
Nivelul legătură de date

Nivelul fizic mută biți, dar nu știe cui. Nivelul legătură de date rezolvă exact această lipsă: împachetează biții în cadre, dă fiecărei interfețe un nume unic pe lume, stabilește cine are voie să vorbească pe un mediu partajat și - prin switch - construiește singur harta rețelei locale, fără ca nimeni să îi spună nimic.

Din capitolul precedent.

- Nivelul **1, fizic** transformă biți în semnal: tensiuni, lumină sau unde radio.
- Mediul poate fi **ghidat** (cupru, fibră) sau **neghidat** (radio). Un mediu partajat înseamnă că mai mulți vorbesc pe același suport.
- Fiecare nivel adaugă propriul **antet** - proces numit **încapsulare**.
- Cursul acesta urcă la **nivelul 2**: adrese MAC, cadre, comutare.

OBIECTIVELE CAPITOLULUI

- Să explicați rolul nivelului 2 și separarea lui în subnivelurile LLC și MAC
- Să citiți un cadru Ethernet câmp cu câmp și să spuneți la ce folosește fiecare
- Să recunoașteți o adresă MAC unicast, multicast sau broadcast
- Să explicați CSMA/CD și să spuneți de ce a dispărut din rețelele moderne
- Să numărați corect domeniile de coliziune și de broadcast dintr-o topologie
- Să descrieți cele trei acțiuni ale unui switch: învățare, comutare, inundare
- Să comparați metodele de comutare store-and-forward și cut-through

3.1 Ce îi lipsește nivelului fizic

Nivelul fizic gestionează un flux de biți. Atât. Are trei lipsuri importante, iar nivelul 2 există exact ca să le compenseze:

Tabelul 3.1 Ce îi lipsește nivelului fizic.

Nivelul fizic nu poate...	Nivelul legătură de date...
să identifice stațiile de pe mediu	le identifică prin adresare
să spună unde începe și unde se termină un mesaj	organizează fluxul în cadre
să comunice direct cu software-ul	oferă un serviciu clar nivelului 3

De ce e nevoie de „cadre”. Imaginați-vă că primiți un șir nesfârșit de litere, fără spații și fără semne de punctuație. Ca să înțelegeți ceva, trebuie să știți unde se termină un cuvânt și unde începe altul.

Un **cadru** (în engleză *frame*) face exact asta pentru biți: îi grupează într-un pachet cu început clar, sfârșit clar, adresa destinatarului și o sumă de control care spune dacă a ajuns întreg.

3.2 Cele două subniveluri: LLC și MAC

Nivelul 2 stă exact la granița dintre hardware și software, așa că a fost împărțit în două subniveluri, fiecare privind în altă direcție.

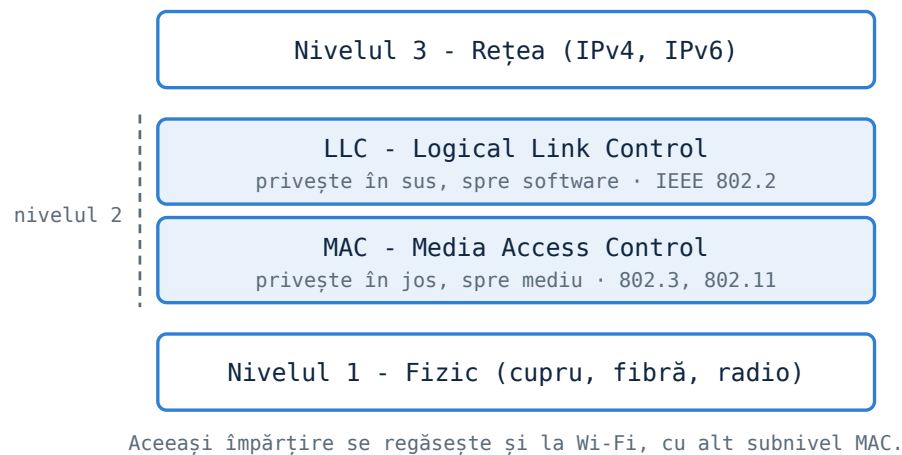


Figura 3.1 Nivelul 2, văzut din interior. LLC este independent de mediu; MAC depinde complet de el.

