

h2

R2:			R3:		
D.N	N.H	D	D.N	N.H	D
205	R2	1	203	INS.DIR	0
208	R3	1	204	INS.DIR	0
202	R3	1	205	INS.DIR	0
209	R1	1	209	R1	1
202	R1	1	202	R1	1
206	R1	1	203	R1	1
207	R1	1	205	R1	1
203	R2	1	206	R1	1
204	R2	1	209	R3	1
209	R2	2			
202	R2	2			
208	R3	1			
209	R3	1			

→ perché R3 dice a R1 209 | R3 | 1.

Per iniziare R1 comunica con R2, e poi con R3. Dopo di che si ha il seguente ordine:

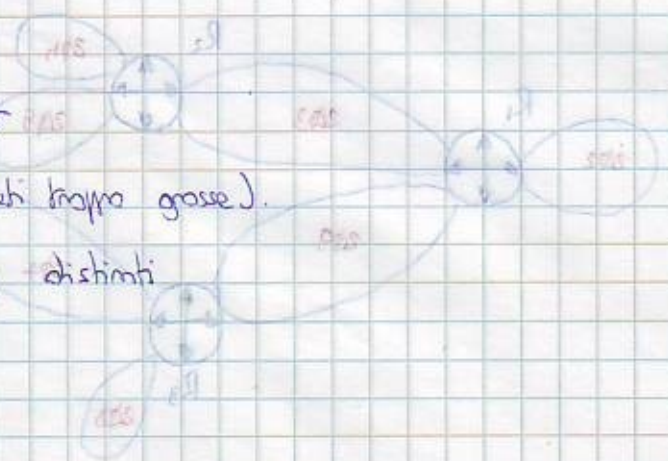
- R2 → R1, R2 → R3
- R3 → R1, R3 → R4
- R4 → R2, R4 → R3

Finisce così il primo ciclo di 30 sec. Nel secondo ciclo (di conferma) si ha:

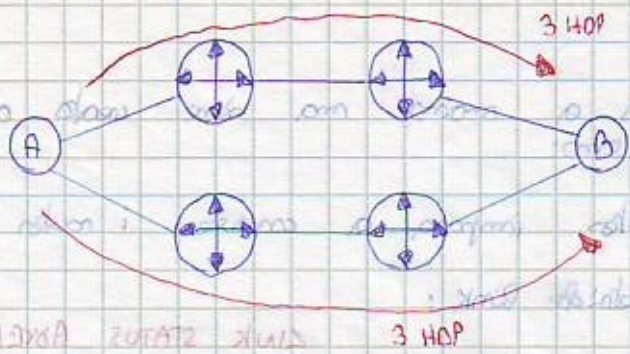
R1:			R2:			R3:			R4:		
D.N	N.H	D	D.N	N.H	D	D.N	N.H	D	D.N	N.H	D
206	R2	2									

Siamo quindi arrivati alla convergenza. Se voglio andare da 202 a 206? R1 passa attraverso R2, ed R2 passa attraverso R4 e poi R3. Questo è il percorso a minimo hop count. Quindi le caratteristiche del RIP2 sono:

- 1) semplicità
- 2) Fornisce cammino a minimi hop count
- 3) È poco scalabile (non riesce a gestire reti troppo grosse).
- 4) È possibile creare domini di routing distinti
- 5) Gestisce le subnetting
- 6) È possibile effettuare il **load sharing**

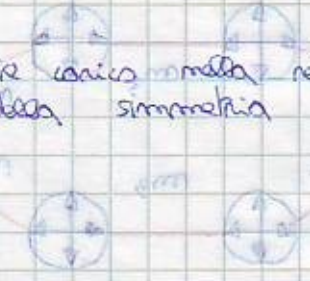
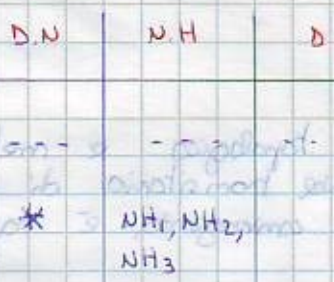


Per quanto riguarda il load sharing, RIP2 può essere configurato in modo che quando capita questa situazione:



RIP 2. E tiene tutti e due i percorsi. Riponisce il unico a 50% nelle due strade.

