RICERCA DEI PERCORSI IP

Prof. Ing. Maurizio Casoni

Dipartimento di Ingegneria “Enzo Ferrari”
Università degli Studi di Modena e Reggio Emilia

INTRODUZIONE

- Crescita esponenziale dei volumi di traffico => crescente richiesta di enormi capacità nella core network
- Offrire qualità di servizio buone significa affrontare tre aspetti nella progettazione di reti IP:
  1. Linee a capacità sempre maggiore
  2. Necessità di architetture negli apparati switch/router sempre più veloci per sostenere le sempre maggiori velocità delle linee di trasmissione (elevati data throughputs)
  3. Elevate velocità di inoltro dei datagrammi IP

1. Risolto: evoluzione dei sistemi di trasmissione in fibra ottica
2. Risolto: adeguate architetture che supportano velocità del Gbit/s
   (v. Sistemi di Commutazione)

- **Obiettivo**: Per il successo delle reti IP di prossima generazione è necessario sviluppare ed implementare nei router di fascia alta metodi per l'elaborazione e l'inoltro dei pacchetti IP ad alta velocità

INOLTRO DEI PACCHETTI

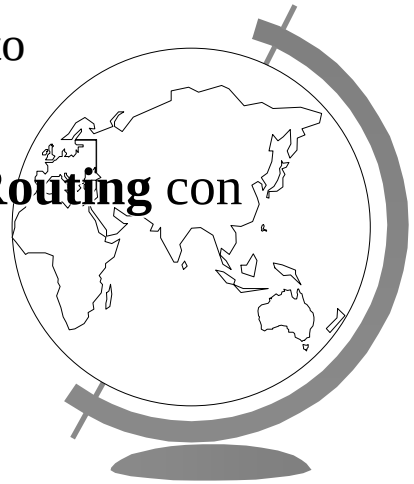
Molti compiti devono essere eseguiti

- Incapsulamento/decapsulamento dell'intestazione del pacchetto IP
- Aggiornamento del campo *Time To Live*
- Eventuale classificazione del pacchetto in apposite code per specifiche classi di servizio
- …..

- Il compito principale e più oneroso in termini di tempo di processamento del pacchetto è la ricerca nella tabella di instradamento dell'informazione di *next hop* relativa al prefisso che soddisfa al meglio l'indirizzo IP del pacchetto : **IP route lookup**

INTERNET: CRESCITA ESPONENZIALE

- Internet non fu concepita per avere estensione mondiale
- Negli anni ’80 il numero di hosts era limitato a qualche centinaia negli Stati Uniti: 32 bit di indirizzo sembravano assai sufficienti
- Inizi anni ’90 con lo sviluppo del WWW il numero di hosts è cresciuto esponenzialmente e la rete si è estesa a livello mondiale: possibile crisi del funzionamento della rete stessa
- Tre grossi pericoli apparvero nel 1991:
  1. Esaurimento degli indirizzi di classe B
  2. Esplosione della dimensione delle tabelle di instradamento
  3. Esaurimento degli indirizzi IP disponibili
- IETF ha definito nel 1992 il **Classless Inter-Domain Routing** con lo scopo di evitare alcuni dei sopracitati pericoli



ESAURIMENTO INDIRIZZI DI CLASSE B

- Indirizzi IP sono raggruppati in 3 classi
  - Classe A (Net-id di 8 bit): 24 bit di host-id => 16.777.216 hosts
  - Classe B (Net-id di 16 bit): 16 bit di host-id => 65.536 hosts
  - Classe C (Net-id di 24 bit): 8 bit di host-id => 256 hosts
- Nel 1994 ci si rese conto che le reti A erano troppo poche e che non potevano essere assegnate facilmente e che le reti C erano troppo piccole e vi sono molte aziende che necessitano di più di 256 indirizzi IP
- Classe B andava abbastanza bene ma in totale ce ne sono 16.384 (solo 14 bit del net-id sono utilizzabili perchè i 2 bit di maggior peso sono fissati a 1 e 0): oggi indirizzi B sono esauriti => occorre trovare una soluzione

