

# Reti di Calcolatori



## Il livello di rete

Università degli Studi di Verona  
Dipartimento di Informatica

Docente: [Damiano Carra](#)

## Acknowledgement

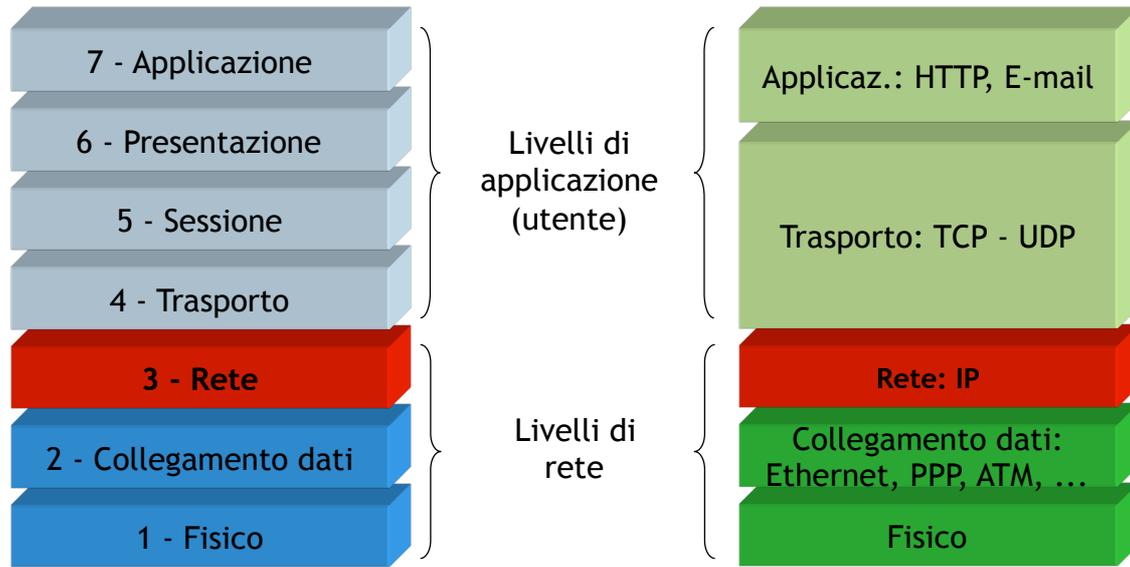
---

### ☐ Credits

- *Part of the material is based on slides provided by the following authors*
  - Jim Kurose, Keith Ross, "Computer Networking: A Top Down Approach," 4th edition, Addison-Wesley, July 2007
  - Douglas Comer, "Computer Networks and Internets," 5th edition, Prentice Hall
  - Behrouz A. Forouzan, Sophia Chung Fegan, "TCP/IP Protocol Suite," McGraw-Hill, January 2005



# Network level

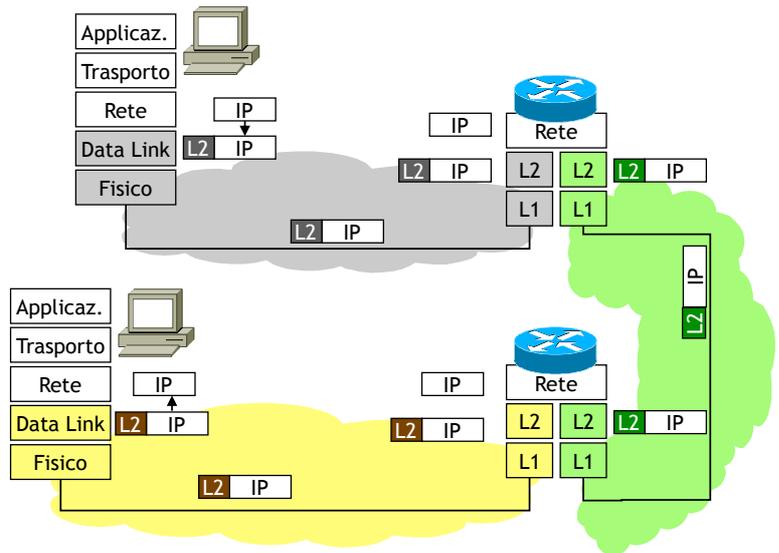


Introduzione



## Visione generale

- ❑ Trasporto dei segmenti dall'host sorgente all'host destinazione
- ❑ Lato sorgente, i segmenti vengono incapsulati in **datagrammi**
- ❑ Lato destinazione, i datagrammi vengono consegnati al livello di trasporto
- ❑ Il livello di rete e' presente in ogni end-host e ogni apparato intermedio (router)
- ❑ I router esaminano i campi dell'header presenti nei datagrammi IP



5



## Funzioni chiave del livello di rete

### ❑ Routing

- Determina il percorso che i pacchetti devono seguire dalla sorgente alla destinazione

#### ➔ Algoritmi di routing

- Analogia: processo di pianificazione di un viaggio, dalla partenza all'arrivo

### ❑ Forwarding

- Dato un router, trasferisce i pacchetti da una porta di input alla porta di out appropriata

- Analogia: nel caso di un viaggio, passaggio attraverso un incrocio, in cui si deve scegliere quale strada prendere

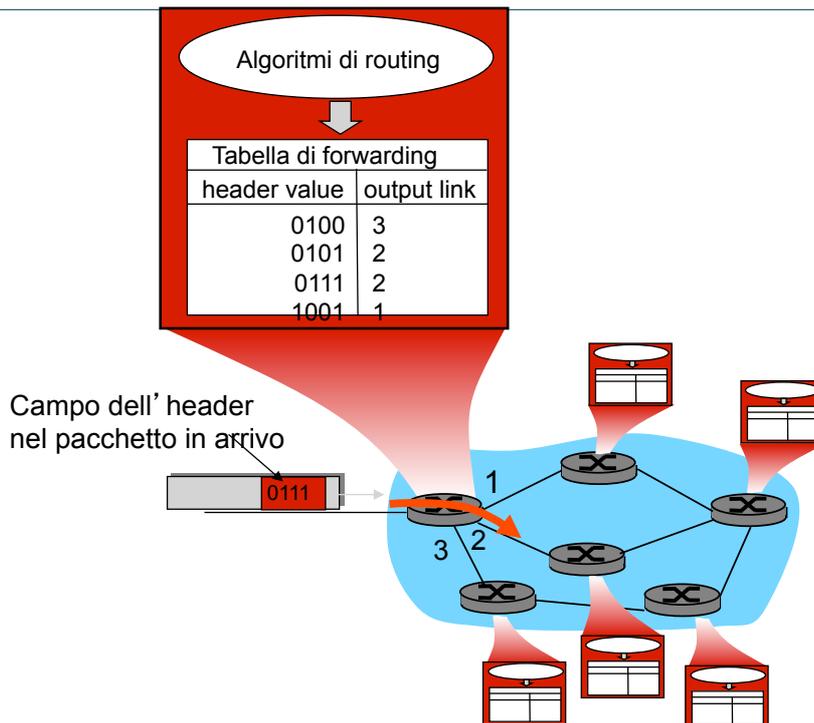
### ❑ Queste funzioni richiedono un componente fondamentale

- l'indirizzamento (**Addressing**)

6



# Interazione tra Routing e Forwarding



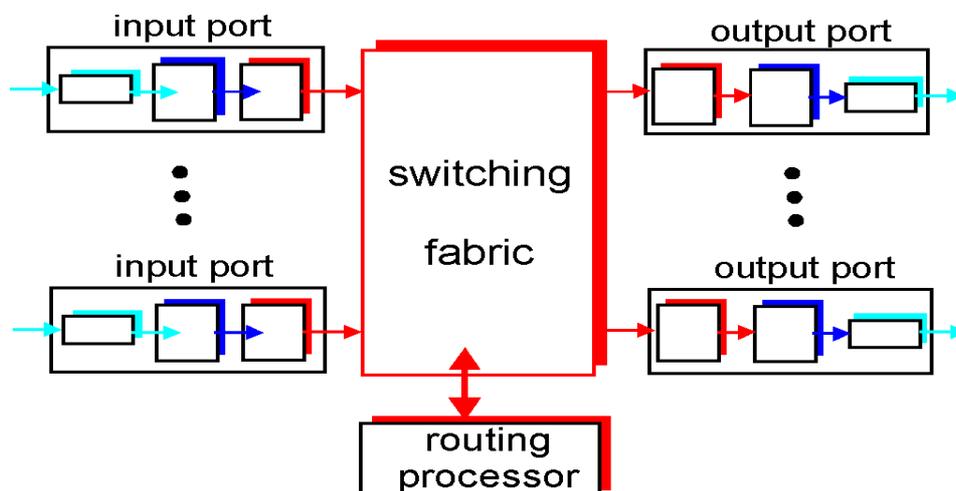
7



# Architettura di un Router: Overview

## □ Funzionalità chiave:

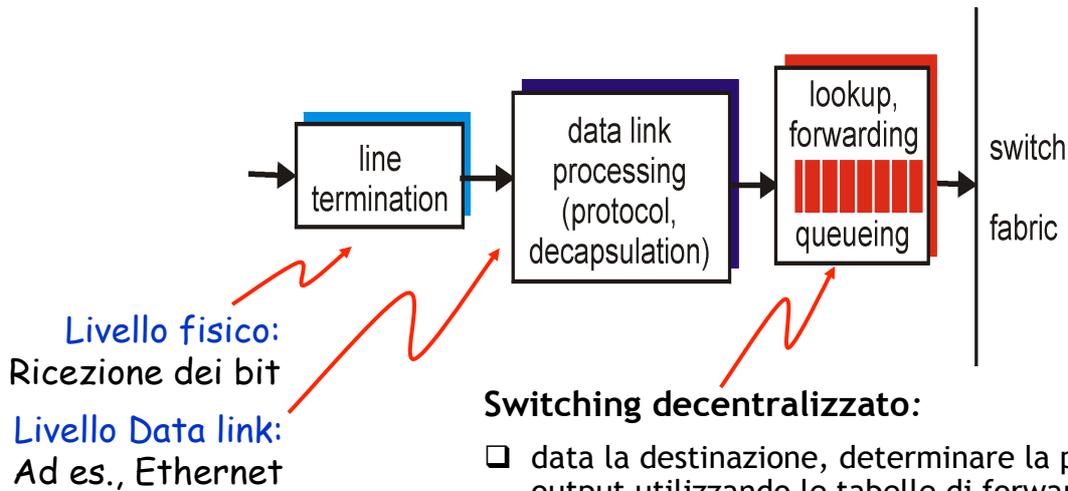
- Eseguire gli algoritmi e i protocolli di routing (RIP, OSPF, BGP)
- Trasferire (commutare) datagrammi dalla porta di input alla porta di output



