



Operazione Rif PA 2021-15946/RER

approvata con DGR 1263/2021 del 02/08/2021

Progetto n. 1 - Edizione n. 1

TECNICO PER LA PROGETTAZIONE E LO SVILUPPO DI APPLICAZIONI INFORMATICHE

**MODULO: N. 6 Titolo: AMMINISTRAZIONE DI SISTEMI SERVER
DURATA : 34 ORE DOCENTE: MARCO PRANDINI**

Richiami di reti di calcolatori



Introduzione alle reti di calcolatori



Marco Prandini
marco.prandini@unibo.it

EnAIP Tecnico Informatico



A cosa serve una rete

Dove viene utilizzata l'informatica esistono:

- **Una molteplicità di risorse**
 - di calcolo
 - di memorizzazione
 - per l'ingresso e l'uscita dei datiassociate a stazioni di lavoro diverse
- **Una continua necessità**
 - di accedere alle risorse
 - di fornire o recuperare informazioniindipendentemente dalla stazione su cui si lavora



Connessione di calcolatori

Idea di fondo: far “parlare” tra loro due calcolatori collegandoli elettricamente.

■ **Vantaggi rispetto allo spostamento fisico dei dati:**

- Capacità (sempre?)
- Trasparenza
- Robustezza
- Interattività

■ **Problemi da risolvere:**

- costruire l’infrastruttura
- concordare un linguaggio comune
- proteggere il sistema dalle intrusioni

EnAIP Tecnico Informatico



Stabilire le regole del dialogo

Perchè i calcolatori connessi si capiscano, è necessario stabilire uno standard sul formato fisico dei segnali, la temporizzazione dei messaggi, ed il significato dei simboli trasmessi dall'uno all'altro.

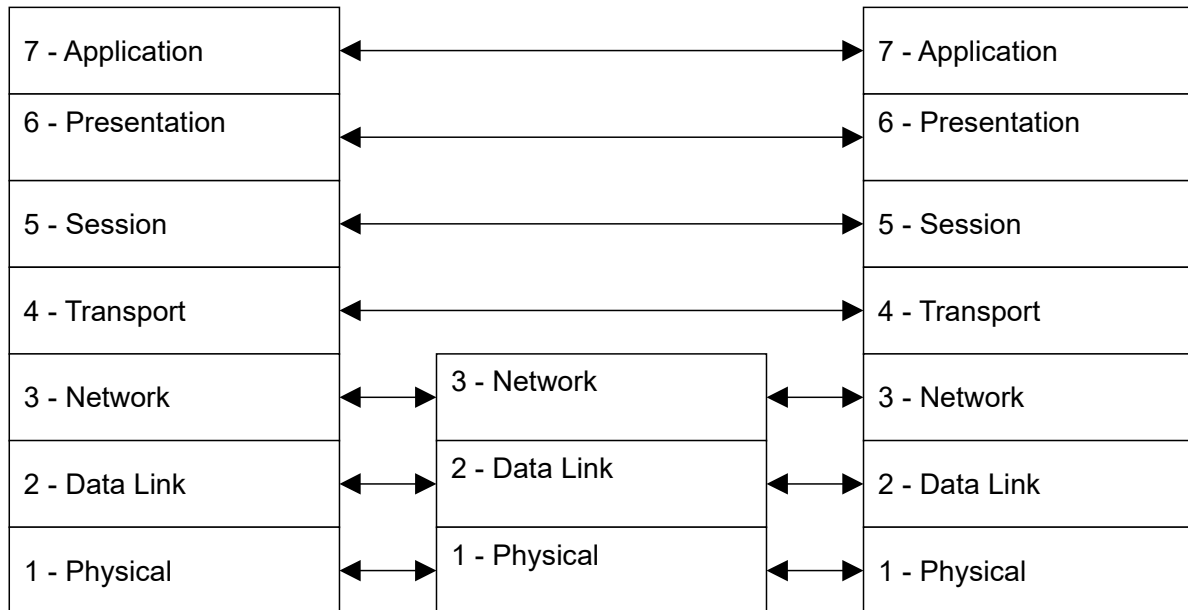
Se affrontato in modo monolitico, il problema risulta complesso e le soluzioni non generalizzabili.

Per questo motivo, l'ISO ha formalizzato il modello OSI, che rappresenta la comunicazione come una pila (stack) di livelli, ognuno deputato a svolgere un ruolo preciso senza che i dettagli implementativi siano noti agli altri livelli.

EnAIP Tecnico Informatico



Modello ISO/OSI



EnAIP Tecnico Informatico



Strati 1-2-3

- Lo strato 1 (Physical) definisce le caratteristiche elettriche e meccaniche dei mezzi di trasporto delle informazioni.
- Lo strato 2 (Data Link) gestisce le comunicazioni tra sistemi connessi alla stessa rete fisica. Esegue il controllo d'errore sui dati trasportati dal livello fisico, definisce i metodi per identificare un sistema all'interno una rete locale e le politiche di accesso dei sistemi al livello fisico.
- Lo strato 3 (Network) si occupa dell'*instradamento (routing)*, ovvero della ricerca del percorso per connettere due sistemi che vogliono comunicare tra loro anche se sono su reti fisiche distinte. I metodi per identificare un sistema su scala globale (come gli indirizzi IP usati in Internet) sono definiti in questo strato.

EnAIP Tecnico Informatico



Strati 4-5-6-7

- Lo strato 4 (Transport) si occupa del trasporto dei dati *end-to-end*. Fornisce allo strato superiore canali affidabili tra i due estremi della comunicazione, occupandosi del controllo di flusso, della correzione degli errori, della divisione e ricomposizione dei dati. Nasconde la frammentazione dei percorsi “reali” che gli sono forniti dallo strato 3.
- Gli strati 5 e 6 (Session-Presentation) sono raramente implementati, e modellano gli strumenti quali le procedure di negoziazione della localizzazione, i controlli di integrità e validità dei documenti, e la sincronizzazione tra applicazioni interattive.
- Lo strato 7 (Application) è quello in cui si collocano le applicazioni che originano e ricevono i dati oggetto della comunicazione

EnAIP Tecnico Informatico

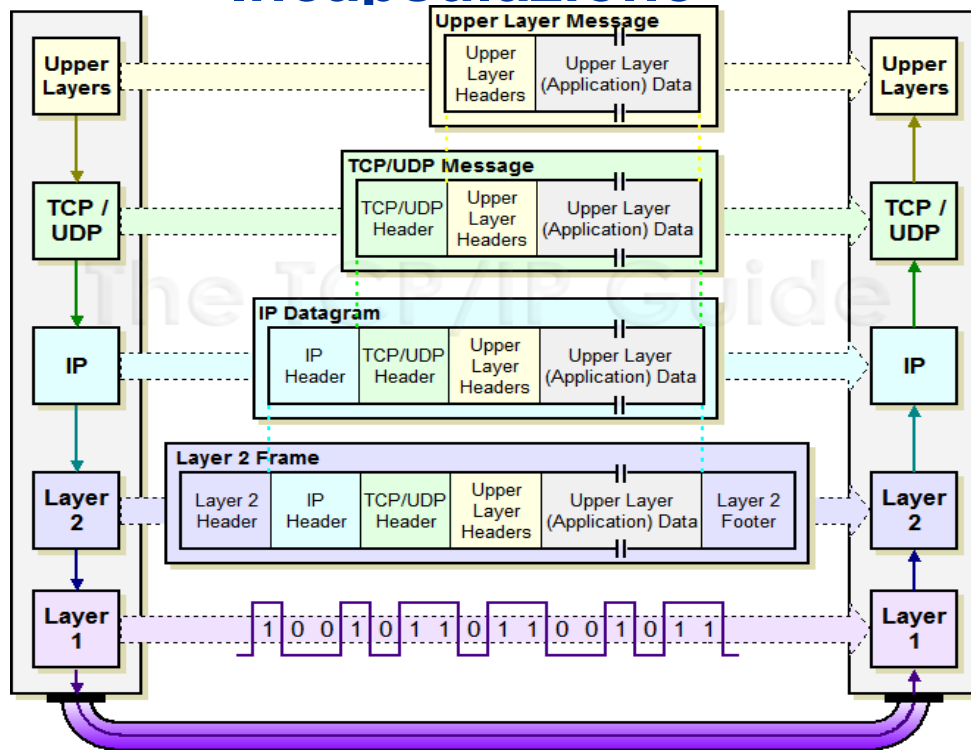


Percorso logico e fisico dei dati

- Fisicamente un messaggio originato da un'applicazione (strato 7) percorre il sistema di origine scendendo di strato in strato, attraverso le *interfacce*.
 - Ogni strato sottopone il messaggio alle elaborazioni necessarie per lo svolgimento del proprio compito, lo *incapsula* in un pacchetto arricchito di metadati in modo che il proprio omologo nel sistema di destinazione possa effettuare le operazioni inverse, e lo passa allo strato inferiore.
- Logicamente, durante l'interazione di due sistemi avviene uno scambio di messaggi tra ciascuno degli strati omologhi, secondo le regole stabilite da un *protocollo*.
 - Il vantaggio della struttura a strati risiede nella possibilità di scegliere qualsiasi protocollo per uno strato senza influenzare quelli circostanti. (Es. passaggio di Microsoft a TCP/IP)

EnAIP Tecnico Informatico

Incapsulazione



EnAIP Tecnico Informatico

Reti Locali

- Una rete locale può essere definita come un insieme di calcolatori connessi da un'infrastruttura tale per cui ogni partecipante ha visibilità fisica diretta di tutti gli altri.
 - Infrastruttura basata sui soli strati 1 e 2
 - Caratterizzate da
 - Topologia (bus, anello, stella)
 - Mezzo trasmissivo (coassiale, doppino, fibra, etere, ...)
 - Politica di accesso (token-based, collisione, ...)
- L'identificazione di ogni nodo avviene per mezzo di un indirizzo che può avere caratteristiche arbitrarie, perchè per definizione deve essere raggiungibile senza istruire intermediari --> problemi di interconnessione

EnAIP Tecnico Informatico



Internetworking

- Usando il solo livello 2 ...
 - Come garantire l'univocità degli indirizzi a livello globale?
 - Come limitare la propagazione del traffico?
- Usando il livello 3 si sovrappone uno schema di indirizzi configurabili all'indirizzamento di livello 2
 - Gli indirizzi sono concessi, sia numericamente che organizzativamente, secondo una gerarchia
 - La contiguità numerica degli indirizzi rispecchia quella fisica dei corrispondenti nodi della rete
- Da questo momento non tratteremo più concetti generali ma faremo riferimento allo standard mondiale di fatto: TCP/IP

EnAIP Tecnico Informatico



Internet

EnAIP Tecnico Informatico

Origini di TCP/IP

Le radici di Internet affondano nella Guerra Fredda.

■ Problema:

- Un sistema di comunicazione a prova di attacco nucleare.

