



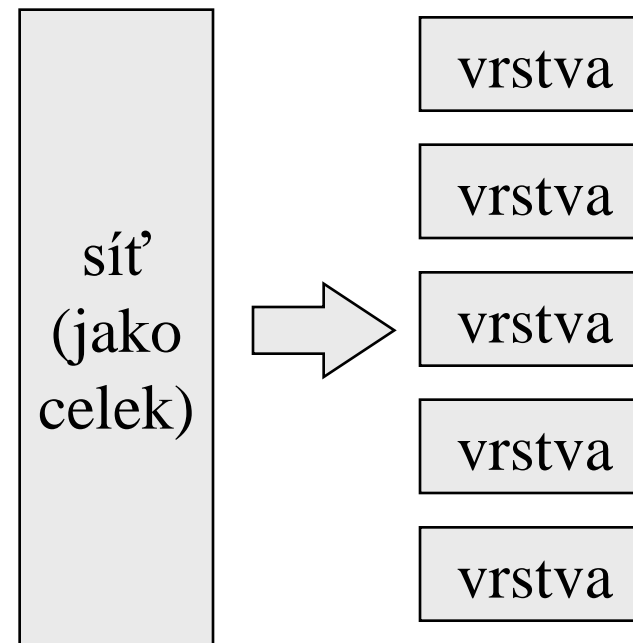
Katedra softwarového inženýrství,  
Matematicko-fyzikální fakulta,  
Univerzita Karlova, Praha



## Lekce 3: Síťové modely a architektury

# „vrstevnatá filozofie“

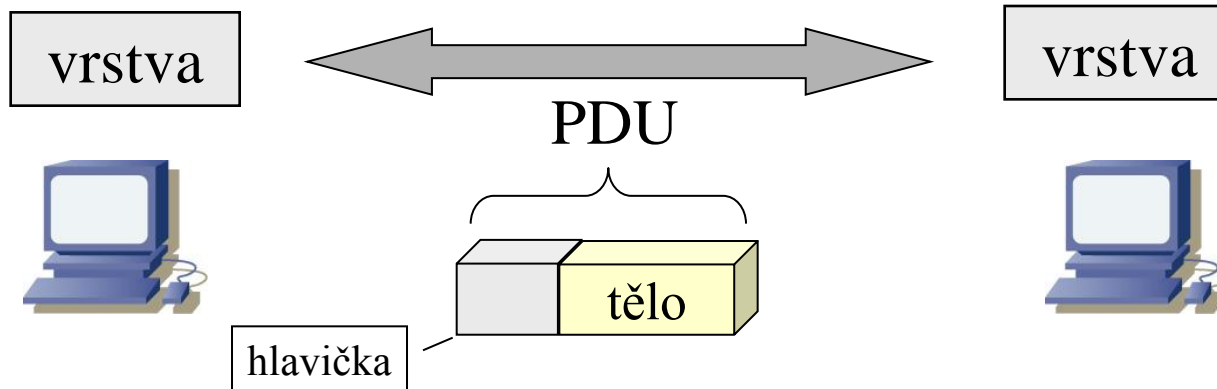
- implementovat funkční síť je hodně složité a náročné
  - stejná situace jako při řešení velkých SW celků
- jde o jeden velký problém, který se vyplatí dekomponovat
  - rozdělit na menší části, které je možné řešit samostatně
- zde: dekompozice se provede po hierarchicky uspořádaných vrstvách
  - dobře to odpovídá povaze řešeného problému
  - přináší to i další výhody
    - možnost alternativních řešení na úrovni nižších vrstev
    - větší modulárnost
    - ....



- musí se vyřešit otázky jako:
  - kolik má být vrstev
  - co má která vrstva dělat
  - jak mají vrstvy spolupracovat
    - vertikálně (v rámci uzlu)
    - horizontálně (mezi uzly)
  - .....



# "horizontální" komunikace mezi vrstvami

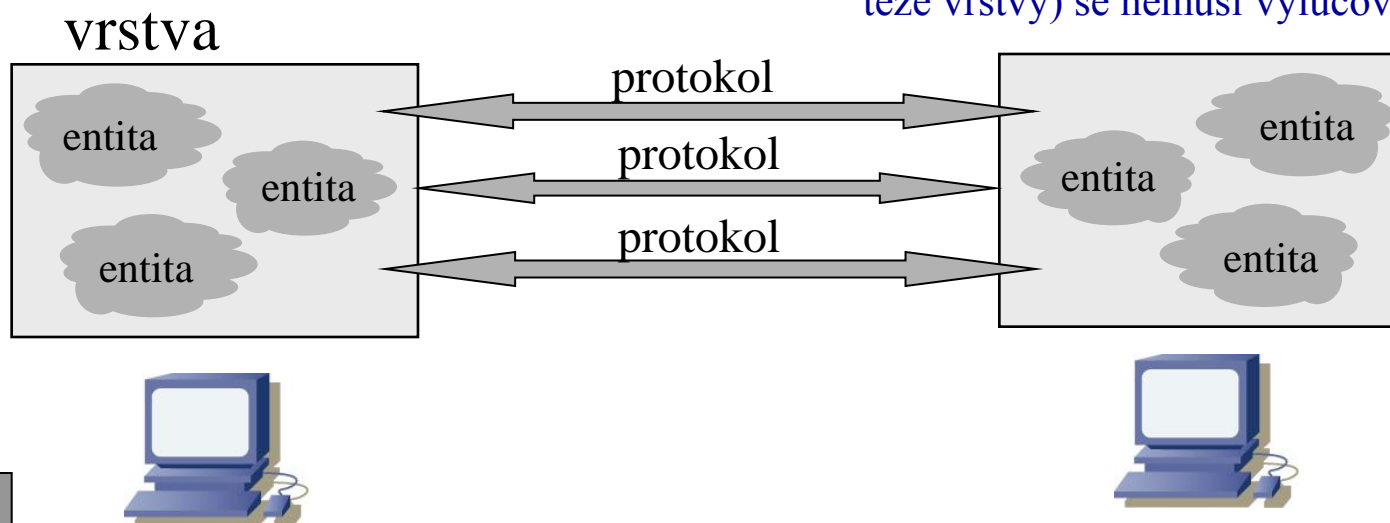


- vzájemná komunikace stejnohlých vrstev různých uzlů se musí řídit předem dohodnutými pravidly
- pravidla této vzájemné komunikace definuje tzv. **protokol**
- protokol definuje (mj.):
  - co si komunikující strany posílají, jaký to má formát a význam, kódování atd.
    - obecně: komunikující strany si předávají tzv. **PDU** (Protocol Data Unit)
    - každý PDU má dvě části: **hlavičku** (header) a **tělo** (náklad, payload)
  - jak komunikace probíhá, jak mají strany reagovat na různé situace atd.
    - ošetřuje standardní i nestandardní situace

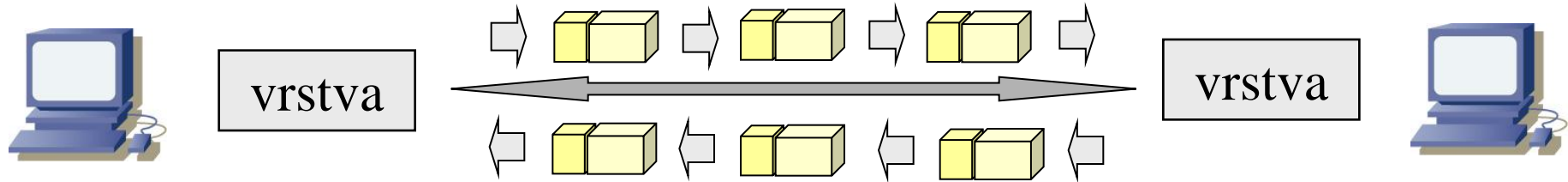
**PDU** se na úrovni různých vrstev nazývá různě:  
**rámec, buňka, paket, segment, zpráva ...**

# upřesnění: entity a protokoly

- vrstvy nejsou „jednotlivé“
- v každé vrstvě může existovat a fungovat několik relativně samostatných **entit**
  - entita může být např. proces, démon, úloha, ....
- entity ve stejné vrstvě mohou
  - plnit rozdílné funkce (nekonkurovat si)
  - plnit stejné/obdobné funkce (ale jiným způsobem, tj. konkurovat si)
- protokol definuje pravidla komunikace mezi entitami stejné vrstvy
  - každý protokol vždy „patří“ do určité konkrétní vrstvy
- protokol určuje způsob, jakým je realizována určitá služba
  - pro každou vrstvu může existovat několik alternativních protokolů
    - jeden např. pro spojovaný přenos, druhý pro nespojovaný
  - současné použití různých protokolů (v rámci téže vrstvy) se nemusí vylučovat



# "horizontální" komunikace mezi vrstvami



- komunikace mezi stejnohlými vrstvami (entitami) různých uzlů má **asynchronní charakter**

- *"pošlu své PDU a čekám na odpověď"*
- odpověď přijde "kdykoli" (nezávisle na tom, co právě dělám)

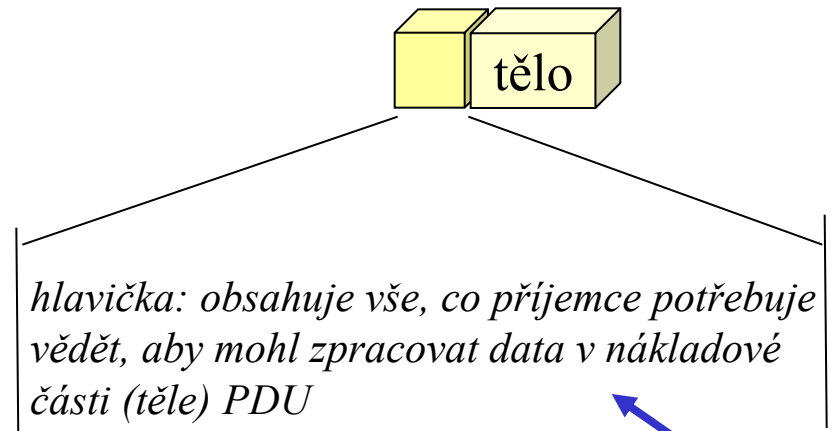
- lze přirovnat k zasílání zpráv

- pošlu zprávu a čekám na odpověď

- veškeré "režijní" informace, určené "protistraně" (partnerské entitě na druhé straně) musí být obsaženy v jednotce PDU

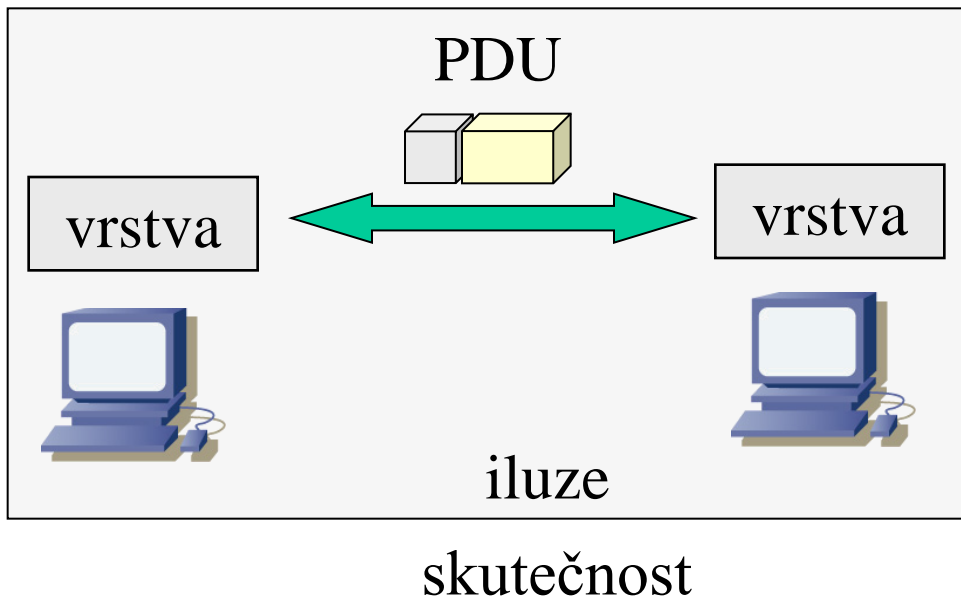
- v její hlavičce

- např. druh zasílaných dat, pořadové číslo PDU, adresa odesílatele a příjemce, kontrolní součet, příznaky, .....

