WHOLESALE WINERY Tour

Clos Network su Mikrotik

Alessandro Campanella

WHOLESALE WINERY Tour

About me
Alessandro Campanella

Lavoro nel campo delle telecomunicazioni da oltre 20 anni come produttore, consulente e formatore, sia in Italia che all'estero.
Sono stato tra i pionieri della tecnologia wireless, conducendo esperimenti sulla mia rete e raccogliendo le esperienze dei numerosi operatori e integratori con cui ho collaborato nel corso degli anni.

MikroTik Evangelist
Design, analysis, and consulting for ISPs
MikroTik Training and Certification

a.campanella@1off.it
@alessandrocampanella

1off
www.1off.it

Ringraziamenti

Sulle spalle dei giganti

Questa presentazione nasce dall'ispirazione e dal lavoro di chi ha contribuito in modo significativo alla comprensione e alla diffusione delle architetture di rete basate su Clos Network.

Un ringraziamento particolare a:

Tiziano Tofoni
per il lavoro divulgativo e l'analisi delle architetture Clos pubblicate su Reiss Romoli, illuminante come sempre.

Petr Lapukhov
autore della RFC 7938 – Use of BGP for Routing in Large-Scale Data Centers,
che ha formalizzato molti dei principi operativi delle moderne architetture spine-leaf.

1. Origine matematica della Clos Network

Clos Network su Mikrotik

Charles Clos (1923–2016)

È stato un ingegnere dei **Bell Laboratories** e una figura fondamentale nella storia delle reti di commutazione.

Il suo lavoro ha posto le basi teoriche delle architetture di rete utilizzate oggi nei **data center spine-leaf** e nelle fabric Clos.

Nel 1953 Clos pubblicò il famoso articolo:
“A Study of Non-Blocking Switching Networks”

1. Origine matematica della Clos Network

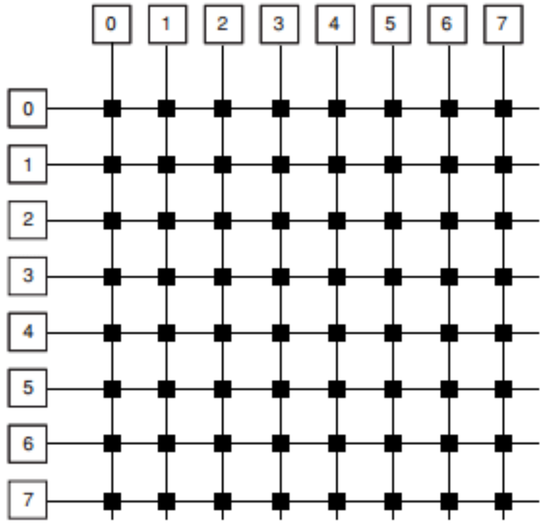
Il problema originario: costruire grandi matrici di commutazione

La Clos Network nasce per risolvere un problema molto concreto: **come realizzare una matrice di commutazione molto grande senza dover costruire un crossbar completo gigantesco**

Nel modello più semplice, se vuoi connettere liberamente:

- N ingressi
- a N uscite

con una piena libertà di connessione, la soluzione più diretta è un **crossbar NxN**.

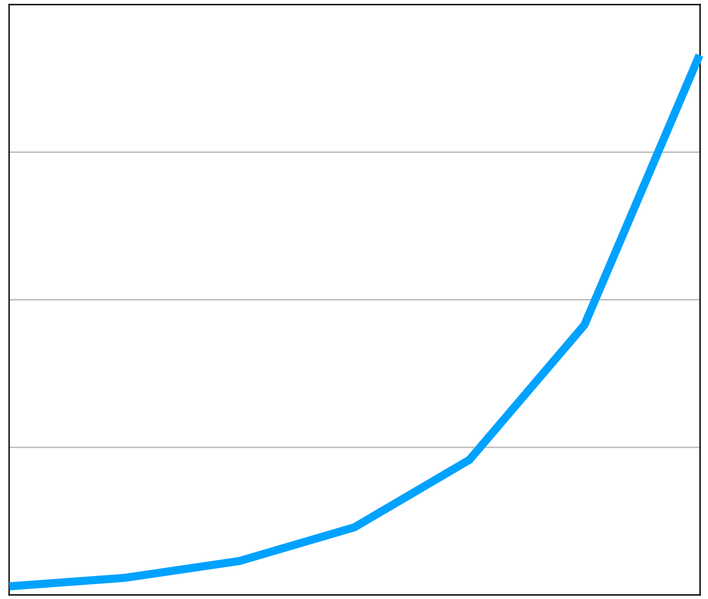


1. Origine matematica della Clos Network

Il problema originario: costruire grandi matrici di commutazione

Crossbar completo: ogni ingresso può essere connesso a ogni uscita.
Numero di punti di commutazione:

N^2



Questa soluzione è:

- semplice da capire
- perfettamente non bloccante
- ma pessima da scalare

Per valori elevati di N, il numero di crosspoint cresce quadraticamente e diventa rapidamente impraticabile.

1. Origine matematica della Clos Network

Il problema originario: costruire grandi matrici di commutazione

Il problema non è solo il numero astratto di crosspoint.
Ci sono almeno quattro limiti pratici:

Complessità hardware
Ogni punto di incrocio richiede un elemento di commutazione fisico o logico.

Costo
La crescita quadratica porta a costi molto elevati.

Ingombro
Più grande è la matrice, più difficile diventa realizzarla fisicamente.

Affidabilità e gestione
Una struttura monolitica enorme diventa più difficile da progettare, raffreddare, alimentare e mantenere.

1. Origine matematica della Clos Network

Il problema originario: costruire grandi matrici di commutazione

L'idea di Charles Clos è elegantissima:
invece di costruire un unico grande switch, conviene costruire più piccoli switch organizzati in più stadi.

Quindi:

- si spezza la grande matrice in blocchi più piccoli
- si collega ogni blocco a uno stadio intermedio
- si ricostruisce il comportamento di una grande matrice usando una rete multistadio

Questa è la vera svolta concettuale:
la connettività completa non richiede necessariamente una matrice monolitica.
In altre parole, una grande rete di switching può essere ottenuta come composizione di elementi più piccoli, con un enorme vantaggio in scalabilità.

1. Origine matematica della Clos Network

Il problema originario: costruire grandi matrici di commutazione

La Clos Network introduce un cambio di paradigma molto importante, che poi ritroveremo nei data center moderni:

Prima: un grande sistema centralizzato
Dopo: una rete di nodi più piccoli, interconnessi in modo regolare

Questa idea è la radice teorica di moltissime architetture moderne:

- **spine-leaf**
- **fat-tree**
- **fabric L3 con ECMP**
- **switching multistadio ad alta capacità**

In pratica, quando oggi parliamo di fabric, stiamo usando una reinterpretazione moderna dello stesso principio.

1. Origine matematica della Clos Network

Il problema originario: costruire grandi matrici di commutazione

È importante chiarire che Clos non propose semplicemente una “topologia furba”.

La forza del modello è che fornisce condizioni precise per sapere quando una rete multistadio è:

- **bloccante**
- **non bloccante**
- **riarrangiabile**

Quindi non è solo una questione di disegnare tre livelli di switch.
È una questione di dimostrare formalmente quanta interconnessione serve tra gli stadi per garantire certe proprietà.

