**Struktur af internettet**

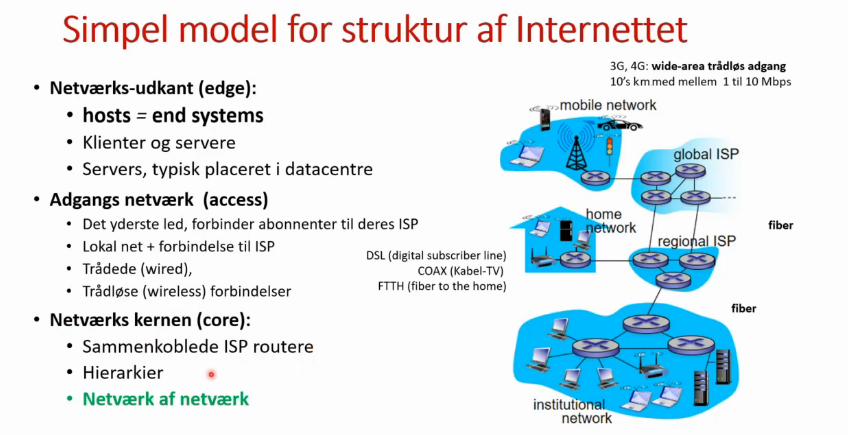
*Hosts* = end systems = PC, servers, laptops, telefoner, cyber-physical systems osv.

*Links* = faste forbindelser (kobber, fiber) eller trådløse forbindelser (wifi).

*Packet Switches* = en enhed der videresender data-pakker i nettet.

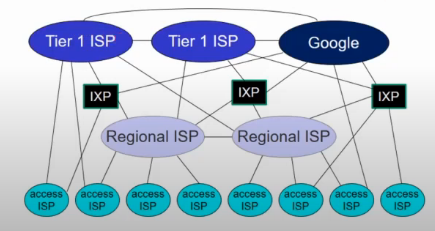
*Protokol* = regler der styrer afsendelse og modtagelse af beskeder (IP, UDP, TCP osv).

*RFC* = Internet standarder.

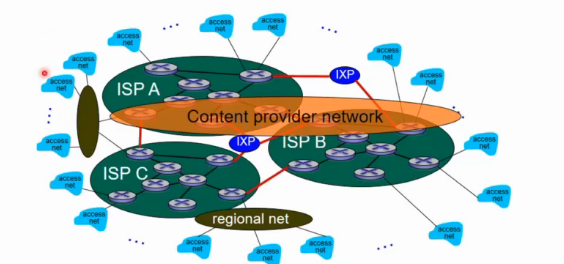


*Adgangs netværk* består (typisk) af lokale netværk brugt af størrer institutioner som typisk er forbundet med *Ethernet* (2 lag) switch.

*Netværk af netværk* består af et hirarki af ISP’er. På de laveste lag er der ”*access ISP*’er” som udbyder internetforbindelse til brugere. Disse Access ISP’er kan være forbundet med en *regional ISP* eller en *national ISP*. Access ISP’er, Regionale og nationale er forbundet til ”*Tier 1 ISP’er*” som er ”de store spillere” og er globale (f.eks Google). Alle ISP’er kan kommunikere direkte med hinanden **eller** med IXP (Internet exchange points) som er en fysisk bygning hvor man udveksler netværks-trafik(f.eks Danish Internet Exchange). Dette betyder der ikke er ét hiraki som er arbitrært. Et eksempel kan ses her:



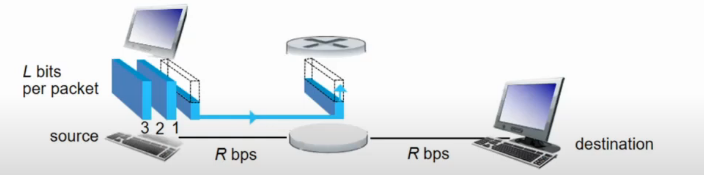
*Interforbundne netværk* består af forbundne *server-farme* som er tættere på end-hosts. Et eksempel kunne være en streaming tjeneste som har flere server farme tættere på brugeren. F.eks Google/Youtube. Figur:



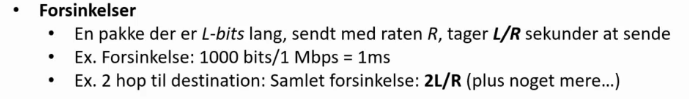
Formålet er at fordele trafik, så der ikke kommer en flaskehals gennem en global server.

**Packet switching**

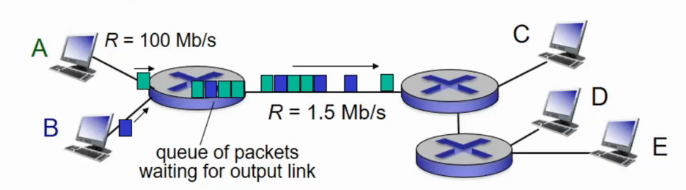
Packet switching består af at hosts deler data op I mindre dele (packets) som bliver sent til den næste router på stien (f.eks fra en host, til en router, til en anden router, til modtager host). Hver packet sendes med den fulde transmissionsrate af linket. Routeren skal modtage hele pakken og gemmes I dens hukkomelse før den bliver vidresendt (store and forward). Eksempel:

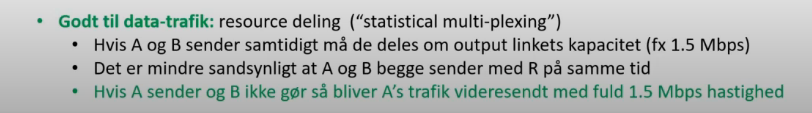


En pakke oplever forsinkelser når den er sendt igennem internettet. En pakke består af **L** *BITS* (8 bit er 1 byte) og den bliver sendt med en rate på **R** (mbps, gbps…osv).



