

## Dispozitive de interconectare pentru LAN-uri

### 1. Obiective:

- Identificarea avantajelor si dezavantajelor a repetoarelor (repeaters), hub-urilor, puntilor (bridges), comutatoarelor (switches) si ruterelor (routers).

### 2. Repetears (repetoare)

Repetorul este dispozitivul de interconectare ce functioneaza la nivel fizic. Deoarece la nivelul fizic nu exista date ci doar biti, repetorul nu este preocupat de identificarea destinatiei sau de verificarea unui cod de corectie, ci doar de semnalul electric pe care-l primeste si de regenerarea acestuia. Principala sa functie este aceea de a extinde suprafata acoperita de o retea locala cu un cost si o latenta foarte scazute.

Sirul de biti generat de o placa de retea este clar, respectand strict nivelurile de tensiune standardizate. Cu cat sirul de biti calatoreste mai mult prin cablu, semnalul electric se deterioreaza si devine din ce in ce mai slab. Pentru a opri deteriorarea semnalului peste o limita ce l?ar face de nerecunoscut pentru destinatie, repetorul ia sirul de biti, il aduce la treptele de semnalizare standardizate si il amplifica. Deprecierea semnalului nu apare doar cand acesta calatoreste prin mediul de cupru, dar si cand atasam prea multe dispozitive la mediul de transmisie, deoarece fiecare nou dispozitiv atasat la mediu va provoca o mica degradare a semnalului.

Exista repetoare pentru toate mediile de transmisie pe cupru - de la cablul coaxial de diferite impedante pana la cel torsadat. Cele mai des intalnite retele locale sunt totusi fara indoiala si in Romania retelele Ethernet.

### 3. Hubs (hub-uri)

In retelele Ethernet intalnim deseori repetoare multiport numite huburi. Huburile vor transmite datele primite pe unul dintre porturi pe toate celelalte porturi. Pentru mediul torsadat acestea indeplinesc o functie suplimentara si anume asigura conectarea tuturor nodurilor la un mediu de transmisie distribuit.

Initial au existat doua tipuri de huburi: pasive si active. Huburile pasive ofera posibilitatea interconectarii la acelasi mediu de transmisie a mai multor dispozitive, fara a regenera semnalul la trecerea prin ele. Huburile active vor oferi in plus fata de primele regenerarea semnalului. Datorita scaderii extrem de rapide a preturilor si avantajelor ce le ofera aceasta regenerare de semnal huburile pasive au disparut de pe piata inca de la sfarsitul anilor '80, din aceasta cauza in continuare prin huburi vom intelege huburi active.

Una din componentele esentiale ale protocolului Ethernet este detectia coliziunilor. Ne intereseaza care este efectul unui repetor asupra coliziunilor.

*Ce sunt domeniile de coliziune?*

**Un domeniu de coliziune** reprezinta acea sectiune dintr-o retea in care se va propaga o coliziune.

*Ce sunt domeniile de difuzare?*

**Un domeniu de difuzare** (domeniu de broadcast) reprezinta acea sectiune dintr-o retea in care se va propaga un pachet de difuzare (broadcast).

Pentru un repetor nu exista notiunea de coliziune, dupa cum nu exista notiunea de pachet de date. Deci repetoarele extind atat domeniile de coliziune cat si pe cele de difuzare. Repetoarele impart reteaua in microsegmente. Aceasta denumire nu este general acceptata, cateodata fiind folosit termenul de segment pentru cele doua sau mai multe seturi de calculatoare pe care le conecteaza un repetor.

Exista o regula foarte importanta pentru proiectarea retelelor Ethernet: regula 5-4-3.

**Regula 5-4-3:** Comunicatia dintre oricare doua calculatoare sau dispozitive dintr-o retea nu trebuie sa treaca prin mai mult de:

5 microsegmente, 4 repetoare consecutive, 3 microsegmente populate.

De unde vine aceasta regula?

Exista o fereastra de timp pentru transmiterea unui bit. Pentru Ethernet, ce ofera o viteza de 10 Mbps, durata transmiterii unui singur bit este de 100 de nanosecunde. Dimensiunea minima a cadrului Ethernet este de 64 de octeti. Rezulta ca timpul necesar transmiterii cadrului de dimensiune minima este de 51,2 microsecunde.

De ce ne intereseaza acest timp? Pentru ca aparitia unei coliziuni trebuie detectata inainte de expirarea acestui interval de timp. Altminteri, aparitia unei coliziuni va fi interpretata ca o coliziune la cel de-al doilea cadru si nu pentru primul. Latenta introdusa de mediul de transmisie va fi data de viteza de propagare a semnalului electric, aceasta fiind aproximativ doua treimi din viteza luminii. Rezulta ca propagarea pe un segment de 100 de metri va dura aproximativ 0,5 microsecunde. Comparativ cu latenta introdusa de un repetor Ethernet de aproximativ 5,6 microsecunde, latenta introdusa de mediul de conectare este cu un ordin de marime mai mica, deci neglijabila.

