

Protokoll i flera skikt

Fragmentering

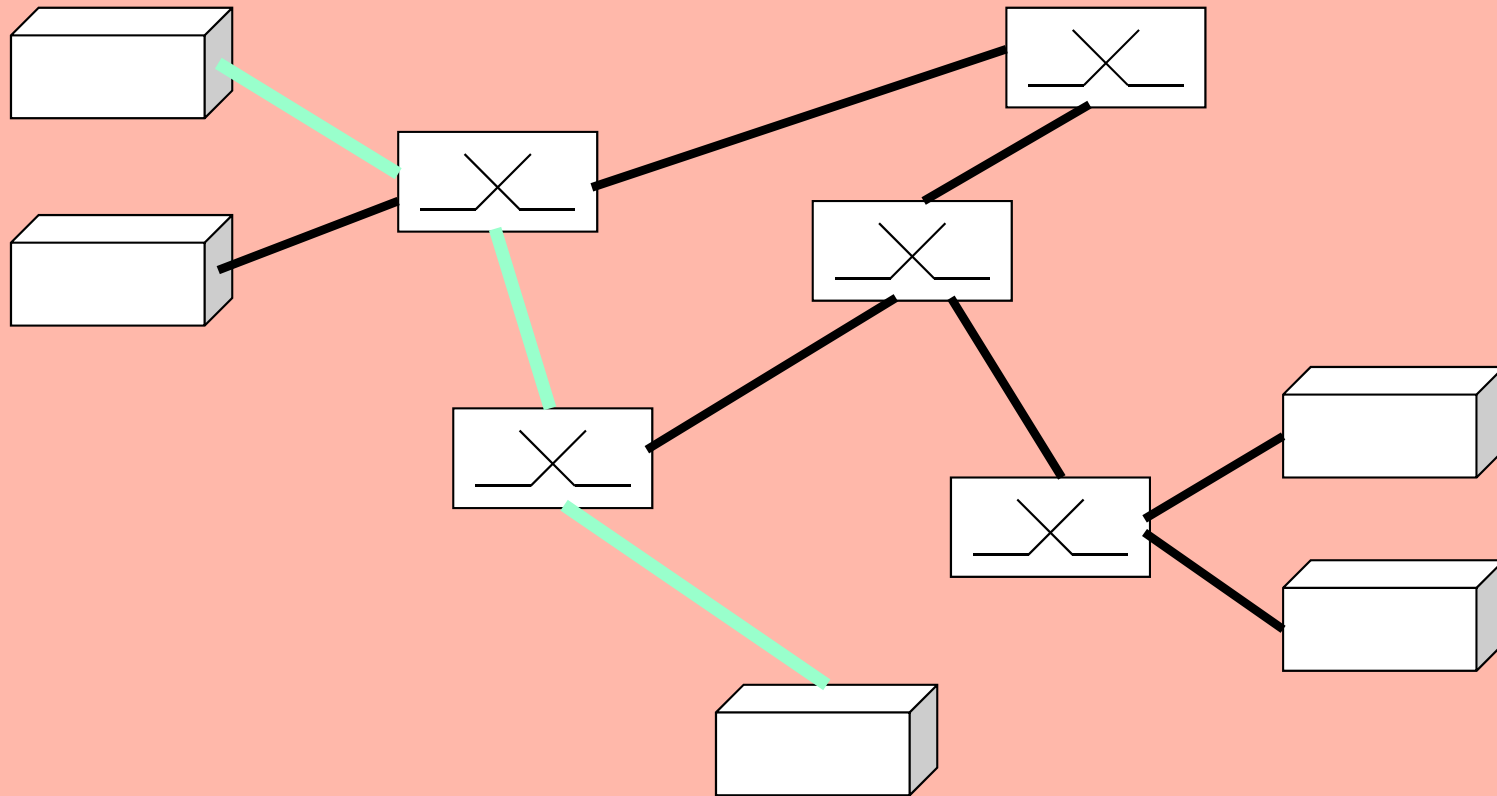
Vägval

DNS

Jens A Andersson

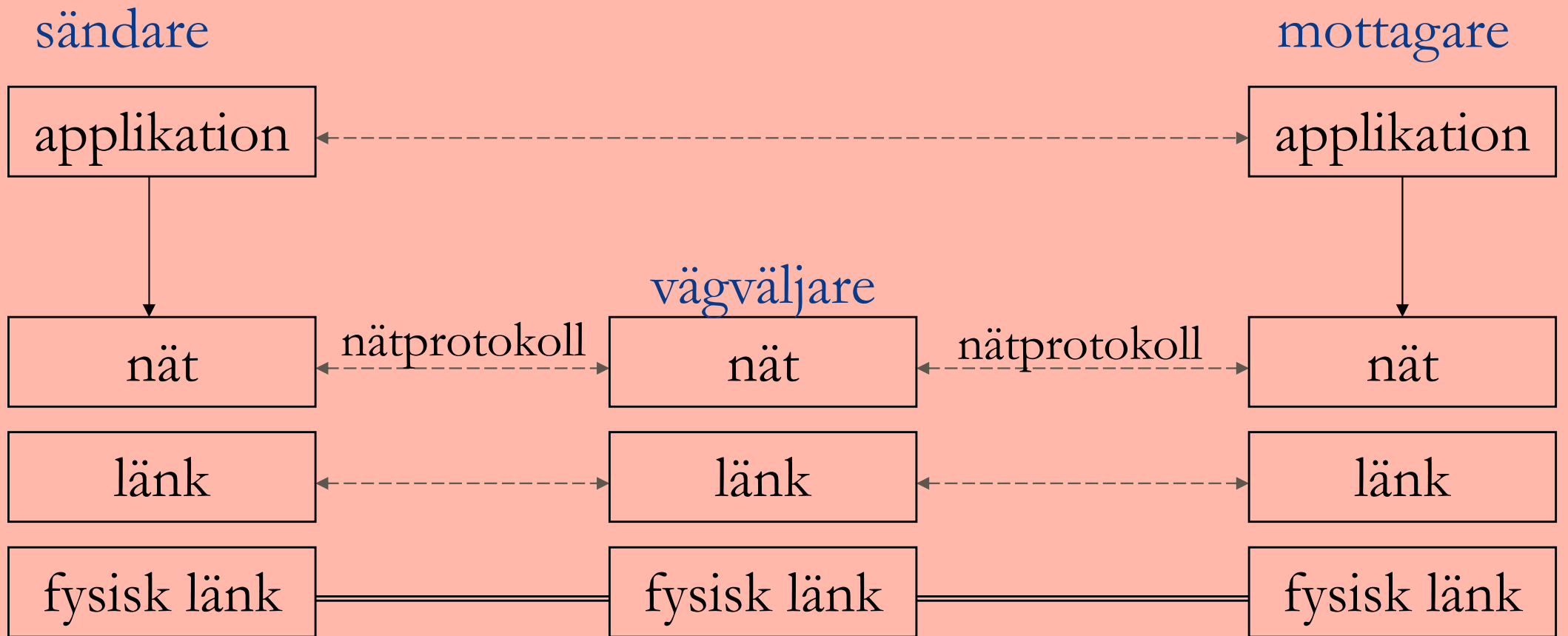


Att skicka data över flera länkar

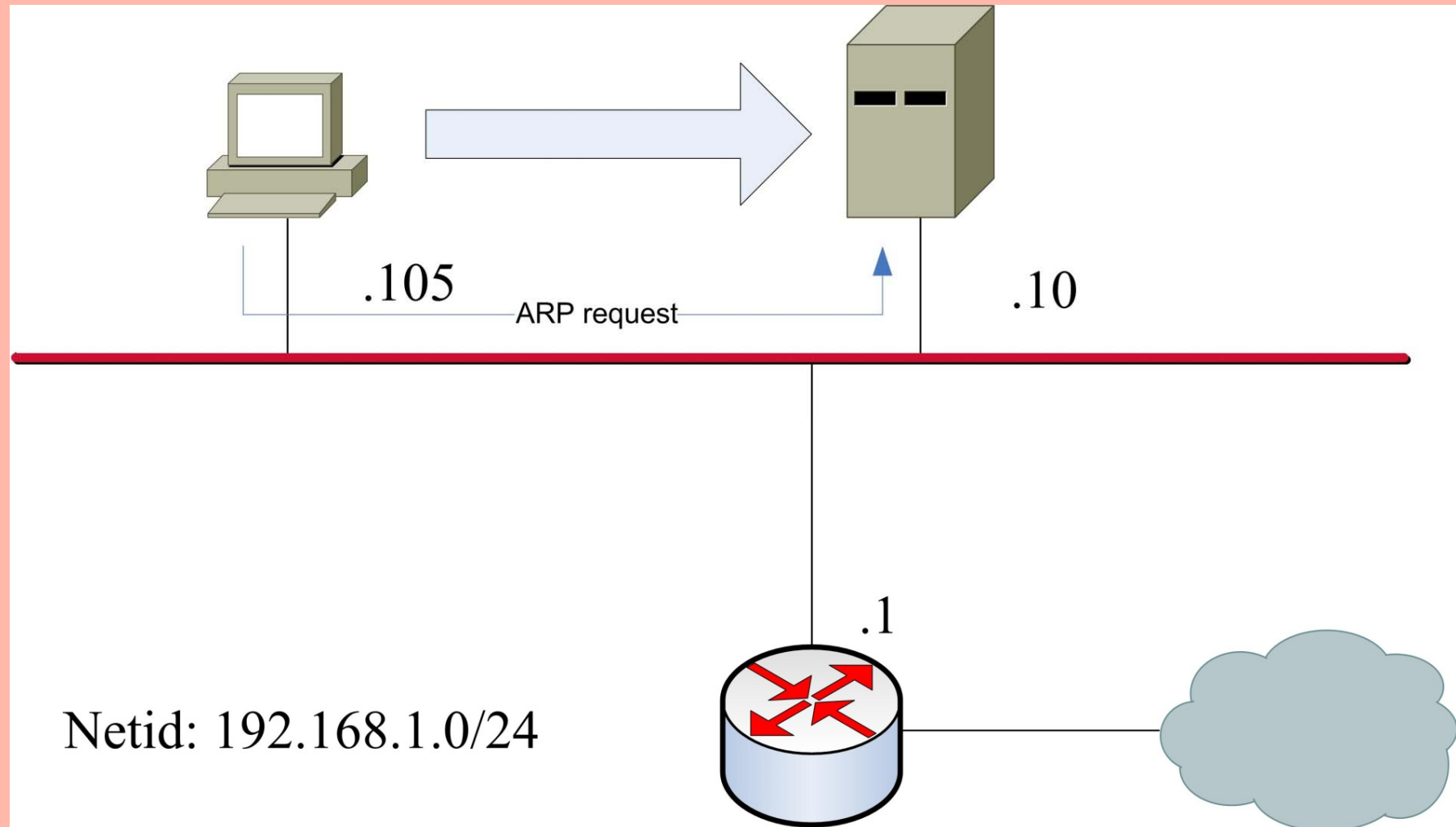


All data som skickas mellan två slutnoder kommer att passera flera vägväljare och länkar på vägen.

