

Reti di Elaboratori

Livello di Rete: Multicast, amministrazione della rete

Thanks to prof. Gaia Maselli

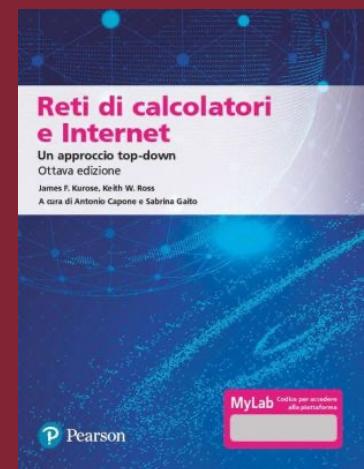


SAPIENZA
UNIVERSITÀ DI ROMA

Alessandro Checco

alessandro.checco@uniroma1.it

Capitolo 5



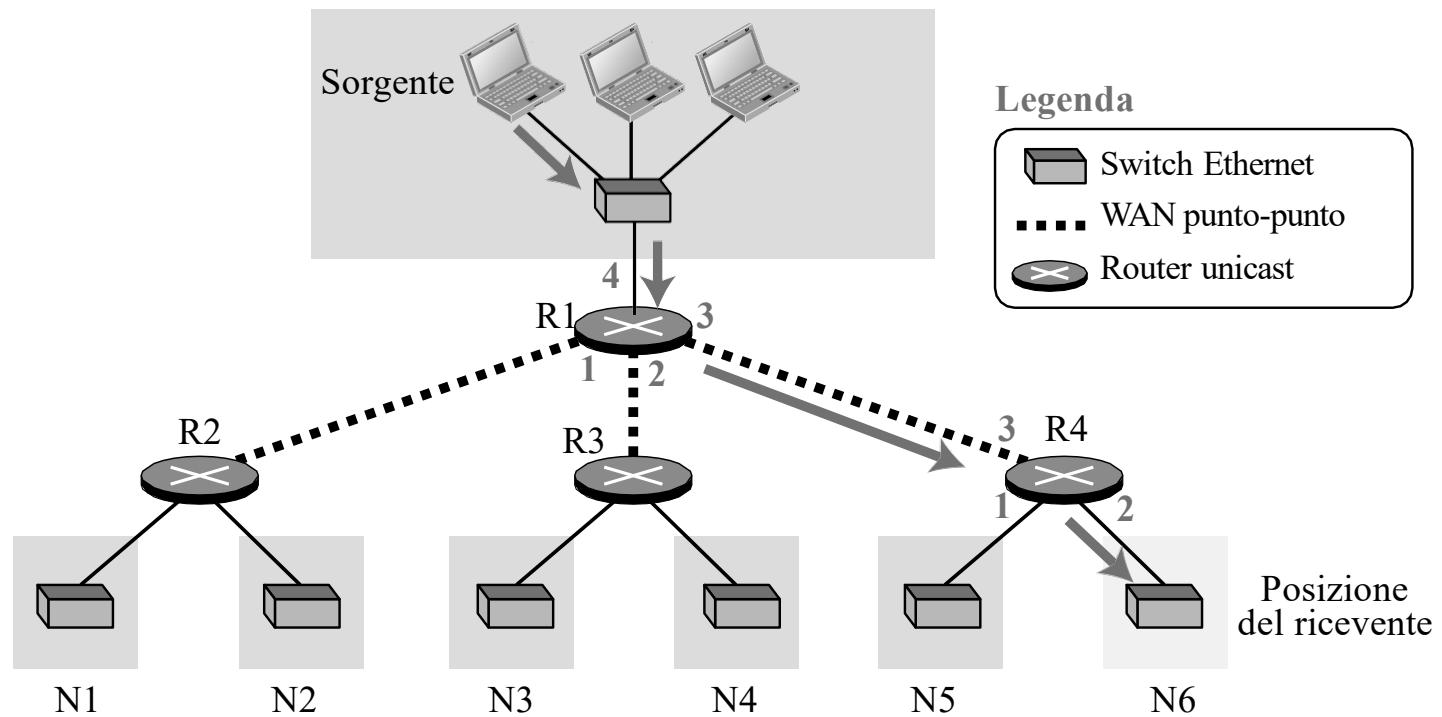
Livello di rete – piano di controllo: sommario

- introduzione
- algoritmi di instradamento
 - link state
 - distance vector
- instradamento intra-ISP: RIP e OSPF
- instradamento tra ISP: BGP
- **multicast e IGMP**
- piano di controllo SDN
- gestione della rete, configurazione
 - SNMP
 - NETCONF/YANG

Routing unicast, broadcast, multicast

Unicast

- UNICAST: comunicazione tra UNA sorgente e UNA destinazione
 - Indirizzo IP sorgente – indirizzo IP destinazione

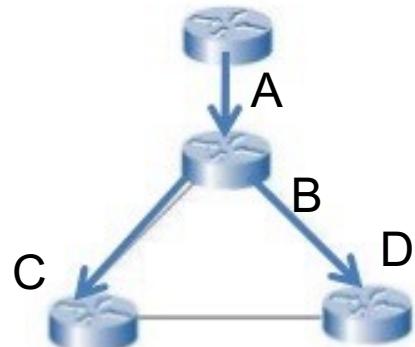


Broadcast

- BROADCAST: invio di un pacchetto da un nodo sorgente a TUTTI i nodi della rete
 - Comunicazione 1 a N, dove N: tutti i nodi della rete
 - <Indirizzo IP sorgente – indirizzo broadcast di destinazione>
- Come eseguire il broadcast?
 - Uncontrolled flooding
 - Controlled flooding
 - Sequence number
 - Reverse path forwarding

Broadcast: uncontrolled flooding

- Uncontrolled flooding
 - Quando un nodo riceve un pacchetto broadcast, lo duplica e lo invia a tutti i nodi vicini (eccetto a quello da cui lo ha ricevuto)
 - Se il grafo ha cicli, una o più copie del pacchetto cycleranno all'infinito (potenzialmente) nella rete

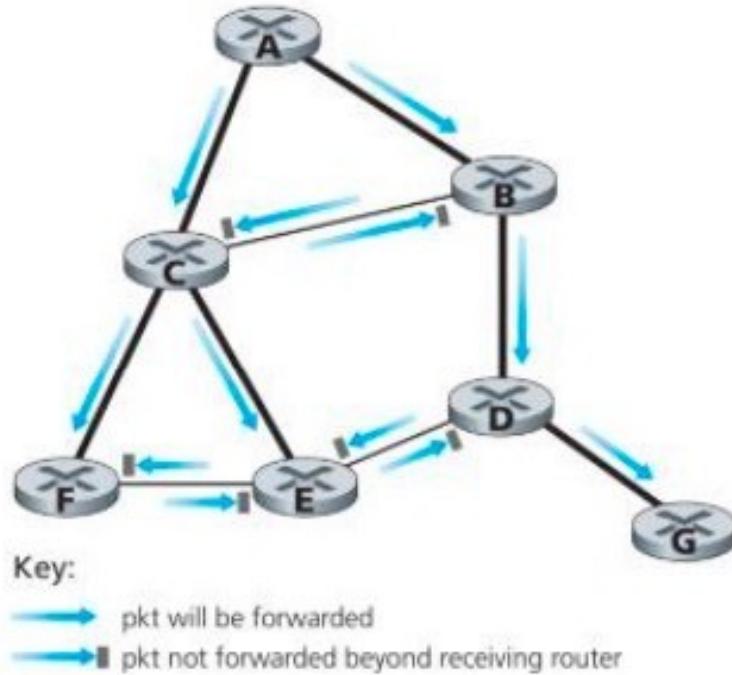


Sequence number controlled flooding

- Non forwardare pacchetti già ricevuti e inoltrati
- Ogni nodo tiene una lista di (indirizzo IP, #seq) dei pacchetti già ricevuti, duplicati, inoltrati
- Quando riceve un pacchetto controlla nella lista, se già inoltrato lo scarta, altrimenti lo forwarda

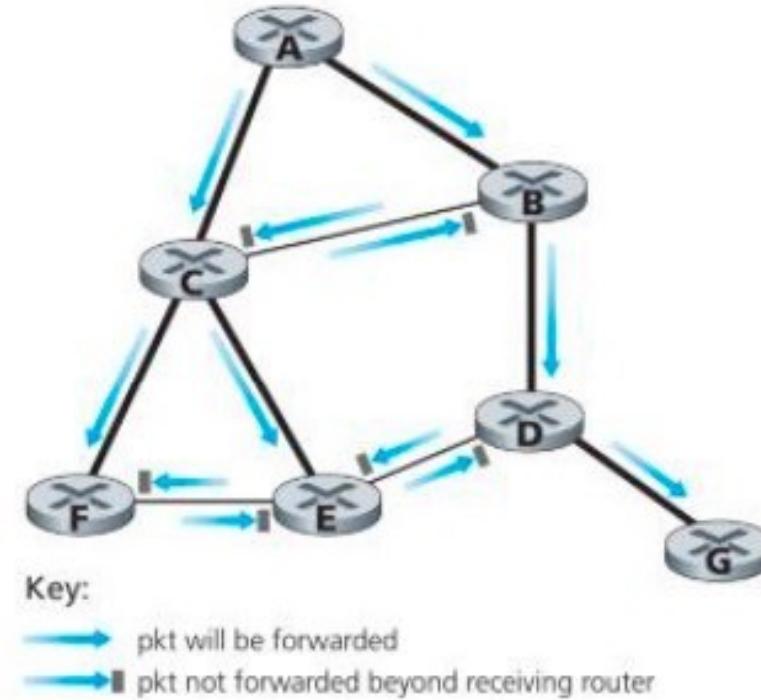
Reverse path forwarding (RPF): example

Forwarda il pacchetto solo se è arrivato dal link che è sul suo shortest path (unicast) verso la **sorgente**



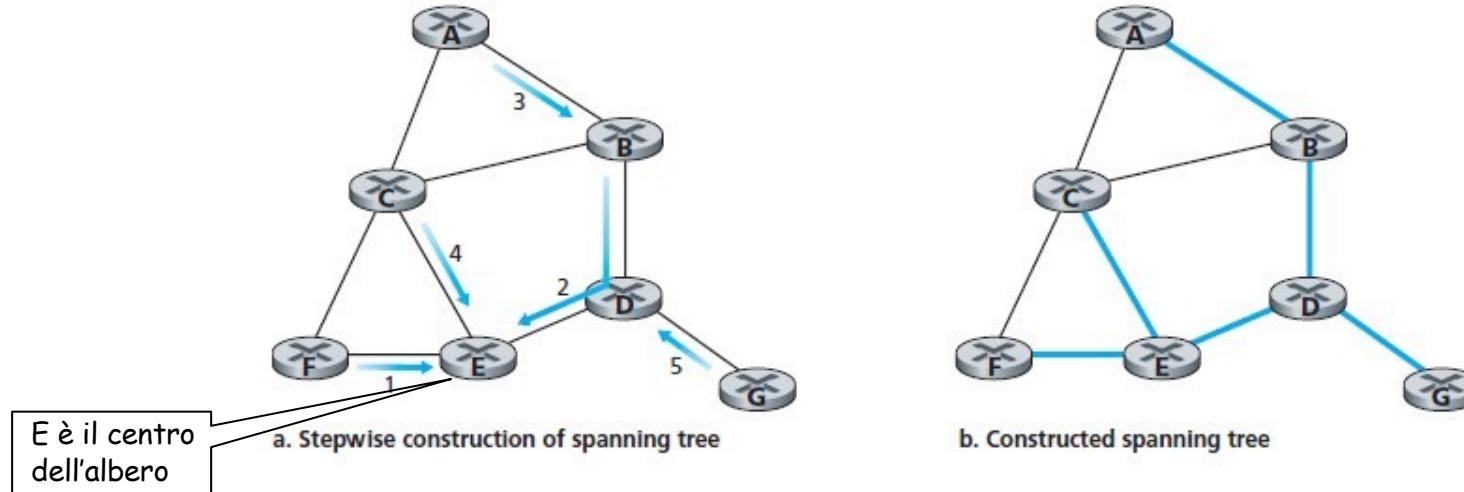
RPF: Pacchetti ridondanti

- RPF elimina il problema di inondare la rete con troppi pacchetti
- **RPF NON elimina completamente la trasmissione di pacchetti ridondanti**
- Esempio: B,C,D,E,F ricevono uno o due pacchetti ridondanti
- Ogni nodo dovrebbe ricevere una sola copia del pacchetto broadcast
- **Soluzione:** costruire lo **spanning tree** prima di inviare i pacchetti broadcast



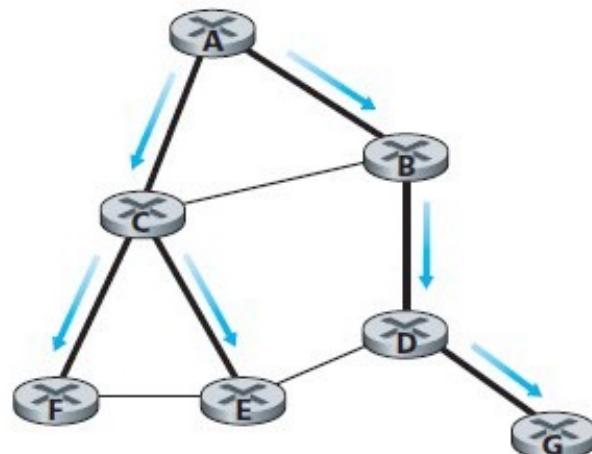
Spanning tree (center-based)

- Si prende un nodo come centro (E)
- Ogni nodo invia un messaggio di join in unicast verso il centro
 - I messaggi vengono inoltrati finchè arrivano:
 - 1) a un nodo che già appartiene all'albero
 - 2) alla radice

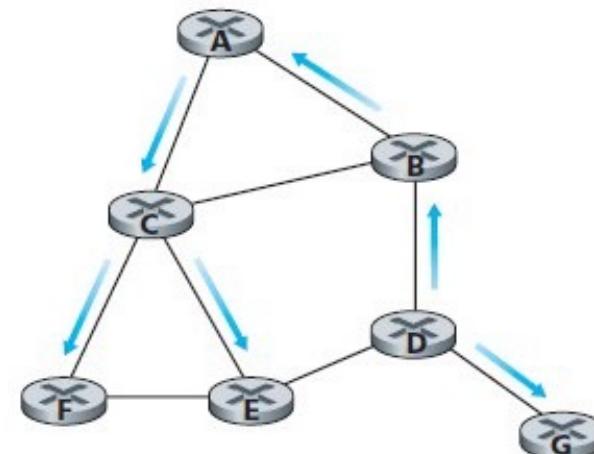


Broadcast sullo spanning tree

- I pacchetti vengono inoltrati solo sui link dell'albero, e ogni nodo riceve solo una copia del pacchetto