Tabelul 3.2 Cele două subniveluri: LLC și MAC.

Subnivel	Cu cine vorbește	Ce face	Standard
LLC	cu software-ul (nivelul 3)	multiplexează protocoalele de nivel superior, oferă control de flux; nu îl interesează mediul	IEEE 802.2
MAC	cu mediul (nivelul 1)	construiește cadrul, decide cine transmite și când; depinde complet de tehnologie	802.3, 802.11, FDDI

De ce împărțirea asta contează. Datorită subnivelului LLC, același pachet IP circulă neschimbat peste Ethernet, peste Wi-Fi sau peste PPP. Schimbarea mediului cere doar înlocuirea subnivelului MAC - restul stivei nu află nimic.

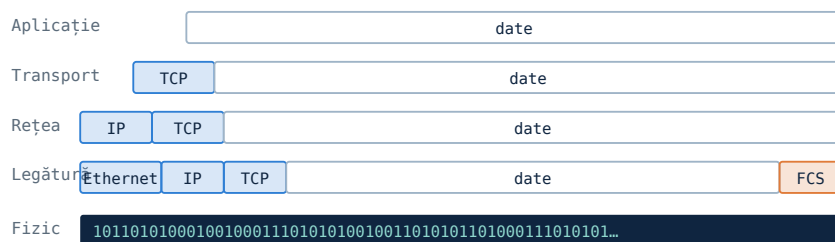
Este exact separarea care permite unui laptop să treacă de pe cablu pe Wi-Fi fără ca vreo aplicație deschisă să observe.

3.3 Încapsularea la nivelul 2

Ca datele să ajungă la destinatarul potrivit e nevoie de informație suplimentară, iar aceasta este adăugată de nivelul 2 și organizată în cadre. Majoritatea protocoalelor de nivel 2 - nu doar Ethernet - folosesc același set de câmpuri:

Tabelul 3.3 Încapsularea la nivelul 2.

Câmp	La ce folosește
Start cadru	o secvență de biți care anunță începutul; fără ea receptorul nu ar ști de unde să înceapă să numere
Adrese	adresele MAC ale sursei și destinației
Tip / lungime	ce protocol de nivel 3 este încapsulat, sau lungimea datelor
Date	mesajul propriu-zis, adică tot ce a construit nivelul 3
CRC / FCS	o sumă de control care permite detectarea erorilor de transmisie



Fiecare nivel privește ce vine de sus ca pe date opace și adaugă doar antetul lui.

Figura 3.2 Încapsularea, văzută de sus în jos. Observați că nivelul 2 este singurul care adaugă ceva și la sfârșitul mesajului: suma de control.

Alte protocoale de nivel 2, pe care le veți întâlni în WAN-uri: **PPP**, **Frame Relay**, **ATM**, iar istoric **Token Ring**. Toate rezolvă aceleași probleme, cu alte compromisuri.

3.4 Cadrul Ethernet, câmp cu câmp

Structura cadrului este aproape identică pentru toate implementările Ethernet, de la 10 Mbps la 100 Gbps. Aceasta e o realizare remarcabilă de inginerie: viteza a crescut de zece mii de ori, iar formatul a rămas.



Figura 3.3 Apăsați pe fiecare câmp al cadrului Ethernet.

Ce e un CRC, pe scurt. **Cyclic Redundancy Check** este un fel de „amprentă” a mesajului: emițătorul o calculează printr-o operație matematică peste toți biții și o atașează la sfârșit. Receptorul recalculează aceeași amprentă și compară.

Dacă valorile diferă, ceva s-a stricat pe drum. CRC-ul nu spune *ce* s-a stricat și nu poate repara - doar ridică mâna. E o alegere deliberată: detectarea e ieftină, corectarea ar fi scumpă, iar la rata de eroare a unui cablu modern e mai eficient să retransmiți întregul cadru.

MTU: cifra care revine tot semestrul. **MTU** (Maximum Transmission Unit) este dimensiunea maximă a încărcăturii utile - 1500 de octeți la Ethernet clasic. Un pachet IP mai mare trebuie fragmentat sau respins.

Veți întâlni MTU-ul din nou la tunelare (cursul 9), unde antetul suplimentar micșorează spațiul disponibil, și la OSPF (cursul 10), unde un MTU nepotrivit între două rutere blochează adiacența într-o stare pe care altfel nu ați putea-o explica.

