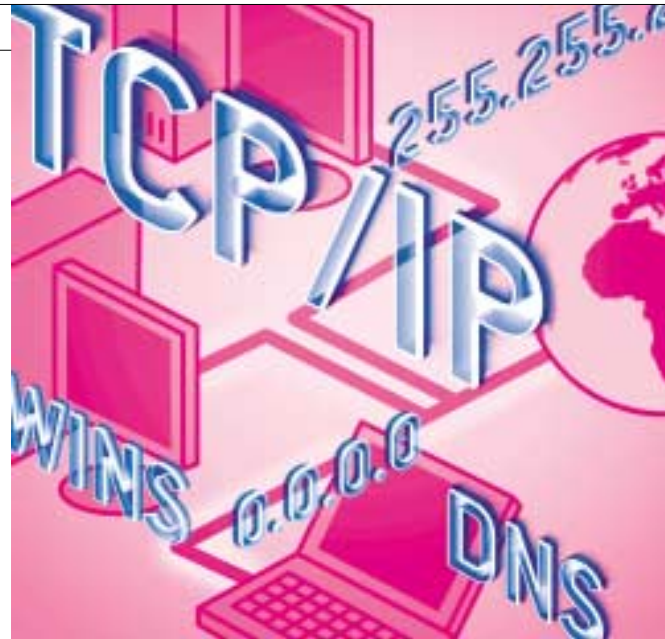


# Báječný svět počítačových sítí

Část třetí – síťové architektury

JÍŘÍ PETERKA

**Ve světě počítačových sítí existuje hned několik ucelených „světonázorů“, neboli celkových pohledů na to, jak by vlastně celá síť měla být koncipována, jak by měla fungovat a jaké služby by měla poskytovat. Jde vlastně o celé síťové architektury, které se mohou v mnohém lišit. Jinak se na svět dívá tzv. referenční model ISO/OSI, jinak zase rodina protokolů TCP/IP, která je dnes asi nejrozšířenější síťovou architekturou.**



**N**avrhnout a realizovat dobře fungující síť není žádná maličkost. Právě naopak, je to pořádně velký oříšek. K jeho rozlousknutí je proto vhodné použít stejný postup, jaký se používá ve stejné situaci i v jiných oblastech lidské činnosti: „Když je něco příliš velké, než aby se to dalo zvládnout jako celek, tak to rozdělme na menší části a ty řešme samostatně“.

Odborně se tomu říká dekompozice, jejím výsledkem by měl být rozklad jednoho velkého a obtížně řešitelného problému na několik menších problémů, z nichž každý je sám o sobě snáze řešitelný.

Při vlastní dekompozici (dělení velkého problému na menší) je samozřejmě třeba dát dobrý pozor na to, aby výsledné dílčí celky byly řešitelné samostatně, resp. aby řešení jedné dílčí části nebylo vázáno na to, jaké řešení se zvolí

v části jiné. Jde tedy o to, kudy a jak vést pomyslný řez velkého problému.

V případě počítačových sítí je „řez“ tradičně veden horizontálně, díky čemuž vznikají hierarchicky uspořádané vrstvy. Kolik takových vrstev má být, co a jak mají dělat, jak mají fungovat atd. – právě v tom se jednotlivé síťové architektury liší.

Až se záhy dostaneme k tzv. referenčnímu modelu ISO/OSI, zjistíme, že předpokládá 7 vrstev. Naopak rodina protokolů TCP/IP, na níž dodnes funguje celý internet, předpokládá pouze 4 vrstvy. A také jí to stačí.

## Výhody vrstevnatých modelů

Vedle schůdnějšího řešení menších problémů místo obtížného řešení jednoho velkého přináší rozdělení do hierarchicky uspořádaných vrstev (tzv. vrstevnatý model) i některé další výhody. Právě u počítačových sítí je velmi důležité, že při vhodném návrhu může být každá vrstva implementována nezávisle na ostatních vrstvách, a to dokonce různými způsoby. To znamená, že nižší vrstvy, zabývající se přenosem dat, mohou existovat ve více variantách a mohou být používány podle toho, jaký konkrétní mechanismus přenosu dat a jaké přenosové médium je právě k dispozici. Vyšší vrstvám to ale může být jedno, mohou tak existovat v jednotném provedení a fun-

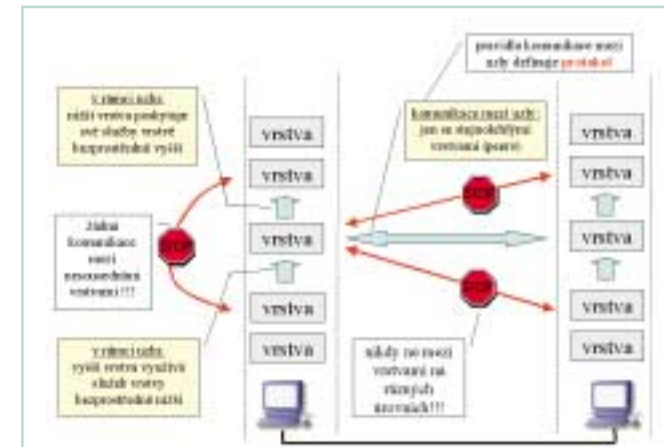
govat vždy stejně bez ohledu na to, jaká nižší vrstva je pod nimi právě „podstrčena“.

