

Rete: livello rete (IP) Parte 1 - Indice

1. Obiettivi
2. Scelte progettuali
3. Algoritmi di instradamento
4. Nodi mobili
5. Controllo di congestione
6. **Internet**

Livello rete (IP)

Architettura degli Elaboratori 2 - T. Vardanega

Pagina 353

Rete: livello rete (IP) Obiettivi - 1

- Ricevere pacchetti dal livello Trasporto del nodo **M** e farli pervenire allo stesso livello del nodo **D**
- Conoscere la topologia dei *router* presenti nella rete, determinando per quali vie raggiungerli
- Superare ogni eventuale disomogeneità tra gli ambienti di rete di **M** e **D**

Livello rete (IP)

Architettura degli Elaboratori 2 - T. Vardanega

Pagina 354

Rete: livello rete (IP) Obiettivi - 2

- Offrire servizi **indipendenti** dalla tecnologia usata per realizzare le *subnet*, nascondendone la complessità al livello Trasporto
- Garantire l'**uniformità** del sistema di indirizzamento superando le diverse tipologie delle reti fisiche collegate (*internetworking*)

Livello rete (IP)

Architettura degli Elaboratori 2 - T. Vardanega

Pagina 355

Rete: livello rete (IP) Scelte progettuali - 1

- Servizi privi di garanzie di connessione (**connectionless**)
 - La scelta progettuale di **Internet**
 - Essendo i *router* **intrinsecamente inaffidabili** (introducono ritardi, causano perdite) conviene delegare interamente al livello Trasporto la responsabilità di effettuare i controlli necessari su eventuali errori trasmissivi
 - Nessun percorso fisso e nessuna informazione di stato sulla conversazione in corso tra **M** e **D**
 - Modello analogo al **servizio postale**
 - Due pacchetti inviati dallo stesso nodo **M** allo stesso nodo **D** possono seguire due percorsi **totalmente diversi**

Livello rete (IP)

Architettura degli Elaboratori 2 - T. Vardanega

Pagina 356

Rete: livello rete (IP) Scelte progettuali - 2

- I pacchetti (**datagram**) circolanti in questa modalità devono designare gli indirizzi dei nodi **D** ed **M**
 - **D** ed **M** devono possedere un indirizzo **unico**
- I *router* usano questa informazione per determinare il balzo (*hop*) successivo da far compiere al pacchetto
 - Quale linea in uscita, verso quale prossimo *router*

Livello rete (IP)

Architettura degli Elaboratori 2 - T. Vardanega

Pagina 357

Rete: livello rete (IP) Scelte progettuali - 3

- Servizi con garanzia di connessione (**connection-oriented**)
 - Per ogni richiesta di comunicazione si configura un **circuito virtuale** (*virtual circuit*) dedicato
 - Il circuito virtuale è **bidirezionale** e tutti i pacchetti vi viaggiano in modo ordinato e controllato
 - Informazione di stato associa i pacchetti di una connessione **M-D** al relativo percorso (circuito)
 - Modello analogo al **servizio telefonico**: i pacchetti scambiati in una stessa conversazione percorrono sempre lo **stesso percorso**

Livello rete (IP)

Architettura degli Elaboratori 2 - T. Vardanega

Pagina 358

Rete: livello rete (IP) Scelte progettuali - 4

- Un circuito virtuale è un tragitto tra nodo **M** e nodo **D** che, una volta configurato (*set up*) è usato per tutto il traffico specifico di quella particolare comunicazione
- Ogni *router* deve poter mantenere **più** circuiti virtuali aperti **simultaneamente** ed avviare ciascun pacchetto sul proprio circuito
- Ogni pacchetto deve perciò identificare il proprio circuito virtuale di appartenenza

Livello rete (IP) Architettura degli Elaboratori 2 - T. Vardanega Pagina 359

Rete: livello rete (IP) Algoritmi di instradamento - 1

- Usati per decidere verso quale linea in uscita e quale successivo *router* avviare pacchetti
 - **Circuito virtuale**: la scelta viene fatta una volta per tutte nel configurare il circuito
 - **Datagram**: la scelta viene fatta per ogni pacchetto in arrivo e può pertanto variare nel tempo
- **2 classi di algoritmi**
 - **Statici**: ogni *router* è dotato di tabelle predefinite che associano percorsi a destinazioni
 - **Dinamici**: tengono conto di eventuali modifiche (anche temporanee) della topologia delle *subnet* e del volume di traffico sui collegamenti disponibili