3.5 Adresa MAC

Ethernet identifică fiecare interfață printr-o adresă de 48 de biți, scrisă în hexazecimal. Este scrisă în memoria ROM a plăcii de rețea, din fabrică.

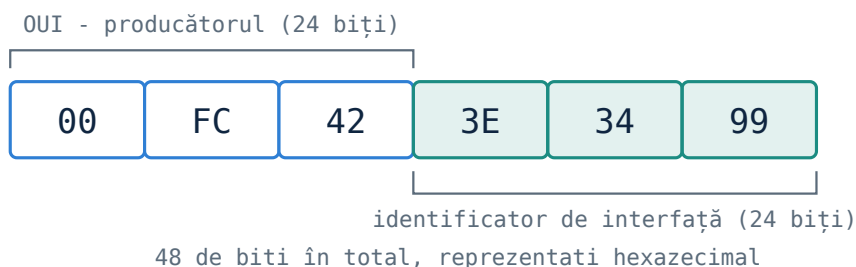


Figura 3.4 Structura unei adrese MAC. Primii 24 de biți formează OUI -ul, atribuit de IEEE producătorului; ultimii 24 sunt aleși de producător, de regulă în ordine de fabricație.

De ce sunt unice pe toată planeta. IEEE vinde blocuri de 24 de biți (OUI - Organizationally Unique Identifier) producătorilor de echipamente. Fiecare producător primește un prefix propriu și îl completează singur cu ce dorește, atât timp cât nu repetă.

Rezultatul: $2^{48} \approx 281$ de mii de miliarde de adrese posibile, împărțite administrativ, fără nicio autoritate care să verifice fiecare placă în parte. Sistemul funcționează pentru că fiecare producător are interesul să nu producă două plăci identice.

Ca observație practică: din primii trei octeți ai unei adrese MAC puteți afla producătorul. Există baze de date publice pentru asta, iar la depanare este surprinzător de util.

Adresa MAC are două proprietăți esențiale, ambele cu consecințe directe asupra a ceea ce urmează în curs:

1. Este **plată** - nu conține nicio ierarhie. Din adresa 00:FC:42:3E:34:99 nu se poate deduce absolut nimic despre locul unde se află acea placă.
2. Are **domeniu local** - este vizibilă și utilizabilă doar în interiorul domeniului de broadcast. Dincolo de primul ruter, nimeni nu o mai vede.

Consecința. Un ruter care ar trebui să dirijeze trafic după adrese MAC ar avea nevoie de o intrare separată pentru **fiecare interfață din lume** - fără nicio posibilitate de grupare, pentru că adresele nu sunt organizate geografic. Tabela nu ar încăpea nicăieri și nu ar putea fi întreținută de nimeni.

Exact aceste două proprietăți fac necesar nivelul 3, subiectul cursului 5.

Cele trei tipuri de adrese

Tabelul 3.4 Cele trei tipuri de adrese.

Tip	Cum se recunoaște	Exemplu	Cine o acceptă
Unicast	primul octet par (bitul cel mai puțin semnificativ = 0)	00:10:A7:22:FE:63	o singură interfață
Multicast	primul octet impar (bitul cel mai puțin semnificativ = 1)	01:00:5E:00:A1:11	stațiile abonate acelui grup
Broadcast	toți cei 48 de biți pe 1	FF:FF:FF:FF:FF:FF	toate stațiile din domeniu

Exercițiu rapid 3.1. Care dintre următoarele adrese sunt multicast?
02:00:00:00:00:01, 01:80:C2:00:00:00, 33:33:00:00:00:01,
FF:FF:FF:FF:FF:FF

Rezolvare. 02 = 00000010 - ultimul bit este 0, deci **unicast**. (Bitul următor, pe 1, înseamnă „administrată local”, adică setată de software, nu din fabrică.)

01 = 00000001 - ultimul bit este 1, deci **multicast**. Această adresă anume, 01:80:C2:00:00:00, este cea pe care circulă mesajele Spanning Tree din cursul 4.

33 = 00110011 - ultimul bit este 1, deci **multicast**. Prefixul 33:33 este rezervat pentru multicast IPv6.