Questa è la parte che rende la Clos Network molto più interessante di una semplice topologia data center.

2. Modello formale della Clos Network

2. Modello formale della Clos Network

Definizione della rete Clos

Una rete Clos classica è una rete di commutazione multistadio a tre livelli.
Formalmente viene indicata come:

C(m,n,r)

dove:
r = numero di switch nel primo stadio
n = numero di ingressi per ogni switch di primo stadio
m = numero di switch nello stadio centrale

Clos Network su Mikrotik

Modello Formale della Clos Network

C(m,n,r) – Rete a tre stadi

1. Input Stage

r Switch di Ingresso

n × m

n ingressi

m uscite

2. Middle Stage

m Switch Centrali

r × r

r ingressi

r uscite

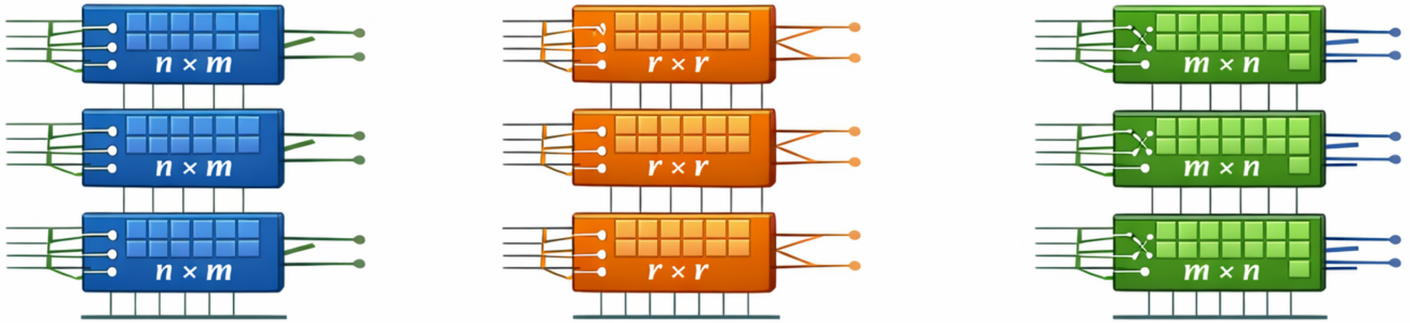
3. Output Stage

r Switch di Uscita

m × n

m ingressi

n uscite



N Ingressi Totali → *Stadio Centrale* ← *Stadio di Uscita*

Matrice di Commutazione N × N

2. Modello formale della Clos Network

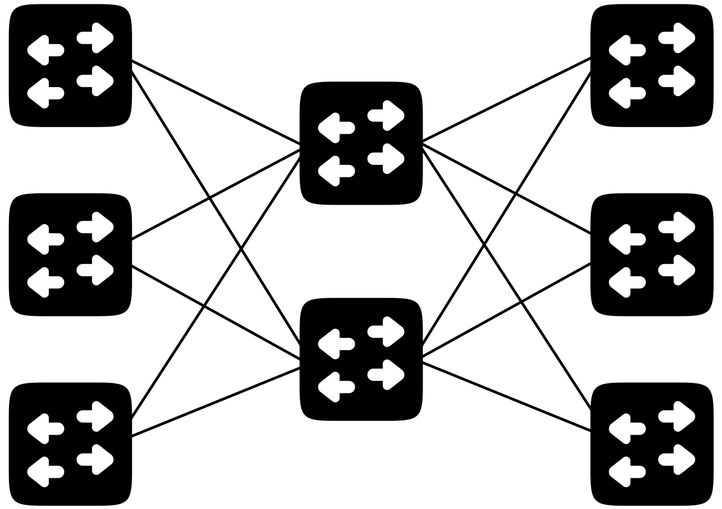
Definizione della rete Clos

Ogni switch dello stage 1 è connesso a tutti gli switch centrali.

Quindi ogni input switch ha:
m uplink

Analogamente, ogni output switch riceve:
m connessioni dagli switch centrali

Questo crea una topologia completamente bipartita tra stage 1 e stage 2, e tra stage 2 e stage 3.



2. Modello formale della Clos Network

Percorso di una connessione

Per stabilire una connessione tra un ingresso e un’uscita:

ingresso	→	input switch
input switch	→	middle switch
middle switch	→	output switch
output switch	→	uscita finale

Quindi ogni connessione attraversa esattamente tre switch.

Questo è uno dei motivi per cui le Clos hanno:

- **latenza deterministica**
- **numero di hop costante**

2. Modello formale della Clos Network

Percorso di una connessione

Supponiamo: **C(4,8,6)**

Significa:
r = 6 input switch
n = 8 ingressi per switch
m = 4 middle switch

Numero totale ingressi:
N=r×n
N=6×8=48

Quindi la rete realizza una matrice: 48×48

ma usando:
6 switch input
4 switch centrali
6 switch output

per un totale di 16 switch piccoli invece di una matrice 48×48.

2. Modello formale della Clos Network

Interpretazione moderna (Data Center)

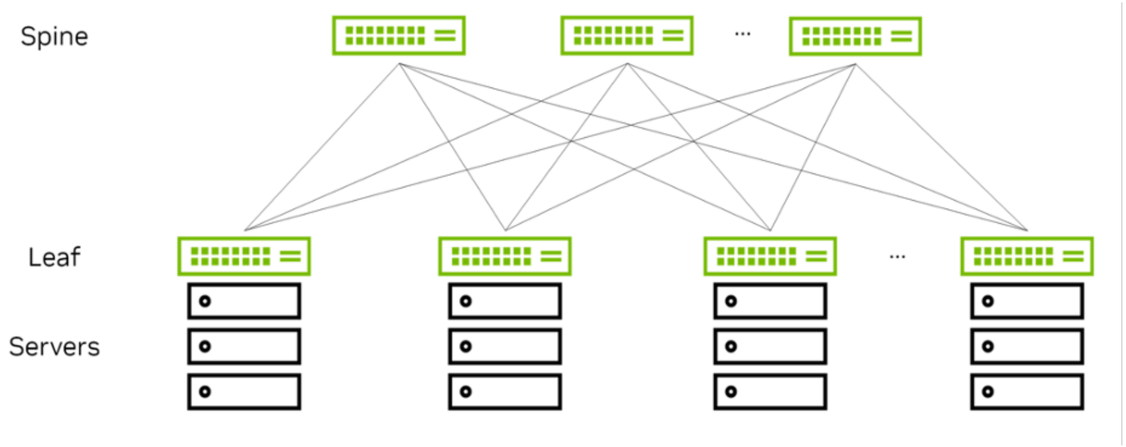
Nel mondo dei data center:

Clos originale	**Data Center**
input stage	leaf
middle stage	spine
output stage	leaf

Quindi la rete diventa:

server - leaf - spine - leaf - server

È esattamente una Clos a 3 stadi.



2. Modello formale della Clos Network

Proprietà chiave del modello

Una Clos ha queste proprietà:

Modularità
può essere costruita usando switch più piccoli.

Scalabilità
aggiungere capacità significa: aumentare r, aumentare m

Ridondanza
più middle switch = più percorsi

Latenza costante
tutte le connessioni hanno 3 stadi di attraversamento.

