



# Počítačové sítě a operační systémy

---

## Protokoly pro přenos dat v síti Směrování

Jaromír Plhák  
[xplhak@fi.muni.cz](mailto:xplhak@fi.muni.cz)



# Rodina protokolů TCP/IP (1)

---

- Rodina protokolů pro komunikaci v počítačové síti
- Protokol
  - Množina pravidel určující význam a syntaxi zpráv při komunikaci

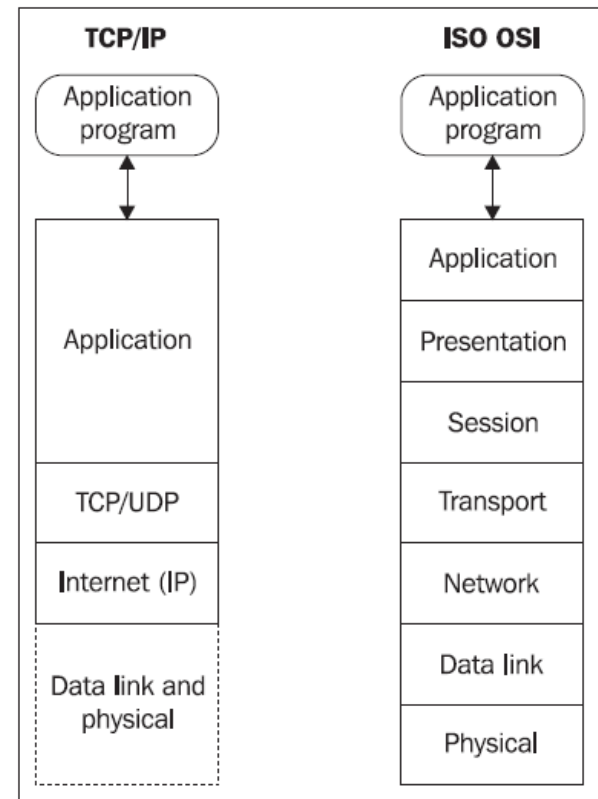


# Rodina protokolů TCP/IP (2)

- Počítačová síť není považována za 100% spolehlivou
  - Protokoly s tím musí umět pracovat
  - Zprávy rozděleny do menších zpráv – paketů
  - Pakety jsou nezávisle adresovány k cíli
  - Nezáleží na tom, kudy se tam dostanou
  - Příjemce potvrzuje přijetí každého paketu
  - Pakety se mohou ztrácet – posílání znovu

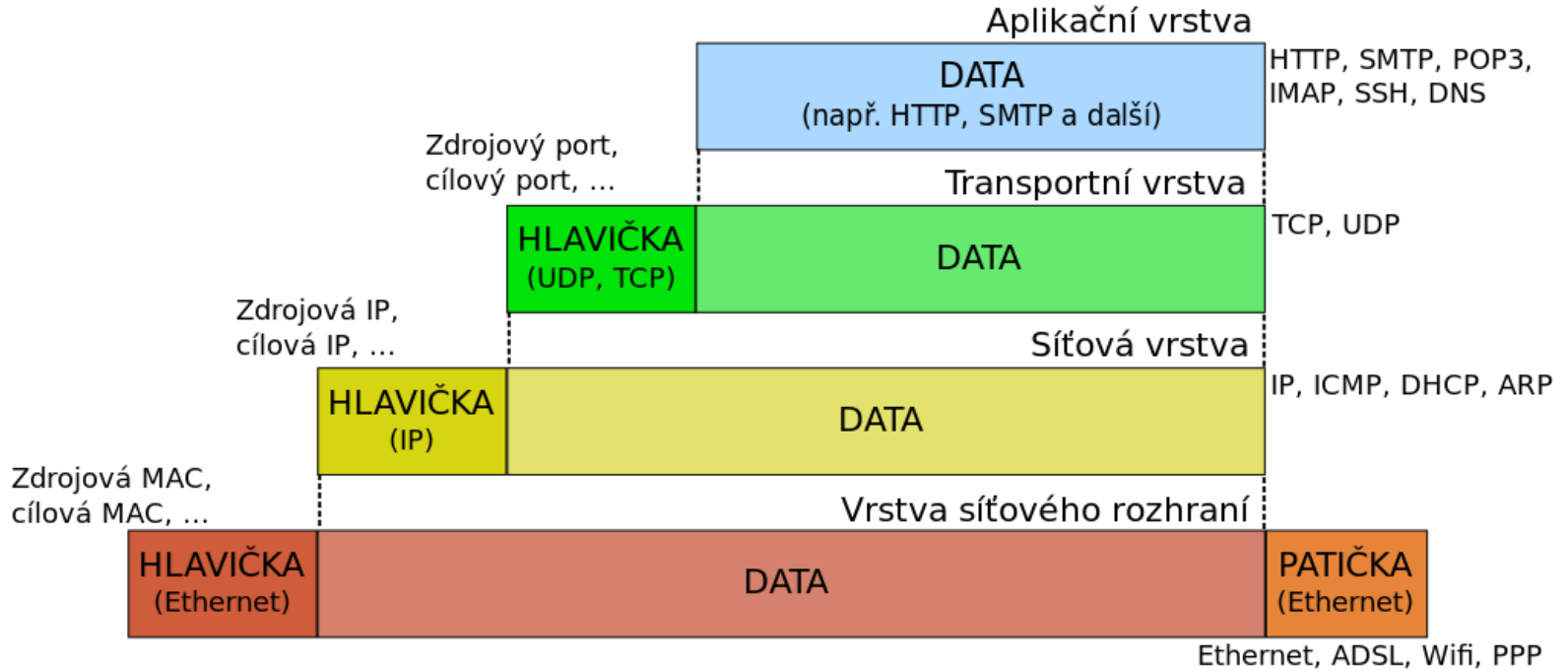
# Architektura TCP/IP (1)

- Komunikace je rozdělena do vrstev
  - Analogicky, jako model ISO/OSI, ale vrstev je méně
- Aplikační
  - DHCP, DNS, FTP, HTTP, Telnet
- Transportní
  - TCP, UDP
- Síťová
  - IP, ARP, ICMP, IPSec
- Síťové rozhraní
  - Ethernet, Token ring...



