina unidad de datos de protocolo (PDU - Protocol Data Unit).

Durante la comunicación, cada capa encapsula las PDU que recibe de la capa superior de acuerdo con el protocolo que se utiliza.

En cada etapa del proceso, una PDU tiene un nombre distinto para reflejar su nuevo aspecto. Aunque no existe una convención universal de nombres para las PDU, en este curso se denominan de acuerdo con los protocolos de la arquitectura TCP/IP:

- **Datos:** el término general para las PDU que se utilizan en la **capa de aplicación**.
- **Segmento:** PDU de la **capa de transporte**.
- **Paquete:** PDU de la capa de **Internetwork**.
- **Trama:** PDU de la **capa de enlace**.
- **Bits:** PDU que se utiliza cuando se transmiten físicamente datos a través de un **medio**.

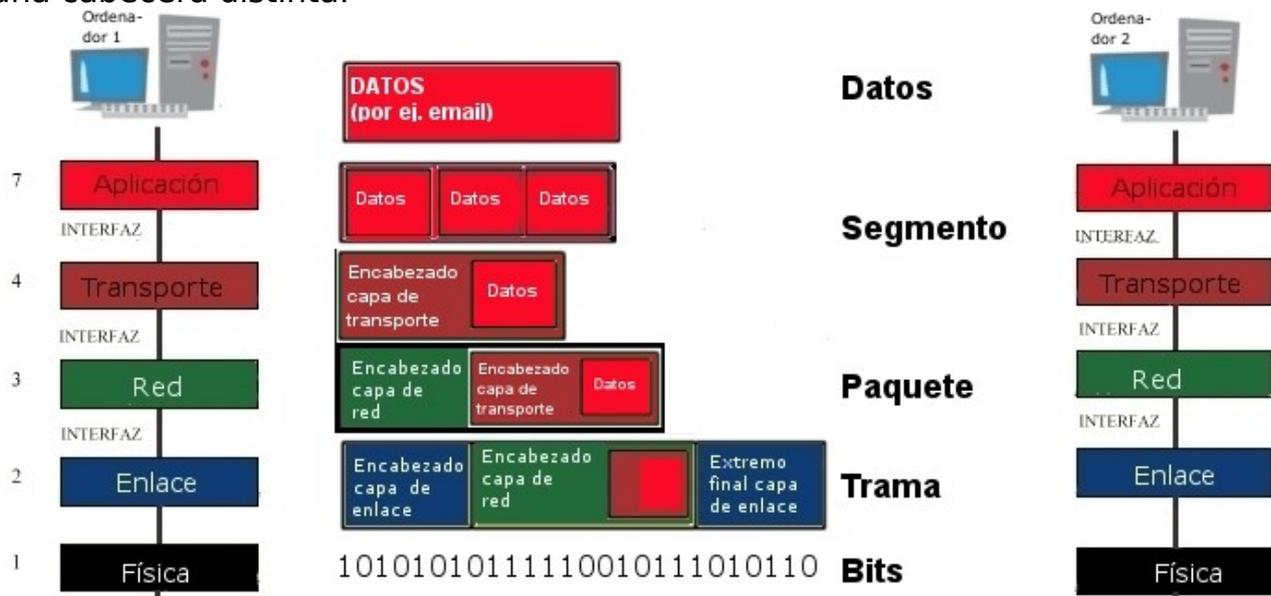


Cuando veamos ejemplos de arquitecturas de redes reales, comprobaremos que en todas sus capas se añaden **cabeceras de control** para la comunicación y todas ellas dependen del **protocolo** usado a ese **nivel**. Como ejemplo, una arquitectura de siete capas añade seis cabeceras de control para transmisión al mensaje original, como se muestra en la figura. La última capa no suele añadir información adicional ya que se encarga de enviar los dígitos binarios por el cable.

Podemos decir, por tanto, que el proceso de

**encapsulamiento** consiste en añadir un encabezado a los datos con la información que requiere cada protocolo a medida que los datos descienden por las capas de la arquitectura de protocolos. En algunos casos puede además de añadirse un encabezado se le añade un pie a los datos.