3. Condizione di Non-Blocking nella Clos Network

3. Condizione di Non-Blocking nella Clos Network

Il problema fondamentale

In una rete di commutazione vogliamo garantire che:

qualsiasi nuova connessione tra un ingresso libero e un’uscita libera possa sempre essere stabilita.

Se questo è sempre possibile senza dover riconfigurare connessioni esistenti, la rete è detta:

strict-sense non-blocking

Questa è la proprietà più forte possibile.

3. Condizione di Non-Blocking nella Clos Network

Worst Case

Supponiamo di voler stabilire una nuova connessione tra:
ingresso I su input switch A - uscita O su output switch B

Il percorso deve passare per uno dei **m** middle switch.

Ma potrebbero verificarsi due conflitti:
conflitti lato input switch: gli altri ingressi dello switch A potrebbero già usare alcuni middle switch. Numero massimo di conflitti: n−1 (perché un ingresso è quello che stiamo usando)

conflitti lato output switch: anche lo switch di uscita B può avere già connessioni attive verso altri middle switch. Anche qui massimo: n−1

3. Condizione di Non-Blocking nella Clos Network

Worst Case

Nel caso peggiore:

n−1 middle switch sono già usati dall'input switch
n−1 middle switch sono già usati dall'output switch

Quindi **il numero massimo di middle switch non disponibili è:**

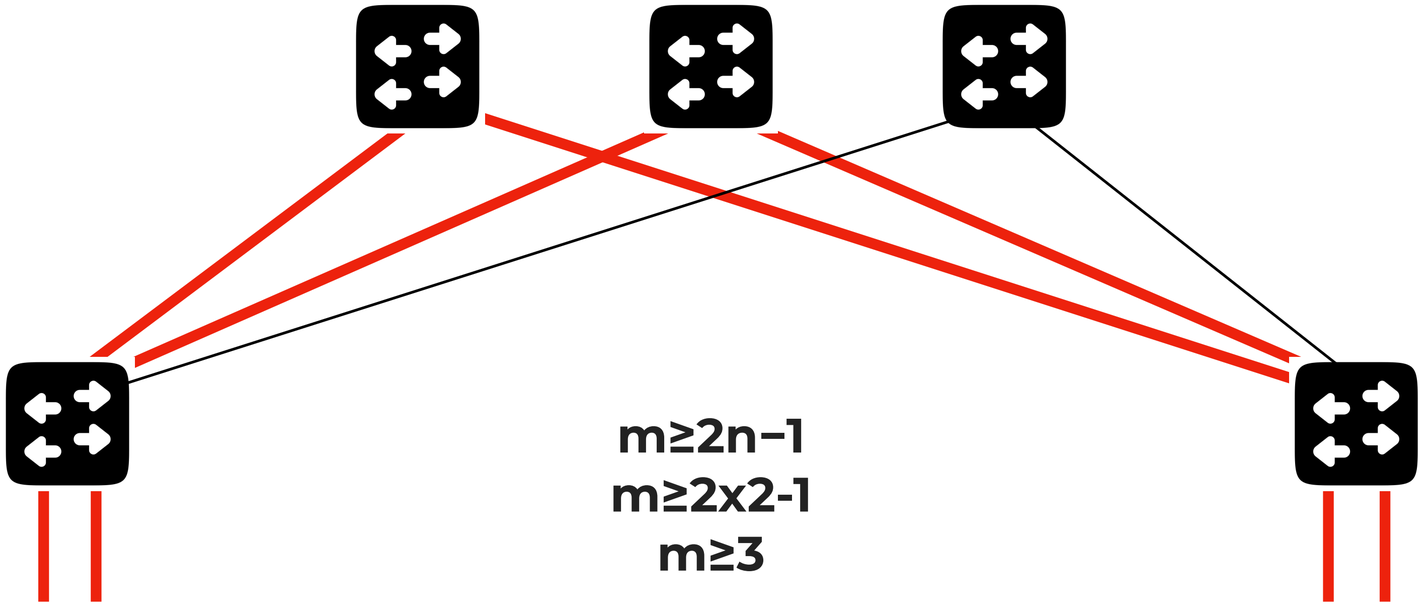
(n−1)+(n−1) = **2n−2**

Per garantire che almeno un middle switch resti disponibile, dobbiamo avere: **m≥2n−1**

Questa è la condizione di Clos per una rete strict-sense non-blocking.

3. Condizione di Non-Blocking nella Clos Network

Worst Case



3. Condizione di Non-Blocking nella Clos Network

Relazione con oversubscription

Oversubscription: il numero di uplink verso spine è minore del numero di downlink verso server

Leaf switch:

48 server ports
4 uplink spine

oversubscription: 48:4 = **12:1**

Questo viola la condizione di Clos, ma è accettabile perché:

- il traffico non è completamente full-mesh
- non tutti i server saturano contemporaneamente.

4. Tipi di Non-Blocking nelle Clos Network

4. Tipi di Non-Blocking nelle Clos Network

Strict-Sense Non-Blocking

È la condizione più forte.

Definizione:
Una rete è strict-sense non-blocking se qualsiasi nuova connessione tra ingresso libero e uscita libera può essere stabilita senza modificare alcuna connessione esistente.
Questa proprietà vale per qualsiasi sequenza di richieste di connessione.

La condizione matematica dimostrata da Clos è:

$m \geq 2n-1$
n = ingressi per input switch
m = middle switch

4. Tipi di Non-Blocking nelle Clos Network

Rearrangeable Non-Blocking

Questa è una proprietà molto più rilassata.

Definizione:
Una rete è rearrangeable non-blocking se ogni configurazione di connessioni è realizzabile, ma può essere necessario riconfigurare alcune connessioni esistenti.
Quindi una nuova connessione potrebbe richiedere:

- spostare alcune connessioni già stabilite
- cambiare i middle switch utilizzati

In questo caso basta che:
$m \geq n$
Quindi molto meno hardware rispetto al caso strict.

4. Tipi di Non-Blocking nelle Clos Network

Wide-Sense Non-Blocking

Questa è una condizione intermedia molto interessante.

Definizione:
Una rete è wide-sense non-blocking se esiste un algoritmo di instradamento che garantisce l'assenza di blocking.
Quindi la rete può funzionare senza blocchi se viene utilizzata una strategia di routing appropriata.

La differenza rispetto allo strict case è che:
- il routing deve essere controllato
- non tutte le configurazioni casuali funzionano.

4. Tipi di Non-Blocking nelle Clos Network

Wide-Sense Non-Blocking

Nei data center moderni quasi sempre siamo nel caso:
wide-sense non-blocking

perché il comportamento dipende da:

- **ECMP**
- **hashing dei flow**
- **distribuzione del traffico**

Non è garantita l'assenza teorica di blocking, ma nella pratica il traffico è distribuito.

4. Tipi di Non-Blocking nelle Clos Network

Collegamento con ECMP

Nel modello Clos originale il middle stage offre m percorsi possibili.
Nei data center questo si traduce in:

leaf → spine → leaf

con più spine disponibili.

L’algoritmo ECMP distribuisce i flussi sui vari spine.
Questo è concettualmente equivalente a un algoritmo di **wide-sense routing**.