8



# Funzioni delle porte di Input

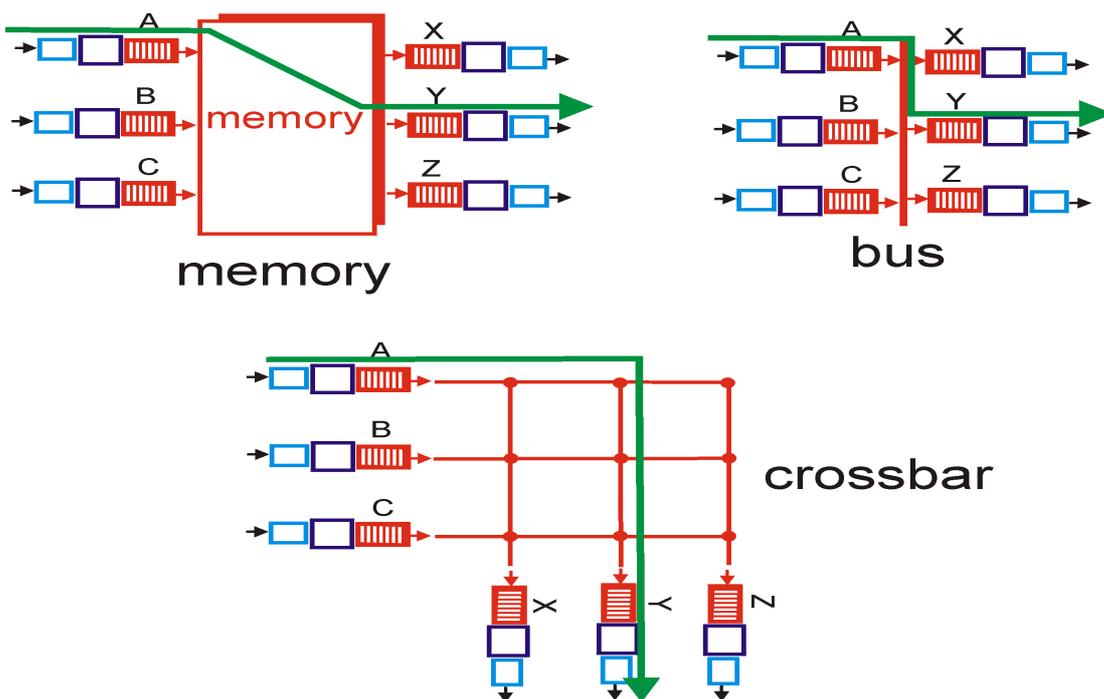


## Switching decentralizzato:

- ❑ data la destinazione, determinare la porta di output utilizzando le tabelle di forwarding in memoria locale
- ❑ scopo: completare l'elaborazione dei pacchetti a velocità di linea
- ❑ queuing: gestire le code in caso i datagrammi arrivino con velocità superiore alla velocità di switching (commutazione) dei pacchetti



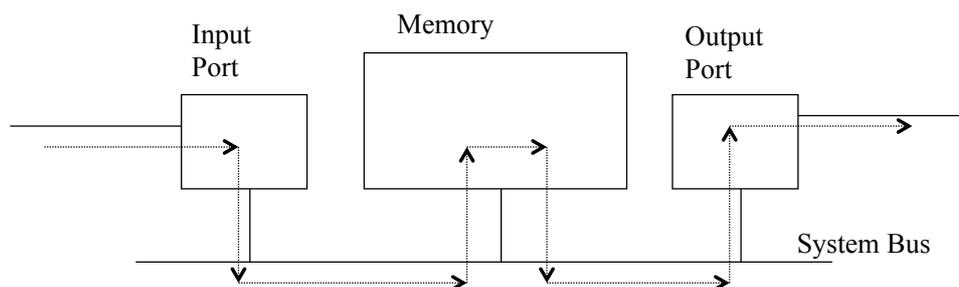
# Esempi di tipologie di matrice di commutazione



# Commutazione basata su memorie

## □ Prima generazione di router:

- computer tradizionali con commutazione controllata dalla CPU
- pacchetto copiato nella memoria di sistema
- velocità limitata dalla velocità della memoria

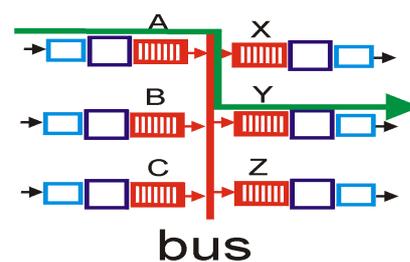


11



# Commutazione via Bus

- Comunicazione tra porte di input e output tramite bus condiviso
- Contesa del bus: velocità di commutazione data dalla velocità del bus
- Esempio:
  - Cisco 5660 con bus a 32 Gbps
  - Sufficiente per reti aziendali e router di accesso



12



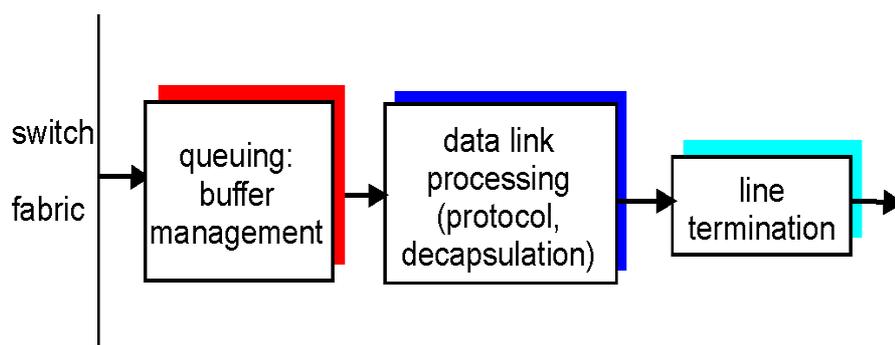
# Commutazione utilizzando una rete interconnessa

- ❑ Risolve i limiti dell'architettura a bus
- ❑ Reti di Banyan
  - reti di interconnessione sviluppate per connettere processori in architetture multiprocessore
- ❑ Architetture avanzate
  - frammentazione dei datagrammi in celle di lunghezza fissa, commutate successivamente in matrici ottimizzate
- ❑ Esempio
  - Cisco 12000: commutazione a 60 Gbps attraverso reti di interconnessione

13



# Porte di Output

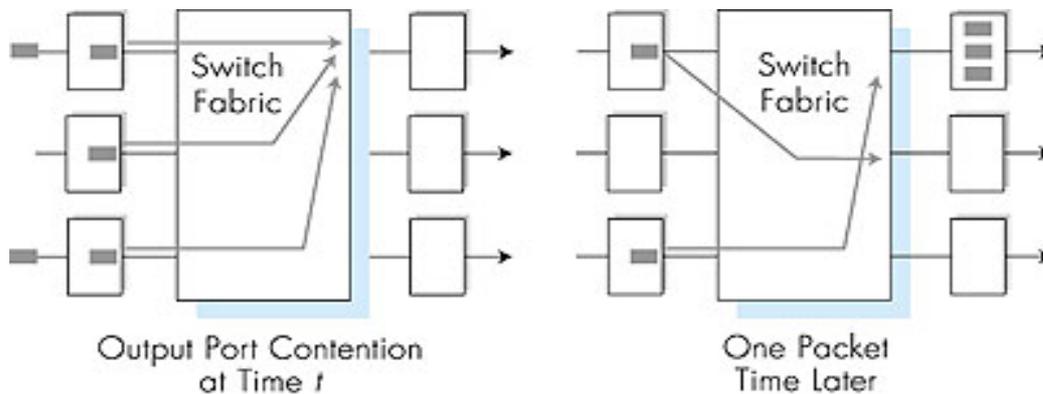


- ❑ **Buffering**: necessario quando la velocità di arrivo dei datagrammi è superiore alla velocità di trasmissione
- ❑ **Scheduling**: utilizzato per determinare l'ordine di trasmissione dei datagrammi in coda nel buffer

14



## Perché è necessario un buffer sulla porta di output



- ❑ Velocita' di arrivo alla porta di output superiore alla velocita' di trasmissione
- ❑ Il ritardo di coda e le perdite sono spesso dovute all' overflow di tale buffer



