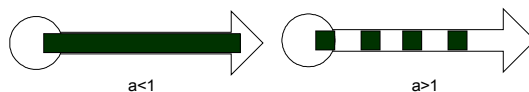


# 11. Architektura komunikačního systému

## Model OSI a IEEE

<b>Linková vrstva</b>	- zajišťuje data proti chybám při přenosu
<b>Síťová vrstva</b>	- definuje způsob pohybů paketů v síti
<b>Transportní vrstva</b> komunikačních spojů	- umožňuje komunikaci aplikačních programů v síti, vytváření dočasných
<b>Relační vrstva</b> body)	- doplňuje logické rozhraní pro aplikační potřeby (poloduplex, synchronizační
<b>Preseňční vrstva</b> utajení)	- transformuje přenášená data (převody pro nekompatibilní počítače, komprese,
<b>Aplikační vrstva</b>	- konečná vrstva standardních aplikací

Lokální x rozlehlé sítě  $a = \text{zpoždění v síti} / \text{střední doba pro přenos jednoho rámce}$



## Přenosová média (optická a metalická)

Základní prvek charakterizující síť. Většinou sériový přenos. (výjimka Cluster One, Twentynet)

### 1) *metalická vedení*

***symetrická vedení*** - kroucený dvoudrát

- jednoduchý (telefonní) nebo dvojité dvojdrát (obvykle na datové přenosy)
- až 155 Mb/s, 100 m
- typicky přenos kódovaných signálů v základním pásmu
- je i varianta pro gigabitový Ethernet
- ATM
  
- UTP kategorie:
  - Cat.3 - mezní frekvence až 16 MHz, 10 Mb/s, Voice Grade
  - Cat.4 - až 20 MHz, 20 Mb/s
  - Cat.5 - až 100 MHz, 100 Mb/s, Data Grade
  - Cat 5e, 5+ - ještě lepší rozdíl mezi přeslechem a utlumem na mezní frekvenci
  - Cat 6 - 200 MHz, navržený
  - Cat 7 - 600 MHz, navržený
  
- STP je od IBM navržen pro Token Ring
  - Type1 – Type8
- kabely pro sériové rozhraní 9.6 - 115.2 kb/s, jednotky metru – standard
- FTP, SFTP - folie stíněné, čtyřpárové

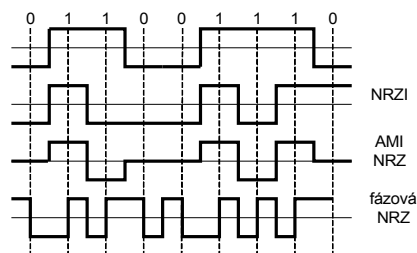
***nesymetrická vedení***

- koaxiální kabely
- přenos kódovaných signálů v základním pásmu 0 - 50 MHz
  - až 20 Mb/s
  - stovky metrů
  - Ethernet 10BASE-5
- přenos modulovaného signálu v přeloženém pásmu 5 - 800 MHz

- skupina kanálů až 40 Mb/s
- kilometry
- kabelové televize + internet

## 2) optická vedení

- jednotky až desítky km
- vysoká odolnost proti rušení
- až 10Gb/s
- optika je standardním přenosovým médiem u FDDI. Použitelné v ethernetu i token ring
- jádro (sklo) + obal (sklo nebo plast, nižší index lomu)
- dříve skoková změna indexu lomu, dnes gradientní (větší podíl energie přenášené s větším úhlem)
- primární ochrana - ultrafialovým světlem tvrditelný akrylát, nebo polymid
- sekundární ochrana - plastická hmota
- napojení vláken: přiložení, slepení, nebo svaření (vždy nutno proměřit odraz, útlum...)
- vysílání zajišťují světloemitující LED nebo laserové ILD diody
- příjem: fotodioda PIN, nebo lavinová APD
- **mnohavidova**
  - šíření s různými úhly odrazu, různá délka cesty paprsky  $\Rightarrow$  vidová disperze
  - 850, 1300nm vlnová délka
  - starsi  $\Rightarrow$  skoková změna indexu lomu, novejsi  $\Rightarrow$  gradientní - větší průměr (snazší napojování)
  - 50/125um, 62.5/125um, 100/140um (starsi)
- **jednovidova**
  - jeden mod, 1300 nebo 1550nm, až 100km



## Kódování

neupravený signál není vhodný pro přenos, obsahuje SS složku (obtížné přenést), nezaručený výskyt elektrických změn.