5. Evoluzione: Fat-Tree e Multi-Stage Clos

5. Evoluzione: Fat-Tree e Multi-Stage Clos

Limite della Clos a 3 stadi

La Clos classica:

C(m,n,r)

è una rete a tre stadi.

Schema: **Input switches → Middle switches → Output switches**

Funziona molto bene fino a una certa dimensione, ma ha un limite: **il numero di porte degli switch cresce rapidamente.**
Per costruire reti molto grandi servirebbero switch centrali con molte porte, spesso più di quelle disponibili nell'hardware reale.

5. Evoluzione: Fat-Tree e Multi-Stage Clos

L’idea della Fat-Tree

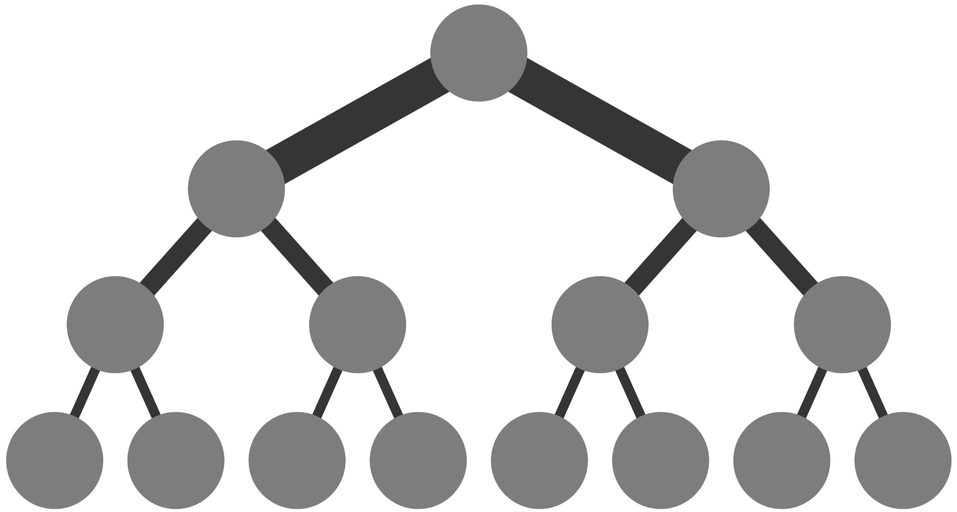
Per superare questo limite si introduce una generalizzazione:

la Fat-Tree network

La Fat-Tree è una struttura gerarchica di Clos ricorsiva, in cui:

- ogni livello superiore ha più capacità aggregata
- il numero di percorsi cresce man mano che si sale nella rete

La rete diventa quindi una Clos multi-stadio.



5. Evoluzione: Fat-Tree e Multi-Stage Clos

Perché si chiama Fat-Tree

In una topologia ad albero classica:

access
|
aggregation
|
root

la capacità diminuisce verso il basso.

Nella Fat-Tree, invece: **i link verso l'alto hanno maggiore capacità aggregata**
Per questo l'albero è "grasso" nella parte alta.

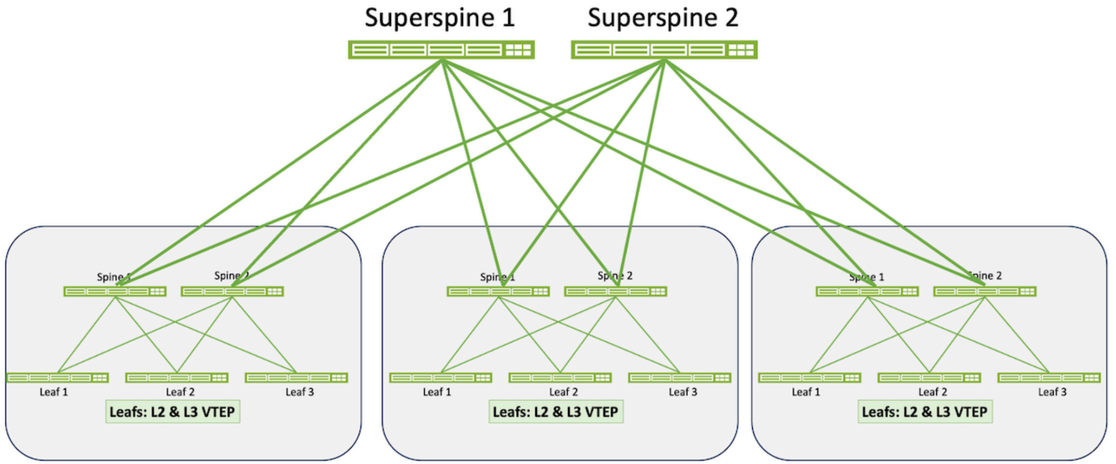
5. Evoluzione: Fat-Tree e Multi-Stage Clos

Clos a 5 stadi

Nei data center moderni la forma più comune è la 5-stage Clos.

leaf - spine - super-spine - spine - leaf

Questa architettura consente di scalare a centinaia di migliaia di server.



5. Evoluzione: Fat-Tree e Multi-Stage Clos

Numero di percorsi possibili

Una proprietà fondamentale della Clos multi-stadio è il numero di percorsi alternativi.

il numero di percorsi possibili tra due leaf cresce rapidamente.

Questo consente:

- **load balancing**
- **fault tolerance**
- **alta capacità aggregata**

5. Evoluzione: Fat-Tree e Multi-Stage Clos

Dimensionamento pratico

Esempio tipico di leaf CRS518 -16XS-2XQ:

16 × 25G server ports
2 × 100G uplink

Ogni leaf ha uplink verso tutti gli spine.

Se abbiamo 6 spine, ogni leaf ha 2 percorsi indipendenti verso ogni altro leaf.

6. Oversubscription e dimensionamento

6. Oversubscription e dimensionamento

Il concetto di oversubscription

Nei data center moderni quasi nessuna rete è strictly non-blocking nel senso teorico di Clos.

Il motivo è semplice:
costruire una rete completamente non bloccante sarebbe troppo costoso.

Si introduce quindi il concetto di **oversubscription**.

Definizione:

$$Oversubscription = \frac{Downlink\ Capacity}{Uplink\ Capacity}$$

6. Oversubscription e dimensionamento

Esempio pratico

Leaf switch CRS518 -16XS-2XQ:

16 × 25G server ports
2 × 100G uplink

Downlink totale: 16×25 = 400Gbps
Uplink totale: 2×100 = 200Gbps
Oversubscription: 400/200 = **2:1**

Questo significa che se tutti i server saturano contemporaneamente, la fabric diventa il collo di bottiglia.

6. Oversubscription e dimensionamento

Perché è accettabile

In pratica il traffico nei data center non è:

all-to-all full bandwidth

ma piuttosto:

- **bursty**
- **distribuito**
- **spesso localizzato**

Quindi una certa oversubscription è accettabile.

Ambiente	Oversubscription
HPC	1:1
hyperscaler	2:1
enterprise	3–5:1
campus	10–20:1

7. Routing nelle Clos Network

7. Routing nelle Clos Network

Perché il fabric non deve essere L2

Le architetture Clos / spine-leaf sono progettate per scalare aumentando il numero di nodi e di percorsi disponibili.

Questo modello funziona correttamente solo se il fabric opera a Layer 3.

Un fabric L2 introduce domini di broadcast molto estesi, nei quali traffico come ARP e unknown unicast tende a propagarsi in tutta la rete.