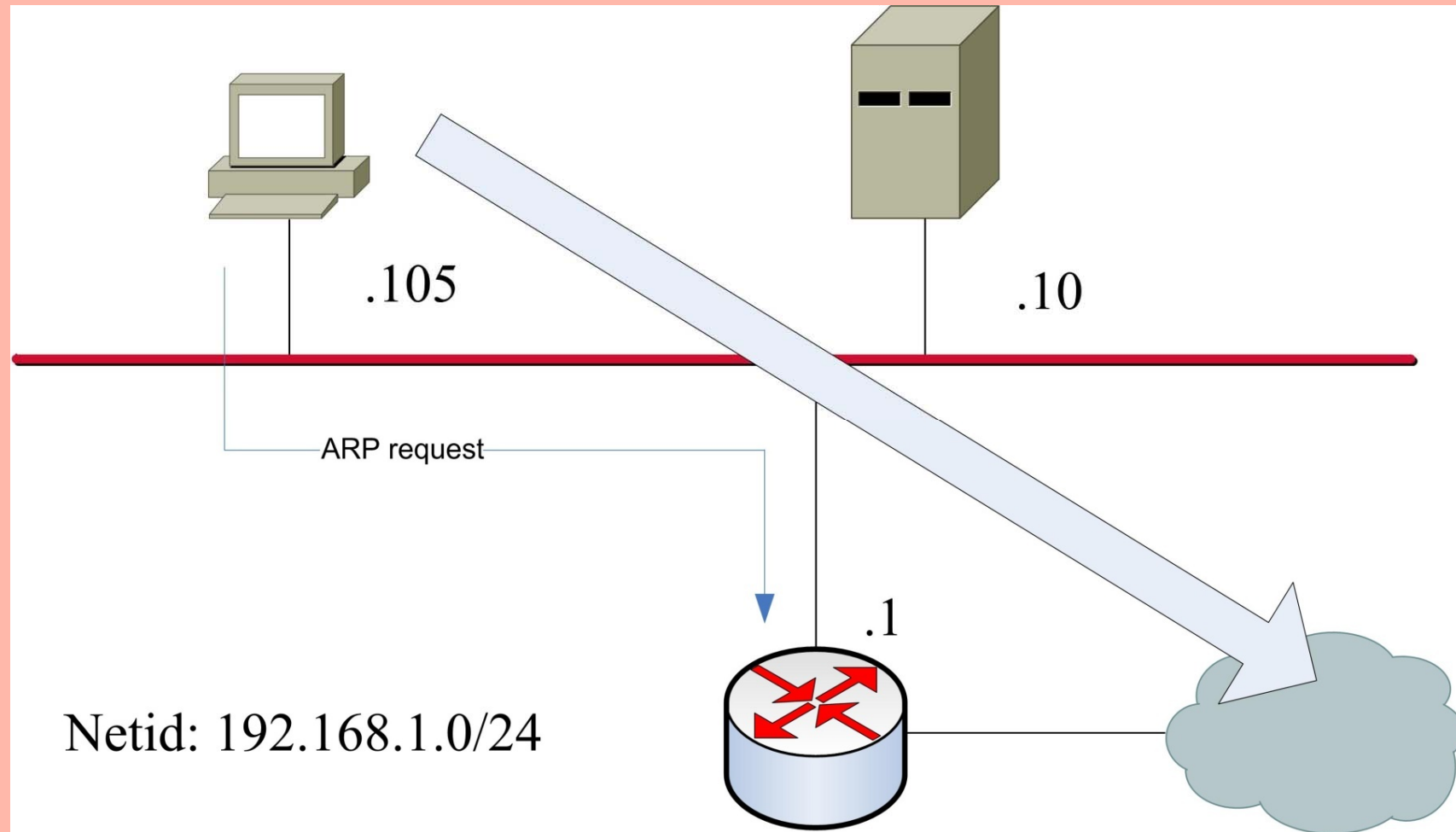
Nätprotokoll



ARP (1)



ARP (2)



IPv6 adresser (2)

- Hela adressrymden är indelad ett antal block
- Varje block igenkänns med ett *block prefix*
 - 001 = *global unicast* (vanlig dator-till-dator)
 - 1111 110 = *unique local unicast*, motsvarar privata IPv4-adresser
 - 1111 1110 11 = *link local* används vid autokonfigurering, kombineras med delar av MAC-adressen

IPv6-adresser (3)

- IPv6-adresser skrivs i 8 grupper om 4 hexadecimala siffror
- Varje grupp skiljs åt med : (kolon)
- Inledande 0or i grupp får strykas
- Flera grupper med 0or får föras samman
 - Noteras med tomt mellan kolon ::
 - Bara en sådan förkortning per adress

IP-adresser (IPv4)

IPv4-adresser består av 32 bitar.

Skrivs som fyra tal med punkter emellan, tex.

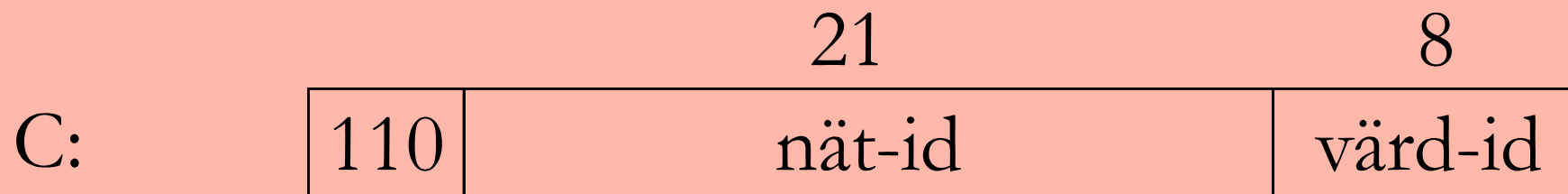
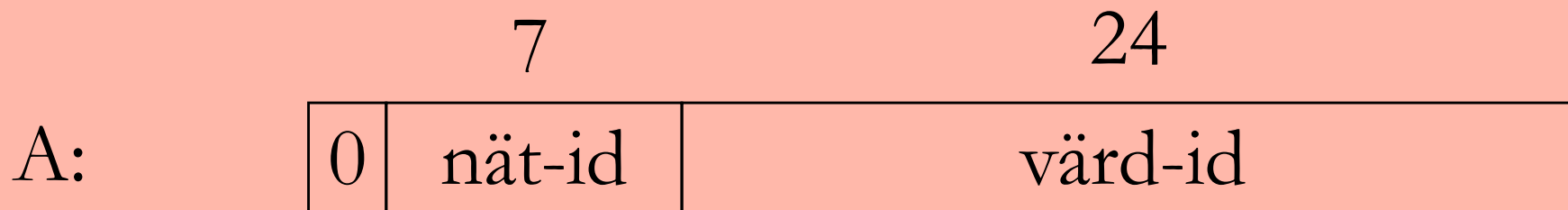
130.235.202.173

Två delar: en **nätidentitet** (nät-id) och en **värdidentitet** (värd-id).

Exempel: 130.235.200 är institutionens nätadress.

IPv4-adresser forts.

Det finns tre olika **adressklasser**: klass A, B och C.



Klasslös adressering

Det finns även **klasslös adressering**.

Man använder en så kallad **mask** för att ange vilka bitar som hör till nät-id respektive värd-id.

En etta i masken indikerar att adressbiten på motsvarande plats ingår i nät-id.

Adressexempel

Adress: 11011110 00010111 01000011 01000100

Mask: 11111111 11111111 11000000 00000000

Nät-id: 11011110 00010111 01000000 00000000

Värd-id: 00000000 00000000 00000011 01000100

IPv6-paket



40 bytes

0-65.535 bytes

Innehållet i IPv6 pakethuvudet

0 4 12 16 24 31

Vers.	Traffic class	Flow label	
Payload length		Next header	Hop limit
Sändaradress (16 bytes)			
Mottagaradress (16 bytes)			

IPv4-paket



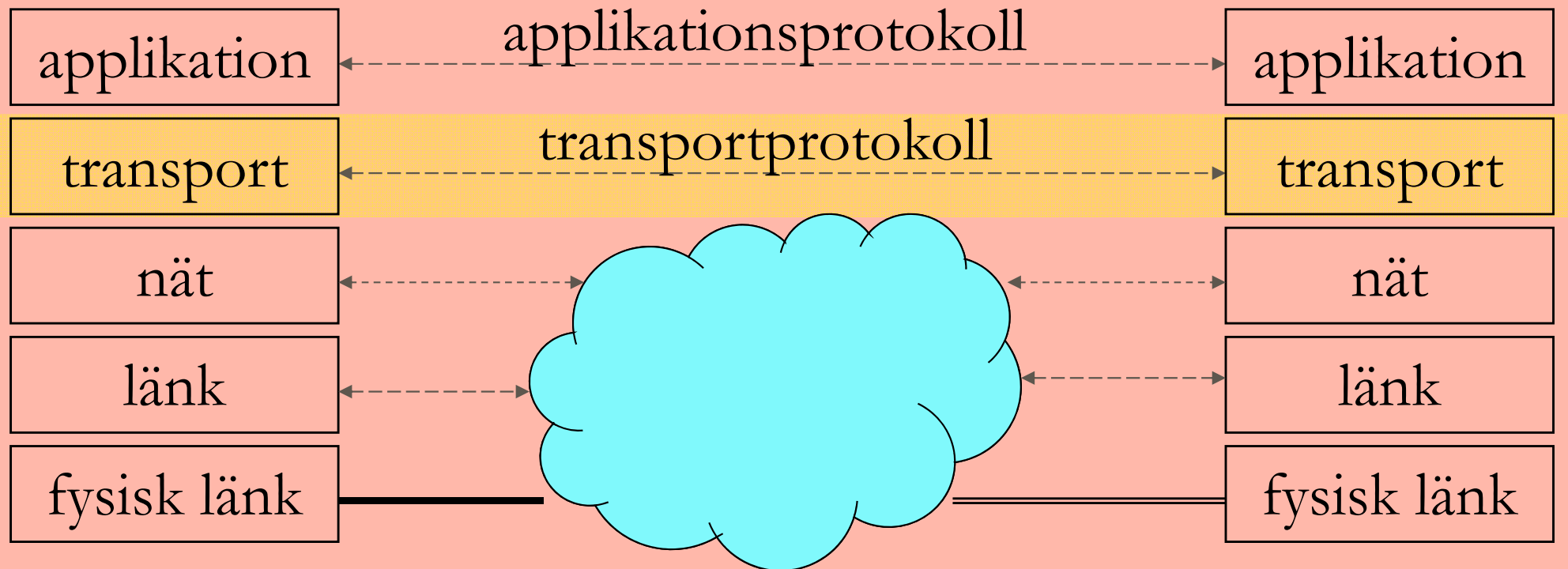
20-60 bytes

0-65.516 bytes

Innehållet i IPv4 pakethuvudet

0	4	8	16	31
vers.	hl.	typ	paketlängd	
sekvensnummer			frg.	fragmentposition
livstid	protokoll		kontrollsumma	
sändaradress				
mottagaradress				
ev. tillval			utfyllnad	

Transportprotokoll



Protokoll i flera skikt

Genom att använda flera protokoll som har sina specifika uppgifter kan dataöverföringen bli enklare och mer flexibel.

Det blir lättare att utveckla nya applikationer och protokoll eftersom de inte behöver kunna allt.

OSI-modellen (1)

I mitten på 70-talet startade ISO ett projekt för att standardisera datorprotokollen.

1983 presenterades Open Systems Interconnection (OSI) Reference Model.