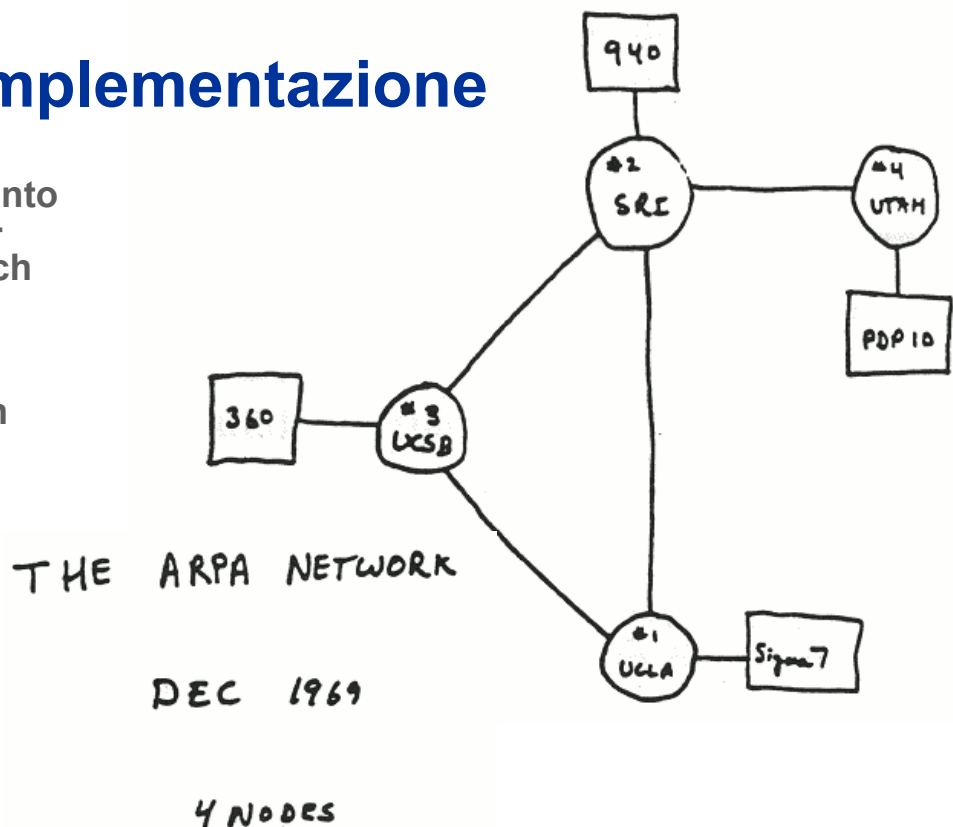
■ Soluzione (RAND Corporation, 1964):

- Una *rete*, anziché un albero di interconnessioni, modello tradizionalmente utilizzato nelle telecomunicazioni
- Moltitudine di nodi nessuno dei quali abbia un “ruolo di comando”
- Moltitudine di connessioni tra i nodi (ridondanza dei percorsi)
- Traffico suddiviso in pacchetti
- La funzione di ogni nodo è il *packet forwarding*

EnAIP Tecnico Informatico

La prima implementazione

- DARPA: Dipartimento della Difesa USA + Advanced Research Projects Agency
- BBN
- RAND Corporation



EnAIP Tecnico Informatico



Alcune tappe fondamentali

- 1972 Invenzione della posta elettronica.
- 1973 Primi nodi internazionali (UK, Norvegia).
Vinton Cerf pubblica TCP.
- 1983 Smilitarizzazione.
Primo uso del termine Internet.
Introduzione dei nomi.
- 1986 NSF collega i propri centri e le università al di fuori di ARPAnet (che ora conta 1000 nodi) usando TCP/IP.
- 1990 ARPAnet viene dismessa.
I nodi TCP/IP superano le 100.000 unità.
- 1992 Nascita del World Wide Web.
Più di 1 milione di nodi.
- 1993 Primo browser grafico (NCSA Mosaic)
Il tasso di crescita di Internet sfiora il 350% annuo.

EnAIP Tecnico Informatico



Organismi, standard, documentazione

L'impulso all'evoluzione di Internet è stato dato da una forma di collaborazione estremamente libera. I documenti fondamentali (Request For Comments – RFC) prodotti dal Network Working Group seguivano queste “regole” (RFC 3):

«Il contenuto di una nota del NWG può essere qualsiasi riflessione, suggerimento, o altro soggetto relativo al software per host o ad aspetti diversi della rete. È incoraggiata la sottomissione tempestiva delle note piuttosto che la cura della forma. Questioni teoriche prive di esempi o applicazioni immediate, suggerimenti specifici, tecniche di implementazione prive di spiegazioni introduttive e di contesto, domande esplicite prive di qualunque tentativo di risposta sono tutti argomenti accettabili. La lunghezza minima per una nota del NWG è di una frase.»

EnAIP Tecnico Informatico



Organismi, standard, documentazione

- La RFC 1 “Host Software” è del 7 aprile 1969. Ora sono oltre 4400.
- Attualmente, esistono diversi organismi che regolano i vari aspetti di Internet, dalla ricerca avanzata ai dettagli applicativi:
 - IAB (Internet Architecture Board) svolge il coordinamento e decide le strategie a lungo termine
 - IRTF (Internet Research Task Force) si occupa della ricerca di base, con progetti a lungo termine
 - IETF (Internet Engineering Task Force) si occupa degli aspetti tecnici, e produce gli Internet Drafts che precorrono le RFC.
 - IANA (Internet Assigned Numbers Authority) si occupa della registrazione degli IP, degli host, e dei servizi TCP e UDP.
- Il coordinamento, le attività di promozione, finanziamento e formazione sono svolte dall’Internet Society.

EnAIP Tecnico Informatico



TCP/IP - generalità

La sigla TCP/IP indica l’insieme dei protocolli usati in ambito Internet.

Alcune caratteristiche:

- È uno standard aperto, slegato da qualsiasi produttore
- È definito ufficialmente dall’IETF nelle Request For Comments
- È indipendente dall’hardware
- Svolge principalmente le funzioni proprie degli strati 3 e 4:
 - Prevede una modalità di indirizzamento globale
 - Supporta diverse modalità di routing
 - È basato sulla commutazione di pacchetto
- Fornisce agli strati superiori connessioni multiple e di diversi tipi

EnAIP Tecnico Informatico



TCP/IP – schema a livelli

Lo stack TCP/IP è più semplice di quello OSI, e prevede solo 4 livelli

<u>Application Layer</u>	(~ 5,6,7)	Protocolli di posta, trasferimento file, web, condivisione dischi, ...
<u>Transport Layer</u>	(~ 4)	Servizi di moltiplicazione e di garanzia della completezza dei dati
<u>Internet Layer</u>	(~ 3)	Indirizzamento dei nodi ed instradamento dei pacchetti
<u>Network Access Layer</u>	(~ 1,2)	Mappatura degli indirizzi Internet sulla LAN ed accesso al mezzo

EnAIP Tecnico Informatico



Internet Protocol

IP (*Internet Protocol*) è il protocollo che svolge le funzioni classificate come strato 3 dal modello OSI; fornisce un servizio di trasporto dati attraverso un'internet, mediante il meccanismo della commutazione di pacchetto:

- tutti i dati consegnati ad IP dallo strato superiore vengono frammentati in unità di dimensione massima prefissata (*pacchetti*)
- per ogni singolo pacchetto viene individuato un percorso che porti dal sistema sorgente al sistema di destinazione, eventualmente attraverso sistemi intermedi

IP è un protocollo di tipo *best effort*, ovvero non dà garanzie:

- sull'effettivo arrivo di un pacchetto
- sull'ordine di arrivo dei pacchetti
- sulla velocità di trasmissione o sui tempi di percorrenza

EnAIP Tecnico Informatico



Indirizzi IP

- Associazione univoca indirizzo → sistema (*host*)
 - (non è imposto il contrario → *multi-homed host*)
- Indirizzi IPv4: 32 bit divisi in 4 byte, normalmente rappresentati con 4 numeri in base 10 separati da punti (*dotted decimal notation*)
 - Esempio: 137.204.59.1
- Ogni indirizzo fa parte di una rete (*subnet*) che inizia da un indirizzo di *network*
- In origine l'estensione era implicita, ora è specificata da una *netmask*
 - Una subnet logica coincide con una LAN fisica
 - Per instradare un pacchetto verso la destinazione non serve considerare il suo indirizzo, basta la subnet cui appartiene

EnAIP Tecnico Informatico



Reti class-based

Originariamente sono state definite *classi* di indirizzi, ovvero raggruppamenti da usare per i sistemi di una rete locale, in cui i byte erano rigidamente divisi tra **network id** e **host id**:

- 128 reti di Classe A (contenenti fino a 16 milioni di host circa):
 - indirizzi da 0.*.* a 127.*.*
- 16.384 reti di Classe B (contenenti fino a 65000 host circa):
 - indirizzi da 128.0.*.* a 191.255.*.*
- 2.097.152 reti di Classe C (contenenti fino a 254 host):
 - indirizzi da 192.0.0.* a 223.255.255.*
- Gli indirizzi da 224.*.* a 239.*.* sono riservati al *multicast*
- Gli indirizzi da 240.*.* a 255.*.* sono riservati a usi futuri

EnAIP Tecnico Informatico



Il vantaggio sulle LAN

- Host su una stessa rete locale:
 - collocati in una stessa area, servita da un determinato apparato di rete
 - numericamente identificati da indirizzi di una stessa subnet



- Per raggiungerli non ho bisogno di sapere dove si trova ognuno, mi basta sapere come raggiungere la subnet

