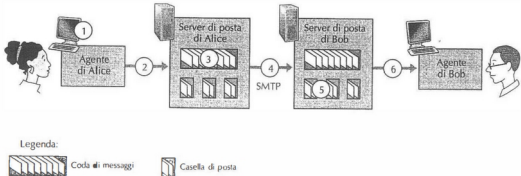


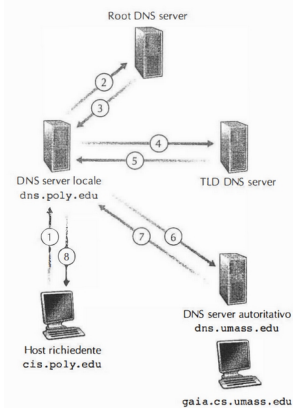
**Tipi di trasferimento di pacchetti sulla rete**  
**Circuit switching:** Imposta un percorso dedicato tra mittente e destinatario prima della trasmissione dei dati. **FDM** divide il canale in bande di frequenza, assegnando una a ciascun circuito, tipicamente da 4 kHz (per la trasmissione telefonica tradizionale). **TDM** suddivide il tempo in slot, assegnando ciascuno a un circuito.  
**Packet switching:** I dati sono suddivisi in pacchetti che viaggiano separatamente sulla rete. Le "**Virtual Circuit Networks**" creano un percorso virtuale dedicato per i pacchetti di dati. Le "**Datagram Networks**" trattano ogni pacchetto individualmente senza stabilire un percorso dedicato.  
**Codifiche:**  
**NRZ** (Non-Return-to-Zero): un livello costante di tensione rappresenta un bit "1" e un altro livello costante rappresenta un bit "0". Non c'è transizione di tensione per rappresentare il bit "0".  
**Manchester:** ogni bit viene rappresentato da una transizione di tensione a metà intervallo di bit. Una transizione ascendente rappresenta un bit "1" mentre una transizione discendente rappresenta un bit "0".  
**Connessioni a internet:**  
**DSL** (Digital Subscriber Line): Una tecnologia di connessione a Internet che sfrutta il filo telefonico esistente per trasmettere dati digitali ad alta velocità. DSL utilizza una larghezza di banda separata dalla voce sulla stessa linea telefonica. download max 10 Mbps e upload max 1Mbps. Usa **Dial-up Modem**: dispositivo che utilizza la modulazione e la demodulazione dei segnali audio per trasmettere dati digitali su una linea analogica.  
**ADSL** (Asymmetric Digital Subscriber Line): un'implementazione della DSL con più download. download max 20 Mbps e upload max 1Mbps.  
**Active Optical Networks:** i segnali luminosi trasmessi lungo la fibra ottica sono amplificati o convertiti elettricamente a intervalli regolari. ciò significa che ci sono dispositivi attivi, come amplificatori ottici o ripetitori, che aumentano la potenza del segnale lungo il percorso. velocità di trasmissione di dati fino a diversi gigabit al secondo.  
**Passive Optical Networks:** non ci sono componenti attivi quindi no amplificazione lungo il percorso.  
**Ritardi:**  
**Di elaborazione:** tempo necessario per esaminare l'intestazione del pacchetto e decidere la destinazione. Include il controllo degli errori di bit. Ordine dei microsecondi o inferiori.  
**Di accodamento:** tempo di attesa per la trasmissione su un collegamento. Dipende dal numero di pacchetti già in coda per la trasmissione sullo stesso collegamento. Ordine dei microsecondi a millisecondi.  
**Di trasmissione:** tempo necessario per trasmettere tutti i bit del pacchetto sul collegamento. Calcolato come *lunghezza del pacchetto / velocità di trasmissione*. Ordine dei microsecondi a millisecondi.  
**Di propagazione:** tempo impiegato per il segnale per viaggiare dal router sorgente a quello di destinazione. Calcolato come *distanza tra i router / velocità di propagazione nel mezzo*. Dipende dal mezzo trasmissivo (fibra ottica, rame, ecc.).  
**Intensità di traffico**  
 $a$  = velocità media di arrivo dei pacchetti nella coda (n pacchetti al secondo)  
 $R$  = velocità di trasmissione, ossia la velocità alla quale i bit vengono trasmessi in uscita dalla coda (in bit al secondo)  
 $L$  = numero di bit che costituisce un pacchetto  
 $L \cdot a$  = velocità media di arrivo dei bit in coda (in bit/s)  
 $(L \cdot a) / R$  = **intensità di traffico**  
 $(L \cdot a) / R > 1 \rightarrow$  la velocità media di arrivo dei bit nella coda supera la velocità alla quale i bit vengono ritrasmessi in uscita da essa. In questa situazione sfortunata, la coda tenderà a crescere senza limiti e il ritardo di coda tenderà all'infinito.  
 $(L \cdot a) / R < 1 \rightarrow$  la natura del traffico in arrivo influisce sul ritardo di coda:  
- i pacchetti arrivano a **cadenza periodica**  $\rightarrow$  ciascun pacchetto troverà una coda vuota e non ci saranno ritardi di accodamento.  
- i pacchetti a **raffiche** periodiche, si possono verificare dei significativi ritardi medi di accodamento  
**Throughput end-to-end istantaneo** = la velocità di trasferimento dati in un momento preciso. misurato come la quantità di dati ricevuti correttamente dalla destinazione nell'arco di un piccolo intervallo di tempo (ad esempio, millisecondi o secondi).  
**Throughput end-to-end medio** = il min(tutte le velocità di trasmissione), cioè la velocità di trasmissione del collegamento che fa da collo di bottiglia (bottleneck).