### Po jednotlivých bitech

#### (a) 2 úrovně napětí:

- **NRZ** – původní data
- **NRZI** - každá jednička - reverzace, každá nula - beze změny signálu
- **PSK - Manchester** – přechod mezi jedničkou a nulou se kóduje dlouhou úrovní, sousední stejné log. úrovně krátkou
  - Ethernet
- **DPSK** – přechod z nuly na jedničku a sousední jedničky - dlouhá úroveň, jinak krátká úroveň
  - Token Ring

#### (b) Víceúrovňové signály:

- **2B1Q**
- **4B3T**
- **ISDN, HDSL**

### (c) Pseudoternární:

- **AMI** - jedničky jako pulsy se střídající polaritou, do dlouhé posloupnosti nul lze vložit synchronizační pulsy
- **B8ZS, HDB3** - modifikace AMI pro ISDN

**Posloupnosti** : Bitové posloupnosti dané délky se převádí na posloupnosti delší, popř. vícehodnotové (ternární, ...) tak, aby signál získaný jednoduchým kódováním (NRZI, ...) měl nulovou stejnosměrnou složku a dostatek hran.

- **4B5B** - FDDI, 100BASE-Tx
- **8B10B** - gigabit. Ethernet.

## Modulace

Kmitočtové spektrum modulovaného harmonického signálu leží v jiné kmitočtové oblasti, než spektrum signálu modulačního - přeložené pásmo.

V lokálních sítích se nejčastěji používá kombinace amplitudové a fázové modulace.

## Sdílení média

### Kmitočtový multiplex

- založen na rozdělení pásma přenášeného médiem na oddělené kmitočtové intervaly, které využijeme pro vytvoření samostatného přenosového kanálu

### Časový multiplex

- spojitý signál vyučívající pásma přenášeného médiem je schopen pracovat s vyšší přenosovou rychlostí, principem v ISDN

### ISDN

- base rate - v 1 kabelu 2 kanály po 64Kb a 1 servisní 10Kb
- 32 kanálů 20x64Kb nebo 1x1920Kb

## Fyzická rozhraní

- modemová
  - připojení pomalejších dvoubodových kanálů do sítě
- RS-232 / V.24 - 19,2 kb/s, 15 m
- RS-423 - 3 kb/s na 1 km nebo 300 kb/s na 10 m
- RS-422 - 100 kb/s na 1 km nebo 10 Mb/s na 10 m
- V.35 CCITT - původně pro modemy 64 kb/s, dnes až 2,048 Mb/s
- RS-530 - až 2,048 Mb/s, částečně kompatibilní s RS-232 / V.24
- X.21 CCITT
  - navrženo pro digitální datové sítě s přepojováním kanálů
  - minimalizuje počet signálů, místo některých speciálních se používají posloupnosti na signálech
- ISDN
  - umožní využití plného přenosového pásma jinak využitého pro hovor (64 kb/s)
  - celkem 2 plně duplexní kanály po 64 kb/s + signalizační 16 kb/s
  - u účastníka rozhraní BRI
- USB
  - 1.1: 1 m, 1,5 Mb/s
  - 2.0: 480 Mb/s
  - signály:
    - [+5 V ; Data- ; Data+ ; uzemnění]
- Ethernet
  - konektor RJ45
  - vývody (2 dvoudráty) [ Tx- ; Tx+ ; Rx- ; Rx+ ]

## Potvrzovací schémata a jejich popis

- schémata spolupráce, která dovolí detekované chyby v přenesených rámcích napravit opakovaným přenosu
- nutností je alespoň detekční (když už ne samoopravné, z důvodu efektivního využití přenosové kapacity) kódování dat
  - iterační a cyklické
- modely - komunikující automaty a Petriho sítě
  - komunikující automaty: dvojice automatů, u nichž výstup jednoho může být vstupem druhého; popisují chování vysílače při potvrzení příjmu nebo zprávy o chybě apod.
  - Petriho sítě: uzly (místa, přechody - "tlusté čárky"), orientované hrany, značení ("tečky"); hrana může spojit jen přechod a místo
    - analýza stavového prostoru - graf dosažitelnosti (automat se shodným chováním)

### *Zpětná vazba*

- potvrzovací: vysílač pošle data i s kontrolní informací, přijímač pošle potvrzení - dvoubodové spoje
- detekční: vysílač pošle jen data, přijímač pošle jen kontrolní informaci přijatých dat - využito vyjimečně
- informační: vysílač pošle data s kontrolní informací, přijímač pošle zpět přijatá data a jejich kontrolní informaci - kruhové lokální a družicové sítě

### *Synchronní simplexní protokol*

- synchronní - nelze pozastavit, simplexní - provoz jedním směrem
- bez zpětného potvrzení - nereaguje na chyby

### *Simplexní protokol s pozitivním potvrzováním*

- nutný alespoň poloduplexní kanál, efektivní pro kanály s malou chybovostí
- vysílač pošle zprávu, čeká (do timeoutu) na potvrzení - nepřijde-li, zopakuje zprávu
- přijímač pošle potvrzení v případě bezchybného příjmu

### *Simplexní protokol s čistě negativním potvrzováním*

- rychlá reakce na chybu, ale
- nedokáže reagovat na ztracený rámec i ztracené potvrzení (odmítnutí)
- => použitelné při kombinaci pozitivního a čistě negativního potvrzování => Simplexní protokol s negativním potvrzováním
- vysílač pošle zprávu a čeká na odmítnutí; nepřijde-li do timeoutu, považuje zprávu za doručenou přijímač pošle odmítnutí při chybě

**Číslování rámců** - Pouhé potvrzování nezabrání ztrátám zpráv (někdy) a především jejich duplicitě. K tomu je třeba číslování.

