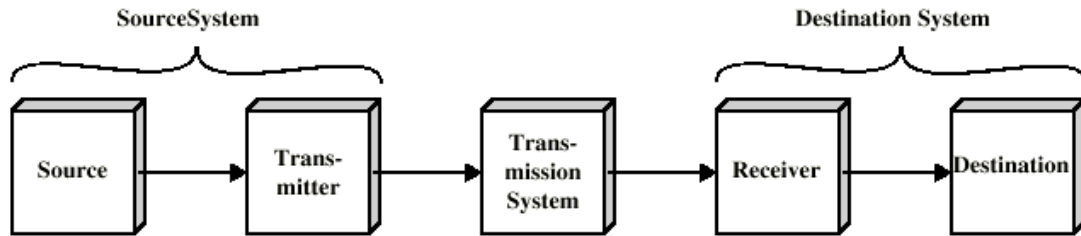
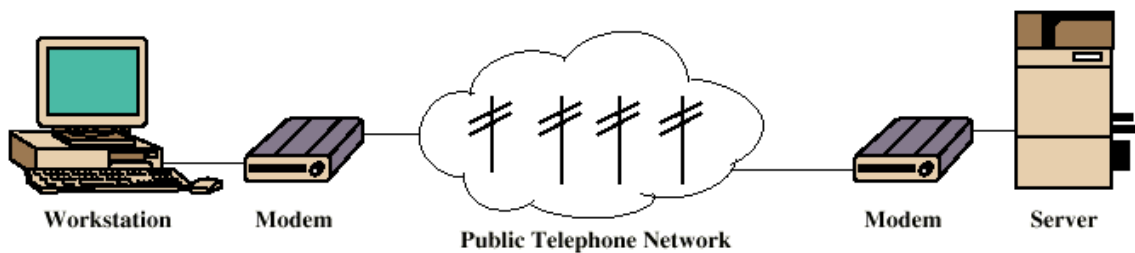


1.- INTRODUCCIÓ AL MODEL DE CAPES.

Tenint present el model de comunicacions de la figura, imaginem-nos el procés de transferència d'un fitxer de dades entre la **font** i el **destí**. A més d'**establir el camí** que seguiran les dades, cal tenir en



(a) General block diagram



(b) Example

compte tot un seguit de funcions o tasques que caldrà realitzar:

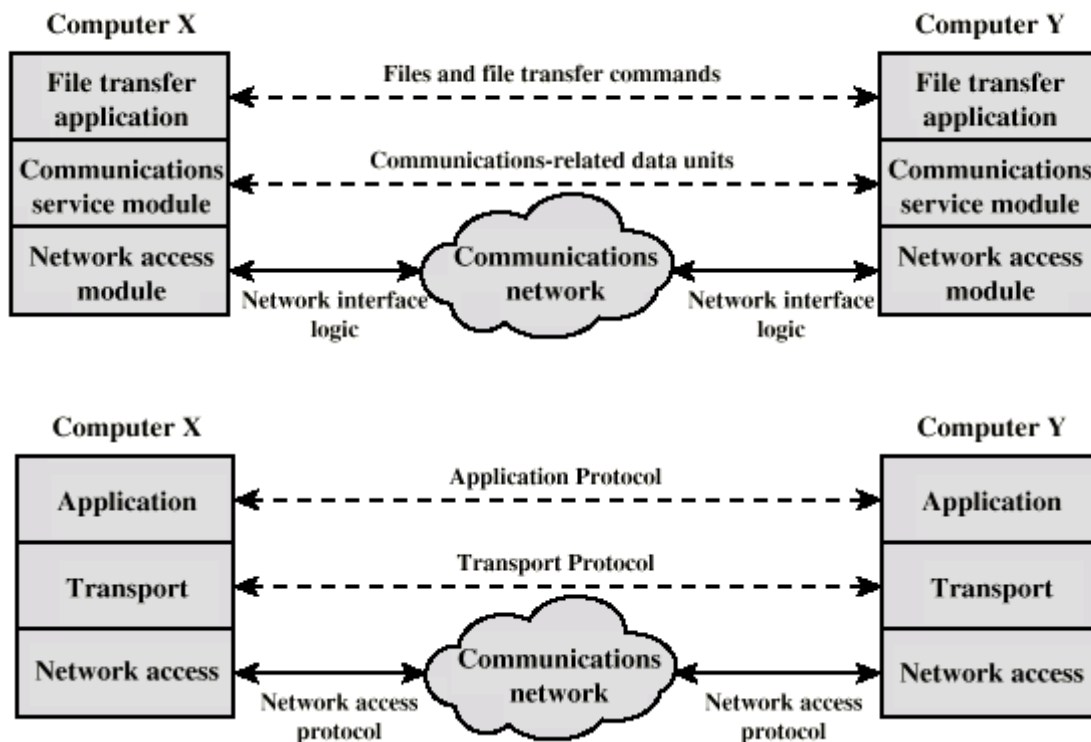
1. El sistema **font** haurà d'establir contacte directe o **indicar** a la **xarxa** la identificació del sistema **destí**.
2. La **font** ha d'assegurar-se que el **destí** està **disposat** per a rebre les **dades**
3. El **programa –aplicació–** que **transfereix** la **informació** en la **font** ha d'estar segur que qui la **gestiona** en el sistema **destí** està **preparat** per a **acceptar-la** i **emmagatzemar-la** adequadament.
4. Si els **formats** de dades requerits pels dos programes –font i destí– **no són compatibles**, un dels dos ha d'efectuar-ne la **traducció**.