Cel mai defavorabil caz se obtine cand sursa si destinatia se afla la distanta maxima, iar coliziunea apare langa destinatie, astfel incat coliziunea ce trebuie detectata si de sursa trebuie sa parcurga de doua ori distanta maxima. Daca vom considera acum ca intre sursa si destinatie se afla cinci repetoare, vom determina ca in cel mai defavorabil caz detectia coliziunii va fi posibila doar dupa cel putin 56 de microsecunde, asta insemand ca un alt doilea pachet deja a fost trimis.

*Ce se intampla daca nu respectam regula 5-4-3?*

In primul rand se va cere retransmisia unui cadru corect, in vreme ce cel pierdut in urma coliziunii va fi considerat ca ajuns la destinatie intact. Astfel responsabilitatea integritatii datelor va fi pasata nivelului superior si anume nivelului retea. Din pacate, acest nivel nu are posibilitatea manipularii de cadre, si va determina ca intregul pachet din care face parte si cadrul eronat este incorect, cerand retransmiterea pachetului. Aceasta practica, desi va asigura integritatea datelor, introduce o latenta semnificativa.

Desi la o incalcare nu prea violenta a acestei reguli (cum ar fi folosirea a cinci repetoare in loc de patru) deprecierea performantelor este mica, daca vrem sa garantam ca reseaua Ethernet instalata va oferi o latime de banda de 10 Mbps, atunci cinci repetoare inseamna deja un repetor in plus. Doar pentru a sesiza flexibilitatea enorma a regulii 5-4-3 sa consideram cazul transmiterii cadrelor Ethernet de lungime maxima, adica 1500 de octeti. Pentru acest caz este necesara o fereastra de timp de 1,2 milisecunde, astfel incat performantele pentru o retea Ethernet ce ar folosi doar cadre de lungime maxima nu ar fi afectate nici in cazul folosirii a 100 de repetoare. Cu toate acestea in realitate se folosesc si cadre de lungime mica si, desi cadrele de lungime minima nu sunt folosite prea des, corectarea erorilor de catre nivelul retea poate constitui un cost prea mare in termeni de latenta. Desi nu este principala sa functie, repetorul ofera, precum orice dispozitiv de interconectare, posibilitatea legarii de grupuri de calculatoare ce difera prin mediul de transmisie folosit, adica prin nivelul fizic. Altfel spus, numeroase dintre huburile actuale ofera pe langa porturile RJ-45 si un port BNC, sau mai rar chiar un port AUI.

#### **4. Bridges (punti)**

Puntea sau bridge-ul este primul dispozitiv de interconectare ce poate lua decizii logice. Pentru el semnalele electrice se transforma in octeti si in date. Puntea este dispozitivul de interconectare ce functioneaza la nivelul legatura de date. Puntile sunt folosite si la interconectarea a grupuri de calculatoare ce difera prin protocolul folosit la nivelul legatura de date sau a mediului de transmisie. Astfel, exista puncti ce conecteaza retele Ethernet cu retele Token Ring, sau retele Token Ring cu retele Token Bus.

Care sunt mecanismele ce ii permit puntii sa ia decizii logice? Cele doua mecanisme ce fac din punte un dispozitiv de interconectare "inteligent" sunt: incapsularea datelor la nivel legatura de date si folosirea unei scheme de adresare pentru livrarea acestora.

Gruparea datelor nu se face la nivel de bit, ci la nivel de cadru, un cadru putand contine pana la 1500 de octeti in cazul cadrului Ethernet, sau chiar 8000 de octeti.

Puntea interconecteaza doua sau mai multe segmente de retea. In plus fata de un simplu calculator, care la nivelul legatura de date se preocupa doar de incapsularea datelor in cadre, o punte trebuie sa ia decizia spre ce segment sa trimita cadrul primit.

*Va regenera puntea semnalul electric?*

Da! In cazul in care pe una dintre interfete primeste un sir de biti ale caror valori nu sunt 0,85V sau -0,85V (in cazul Ethernetului), va incerca sa-si dea seama care au fost valorile initiale a acestor biti pentru a putea intelege cadrul primit. Odata obtinut un cadru valid, adica dupa corectarea bitilor ce nu mai aveau niveluri de tensiune corecta, puntea va desface antetul cadrului si va analiza informatiile legate de adresa destinatie. Dupa determinarea interfetei pe care trebuie trimis cadrul, placa de retea il va transforma in biti, trecandu-l

la nivelul fizic. Placa de retea poate genera doar cateva niveluri de tensiune, astfel incat nici nu ar fi posibila trimiterea sirului de biti depreciat.

• **Principala functie a unei punti este filtrarea traficului pe baza adresei fizice.**

Sa ne aplecam un pic asupra procesului prin care putea ia decizii de comutare a unui cadru. Pentru a putea lua astfel de decizii puntile folosesc o tabela, numita tabela de comutare (bridging / switching table) in care fiecarei adrese fizice ii este asociata una dintre interfetele sale. In figura de mai jos avem o astfel de tabela.