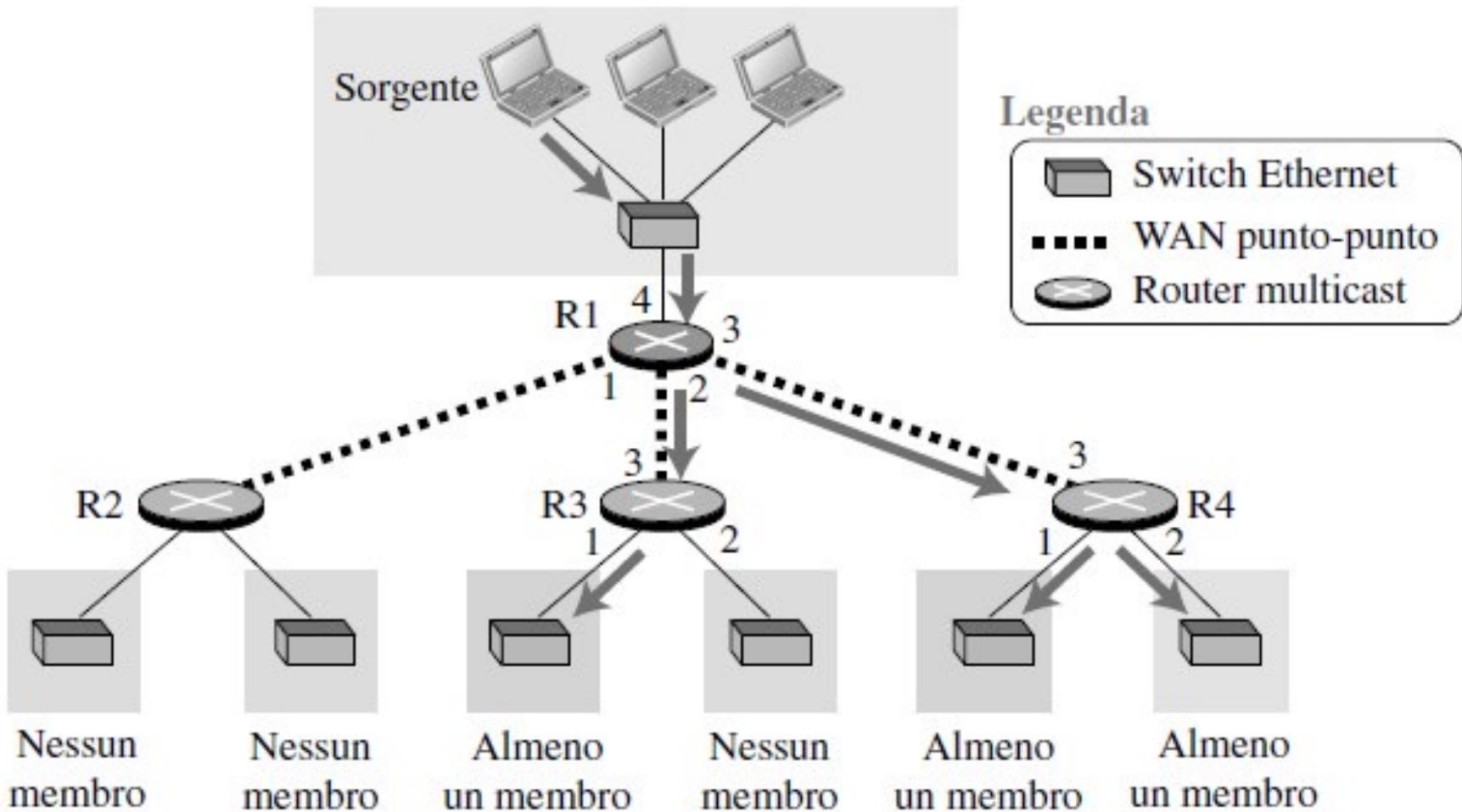
a. Broadcast initiated at A



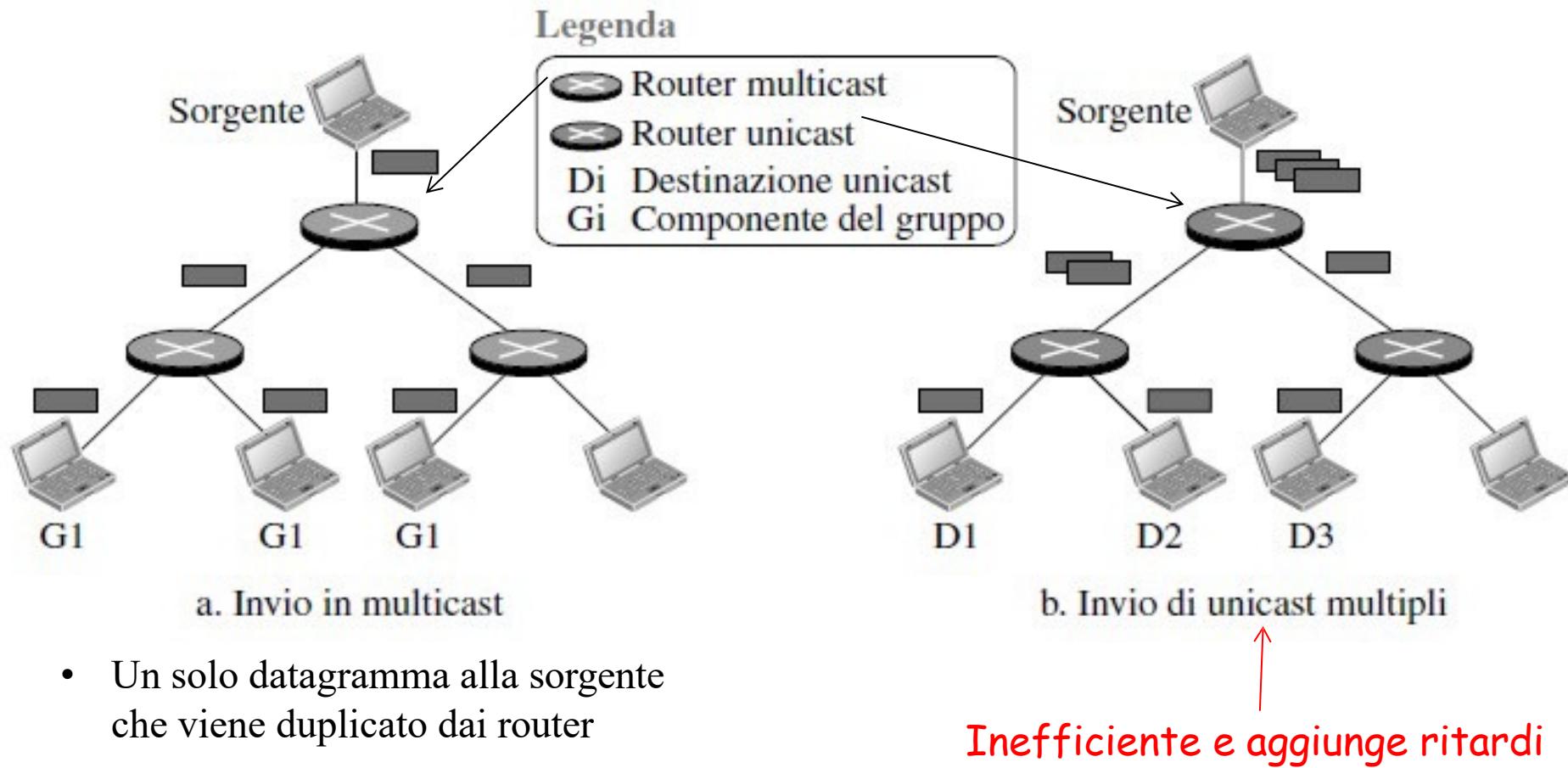
b. Broadcast initiated at D

Multicast

- MULTICAST: comunicazione tra una sorgente e un gruppo di destinazioni



Confronto tra multicast e unicast multiplo



Instradamento multicast

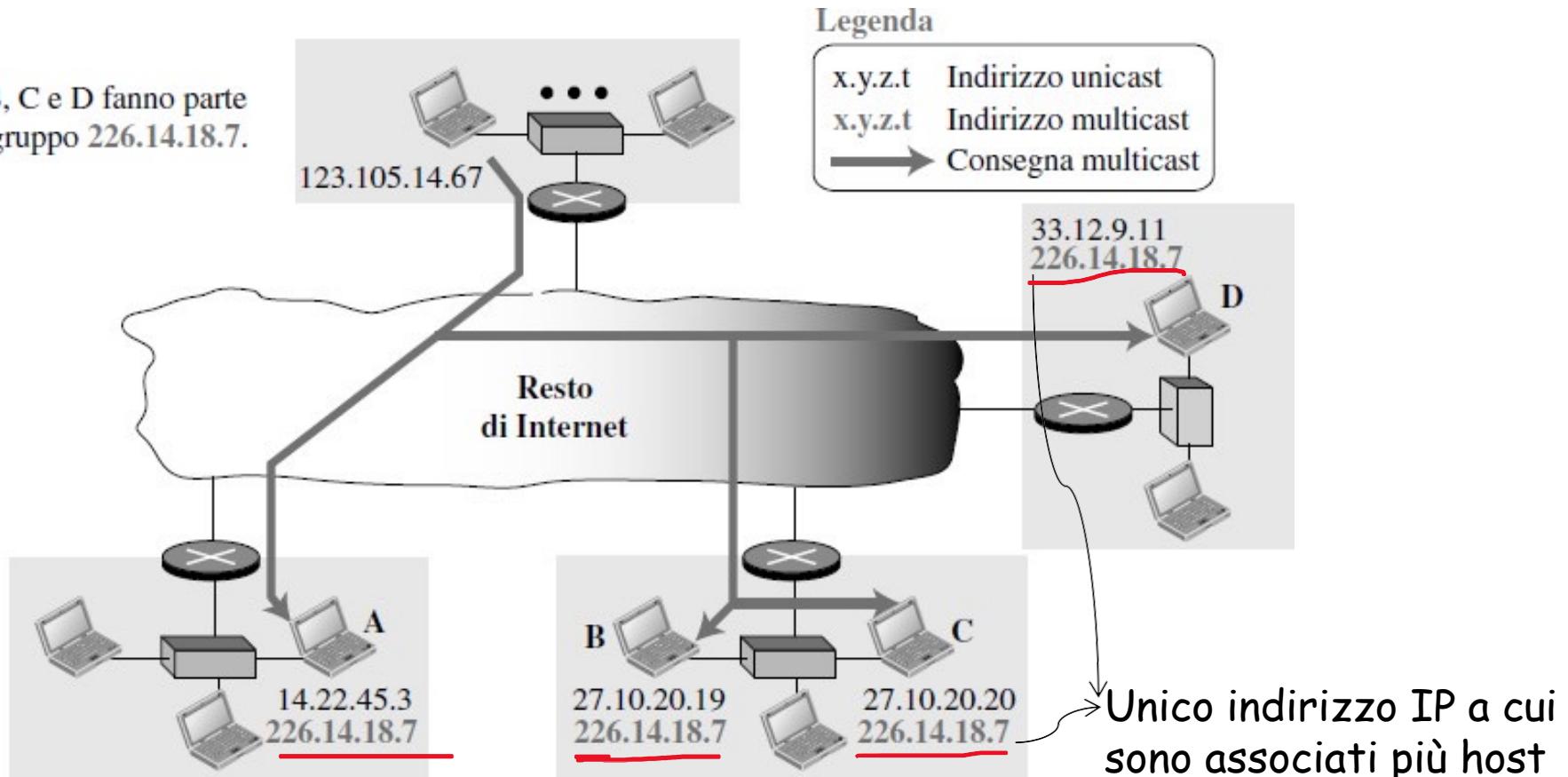
- Molte applicazioni richiedono il trasferimento di pacchetti da uno o più mittenti ad un gruppo di destinatari
 - ○ trasferimento di un aggiornamento SW su un gruppo di macchine
 - ○ streaming (audio/video) ad un gruppo di utenti o studenti
 - ○ applicazioni con dati condivisi (lavagna elettronica condivisa da più utenti)
 - ○ aggiornamento di dati (andamento di borsa)
 - ○ giochi multi-player interattivi

Problema dell'indirizzamento

- Come è possibile comunicare con host che partecipano a un gruppo ma appartengono a reti diverse?
- ES. Un gioco multi-player interattivo può coinvolgere host appartenenti a continenti diversi
- L'indirizzo di destinazione nell'IP può essere uno solo
- Soluzione: unico indirizzo per tutto il gruppo ovvero *indirizzo multicast*

Gruppo multicast

A, B, C e D fanno parte del gruppo 226.14.18.7.



I router devono sapere quali host sono associati a un gruppo multicast !!!

Indirizzi multicast

- Blocco di indirizzi riservati per il multicast
- In IPv4
 - 224.0.0.0/4
 - 1110---identificatore del gruppo---
(da 224.0.0.0 a 239.255.255.255)
 - Numero di gruppi: 2^{28}

Indirizzi multicast:

1110	group identifier
------	------------------

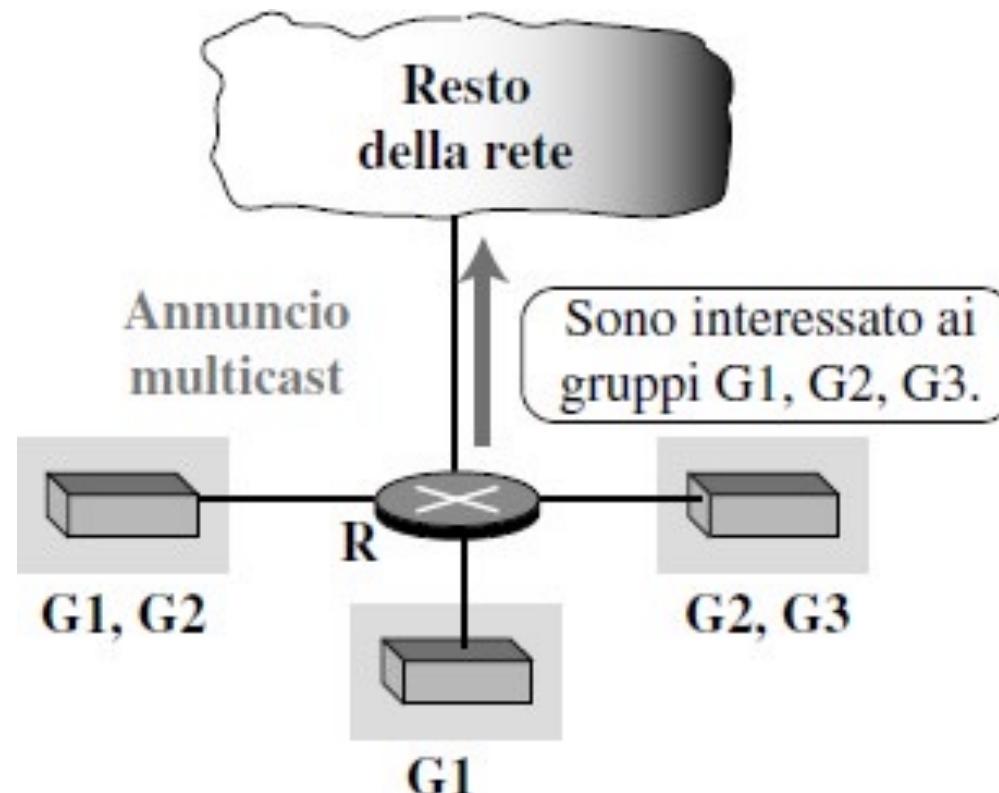
First byte: 224 to 239

Gruppi multicast

- L'appartenenza a un gruppo non ha alcuna relazione con il prefisso associato alla rete
- Un host che appartiene a un gruppo ha un indirizzo multicast separato e aggiuntivo rispetto al primario
- L'appartenenza non è un attributo fisso dell'host (periodo di appartenenza può essere limitato)
- Come può un router sapere quali host appartengono a un gruppo?