## Il datagramma IP

---

- ❑ Nello stack TCP/IP si utilizza il termine “datagramma IP” per riferirsi ad un pacchetto
- ❑ Ciascun datagramma e' formato da un **header**
  - lungo da 20 a 60 bytes, contenente informazioni essenziali per l'instradamento e la consegna del datagramma stesso
- ❑ seguito dai dati (**payload**)
  - La dimensione dei dati non e' fissa, ma e' determinata dall'applicazione che invia i dati
  - Un datagramma puo' contenere un solo byte o fino a 64K byte

17



## Il formato dell' header del datagramma IP

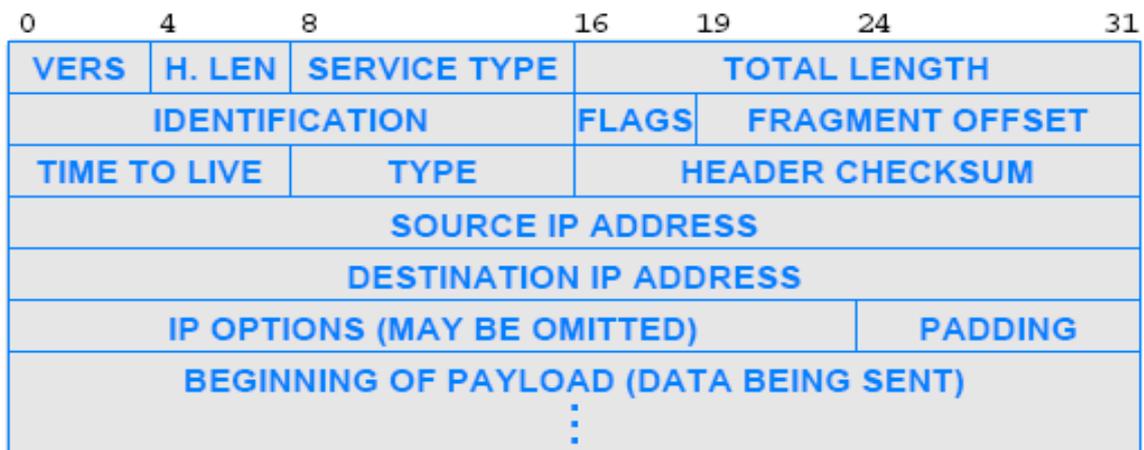
---

- ❑ Cosa contiene l' header del datagramma IP?
  - Contiene informazioni utili per trasferire il datagramma stesso
- ❑ Le informazioni dell' header includono:
  - l' indirizzo della sorgente (chi ha inviato inizialmente il datagramma)
  - l' indirizzo della destinazione (a chi va consegnato)
  - un campo che specifica il tipo di dati trasportato nel payload
- ❑ Gli indirizzi negli header sono **indirizzi IP**
  - formato standard che vedremo successivamente
- ❑ Ciascun campo dell' header ha una dimensione fissa
  - in tal modo il processing dell' header puo' essere fatto in maniera efficiente

18



## Il formato dell' header del datagramma IP



19



## Il formato dell' header del datagramma IP

### VERS

- 4 bit che specificano la versione del protocollo

### H.LEN (header length)

- 4-bit utilizzati per specificare la dimensione dell' header (numero totale di byte / 4)
- Es.: se non ci sono opzioni, il valore e' 5 (20 byte totali / 4 = 5)

### SERVICE TYPE

- 8-bit che identificano la classe di servizio del datagramma (usato raramente)

### TOTAL LENGTH

- intero a 16-bit che specifica il numero totale di byte del datagramma intero (header + payload)

20



## Il formato dell' header del datagramma IP (cont)

---

### IDENTIFICATION

- numero di 16-bit (di solito sequenziale) assegnato al datagramma
  - utilizzato per ricomporre un datagramma nel caso in cui venga frammentato

### FLAGS

- 3-bit, dove ciascun bit specifica se il datagramma e' un frammento o meno, ed eventualmente se e' l' ultimo frammento

### FRAGMENT OFFSET

- 13-bit che specificano l' offset del frammento rispetto al datagramma originale
- il valore del campo deve essere moltiplicato per 8 per ottenere il vero offset

21



## Il formato dell' header del datagramma IP(cont)

---

### TIME TO LIVE

- intero a 8-bit inizializzato dalla sorgente
- viene decrementato da ciascun router attraversato dal datagramma
- se raggiunge il valore 0, il datagramma viene scartato e un messaggio di errore viene inviato alla sorgente

### TYPE

- 8-bit che specificano il tipo di dati trasportato nel payload

### HEADER CHECKSUM

- 16-bit checksum dell' header

### SOURCE IP ADDRESS

- indirizzo Internet di 32 bit della sorgente

22



## Il formato dell' header del datagramma IP(cont)

---

### DESTINATION IP ADDRESS

- indirizzo Internet di 32 bit della destinazione

### IP OPTIONS

- Campi opzionali (non necessariamente presenti) con informazioni addizionali

### PADDING

- Se il campo "Options" e' presente e la sua dimensione non e' un multiplo di 32 bit, vengono messi degli "0" per raggiungere il multiplo di 32 bit



## Frammentazione IP



## MTU e Frammentazione del datagramma

---

- ❑ A seconda della tecnologia hardware, i diversi tratti di rete possono trasportare trame con una lunghezza massima predefinita
  - Il limite e' noto come **Maximum Transmission Unit (MTU)**
- ❑ L' hardware di rete non e' in grado di accettare o gestire trame piu' grandi della MTU
- ❑ Internet e' composto da un insieme eterogeneo di segmenti di rete
  - La restrizione sulla MTU puo' dunque creare problemi
- ❑ Un router puo' connettere reti con MTU diverse
  - Un datagramma ricevuto da un' interfaccia potrebbe essere troppo grande da spedire sull' interfaccia successiva

25



## MTU per alcune tecnologie

---

<i>Protocol</i>	<i>MTU</i>
Hyperchannel	65,535
Token Ring (16 Mbps)	17,914
Token Ring (4 Mbps)	4,464
FDDI	4,352
Ethernet	1,500
X.25	576
PPP	296

26



## MTU e Frammentazione del datagramma



- ❑ Esempio: un router (R<sub>1</sub>) interconnette due reti con valori di MTU di 1500 e 1000 byte rispettivamente
  - Gli host H<sub>1</sub> e H<sub>2</sub> sono connessi ad una rete con MTU = 1500 byte
  - I datagrammi inviati da H<sub>1</sub> a H<sub>2</sub> possono passare attraverso la rete Net 2 che ha MTU = 1000 byte
    - Su tale rete non si possono inviare o ricevere datagrammi di dimensione maggiore a 1000 byte
  - Se l'host H<sub>1</sub> invia un datagramma di 1500 byte all'host H<sub>2</sub>
    - il router R non riesce a inviare il datagramma a destinazione

27



## MTU e Frammentazione del datagramma

- ❑ Quando la dimensione di un datagramma è superiore alla massima MTU della rete verso cui deve essere inviato
  - il router divide il datagramma in pezzi più piccoli chiamati "frammenti"
  - e invia ciascun frammento in modo indipendente
- ❑ Un frammento ha lo stesso formato degli altri datagrammi
  - un bit nel campo dei FLAG dell'header indica se il datagramma è un frammento o un datagramma completo
- ❑ Vengono utilizzati altri campi dell'header per trasportare informazioni utili a riassemblare i frammenti a destinazione
  - in modo da ottenere il datagramma originale
- ❑ Ad. es., il campo FRAGMENT OFFSET specifica in che punto il frammento va riposizionato

28



## MTU e Frammentazione del datagramma

---

- Un router usa i valori di MTU e di dimensione dell'header per calcolare
  - la dimensione massima dei dati che possono essere inviati in ciascun frammento
  - e il numero di frammenti necessario
- Il router crea i frammenti
  - Usa i campi dell'header originale per creare l'header del frammento
    - Ad es., il router copia gli indirizzi IP sorgente e destinazione dall'header del datagramma all'header del frammento
  - Copia la porzione di dati dal datagramma originale al frammento
  - Trasmette il risultato