Aunque a primera vista parezca que la transmisión de un mensaje necesita el envío de gran cantidad de información de control (hay veces que se envía más cantidad que datos), este desperdicio no es muy superior al que sucede en una arquitectura diseñada sin niveles. La razón fundamental es que, como se ha dicho, cada capa se encarga de realizar una función diferente y necesita de una cabecera distinta.



A modo de ejemplo, para entender como es el proceso de comunicación entre dos ordenadores conectados en red vamos a utilizar como ejemplo una de las arquitecturas por niveles más extendida: la arquitectura/modelo TCP/IP.

Una aplicación (proceso 1) en el ordenador 1 desea enviar cierto mensaje (por ejemplo un email) a otra aplicación (proceso 2) del ordenador 2. El proceso de comunicación incluye los siguientes pasos:

1. El proceso 1 crea los datos a nivel de la **capa de aplicación** y realiza una **llamada al servicio** correspondiente en la capa de transporte y le entrega el mensaje
2. La **capa de transporte** añade una **cabecera** o encabezado (encapsulación) generando lo que se llama un segmento y lo entrega a la capa de red a través de la correspondiente solicitud de servicio. Puede ocurrir que debido al tamaño de los datos generados, la capa de transporte se vea obligada a partir el mensaje (segmentación).
3. La **capa de red** añade una **cabecera** o encabezado generando lo que se llama un **paquete** y lo entrega a la enlace solicitando el servicio asociado.
4. En la **capa de enlace** se repite el mismo proceso hasta llegar a la capa

física, con la particularidad de que en esta capa también se añade información de control en el **extremo final** del paquete que recibe de la capa superior.

5. La **Trama** es transmitida en forma de **bits** (señales eléctricas realmente), a través de la red, hasta el ordenador 2.
6. Cuando los datos llegan al **nivel físico** de la **máquina destino** (ordenador 2), se produce justamente el proceso contrario al anterior: cada capa recibe los datos, le quita y analiza su cabecera (**desencapsulación**) correspondiente y si todo es correcto los pasa al **nivel superior**. En el caso de que los datos se hayan segmentado, deberán ser **rearmados** por la capa de transporte.
7. Por último, la **capa de transporte** hace entrega de los datos al proceso 2 (ordenador 2).



Para entender en qué consiste el proceso de encapsulación, se puede asemejar a las muñecas rusas o matrioska. Su originalidad consiste en que se encuentran huecas por dentro, de tal manera que en su interior albergan una nueva muñeca, y ésta a su vez a otra, y ésta a su vez otra, en un número variable.

## El modelo de referencia OSI

El modelo **OSI** (Open Systems Interconnection o Interconexión de Sistemas Abiertos) está basado en una propuesta establecida en el año 1983 por la organización internacional de normas **ISO (ISO 7498)** como un avance hacia la **normalización** a nivel mundial de **protocolos**. El modelo se llama **modelo de referencia OSI de la ISO**, puesto que se ocupa de la conexión de sistemas abiertos, esto es, sistemas que están preparados para la comunicación con sistemas diferentes. Usualmente lo llamaremos sólo **modelo OSI** para acortar.

OSI emplea una **arquitectura en niveles** a fin de dividir los problemas de interconexión en partes manejables. Posteriores estándares de ISO definieron las implementaciones en cada nivel para asegurar que se consigue una

compatibilidad total entre ellos.

Como se vio en el apartado anterior, la aproximación en niveles asegura **modularidad** y facilita que el software pueda mejorarse sin necesidad de introducir cambios revolucionarios, además de permitir la compatibilidad entre equipos diferentes. Consta de **siete niveles**, mostrados en la tabla siguiente.