FF:FF:FF:FF:FF:FF este **broadcast** - tehnic, un caz particular de multicast, în care grupul este „toată lumea”.

3.6 Timpii caracteristici ai unei rețele Ethernet

Trei mărimi de timp explică aproape toate valorile „ciudate” din standardul Ethernet.

Tabelul 3.5 Timpii caracteristici ai unei rețele Ethernet.

Mărime	Definiție	100		
		10 Mbps	Mbps	1 Gbps
Bit time	durata unui singur bit pe fir	100 ns	10 ns	1 ns
Slot time	timpul necesar semnalului să parcurgă cel mai lung segment și să se întoarcă	512 bit times = 64 de octeți		4096 bit times = 512 octeți
Interframe gap	pauza minimă obligatorie între două cadre succesive	96 bit times		

De unde vine dimensiunea minimă de 64 de octeți. Slot time-ul explică o valoare care altfel pare complet arbitrară.

Într-un mediu partajat, o stație trebuie să **încă transmită** în momentul în care o eventuală coliziune se întoarce la ea. Dacă ar termina mai devreme, ar pleca liniștită mai departe, convinsă că totul a mers bine - și nimeni nu ar retransmite cadrul pierdut.

Cadrul minim este exact atât cât să dureze un slot time. Cadrele mai scurte, rezultate din coliziuni, se numesc **runt frames** și sunt aruncate.

Interframe gap-ul are alt rol: dă stațiilor lente timp să proceseze cadrul curent și să se pregătească pentru următorul. La 1 Gbps înseamnă 96 de nanosecunde - o eternitate pentru electronică, o nimica toată pentru trafic.



Figura 3.5 Calculator de timpi Ethernet.

3.7 Mediul partajat și CSMA/CD

Ethernet a fost proiectat pentru un mediu multiacces: un singur cablu coaxial la care se conectau toate stațiile. Într-un astfel de mediu, două stații pot începe să transmită exact în același timp, iar semnalele lor se suprapun. Rezultatul se numește **coliziune**, iar cadrele implicate devin inutilizabile.

Analogie O ședință fără moderator, la telefon. Regula nescrisă este: ascuți, iar dacă nu vorbește nimeni, începi. Dacă doi încep simultan, se aude un talmeș-balmeș, amândoi se opresc, fiecare așteaptă un timp **diferit** - altfel ar reporni tot simultan - și încearcă din nou. Aceasta e, cuvânt cu cuvânt, descrierea CSMA/CD.



Figura 3.6 CSMA/CD, pas cu pas.

CSMA/CD este astăzi istorie - dar nu inutilă. Într-o rețea comutată full-duplex, fiecare port al switch-ului formează, împreună cu stația conectată, un domeniu de coliziune propriu, în care există exact doi participanți și **fire separate pentru fiecare sens**. Coliziunile nu mai pot apărea fizic. 10 Gigabit Ethernet nici măcar nu mai definește modul half-duplex.

Mecanismul rămâne totuși esențial de înțeles, din trei motive: explică dimensiunea minimă a cadrului, explică distanțele maxime ale segmentelor și - mai ales - este ideea pe care Wi-Fi a trebuit să o reinventeze, sub forma CSMA/CA din cursul 11, pentru că în aer coliziunile **nu pot fi detectate**.

3.8 Domenii de coliziune și de broadcast

DEFINIȚIA 3.1 (CELE DOUĂ DOMENII). Domeniul de coliziune este grupul de segmente în care două transmisii simultane se pot ciocni. Cu cât e mai mic, cu atât mai bine - ideal, doi participanți.

Domeniul de broadcast este grupul de stații care primesc un cadru de broadcast emis de oricare dintre ele. Cu cât e mai mare, cu atât mai mult traficul inutil procesează fiecare stație.

Tabelul 3.6 Domenii de coliziune și de broadcast.

Echipament	Domeniul de coliziune	Domeniul de broadcast
Hub / repetor	îl extinde	îl extinde
Switch / bridge	îl delimitează - câte unul per port	îl extinde
Ruter	îl delimitează	îl delimitează - câte unul per interfață

Cum se numără, practic. Domenii de coliziune: porniți de la fiecare cablu. Un cablu care duce la un port de switch sau de ruter își are propriul domeniu. Un hub, în schimb, unește într-un singur domeniu **toate** cablurile conectate la el.

