



Modelli di comunicazione

Luca Veltri

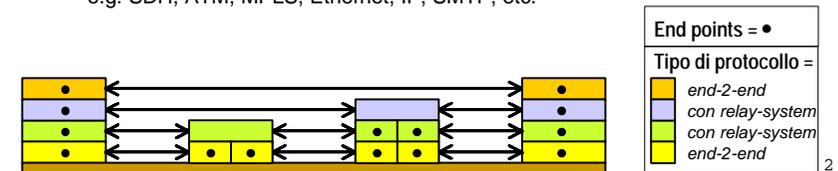
(mail.to: luca.veltri@unipr.it)

Corso di Reti di Telecomunicazioni A, a.a. 2007/2008

<http://www.tlc.unipr.it/veltri>

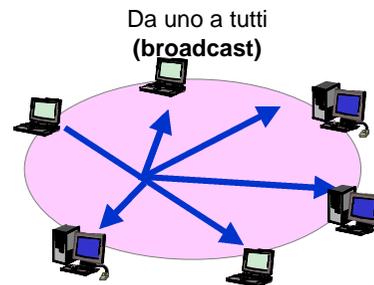
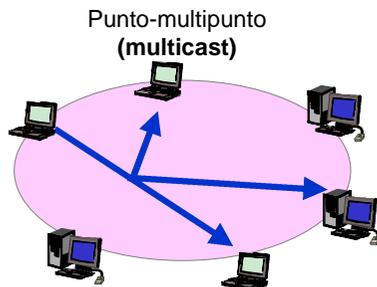
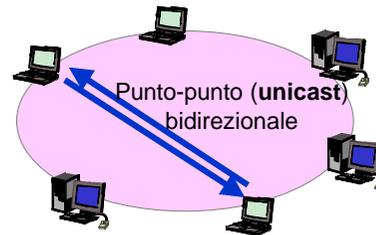
Comunicazione End-to-end o Relayed

- Comunicazione end-to-end
 - quando la comunicazione avviene direttamente tra entità sorgente e destinazione
 - non richiede l'indirizzamento di nodo ma solo del utente del servizio
- Comunicazione rilanciata (relayed)
 - quando la comunicazione avviene attraverso il rilancio di 1 o più nodi intermedi (relay system)
- Per estensione, un protocollo di comunicazione potrà essere di tipo:
 - end-to-end
 - e.g. SLIP, PPP, LLC, AAL, TCP, UDP, POP3, Telnet, etc
 - relayed
 - e.g. SDH, ATM, MPLS, Ethernet, IP, SMTP, etc.



2

Comunicazione Unicast, Multicast, Broadcast



3

Comunicazione Unicast, Multicast, Broadcast

- punto-punto (Unicast)
 - una sorgente e una destinazione
 - può essere uni o bi-direzionale
 - esempio: telefonia classica (fissa e mobile), circuiti dedicati, etc
- punto-multipunto (Multicast)
 - trasmissione contemporanea da una sorgente a più destinazioni
 - esempio: servizi telefonici avanzati (telefonata a tre), IP multicast
- diffusive (Broadcast)
 - trasmissione contemporanea da una sorgente a tutte le possibili destinazioni/terminali
 - esempio: TV, LAN..

4

Protocolli unicast, multicast, broadcast

- Alcuni protocolli supportano solo
 - la **modalità unicast**,
 - **altri sia unicast che broadcast (e.g. Ethernet..)**
 - **altri ancora tutte e tre le modalità di comunicazione (unicast, multicast e broadcast) (e.g. IP)**
- Normalmente la funzionalità di comunicazione broadcast/multicast di un protocollo è realizzata rispetto al protocollo sottostante
 - **sfruttando la funzionalità broadcast/multicast del protocollo sottostante (e.g. IP broadcast → Ethernet broadcast), oppure**
 - **mappando tale comunicazione su una molteplicità di comunicazioni unicast al livello sottostante**

5

Comunicazione CO e CL

A seconda delle modalità di gestione della comunicazione, è possibile definire due differenti modi di trasferimento dell'informazione:

- Senza connessione (CL, Connection Less)
 - **una unica fase temporale**
 - **assenza di negoziazione, indipendenza e autoconsistenza dei segmenti informativi**
 - **accordo solo tra N-utente e (N-1)-fornitore**
- Con connessione (CO, Connection Oriented)
 - **strutturazione in tre fasi temporali**
 - **accordo tra gli estremi della connessione**
 - **negoziazione dei parametri di trasferimento**
 - **una volta stabilita la connessione, il trasferimento delle UI avviene come attraverso un "tubo":
le UI vengono inviate dal sorgente ed estratte in modo ordinato dal destinatario**
 - uso di identificatori di connessione
 - è presente una relazione tra i segmenti informativi scambiati