EnAIP Tecnico Informatico



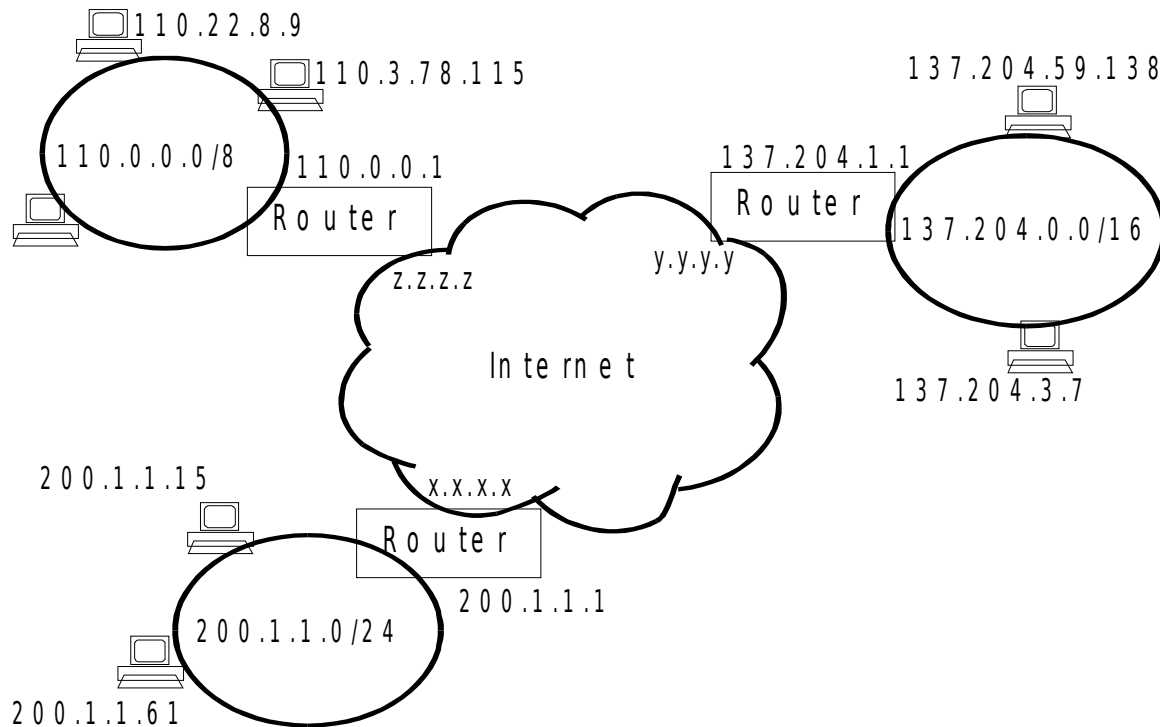
Instradamento del traffico

- Ogni host di una rete locale IP “sa” quali altri host può raggiungere direttamente, grazie a network address e netmask.
- Ogni host può essere configurato per raggiungere destinazioni al di fuori della rete locale usando degli intermediari: i *router*
 - Un router per instradare traffico tra LAN diverse deve avere un'interfaccia connessa ad ognuna, a cui deve essere associato un indirizzo della corrispondente subnet.
 - Più in generale, un router sa come raggiungere *ogni* host/subnet, direttamente o attraverso altri router
 - L'elenco delle destinazioni non è conservato da ogni router in modo estensivo, ma tipicamente contiene l'indicazione di un router a cui delegare l'instradamento verso tutte le subnet non esplicitamente note: il *default gateway*.

EnAIP Tecnico Informatico



Internetworking con TCP/IP



EnAIP Tecnico Informatico



CIDR

- Poche classi di dimensioni fissate = spreco di indirizzi
 - Soluzione: CIDR (*classless inter-domain routing*)
 - Con classless-IP gli indirizzi sono visti come una stringa di 32 bit divisa in net-id e host-id in un punto arbitrario, anziché per byte.
 - Unico vincolo (ovvio): la dimensione di una rete è potenza di 2 (in questo esempio $2^6 = 64$ indirizzi)
- 144 156 166 151

1 0 0 1 0 0 0 0 1 0 0 1 1 1 0 0 1 0 1 0 0 1 1 0 1 0 0 1 0 1 1 1
- 26 bit net-id 6 bit host-id
- Una rete locale è identificata per mezzo di un *network address* e di una *netmask*, noti a tutti gli host che ne fanno parte.

EnAIP Tecnico Informatico



Netmask, network, broadcast

- Serve un modo per specificare dove cade la divisione: la *netmask* è un valore di 32 bit composto da tanti “1” quanti sono i bit che identificano la subnet, e tanti “0” quanti sono i bit che specificano l'host al suo interno
 - Nell'esempio precedente:
11111111.11111111.11111111.11000000 = 255.255.255.192
- Due valori in ogni subnet hanno un significato speciale e non possono essere usati per indirizzare un host:
 - Network address (ottenuto mettendo a 0 tutti i bit dell'host-id)
 - **10010000.10011100.10100110.10000000** = 144.156.166.128
 - Broadcast (ottenuto mettendo a 1 tutti i bit dell'host-id)
 - **10010000.10011100.10100110.10111111** = 144.156.166.191

EnAIP Tecnico Informatico



Netmask, network, broadcast... perchè?

- Il sistema è stato pensato per velocizzare il lavoro degli apparati di instradamento
 - ogni pacchetto contiene un indirizzo di destinazione
 - deve essere inviato verso la subnet che contiene tale indirizzo
 - i router conoscono il modo di raggiungere le subnet per mezzo di una tabella che elenca (network, netmask, interfaccia da usare)
 - l'operazione di AND bit-a-bit tra un indirizzo e una netmask restituisce il valore da confrontare con il network
- Facile costruzione di gerarchie di subnet, per diminuire ulteriormente il numero di regole di instradamento memorizzata su ogni router

EnAIP Tecnico Informatico



Il livello di trasporto

- TCP = Transmission Control Protocol
- UDP = User Datagram Protocol
- Le funzioni svolte da questi protocolli sono classificabili nell'ambito dello strato 4 del modello OSI. Conseguentemente, operano su di un canale di comunicazione dei dati end-to-end, fornito da IP che nasconde i dettagli della gestione dei pacchetti.
 - IP non permette di distinguere diverse applicazioni sorgente o destinazione dei pacchetti su di uno stesso host
 - IP non fornisce garanzie sulla consegna dei pacchetti

EnAIP Tecnico Informatico



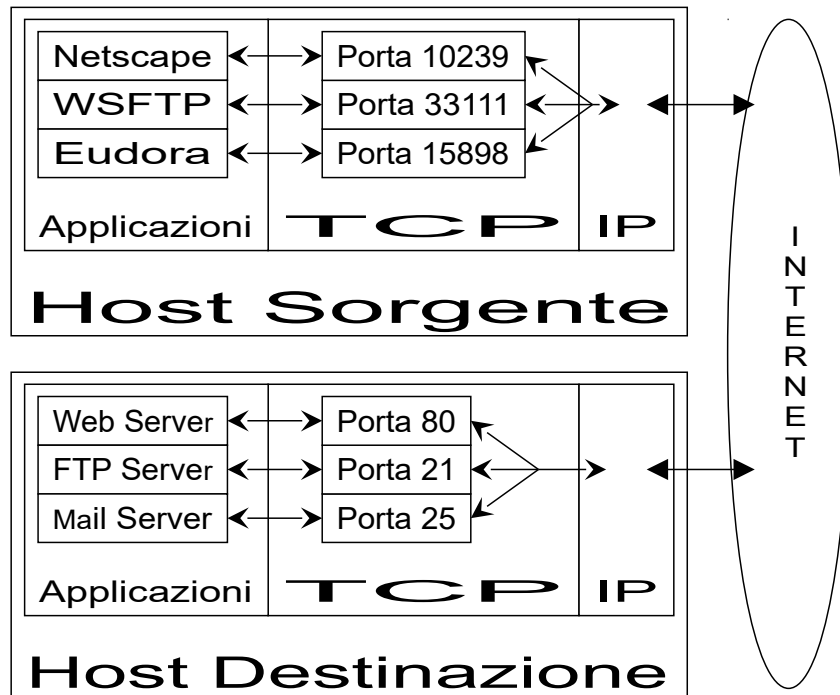
Multiplexing

- Il multiplexing è la funzione svolta sia da TCP che da UDP è per consentire l'indirizzamento di applicazioni specifiche all'interno di un host
 - IP fornisce un canale trasmissivo tra due host, TCP ed UDP permettono a più applicazioni di usare questo canale contemporaneamente-
 - Questo servizio viene offerto associando ogni applicazione ad una *porta*.
 - Una porta è indicata da un numero compreso tra 0 e 65535.
 - In ogni istante, su di un host, ciascuna porta non può essere utilizzata da più di un'applicazione.
- Per raggiungere un'applicazione su di un host, bisogna conoscere la porta associata. Per questo i servizi di ampio interesse sono collocati su porte standard, dette *well-known ports*

EnAIP Tecnico Informatico



Multiplexing



EnAIP Tecnico Informatico



TCP

- UDP è un protocollo connection-less, ovvero i messaggi vengono inviati senza accordi preliminari, di conseguenza è molto efficiente per l'invio di piccole quantità di dati la cui perdita non è critica (es.: NTP, domain)
- TCP è un protocollo connection-oriented, ovvero:
 - ogni scambio di messaggi è preceduto da una fase preliminare in cui le due parti concordano di voler comunicare tra loro, che permette di marcare i pacchetti come appartenenti ad una precisa connessione e di numerarli in sequenza
 - di conseguenza:
 - richiede un tempo iniziale aggiuntivo per stabilire la connessione
 - è affidabile perché rileva la perdita di pacchetti
 - esegue il riordino dei pacchetti che arrivano fuori sequenza

EnAIP Tecnico Informatico



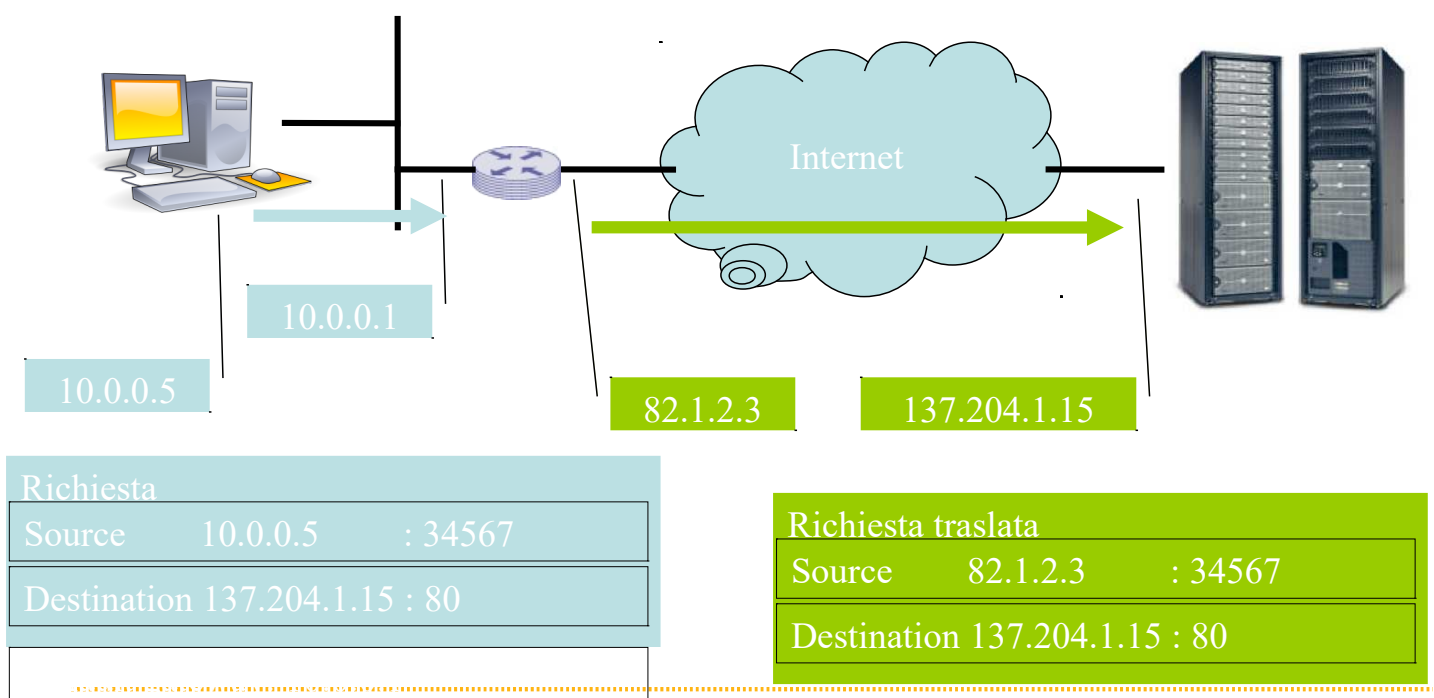
Network Address Translation (NAT)