Pero, ¿cómo llegó la ISO, partiendo desde cero, a definir una arquitectura a siete niveles de esas características? Los principios teóricos en los que se basaron para la realización de OSI podemos resumirlos en los siguientes:

- Cada capa de la arquitectura está pensada para realizar **una función bien definida**.
- El **número de niveles** debe ser suficiente para que no se agrupen funciones distintas, pero no tan grande que haga la arquitectura inmanejable.
- Debe crearse una **nueva capa** siempre que se necesite realizar una función bien diferenciada del resto.
- Las **divisiones** en las capas deben establecerse de forma que se minimice el flujo de información entre ellas, es decir, que la **interfaz** sea más sencilla.
- Permitir que las **modificaciones de funciones o protocolos** que se realicen en una capa no afecten a los niveles contiguos.
- Utilizar la **experiencia de protocolos anteriores**. Las fronteras entre niveles deben situarse donde la experiencia ha demostrado que son convenientes.
- Cada nivel debe **interaccionar únicamente** con los **niveles contiguos** a él (es decir, el superior y el inferior).
- La función de cada capa se debe elegir pensando en la definición de protocolos estandarizados internacionalmente.

OSI está definido más bien como **modelo**, y no como arquitectura. La razón principal es que la ISO definió solamente la **función general** que debe realizar cada capa, pero no mencionó en absoluto los **servicios y protocolos** que se deben usar en cada una de ellas.

Las funciones encomendadas a cada una de las capas de OSI son las siguientes:

# ARQUITECTURA OSI



## Nivel físico

Tiene que ver con la transmisión de dígitos binarios por un canal de comunicación. Las consideraciones de diseño tienen que ver con el propósito de asegurarse de que, cuando un lado envíe un "1", se reciba en el otro lado como "1", no como "0". Las preguntas típicas aquí son: ¿qué voltaje deberá usarse para representar un 1 y para representar un 0?, ¿cuántos microsegundos dura cada dígito?, ¿en qué frecuencia de radio se va a transmitir?, ¿cuántas puntas tiene el conector de la red y para qué sirve cada una?, etc. Aquí las consideraciones de diseño tienen mucho que ver con las interfaces mecánica, eléctrica y de procedimientos y con el medio de transmisión físico que está bajo la capa física.

## Nivel de enlace de datos

Su tarea principal es detectar y corregir todos los errores que se produzcan en la línea de comunicación. También se encarga de controlar que un emisor rápido no sature a un receptor lento, ni se pierdan datos innecesariamente. Finalmente, en redes donde existe un único medio compartido por el que

circula la información, este nivel se encarga de repartir su utilización entre las estaciones. La unidad mínima de datos que se transfiere entre entidades pares a este nivel se llama trama o marco.

## **Nivel de red**

Se ocupa de determinar cuál es la mejor ruta por la que enviar la información. Esta decisión tiene que ver con el camino más corto, el más rápido, el que tenga menor tráfico, etc. Por todo esto, la capa de red debe controlar también la congestión de la red, intentando repartir la carga lo más equilibrada posible entre las distintas rutas. También a este nivel se realiza gran parte del trabajo de convertir y adaptar los mensajes que circulan entre redes heterogéneas. La unidad mínima de información que se transfiere a este nivel se llama paquete.

## **Nivel de transporte**

Es el nivel más bajo que tiene independencia total del tipo de red utilizada y su función básica es tomar los datos procedentes del nivel de sesión y pasarlos a la capa de red, asegurando que lleguen correctamente al nivel de sesión del otro extremo. A este nivel, la conexión es realmente de extremo a extremo ya que no se establece ninguna conversación con los niveles de transporte de todas las máquinas intermedias.

## **Nivel de sesión**

A este nivel se establecen, mantienen y finalizan las sesiones (conexiones) de comunicación entre los dos extremos para el transporte ordinario de datos. A diferencia del nivel de transporte, a este nivel se proporcionan algunos servicios mejorados, como la reanudación de la conversación después de un fallo en la red o una interrupción, etc.

## **Nivel de presentación**

A este nivel se controla el significado de la información que se transmite, lo que permite la traducción de los datos entre las estaciones. Por ejemplo, si una estación trabaja con un código concreto y la estación del otro extremo maneja uno diferente, el nivel de presentación es el encargado de realizar esta conversión. Para conversaciones confidenciales, este nivel también codifica y encripta los datos para hacerlos incomprensibles a posibles escuchas ilegales.

## **Nivel de aplicación**

Es el nivel que está en contacto directo con los programas o aplicaciones informáticas de las estaciones y contiene los servicios de comunicación más utilizados en las redes. Como ejemplos de servicios a este nivel se puede mencionar la transferencia de archivos, el correo electrónico, etc.