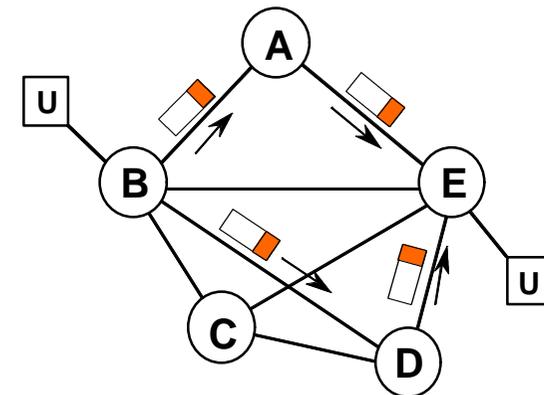
6

Comunicazione CL (a datagramma)

- Nella modalità di trasferimento a DATAGRAMMA, il trasferimento delle UI (pacchetti) avviene senza accertare preventivamente la disponibilità dell'utente destinatario e/o delle risorse di rete all'effettuazione del trasferimento
- Non esistono le fasi di instaurazione e di abbattimento di una chiamata ed ogni pacchetto o datagramma (UI) è gestito dalla rete indipendentemente dagli altri, anche se fanno parte della stessa comunicazione
- Se sono presenti nodi di commutazione, questi operano la funzione di instradamento solo sulla base dei singoli pacchetti (UI)
- Ogni pacchetto è completamente indipendente
- E' possibile che i pacchetti vengano consegnati fuori sequenza

7

Esempio di comunicazione CL (a datagramma)



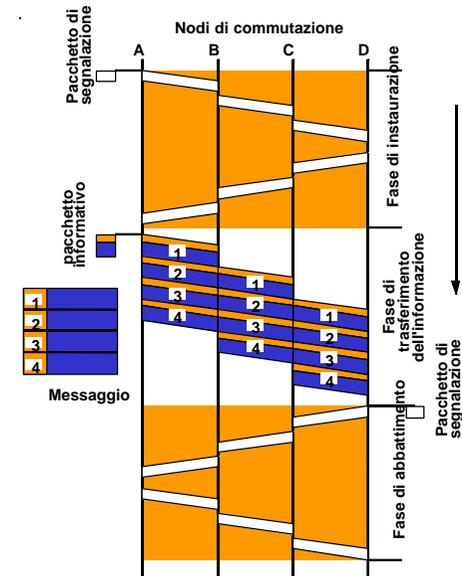
8

Comunicazione CO

- Nella modalità di trasferimento con connessione, il servizio è articolato in tre fasi:
 - fase di instaurazione della connessione
 - fase di trasferimento informativo
 - fase di abbattimento della connessione
- Durante la fase di instaurazione normalmente viene:
 - a) accertata la possibilità di instaurare la connessione (controllo di accettazione di chiamata) ed eventualmente effettuata un'assegnazione logica delle risorse necessarie
 - b) assegnati opportuni identificatori della chiamata che saranno trasportati da tutte le UI appartenenti alla chiamata stessa (connessione)
 - c) nel caso di attraversamento di più nodi, viene determinato il cammino che i pacchetti seguiranno in rete (funzione di istradamento)
- Durante la fase di abbattimento vengono rilasciate le risorse precedentemente allocate