# Architektura TCP/IP (2)

## ZAPOUZDŘENÍ DAT V SÍTI TCP/IP

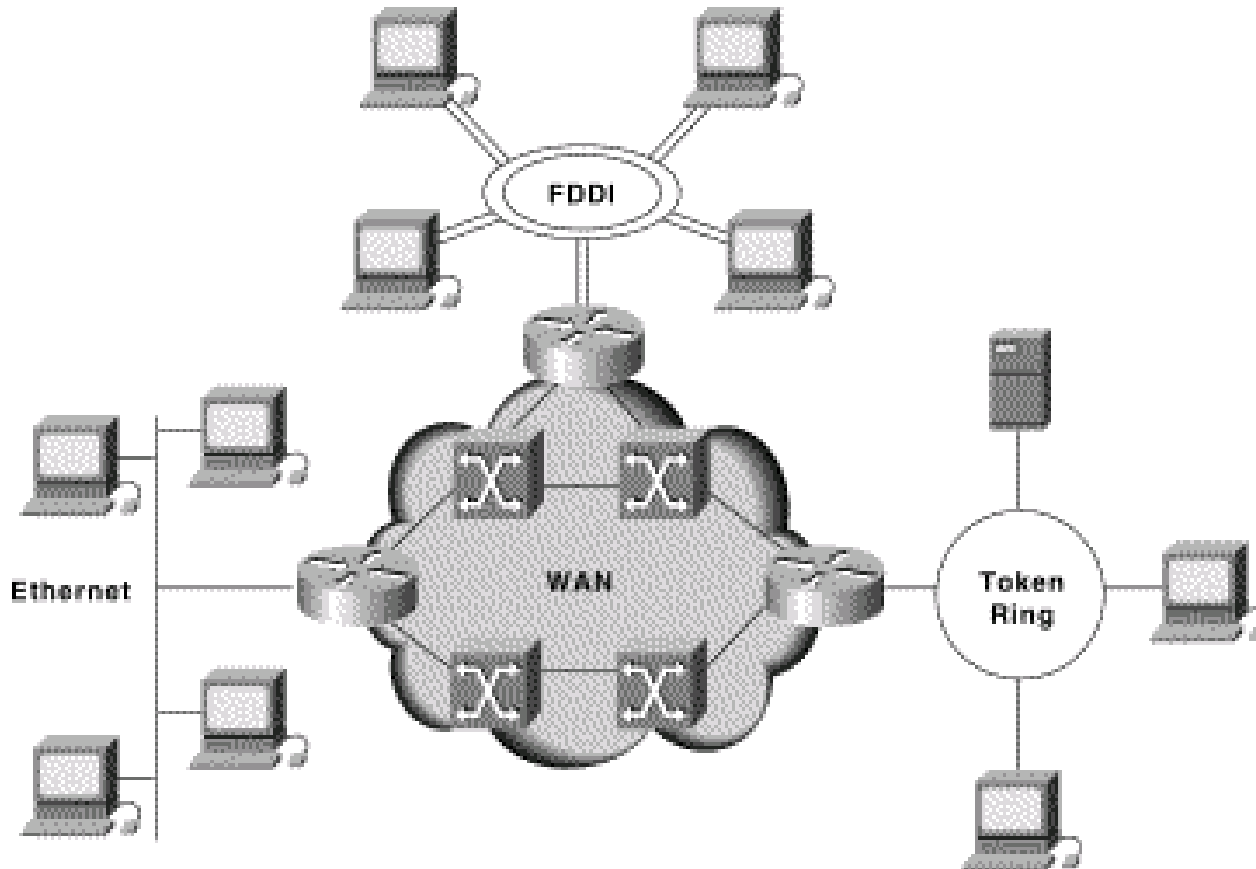


# Síťová vrstva (1)

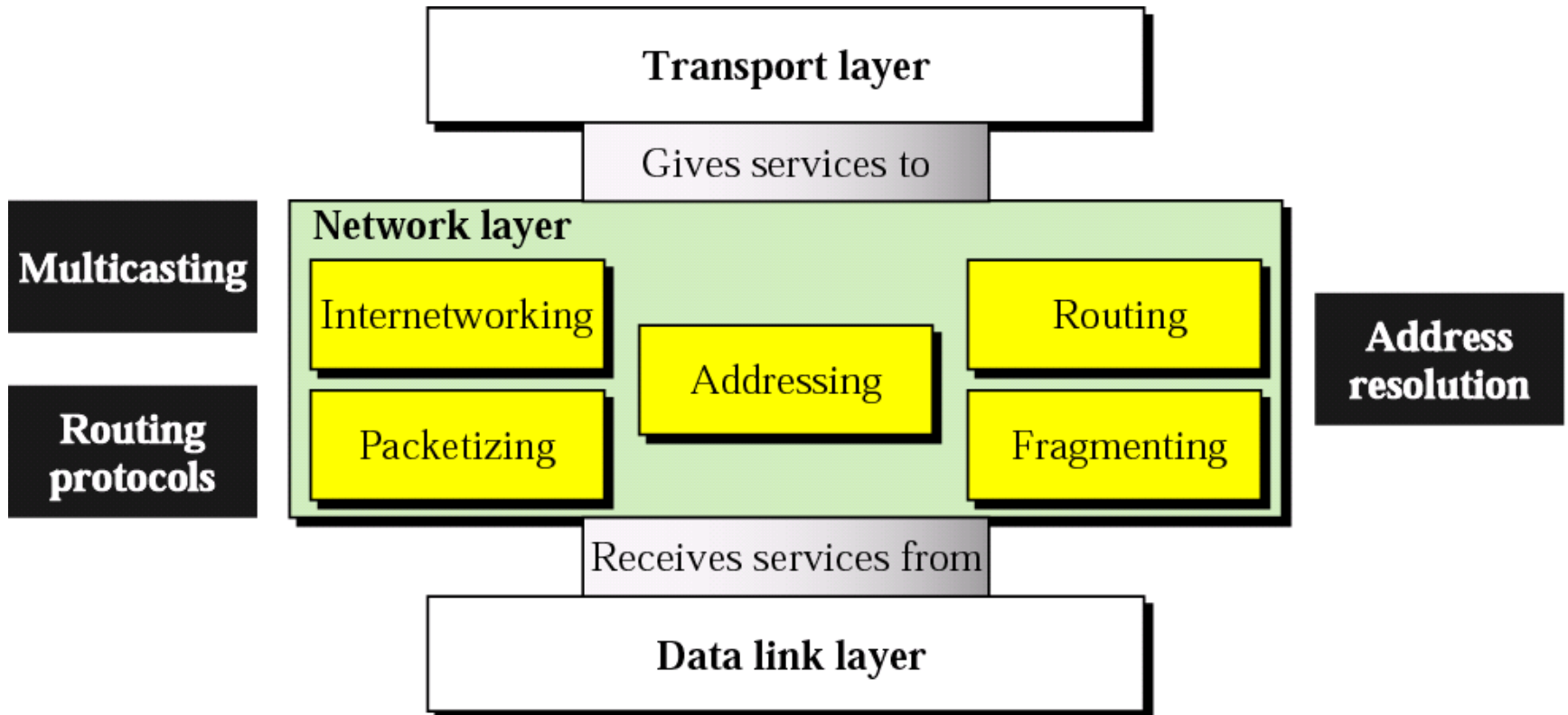
---

- Proč nám nestačí vrstva datového spoje?
  - Nemožnost vybudovat geograficky libovolně rozsáhlé sítě
  - Neuniformní prostředí různých sítí
- Chceme mít možnost vytvořit komunikační kanál mezi libovolnými stanicemi v Internetu
  - Skrze více samostatných fyzických sítí (LANs)

# Síťová vrstva (2)



# Služby síťové vrstvy





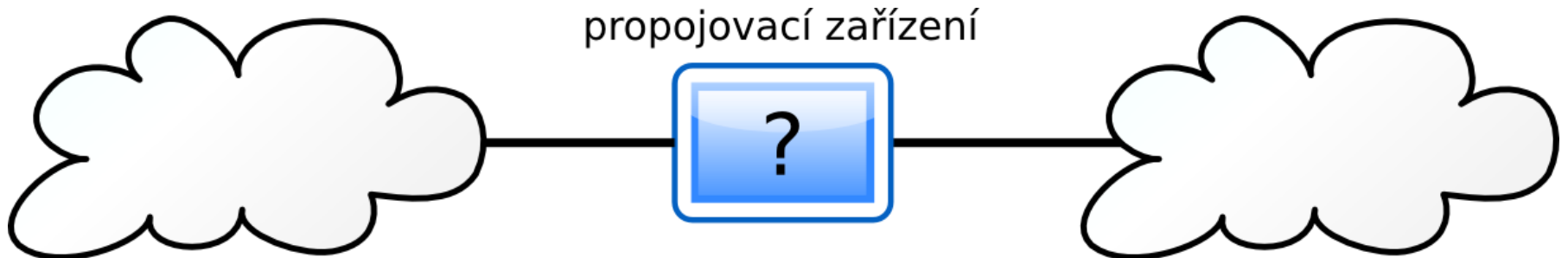


# Důvody propojování sítí

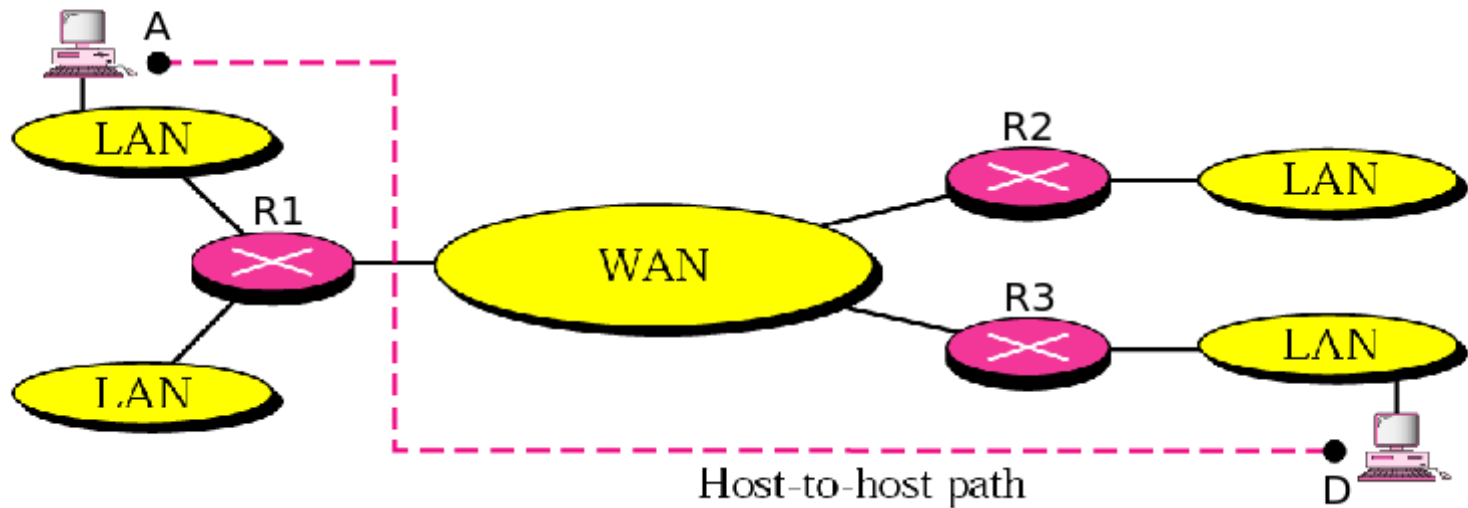
- Překonání technických omezení/překážek
  - Např. omezený dosah kabelových segmentů
- Optimalizace fungování sítě
  - Snaha regulovat tok dat, zamezení zbytečného šíření provozu
- Zpřístupnění vzdálených zdrojů
  - Přístup ke vzdáleným serverům
- Zvětšení dosahu poskytovaných služeb
  - Elektronická pošta, internetové telefonování atd.

# Propojování sítí dle vrstev (1)

- Fyzická vrstva – opakovač (repeater)
- Vrstva datového spoje – můstek (bridge),  
přepínač (switch)
- Síťová vrstva – směrovač (router)
- Aplikační vrstva – brána (gateway)

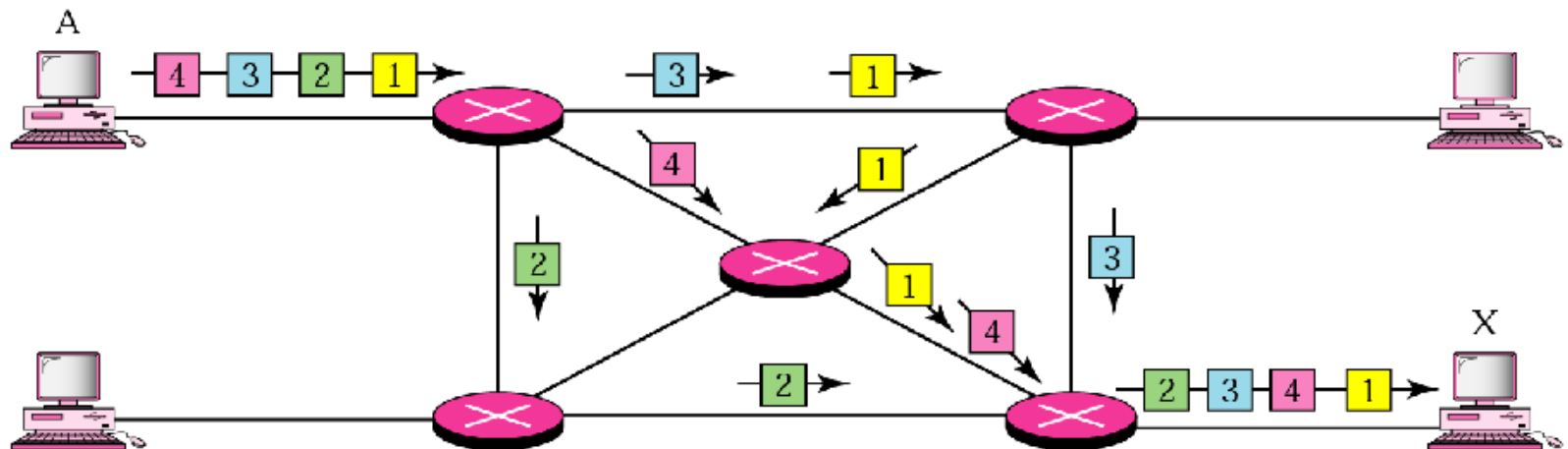


# Propojování sítí dle vrstev (2)



# Modely propojování sítí

- Přepínání okruhů (Circuit Switching)
  - Ustavení přímého fyzického spojení mezi odesílatelem a příjemcem
  - Fyzická vrstva, využito ve spojovaných sítích
- Přepínání paketu (Packet Switching)
  - Zaslání nezávislých datových jednotek (paketů)
  - Může být spojovaná i nespojovaná

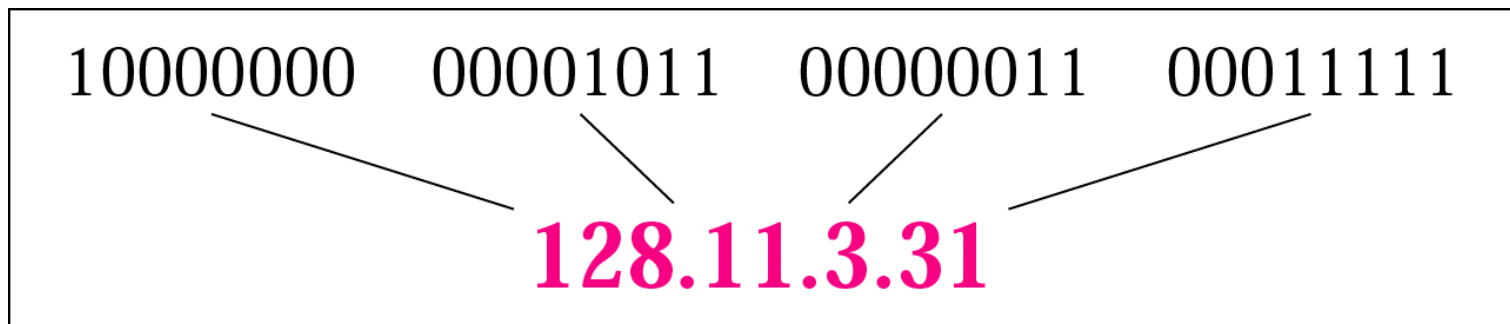


# Protokol IP

- Zajišťuje směrování paketů (datagramů) na základě IP adres
  - Nejen v rámci jedné sítě
- Každý paket obsahuje informace o zdroji a cíli
- Zodpovědnost za pořadí a správné doručení má vyšší vrstva (TCP)
- Výběr optimální cesty v síti
- IPv4 a IPv6

# Adresace IP

- Požadavek jednoznačné identifikace každého zařízení připojeného k Internetu
- Nutnost systematického přidělování adres za účelem snadnějšího směrování
- Každému zařízení/rozhraní přiřazena IP adresa
  - IPv4 adresa (32 bitů) vs. IPv6 adresa (128 bitů)





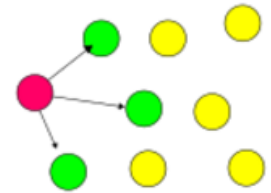
# IPv6 – struktura adres

- Typicky se skládá z prefixu a adresy hosta (64b, 64b)
  - Adresa hosta je buď MAC adresa jeho síťové karty nebo je přiřazena jiným způsobem
- Notace
  - Skupiny hexadecimálních čísel
  - 2001:0db8:85a3:08d3:1319:8a2e:0370:7334
  - 2001:0db8:0000:0000:0000:0000:1428:57ab
  - 2001:0db8:0000:0000:0000::1428:57ab
  - 2001:0db8:0:0:0:0:1428:57ab
  - 2001:0db8:0:0::1428:57ab
  - 2001:0db8::1428:57ab
  - 2001:db8::1428:57ab
  - Poslední část adresy může obsahovat IPv4 adresu
    - ::ffff:12.34.56.78 (z důvodů smíšeného prostředí)

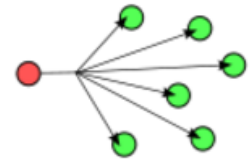
# Typy adres

- Individuální (unicast) adresy
  - Identifikace jediného odesílatele/příjemce
- Broadcast adresy
  - Slouží pro zaslání dat všem možným příjemcům na dané LAN
- Skupinové (multicast) adresy
  - Slouží pro adresování skupiny příjemců, kteří o data projevili zájem
  - Data směrovači rozaslána všem členům skupiny
- Výběrové (anycast) adresy (IPv6)
  - Data se doručí jen jedinému členu ze skupiny příjemců – tomu, který je nejbližší

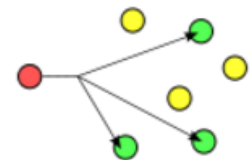
anycast



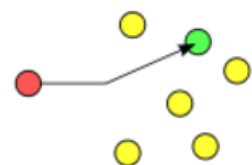
broadcast



multicast



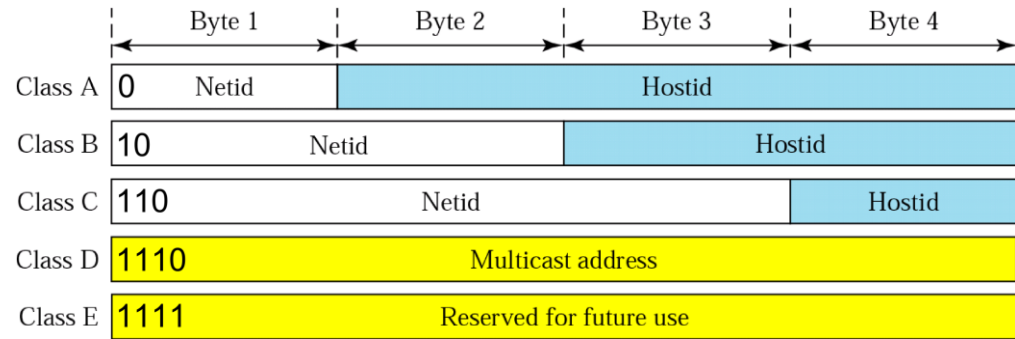
unicast



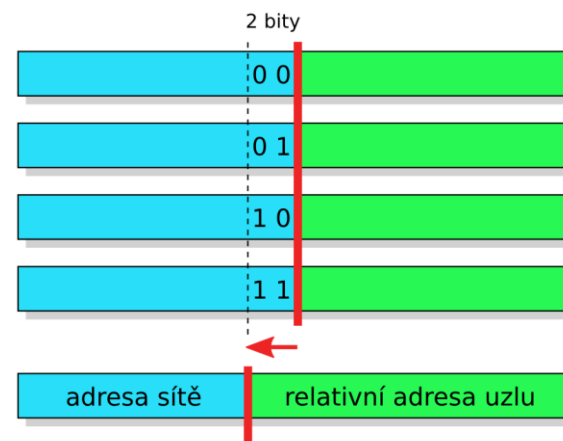


# Přidělování adres (1)

- Classful Addressing
  - Rozdělení do tříd
    - Již se nepoužívá
  - Subnetting
  - Supernetting



(a) Subnetting



(b) Supernetting

# Maska sítě

- Oba způsoby vyžadují mechanismus pro identifikaci bitů, které identifikují síť
- Masky sítě
  - 32-bitový řetězec (v rámci IPv4)
  - Obsahuje 1 v těch bitech, které odpovídají síťové části adresy, 0 tam, kde jde o relativní adresu uzlu v rámci sítě
- IP adresa uzlu && maska sítě = adresa sítě

| Class | Binary form                                | Decimal form           | Using slash |
|-------|--|------------------------|-------------|
| A     | <b>11111111</b> 00000000 00000000 00000000 | <b>255.0.0.0</b>       | /8          |
| B     | <b>11111111 11111111</b> 00000000 00000000 | <b>255.255.0.0</b>     | /16         |
| C     | <b>11111111 11111111 11111111</b> 00000000 | <b>255.255.255.0</b>   | /24         |
| ---   | <b>11111111 11111000</b> 00000000 00000000 | <b>255.248.0.0</b>     | /13         |
| ---   | <b>11111111 11111111 11111111 10000000</b> | <b>255.255.255.128</b> | /25         |

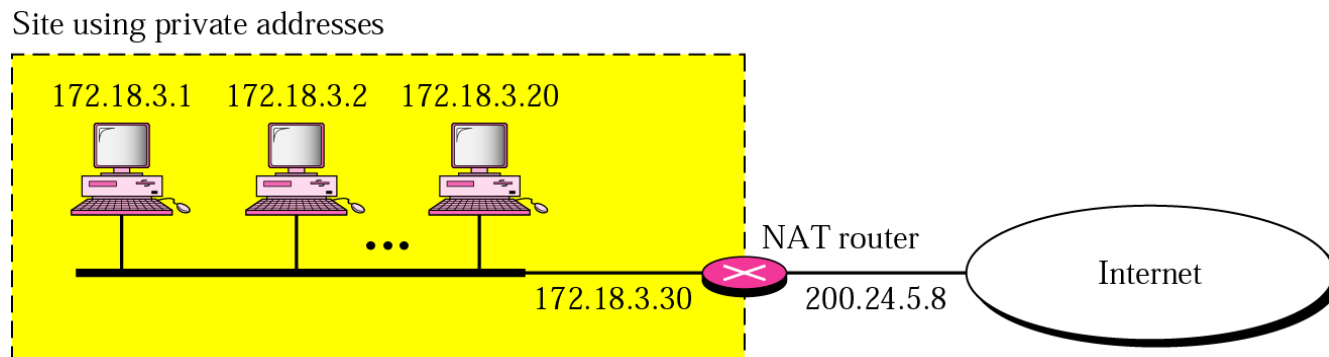
# Přidělování adres (2)

---

- Classless Addressing
  - Zobecnění a rozšíření subnettingu/supernettingu
  - Zavádí zcela variabilní délku bloku adresy sítě
    - Identifikace sítě = adresa sítě a maska sítě
  - Adresy se přidělují hierarchicky
    - Umožnění agregace směrování
    - Snaha o minimalizaci velikosti směrovacích tabulek
  - Opodstatnění subnettingu zůstává

# NAT (1)

- Překlad adres, síťová maškaráda
  - Způsob adresování ve vnitřní síti
  - Router prepisuje zdrojovou nebo cílovou adresu
  - Využívá rezervované privátní adresy
  - Ve směrovací tabulce si udržuje informace o spojení a odpověď správně předá

