

---

# Sisteme distribuite – teorie

## 5. Comunicatie in grup

---

---

# Comunicarea in grup

- RPC: doua parti, clientul si serverul
  - Contra-exemplu: un grup de servere de fisiere care coopereaza pentru a oferi un singur sistem de fisiere tolerant la esecuri
    - Clientul expediază un mesaj la toate serverele pentru a se asigura ca cererea este onorata chiar daca unul dintre acestea cedeaza
    - RPC nu poate trata comunicarea de la un expeditor la mai multi receptori (in afara cazului in care efectueaza RPC-uri separate cu fiecare dintre ele)
-

---

# Grup

- = colectie de procese care actioneaza impreuna intr-o modalitate specifica sistemului sau utilizatorului.
  - Scop: permite proceselor sa trateze colectii de procese ca o singura abstractizare -> un proces poate trimite un mesaj la un grup de servere fara a cunoaste cate sunt si unde sunt (caracteristici ce se pot schimba de la un apel la altul)
  - Proprietate cheie: cand un mesaj este expedit la un grup specific, toti membrii grupului il receptioneaza
    - forma de comunicare unul-la mai multi in contrast cu comunicarea punct-la-punct
  - Dinamicitate (analogie cu organizarea sociala !)
    - Grupuri noi pot fi create, grupuri vechi pot fi distruse.
    - Un proces se poate alatura unui grup sau il poate parasi
    - Un proces poate fi membru la mai multe grupuri in acelasi timp-> Sunt necesare mecanisme pentru a administra grupurile si apartenenta la grupuri
-

---

# Implementari ale comunicarii in grup

1. Tehnica multi-casting (difuzare multipla)
    - ❑ Crearea unui adrese speciale de retea (de exemplu indicare prin setarea bitilor celor mai semnificativi pe 1), la care asculta masini multiple
    - ❑ Cand este expediat un pachet la una dintre aceste adrese, este automat livrat la toate masinile care asculta la respectiva adresa
    - ❑ Implementarea grupurilor utilizand multi-casting este imediata: asignarea fiecarui grup la adrese multi-cast diferite
  2. Tehnica broad-casting (difuzare larga)
    - ❑ Pachelete care contin o anumita adresa sunt livrate la toate masinile
    - ❑ Poate fi utilizata pentru a implementa grupurile, dar este mai putin eficienta:
      - Fiecare masina primeste fiecare difuzare si softwareul sau trebuie sa verifice daca pachetul ii este adresat
      - Daca nu, pachetul este ignorat, dar un anumit timp este pierdut pentru a tratarea intreruperii
    - ❑ Necesita doar un pachet care ajunge la toti membrii grupului
-

---

# Implementari ale comunicarii in grup

1. Multi-casting
  2. Broad-casting, daca multicasting nu este permis
  3. Uni-casting (transmitere punct-la-punct),  
daca mc sau bc nu sunt permise
    - ❑ Expedierea unui mesaj de la un singur expeditor la un singur destinatar
    - ❑ Expeditorul transmite pachete separate pentru fiecare membru din grup.
    - ❑ Pentru un grup cu  $n$  membrii, sunt necesare  $n$  pachete, spre deosebire de un pachet cand este utilizat mc sau bc
    - ❑ Desi mai putin eficienta, aceasta implementare este functionala daca grupurile sunt mici
-

---

# Designul comunicărilor în grup

- Drept cazuri de transmitere de mesaje:
    - Cu buffer vs. fără buffer
    - Cu blocare vs. fără blocare
    - Etc
  - Opțiuni adiționale
    - Grupuri închise vs. Grupuri deschise.
    - Grupuri de semeni vs. Grupuri ierarhice
  - Alte probleme
    - Apartenența la grup
    - Adresarea grupului
    - Primitivele Send și Receive
    - Atomicitate
    - Ordonarea mesajelor
    - Grupuri cu suprapunere
    - Scalabilitate
-

---

# Grupuri inchise vs Grupuri deschise

## ■ Grupuri inchise:

- ❑ In care numai membrii grupului pot trimite mesaje la grup.
- ❑ Cei din exteriori nu pot trimite mesaje la grup ca tot, desi pot trimite mesaje la membrii individuali
- ❑ Exemplu: O colectie de procese care lucreaza impreuna pentru un joc de sah pot forma un grup inchis; au propriul scop si nu interactioneaza cu lumea inconjuratoare.