**Application-layer protocols:** quali tipi di messaggi (ex. Richiesta?, Risposta?), sintassi, semantica, regole di comunicazione  
**Protocolli di pubblico dominio:** definiti in documenti RFC (Request for Comments). progettati per essere interoperabili, ovvero consentono a software diversi di comunicare tra loro. Ex. HTTP, SMTP ecc..  
**Protocolli proprietari:** sono specifici di un software. non sono interoperabili con software di altri fornitori. Ex. KaZaA  
**Porta:** il numero di porta identifica il processo specifico su un dispositivo. 16 bit address (0-65535). Common port numbers: FTP 20, TELNET 23, SMTP 25, HTTP 80, POP3 110. Memorizzate in *transport protocol header* (TCP or UDP). IP address in *IP packet header*.  
**HTTP** (Hypertext Transfer Protocol): protocollo per la comunicazione tra browser web e server web. le regole per la richiesta e la ricezione di pagine web e altri contenuti web. Modalità:  
**non persistente** (HTTP 1.0): viene aperta una nuova connessione per ogni richiesta-risposta. (inefficiente, ogni volta che si apre una connessione TCP, si pagano 2RTT)  
**persistente:** singola connessione.  
- **senza pipelining:** aspetta una risposta completa prima di inviare una nuova richiesta. Ogni oggetto web richiede un RTT (Round Trip Time).  
- **con pipelining** (HTTP 1.1): invia nuove richieste senza aspettare risposta completa, minimizzando i RTT complessivi. (efficiente).  
**HTTP 2.0:** streaming multiplexing (flusso bidirezionale di informazioni), controllo flusso, priorizzazione, compressione degli header, HTTPS (sicurezza e crittografia).  
**Proxy:** sono degli instradatori di messaggi di livello applicativo. intermediari che si comportano sia come server sia come client.  
**Regular Proxy:** inoltra semplicemente le richieste e le risposte tra client e server.  
**Caching Proxy:** memorizza le risposte a richieste frequenti per un accesso più rapido.  
**Forwarding Proxy:** si posiziona tra un client e internet, filtrando e gestendo il traffico.  
**Reverse Proxy:** si posiziona di fronte a un server web, distribuendo le richieste su più server web e fornendo servizi di sicurezza e caching.  
**Interception Proxy:** intercetta e modifica il traffico di rete per scopi di sicurezza o monitoraggio.  
**Cookie:** un altro modo per riconoscere richieste successive di uno stesso utente. Ogni utente è identificato da un numero di cookie. Il numero di cookie inviato dal server viene memorizzato in un opportuno file.  
**FTP** (File Transfer Protocol): protocollo per il trasferimento di file tra un client e un server su una rete TCP/IP. Numero di porta predefinito: 21. Definito nel documento RFC 958. Supporta autenticazione utente e password. Consente trasferire file in modalità binaria o ASCII. Funzionalità per navigare tra le directory e manipolare i file sul server remoto. Server TCP deve tener traccia dello stato (ex. directoy) dell'utente connesso. Funziona su connessioni TCP separate per il controllo (*formato ASCII 7 bit*) e i dati.  
**Posta Elettronica:** componenti user agent, mail server, protocollo SMTP.