- Jednoznačné - nutnost neomezeného rozsahu číslování.
- Modulární - 2, 8, 128.
- Číslovat lze rámce / příkazy i potvrzení / odpovědi.

### *Skupinové potvrzování*

- Pro snížení zpoždění / zvýšení propustnosti.
- Nepotvrzují se jednotlivé rámce, ale posloupnosti společně.
- Potvrzování např. číslem očekávaného (tedy zatím bezchybně nepřijatého) rámce.

### *Kontinuální potvrzování*

- nepřetržitý tok rámců - vysílač nečeká na potvrzení
- po uplynutí timeoutu na příjem již odvysílaného rámce nebo příjmu odmítnutí zopakuje
- počet rámců odeslaných bez potvrzení - okénko => okénkové schéma

- opakování rámců od určeného rámce, pokračuje se následujícími

### ***Selektivní opakování***

- opakuje se jenom ten rámec, na který bylo přijato odmítnutí
- omezení velikosti okénka na max. polovinu číslovacího modulu

### ***Nesamostatné potvrzení***

- zvláště u plně duplexního přenosu - vhodné doplnit informaci o potvrzení do rámce (tak se potvrzují rámce poslané opačným směrem)

## ***Linkové protokoly***

### Znakově orientované

#### SLIP

- pouze rozděluje data na rámce - tam ukládá IP pakety
- neobsahuje detekci chyb ani potvrzování
- jednotlivé IP pakety odděluje znakem 0xC0; objeví-li se tento v datech, je nahrazen dvojicí 0xDB 0xDC, objeví-li se v datech 0xDB, je prefixován také 0xDB
- neschopnost rozlišit protokoly vyšších vrstev (proto nahrazen PPP)

#### BSC

- pro poloduplexní přenos po mnohabodovém spoji
- zajišťuje segmentaci zprávy (na několik menších)
- rámec uvozuje ASCII znakem SYN
- zabezpečení cyklickým kódem nebo kontrolním součtem
- výskyt řídicích znaků v datech prefixován 0x10
- střídavé potvrzování - číslování pouze potvrzení (číslo očekávaného rámce)

### Bitově orientované

- zabezpečují transparentci dat technickými prostředky
  - vkládání bitů - za každých 5 jedniček přidá vysílač nulu (začátek a konec rámce je označen 6 jedničkami)
- opírají se o obvodový výpočet cyklických kódů
- okénkové potvrzování
- umožňují plně duplexní přenos

#### SDLC

- nejstarší, vychází z něj doporučení HDLC a z něj mnoho dalších protokolů
- rámec obsahuje adresu stanice, možnost potvrzení

#### PPP

- vychází z HDLC
- pro 2bodové spoje
- dovoluje detekovat chyby, ale opravu nechává na protokolech vyšších vrstev
- dovoluje rozlišit protokoly síťové vrstvy

#### MPPP

- pro více paralelních spojů (2 kanály ISDN apod.)
- dělení dlouhých paketů do fragmentů

### Protokoly lokálních sítí

adresace stanic na sdíleném médiu - např. MAC

podpora více současně užívaných protokolů síťové vrstvy

LLC1

LLC2 - okénkové potvrzování

LLC3 - střídavé potvrzování

LLC - Token Ring, FDDI

Ethernet - místo délky paketů v LLC uchovává typ protokolu

### **LLC1**

- Datagramová služba bez potvrzování
- Point to point, Multicast, Broadcast

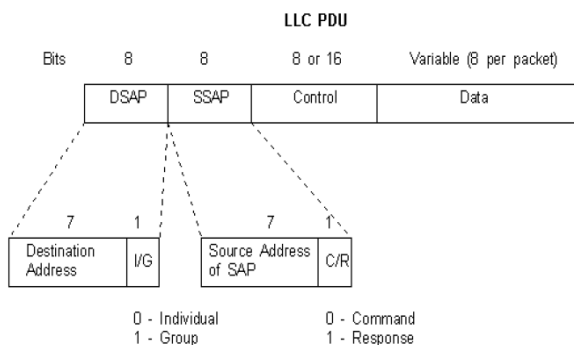
### **LLC2**

- Logické spojení (connection mode service)
- protokol který zajišťuje během přenosu, že všechna odeslaná data budou předána protějšku
- okénkové potvrzování

### **LLC3**

- Potvrzovaná datagramová služba (střídavé potvrzování)
- DL-DATA-ACK – potvrzovaný přenos dat
- DL-REPLY – požádání vzdálené aplikace o předem připravená data
- Token Ring, FDDI