Per tal d'aconseguir èxit en la comunicació cal **un conjunt de normes que controlin i governin l'intercanvi d'informació entre les aplicacions** que s'executen en els dos equips finals. Aquest conjunt de normes és, com sabem, el que anomenem un **protocol**. Un conjunt d'aquests és el que fa que ambdues aplicacions arribin a entendre's. Els protocols defineixen aspectes com el format de les dades, com es controlarà el flux d'informació i es manejaran els errors, quin significat tenen les dades transmeses,...

Tenint en compte que els dos sistemes físics que dialoguen ho poden fer per diversos motius –consulta d'una base de dades, aplicacions de correu electrònic, canvis des de l'equip font de fitxers d'aplicació residents en l'equip destí,... –, es prefereix no carregar tota la feina associada al procés de la comunicació a les aplicacions finals, i implementar-la mitjançant un conjunt de mòduls jeràrquics que es parlen entre ells tant en sentit vertical –dins el mateix sistema– com, virtualment, en sentit transversal –entre sistemes.

La figura de la pàgina següent en mostra una possible solució basada en la transferència del fitxer de dades plantejat a l'inici de l'apartat. Podem pensar en una estructura, anomenada **arquitectura de protocols**, formada per tres mòduls –o capes–:

- Un d'**aplicació**, encarregat dels aspectes associats a la generació i interpretació de les ordres específiques associades a la transferència i a la compatibilitat de les dades, és a dir als punts **3** i **4** indicats anteriorment.



- Un altre de **comunicacions**, que tindria cura de que l'intercanvi d'ordres i dades es fes en forma fiable –primera i segona tasca.
- Com els dos sistemes no tenen perquè està connectats a la xarxa de la mateixa forma, podem pensar en un tercer mòdul, anomenat de **xarxa**, que hi fes possible l'accés de cadascun d'ells. D'aquesta forma, un canvi en la forma física de connexió, només afectarà a aquest darrer mòdul.

Si partim de l'estructura anterior podem pensar en organitzar la feina de la comunicació en tres capes amb els següents noms i característiques:

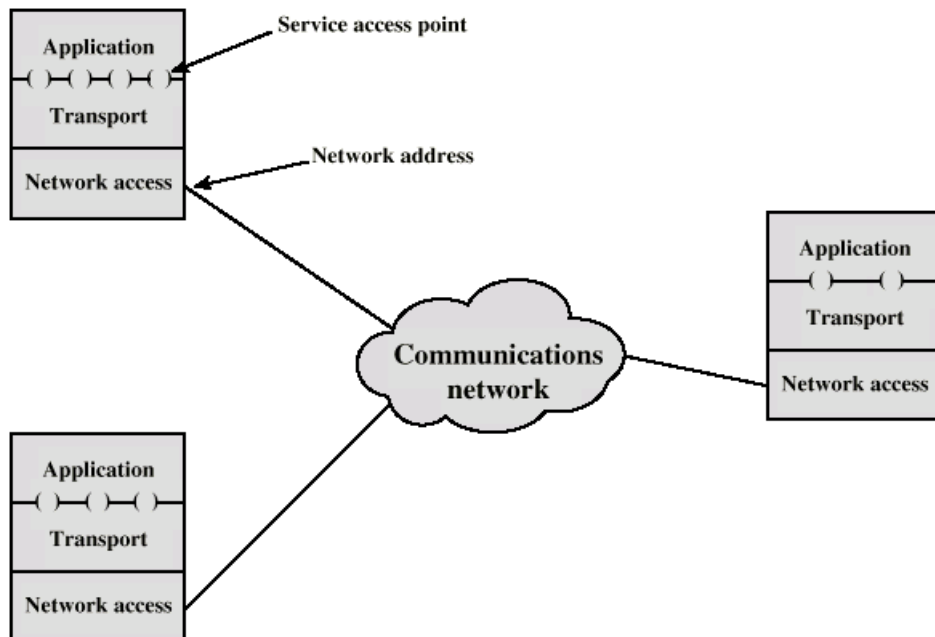
- ❑ **Accés a xarxa:** encarregada de l'intercanvi de dades entre màquina i xarxa.
El software d'aquesta capa depèn de la xarxa a que està connectat l'equip
- ❑ **Transport:** encarregada de l'intercanvi segur de dades, independentment de la xarxa i de l'aplicació.
- ❑ **Aplicació**

Anem a analitzar el funcionament del model considerant tres ordinadors connectats a una xarxa. Per tal de garantir una comunicació amb èxit, cada aplicació ha de tenir un adreçament únic. En aquest sentit cal parlar de dos tipus d'adreçament:

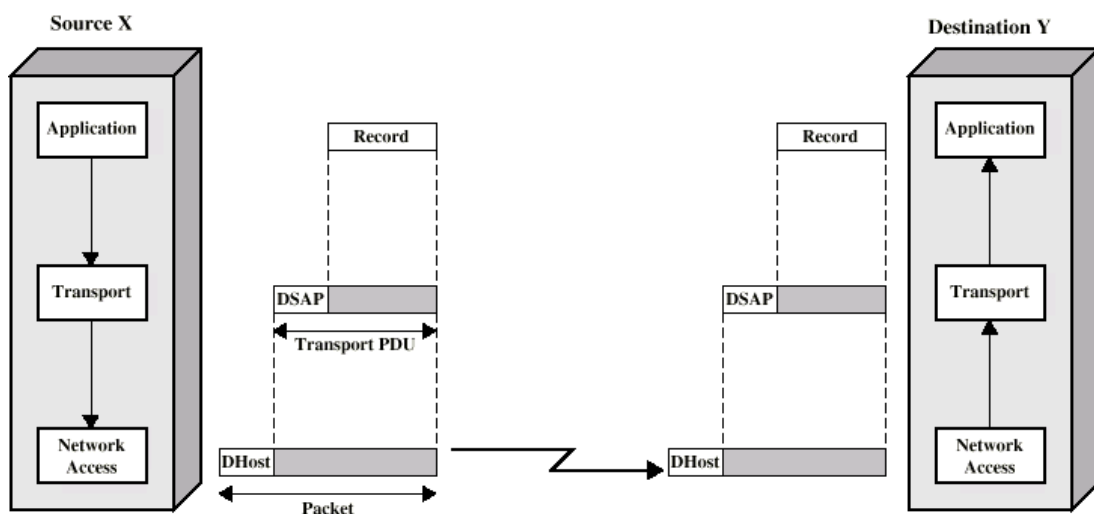
- ❑ El que té cada màquina en la xarxa
- ❑ El que té cada aplicació dins una màquina determinada, de forma que la capa de transport pugui servir-la adequadament.