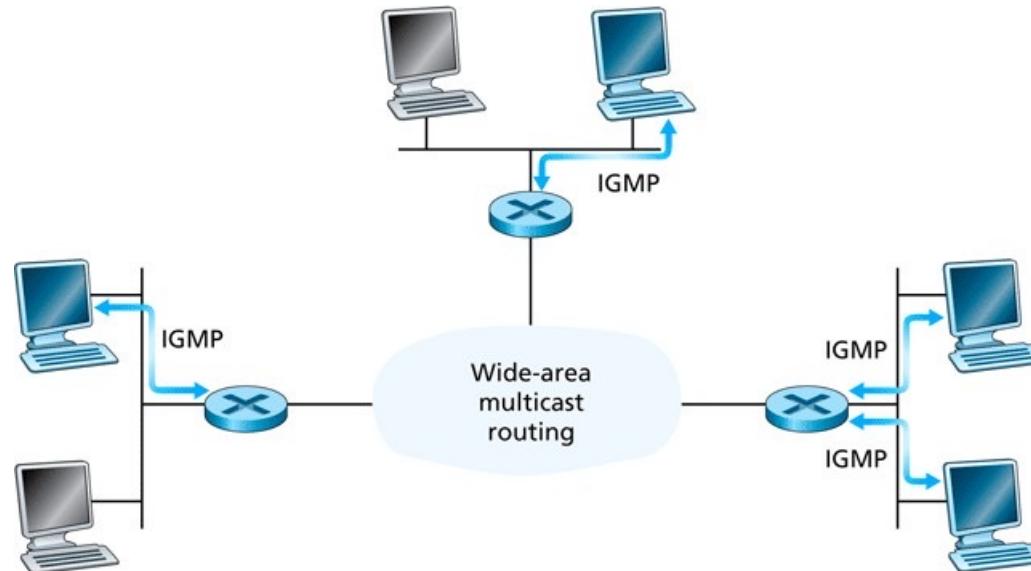
Gruppi multicast

- Un router deve scoprire quali gruppi sono presenti in ciascuna delle sue interfacce
- Il router deve propagare le informazioni agli altri router



Internet Group Management Protocol (IGMP)

- Lavora **tra un host e il router** che gli è direttamente connesso
 - Offre agli host il mezzo di informare i router ad essi connessi del fatto che un'applicazione in esecuzione vuole aderire ad uno specifico gruppo multicast



IGMP

- Messaggi incapsulati in datagrammi IP, con IP protocol number 2
 - Mandati con TTL a 1

- Messaggi IGMP

- **Membership query**: router → host, per determinare a quali gruppi hanno aderito gli host su ogni interfaccia (invia periodicamente)
- **Membership report**: host → router, per informare il router su un'adesione, anche non in seguito a una query (al momento dell'adesione)
- **Leave group**: host → router, quando si lascia un gruppo
Il leave group è opzionale: il router può capire che non ci sono più host associati a un gruppo quando non riceve report in risposta a query

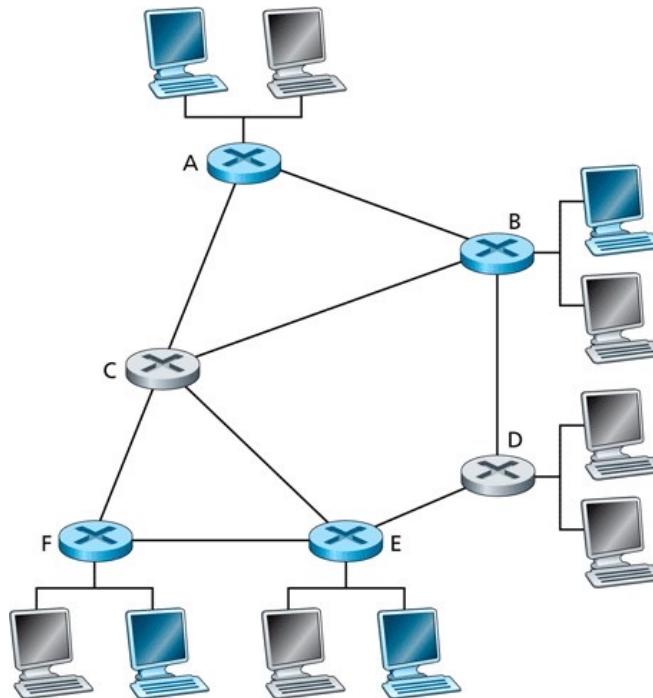


IGMP

- Un router multicast tiene una lista per ciascuna sottorete dei gruppi multicast (multicast group membership → almeno un elemento del gruppo fa parte della sottorete) con un timer per membership
 - la membership deve essere aggiornata da report inviati prima della scadenza del timer
 - può essere anche aggiornata tramite messaggi di leave esplicativi

Problema del routing multicast

- Fra la popolazione complessiva di router solo alcuni (quelli collegati a host del gruppo multicast) dovranno ricevere traffico multicast
- È necessario un protocollo che coordini i router multicast in Internet (instradare pacchetti multicast dalla sorgente alla destinazione)



A,B,E,F sono router che devono ricevere traffico multicast

Obiettivo: trovare un albero che colleghi tutti i router connessi ad host che appartengono al gruppo multicast. I pacchetti verranno instradati su questo albero. Albero può essere unico per tutto il gruppo o diverso a seconda della sorgente

Instradamento multicast in Internet

Intra-dominio multicast (interno a un sistema autonomo)

- DVMRP: distance-vector multicast routing protocol
- MOSPF: multicast open shortest path first
- PIM: protocol independent multicast

Inter-dominio multicast (tra sistemi autonomi)

- MBGP: multicast border gateway protocol

Livello di rete – piano di controllo: sommario

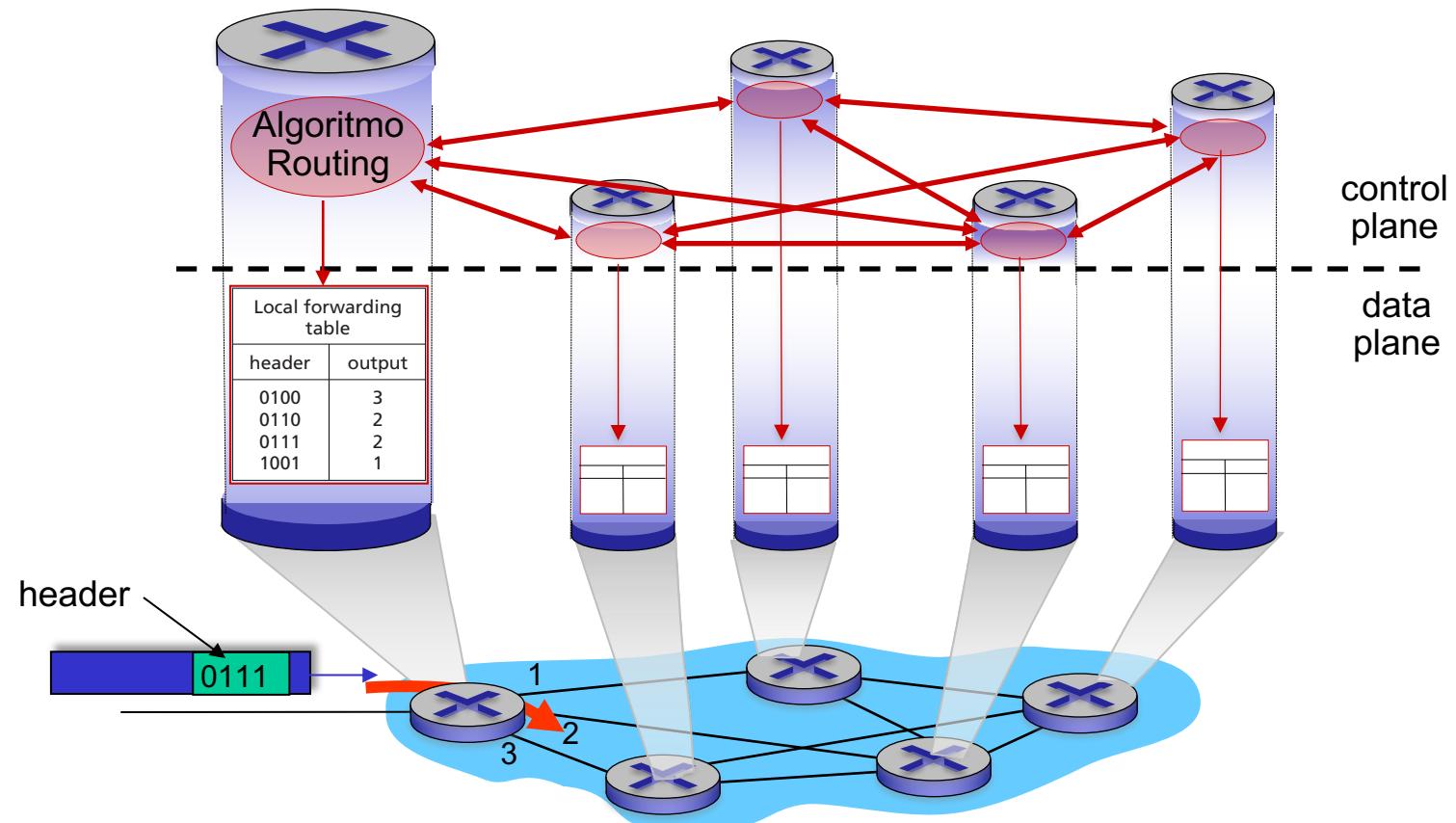
- introduzione
- algoritmi di instradamento
 - link state
 - distance vector
- instradamento intra-ISP: RIP e OSPF
- instradamento tra ISP: BGP
- multicast e IGMP
- **piano di controllo SDN**
- gestione della rete, configurazione
 - SNMP
 - NETCONF/YANG

Software defined networking (SDN)

- Livello di rete Internet: storicamente implementato tramite un approccio di controllo distribuito sui router:
 - un router *monolitico* contiene hardware di commutazione, esegue implementazioni proprietarie dei protocolli standard Internet (IP, RIP, IS-IS, OSPF, BGP) in sistemi operativi proprietari (ad es. Cisco IOS)
 - diversi "middlebox" per diverse funzioni del livello di rete: firewall, load balancing, NAT, ..
- ~2005: rinnovato interesse nel ripensare il piano di controllo della rete

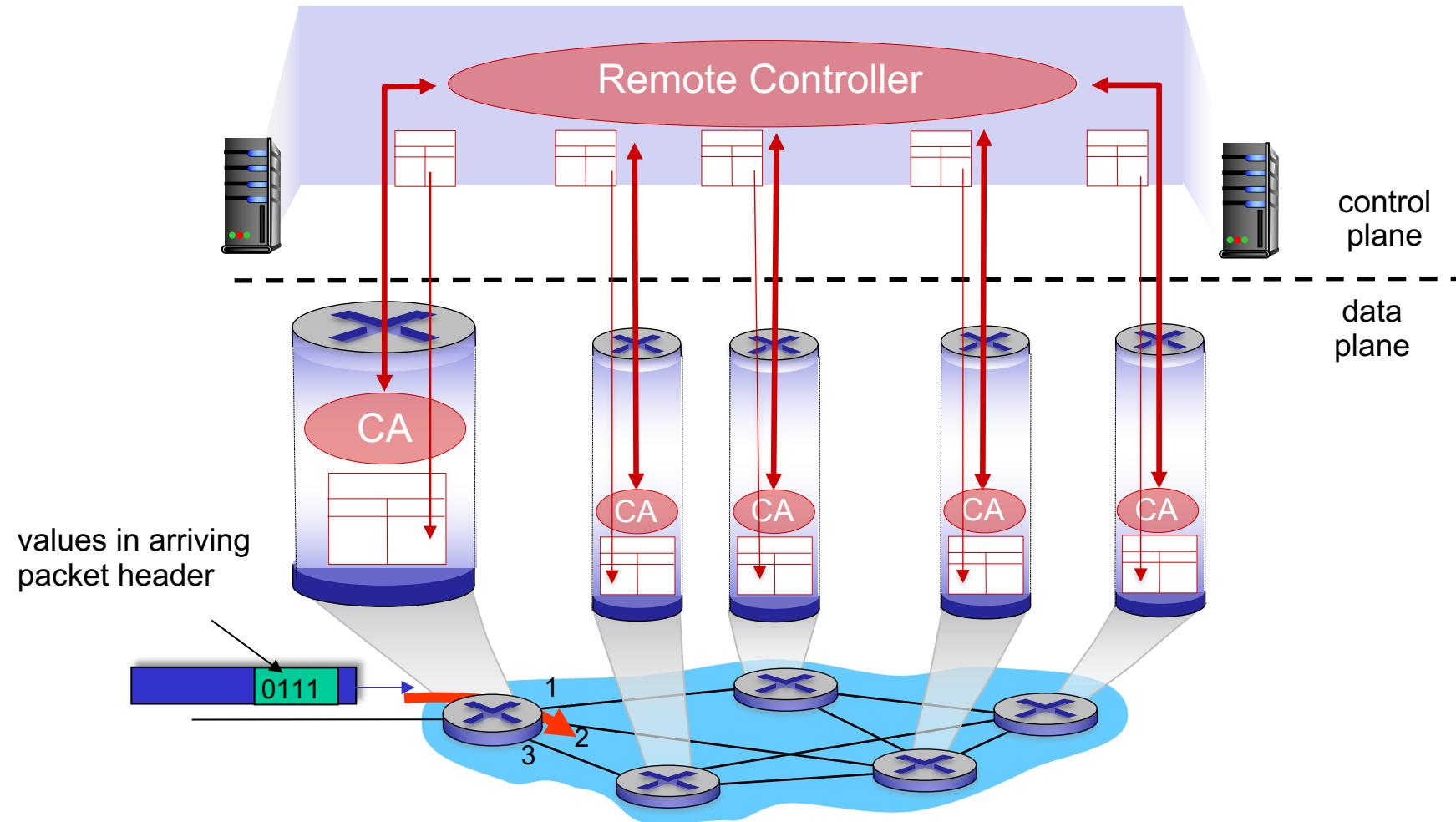
Piano di controllo di ogni router

I singoli componenti dell'algoritmo di routing *in ogni singolo router* interagiscono nel piano di controllo



Software-Defined Networking (SDN) control plane

Il controller remoto calcola e poi installa le tabelle di inoltro nei router

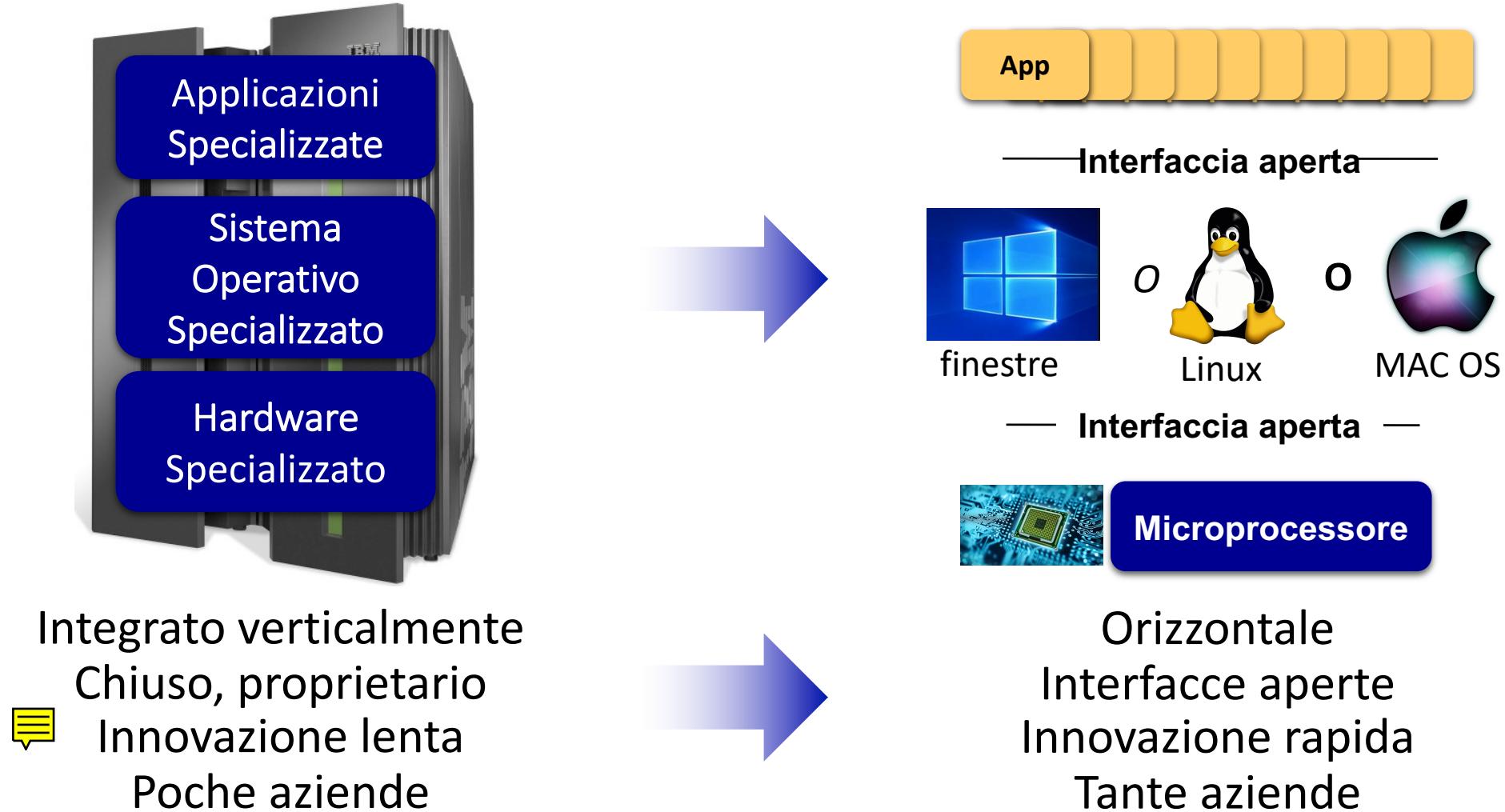


Software defined networking (SDN)