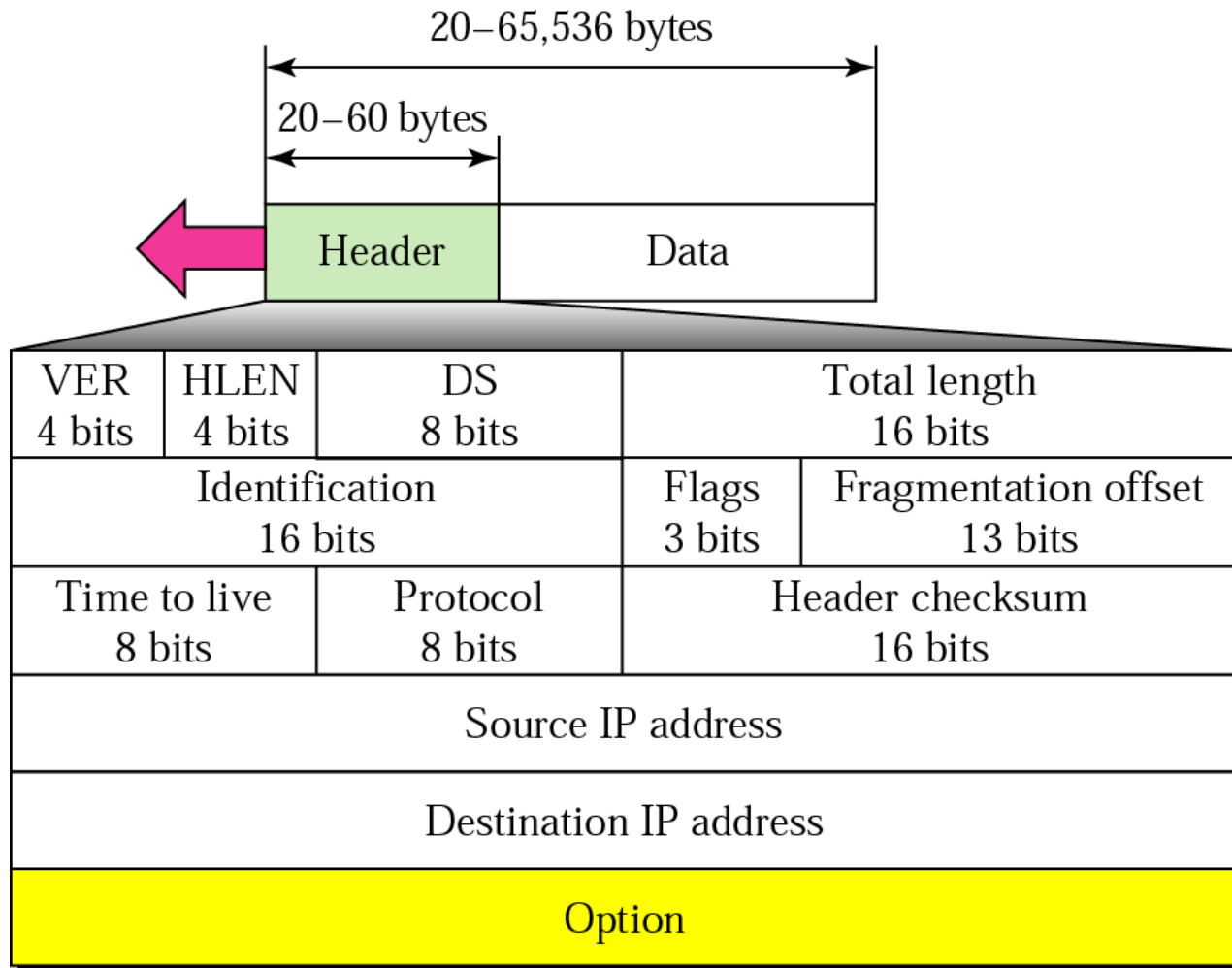


# NAT (2)

- Umožňuje připojit více počítačů za jednu veřejnou IP
- Zvyšuje bezpečnost počítačů za NATem
- Problémy s příchozím provozem (FTP, HTTPd)
- Využívá překladové tabulky

| <i>Private Address</i> | <i>Private Port</i> | <i>External Address</i> | <i>External Port</i> | <i>Transport Protocol</i> |
|------------------------|---------------------|-------------------------|----------------------|---------------------------|
| 172.18.3.1             | 1400                | 25.8.3.2                | 80                   | TCP                       |
| 172.18.3.2             | 1401                | 25.8.3.2                | 80                   | TCP                       |
| ...                    | ...                 | ...                     | ...                  | ...                       |

# Struktura datagramu





# Fragmentace datagramů (1)

- Datagram při cestě k cíli prochází různými sítěmi
  - Všechny sítě (resp. využitě protokoly linkové vrstvy) **nemohou** přenášet data stejné velikosti
- Maximum Transfer Unit (MTU)
  - Maximální velikost dat, které lze přenést využitým protokolem nižší vrstvy
  - Určuje maximální velikost přenositelného IP datagramu (Total size)

## Fragmentace datagramů (2)

- Zdrojový uzel chce odeslat datagram, který je větší než MTU výstupní linky
- Směrovač přijme datagram, který je větší než MTU výstupní linky
- Lze řešit provedením tzv. fragmentace IP datagramu
- Původní datagram je rozdělen na několik menších datagramů (tzv. fragmenty)
- Každý fragment získá svou vlastní IP hlavičku
  - Stane se z něj nový, plnohodnotný datagram



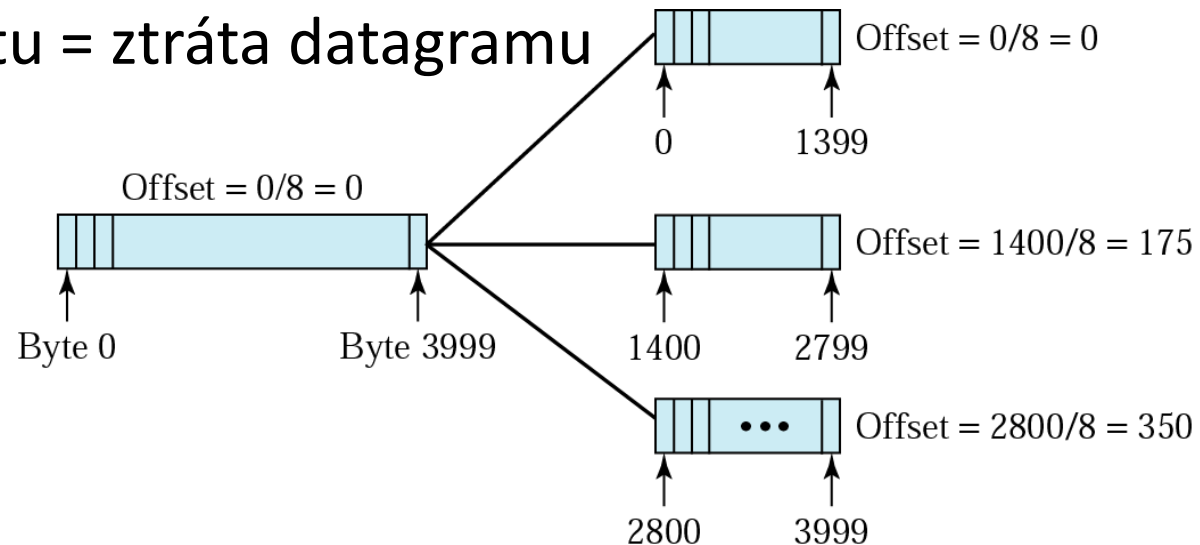
# Fragmentace datagramů (3)

---

- Fragmenty na cílovém uzlu jsou složeny do původního datagramu
  - Před předáním transportnímu protokolu
- Složení fragmentu do původního datagramu vyžaduje
  - Identikaci datagramu, kterému fragmenty náleží
  - Znalost počtu fragmentů
  - Znalost pozice každého fragmentu v původním datagramu

# Fragmentace datagramů (4)

- Kde se fragmentace provádí?
  - Na zdrojovém uzlu
  - Na směrovači/směrovačích
- Kde se provádí skládání fragmentu?
  - Pouze na cílovém uzlu
  - Ztráta fragmentu = ztráta datagramu



# IPv6 – motivace

---

- Prostor IPv4 dochází nebo už došel (Asie)
- Problémy IPv4
  - Slabá podpora přenosu aplikací reálného času
  - Žádná podpora zabezpečené komunikace na úrovni IP
  - Žádná podpora autokonfigurace zařízení
  - Žádná podpora mobility

# IPv6

- Změny

- Ohromný adresní prostor

- 128b adresa
- $2^{128}$  adres –  $3,4 \times 10^{38}$

- Jednodušší hlavička – 40B

- Podpora přenosů reálného času – značkování toku, prioritizace provozu

- Podpora zabezpečení přenosu – podpora autentizace, šifrování a verifikace integrity přenášených dat

- Podpora mobility – skrze tzv. domácí agenty

- Podpora autokonfigurace zařízení

- Nejsou nutné drastické změny v aplikacích

| Verze              | Třída provozu | Značka toku    |            | bitů |
|--------------------|---------------|----------------|------------|------|
| 8                  | 8             | 8              | 8          |      |
|                    | Délka dat     | Další hlavička | Max. skoků |      |
| Adresa odesílatele |               |                |            |      |
| Cílová adresa      |               |                |            |      |

# Směrování

- Proces nalezení cesty v počítačové síti mezi dvěma uzly
  - Co možná nejefektivnější cesta pro doručení paketu
  - Cesta musí splňovat určité omezující podmínky
  - Je ovlivněna topologií a zátěží sítě
- V rámci směrování se řeší komu paket poslat dál, nikoliv celá cesta
  - Někomu blíže k cíli
  - A ten pak rozhoduje, co s paketem dál
- Analogie při rozhodování na křižovatkách
- Směrovače mají směrovací tabulky
- Na směrování lze nahlížet jako na problém teorie grafů

# Cena komunikace

- Určení ceny (ohodnocení) linky – metrika
- Všechny linky mají stejnou cenu
  - Minimalizace ceny = minimalizace počtu skoků
  - Nejjednodušší, nejčastěji využívané
- Cena linky = převrácená hodnota kapacity
- Cena linky = zpoždění linky
- Cena linky = využití linky
- Cena linky = reálna cena (platba) za využití linky
  - Staticky přiřazeno administrátorem
- Atd.

# Směrovací tabulka

- Základní datová struktura
  - Sada ukazatelů, podle kterých se rozhoduje, co udělat s kterým paketem
  - Obsahuje cesty k prefixům (počáteční IP adresa a blok)
- Agregace záznamu
  - Hledá se nejdelší prefix, který vyhovuje požadavku

|                 | Mask | Destination address | Next-hop address | Interface |
|-----------------|------|---------------------|------------------|-----------|
|                 | /8   | 14.0.0.0            | 118.45.23.8      | m1        |
| Host-specific → | /32  | 192.16.7.1          | 202.45.9.3       | m0        |
|                 | /22  | 193.14.4.0          | 84.12.6.20       | m1        |
|                 | /24  | 193.14.5.0          | 84.78.4.12       | m2        |
| Default →       | /0   | /0                  | 145.11.10.6      | m0        |



# Problém globálního pohledu

- Získání globální znalosti topologie celé sítě je problematické
  - Když už se to podaří, není aktuální
  - Musí být lokálně relevantní
- Lokální představu o topologii reprezentuje směrovací tabulka
- Rozpor mezi lokální a globální znalostí může způsobit
  - Cykly (černé díry)
  - Oscilace (adaptace na zátěž)





# Směrování – základní přístupy

- Statické
  - Směrovací tabulky jsou pevně (ručně) dané
  - Jednodušší, ale málo flexibilní
- Dynamické
  - Směrovací tabulky se upravují podle topologie sítě
  - Nutné pravidelné aktualizace směrovacích tabulek
  - Nezaručuje pořadí doručení
  - Síť musí poskytovat informace o svém stavu
    - Centralizovaně
    - Izolovaně
    - Distribuovaně

# Centralizované směrování

- V síti je Routing Control Center (RCC)
  - Každý směrovač mu posílá zprávy o své situaci (stavu)
  - RCC informace sbírá, vypočte optimální cesty a rozešle směrovačům jejich tabulky
- Výhody
  - Globální informace = optimální řešení
  - Ulehčení práce směrovačů
- Nevýhody
  - Špatně škáluje – nelze využít pro velké sítě
  - Pomalé
  - Při výpadku centra se přestane aktualizovat