- La prospettiva di un esaurimento degli IP disponibili ha fatto esplodere l'utilizzo di una tecnica che consente di utilizzare un solo indirizzo pubblico, senza rinunciare alla possibilità di realizzare in tecnologia TCP/IP una rete anche di grandi dimensioni e di connetterla ad Internet.
- L'efficacia della tecnica si basa sull'osservazione che la gran parte degli host è client e non server
 - non necessitano di essere raggiunti da richieste
 - originano richieste e devono poter essere raggiunti dalle risposte

EnAIP Tecnico Informatico



Network Address Translation (NAT)



EnAIP Tecnico Informatico



Network Address Translation (NAT)

- La quintupla
(*protocollo, ip_sorgente, porta_sorgente, ip_destinazione, porta_destinazione*)
identifica univocamente una connessione
- Nel NAT molti IP sorgente vengono sostituiti dall'unico IP pubblico del router
 - possibilità di modificare la porta sorgente per disambiguare le connessioni originate con tutti i parametri identici a parte l'IP sorgente
 - memorizzazione delle traslazioni per poter riconoscere il destinatario delle risposte

Source IP	Source port	Router IP	Router port	Dest. IP	Dest. port
10.0.0.5	11111	82.1.2.3	11111	137.204.1.15	80
10.0.0.9	11111	82.1.2.3	11111	137.204.1.15	80

EnAIP Tecnico Informatico



Network Address Translation (NAT)

- La quintupla
(*protocollo, ip_sorgente, porta_sorgente, ip_destinazione, porta_destinazione*)
identifica univocamente una connessione
- Nel NAT molti IP sorgente vengono sostituiti dall'unico IP pubblico del router
 - possibilità di modificare la porta sorgente per disambiguare le connessioni originate con tutti i parametri identici a parte l'IP sorgente
 - memorizzazione delle traslazioni per poter riconoscere il destinatario delle risposte

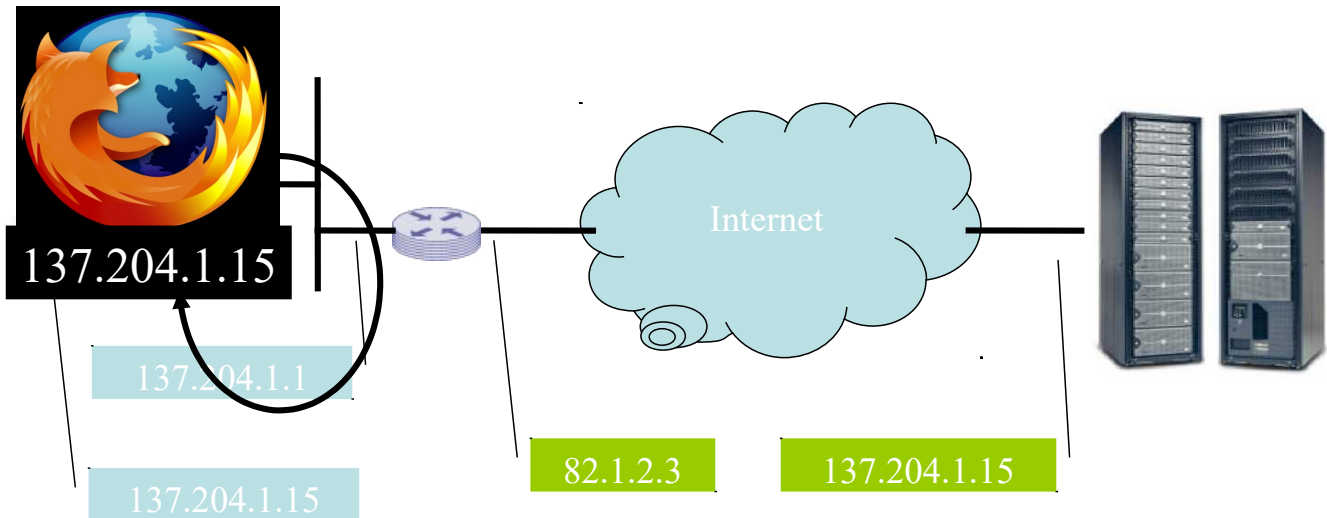
Source IP	Source port	Router IP	Router port	Dest. IP	Dest. port
10.0.0.5	11111	82.1.2.3	11111	137.204.1.15	80
10.0.0.9	11111	82.1.2.3	22222	137.204.1.15	80

EnAIP Tecnico Informatico



Network Address Translation (NAT)

- Gli IP della rete interna risultano del tutto nascosti
- Cosa capiterebbe scegliendoli arbitrariamente?



EnAIP Tecnico Informatico



IP privati (RFC 1918)

- Per evitare il problema del possibile "oscuramento" di IP validi, uno standard definisce alcuni intervalli di indirizzi che non possono essere utilizzati su Internet
 - 10.x.y.z
 - 172.16.x.y ... 172.31.x.y
 - 192.168.x.y

EnAIP Tecnico Informatico



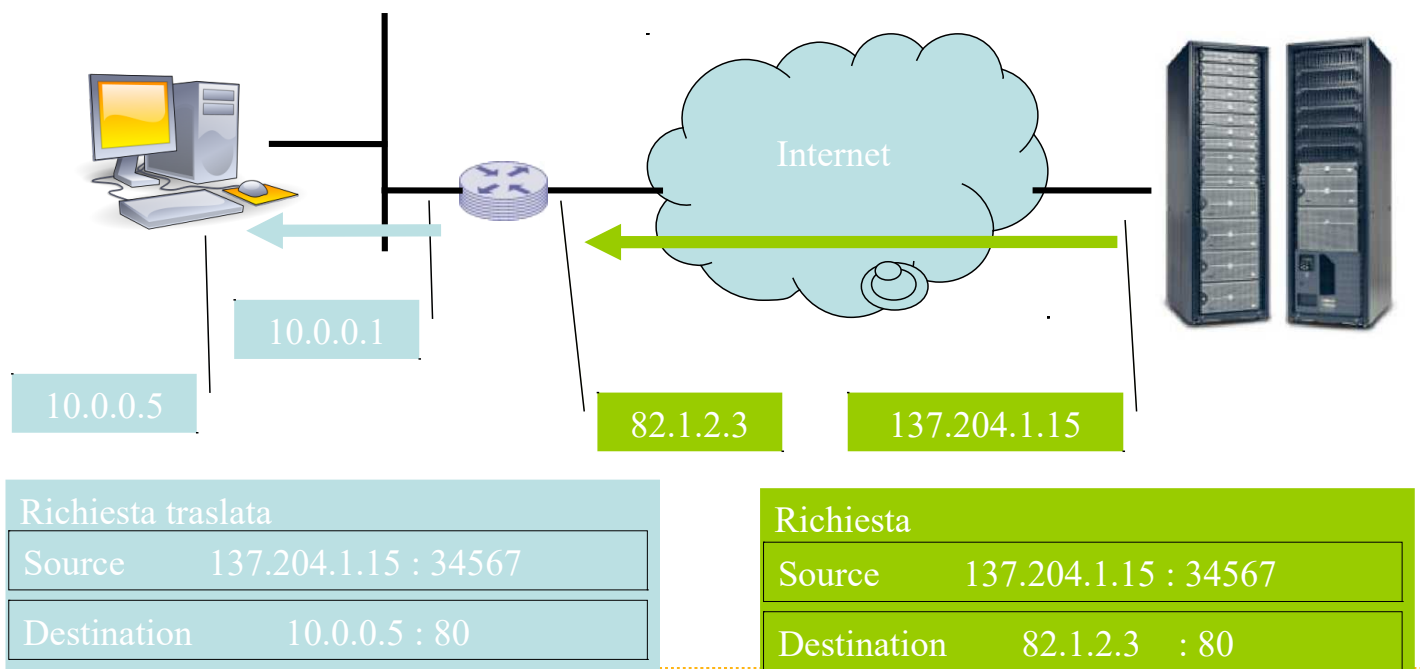
SNAT / DNAT

- Per poter utilizzare una rete di client con un solo IP pubblico si modifica l'IP sorgente → Source NAT (SNAT)
 - comportamento trasparente ed automatico
- Lo stesso dispositivo consente anche di rendere raggiungibili alcuni host della rete privata, modificando l'indirizzo di destinazione quando riceve richieste dall'esterno → Destination NAT (DNAT)
 - la mappatura tra porta (servizio) di destinazione ed host interno a cui inoltrare la richiesta va esplicitamente configurata

EnAIP Tecnico Informatico



DNAT



EnAIP Tecnico Informatico



Aspetti gestionali

- La concessione degli IP è compito dello IANA.
 - <http://www.iana.org/ipaddress/ip-addresses.htm>
- All'atto pratico vengono affidati blocchi di IP a grandi gestori (Regional Internet Registrar - RIR), che a loro volta ne assegnano sottoinsiemi ai gestori locali (Local Internet Registrar - LIR), come ad esempio i maggiori ISP (Internet Service Provider). I RIR sono:
 - AfriNIC serving Africa
 - APNIC serving the Asia Pacific region
 - ARIN serving North America
 - LACNIC serving South America and the Caribbean
 - RIPE NCC serving Europe, Central Asia and the Middle East.