29



## Campo con Flags

---

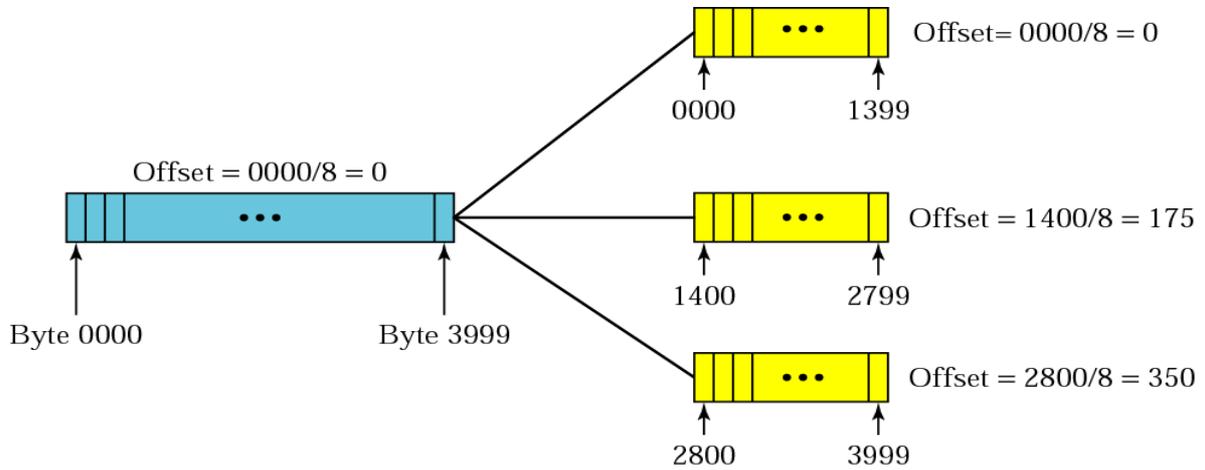
D: Do not fragment  
M: More fragments



30



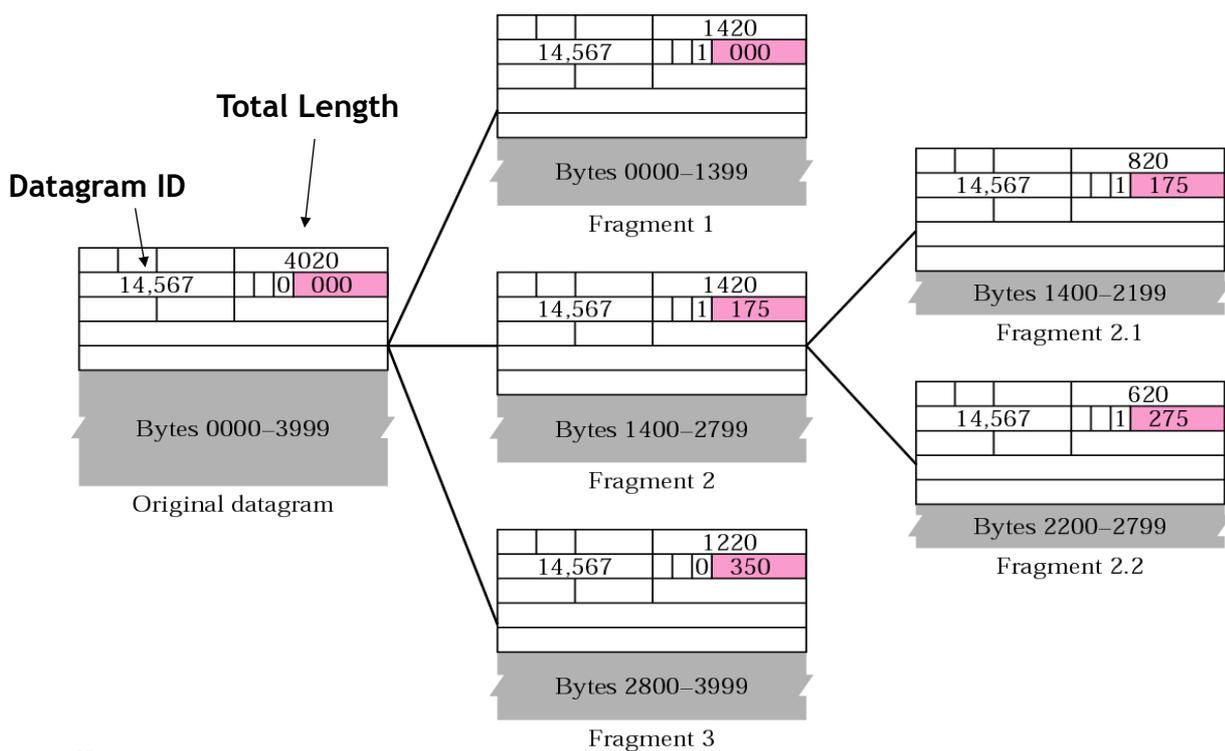
# Esempio di frammetazione



31



# Esempio di frammetazione



--



## Domande

---

- ❑ Un pacchetto ha il flag M pari a 0. Si tratta del primo segmento, l'ultimo segmento o un segmento intermedio? Si può dire se il pacchetto è stato frammentato?
  - Se il flag M è pari a 0, vuol dire che non ci sono ulteriori frammenti, per cui il frammento è l'ultimo; con la sola informazione su M non possiamo dire se il pacchetto originale è stato frammentato (serve anche l'offset).
  
- ❑ Un pacchetto ha il flag M pari a 1. Si tratta del primo segmento, l'ultimo segmento o un segmento intermedio? Si può dire se il pacchetto è stato frammentato?
  - Se il flag M è pari a 1, allora esiste almeno un altro frammento, quindi questo frammento può essere il primo o un segmento intermedio -- non si può dire quale perché serve altra informazione, ovvero l'offset

33



## Domande

---

- ❑ Un pacchetto ha il flag M pari a 1 e un offset di frammentazione pari a zero. Si tratta del primo segmento, l'ultimo segmento o un segmento intermedio?
  - Essendo il flag M pari a 1, allora si tratta del primo segmento o di uno intermedio; poiché l'offset è pari a 0, allora si tratta del primo segmento
  
- ❑ Un pacchetto ha il campo offset posto a 100. Qual'è il numero del primo byte (rispetto al datagramma originale)? È possibile risalire al numero totale di byte del datagramma originale?
  - Per trovare il numero del primo byte, basta moltiplicare l'offset per 8, ottenendo 800. Con le informazioni fornite, non è possibile affermare altro. Se il campo M fosse 0, sapremmo che si tratta dell'ultimo frammento; se, in aggiunta, conoscessimo la dimensione totale del frammento, potremmo ricostruire la dimensione originale sommando all'offset la dimensione del segmento.

34



## Riassemblaggio di un datagramma dai frammenti



### □ Esempio: pacchetti inviati da H<sub>1</sub> a H<sub>2</sub>

- se l'host H<sub>1</sub> manda datagrammi da 1500 byte, il router R<sub>1</sub> li divide in due frammenti, che vengono spediti a R<sub>2</sub>
- Il router R<sub>2</sub> non riassembla i frammenti
  - R<sub>2</sub> utilizza l'indirizzo IP di destinazione per inviare i frammenti come datagrammi a se' stanti
- Solo la destinazione finale, l'host H<sub>2</sub>, memorizza i frammenti e li riassembla per ottenere il datagramma originale



35

## Riassemblaggio di un datagramma dai frammenti

### □ Far riassemblare i datagrammi a destinazione ha due vantaggi:

- riduce la quantità di dati da memorizzare nei router
  - Per l'operazione di forwarding, un router non ha bisogno di sapere se il datagramma è intero o è un frammento
- permette di far cambiare percorso ai datagrammi in maniera dinamica
  - se ci fosse un router intermedio che fa riassemblaggio, tutti i frammenti dovrebbero passare necessariamente da quel router

### □ Rimandando il riassemblaggio alla destinazione

- il protocollo IP è libero di far percorrere ai diversi frammenti il percorso più opportuno in quel momento



36

## Perdita dei frammenti

---

- Si inizia a riassemblare un datagramma solo quando tutti i frammenti sono presenti
- La destinazione deve salvare (in un buffer) i frammenti
  - poiché' i diversi frammenti potrebbero avere ritardi diversi
  - tuttavia, i frammenti non possono essere memorizzati per sempre
- IP specifica un tempo massimo di memorizzazione dei frammenti
  - Quando arriva il primo frammento di un datagramma, la destinazione fa partire un timer
- Se tutti i frammenti di un datagramma arrivano prima che il timer scada
  - la destinazione cancella il timer e riassembla il datagramma
- Viceversa, i frammenti arrivati vengono scartati

37



## Perdita dei frammenti

---

- L' utilizzo di un timer per il riassettaggio implica una politica "tutto o niente" :
  - o tutti i frammenti arrivano e il datagramma viene ricostruito,
  - oppure il datagramma incompleto viene scartato
- Non ci sono meccanismi per far si che la destinazione comunichi i frammenti arrivati
  - La sorgente non ha informazioni riguardo alla frammentazione
- Se la sorgente ritrasmette il datagramma, il percorso seguito potrebbe essere diverso dal precedente
  - ovvero, possono essere attraversate porzioni differenti di reti
    - e non ci sono garanzie che il datagramma ritrasmesso venga frammentato nello stesso modo del precedente

38



# Reti di Calcolatori



## L'indirizzamento nel livello di rete

Universtità degli studi di Verona  
Facoltà di Scienze MM.FF.NN.  
Laurea in Informatica  
Docente: [Damiano Carra](#)

## Funzioni chiave del livello di rete

### Routing

- Determina il percorso che o pacchetti devono prendere dalla sorgente alla destinazione
- Algoritmi di routing
  - Analogia: processo di pianificazione di un viaggio, dalla partenza all' arrivo

### Forwarding

- Dato un router, trasferisce i pacchetti da una porta di input all porta di out appropriata
  - Analogia: nel caso di un viaggio, passaggio attraverso un incrocio, in cui si deve scegliere quale strada prendere

### Queste funzioni richiedono un componente fondamentale

- l' indirizzamento (**Addressing**)



# Indirizzi per Internet

---

- ❑ L'indirizzamento e' un componente fondamentale di Internet
- ❑ Tutti gli host devono utilizzare uno schema di indirizzamento **comune**
  - una coppia arbitraria di applicativi piu' comunicare senza preoccuparsi del tipo di hardware di rete utilizzato
- ❑ Ciascun indirizzo deve essere **unico**
- ❑ Gli indirizzi MAC (livello Data Link) non possono essere usati perche'
  - Internet puo' contenere diverse tecnologie di rete
  - a ciascuna tecnologia puo' avere il suo indirizzo MAC (con formati diversi)
- ❑ Gli indirizzi IP sono assegnati da un protocollo in software
  - Non sono "hard-coded" nella tecnologia utilizzata