Inoltre, protocolli come Spanning Tree disabilitano parte dei link per evitare loop, impedendo di sfruttare pienamente i percorsi multipli.

Per questo motivo nei data center il fabric è L3, mentre la connettività Layer 2 viene fornita sopra il fabric tramite overlay come VXLAN.

7. Routing nelle Clos Network

Quando utilizzare un IGP

Un modo semplice per costruire il fabric è **utilizzare un IGP come OSPF o IS-IS**.

Questo approccio funziona bene in **fabric di dimensioni moderate**, dove il numero di nodi è limitato e la topologia è relativamente stabile.

Gli IGP offrono una convergenza rapida e **supportano naturalmente il multipath routing**, fondamentale nelle architetture spine-leaf.

Tuttavia, gli IGP mantengono una visione globale della topologia, che può diventare difficile da gestire quando il fabric cresce molto.

7. Routing nelle Clos Network

Quando utilizzare eBGP

Utilizziamo **eBGP anche nell'underlay del fabric**.

In questo modello ogni collegamento leaf–spine è una **sessione eBGP point-to-point**.

Questo permette di costruire fabric molto grandi mantenendo un comportamento semplice e prevedibile.

BGP offre inoltre **maggiore controllo sulle rotte** e consente di mantenere il piano di controllo distribuito, caratteristica che rende questo approccio particolarmente adatto ai data center di grandi dimensioni.

7. Routing nelle Clos Network

Il problema del Valley Routing

Quando si usa BGP in una topologia gerarchica, entra in gioco un principio fondamentale: **il valley-free routing**.

Nel mondo Internet questo concetto nasce dalle relazioni economiche tra customer, provider e peer. Un percorso valido può “salire” verso un provider e poi “scendere” verso il destinatario, ma non dovrebbe tornare a salire di nuovo, perché questo violerebbe la gerarchia del routing e porterebbe a percorsi anomali o non desiderati.

Trasportato in una topologia Clos, il principio resta lo stesso anche se cambia il contesto: **il traffico deve seguire una gerarchia ordinata, non muoversi in modo arbitrario nel fabric.**

7. Routing nelle Clos Network

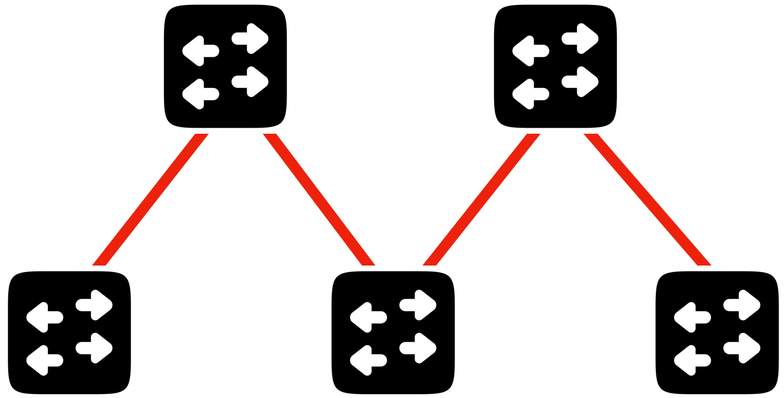
Valley-free routing applicato a una Clos

una fabric Clos / spine-leaf, la gerarchia è molto semplice:
leaf → spine → leaf

Un pacchetto parte da un leaf di ingresso, sale verso uno spine e poi scende verso il leaf di uscita. Questo è il comportamento naturale della topologia.

Il problema nasce quando il controllo del routing consente percorsi del tipo: **leaf → spine → leaf → spine**

In una Clos ben progettata, il traffico non deve mai "rimbalzare" tra livelli diversi in modo incoerente.



7. Routing nelle Clos Network

Come si evita il valley routing

Per evitare questo problema, una Clos costruita con BGP impone una gerarchia molto precisa nel piano di controllo.

Le tecniche principali sono due:

1. **Assegnare Autonomous System distinti ai diversi livelli della fabric, oppure a ogni singolo switch**. In questo modo la topologia gerarchica viene riflessa anche nella struttura degli AS.

2. **Definire regole di propagazione BGP coerenti con la gerarchia.** Un leaf annuncia verso gli spine, uno spine riannuncia verso i leaf, ma non deve diventare un punto da cui il traffico o gli annunci possano risalire di nuovo dopo essere già scesi.

7. Routing nelle Clos Network

Modello 1: un unico AS

Il primo modello possibile è quello con **un unico AS per tutta la fabric**. In questo schema tutti i router appartengono allo stesso sistema autonomo.

Dal punto di vista concettuale è il modello più semplice da immaginare, ma nella pratica introduce complessità operative importanti.

Essendo tutti nello stesso AS, **il routing tra i nodi del fabric richiede iBGP**. Questo significa che non basta collegare semplicemente leaf e spine: occorre risolvere le limitazioni dell'iBGP, **che non propaga le rotte apprese da un peer iBGP verso un altro peer iBGP**.

Di conseguenza **diventano necessari** meccanismi aggiuntivi, tipicamente **route reflector**, che aumentano la complessità del progetto.

7. Routing nelle Clos Network

Modello 2: un AS per ruolo

Un secondo modello, molto più coerente con la struttura della Clos, consiste nell'**assegnare un AS a ciascun livello della gerarchia**.

tutti i leaf in un AS, tutti gli spine in un altro AS

Questo schema è già molto più efficace, perché trasfo**rma i collegamenti leaf–spine in sessioni eBGP**. La separazione tra livelli diventa così evidente anche nel piano di controllo.

Il vantaggio principale è che la gerarchia logica della fabric viene rappresentata direttamente nella gerarchia degli AS. Questo rende molto più naturale applicare politiche che impediscano percorsi non desiderati e aiuta a mantenere il routing valley-free.

7. Routing nelle Clos Network

Modello 3: un AS per switch

Il terzo modello è quello più radicale e, in molti contesti hyperscale, anche il più elegante: **ogni switch ha il proprio AS**.

Leaf1 → AS65101
Leaf2 → AS65102
Spine1 → AS65201
Spine2 → AS65202

In questo modo **ogni collegamento leaf–spine è una normale sessione eBGP tra due AS distinti**. Il fabric diventa una collezione di peering eBGP point-to-point perfettamente aderente alla topologia fisica.

Questo modello è spesso chiamato **eBGP Clos Fabric**.

7. Routing nelle Clos Network

Il modello “AS per switch” è efficace

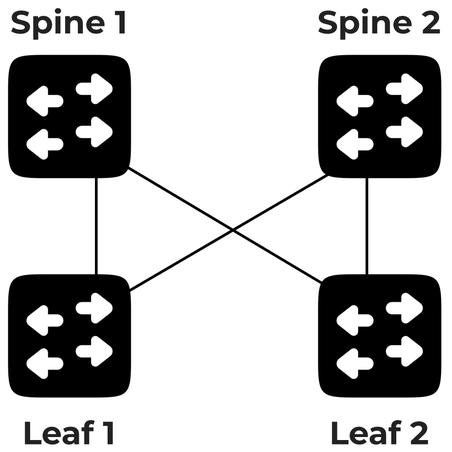
Assegnare un AS a ogni switch porta diversi vantaggi molto concreti.