Este modelo no es perfecto y, de hecho, algunas cuestiones adolecen de un mal diseño. La más importante, en lo que se refiere a las capas, es que algunas de ellas están prácticamente vacías (es decir, hay muy pocos protocolos definidos dentro de éstas y a la vez son bastante triviales), mientras que otras están llenas a rebosar. Por ejemplo, las capas de sesión y presentación no se usan en la mayoría de las aplicaciones, mientras que las capas más inferiores

están tan saturadas que en revisiones posteriores se han dividido en múltiples subcapas.

Otro problema que tiene OSI es que existen algunas funciones que se repiten en muchas de las capas, lo que hace que muchos servicios y programas estén duplicados, a la vez que contribuye a un aumento del tamaño de las cabeceras de control de los bloques de información que se transmiten.

## Arquitectura TCP/IP



**TCP/IP** se suele confundir muchas veces con un protocolo de comunicaciones concreto, cuando, en realidad, es una compleja arquitectura de red que incluye varios de ellos, apilados por capas. Es, sin lugar a dudas, la más utilizada del mundo, ya que es la base de comunicación de Internet y también se utiliza ampliamente en las distintas versiones de los sistemas operativos Unix y Linux (aunque debido a su gran utilización ha sido también implantado en otros sistemas como Windows).

En el año 1973, el DoD (Departamento de Defensa de Estados Unidos) inició un programa de investigación para el desarrollo de tecnologías de comunicación de redes de transmisión de datos. El objetivo fundamental era desarrollar una red de comunicación que cumpliera las siguientes características:

- Permita interconectar **redes diferentes**. Esto quiere decir que la red en general puede estar formada por tramos que usan tecnología de transmisión diferente.
- Sea **tolerante a fallos**. El DoD deseaba una red que fuera capaz de soportar ataques terroristas o incluso alguna guerra nuclear sin perderse datos y manteniendo las comunicaciones establecidas.
- Permita el uso de **aplicaciones diferentes**: transferencia de archivos, comunicación en tiempo real, etc.



Todos estos objetivos implicaron el diseño de una red con **topología irregular** donde la información se **fragmentaba** (dividía en trozos) para seguir **rutas diferentes** hacia el destinatario. Si alguna de esas rutas fallaba

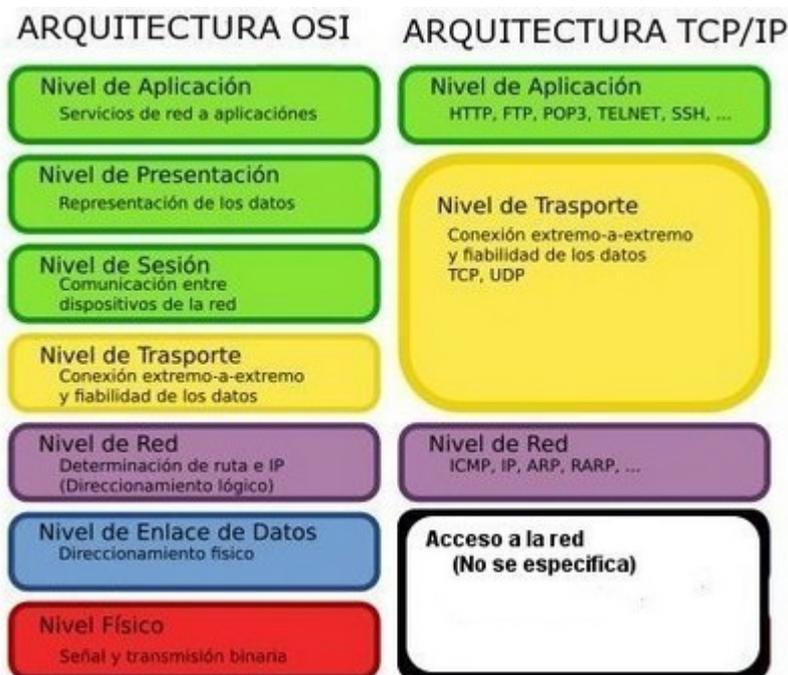
repentinamente, la información podría seguir **rutas alternativas**. Así, surgieron dos redes distintas: una dedicada a la investigación, ARPANET, y otra de uso exclusivamente militar, MILNET.