Praktickým důsledkem je například to, že internet funguje v principu všude stejně, bez ohledu na to, zda jste právě připojeni přes telefon, ADSL, kabelovou přípojku, nějakou formu bezdrátového připojení apod. Rychlost odezvy se nejspíše bude lišit, ale to by měl být jediný rozdíl – jinak by vše ostatní mělo fungovat stejně.

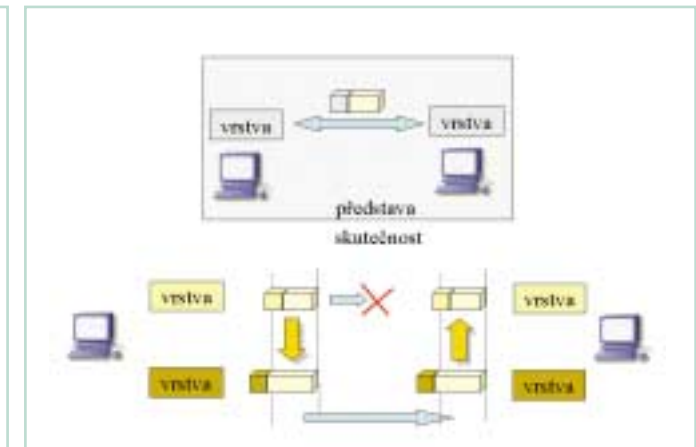
Pokud se v budoucnu objeví nějaká nová technologie, schopná přenášet data a zajistit připojení k internetu, bude třeba vyvinout novou variantu implementace té vrstvy, která tuto novou technologii využívá a přenáší po ní data. Tato nová varianta nižší vrstvy se pak „podstrčí“ pod vrstvy vyšší, které mohou zůstat beze změny. Přesto by vše mělo fungovat tak, jak má. Internet bude rázem dostupný i po této nové přenosové technologii.

## Vrstvy, služby, protokoly

Předchozí příklad dává tušit, že jednotlivé vrstvy mají své konkrétní úkoly, které plní. Jedna vrstva může zajišťovat přenos dat k přímým sousedům, zatímco jiná hledá cestu po celé soustavě sítí, aby se přenášená data dostala až ke svému cíli. Další vrstva pak rozumí přenášeným datům a zajišťuje fungování služeb jako elektronická pošta či WWW.



▲ Představa vrstev a protokolů.



▲ Horizontální komunikace mezi vrstvami, představa a skutečnost.

Obecně tedy každá vrstva plní své úkoly tak, že poskytuje určité služby. Nejvyšší vrstva je poskytuje přímo uživatelům (jako již zmiňovanou elektronickou poštu, WWW stránky, přenos souborů atd.), ale ostatní vrstvy si je poskytují navzájem: konkrétní vrstva nabízí své služby vrstvě bezprostředně vyšší (v rámci daného uzlu). Sama přitom ke svému fungování využívá služeb, které jí poskytuje vrstva bezprostředně nižší. Vše naznačuje i následující obrázek.

V prostředí sítí je přirozené, že vrstvy jednotlivých uzlů spolupracují při plnění svých úkolů s vrstvami jiných uzlů. Tato spolupráce se ale vždy odehrává na stejné úrovni, resp. na úrovni stejnohlých vrstev. Nikdy by naopak neměly spolupracovat vrstvy na nestejných úrovních.

Důležitý je i konkrétní způsob vzájemné komunikace stejnohlých vrstev. Pravidla této komunikace a vzájemné interakce obou stran definuje tzv. **protokol**. Ten například říká, jaký formát a význam mají data, která si obě strany vzájemně předávají, jak má jedna strana reagovat na to, co jí posílá protistrana atd. Protokoly přitom přísluší jednotlivým vrstvám – různé vrstvy stejných uzlů tedy používají ke vzájemné komunikaci různé uzly. Nebo obráceně: každý protokol vždy „patří“ do určité vrstvy.