41



# Lo schema di indirizzamento IP

---

- ❑ A ciascun host viene assegnato un numero unico di 32 bit
  - noto come "indirizzo IP" o "indirizzo Internet" dell' host
- ❑ Quando un host vuole inviare un pacchetto in Internet, la sorgente deve specificare:
  - il proprio indirizzo IP (indirizzo sorgente)
  - e l' indirizzo IP della destinazione

42



## Notazione decimale puntata

- ❑ Per semplificare la gestione degli indirizzi (da parte degli utenti), invece di indicare il valore dei 32 bit, si utilizza un'altra notazione
- ❑ Tale notazione, nota come “**dotted decimal notation**” (notazione decimale puntata), prevede di:
  - dividere i 32 bit in 4 sezioni, ciascuna da 8 bit
  - esprimere ciascuna sezione nel corrispondente valore decimale
  - dividere con un punto le diverse sezioni
- ❑ La notazione considera ogni sezione (8 bit = 1 byte) come un intero senza segno
  - il valore più piccolo è 0
    - tutti gli 8 bit hanno valore zero (0)
  - il valore più grande è 255
    - tutti gli 8 bit hanno valore uno (1)
  - Il range dei valori è costituito dai seguenti estremi  
da 0.0.0.0 fino a 255.255.255.255

43



## Notazione decimale puntata: esempi

32-bit Binary Number	Equivalent Dotted Decimal
1000001 00110100 0000110 0000000	129 . 52 . 6 . 0
1100000 0000101 0011000 0000011	192 . 5 . 48 . 3
00001010 0000010 0000000 00100101	10 . 2 . 0 . 37
1000000 00001010 0000010 0000011	128 . 10 . 2 . 3
1000000 1000000 1111111 0000000	128 . 128 . 255 . 0

44



# Gerarchia degli indirizzi IP

- ❑ Gli indirizzi IP sono divisi concettualmente in due parti:
- ❑ Un **prefisso** → identifica la rete fisica a cui l'host è connesso (noto anche come NetID)
  - A ciascuna rete in Internet viene assegnato un unico numero di rete
- ❑ Un **suffisso** → identifica un host specifico all'interno di una rete (noto anche come HostID)
  - A ciascun host su una data rete viene assegnato un suffisso unico
- ❑ Lo schema degli indirizzi IP garantisce due proprietà:
  - A ciascun host viene assegnato un indirizzo unico
  - L'assegnazione dei numeri di rete (prefissi) viene coordinata globalmente
    - I suffissi vengono assegnati localmente, senza bisogno di coordinazione globale



## Indirizzamento Classful



## Classi di indirizzi IP: tradeoff dei bit

---

### ❑ Problema: quanti bit usare per il prefisso e il suffisso?

- Il prefisso ha bisogno di un numero di bit sufficientemente grande per identificare tutte le possibili reti fisiche in Internet
- Il suffisso ha bisogno di un numero di bit sufficientemente grande da poter specificare tutti i possibili host connessi ad una data rete

### ❑ Non esiste una scelta semplice per l'allocazione dei bit!

- Utilizzando tanti bit per il prefisso si possono specificare molte reti
  - ma ciascuna rete avrà dimensione limitata
- Utilizzando tanti bit per il suffisso si possono avere molti host su una data rete
  - ma il numero totale di reti sarà limitato

47



## Classi di indirizzi IP: soluzione originale

---

### ❑ Internet contiene poche reti molto grandi e molte reti piccole

- i progettisti hanno scelto uno schema che permettesse la convivenza di combinazioni di reti grandi e piccole

### ❑ L'indirizzamento IP originale, chiamato **classful**, divideva lo spazio di indirizzamento in 3 classi primarie

- ciascuna classe aveva una dimensione del prefisso / suffisso differente

### ❑ I primi 4 bit di un indirizzo IP determinavano la classe di indirizzamento di appartenenza

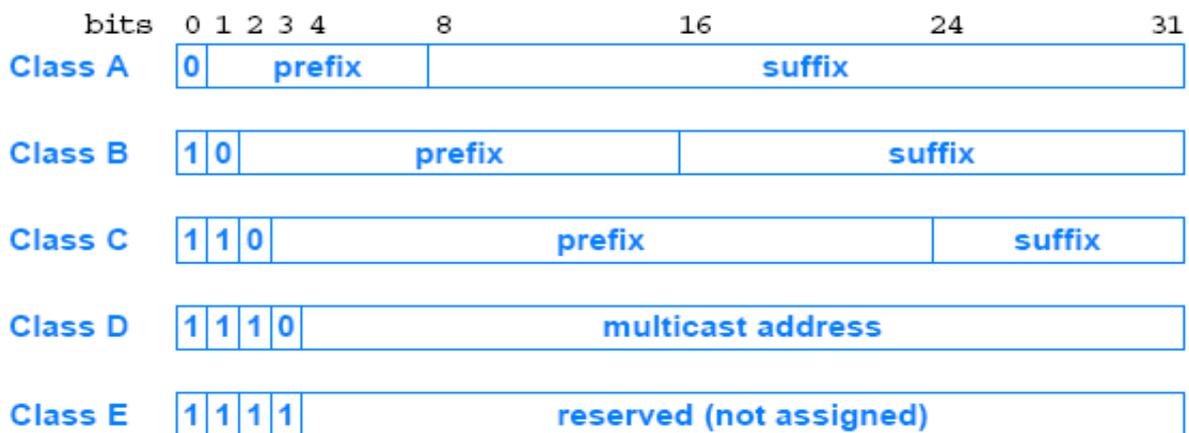
- specificavano come il resto dell'indirizzo doveva essere diviso tra prefisso e suffisso

**NOTA: In questa slide (e in tutte quelle relative allo schema di indirizzamento classful) viene usato il tempo al passato**

48



## Classi di indirizzi IP: soluzione originale



49



## Divisione dello spazio di indirizzamento

Address Class	Bits In Prefix	Maximum Number of Networks	Bits In Suffix	Maximum Number Of Hosts Per Network
A	7	128	24	16777216
B	14	16384	16	65536
C	21	2097152	8	256

50



## Divisione dello spazio di indirizzamento

---

- ❑ Lo schema classful divideva lo spazio di indirizzamento in porzioni di dimensione non uguali tra loro
- ❑ I progettisti scelsero questa soluzione per poter includere diversi scenari
  - Meta' degli indirizzi IP disponibili appartengono alla classe A
  - Il numero di reti di classe A e' pero' limitato a 128
    - Lo scopo era permettere ai principali ISP di creare ciascuno una grande rete che connettesse milioni di host
  - In maniera analoga, la classe C e' stata creata per permettere ad una piccola organizzazione di connettere alcuni calcolatori ad una LAN

51



## Autorita' per gli indirizzi

---

- ❑ E' stata creata un' autorita' per l' assegnazione degli indirizzi e gestire le relative dispute
  - Internet Corporation for Assigned Names and Numbers ([ICANN](#))
- ❑ ICANN non assegna direttamente i prefissi
  - ma autorizza un insieme di [registrars](#) ufficiali a farlo
- ❑ I Registrars assegnano i blocchi di indirizzi agli ISP
  - Gli ISP forniscono gli indirizzi ai loro utenti
- ❑ Per ottenere un prefisso, una compagnia / societa' contatta di solito un ISP

52



# Indirizzamento Classless



## Subnet e indirizzamento classless

- ❑ Con l'espansione di Internet lo schema classful originale si e' rivelato limitante
- ❑ Tutti chiedevano indirizzi di classe A o B
  - In modo da poter avere un numero sufficiente di indirizzi per eventuali espansioni
    - conseguente sottoutilizzo di indirizzi all'interno di ogni classe
- ❑ Sono stati proposti due meccanismi correlati per risolvere tali limitazioni
  - Subnet
  - Indirizzamento Classless
- ❑ Invece di avere un insieme ristretto di lunghezze per i prefissi / suffissi, la scelta della lunghezza viene resa arbitraria



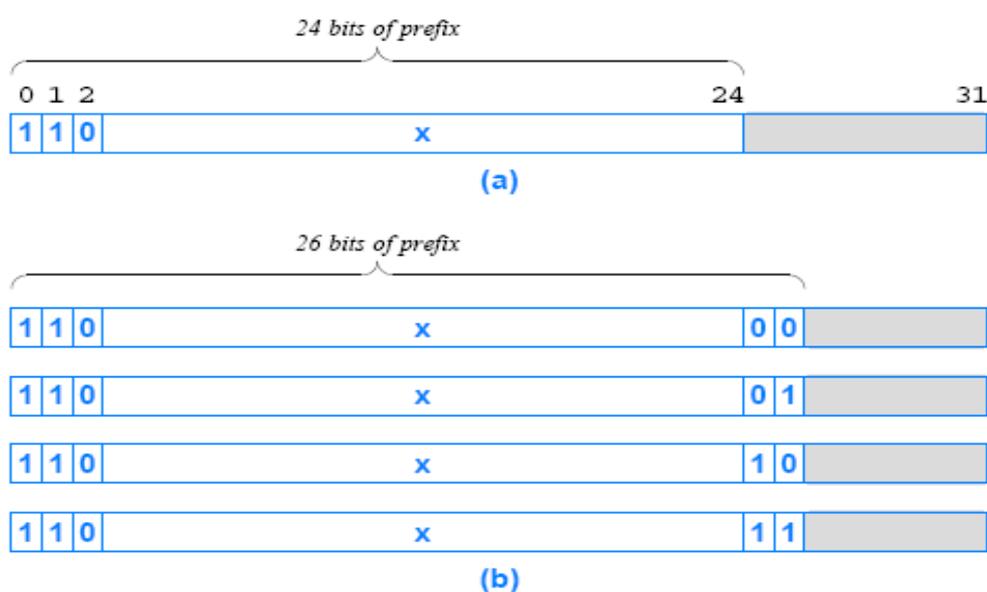
## Subnet e indirizzamento classless: Motivazioni

