

REȚELE LOCALE DE CALCULATOARE

Note de curs 2012

sl. dr. ing. IONESCU Valeriu Manuel

CURS 1

Cunoștințe fundamentale de rețele de calculatoare necesare pentru cursul de rețele locale de calculatoare

Tipuri de comutație:

- comutația de circuite (circuite fizice sau virtuale);
- comutația de pachete (în Rețele de calculatoare se folosește multiplicarea statistică pentru a folosi eficient canalul de comunicații) – sloturile alocate în funcție de cerere – rezultă folosire eficientă;
- comutația de celule (în rețele ATM) – dimensiunea frame-ului este fixă.

Modele folosite în RC (arhitecturi):

- OSI (7 straturi) – nu este folosit în practică;
- TCP/IP (4 straturi) – este folosit în practică.

Încapsularea datelor: segment, pachet, frame, biți, date.

Echipamente folosite în RC și repartizarea acestora pe nivelurile OSI:

- la nivelul fizic: hub-uri, repeatoare, transceiver, convertoare;
- la nivel date: switch/bridge;
- la nivel rețea: router.

Topologii fizice și logice în RC:

- fizice – așezarea/dispunerea fizica a calculatoarelor în rețea (stea, magistrala, inel, mesh, etc.);
- logice – se referă la modul în care pachetele sunt transmise: topologia de tip broadcast și cea de tip token passing.

Medii de transmisie folosite în rețele de calculatoare: cablu coaxial, cablu torsadat, fibră optică, wireless.

Structura plăcii de rețea (hard și soft) și funcționarea ei

Modul de funcționare al algoritmului CSMA/CD folosit în rețelele de calculatoare bazate pe Ethernet.

Adresele IP:

- Clasificare
- Mască
- Gateway
- Tabela de routare
- Calcul AR și AB

Protocoale de rutere:

- RIP – routing information protocol (rutere pas cu pas)
- OSPF

TCP vs. UDP

TCP – orientat pe conexiune;

UDP – nu are conexiune.

Servicii în rețele de calculatoare:

- WEB
- DNS
- DHCP
- FTP
- TelNet
- SSH

Metode pentru securizarea Rețelelor de calculatoare:

- autentificarea;
- criptarea datelor: Tipuri de criptare: cu cheie publică ; cu cheie privată.
- algoritmi: RSA - pentru criptare cu cheie publică;
AES – pentru criptare cu cheie privată.

Traficul în rețele de calculatoare

Traficul este o componentă a rețelelor ce nu apare într-o prezentare structurală.

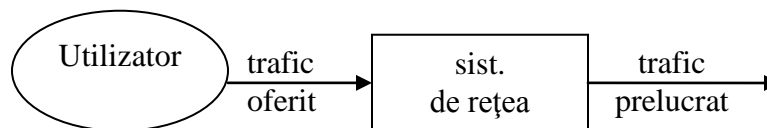
Traficul se face simțit prin ocuparea circuitelor și dispozitivelor.

Traficul nu poate fi stocat → prelucrarea se face în timp real.

Traficul poate fi:

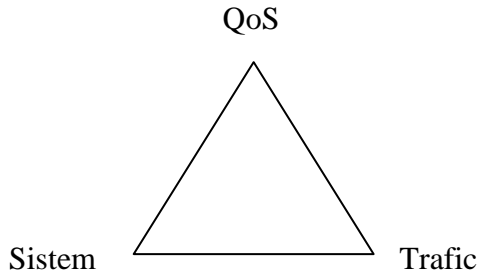
- oferit – se produce prin solicitările utilizatorilor;
- prelucrat – circulă printr-o rețea de calculatoare, dimensionat corespunzător.

Traficul diferă în funcție de modul de comutare folosit.



Un sistem trebuie să răspundă la momentul proiectării la una din următoarele întrebări:

1. Fiind dat un sistem cu un anumit trafic oferit, care este calitatea serviciului care poate fi garantat?



2. Cum trebuie dimensionat sistemul dacă știu traficul oferit și calitatea dorită?
3. Pentru un sistem și o calitate dorită, care este traficul maxim suportat?

Există trei factori ce trebuie luați în calcul în proiectarea unei rețele: QoS, trafic și capacitatea sistemului.

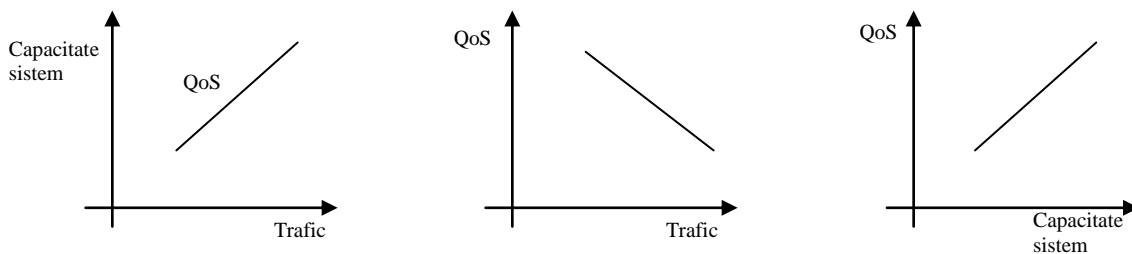
Sistemul poate fi un singur element, întreaga rețea sau o parte a acesteia. Traficul caracterizat prin fluxuri, conexiuni, biți, pachete (depinde modul în care se realizează schimbul de date) – depinde de perioada de timp (transferul).

Calitatea serviciului în rețele de calculatoare se refera la posibilitatea asigurării de priorități diferite pentru aplicații, utilizatori, fluxuri de date sau garantarea unui anumit nivel de performanță. Calitatea serviciului poate fi descrisă atât din punct de vedere al utilizatorului (în acest caz privește întârzierea, latența, viteza de transmisie, întreruperea conexiunii) cat și din punctul de vedere al sistemului când se folosește termenul de performanță în utilizarea rețelei.

Există trei elemente interdependente după cum urmează:

- QoS
- trafic
- capacitate sistem

Relația între cei 3 factori:



Pentru a descrie relația din punct de vedere cantitativ sunt necesare metode matematice.

Curs 2

Modele de trafic

Modelele de trafic sunt de tip probabilistic (stochastic) deoarece momentul efectuării schimbului de date nu poate fi prevazut.

Elementul probabilistic este determinat de variabile aleatoare precum numarul de conexiuni, numarul de pachete din buffer si dimensiunea pachetelor.

S-au elaborat modele probabilistice care incearca sa descrie parțial sistemele reale. Scopul practic al elaborarii modelelor este proiectarea corecta a retelelor de calculatoare. Aceasta are doua aspecte: planificarea retelelor si managementul retelelor.

Planificarea cuprinde dimensionarea, optimizarea si analiza performantelor.

Managementul se refera la operarea eficienta, recuperarea erorilor si poate si contabilitatea atunci cand este vorba de un trafic platit.

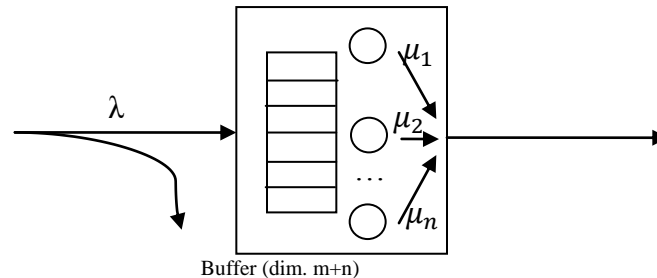
Exista 3 tipuri de sisteme model:

- cu pierderi
- cu cozi de așteptare
- cu partajare

Acestea sunt modele simple care in realitate se pot combina pentru a crea modele complexe.

Consideram următoarele elemente care descriu un model de trafic:

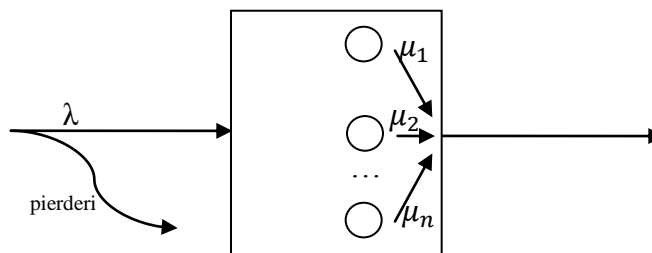
- λ - rata de sosirea a utilizatorilor
- $1/\lambda$ - perioada intre sosirile utilizatorilor
- μ - rata serviciului
- $1/\mu$ - timpul mediu de asteptare pentru un utilizator.



Bufferul poate avea n pozitii de lucru si m pozitii de asteptare.

1. Sistem cu pierderi

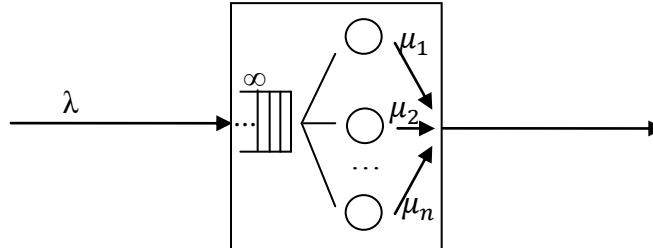
- Nu exista pozitii de asteptare ($m=0$);
- Utilizatorii peste capacitate sunt pierduti, daca sistemul este plin.
- de interes este probabilitatea ca sistemul sa fie plin cand utilizatororii sosesc



2. Sistemul infinit

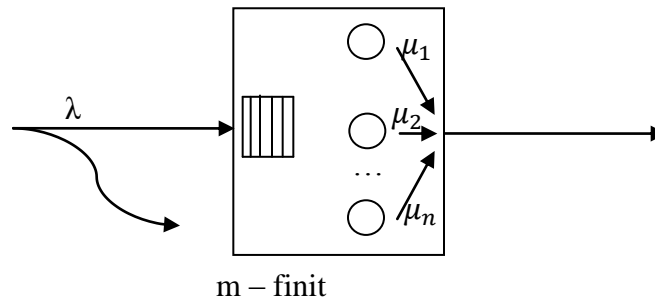
- Acesta este un model teoretic care are un număr infinit de servere care procesează cererile clienților;
- Nu există pierderi și nici așteptare;
- Este folosit pentru a determina limita superioară a unui sistem real.

3. Sistem pur cu așteptare



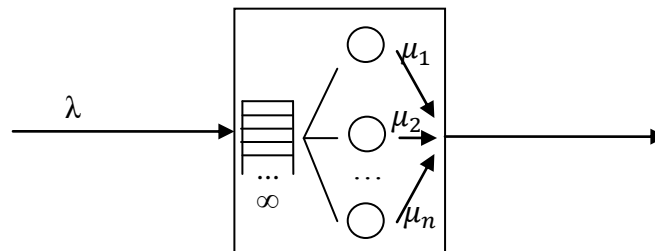
- Nu are pierderi;
 - Are un număr finit de poziții de lucru;
 - Dimensiunea bufferului de așteptare este infinită;
 - Dacă toate serverele sunt ocupate când sosește un client, atunci acesta ocupă o poziție de așteptare.
- Dezavantaj: - unii clienți ar putea să aștepte foarte mult.