EnAIP Tecnico Informatico



Configurazione di Linux

- Si può verificare la configurazione delle interfacce di rete con il comando *ifconfig*
- ```
eth0 Link encap:Ethernet HWaddr 00:0F:B0:93:FC:D5
 inet addr:137.204.57.118 Bcast:137.204.57.255 Mask:255.255.255.0
 UP BROADCAST MULTICAST MTU:1500 Metric:1
```
- Si può verificare la configurazione delle tabelle di routing con il comando *route*:

### Kernel IP routing table

| Destination  | Gateway        | Genmask       | Flags | Metric | Ref | Use | Iface |
|--------------|----------------|---------------|-------|--------|-----|-----|-------|
| 137.204.57.0 | 0.0.0.0        | 255.255.255.0 | U     | 0      | 0   | 0   | eth0  |
| 0.0.0.0      | 137.204.57.254 | 0.0.0.0       | UG    | 0      | 0   | 0   | eth0  |

---

EnAIP Tecnico Informatico



# Diagnostica della connettività IP

## ■ ping

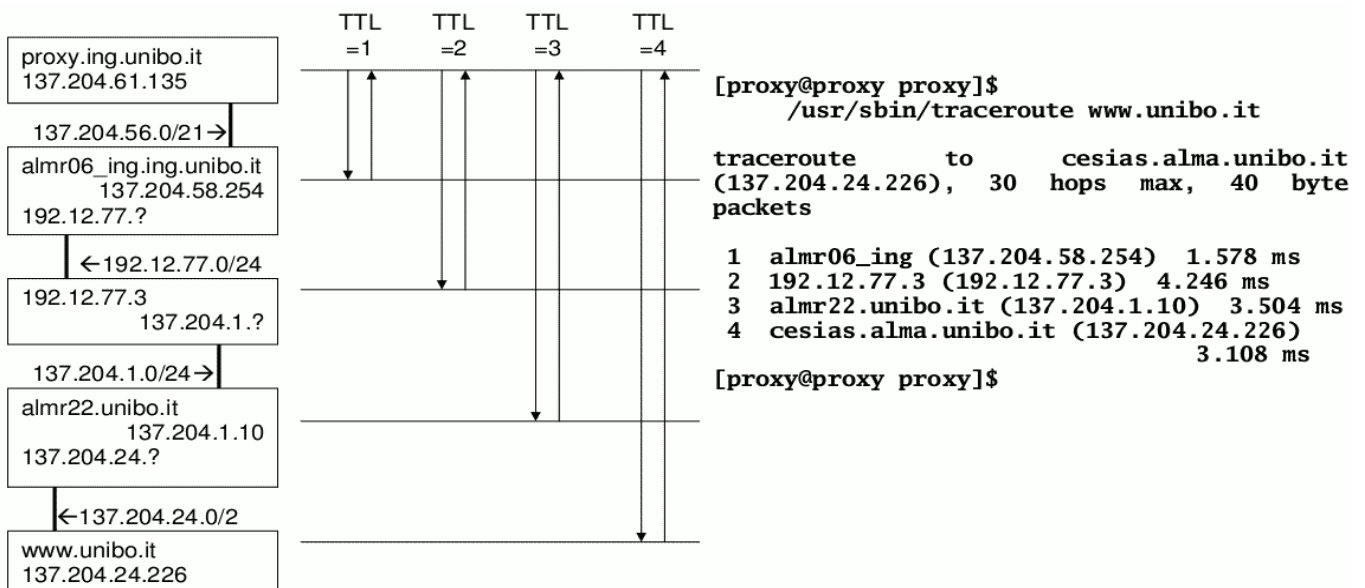
```
sansone.ing.unibo.it%~% ping -s www.netscape.com
PING www-de1.netscape.com: 56 data bytes
64 bytes from 194.25.242.193: icmp_seq=2. time=172. ms
64 bytes from 194.25.242.193: icmp_seq=6. time=165. ms
64 bytes from 194.25.242.193: icmp_seq=7. time=285. ms
64 bytes from 194.25.242.193: icmp_seq=8. time=371. ms
^C
----www-de1.netscape.com PING Statistics----
10 packets transmitted, 4 packets received, 60% packet loss
round-trip (ms) min/avg/max = 165/248/371
sansone.ing.unibo.it%~%
```

EnAIP Tecnico Informatico



# Diagnostica della connettività IP

## ■ traceroute



EnAIP Tecnico Informatico



# Servizi

---

EnAIP Tecnico Informatico



## Il modello client-server

- Con il termine *server* indichiamo un'applicazione che rende disponibile, mediante un'interfaccia standard, un servizio.
- Con il termine *client* indichiamo un'applicazione in grado di utilizzare i servizi messi a disposizione da un server.

Attenzione: a volte gli stessi termini vengono usati impropriamente per indicare l'host che ospita l'applicazione

- Il modello CS centralizza dati e metodi di elaborazione
  - evita duplicazioni, incoerenze di versione, proliferazione di interfacce
  - introduce problemi di disponibilità e prestazioni

---

EnAIP Tecnico Informatico





# Domain Name Service (DNS)

- Gli host sono individuati da indirizzi IP, con cui gli apparati lavorano bene perché sono numeri binari.
- Gli esseri umani faticano a ricordare dei numeri, preferiscono usare un nome significativo della funzione dell'host.
- È utile disporre di un servizio che associ un nome all'indirizzo
- Ogni host viene identificato per mezzo del proprio nome (*hostname*) e del nome del proprio dominio (*domain name*).
- I domini sono definiti in maniera gerarchica.
- I componenti della gerarchia sono separati con un punto  
es.: `www.deis.unibo.it`

---

EnAIP Tecnico Informatico



## Top Level Domain (TLD)

- Il primo componente a destra (la parte più elevata della gerarchia) è detto TLD (Top Level Domain) e deve essere scelto essenzialmente:
  - fra i TLD nazionali:
    - sigle di due lettere concesse solo alle nazioni come definite e riconosciute dall'ONU
    - es.: `.it`, `.uk`, `.fr`, `.de`, `.tv`, ...
  - fra le sigle identificative di determinate categorie (gTLD):
    - dal 1985 esistono: `.com`, `.org`, `.net`, `.edu`, `.gov`, `.mil`, `.int`
    - dal 2001 anche: `.aero`, `.biz`, `.coop`, `.info`, `.name`, `.museum`, `.pro`,
    - dal 2005 anche: `.jobs`, `.travel`, `.cat`
    - dal 2006 anche: `.mobi`
    - Poi `.asia`, `.tel`, `.xxx`, ma altri lasciati in attesa di decisione per anni
  - La critica dell'arbitrarietà delle procedure per ottenere l'attivazione di un gTLD ha infine portato alla possibilità di richiedere gTLD qualsiasi (decisione 2008, avvio 2012, attivazione 2013)
    - 185.000\$ + 25.000\$/anno

---

EnAIP Tecnico Informatico



## Aspetti gestionali

- La gestione dei TLD è in carico allo IANA.
  - <http://www.iana.org/domain-names.htm>
- All'atto pratico la gestione di un dominio viene esercitata per mezzo di deleghe ai gestori dei sottodomini:
  - sponsor o operatori dei TLD generici
    - .com --> Verisign, .aero --> SITA, ...
  - enti nazionali responsabili dei TLD geografici
    - .it --> IIT-CNR, ...
  - qualsiasi ente o persona responsabile dei domini al di sotto dei TLD (domini di 2° livello)

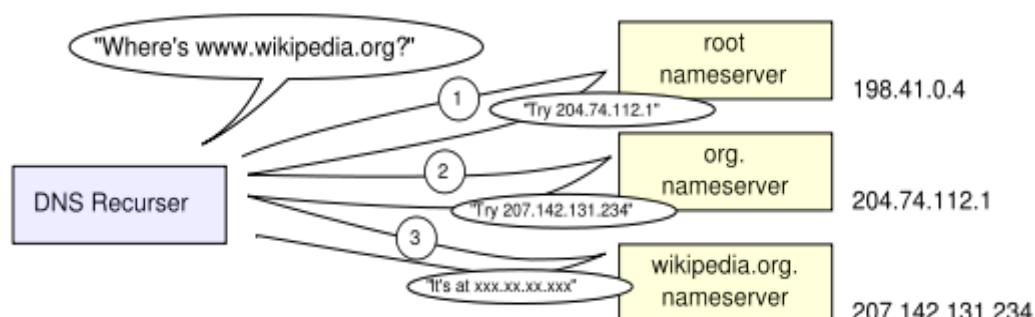
---

EnAIP Tecnico Informatico



## Risoluzione dei nomi in indirizzi

- Il DNS è un sistema client-server.
  - Il client prende il nome di *resolver*, e rivolge le proprie interrogazioni (*query*) tramite il protocollo UDP ad un server in ascolto sulla porta 53.
  - Il server può farsi carico del reperimento dell'indirizzo richiesto, o solo di fornire un puntatore da cui proseguire le ricerche



---

EnAIP Tecnico Informatico



## Risoluzione inversa

- Per la mappatura inversa (da IP a nome) si usa lo stesso meccanismo con un “trucco”: si sfrutta la gerarchia degli IP e si definisce un spazio dei nomi con origine in-addr.arpa. Si noti che
  - gli IP sono scritti dal byte che rappresenta la rete più ampia a quello che rappresenta la parte più specifica dell'indirizzo
    - Es: 137.204.57.1 == RIPE --> UNIBO --> DEIS --> PROMET1
  - i nomi sono scritti a partire dal nome specifico dell'host verso il TLD che rappresenta il dominio più ampio nella gerarchia
    - [www.deis.unibo.it](http://www.deis.unibo.it)
- quindi per usare la stessa infrastruttura la ricerca di 137.204.57.1 viene effettuata con una query del nome 1.57.204.137.in-addr.arpa

---

EnAIP Tecnico Informatico



## Identificazione dei domini e degli IP

- I registri degli indirizzi IP e dei nomi di dominio sono pubblicamente consultabili per poter risalire ai titolari
  - utilizzo delle query DNS per associare nome <--> indirizzo
  - utilizzo dei servizi *whois* per identificare i responsabili tecnici e legali del domino o del blocco di indirizzi
    - i registri non sono integrati a livello globale
    - utilizzo dei servizi whois del RIR appropriato per l'indirizzo
    - utilizzo dei servizi whois del registrar appropriato per il TLD

---

EnAIP Tecnico Informatico



# Posta elettronica

La posta elettronica (e-mail) è stato uno dei primi servizi introdotti su ARPAnet, ed è tuttora uno dei più utilizzati su Internet.