**SMTP** (Simple Mail Transfer Protocol): usa codifica ASCII per tutti i tipi di messaggi (penalizzante per allegati) (diverso da HTTP dove multimedia trasmessi in binario perché no vincolo). Non usa server intermedi.  
**POP3** (Post Office Protocol): fasi autorizzazione, transizione (modalità scaricare e cancellare o scaricare e rimanere), aggiornamento.  
**IMAP** (Internet Message Access Protocol): + funzioni rispetto a pop3. Creare cartelle remote (default in INBOX). Cercare messaggi in base a criteri. Scaricare solo parti di mail.  
**HTTP per Mail:** accesso a posta tramite web browser. Prima Hotmail negli anni 90'.

**DNS** (Domain Name System): traduce hostname in indirizzi IP. DNS sono macchine UNIX che eseguono il software BIND (Berkeley Internet Name Domain). Porta default: 53. Altri servizi: Host Aliasing, Mail Server Aliasing, Load Distribution. Ad host molto trafficati corrispondono un insieme di indirizzi IP, ogni volta cambia ordine della lista così host diversi accedono a server diversi. Il database con la corrispondenza ip  $\leftrightarrow$  hostname è distribuito e gerarchico. Gerarchia:  
**Root server:** restituisce uno o più indirizzi IP relativi al server TLD per uno specifico dominio. In Internet esistono 13 (in realtà cluster di server replicati) root server (etichettati da A a M). La maggior parte situati in Nord America.  
**Top-level domain (TLD) server:** relativi ai domini di primo livello e ai vari paesi. La Verisign Global Registry Services gestisce i TLD server per il dominio com  
**Server autoritativi:** coloro che hanno hostname accessibili tramite internet ne devono avere uno o pagare un fornitore per ospitare i propri record.  
**DNS Server locali** (o Default Name Server): ogni uni, azienda ne possiede uno. Si occupano di inoltrare la richiesta dell'utente alla gerarchia vista sopra. Ogni dns server locale può gestire più richieste da host diversi.  
Nel caso domestico l'ISP quando gli arriva una richiesta la smista tra i suoi dns server locali assegnandone uno. Ogni dns server locale ha una cache con il record hostname e relativo IP. Ogni record ha un periodo di permanenza di 2 giorni in genere. Types: A = hostname standard, NS = DNS autoritativo del dominio, CNAME = hostaliasing, MX = mail server aliasing (per far avere nomi semplici ai mail server).  
**Messaggi DNS:** Identificazione = viene copiato nei messaggi di risposta a una richiesta (x per far corrispondere query e risposte), Flag = bit domanda o risposta, (risposta) bit dns autoritativo o no?, (richiesta) fai ricorsione o no?, (risposta) ricorsione disponibile?, Sezione domande = nome richiesto e tipo di domanda (a, mx ecc.), Sezione risposte, Sezione autoritativa, Sezione aggiuntiva.



**P2P** (Peer-to-peer):

**BitTorrent:**

**Audio:** per catturare dell'audio da inviare viene campionato un segnale audio analogico ad una rate costante (ex. telefono 8000 campioni/sec.) Ogni campione viene quantizzato (cioè arrotondato). Viene quantizzato perché deve passare da analogico a digitale. Disporremo di un tot di bit per campione. Rappresentato con un grafico con andamento del *segnale nel tempo* o grafico dello *spettro del segnale* (diverse componenti in frequenza) (notiamo che le componenti di frequenza più significative della voce si trovano tra i 300Hz e i 3400Hz). Tipi di quantizzazione:

**Quantizzazione uniforme:** errore di quantizzazione è fisso. Se q è il passo di quantizzazione, allora l'errore di quantizzazione sarà minore di q/2.