Aquestes adreces internes s'anomenen **punts d'accés al servei –SAP Services Access Point–** o **ports** i serveixen per a que les distintes aplicacions dins una màquina puguin accedir a la capa de transport. Suposem que la aplicació associada a **SAP1** en el **Sistema 1** vol passar informació a **SAP2** del **Sistema 3**. Els passos a seguir serien els següents:

- L'**aplicació** passa la informació a la **seva capa de transport** amb la instrucció de que la faci arribar a **SAP2** del **Sistema 3**.



- La capa de **transport** pot **trosetejar** el **missatge**, afegint a cada bloc una **capçalera-cua** de control: **SAP** destí, número de seqüència, codi de detecció d'errors.
- El **conjunt de dades** més la **capçalera-cua** s'anomena **Unitat de Dades del Protocol –PDU–**. Aquesta serà rebuda posteriorment per la mateixa **capa** del sistema **3**, la qual serà capaç d'**interpretar** i **verificar** les dades.
- La capa de **transport** passa cada **PDU** a la **capa de xarxa** dient que la transmeti al sistema **3**. El **protocol** d'accés a xarxa hi afegeix la **capçalera** de xarxa –adreça equip destí, demandes de facilitats,... –, formant la seva **PDU**.
- La **xarxa transmet** la **PDU** de xarxa, que es **rebuda** per la **capa de xarxa** del sistema **3** que, si l'**accepta**, **elimina** la **capçalera** de xarxa i **transmet** la **informació** a la seva **capa** de



transport, la qual **interpretarà** i eliminarà la capçalera de transport i **donarà** a l'**aplicació SAP2** les **dades** pertinents.

En el món de les comunicacions i xarxes existeixen dos models d'arquitectura de protocols dels que podem dir que són en certa forma estàndards. Un d'ells és l'anomenat model de referència **OSI** –*Open System Interconnection*–, i l'altre el conjunt de protocols **TCP/IP** –*Transmission Control Protocol/Internet Protocol*. Mentre el primer d'ells va néixer ja amb la vocació d'estàndard però no ha aconseguit ser-ho, el segon s'hi ha convertit de facto.

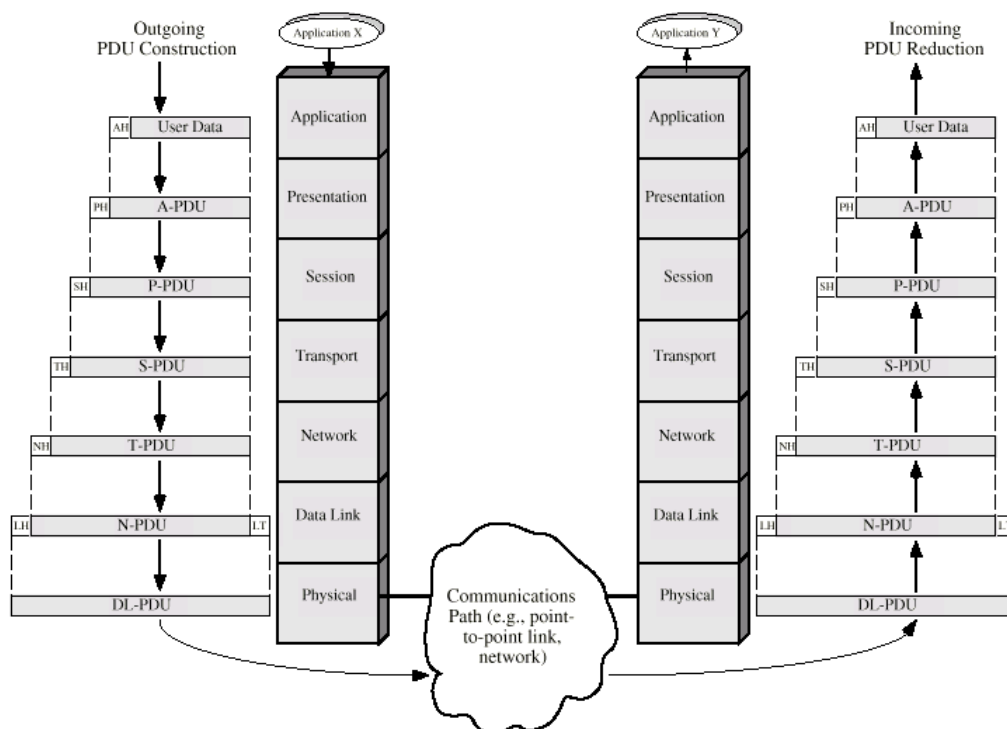
2.- EL MODEL OSI.