1. **Non servono route reflector**, perché tutto il fabric è costruito su eBGP. Questo elimina una delle principali fonti di complessità dei design iBGP.
2. **Le policy diventano più semplici**. Ogni peering rappresenta una relazione chiara tra due nodi adiacenti, e le regole di import/export sono più facili da comprendere e da controllare.
3. **La convergenza tende a essere rapida** e il comportamento del controllo più lineare. Inoltre, la presenza dell’AS_PATH introduce un meccanismo naturale di protezione contro percorsi che violano la gerarchia del fabric.

La topologia fisica, la topologia logica e il modello di routing tendono a coincidere.

7. Routing nelle Clos Network

Esempio di BGP Clos fabric



Leaf1 stabilisce sessioni eBGP verso Spine1 e verso Spine2.
Allo stesso modo ogni spine mantiene sessioni verso ciascun leaf.

In questo modello ogni leaf annuncia tipicamente:

- **la propria loopback**
- **le subnet direttamente connesse**

Gli spine hanno il compito di fungere da livello di transito e redistribuzione dei prefissi appresi dai leaf verso gli altri leaf.

Questo comportamento ricorda una forma di “riflessione”, ma avviene in modo naturale tramite eBGP, **senza introdurre veri route reflector**.

8. ECMP nelle Clos Network

8. ECMP nelle Clos Network

Equal Cost Multi Path

Una delle proprietà più eleganti delle Clos network è che la topologia stessa genera molti percorsi con costo identico tra ogni coppia di nodi.

In una struttura leaf–spine, il percorso tra due server è: **server → leaf → spine → leaf → server**

Se il fabric ha k spine, tra due leaf esistono esattamente k percorsi distinti.

Tutti questi percorsi hanno:

- **stesso numero di hop**
- **stessa latenza**
- **stesso costo di routing**

Questa è esattamente la condizione richiesta da ECMP (Equal Cost Multi Path).

8. ECMP nelle Clos Network

Che cosa richiede ECMP

ECMP può essere applicato quando esistono più percorsi con costo identico verso una destinazione.

$Cost(P_1) = Cost(P_2) = ... = Cost(P_n)$

Quando questa condizione è soddisfatta, il router può installare più next-hop nella FIB e distribuire il traffico tra di essi.

Le Clos network sono progettate esattamente per creare questa situazione.
La topologia completamente bipartita tra leaf e spine garantisce che tra ogni coppia di leaf esista lo stesso numero di percorsi equivalenti.

La Clos non è semplicemente compatibile con ECMP: è costruita per sfruttarlo.

8. ECMP nelle Clos Network

Simmetria della topologia Clos

Un'altra proprietà fondamentale della Clos è la simmetria strutturale.

Ogni leaf ha:

- **lo stesso numero di uplink**
- **la stessa capacità verso il fabric**
- **la stessa distanza da tutti gli altri leaf**

Di conseguenza **nessun percorso è privilegiato rispetto agli altri**.

Questo elimina uno dei problemi principali delle architetture gerarchiche tradizionali, nelle quali alcuni link diventano punti di congestione mentre altri restano sottoutilizzati.
Tutti i percorsi sono equivalenti e utilizzabili contemporaneamente.

8. ECMP nelle Clos Network

Distribuzione del traffico con ECMP

I router non dividono quasi mai il traffico per pacchetto, ma utilizzano un **meccanismo di flow hashing**.

Il router calcola un hash su alcuni campi del traffico, ad esempio:

- **indirizzo IP sorgente**
- **indirizzo IP destinazione**
- **porta sorgente**
- **porta destinazione**
- **protocollo**

Il risultato dell'hash determina quale percorso ECMP verrà utilizzato. In questo modo flussi diversi possono essere distribuiti tra spine differenti, mantenendo coerenza per singolo flusso.

8. ECMP nelle Clos Network

Capacità lineare della fabric

Una conseguenza diretta di questa proprietà è che la capacità della fabric cresce quasi linearmente con il numero di spine.

Se ogni leaf è collegato a k spine, la capacità teorica del fabric tra due leaf è circa:

fabric bandwidth ≈ k × bandwidth del link verso spine

Esempio:

6 spine, 100 Gbit/s per link
capacità potenziale tra due leaf: ≈ 600 Gbit/s

8. ECMP nelle Clos Network

Ridondanza naturale

La presenza di molti percorsi equivalenti introduce anche una ridondanza naturale.

Se uno spine fallisce, il numero di percorsi disponibili passa semplicemente da:

k → k − 1

Il protocollo di routing aggiorna automaticamente la FIB e il traffico continua a essere distribuito sugli altri spine.

La resilienza è una proprietà intrinseca della topologia.

8. ECMP nelle Clos Network

ECMP nelle fabric MikroTik

In RouterOS la distribuzione ECMP è controllata dal parametro:

/ip/settings ipv4-multipath-hash-policy

Le modalità disponibili sono:

- **l3 — hashing basato su src IP e dst IP**
- **l4 — hashing sulla 5-tuple (IP, protocollo, porte)**
- **l3-inner — hashing sugli indirizzi interni nel traffico incapsulato**

L’efficacia di ECMP dipende dall’entropia disponibile nei campi usati dall’hash.
Più campi vengono utilizzati, più uniforme sarà la distribuzione del traffico tra gli spine.

8. ECMP nelle Clos Network

ECMP con L3 Hardware Offloading

Quando il routing viene offloadato nell'ASIC tramite **L3 Hardware Offloading**, il forwarding non passa più dalla CPU ma avviene direttamente nello switch chip.
Questo significa che anche **la selezione del next-hop ECMP viene eseguita in hardware**.

Gli ASIC utilizzati negli switch MikroTik operano tipicamente fino al livello IP, quindi l'hash ECMP utilizzato dal forwarding hardware si basa sui campi:

- **source IP**
- **destination IP**

Tutti i flussi tra la stessa coppia di host tendono a seguire lo stesso percorso ECMP.

8. ECMP nelle Clos Network

Implicazioni progettuali per una Clos MikroTik

La distribuzione del traffico dipende dalla diversità delle coppie di indirizzi IP presenti nel traffico.

Nei data center, molti server comunicano con molti altri server, generando naturalmente un numero elevato di combinazioni IP. Anche con hashing L3 la distribuzione dei flussi tra gli spine risulta generalmente uniforme.

Lo stesso vale nelle reti ISP: il traffico proviene da migliaia di utenti e verso un'enorme varietà di destinazioni su Internet. Questa diversità di coppie src/dst IP fornisce entropia sufficiente affinché ECMP L3 distribuisca efficacemente il traffico tra i percorsi disponibili.

Con hashing basato solo su Layer 3, una fabric Clos può sfruttare tutti i percorsi ECMP.

8. ECMP nelle Clos Network

Introduzione al BGP Multipath

Tradizionalmente **BGP seleziona un solo “best path” per ogni prefisso** e lo installa nella tabella di routing. Questo comportamento garantisce stabilità nel piano di controllo, ma impedisce di utilizzare simultaneamente più percorsi equivalenti.

Nelle versioni recenti di RouterOS v7 è stato introdotto il supporto a BGP Multipath, che consente di installare più percorsi BGP equivalenti verso lo stesso prefisso.