ESPLOSIONE DELLE TABELLE DI INSTRADAMENTO

- All’ aumentare del numero delle reti interconnesse aumenta la quantità di memoria necessaria nei router per la loro memorizzazione e questo dipende molto dal tipo di protocollo di routing usato e dall’architettura del router
- Inizi anni ’90 i router di transito superarono la soglia di 4000 reti interconnesse: inviare periodicamente informazioni di aggiornamento comportava creare pacchetti molto lunghi frammentati in molteplici datagrammi IP che mandarono fuori servizio i router stessi
- Siccome il processamento degli indirizzi IP deve essere veloce, occorre usare memorie veloci nelle schede di rete che però non sono grandi e presto si rivelarono insufficienti appena si superarono le 10000 reti interconnesse (1993); d’altra parte processare in grandi memorie centrali rallentava il forwarding
- Si decise quindi di mettere in memorie cache nelle schede di rete gli indirizzi “più usati” e processare tutti gli altri mediante il processore centrale

INDIRIZZI CLASSLESS

- ***OSSERVAZIONE***: molte aziende hanno più di 256 hosts mentre poche ne hanno più di alcune migliaia
- ***POSSIBILE IDEA***: anzichè richiedere una classe B perchè non richiedere un numero di reti di classe C tale da soddisfare le proprie necessità ?
- Reti di classe C sono più di 2 milioni e mediamente con 8 di esse (2048 indirizzi) si soddisfa la maggior parte delle organizzazioni
- In tal modo si può alleviare il problema dell'esaurimento degli indirizzi di classe B ma non si risolve ancora il problema dell'esplosione delle tabelle di instradamento
- **STRATEGIA DI ASSEGNAZIONE CIDR degli indirizzi**: i numeri delle reti di classe C assegnati ad un'organizzazione non devono essere casuali ma contigui in modo da condividere un prefisso di bits più significativi
  - Unica entry per l'intero insieme di reti di classe C (**supernet**)
  - Possibile allocazione di indirizzi in modo geografico con prefisso che identifica un certa area geografica (aggregazione della tabella di instradamento)

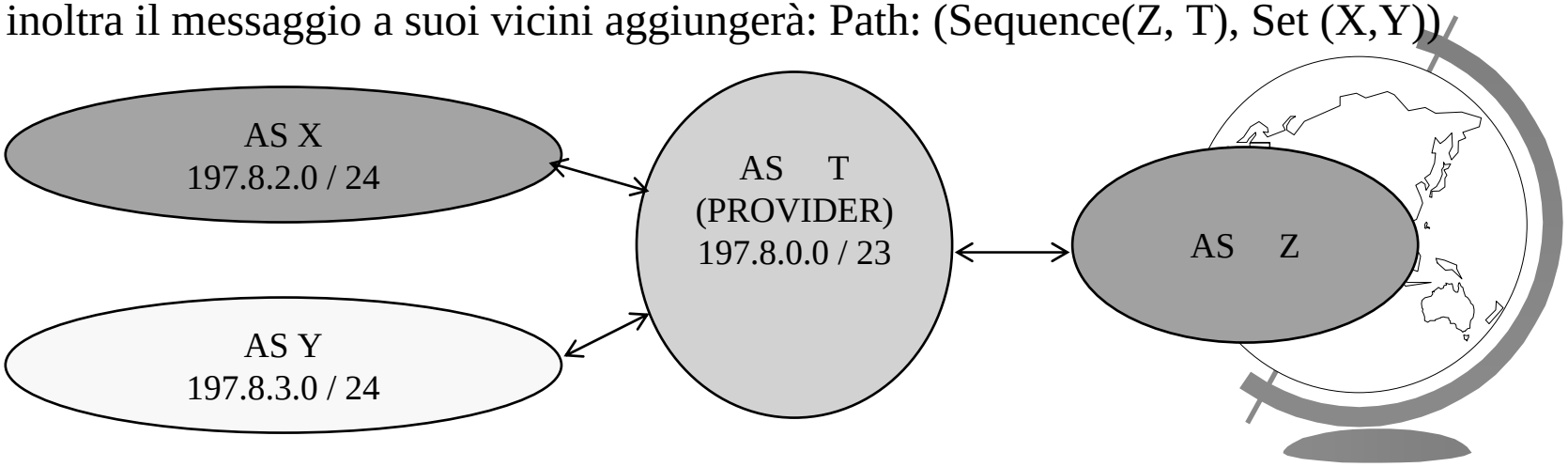
AGGREGAZIONE DELLA TABELLA DI INSTRADAMENTO