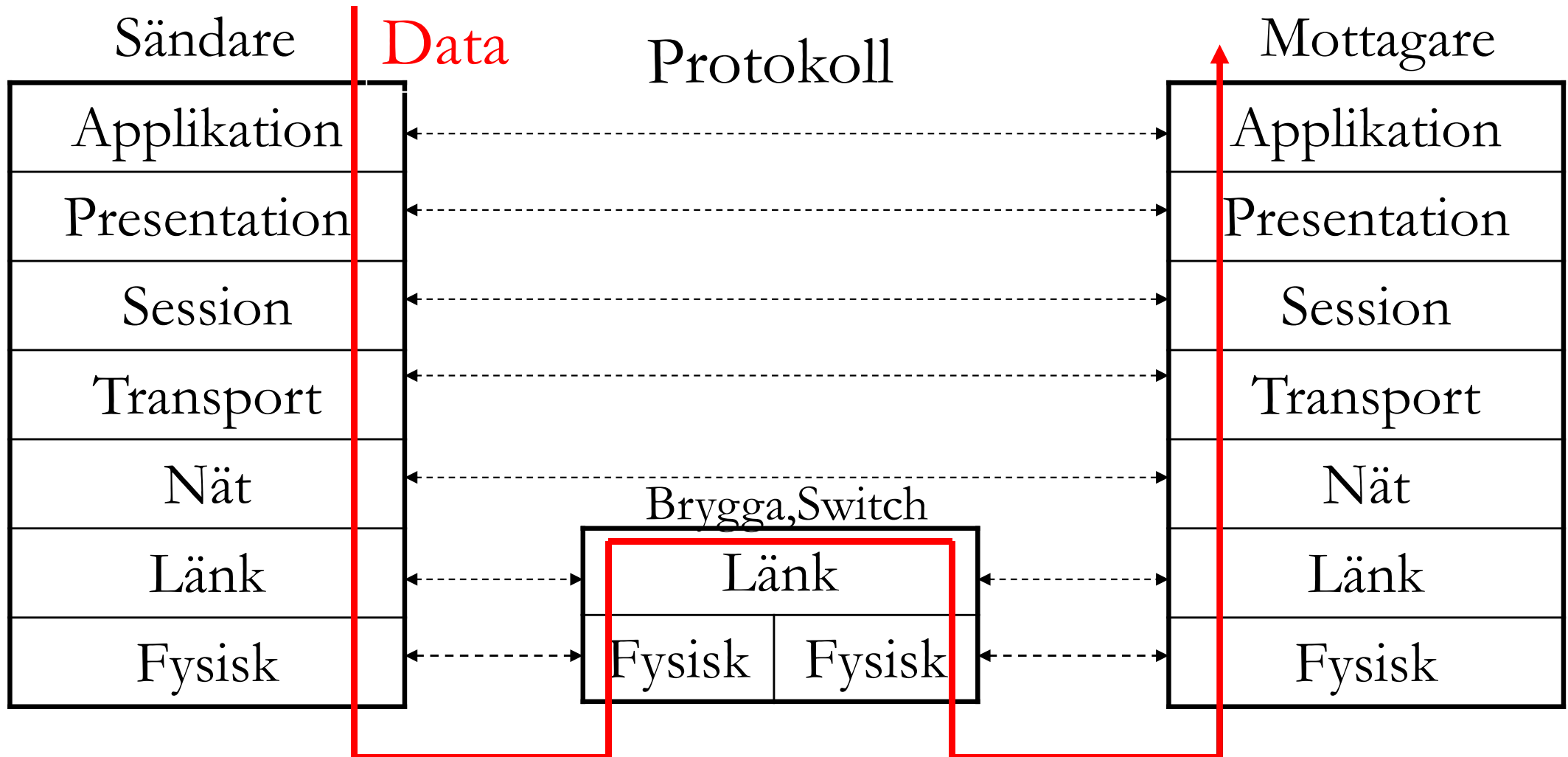
OSI-modellen är modell (framework) för hur datorprotokoll skall utvecklas.

OSI-modellen (2)

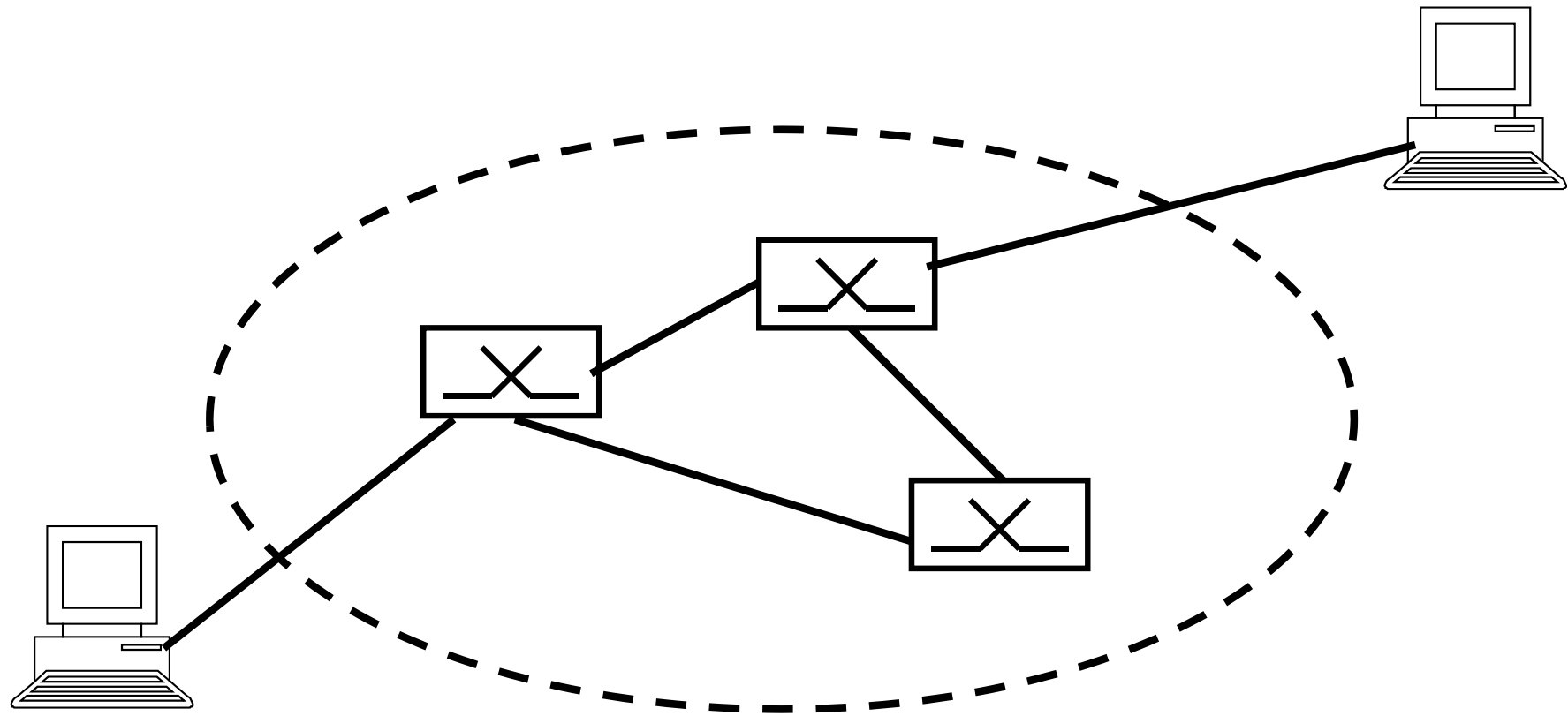
OSI-modellen innehåller 7 skikt el. nivåer (layers).

Applikation	7
Presentation	6
Session	5
Transport	4
Nät	3
Länk	2
Fysisk	1

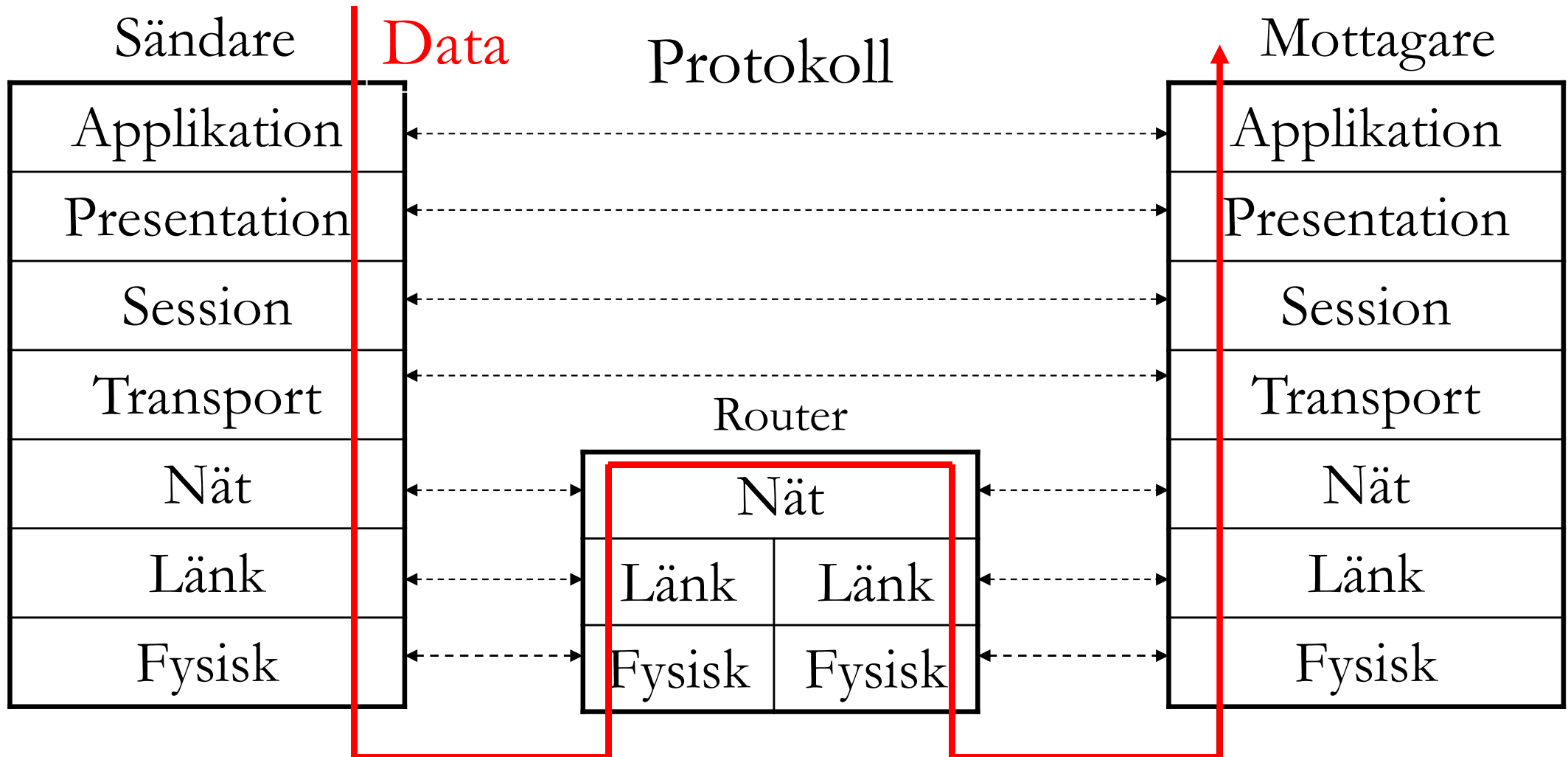
Protokollstack i OSI-modellen (1)