Questa ci permette di spongere meglio le uniche malfa nelle ottenendo casi migliori performance. Si cerca quindi di fare della simmetria sui link. Se ho una situazione del seguente tipo:

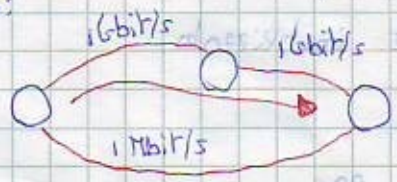


Ha una molteplicità di Next hop. Si può quindi fare la ripartizione uniforme del carico.

Analizziamo ora l'algoritmo OSPF. OSPF sta per Open Shortest Path First. Tale algoritmo si basa sul **link state**. Questa tecnica risulta essere più fine del **DISTANCE VECTOR** ma più complessa. Quindi l'algoritmo di routing sono più efficienti e scalabili ma risulta essere più difficile da gestire. Quando si applica il **link state** il router deve possedere un **DATABASE TOPOLOGICO** cioè un **DATABASE METRICHE LINK**. In tale algoritmo ogni link ha una metrica di valore:

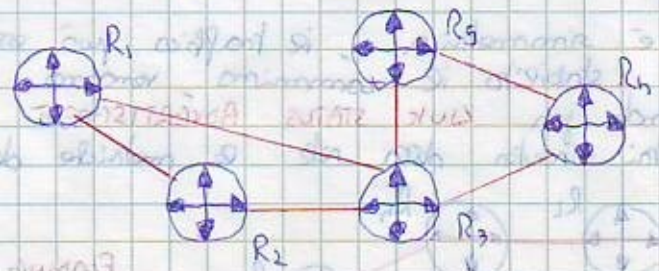
METRIKA LINK = $\frac{1}{\text{capacità del link}}$

Si può dire la metrica di un percorso = \sum delle metriche dei link del percorso. Per esempio se ho:



Quindi la metrica si può basare sulla capacità. In questo caso si parla di **METRIKA NON TOPOLOGICA**.

Poi ci sono anche le **metriche statistiche** cioè quelle metriche che non cambiano al variare del traffico in rete. Il database topologico è una rappresentazione topologica della rete. È una struttura dati e più precisamente è una matrice $m \times m$, con m = numero di router. Per esempio:



Ogni router ha un numero convenzionale all'interno della rete. Quindi:

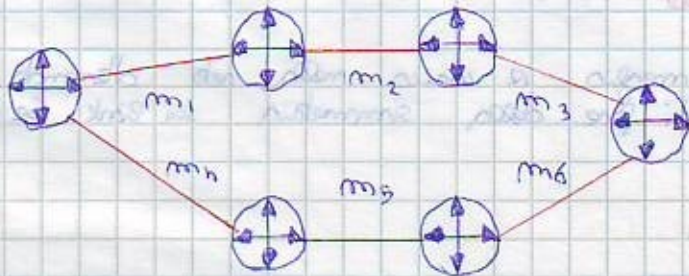
	1	2	3	...	m
1	1	0	1		
2	0	1			
...					
m					1

NB: È una matrice di incidenza.

Il router R_i è connesso al router R_j .

Il database topologico non viene inserito a mano, ma viene creato automaticamente, quindi è fuori dell'algoritmo stesso.

- 1) **Network Discovery** dove ogni router impara a conoscere i router vicini
- 2) viene costruito un database mediante link:



LINK STATUS ADVERTISEMENT

viene applicato l'algoritmo shortest path.

Se tutti i router hanno lo stesso database topologico e metriche, tutti i router hanno gli stessi percorsi coerenti. Nel transitorio di convergenza i percorsi possono non essere coerenti. Se la convergenza è rapida, questo problema non causa gravi ripercussioni.

L'OSPF è l'algoritmo più diffuso e si applica su IP. I messaggi OSPF vengono incapsulati dentro ad IP. Il campo PROTOCOL dell'IP si ricorda che è protocollo 89, e questo viene "spedito" ad OSPF.