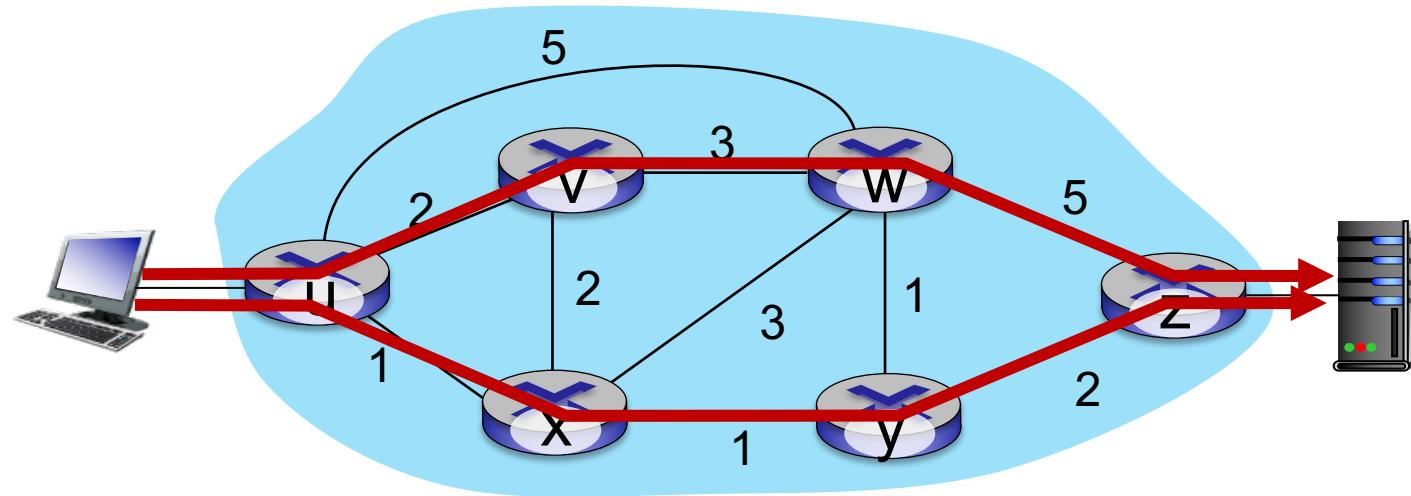
*Perché UN piano di controllo *logicamente centralizzato* ?*

- gestione della rete più semplice: evitare errori di configurazione del router, maggiore flessibilità dei flussi di traffico
- match+action basato su tabella (cfr OpenFlow API) consente la "programmazione" dei router
 - "programmazione" centralizzata più semplice: calcola le tabelle centralmente e distribuisce
 - "programmazione" distribuita più difficile: calcolo delle tabelle come risultato dell'algoritmo distribuito (protocollo) implementato in ogni singolo router
- implementazione aperta (non proprietaria) del piano di controllo
 - favorire l'innovazione

Analogia SDN: rivoluzione da mainframe a PC



Traffic engineering: difficile con il routing tradizionale

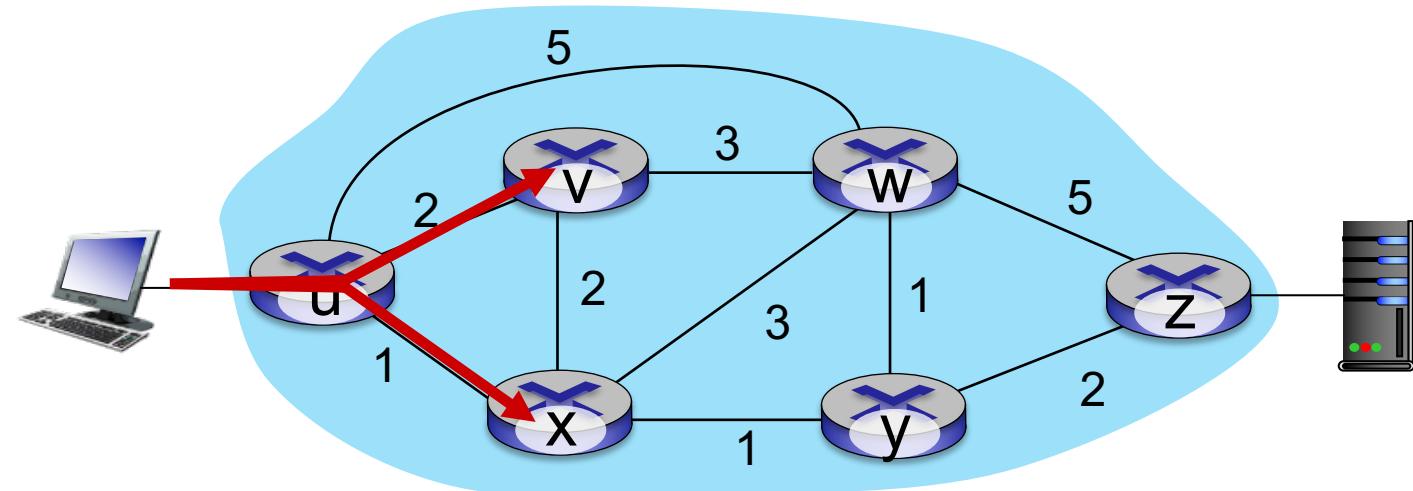


D: Come fare se l'operatore di rete desidera che il traffico da u a z scorra lungo uvwz, anziché uxzy?

R: è necessario ridefinire i pesi dei collegamenti in modo che l'algoritmo di instradamento del traffico calcoli i percorsi di conseguenza (o è necessario un nuovo algoritmo di instradamento)!

i costi dei link sono le uniche “manopole” di controllo: poco controllo!

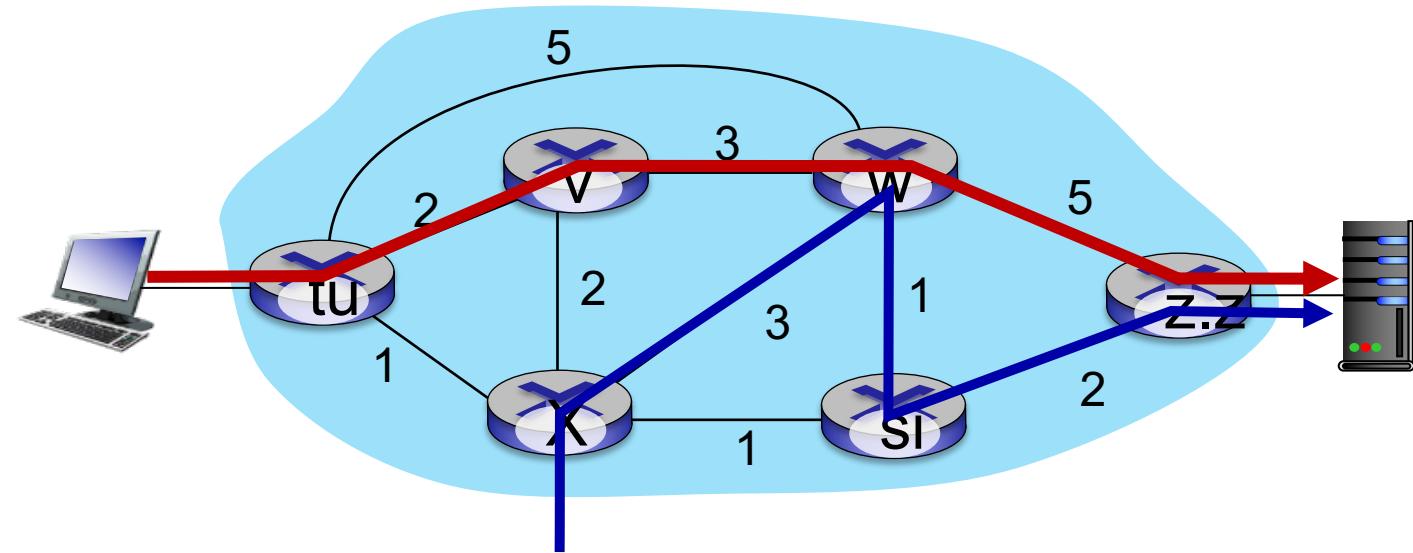
Traffic engineering: difficile con il routing tradizionale



D: come fare se l'operatore di rete desidera suddividere il traffico u-z tra uvwz **e** uxyz (bilanciamento del carico)?

R: non può farlo (o serve un nuovo algoritmo di routing)

Ingegneria del traffico: difficile con il routing tradizionale



D: come fare se w vuole instradare il traffico blu e rosso in modo diverso da w a z?

R: non posso farlo (con inoltro basato sulla destinazione e instradamento LS, DV)

Abbiamo appreso nel Capitolo 4 che l'inoltro generalizzato e le SDN possono essere utilizzate per ottenere *qualsiasi* instradamento desiderato

Software defined networking (SDN)



4. applicazioni di controllo programmabili

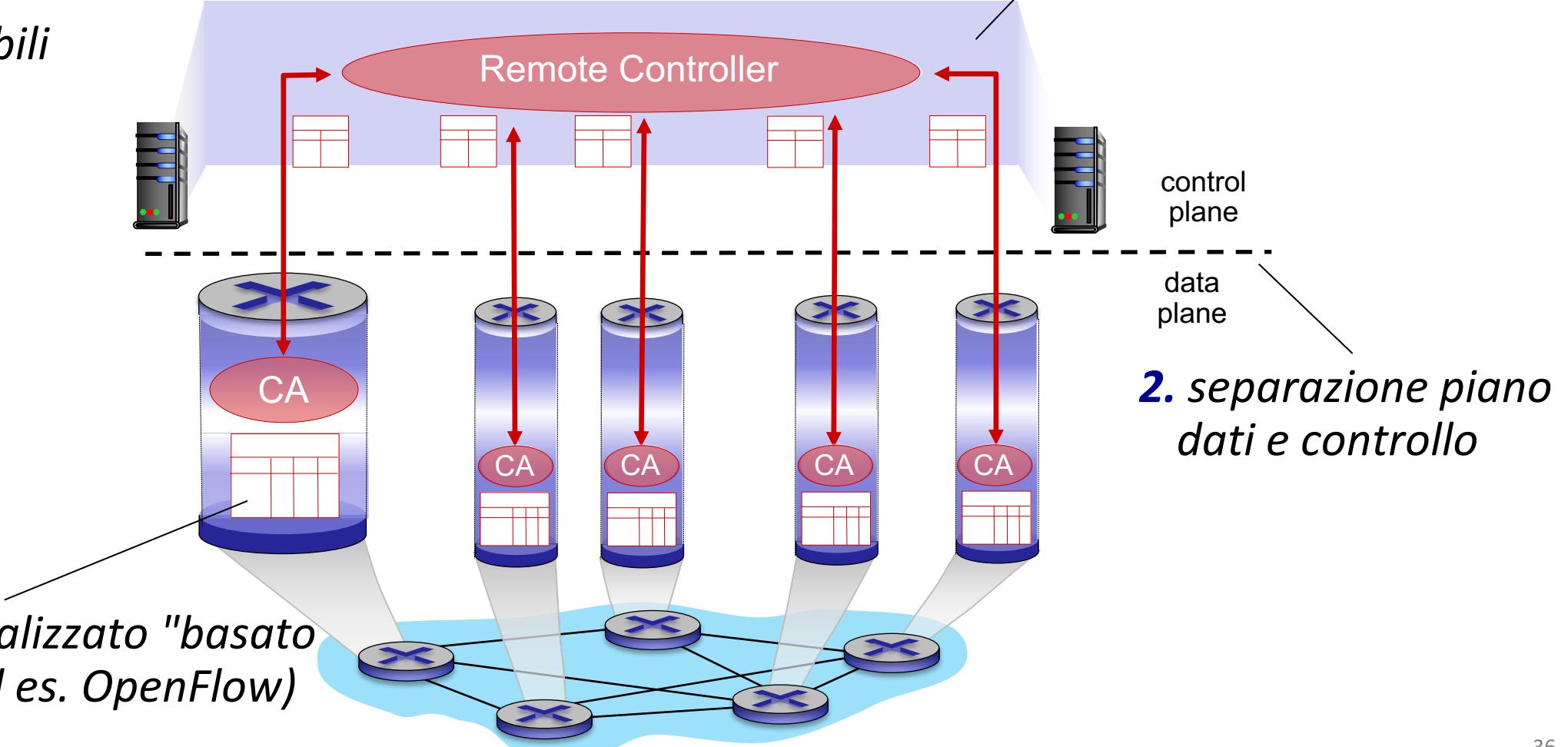
routing

access control

...

load balance

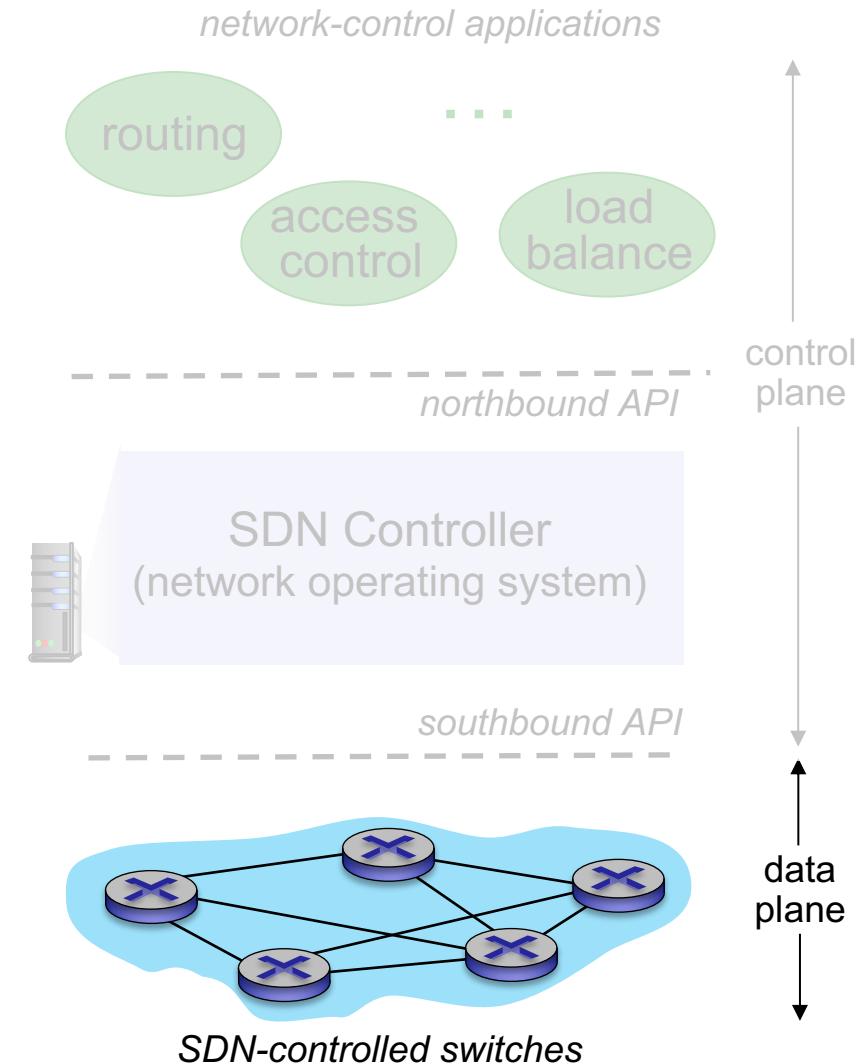
3. funzioni del piano di controllo addizionali al classico switching del data plane