- ❑ Si consideri un ISP che distribuisce prefissi. Si assuma che un cliente chieda un prefisso per una rete che ha 55 host
- ❑ In caso di indirizzamento classful:
  - Si dovrebbe assegnare una classe C (254 indirizzi host)
  - Sarebbero sufficienti 6 bit per rappresentare tutti valori degli host
    - 190 indirizzi su 254 non sarebbero usati
  - La maggior parte dello spazio di indirizzamento sarebbe sprecato
- ❑ In caso di indirizzamento classless:
  - L'ISP puo' assegnare liberamente la dimensione del prefisso
    - Ad es., 26 bit per il prefisso
    - e 6 bit per il suffisso

55



## Subnet e indirizzamento classless: Esempio



56



# Maschere

---

- ❑ Come puo' un router conoscere la lunghezza del prefisso?
  - Le decisioni di routing si prendono solo analizzando il prefisso, che indica la rete di appartenenza
  - Con l'indirizzamento classful, i primi bit indicavano la classe e quindi la lunghezza del prefisso
- ❑ Nel caso di indirizzamento classless, serve aggiungere un pezzo di informazione
  - la suddivisione tra prefisso e suffisso
- ❑ Invece di aggiungere la dimensione del prefisso esplicitamente, si preferisce usare un'altra tecnica nota come **maschera di indirizzo** o **maschera di subnet**
  - Una maschera non e' altro che un valore a 32 bit in cui sono posti a 1 tutti i bit fino a raggiungere la lunghezza del prefisso
  - Le maschere rendono il processing piu' efficiente

57



# Maschere

---

- ❑ Un router mantiene in memoria una serie di
  - reti di destinazione (prefissi di rete)
  - e le corrispondenti maschere
- ❑ Quando arriva un pacchetto con indirizzo IP generico
  - il router confronta l'indirizzo di destinazione con i prefissi in memoria
  - fa il forward del pacchetto in base alla destinazione
- ❑ Il confronto non viene fatto su tutti i 32 bit
  - per ogni destinazione (prefisso), si considera la maschera
  - viene fatto l'AND bit a bit tra maschera e indirizzo IP del pacchetto
  - si confronta il risultato con il prefisso in memoria
  - se sono uguali, allora e' stata determinata la destinazione del pacchetto

58



## Maschere: Esempio

- ❑ Si considerino i prefissi e le relative maschere memorizzate in un router:

NetA            10000000 00001010 00000000 00000000    → 128.10.0.0

MaskA           11111111 11111111 00000000 00000000    → 255.255.0.0

NetB            01000000 00001010 00000010 00000000    → 64.10.2.0

MaskB           11111111 11111111 11111111 00000000    → 255.255.255.0

- ❑ Il router analizza un pacchetto con il seguente indirizzo IP

IP addr         10000000 00001010 00000010 00000011    → 128.10.2.3

- ❑ Il router applica le diverse maschere al pacchetto e confronta il risultato con i prefissi

IP & MaskA     10000000 00001010 00000000 00000000    → 128.10.0.0

IP & MaskB     10000000 00001010 00000010 00000000    → 128.10.2.0

- ❑ Essendo (IP & MaskA) = NetA, allora il pacchetto ha come destinazione la rete NetA



## Notazione CIDR

- ❑ Classless Inter-Domain Routing (CIDR)
- ❑ L' utilizzo di maschere per specificare la dimensione del prefisso viene fatta per questioni di efficienza
  - operazioni di AND bit a bit molto veloci in hardware
- ❑ Tuttavia, per facilitare la gestione da parte degli utenti, si utilizza una notazione piu' semplice e diretta
  - viene specificata la dimensione del prefisso
- ❑ In particolare, la notazione CIDR prevede la seguente forma:  
**ddd.ddd.ddd.ddd/m**
  - **ddd** e' il valore decimale nella notazione decimale puntata
  - **m** di bit del prefisso
- ❑ Nell' esempio precedente, il router ha memorizzato due reti:
  - NetA: 128.10.0.0/16
  - NetB: 64.10.2.0 /24



## CIDR: indirizzi per gli host

---

- ❑ Dopo aver ricevuto il prefisso CIDR da un ISP, un cliente puo' assegnare liberamente gli indirizzi di host ai propri utenti
  
- ❑ Svantaggi dell' indirizzamento classless
  - maggiore informazione da memorizzare nei router, e conseguenti operazioni da svolgere per il processing di un pacchetto
  - poiche' la divisione tra prefisso e suffisso e' arbitraria, usando la notazione decimale puntata non e' sempre facile riuscire a leggere gli indirizzi



## Indirizzi IP speciali



## Indirizzi IP speciali

---

- Il protocollo IP definisce un insieme di indirizzi IP speciali che sono riservati
  - gli indirizzi IP speciali non possono essere assegnati agli host
- Esempi:
  - Indirizzi di rete
  - Indirizzi di “Directed Broadcast”
  - Indirizzi di “Limited Broadcast”
  - Indirizzo “Questo Host”
  - Indirizzo di Loopback

63



## Indirizzo di rete

---

- E' utile avere un indirizzo che denota il solo prefisso assegnato ad una rete
- L' indirizzo IP con host address a zero viene riservato
  - e utilizzato per identificare la rete
- Quindi, l' indirizzo 128.211.0.16/28 identifica una rete
  - poiche' i bit oltre il 28-esimo sono posti a zero
  - 10000000 11010011 00000000 00010000
- Un indirizzo di rete non deve mai comparire come indirizzo di destinazione di un pacchetto

64



## Indirizzi di “Directed Broadcast”

---

- Serve per semplificare il broadcasting (invio a tutti)
  - Viene definito un indirizzo di broadcast diretto diverso per ciascuna rete
- Quando viene inviato un pacchetto con indirizzo di broadcast diretto
  - un solo pacchetto viaggia su Internet, finché non raggiunge la rete specificata
  - il pacchetto viene poi consegnato a tutti gli host della rete specificata
- L'indirizzo di broadcast diretto è costruito ponendo a 1 tutti i bit del suffisso
  - 10000000 11010011 00000000 00011111

65



## Indirizzi di “Limited Broadcast”

---

- Broadcast limitato si riferisce al broadcast verso gli host connessi direttamente alla rete a cui è connesso l'host che invia il pacchetto
  - Utilizzato durante lo startup del sistema, quando l'host non conosce ancora l'indirizzo di rete
- L'indirizzo è formato ponendo a 1 tutti i 32 bit
  - 11111111 11111111 11111111 11111111
- Il pacchetto raggiungerà tutti gli host appartenenti alla stessa rete locale del nodo che ha originato il pacchetto

66



## Indirizzo “Questo Host”

---

- ❑ Un host deve conoscere il proprio indirizzo IP
  - prima di ricevere o inviare pacchetti su Internet
- ❑ Lo stack TCP/IP contiene dei protocolli che possono essere utilizzati per ottenere un indirizzo IP automaticamente all' accensione dell' host
  - ... ma il protocollo di startup utilizza il protocollo IP per comunicare
- ❑ Durante lo startup
  - un host non puo' indicare un indirizzo di sorgente corretto (non lo possiede ancora)
  - a tale scopo, l' indirizzo formato da tutti 0 viene riservato per indicare “questo host”
    - ➔ 00000000 00000000 00000000 00000000
  - Vedremo nelle prossime lezioni nel dettaglio la sequenza dei messaggi di startup

67



## Indirizzo di Loopback

---

- ❑ L' indirizzo di Loopback e' usato per testare applicaz. di rete
  - ad es., per il debugging di applicazioni di rete in fase di sviluppo
- ❑ Un programmatore deve avere due applicazioni che comunicano attraverso una rete
- ❑ Invece di eseguire ciascun programma su host separati
  - il programmatore fa girare entrambi i programmi su un solo host
  - e configura tali programmi per utilizzare gli indirizzi di loopback per comunicare
- ❑ Quando un' applicazione invia dati ad un' altra
  - i dati viaggiano lungo lo stack protocollare fino al livello IP
  - il livello IP rigira il pacchetto, passando di nuovo attraverso lo stack all' altra applicazione

68



## Indirizzo di Loopback (cont.)

---

- IP riserva il prefisso **127.0.0.0/8** per il loopback
- L' indirizzo di host (suffisso) e' irrilevante
  - di solito si utilizza il primo host disponibile, ovvero **127.0.0.1**
- Durante il testing con uso di loopback, nessun pacchetto di fatto viene trasmesso
  - e' lo strato software che gestisce il protocollo IP che rigira i pacchetti da un' applicazione ad un' altra
- L' indirizzo di loopback non appare mai nei pacchetti che viaggiano su qualsiasi rete (locale, WAN, ...)