## ■ Grupuri deschise:

- ❑ fiecare proces din sistem poate trimite la oricare grup
  - ❑ Exemplu: grup de servere replicate; este important ca proceserel care nu sunt membre (clientii) sa poate expedia mesaje la grup.
-

---

# Grupuri de semeni vs. Grupuri ierarhice

- Grupuri de semeni (Peer Groups)
    - Toate procesele sunt egale.
    - Nici un proces nu este sef si toate deciziile sunt loate in colectiv
    - Grupul este simetric si nu are un punct singular de esec
    - Daca procesesele esueaza, grupul devine mai mic, dar continua
    - Dezavantaj: procesul de decizie este complicat – pentru a decide orice, este necesar un vot ceea ce duce la intarzieri si surplus in timp
  - Grupuri ierarhice
    - Un proces este cordonator si celelalte sunt lucratori.
    - Cand este generata o cerere de lucru fie de catre un client extern sau de unul dintre lucratori, este trimisa la coordonator care decide care lucrator este cel mai adecvat sa efectueze lucru si i-l expediaza
    - Pierderea coordonatorului duce intregul grup la o oprire, dar atat timp cat functioneaza deciziile se iau fara a deranja pe altii
    - Ex: un grup ierarhic este indicat pentru un program de sah:
      - Coordonatorul considera situatia curenta, genereaza toate miscarile posibile din aceasta si le trimite lucratorilor pentru evaluare
      - Ca urmare a evaluarii noile situatii sunt expediate la coordonator pentru a le evalua
      - Cand un lucrator nu este ocupat, cere coordonatorului de lucru
      - Coordonatorul contrleaza strategia de cautare
-



---

# Apartenența la grup

- Anumite metode sunt necesare
    - pentru a crea și șterge grupuri,
    - pentru a permite proceselor să se alature sau să părăsească grupuri
  - Abordări:
    1. Există un server de grup la care sunt transmise toate cererile
    2. Administrează apartenența la grup într-o modalitate distribuită.
      - Într-un grup deschis, un outsider poate trimite un mesaj la toți membrii grupului pentru a-și anunța prezența
      - Într-un grup închis, este necesar ceva similar
      - Pentru a părăsi grupul, un membru trimite mesaj către fiecare membru
-

---

# Adresarea grupului – abordari

1. Fiecare grup are o adresa unica, precum o adresa de proces.
    - Permis multi-cast-ul: adresa de grup poate fi asociata cu o adresa multi-cast
    - Permis broad-cast-ul: mesajul poate fi difuzat.
    - Numai uni-cast-ul est permis: necesar o lista de masini care au procese ce apartin grupului.
  2. Cere expeditorului sa ofere o lista explicita a tuturor destinatiilor
  3. Fiecare mesaj este expediat la membrii grupului prin una din metodele descrise anterior, dar in plus :
    - Fiecare mesaj contine un predicat (expresie Booleana) de evaluat.
    - Predicatul poate include numarul masinii destinatare, variabilele sale locale si alti factori.
    - Daca predicatul este evaluat prin TRUE, mesajul este acceptat.
    - Daca este evalaut prin FALSE, mesajul este ignorat.
    - Exemplu: expediaza un mesaj numai la acele masini care au cel putin 4M de memorie libera si sunt dispuse sa preia noul proces
-

---

# Primitivele Send si Receive

1. Uneste cele doua forme de comunicare : group & point2point?
    - Send:
      - Parametru -- destinatie
        - O adresa de proces, un singur mesaj este trimis la un proces.
        - O adresa de grup (sau o referinta la o lista de destinatii), un mesaj este trimis la toti membrii grupului
    - Receive:
      - Complet cand soseste un mesaj punct-la-punct sau de grup
  2. Proceduri noi de biblioteca:
    - group-send
    - group-recv
-

---

# Atomicitate

- Dezirabila deoarece face programarea sistemelor distribuite mai usoara.
  - Un proces care expediază un mesaj la un grup nu trebuie sa-si faca griji ce sa faca daca unul dintre membrii nu l-a primit
  - Exemplu – intr-un sistem de baze de date distribuite si cu replicare:
    - Un proces expediază un mesaj la toate masinile care detin baza de date pentru o noua inregistrare in baza de date
      - Are certitudinea ca data a fost scrisa in toate replicile
-

---

# Ordonarea mesajelor

- Ordonarea in timp global:
    - Se asteapta ca toate mesajele sa fie licrate instantaneu si in ordinea in care au fost expediate
    - Toate destinatiile obtin toate mesajele in exact aceiasi ordine
    - Ignora din motive de convenienta faptul ca nu exista un timp global absolut!
  - Ordonarea in timp absolut nu este intotdeauna uso de implementat! Anumite sisteme ofera variatiuni pe aceasta tema. De exemplu:
  - Ordonare in timp consistenta:
    - Daca doua mesaje, fie A si B sunt expediate apropiat in timp, sistemul preia unul dintre acestea ca fiind "primul" si il livreaza la toti membrii grupului urmat de celalalt.
    - Se intampla ca cel ales sa nu fie cel real, dar computarea sistemului nu depinde de acesta
    - Mesajele ajung la toti membrii grupului in aceeasi ordine, dar ordinea nu neaparat este ordinea reala in care au fost expediate.
-

---

# Grupuri cu suprapunere

- Deși există o ordonare în timp în fiecare grup, nu este în mod necesar și o coordonare între grupuri multiple
  - Anumite sisteme suportă ordonarea în timp între grupuri suprapuse, altele nu
  - Implementarea ordonării în timp între grupuri diferite este dificilă
-