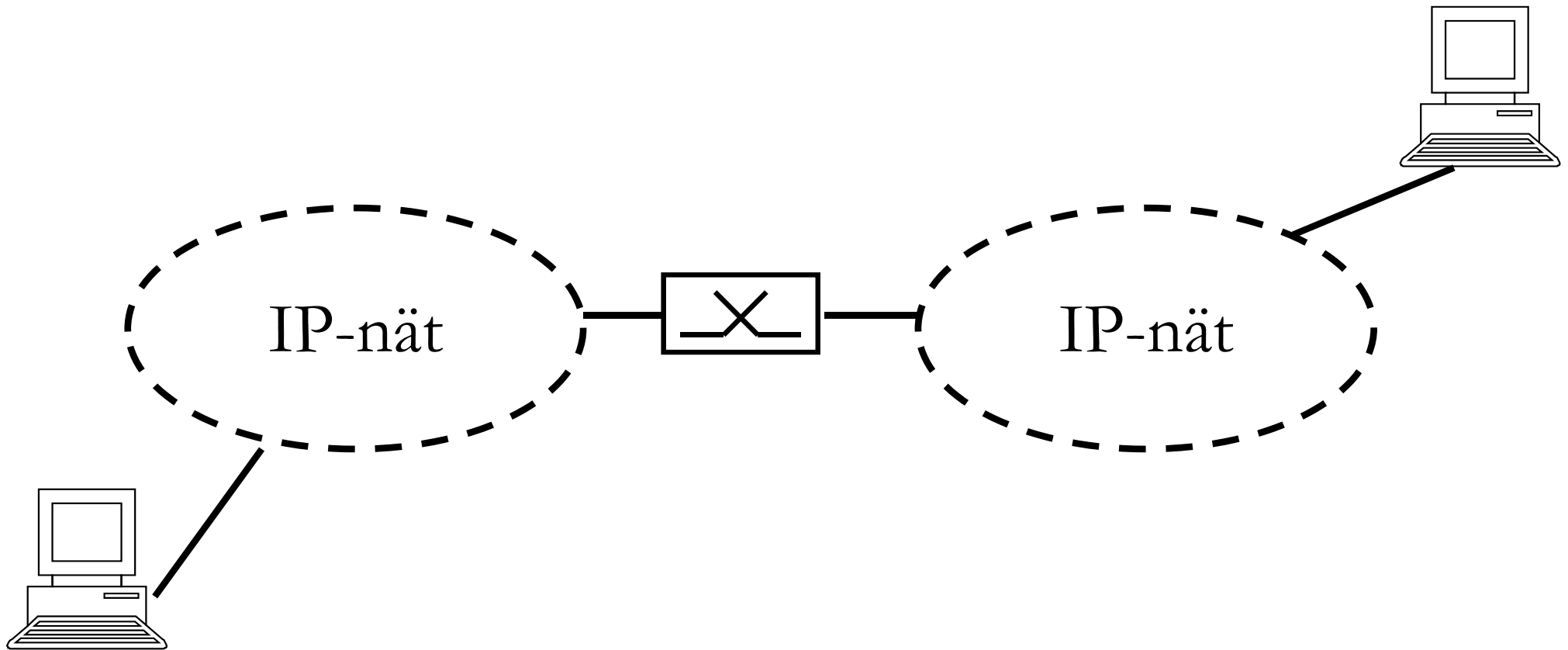
Switchar används inom ett nät



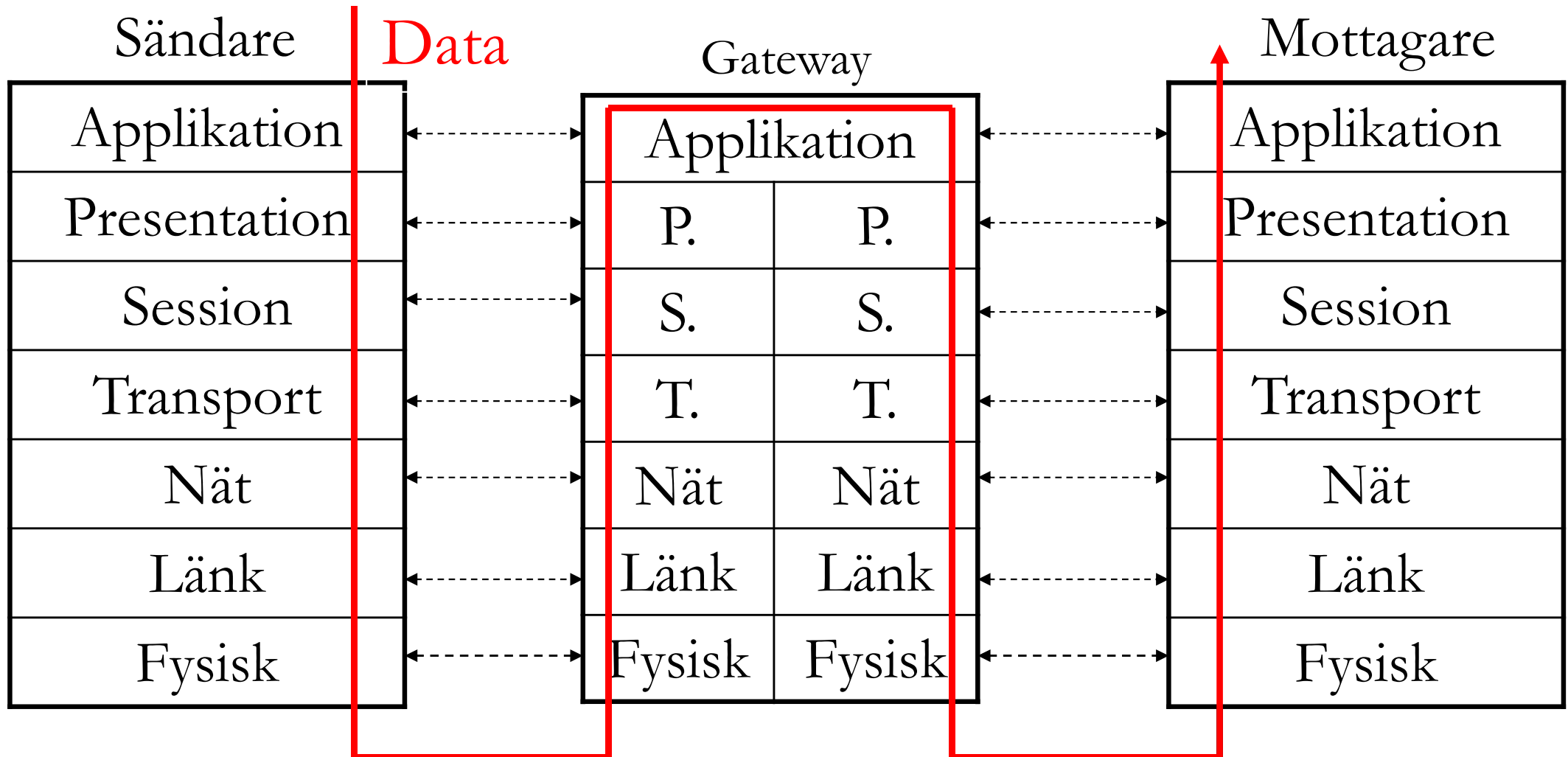
Protokollstack i OSI-modellen (2)



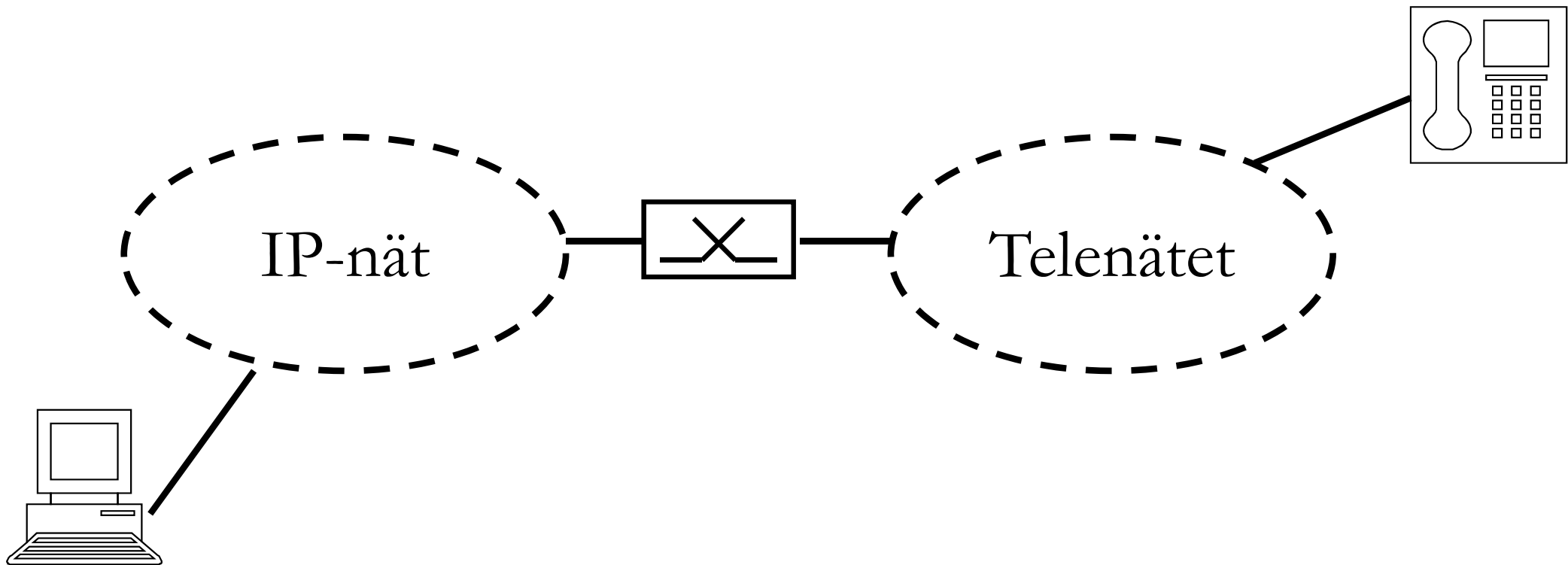
Routers används mellan nät med samma nätprotokoll.



Protokollstack i OSI-modellen (3)

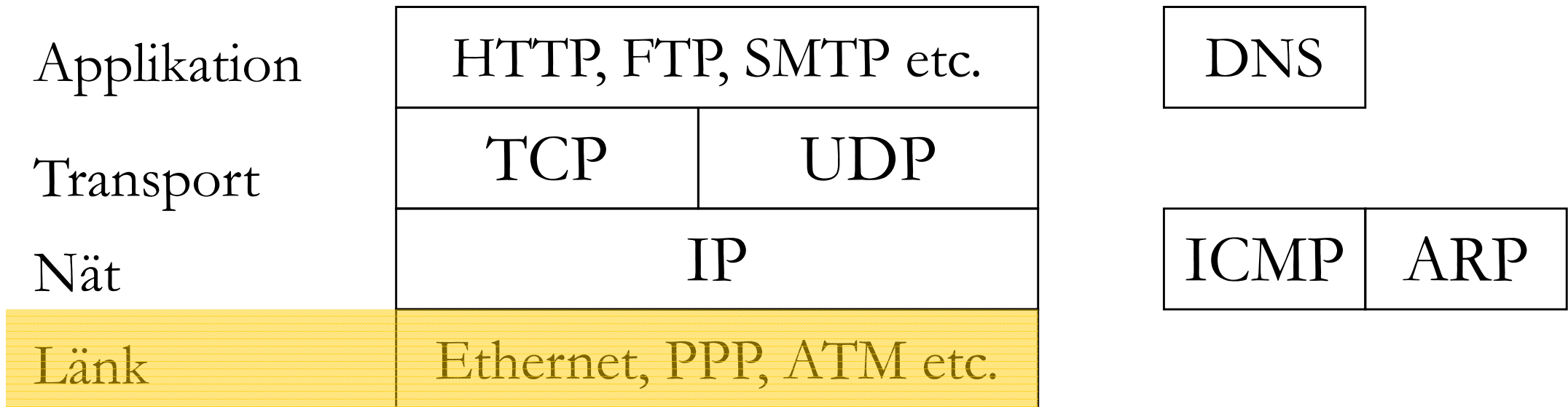


Gateways används mellan nät med olika applikationsprotokoll



Internets protokollstack

Internet har endast 3 skikt i sin protokollstack.



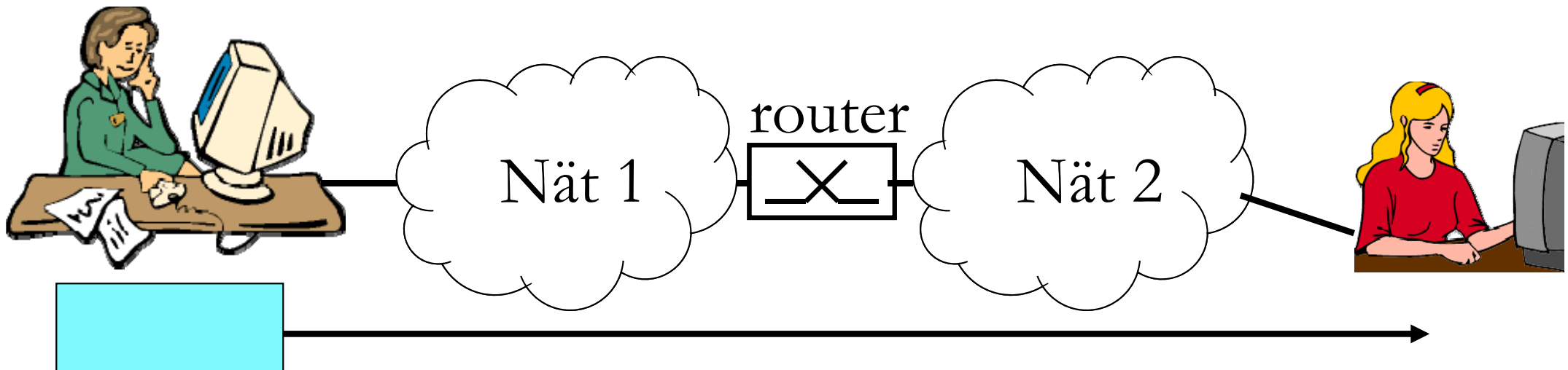
Jämförelse med OSI-modellen

OSI-modellen	TCP/IP-modellen
Applikation	Applikation
Presentation	
Session	
Transport	Transport
Nät	Nät
Länk	IP-bärande nät
Fysisk	

Vad händer med informationen?