Bylo by ale chybou myslet si, že mezi protokoly a vrstvami platí poměr 1 : 1, neboli co protokol, to vrstva a naopak. Jedna vrstva může ke svému fungování využívat více protokolů, a to i souběžně. Například na jedné vrstvě (transportní, viz dále) se mohou používat dva různé protokoly, jeden pro spolehlivý přenos (zabezpečený proti ztrátám a poškození dat), druhý nespolehlivý (který takto zabezpečen není). Na nejvyšší úrovni (aplikační, viz dále) může být používán jeden protokol pro přenos WWW stránek, jiný protokol pro přenos zpráv elektronické pošty a ještě jiný pro přenos souborů atd.

Zpět ale k protokolům jako takovým: ty musí být oběma stranám známy předem, aniž by se na jejich podobě musely nějak domlouvat. Proč tomu tak musí být vyplývá ze skutečnosti, že spolu mohou chtít komunikovat dva uzly, které až dosud neměly nic společného. Vůbec se neznají,

## ► Horizontální komunikace mezi vrstvami.

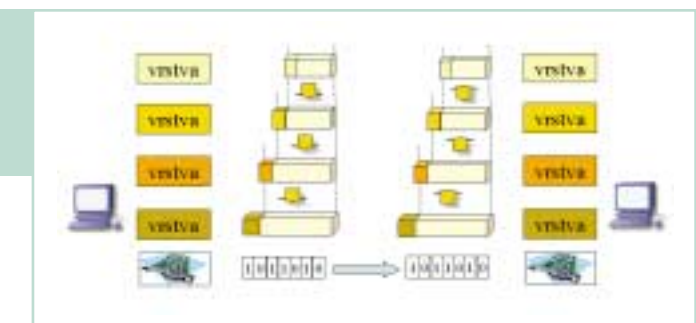
stojí na různých systémových platformách, mají jiné operační systémy, používají jiné aplikace – a přesto si musí hned napoprvé dobře porozumět. Musí tedy příslušný protokol znát „odjinud“, a to v jeho definitivní a „závazné“ podobě. Takovouto „předem známou“ podobu protokolů nabízí standardizace, resp. vydávání standardů popisujících dané protokoly.

## Horizontální komunikace mezi vrstvami

Pravidla vzájemné komunikace mezi stejnohlými vrstvami různých uzlů (horizontální komunikace), definovaná konkrétním protokolem, obvykle předpokládají, že jedna strana pošle druhé straně určitý „balíček“, obsahující jak data, tak instrukce týkající se toho, co se s daty má stát (jak mají být dále zpracovány atd.). Lze si to představit také tak, že jde o obálku: uvnitř této obálky jsou data, na obálce je napsán vzkaz protistraně (instrukce).

Ovšem představa, že by tuto obálku skutečně jedna strana předala přímo své partnerské straně (stejnohlé vrstvě), není většinou správná. Kromě nejnižších vrstev totiž spolu žádné jiné vrstvy fakticky přímo nekomunikují. Pokud si potřebují něco předat, udělají to tak, že příslušnou „obálku“ předají k doručení bezprostředně nižší vrstvě. Ta se zachová stejně: obálku od vyšší vrstvy vloží do jiné (poněkud větší) obálky, na tu napíše vzkaz pro svou partnerskou (stejnohlé) vrstvu a předá ji protistraně. Opět ale nikoli přímo, ale tak, že tuto obálku předá své bezprostřední vrstvě...

Takto vše pokračuje, dokud se obálka nedostane až k nejnižší vrstvě. Teprve ta totiž něco



fakticky přenáší – vezme tedy obálku od druhé nejnižší vrstvy (fakticky určitý blok dat) a celou ji předá, bit po bitu, druhé straně.

## Síťové architektury

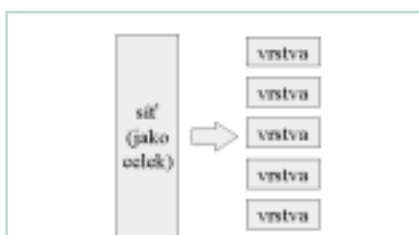
Pokud už tušíme, co je protokol a co jsou vrstvy, můžeme si vysvětlit i pojem „síťová architektura“. Představuje ucelenou představu o tom, kolik by mělo existovat vrstev a co by mělo být jejich úkolem. Kromě toho je součástí síťové architektury i konkrétní představa o tom, jak by jednotlivé vrstvy měly své úkoly plnit – tedy představa o konkrétních protokolech, patřících do jednotlivých vrstev.

Již avizovaným příkladem síťové architektury je rodina protokolů TCP/IP, vytvořená pro potřeby internetu, dnes jednoznačně nejrozšířenější a nejpoužívanější. Rozhodně ale není síťovou architekturou jedinou.