Livello rete (IP) Architettura degli Elaboratori 2 - T. Vardanega Pagina 360

Rete: livello rete (IP) Algoritmi di instradamento - 2

Modalità **datagram**: i pacchetti viaggiano su percorsi **indipendenti**

A's table		C's table		E's table	
initially	later	initially	later	initially	later
A: -	A: A	A: A	A: C	A: C	A: C
B: B	B: B	B: A	B: D	B: D	B: D
C: C	C: C	C: -	C: C	C: C	C: C
D: B	D: B	D: D	D: D	D: D	D: D
E: C	E: B	E: E	E: -	E: -	E: -
F: C	F: B	F: E	F: F	F: F	F: F

Ciascuna tabella di routing indica il passo successivo (1 balzo) verso ciascuna delle destinazioni note

Livello rete (IP) Architettura degli Elaboratori 2 - T. Vardanega Pagina 361

Rete: livello rete (IP) Algoritmi di instradamento - 3

Circuito virtuale: i pacchetti di una stessa conversazione viaggiano **tutti** sullo **stesso** percorso

A's table		C's table		E's table	
In	Out	In	Out	In	Out
H1: 1	C: 1	A: 1	E: 1	C: 1	F: 1
H3: 1	C: 2	A: 2	E: 2	C: 2	F: 2

Ciascuna tabella di routing fissa il percorso di ogni singolo circuito

Livello rete (IP) Architettura degli Elaboratori 2 - T. Vardanega Pagina 362

Rete: livello rete (IP) Algoritmi di instradamento - 4

- **Principio di ottimalità**: se *J* si trova sul **cammino ottimale** da *I* a *K*, allora quello è anche il cammino ottimale da *J* a *K*
 - **Metrica di ottimalità** è la lunghezza del percorso, espressa, p.es., in numero di balzi
- L'insieme dei cammini ottimali da ogni possibile sorgente ad una stessa destinazione è un albero con radice nella destinazione (**sink tree**)
 - Questa proprietà garantisce l'arrivo a destinazione, in tempo finito, di ogni pacchetto inviato
 - Ma solo se i collegamenti non subiscono modifiche (se l'albero resta albero)

Livello rete (IP) Architettura degli Elaboratori 2 - T. Vardanega Pagina 363

Rete: livello rete (IP) Algoritmi di instradamento - 5

Le connessioni fornite da una subnet danno origine a ≥ 1 sink tree per ogni possibile destinazione (p.es. B)

Livello rete (IP) Architettura degli Elaboratori 2 - T. Vardanega Pagina 364

Rete: livello rete (IP)

Algoritmi di instradamento - 6

- Varie metriche possono essere usate per determinare la lunghezza di un cammino
 - **Numero di balzi**
 - # archi attraversati sul grafo
 - **Distanza geografica**
 - Σ distanze espresse come attributo di ciascun arco
 - **Tempo medio di attesa** presso un certo *router*
 - Attributo di ciascun nodo
 - **Capacità** (portanza) del collegamento
 - Attributo di arco
 - ...

Livello rete (IP)
Architettura degli Elaboratori 2 - T. Vardanega
Pagina 365

Rete: livello rete (IP)

Algoritmi di instradamento - 7

Determinazione del percorso più breve tra **A** e **D** con l'algoritmo di Dijkstra

Livello rete (IP)
Architettura degli Elaboratori 2 - T. Vardanega
Pagina 366

Rete: livello rete (IP)

Algoritmi di instradamento - 8

- **Algoritmo di Dijkstra (statico)**
 - **(a)** ogni arco ha un attributo iniziale che ne indica la lunghezza
 - **(b)** ogni nodo raggiungibile da **A** in 1 balzo riceve un'etichetta indicante la distanza da **A** ed il nodo adiacente di provenienza nel cammino
 - Ogni nodo raggiunto che risulta più vicino nel cammino da **A** viene marcato come "permanente", gli altri come "tentativi"
 - I nodi non ancora toccati sono considerati a distanza infinita
 - **(c-f)** si visita ogni nodo adiacente ad ogni nodo marcato permanente e si aggiorna la distanza da **A**, sommandovi la lunghezza percorsa sul relativo arco

Livello rete (IP)
Architettura degli Elaboratori 2 - T. Vardanega
Pagina 367

Rete: livello rete (IP)