OSI s'estructura en set capes:

- Aplicació
- Presentació
- Sessió
- Transport
- Xarxa
- Enllaç de Dades
- Física

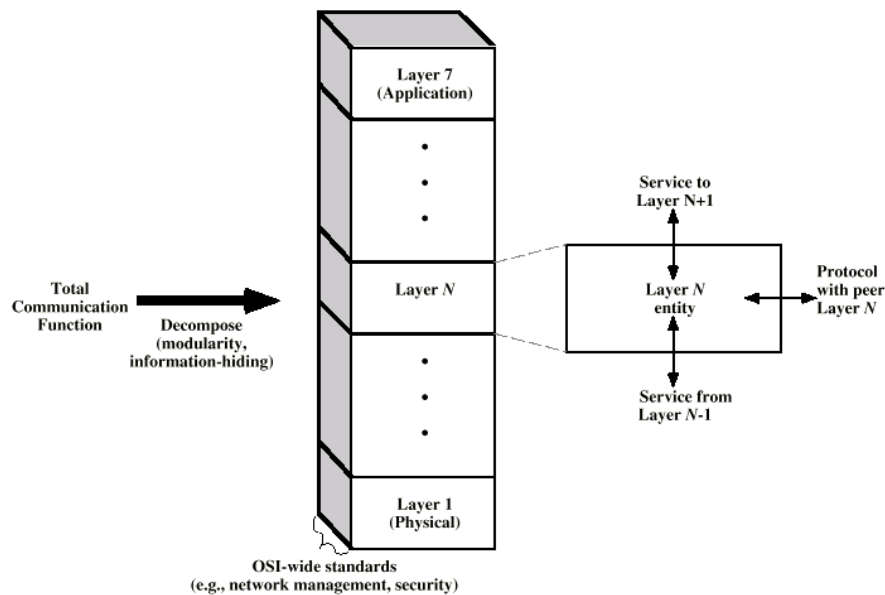
D'acord amb les idees expressades en l'apartat anterior, el sistema es basa en els següents punts:

- la feina associada a la comunicació entre dues entitats s'articula en un seguit de funcions que es divideixen entre les distintes capes que constitueixen l'arquitectura, cosa que facilita el procés en fer-lo més manejable;
- cadascuna de les capes proporciona serveis, indicats mitjançant primitives, a la capa superior; la capa es limita a desenvolupar seva feina que li encarreguen i lliurar el missatge a la capa inferior en l'arquitectura;
- cada capa de la banda de l'emissor es comunica –"s'entén"– amb la capa corresponent de la banda del destinatari
- les capes, i les funcions associades, estan definides de forma tal que canvis en una d'elles no



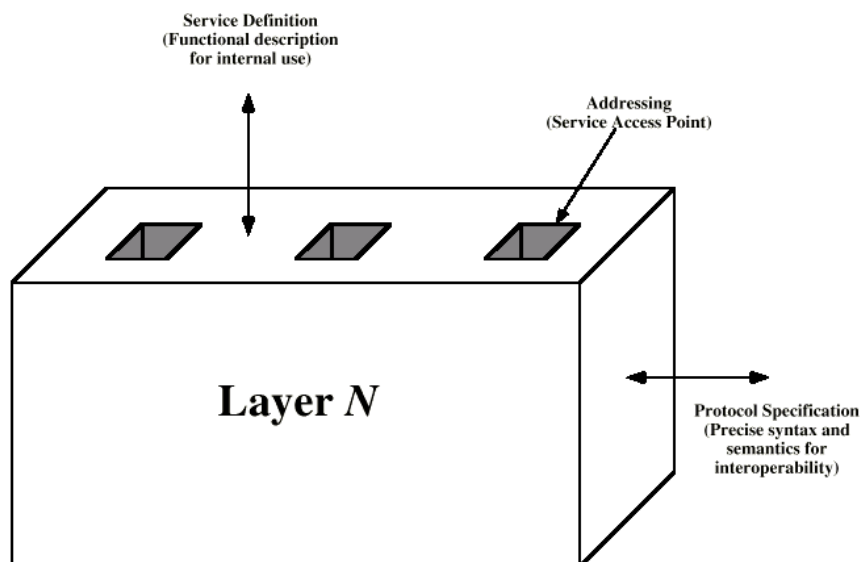
requereixen canvis en les altres.

La figura anterior mostra com funciona el model quan una aplicació **X** que corre en una determinada màquina transmet dades a l'aplicació **Y** que es desenrotlla en un altre sistema informàtic: observem la



formació de les distintes **PDU** en el sistema font i la posterior descodificació en el sistema destí. Excepte en la capa física, no existeix comunicació directa entre capes anàlogues, tot i que entre elles s'estableixen lligams lògics ja que s'utilitza, entre capes paritàries, el mateix protocol.

OSI és un model de referència que defineix les funcions que ha de realitzar cada capa, i que permet dins el model el desenvolupament de distints protocols per a una mateixa capa. Dins cada capa, es subministren serveis a la capa superior i s'elabora el protocol que permet el diàleg amb la capa paritària



de l'altre sistema connectat.

La figura mostra els elements clau de la normalització:

- **Protocol:** format **PDU**, significat distints camps i seqüenciació, si cal.
- **Servei:** quins serveis proporciona cada capa a l'anterior.
- **Adreçament:** per a subministrar serveis cal saber a qui –ús dels punts d'accés al servei.

2.1.- LES CAPES DEL MODEL OSI.

Tot seguit s'explica breument la funció de cadascuna de les capes **OSI**.

Capa Física: comunica directament amb el mitjà de comunicació i té la responsabilitat d'enviar i rebre bits,

- ☑ té cura, per tant, de la interfície física entre els dispositius i de les regles que fan possible l'intercanvi de bits.
- ☑ Ha d'especificar clarament les característiques mecàniques, elèctriques i funcionals de la connexió.

Com exemple normatiu podem citar l'estàndard **RS-232**.