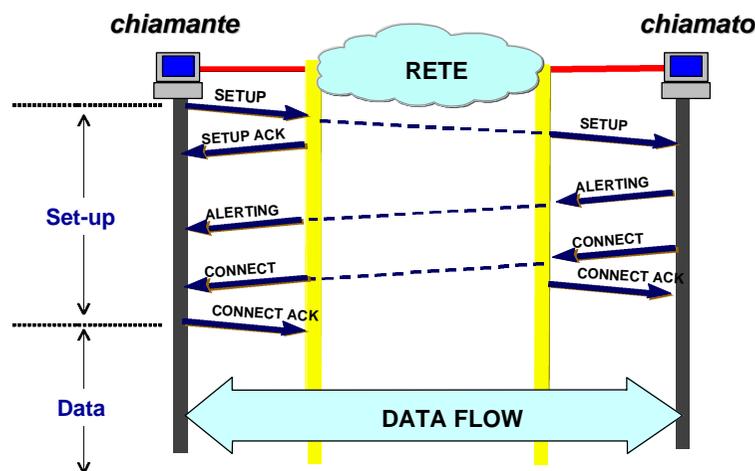
9

Esempio di comunicazione CO



10

Esempio CO: Chiamata Q.931



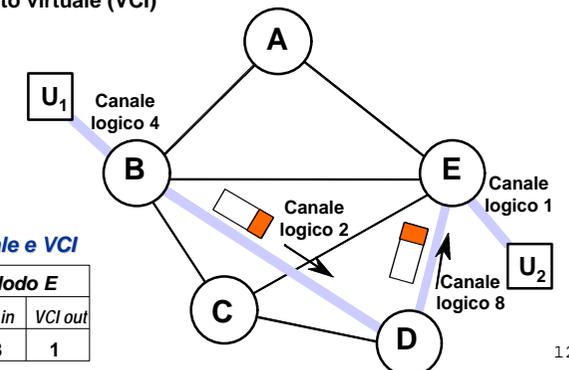
11

Comunicazione CO

- Durante la fase di trasferimento le UI vengono trattate dai terminali e, eventualmente, dai nodi di rete sulla base della appartenenza delle stesse UI ad una specifica connessione (precedentemente instaurata)
 - a tal fine vengono considerati degli identificatori logici associati alle singole UI
 - tali identificatori vengono a volte chiamati numero di canale logico, o identificatore di circuito virtuale (VCI)

Esempio di corrispondenza tra circuito virtuale e VCI

Nodo B		Nodo D		Nodo E	
VCI in	VCI out	VCI in	VCI out	VCI in	VCI out
4	2	2	8	8	1



12

Comunicazione CO

- In genere, durante la fase di trasferimento:
 - **le UI preservano il loro ordinamento (sequenza)**
 - **rispetto a comunicazioni in modalità CL, la probabilità di perdita può essere minore (aumentando l'integrità informativa)**
 - grazie ad eventuali meccanismi di recupero di errore
 - grazie ad eventuali meccanismi di riservazione/gestione delle risorse di rete
 - **rispetto a comunicazioni in modalità CL, il ritardo end-to-end può essere minore (aumentando la trasparenza temporale)**
 - nel caso lo strato CO esegua la funzione di commutazione (ovvero prevede nodi intermedi che operano in modalità CO)

13

Comunicazione Affidabile e Non affidabile

- Una comunicazione viene detta "*affidabile*" se garantisce l'integrità informativa dei dati trasferiti
 - **in particolare deve preservare sia l'integrità delle singole UI e sia il loro ordinamento**
- Nel caso contrario, la comunicazione è detta "*non affidabile*"
- Se un protocollo implementa funzionalità che permettono di garantire l'affidabilità della comunicazione, tale protocollo è detto *affidabile*
 - e.g. **TCP**
- Nel caso contrario il protocollo è detto *inaffidabile*
 - e.g. **Ethernet, IP, UDP**
- Per estensione, spesso un protocollo viene detto *affidabile* anche quando non implementa direttamente tali funzionalità, ma si poggia comunque su un protocollo sottostante di tipo *affidabile*
 - e.g. **SMTP, HTTP, etc.**

14

Comunicazione Real-time e Non real-time

- Una comunicazione è detta di tipo real-time se nel trasferimento preserva (o deve preservare) la trasparenza temporale
- La trasparenza temporale può essere garantita in termini di:
 - **mantenimento/ricostruzione in ricezione della stessa cadenza temporale delle UI inviate**
 - **basso ritardo end-to-end**
- Il primo punto può essere garantito attraverso opportune funzioni di riequalizzazione dei ritardi
 - **soprattutto in caso di bassa variabilità (jitter) dei ritardi end-to-end**
- Il secondo punto richiede specifiche caratteristiche di trasferimento nella rete attraversata
 - **basso ritardo massimo end-to-end**

15

Comunicazione Message o Stream oriented

- A seconda della modalità con cui i dati vengono inviati e ricevuti, si possono distinguere due modalità di comunicazione:
 - **Message oriented**
 - vengono inviati e ricevuti dallo strato superiore specifici blocchi di dati
 - in ricezione viene garantita l'appartenenza dei dati ad un specifico blocco (messaggio)
 - e.g. PPP, Ethernet, IP, UDP, HTTP, SMTP, etc.
 - **Stream oriented**
 - i dati dello strato superiore vengono gestiti come un flusso continuo (non necessariamente temporalmente continuo) di bit o byte
 - in ricezione viene garantita la medesima sequenza dei dati e riconsegnati allo strato superiore in ordine
 - i dati non vengono strutturati e delimitati in blocchi, ma al contrario vengono consegnati allo strato superiore come un unico flusso
 - e.g. PCM, SDH, in generale la maggior parte degli strati PH, il TCP, Telnet/SSH

16

Comunicazione Message o Stream oriented