4. Model complex: sistem de așteptare cu pierderi.



- Dimensiunea bufferului (m) este finită;
- Dacă bufferul este gol, clienții sunt puși în așteptare;
- Dacă bufferul este plin, clienții sunt pierduți.

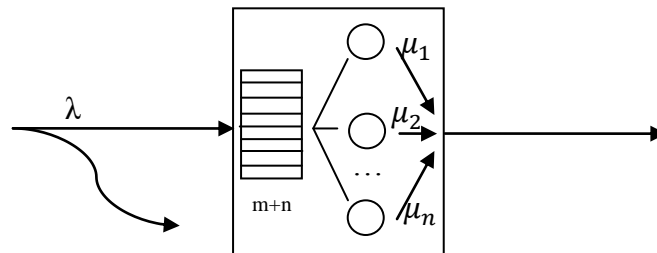
5. Sistem pur cu partajare.



- Niciun client nu este pierdut;

- Niciun client nu trebuie sa astepte;
- Daca numarul utilizatorilor este mai mic sau egal decat numarul de servere, fiecare are propriul server;
- Daca numarul de clienti este mai mare decat numărul de servere (n) atunci rata de serviciu este impartita \rightarrow serverele proceseaza pe rand si clientii suplimentari. O problema poate fi intarzierea procesarii cererilor.

6. Sistemul cu partajare si pierderi.



- Dimensiunea bufferului este finita;
- Niciun client nu asteapta, dar unii pot fi pierduti.

Legea lui Little

Considerând un sistem stabil cu rata medie a sosirilor λ , \bar{N} numarul mediu de clienti din sistem si \bar{T} timpul petrecut in medie de un client in sistem, următoarea relație este valabila: $\bar{N} = \lambda * \bar{T}$,

Aceasta lege poate fi reformulata astfel: Timpul mediu petrecut intr-un sistem este egal cu timpul mediu de așteptare plus cu timpul mediu necesar pentru primirea serviciului.

Aceasta relatie este importanta deoarece atata faptul ca factori precum: nu au nici o de influenta in relatia dintre aceste elemente. Legea este utilizata in gestionarea sistemelor complexe (formate din mai multe sub-sisteme) pentru stabilirea timpului de traversare sau al ritmului de productie. In retelele de calculatoare, de exemplu, poate fi vorba despre timpul necesar unui client sa primeasca serviciul de la server (de la momentul in care cererea clientului ajunge la server pana la momentul in care datele procesate pleaca de la server catre client).

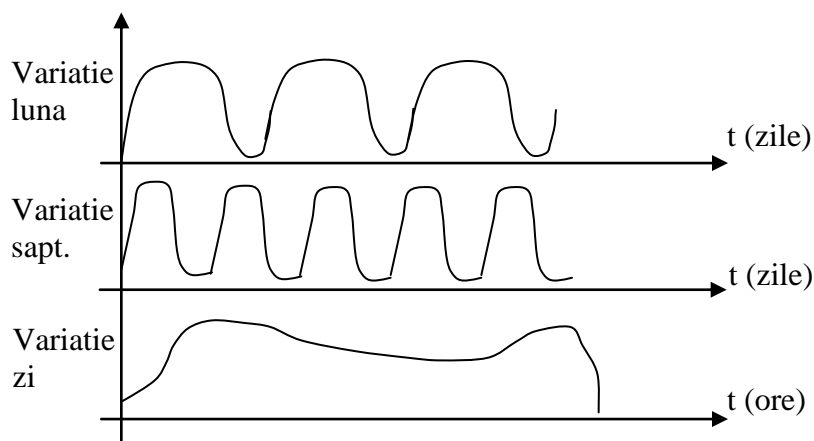
Traficul in rețele de calculatoare

Traficul este cantitatea informatională în mișcare care implică noțiunea de debit. Aspectele traficului sunt: intensitate, fluiditate și situații de blocare sau saturare.

Traficul are diferite forme: trafic oferit (dpdv al providerului), trafic prelucrat, trafic refuzat, trafic de așteptare, trafic de comandă.

Clasificarea traficului

Traficul în rețea nu este constant, el având variații predictibile la nivel de an, săptămână, zi.



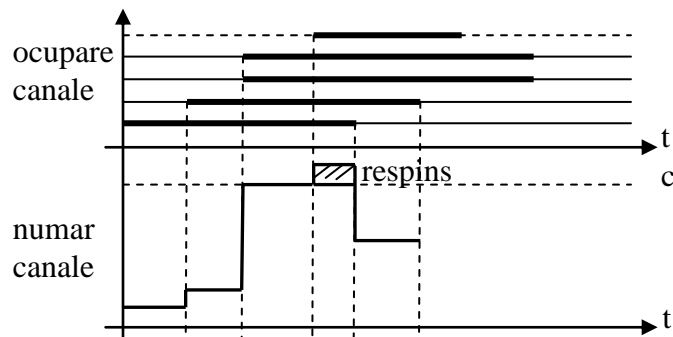
Pentru dimensionarea traficului este necesară o estimare a acestuia; în estimare se vor considera în special variațiile predictibile și mai puțin variațiile nepredictibile.

Variații nepredictibile: variații la nivel de minut, variații aleatoare pe termen lung privind de exemplu modificarea profilului utilizatorului, variații datorate evenimentelor neprevăzute.

Trafic:

- Comutație de circuite;
- Comutație de pachete:
 - o La nivel de pachet (IP);
 - o Flux:
 - UDP;
 - TCP.

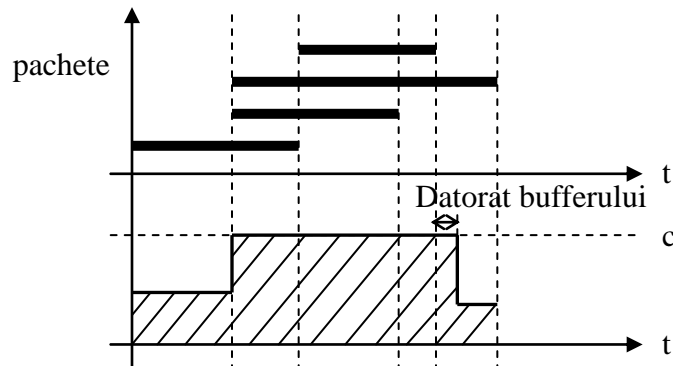
Traficul în comutație de circuite – este un sistem pur cu pierderi în care un server corespunde unui canal, rata de serviciu depinde de timpul mediu de ocupare, numărul de servere depinde de capacitatea liniei, când toate canalele sunt ocupate apelurile suplimentare sunt respinse.



In comutia de circuite se foloseste tehnica TDM (time division multiplexing).

Traficul la nivelul pachetelor

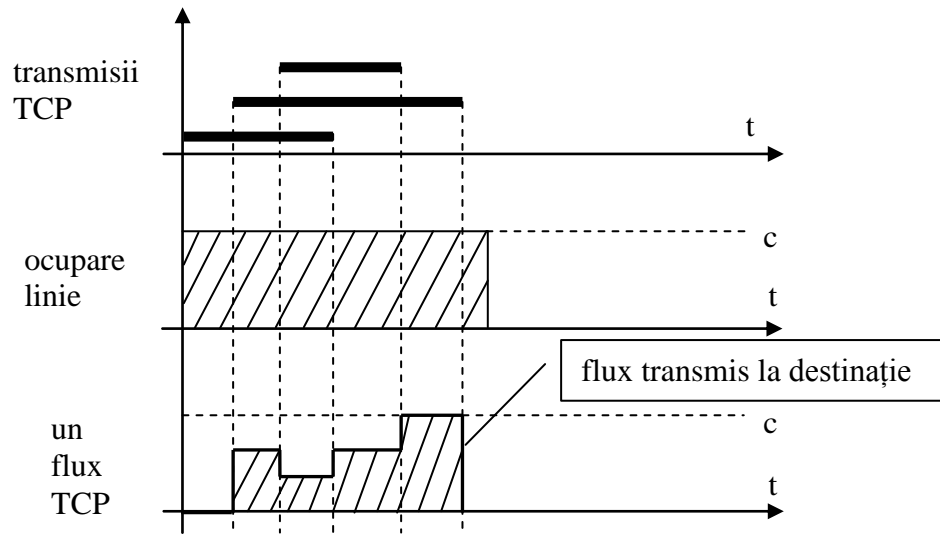
- Este un sistem cu asteptare bazat pe un singur server;
- Rata de serviciu depinde atat de capacitatea liniei cat si de lungimea medie a pachetelor;
- In comutatia de pachete utilizatorii concursa pentru resursele retelei;
- Cand linia este ocupata pachetele noi sunt trecute in buffer; daca acesta este plin, pachetele sunt pierdute.



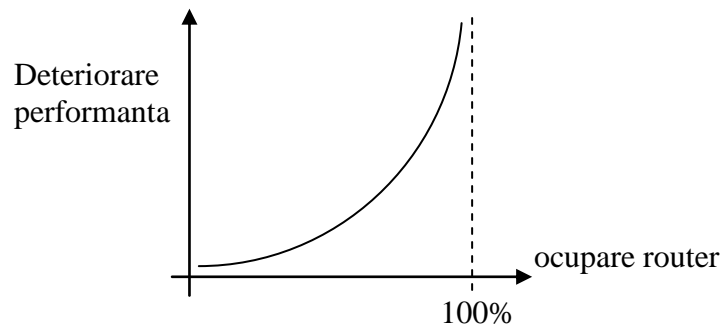
Traficul la nivel de flux.

Traficul TCP se mai numeste si trafic elastic deoarece isi poate redimensiona fluxul in timpul realizarii schimbului de date. Modelul este un sistem cu partajare deoarece nu exista controlul intrarilor deci niciun flux de intrare nu este respins.

Rata de serviciu depinde de capacitatea liniei si marimea fluxului.

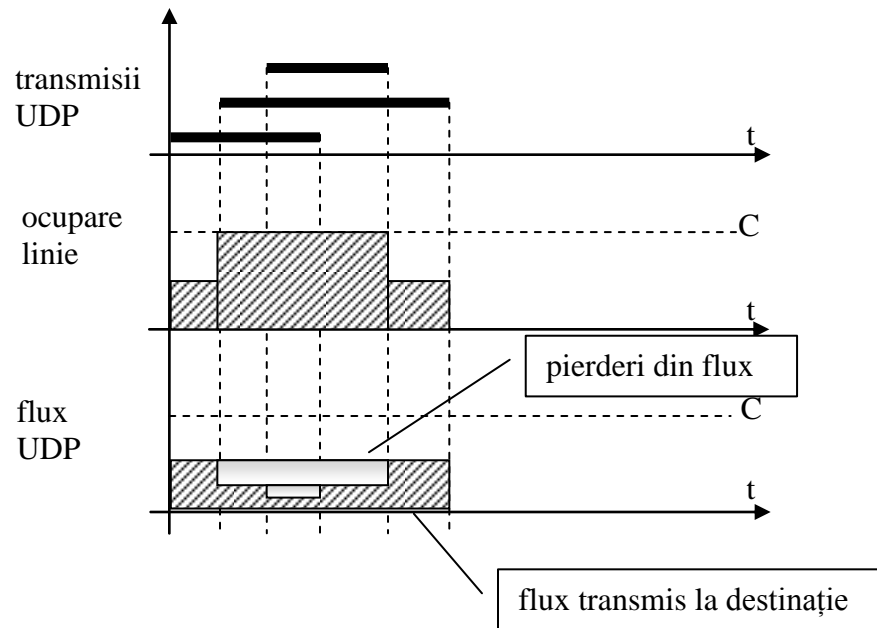


Se observa cum in cazul a trei transmisii TCP simultane banda disponibila se imparte in mod egal la cele trei fluxuri, fara sa aiba loc pierderi de date. Tehnica folosita de fluxul TCP pentru creștere este creșterea aditiva (treptat) si descreșterea multiplicativa (cu 1/2).



Traficul de flux tip UDP (se mai numeste si streaming):

- Modelul este cel al unui sistem infinit;
- Rata de prelucrare depinde de lungimea fluxului;
- Nu exist bufferi → traficul suplimentar este pierdut uniform din toate streamurilor;
- Nu exista control al intrarilor.



Spre deosebire de fluxurile TCP care pot detecta congestia liniei de comunicație (și în consecință își se vor redimensiona), fluxurile UDP care însumate depășesc capacitatea liniei de comunicației nu se vor modifica, consecința fiind ca se vor pierde date din toate aceste fluxuri.

În figura se observă cum fiecare flux UDP are pierderi, însă, deoarece în UDP nu este implementat un mecanism de detecție a apariției congestiei rețelei, acestea vor persista cât timp este depășită capacitatea canalului de transmisie. Considerând că un flux UDP ocupă 50% din capacitatea liniei de comunicație, la apariția mai multor fluxuri UDP, doar o mică parte din date vor mai ajunge la destinație, celelalte fiind pierdute.

CURS 4

Calitatea serviciului (Qos- quality of service) ~Definirea calitatii serviciului si a performantei retelei~

Calitatea serviciului in rețele de calculatoare se refera la posibilitatea asigurării de prioritați diferite pentru aplicații, utilizatori, fluxuri de date sau garantarea unui anumit nivel de performanta.

Efectul performantelor serviciului dpdv al utilizatorului = calitatea serviciului.

Performanta retelei = capacitatea retelei de a realiza functiile de comunicatie intre utilizatori:

- Accesul la mediul de transmisie;
- Stabilitatea comunicatiei;
- Calitatea comunicatiei.

Calitatea serviciului are in vedere urmatoarele elemente:

1. Adaptarea retelei la nivelul de performanta solicitat.
2. Accesul la comunicati de voce, video si date.

Performanta retelei, dpdv al providerului, trebuie sa asigure controlul urmatorilor parametri:

- Latime de banda (band width \neq through put = viteza de transfer instantanee);

Exp. O linie de 10 Mb/sec latime de banda

Transfer de 5 Mb/sec

Through put < latime de banda (exceptie cand se foloseste compresia).

- Intarzierea;
- Variatia intarzierii (jitter);
- Pierderea de pachete.

In schimbul de date, entitatile implicate sunt urmatoarele:

1. Clientul – este partea care plateste pentru serviciile asigurare.
2. Furnizorul de retea – asigura conectivitatea, dar nu neaparat serviciile.
3. Furnizorul de servicii – asigura restul serviciilor (exp. DNS – serviciul de nume).
4. Utilizatorul – cel care foloseste serviciile oferite.

~Cerinte de performanta impuse retelelor IP~

Latimea de banda nu este un element dinamic, deci nu suporta criterii de performanta.

Intarzierea este diferenta intre timpul la care este receptionat si timpul la care a fost transmis. Ea este determinata de:

- Modul de serializare a informatiei (dimensiunea pachetului);

- Timpul de propagare (de ordinul μ s-elor);
- Timpul de comutatie care creste in functie de marimea cozii de asteptare (router).

Jitterul (variatiia intarzierii) – pentru reducerea acestuia se folosesc bufferi. Din pacate bufferii, maresc intarzierea.

Pierderea de informatie este determinata de erori de bit sau de pierderi de pachete si poate fi imbunatatita prin marirea compresiei datelor.

In concluzie, putem clasifica cele doua elemente componente ale calitatii serviciului astfel:

1. QoS orientat pe utilizator, vizeaza serviciile retelei; se manifesta intre punctele de acces ale serviciului.
2. Network performance este orientat pe provider (furnizor); priveste elementele de conectivitate si se manifesta intre elementele de conectivitate.

In prezent, sunt definite mai multe niveluri pentru calitatea serviciului (sunt numerotate de la 0 la 5 si apar in pachetul IP):

- 0 – servicii de timp real; retea asigura cozi de asteptare cu servire preferentiala;
- 1 – tot pentru servicii de voce si timp real, de prioritate redusa;
- 2 – serviciile de semnalizare in retea (confirmari, acknowledgement etc.);
- 3 – servicii de date cu interactivitate redusa;
- 2 si 3 – retea asigura cozi de asteptare separate prioritare la pierderi;
- 4 – aplicatii cu pierderi mici (transferuri scurte, videostreaming); cozile de aplicatii sunt lungi si directionarea pachetelor se face pe orice traseu disponibil;
- 5 – servicii (restul serviciilor IP).

Pentru clasa 0 avem jitter sub 100 ms.

1	sub 400 ms.
2	sub 100 ms (cu pierderi de pachete).
3	sub 400 ms.
4	sub o secunda.
5	nicio restrictie (exista pierderi de pachete).

Importanta calitatii serviciului pentru diferite servicii IP:

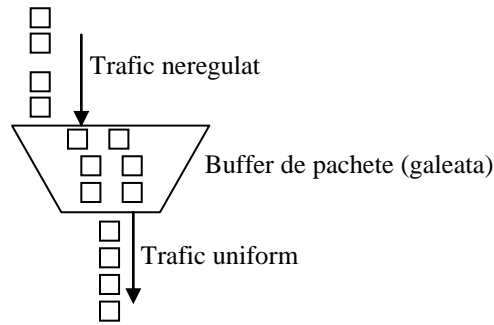
Posta electronica (email)	Mica	Mica	Mica	Mare
Traficul web	Medie	Mare	Mica	Mare
Remote desktop	Medie	Mare	Medie	Mare
Videostreaming	Mare	Mare	Mare	Medie
Video conferinta	Mare	Mare	Mare	Medie
	Latime de banda	Latenta	Jitter	Pierderi

Rețelele IP au fost proiectate sa fie de tipul best-effort, adica pierderea unui pachet sa nu fie semnalizata inasa se incearca transmiterea acestora in proportie cat mai mare. Din acest motiv, protocoalele de nivel superior precum TCP pot asigura un anumit nivel al QoS.

Metode de asigurare a QoS:

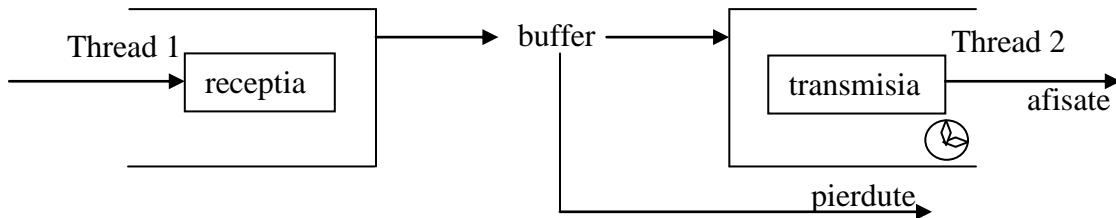
1. Supra-aprovizionarea este rareori aleasa deoarece implica costuri suplimentare, sau nu este disponibila.
2. Memorarea temporara – este o solutie pentru jitter in detrimentul latentei. Nu sunt afectate fiabilitatea si latimea de banda.
3. Trafic shaping (modelarea traficului) – este metoda preferata de provideri pentru a monitoriza / controla fluxul de date. Foloseste tehnici multiple pentru evitarea congestiei / aglomerarii rezelei.
4. Metode de control al traficului intr-un nod al rezelei

Algoritmul Leaky-bucket (galeata sparta).

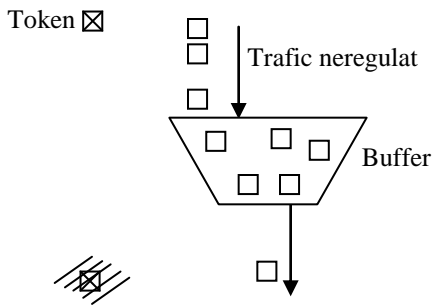


Aceasta solutie presupune folosirea unui buffer pentru a uniformiza traficul. Pachetele ies din buffer cu o anumita frecventa independenta de rata de intrare. Daca bufferul se umple, pachetele sunt pierdute.

Aplicatia server are doua componente:



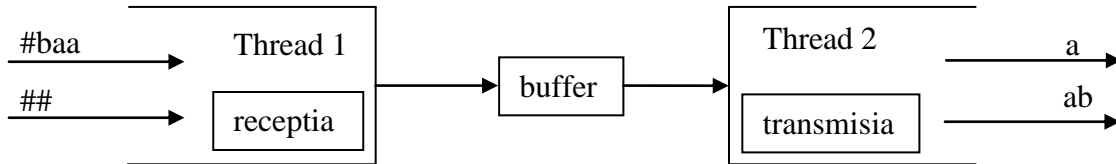
Algoritmul Token-bucket



Bufferul de pachete scoate date doar daca se primesc caractere de control (tokeni / jeton) pentru un pachet iesit se distruge un token.

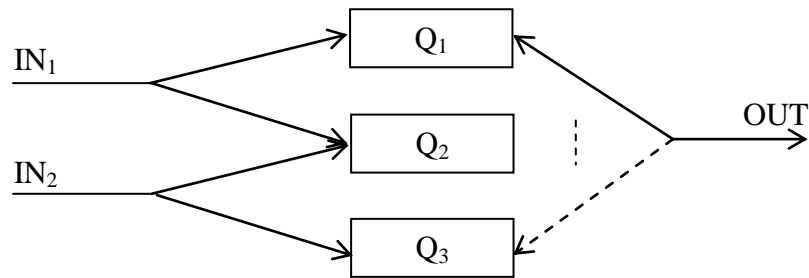
Algoritmul ignora/elimina tokenurile daca bufferul este plin.

Implementarea permite controlul iesirii de la sursa; din buffer nu se pierde pachete; este algoritmul folosit la comunicatia intre routere.



5. Rezervarea resurselor implica stabilirea resurselor necesare la momentul conexiunii si utilizarea lor exclusiva.
 - Se foloseste un protocol dedicat RSVP.
6. Controlul accesului – routerele primesc cereri de QoS specificate prin parametrii QoS doriti si alti parametri aditionali precum dimensiunea pachetelor.
7. CQS (classification Queing Scheduling) – aceasta metoda presupune clasificarea pachetelor, introducerea in cozile de asteptare (modul in care se alege din fiecare coada de asteptare sa fie transmis) si programarea.

CURS 5



Clasificate (C)

Coadă de asteptare (Q)

Planificare (S)

IntServ – mecanism ce are algoritmi orientati pe flux si foloseste protocolul RSVP (resource reservation protocol).

Dezavantajele modelului IntServ sunt:

- Fiecare flux trebuie semnalizat in retea, lucru problematic in cazul unui numar mare de fluxuri;
- Echipamentele necesita bufferi de dimensiuni mari;
- Schimbul de informatie intre routere este complex.

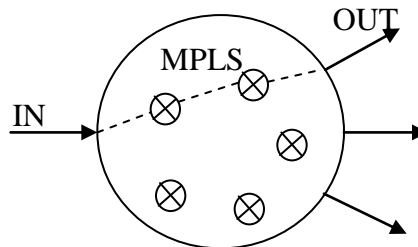
Deși IntServ are avantaje privind eficiența rezervării resurselor, costurile suplimentare au făcut ca în prezent să fie răspândite metodele DiffServ.

DiffServ clasifică fluxurile în funcție de calitatea serviciului oferit.

DiffServ:

- Marchează pachetelor prin scrierea câmpurilor ToS (type of service) – nu necesită rezervarea de resurse;
- Traficul este extrem de redus.

În prezent, se folosește din ce în ce mai mult tehnologia MPLS pentru asigurarea calității serviciului. Inițial, MPLS a fost proiectată drept comutație de pachete orientată pe conexiune.



MPLS:

- Definiște o zonă în care traseul de rutare poate fi definit explicit;
- Fiecare flux este marcat printr-o etichetă care identifică traseul ce trebuie urmat în rețeaua MPLS. În interiorul rețelei se găsesc routere numite label switch router (LSR) care recepționează un pachet etichetat, citește destinația și înlocuiește eticheta cu una nouă prin care identifică următorul router LSR.

Avantajul este analiza unică a etichetei.

Deoarece MPLS simplifică mult direcționarea pachetelor putând să grupeze fluxuri de același tip prin etichete similare; posibilitatea direcționării atât a pachetelor IP cât și a ATM mecanismul MPLS a permis implementarea garanției calității serviciului.

~Metode de planificare a pachetelor~

Planificarea pachetelor este ultima etapă de procesare înainte de transmiterea pachetelor către interfața de ieșire.

Planificatorul este un arbitru / bloc de decizie ce hotărăște care pachet memorat în cozile de așteptare va fi trimis spre ieșire.

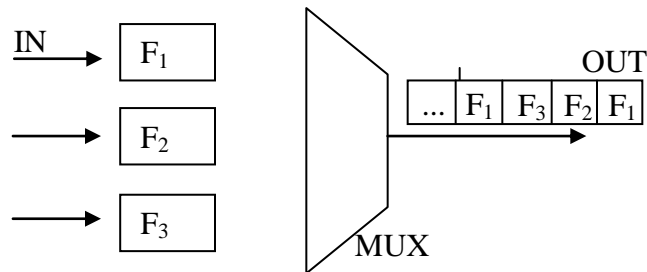
Se urmăresc în luarea deciziilor:

- *Asigurarea vitezei;*
- *Low balancing între conexiuni concurente;*
- *Garantarea ratei de pierderi;*
- *Garantarea întârzierii și jitterului.*

Metode de planificare:

1. FIFO

- Routerelor tradiționale au o singura coada de așteptare asociată interfeței de ieșire;

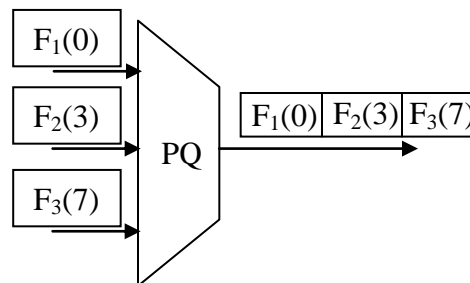


- Algoritmul este simplu (costuri reduse);
- Nu poate garanta întârzierea și viteza de transfer;

In routerelor moderne ce suporta functii de calitatea serviciului, planificatorul controleaza iesirea prin disciplina de planificare integrata in router.

2. Priority Queuing (PQ)

- Aceste routere analizeaza campul prioritate din antetul IP. In IP v6 acesta are valori de la 0-7 in care 0 este asociat traficului neclasificat, iar 7 este asociat traficului de prioritate maxima (exp. Traficul de routere);



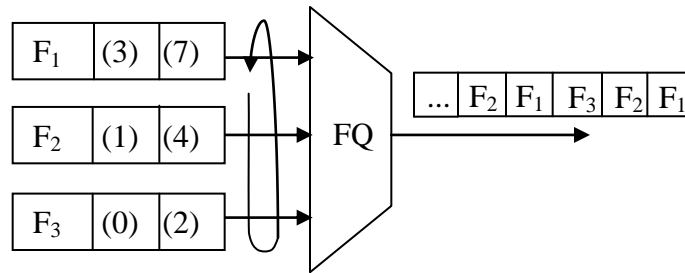
- Aceasta schema este strict prioritara.

Dezavantaj:

- poate bloca fluxurile cu prioritate redusa;
- timpul mediu de intarziere este dependent de valoarea MTU (maximum transmission unit).

3. FQ (fair queuing)

- Pentru aceasta tehnica / metoda planificatorul isi adapteaza dinamic disciplina de servire a cozilor;
- In fiecare etapa de transmisie orice flux are dreptul sa transmita un pachet;
- Coada de asteptare va fi reorganizata pentru fiecare flux in parte;



- FQ asigura un tratament egal pentru toate cozile de asteptare;
- Daca dimensiunea pachetului este fixa (retele ATM) se asigura o alocare echitabila a vitezei de transmisie;
- Daca pachetele au dimensiuni diferite sunt favorizate pachetele de dimensiuni mari.

4. WFQ (waited fair query)

- Permite diferentierea fluxurilor prin indrumarea lor in cozi de asteptare diferite in raport cu clasa de trafic asociata;
- Metoda foloseste campul IP precedence;
- Ponderile sunt stabilite de acest camp si nu pot fi modificate;
- Complexitatea planificarii depinde de numarul pachetelor in asteptare;
- Fluxurile care au campuri IP precedente setat vor obtine o rata de prioritate mai mare.

Exp. Daca avem 8 fluxuri cu ponderi de la 0 la 7, atunci traficul care are IP precedence setat in 0 va obtine:

$$IP_{precedence=0} = \frac{1}{1 + 2 + \dots + 8} = \frac{1}{36}$$

$$IP_{precedence=2} = \frac{3}{1 + 2 + \dots + 8} = \frac{3}{36} = \frac{1}{12}$$

5. CBWFQ (class based WFQ)

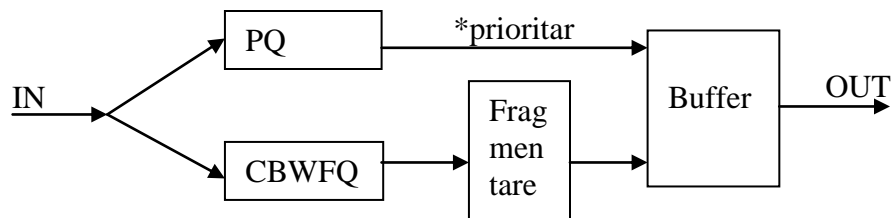
- Permite crearea a 256 clase de trafic, fiecare clasa avand propria coada de asteptare. Pentru fiecare clasa se garanteaza latimea de banda insa nu se garanteaza intarzierea.

CURS 6

Alte metode de planificare

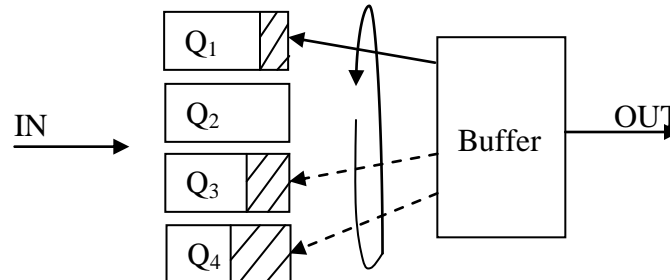
1. Low balancing quering

Acest mecanism combina metoda de clasificare dupa prioritate si metode de prioritizare in functie de clasa. Traficul asociat cozilor tratate cu algoritmul PQ este servit prioritar inaintea celui CBWFQ.



2. Round Robin

Algoritmul permite unui router sa organizeze mai multe cozi de asteptare. Pachetele se extrag pe rand din fiecare coada care nu este goala.



Q_i = cozi de asteptare

Aceasta este metoda de utilizare a bufferilor hardware din placa de retea.

Dezavantajul acestui algoritm este ca acorda latime de banda mai mare echipamentelor care folosesc pachete de dimensiune mare. O solutie alternativa ce rezolva aceasta problema este prelucrarea datelor octet cu octet.

3. Deficit Round Robin

Acest algoritm foloseste o variabila pentru a determina numarul de biti dintr-o coada de asteptare care trebuiesc preluati intr-o runda. Daca fiecare coada de asteptare are pachete de transmis, iar lungimea pachetului este mai mica sau egala cu valoarea variabilei, atunci pachetul este transmis si se decrementeaza variabila cu lungimea pachetului.

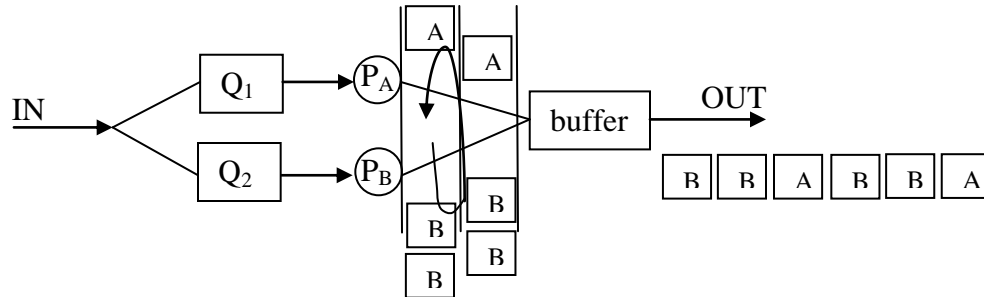
Daca lungimea pachetului este mai mare decat valoarea variabilei se asteapta runda urmatoare in care variabila este incrementata in mod automat.

Algoritmul poate defavoriza unele cozi de transmisie intr-o runda insa acestea vor recupera in una din rundele urmatoare.

4. Weighted Round Robin

Aceasta metoda preia n pachete dintr-o coada de asteptare intr-un ciclu cu n specific fiecărei cozi, valoarea n fiind aleasa astfel incat sa ocupe o anumita fractiune din latimea de banda.

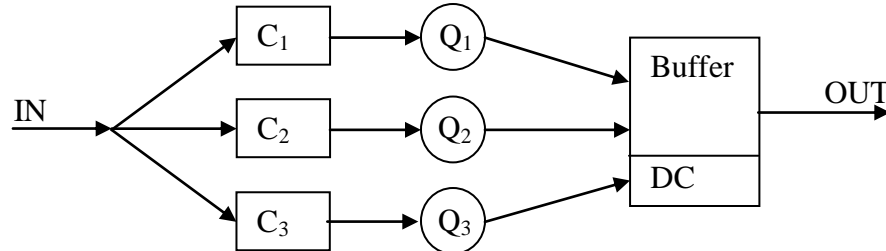
Exp. Daca in router intra doua fluxuri A si B, din router sunt extrase



Acest algoritm lucreaza bine cand dimensiunea pachetelor este fixa.