**Quantizzazione NON uniforme** (o non lineare). Valori grandi possono supportare errori maggiori.

**Codifica:** processo di cattura del segnale audio analogico e della trasformazione in digitale. Fasi: filtraggio, campionamento, codifica e compressione.

**Decodifica:** fasi sono espansione, decodifica e ricostruzione

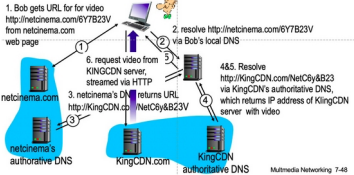
**Streaming stored video:** necessità di un *buffer lato client* per soddisfare i requisiti di riproduzione continua e un *ritardo di riproduzione* per riempire tale buffer. Motivo: ridurre i ritardi di rete sono variabili (jitter). tasso di riempimento ( $> o <$ ) tasso di riproduzione. Alternative:

**UDP+RTP** = Il server di streaming invia i dati alla velocità appropriata per il client. Spesso, la velocità di invio è uguale alla velocità di codifica ed è costante nel tempo. Velocità di trasmissione può essere indifferente ai livelli di congestione della rete, il che significa che non si adatta dinamicamente alla congestione. ritardo di riproduzione 2-5 secondi. UDP potrebbe incontrare difficoltà nel passare attraverso i firewall poiché è un protocollo senza connessione e non prevede il controllo di flusso.

**TCP+HTTP** = file recuperati tramite richieste HTTP GET (GET) (specificando l'intervallo specifico di byte che il client desidera attualmente recuperare dal video desiderato), inviati alla massima velocità possibile sotto il protocollo TCP (Transmission Control Protocol). Il tasso di riempimento varia a causa del controllo della congestione TCP e delle ritrasmissioni (consegna in ordine). Si ha un ritardo di riproduzione più elevato per garantire una consegna uniforme tramite il protocollo TCP. HTTP/TCP passa più facilmente attraverso i firewall.

**DASH** (Dynamic Adaptive Streaming over HTTP): Streaming Dinamico Adattivo su HTTP. Funziona con qualsiasi server web HTTP standard. Permette streaming di alta qualità anche con connessioni internet lente, NO interruzioni e NO buffering. Server = Divide video in segmenti (chunk), memorizza ogni chunk codificato a diverse velocità (bitrate), crea un file manifest che contiene gli indirizzi (URL) per scaricare i diversi chunk. Client = Misura periodicamente la banda, consulta il file manifest, richiede il chunk con la massima qualità video sostenibile dalla connessione in quel momento.

**CDN** (Content Delivery Network): li usa netflix. vantaggi = riduzione della latenza (CDN distribuiti meglio geograficamente), incremento del throughput (CDN grande capacità di banda e sono distribuiti), migliora disponibilità contenuto (CDN replicano su più server).



**RTSP** (Real-Time Streaming Protocol): telecomando per un media server. Funziona su una porta diversa (default: 554) dal flusso multimediale. Il lettore multimediale del client analizza la *presentation description file* ricevuto dalla richiesta HTTP GET e utilizza RTSP per controllare il server di streaming inviando istruzioni varie.

**VOIP** (Voice over IP): tecnologia che rende possibile effettuare una chiamata sfruttando una connessione internet. Richiede uno scambio di informazioni prima della chiamata. Rileva il silenzio e genera pacchetti solo durante la conversazione (pacchetti da 20ms a 8Kbps). Problemi: **ritardi** = meno di 150ms buona qualità, oltre 400ms scarsa qualità & **perdita pacchetti** = tollerabile tra l'1% e il 10%.

**Tipi di ritardo di playout:** **fisso** (q) = un q alto significa scartare meno pacchetti a causa del ritardo, ma peggiora la qualità dell'esperienza perché c'è più ritardo tra il momento in cui la persona parla e il momento in cui senti la sua voce. un q basso significa una minore latenza e un'esperienza più interattiva, ma aumenta la possibilità di perdere pacchetti e quindi la qualità della voce ne risente. **Adattivo** = attraverso la tecnica chiamata *media mobile ponderata esponenzialmente* (EWMA).