Interfata	Adresa MAC
E0	00.48.C2.01.78.12
E0	00.00.2E.00.59.91
E1	00.00.54.91.01.4A

Tabela de comutare cu 3 intrari

De exemplu, prima intrare are urmatoarea semnificatie: destinatia 00.48.C2.01.78.12 se afla pe segmentul conectat pe interfata E0 a puntii (E0 este prescurtarea de la Ethernet 0, prima interfata Ethernet).

• **Puntea izoleaza comunicatia intre statii aflate in acelasi segment la nivelul segmentului.**

Consecintele acestui fapt sunt extrem de importante. In primul rand, puntea va margini domeniile de coliziune. Totodata ea va oferi mai multa banda disponibila, deoarece comunicatia in interiorul aceluiasi segment nu va consuma din banda disponibila a intregii retele.

O alta consecinta o reprezinta minimizarea riscurilor de securitate legate de atacurile din interiorul retelei locale. Unul dintre cele mai populare atacuri este ascultarea liniei (sniffing attack), prin care pe una dintre statiile conectate la mediul distribuit se forteaza nivelul legatura de date sa trimita spre nivelurile superioare toate cadrele – inclusiv cele ce nu sunt destinate acestei statii. Cu ajutorul unor aplicatii dedicate datele sunt reasamblate si astfel va fi monitorizat tot traficul ce traverseaza segmentul de retea. Prin folosirea puntilor putem izola de restul retelei statiile ce prezinta un risc de securitate.

*Care este rolul puntii in comunicatia dintre segmente?*

Pentru acest caz vom considera aceeasi retea din figura precedenta si un trafic intre statia A1 si B1. Statia A1 va asculta mediul si cand acesta va fi liber va transmite un cadru. Cadrul se va propaga spre statiile A2, A3 si spre puntea 1. Statiile vor ignora cadrul, acesta nefiind adresat lor, in schimb puntea va cauta adresa destinatie in tabela sa de comutare. Va determina interfata pe care trebuie trimis cadrul si apoi va decide ca aceasta interfata este diferita de cea pe care cadrul a fost primit. Astfel incat puntea va transmite cadrul primit din segmentul A, doar pe segmentul B. Cadrul va fi receptionat atat de B1, cat si de B2, dar doar B1 il va prelucra.

*Care este suprafata maxima pe care o poate ocupa o retea ce foloseste doar punti?*

Suprafata maxima pe care se poate intinde o retea folosind doar punti nu face obiectul nici unei reglementari explicite. Cu toate acestea, in plus fata de avantajele prezentate mai sus, puntea aduce si o serie de dezavantaje, facand astfel ca procesul de proiectare a unei retele locale sa fie un lucru foarte delicat. In comparatie cu repetoarul, puntea inlatura limitarile impuse de regula 5<sup>4</sup>?3, izoleaza traficul din interiorul unui segment la nivelul segmentului si ofera posibilitatea interconectarii unor segmente de retea ce folosesc protocoale de nivel legatura de date diferite. Puntile vor extinde domeniile de difuzare, desi le limiteaza pe cele de coliziune. In acelasi timp, costul unei punti este cu cel putin un ordin de marime mai mare decat cel al unui repetoar.

Inlocuirea repetoarelor cu punti duce o crestere a latentei in retea cu 10-30 %, datorita timpului necesar prelucrarii informatiei de nivel legatura de date. In cazul unui trafic intens intre statii aflate in segmente diferite puntea poate duce la o gatuire a traficului. Pentru a putea functiona eficient o punte trebuie sa aiba la dispozitie o tabela de comutare ce contine cate o intrare pentru fiecare dintre statiile din acea retea locala. Cautarea in aceasta tabela este o cautare secventiala, deci extrem de ineficienta pentru o dimensiune prea mare a tablei. Astfel dimensionarea optima a retelei ce foloseste doar punti, desi nu se supune nici unei restrictii de lungime, va fi puternic influentata de numarul de statii, precum si de tipul traficului, mai exact de latentă pe care ne-o putem asuma pentru diferite tipuri de trafic.

## 5. Switches (comutatoare)

O punte multiport se numeste comutator sau switch. Fata de punti, comutatoarele in general implementeaza metode de comutare mai rapide. Uneori comutatorul este privit ca un dispozitiv de interconectare ce actioneaza atat la nivel fizic, cat si la nivel legatura de date. Aceasta nu se datoreaza unei latente mai mici sau unui cost mai scazut comparativ cu o punte, ci datorita faptului ca in retelele Ethernet ce folosesc mediul torsadat comutatorul preia functia principala a hubului, si anume aceea de a asigura conectarea tuturor nodurilor la un mediu de transmisie.

Exista doua paradigme in retelele de calculatoare: arhitecturi bazate pe magistrala si indirect pe difuzare si arhitecturi bazate pe comutare. Optarea pentru una dintre cele doua paradigme se traduce in decizia de a folosi un comutator sau un hub.