5. Deficit WRR

Fluxurile sunt clasificate in cozi de asteptare distincte dupa calitatea serviciului. Fiecare coada foloseste un indicator pentru numarul de octeti care vor fi preluati (Q) la un ciclu si un contor de deficit care cumuleaza intr-o variabila ce determina numarul total de octeti ce pot fi extrasi in runda curenta.



6. Planificarea prin eliminarea pachetelor

In schemele anterioare niciun pachet nu era pierdut. Prin aceasta metoda daca respectarea calitatii serviciului este pusa in pericol se ignora pachete din fiecare coada pentru a reduce intarzierea celorlalte pachete.

Pierderea pachetelor se face doar cu respectarea integritatii serviciului (se foloseste in cazul protocolului TCP mecanismul de evitare a congestiei).

CURS 7

Virtualizarea

Pentru calculatoare virtualizarea consta in crearea unei versiuni virtuale de platforma hardware, S.O., retele sau stocare.

Conceptul de virtualizare a aparut din mai multe motive:

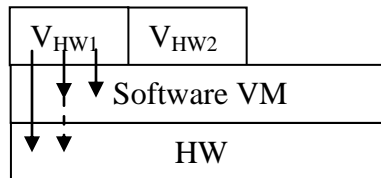
- Usurarea sarcinilor administrative;
- Imbunatatirea scalabilitatii;
- Din motive economice.

~Categorii / tipuri de virtualizare~

1. Virtualizarea hardware.
2. Virtualizarea software-ului.
3. Virtualizarea memoriei.
4. Virtualizarea stocarii.
5. Virtualizarea retelelor.

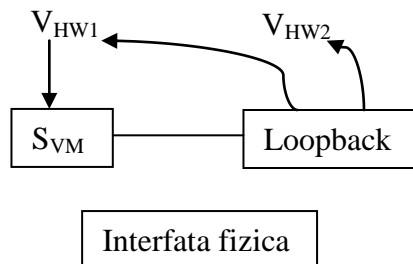
1. Virtualizarea hardware.

- Ascunde caracteristicile platformei pentru utilizatori prezentand un raport diferit (abstract);
- Virtualizarea este controlata de un software de monitorizare;
- Sistemul gazda al masinii virtuale acceseaza direct resursele hardware si le pune la dispozitia masinii virtuale cu anumite limitari.



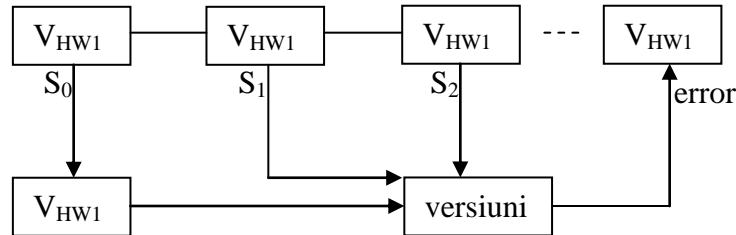
Hardware-ul virtual este de multe ori restrictionat in accesarea dispozitivelor periferice, a resurselor de retea sau afisajul.

Exp. Limitarea accesului la retea se face prin utilizarea accesului loop back.



Motivatia virtualizarii hardware:

- a) Datorita modului de functionare masinile virtuale pot fi mutatede pe un hardware pe altul foarte usor.
- b) Securitatea – expunerea masinii virtuale pentru demonstratii poate duce la afectarea securitatii sistemului. O masina virtuala poate fi resetata usor la o stare anterioara.



- c) Testarea aplicatiilor pentru mai multe configuratii hardware.
- d) Masina virtuala poate fi scalata fara achizitionarea in prealabil a unor resurse hardware.

Nivelurile de virtualizare

Virtualizarea hardware-ului poate fi: completa, partiala, paravirtualizare.

Acestea pot sa includa:

- a) Virtualizarea completa:
 - VMWare;
 - Virtual Box;
 - Virtual PC;
 - Qemu;

Virtualizarea hardware-ului este ajutata de anumite extensii implementate in microprocesor, tehnologiile: AMD-V, Intel VTx. Nu toate procesoarele au acest suport. Suplimentar, suportul nu este implicit activat (de obicei activarea se face in BIOS).

- b) Virtualizarea partiala** se foloseste in cazurile in care mediul hardware nu poate fi complet virtualizat. Masinile care ruleaza virtualizarea partiala au propriul spatiu de adrese in care ruleaza aplicatia. Rolul virtualizarii partiale este de a **suporta anumite aplicatii care partajeaza resurselor intre mai multi utilizatori.**

Avantaje:

- Partajarea resurselor;
- Asigurarea compatibilității.

Dezavantaje:

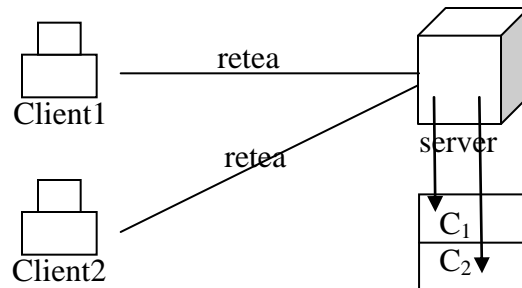
- Accesul direct la hardware poate afecta stabilitatea mașinii gazda.

- c) **Paravirtualizarea.** Aceasta tehnica nu simuleaza hardware ci ofera un API care poate fi folosita de S.O. virtual. Paravirtualizarea prezinta programelor un spatiu de adrese separat, iar aplicatiile trebuiesc modificate

special pentru a rula in acest mediu. Aplicatiile portabile folosesc paravirtualizarea.

2. Virtualizarea software-ului

O forma de virtualizare software intalnita in retelele de calculatoare se bazeaza pe modelul client-server si pune la dispozitia utilizatorului o interfata virtuala astfel incat aceasta are impresia ca ruleaza pe propria masina.



Acest mecanism poate fi utilizat doar daca serverul si latimea de banda permit conectarea simultana a mai multor utilizatori.

Clientii care virtualizeaza desktop-ul se mai numesc si *thin client*.

Virtualizarea la nivelul sistemului de operare are loc atunci cand kernelul unui sistem de operare permite existenta / prezenta mai multor instante izolate. Dpdv al clientului ele au toate caracteristicile calculatorului serverului insa sunt limitate in interactiunea cu ceilalti utilizatori conectati simultan la acelasi server.

In Linux-Unix comanda care permite virtualizarea este „chroot” (ch=change). Aceasta permite instalarea intr-un director a unei copii a sistemului de operare si orice utilizator marcat corespunzator va accesa acel sistem de operare care nu are legatura cu sistemul de operare real al serverului.

Aceste mecanisme pot fi usor evitate, din acest motiv au fost create alte aplicatii / servicii care au mecanisme de securitate mult mai bune.

Exp. Parallels, free BDS Jail.

3. Virtualizarea aplicatiilor.

Termenul descrie multe tehnologii pentru imbunatatirea portabilitatii si a compatibilitatii.

Exemplul des intalnit este al executiei aplicatiilor Win create pentru versiuni mai vechi. Setarile permit modificarea rezolutiei, rulara in full screen / fereastra, modificarea identificatorului sistemului de operare si accesul la memorie. Aplicatia WINE (Win emulator) permite rulara aplicatiilor Win sub Linux si executia programului Win inclusiv cel cu suport OPEN GL si DIRECT X.

CURS 8

Evolutia structurilor pentru virtualizare

Punctele de reper sunt:

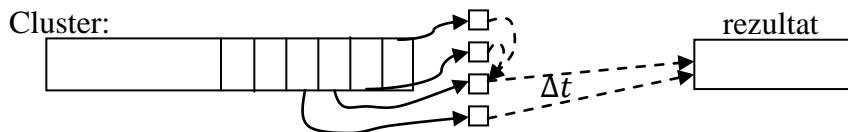
- a) Mainframe;

- b) Cluster;
- c) Grid;
- d) Cloud.

a) Mainframe

- A aparut incepand cu anii `60;
- Constau in grupuri mari de calculatoare, cu putere mare de calcul ale caror caracteristici principala era securitatea si redundanta hardware (mainframe-ul asigura o disponibilitate continua);
- Datele de intrare erau introduse prin istoric: cartele gaurite, apoi benzi magnetice, in prezent sistemele pot fi configurate din interfata WEB;
- Necesarul de putere s-a redus dramatic in anii `90 cand s-a realizat trecerea de la tehnologia bipolară la tehnologia CMOS;
- Scopul era realizarea aplicațiilor care necesitau putere mare de calcul si aveau surse multiple de date.

b) Cluster



Datorita evolutiei performantelor procesoarelor s-a dezvoltat o ramura specializata in procesarea de mare viteza folosind ca mediu de transfer intre calculatoare retelele locale. Acestea au ajuns la un nivel in care ofera latime de banda mare si latenta mica;

- Calculatoarele dintr-un cluster functioneaza impreuna ca un singur calculator;
- Exista mai multe tipuri de clustere:
 - o High availability cluster (HA) → au scopul de a imbunatati disponibilitatea serviciilor oferite;
 - o Load balancing cluster (LB) → apare cand mai multe calculatoare impart sarcinile de lucru funcționând ca un singur calculator virtual;
 - o Clustere pentru calcul (CC) → rolul lor este de a efectua calcule;
- poate necesita comunicatii mai frecvente între noduri decat alte clustere;

În informatică FLOPS este un acronim ce provine de la expresia engleză floating point operations per second (tradus: operații în virgulă mobilă pe secundă). FLOPS reprezintă o unitate de măsură a puterii de calcul a unui calculator sau sistem de calcul, măsurând numărul maxim de operații în virgulă mobilă, (de regulă adunări și înmulțiri), ce sunt executate pe secundă. Unitatea FLOPS își găsește folosul mai ales în domeniul calculului științifice, la care calculul în virgulă mobilă este frecvent folosit. estimativ: 2010, the fastest six-core PC processor reaches 109 gigaFLOPS (Intel Core i7 980 XE; Nvidia Tesla C2050 GPU computing processors perform around 515 gigaFLOPS.

In prezent se intocmesc ierarhii cu cele mai performante calculatoare virtuale; ordinul de marime privind performanta este de peta flops (peta= 10^{15} operatii in virgula mobila).

Exemple de cluster: Beowulf. Cateva din distributiile de linux care implementeaza acest tip de cluster: MOSIX, geared toward computationally intensive, IO-low, applications; ClusterKnoppix, based on Knoppix; Kerrighed.