Algoritmi di instradamento - 9

- **Alluvione, flooding (statico)**
 - Ogni pacchetto in ingresso è emesso, **in copia**, su ciascuna linea in uscita
 - Per limitare le emissioni inutili, ci si può limitare ai soli collegamenti nella "direzione giusta"
 - Ogni pacchetto può percorrere al più un dato **numero di balzi**: un apposito attributo contatore viene decrementato ad ogni balzo
 - Quando il contatore giunge a 0 quella copia del pacchetto viene distrutta

Livello rete (IP)
Architettura degli Elaboratori 2 - T. Vardanega
Pagina 368

Rete: livello rete (IP)

Algoritmi di instradamento - 10

- Stato del collegamento, **link state** (dinamico)
 - Ogni *router* si aggiorna sull'indirizzo e la distanza dei propri vicini e, tramite invio di **pacchetti di prova**, determina il cammino più breve verso di essi
 - I nomi dei *router* devono essere **unici**
 - L'aggiornamento deve essere ripetuto **periodicamente**
 - I *router* si scambiano pacchetti contenenti la propria informazione topologica
 - L'aggiornamento asincrono delle tabelle dei *router* può rendere instabile la determinazione dei cammini
 - Ottenuta tutta l'informazione necessaria, ciascun *router* usa l'algoritmo di Dijkstra per i cammini minimi

Livello rete (IP)
Architettura degli Elaboratori 2 - T. Vardanega
Pagina 369

Rete: livello rete (IP)

Nodi mobili - 1

- 2 tipi di utenti (= nodi) **mobili** in una rete
 - Utenti che si connettono in tempi diversi da diverse **postazioni fisse**
 - Utenti di **postazioni mobili (roaming)**
- Ogni utente ha una base di riferimento fissa **HB**
- La rete globale è suddivisa in aree, ciascuna delle quali possiede:
 - ≥ 1 agenti **FA** degli utenti mobili (*foreign*) connessi
 - Gli **FA** registrano la presenza di nuovi utenti mobili
 - 1 agente **HA** degli utenti con base (*home*) nell'area

Livello rete (IP)
Architettura degli Elaboratori 2 - T. Vardanega
Pagina 370

Rete: livello rete (IP) Nodi mobili - 2

- Ogni utente mobile **MU** si registra nell'area ove si trova al momento
 - **FA** → *broadcast* periodico a tutti i possibili **MU** nell'area
 - **MU** → **FA** : **HB(MU)** = posizione attuale ed accreditato
 - **FA** → **HA(MU)** : indirizzo **FA**, accreditato
 - **HA(MU)** → **FA** : autorizzazione a registrare **MU**
 - **FA** registra **MU** nella sua anagrafe locale
- Ogni comunicazione destinata ad **MU** va alla sua **HB**, il cui **HA** la invia al **FA** presso il quale esso è registrato
 - I pacchetti originari vengono inviati come parte dati dei pacchetti inviati da **HA** ad **FA** (*tunneling*)

Livello rete (IP)
Architettura degli Elaboratori 2 - T. Vardanega
Pagina 371

Rete: livello rete (IP) Controllo di congestione - 1

- Si ha congestione quando troppi pacchetti si trovano entro una stessa *subnet*
 - I pacchetti in ingresso ad un *router* le cui code di lavoro sono piene vanno persi
 - Code più ampie peggiorano il problema: i pacchetti stanno più a lungo in coda, **M** li considera persi e li replica, aumentando così il traffico!
- **Controllo di congestione** garantisce che la *subnet* sia sempre capace di trattare tutto il suo traffico in ingresso
- **Controllo di flusso** evita che **M** possa mai congestionare **D**

Livello rete (IP)
Architettura degli Elaboratori 2 - T. Vardanega
Pagina 372

Rete: livello rete (IP) Controllo di congestione - 2

- Le congestioni possono essere prevenute per via strutturale o per controllo d'esecuzione
- **Dimensionamento strutturale di traffico**
 - **Leaky bucket** : fissa il flusso massimo di uscita, vi dimensiona la coda di ingresso e scarta tutto il traffico in eccesso
 - Il flusso d'uscita non può avere picchi
 - Il "secchio" è la coda e perde pacchetti se si riempie
 - **Token bucket** : il controllore produce gettoni con periodo T; ogni pacchetto in arrivo viene accettato se vi è un gettone per lui, altrimenti resta in coda
 - I gettoni possono essere accumulati fino alla capacità massima del "secchio" → il flusso d'uscita può avere picchi
 - Il "secchio" accoda (e perde) gettoni e non pacchetti