69



## Indirizzi IP speciali: schema riassuntivo

---

Prefix	Suffix	Type Of Address	Purpose
all-0s	all-0s	this computer	used during bootstrap
network	all-0s	network	identifies a network
network	all-1s	directed broadcast	broadcast on specified net
all-1s	all-1s	limited broadcast	broadcast on local net
127/8	any	loopback	testing

70



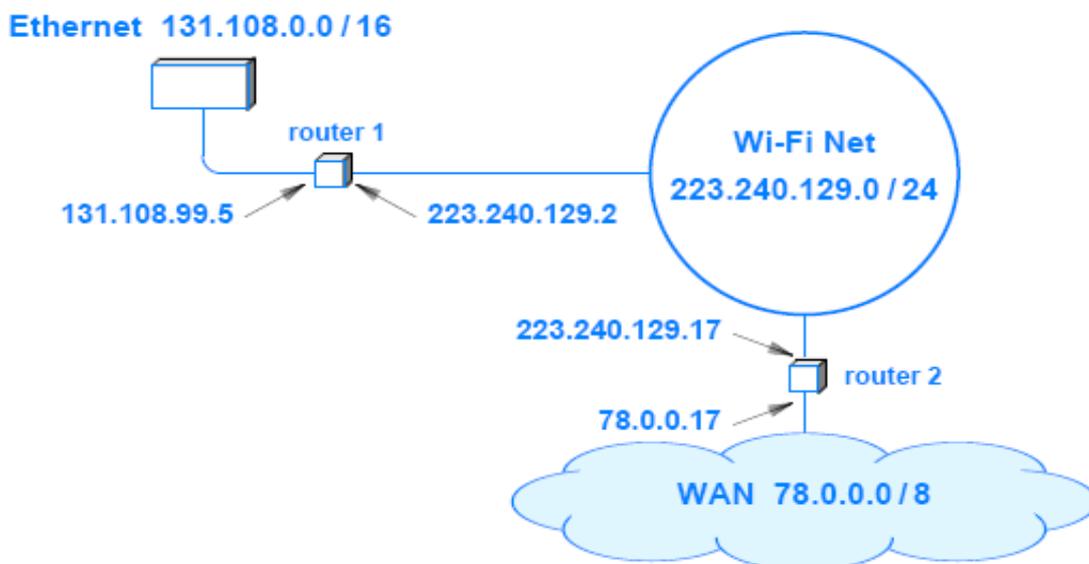
## Router e il principio di indirizzamento

- ❑ A ciascun router vengono assegnati 2 o più indirizzi IP
  - un indirizzo per ciascuna rete a cui il router è connesso
- ❑ Per comprenderne il motivo, occorre ricordare che:
  - un router ha connessione verso più reti fisiche
  - ciascun indirizzo IP contiene un prefisso che specifica una rete fisica
- ❑ Un solo indirizzo IP non è sufficiente per un router
  - perché ciascun router connette diverse reti
  - e ciascuna rete ha un prefisso unico
- ❑ Lo schema degli indirizzi IP può essere riassunto con un principio:
  - Un indirizzo IP non identifica un computer specifico: esso identifica la connessione tra un computer e la rete
  - Un computer con più di una connessione di rete (ad es., un router) deve avere più di un indirizzo IP, uno per ogni connessione

71



## Routers and the IP Addressing Principle



72



# Reti di Calcolatori



## Esercizi su indirizzamento IP

Universtità degli studi di Verona  
Facoltà di Scienze MM.FF.NN.  
Laurea in Informatica  
Docente: [Damiano Carra](#)

### Esercizio 1

*Trasformare i seguenti indirizzi IP nella notazione decimale puntata.*

- a. 10000001 00001011 00001011 11101111*
- b. 11000001 10000011 00011011 11111111*
- c. 11100111 11011011 10001011 01101111*
- d. 11111001 10011011 11111011 00001111*

#### *Soluzione*

- a. 129.11.11.239*
- b. 193.131.27.255*
- c. 231.219.139.111*
- d. 249.155.251.15*



## Esercizio 2

---

*Trasformare i seguenti indirizzi IP nella notazione binaria.*

- a. 111.56.45.78*
- b. 221.34.7.82*
- c. 241.8.56.12*
- d. 75.45.34.78*

### *Soluzione*

- a. 01101111 00111000 00101101 01001110*
- b. 11011101 00100010 00000111 01010010*
- c. 11110001 00001000 00111000 00001100*
- d. 01001011 00101101 00100010 01001110*

75



## Esercizio 3

---

*Trovare l'errore, se esiste, nei seguenti indirizzi IP:*

- a. 111.56.045.78*
- b. 221.34.7.8.20*
- c. 75.45.301.14*
- d. 11100010.23.14.67*

### *Soluzione*

- a. Non si antepongono zeri nella notazione decimale puntata (045).*
- b. Non si possono avere piu' di 4 sezioni nella notazione decimale puntata.*
- c. Nella notazione decimale puntata, ciascun numero deve essere inferiore o uguale a 255; 301 e' fuori range.*
- d. Non e' permesso mescolare notazione decimale puntata e binaria.*

76



## Esercizio 4

---

*Qual e' l'indirizzo di rete se uno degli indirizzi e' 167.199.170.82/27?*

### *Soluzione*

*La lunghezza del prefisso e' 27 bit, per cui si mantengono fissi i primi 27 bit e si pongono a zero i rimanenti 5*

*Indirizzo in binario: 10100111 11000111 10101010 01010010*

*Ultimi 5 bit a zero: 10100111 11000111 10101010 01000000*

*Risultato in notazione CIDR: 167.199.170.64 / 27*

77



## Esercizio 5a e 5b

---

*Qual e' il numero di indirizzi (inclusi gli indirizzi riservati) del blocco se uno degli indirizzi e' 140.120.84.24/20.*

### *Soluzione*

*Il prefisso e' lungo 20 bit, quindi il blocco ha  $2^{32-20} = 2^{12} = 4096$  indirizzi.*

*Qual e' l'indirizzo di rete se uno degli indirizzi e' 140.120.84.24/20?*

### *Soluzione*

*L'indirizzo di rete e' 140.120.80.0/20.*

78



## Esercizio 6a e 6b

*Si trovi il blocco CIDR se uno degli indirizzi e' 190.87.140.202/29.*

### *Soluzione*

*Il numero di indirizzi e'  $2^{32-29} = 8$ . L'indirizzo di rete e' 190.87.140.200/29, l'indirizzo di broadcast e' 190.87.140.207/29*

*Si mostri una configurazione di rete per il blocco del precedente esempio.*

### *Soluzione*

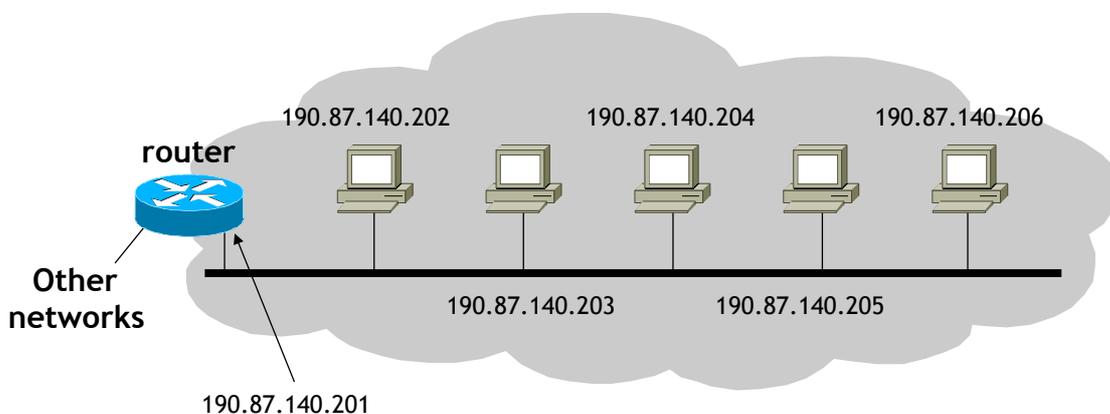
*Ci sono due indirizzi speciali che non possono essere usati per gli host: l'indirizzo di rete (bit dell'HostID tutti a 0) e l'indirizzo di broadcast limitato (bit dell'HostID tutti a 1). Si veda la prossima slide*

79



## Esercizio 6b (cont)

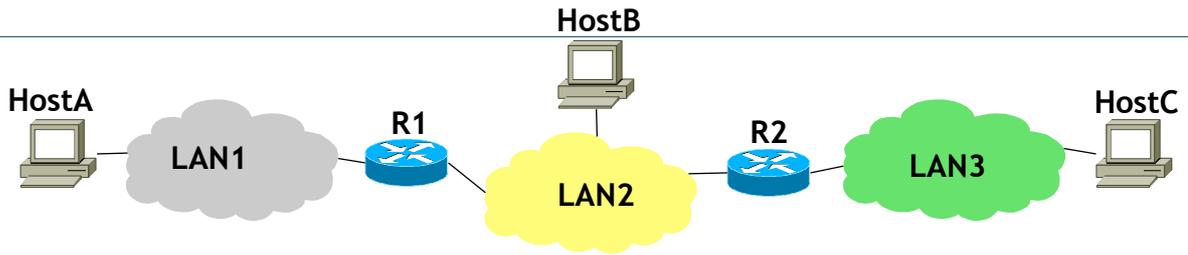
Indirizzi speciali	
Indirizzo di rete	→ 190.87.140.200 / 29
Indirizzo di Broadcast	→ 190.87.140.207 / 29