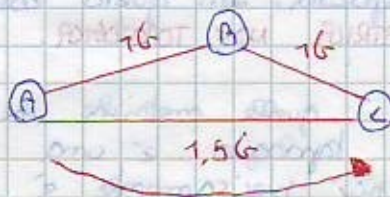
OSPF è in grado di gestire le diverse TYPE OF SERVICE dei pacchetti IP. Le metriche sono potenzialmente eterogenee, ma oggi OSPF usa una metrica del tipo 1/c. L'obiettivo OSPF tende a scegliere i link più grandi, cioè quelli che hanno più capacità. Per esempio:



questo link è inutilizzato.

Quindi OSPF concentra il traffico, e non sempre questa è una cosa buona.

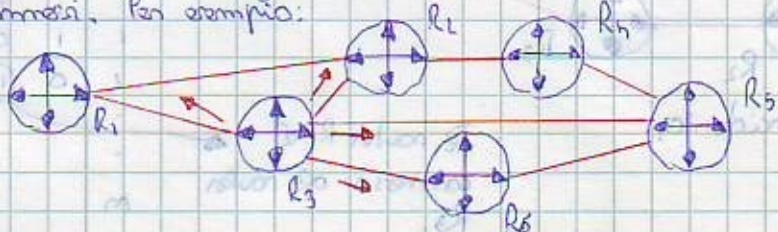
Per esempio:



$$ABC = 1 + 1 = 2$$

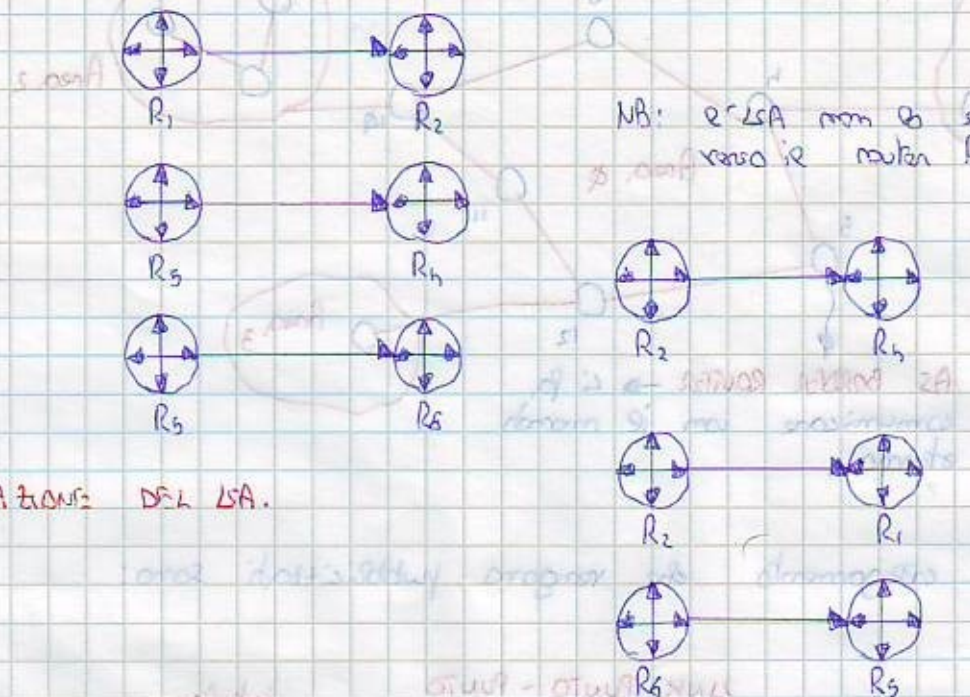
$$AC = 1 / 1,5 \approx 0,6$$

Quindi anche se la topologia è simmetrica, il traffico può essere asimmetrico. Una volta che i router hanno stabilito le connessioni vengono pubblicizzati i messaggi. La richiesta ora si intende per LINK STATUS ADVERTISEMENT (LSA). Ogni router comunica a tutti gli altri router della rete le metriche dei link a cui direttamente connessi. Per esempio:



FLOODING DI LSA.