### **LLC Frame**



## **Směrovací algoritmy, datagram a virtuální kanál**

**Záplavové** - každý uzel kromě příjemce vyšle přijatý paket do všech směrů

**Náhodné** - uzel odešle paket do náhodně vybraného směru

**Izolované** - bere v úvahu pouze lokální informace a už ne inf. ostatních uzlů sítě

**Horký brambor** - paket se odešle na linku, která má nejkratší frontu

**Zpětné učení** - uzel vytváří směrovací tabulku z informací z příchozích paketů - hlavně čas přenosu nebo počet prošlých uzlů

**statické** - směrovací tabulky se tvoří při návrhu sítě, vhodné určit i alternativní směry

**adaptivní** - směrovací tabulky se přizpůsobují momentálnímu stavu

**centralizované / distribuované** - vhodná je kombinace občasného (ne častého) přepočtu směrovacích

tabulek s využitím lokálních informací (zrychlení přizpůsobení)

### **RIP**

- Ford-Fulkerson
- distribuovaný
- uzly si v pravidelných intervalech vyměňují informace o vzdálenostech (odhadech) k dalším uzlům
  - distance vector - kam, soused, délka
- může způsobit směřování v cyklech
- rychle reaguje na "dobré", ale pomalu na "špatné" zprávy
  - split horizon - uzel nepředá informaci směrem, ze kterého ji získal (urychlení aktualizace při výpadku)
  - inverse poison - ve směru, odkud dostávám informaci o nejlepší cestě, posílám nekonečno

### **OSPF (Open Shortest Path First)**

- Dijkstra
- stavy linek se šíří broadcastem
- pořadí broadcastu - číslování
- restart směrovače - pošle broadcast s nepoužívaným číslem (aby nebyl ignorován jako starý)

### **Hiearchické směřování**

- rozdělení adresy: prefix oblasti + adresa uzlu uvnitř
- adresování respektuje topologii

### **Dijkstrův algoritmus**

Každý uzel označen L - dosud zjištěná vzdálenost k uzlu, D - index uzlu který je sousedem na nejkratší dosud známé cestě, P - že L a D se dále již nemění. Pouze 1 řádek tabulky, nutné provést pro každý uzel.

- 1) Start = s,  $L(s)=0$ ,  $D(s)=s$ ,  $P(s)=1$  pro s a  $L(v)=\infty$ ,  $D(v)=(v)$ ,  $P(v)=0$ ,  $u=s$
- 2) Po každý v jež je soused u a  $P(v)=0$ ,  $M=\min(L(v), L(u)+len(u,v))$  if  $M<L(v)$  then  $L(v)=M$  a  $D(v)=u$
- 3) Mezi uzly v pro které platí  $P(v)=0$ , vybereme  $\min(L(v))$ , jeli jich více vybereme nejkratší z hlediska hran, a nastavíme  $P(v)=1$ .
- 4) Dělej dokud alespoň jeden je  $P(x)=0$

**Ford-Fulkersonův** - modifikace Dajkstry, lze jej jednoduše implementovat. Krok 2 odpovídá výměně paketů mezi sousedy. V praxi s vektorem vzdáleností a vytvářejí se přímo směrovací tabulky.

- 1) Start = s,  $L(s)=0$ ,  $D(s)=s$  pro s a  $L(v)=\infty$ ,  $D(v)=(v)$
- 2) Pro každý uzel  $v \in G$ ,  $M=\min(L(v), L(u)+len(u,v))$
- 3) Pro každý uzel, pro který platí  $M(v) < L(v)$ ,  $L(v)=M(v)$ ,  $D(v)=u$ , kde u je označení sousedního uzlu, pro který bylo min zjištěno
- 4) if změna alespoň u 1 uzlu tak opakuj od 2.

**řízení toku** - mechanismy, které zabraňují lokálnímu, nebo globálnímu přetížení sítě.

**datagram a virtuální kanál** - zprávy v síti s přepojováním lze doručovat různě. Rozdělíme ji do paketů (data + zdroj +cíl), Datagram = předávám nezávisle na předávání ostatních, nebo vytvořím virtuální kanál a uložíme informaci v uzlech.

### **Řízení toku**

- nutný pro co nejlepší využití kapacit sítě
- průchodnost může s rostoucí zátěží (bez řízení toku) i klesat - přeplněné fronty apod.
- odmítání paketů - může vést k naplnění výstupních front sousedních uzlů a tím šíření zahlcení po síti
- zahazování paketů

- omezení šíření zahlcení
- výpadek paketu musí řešit koncové potvrzování
- vhodné omezení délky front a zahazování paketů už když se blíží vyčerpání jejich kapacit

### Diferencovaná obsluha

- např. QoS
- fronty FIFO - nediferencovaná
- prioritní obsluha - více front podle tříd toků
- WQF - snižuje nevýhodu absolutního zvýhodnění některých toků - cyklicky střídá (ne pravidelně cyklicky)