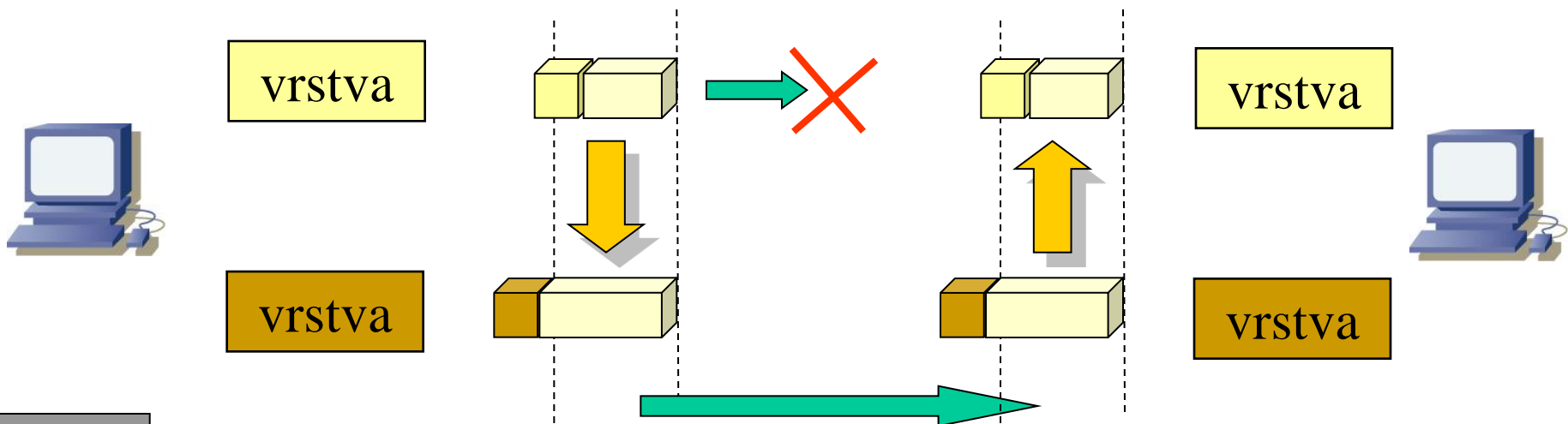


*hlavička: obsahuje vše, co příjemce potřebuje vědět, aby mohl zpracovat data v nákladové části (těle) PDU*

# "horizontální" komunikace mezi vrstvami

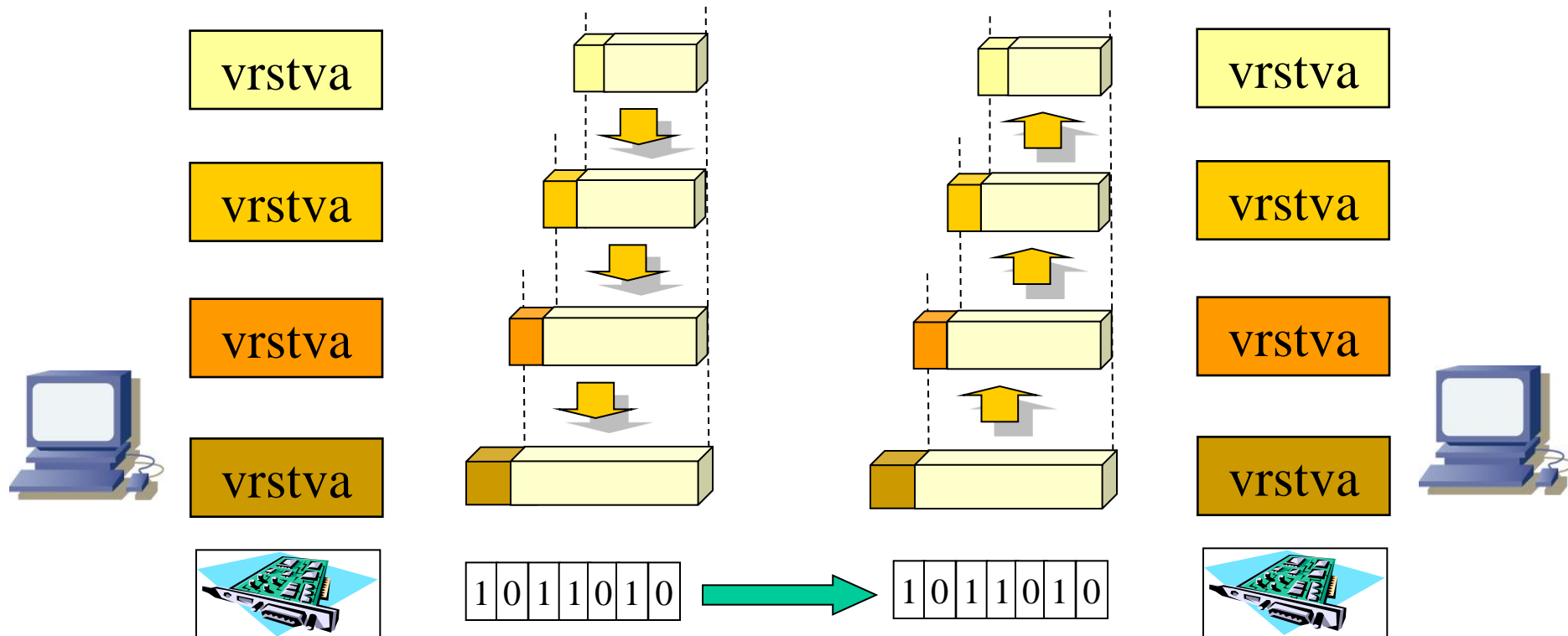


- představa, že si stejnohlé vrstvy skutečně předávají (přímo) mezi sebou jednotky PDU (*rámce, pakety, segmenty, ...*) je POUZE ILUZÍ
  - uměle navozenou fikcí
- ve skutečnosti je předávají bezprostředně nižší vrstvě, s požadavkem na doručení druhé straně !!!
  - nižší vrstva je vloží do svého PDU



# "horizontální" komunikace mezi vrstvami

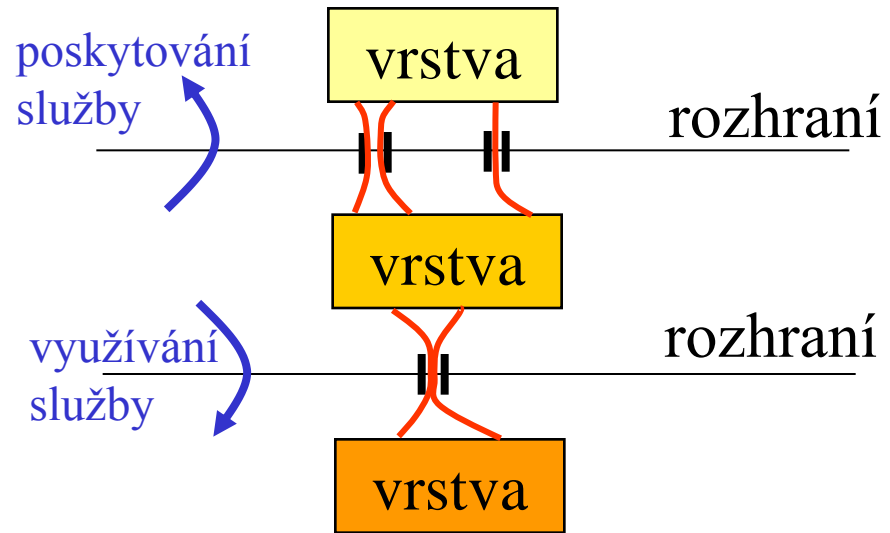
- předávání PDU (bezprostředně) nižší vrstvě k doručení pokračuje až k nejnižší vrstvě (fyzické vrstvě)
- pouze nejnižší vrstva (fyzická vrstva) skutečně přenáší nějaká data
  - po jednotlivých bitech !!!





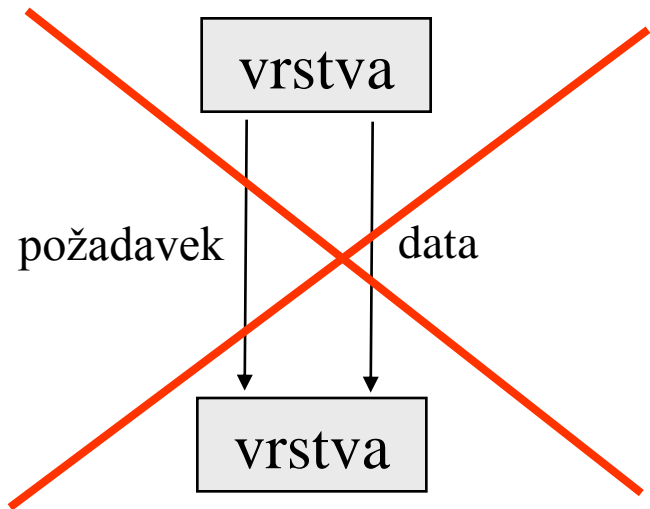
# "vertikální" komunikace mezi vrstvami

- má charakter:
  - poskytování služeb
    - vrstva poskytuje své služby bezprostředně vyšší vrstvě
  - využívání služeb
    - vrstva využívá služeb, poskytovaných bezprostředně nižší vrstvou
      - využívá je k plnění svých úkolů



- příklad:
  - nejnižší (fyzická) vrstva přenáší jednotlivé bity
  - bezprostředně vyšší (linková) vrstva využívá přenosu jednotlivých bitů k tomu, aby přenášela celé bloky dat
    - tzv. rámce (své PDU)
- mezi vrstvami je definované rozhraní
  - je implementačně závislé, není "vidět" mimo daný uzel !!!
- komunikace mezi vrstvami probíhá skrze "přechodové body" v tomto rozhraní
  - ISO/OSI: body **SAP** (Service Access Points)
  - TCP/IP: **porty**
- identifikace přechodových bodů je viditelná i "z vně" !!!
  - už nemůže být implementačně závislá !!!!!