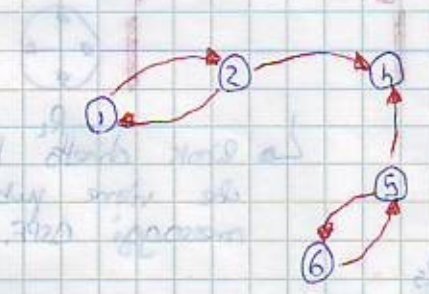
Abbiamo ipotizzato che il router R3 abbia voluto spedire LSA ai router R1, R2, R5, R6. Quindi la propagazione avviene in questo modo:



NB: un LSA non si ripropaga di nuovo verso il router R3

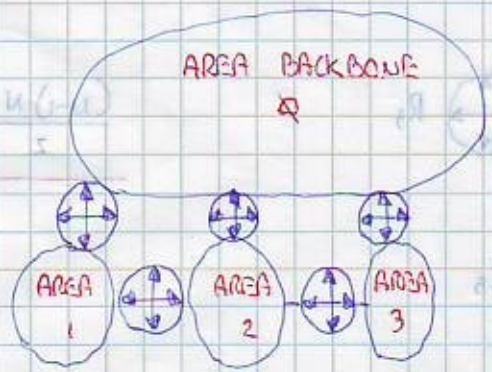
PROPAGAZIONE DEL LSA.

Quindi il diagramma temporale è:



NB: Ogni router invia i suoi messaggi
NB: I messaggi LSA duplicati non vengono propagati.

Quando il router invia un messaggio LSA è ricorrenza in realtà gli invia un ACKNOWLEDGEMENT. Ogni T_{max} si riceve il router mandando LSA. OSPF è scalabile perché è in grado di gestire la rete in modo gerarchico, cioè la rete può essere suddivisa in aree.

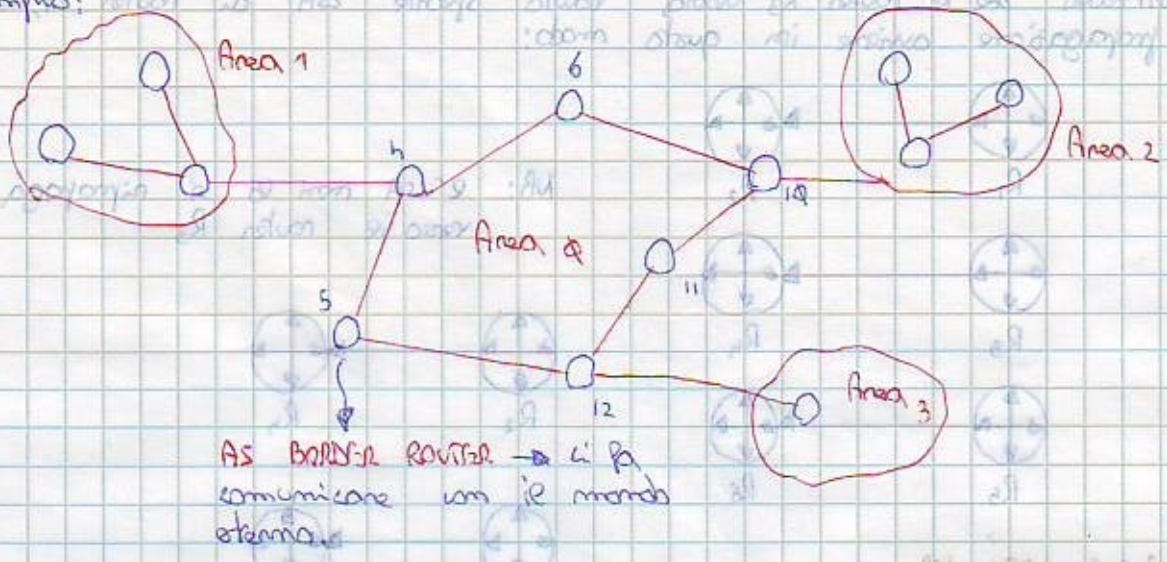


Ogni area ha un numero identificativo.

Così facendo si riduce notevolmente il numero di LSA e quindi aumenta la dimensione della rete. La messaggistica cresce in maniera più che proporzionale alla dimensione della rete. I router che fungono da interfaccia tra due aree vengono detti AREA BOUNDARY ROUTERS. Quindi il routing è differente se è INTER-AREA o INTRA-AREA. Per scegliere un percorso pulito è necessario eseguire un buon sub-routing, in modo che le sottoreti non si sovrappongano.