Software defined networking (SDN)

Switch del piano dati:

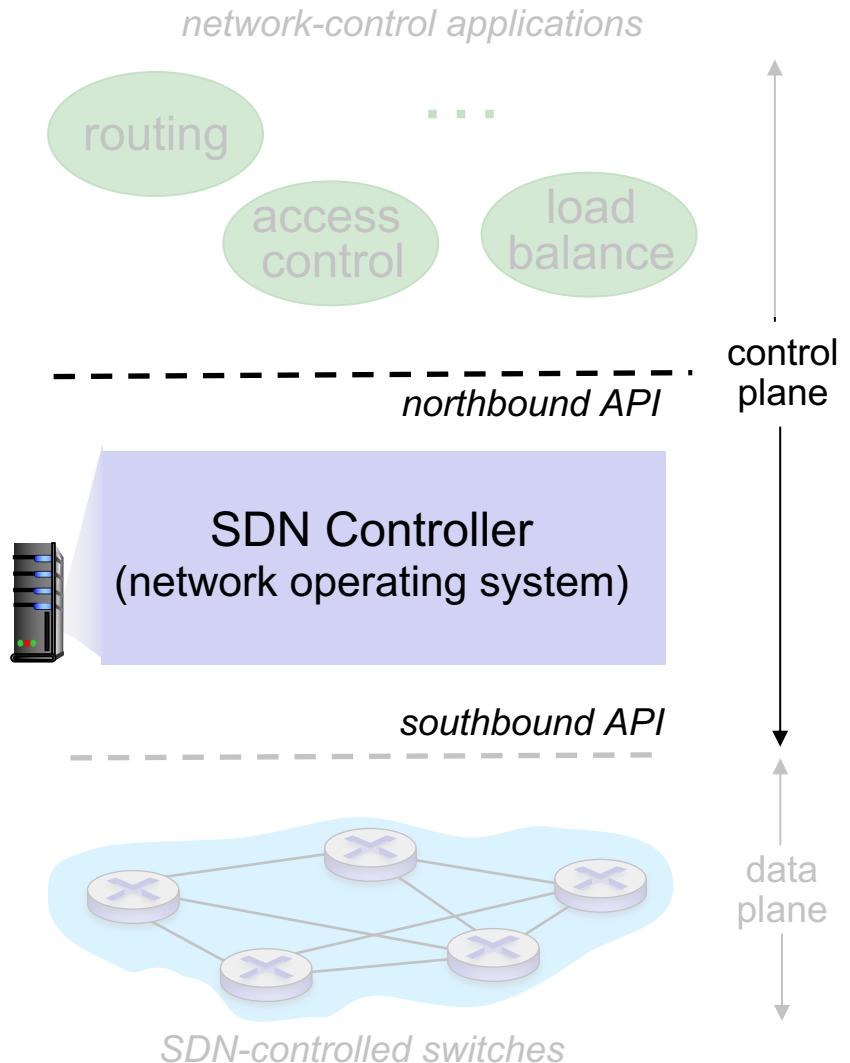
- Abbiamo visto che esistono switch veloci e semplici che implementano l'inoltro generalizzato del piano dati lato hardware
- tabella di flusso calcolata e installata sotto la supervisione del controller
- API per il controllo degli switch basato su tabella (ad es. OpenFlow)
 - definisce ciò che è controllabile
- protocollo per la comunicazione con il controller (ad es. OpenFlow)



Software defined networking (SDN)

Controller SDN (sistema operativo di rete):

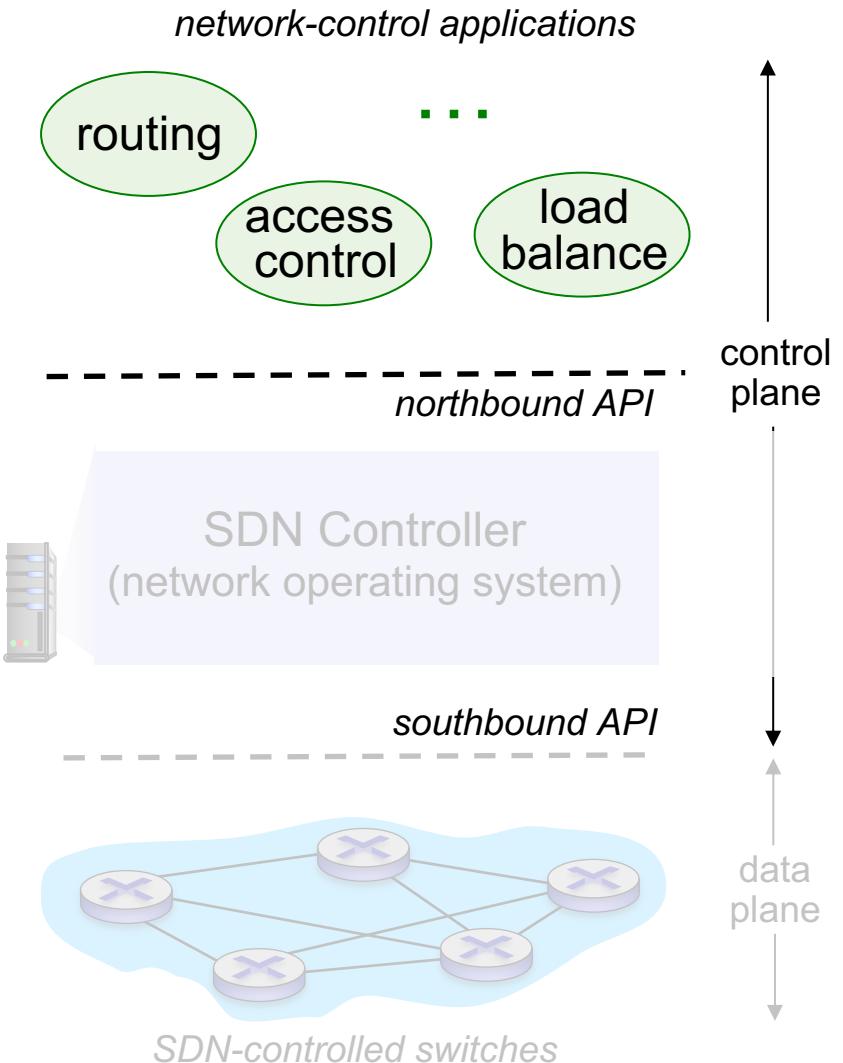
- mantenere informazioni sullo stato della rete
- interagisce con le applicazioni di controllo della rete al di sopra tramite l'API Northbound
- interagisce con gli switch di rete sotto tramite l'API southbound
- implementato come sistema distribuito per prestazioni, scalabilità, tolleranza ai guasti, robustezza



Software defined networking (SDN)

network-control apps:

- "cervelli" di controllo: implementano funzioni di controllo utilizzando servizi di livello inferiore, API fornite dal controller SDN
- *unbundled*: può essere fornito da terze parti distinte dal fornitore degli switch o del controller SDN

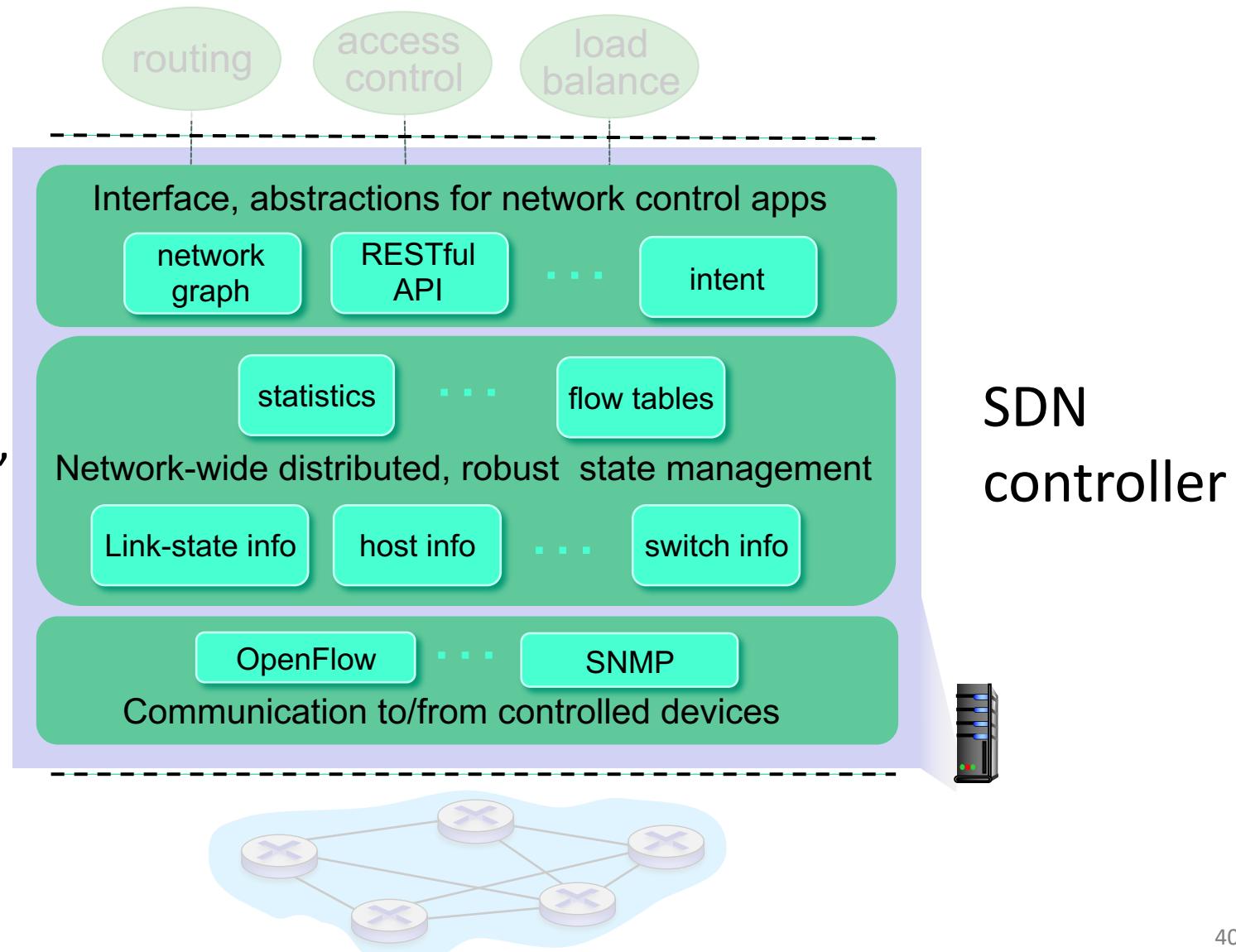


Componenti del controller SDN

interface layer to network control apps: API di astrazione

network-wide state management: stato dei collegamenti, degli switch, dei servizi delle reti: un *database distribuito*

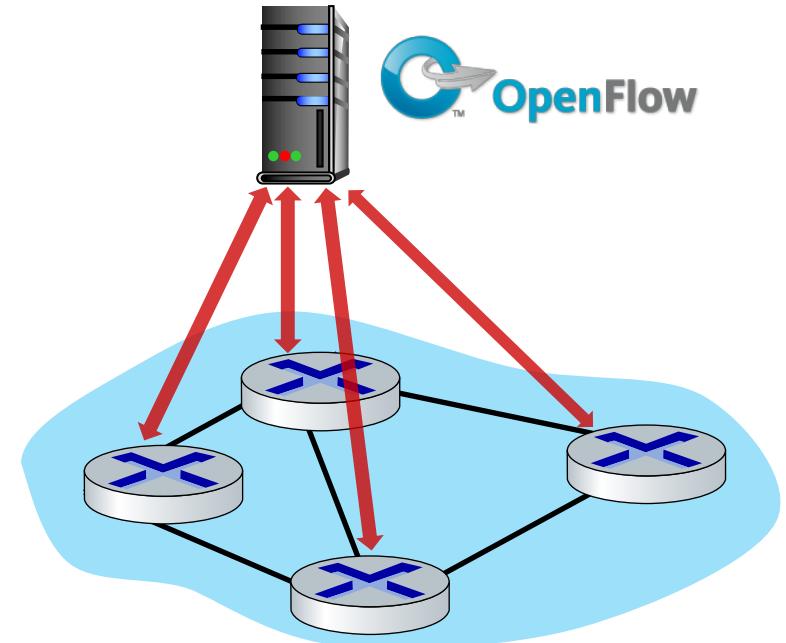
communication: comunicazione tra il controller SDN e gli switch controllati



Protocollo OpenFlow

- comunicazione tra controller e switch
- TCP usato per scambiare messaggi
 - crittografia opzionale
- tre classi di messaggi OpenFlow:
 - controller-to-switch
 - asincrono (dallo switch al controller)
 - simmetrico
- DIVERSO dall'API OpenFlow
 - API utilizzata per specificare azioni di inoltro generalizzate

Controller OpenFlow

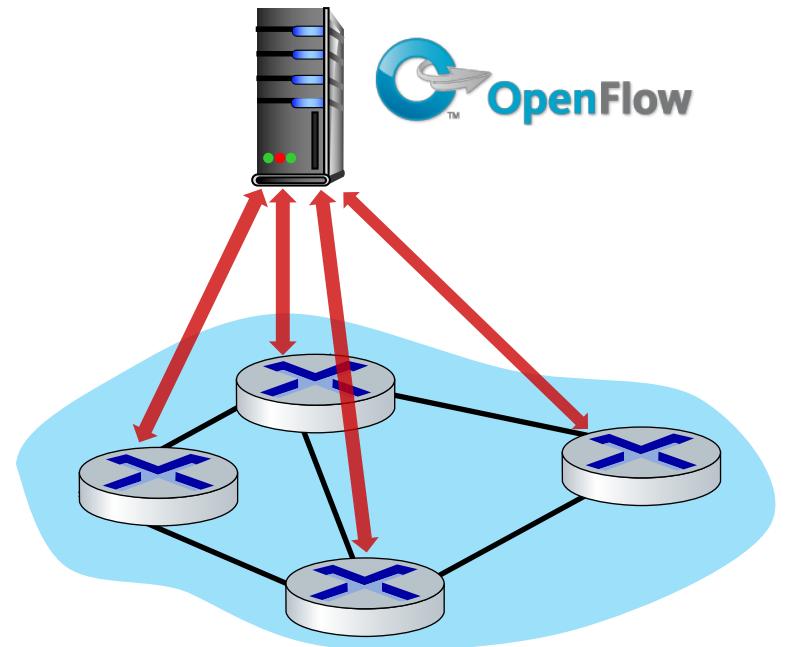


OpenFlow: messaggi da controller a switch

Messaggi chiave da controller a switch

- ***features***: query del controller per conoscere le funzionalità supportate dallo switch (ad es rate limiting può non essere implementato)
- ***configure***: il controller interroga/imposta parametri di configurazione dello switch
- ***modify-state***: aggiungi, elimina, modifica campi nelle flow table OpenFlow
- ***packet-out***: il controller può inviare un pacchetto da una specifica porta dello switch

Controller OpenFlow

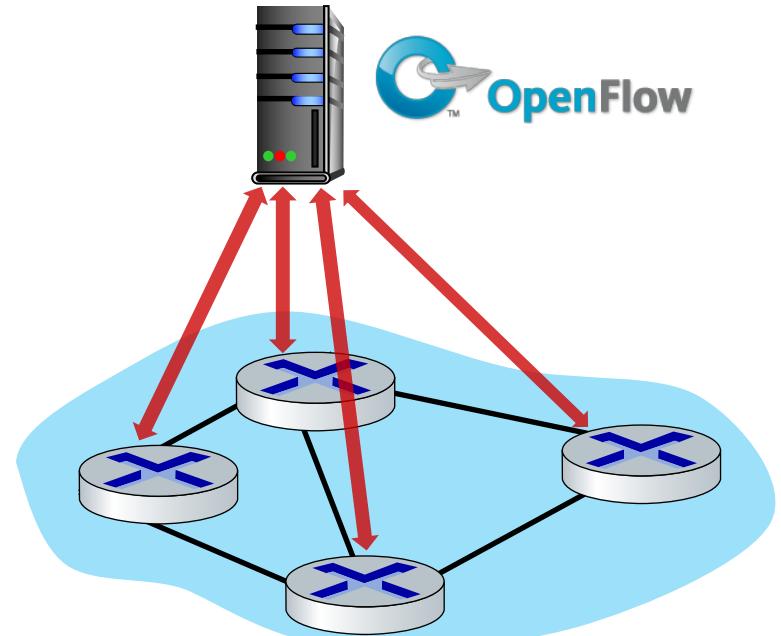


OpenFlow: messaggi dallo switch al controller

Messaggi chiave dallo switch al controller

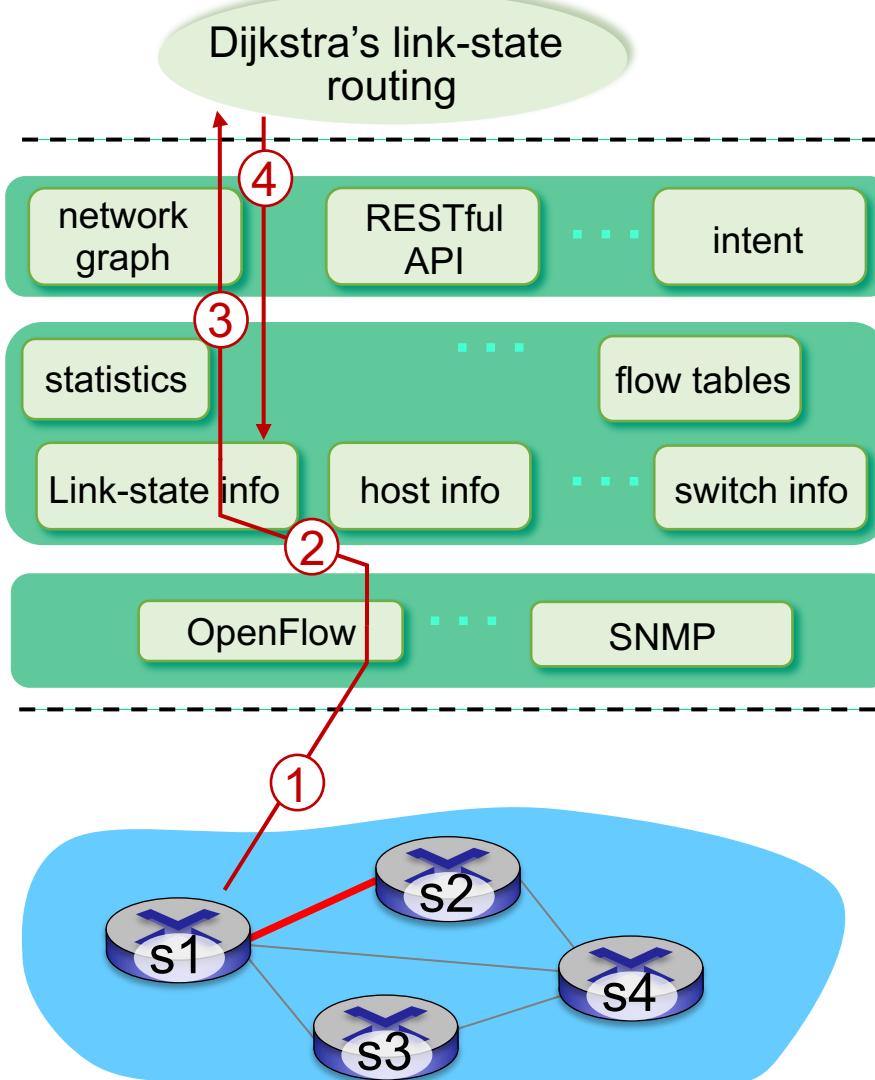
- ***packet-in***: trasferisce il pacchetto al controller (poi con packet-out il controller può gestirlo)
- ***flow-removed***: riga della tabella di flusso eliminata allo switch
- ***port status***: informa il controllore di una modifica su una porta

Controller OpenFlow



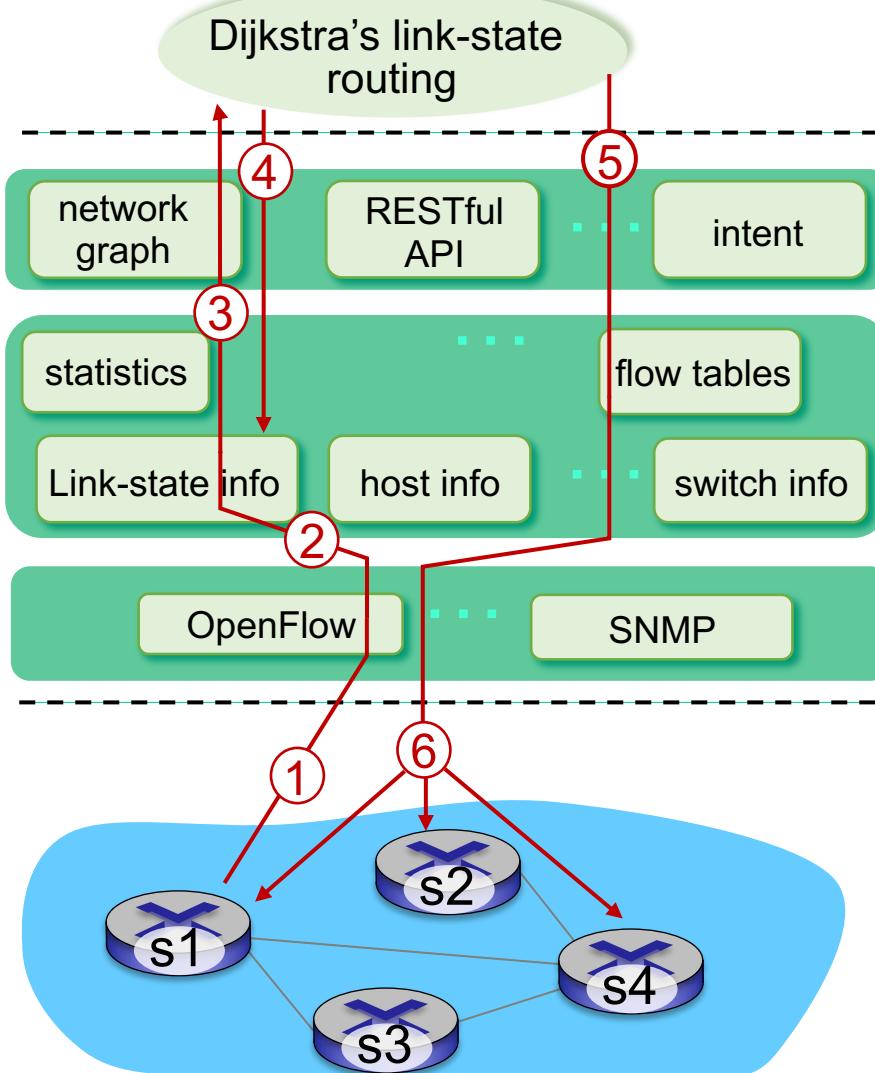
Gli operatori di rete non "programmano" gli switch inviando direttamente messaggi OpenFlow. Usano invece l'astrazione di livello superiore al controller

SDN: esempio di interazione piano dati/controllo



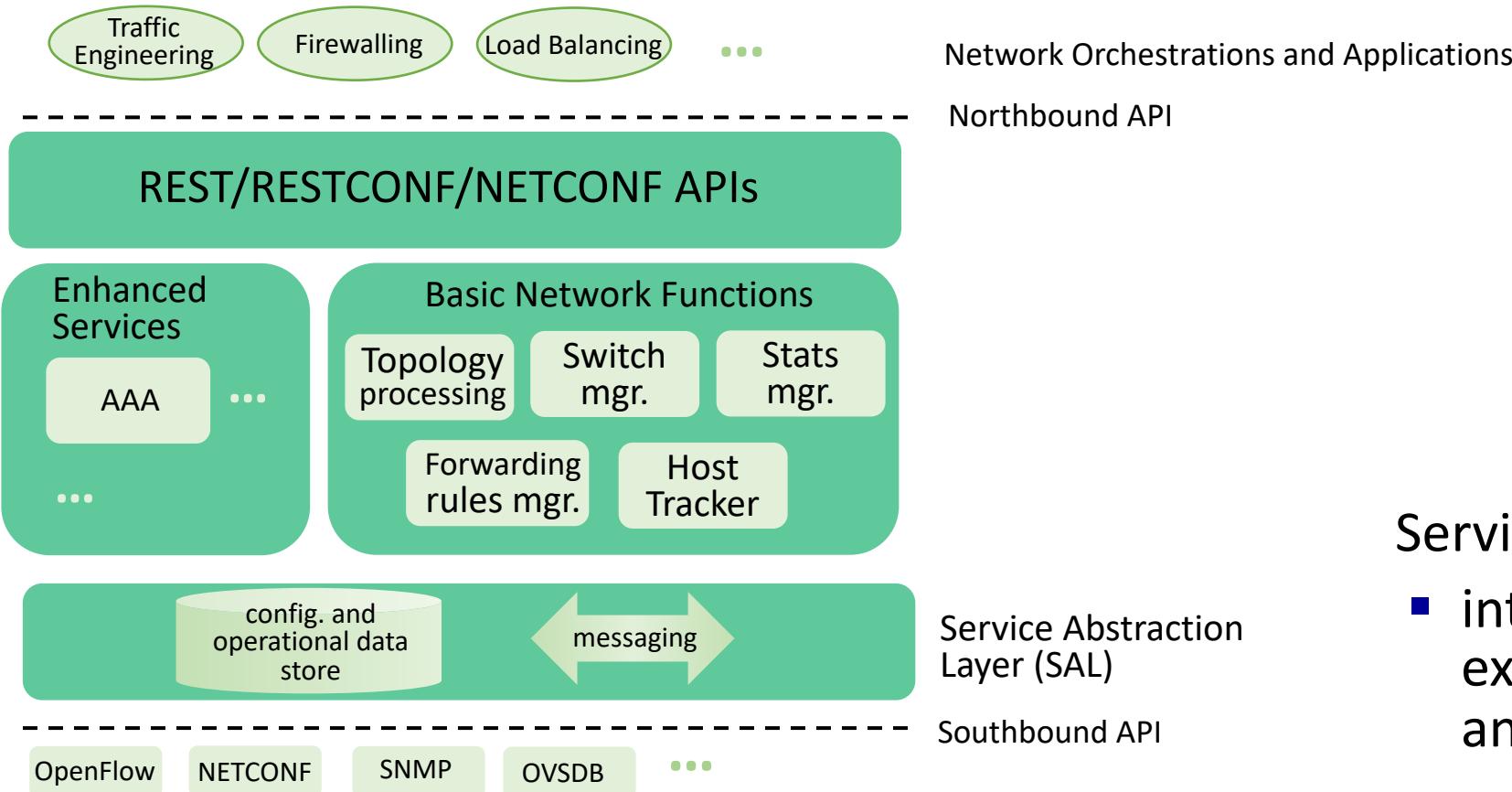
- ① S1, in cui si verifica un errore di collegamento, utilizza il messaggio di stato della porta OpenFlow per informare il controller
- ② Il controller SDN riceve il messaggio OpenFlow, aggiorna le informazioni sullo stato del collegamento
- ③ L'applicazione dell'algoritmo di routing (Dijkstra) è stata precedentemente registrata per essere chiamata ogni volta che lo stato del collegamento cambia. Viene chiamata
- ④ L'algoritmo di instradamento di Dijkstra accede alle informazioni sul grafo di rete, alle informazioni sullo stato dei collegamenti nel controller, calcola nuovi percorsi

SDN: esempio di interazione piano dati/controllo



- ⑤ l'app di routing di link state interagisce con il componente di calcolo della tabella di flusso nel controller SDN, che calcola le nuove tabelle di flusso necessarie
- ⑥ il controller utilizza OpenFlow per installare nuove tabelle negli switch che devono essere aggiornati

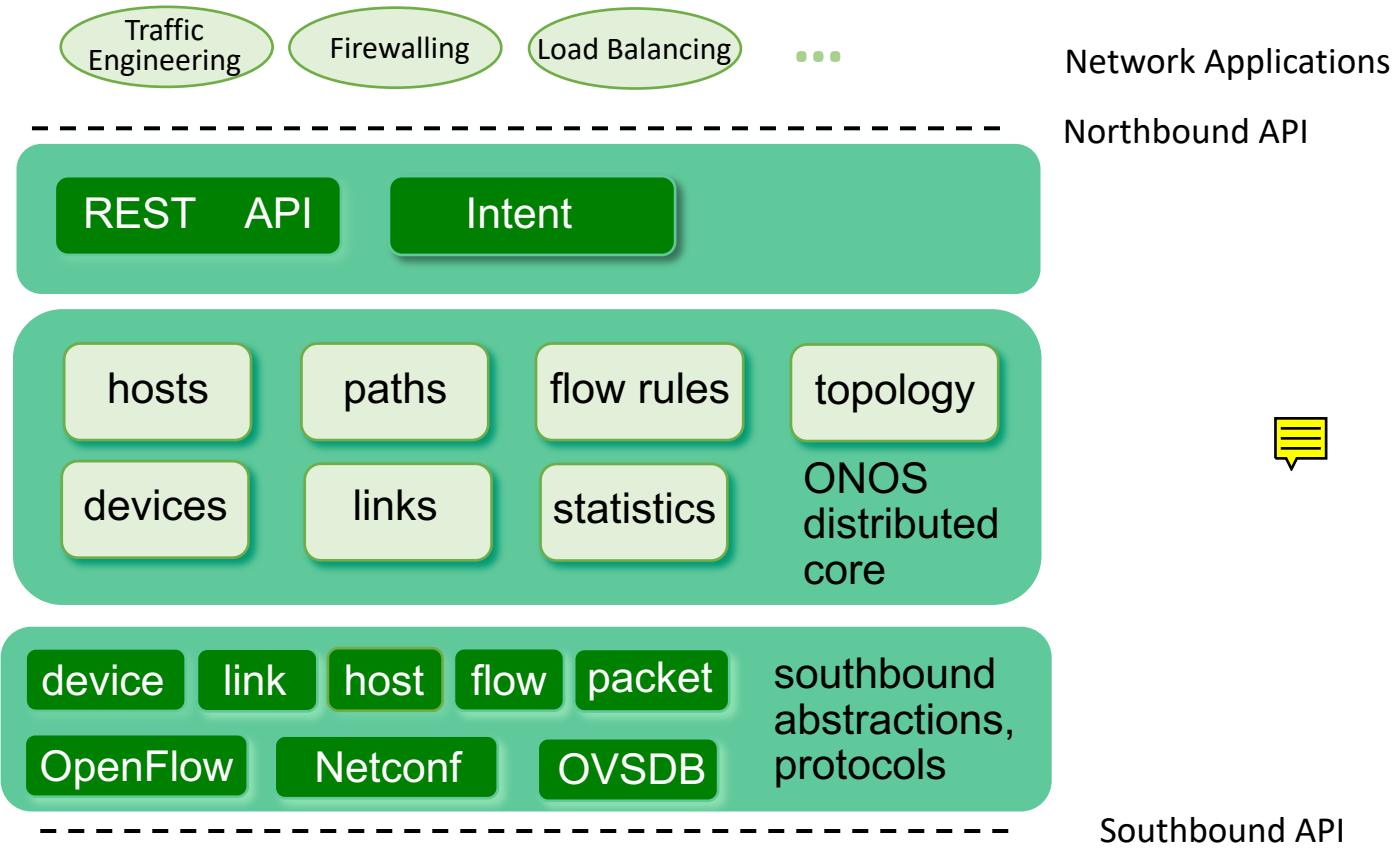
OpenDaylight (ODL) controller



Service Abstraction Layer:

- interconnects internal, external applications and services

ONOS controller



- control apps separate from controller
- intent framework: high-level specification of service: what rather than how
- considerable emphasis on distributed core: service reliability, replication performance scaling

SDN: challenges

- rafforzamento del piano di controllo: sistema distribuito affidabile, scalabile in termini di prestazioni e sicuro
 - robustezza ai guasti: sfruttare la teoria dei sistemi distribuito affidabile per il piano di controllo
 - affidabilità, sicurezza: fin dal primo giorno? Quasi (versione aggiornata di OpenFlow usa autenticazione)
- reti e protocolli che soddisfino specifici requisiti in un framework best effort (IP)
 - ad esempio: tempo reale, ultra affidabile, ultra sicuro
- Scalabilità Internet: come andare oltre un singolo AS? Per ora SDN è solo un sostituto del routing intra-AS tradizionale
- SDN fondamentale nelle reti cellulari 5G



SDN e il futuro dei protocolli di rete tradizionali

- Tabelle di inoltro calcolate da SDN invece che a livello di router
 - È solo un esempio di calcolo centralizzato
 - si potrebbe immaginare ad es. controllo della congestione calcolato da SDN:
 - il controller imposta i rate del mittente in base ai livelli di congestione segnalati dal router (al controller).