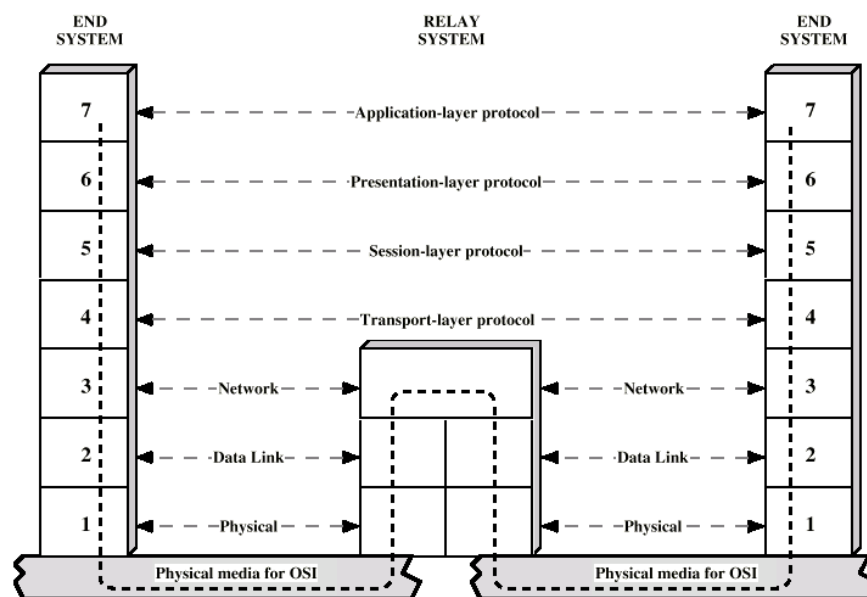
Capa d'Enllaç de Dades: s'encarrega d'enviar dades a través de la connexió física. Molt sovint aquest enllaç s'efectua a través d'una xarxa local, per la qual cosa aquesta capa té la responsabilitat de possibilitar la comunicació entre els distints nodes que la formen. Per aquest motiu ha de proporcionar les adreces dels interlocutors i ha de traduir en bits interpretables per la capa física els missatges provinents de les capes superiors. Per tal d'aconseguir-ho,

- ☑ activa, manté i desactiva l'enllaç, fent que aquest sigui segur: controla flux i errors.
- ☑ Quan els sistemes no estan directament connectats existeixen normalment distints enllaços operant independentment, amb la qual cosa les capes superiors han de participar en el control del flux i dels possibles errors en la comunicació entre els interlocutors finals.

Com exemples podem indicar **HDLC** i **LLC**.

Capa de Xarxa: molt sovint els interlocutors no es troben directament connectats ni tant sols participen en la mateixa xarxa. Aquesta capa té com a missió el fer possible la comunicació quan intervien dispositius que es troben en xarxes distintes,

- ☑ proporcionant un mètode d'adreçament que permeti identificar les distintes xarxes;
- ☑ cercant la forma més eficient de comunicació entre ambdues xarxes;
- ☑ tenint cura de la transferència d'informació a través d'alguna xarxa de comunicacions;
- ☑ l'ordinador utilitza aquesta capa per a dialogar amb la xarxa indicant el destí, petició de facilitats,...



- ☑ naturalment, no es necessita en comunicacions punt a punt.

Com exemple podem esmentar el protocol **IP**.

Capa de Transport: sovint la informació a enviar s'especeja –degut a la quantitat de dades–, o bé es treballa amb distintes aplicacions a l'hora i s'ha de multiplexar-demultiplexar la informació. La capa de transport, proporciona els mecanismes per a intercanviar dades entre els sistemes finals,

- ☑ assegurant que les dades es transmetin sense errors,
- ☑ es lliurin amb la seqüenciació adequada a les capes superiors,
- ☑ sense pèrdues o duplicitats,
- ☑ i fent ús en forma òptima dels serveis de xarxa i donant la qualitat sol·licitada.

Com exemples podem indicar **TCP** i **UDP**.

Capa de Sessió: controla el diàleg entre aplicacions en els sistemes finals. Mitjançant aquest diàleg els nodes intervinents acorden com intercanviaran dades,

- ☑ marcant la forma –*simplex*, *half-duplex* o *full-duplex*–,
- ☑ indicant si es desitja treballar en mode connexió i portant-lo a la pràctica: establiment, transferència de dades i alliberament de la connexió,
- ☑ pactant la incorporació de serveis de comprovació, agrupament i recuperació.

Capa de Presentació: prepara les dades pel viatge i les disposa per a ser utilitzades en l'aplicació de l'usuari,

- ☑ defineix el format de dades que s'intercanvien entre aplicacions,
- ☑ tradueix, si cal, les dades que la capa d'aplicació vol enviar,
- ☑ ofereix serveis de seguretat –encriptat– i transformació –compressió– de dades

Exemples de protocols d'aquesta capa serien **HTML** i **XML**.

Capa d'Aplicació:

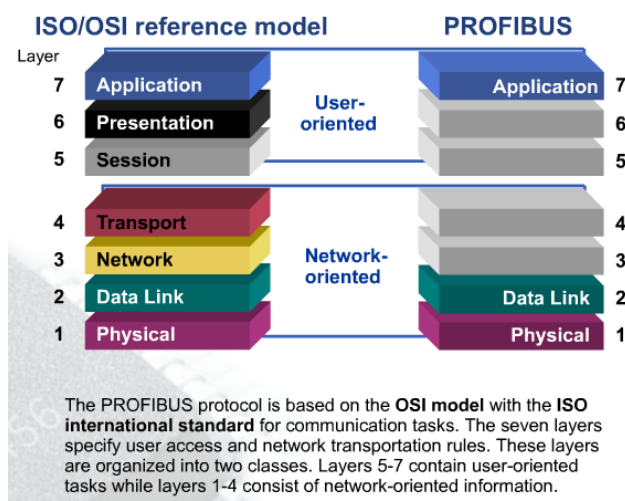
- ☑ Proporciona la forma a través de la que els programes poden accedir al model **OSI**.
- ☑ Conté aplicacions de propòsit general: transferència de fitxers, correu i accés a d'altres màquines emulant terminals.