Caracteristici sisteme cluster

- Sisteme strans cuplate (Tightly coupled systems);
- Creaza impresia unui singur sistem de calcul(Single system image);
- Sistem centralizat de Job management si planificare

c) Grid

Sistemele Grid sunt similare cu sistemele pentru ca fac uz de mai multe calculatoare conectate pentru a rezolva o problemă mare. Exista insa si diferente. Marea diferență este că un cluster este omogen în timp ce rețelele sunt eterogene. Calculatoarele care fac parte dintr-o rețea grid pot rula sisteme de operare diferite și au hardware diferit in timp ce calculatoarele dintr-un cluster au toate au același hardware și sistemul de operare. O rețea grid poate face uz de puterea de calcul de rezervă a unui computer desktop, în timp ce mașinile dintr-un cluster sunt dedicate pentru a lucra ca o singură unitate și nimic altceva. Retelele Grid sunt în mod inerent distribuite într-o rețea LAN, WAN sau metropolitana. Pe de altă parte, calculatoarele din cluster sunt în mod normal conținute într-un singur loc.

O altă diferență constă în modul în care resursele sunt manipulate. În cazul unui cluster, întregul sistem (toate nodurile) se comportă ca un sistem de vizualizare unică și resursele sunt gestionate de către managerul de resurse centralizate. În cazul unui Grid, fiecare nod este autonom adică are managerul de resurse proprii și se comportă ca o entitate independentă.

Avantajul principal al calculului distribuit (grid) este că fiecare nod poate fi achiziționat individual, fara restrictii de configuratie si care, atunci este combinat cu alte calculatoare in grid, poate produce o resursă de calcul similară cu a supercomputerelor, dar la un cost mai mic. Dezavantajul principal este performanta deoarece schimbul de date între procesoarele plasate in diverse zone se poate realiza prin conexiuni care nu au conexiuni de mare viteză. Sistemele Grid sunt, așadar, potrivite pentru aplicații în care calculele multiple paralele pot avea loc independent, fără nevoia de a comunica rezultatele intermediare între procesoare. Pentru aceste aplicatii scalabilitatea rețelelor grid, dispersate geografic, este un avantaj mult mai mare decat viteza mica de conectivitate între elementele grid-ului.

- *Structura de grid identifica o conexiune între mai multe masini in scopul analizei datelor experimentale care nu pot fi analizate eficient intr-o singura locatie;*
- *Un grid este o forma de calcul distribuit in care mai multe calculatoare sunt conectate la distanta prin retele locale sau prin internet; Diferenta fata de cluster este distribuirea pe zone mari geografice si structura eterogena a unitatilor de calcul;*
- *Datorita lipsei de control centralizat asupra hardware-ului modelul de procesare difera fata de cluster prin asignarea de sarcini de lucru foarte mari si prelucrarea rezultatelor dupa un anumit timp.*

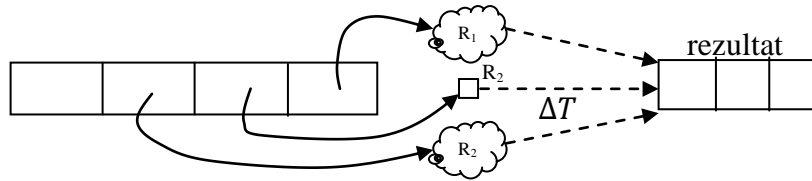
Caracteristici sisteme grid

- Loosely coupled (Decentralization)
- Diversitate si dinamism (datorata faptului ca nu exista un control unic asupra tuturor sistemelor de calcul din grid)
- Job Management si planificare distribuite

Areas of Grid Computing and it's applications for modeling and computing

- 1.Predictive Modeling and Simulations
- 2.Engineering Design and Automation
- 3.Energy Resources Exploration
- 4.Medical, Military and Basic Research
- 5.Visualization

Grid. Imaginea ilustreaza faptul ca sarcinile sunt distribuite între calculatoare eterogene cu puteri de calcul diferite.



$$\Delta T(\text{grid}) \gg \Delta t(\text{cluster})$$

OBS. Acest model de calcul numit peer-to-peer cu mouting este similar cu gridul inasa nodurile componente sunt utilizatori voluntari.

Modulul foloseste arhitectura client-server si se bazeaza pe numarul mare al participantilor. Nivelul de performanta atins este de ordinul Peta flops. Bitcoin network – 168.26 PFLOPS in iulie 2012 si Folding@Home – 5 PFLOPS in 2009, SETI@Home in 2010 - 730 TFLOPS .

Avantajul principal al calculului in grid este costul scazut, deoarece pot fi folosite iferite tipuri de echipamente.

Un termen asociat este CPU scavenging (furtul de cicluri CPU).

De multe ori nu utilizeaza la intreaga capacitate echipamentele disponibile (exp. noaptea). In acest moemnt proceosoarele pot fi „imprumutate” de grid pentru a realiza calcule pana dimineata urmatoare. Imprumutarea se realizeaza fie printr-un modul integrat in SO (WIN HPC) sau aplicatii dedicate (CHAOS).

Tot in anii `60 a aparut termenul de utility computing in care utilizatorii pot accesa printr-o retea de disponibilitate mare resurse situate la distanta (reseaua electrica).

Conceptul de utility computing a fost extins la retelele de internet dand nastere la cloud.

d) Cloud

Aceasta metoda de interconectare se bazeaza pe conceptul software as service (SaS). Conceptul permite inchirierea serviciilor pentru o durata de timp determinata fara resurse hardware dedicate in scopul facturarii mai eficiente.

-amortizare costuri de evolutie hardware prin inchiriere hardware;

-evitare upgrade hardware prin inchiriere putere de calcul

Un alt avantaj este ca serviciile pot fi oferite la costuri reduse sau gratuit, inasa impreuna cu publicitate.

3. Virtualizarea memoriei – permite unui calculator sa utilizeze resursele virtuale dintr-un centru de date drept memorie virtuala; pentru utilizarea eficienta in majoritatea cazurilor calculatorul este un cluster.

Memoria este accesata de SO si/sau aplicatii in plus fata de memoria SO-ului local. Rolul ei poate fi de stocare, transmitere mesaje sau partajare date intre CPU si GPU (graphic processing unit).

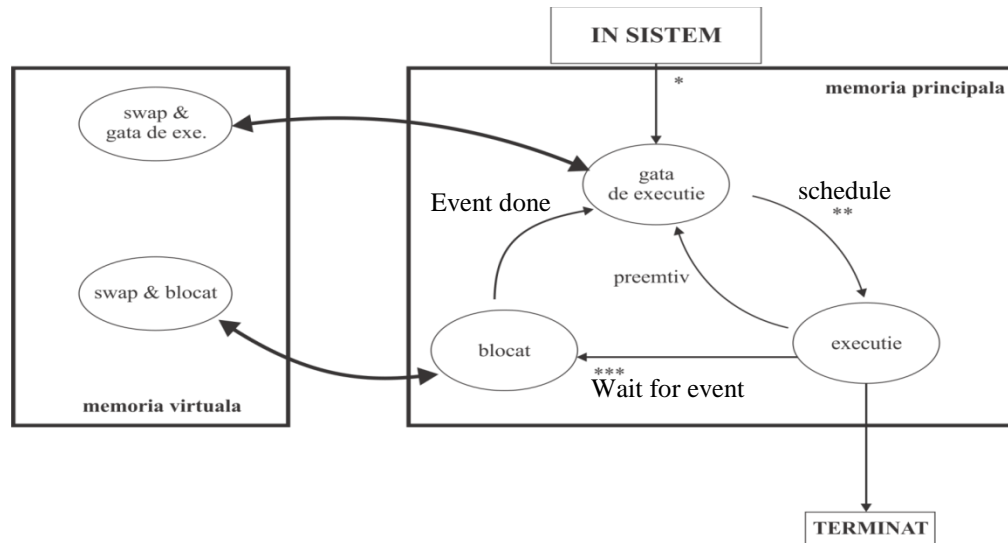
Exista doua forme de implementare la nivel de aplicatie in care accesarea memoriei virtuale se face printr-un API dedicat sau un sistem de fisier si integrarea SO caz in care se creeaza spatii de memorie puse la dispozitia aplicatiilor.

Memoria virtuala:

- Permite programelor rulanad in kernel-uri multi task sa acceseze diferite tipuri de memorii in mod unitar;
- Cea mai vizibila e memoria virtuala a WIN ce poate fi repartizata atat pe hard disk cat si pe discuri flash sau alte medii de stocare;
- Ascunde fragmentarea, realocarea memoriei si adresarea relativa.

Stările procesului: un proces poate sa aiba 3 stări:Gata de executie ;Execuție; Blocat.
 Acestea 3 daca ne aflam in memoria principala si inca 2 stari in memoria virtuala:

- SWAP si gata de execuție
- SWAP si blocat



4. Virtualizarea stocării datelor

Este un concept similar cu virtualizarea memoriei deoarece are rolul de a abstractiza (separa) stocarea locala de resursele fizice. Ea permite extinderea definiției de sistem de stocare de la echipamentul de stocare situat local la cel care folosește (de obicei) rețeaua locala;

Sistemele de stocare permit păstrarea datelor pentru procesare rapida si fiabila.

Caracteristici ale virtualizării stocării:

- Realocarea spațiului de adrese: sistemul de virtualizare poate realoca adresele fizice utilizate de anumite date;
- Folosirea metadatelor: aceste informații folosesc pentru a menține o vedere de ansamblu asupra dispozitivului virtualizat;
- Redirectarea cererilor de intrare-iesire: sistemele de virtualizare redirecționează cererile de intrare-iesire din spatiul virtual in cel real;
- Duplicarea (copierea) spatiului de stocare: aceasta functie permite copierea dispozitivelor logice. Este una din functiile cel mai greu de implementat;
- Managementul spatiului de stocare: asigura crearea, stergerea si modificarea dispozitivului de stocare virtuala. De exemplu, marirea spatiului de stocare poate necesita un nou echipament de stocare.

Problemele apar datorita lipsei compatibilității, realizarea unui back up incorect, complexitatea ridicata si scalabilitatea redusa.

Modalitati de implementare:

- a) Implementarea la nivel de host implica rularea unui soft aditional in majoritatea cazurilor integrat cu SO;
- b) La nivelul de dispozitiv de stocare: **aceasta asigurata in hard in cazul dispozitivelor write.** Aceasta metoda de implementare asigura functii precum clonarea si realizarea back-upului;
- c) La nivelul de retea implica folosirea unui server care gestioneaza o retea de mare viteza numita SAN (Storage Area Network).

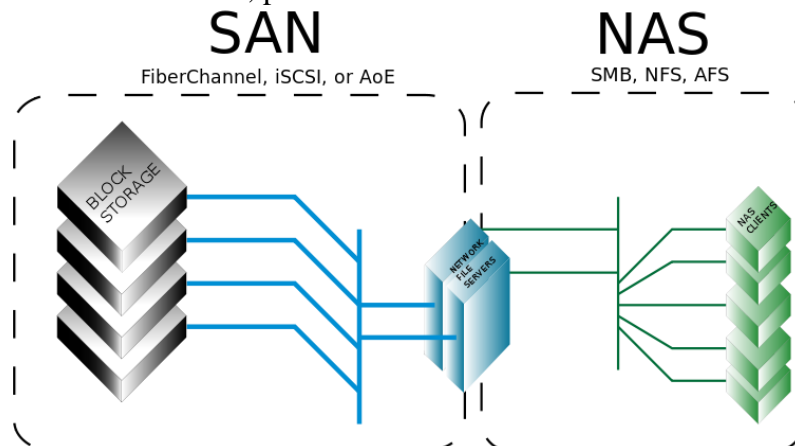
Exista doua moduri de virtualizare:

- o **virtualizarea la nivel de bloc** → se refera la separarea intre spatiul de stocare local si spatiul fizic de stocare astfel incat sa poata fi accesat in mod unitar desi spatiul de stocare fizic poate fi format din mai multe componente eterogene. Protocoalele care folosesc aceasta metoda sunt: Fibre Channel, iSCSI (Internet Small Computer System Interface), SAS (Serial attached SCSI).