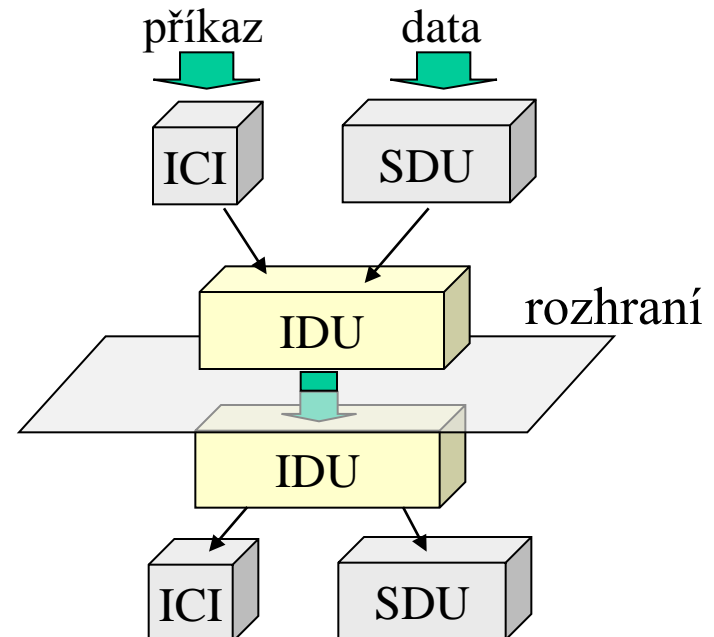
# "vertikální" komunikace mezi vrstvami



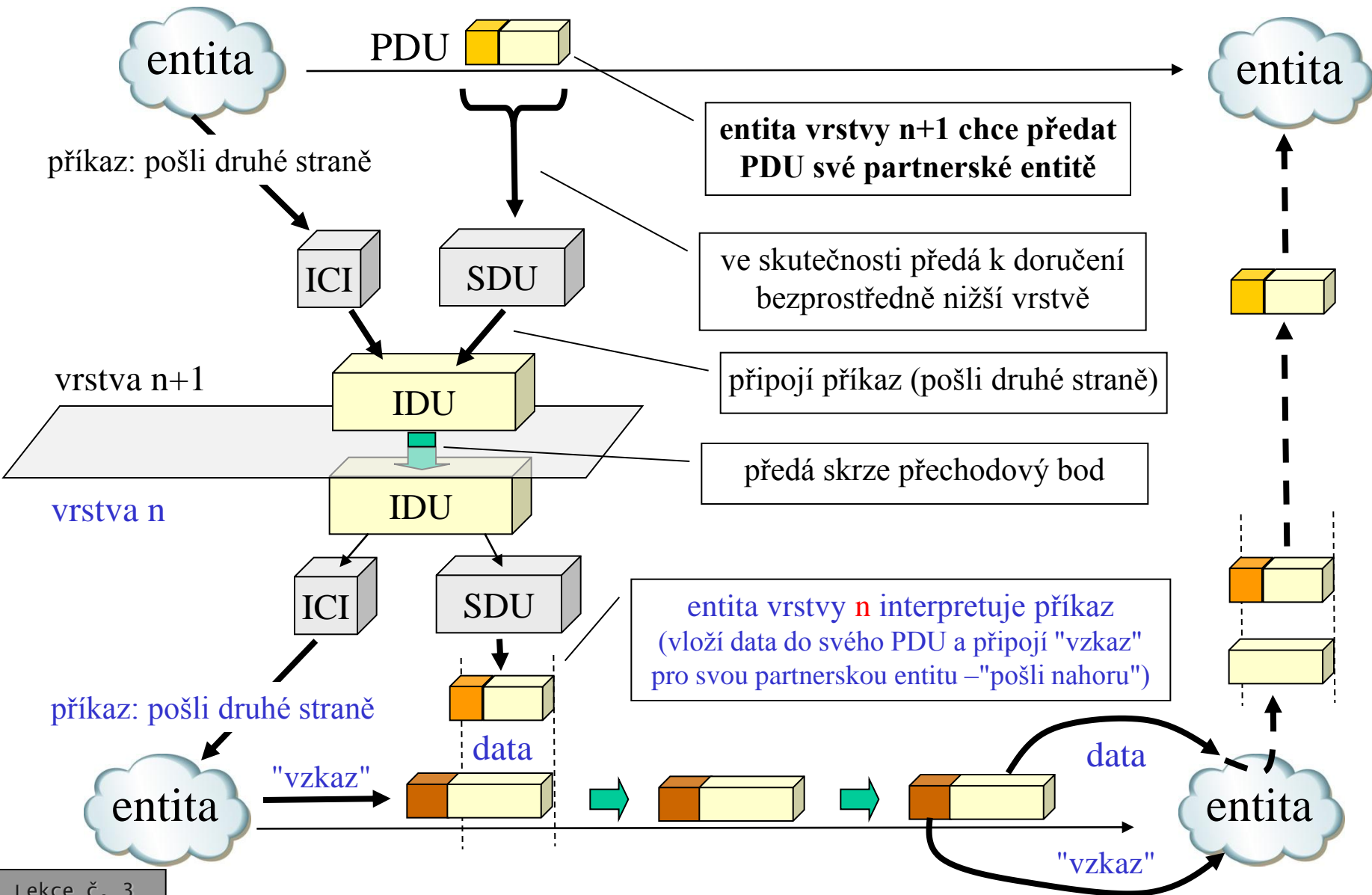
- místo toho:
  - požadavek (příkaz) se "zabalí" spolu s daty do jednoho "balíčku" (IDU) a ten se předá skrze přechodový bod
  - IDU (Interface Data Unit):
    - ICI (Interface Control Info)
    - SDU (Service Data Unit)

tzv. marshalling

- neprobíhá po 2 samostatných kanálech
  - 1x pro požadavek (příkaz, operace)
  - 1x pro data
- důvod: byly by nutné 2 přechodové body
  - nebo nějaké zdvojení přechodových bodů



# představa "vertikální" & "horizontální" komunikace



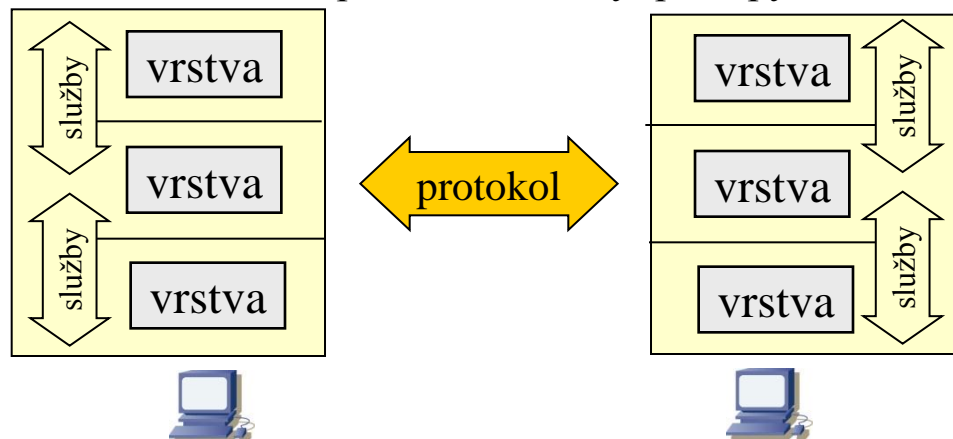
# shrnutí

## služby (angl.: services)

- týkají se vertikální komunikace
  - mezi vrstvami
  - jsou poskytovány "skrze" rozhraní mezi vrstvami
- *služby ani rozhraní nejsou "vidět" z vně daného uzlu*
  - **výjimka:** identifikace přechodových bodů, slouží k identifikaci entit v rámci vrstvy
- rozhraní mezi vrstvami nemusí být standardizováno
  - může být (a bývá) na různých uzlech/platformách různé
- nemusí být (a není) standardizován ani způsob realizace služeb
  - není potřeba, aby byly na všech uzlech realizovány stejně

## protokoly (angl.: protocols)

- týkají se "horizontální" komunikace
  - mezi stejnohlými vrstvami různých uzlů
- *protokoly jsou "vidět" z vně daného uzlu*
  - jejich implementace může zůstat skryta
- protokoly musí být standardizovány
  - musí být dopředu a všem známo, jaké jsou ...
  - pravidla, formáty, postupy atd.



# síťový model a síťová architektura

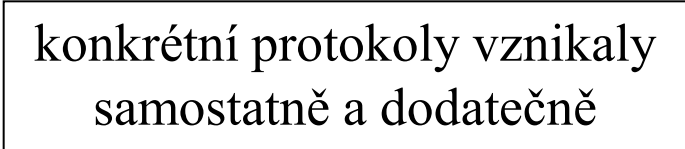
- **síťový model** je ucelená představa o tom, jak mají být sítě řešeny
  - zahrnuje:
    - představu o počtu vrstev
    - představu o tom, co má mít která vrstva na starosti
  - nezahrnuje:
    - konkrétní představu o tom, jak má která vrstva své úkoly plnit
      - tedy konkrétní protokoly
- příklad síťového modelu:
  - **referenční model ISO/OSI**
- **síťová architektura** obsahuje navíc také:
  - konkrétní protokoly
- příklad síťové architektury:
  - **rodina protokolů TCP/IP**



má 4 vrstvy



má 7 vrstev



konkrétní protokoly vznikaly samostatně a dodatečně

# referenční model ISO/OSI

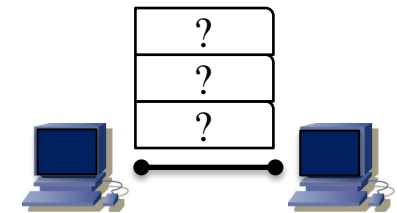
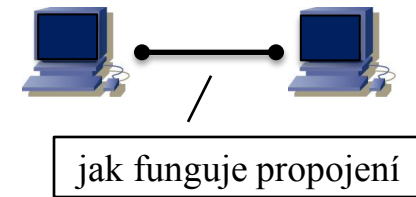
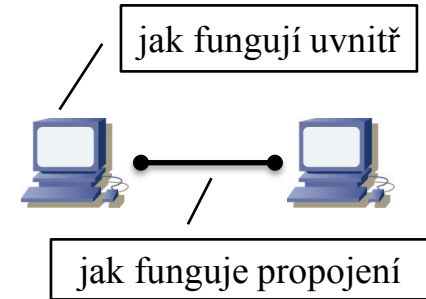
- byl pokusem vytvořit univerzální síťovou architekturu,
  - skončil jako síťový model
    - bez protokolů, ty se dodělávaly postupně
- pochází „ze světa spojů“
  - od organizace ISO (International Standards Organization, správně: International Organization for Standardization)
    - v češtině: **Mezinárodní organizace pro normalizaci**
    - členy ISO jsou národní normalizační instituce
      - za ČR organizace ČSNI
- byl „oficiálním řešením“
  - řešením, které prosazovaly "orgány státu" a chtěly jej nasadit do praxe
  - dnes je prakticky odepsaný, prohrál v souboji s TCP/IP



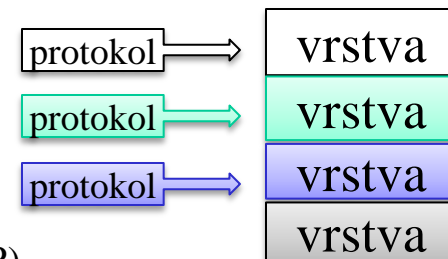
# geneze RM ISO/OSI

- geneze:

- prvotní záměr: definovat, jak mají vypadat otevřené systémy
  - tj. definovat jejich chování jak „uvnitř“, tak i „mezi sebou“
  - odsud: **Open Systems Architecture**
    - ukázalo se jako příliš náročné, dochází k redukci ambic
- revidovaný záměr: definovat pouze vzájemné propojení otevřených systémů
  - změna názvu: **Open Systems Interconnection Architecture**
    - opět se ukázalo jako příliš náročné, nedalo se stihnout
- nakonec: nebude to obsahovat konkrétní protokoly
  - ale jen představu o počtu vrstev a o tom, co má která vrstva dělat
    - aby se to „vůbec stihlo“
  - změna názvu: **Open Systems Interconnection** (OSI, ISO/OSI)



RM ISO/OSI



- proč Referenční model (RM)?

