Conceptos de Sistemas Distribuidos

Desarrollo de Aplicaciones para la nube

Ulises Ancona Graniel

PROTOCOLO DE COMUNICACIÓN

- En esencia, un protocolo es un acuerdo entre las partes que se comunican para establecer la forma en que se llevará a cabo la comunicación entre una máquina con otra.

- Un protocolo es un conjunto de reglas y convenciones que rigen el formato y el significado de los paquetes o mensajes que intercambian las entidades iguales en una capa.

- Si se viola el protocolo se hará más difícil la comunicación, si no es que se vuelve imposible.

PROTOCOLO DE COMUNICACIÓN

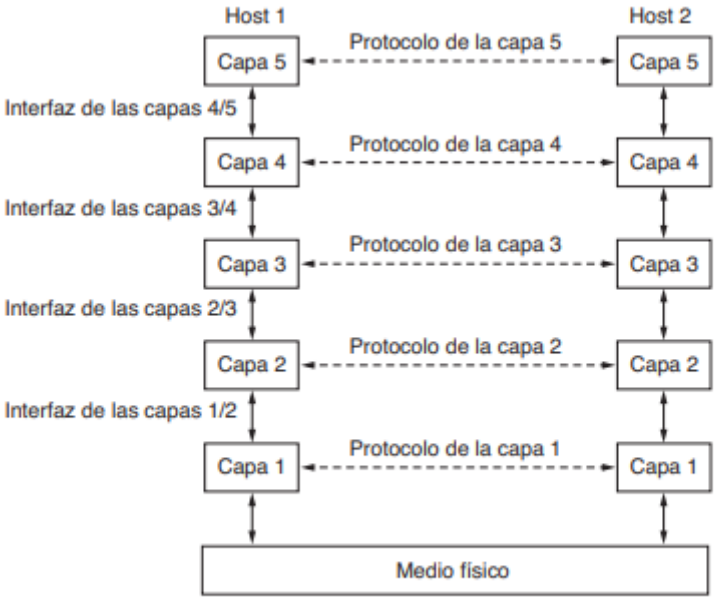


Figura 1-13. Capas, protocolos e interfaces.

PROTOCOLO DE COMUNICACIÓN

- **Ejemplos**

Capa 1: Nivel físico
Cable coaxial o UTP (categoría 5, categoría 5e, categoría 6, categoría 6a), Cable de fibra óptica, cable de par trenzado, Microondas, Radio, RS-232, RS-485.
Capa 2: Nivel de enlace de datos
ARP, RARP, Ethernet, Fast Ethernet, Gigabit Ethernet, Token Ring, FDDI, ATM, HDLC, CDP.
Capa 3: Nivel de red
IP (IPv4, IPv6), X.25, ICMP, IGMP, NetBEUI, IPX, Appletalk.
Capa 4: Nivel de transporte
TCP, UDP, SPX.
Capa 5: Nivel de sesión
NetBIOS, RPC, SSL.
Capa 6: Nivel de presentación
ASN.1.
Capa 7: Nivel de aplicación
SNMP, SMTP, NNTP, FTP, SSH, HTTP, CIFS (también llamado SMB), NFS, Telnet, IRC, POP3, IMAP, LDAP, Internet Mail 2000, y en cierto sentido, WAIS y el desaparecido GOPHER.

PROTOCOLO DE COMUNICACIÓN

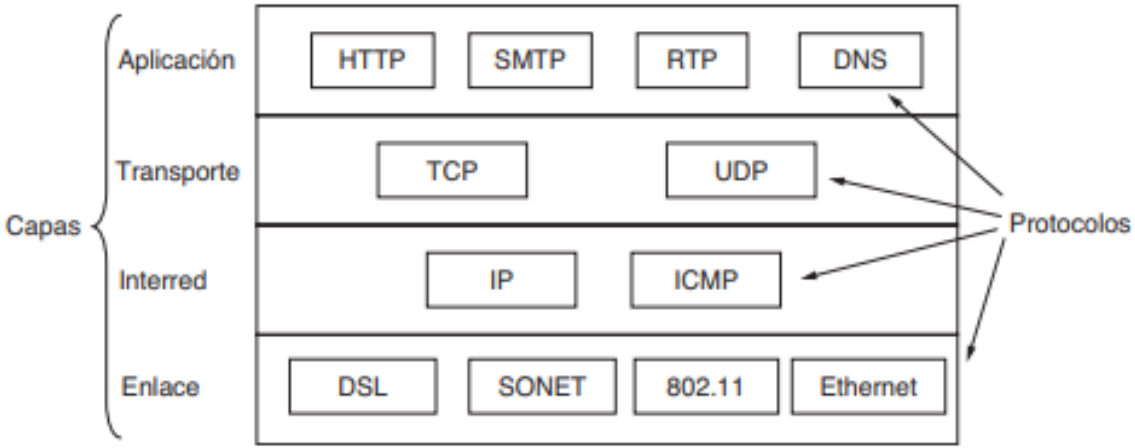


Figura 1-22. El modelo TCP/IP con algunos de los protocolos.

