

Kaksivaiheinen lukituskäytäntö (2PL, two-phase locking)

Eristyneisyysanomaliat ovat mahdollisia, koska samaan tietoalkioon operointia ei rajoiteta tarpeeksi. Edellisessä esimerkissä (s. 45) kaikki lukot ovat (hyvin) lyhytaikaisia.

Kaksivaiheinen lukitus:

- mitään lukkoa ei vapauteta ennenkuin kaikki transaktion tarvitsemat lukot on varattu
- transaktio jakaantuu kahteen vaiheeseen:
 - kasvuvaihe, jonka aikana kaikki lukot varataan,
 - kutistumisvaihe, jonka aikana lukot vapautetaan.

Lukulukon korotus kirjoituslukoksi tulkitaan lukon varaukseksi eli on tehtävä kasvuvaiheen aikana. (vastaavasti kirjoituslukon alennus lukulukoksi kutistumisvaiheessa)

Kaksivaiheinen lukitus voi olla perusmuodon lisäksi mm.

- ankara (strict): kaikki lukot vapautetaan vasta transaktion sitoutuessa (tai peruuntuessa)
- konservatiivinen: kaikki lukot varataan transaktion alussa
 - tulee tietää tarvittavien tietoalkioiden joukot: read-set ja write-set

Esim. seuraavat transaktiot toteuttavat 2PL-ehdon:

T1'	T2'
read_lock(Y);	read_lock(X);
read_item(Y);	read_item(X);
write_lock(X);	write_lock(Y);
unlock(Y);	unlock(X);
read_item(X);	read_item(Y);
X:= X + Y;	Y:= X + Y;
write_item(X);	write_item(Y);
commit;	commit;
unlock(X);	unlock(Y);

- lukkiutuma (deadlock) on mahdollinen!
 - T1' varaa Y:n kirjoituslukon, T2' X:n
 - erilaisia menetelmiä lukkiutuman hoitamiseen
- esimerkissä lukulukon vapautus ei päästä toista transaktiota eteenpäin eli toiminta vastaa ankaraa 2PL-käytäntöä (käytännössä tässä jompikumpi sarjallinen suoritus)

(Huom. E&N esittelee useampia vaihtoehtoja:

- conservative: kuten edellä
- rigorous: kuten strict edellä
- strict: kirjoituslukot pidetään sitoutumiseen asti)

Konservatiivinen menetelmä:

- voi olla liian varovainen: transaktion vaikea päästä alkuun
- luku- ja kirjoitusjoukkojen määräytyminen etukäteen hankalaa
- + transaktiot eivät voi lukkiutua

Ankara kaksivaiheinen lukitus on käytännössä yleisin. Se takaa transaktion sarjallisuuden, jos kaikki transaktiot noudattavat samaa käytäntöä.

Käytännön merkitys: ei tarvitse tutkia erikseen ajoituksen sarjallisuutta (read/write-suhteiden verkko; verkon syklittömyys), lukkojen varaus- ja vapautusperiaate riittää.

Transaktioista voidaan muodostaa verkko, jonka solmuina ovat transaktiot ja särminä transaktioita yhdistävät konfliktointien operaatioiden (r/w, w/r, w/w) parit (sarjallisuusverkko). Jos verkossa on sykli, vastaava ajoitus ei ole sarjallistuva, muuten on.

Kuinka ankara 2PL estää eristyneisyysanomalia?

1° likainen kirjoitus

Esim. T1: write_lock(X);
 write_item(X, u);
 T2: write_lock(X);
 T2: write_item(X, v);
 T1: commit; (tai rollback);
 T1: unlock(X);
 ...

Ajoitus ei ole mahdollinen, koska T2 ei voi saada X:n kirjoituslukkoa eikä siten tehdä likaista kirjoitusta. Olennaista on, että T1:n kirjoituslukko on pitkäaikainen; yleisesti kirjoittaja ottaa kirjoituslukon ja kaikilla muilla on pitkäaikaiset kirjoituslukot.

2° likainen luku

Esim. T1: write_lock(X);
 T1: write_item(X, u);
 T2: read_lock(X);
 T2: read_item(X, v);
 T1: commit; (tai rollback);
 T1: unlock(X);
 ...

T2 ei voi saada edes lukulukkoa, kun X:llä on pitkäaikainen kirjoituslukko.

Huom. Lyhytaikainenkin lukulukko riittää estämään likaisen luvun, kun muilla on pitkäaikaiset kirjoituslukot.

3° toistokelvoton luku

```
Esim.  T1: read_lock(X);
        T1: read_item(X, u);
           T2: write_lock(X);
           T2: write_item(X, v);
        T1: commit;      (tai: rollback;)
        T1: unlock(X);
        ...
```

Toistokelvoton luku estyy: T2 ei voi saada kirjoituslukkoa X:ään eli ei voi muuttaa X:n arvoa, kun T1:llä on pitkäaikainen lukulukko. (Lyhytaikainen lukulukko ei riitä.)

On luontevaa ajatella, että sama lukituskäytäntö koskee kaikkia transaktioita. Transaktiokohtaisia muutoksia voidaan kuitenkin tehdä (eristyneisyystason asetus lauseella set transaction ...).