- fakticky jde o síťový model
  - pouze představa o vrstvách a jejich úkolech, bez konkrétních protokolů
- jednotlivé protokoly vznikaly dodatečně (a postupně)
  - ale moc úspěšné nebyly a v praxi se moc nepoužívaly
  - nevznikly všechny, částečné využití našly jen protokoly:
    - X.400 (pro elektronickou poštu)
    - X.500 (pro adresářovou službu – základem pro protokol LDAP z TCP/IP)

# filosofie RM ISO/OSI

- vznikal „od zeleného stolu“
  - a pak byl „nadiktován“ uživatelům
- vznikal maximalistickým způsobem
  - autoři se snažili zahrnout „vše, co by někdy někomu mohlo hodit“
- výsledek byl dosti odtažitý od reálné praxe
  - celá řešení se často ukázala jako nerealizovatelná, a hledala se implementovatelná podmnožina
  - vznikaly různé implementovatelné podmnožiny, které nebyly vzájemně kompatibilní
- mnohé výchozí předpoklady se ukázaly jako chybné



- profily GOSIP
  - když si státní (veřejné) instituce kupovaly síťové vybavení, trvaly na tom, aby vycházelo z ISO/OSI
    - ale RM ISO/OSI je příliš široký, a pro praktické použití je nutné jej „zúžit“ (vybrat implementovatelnou podmnožinu)
  - státní (veřejné) instituce musely takovou podmnožinu přesně vyspecifikovat
    - tím vznikly profily **GOSIP** (Government OSI Profile)

pro srovnání:

- rodina protokolů TCP/IP vznikala postupným obohacováním
  - nejprve se navrhlo jednoduché řešení
  - teprve postupně se "obohacovalo"
    - až na základě skutečné potřeby
    - nové řešení muselo nejprve prokázat svou životaschopnost (možnost implementace, provozní zkušenosti)

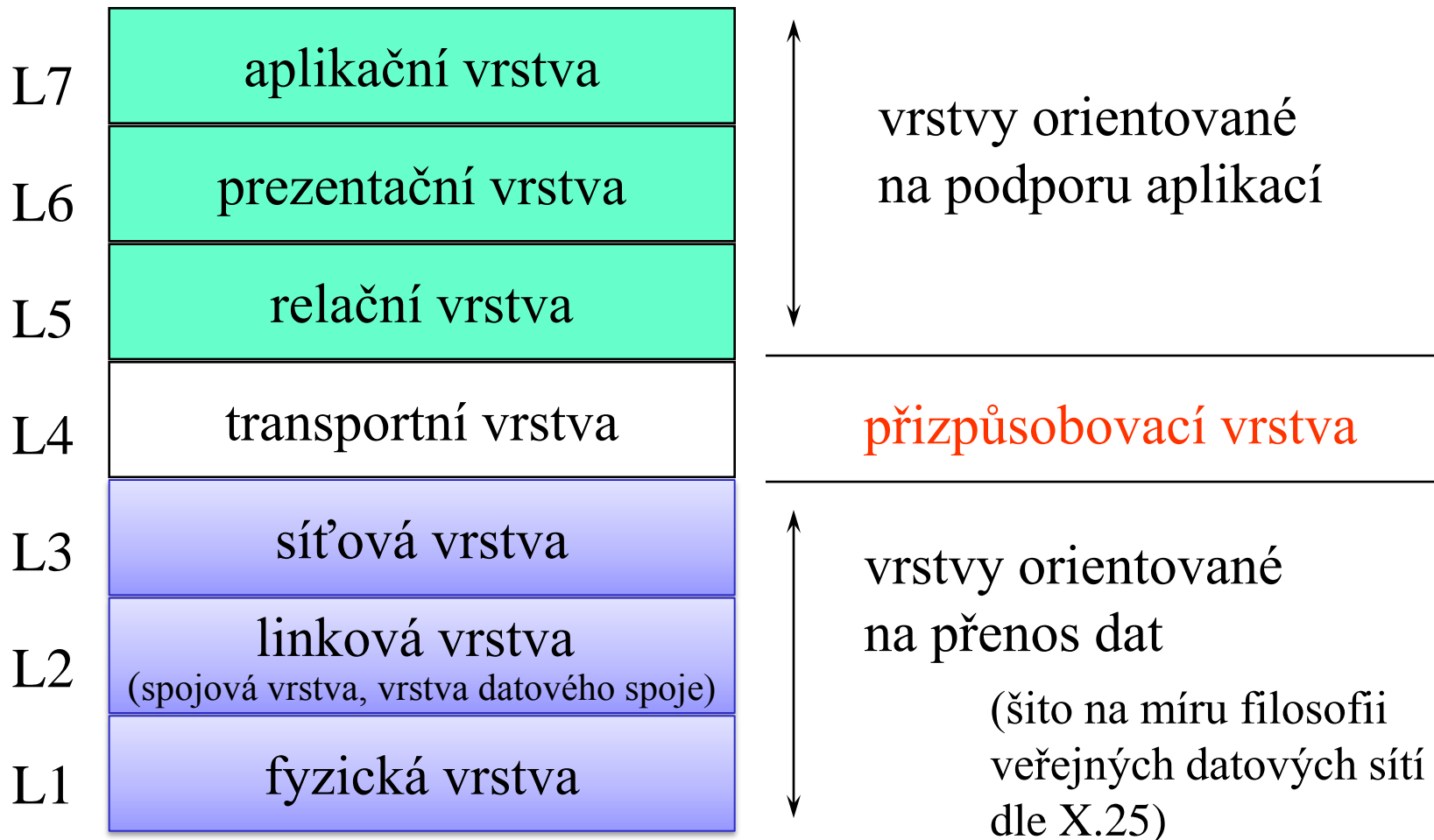


# sedm vrstev ISO/OSI

- autoři ISO/OSI se dosti dlouho přeli o počtu vrstev
- kritéria pro volbu vrstev:
  - činnosti na stejném stupni abstrakce mají patřit do stejné vrstvy
  - odlišné funkce by měly patřit do odlišných vrstev
  - aby bylo možné převzít již existující standardy
  - aby datové toky mezi vrstvami byly co nejmenší
  - aby vrstvy byly rovnoměrně vytíženy
  - ....
- nakonec zvítězil návrh na 7 vrstev
- dnes se 7 vrstev zdá být zbytečně mnoho
  - např. rodina protokolů TCP/IP má jen 4 vrstvy
- některé vrstvy ISO/OSI jsou "málo vytížené"
  - např. vrstva relační a prezentační
- jedna vrstva ISO/OSI (linková) je naopak "přetížená" a rozpadla se na dvě podvrstvy
  - podvrstvu LLC
  - podvrstvu MAC

zde šlo hlavně o již existující standard X.25

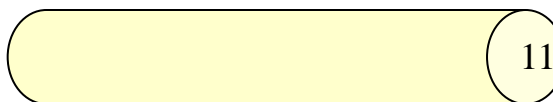
# sedm vrstev ISO/OSI



# fyzická vrstva (L1)

- zabývá se výhradně přenosem bitů
  - a v rámci toho otázkami typu:
    - kódování,
    - modulace,
    - časování,
    - synchronizace,
    - el. parametry signálů, konektory,
    - řídicí signály rozhraní, ....
- nabízí služby typu
  - přijmi bit,
  - odešli bit
- nijak neinterpretuje to, co přenáší
  - jednotlivých bitům nepřisuzuje žádný specifický význam
  - každý bit přenáší stejně
    - datový, režijní, ....
- na úrovni fyzické vrstvy se rozlišuje:
  - paralelní a sériový přenos
  - synchronní, asynchronní a arytmičtý přenos
  - přenos v základním a přeloženém pásmu
- pro přenos bitů mohou být využívána různá přenosová média
  - drátová
  - bezdrátová
- na úrovni fyzické vrstvy se pracuje s veličinami jako je:
  - šířka pásma (bandwidth)
  - modulační rychlost
  - přenosová rychlost

11001001



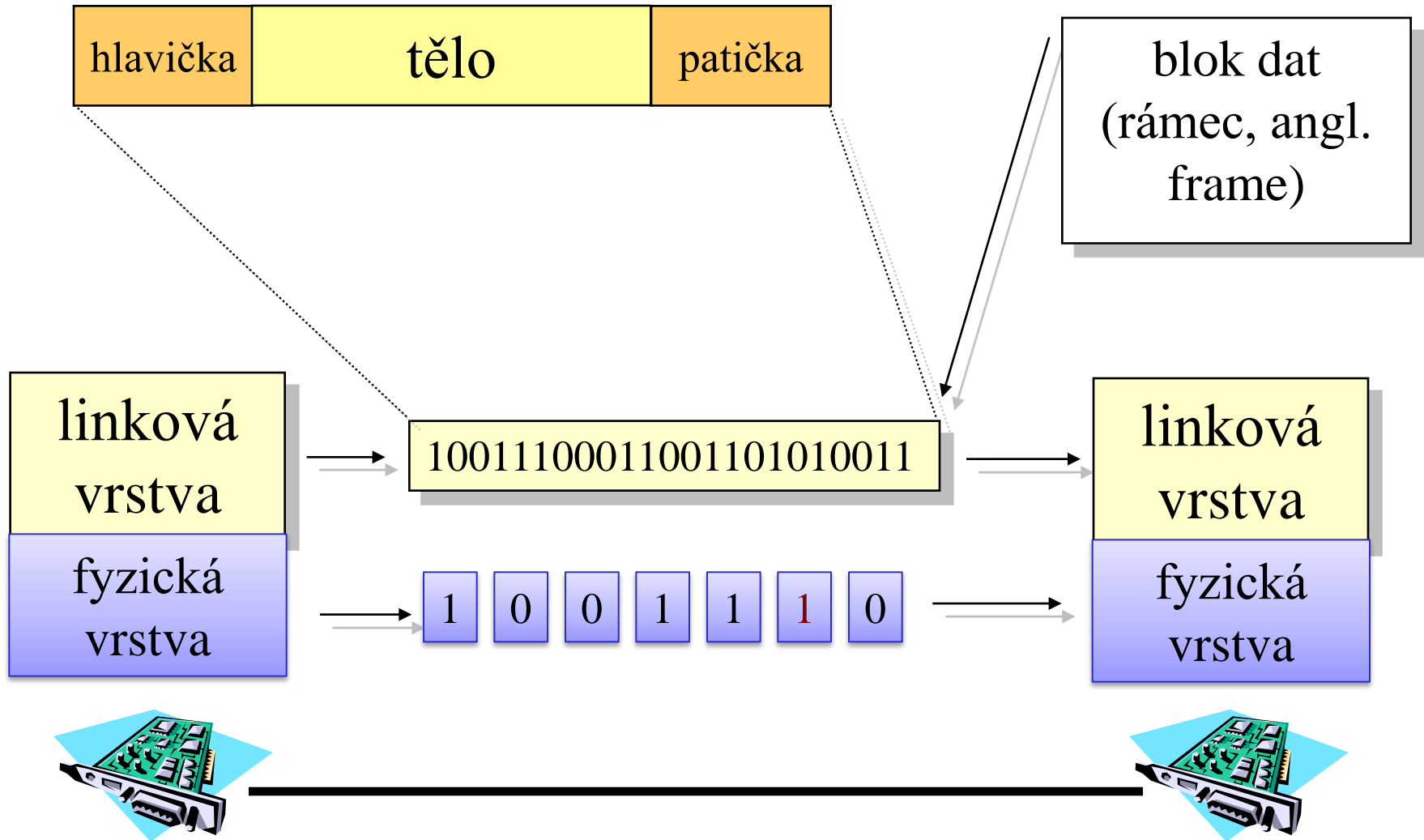
11001001010101001010101010010

# linková vrstva (spojová vrstva, L2)

- přenáší celé bloky dat
  - tzv. **rámce** (frames)
  - využívá k tomu služby fyzické vrstvy
    - přijmi/odešli bit
- přenos rámců zajišťuje pouze k přímým sousedům
  - pouze v dosahu přímého spojení, bez "přestupu" přes mezilehlé uzly
- může fungovat
  - spolehlivě či nespolehlivě
  - spojovaně či nespojovaně
  - best effort / QoS
- může využívat různé přenosové technologie
  - na úrovni fyzické vrstvy
- úkoly linkové vrstvy:
  - synchronizace na úrovni rámců
    - správné rozpoznání začátku a konce rámce, i všech jeho částí
  - zajištění spolehlivosti (pokud je požadováno)
    - detekce chyb a jejich náprava
  - řízení toku
    - zajištění toho, aby vysílající nezahltl příjemce
  - přístup ke sdílenému médium
    - řeší konflikty při vícenásobném přístupu ke sdílenému médium
    - tento úkol nebyl mezi původně uvažovanými – následně způsobil rozpad linkové vrstvy na dvě podvrstvy