*Pakke-kø* og *pakke-tab* kan ske I et netværk. En *pakke-kø* kan opstå ved at ankomst raten overstiger transmissionsraten over en kort periode, her bliver pakken gemt I en buffer med first-in-first-out princippet (en kø struktur). *Pakke-tab* sker hvis bufferen løber tør for plads, så routeren bliver nødtil at droppe pakken. Forstoppelse (*congestion*) giver mulighed for pakke-tab (eksempel næste side).

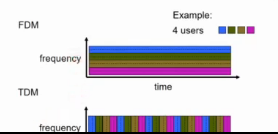




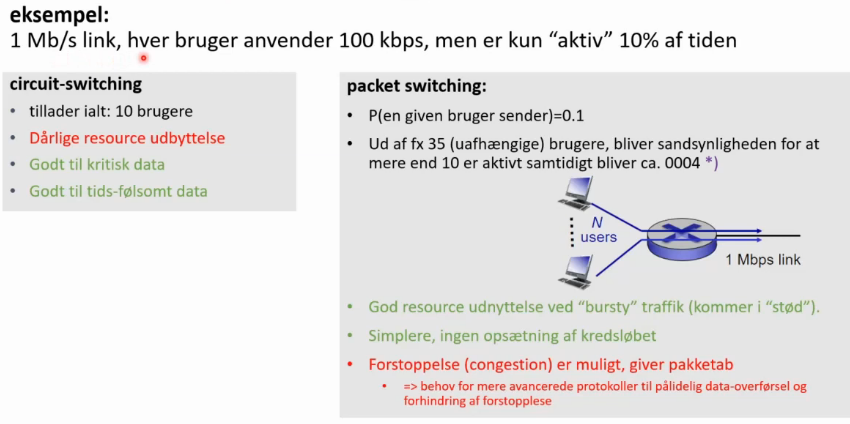
* Godt til data-trafik som optræder I stød.

**Circuit Switching**

Circuit switching er en alternativ løsning hvor der oprettes en forbindelse mellem hosts og der reserveres en *fast* transmissionsrate igennem hele netværket. Måden man deler linkets kapacitet op, er ved FDM eller TDM. Eksempel:



Packet switching vs circuit switching:



**Forsinkelse**

Kilder til forsinkelse af routeren består af behandlingtid, kø-tid, transmissionstid og udbredings delay (propagation delay). Man bestemmer forsinkelsen med følgende formel:



dproc = data-behandlingstid, checksum fejl, bestemme output link. Dproc er typisk < 1 milisekund.

dqueue = Ventetiden i køen. Tallet afhænger af forstoppelses-niveauet (congestion level).

dtrans = L/R – Længden af pakken(i BITS) divideret med transmissionsraten (bits per second).

dprop = d/s – Distancen af det fysiske link divideret med udbredselshastigheden s = 2 \* 10^8 m/s.

(traceroute.org kan bruges til at måle rigtige forsinkelser på internettet)

**Trafik intensitet**

Trafik intensitet er defineret som forholdet imellem indkommende og udkommende trafik rate. Man bestemmer trafik intensiteten med følgende formel:

 - (L \* A) / R

R – Transmissions rate (bps)

L – Længden af pakken i bits

A – Gennemsnitlig pakke ankomst (pakker per sekund).

Hvis trafik intensiteten overstiger 1 vil der akkumuleres pakker og der vil komme pakke-tab.

Hvis trafik intensiteten nærmer sig 0 er der ”klar bane” og alt er godt.

Hvis trafik intensiteten nærmer sig 1 vil hver pakke have lang ventetid.

**Throughput (flaske halse)**

Throughput er den opnåelige transmissionsrate i en forbindelse mellem 2 end-hosts. Throughput bestemmes af linket med mindst rate:



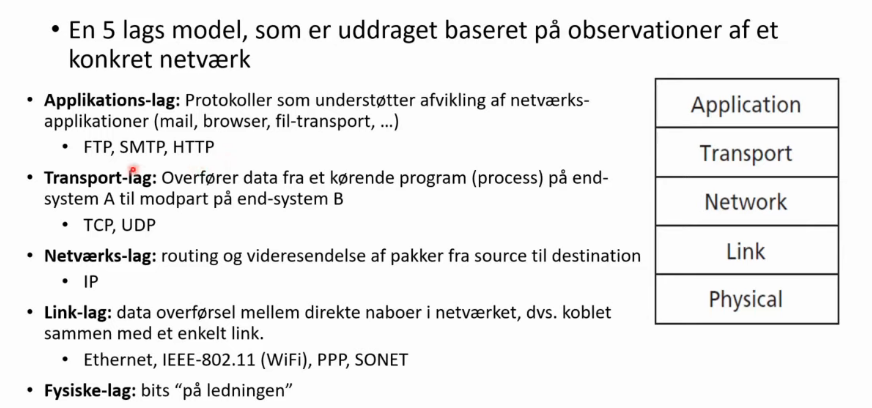
Rc = Transmissions raten af clienten (eller 1 end-host)

Rs = Transmissionsraten af serveren (eller en anden end-host)

R/n – R = Transmissionsraten og n er antallet af forbindelser. Dette fordeler transmissions raten fair.

Typisk kommer der en flaskehals ved clienten eller serveren.

**Internet protokol stakken**



**Header-payload modellen (Encapsulation)**