O retea bazata pe huburi are un cost mai scazut si o latentă mai mica. Principalul avantaj al inlocuirii huburilor cu comutatoare nu il reprezinta inlaturarea restrictiilor impuse de regula 5-4-3, ci reducerea numarului de utilizatori ce partajeaza aceeasi latime de banda.

Comutatoarele vor oferi protectie impotriva atacurilor prin ascultare a liniei. "Razboiul" hub versus comutator opune costul si latentă mai scazute, pe de o parte, cu cerintele crescande de latime de banda disponibila si de securitate, pe de alta parte.

*Am lamurit care sunt diferentele dintre comutatoare si repetoare, dar care sunt diferentele intre punti si comutatoare?*

Numarul de interfete sau porturi este fara indoiala cea mai usor de observat diferenta. Diferenta cea mai importanta consta totusi in modul de comutare a pachetelor: comutatoarele folosind comutarea directa. In acelasi timp cerintele de latentă pentru o punte cu doua interfete sunt mult mai relaxate decat pentru un comutator. Din aceasta cauza puntile, in general, comuta pachete folosind componente software, in vreme ce comutatoarele vor lua toate deciziile la nivel hardware.

Dupa cum am vazut, puntea reface semnalul la nivel de bit, pentru a obtine un cadru, apoi despacheteaza cadrul, foloseste informatiile din campul adresa destinatie pentru a filtra sau nu cadrul, iar adresa sursa va fi folosita pentru construirea tabelii de comutare. Dar una dintre functiile nivelului legatura de date este acela de a oferi mecanisme de corectie a datelor la nivel de cadru.

*Ofera comutatoarele astfel de mecanisme?*

In figura de mai sus este prezentata structura cadrului Ethernet. Este important de remarcat ca informatiile de detectie sau corectie a erorilor se afla in finalul cadrului. Asta inseamna ca, pentru a putea detecta erorile dintr-un cadru, trebuie mai intai asteptata receptionarea integrala a acestuia. Problema care apare in acest caz este ca latentă pentru un cadru de dimensiune maxima pentru Ethernet va fi de 1,2 milisecunde. Aparent, mutarea campului de control in antetul cadrului ar rezolva problema. Practic, acest lucru este imposibil, deoarece suma de control nu este un sir continuu de biti aflati dupa zona bitilor de date, ci bitii ce compun cei 2 sau 4 octeti ai sumei de control sunt intercalati cu bitii de date.

*Care sunt tipurile de comutare folosite de un comutator?*

Exista doua metode de comutare a pachetelor: comutare directa (cut through) si comutare dupa stocare (store and forward).

Metoda de comutare dupa stocare se bazeaza pe receptionarea intregului cadru inainte de a incepe retransmisia acestuia. Latenta acestei metode creste odata cu dimensiunea campului de date. Cu toate acestea, performantele metodei de comutare dupa stocare pot fi superioare celor oferite de comutarea directa, mai ales in cazul linilor expuse unor interferente puternice. Mecanismele de detectie a erorilor pe care le ofera aceasta metoda de comutare permite asigurarea unei conexiuni sigure la nivelul legatura de date.

Aparent, metoda de comutare dupa stocare ridica si problema asigurarii memoriei pentru stocarea cadrelor. Sa luam exemplul unui comutator cu 24 de porturi. Acesta va trebui sa poata gestiona 12 comunicatii simultane, care in cel mai defavorabil caz posibil vor transfera cadre de lungime maxima. Am ajuns astfel la o dimensionare a memoriei RAM pentru stocarea cadrelor de aproape 18 ko. Putem trage concluzia ca dimensionarea memoriei RAM folosite pentru stocarea cadrelor a incetat sa mai fie o problema de actualitate.

Comutarea directa presupune ca puntea sa inceapa transmiterea cadrului pe portul destinatie imediat ce adresa destinatie a fost trecuta prin tabela de comutare si interfata de plecare a fost determinata. Cel mai adesea se intampla ca transmisia cadrului sa inceapa inainte de receptionarea integrala a cadrului. Astfel

comutatorul va primi pe una dintre interfețe octeti ce compun cadrul, transmitând în același timp pe portul destinație octeti din același cadru primiti mai devreme.

Pentru comutarea directă nu este necesară nici măcar recepționarea integrală a antetului cadrului, adresa destinație fiind suficientă. Această metodă se numește comutare directă rapidă (fast forward) și oferă o latență de aproximativ 21 de microsecunde. Datorită faptului că retransmisia cadrului începe imediat după citirea adresei destinație, cadrele eronate vor fi transmise cu erori. Deși aceste cadre sunt respinse la nivelul legăturii de date al destinației (de către placa de rețea), traficul generat de retransmisia lor poate, în cazul unui mediu de transmisie cu multe erori, să ducă la o depreciere severă a performanțelor rețelei.