I problemi affrontati per la ricezione dei messaggi sono:

- È richiesta l'identificazione univoca del destinatario
  - □ indirizzo costruito in modo gerarchico: utente@host.dominio
- È necessario poter depositare i messaggi fino alla effettiva lettura
  - casella postale, ovvero spazio su di un server di posta
- È utile un sistema di trasporto che tolleri la temporanea irraggiungibilità del server di destinazione
  - ridondanza, per ogni destinazione sono indicati punti di appoggio alternativi (inizialmente una vera “rete sulla rete”)

---

EnAIP Tecnico Informatico



# Posta elettronica

Il reperimento delle informazioni per individuare i server di posta di un dominio è affidato al DNS - stretta integrazione tra i sistemi

I problemi affrontati per la spedizione dei messaggi sono:

- È desiderabile che l'utente veda un'interazione veloce
  - queuing, il server accetta i messaggi e li accoda per spedirli
- L'utente non deve essere obbligato a mantenere la connessione al server durante la lettura e la composizione dei messaggi
  - client di posta elettronica e protocolli di colloquio col server

---

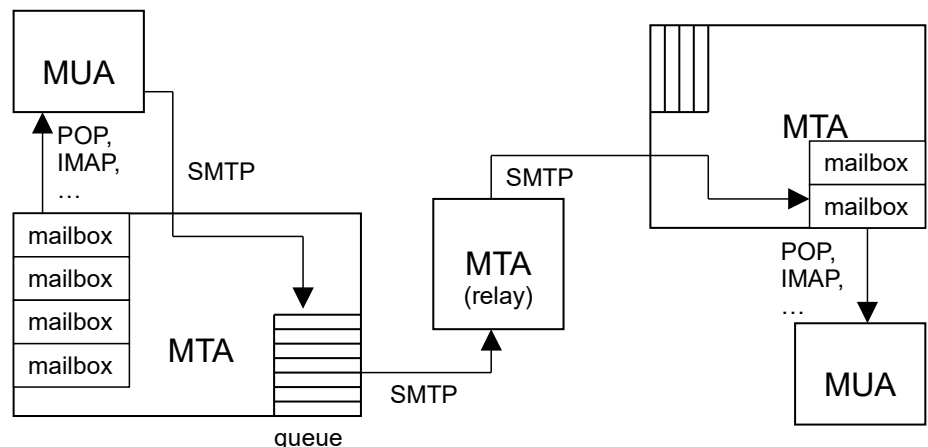
EnAIP Tecnico Informatico



# Posta elettronica

## ■ Applicazioni:

- MUA (Mail User Agent): Client (es. Eudora)
- MTA (Mail Transport Agent): Server di transito e smistamento



## ■ Protocolli:

- SMTP (Simple Mail Transfer Protocol), via TCP alla porta 25
- POP (Post Office Protocol), via TCP alla porta 110
- IMAP (Internet Mail Access Protocol), via TCP alla porta 143



# Posta elettronica

- Originariamente pensata per trasferire solo testi
  - *header*, contenente le informazioni per la consegna del messaggio (destinatario, mittente, soggetto, data, ...)
  - *body*, contenente il testo del messaggio.
- Presto riconosciuta come utile per trasferire dati di ogni genere
  - estendere le possibilità di codifica del body con lo standard MIME (Multipurpose Internet Mail Extension)
  - Il corpo della mail rimane in formato ascii, ma in esso sono codificate le informazioni binarie originali (attach).
    - Possibilità di spedire più dati distinti in un solo messaggio
    - Possibilità di spedire una indicazione del tipo di dato platform-independent
    - Consente di definire nuovi tipi di dato
  - del tutto trasparente per i server



# Formato di un messaggio

```
From: luca.ghedini@mail.ing.unibo.it Wed Nov 12 23:06:49 1997
Received: (from ghedo@localhost)
 by mail.ing.unibo.it (8.8.8/8.8.5) id XAA04169
 for luca.ghedini; Wed, 12 Nov 1997 23:06:49 +0100 (MET)
Date: Wed, 12 Nov 1997 23:06:49 +0100 (MET)
From: Luca Ghedini <luca.ghedini@mail.ing.unibo.it>
To: luca.ghedini
Subject: Saluti
Mime-Version: 1.0
Content-Type: multipart/mixed;
boundary=6e57_3bae-6e62_7e55-1cf_41
Content-Length: 590

--6e57_3bae-6e62_7e55-1cf_41
Content-Type: text/plain; charset=us-ascii
Content-Transfer-Encoding: 7bit
Content-MD5: DjPy0vmsowwWAIx7HqbxRg==
X-Sun-Data-Type: text

Ciao, ti mando le cose che mi avevi chiesto
--6e57_3bae-6e62_7e55-1cf_41
Content-Type: image/gif; name=2torri.gif
Content-Transfer-Encoding: base64
Content-MD5: LcOMV2MEgHKz/Ufm58vvlQ==
Content-Description: 2torri.gif
Content-Disposition: attachment; filename=2torri.gif
X-Sun-Data-Type: gif-file

R0lGODlh2AA0AfcAACAgQCAgAg/yBAQCBawCBA/yBgQCBggCBgwCBg/yCAQCCA
cmtzIEluYy4NDVVzZSBubyBob29rcwA7
--6e57_3bae-6e62_7e55-1cf_41--
```

---

EnAIP Tecnico Informatico

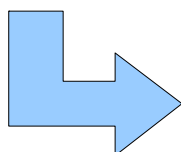


## World Wide Web

Il problema: recuperare informazioni in un contesto altamente eterogeneo.

Cosa esisteva prima del web ?

- programmi specializzati per accedere a informazioni codificate in differenti maniere.
- difficoltà di integrazione delle informazioni provenienti da fonti non omogenee
- interfacce di accesso non consistenti
  - Costi elevati
  - Difficoltà di apprendimento
  - Rapida obsolescenza dei programmi
  - Necessità di “aggiustamenti” manuali



---

EnAIP Tecnico Informatico



# Scopi del WWW (CERN 1989)

- differenti rappresentazioni delle informazioni (es. per le immagini : GIF, JPEG, BMP,...)
- differenti modalità di recupero delle informazioni (database query, ftp site, ...)
- differenti modalità di autenticazione (password, chiavi pubbliche, DES,...)
- trasparenza accesso e allocazione
- presentazione multimediale
- unicità di interfaccia utente

---

EnAIP Tecnico Informatico



## Ipertesti

L'elemento base del www è l'ipertesto.

- Un ipertesto e' un testo arricchito di immagini, suoni, e riferimenti ad altre informazioni (link)
- Ciascun link punta ad un'altra informazione che può essere ovunque (World Wide)
- L'insieme dei link forma una rete (Web) in cui si incrociano i percorsi logici secondo cui l'informazione può essere esplorata.
- Il risultato è un ipertesto distribuito su una rete di calcolatori

---

EnAIP Tecnico Informatico



# Componenti del WWW

- **Browser**
  - accetta le richieste dell'utente e gli presenta i risultati della esplorazione
- **Server:**
  - Immagazzina gli ipertesti, elabora le richieste ricevute dai browser ed invia loro i risultati della elaborazione
- **Helper**
  - moduli per aggiungere ai browser particolari funzionalità
- **Common Gateway Interface/Server Side Scripting**
  - permettono agli utenti di eseguire programmi sui server (esecuzione remota).

---

EnAIP Tecnico Informatico



# Standard relativi al WWW

Occorrono degli standard:

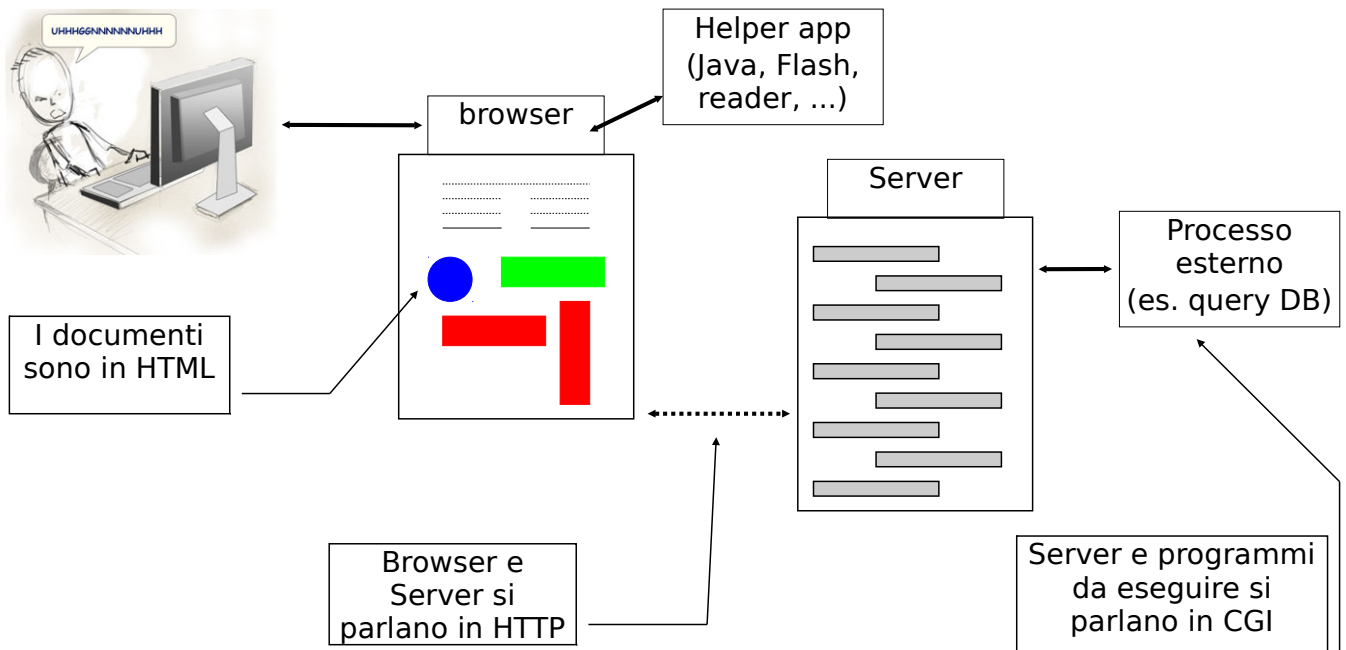
- per identificare le risorse in rete
  - Uniform Resource Identifier/Locator (URI/URL)
- per far comunicare server e browser
  - HyperText Transfer Protocol (HTTP)
- per permettere al browser di presentare i documenti all'utente
  - HyperText Markup Language (HTML)
  - Cascading Style Sheets (CSS)
- per gestire l'esecuzione lato server
  - Common Gateway Interface (CGI)

---

EnAIP Tecnico Informatico



# Standard relativi al WWW



EnAIP Tecnico Informatico

## Transazione HTTP

- Il protocollo è il più semplice possibile
  - V. 1.1 (RFC 2068)
  - REQUEST/RESPONSE
    - nessuna memoria del passato influenza le risposte successive
  - Capacità di negoziazione
    - il browser indica che tipi di dato, lingue, codifiche è in grado di gestire
    - è possibile elencare alternative in ordine di preferenza
  - Riutilizzo degli standard
    - messaggi codificati secondo MIME

GET /pippo.gif HTTP/1.0



```
HTTP/1.0 200 OK
Date: Sunday, 20-Apr-1995 20:23:57 GMT
Server: apache/1.2
MIME-version: 1.0
Content-type:image/gif
last-modified: Sunday, 20-Apr-1995
20:23:57 GMT
Content-lenght: 2245

.
. (dati della immagine)
.
```

EnAIP Tecnico Informatico



# HTTP Request

- 2 parti separate da una riga vuota:
  - header                      <metodo> <URL> <versione>  
                                    <opzioni>
  - body                         (opzionale) dati relativi alla richiesta
- Metodo:                      Es. GET, HEAD, POST, PUT
- URL                           solo la parte relativa (tolto protocollo, host, porta)
- Versione                      HTTP/1.0 o HTTP/1.1
- Opzioni                       Es. Accept:..., If-modified-since:..., Referer:...,  
                                    User-Agent:...,

---

EnAIP Tecnico Informatico



# HTTP Response

- 2 parti separate da una riga vuota:
  - header                      <status code> <status description>  
                                    <opzioni>
  - body                         <risposta>
- Status:                       1xx                      Information  
                                    2xx                      Success  
                                    3xx                      Redirection  
                                    4xx                      Client Error  
                                    5xx                      Server Error
- Opzioni                       Es. Content-type:..., Expires:...,  
                                    Content-encoding:...,