Quando più route hanno attributi compatibili (stesso costo e parametri di selezione equivalenti), il router può installare più next-hop nella FIB, rendendoli disponibili per il forwarding.

Questo permette di sfruttare più collegamenti verso la stessa destinazione.

8. ECMP nelle Clos Network

Dal BGP Multipath a ECMP

Una volta che più percorsi BGP equivalenti vengono installati nella FIB, il forwarding può utilizzare ECMP (Equal Cost Multi Path) per distribuire il traffico tra i diversi next-hop.

Il router applica una funzione di hash sui campi IP del traffico per determinare quale percorso utilizzare, garantendo che flussi diversi possano seguire percorsi diversi.

In questo modo BGP Multipath e ECMP lavorano insieme:

- **BGP distribuisce le rotte e rende disponibili più percorsi equivalenti**
- **ECMP distribuisce i flussi tra questi percorsi**

Questo meccanismo è particolarmente efficace nelle architetture Clos / spine-leaf, dove tra due nodi esistono naturalmente molti percorsi con costo identico.

9. Underlay e Overlay nelle Clos Fabric

9. Underlay e Overlay nelle Clos Fabric

I due livelli logici

Underlay
la rete IP fisica che collega gli switch del fabric.

Overlay
la rete virtuale che trasporta i servizi di livello superiore (L2 o L3).

Questa separazione permette di mantenere il fabric semplice, stabile e altamente scalabile, mentre i servizi di rete vengono implementati sopra di esso.

L’overlay è spesso realizzato con VXLAN e controllato tramite EVPN.

9. Underlay e Overlay nelle Clos Fabric

Che cosa resta nell'Underlay

L'underlay ha un obiettivo molto preciso: fornire connettività IP affidabile tra tutti i nodi del fabric.

Per questo motivo l'underlay dovrebbe contenere solo pochi elementi fondamentali:

- **routing IP (IGP o eBGP)**
- **loopback dei router**
- **indirizzi dei VTEP**
- **ECMP tra spine e leaf**

Il traffico nell'underlay è quindi puro traffico IP.
Questo approccio permette di sfruttare completamente ECMP e L3 Hardware Offloading, consentendo al fabric di inoltrare pacchetti a wire speed direttamente nello switch ASIC invece che nella CPU.

9. Underlay e Overlay nelle Clos Fabric

Che cosa va nell'Overlay

Tutto ciò che richiede astrazione o virtualizzazione viene invece spostato nell'overlay.

Tipicamente nell'overlay troviamo:

- **domini Layer-2 estesi**
- **segmentazione multi-tenant**
- **mobilità delle VM**
- **servizi virtuali**
- **traffico utenti**

Questo viene realizzato **incapsulando i frame Ethernet dentro VXLAN**, che permette di trasportare Layer-2 sopra una rete Layer-3.
Ogni dominio logico viene identificato da un VNI (VXLAN Network Identifier).

9. Underlay e Overlay nelle Clos Fabric

VXLAN: il data plane dell’overlay

VXLAN incapsula frame Ethernet in pacchetti UDP/IP, permettendo di trasportare traffico L2 sopra una rete L3.

Gli endpoint di incapsulamento sono chiamati:
VTEP — VXLAN Tunnel Endpoints

Il percorso diventa quindi: **Host → Leaf (VTEP) → IP Fabric → Leaf (VTEP) → Host**

Questo significa che:

- **l’underlay vede solo traffico IP**
- **la complessità L2 resta confinata nell’overlay.**

9. Underlay e Overlay nelle Clos Fabric

EVPN: il control plane dell’overlay

Se VXLAN è il data plane, EVPN è il control plane.

EVPN utilizza MP-BGP per distribuire nel fabric informazioni come:

- **MAC address**
- **binding MAC/IP**
- **membership dei domini VXLAN**

In questo modo **gli switch non devono più imparare MAC tramite flooding**: le informazioni vengono annunciate via BGP.

Questo riduce drasticamente il traffico di broadcast e rende il sistema molto più scalabile.

9. Underlay e Overlay nelle Clos Fabric

Hardware Offloading e Overlay

Una delle sfide delle architetture overlay è mantenere prestazioni elevate nonostante l'incapsulamento.

Le piattaforme MikroTik permettono ora di combinare:

- **L3 hardware offloading**
- **forwarding ECMP**
- **VXLAN**

Quando supportato dall'hardware, parte del forwarding VXLAN può essere gestito direttamente dall'ASIC, permettendo di mantenere throughput molto elevato anche con traffico incapsulato.

È possibile costruire fabric Clos con overlay VXLAN mantenendo prestazioni quasi lineari.

10. Progettazione di una Clos Network MikroTik

10. Progettazione di una Clos Network MikroTik

Dimensionamento di una Clos Fabric

Il dimensionamento di una Clos / spine-leaf non parte dal numero di spine, ma dal leaf.

Il leaf determina:

- **numero di endpoint collegati**
- **velocità delle porte di accesso**
- **traffico atteso**

Da questi parametri si calcola la capacità totale sud (southbound) del leaf.

Il dimensionamento del fabric consiste nel determinare quanta capacità nord (northbound) serve per ottenere il rapporto di oversubscription desiderato.

10. Progettazione di una Clos Network MikroTik

Dimensionare il numero di Spine

In una Clos spine-leaf ogni leaf è collegato a tutti gli spine.

Il numero di spine coincide quindi normalmente con il numero di uplink per leaf.

Esempio:
Leaf uplink: 2 × 100G → fabric con 2 spine

Questo produce:

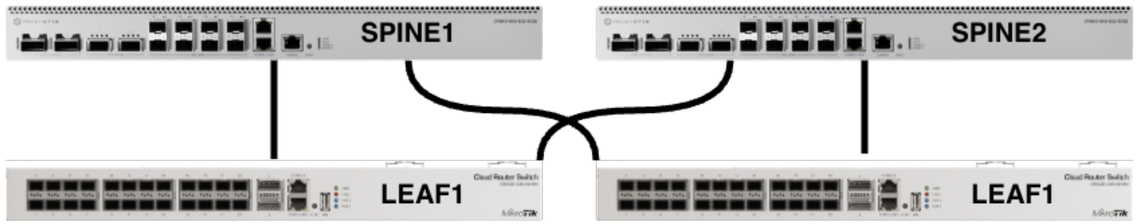
- **2 percorsi ECMP**
- **massima simmetria**
- **bilanciamento naturale del traffico**

10. Progettazione di una Clos Network MikroTik

Dimensionare il numero di Spine

Leaf **CRS326-24S+2Q+**
Spine **CRS812-8DS-2DQ-2DDQ**

Downlink: 24x10G=240G
Uplink: 2x40G=80G
Oversubscription: **1:3**



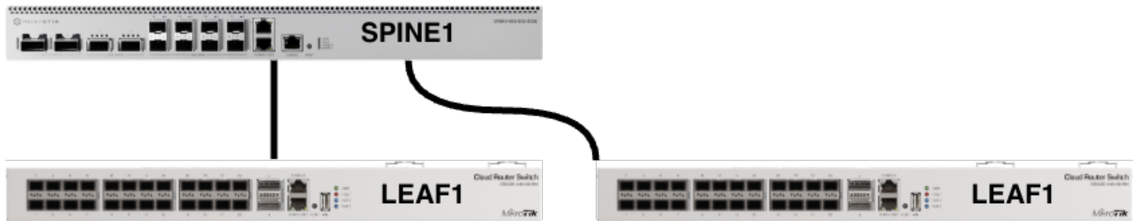
10. Progettazione di una Clos Network MikroTik

Dimensionamento N-1

Una best practice importante consiste nel dimensionare il fabric considerando la perdita di uno spine.