Al doilea tip de comutare directă este comutarea fără fragmente (fragment free). Pentru această metodă de comutare vor fi filtrate fragmentele de cadre rezultate în urma unei coliziuni. Într-o rețea ce respectă specificațiile standardului Ethernet dimensiunea fragmentelor de coliziuni nu poate depăși 64 de octeți. Pentru comutarea fără fragmente, comutatorul va determina ca șirul de octeți recepționați nu fac parte dintr-un fragment de coliziune și abia apoi va începe retransmisia pe portul destinație. Latența în acest caz este de minim 51,2 microsecunde, ceea ce reprezintă timpul necesar recepționării a 64 de octeți.

*Care sunt tipurile de comutare folosite de o punte?*

În general punțile implementează doar comutarea după stocare, aceasta fiind una din principalele deosebiri față de comutatoare care nu vor mai fi preocupate de detectarea erorilor, ci de filtrarea pachetelor și de asigurarea unei latențe cât mai scăzute.

*Care este rolul comutatoarelor în implementarea conexiunilor Ethernet halfduplex?*

Comunicația semi-duplex (half-duplex) permite doar unui singur nod să transmită date. În Ethernet aceasta este controlată cu ajutorul coliziunilor. Dacă două sau mai multe stații încearcă să comunice simultan rezultatul va fi o coliziune.

Pe interfețele unui comutator putem conecta o stație sau un segment întreg. Cu toate acestea, rețelele comutate sunt răspunsive pentru cerințele crescânde de securitate și de lățime de bandă pentru fiecare nod. Rețelele comutate vor folosi câte un port pentru fiecare stație, reducând dimensiunea domeniilor de coliziune la doar două noduri (unul fiind placa de rețea din respectiva stație, iar cel de-al doilea portul din comutator ce o conectează pe aceasta).

Altfel spus, comutatoarele oferă suportul pentru implementarea rețelelor comutate, rețele în care domeniile de coliziune nu depășesc două noduri.

*Care este rolul comutatoarelor în implementarea conexiunilor Ethernet fullduplex?*

Ethernetul full-duplex permite trimiterea și recepționarea simultană.

*Ce tipuri de arhitecturi pot fi implementate cu ajutorul comutatoarelor?*

### **STP**

O buclă de nivel legătură de date apare într-o rețea când între două dispozitive ale acesteia există două sau mai multe legături active, fiecare conexiune folosind doar dispozitive de interconectare ce pot analiza cel mult informații de nivel legătură de date.

Apariția buclelor de nivel legătură de date este corelată cu faptul că punțile și comutatoarele nu filtrează pachetele de difuzare și duc la o depreciere semnificativă a performanțelor rețelei prin determinarea unor avalanșe de difuzări (broadcast storm).

*Cum putem preveni apariția avalanșelor de difuzări?*

Soluția trivială ar fi să instruim punțile și comutatoarele să nu retransmită cadrele de difuzare. Din păcate acest lucru nu este posibil, deoarece o serie de protocoale folosesc cadre de difuzare pentru a funcționa corect, unul dintre acestea fiind chiar ARP – Address Resolution Protocol. Altfel spus, filtrarea cadrelor de difuzare de către punți ar presupune rescrierea protocoalelor fundamentale ce asigură suportul de comunicație.

Soluția validă presupune identificarea buclelor și întreruperea lor. Protocolul ce realizează aceasta se numește STP - Spanning Tree Protocol, și presupune construirea unui arbore de acoperire pe graful determinat de dispozitivele de interconectare și de conexiunile dintre acestea.

### *Cum functioneaza STP?*

Functionarea acestui protocol se bazeaza pe crearea topologiei rezei folosind niste cadre speciale numite cadre BPDU (Bridge Protocol Data Unit). Aceste cadre speciale sunt folosite intens la initializarea comutatoarelor; ulterior, la fiecare doua secunde vor fi schimbate cadre BPDU, pentru a verifica daca nu au aparut modificari. Totodata sunt definite cinci stari in care se poate afla o interfata a comutatorului: starea blocat, de ascultare, de invatare, de comutare de cadre si nefunctional (blocking, listening, learning, forwarding, disabled). In starea blocat nu se accepta decat cadre BPDU, in cea de ascultare se primesc si cadre, dar acestea nu sunt retransmise. In starea de invatare, in plus fata de starea de ascultare, este inspectata adresa sursa a cadrelor primite, permitand astfel construirea tabelii de comutare. In starea de comutare cadrele primite sunt retransmise, iar tabela de comutare este actualizata. In starea nefunctional nu se vor accepta nici cadre BPDU.

Pentru construirea arborelui de acoperire sunt necesare aproximativ 30 de secunde, timp in care toate porturile comutatoarelor sunt in starea blocat. Exista trei pasi ce trebuie urmati pentru construirea arborelui de acoperire: mai intai trebuie aleasa radacina arborelui (root bridge), apoi trebuie alese porturile radacina, pentru ca in final sa fie determinate porturile active.

Prioritatea puntii este o valoare numerica pastrata in memoria nevolatila a fiecarei punti. Pe baza compararii prioritatilor tuturor puntilor din retea se va determina puntea cu prioritatea cea mai scazuta, aceasta devenind radacina arborelui de acoperire.