Com a exemples podem citar **HTTP**, **FTP** i **Profinet**.

3.- MODEL OSI REDUÏT APLICAT A LES COMUNICACIONS INDUSTRIALS.

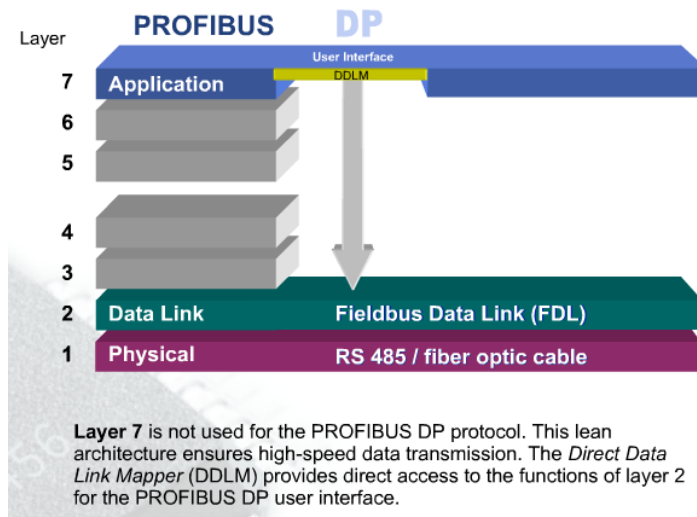
Com ja s'ha indicat, el model **OSI** mai ha estat un èxit, ja que el que podríem anomenar el mercat s'ha decantat pel model **TCP/IP**. A més, la sobrecàrrega –el nombre de capes i les seves relacions– que el model comporta s'ha demostrat ineficient a nivell industrial ja que en aquest camp es requereix assegurar la comunicació i que aquesta es produeixi de la forma més ràpida possible.

En aquest sentit, les xarxes industrials que es construeixen a partir del model **OSI**, per exemple **Profibus**, fan servir només tres de les capes **OSI**: la física, la capa d'enllaç i la d'aplicació.

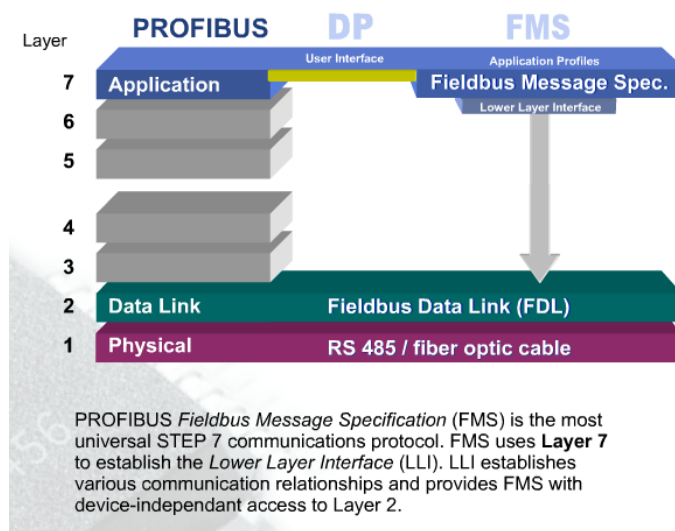


En gairebé totes les xarxes –com veurem en un altre document, **Ethernet** n'és un exemple–, en la capa d'enllaç se situa el subnivell d'accés al medi –**MAC**, *Medium Acces Control*. En les xarxes industrials interessa que el mecanisme d'accés al medi tingui un comportament determinista, és a dir, que asseguri el lliurament de la informació dins un límit de temps determinat, per la qual cosa solen utilitzar-se sovint polítiques de prioritat. A més, els protocols associats al subnivell MAC han de ser molt senzills per tal de permetre la implementació per maquinari.

La capa d'aplicació defineix la interfície entre l'usuari i el sistema, és a dir, és la forma en que l'usuari veu el sistema de comunicacions. Aquesta capa ha d'incorporar els subnivells a través dels quals accedeix als serveis que li proporciona la capa d'enllaç.



En aquest nivell, els estàndards proporcionen elements específics per a distints dominis d'aplicació: perifèria descentralitzada, serveis adreçats a missatges, control de processos, robòtica, ... Són els anomenats perfils –*profiles*.



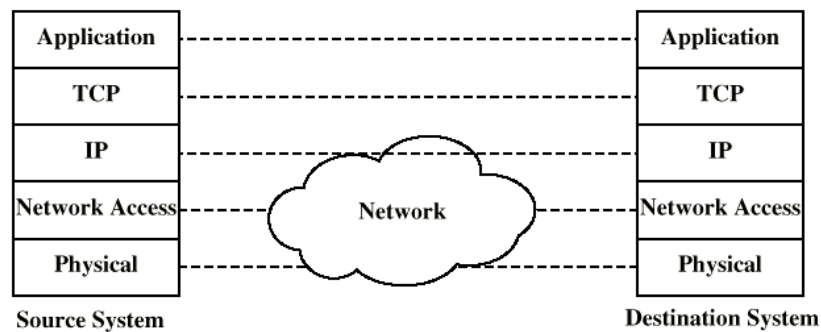
4.- L'ARQUITECTURA TCP/IP.