Antag att ett användarapplikation har ett **meddelande** den vill sända till en annan användare.

Vad händer med detta meddelandet i de olika protokollen på vägen från sändare till mottagaren?

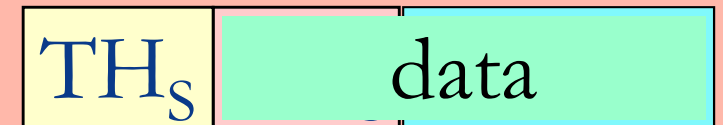


Sändarsidan

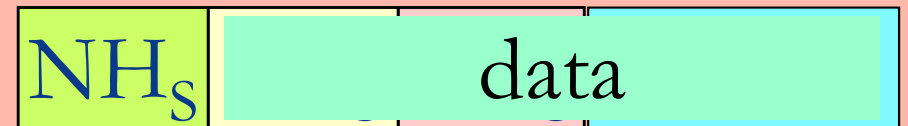
applikation



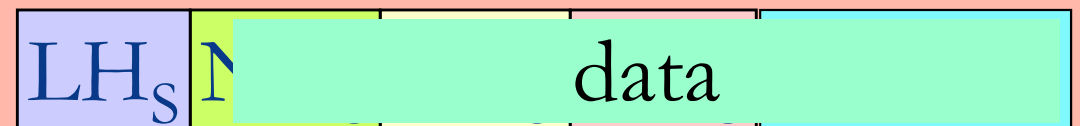
transport



nät



länk



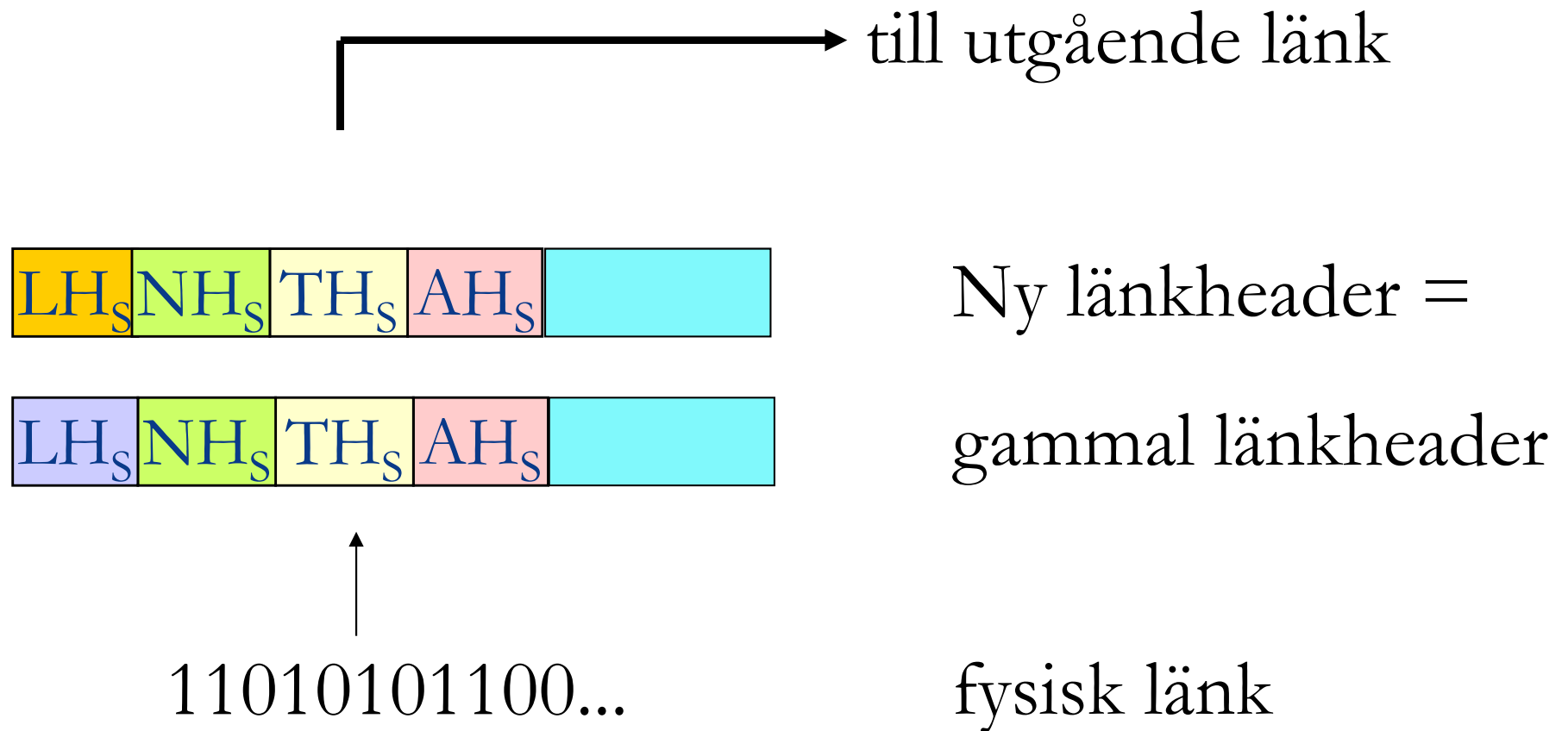
fysisk länk

11010101100011100011....



Switchar

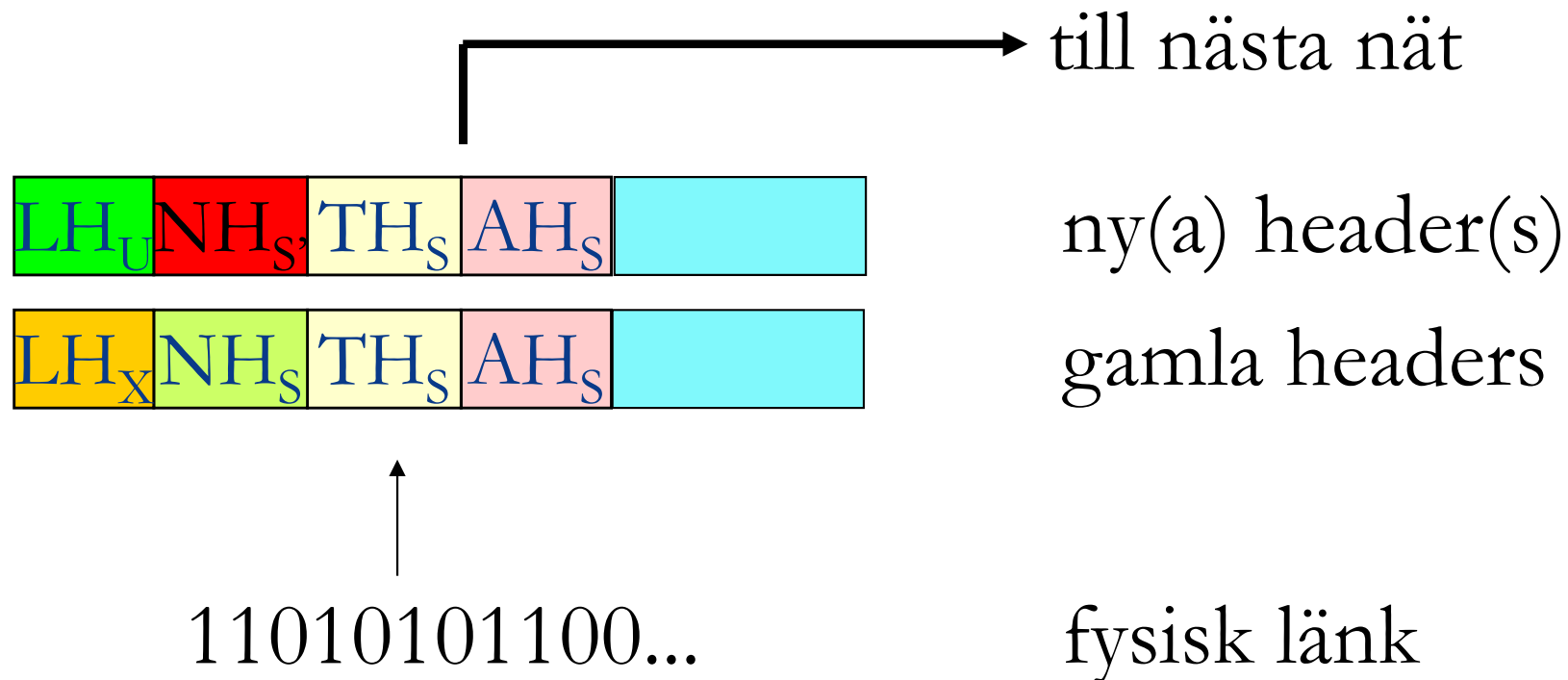
Ethernet-switchar ändrar normalt inte någon header.



Router

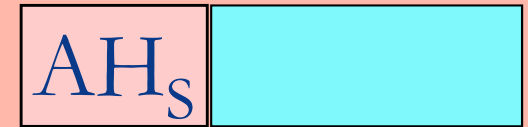
Routern gör en ny länk-header som passar.

Nät-header är *i princip(?)* oförändrad.



Mottagarsidan

applikation



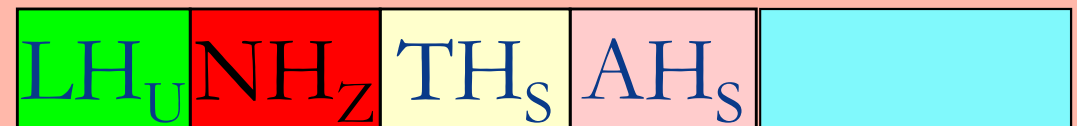
transport



nät



länk



fysisk länk

110100111011000011....