### Koncové řízení toku

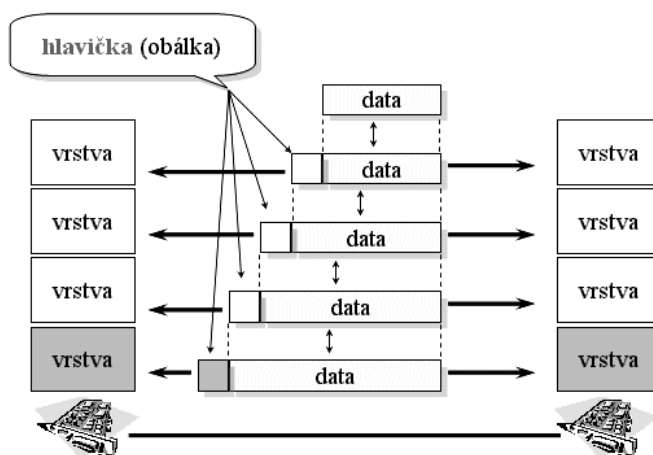
- Token bucket - formování toku - vstupní tok do sítě omezen kredity - po vyčerpání se pakety ukládají do fronty
- Leaky Bucket - klasifikace - označování paketů, které překračují / nepřekračují dohodnutý limit (střední tok apod.); na základě toho jsou pakety vyloučeny ze sítě úplně nebo jen přednostně vyhazovány ze zatížených uzlů
- zpětnovazební mechanismy
  - při přetížení je koncové stanici odeslán škrtící paket - reaguje přivřením potvrzovacího okénka
  - indikace zahlcení - paketům jsou při průchodu zahlcenou oblastí přidány příznaky

## Struktura přepojovacího uzlu, směrovače

### Nejdříve trochu teorie

Pokud naše zařízení (např. switch) dokáže správně interpretovat alespoň část obsahu přenášeného bloku, otevírají se mu obrovské možnosti v tom, jak s blokem může naložit. Může například dojít k závěru, že jej má předat jen jedním směrem (cíleně do jednoho ze segmentů, ke kterým je připojena). Nebo že jej nemusí vůbec nikam předávat a může jej zahodit (pokud takovýto datový blok nemusí resp. nemá opustit segment ze kterého přišel).

Zde je nutné si uvědomit, jak takovýto datový blok obecně vzniká - tak, že na nejvyšší (aplikační) vrstvě je zadán k přenosu určitý "užitečný" blok dat, a každá z vrstev k těmto datům přidá svou vlastní hlavičku:



Obr. 5: Obecná představa zapouzdřování dat ("vkládání do obálek")

Pokud použijeme tuto metaforu s obálkami, pak poslední (největší, nejvíce vnější) obálka odpovídá linkové vrstvě (2. vrstvě sedmivrstvého modelu ISO/OSI), a obsahuje mj. linkovou (např. ethernetovou) adresu odesílatele a příjemce. Další (více vnitřní) obálkou je obálka s hlavičkou patřící síťové vrstvě - zde jsou především síťové adresy odesílatele a příjemce (nebo identifikátor virtuálního přenosového okruhu). Další (ještě více vnitřní) obálkou je hlavička patřící vrstvě transportní a obsahuje tzv. čísla portů odesílatele a příjemce (vypovídající o typu aplikace, které data patří). No a ještě hlouběji pak jsou



samotná aplikační data - například URL odkaz na stránku, kterou si WWW prohlížeč právě vyžádal na určitém WWW serveru (nebo je to již obsah příslušné stránky, jako odpověď na požadavek).

Tomu, že z jedné strany přijme nějaký datový blok a následně jej předá dál do jiné strany (segmentu, sítě apod.) se obvykle říká "přepojování". Potom také můžeme naši "krabičce" začít říkat "přepojovací uzel". Skutečnost, že uvnitř přepojovacího uzlu dochází k přepojování celých datových bloků, se obvykle zdůrazňuje termínem "**přepojování paketů**" (packet switching).

**Switch** (přepínač) - Pokud se přepojovací uzel rozhoduje podle údajů příslušejících linkové vrstvě (největší, resp. nejvíce vnější obálce ve smyslu předchozí metafory), jedná se o přepojování na linkové vrstvě (na 2. vrstvě ISO/OSI). Na úrovni této vrstvy jsou dostupné především tzv. linkové adresy, které jsou dány použitou linkovou technologií. U Ethernet jde o 48-bitové ethernetové adresy atd. Dále může být na této úrovni znám typ nákladu (protokol, podle kterého jsou formátována data v datové části - například že jde o paket protokolu IP z rodiny TCP/IP, nebo že jde o paket novellských protokolů IPX apod.). Datovému bloku, který se přenáší na úrovni linkové vrstvy, se říká rámec (frame).