Tal com s'ha esmentat en el primer apartat, l'arquitectura de protocols **TCP/IP** ha esdevingut l'estàndard real en el món de les comunicacions, entre d'altres motius per la implantació creixent d'**Internet**. Respecte a **OSI** podem indicar el següent:

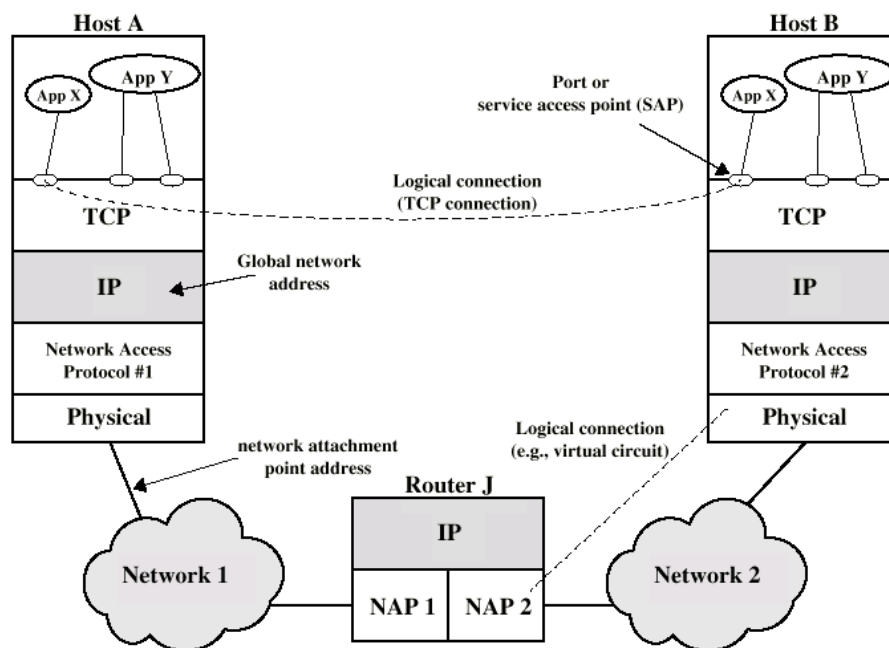
- ☑ Més que un model de referència, **TCP/IP** és un conjunt de protocols.
- ☑ No existeix un model oficial.
- ☑ No és obligatori l'ús de tots els mòduls o capes.

Tot i que no ve definit així en **TCP/IP**, per tal de fer-ne una anàlisi, convé, però, considerar l'arquitectura com un model de quatre o cinc capes:

- **Aplicació:** encarregada de gestionar la comunicació entre processos o aplicacions.
- **Transport** o Ordinador-a-Ordinador: que assegura els serveis de transferència de dades extrem a extrem.



- **Internet:** encarregada de l'encaminament de dades *-routing-* des de l'origen al destí a través de les distintes xarxes connectades
- Accés a **Xarxa**, que efectua la interfície entre l'equip final i la xarxa –anomenada aquí subxarxa.
- **Física**, en què es defineixen les característiques del mitjà de transmissió, velocitat, codificació de senyals,...



4.1.- FUNCIONAMENT BÀSIC DE TCP/IP.

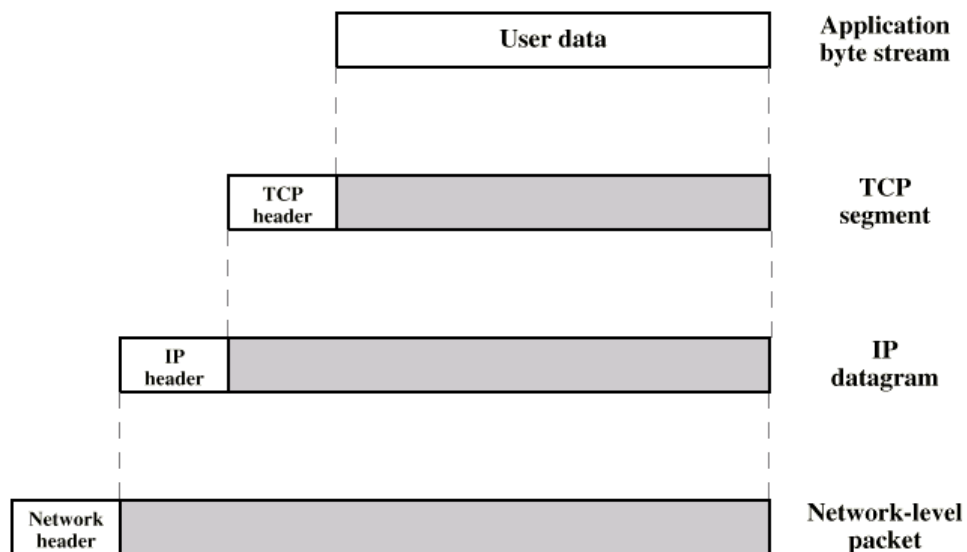
Podem considerar que el conjunt de protocols es basa en una sèrie d'idees molt senzilles:

- ☑ el sistema de comunicació es forma a partir de múltiples xarxes que anomenarem subxarxes.

- ☑ Per a comunicar un ordinador amb una subxarxa s'usa algun dels protocols normals d'accés a xarxa –per exemple, **CSMA/CD** o **Token-ring**.
- ☑ Això permet a una màquina enviar dades a una altra de la mateixa subxarxa o a un encaminador o *router*.
- ☑ El protocol **IP** –*Internet Protocol*– està implementat en totes les màquines –ordinadors i *routers*– i actua com a retransmissor.
- ☑ El protocol **TCP** –o d'altres protocols paritaris– implementat en els equips finals assegura que les dades es lliuren de forma segura a l'aplicació adequada.
- ☑ Per tal d'assegurar la comunicació entre aplicacions, cada procés o aplicació ha de tenir una adreça única en el sistema.
- ☑ Aquesta adreça es forma a base de la unió de la adreça internet **IP** que identifica cada ordinador dins la subxarxa a què està connectat, i de la adreça –**port**– que identifica l'aplicació dins l'ordinador, la qual cosa permet al protocol extrem-a-extrem lliurar les dades al procés adequat.