Prioritatea puntii are o valoare implicita atribuita de producator, valoare ce poate fi modificata ulterior. In cazul folosirii mai multor echipamente produse de aceeaasi firma, se intampla adesea sa existe mai multe punti ce vor avea aceeasi prioritate.

### *Cum vom putea decide care dintre doua sau mai multe punti cu aceeasi prioritate sa devina radacina arborelui?*

Pe baza adresei fizice. Puntea cu cea mai mica adresa fizica va deveni radacina arborelui de acoperire.

Pasul al doilea presupune identificarea cailor redundante dintre fiecare punte si puntea radacina, apoi selectarea unei sigure cai intre respectiva punte si radacina si, in final, dezactivarea celorlalte.

Pentru evaluarea unei cai vom determina costul caii, care va fi definit ca suma a costurilor porturilor prin care trece calea. Costul unui port este definit pe latimea de banda pe care o ofera portul, sau uneori chiar pe mediul de transmisie folosit pentru conectarea la port. De exemplu, pentru comutatoarele CISCO costul portului este determinat prin impartirea lui 1000 la latimea de banda pe care o ofera portul, astfel incat un port Ethernet va avea costul 100.

Pentru alegerea porturilor radacina vor avea prioritate porturile conectate direct la radacina arborelui de acoperire. In cazul in care nu exista nici un port cu o conexiune directa spre puntea radacina, sau cand avem mai mult de un singur port cu conexiune directa spre radacina, va fi ales portul ce are cel mai scazut cost al caii spre radacina.

## **6. Routers (rutere)**

Ruterul este dispozitivul de interconectare ce are rolul de a determina calea ce trebuie urmata de un pachet pentru a ajunge la destinatie, de a interconecta si a schimba pachete intre rezele diferite. Ruterul este un dispozitiv de interconectare ce poate fi intalnit mai ales la nivelWAN, dar si la nivelul rezei locale, una din functiile sale principale fiind si cea de a oferi posibilitatea conectarii LAN-urilor la WAN.

Procesul de rutare sau de determinare a caii optime se bazeaza pe construirea si mentinerea unei tabele de rutare. O intrare intr-o tabela de rutare se numeste ruta si este compusa din minim 3 elemente: adresa de retea, masca de retea, adresa urmatorului ruter si/sau interfata de plecare.

### *Ce se intampla cu un pachet ajuns la un ruter?*

Antetul de nivel legatura de date este despachetat. Acesta va contine doar adresa logica a destinatiei si nu si masca de retea. Ruterul va verifica mai intai daca adresa destinatie nu este cumva una dintre adresele sale. Daca este atunci cadrul va fi trecut la nivelul superior, daca nu ruterul va verifica daca adresa destinatie nu este in accesi retea cu interfata de pe care a primit pachetul. Daca este atunci ruterul nu va mai face nimic cu acest pachet si va trece la urmatorul. Daca este atunci va abandona prelucrarile asupra respectivului pachet si va lua urmatorul pachet. In cazul in care destinatia nu este nici el si nici nu se afla pe accesi interfata de unde a primit pachetul, atunci va incepe procesarea tabelii de rutare. Va extrage prima ruta din tabela si va aplica masca de retea adresei destinatie continuta in antetul pachetului. Rezultatul il va compara

cu adresa de retea a respectivei rute. Dacă cele două coincid pachetul va fi trimis pe interfața specificată de ruta. Dacă nu este extrasă o nouă ruta din tabelă. Procesul se repetă până la ultima ruta din tabelă sau până la găsirea primei potriviri. Dacă pachetul nu corespunde nici ultimei rute atunci acesta este abandonat și se trece la pachetul următor. Înainte de a trimite pachetul sau îl abandona tabelă ARP a interfeței pe care a sosit pachetul va fi actualizată folosindu-se adresa MAC și cea IP a sursei.

*Care este efectul rutelor asupra domeniilor de difuzare și a domeniilor de coliziune?*

La nivelul legăturii de date punțile detectau coliziunile și nu le transmiteau mai departe, dar cadrele de detectare a coliziunilor este una din principalele funcții ale nivelului legăturii de date, iar rutatorul însuși și toate funcțiile primelor niveluri va face atât regenerarea semnalului cât și detectarea coliziunilor. În plus ruterele, spre deosebire de punți au acces și la informațiile de nivel rețea această permitându-le controlul difuzării și a pachetelor de multicast. În mod implicit ruterele nu transferă pachetele de difuzare sau de multicast.

Concluzionând putem spune că ruterele marginesc atât domeniile de coliziune, cât și pe cele de difuzare.

## 7. Partea practică

1. **Obiectiv :** *Care este rolul punții în comunicarea din interiorul aceluiași segment?*

Protocolul Ethernet oferă un mediu de comunicare distribuit, adică comunicarea dintre două stații și accesibilă nivelului legăturii de date a oricărei alte stații conectate pe același segment. Pentru fiecare cadru primit de o stație, nivelul legăturii de date va verifica dacă aceasta stație este sau nu destinația. În cazul afirmativ cadrul va fi pasat nivelului rețea, altminteri va fi ignorat. Pentru cazul comunicării în interiorul aceluiași segment, să considerăm rețeaua din figura de mai jos. Presupunem că stația A1 vrea să transmită date stației A2.