Domenii de broadcast: același raționament, dar de data asta și switch-urile unesc - doar ruterele separă. Practic: numărați câte „insule” rămân dacă tăiați rețeaua la fiecare interfață de ruter.



Figura 3.7 Exercițiu: numărați domeniile din această topologie.

Exercițiu rezolvat - verificați-vă raționamentul 3.2. Un ruter are două interfețe. Pe prima este conectat un switch cu 4 stații; pe a doua, un hub cu 3 stații. Câte domenii de coliziune și câte domenii de broadcast există?

Rezolvare. Domenii de coliziune: 6. Switch-ul creează câte unul pentru fiecare dintre cele 4 legături spre stații, plus unul pentru legătura cu ruterul - deci 5. Hub-ul, împreună cu cele 3 stații și cu legătura spre ruter, formează un singur domeniu. Total $5 + 1 = 6$.

Domenii de broadcast: 2. Câte unul pentru fiecare interfață a ruterului. Toate cele 4 stații de pe switch se aud în broadcast între ele; toate cele 3 de pe hub, la fel; dar cele două grupuri nu se aud reciproc.

Capcana clasică: legătura switch–ruter este un domeniu de coliziune propriu, ușor de uitat. Iar legătura hub–ruter **nu** este un domeniu separat, ci face parte din domeniul hub-ului.

3.9 Cum funcționează un switch

Aici se află partea cea mai elegantă din tot cursul. Un switch nu este configurat de nimeni cu harta rețelei: și-o construiește singur, dintr-o singură observație inteligentă.

Observația care face totul posibil. Adresa MAC sursă a unui cadru spune pe ce port se află expeditorul.

Nu e nevoie de nicio configurare, de niciun protocol de descoperire, de nicio autoritate centrală. Informația este deja acolo, în fiecare cadru care trece - trebuie doar citită și reținută.

Asocierile se păstrează în **tabela CAM** (Content Addressable Memory), numită și tabela de adrese MAC. Pentru fiecare cadru primit, switch-ul execută întotdeauna aceiași doi pași, în această ordine:

1. **Învățare.** Citește MAC-ul **sursă**. Dacă nu există în tabelă, adaugă perechea (MAC sursă, port de intrare). Dacă există deja, îi resetează contorul de vârstă.
2. **Comutare.** Citește MAC-ul **destinație** și îl caută în tabelă. Trei rezultate posibile:
 - găsit, pe **alt** port → **unicast**: trimite cadrul doar pe acel port;
 - găsit, pe chiar **portul de intrare** → **drop**: destinația e de aceeași parte, retrimiteria ar fi inutilă;
 - **negăsit**, sau adresă de broadcast/multicast → **flood**: trimite pe toate porturile în afară de cel de intrare.



Figura 3.8 Simulator: tabela CAM a unui switch cu patru stații.

1. Goliți tabela. Trimiteți un cadru de la **A la C**. Observați: switch-ul învață A, dar nu știe unde e C, deci **inundă**.
2. Trimiteți acum de la **C la A**. Switch-ul învață și C, iar A îl știe deja, deci **unicast**.
3. Trimiteți din nou de la **A la C**. De data asta este direct unicast.

Experimentul de făcut, în ordinea asta. Ce ați văzut: rețeaua **se învață singură** după primul schimb, iar traficul inutil dispare de la sine. Niciun administrator nu a intervenit.

Îmbătrânirea intrărilor

Fiecare intrare din tabelă are asociată o **vârstă**, resetată la fiecare cadru primit de la acea adresă. Când vârsta depășește un prag - implicit 300 de secunde pe echipamentele Cisco - intrarea este ștersă.

Exemplul 3.3. De ce trebuie șterse intrările? Ce s-ar întâmpla dacă tabela ar fi permanentă?

Rezolvare. Dacă mutați un calculator de pe portul 3 pe portul 7, switch-ul ar continua să trimită traficul pentru el pe portul 3 - la nesfârșit, deși acolo nu mai e nimeni.

În practică, problema se rezolvă de la sine mult mai repede: primul cadru emis de calculator de pe noul port corectează imediat intrarea. Îmbătrânirea acoperă cazul în care stația a plecat și **nu mai vorbește deloc** - altfel tabela s-ar umple, în timp, cu fantome.