SISTEMA DISTRIBUIDO

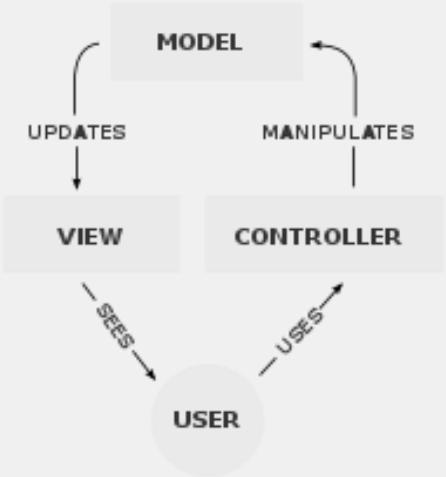
Un sistema distribuido es una colección de computadoras independientes que dan al usuario la impresión de constituir un único sistema coherente.

- **En primer lugar, tenemos que un sistema distribuido consta de componentes (es decir, computadoras) autónomos.**

- **El segundo aspecto es que los usuarios (personas o programas) creen que realmente interactúan con un sistema único. Los componentes colaboran entre sí.**

ARQUITECTURAS DE SOFTWARE

Un **patrón arquitectónico** es una solución general y reutilizable a un problema común en la arquitectura de software dentro de un contexto dado. Los patrones arquitectónicos son similares al patrón de diseño de software pero tienen un alcance más amplio.



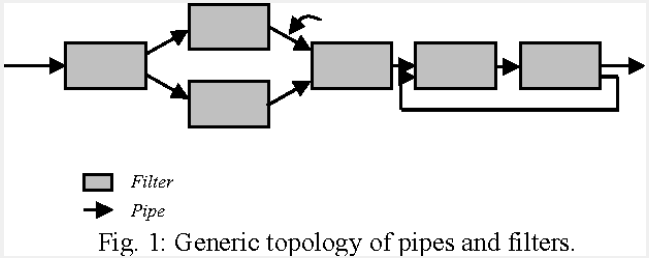
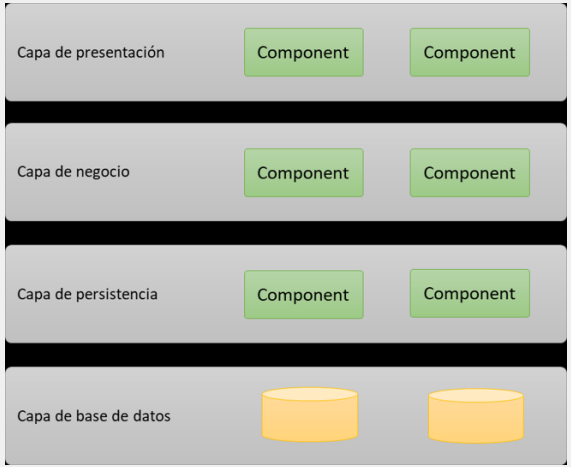


Fig. 1: Generic topology of pipes and filters.



PATRONES DE DISEÑO DE SOFTWARE

Un **patrón de diseño** es una solución repetible general a un problema que ocurre comúnmente en el diseño de software.

Un patrón de diseño no es un diseño terminado que se puede transformar directamente en código. Es una descripción o plantilla de cómo resolver un problema que se puede utilizar en muchas situaciones diferentes.

Creational	Structural	Behavioral
• Factory Method	• Adapter	• Interperter
• Abstract Factory • Builder • Prototype • Singleton	• Adapter • Bridge • Composite • Decorator • Facade • Flyweight • Proxy	• Chain of Responsibility • Command • Iterator • Mediator • Momento • Observer • State • Strategy • Visitor

DIFERENCIA ENTRE ARQUITECTURA Y PATRON DE DISEÑO

Un patrón arquitectónico expresa un esquema de organización estructural esencial para un sistema de software, que consta de subsistemas, sus responsabilidades e interrelaciones.

Los patrones de diseño Ayudan a evitar problemas sutiles en el que pueden causar mas problemas y mejora la legibilidad del código

En comparación con los patrones de diseño (es relevante a un subcomponente), los patrones arquitectónicos tienen un nivel de abstracción mayor (es relevante a la totalidad del sistema)