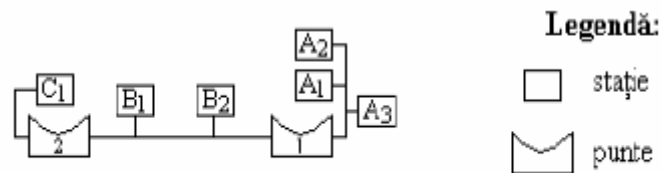


Fig. 1: Rețea segmentată cu punți

Suntem într-o rețea Ethernet, așa că primul lucru pe care-l va face stația A1 va fi ascultarea mediului. Dacă mediul este liber va începe transmiterea datelor. Cadrul emis de A1 se va propaga către toate stațiile conectate pe acest segment, inclusiv către punte. Stația A2 va trece cadrul către nivelul rețea, stația A3 îl va ignora. Odată ajuns la punte cadrul este despachetat și adresa destinație este căutată în tabelă de comutare a punții. Puntea va decide dacă destinația se află chiar pe interfața pe care a primit cadrul. În acest caz puntea ia decizia că acest cadru nu mai trebuie transmis, deoarece retransmiterea cadrului ar duce la o duplicare a acestuia la destinație.

Cum va acționa puntea 1 în cazul comunicării între B1 și B2?

Ambele punți (deși vor recepționa cadrele) vor lua decizia de a nu le mai retransmite. Dar să presupunem că cele două comunicații apar simultan: atât A1 transmite către A2, cât și B1 către B2. Va apărea în acest caz o coliziune? Dacă în loc de puntea 1 am fi folosit un repetor, cu siguranță am fi avut o coliziune. În cazul nostru, de vreme ce nici un cadru din comunicarea dintre A1 și A2 nu va ajunge pe segmentul B, și nici un cadru din comunicarea dintre B1 și B2 nu va ajunge pe segmentul A, este imposibil să apară o coliziune.

2. **Obiectiv :** *Cum își construiește puntea tabelă de comutare?*

În exemplele anterioare am presupus că tabelă de comutare era deja construită. Această tabelă este păstrată bineînțeles în memoria RAM a punții, prin urmare se va pierde dacă reinitializăm puntea. În plus, o punte trebuie să fie în stare să includă dinamic în tabelă de comutare informații despre o nouă stație conectată în rețea.

Sa consideram retea din figura de mai jos, unde puntea 1 a fost reinitializata, si statia A1 vrea sa comunice cu statia B1.

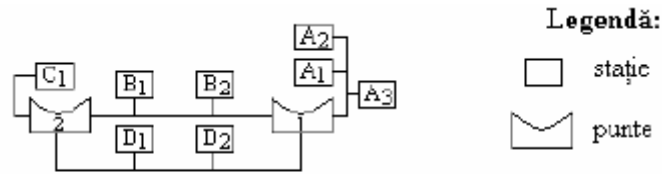


Fig. 2: Construirea tabeli pentru puntea I

Statia A1 asculta mediul, iar cand acesta este liber trimite un cadru ce are ca destinatie statia B1. Statiile A2 si A3 vor ignora cadrul. Puntea 1 va primi cadrul si va incerca sa gaseasca adresa destinatie in tabela sa de comutare. Puntea nu va reusi sa gaseasca destinatia, deoarece tabela sa de comutare era goala, astfel incat va retransmite cadrul pe toate segmentele la care este ea conectata, in afara de segmentul de pe care a fost primit cadrul. Inainte de a retransmite cadrul puntea va verifica daca adresa sursa este prezenta in tabela sa de comutare. In cazul nostru ea nu este, astfel incat puntea va crea prima intrare in tabela de comutare ce va contine adresa fizica a statiei A1 si interfata ce conecteaza segmentul A. Cadrul va ajunge atat pe segmentul D, unde statiile D1 si D2 vor determina ca nu acesta nu le este adresat lor, deci il vor ignora; cat si pe segmentul B, la statia B1, B2 si puntea 2. Puntea 2 va determina ca destinatia este in acelasi segment din care a primit cadrul si va decide sa nu?l mai retransmita, iar statia B1 va determina ca ea este destinatarul cadrului.

Chiar si comunicatia intre doua statii aflate in acelasi segment poate afecta latimea de banda din intreaga retea daca puntea nu a apucat sa-si construiasca tabela de comutare. Dupa cadrul trimis catre statia B1, sa consideram ca statia A1 va trimite un cadru pentru A2. Cadrul va ajunge la destinatie fara ajutorul puntii, dar puntea, neidentificand destinatia in tabela sa de comutare, va retransmite cadrul atat pe segmentul B, cat si pe segmentul D.