80



## Esercizio 7



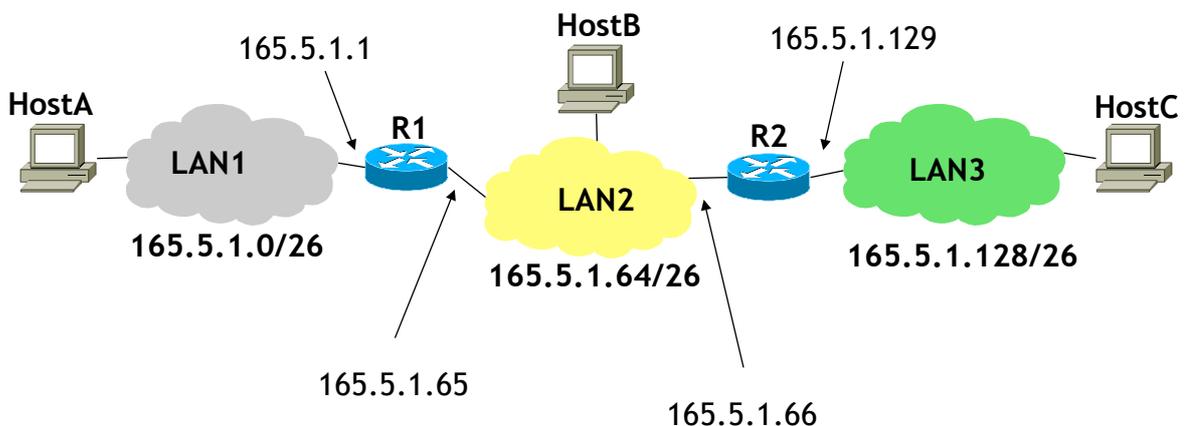
□ Al sito rappresentato e' stato assegnato il blocco CIDR 165.5.1.0/24

- si assegnino gli indirizzi di rete alle tre LAN partizionando il blocco in modo che ciascuna LAN possa contenere lo stesso numero di stazioni (massimizzando tale numero)
- per ogni LAN si specifichi l' indirizzo di broadcast spiegando come si ottiene
- si assegni il primo (o i primi) indirizzo(i) disponibile(i) per ogni LAN all'interfaccia del router ad essa collegato



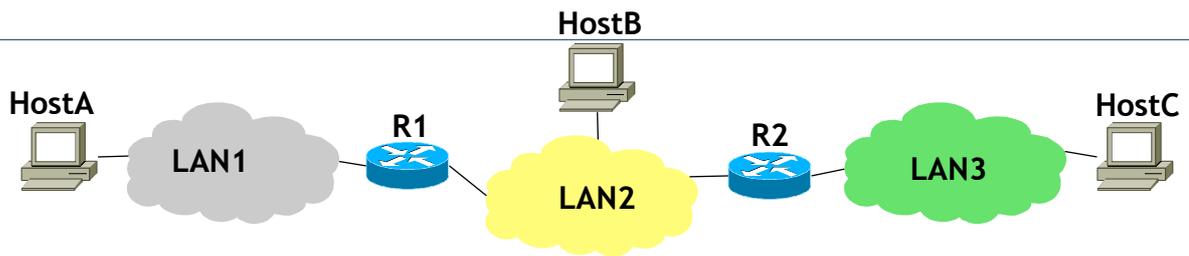
81

## Esercizio 7 - Soluzione



82

## Esercizio 8



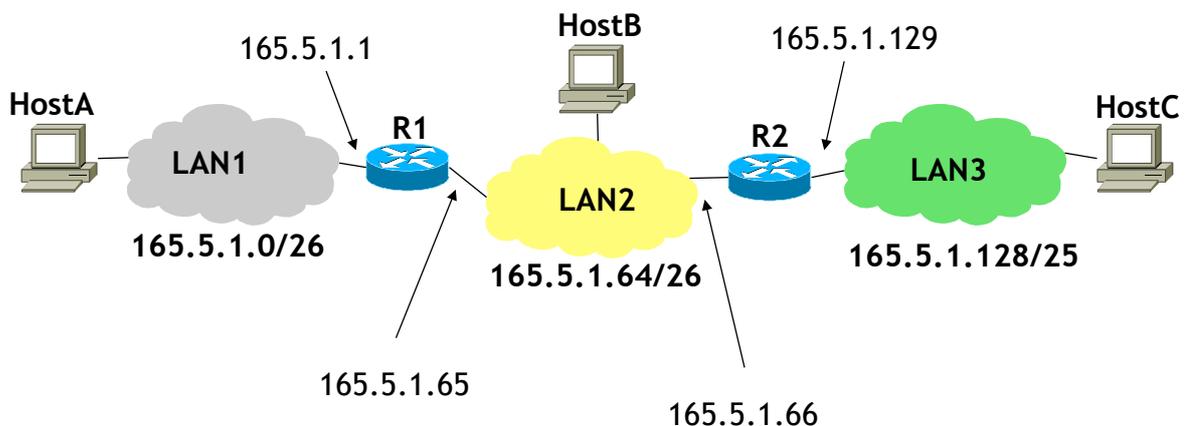
□ Al sito rappresentato e' stato assegnato il blocco CIDR 165.5.1.0/24

- si assegnino gli indirizzi di rete alle tre LAN partizionando il blocco in modo che ciascuna LAN1 e LAN2 possano contenere almeno 32 stazioni ciascuna, mentre LAN3 ne possa contenere almeno 64
- per ogni LAN si specifichi l' indirizzo di broadcast spiegando come si ottiene
- si assegni il primo (o i primi) indirizzo(i) disponibile(i) per ogni LAN all'interfaccia del router ad essa collegato



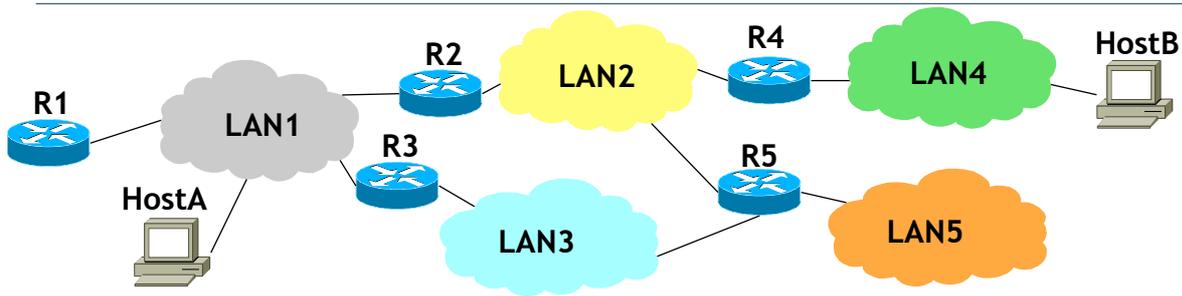
83

## Esercizio 8 - Soluzione



84

## Esercizio 9



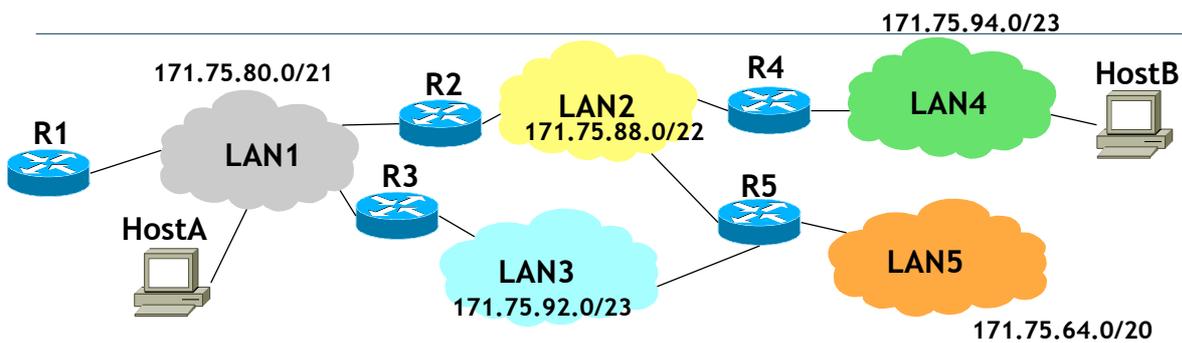
□ Con riferimento alla figura:

- Si scriva il blocco CIDR a dimensione minima contenente gli indirizzi 101.75.79.255 e 101.75.80.0.
- Si utilizzi il blocco calcolato al punto precedente per assegnare il piano di indirizzamento alle reti LAN1/2/3/4/5 rispettando i seguenti vincoli:
  - LAN 1 ha netmask /21,
  - LAN 2 deve ospitare 1000 host,
  - LAN 3 ha netmask /23,
  - LAN 4 deve ospitare 400 host,
  - LAN 5 ha a disposizione metà dell'intero blocco di indirizzi.

85



## Esercizio 9 - Soluzione



- |                 |   |                 |                 |                 |                 |
|-----------------|---|-----------------|-----------------|-----------------|-----------------|
| □ 101.75.79.255 | → | 01100101        | 01001011        | 01001111        | 11111111        |
| □ 101.75.80.0   | → | 01100101        | 01001011        | 01010000        | 00000000        |
| □ Parte comune  | → | <u>01100101</u> | <u>01001011</u> | <u>01000000</u> | <u>00000000</u> |
- Che corrisponde a 101.75.64.0/19

- LAN1 → /21
- LAN2 → 1000 host, servono almeno 10 bit per l'HostID → /22
- LAN3 → /23
- LAN4 → 400 host, servono almeno 9 bit per l'HostID → /23
- LAN5 → /20

86