---

EnAIP Tecnico Informatico



# Web Server

- Il web server di base è un *mappatore di URL su risorse locali del server*
  - Risorse locali *statiche* (HTML, CSS, immagini ed altri media, programmi da eseguire lato client, ...)
    - Può effettuare operazioni sofisticate di negoziazione della versione da erogare, compressione, definizione degli header, ecc..., ma di base deve semplicemente trasformare il path della URL in un path del filesystem
    - Es: `http://www.example.com/docs/faq.html` →  
`/var/www/sites/example.com/pages/docs/faq.html`
  - Risorse locali *dinamiche*
    - vengono generate su richiesta
    - il risultato può essere un qualunque formato supportato dal browser, esattamente come per le risorse statiche
    - In questo caso la mappatura è tra URL e programma
      - Serve un metodo per codificare i parametri tra browser e server
      - Serve un metodo per passare i parametri tra server e programma

---

EnAIP Tecnico Informatico



## Installazione di Apache

- I sorgenti di Apache possono essere scaricati da <http://httpd.apache.org/download.cgi>
- Supponendo di selezionare la più recente release della versione 2.0, otterremo il file *httpd-2.0.59.tar.bz2*
  - scegliamo e creiamo una directory per i sorgenti, es. ~/src, entriamo in tale directory ed estraiamo l'archivio scaricato:

```
mkdir ~/src
cd ~/src
tar xvfj ~/Desktop/httpd-2.0.59.tar.bz2
```
  - entriamo nella directory risultante dal questo comando

```
cd httpd-2.0.59
```
  - consultiamo le opzioni di configurazione disponibili

```
./configure --help | less
```

---

EnAIP Tecnico Informatico



# Compilazione, installazione e test di Apache

- Supponendo che la configurazione di default sia accettabile, ed in particolare avendo verificato la destinazione in cui il software verrà collocato (/usr/local/apache2), lanciamo semplicemente  
`./configure`
- Successivamente, per compilare basta lanciare  
`make`
- Dal momento che solitamente il software viene installato in una directory di sistema, la cui modifica è proibita agli utenti comuni, il comando di installazione deve essere lanciato con i privilegi di amministrazione:  
`sudo make install`
- Si può immediatamente verificare se il server funziona lanciando  
`/usr/local/apache2/bin/apachectl -k start`  
e puntando un browser della macchina virtuale a `http://127.0.0.1/`

---

EnAIP Tecnico Informatico



## Architettura di Apache

- Apache è un server modulare, offre cioè un set di funzionalità di base e specifica uno standard che consente di inserirne di nuove per mezzo di frammenti di codice scritti appositamente (moduli)
  - Un modulo può essere compilato indipendentemente da Apache
  - Un modulo compilato può non essere attivato nel file di configurazione
    - occupa solo lo spazio disco relativo al file che contiene il suo eseguibile
  - Un modulo attivato entra a far parte integrante del processo server
    - occupa ram anche se non viene mai utilizzato
- Come ogni server web, Apache è essenzialmente un “mappatore” di indirizzi internet in nomi di file, e serve
  - pagine statiche: legge il contenuto del file e lo invia al browser correttamente inserito in un pacchetto HTTP
  - pagine dinamiche: esegue il file ed invia al browser il risultato

---

EnAIP Tecnico Informatico



# Configurazione di Apache

- I parametri essenziali di configurazione riguardano quindi:
  - come individuare sul disco i file relativi all'URL richiesta
  - come distinguere le pagine statiche da quelle dinamiche
- I parametri sono interpretati dai moduli, per cui il caricamento di un modulo rende possibile inserirli nella configurazione, mentre la sua assenza rende i corrispondenti parametri incomprensibili al server
- I parametri possono essere inseriti in diversi contesti
  - globale
  - virtual host
  - directory/file
  - htaccess

---

EnAIP Tecnico Informatico



# Configurazione di Apache

- Alcuni parametri essenziali per erogare contenuti
  - **ServerName**      *nome*      (server)
    - specifica il nome canonico del server usato per auto-identificarsi
  - **DocumentRoot**    *directory*      (server, vhost)
    - specifica la directory che verrà fatta corrispondere alla URL "/"
    - Es: **DocumentRoot** */var/www/html*
      - alla richiesta di *http://127.0.0.1/test.html*  
verrà servito il file */var/www/html/test.html*
  - **Alias**      *url*      *directory*      (server, vhost)
    - specifica una directory che verrà fatta corrispondere alla URL indicata
    - Es: **Alias** */icons/* */var/images/new/*
      - alla richiesta di *http://127.0.0.1/icons/up.png*  
verrà servito il file */var/images/new/up.png*

---

EnAIP Tecnico Informatico



# Configurazione di Apache

## ■ Alcuni parametri essenziali per erogare contenuti

### – **DirectoryIndex** *file*

- Specifica di usare file per rispondere a richieste che riguardano una directory
- Es: **DocumentRoot** */var/www/html*  
**DirectoryIndex** *welcome.html*  
→ alla richiesta di *http://127.0.0.1/*  
verrà servito il file */var/www/html/welcome.html*

## ■ Alcuni parametri essenziali per il monitoraggio

### – **LogFormat** *parametri* *nomeformato*

- Specifica che informazioni mettere nei log di tipo “nomeformato”

### – **Customlog** *nomefile* *nomeformato*

- Attiva il logging su nomefile delle informazioni definite da nomeformato

### – **ErrorLog** *file*

- specifica il file in cui registrare qualsiasi evento anomalo o errore

EnAIP Tecnico Informatico



# Configurazione di Apache

- Per poter realizzare siti *dinamici*, in cui le pagine siano cioè generate da un programma all'atto della loro richiesta, bisogna istruire il server web perchè sappia che il file corrispondente all'URL è un programma da eseguire e non semplicemente uno stream di dati da inviare al browser.

## ■ Esistono due modalità di esecuzione:

### – CGI (common gateway interface)

- In questo caso il server lancia un programma esterno che si occupa della generazione dell'intera risposta, header HTTP compresi

### – Server Side Scripting

- In questo caso il server viene dotato, tipicamente grazie ad un modulo, della capacità di interpretare codice inserito in una pagina che può contenere sia parti statiche (che il server invia al browser inalterate), sia parti dinamiche, il cui effetto è che il server invii al browser il risultato dell'esecuzione di codice



## Esempio di esecuzione CGI

- La direttiva `ScriptAlias` indica al server che le pagine riferite ad una certa URL sono da generare lanciando i corrispondenti programmi CGI:
  - `ScriptAlias /cgi-bin/ /usr/local/apache2/cgi-bin/`
- Supponiamo che il file `/usr/local/apache2/cgi-bin/data` sia eseguibile e contenga queste istruzioni:

```
#!/bin/bash
echo Content-type: text/html
echo
echo '<html><body><h1>'
/bin/date
echo $
echo '</h1></body></html>'
```
- Invocando l'URL <http://127.0.0.1/cgi-bin/data> cosa accade?

---

EnAIP Tecnico Informatico



## Configurazione di Apache

- Altre direttive riguardano:
  - il controllo dell'accesso (per IP, dominio, utente, ...)
  - il logging delle pagine richieste
  - la negoziazione delle forme alternative per i contenuti
  - la generazione degli header HTTP
  - il virtual hosting
  - le connessioni sicure (HTTPS)
- Per un elenco esaustivo dei moduli disponibili nella distribuzione standard di Apache e delle loro direttive di configurazione si visiti:  
<http://httpd.apache.org/docs/>

---

EnAIP Tecnico Informatico



# Installazione di PHP

- I sorgenti di PHP possono essere scaricati da <http://www.php.net/downloads.php>
- Supponendo di selezionare la versione più recente (5.1.6), otterremo il file *php-5.1.6.tar.bz2*
  - scegliamo e creiamo una directory per i sorgenti, es. *~/src*, entriamo in tale directory ed estraiamo l'archivio scaricato:

```
mkdir ~/src
cd ~/src
tar xvfj ~/Desktop/php-5.1.6.tar.bz2
```
  - entriamo nella directory risultante dal questo comando

```
cd php-5.1.6
```
  - consultiamo le opzioni di configurazione disponibili

```
./configure --help | less
```

---

EnAIP Tecnico Informatico



## Configurazione dei sorgenti di PHP

- Esaminando i parametri di configurazione disponibili, bisogna notare che:
    - è opportuno specificare in che modo PHP si debba integrare in Apache. Nella sezione “SAPI modules” sono elencate le interfacce verso diversi server web; limitandosi ad Apache2 ed escludendo le soluzioni non stabili (EXPERIMENTAL) la scelta cade sull'opzione `--with-apxs2`.
    - è utile indicare un percorso di installazione più descrittivo rispetto al default (*/usr/local*)
    - è ragionevole per ora disattivare le estensioni che, pur essendo attivate di default, non si prevede di usare, specialmente perchè molte di queste richiederebbero l'installazione di ulteriore software di supporto
- Si ottiene quindi la riga di comando:
- ```
./configure --with-apxs2=/usr/local/apache2/bin/apxs
--prefix=/usr/local/php5 --disable-all
```

EnAIP Tecnico Informatico



Compilazione con ulteriori funzioni

Supponiamo di voler installare un prodotto basato su Apache/PHP che richieda che PHP supporti le estensioni mbstring, xml e zlib.

Il supporto può essere attivato includendo nel comando configure

```
--enable-mbstring=all --with-zlib --with-xml --with-iconv
```

ma ciò non è sufficiente, perchè PHP offre un supporto interno completo di alcune funzionalità, mentre in altri casi le direttive mostrate indicano solo a PHP di utilizzare funzionalità che devono già essere presenti sul sistema.