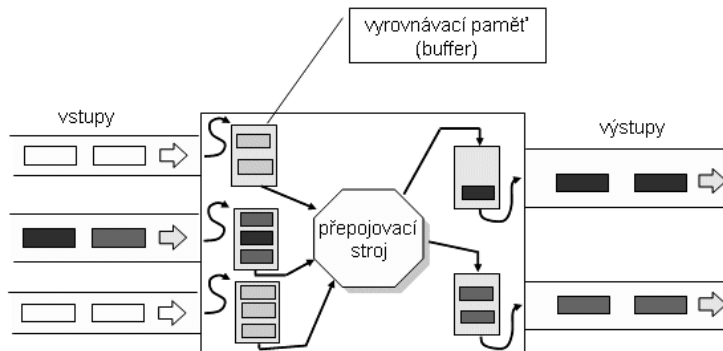
**Router** (směrovač) - Pokud přepojovací uzel dokáže správně interpretovat údaje příslušející síťové vrstvě a rozhodovat se podle nich, pak se jedná o přepojování na síťové vrstvě (na 3. vrstvě ISO/OSI). Na úrovni této vrstvy jsou dostupné především tzv. síťové adresy - například 32-bitové IP adresy (v sítích na bázi protokolů TCP/IP), či IPX adresy apod. Dále na této úrovni mohou být dostupné informace o typu "nákladu" (např. že data patří protokolu TCP, nebo UDP...

Dokáže-li přepojovací uzel správně interpretovat údaje příslušející transportní vrstvě a rozhodovat se podle nich, pak se jedná o přepojování na transportní vrstvě (na 4. vrstvě ISO/OSI). Hlavní informace, dostupné na této vrstvě, jsou tzv. čísla portů (port numbers). Příklad: data adresovaná portu č. 80 jsou určena pro WWW server, a tudíž představují nějaký požadavek na tento server.

Dokáže-li přepojovací uzel interpretovat i samotná data patřící konkrétní aplikaci (a rozhodovat se podle nich), pak se jedná o přepojování na aplikační vrstvě (na 7. vrstvě ISO/OSI).

Většina dnešních sítí je budována nikoli na bázi síťové architektury ISO/OSI, která má 7 vrstev, ale na bázi architektury TCP/IP, která má pouze 4 vrstvy (viz dále), proto ve výčtu chybí přepojování na 5. a 6. vrstvě (relační a prezentační).

**Fungování přepojovacích uzlů** - Vnitřek přepojovacího uzlu si lze představit jako blok, který má určitý počet vstupů a určitý počet výstupů. Pro každý vstup resp. výstup je uvnitř přepojovacího uzlu tzv. buffer (vyrovnávací paměť), kde se přichází resp. odchozí pakety hromadí. Další částí přepojovacího uzlu je "přepojovací stroj" (tzv. engine), který si lze představit jako aktivní prvek který podle určitého pravidla odebírá bloky nahromaděné ve vstupních bufferech, analyzuje jejich obsah a rozhoduje se, jak s nimi dál. Pokud se zrovna nerozhodne, že daný blok může zahodit, zvolí jeden z výstupních směrů a příslušný datový blok vloží do bufferu, který je s tímto výstupem spojen.



Obr. 8: Obecná představa přepojovacího uzlu

Do každého vstupu vstupuje každý datový blok na úrovni linkové vrstvy, neboli jako linkový rámeček. Stejně tak z každého výstupu vystupuje datový blok na úrovni linkové vrstvy, tj. jako linkový rámeček. Pozor ale na to, že ne všechny vstupy a výstupy musí vždy používat stejnou linkovou technologii!! Například směrovač může propojovat jeden ethernetový segment lokální sítě s virtuálním okruhem na bázi technologie Frame Relay (dvoubodový spoj).

### ***Na jakem principu a k čemu slouží technologie Virtualních lokálních sítí (VLAN)?***

Mikrosegmentovaná LAN, kde jsou jednotlivá zařízení připojena ke switchům. Ve vstupním přepínači jsou data opatřena identifikátorem skupiny – VI (VLAN Identifier) ( $2^{12}$  hodnot). Výstupní switch zkontroluje, zda jsou data určena stanici, ke které je switch připojen, pokud ano, tak switch VI odstraní a předá cílové stanici. V opačném případě přepínač rámeček zlikviduje. Rámeček určený pro tyto tagovací mechanismy obsahuje také VLAN-TID (Tag Protocol Identifier)