# Izolované směrování

- Neposílají se žádné informace o stavu sítě, každý se rozhoduje sám za sebe
- Příklady
  - Náhodná procházka – paket pošle do náhodně vybrané linky
  - Horký brambor (hot potatoe) – paket pošle do linky s nejkratší frontou
  - Záplava (flooding) – paket pošle do všech linek kromě té, po níž přišel
  - Zpětné učení (backward learning) – učí se z procházejících paketů
    - Směrovač se dozví, že příchozí linkou vede cesta k odesílateli nanejvýš dané délky



# Distribuované směrování

- Směrovací informace si vyměňují sousedé či malé skupiny směrovačů
- Na základě periodicky šířených informací se (podle určitého algoritmu) vypočítávají mapy sítě
- Mezi směrovači musí být dohoda o implementaci určitého směrovacího algoritmu
- Dostatečně pružné a robustní, vhodné i pro rozlehlé sítě
- Standardní přístup ke směrování v síti Internet

# Směrování v Internetu

distribuované

vs. centralizované

"krok za krokem"

vs. zdrojové

deterministické

vs. stochastické

jedno

vs. více cestné

dynamický

vs. statický výběr cest

**INTERNET**

# Transportní vrstva (1)

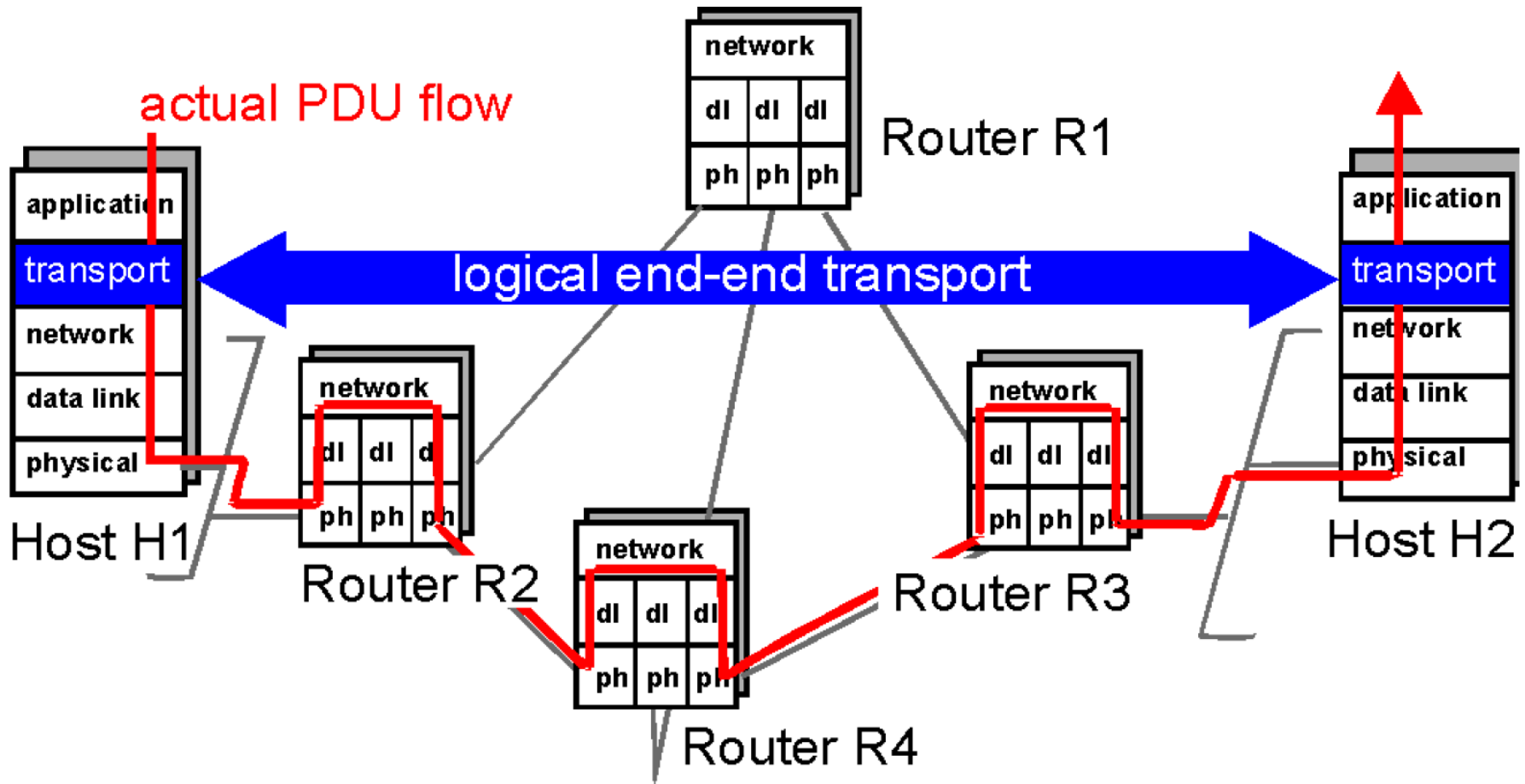
---

- Proč nám nestačí síťová vrstva?
  - Nelze identifikovat aplikaci, které jsou data určena
    - Na každém uzlu by tak mohla jet jen jedna aplikace
  - Neřeší defekty sítě (ztrátu/znásobení datagramu, zahlcení sítě, atp.)
- Dva protokoly na transportní vrstvě
  - TCP – Transport Control Protocol
  - UDP – User Datagram Protocol

# Transportní vrstva (2)

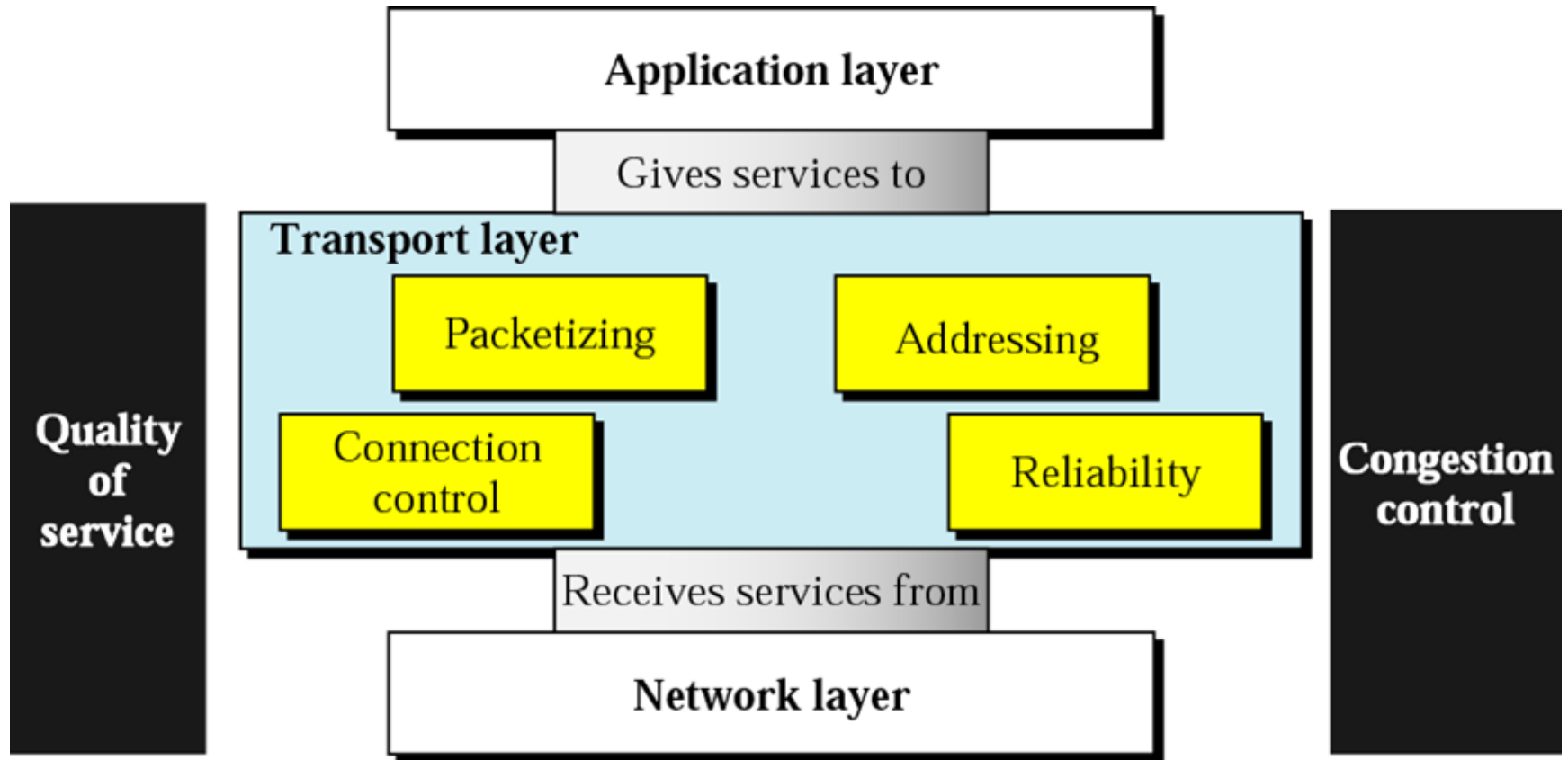
- Poskytuje služby pro aplikační vrstvu
  - Přijímá data odesílací aplikace, které transformuje do segmentu
  - Přijaté segmenty pak předává cílové aplikaci
- Ve spolupráci se síťovou vrstvou zajišťuje doručení dat (segmentů) mezi komunikujícími aplikacemi/procesy
  - S případným zajištěním spolehlivosti přenosu
  - Poskytuje jim logický komunikační kanál
  - Tzv. process-to-process delivery
- Nejnižší vrstva poskytující tzv. end-to-end služby
  - Hlavičky generované na straně odesílatele jsou interpretovány jen na straně příjemce
  - Směrovače vidí data transportní vrstvy jako payload přenášených paketů