- Operazione possibile solo se gli indirizzi vengono assegnati in modo coordinato
- Assegnamento coordinato con piano degli indirizzi delle reti di classe C per continente
  - Multiregionale: 192.0.0.0 – 193.255.255.255
  - Europa: 194.0.0.0 – 195.255.255.255
  - Nord America: 198.0.0.0 – 199.255.255.255
  - America Centro/Sud: 200.0.0.0 – 201.255.255.255
- Scopo: avere unico prefisso per continente, p.es. Europa caratterizzata da unico prefisso a 7 bit: 1100001

- Dibattito aperto: assegnazione degli indirizzi in base all'area geografica o in base al provider?
- Ad ogni modo, aggregazione fornisce uno strumento non la soluzione: occorre che i protocolli di routing inter-domain la mettano in pratica mediante la gestione dell'instradamento delle supernet e l'aggiornamento automatico

CIDR e BGP

- BGP v.4 (operativo dal 1994) supporta il supernetting
- Per ridurre le tabelle di instradamento dobbiamo poter aggregare i prefissi
- AS T è un provider: gli indirizzi scelti da 2 set contigui di classe C, 197.8.0.XX e 197.8.1.XX, rappresentati da un prefisso di 23 bit
- T ha 2 clienti, X e Y, con reti di classe C: che percorsi verranno annunciati da T a Z?
- Ideale sarebbe annunciare Path1: reaches 197.8.0.0/22 che unisce i prefissi di T, X,Y in unico prefisso a 22 bit che li comprende tutti
- BGP4 struttura l'attributo "AS path" in 2 componenti, lista ordinata e non ordinata:
  - Path: (Sequence(T), Set (X,Y))
- Se Z inoltra il messaggio a suoi vicini aggiungerà: Path: (Sequence(Z, T), Set (X,Y))



CIDR

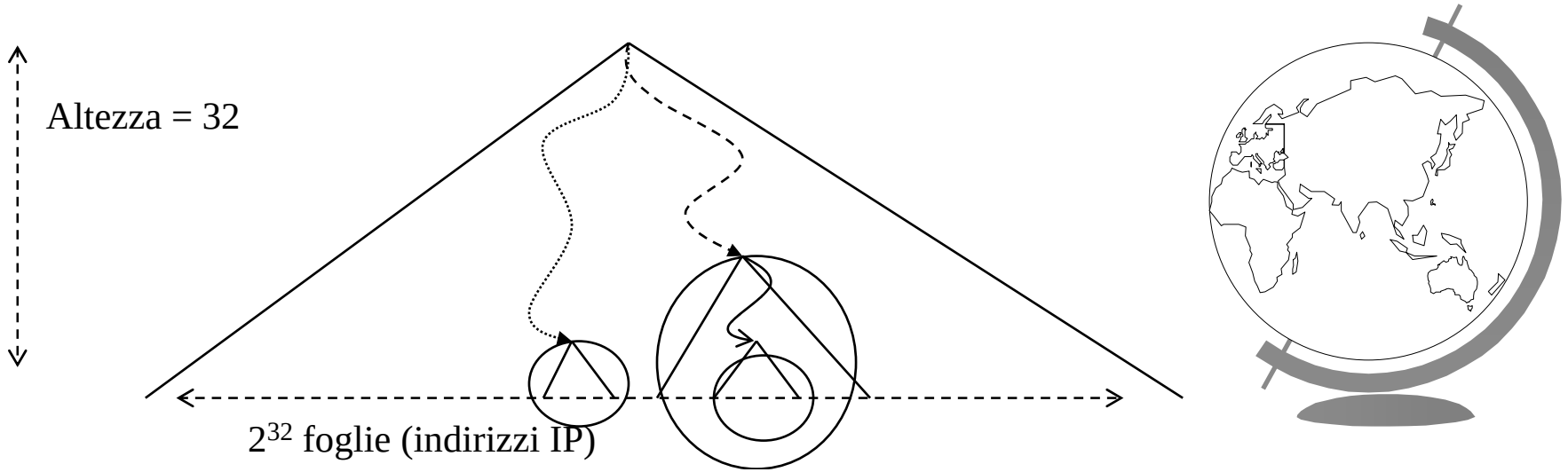
- Si osservi che CIDR ha consentito di dare una temporanea risposta a 2 dei 3 grossi problemi di Internet con IPv4
  1. Esaurimento degli indirizzi di classe B
  2. Esplosione della dimensione delle tabelle di instradamento
- ma non può fare nulla per il terzo
  3. Esaurimento degli indirizzi IP disponibili
- IETF ha quindi studiato e proposto una versione di IP che risolvesse anche il terzo problema e consentisse inoltre il supporto della mobilità, di applicazioni real-time e di multicast: ***IP v.6***