Imaginem-nos, per exemple, que el procés associat al port **1** en la màquina **A** –adreçament **A.1**– vol enviar un missatge al procés lligat al port **2** de la màquina **B** –**B.2**. El passos que es segueixen són semblants als següents:

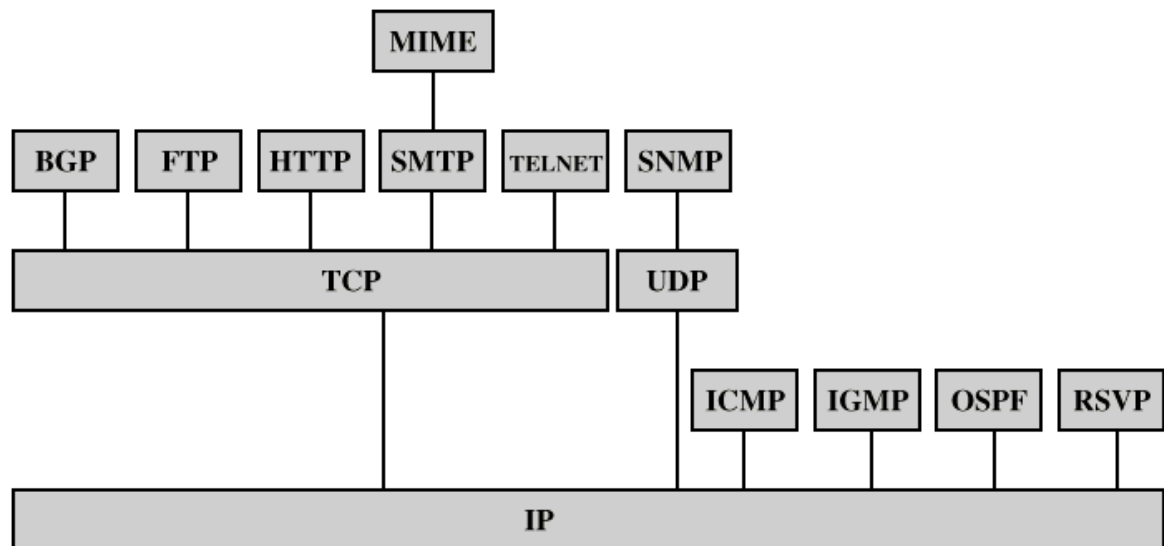
- el **procés en A** passa el **missatge** a **TCP** indicant-li que ho **envii** a **B.2**.
- **TCP**, que pot trossejar el missatge, **elabora** un, o més d'un, **segment TCP afegint** a la informació transmesa per la capa d'aplicació una **capçalera** amb: port destí, número de seqüència i camp de comprovació.



- **TCP en A** passa el **segment** a la **capa IP** amb la **indicació** de que el **transmeti** a **B** –no cal que indiqui el port destí, contingut en la capçalera **TCP**.
- **IP** li **adjunta** la seva **capçalera** i li **passa** el **datagrama** així format a la **capa** d'accés a **xarxa**, per exemple **Ethernet**, indicant-li que ho **transmeti** a un dispositiu **encaminador –J**.
- El software de **xarxa** li **afegeix** la seva **capçalera** –adreça destí (*router*), petició de facilitats,...
- Els **paquets** així formats **viatgen** per la **subxarxa** fins el **router**. En **arribar**-hi, s'**elimina** la **capçalera de xarxa** i s'**analitza** la **capçalera IP** –en la capa corresponent–, la qual cosa **serveix** per a **encaminar** el paquet cap un **nou router o adreçar-lo directament** a la màquina **B**. Per tal d'aconseguir-ho s'hi afegeix una nova capçalera **IP** i es passa el paquet a la nova subxarxa.

- El procés es va repetint fins que **el missatge arriba a B**. Cadascuna de les capes **interpreta la capçalera** que l'afecta i, si tot és correcta, l'elimina i **lliura les dades rebudes a la capa superior**, fins que la capa **TCP** l'envia al **procés** associat al **port 2**.

Tal com hem indicat anteriorment l'arquitectura **TCP/IP** no exigeix l'ús de totes les capes. Tal com mostra la figura adjunta en que es mostren alguns dels protocols es poden desenvolupar aplicacions que fan ús directe d'alguna de les capes sense passar per les superiors.



BGP = Border Gateway Protocol	OSPF = Open Shortest Path First
FTP = File Transfer Protocol	RSVP = Resource ReSerVation Protocol
HTTP = Hypertext Transfer Protocol	SMTP = Simple Mail Transfer Protocol
ICMP = Internet Control Message Protocol	SNMP = Simple Network Management Protocol
IGMP = Internet Group Management Protocol	TCP = Transmission Control Protocol
IP = Internet Protocol	UDP = User Datagram Protocol
MIME = Multi-Purpose Internet Mail Extension	

BIBLIOGRAFIA EMPRADA:

Stallings, W. (2008) *Comunicaciones y Redes de Computadores*. Prentice Hall.

Castro Gil i d'altres. (2007) *Comunicaciones Industriales: Principios básicos*. UNED.

Castro Gil i d'altres. (2007) *Comunicaciones Industriales: Sistemas distribuidos i aplicaciones*, UNED.

Halsall, F. (1998) *Comunicaciones de Datos, Redes de Computadores y Sistemas Abiertos*. (3a Edició). Addison Wesley.

Schneider Electric. [Guía de Soluciones de Automatización. Redes Industriales](#).