# Transportní vrstva (3)



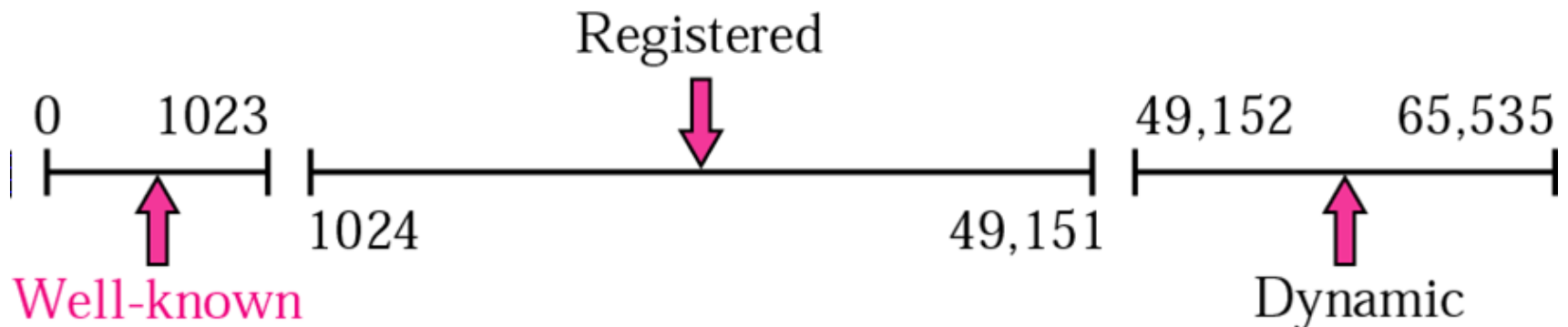


# Služby transportní vrstvy

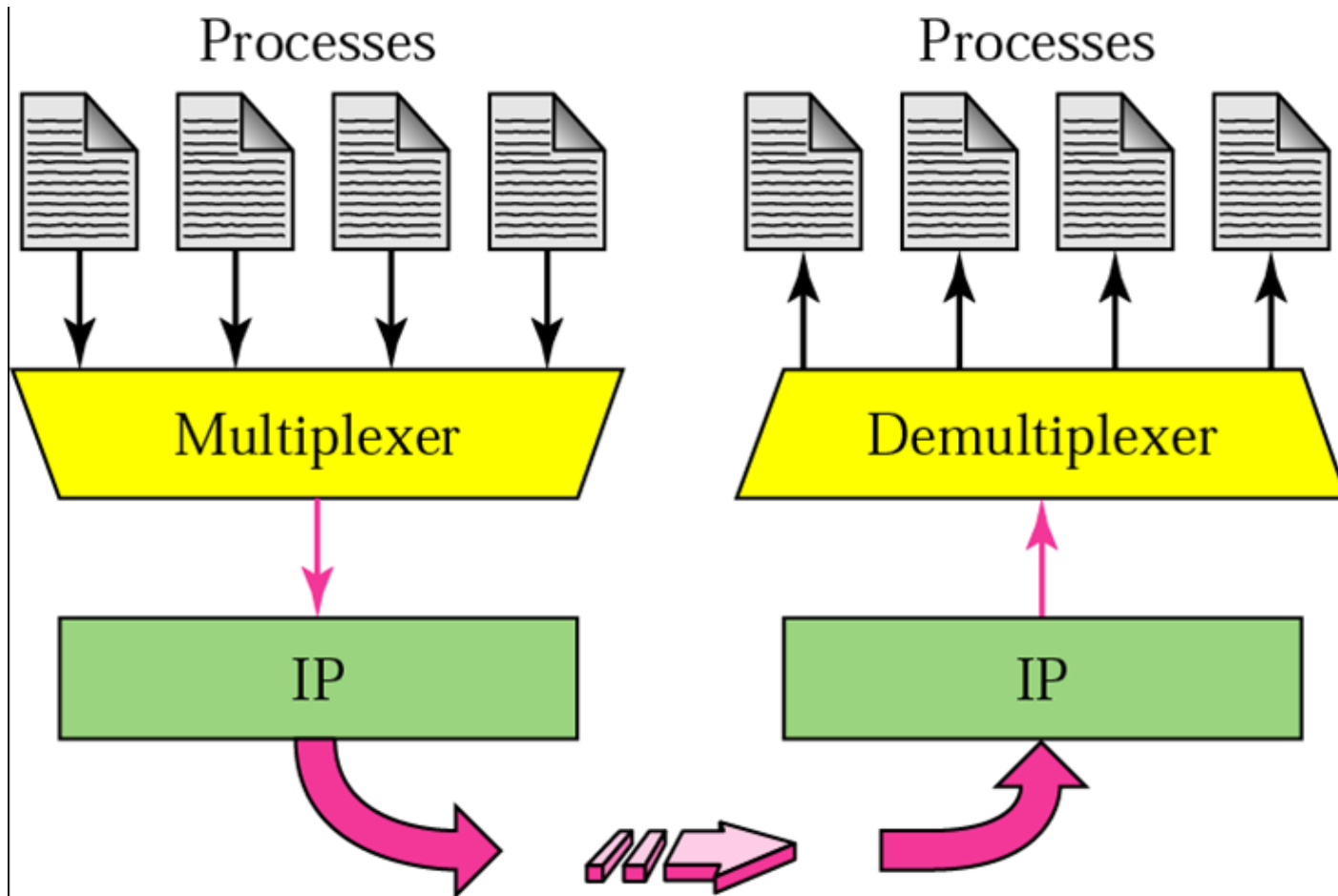


# Adresace transportní vrstvy

- Adresy na transportní vrstvě – čísla portů (ports, port numbers)
  - Neboli adresy služeb
  - Identifikují odesílací aplikaci na zdrojovém uzlu (identifikován IP adresou)
  - Identifikují přijímající aplikaci na cílovém uzlu (identifikován IP adresou)
- Identifikace portu 16bitovým číslem
- Rozsah 0-65535



# Adresace – Multiplexing vs. Demultiplexing

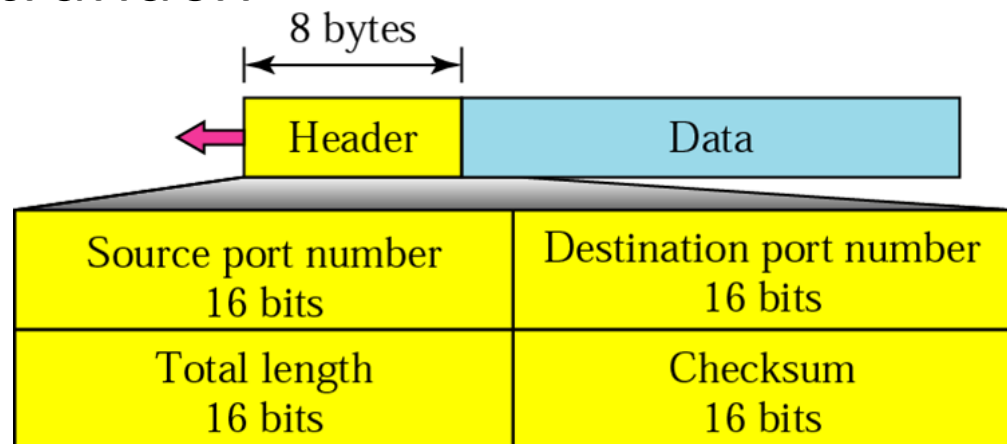


# UDP

- User datagram protokol
- „Nespolehlivá“ transportní služba
- Využitelné pro aplikace, které nevyžadují spolehlivost přenosu
  - Streamované video, rádio, videokonference
  - DHCP, DNS
- Použití portů pro rozlišení různých aplikací

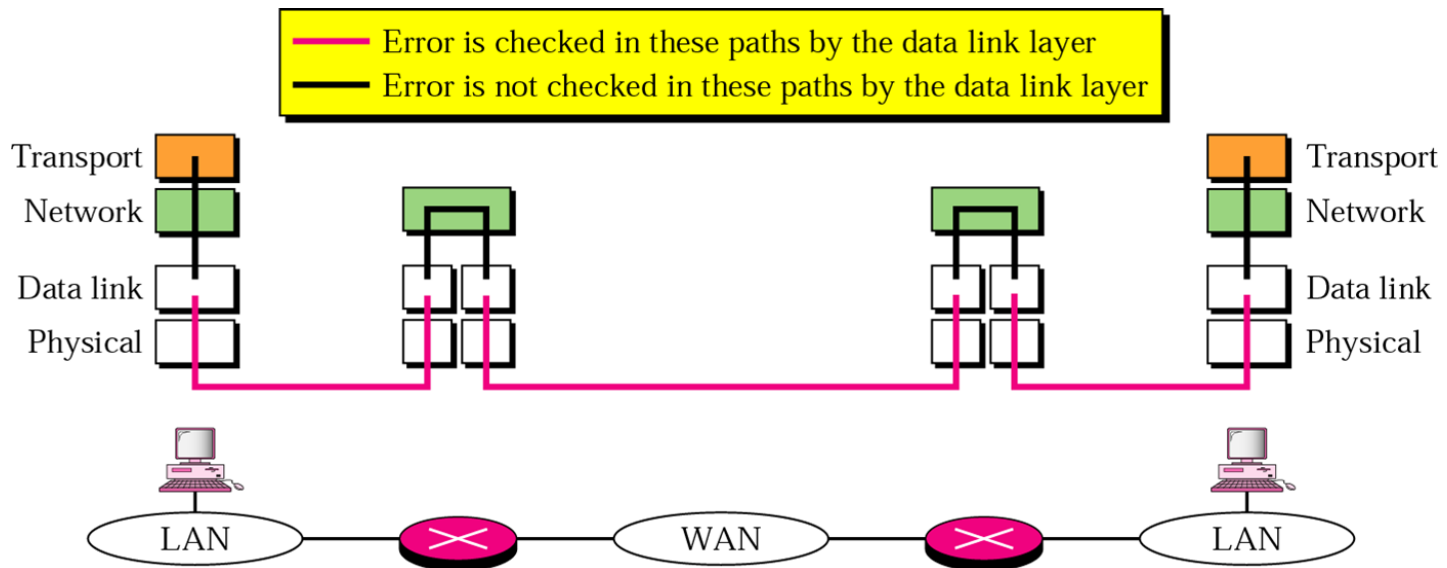
# Přednosti UDP

- Nemá fázi navázání a ukončení spojení
- Žádná nutnost ustavení spojení (přináší zpoždění na začátku přenosu)
- Žádná nutnost uchovávání stavových informací na komunikujících stranách
- Malá hlavička



# Zajištění spolehlivého přenosu (1)

- K čemu je řízení chyb na transportní vrstvě, když už je toto poskytováno linkovou vrstvou?
  - Linková vrstva poskytuje řízení chyb vždy pouze mezi dvěma uzly na cestě, ne mezi koncovými stanicemi



# Zajištění spolehlivého přenosu (2)

- Spolehlivost přenosu zajištěna mechanismem potvrzování (acknowledgement)
  - Pakety číslovány tzv. sekvenčními čísly (Sequence Numbers, SEQ)
  - Pozitivní potvrzování (positive acknowledgement)
    - Potvrzení úspěšného přijetí paketu (přijato v pořádku)
  - Negativní potvrzování (negative acknowledgement)
    - Informace o neúspěšném přijetí/ztrátě datagramu (zopakuj)
- V případě výskytu chyby jsou data opětovně přeposlána
  - Mechanismy ARQ (Automatic Repeat reQuest)
  - Nutnost vypořádat se s duplicitami!