Datorita dificultatii cautarii intr-o multime neordonata, in tabela de comutare nu se vor pastra toate adresele statiilor din retea locala, ci doar a celor ce au o probabilitate mare sa transmita in viitorul apropiat, mai exact a ultimilor statiilor ce au transmis. Pentru implementarea acestui concept, o intrare intr-o tabela de comutare va avea, pe langa adresa MAC si interfata, si o eticheta de timp. Aceasta eticheta de timp este actualizata la o noua primire a unui cadru cu aceeasi adresa sursa. Acest mecanism permite inlaturarea intrarilor invecinate si deci restrangerea dimensiunii tabeli de comutare. Pretul platit pentru aceasta este consumul din latimea de banda a tuturor segmentelor din retea in cazul in care o statie nu transmite nici un cadru un interval de timp.

### 3. Obiectiv : Care este efectul aparitiei buclelor de nivel legatura de date?

Aparitia buclelor de nivel legatura de date este corelata cu faptul ca puntile si comutatoarele nu filtreaza pachetele de difuzare si duc la o depreciere semnificativa a performantelor retelei prin determinarea unor avalanse de difuzari (broadcast storm).

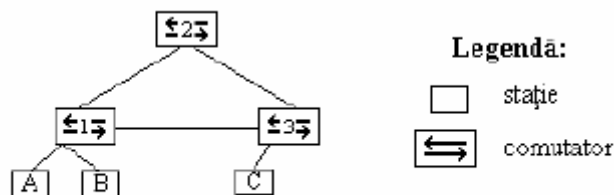


Fig. 4: Retea in care s-a creat o bucla

Sa consideram retea din figura de mai sus. Presupunem ca statia A1 trimite un cadru de difuzare. Comutatorul 1 nu va gasi adresa destinatie in tabela sa de comutare, astfel incat va transmite cadrul pe celelalte segmente: segmentul ce contine statia B, segmentul dintre comutatoarele 1 si 2, si segmentul



dintre comutatoarele 1 si 3. Statia B va examina cadrul, va decide ca ii este adresat si il va trece spre nivelul legatura de date.

Comutatorul 2 va lua decizia de a transmite cadrul pe toate interfețele sale, cu exceptia celei de pe care a primit cadrul. Am ajuns ca sa avem in retea doua cadre destinate statiei FF.FF.FF.FF.FF.FF, adica doua cadre de difuzare. Indiferent de ordinea in care acestea ajung la comutatorul 3, acesta va determina ca nu cunoaste adresa destinatie si le va retransmite catre statia C, dar si catre celelalte comutatoare.

Avalansa de difuzari consuma din banda utila a rețelei, ducand la o micșorare a bandei efective disponibile. O avalansa de difuzari se va opri doar in cazul intreruperii buclei.

4. **Obiectiv** : Contruirea arborelui de acoperire a switch-ului.

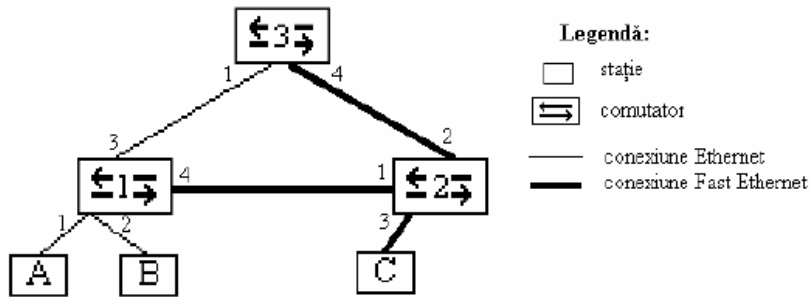


Fig. 5: Construirea arborelui de acoperire

Fie rețeaua din figura de mai sus. Vom urmări pentru această rețea etapele construirii arborelui de acoperire.

Prima întrebare pe care trebuie să ne-o punem este: care este prioritatea fiecărui comutator? Să considerăm că toate cele trei comutatoare sunt produse de același fabricant și în plus sunt abia scoase din cutie. Asta înseamnă că toate comutatoarele vor avea aceeași prioritate. În acest caz va trebui să aflăm adresele fizice.

comutatorul 1	00.C2.45.26.57.A1
comutatorul 2	00.C2.45.2E.08.EF
comutatorul 3	00.C2.45.A2.11.49

Din analiza tabelului de mai sus, rezulta că rădăcina arborelui de acoperire va fi comutatorul 1. În continuare vom determina pentru restul comutatoarelor costurile porturilor ce oferă calea spre comutatorul rădăcină. Pentru comutatorul 2 costul portului 1 va fi 10 (=1000/100), iar pentru portul 2 va fi 110 (10 + costul portului 1 din comutatorul 3).

Pentru comutatorul 3, portul 1 va avea costul 100, iar portul 4 costul 20.

Pentru comutatorul 2 portul rădăcină va fi portul 1, astfel încât portul 1 trece în starea de comutare, în vreme ce portul 2 va rămâne în starea de blocat. Pentru comutatorul 3 portul rădăcină va fi portul 1, deoarece, chiar dacă are un cost mai mare decât portul 4, este direct conectat la rădăcină, astfel încât portul 1 va trece în starea de comutare.