56

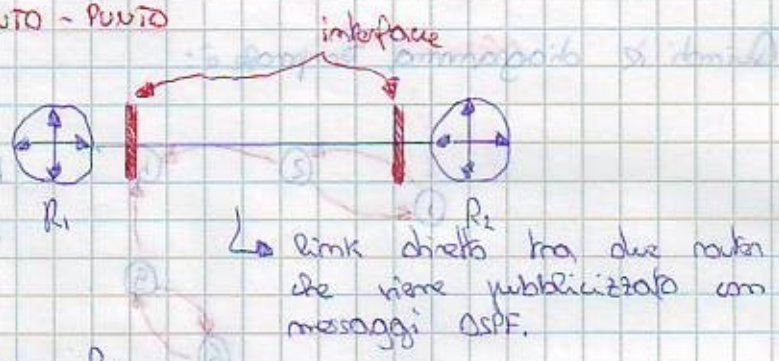
Per esempio:



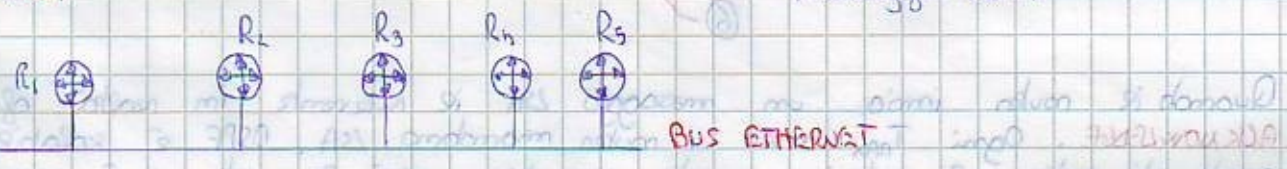
In OSPF i tipi di collegamento che vengono pubblicizzati sono:

- 1) PUNTO - PUNTO
- 2) TRANSIENTE
- 3) FOGLIA (STUB)
- 4) VIRTUALE

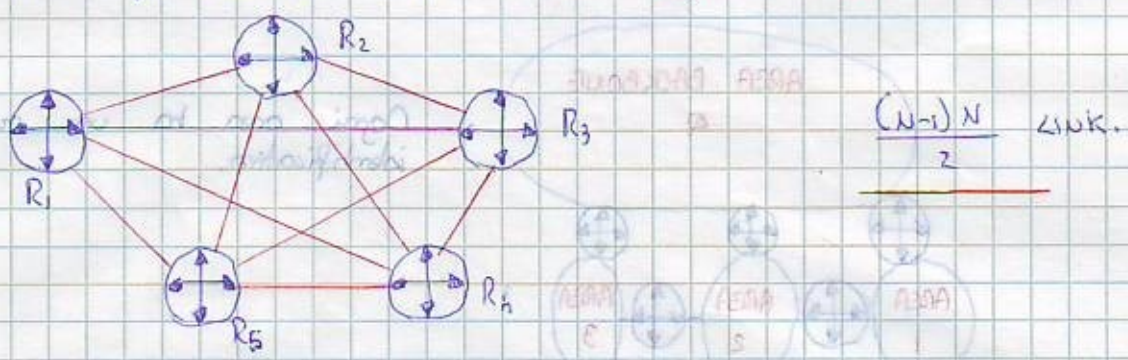
LINK PUNTO - PUNTO



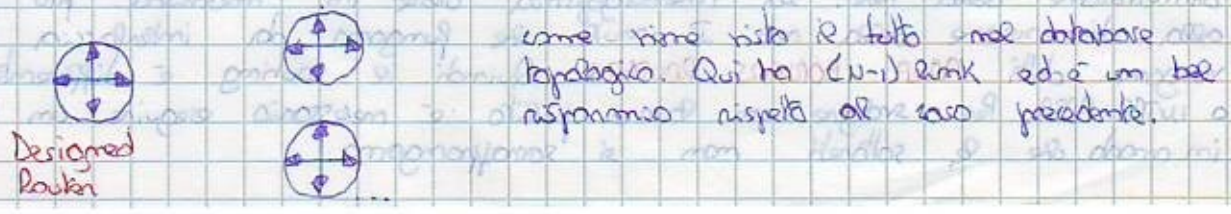
LINK TRANSIENTE:



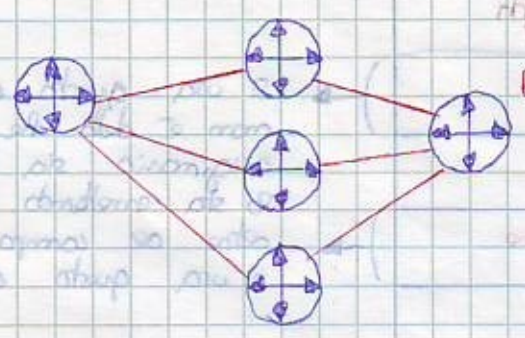
Questo link viene pubblicizzato come una maglia completa esistente fra questi router.



All'interno del bus ETHERNET si elegge, per ridurne le traffico LSA, un router particolare chiamato **Designated Router** che pubblica i link della rete. Quindi abbiamo:

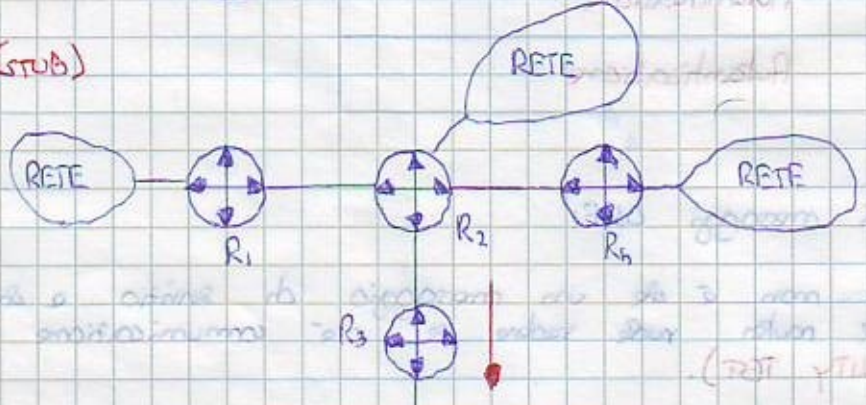


Si noti che, così facendo non si garantisce un'adatta affidabilità. Infatti se collegi le **designated router**, tutti gli altri router si isola. Per ovviare a ciò, un meccanismo manager può designare un **designated router** e un **backup designated router** in questo modo:



BACKUP In caso facendo si pubblicizzano $2(N-2)$ link. Comunque riduce l'affidabilità.

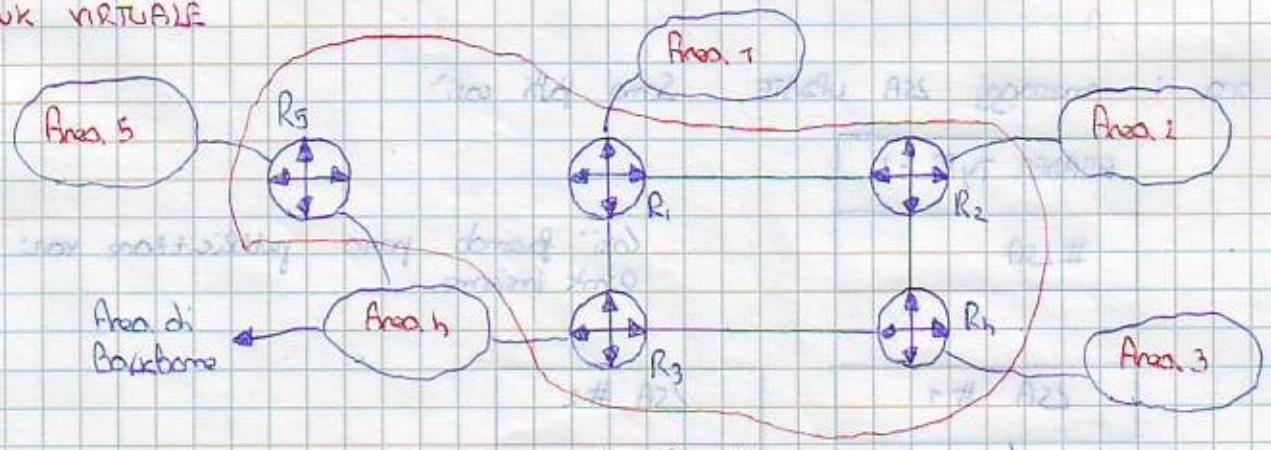
LINK FOGLIA (STUB)