El DoD permitió a varias universidades que colaboraran en el proyecto, y ARPANET se expandió gracias a la interconexión de esas universidades e instalaciones del Gobierno. Este modelo se nombró después como **arquitectura TCP/IP**, por las iniciales de sus **dos protocolos más importantes**. En 1980, TCP/IP se incluyó en Unix 4.2 de Berkeley y fue el protocolo militar estándar en 1983. En ese mismo año nació la red global Internet, que utiliza también esta arquitectura de comunicación. ARPANET dejó de funcionar oficialmente en 1990.

Algunos de los motivos de la popularidad alcanzada por esta arquitectura son:

- Es **independiente de los fabricantes** y las marcas comerciales.
- Soporta **múltiples tecnologías** de redes.
- Es capaz de **interconectar redes** de diferentes tecnologías y fabricantes.
- Puede funcionar en máquinas de cualquier tamaño, desde ordenadores personales a grandes supercomputadores.
- Se ha convertido en estándar de comunicación en EEUU desde 1983.

La arquitectura de TCP/IP se construyó diseñando inicialmente los protocolos para, posteriormente, integrarlos por capas en la arquitectura. Por esta razón, a TCP/IP muchas veces se la califica como pila de protocolos. Su modelo por niveles es algo diferente al modelo OSI de ISO, como demuestra la tabla siguiente:



*Comparativa OSI - TCP/IP*

Obsérvese que TCP/IP sólo tiene definida 5 capas (si incluimos enlace y física) mientras que OSI tiene siete. Las funciones que realizan cada una de ellas son las siguientes:

## Capa de acceso a la red

El modelo no da mucha información de esta capa, y solamente se especifica que debe existir algún **protocolo** que conecte la estación con la red. La razón fundamental es que, como TCP/IP se diseñó para su funcionamiento sobre redes diferentes, esta capa **depende de la tecnología** utilizada y no se especifica de antemano. Piensa que una red puede estar interconectada por diferentes tipos de cables o de forma inalámbrica y en este nivel se definen los protocolos asociados a los dispositivos de bajo nivel de cada una de estas tecnologías.

## Capa de red o de internet

Esta capa es la más importante de la arquitectura y su misión consiste en permitir que las estaciones envíen información (paquetes) a la red y los hagan viajar de forma independiente hacia su **destino**. Durante ese viaje, los paquetes pueden **atravesar redes diferentes** y llegar **desordenados**. Esta capa no se responsabiliza de la tarea de ordenar de nuevo los mensajes en el destino. El protocolo más importante de esta capa se llama **IP** (Internet Protocol Protocolo de Interred), aunque también existen otros protocolos.

## Capa de transporte

ésta cumple la función de **establecer una conversación** entre el origen y el destino, de igual forma que hace la capa de transporte en el modelo OSI. Puesto que las capas inferiores no se responsabilizan del **control de errores** ni de la **ordenación de los mensajes**, ésta debe realizar todo ese trabajo. Aquí también se han definido varios protocolos, entre los que destacan **TCP** (Transmission Control Protocol Protocolo de Control de la Transmisión) y **UDP** (User Datagram Protocol Protocolo de Datagrama de Usuario)

## Capa de aplicación

Esta capa contiene, al igual que la capa de aplicación de OSI, todos los protocolos de alto nivel que utilizan los **programas** para comunicarse. Aquí se encuentra el protocolo de terminal virtual (TELNET), el de transferencia de archivos (FTP), el protocolo HTTP que usan los navegadores para recuperar páginas en la World Wide Web, los protocolos de gestión del correo electrónico, etc.

Las capas de **sesión y presentación no existen** en la arquitectura TCP/IP, ya que los diseñadores pensaron que no se necesitaban. La experiencia obtenida con los trabajos realizados en el modelo OSI ha comprobado que esta visión fue correcta ya que se utilizan muy poco en la mayoría de las aplicaciones de comunicación.