IP ROUTE LOOKUP

- Nella progettazione della struttura dati usata nella *forwarding table*, l'obiettivo primario è ***rendere minimo il lookup time***
- Due parametri devono contemporaneamente essere minimizzati
  - Il numero degli accessi alla memoria durante il lookup
  - La dimensione della struttura dati
- Ridurre il numero degli accessi è importante perchè essi sono di solito relativamente lenti e rappresentano il collo di bottiglia delle procedure di lookup
- Se la struttura dati è piccola si può pensare di metterla nella cache di un microprocessore rendendo gli accessi ordini di grandezza più veloci rispetto a DRAM
- Anche se l'intera *forwarding table* non ci sta nella cache si può pensare di metterci una sua parte, quella che si ritiene sarà maggiormente consultata
- Come obiettivi secondari, la struttura dati dovrebbe
  - Necessitare di poche istruzioni durante il *lookup*
  - Mantenere i suoi elementi in organizzazione ottimale per evitare sprechi ed inefficienze nella ricerca

STRUTTURA DATI

- Si consideri un albero binario che copre l'intero spazio degli indirizzi IP
  - La sua altezza sarà 32 ed il numero delle foglie 2^{32}, uno per ogni possibile indirizzo IP
- Il prefisso di entrata alla *routing table* definisce un percorso nell'albero che termina in un certo nodo: tutti gli indirizzi IP (foglie) nel sottoalbero avente questo nodo come radice dovrebbero essere instradati allo stesso modo
- Conseguenza: ogni *entry* alla *routing table* definisce un insieme di indirizzi IP aventi le stesse informazioni di routing (indirizzo IP di next-hop)
- Se d'altra parte diverse *entry* alla tabella coprono lo stesso indirizzo IP, si applica la regola del ***longest matching prefix***: dato un certo indirizzo IP si deve scegliere la entry che scende più in profondità nell'albero