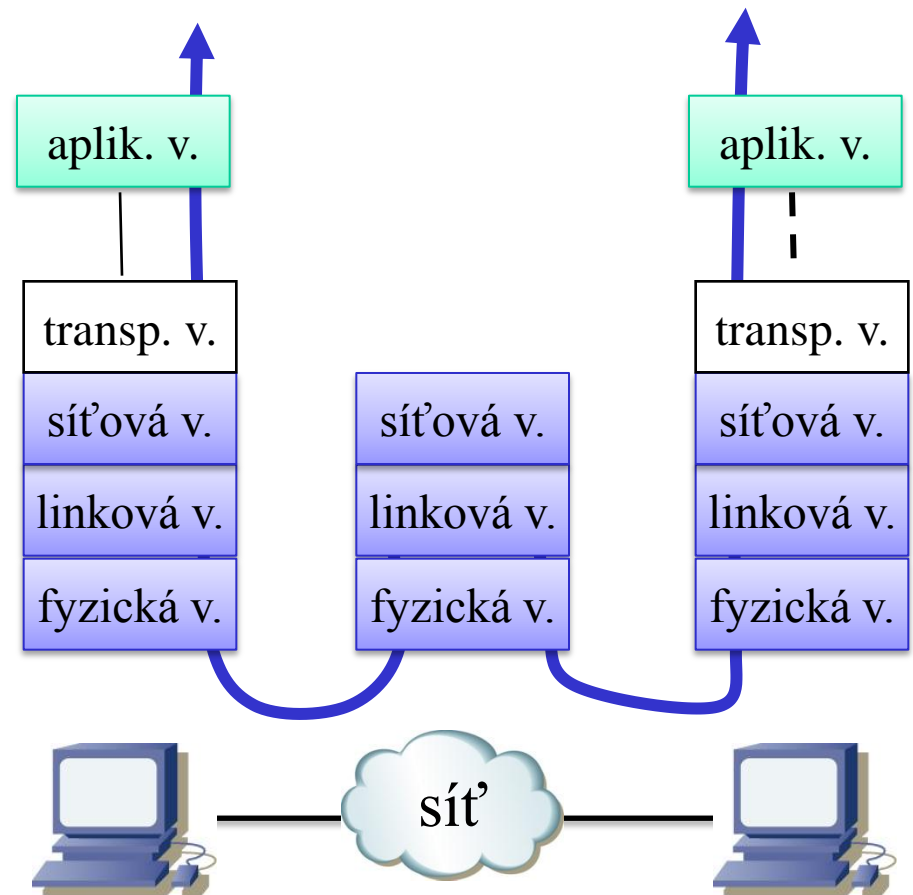
# představa fungování linkové vrstvy (L2)



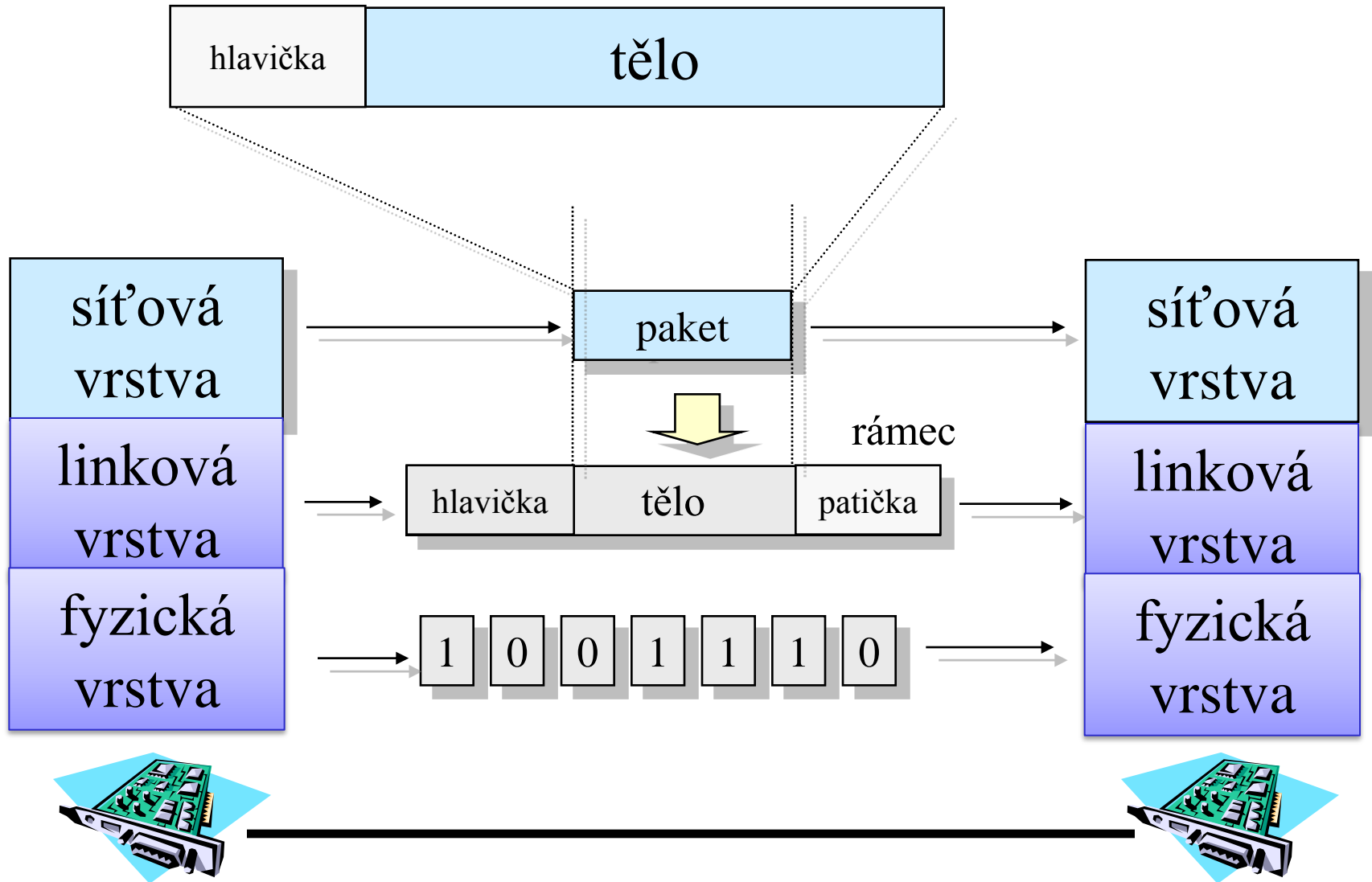
# síťová vrstva (L3)

- přenáší bloky dat označované jako **pakety** (packets)
  - fakticky: vkládá je do linkových rámců
    - předává linkové vrstvě k doručení
- zajišťuje doručení paketů až ke konečnému adresátovi
  - tj. přes různé mezilehlé uzly
    - tzv. směrovače
  - hledá vhodnou cestu až k cíli
    - zajišťuje tzv. **směrování** (routing)
- může používat různé algoritmy směrování:
  - adaptivní, neadaptivní
  - izolované, distribuované, centralizované, ..

- je poslední vrstvou, kterou musí mít i "přestupní" uzly
  - tzv. směrovače (zajišťující směrování)

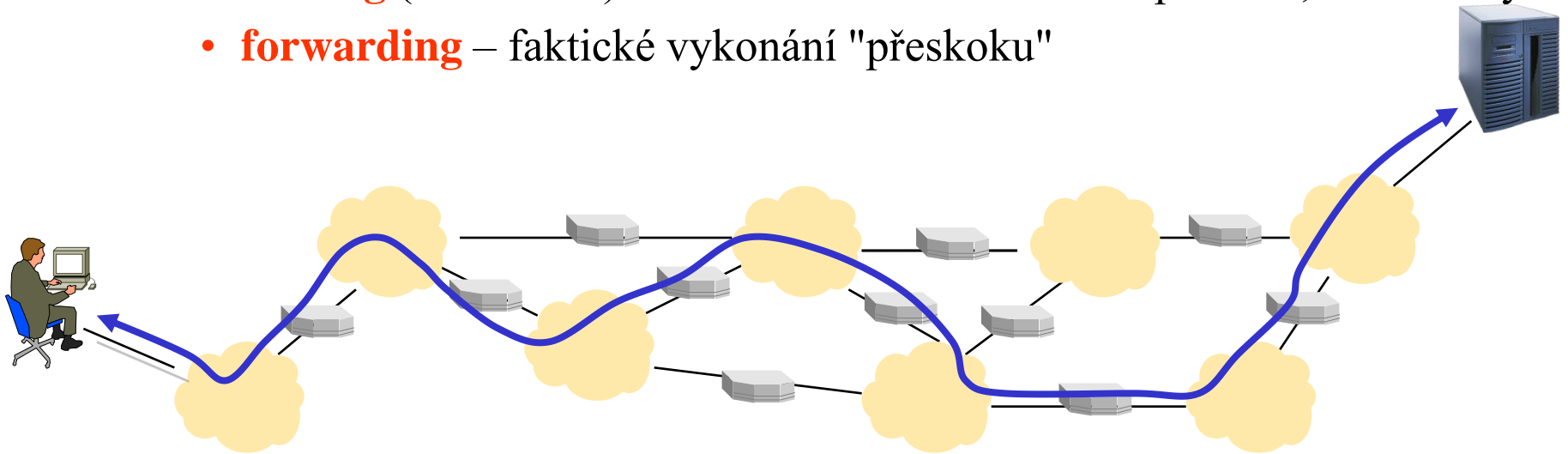


# představa fungování síťové vrstvy (L3)



# hlavní úkol síťové vrstvy (L3)

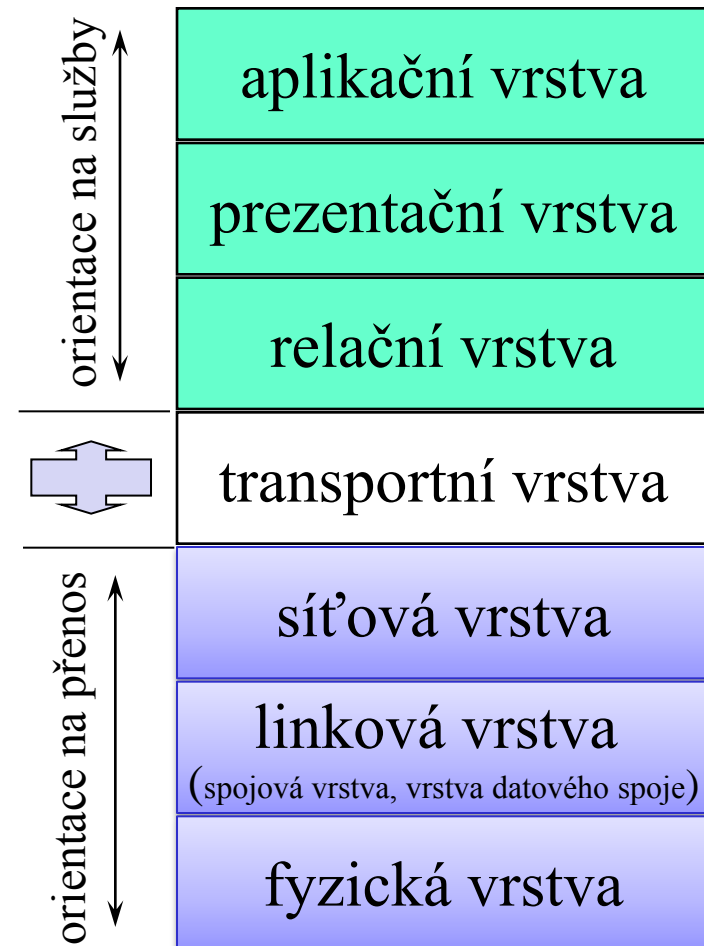
- doručovat data od jejich zdroje až k jejich koncovým adresátům
  - což může obnášet „přeskok“ přes různé mezilehlé uzly
    - linková vrstva se stará jen o doručování k přímým sousedům (v dosahu přímého spojení) a nezabývá se přeskoky
  - „přeskok“ vyžaduje:
    - **routing** (směrování) - rozhodnutí o dalším směru přenosu, volba trasy
    - **forwarding** – faktické vykonání "přeskoku"



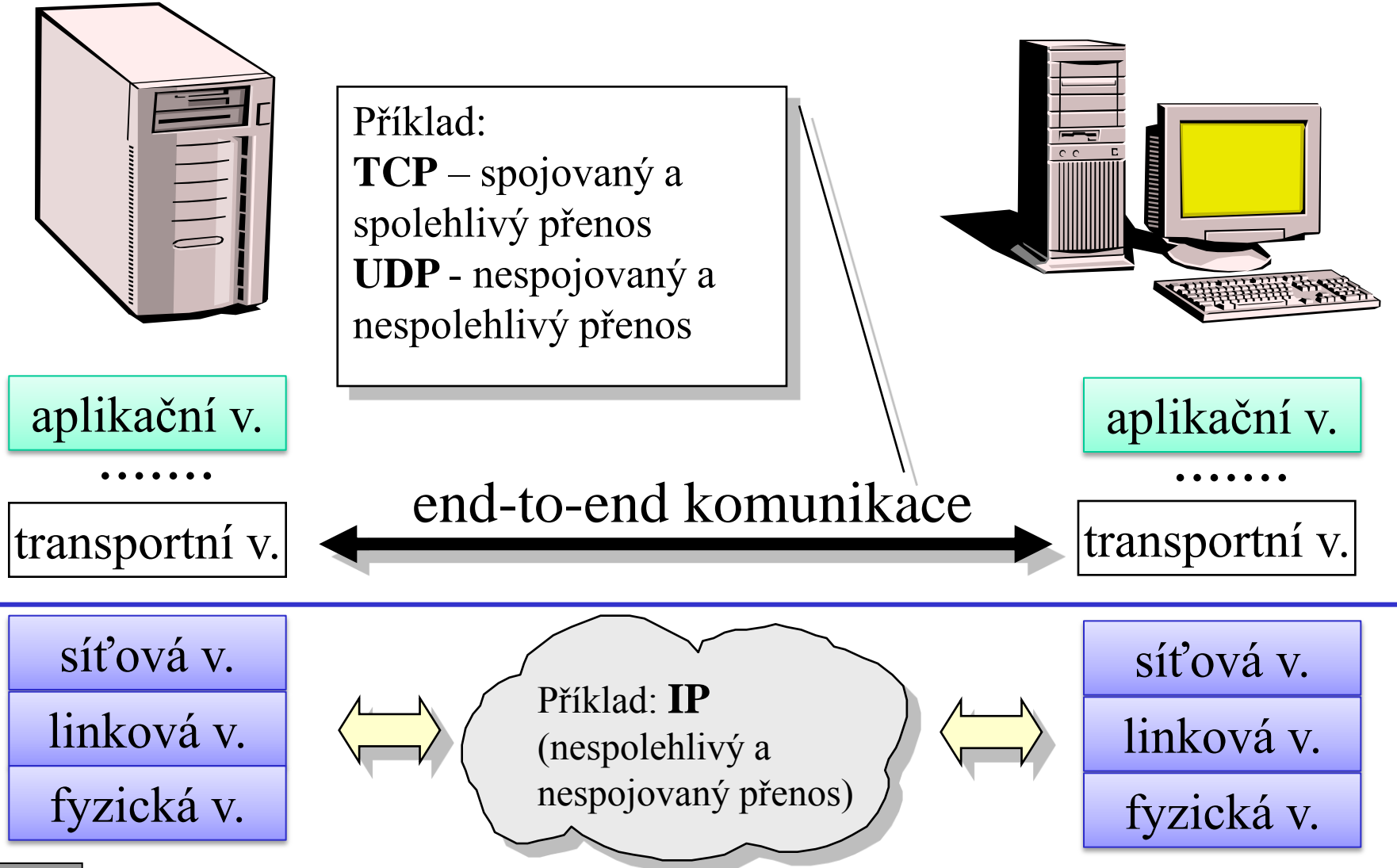


# transportní vrstva (L4)

- teze:
  - nelze „hýbat“ s vlastnostmi a funkcemi nižších vrstev
    - třeba proto že patří někomu jinému
  - vyšší vrstvy mohou chtít něco jiného, než co nabízí nižší vrstvy
- je úkolem transportní vrstvy zajistit potřebné přizpůsobení!
- zajišťuje:
  - komunikaci mezi koncovými účastníky (end-to-end komunikaci)
- může měnit
  - nespolehlivý charakter přenosu na spolehlivý
  - méně spolehlivý přenos na více spolehlivý
  - nespojovaný přenos na spojovaný

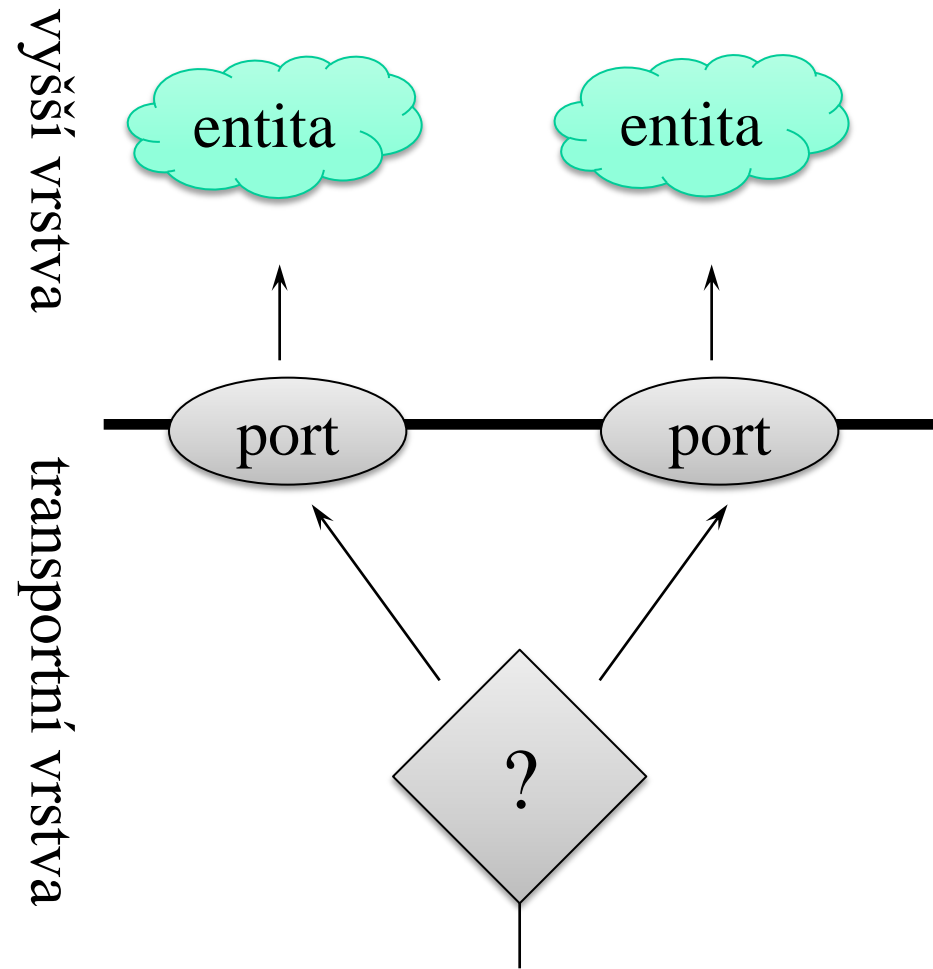


# představa transportní vrstvy (L4)



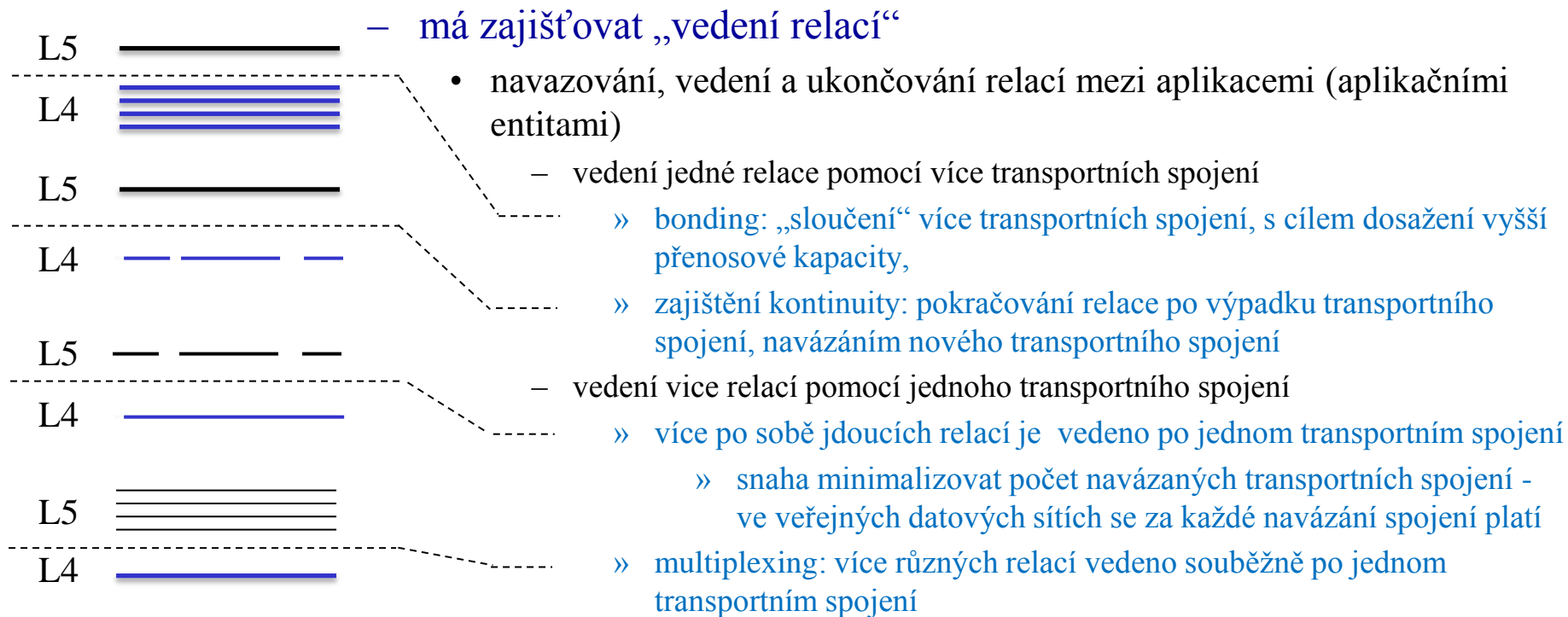
# další úkol transportní vrstvy (L4)

- do vrstvy síťové (včetně) se uzly chápou jako nedělitelné celky
  - síťové adresy reprezentují celé uzly
  - například: IP adresy v TCP/IP
- transportní vrstva již rozlišuje konkrétní entity v rámci každého uzlu
  - jednotlivé procesy, démony, úlohy ....
    - rozlišuje je obvykle nepřímo, skrze přechodové body (**body SAP, porty**) ke kterým jsou tyto entity asociovány
    - např. čísla portů v TCP/IP

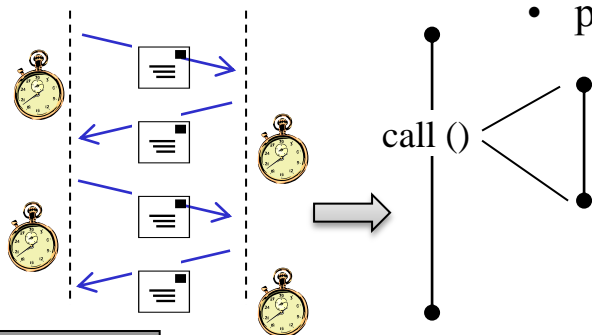


# relační vrstva (L5)

- původní představa:



- přenos jednotlivých částí (dat) v rámci probíhající relace



- například ve smyslu toho, jak dnes v TCP/IP funguje RPC (Remote Procedure Call)

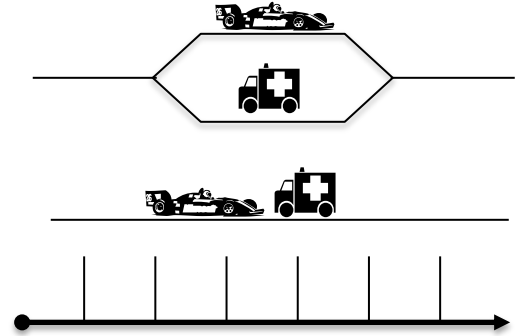
- » komunikace je ve skutečnost asynchronní, má charakter odesílání zpráv a čekání na odpovědi
  - » RPC fakticky mění komunikaci na synchronní: má charakter volání procedury
  - » bez asynchronního čekání, s transformací parametrů atd.