TEST PRATICO:

- rientrare nella directory dei sorgenti di PHP
- ripulire i risultati della precedente compilazione
`make clean`
- riconfigurare i sorgenti
`./configure --with-apxs2=/usr/local/apache2/bin/apxs
--enable-mbstring=all --with-zlib --with-xml --with-iconv`
- osservare il messaggio d'errore

EnAIP Tecnico Informatico



Compilazione con ulteriori funzioni

E' quindi necessario:

- individuare ed installare i pacchetti che forniscono le funzionalità mancanti, oppure gli analoghi sorgenti
- ripetere i passi di configurazione, compilazione ed installazione di PHP

In questo caso, i pacchetti necessari sono *libxml2-dev* e *zlib1g-dev* reperibili nella sezione “Libraries – development” di Synaptic

Si noti una particolarità del sistema a pacchetti: le funzionalità xml2 e zlib sono in realtà già presenti sul sistema (verificare nella sezione “Libraries”)

- se si dispone di un programma eseguibile che ne fa uso in modo corretto non serve altro
- se si vuole *compilare dai sorgenti* un programma che ne faccia uso servono le “interfacce di programmazione” fornite dai pacchetti “-dev”

EnAIP Tecnico Informatico



Compilazione dei sorgenti Installazione di PHP da pacchetti

- Una volta completata la fase di configurazione, per compilare ed installare basta lanciare gli stessi comandi visti per Apache

```
make
sudo make install
```
- Per installare direttamente da pacchetti della distribuzione:
 - su piattaforma Debian/Ubuntu
 - `apt-get install php5`
 - richiede la modalità *prefork* di Apache e rispettivo modulo
 - include i pacchetti essenziali per l'uso da Apache e a riga di comando
 - le funzioni aggiuntive attivabili durante la compilazione dei sorgenti sono disponibili come moduli in pacchetti separati (php5-...)
 - su piattaforma RedHat/CentOS
 - `yum install php`
 - come sopra (il repository *epe/* contiene molte decine di pacchetti software basati su php oltre a quelli standard della distribuzione)

EnAIP Tecnico Informatico



Attivazione di PHP

- La procedura di installazione modifica automaticamente il file di configurazione di Apache perchè all'avvio del web server venga attivato il modulo PHP. Si può verificare che nel file di configurazione di Apache è comparsa la linea

```
LoadModule      php5_module                modules/libphp5.so
```

Questo non è però sufficiente, poichè bisogna “insegnare” ad Apache quali file devono essere processati da tale modulo, aggiungendo la riga

```
AddType    application/x-httpd-php    .php
```

Complessivamente, queste direttive predispongono il server all'esecuzione di pagine PHP in modalità Server Side Scripting
- Per attivare il modulo si deve riavviare il server

EnAIP Tecnico Informatico



Test di PHP

- Avendo verificato che la DocumentRoot di default di Apache è `/usr/local/apache2/htdocs`, si crea un file in tale directory con un editor di testo:

```
sudo gedit /usr/local/apache2/htdocs/info.php
```

contenente le direttive

```
<?
phpinfo();
?>
```
- Caricando la pagina <http://127.0.0.1/info.php> si dovrebbe ottenere il resoconto di tutti i parametri di funzionamento dell'interprete, generato dinamicamente come risultato dell'istruzione `phpinfo`
- Per una descrizione dettagliata della procedura di installazione si faccia riferimento al file `INSTALL` nella directory dei sorgenti di PHP

EnAIP Tecnico Informatico



Configurazione di PHP

- Anche PHP è un software modulare. Il file di configurazione *php.ini*, contenuto nella directory *etc* (di sistema o collocata al di sotto del percorso di installazione scelto con *configure*) raccoglie, tra le altre:
 - direttive per il caricamento di moduli aggiuntivi
 - direttive per la limitazione dell'uso delle risorse di sistema
 - direttive per la configurazione del logging
- Il file di default è ampiamente commentato per rendere comprensibile il significato di ogni direttiva, i manuali dettagliati sono disponibili in rete agli indirizzi <http://it.php.net/manual/en/configuration.php> e <http://it.php.net/manual/en/ini.php>

EnAIP Tecnico Informatico

DNS – NTP

Breve descrizione e configurazione dei servizi su Linux

Marco Prandini

Risoluzione dei nomi - generalità

- La mappatura da nomi di host a indirizzi IP e viceversa è uno dei tanti casi in cui il sistema ha bisogno di un dizionario di nomi
- Il primo accorgimento adottato da GNU/Linux riguarda la *scelta della sorgente di informazioni*
 - *Name Service Switch*
 - svolta dalla libreria C di sistema
 - supporta un set fisso di possibili database
 - configurata tramite `/etc/nsswitch.conf`
 - vedi man page omonima

NSS

■ Sintassi di nsswitch.conf

- `<entry> ::= <database> ":" [<source> [<criteria>]]*`
- `<criteria> ::= "[" <criterion> + "]"`
- `<criterion> ::= <status> "=" <action>`
- `<status> ::= "success" | "notfound" | "unavail" | "tryagain"`
- `<action> ::= "return" | "continue"`

risposta
ricevuta

la sorgente esiste
ma non sa rispondere

la sorgente esiste
ma è occupata

la sorgente non
è raggiungibile

(i colori indicano
l'azione di default)

■ Es.

`passwd: files nis ldap`

`group: files ldap`

`hosts: ldap [NOTFOUND=return] dns files`



Risoluzione dei nomi – host e IP

`hosts: ldap [NOTFOUND=return] dns files`

■ **files** → la sorgente di informazioni è il file `/etc/hosts`

- formato: `<IP> <FQDN> [<ALIAS> ...]`
- esempio: `8.8.8.8 dns.google.com gdns`

■ **dns** → la sorgente di informazioni è il sistema DNS

- l'interrogazione di server DNS è un'ulteriore set di funzioni della libreria C di sistema, il *resolver*
- si configura attraverso `/etc/resolv.conf`
- esempio
`nameserver 137.204.58.1`
`domain disi.unibo.it`
`search ing.unibo.it`



DNS caching

■ Spesso si trova un server DNS locale

- Miglioramento prestazioni
- Maggiore flessibilità per contesti dinamici

```
~$ cat /etc/resolv.conf
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)
#     DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
nameserver 127.0.1.1
```

- Tutti gli IP che iniziano per 127 puntano a localhost

```
sudo ss -naup | grep 127.0.1.1:53
...
UNCONN      0          0      127.0.1.1:53      *:~      users:(( "dnsmasq",pid=2154,fd=4))
```

Risoluzione di nomi via NSS

■ Il comando getent permette di interrogare i database del name service switch

getent <db name> <keyword>

Esempi:

```
$ getent passwd las
las:x:1000:1000:Lab Amministrazione Sistemi,,,:/home/las:/bin/bash
```

```
$ getent hosts www.unibo.it
137.204.24.35   atrproxy.unibo.it www.unibo.it
```

Risoluzione nomi DNS diretta

- Per interrogare direttamente il DNS e avere più controllo sulle query si usano tipicamente `host` e `dig`
 - non considerano `nsswitch`
 - usano i `nameserver` di `resolv.conf` di default
 - possono interrogare un server specifico
- *host* (tipicamente per conversioni IP \longleftrightarrow nome)
 - query di un nome: `host www.unibo.it`
 - query a un server specifico: `host www.unibo.it 8.8.8.8`
- *dig* (tipicamente per ottenere informazioni legate a un dominio diverse da nomi `host`)
 - conoscere i Mail eXchanger: `dig mx example.com`
 - conoscere i Name Server: `dig ns example.com`

Sincronizzazione

- L'allineamento dell'ora di un sistema ad un orologio di riferimento è cruciale
 - per la diagnostica dei problemi (timestamp su log)
 - per i protocolli di autenticazione e autorizzazione (i messaggi hanno una vita limitata)
 - per la sincronizzazione di azioni distribuite
 - per il valore legale di azioni compiute attraverso i computer
- È possibile usare un protocollo che compensa i ritardi di rete per ottenere informazioni precise via Internet:
Network Time Protocol (NTP)
 - sito ufficiale: <http://www.ntp.org/>
 - grande quantità di informazioni su:
<http://www.eecis.udel.edu/~mills/ntp.html>

NTP in breve

■ Preciso

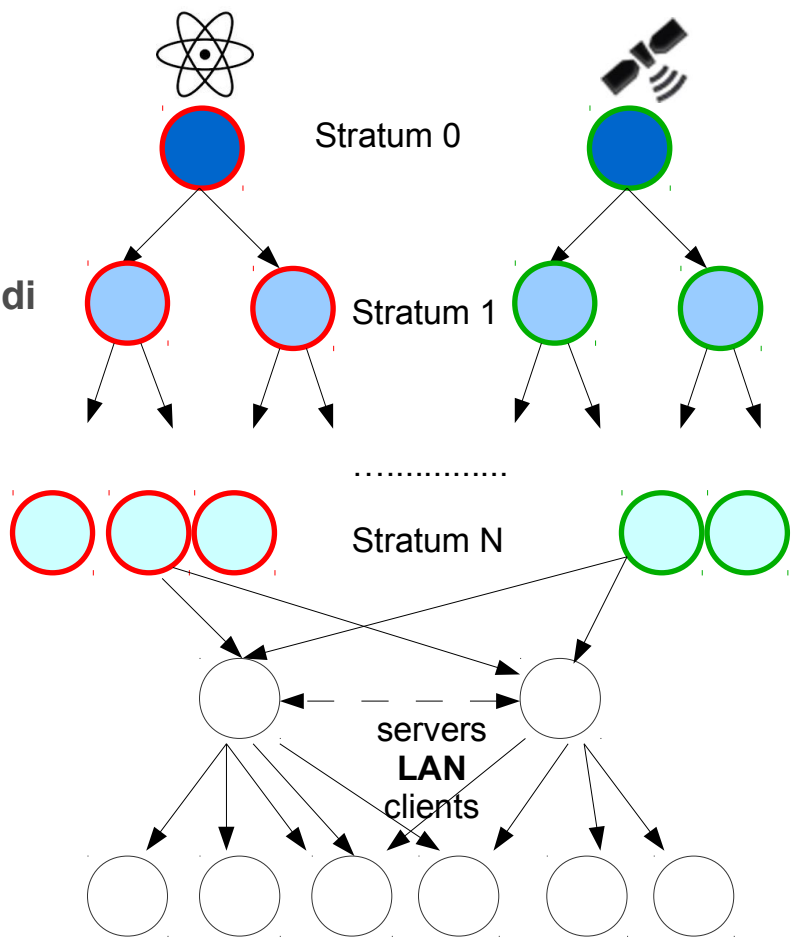
- poche decine di millisecondi di scarto su WAN
- <1 millisecondo su LAN
- supporto di sorgenti HW (oscillatori, GPS, ...)

■ Standard

- portato su ogni architettura nota

■ Scalabile e affidabile

- *multi-server*
- *strata*
- *peering*
- *auto-keying*



NTP su Linux

■ Il demone *ntpd* è client e/o server in funzione della configurazione

■ */etc/ntp.conf* – esempio

```
server 0.ubuntu.pool.ntp.org
```

```
server 1.ubuntu.pool.ntp.org
```

```
peer fellow.server.lan
```

```
# By default, exchange time with everybody, but don't allow configuration.
```

```
restrict -4 default kod notrap nomodify nopeer noquery
```

```
restrict -6 default kod notrap nomodify nopeer noquery
```

```
# Local users may interrogate the ntp server more closely.
```

```
restrict 127.0.0.1
```

```
restrict ::1
```


NTP – inizializzazione e uso sporadico

■ Il tool **ntpd** permette di sincronizzare l'orologio locale a un server NTP

- senza parametri usa i server in **ntp.conf**
 - **ntpd** non deve essere attivo
- accetta come parametro un server specifico

■ L'ora viene modificata in due modi

- se la differenza è più di 0.5 secondi: step
- se la differenza è meno di 0.5 secondi: slew con `adjtime()`

■ Non rimpiazza **ntpd**, che usa algoritmi sofisticati

- per compensare errori e ritardi dei pacchetti dai server
- per profilare il comportamento dell'orologio locale