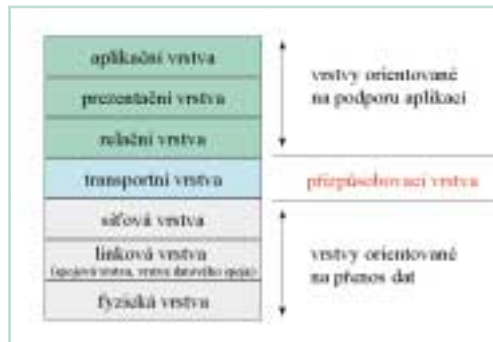
Síťové architektury se objevily již v době vzniku prvních počítačových sítí, ale zpočátku jako tzv. proprietární (tj. jako vlastní řešení jedné konkrétní firmy). Takto vznikly například architektury SNA (Systems Network Architecture firmy IBM) či DECNET od společnosti Digital Equipment Corporation). Záhy se ale objevila potřeba vytvořit něco nikoliv proprietárního, čemu by „nevelela“ jen jedna firma, ale co by bylo naopak dostatečně otevřené.

## Geneze referenčního modelu ISO/OSI

Úkolu vypracovat otevřenou síťovou architekturu se nakonec chopila organizace ISO (Internati-



▲ Představa dekompozice.



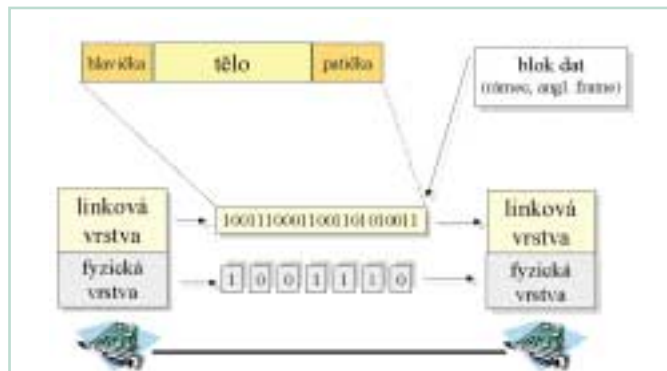
▲ Sedm vrstev ISO/OSI.

onal Organization for Standardization). Ta pochází spíše ze světa spojů, je mezinárodní organizací pro normalizaci s formálním mandátem od jednotlivých členských zemí, jejími členy jsou národní normalizační instituce. Za ČR je členem ČSNI (Český normalizační institut).

Organizace ISO však původně chtěla vytvořit mnohem více než jen síťovou architekturu ve výše uvedeném smyslu. Předsevzala si, že formuluje obecnou a ucelenou představu toho, jak mají vypadat tzv. otevřené systémy. Přesněji: architekturu otevřených systémů (Open Systems Architecture), zahrnující kromě sítí i samotné uzly a jejich fungování „uvnitř“.

To se ale záhy ukázalo jako příliš velké sousto, a tak ISO svůj záměr zredukovala – odebrala „vnitřní fungování“ jednotlivých uzlů a rozhodla se soustředit jen na jejich vnější projevy při vzájemném propojení. Výsledkem byla i změna názvu toho, co mělo vzniknout. Nyní už šlo o „pouhé“ Open Systems Interconnection Architecture (aneb architektura propojování otevřených systémů).

Bohužel i to se nakonec ukázalo jako příliš velké sousto, a tak ISO musela slevit ještě jednou. Připravila sice představu o počtu vrstev a jejich úkolech, ale již nestihla včas vymyslet jednotlivé protokoly, které by jednotlivé vrstvy používaly. Proto nakonec zredukovala zadání na vytvoření pouhého „referenčního modelu“ (jakéhosi všeobecného rámce) a z předchozího názvu (Open Systems Interconnection Architecture, OSIA) muselo zmizet slovo „Architecture“. Tak vznikl konečný název Open Systems Interconnection, OSI.



▲ Představa fungování linkové vrstvy.

Výsledkem práce organizace ISO tedy byl referenční model ISO/OSI. Ten měl být „oficiálním“ řešením, které by jednotlivé členské státy následně zavedly do praxe. Některé se o to skutečně i snažily – ale neuspěly. Narazily totiž na to, že na trhu nebyly prakticky žádné produkty, které by z referenčního modelu ISO/OSI vycházely.

Důvod byl dosti příznačný. Celá koncepce RM ISO/OSI vznikala u zeleného stolu, bez kontaktu s realitou a bez respektování toho, co a jak se dá prakticky implementovat a co se také implementovat vyplatí a je pro praxi potřebné.