### ***Most vs Switch (co je spanning tree), Transparentní most - vytváření tabulek?***

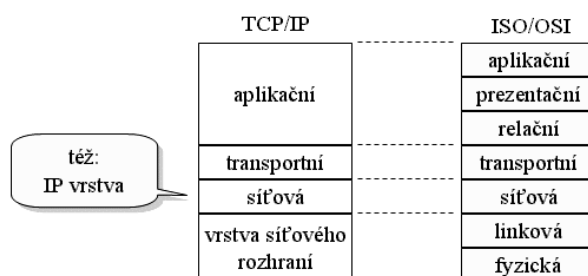
- Bridge, rozhoduje se na základě MAC adres (prijme, ignoruje), Translation Bridges - propojují síť s různými formáty rámečků (Ethernet, Token Ring, FDDI)
- Switch více než dvě porty Bridge, musí mít cut-through
- Tabulky: statické x dynamické (transparentní mosty)
- Transparentní - do směrovací tabulky ukládají adresy sítí odesílatele
  - správa patří do stejné sítě  $\Rightarrow$  zahodí se
  - cíl zprávy v tabulce  $\Rightarrow$  poslat do příslušné sítě
  - neznámý cíl nebo broadcast  $\Rightarrow$  broadcast
  - vyžaduje stromovou strukturu  $\Rightarrow$  spanning tree algoritmus (hledání kostry)
    - mosty mají ID, komunikace služebními rámečky BPDU
      1. nejprve vyber kořenu - ten s nejnižším ID
      2. kořen rozesílá na všechny výstupy BPDU obsahující cenu cesty pro daný výstup.
    - volba mostu s nejnižší cenou cesty ke kořenu – designated
    - sousedící rozhraní s kořenem označeno *root port*

## **Propojování sítí (internetworking), protokoly TCP/IP**

Propojování sítí (internetworking) spočívá v propojování samostatných segmentů sítě v objemný celek. Je možné jej realizovat např. pomocí: Repeaterů („Zesilovač“, který předává veškeré informace z jednoho síťového segmentu na druhý. Používá se k prodloužení síťového segmentu) nebo pomocí Routerů (směrovač – viz výše).

**TCP/IP** - Vzhledem ke složitosti problémů je síťová komunikace rozdělena do tzv. vrstev, které znázorňují hierarchii činností. Výměna informací mezi vrstvami je přesně definována. Každá vrstva

využívá služeb vrstvy nižší a poskytuje své služby vrstvě vyšší. Celý význam slova TCP/IP je *Transmission Control Protocol/Internet Protocol*.



Obr. 7: Vrstvy ISO/OSI a TCP/IP

**Vrstva síťového rozhraní** – Nejnižší vrstva umožňuje přístup k fyzickému přenosovému médium. Je specifická pro každou síť v závislosti na její implementaci. Příklady sítí: Ethernet, Token ring, FDDI, X.25, SMDS.

**Síťová vrstva** – Vrstva zajišťuje především síťovou adresaci, směrování a předávání datagramů. Protokoly: IP, ARP, RARP, ICMP, IGMP, IGRP, IPSEC. Je implementována ve všech prvcích sítě - směrovačích i koncových zařízeních.

**Transportní vrstva** – Transportní vrstva je implementována až v koncových zařízeních (počítačích) a umožňuje proto přizpůsobit chování sítě potřebám aplikace. Poskytuje spojované (protokol TCP, spolehlivý) či nespojované (UDP, nespolehlivý) transportní služby.

**Aplikační vrstva** – Vrstva aplikací. To jsou programy (procesy), které využívají přenosu dat po síti ke konkrétním službám pro uživatele. Příklady: Telnet, FTP, HTTP, DHCP, DNS.

## Propojování sítí (internetworking), protokoly TCP/IP.

internetwork - soustava více propojených lokálních sítí

**Most** - nejjednodušší prvek, slouží k propojení dvou lokálních sítí

- přijímá všechny rámce ze všech lokálních sítí
- analyzuje adresy přijatých rámců a rozhoduje se zda je odešle do některé lokální sítě, nebo zničí
- uloží přenášení rámce do své paměti a nakonec je odešle

**Transparentní most** - pokud jsou stanice jednoznačně adresovány MAC adr. Pouze v síti se stromovou strukturou. Algorytmus (distribuovaný) **výběru kostry**. Založen na jednoznačné identifikacím, Bridge Protocol Data Unit.

- sleduje veškerý provoz v síti. Vede si evidenci stanic, jejichž adresy jsou uvedené jako odesílatel ze sítě ve formě směrovací tabulky. Pro každou adresu, která se objeví v poli odesílatel, je uvedena síť ze které přišla. ukládání = učení
- Na každou přijatou zprávu reaguje
  - 1) ví, že přišla ze směru odkud byla poslána = **zahodí**
  - 2) ví, že leží v jiné síti, než ze které byla poslána = **převede ji do ní**
  - 3) určená pro stanici, kterou nezná = **rozešle do všech směrů**

**Explicitní směrování** - Informace o směru a cestě si nese zpráva sama. Buď staticky, nebo dynamické zjištění nevhodnější cesty. Informace se zjistí při otevírání spojení. Vyšle se speciální rámec, který je mosty rozeslán do všech podítí, cestou doplňován o adresy mostů a sítí, jimiž prošel. Adresát vrací odpověď na každý došlý rámec. Odesílatel si pak vybere ten nej.

**Router** - v případech, kdy nelze zajistit MAC adresy. Použití médií, které to třeba vůbec nepodporovaly, (Internet) Není rozhodující o jakou síť se jedná, Zabaleno do obálky, a je vytvářena pro každou síť, Problém s limitovanou délkou paketu.