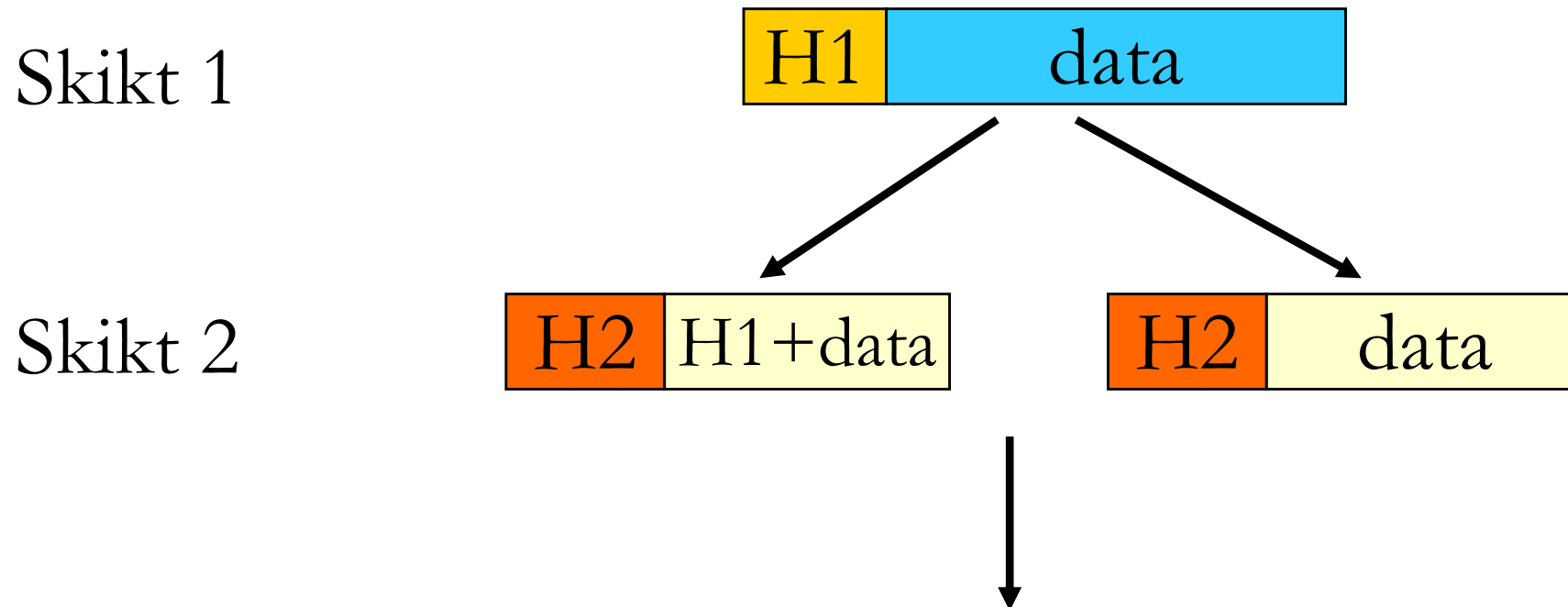
MTU

Maximum Transmission Unit

- Hur mycket data en ram kan bära
- Exempel:
 - Ethernet MTU = 1500
 - PPP länk över modem = 256/512
 - Gigabit Ethernet Jumbo Frames = 9600
 - IPv4 = 65536 inkl header
 - IPv6 = 65536 exkl header

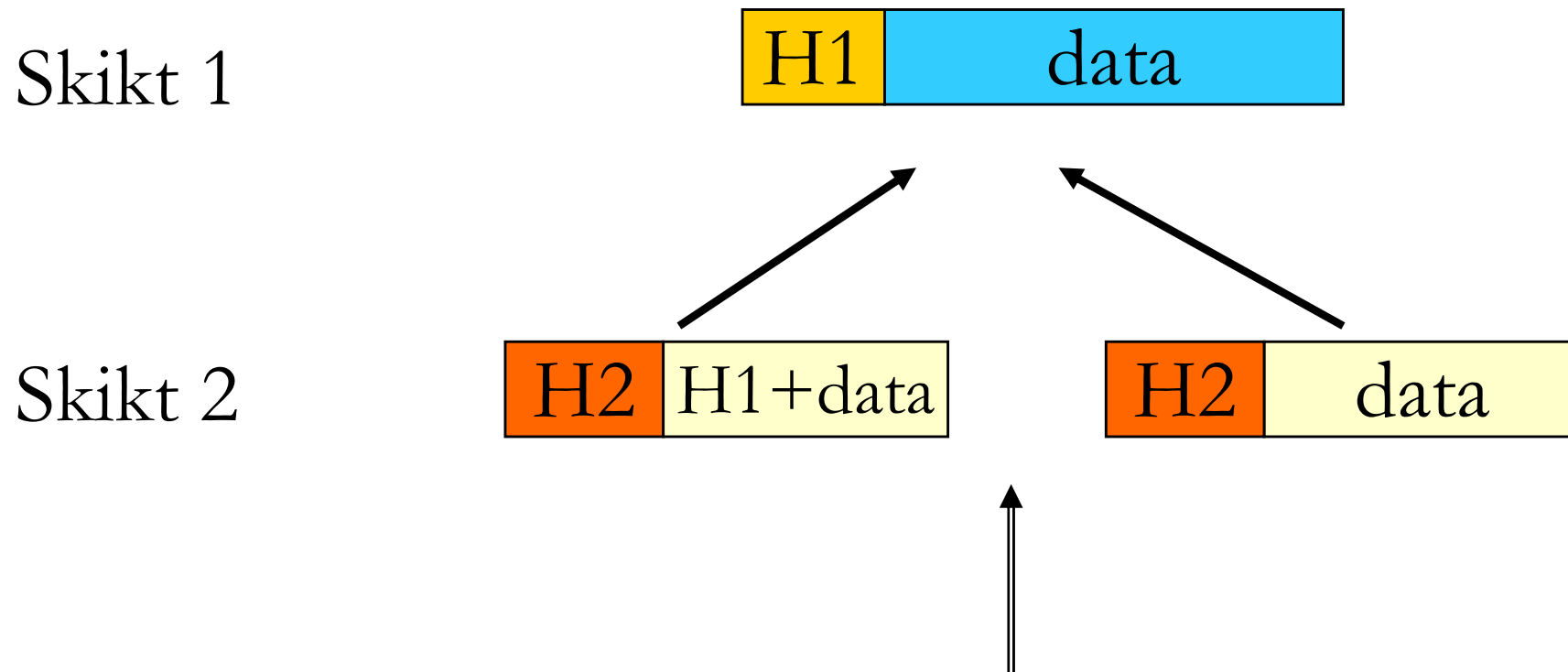
Fragmentering

Om det kommer data från ett övre skikt som inte får plats i ett enda datapaket sker så kallad **fragmentering**.



Hopsättning

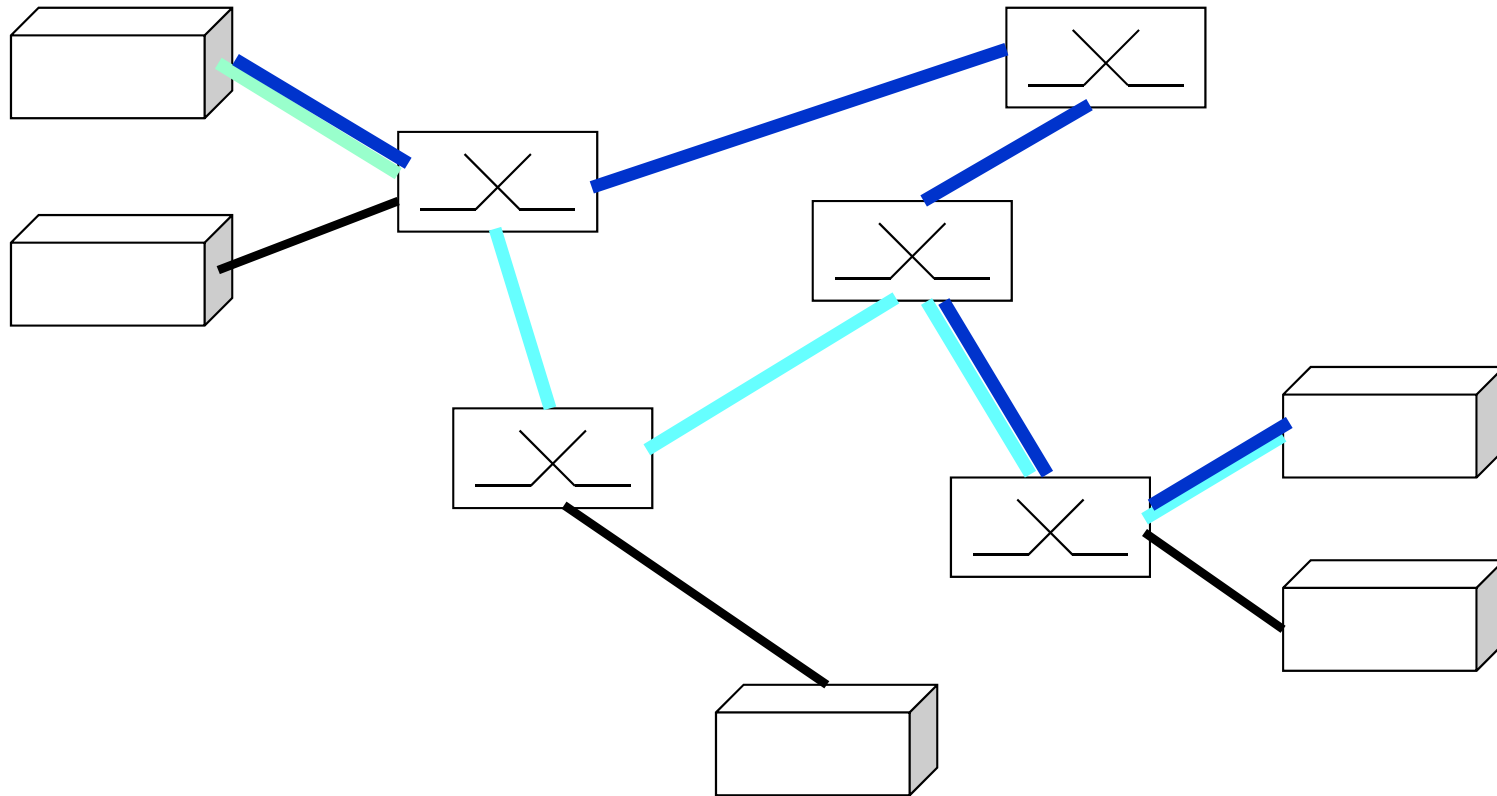
På mottagarsidan sätts datan ihop igen.



Fragmentering på nätnivån

- Om länknivåns ram är mindre än IP-paketet måste IP-paketet fragmenteras
- I IPv6 får bara ursprunglig avsändare fragmentera.
- I IPv4 får varje vägväljare/router fragmentera.
- Defragmentering görs alltid i mottagaren.

Hur väljs en väg för paketen?



I stora datanät finns det oftast flera möjliga vägar för ett paket.

Vägvalsalgoritmer

Vägväljarna använder en [vägvalsalgoritm](#) för att hitta en väg genom nätet.

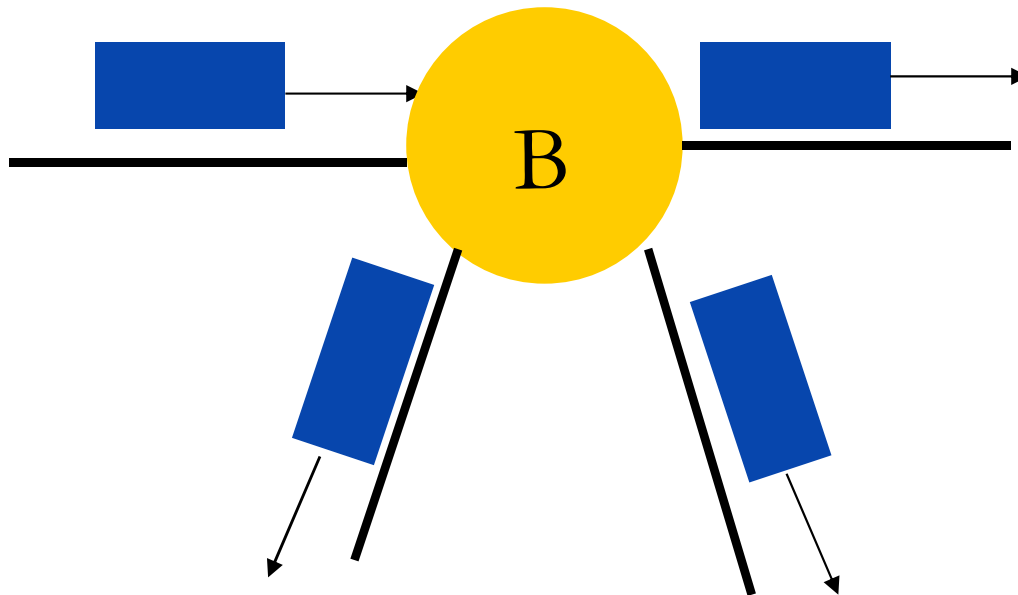
Syftet med vägvalsalgoritmen är att hitta den bästa möjliga vägen genom nätet för varje sändar-mottagarpar.