Zjednodušeně by se dalo říci, že referenční model ISO/OSI vznikl následujícím postupem: jeho autoři postupně „přihazovali“ další a další vlastnosti a funkce, které by celé řešení podle jejich názoru mělo obsahovat. Když už je nic dalšího nenapadlo, vše sečetli a udělali z toho závazný standard. Teprve následně (a často to byli jiní lidé) se začali zamýšlet nad tím, zda a jak by šlo vše realizovat. A tu se obvykle zjistilo, že standardem požadované řešení je příliš bohaté, příliš „velké“ a prakticky nerealizovatelné. Musely se hledat prakticky implementovatelné podmnožiny... než se ale našly a prosadily do praxe, ujala se vlády nad počítačovými sítěmi jiná architektura: rodina protokolů TCP/IP. Na tu se podrobněji podíváme v příštím dílu tohoto seriálu.

### Současnost ISO/OSI

Ačkoli referenční model ISO/OSI v souboji s protokoly TCP/IP prohrál (i v souvislosti s úspěchem TCP/IP v rámci internetu), přesto není správné se na něj dívat jako na něco zcela odepsaného, o čem ani nemá smysl se zmiňovat. To určitě ne. Některé z protokolů, které byly vyvinuty pro ISO/OSI a (dodatečně) dosazeny do jeho referenčního modelu, se skutečně používaly nebo se staly alespoň základem pro jiné úspěšné protokoly. Za všechny lze zmínit například protokol X.400, který sloužil potřebám elektronické pošty a byly na něm založeny i některé reálné fungující poštovní systémy. Protokol X.500 zase prošel odtučňovací kúrou a stal se základem pro protokol LDAP (Lightweight Directory Access Protocol) z rodiny protokolů TCP/IP.

Kromě toho referenční model ISO/OSI – a to hlavně jeho vrstevový model – je dodnes základem pro většinu učebních textů, jež seznamují zájemce se základními principy světa počítačových sítí. Proto s ním začneme i my.

### Sedm vrstev ISO/OSI

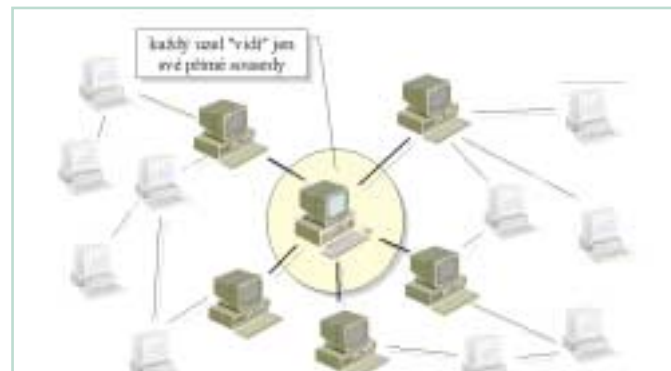
Referenční model ISO/OSI předpokládá existenci sedmi vrstev, jak ostatně naznačuje i obrázek. Tři nejnižší vrstvy se primárně zabývají přenosem dat a přenášená data nijak nepracovávají ani neinterpretují. Naopak tři nejvyšší vrstvy již přenášená data nějakým způsobem zpracovávají či alespoň interpretují, snaží se tedy vycházet vstříc potřebám jednotlivých aplikací. Mezi těmito dvěma trojicemi pak existuje ještě jedna vrstva, která má fungovat jako jakási přizpůsobení, resp. přizpůsobovací vrstva.

Jen pro srovnání – rodina protokolů TCP/IP má pouze čtyři vrstvy a také si s nimi bohatě vystačí.

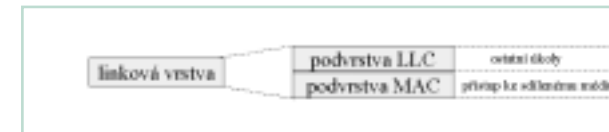
### Fyzická vrstva

Úkolem fyzické vrstvy, jak už její název napovídá, je „fyzický“ přenos jednotlivých bitů. Jak jsme si již uvedli výše, jde vlastně o jedinou vrstvu, která skutečně přenáší nějaká data. Přenáší je po bitech a bezprostředně vyšší vrstvě (vrstvě linkové) tedy nabízí dvě služby: odeslání bitu a příjem bitu. Aby tak mohla činit, musí se fyzická vrstva zabývat tím, jak jsou jednotlivé bity na přenášeném médiu znázorněny – zda a jak jsou kódovány či modulovány, jak je vyřešeno časování a synchronizace, případně jaké jsou používány konektory a jednotlivá rozhraní, jaké jsou jejich řídicí signály atd. Podle toho se pak rozlišuje například synchronní a asynchronní přenos, přenos modulovaný a nedomulovaný (resp. v základním pásmu), případně přenos sériový a paralelní atd. Bere se v úvahu také přenosová rychlost (v bitech za sekundu, resp. v násobcích), modulační rychlost (v Baudech), volí se různé způsoby modulace atd.

Fyzická vrstva naopak nijak neinterpretuje bity, které přenáší, s každým nakládá stejně jako s ostatními. Nepozná ani to, že některé bity „patří k sobě“ a představují nějaký ucelenější blok dat – například linkový rámec.