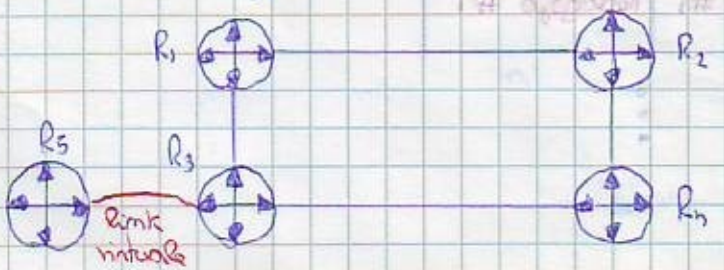
RETE rete foglia, cioè la rete non raggiungibile solo da un nodo.

Esiste un particolare messaggio per pubblicizzare questo link, perché è frequente.

LINK VIRTUALE

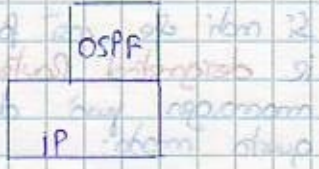


A volte conviene avere un'area di Backbone non contigua, cioè con una connettività non diretta. Un **canale virtuale** è un canale che per esempio collega R3 con R5. Logicamente si ha:



Tutto ciò per il Backbone appare ovvio.

Vediamo ora la struttura di un messaggio OSPF:



Versione	Type	Message Length
Source Router IP address		
Area ID		
CHECKSUM	Autenticazione	
Autenticazione		
Autenticazione		

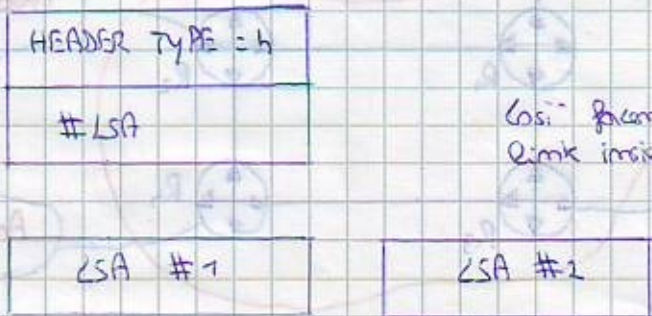
Si usa questo campo perché non è detto che il mittente originario sia il router che lo sta emettendo. Bisogna anche al campo indirizzo IP se usa questo campo.

Procedura di sicurezza

Vediamo ora i tipi di messaggi OSPF:

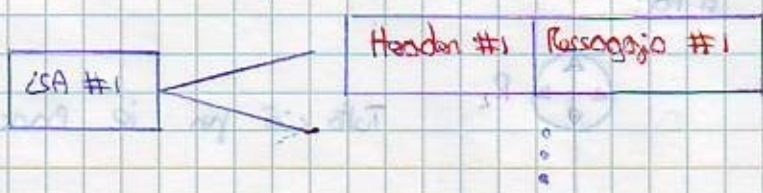
- 1) HELLO che altro non è che un messaggio di servizio e che viene usato quando il router vuole vedere se c'è comunicazione con i vicini. (REACHABILITY TEST).
- 2) DATABASE DESCRIPTION
- 3) LSA REQUEST
- 4) PACCHETTI LSA UPDATE che sono anche i messaggi principali
- 5) LSA ACKNOWLEDGE.

Analizziamo ora i messaggi LSA UPDATE. Sono fatti così:



Così facendo posso pubblicizzare vari link insieme.

I vari LSA poi a loro volta possono essere costituiti da un header e un messaggio



Un LSA oser interno è così composto: