
Sisteme distribuite – teorie

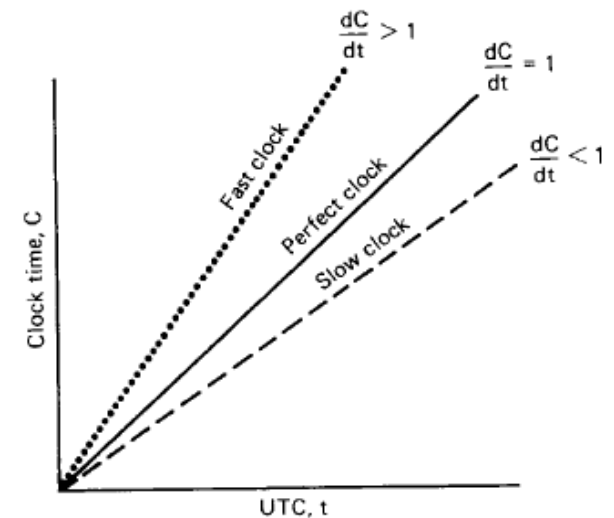
7. Alg. pt. sincronizarea ceasurilor

Scop si presupuneri

- Presup. 1: o masina are un receptor UTC,
 - Scop: mentinerea masinilor din sistem sincronizate
- Presup. 1': masinile au receptoare UTC, fiecare masina tine evidenta propriului timp
 - Scop: mentinerea ceasurilor individuale cat mai apropiate
- Presup. 2: Fiecare masina se presupune ca fiecare masina are un timer care cauzeaza H intreruperi pe secunda
 - Cand acest timer ajunge la zero, handlerul de intrerupere adauga 1 la *ceasul software* care mentine evidenta numarului de intreruperi de la momentul unei sincronizari anterioare
 - Aceasta valoare a ceasului este notata C .
 - Cand valoare timpului UTC este t , valoarea ceasului masinii p : $C_p(t)$.
 - Intr-o lume perfecta, ar avea loc egalitatea $C_p(t) = t$ pentru toate p si toti t . Cu alte cuvinte, dC/dt ideal ar trebui sa fie 1.

Rata de derivatie

- Timerele reale nu reusesc sa realizeze intreruperile exact de H ori pe secunda.
 - Theoretic, un timer cu $H = 60$ ar trebui sa genereze 216 000 ticuri pe ora.
 - In practica, eroarea relativa obtinuta de cipurile timer curente poate fi in domeniul 215 998 to 216 002 ticuri pe ora.
- Daca exista o anume constanta astfel incat $1-r \leq dC/dt < 1+r$ se spune ca timerul lucreaza conform specificatiei.
- Constanta r este specificata de producator si este cunoscuta drept rata maxima de deviere.

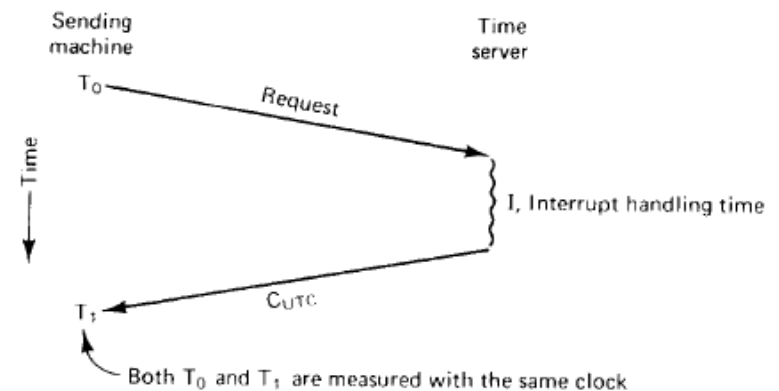


Sincronizare

- Daca doua ceasuri au deviat de la timpul UTC in directii opuse, la un timp dt dupa ce au fost sincronizate, pot fi departate cu $2 r dt$.
- Daca designerii sistemului distribuit doresc sa garanteze ca oricare doua ceasuri nu se indeparteaza mai mult de d , ceasurile trebuie resincronizate (prin software) la cel putin fiecare $d/2r$ secunde.
- Algoritmii difera in precizia cu care este realizata sesincronizarea.

Algoritmul lui Cristian – principiu

- Presupunete: SD in care o masina are un receiver UTC
 - Numim masina care are acest receptor *server de timp*.
- Scopul este acela de a sincroniza masinile cu acesta.
- Periodic, nu mai tarziu de fiecare $d/2r$ secunde, fiecare masina expediază un mesaj cerandu-l timpul curent.
- Serverul de timp raspunde cat de repede poate cu un mesaj care contine timpul sau curent C_{UTC}
- Cand expeditorul gets the reply, it can just set its clock to C_{UTC} .



Algoritmul lui Cristian – probleme

- Algoritmul are doua probleme:
 - Problema majora: timpul nu trebuie sa fie dat inapoi
 - Daca ceasul expeditorului este rapid, va fi setat la o valoare mai mica decat valoarea curenta a expeditorului.
 - Reaintere consecinte: un cod obiect compilat dupa schimbarea ceasului va avea un timp inaintea celui a sursei care a fost modificata inainte de schimbarea ceasului.
 - Problema minora: este necesar un timp pentru ca raspunsul serverului sa ajunga la expeditor
 - Aceasta intarziere poate fi destul de mare si sa varieze functie de incarcarea retelei

Tratarea problemei majore

- Schimbarea trebuie introduse gradual.
- O modalitate este urmatoarea:
 - Presupunem ca timerul este setat pentru a genera 100 intreruperi pe secunda.
 - Normal, fiecare intrerupere va aduaga 10 msec la timp.
 - Cand se doreste incetinirea rutina de intrerupere aduaga numai 9 msec de fiecare data, pana cand a fost facuta corectia
 - Similar, ceasul poate fi avansat gradual prin adugarea a 11 msec la fiecare intrerupere in loc de un salt dintr-o data.

Tratarea problemei minore

- Modalitatea lui Cristian de tratare: incercare de a masura intarzierea retelei
- Expeditorul inregistreaza cu acuratete intervalul intre expedierea cererii de timp catre server si sosirea raspunsului
 - Timpul de start, T_0 si timpul final, T_1 , sunt masurate utilizand acelasi ceas,
 - Intervalul va avea o acuratete relativa, chiar daca ceasul expeditorului este deplasat de la timpul UTC cu o anumite cantitate
- In absenta unei alte estimari mai bune a timpului de propagare a mesajului, acesta este $(T_1 - T_0)/2$.
- Cand soseste raspunsul, valoarea din mesaj poate fi incrementata cu aceasta cantitate pentru a oferi o estimare a timpului curent a serverului
- Daca timpul de propagare minim si teoretic este cunoscut, o estimare a timpului este mai usor de calculat

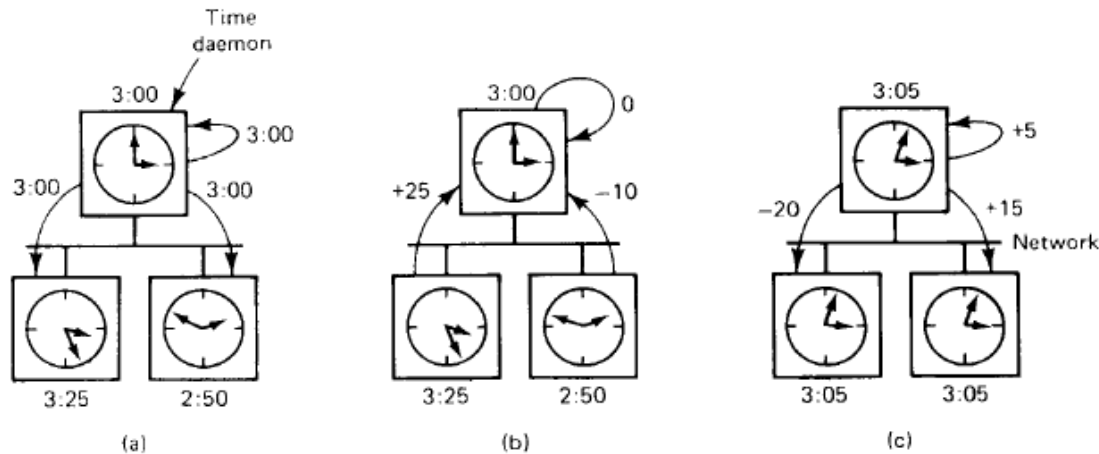
Imbunatatirea estimarii timpului

- Estimarea poate fi imbunatatita daca este cunoscut cat timp este necesar serverului de timp sa trateze o intrerupere si sa proceseze mesajul sosit
- Fie timpul de tratare a intreruperii I .
- O masura a intervalului de timp dintre T_0 si T_1 care este dedicat propogarii mesajului este $T_1 - T_0 - I$ astfel ca o estimare buna este a timpului de propagare este jumatare din aceasta
- Dar ... Exista sisteme in care mesajele de la A la B in mod sistematic au o ruta diferita de mesajele de la B la A, si astfel au timpi de propagare diferiti.
- Pentru a imbunatati acuratetea estimarii, Cristian a sugerat sa nu se faca numai o masuratoare, ci o sereie de masuratori.
 - Fiecare masuratoare in care $T_1 - T_0$ depaseste o anumita valoare de prag este ignorata ca fiind victima congestiilor de retea si de aceea de ne-incredere
 - Estimările derivate din probele ramase pot fi folosite la o mediere pentru a obtine o valoare mai buna
 - Alternativ, mesajul care vine inapoi mai repede poate fi considerat cel mai bun deoarece a intalnit un trafic redus si este cel mai reprezentativ pentru timpul de propagare pur

Algoritmul Berkeley (1 / 2)

- In algoritmul lui Cristian: timpul serverului este pasiv.
 - Fiecare masina il solicita periodic
 - Tot ceea ce face este sa raspunda la solicitari
- Algoritmul Berkeley: considera o abordare opusa
 - Serverul de timp este activ (de fapt este un proces demon), intreband fiecare masina periodic ce valoare a timpului cunoaste
 - Bazandu-se pe raspunsuri, calculeaza un timp mediu si spune celorlalte masini sa avanseze sau incetineasca ceasurile lor la timpul nou pana cand a fost realizata o anumita reducere a diferentei.
 - Aceasta metoda este adecvata pentru un sistem in care nici o masina nu are un receptor UTC.
 - Timpul demonului trebuie setat manual de catre operator periodic.

Algoritmul Berkeley (2/2)



- a) La 3:00, demonul de imp spune celorlalte masini timpul sau si le cere celorlalte.
- b) Acestea raspund cat de departe sunt fata de timpul demonului
- c) Inarmat cu aceste numere, demonul de timp calculeaza media si spune fiecarei masini cum sa ajusteze ceasul sau

Algoritmi de mediere

- Ambele metode descrise mai sus sunt centralizate, cu avantajele uzuale
- Algoritmi descentralizati sunt cunoscuti!
- O clasa de algoritmi de sincronizare descentralizata a ceasurilor lucreaza prin divizarea timpului in intervale de resincronizare de lungime fixa
 - Intervalul i lea porneste la T_0+iR si ruleaza pana la $T_0+(i+1)R$ unde T_0 este un moment agreat in trecut, iar R este un parametru de sistem.
 - La inceputul fiecarui interval, fiecare masina difuzeaza timpul curent conform ceasului sau
 - Deoarece ceasurile de pe masini diferite nu ruleaza la exact aceeasi viteza, aceste difuzari nu se intampla exact simultan
 - Dupa ce o masina a difuzat timpul sau, porneste un timer local pentru a colecta toate celelalte difuzari care sosesc intr-un anumit interval S .
 - Cand toate difuzarile au sosit este rulat un algoritm care calculeaza noul timp
 1. Cel mai simplu algoritm face media valorilor de la toate celelalte masini
 2. O variatie pe aceasta tema este aceea de a inlura inainte de m cele mai mari si m cele mai mici valori si a face media restului
 - Eliminarea valorilor extreme poate fi privita ca un o aparare contra a m ceasuri eronate care trimit valori fara sens
 3. O alta variatie este aceea de a incerca corectarea fiecarui mesaj prin adugarea la acesta a unui estimari a timpului de propagare de la sursa.
 - Aceasta estimare poate fi realizata cunoscand topologia retelri sau timpul necesar unei probe mesaj care este receptionata in ecou

Surse multiple de timp extern

- Pentru sistemele in care sincronizarea cu acuratee extrema cu UTC este necesara, este posibila echiparea sistemului cu receptori multipli
- Datorita inacuratetei inerente in sursa de timp in sine + fluctuatiile in drumul semnalului ceea ce poate un SD sa realizeze este sa stabileasca un domeniu (interval de timp) in care se afla UTC.
- In general, sursele variate de timp vor produce diferite domenii ceea ce implica faptul ca masinile atasate la acestea trebuie sa ajunga la un acord.
- Pentru a ajunge la acest acord, fiecare procesor cu o sursa UTC poate difuza domeniul sau periodic, de exemplu la inceputul fiecarui minut UTC
- Nici un procesor nu va primi aceasta informatie instantaneu
- Mai mult, intarzierea intre expediere si receptionare depinde de distanta pe cablu si numarul de gateways pe care pachetele trebuie sa le traverseze, ceea ce este diferit pentru fiecare pereche (sursa UTC, procesor).
- Fiecare factor joaca de asemenea un rol, precum intarzierile datorate coliziunilor cand masini multiple incearca sa transmita pe retea in acelasi timp
- Daca un procesor este ocupat cu tratarea unui pachet anterior, e posibil sa nu urmareasca un pachet de timp pentru un numar considerabil de miliseunde, introducand o incertitudine aditionala la timp

Utilizarea ceasurilor sincronizate

- Tichete de timp in autentificare
- Tratarea commit in tranzactiile atomice
- Consistenta cache bazata pe ceas
- Livrarea o singura data a mesajelor