▲ Představa „viditelnosti uzlů“ na úrovni linkové vrstvy.



◀ Podvrstvy LLC a MAC.



◀ Představa fungování síťové vrstvy: doručuje síťové pakety až k jejich koncovým příjemcům.

### Linková vrstva

Sestavovat jednotlivé bity do větších celků – tzv. linkových rámců (anglicky frames) – má za úkol až vrstva linková. Ta tedy sama využívá služeb fyzické vrstvy (typu „přijmi bit, odešli bit“) a sama musí zajistit korektní rozpoznání začátku i konce každého jednotlivého rámce.

Důležité také je, že linková vrstva přenáší linkové rámce jen ke svým přímým sousedům. Tedy k uzlům, s nimiž má přímé spojení (a nikoli takové, které vede zprostředkovaně přes jiný uzel). Lze si to představit i tak, že na úrovni linkové vrstvy každý uzel „vidí“ jen své přímé sousedy, zatímco existenci dalších uzlů si nemusí uvědomovat.

Na linkovou vrstvu však mohou být kladeny ještě kromě samotného přenosu linkových rámců další požadavky. Například tzv. řízení toku, které má zabránit tomu, aby odesílatel zahlcoval příjemce – tedy aby mu posílal data rychleji, než je příjemce schopen je zpracovávat.

Po linkové vrstvě může být také požadováno, aby zajišťovala spolehlivý přenos – tedy aby kontrolovala, zda při přenosu nedošlo k nějaké chybě. Pokud snad ano, pak je povinná se postarat o nápravu (typicky tak, že si nechá poslat poškozená či ztracená data znovu). V opačném případě, pokud tedy není spolehlivost požadována, se takový přenos označuje jako nespolehlivý.

### Podvrstvy linkové vrstvy

Linková vrstva referenčního modelu ISO/OSI původně počítala spíše s rozlehlými sítěmi (sítěmi WAN), které propojují své uzly pomocí vyhrazených dvoubodových spojů – jako na obrázku. Velmi brzy se ale na scéně objevily také lokální sítě (LAN), které mají sdílenou topologii, protože používají sdílené přenosové médium. Tedy takové, k němuž je připojeno více uzlů najednou a všechny tyto uzly mohou také současně přijímat. Ovšem vysílat by měl vždy nejvýše jeden uzel, protože jinak dochází k tzv. kolizi a „promíchání“ signálů od více vysílajících uzlů.

V takových lokálních sítích proto musí být použita nějaká metoda řízení přístupu jednotlivých uzlů ke sdílenému médiu. Ta ale musí být implementována nad fyzickou vrstvou, protože sama ke svému fungování využívá vysílání a příjem jednotlivých bitů. Stejně tak ale musí být přístupová metoda implementována ještě pod linkovou vrstvou, protože když tato vrstva přenáší ce-

lé rámce, měla by již mít zajištěn výlučný přístup ke sdílenému médiu pro potřeby vysílání.

Autoři ISO/OSI tento protichůdný požadavek vyřešili šalamounsky. V zásadě mezi původní fyzickou a linkovou vrstvou vsunuli jednu další vrstvu. Aby tím ale nezvyšovali již tak vysoký počet vrstev, označili to jako rozdělení vrstvy linkové na dvě podvrstvy:

- (vyšší) podvrstvu řízení linkového spoje (podvrstvu LLC, z anglického Logical Link Control), která dělá v zásadě vše, co původní linková vrstva,
- (nižší) podvrstvu řízení přístupu ke sdílenému médiu (podvrstvu MAC, z anglického Media Access Control), jež implementuje přístupovou metodu.

### Síťová vrstva

Zatímco linková vrstva přenáší linkové rámce pouze ke svým sousedům, v praxi bývá nutné přenést nějaká data ještě dále – přes mezilehlé uzly, do dalších sítí. To už ale má za úkol vrstva síťová. Blokům dat, které přenáší, se říká pakety (anglicky packets). Síťová vrstva je přenáší přes různé mezilehlé uzly tak dlouho, dokud je nedoručí jejich konečnému příjemci. Aby se tak stalo, musí být síťová vrstva schopna provádět tzv. směrování, neboli hledat cesty skrze sítě, vedoucí až ke kýženému cílovému uzlu.

Metod, jak hledat nejhodnější cestu k cílovému uzlu, existuje celá řada a jejich podrobnější popis vydá na jeden z příštích dílů tohoto seriálu. Zde si jen naznačíme, že existují různé inteligentní (promyšlené) algoritmy směrování. Mezi ty „méně promyšlené“ patří například metoda tzv. záplavového směrování, kdy každý mezilehlý uzel předá každý paket dál ve všech směrech, jež vedou z daného uzlu (kromě přichozího směru). Tím sice vznikají duplicitní pakety, které je třeba následně eliminovat, ale je zaručené, že se „záplava“ dříve či později dostane ke svému cíli.

Promyšlenější algoritmy směrování hledají cestu k cílovému uzlu jinak, na základě znalosti celé soustavy vzájemně propojených sítí. Na úrovni síťové vrstvy by si proto jednotlivé (mezilehlé) uzly měly plně uvědomovat celou topologii soustavy sítí a nikoli jen bezprostřední sousedy.

### Transportní vrstva

O transportní vrstvě již víme, že je jakousi přizpůsobovací vrstvou mezi trojicí nejspodnějších

**ZyXEL**  
Intelligence. Power.

SKY IS THE LIMIT

Wi-Fi

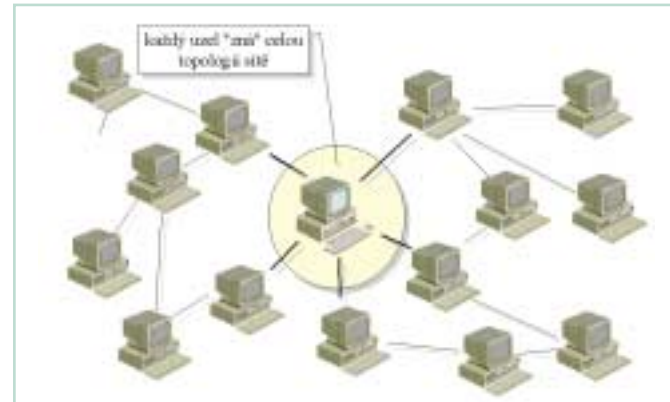
ADSL

Internet Security

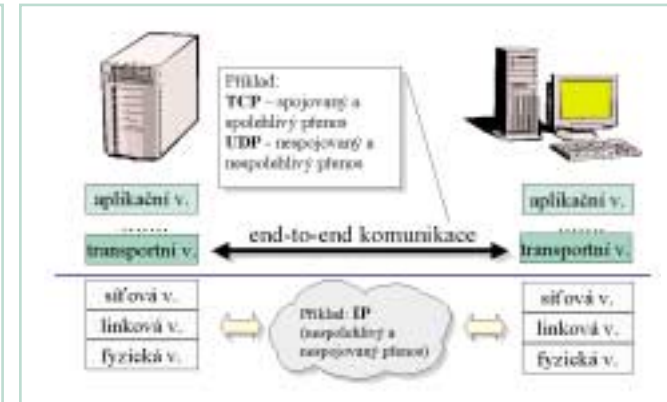
Gigabit/Fast Ethernet

VoIP

ZyXEL Communications Czech s.r.o.  
Mod Fenská 621, 143 01 Praha 4  
Tel.: 241 091 350, Fax: 241 091 359  
<http://cz.zyxel.com>



▲ Viditelnost uzlů na úrovni síťové vrstvy.



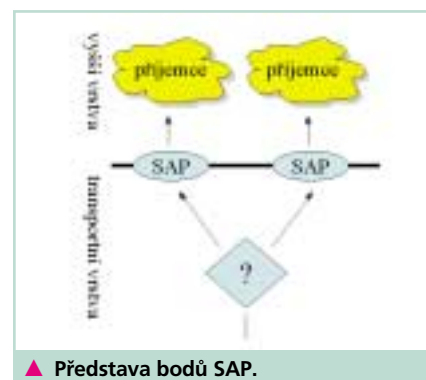
▲ Představa transportní vrstvy.

vrstev, orientovaných na přenos dat, a trojici nejvyšších vrstev, orientovaných na podporu aplikací. Je také první vrstvou, která je přítomna až v koncových uzlech a naopak není přítomna v uzlech mezilehlých, propojujících jednotlivé sítě (v tzv. směrovačích). Na rozdíl od nižších vrstev tedy nezajišťuje komunikaci mezi přímými sousedy, ale až komunikaci mezi dvěma koncovými uzly (tzv. end-to-end komunikaci).

Funguje-li transportní vrstva jako přizpůsobovací vrstva, pak přizpůsobuje možnosti a způsob fungování trojice nižších vrstev tomu, co požadují tři nejvyšší vrstvy. Rozdíl mezi možnostmi a požadavky může být například v tom, že nejnižší tři vrstvy fungují nespolehlivě (nepovažují za svou povinnost postarat se o nápravu případných chyb při přenosu), zatímco vyšší vrstvy požadují spolehlivý přenos. Nižší vrstvy také fungují tzv. nespojovaně (nenavazují spojení na začátku přenosu), zatímco vyšší vrstvy naopak chtějí fungovat spojovaně.

V prostředí sítí na bázi protokolů TCP/IP je tomu přesně tak: síťová vrstva (a na ní protokol IP) funguje nespolehlivě a nespojovaně. Na úrovni transportní vrstvy pak existují dva vzájemně alternativní protokoly, TCP a UDP. UDP nemění způsob fungování protokolů IP (také funguje nespojovaně a nespolehlivě), ale protokol TCP funguje spojovaně a spolehlivě. Vyšší vrstvy si pak mohou vybrat, kterou vrstvu budou používat.

Transportní vrstva má však ještě jeden důležitý úkol, rozlišování různých příjemců a odesílateľů v rámci každého uzlu. Ještě síťová vrstva se totiž dívala na každý uzel jako na dále nedělitel-



▲ Představa bodů SAP.

ný celek a nerozlišovala například mezi tím, že v rámci jednoho uzlu mohou přijímaná data „přít“ například WWW serveru, poštovnímu serveru nebo naopak WWW browseru, poštovnímu klientovi či nějaké jiné aplikaci.

Takové rozlišení různých odesílateľů a příjemců v rámci uzlu zajišťuje až transportní vrstva. Využívá k tomu přechodové body mezi sebou a bezprostředně vyšší vrstvou, které se v terminologii referenčního modelu ISO/OSI označují jako „body SAP“ (Service Access Point) a v terminologii TCP/IP jako tzv. porty.

Jednotlivé pakety pak specifikují své odesílatele a příjemce tím, že obsahují číselné identifikátory příslušných přechodových bodů. Přes ně je pak příslušný příjemce převezme, resp. odezdá.

### Relační vrstva

O relační vrstvě referenčního modelu ISO/OSI se říká, že vlastně nemá nic na práci a je tedy ve své podstatě zbytečná. Původně tomu ale tak nemělo být, relační vrstva měla zajišťovat řadu činností, spojených s „vedením relace“ mezi dvěma komunikujícími stranami. Například měla zajišťovat podporu transakcí nebo zabezpečení přenášených dat (jejich šifrování), případně i řízení toku a poloduplexnosti (aby například klient nezahltl server příliš mnoha požadavky).

Nakonec se ale ukázalo, že takové funkce buď nejsou vůbec zapotřebí, nebo sice zapotřebí jsou, ale stejně si je vyšší vrstva (hlavně vrstva aplikační) zajistí podle svého.

### Prezentační vrstva

Prezentační vrstva má již větší opodstatnění. Řeší totiž problém s tím, že stejný řetězec bitů může mít jiný význam pro odesílatele a jiný pro příjemce. Důvodem může být jiné kódování znaků – pokud si dvě strany posílají nějaký text, ale každá používá jiné kódování znaků, pak příjemce nebude přijatému textu vůbec rozumět. Podobně pro znázornění celých čísel, čísel s plovoucí desetinnou čárkou (a obecně pro všechny datové struktury), případně pro pouhé uložení vícebi-

tových dat v paměti. I zde existují dvě různé konvence (označované jako Little Endian a Big Endian), mezi nimiž je třeba řádně rozlišovat.

Pokud se bezprostředně nižší vrstvy snaží stůj co stůj přenést všechny bity tak, jak jsou, aniž by se na nich cokoli změnilo, pak to může být špatně a příjemce může přijatým datům rozumět jinak než jim rozuměl odesílatel. Proto jsou zapotřebí vhodné konverze přenášených dat – a právě tyto konverze má na starosti prezentační vrstva.

### Aplikační vrstva

Poslání aplikační vrstvy by na první pohled mohlo být jasné a přímočaré: v této vrstvě jsou provozovány jednotlivé aplikace. Není to ale pravda, a to z dosti kuriózního důvodu: pokud by se



▲ Představa aplikační vrstvy.

v aplikační vrstvě nacházely skutečně celé aplikace se vším všudy, pak by to znamenalo, že by jejich fungování muselo být standardizováno a tudíž i sjednoceno. V praxi by všechna okénka musela být stejná (standardizovaná), stejné (standardizované) by musely být i způsoby ovládní aplikací atd. To samozřejmě nemá smysl, protože by to bránilo autorům aplikací v rozvoji a v unikátnosti jejich produktů.

V aplikační vrstvě proto nejsou celé aplikace, ale jen jejich části – takové, které má smysl standardizovat a tím podřítit společným konvencím, aby si rozuměly s jinými implementacemi téže aplikace. Jsou to obvykle „základy“ aplikací, jako například mechanismy přenosu zpráv elektronické pošty. Naopak uživatelské rozhraní, sloužící k psaní zpráv, k jejich čtení, třídění, tisku, odpovídání atd. zajišťuje část aplikace, která již je „nad“ aplikační vrstvou a tudíž již nemusí být standardizována.