# TCP (1)

- Transportní protokol poskytující spojovanou a plně spolehlivou (= zajištěnou) službu
  - Pokud je to možné, odesílaná data budou přijímající aplikací doručena kompletní a ve správném pořadí
  - Oproti UDP orientován na přenos proudu bytů
- Multiplexing/demultiplexing a detekce chyb stejně jako v UDP
- Rozlišování aplikací pomocí portů (80, 21, 22)
- TCP neřeší bezpečnost přenosu dat

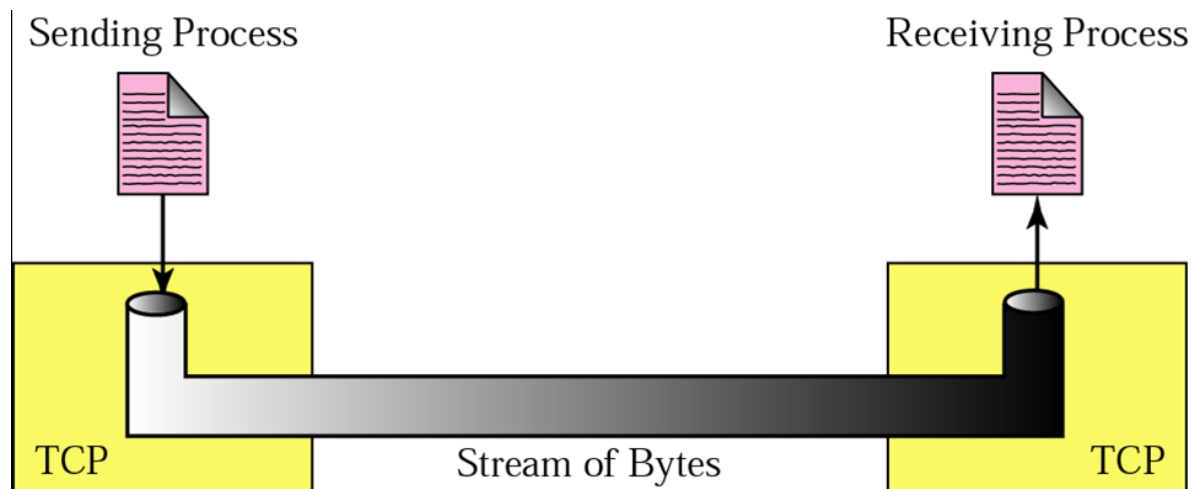


# TCP (2)

- Před začátkem přenosu nutnost ustavení spojení mezi odesílací a přijímající stranou
  - Tzv. handshake před začátkem přenosu zahrnuje výměnu všech potřebných parametrů
  - Spojení rozeznatelné jen na koncových uzlech (end-to-end služba)
    - Směrovače tato spojení nevidí
  - Ustavené spojení možno využít pro plně duplexní komunikaci
    - Řídící data přibalována do dat jdoucích opačným směrem (Piggybacking)
  - Spojení může být pouze dvoubodové (point-to-point)
    - Komunikace mezi více partnery není podporována

# Přenos dat v rámci TCP

- Aplikace předává TCP protokolu proud bytů, které TCP segmentuje, opatřuje hlavičkou a předává síťovému protokolu
- Aplikacím poskytuje iluzi roury, která přenáší jejich data

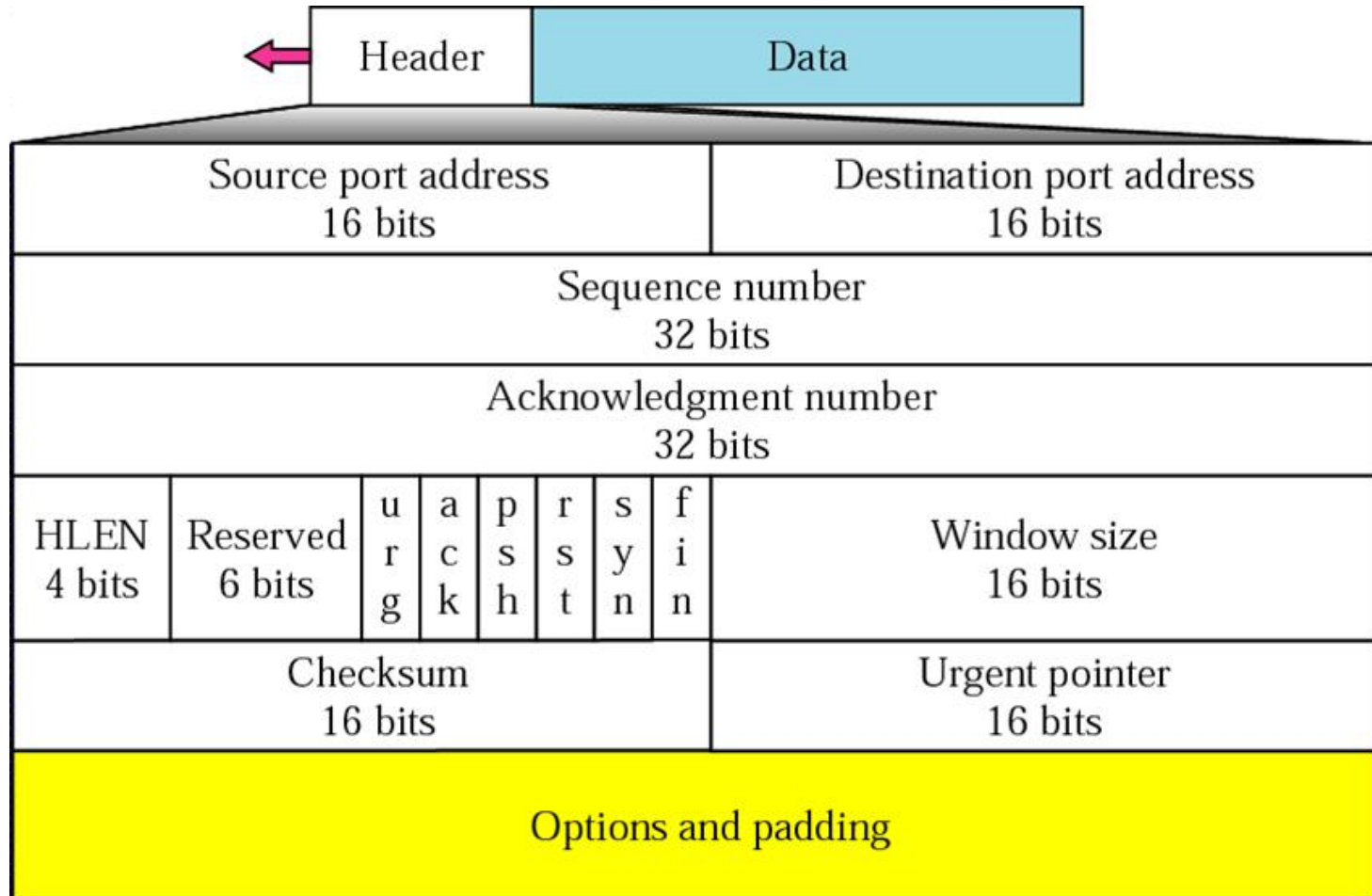




# TCP – segmentace dat

- Aplikace TCP protokolu předává proud bytů
- Síťová vrstva (IP protokol) očekává bloky dat
  - Nutnost tvorby bloku dat (segmentu)
  - Velikost segmentu omezena hodnotou Maximum Segment Size (MSS)
    - Identifikuje maximální velikost uživatelských dat v segmentu (ne velikost celého segmentu)
  - Segmenty následně opatřeny TCP hlavičkou a předány síťovému protokolu
  - Číslovány nejsou bloky dat (segmenty), ale jednotlivé přenášené bajty

# TCP hlavička segmentů

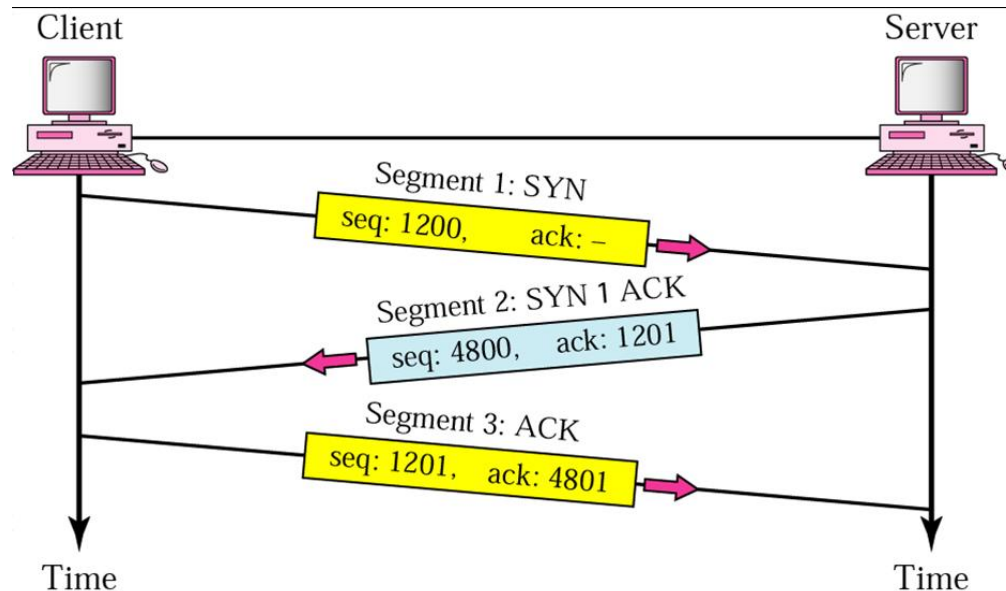


# Well-known TCP aplikace

| Port | Protocol     | Description                                   |
|------|--------------|---|
| 7    | Echo         | Echoes a received datagram back to the sender |
| 9    | Discard      | Discards any datagram that is received        |
| 11   | Users        | Active users                                  |
| 13   | Daytime      | Returns the date and the time                 |
| 17   | Quote        | Returns a quote of the day                    |
| 19   | Chargen      | Returns a string of characters                |
| 20   | FTP, Data    | File Transfer Protocol (data connection)      |
| 21   | FTP, Control | File Transfer Protocol (control connection)   |
| 23   | TELNET       | Terminal Network                              |
| 25   | SMTP         | Simple Mail Transfer Protocol                 |
| 53   | DNS          | Domain Name Server                            |
| 67   | BOOTP        | Bootstrap Protocol                            |
| 79   | Finger       | Finger  |
| 80   | HTTP         | Hypertext Transfer Protocol                   |
| 111  | RPC          | Remote Procedure Call                         |