Come evolverà
l'implementazione della
funzionalità di rete?



Livello di rete – piano di controllo: sommario

- introduzione
- algoritmi di instradamento
 - link state
 - distance vector
- instradamento intra-ISP: RIP e OSPF
- instradamento tra ISP: BGP
- multicast e IGMP
- piano di controllo SDN
- gestione della rete, configurazione
 - SNMP
 - NETCONF/YANG

Cos'è il network management?

- sistemi autonomi (domini o sottorete): migliaia di componenti hardware/software interagenti
- altri sistemi complessi che richiedono monitoraggio, configurazione, controllo:
 - aereo a reazione, centrale nucleare,...



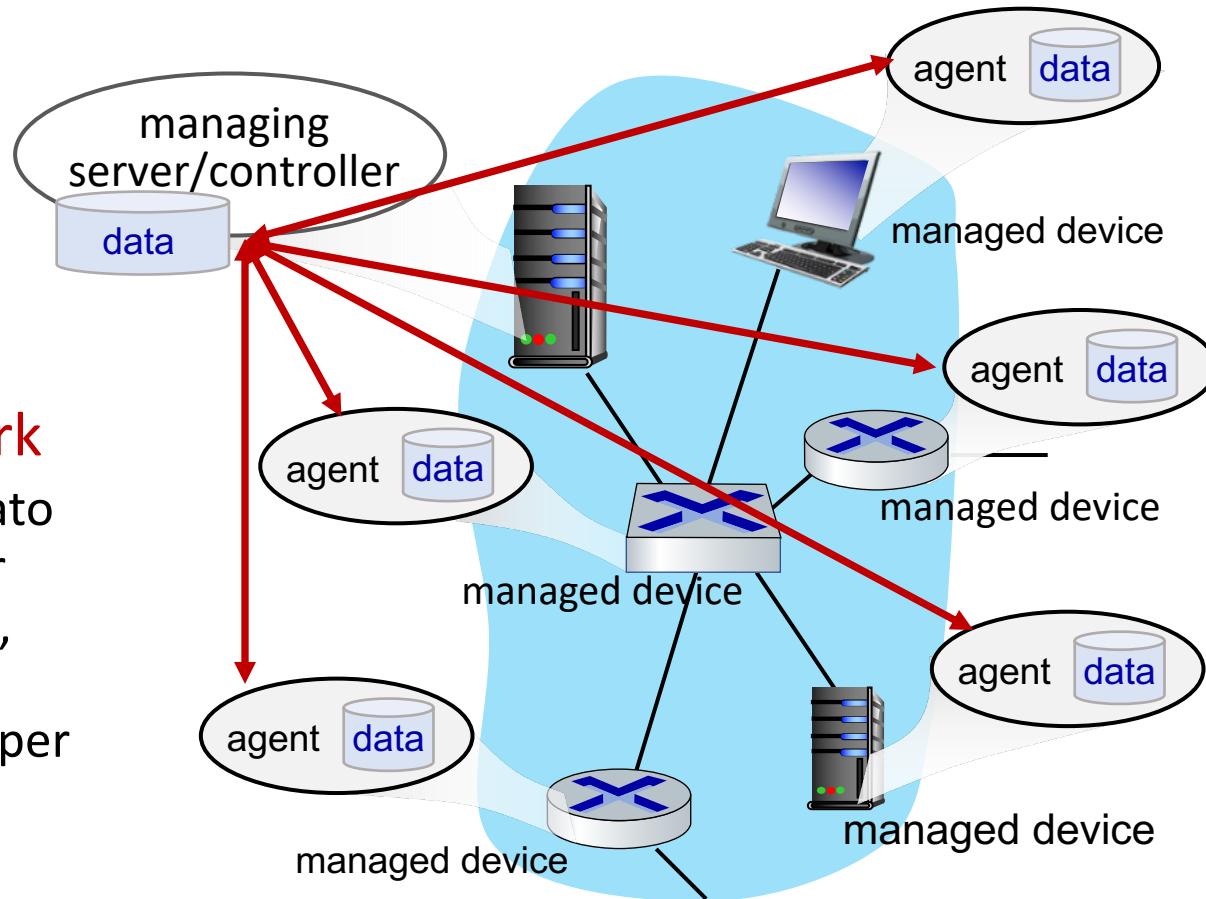
"La gestione della rete include la distribuzione, l'integrazione e coordinamento dell'hardware, del software e di elementi umani per monitorare, testare, sondare, configurare, analizzare, valutare, e controllare la rete e le risorse per soddisfare i requisiti di prestazioni operative e qualità del servizio a un costo ragionevole".



Componenti del network management

Managing server:
applicazione,
tipicamente con
amministratori della
rete (humans) in the
loop

**Protocollo di network
management:** utilizzato
dal managing server per
interrogare, configurare,
gestire il dispositivo;
utilizzato dai dispositivi per
informare il server di
gestione di dati, eventi.



Managed device:
apparecchiature con
componenti hardware e
software gestibili e
configurabili

Dati: dati di
configurazione dello
"stato" del dispositivo,
dati operativi,
statistiche del
dispositivo

L'approccio dell'amministratore di rete alla gestione

CLI (Command Line Interface)

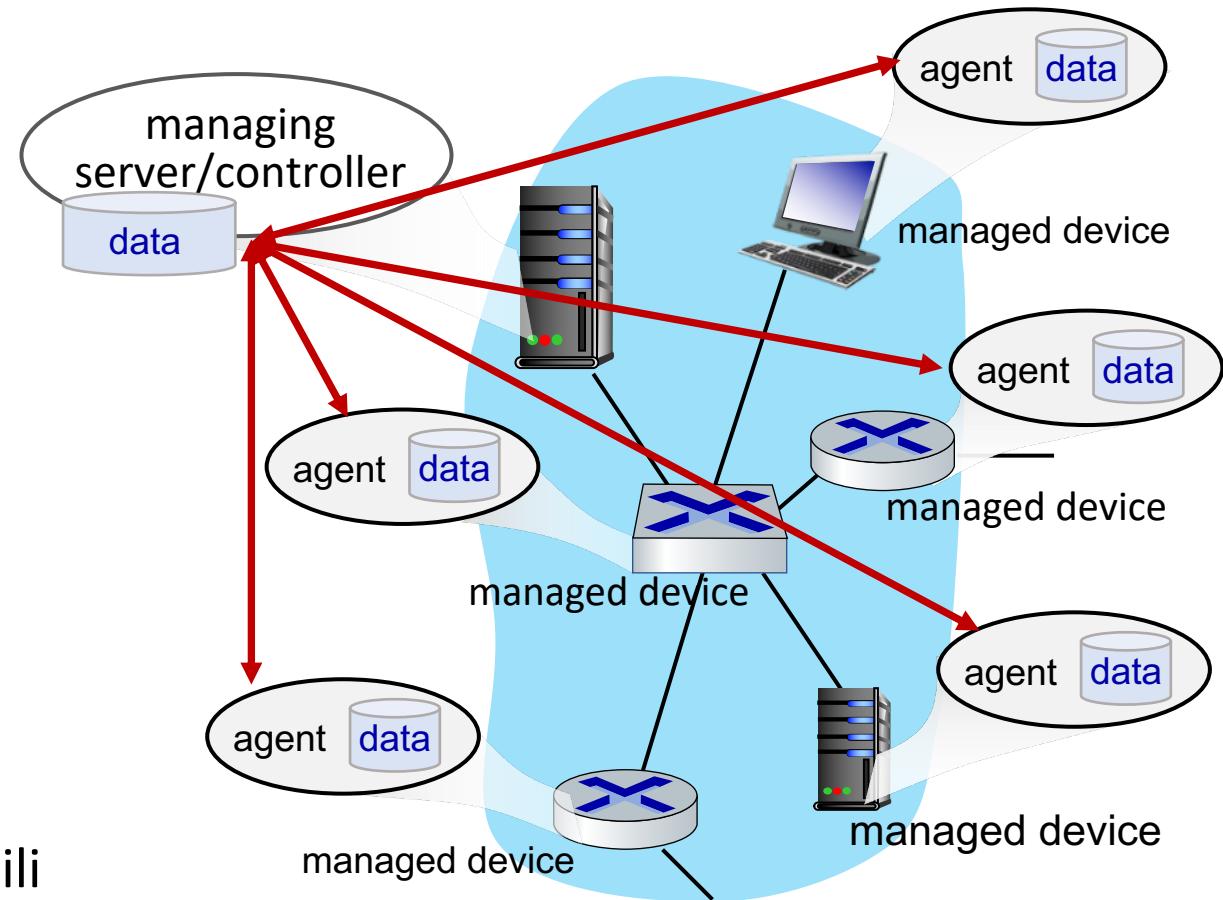
- l'operatore esegue comandi diretti ai singoli dispositivi (ad esempio, vis ssh)

SNMP/MIB

- l'operatore interroga/imposta i dati dei dispositivi (MIB) utilizzando il Simple Network Management Protocol (SNMP)

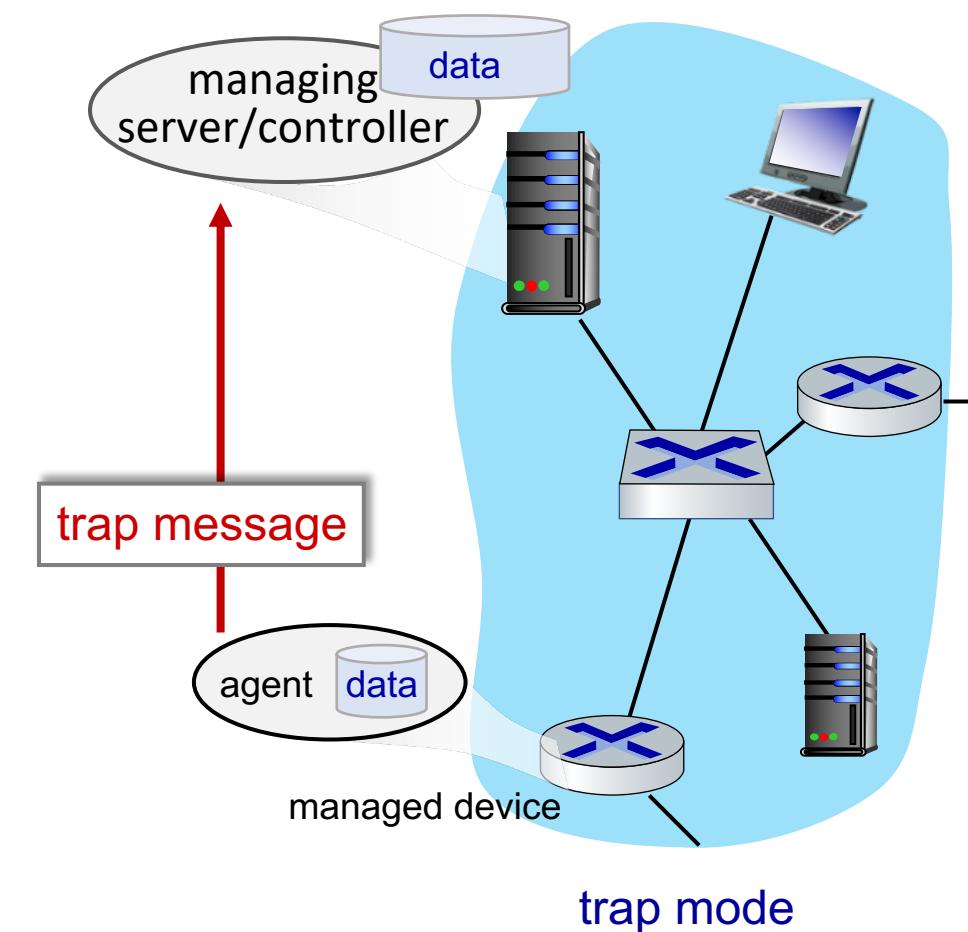
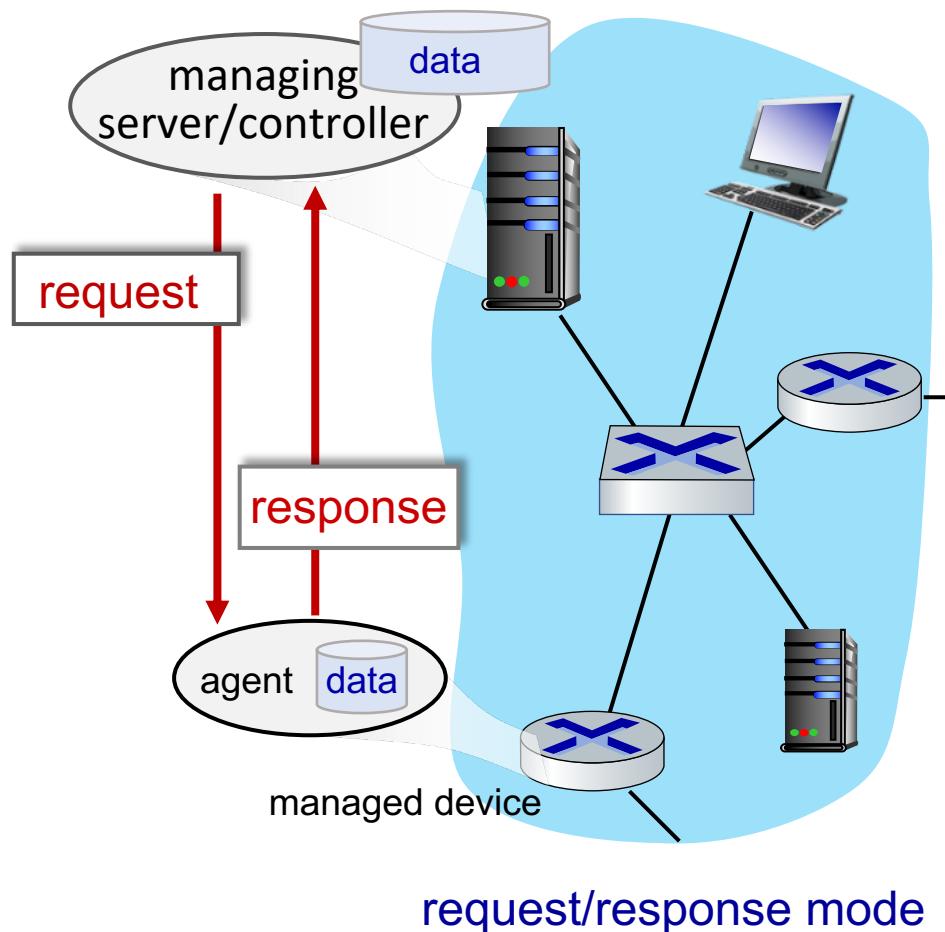
NETCONF/YANG