SAN(Storage Area Network) asigura stocare la nivel de bloc care lasa problemele legate de organizarea sistemului de fisiere la latitudinea clientului.

Protocoalele SAN includ SCSI, Fibre Channel, iSCSI, ATA over Ethernet (AoE), sau HyperSCSI.

Spatiul disponibil prin reteaua SAN apare pentru sistemul de operare al clientului ca o unitate de stocare, vizibila in utilitarele de gestionare a spatiilor de stocare alaturi de unitatile de disc locale, putand fi formatat si mountat in mod obisnuit.



- o **Virtualizarea fisierelor** → prezenta in special in dispozitivele de stocare NAS (*Network Attached Storage*) ce elimina dependenta intre datele accesate la nivel de fisier si locatia fizica a acestora. NAS apare pentru sistemul de operare al clientului ca un file server care poate fi mapat de client ca un network drive. De obicei este folosit protocolul pentru sisteme distribuite de fisiere NFS (*Network File System*); Performanta acestor sisteme depinde de viteza retelei. Nodul central (cel care gestionează) nu trebuie sa fie foarte performant. Avantajele acestei solutii sunt:

- Folosirea protocoalelor de comunicare pentru schimbul de date;
- Folosirea listelor de acces pentru utilizatori;
- Usureaza sarcini precum copierea si toleranta la erori.

NAS (*Network Attached Storage*)

Este o solutie pentru stocarea datelor cu accesare la nivel de fisier conectate la o retea de calculatoare, asigurand acces la date intr-un grup eterogen de clienti. NAS asigura atat stocarea cat si sistemul de fisiere.

[FreeNAS](#), OpenMediaVault si [Openfiler](#) sunt sisteme de operare pentru solutii NAS.

Sistemele NAS au un consum redus de energie, iar viteza de transfer variaza in functie de implementare intre 40MB/s si 120MB/s.

NAS este un mai degrabă calculator specializat, construit pentru stocarea fișierelor și servire - decât pur și simplu un calculator de uz general fiind utilizate pentru rol. Sistemele NAS sunt interconectate aparatele care conțin unul sau mai multe hard disk-uri, de multe ori aranjate în recipiente de depozitare logice, redundante sau RAID. Unitățile NAS, de obicei, nu au o tastatură sau ecran, și sunt controlate și configurate în rețea, folosind adesea un browser.

Alternative la NAS:

- Direct-Attached Storage (DAS): este o extensie a unui server care nu implica in mod obligatiu rețelele de calculatoare;
- Clustered NAS este un NAS care se utilizează un sistem de fișiere distribuit care rulează simultan pe mai multe servere. Diferența esențială dintre un NAS cluster și tradițional este abilitatea de a distribui datele și metadatele peste nodurile cluster sau dispozitive de stocare. Cluster NAS, cum ar fi una tradițională, încă mai oferă acces unificat la fișierele din oricare din nodurile clusterului, nu au legătură cu localizarea efectivă a datelor.

NFS (*Network File System*);

- este un protocol care permite accesarea fișierelor prin intermediul rețelelor de calculatoare;
- este posibilă accesarea de către mai mulți utilizatori în paralel a unuia sau mai multor fișiere.
- NFS elimină responsabilitatea de distribuire a fișierelor de către alte servere din rețea. NFS oferă de obicei acces la fișiere utilizând protocoale de rețea de partajare de fișiere, cum ar fi NFS (*Network File System*), SMB (*Server Message Block - SO Windows*) sau AFP (*Apple*).

În 2009, vânzătorii NAS (în special la rețelele CTERA și NETGEAR) a început să introducă soluții Online Backup integrate în aparatele lor NAS, pentru recuperare caz de dezastru on-line.

Aplicatii pentru managementul stocarii

- Sistemele pentru managementul stocarii ruleaza de obicei intr-o masina virtuala;
- Sunt portabile deoarece masinile virtuale pot fi instalate pentru orice suport de SO;

- Sistemele de management trebuie sa asigure compatibilitatea intre domenii diferite de stocare si sa extinda facilitatile acestor dispozitive;
- Exista mai multe tipuri de dispozitive de management in functie de pret si performanta dorita;
- Avantajul acestor sisteme soft este posibilitatea update-ului acestora in timp.

Virtualizarea datelor

Virtualizarea datelor privește reprezentarea datelor independent de modul de stocare al acestora. Virtualizarea bazelor de date privește separarea nivelului aplicație de nivelul stocare.

Partitionarea bazelor de date este necesara datorita traficului susținut, generat in retea. Cautari simultane efectuate intr-o singura baza de date duce la creșterea latentei.

Separarea componentelor bazelor de date se poate realiza in doua moduri:

- **Shared all database** – datele sunt repartizate pe calculatoare diferite inasa este permisa comunicarea intre nodurile rețelei pentru a mentine sincronizarea datelor. Aceasta tehnica este folosita in cadrul clusterelor de dimensiuni mici;
- **Shared nothing** – in care sistemele independente de baze de date sunt gestionate in mod individual. Nu toate bazele de date se preteaza pentru o astfel de implementare. Aceste sisteme necesita partajare manuala

Performanta acestor sisteme este mare datorita fragmentarii, iar realocarea in cazul schimbului hard-ului este facila.

Virtualizarea bazelor de date mareste performanta, usureaza administrarea si asigura flexibilitatea sistemului de calcul.

5. Virtualizarea rețelelor.

Network virtualization is also a vital component for application development and testing environments. In the pre-production stages of the software development life cycle, network virtualization is the process of recreating network conditions from the production - or real world - environment within the test environment.

Procesul implica imbinarea resurselor soft si hard intr-o singura retea virtuala de tip software.

Exista doua tipuri de virtualizare:

- Virtualizarea externa: îmbina resurse care sunt partea externa a rețelei de calculatoare.

- Virtualizarea interna: asigura conectivitatea de retea intre containerele interne a unui singur sistem de calcul.

Componentele unei rețele virtuale sunt:

- Hardware de retea

- Switch-uri;
- Parti de retea.
- Echipamente de protectie (firewall sau echipamente load balancing);
- Retele virtuale (V-LAN) → folosite pentru a separa rețeaua la nivel de switch;
- Echipamente mobile;
- Mediul de transmisie.

CURS 10

Sisteme Cluster

Single system image (SSI)

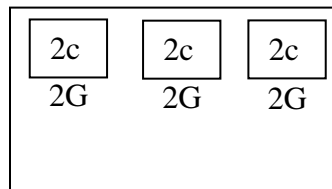
Un cluster este un grup de calculatoare functionand unitar si care se comporta in multe aspecte ca un singur calculator. In majoritatea cazurilor clustererele sunt conectate la retele de calculatoare.

Cele trei tipuri de clusterere sunt:

- High availability cluster (HA) – stau online mereu;
- Load balancing cluster (LB) – rol de a balansa incarcarea echipamentelor;
- Clusterere pentru calcul (pentru procesare).

Clustererele de tipul SSI ruleaza un SO unitar care repartizeaza sarcinile prin intermediul funcțiilor specifice pe diferite noduri.

SSI cluster:
6 core-uri
6G memorie



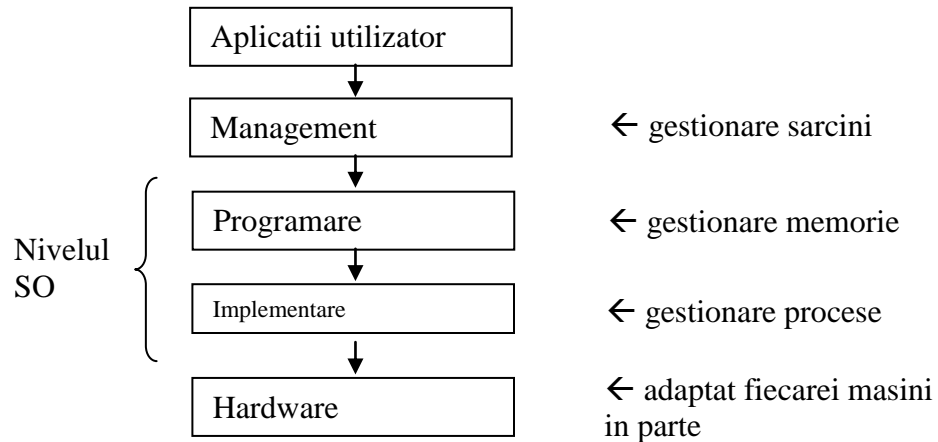
Comportamentul este apropiat de cel al SO distribuite, insa functiile oferite sunt mai restranse fata de un SO distribuit.

Principala activitate gestionata este migrarea proceselor.

Avantajele clusterelor SSI:

- Punct unic de intrare pentru utilizatorii sau procesele care se conecteaza la cluster;
- Interfata utilizator unica (oferind inclusiv interfata grafica);
- Spatiu unic de adrese pentru procese (implica identificatori unici la nivelul clusterului pentru procese simplificand comunicatia intre acestia);
- Spatiu unic de intrare-iesire;
- Sistem unic de fisiere (deobicei NFS);
- Gestionarea unitara;
- Salvarea unitara a starilor intermediare.

Se disting urmatoarele niveluri in arhitectura unui cluster:



1. Nivelul hardware preia resursele fiecarui calculator și le pune la dispoziția SO al clusterului.
 - Transmisia între noduri poate fi broadcast sau unicast;
 - Se asigură mecanisme pentru detectarea erorilor în situația în care anumite stații devin indisponibile.
2. Nivelul SO
 - Scopul este paralelizarea aplicațiilor secvențiale;
 - Rolul este planificarea proceselor, identificarea resurselor libere, migrarea proceselor și balansarea încărcării (load balancing). Accesarea acestor funcții se face fără comenzi suplimentare.
3. Nivelul middleware (management)
 - Se interpune între SO și aplicațiile utilizator cu rol de management al sarcinilor care sunt executate;
 - Planificarea globală a sarcinilor oferă un nivel ridicat de transparență utilizatorului.
4. Nivelul aplicație
 - Este nivelul care rulează aplicațiile utilizator și interfața grafică (GUI) a SO.