Vägvalsalgoritmen måste också klara av förändringar i nätet.

Flooding

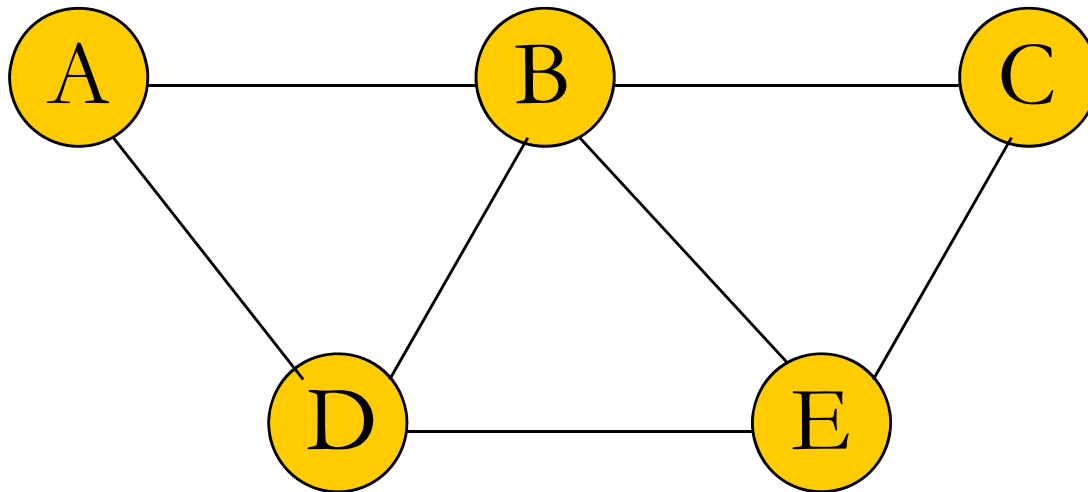
I Flooding skickas ett inkommande paket ut på samtliga länkar.

En hop-count används för att inte skapa loopar.



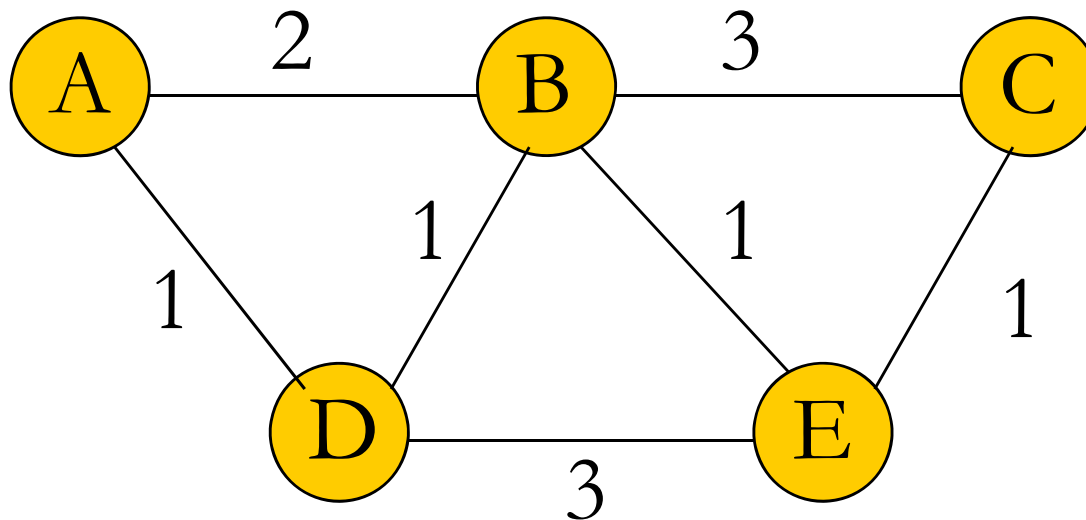
Nätgraf

I vägvalsalgoritmerna används en nätgraf som består av noder och länkar.



Nätgraf forts.

Varje länk i grafen har en kostnad som anger hur "dyrt" det är att skicka ett paket över länken.



Länkkostnad

Länkkostnaden kan bero på flera saker:

⌘ Kapacitet

⌘ Belastning

⌘ Sträcka

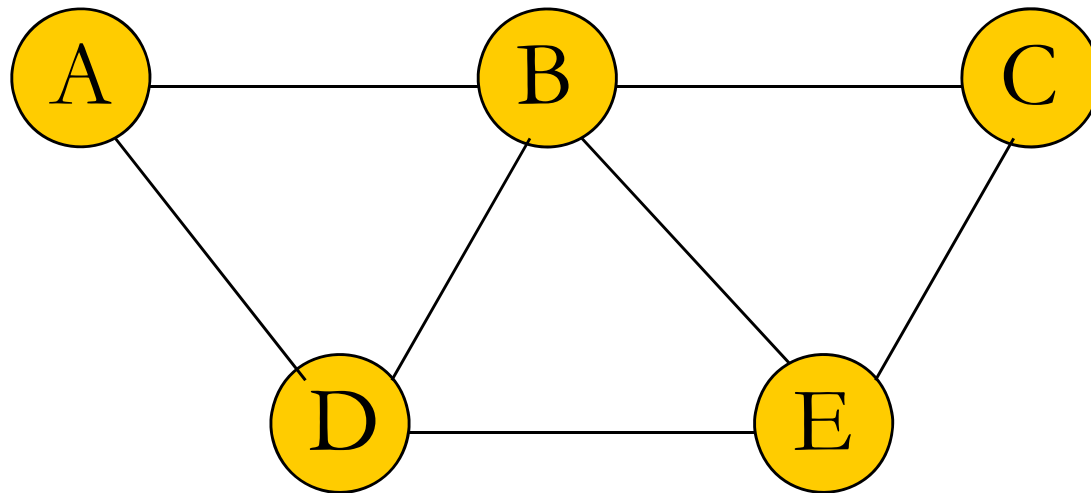
⌘ Utbredningsmedium

⌘ Osv...

Least-hop path

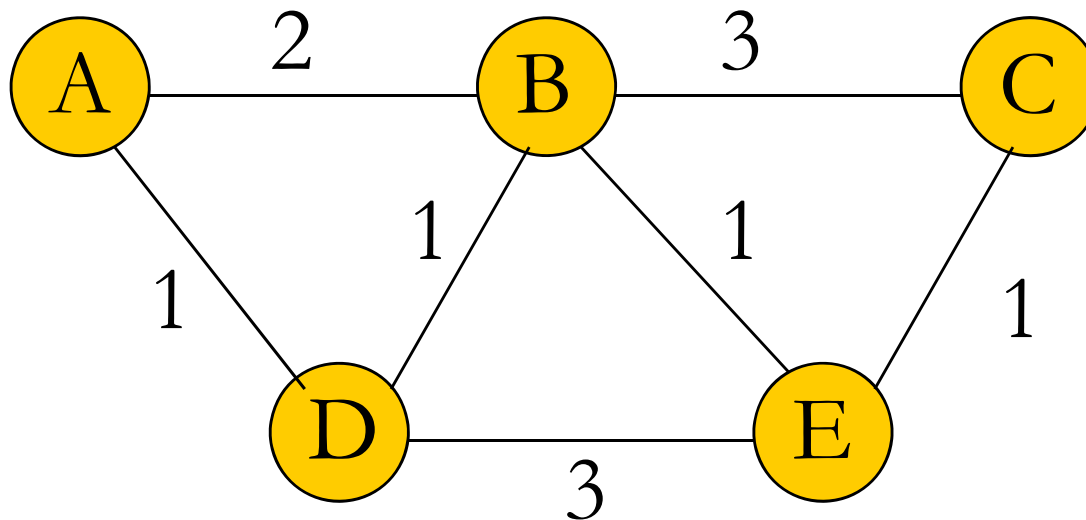
Least-hop path fungerar bäst om alla länkar har samma kostnad.

Den väg som innehåller minst antal steg är bäst.



Least-cost path

I Least-cost path väljs de vägar ut som kostar minst.



Symboliska namn

Eftersom IP-adresserna inte är speciellt användarvänliga, har man utvecklat ett system med [symboliska namn](#) på varje värddator eller tjänst.

Tex. min Unix-dator har IP-adressen 130.235.201.50 men heter också [lindyhop.eit.lth.se](#)

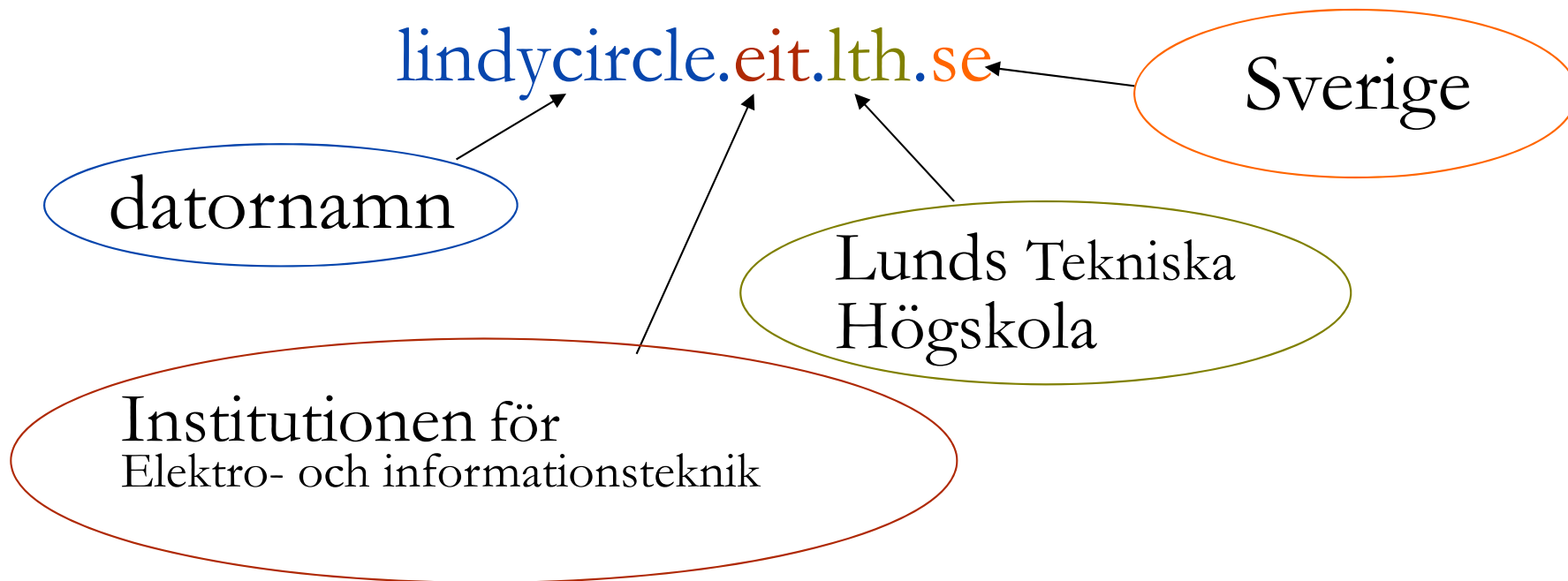
De symboliska namnen är hierarkiska och specificeras med hjälp av [Domain Name System \(DNS\)](#).

Domain Name System (DNS)

- Hierarkiskt namnsystem som består av ett antal nivåer.
- Internet delas in i ett antal **domäner** och varje domän får sin egen kod.
- Överst finns geografiska och organisatoriska domäner.
- Varje domän delas sedan in i underdomäner.