# relační vrstva (L5)

- původní představa:

- relační vrstva bude zajišťovat i další úkoly, například:

- synchronizaci komunikace
  - zajištění poloduplexní či plně duplexní komunikace
  - ochrana před zablokováním (deadlock, ....)
- podporu přenosů
  - checkpointing
  - nastavování „zarážek“ (bodů obnovy), od kterých je možné pokračovat v dříve přerušném přenosu
- podporu transakcí
  - např. 2. fázový commit
- zabezpečení
  - zajištění identifikace a autentizace komunikujících stran
  - zajištění důvěrnosti přenášených dat (šifrováním, ...)
  - autorizace (řízení/kontrola oprávnění ke komunikaci)
- „nalezení protistrany“
  - vyhledání aktuální polohy konkrétního uzlu
    - » například v tom smyslu, jak jej dnes řeší protokol SIP (Session Initiation Protocol)



- realita:

- relační vrstva nedělá skoro nic, často není ani implementována

- je kritizována jako nejméně „vytížená“ vrstva RM ISO/OSI



# prezentační vrstva (L6)

- nižší vrstvy (L1 až L5) přenáší data „tak jak jsou“, beze změny
  - usilují o to, aby nedošlo ke změně ani v jediném bitu
- to ale nemusí být správné a žádoucí !
  - různé strany mohou „rozumět“ stejným datům různě !
    - na různých platformách mohou stejná data „vypadat“ různě
      - mohou se lišit svou hodnotou, strukturou .....

} o řešení se má postarat právě prezentační vrstva, prostřednictvím konverzí

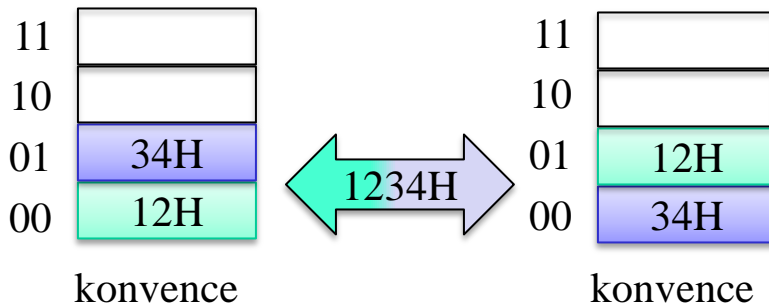
- příklady odlišností:

- kódování textu

- jednotlivé znaky lze kódovat např. v ASCII, EBCDIC, UTF-8, UTF-16 atd.

- pořadí bytů

- ve vícebytových položkách může být pořadí jednotlivých bytů různé



- formáty čísel

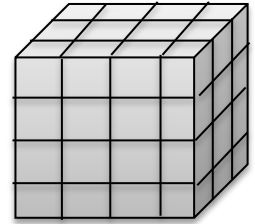
- v pevné i pohyblivé řádové čárce
  - různé velikosti mantisy a exponentů, různé základy, ....


- datové struktury

- pole, záznamy, fronty atd. ....
  - dají se konvertovat
- struktury provázané pointry
  - pointry nelze konvertovat
    - » adresové prostory příjemce a odesilatele mohou být různé

– .....

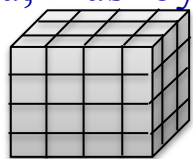
# prezentační vrstva (L6)



- další úkol prezentační vrstvy:
  - převést data do takové podoby, aby se dala přenést
- v čem je problém?
  - data, určená k přenosu (datové struktury) mohou být vícerozměrné
    - například vícerozměrná pole, struktury provázané pointerem .....
  - přenosový kanál je pouze jednorozměrný 11010010101010... 
    - přenáší pouze lineární data (lineární posloupnosti bitů)

- důsledek
  - data, určená k přenosu, musí být převedena do tvaru, který je vhodný pro přenos

možnosti řešení:



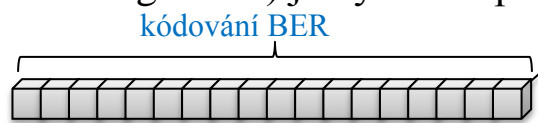
tzv. serializace .....



- „specifické“
  - řešení v rámci jedné aplikace (protokolu), která si vše vyřeší tak, jak považuje za vhodné
    - nelze jej ale použít univerzálně, pro jiné aplikace/protokoly

jakási průvodka, podle ní příjemce pozná, co data znamenají

- univerzální:
  - pomocí vhodného jazyka (např. ASN.1) je popsána obecná struktura dat – jejich význam
    - abstraktní syntaxe (abstract syntax)
  - pomocí vhodného kódování (např. BER, Basic Encoding Rules) je vytvořen přenosový (serializovaný) tvar dat, určených k přenosu
    - přenosová syntaxe (transfer syntax)

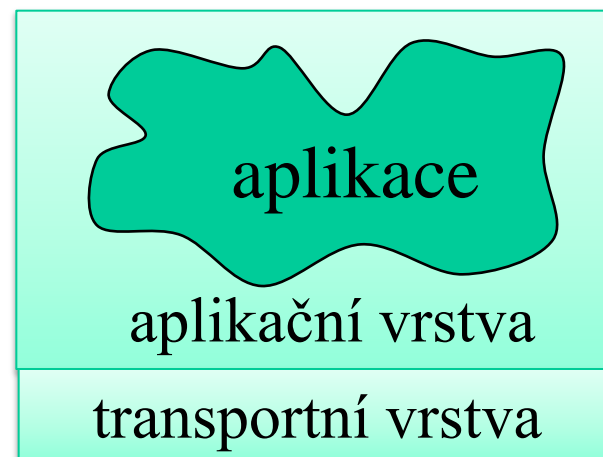


kódování BER



# aplikační vrstva (L7)

- původní představa:
  - bude obsahovat aplikace
  - problém: aplikací je moc, musely by být všechny standardizovány
    - to nejde stihnout
    - nemělo by to ani smysl
- později:
  - aplikační vrstva bude obsahovat pouze „jádro“ aplikací, které má smysl standardizovat
    - například přenosové mechanismy el. pošty
    - ostatní části aplikací (typicky: uživatelská rozhraní) byly vysunuty nad aplikační vrstvu





# srovnání RM ISO/OSI a TCP/IP

## • RM ISO/OSI

- není síťovou architekturou
  - neobsahuje (všechny) protokoly
- vzniká ve světě spojů
  - koncepce ovlivněna filosofií a „logikou“ světa spojů
- vytvářeli jej lidé, zvyklí na to, že služby jsou někomu prodávány
  - poskytovány za úplatu
  - proto: důraz spíše na bohatší a komplexnější služby
    - viz „chytrá síť, hloupé uzly“
    - preference spojovaného a spolehlivého způsobu přenosu
    - spíše podpora QoS
- při vzniku menší vazba na praxi
  - více „teoretický“ přístup
    - nejdříve se vymyslí standard, pak se řeší možnost implementace
  - i proto: více vrstev (7)

## • TCP/IP

- je síťovou architekturou
  - vznikalo „obráceně“: nejprve protokoly, pak představa o vrstvách
- vzniká ve světě počítačů
  - v akademické sféře, za grantové peníze
- vytvářeli jej lidé, kteří nepotřebovali nikomu nic prodávat
  - ani poskytovat za úplatu
  - proto: důraz spíše na jednoduchost a efektivnost
    - viz „hloupá síť, chytré uzly“
    - preference nespojovaného a nespolehlivého přenosu
    - princip best effort
- při vzniku větší vazba na praxi
  - více „praktický“ přístup
    - nejprve se ověří možnost implementace, pak standardizace
  - i proto: méně vrstev (4)

# filosofie TCP/IP

- obecně:

- preference nespojovaného způsobu přenosu

- protože ten se lépe vyrovnává s výpadky sítě či změnami v topologii

- preference nespolehlivého přenosu

- protože ne každá aplikace/služba potřebuje spolehlivost

- například pro multimediální aplikace je důležitější pravidelnost a včasnost doručování jejich dat
  - » zajištění spolehlivosti zdržuje a zavádí nepravidelnost (když se přenos opakuje)

- protože ne každé aplikaci vyhovuje stejná míra spolehlivosti

- spolehlivost není absolutní (ano/ne), ale vždy jen relativní (některé chyby jsou odhaleny a napraveny, jiné nikoli)

- » některé aplikace mohou mít vyšší požadavky na míru spolehlivosti, než je ta společná

- protože spolehlivost se snáze (levněji) zajistí v koncových uzlech, než v přenosové síti

- zabudovat potřebnou inteligenci do koncových uzlů je jednodušší a levnější

- » jsou to „univerzální“ počítače, mají levnější HW i levnější tvorbu/úpravy SW

- různé koncové uzly i různé aplikace si mohou zajistit různou míru spolehlivosti

---

- preference principu best effort (oproti podpoře QoS)

- protože to vede na podstatně jednodušší, efektivnější (i levnější) implementaci

- se všemi přenášenými daty je nakládáno stejně, nerozlišují se různá data

není to optimální pro multimediální služby (např. přenos hlasu a obrazu)