Există și un motiv de securitate: o tabelă plină face switch-ul să inunde tot traficul, ceea ce este exact scopul atacului *CAM overflow* din cursul 12.



Figura 3.9 Puneți în ordine pașii pe care îi face un switch.

3.10 Metode de comutare

Cât așteaptă un switch înainte să ia decizia? Există trei răspunsuri, cu trei compromisuri diferite.

Tabelul 3.7 Metode de comutare.

Metodă	Ce așteaptă înainte de a comuta	Latență	Cadre eronate
Store-and-forward	cadrul întreg; verifică lungimea și FCS-ul	mare, crește cu dimensiunea cadrului	oprite
Fragment-free	primii 64 de octeți	medie, fixă	runt frames oprite, restul trec
Fast-forward (cut-through)	doar adresa destinație - primii 6 octeți după preambul	minimă, fixă	toate trec

Compromisul e limpede: cu cât switch-ul citește mai mult înainte de a decide, cu atât filtrează mai bine, dar cu atât întârzie mai mult.

Ce se folosește în practică. Echipamentele moderne rulează, aproape întotdeauna, **store-and-forward**. Costul în latență a devenit neglijabil la vitezele actuale, iar verificarea FCS-ului este obligatorie: un switch cut-through **propagă** cadrele corupte mai departe în rețea, unde vor fi aruncate oricum - dar după ce au consumat bandă.

Cut-through supraviețuiește în centre de date și în tranzacții financiare, unde micro-secondele contează cu adevărat.

3.11 Familia Ethernet, pe scurt

Ethernet a apărut în 1973 la Xerox PARC, realizat de Bob Metcalfe și David Boggs, cu o lățime de bandă de 2,94 Mbps. Numele vine de la „eter” - substanța ipotetică prin care se credea cândva că se propagă lumina. În 1983, IEEE a transformat standardul industrial DIX (DEC–Intel–Xerox) în standardul **802.3**, care rezistă și azi.

Tabelul 3.8 Familia Ethernet, pe scurt.

Generație	An	Standard	Viteză	Codare	CSMA/CD
Fast Ethernet	1995	100BASE-TX / FX	100 Mbps	4B/5B + MLT-3 sau NRZ-I	da (half-duplex)
Gigabit Ethernet	1998–1999	1000BASE-T / SX / LX	1 Gbps	PAM-5 sau 8B/10B	opțional
10 Gigabit	2002	10GBASE-T / SR / LR	10 Gbps	diverse	nu - doar full-duplex
40 / 100 Gigabit	2010	802.3ba	40 / 100 Gbps	multiplă, pe mai multe benzi	nu

Ce s-a schimbat și ce a rămas. Viteza a crescut de zece mii de ori. Mediul s-a schimbat de trei ori. CSMA/CD a dispărut complet.

Formatul cadrului a rămas același. Un cadru capturat astăzi pe o legătură de 100 Gbps are exact aceleași câmpuri ca unul din 1983. Aceasta e cea mai bună lecție de proiectare pe care o oferă acest curs: o interfață bine gândită supraviețuiește implementărilor ei.

3.12 Greșeli frecvente

Greșeli frecvente.

„Switch-ul învață din adresa MAC destinație”

Nu. Învață exclusiv din adresa sursă - singura care spune sigur de unde a venit cadrul. Destinația se folosește doar pentru a decide unde îl trimite.

Rețineți ordinea: întâi învață din sursă, apoi caută destinația.

„Un cadru de broadcast este trimis și înapoi pe portul de intrare”

Niciodată. Flood înseamnă „pe toate porturile în afară de cel pe care a venit”.

Altfel s-ar crea o buclă instantanee, chiar și fără cabluri redundante.

„Dimensiunea minimă de 64 de octeți e o convenție arbitrară”

Vine direct din slot time: stația trebuie să încă transmită când coliziunea se întoarce la ea.

64 octeți = 512 bit times = slot time la 10 și 100 Mbps.

„Un switch cu 24 de porturi are 24 de domenii de broadcast”

Are 24 de domenii de coliziune și unul singur de broadcast. Switch-ul separă coliziunea, dar extinde broadcast-ul.