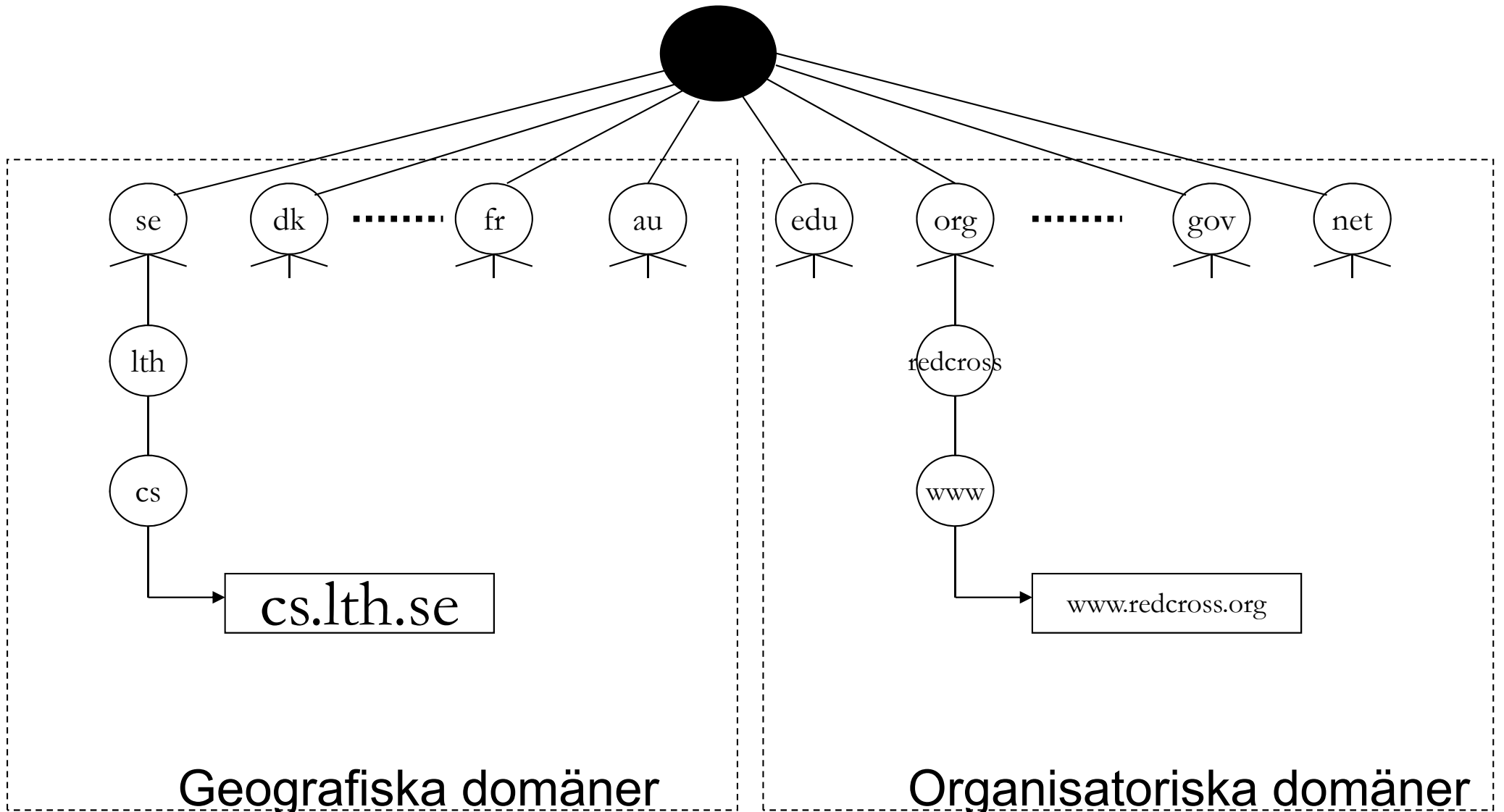
DNS-exempel

Det symboliska namnet för min dator är



Domäner

Rotdomän



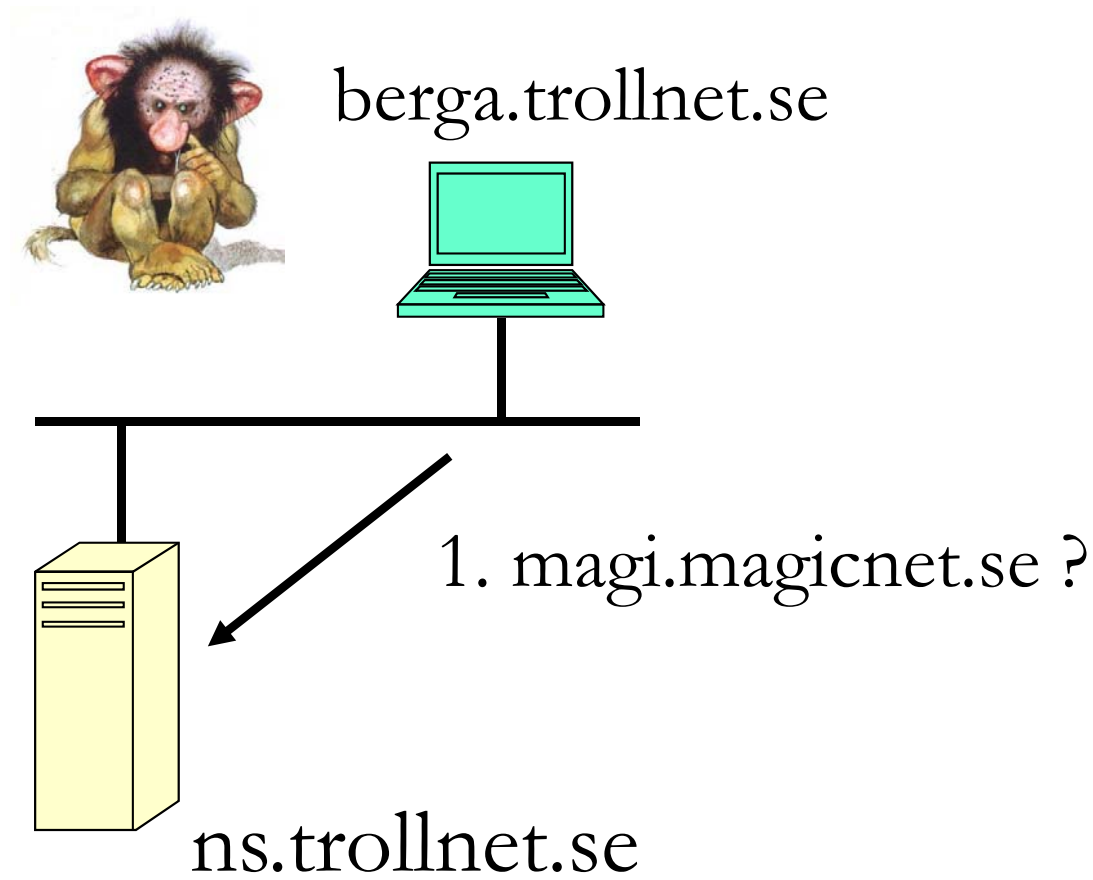
Från datornamn till IP-adress (1)

Eftersom de symboliska namnen inte går att använda som IP-adress, måste de översättas för att informationen skall nå rätt mottagare.

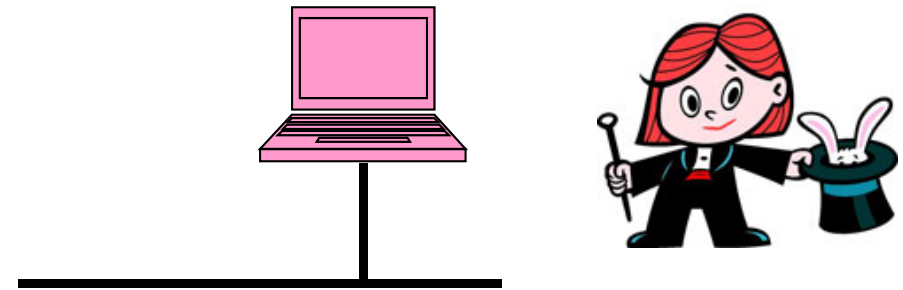
Det finns speciella databaser som har översättningstabeller.

Databaserna kallas för name servers.

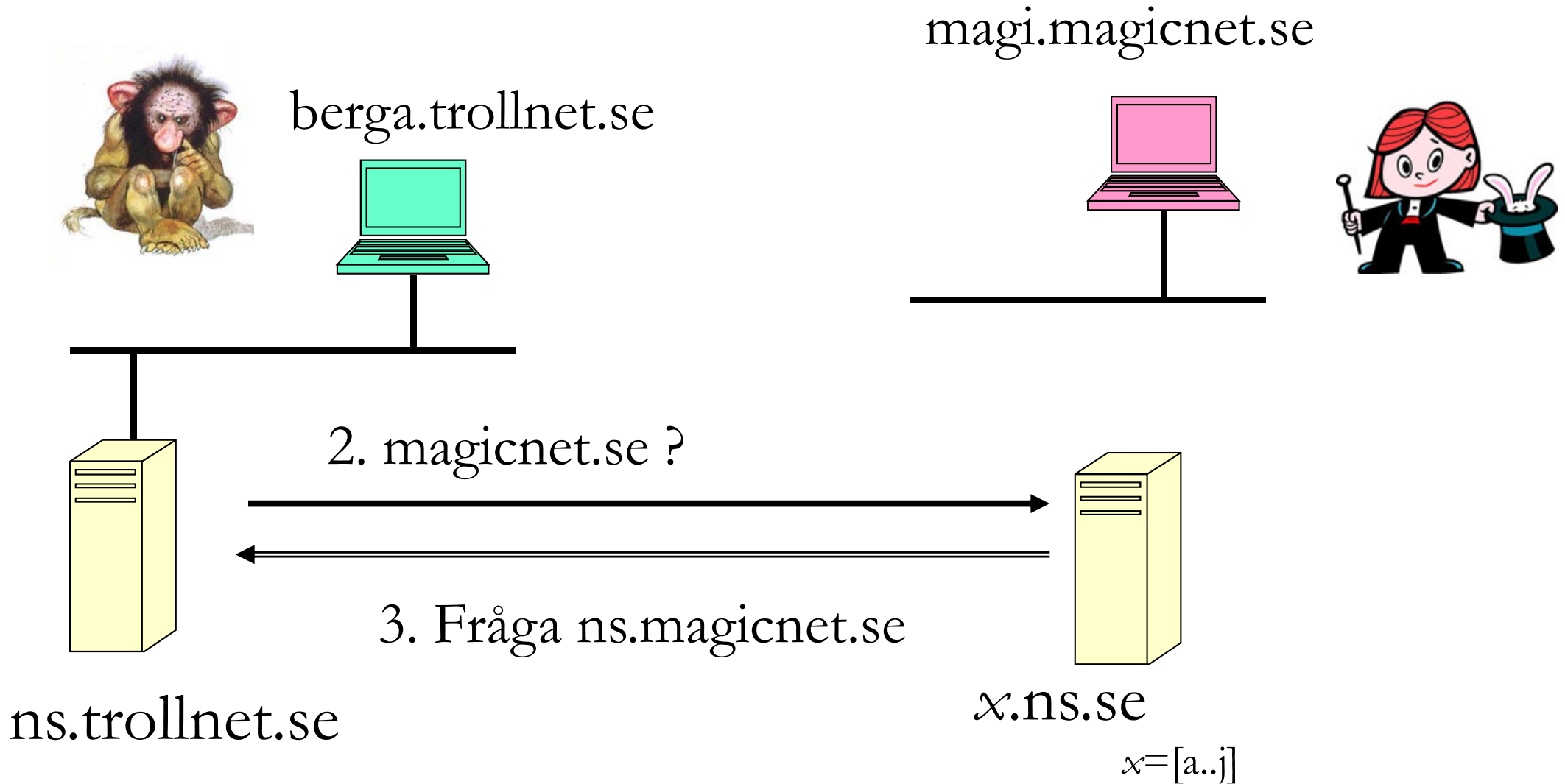
Från datornamn till IP-adress (2)



magi.magicnet.se



Från datornamn till IP-adress (3)



Från datornamn till IP-adress (4)