Leaf **CRS326-24S+2Q+**
Spine **CRS812-8DS-2DQ-2DDQ**

Downlink: 24x10G=240G
Uplink: **1**x40G=40G
Oversubscription: **1:6**

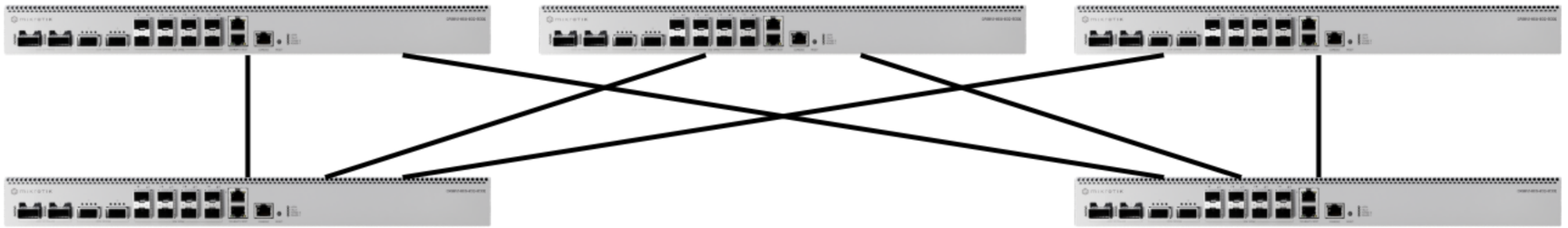


10. Progettazione di una Clos Network MikroTik

Dimensionamento N-1

Leaf **CRS812-8DS-2DQ-2DDQ**
Spine **CRS812-8DS-2DQ-2DDQ**

Downlink: 8x50G+1x200G=600G
Uplink: 3x200G=600G / **2x200G=400G** *(tutti gli uplink a 200G)*
Oversubscription: 1:1 / **1:1.5**



10. Progettazione di una Clos Network MikroTik

Evoluzione degli switch CRS per fabric spine-leaf

La famiglia CRS non è più limitata a switch con CPU minimale. Alcuni modelli includono CPU ARM64 multi-core insieme a switch ASIC ad alte prestazioni.

CRS804-4DDQ-hRM
CRS812-8DS-2DQ-2DDQ-RM
CRS520-4XS-16XQ-RM
CRS418-8P-8G-2S+-RM

Queste piattaforme integrano CPU ARM64 quad-core 2GHz e circa 4GB di RAM, insieme a switch chip Marvell Prestera che garantiscono forwarding a wire-speed.

10. Progettazione di una Clos Network MikroTik

Perché ARM64 è importante nei leaf e spine

In una fabric spine-leaf gli switch non fanno solo switching:

- **partecipano a BGP / EVPN**
- **gestiscono il control-plane dell'overlay**
- **eseguono policy e routing**

Queste operazioni benché offloadate nello switch chip, dipendono anche dalla CPU di controllo.

Per questo motivo sono adatti come leaf o spine della fabric, non solo come switch.

10. Progettazione di una Clos Network MikroTik

Dove usare switch con CPU più limitata

Switch con CPU più piccola (ad esempio modelli storicamente basati su MIPS) hanno senso quando il traffico rimane quasi completamente nello switch ASIC.

- **switching L2 puro**
- **bridging VLAN**
- **VXLAN completamente offloaded**

In questo scenario la CPU gestisce solo:

- **management**
- **control-plane minimo**

Il traffico invece resta in hardware forwarding path.

10. Progettazione di una Clos Network MikroTik

VXLAN hardware offloaded: vantaggi e limiti

RouterOS supporta VXLAN hardware-offloaded data plane, che consente di incapsulare traffico overlay direttamente nello switch chip.

- **throughput elevato**
- **bassa latenza**
- **CPU quasi inattiva**

Tuttavia esistono vincoli architetturali importanti:

- **mapping VLAN ↔ VNI 1:1**
- **VTEP solo su interfacce Ethernet routed**
- **VLAN tagging over VXLAN non supportato**

10. Progettazione di una Clos Network MikroTik

Quando disabilitare l’offloading

Se il progetto richiede **trasporto VLAN dentro VXLAN** o **elaborazioni non supportate dall’ASIC** l’offloading deve essere disabilitato.

In questi casi è preferibile usare piattaforme ARM64 con CPU molto potente, come router **CCR2216** o switch CRS con CPU ARM64, che possono gestire l’overlay in software mantenendo comunque throughput elevato.

11. Conclusioni

11. Conclusioni

Perché le Clos dominano i Data Center

Le reti moderne richiedono:

- **capacità di crescita quasi lineare**
- **multipli percorsi equivalenti**
- **resilienza intrinseca**
- **latenza prevedibile**

Le architetture tradizionali non soddisfano questi requisiti. La Clos network invece offre:

- **scalabilità modulare**
- **ECMP naturale**
- **capacità aggregata crescente**
- **fault tolerance strutturale**

Per questo motivo oggi costituisce la base di quasi tutte le fabric data center.

11. Conclusioni

Il ruolo di ECMP e della simmetria

La forza della Clos non è solo topologica.

È la combinazione di:

- **topologia completamente bipartita**
- **percorsi equivalenti**
- **routing multipath**

Tra due leaf esistono k percorsi identici, dove k è il numero di spine.

Questo permette:

- **bilanciamento del traffico**
- **resilienza automatica**
- **crescita lineare della capacità della fabric.**

11. Conclusioni

MikroTik come piattaforma per Clos Fabric

Le piattaforme MikroTik attuali rendono possibile costruire fabric spine-leaf ad alte prestazioni.

- **switch ASIC con forwarding wire-speed**
- **supporto ECMP e BGP multipath**
- **VXLAN hardware offload**
- **CPU ARM64 per il control-plane**

Queste piattaforme permettono di realizzare fabric Clos:

- **scalabili**
- **efficienti**
- **economicamente sostenibili.**

11. Conclusioni

Una Clos network non è semplicemente una topologia.

È un modello matematico di scalabilità delle reti.

Dal lavoro di Charles Clos nel 1953 alle fabric moderne, lo stesso principio rimane valido:
costruire grandi sistemi distribuendo la complessità su molti elementi più piccoli.

Le architetture spine-leaf non sono quindi una moda dei data center.
Sono la naturale evoluzione delle reti di commutazione su larga scala.

11. Conclusioni

Dalla teoria alla pratica

I principi descritti in questa presentazione non sono soltanto modelli teorici.

Nell’ultimo anno ho **progettato e realizzato** numerose architetture di rete basate su fabric Clos e topologie spine-leaf utilizzando piattaforme MikroTik, soprattutto in contesti ISP e infrastrutture ad alta capacità.

Queste architetture si sono dimostrate **battle tested**: reti scalabili, resilienti e operative ogni giorno in ambienti di produzione reali.

WHOLESALE WINERY Tour

THANK YOU!

Alessandro Campanella