Livello rete (IP)
Architettura degli Elaboratori 2 - T. Vardanega
Pagina 373

Rete: livello rete (IP) Controllo di congestione - 3

- **Soluzioni dinamiche a controllo di traffico**
 - **Soffocamento di linea** : se il flusso in ingresso da ciascuna linea supera una data soglia, si invia una notifica (**choke packet**) ad **M**, marcando il suo ultimo pacchetto accettato come "avvisato" e facendolo proseguire; ricevuta la notifica, **M** riduce il flusso in uscita
 - Metodo rischioso: si basa sulla buona fede di **M**
 - **Load shedding**: il *router* in sovraccarico scarta pacchetti secondo criteri arbitrari e/o secondo l'importanza loro attribuita da **M**

Livello rete (IP)
Architettura degli Elaboratori 2 - T. Vardanega
Pagina 374

Rete: livello rete (IP) Internet - 1

- Il termine **internet**, originariamente coniato per denominare una **singola** interconnessione tra reti locali, denota ora un insieme di reti connesse da un **protocollo comune** in modo da formare una singola rete virtuale
- Il termine **Internet** denota la confederazione di reti **eterogenee** connesse da **IP** per formare una singola rete virtuale **globale**

Livello rete (IP)
Architettura degli Elaboratori 2 - T. Vardanega
Pagina 375

Rete: livello rete (IP) Internet - 2

- **IP** si propone 5 obiettivi
 - Definire l'unità di comunicazione su rete
 - **Datagram**
 - Definire lo schema di indirizzamento di **tutti** i nodi della rete (non solo i *router*!)
 - Offrire i servizi attesi dal livello Trasporto (*host-to-host*) utilizzando i servizi forniti dal livello inferiore
 - Instradare i **datagram** verso le rispettive **D**
 - Frammentare i **datagram** all'invio e ricostituirli alla ricezione

Livello rete (IP)
Architettura degli Elaboratori 2 - T. Vardanega
Pagina 376

Rete: livello rete (IP) Internet - 3

Il formato di un **datagram**

Indirizzi IP

Livello rete (IP) Architettura degli Elaboratori 2 - T. Vardanega Pagina 377

Rete: livello rete (IP) Internet - 4

- L'intestazione (*header*) occupa sempre le prime 5 parole da 32 *bit* del **datagram**
 - Anche la 6ª parola, se così specificato nel campo **IHL** (*Internet Header Length*)
- L'intestazione fornisce le informazioni che permettono la consegna a destinazione
- Il **datagram** viene trattato e trasmesso in ordine *Big-Endian*
 - La parte più significativa di ciascun campo si trova all'indirizzo più alto (e viene trasmessa per prima)

Livello rete (IP) Architettura degli Elaboratori 2 - T. Vardanega Pagina 378

Rete: livello rete (IP) Internet - 5

- **Versione:** versione del protocollo cui appartiene il **datagram**
- **Tipo di servizio:** caratteristiche richieste alla trasmissione (veloce, sicura); spesso ignorato
- **Lunghezza totale:** specifica la dimensione del **datagram**, e dunque della sua area dati ($\leq 64k-1$ B)
- **Identificativo:** flusso di comunicazione cui appartengono i dati del **datagram**
- **Flags e Fragmentation Offset.** indicano se il **datagram** possa essere frammentato e, nel caso, la posizione relativa del frammento nella sequenza
 - Reti fisiche diverse impongono limiti diversi sulla massima dimensione delle unità trasportate

Livello rete (IP) Architettura degli Elaboratori 2 - T. Vardanega Pagina 379

Rete: livello rete (IP) Internet - 6

- **Tempo di vita:** numero di prossimi balzi (*hop*) ancora consentiti al **datagram**
 - Decrementato ad ogni balzo: quando va a 0, causa la distruzione del **datagram** e l'invio di una notifica ad **M**
- **Protocollo:** tipo di trattamento richiesto dal **datagram** a livello trasporto (TCP o UDP)
- **Controllo (*checksum*) di intestazione:** per verifica di integrità dell'intestazione
 - Deve essere calcolato ogni volta che cambi il contenuto dell'intestazione, e dunque ad ogni balzo
- **Opzioni:** riservato per sperimentazioni o future estensioni del protocollo

Livello rete (IP) Architettura degli Elaboratori 2 - T. Vardanega Pagina 380