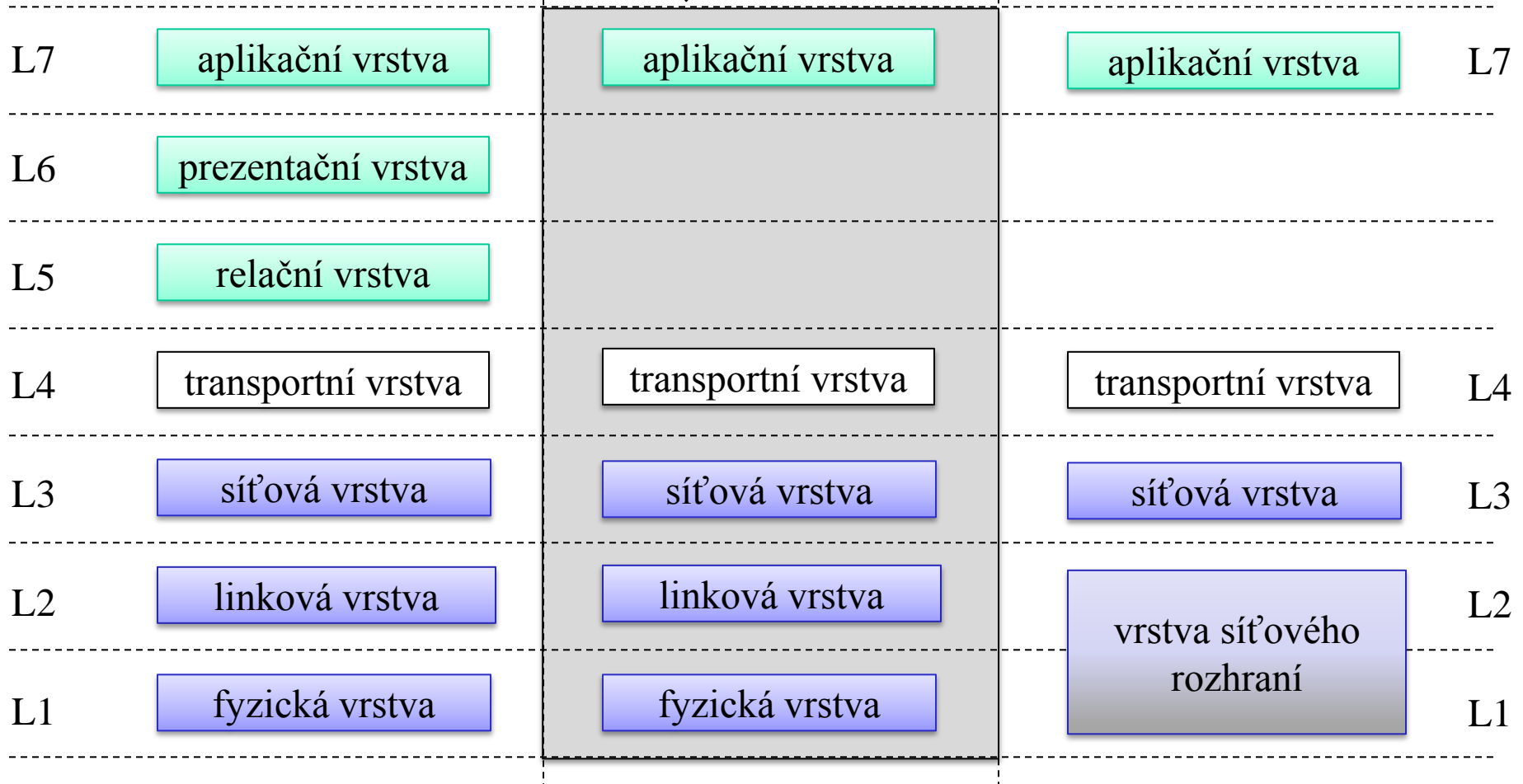
dodnes se ukazuje jako správné a účelné

# počty vrstev RM ISO/OSI a TCP/IP

referenční model  
ISO/OSI

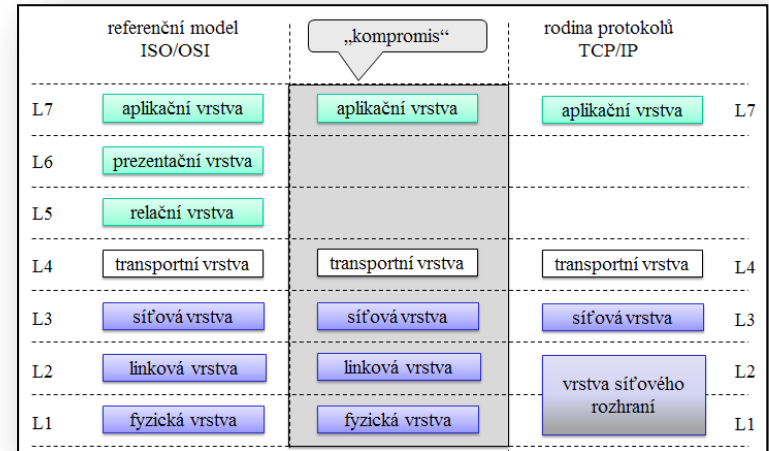
„kompromis“

rodina protokolů  
TCP/IP



# vrstvy (L1 a L2) TCP/IP

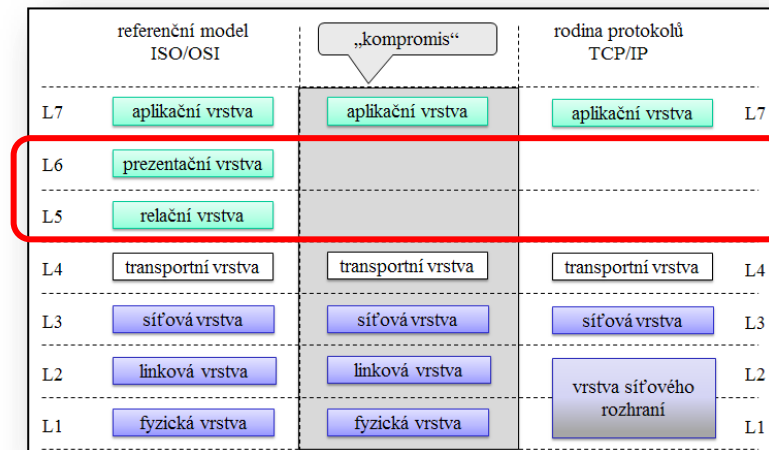
- filosofie TCP/IP:
  - nevymýšlet znovu to, co už bylo vymyšleno
  - konkrétně:
    - pokud již existuje nějaká přenosová technologie (na úrovni fyzické a linkové vrstvy), pak ji rovnou použijme
      - například: Ethernet
    - a soustředme se na to, jak ji využít co nejlépe
      - jak co nejlépe „balit“ IP pakety do Ethernetových rámců



- důsledek:
  - TCP/IP nepokrývá linkovou (L2) ani fyzickou (L1) vrstvu
    - v tom smyslu, že by definoval vlastní protokoly pro tyto vrstvy
      - » výjimka z pravidla: protokoly SLIP a PPP
        - » jako řešení pro dvoubodové spoje, kde i Ethernet je „overkill“
  - TCP/IP nerozlišuje mezi linkovou a fyzickou vrstvou
    - místo toho je „sdružuje“ do jedné vrstvy, které říká vrstva síťového rozhraní
      - » anglicky: **network interface layer**,
        - » někdy též **network access layer**, nebo jen **link layer**
    - zde se používají ona „cizí“ řešení (např. Ethernet) která nejsou součástí TCP/IP

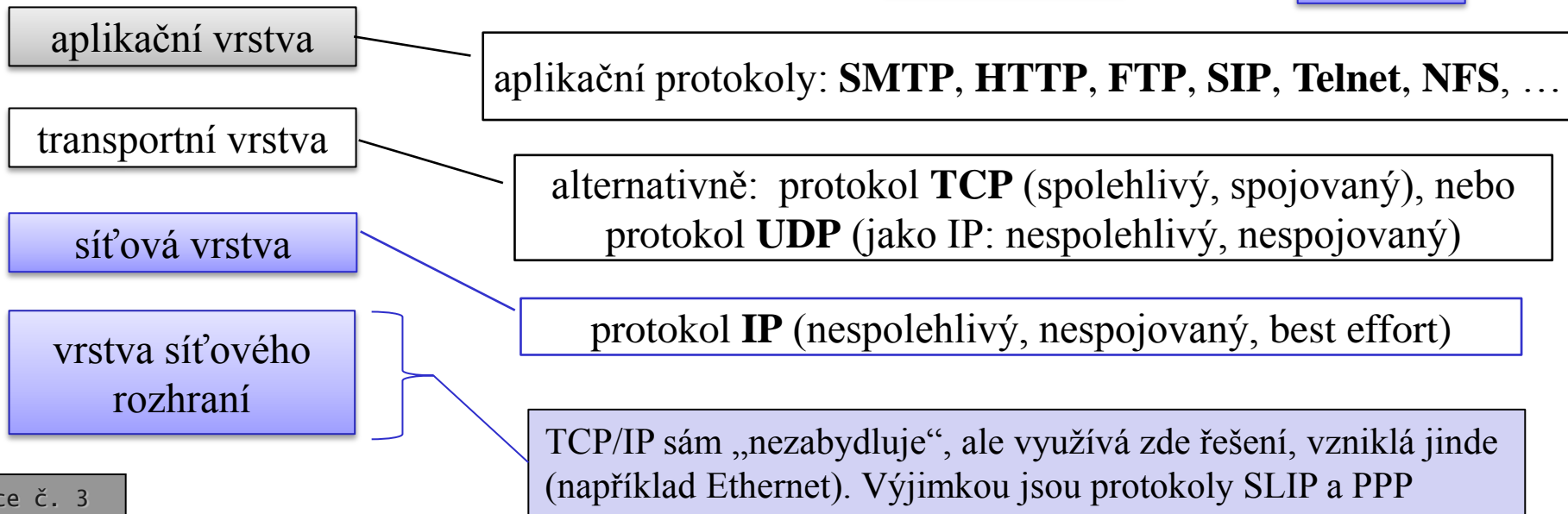
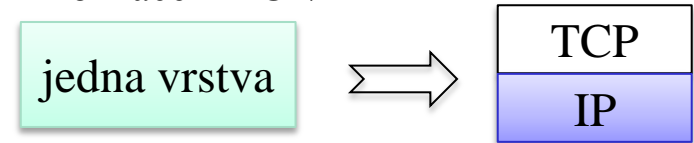
# absence vrstev L5 a L6 v TCP/IP

- filosofie TCP/IP:
  - nevnucovat nikomu něco, co nutně nepotřebuje
  - konkrétně:
    - služby relační a prezentační vrstvy nemusí potřebovat každá aplikace
    - úvaha autorů TCP/IP:
      - potřebovat je bude jen menšina aplikací
      - ti, kteří je potřebují, většina si je raději (a lépe) zajistí samy, podle svých představ
      - proto: nedělejme samostatnou relační a prezentační vrstvu
        - » protože jejich samotná existence zvyšuje režii na celkové fungování
  - důsledek:
    - TCP/IP nemá ani relační (L5), ani prezentační (L6) vrstvu
  - srovnání:
    - autoři RM ISO/OSI ve stejné situaci dospěli k závěru, že relační a prezentační služby bude potřebovat většina aplikací
      - a to ve stejné podobě/provedení
        - » v praxi se to nepotvrdilo !
    - proto zavedli samostatnou relační a prezentační vrstvu
      - které jsou dnes kritizovány jako zbytečné



# vrstvy TCP/IP

- TCP/IP má jen 4 vrstvy
  - z nichž jednu vůbec „nezabývá“
    - vrstvu síťového rozhraní
  - a také mu to stačí !
- role vrstev TCP/IP odpovídá jejich rolím v RM ISO/OSI
  - liší se ale představa o tom, jak by daná vrstva měla svou roli plnit
- proč se TCP/IP jmenuje právě TCP/IP?
  - protože původně měl mít ještě méně vrstev
    - transportní a síťová vrstva byly spojené
  - pak došlo k rozdělení na transportní vrstvu (TCP) a síťovou vrstvu (IP)
    - ke zdůraznění tohoto rozdělení se zavedlo označení TCP/IP



# TCP/IP je více než síťová architektura

- TCP/IP je síťovou architekturou
  - protože zahrnuje představu o vrstvách a také o konkrétních protokolech
- ale k TCP/IP „patří“ i další věci
  - standardizační proces
    - vše kolem vzniku technických řešení, která se stávají standardy
      - orgány IETF, IAB, IANA, ICANN, .....  
které se standardizací zabývají
  - publikační mechanismus
    - dokumenty RFC (Request for Comment),
    - ale také: STD, FYI, BCP, Internet drafts
  - správa jmenného prostoru
    - koncept DNS
    - systém domén nejvyšší úrovně (TLD)
      - zřizování nových TLD
      - správa a provoz kořenových NS
    - pravidla pro správu a fungování domén nižších úrovní
  - správa adresového prostoru
    - koncepce IP adres
      - IPv4 i IPv6
    - pravidla „používání“ IP adres (přidělování IP adres koncovým uzlům, ...)
      - ruční konfigurace, DHCP, autokonfigurace (v IPv6), .....
    - pravidla přidělování IP adres
      - pravidla a systém distribuce IP adres
        - » IANA (Internet Assigned Numbers Authority)
        - » RIR (Regional Internet Registries)
        - » LIR (Local Internet Registries)
    - koordinace číselných parametrů
      - tzv. dobře známé porty
        - » spravuje IANA