Studiu de caz

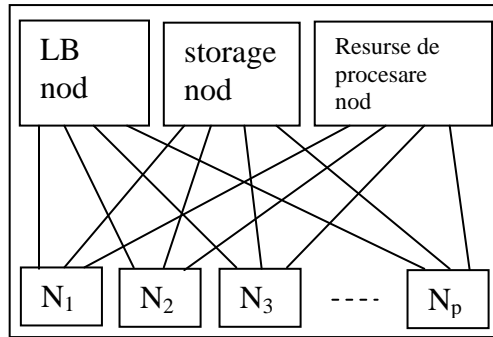
Linux Virtual Server (LVS)

- Are arhitectura transparentă pentru utilizator care ascunde existența calculatoarelor interconectate;
- Selectarea serverelor se realizează prin 4 algoritmi:
 - Round Robin (RR);
 - Weighted Round Robin (WRR);
 - Least Connections (LC);
 - Weighted Least Connections (WLC).

Least connections

- Cele mai puține conexiuni;
- Algoritmul alocă sarcinile în funcție de numărul de conexiuni existente.

Structura acestui cluster SSI este urmatoare:



nodStorage = stocheaza sistemul de fisiere pe cluster.

nod de procesare = implementeaza serviciile oferite de cluster.

Nodul cu resursele de procesare este cel care implementeaza serviciile oferite de cluster. Mod de functionare:

- Nodul LB primeste cererile din internet, selecteaza serverele din lista citita de la nodul cu resurse de procesare;
- Mentine starea conexiunilor concurente / paralele;
- Transmite pachete procesate in interiorul retelei.

SO mentine un serviciu de monitorizare a intregului cluster. Acesta este stocat pe nodul cu resursele de procesare si trimite pachete ping la anumite intervale de timp catre nodurile din retea locala a clusterului. Daca nodul nu raspunde, el este eliminat din lista cu resurse de procesare, dar si din lista nodului LB si a celui de gestionare storage.

Pentru a preveni functionarea incorecta a nodului de LB, de obicei acesta e dublat de un nod de back-up.

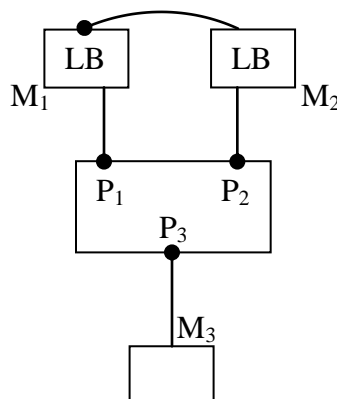
In cazul in care nodul principal executa o operatie eronata, serverul de back-up realizeaza operatia de impersonare ARP (ARP spoofing) pentru a prelua rolul / adresa IP a LB / calculatorului principal.

Cand nodul principal a redevenit functional el contacteaza nodul de back-up si acesta elibereaza adresa IP.

Operatia de preluare a adresei IP duce la terminarea conexiunilor curente, deci clientii vor retransmite cererile catre cluster.

ARP = adress resolution protocol.

Operatia de ARP spoofing:



Operatia ARP spoofing se bazeaza pe proprietatea switchurilor de a lucra doar cu adrese MAC.

Inainte	Dupa
P ₁ -M ₁	P ₂ -M ₁
P ₂ -M ₂	P ₃ -M ₃
P ₃ -M ₃	

Cererile ulterioare ale M₃ vor fi directionate transparent catre portul P₂.

Cateva sisteme de tipul cluster SSI sunt: OPEN SSI, OPEN MOSIX si KERRIGHED.

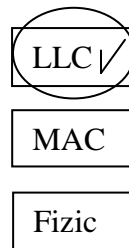
CURS 11

Retele CAN (Controller Area Network)

Aceste retele au fost create la jumatatea anilor '80, in industria automobilelor din Germania pentru a permite comutarii seriale rezistente la erori.

Protocolul folosit de retelele CAN s-a raspandit rapid nu doar in industria automobilelor ci si in alte domenii, de exemplu industria echipamentelor medicale si a aparatelor de testare.

Reteaua CAN functioneaza in straturile de jos ale nivelului OSI: legatura de date si fizic. Din acest motiv, o mare parte a protocolului e dependeta de mediul de transmisie insa exista si o portiune independenta de mediul de transmisie.



Protocoalele de nivel superior vor fi implementate de producatorul software.

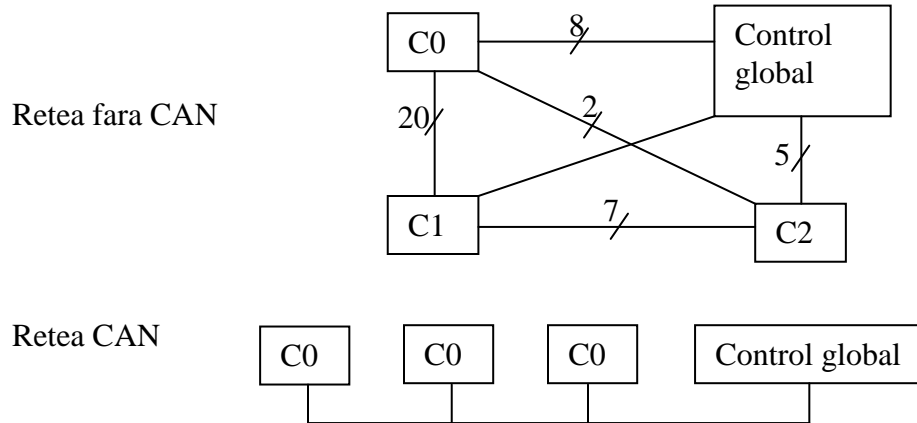
Protocolul CAN a fost optimizat sa transmita cantitati mici de date, spre deosebire de Ethernet sau USB care sunt proiectate pentru a transmite blocuri mari de date.

Protocolul este orientat pe mesaje si nu pe adrese, realizand astfel comunicatii nod-la-nod (peer-to-peer) si multicast fara a modifica tipul mesajului.

Scopul protocolului CAN este realizarea comunicatiei rapide si sigure.

Caracteristicile protocolului CAN

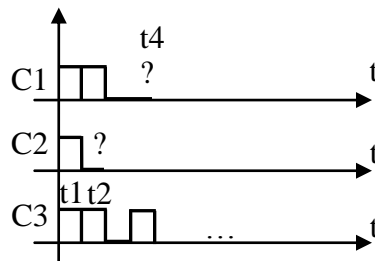
Protocolul foloseste CSMA/CD pentru a depista coliziunile. Acest protocol precizeaza ca fiecare nod din retea trebuie sa monitorizeze mediul de transmisie pentru a depista o perioada de inactivitate inainte de realizarea transmisiei. Daca o perioada de inactivitate este determinata, oricare nod are prioritate egala de transmisie. Daca doua noduri transmit simultan, este detectata coliziunea si se iau masuri corespunzatoare.



Spre deosebire de algoritmi folositi in retelele ethernet, protocolul CAN implementeaza o metoda nedestrutiva privind mesajele transmise la aparitia coliziunii, adica mesajele raman intacte chiar daca coliziunea a fost detectata.

Protocolul CAN foloseste doua tipuri de logica: master si slave, in care semnalul / bitul de 1 este master el avand efect prioritar asupra bitului de 0 care este slave. Daca trei controllere actioneaza simultan mediul de transmisie, lucrurile se intampla in modul urmator: fiecare controller are asociat un identificator. Acesta defineste prioritatea de transmisie a dispozitivului.

controller	cod
C1	1100
C2	1000
C3	1101



La momentul t2, controllerul C2 depisteaza faptul ca semnalul din linie nu corespunde cu semnalul transmis. Aceasta eroare nu este semnalizata deoarece presupune ca un nod prioritar transmite simultan.

La momentul de timp t4, controllerul c1 realizeaza ca un controller prioritar transmite simultan si isi inceteaza transmisia.

C3 fi singurul dintre cele trei care va termina de transmis datele deoarece are proritare maxima.

Comunicare prin mesaje.

Protocolul CAN foloseste mesaje si nu foloseste adrese (Ethernetul este un protocol ce se bazeaza pe adrese) → toate nodurile din sistem primesc mesajul si fiecare

nod decide daca mesajul va fi procesat sau eliminat. Un mesaj poate fi destinat unuia sau mai multor noduri in functie de proiectarea retelei.

Exp. Senzorul de airbag poate transmite un mesaj care va fi receptionat de mai multe controllere.

Protocolul CAN implementeaza o functie numita Remote Transmit Request prin care un nod poate solicita date de raspuns de la un alt nod.

Structura frame-ului CAN

Unitatea de protocol se numeste cadru / frame.

	11 biti	18 biti			max 8 octeti			
S O F	ID	ID2	L	RTR	DATE	CRC 15 biti	A C K	E O F

SOF=start of frame

EOF=end of frame

ID=identificator pe 11 biti determina prioritatea mesajului. In unele situatii se folosesc 18 biti aditionali pentru identificarea prioritatii in functie de proiectarea sistemului.

L=dimensiunea campului de date

RTR=remote transmit request

Ack este folosit pentru confirmarea receptiei; in cazul in care nu se receptioneaza de la destinatie un acknowledge, mesajul este retransmis. In multe sisteme, mecanismul de Ack este dezactivat pentru a reduce numarul de transmisii din sistem.

Detectia erorilor folosind protocolul CAN

Acest protocol foloseste transmisii pe doua fire. Viteza de transmisie a fost marita de la prima versiune 1.0 pana la cea de-a doua versiune 2.0 de 8 ori atingand viteza de 1Mb/sec.

In 1986 cand a aparut primul automobil cu retea CAN, lungimea firelor necesare cablarii masinilor a fost redusa cu 2 km rezultand intr-un castig de 50 kg.

In 2008, peste 90% din masini utilizeaza in interior protocolul CAN.

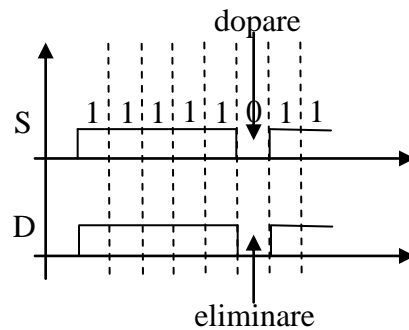
1. Detectie prin CRC.
 - Daca se receptioneaza un frame cu CRC incorect e generata o eroare corespunzatoare;
 - Daca cel putin un nod nu a primit mesajul corect, acesta este retransmis.
2. Eroare de confirmare.

- Daca bitul de Ack contine valoarea 1, mesajul a fost receptionat corect; valoarea 0 indica receptia incorecta.
3. Eroarea de format.
 - Daca oricare nod detecteaza un bit 1 in oricare din campurile de delimitare a frame-urilor (SOF, EOF), o eroare de format este transmisa si mesajul original este retransmis.
 4. Eroarea de bit.
 - Este transmisa in alte situatii decat momentul negocierii transmisiei.
 5. Eroare de dopare.

Deoarece CAN foloseste metoda NRZ (pe durata intregului bit semnalul nu-si modifica starea) pentru transmisie, este posibila aparitia unor secvente lungi de 0 sau 1.

Doparea cu biti inseamna adaugarea unui bit suplimentar dupa un numar bine stabilit de biti cu aceeasi valoare.

S=11111011 (sursa)
 D=1111111 (destinatia il primeste pe 0, dar il sare la compunerea secventei).



Exista si alte erori detectate de protocolul CAN, majoritatea fiind legate de starile in care se afla magistrala de comunicatie.