**Adresy** ClassA 0+net+host, ClassB 10+net+host, ClassC 110+net+host, ClassD 1110+net+host, ClassE 11110+net+host.

**ARP** - překlad IP na MAC - pošle dotaz broadcastem na IP. stanice která má dané IP odpoví ARP Reply (IP+MAC) Tazatel si uloží IP+MAC do ARP Cache

**RARP**- automatické přidělení IP, stanice po připojení rozešle RARP s žádostí o IP, IP odpoví RARP server RARP Requestem

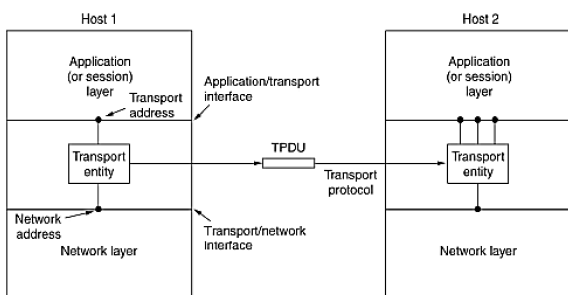
**IP** Internet Protocol - navržen jako datagramová služba. přenos je zajištěn na základě IP, neoprotvrzován.

verze prot	head. len	tot. len	
identification	flags	offset	
time to live	protocol	checksum	
source adr.			
destination adr.			
options		padd	
data			

Flags - frgmentace, poslední frgment,  
Service - priorita, zpoždění, kapacita, spolehlivost  
Ident - podpora fragmentace  
offset adresa relativní fragmentu  
Protocol - vyšší vrstvy UDP, TCP, ...

## Transportní služba, rozhraní TCP a UDP.

Služby prováděné na nejvyšší úrovni



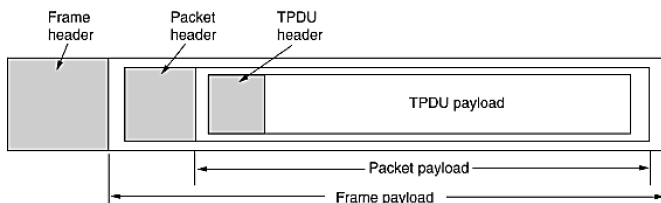
Síťová, transportní a aplikační úroveň

Primitivní služby transportní úrovně

Primitive	Packet sent	Meaning
LISTEN	(none)	Block until some process tries to connect
CONNECT	CONNECTION REQ.	Actively attempt to establish a connection
SEND	DATA	Send information
RECEIVE	(none)	Block until a DATA packet arrives
DISCONNECT	DISCONNECTION REQ.	This side wants to release the connection

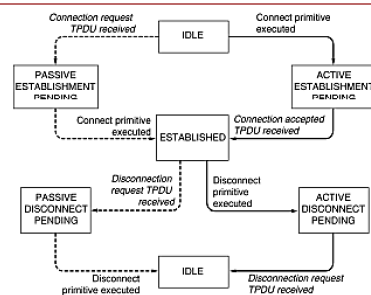
Primitivní funkce pro jednoduché transportní služby.

Primitivní služby transportní úrovně (2)



Zapouzdření TPDU, paketů a rámců.

Primitivní služby transportní úrovně (3)



Stavový diagram pro jednoduchý transportní protokol. Plné čáry představují přechody klienta, tečkované přechody serveru. Přechody popsané skloněným písmem jsou způsobeny příjmem paketů.

**TCP** - poskytuje zabezpečený přenos. **stream** - jako posloupnost slabik

source adr		
destination adr.		
zero	protocol	TCP length
source port		dest port
seq numb		
ack numb.		
head. len	code	window
check sum		urgent pointer
options		padding
data		

seq.numb - pořadové číslo Ack.numb. jeho potvrzení  
code - URG,ACK,PSH,RST,SYN,FIN  
Window - kredit mechanismu řízení toku

**UDP** - datagramový protokol, nadstavba IP, nezabezpečený přenos

source adr		
destination adr.		
zero	protocol	TCP length
source port		dest port
udp len		checksum
data		

checksum- all1=nulový CRC, all0 = nezabezpečený

**Datagramová služba bez potvrzování** - síť nezajišťuje bezpečné dodání, ani neinformuje odesílatele o doručení. jen primitiva pro odeslání a příjem. Vhodná pro sběr dat, kde se periodicky vysílání opakuje. VoIP hovor, telemetrie. Není zaručeno ani pořadí paketů.

**logické spojení** - vytvářet, využívat a rušit logické spoje. Při navazování se dohodnou o spojení a inicializují spojení. zajišťuje předání všech dat a předána v odeslaném pořadí. V případě poruchy odesílatel je informován.

**Potvrzovaná datagramová služba** - potvrzovaný přenos dat, požádat vzdálenou aplikaci a předem připravená data