**Recupero perdita pacchetti con Correzione d'errore in avanti (FEC)** = Invece di aspettare un ACK/NAK (segnale di ricezione positiva/negativa) che richiede un tempo di andata e ritorno (RTT) aggiuntivo, il mittente può inviare informazioni ridondanti insieme ai dati originali. Questo permette al ricevitore di ricostruire i pacchetti persi senza bisogno di una ritrasmissione. Tipi di FEC: **Piggyback a basso bitrate** = informazione ridondante nel pacchetto successivo, **interleaving** = informazione ridondante in pacchetti diversi precedenti e successivi.

**Skype:** utilizza un protocollo proprietario a livello applicativo (scoperto tramite reverse engineering). Si basa su una struttura peer-to-peer (P2P). Funzionamento = il client si unisce alla rete contattando un supernodo (SN) e connettendosi con TCP; il client esegue il login utilizzando nome utente e password su un server di login centralizzato di Skype; il client ottiene l'indirizzo IP del destinatario della chiamata dal supernodo tramite la rete overlay (rete con database distribuito di addressIP e porte degli skype clients) o dall'elenco contatti del client, se già presente; avvia la chiamata stabilendo una connessione UDP con il destinatario (se firewall presenti TCP).

**RTCP** (Real-Time Protocol): viene eseguito "al di sopra" del protocollo UDP. Permette l'interoperabilità = se due applicazioni VoIP (Voice over IP) utilizzano RTP, possono funzionare insieme, essendo uno standard comune. Hader = Tipo di payload (7 bit) → indica il tipo di codifica attualmente utilizzata. Se chi invia modifica la codifica durante la chiamata, lo comunica al ricevente tramite questo campo; Numero sequenza (16 bit) → aumenta di uno per ogni pacchetto RTP inviato per rilevare perdita di pacchetti; Timestamp (32 bit) → indica l'istante di campionamento del primo byte in questo pacchetto di dati RTP. L'orologio dei timestamp continua ad aumentare a ritmo costante anche quando la sorgente è inattiva; Campo SSRC (32 bit) → identifica la sorgente del flusso RTP. Ogni flusso in una sessione RTP ha un SSRC distinto.

**RTCP** (Real-Time Control Protocol): lavora insieme a RTP e fornisce informazioni e controlla le prestazioni di RTP. Scambia di statistiche tra partecipanti di una sessione RTP (numero di pacchetti inviati, numero di pacchetti persi, variazione di ritardo tra l'arrivo di pacchetti consecutivi (jitter)). Ogni partecipante riduce la quantità di pacchetti RTCP che invia man mano che il numero di partecipanti alla conferenza aumenta. I pacchetti RTP e RTCP hanno due numeri di porta diversi. Cerca di limitare il suo traffico al 5% della larghezza di banda utilizzata dalla sessione. Permette sincro audio-video = il pacchetto RTCP Sender Report contiene, per ogni flusso RTP associato, il timestamp dell'ultimo pacchetto RTP inviato e l'orario reale (wall-clock time) in cui è stato creato, così, il ricevitore, grazie a questa associazione tra timestamp e orario reale, riesce a sincronizzare la riproduzione dell'audio e del video.

**SIP** (Session Initiation Protocol): usato per avviare sessioni su internet. Si occupa di stabilire la connessione e individuare l'indirizzo IP corrente del chiamato. Permette di aggiungere nuovi flussi multimediali durante una conversazione (video to audio), modificare il formato di codifica, aggiunta di altri partecipanti, trasferimento e messa in attesa chiamate. Risolve problema trovare il numero di IP address di una persona quando si ha solo il suo indirizzo: **Registrar SIP** = mantiene una directory di utenti SIP registrati (indirizzo SIP e indirizzo IP corrente) perché quando avvio il client invio subito un messaggio di registrazione al mio Registrar SIP. **Proxy SIP** = agisce come intermediario per le chiamate SIP, simile a un server DNS locale per la risoluzione dei nomi in indirizzi IP.

**Trasporto TCP**

**UDP**