Separarea broadcast-ului cere un ruter - sau VLAN-uri, din cursul 4.

„FCS-ul corectează erorile”

Doar le detectează. Un cadru cu FCS greșit se aruncă, iar retransmiterea rămâne treaba nivelurilor superioare.

Detectare la nivelul 2, recuperare la nivelul 4.

3.13 Rezumat și glosar

1. Nivelul 2 adaugă ce lipsea nivelului fizic: **identitate** (adrese MAC), **delimitare** (cadre) și **detectarea erorilor** (FCS).
2. Adresa MAC este **plată** și **locală** - două limitări care fac necesar nivelul 3.

3. CSMA/CD a rezolvat problema mediului partajat; comutarea full-duplex a făcut-o inutilă.
4. Un switch își construiește singur harta rețelei, din adresele MAC sursă ale cadrelor care trec prin el.
5. Switch-ul **separă** domeniile de coliziune și **extinde** pe cele de broadcast - de aici pornește tot cursul următor.

Exerciții

1. Un switch primește un cadru a cărui adresa MAC destinație nu se afla în tabela CAM. Ce face?
 - a. Il arunca
 - b. Il trimite pe toate porturile în afara de cel de intrare
 - c. Il trimite înapoi pe portul de intrare
 - d. Cere tabela CAM de la switch-ul vecin
2. De ce cadrul Ethernet minim are 64 de octeți?
 - a. Ca să încapă antetul IP
 - b. Pentru ca transmisia trebuie să dureze cel puțin un slot time, altfel stația nu ar detecta coliziunea
 - c. Pentru că FCS ocupă 4 octeți
 - d. Este o valoare aleasă arbitrar de IEEE
3. Care adresa MAC este multicast?
 - a. FF:FF:FF:FF:FF:FF
 - b. 00:10:A7:22:FE:63
 - c. 01:00:5E:00:A1:11
 - d. 02:00:00:00:00:01
4. Câte domenii de broadcast are un switch cu 24 de porturi, fără VLAN-uri?
 - a. 24
 - b. 12
 - c. 1
 - d. 0

5. Ce conține câmpul Tip/Lungime când are valoarea 0x0800?
 - a. Lungimea cadrului în octeți
 - b. Protocolul de nivel 3 încapsulat, adică IPv4
 - c. Numărul VLAN-ului
 - d. Suma de control

6. Din ce câmpuri învață un switch, ca să își construiască tabela CAM?
 - a. Din adresa MAC destinație
 - b. Din adresa MAC sursă
 - c. Din adresa IP sursă
 - d. Din câmpul EtherType

7. De ce nu mai apar coliziuni într-o rețea comutată full-duplex?
 - a. Pentru că switch-urile sunt mai rapide
 - b. Pentru că fiecare port are doi participanți și fire separate pentru fiecare sens
 - c. Pentru că CSMA/CD a fost îmbunătățit
 - d. Pentru că se folosește fibra optică

8. Un switch cut-through primește un cadru corupt. Ce se întâmplă?
 - a. Îl aruncă
 - b. Îl trimite mai departe, pentru că a decis deja pe baza adresei destinație
 - c. Îl repară folosind FCS
 - d. Îl reține până la verificare

Note bibliografice

Un switch rezolvă coliziunile, dar lasă un singur domeniu de broadcast oricât de mare ar fi rețeaua - și nu tolerează bucle. Cursul următor atacă ambele probleme: **VLAN-uri** pentru segmentare și **Spanning Tree** pentru redundanță fără bucle.

La **laboratorul 2** veți goli o tabelă CAM reală, o veți umple urmărind-o cadru cu cadru și veți compara, pe aceeași topologie, un switch cu un hub.

- IEEE 802.3 - formatul cadrului și mecanismul CSMA/CD
- IEEE 802.1D - operarea unui bridge transparent
- Registrul public IEEE OUI, pentru identificarea producătorului după adresa MAC
- Andrew S. Tanenbaum, *Computer Networks*, capitolul 4 - subnivelul de acces la mediu

- Charles Spurgeon, *Ethernet: The Definitive Guide*

Extras din volumul „Rețele Locale - Note de curs și lucrări de laborator”. Varianta online a acestui capitol conține figurile în formă interactivă.