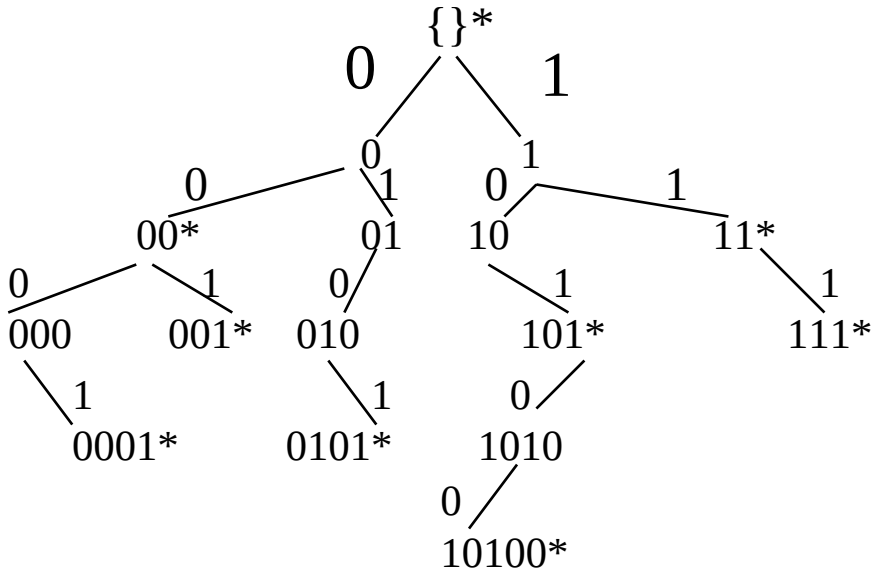


IP ROUTE LOOKUP: TRIE STRUCTURE

- La ***struttura trie*** è un albero multi percorso in cui ogni nodo contiene o zero o più puntatori a nodi figli
- Nella struttura *trie* ad 1 bit ogni nodo contiene 2 puntatori, il puntatore a 0 e a 1
- Un nodo X al livello h rappresenta l'insieme di tutti i prefissi di percorsi che hanno gli stessi primi h bits ed in funzione del (h+1)esimo bit, 0 o 1, un puntatore punta un diverso sottoalbero (se esiste) che rappresenta l'insieme di tutti i prefissi di percorsi che hanno gli stessi primi (h+1) bits
- Ogni ricerca (*lookup*) comincia dal nodo radice del trie ed in funzione del valore di ogni bit successivo si ricava il nodo seguente da visitare
- Durante l'attraversamento del trie si tiene traccia del *next hop* di *matching prefix* più lungo trovato

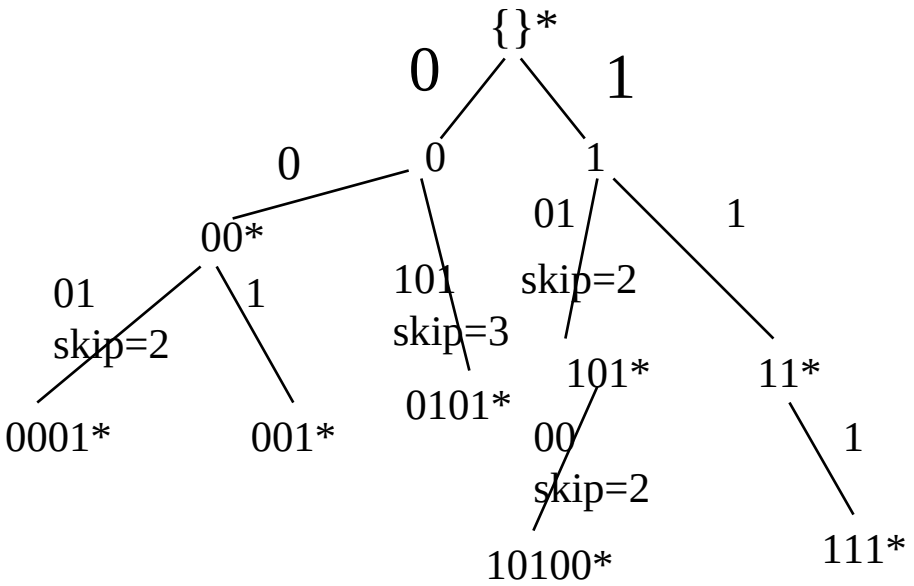
TRIE STRUCTURE: ESEMPIO

- Ogni nodo ha un flag, asterisco, che indica se il prefisso che termina in quel nodo esiste nella tabella di routing
- Si supponga di processare l'indirizzo di destinazione 10101101: la ricerca termina nel nodo 1010 ed il longest prefix matching restituisce 101
- Svantaggi della struttura ad 1 bit: accessi alla memoria nel caso peggiore sono 32



PATRICIA TREE

- Tradizionali implementazioni delle tabelle di instradamento fanno uso di Patricia trees, miglioramento della struttura trie a 1 bit
- ***Osservazione***: ogni nodo interno che non corrisponde a prefisso nella tabella ed ha solo un figlio può essere rimosso per abbreviare il percorso
- Occorre definire meccanismo per tenere traccia dei nodi mancanti: memorizzare un numero, *skip value*, in ogni nodo che indichi quanti bit sono stati saltati lungo il percorso
- Supponiamo di ricercare l'indirizzo 10001101: {} viene restituito
- Con circa 40000 prefix entry nella routing table ci vogliono circa 2 Mbyte: struttura comunque grande con troppo accessi (15-16): ulteriori tecniche di lookup



CENNI DI PRESTAZIONI

- Alcuni miglioramenti consentono da una grande tabella di instradamento con 40000 entries di avere una struttura dati di 150-160 Kbytes che può essere messa nella cache di un processore general-purpose
  - Pentium Pro può eseguire svariati milioni di IP lookup al secondo

- Uleriori miglioramenti con hardware customizzato sono stati proposti per fornire un lookup ad ogni accesso di memoria: con memorie DRAM a 50 nsec vuol dire 20 milioni di pacchetti al secondo