- più astratto, a livello di rete, olistico
- enfasi sulla gestione della configurazione multi-dispositivo.
- YANG: linguaggio di modellazione dei dati
- NETCONF: comunica azioni/dati compatibili con YANG a/da/tra dispositivi remoti



SNMP - Simple Network Management Protocol

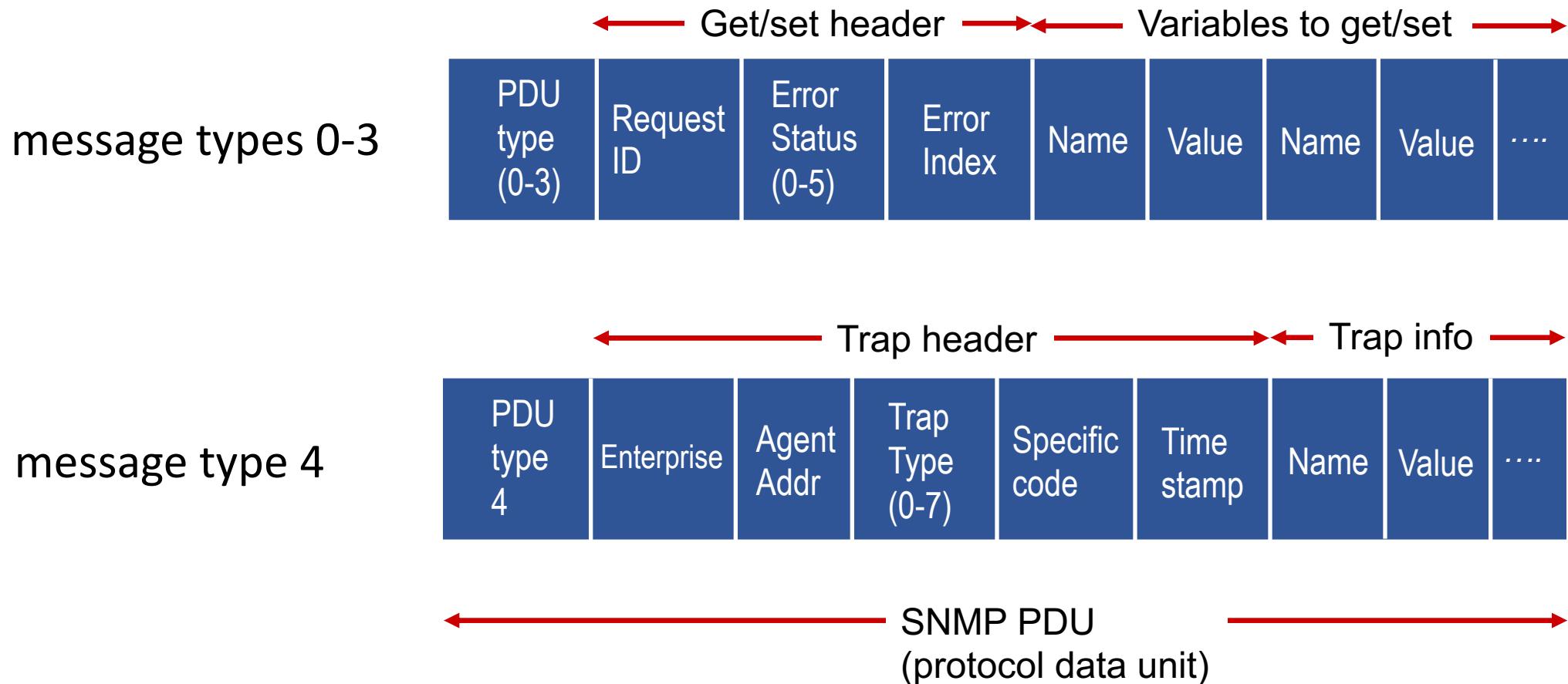
Due modi per trasmettere informazioni MIB e comandi:



Protocollo SNMP: tipi di messaggio

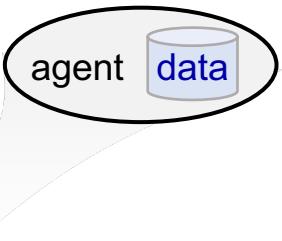
Message type	Function
GetRequest GetNextRequest GetBulkRequest	manager-to-agent: “get me data” (data instance, next data in list, block of data).
SetRequest	manager-to-agent: set MIB value
Response	Agent-to-manager: value, response to Request
Trap	Agent-to-manager: inform manager of exceptional event

Protocollo SNMP: formati dei messaggi



SNMP: Management Information Base (MIB)

- Gli **agenti** memorizzano dati sullo stato e configuraz. del device
- raccolti nel **modulo MIB del dispositivo**
 - 400 moduli MIB definiti in RFC; molti altri MIB specifici del fornitore
- **Structure of Management Information (SMI):** data definition language
- Esempio di variabili MIB per il protocollo UDP:



Object ID	Name	Type	Comments
1.3.6.1.2.1.7.1	UDPIInDatagrams	32-bit counter	total # datagrams delivered
1.3.6.1.2.1.7.2	UDPNoPorts	32-bit counter	# undeliverable datagrams (no application at port)
1.3.6.1.2.1.7.3	UDInErrors	32-bit counter	# undeliverable datagrams (all other reasons)
1.3.6.1.2.1.7.4	UDPOutDatagrams	32-bit counter	total # datagrams sent
1.3.6.1.2.1.7.5	udpTable	SEQUENCE	one entry for each port currently in use

SNMP da v1 a v3

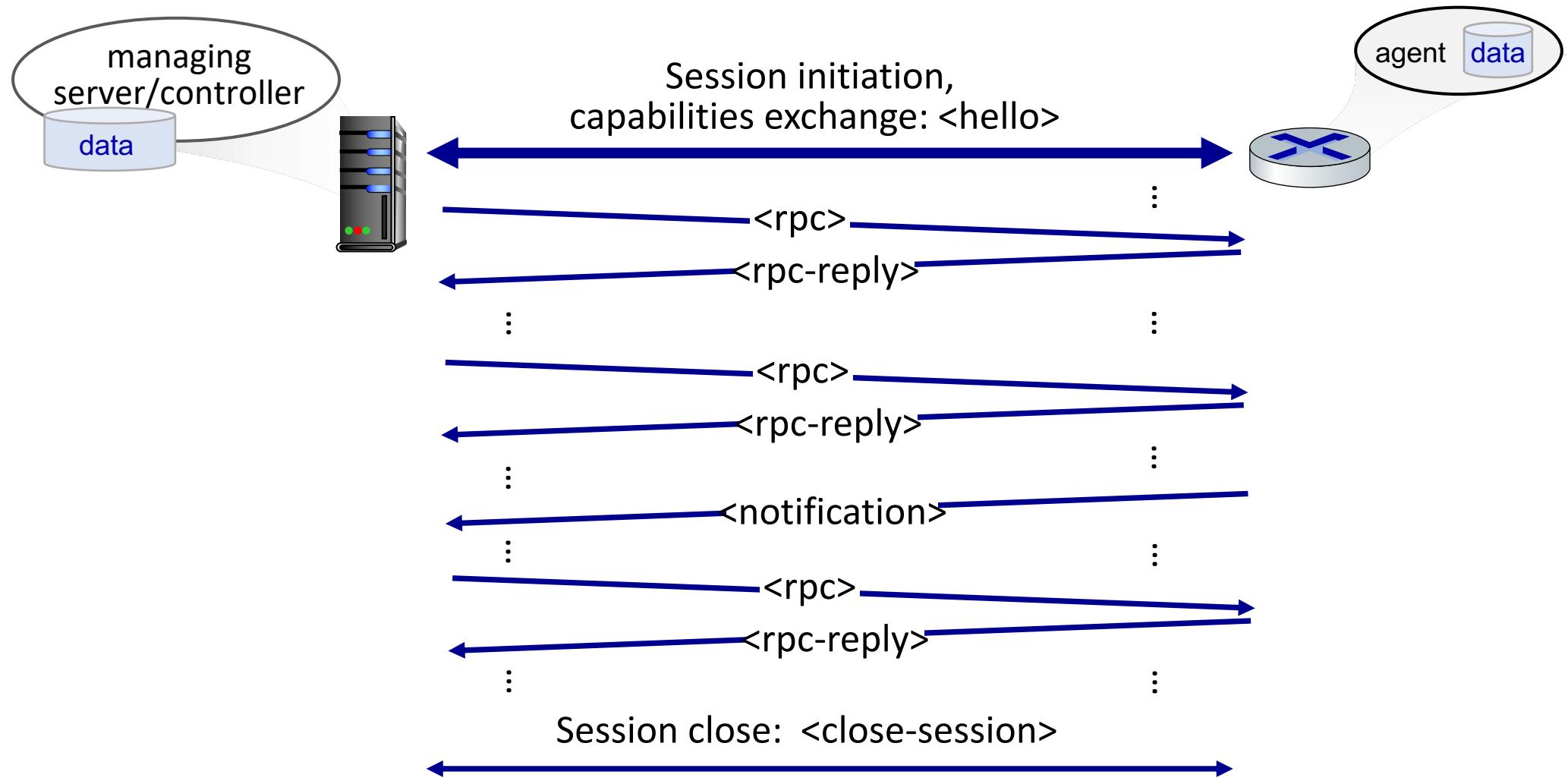
- Tradizionalmente usato su UDP (porta 161) per evitare problemi di congestione di rete, ma può essere implementato anche su TCP
- SNMP tradizionale è vulnerabile ad attacchi di IP spoofing
- SNMPv3 include autenticazione e crittografia

NETCONF overview



- **obiettivo:** gestire/**configurare attivamente** i dispositivi a livello di rete
- opera tra il server di gestione e i dispositivi di rete gestiti
 - azioni: recuperare, impostare, modificare, attivare configurazioni
 - **commit atomico** su più dispositivi
 - interrogare dati operativi e statistiche
 - subscription alle notifiche dai dispositivi
- paradigma di chiamata di procedura remota (RPC, remote procedure call)
 - Messaggi del protocollo NETCONF codificati in XML
 - protocollo di trasporto sicuro e affidabile (ad es. TLS)

NETCONF inizializzazione, scambio, chiusura



NETCONF Operations (non tutte)

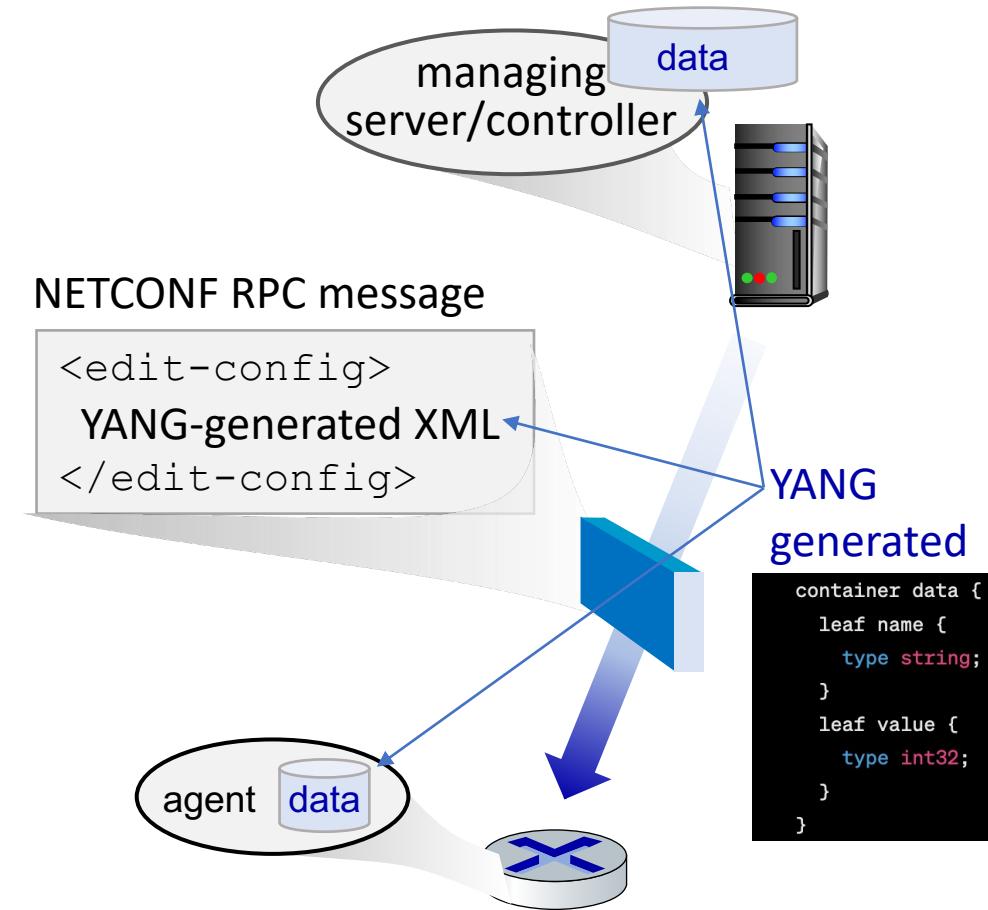
NETCONF	Operation Description
<get-config>	Retrieve all or part of a given configuration. A device may have multiple configurations.
<get>	Retrieve all or part of both configuration state and operational state data.
<edit-config>	Change specified (possibly running) configuration at managed device. Managed device <rpc-reply> contains <ok> or <rpcerror> with rollback.
<lock>, <unlock>	Lock (unlock) configuration datastore at managed device (to lock out NETCONF, SNMP, or CLIs commands from other sources).
<create-subscription>, <notification>	Enable event notification subscription from managed device

Esempio di messaggio NETCONF RPC

```
01 <?xml version="1.0" encoding="UTF-8"?>
02 <rpc message-id="101" note message id
03   xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
04     <edit-config> change a configuration
05       <target>
06         <running/> change the running configuration
07       </target>
08     <config>
09       <top xmlns="http://example.com/schema/
1.2/config">
10         <interface>
11           <name>Ethernet0/0</name> change MTU of Ethernet 0/0 interface to 1500
12           <mtu>1500</mtu>
13         </interface>
14       </top>
15     </config>
16   </edit-config>
17 </rpc>
```

YANG

- linguaggio di modellazione dei dati utilizzato per specificare la struttura, la sintassi e la semantica dei dati di gestione della rete NETCONF
 - definisce tipi di dati, come SMI
- Documento XML che descrive il dispositivo e le sue funzionalità possono essere generate a partire da una descrizione YANG
- può esprimere vincoli tra i dati che devono essere soddisfatti da una configurazione NETCONF valida
 - assicurarsi che le configurazioni NETCONF soddisfino i vincoli di correttezza e coerenza



Livello di rete: riepilogo

- approcci al piano di controllo della rete
 - controllo per router (tradizionale)
 - controllo logicamente centralizzato (network definito dal software)
- algoritmi di routing tradizionali
 - implementazione in Internet: OSPF, BGP
 - routing gerarchico: ogni AS è indipendente
- Controller SDN
 - implementazione pratica: ODL, ONOS
- ICMP, SNMP, NAT, DHCP, IGMP
- gestione della rete (SNMP, NETCONF, YANG)

prossima lezione: livello di collegamento, spostare pacchetti su link della stessa rete