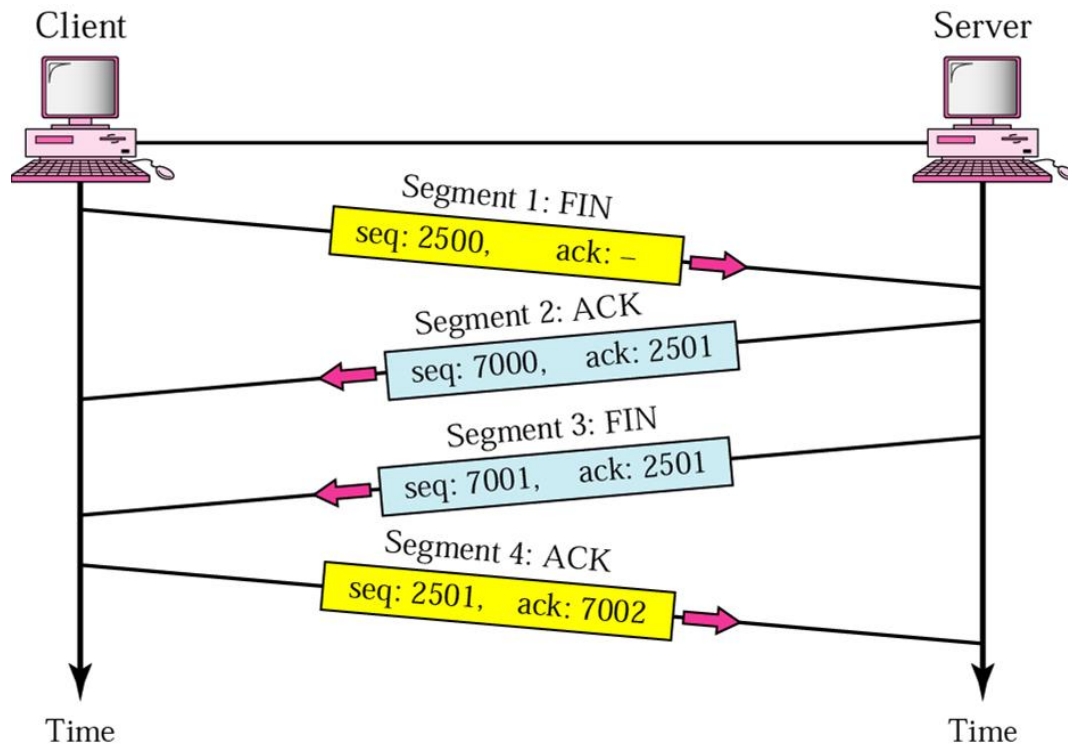
# TCP – ustavení spojení

- Full-duplexní přenos
  - Obě strany musí iniciovat spojení
- Mechanismus známý jako třicestný handshake (three-way handshake)



# TCP – ukončení spojení

- Iniciováno jednou z komunikujících stran
- Spojení musí být uzavřeno oběma stranami

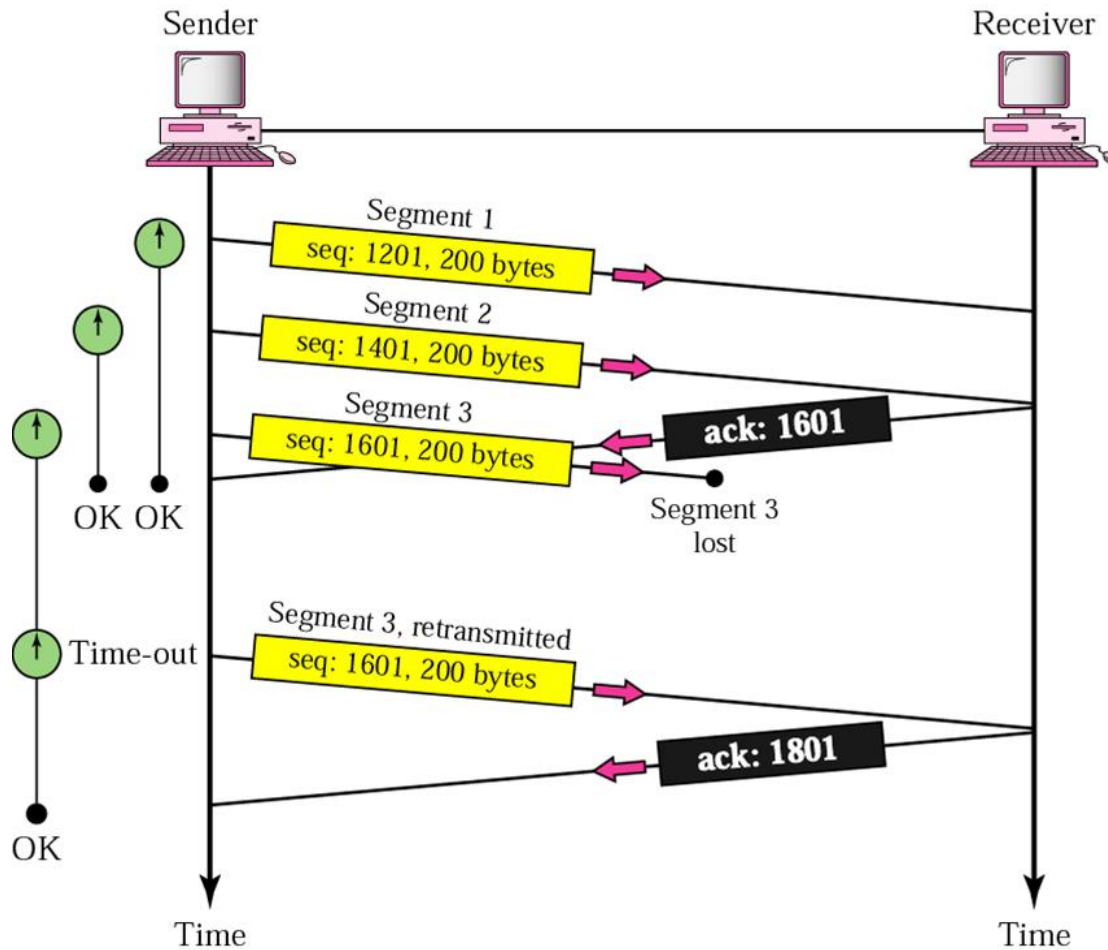


# TCP – řízení chyb

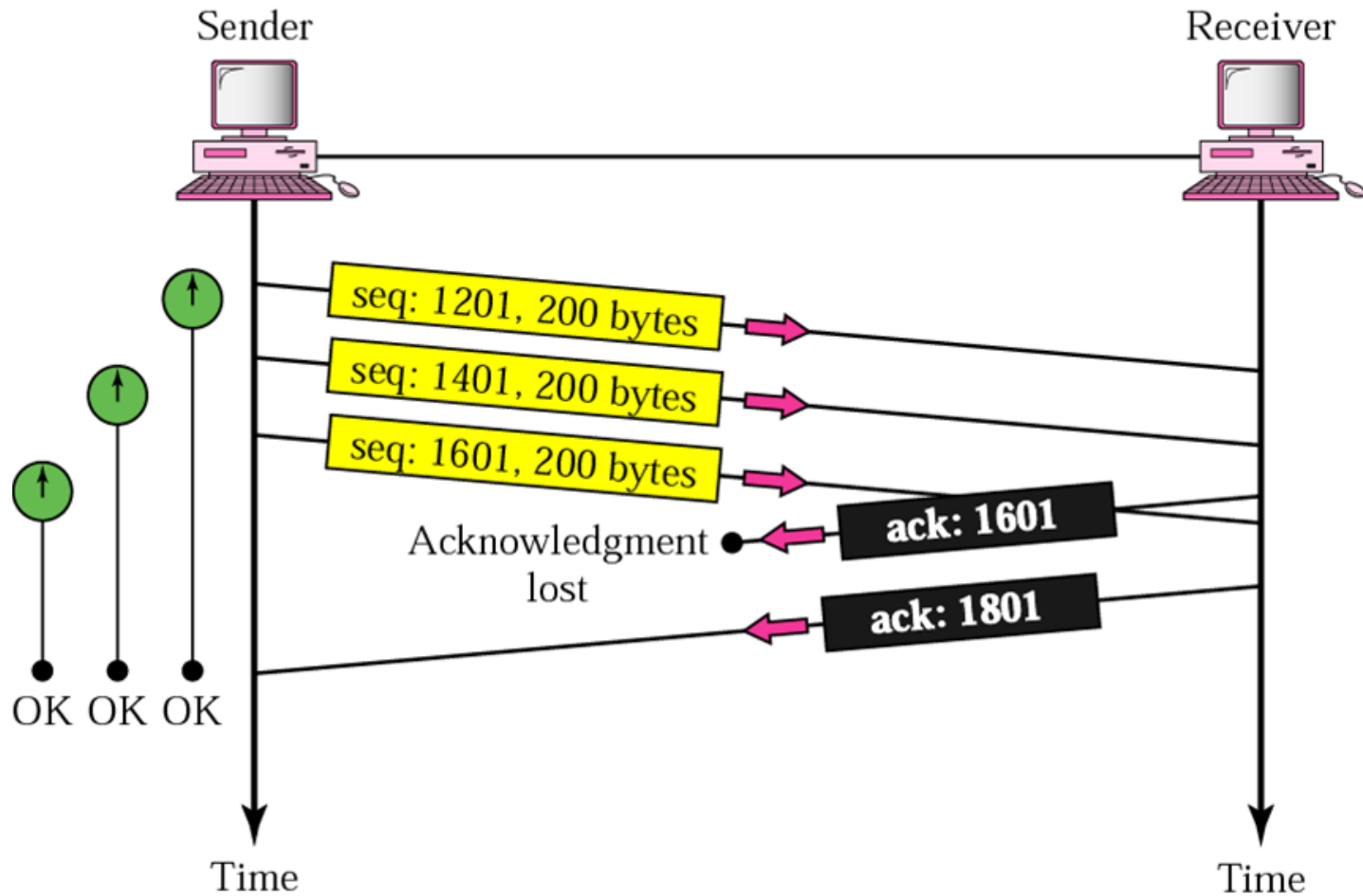
- Během přenosu je nutno detekovat poškozené, ztracené, duplikované a out-of-order segmenty
- TCP mechanismy pro zajištění spolehlivého přenosu
  - Kontrolní součty (detekce poškozených segmentů)
  - Potvrzování přijatých segmentů (acknowledgements)
    - Detekce ztracených (na straně příjemce), duplikovaných a out-of-order segmentů
    - Zajištěno mechanismem pozitivního potvrzování
    - Využito kumulativní potvrzování
  - Timeouty
    - Detekce ztracených segmentu (na straně odesílatele)



# Ztráta paketu



# Ztráta potvrzení



# Timeouty

- Timeout = doba, po kterou se čeká na potvrzení odeslaného segmentu
- Založeno na tzv. Round-Trip Time (RTT)
- Čas potřebný pro cestu segmentu od odesílatele k příjemci a zpět
- Typicky je timeout roven dvojnásobku RTT