On helppo nähdä, että ankara 2PL rajoittaa usein 'liian paljon' samanaikaisuutta: tietyn tietoalkion lukko voitaisiin vapauttaa heti, kun siihen kohdistuvat operaatiot on tehty. Tätä 'hintaa' pidetään järkevänä verrattuna kunkin ajoituksen sarjallisuuden selvittämiseen erikseen.

Tapauksessa 1 käytetään siis rajoitavampia lukkoja, mutta lyhytaikaisesti; tapauksessa 2 yleensä jaettu lukulukkoja pitempään.

Hakemistojen lukituksella voidaan vaikuttaa myös ns. haamutietueiden havaitsemiseen: tiheän hakemiston tapauksessa hakemistosivun lukitus estää päivittävää ja lukevaa transaktiota pääsemästä samanaikaisesti tietosivulle

```
esim. T: insert into employee values (... , dno=5)
        T': select sum(salary) ... where dno=5
```

- jompikumpi varaa hakemistosivun lukon eli rivin lisäys ei tapahdu kesken laskennan (harva hakemisto: samoin, mutta estävä vaikutus ulottuu laajemmalle)

Lukitus ja hakemistot

Kaksivaiheinen lukitus ei sovellu hyvin hakemistojen käsittelyyn:

- hierarkkisen hakemiston käsittely alkaa juuresta
 - alempien tasojen ja tietosivujen käsittelyssä tarvitaan lukkoja myöhemmin, jolloin ei enää välttämättä tarvita ylempien tasojen sivuja
- lukkojen vapautus vasta kutistusvaiheessa rajoittaa huomattavasti samanaikaisuutta

Useimmiten isäsivun lukko voitaisiin vapauttaa, kun lapsisivuun on saatu lukulukko. Jos tulee tarvetta päivittää hakemistosivua, varataan päivitystä varten lukko uudelleen.

Esim. B+ -puu:

- 1) konservatiivinen tapa:
 - varataan tasoitain kirjoituslukkoja; vapautetaan ne, kun on saatu lukko seuraavalle tasolle ja todettu, ettei tarvitse palata (sivulla on tilaa uudelle alkioille)
- 2) optimistinen tapa:
 - varataan lukulukkoja tasojen alas edettäessä; jos lehtitaso jaetaan ja ylempää tasoa pitää päivittää, korotetaan lukulukkoja kirjoituslukeiksi (tarpeen mukaan)

Lukkiutum

• kaksi tai useampia transaktioita yrittää lukita jotakin toisen transaktion käytössä olevaa tietoalkiota: transaktiot ovat siis lukitusalgoritmeja suorittaessaan odottavien transaktioiden jonossa

Esim. T1', T2' sivulla 51

Mitä voidaan tehdä?

- 1) lukkiutuman estävä käytäntö
 - konservatiivinen 2PL: kaikki lukot alussa
 - varataan paljon lukkoja turhan aikaisin
 - samanaikaisuus vähäistä
 - vaikea tietää, mitä tietoalkioita tarvitaan
- 2) tietoalkioiden järjestykseen perustuva varaaminen
 - estää ristikkäiset varaukset
 - järjestys, sen ylläpito?
- 3) no wait -periaate: transaktio ei odota koskaan, vaan käynnistyy myöhemmin uudelleen
- 4) aikaleimoihin perustuva lukkiutuman ratkaisu:
 - esim. nuorempi peruutetaan (aloitetaan myöhemmin uudelleen samalla aikaleimalla)
 - (wait - die')

- 5) lukkiutumien havaitseminen ja purku: odotusverkko
- odotusverkon ylläpito: verkon sykli merkitsee lukkiutumien syntyä
 - purku: peruutetaan jokin transaktio ja aloitetaan se myöhemmin uudelleen

Transaktioiden odotus lukinnassa tai lukkiutumia purettaessa voi johtaa nälkiintymiseen (starvation):

- lukkiutumien odotus: vaikka ei synny lukkiutumia, jokin transaktio häviää aina kilpailun lukitusvuorosta
 - ratkaisuja: FIFO, prioriteetin kasvatus odottaessa
- lukkiutumien purku: transaktio ei saa vuoroa uudelleenaloituksissa (vrt. wait-die: alkuperäisen aloitajan säilyminen)

3. Oletetaan, että tietokannan hallintajärjestelmässä on eristyneisyystason oletusarvona 'read uncommitted' eli vain likaiset kirjoitukset on kielletty. Kaikkien työntekijöiden palkkasumma lasketaan transaktiolla T1, jonka eristyneisyystasoksi määritellään a) repeatable read, tai b) read committed. Siis esimerkiksi (a)

```
set transaction isolation level 'repeatable read';
begin transaction
  select sum(salary) from employee;
commit transaction;
```

Oletetaan, että samanaikaisesti T1:n kanssa on suorituksessa transaktio T2, joka päivittää kahden työntekijän palkkaa.

Mitä lukkoja transaktiossa varataan ja milloin ne vapautetaan?

Millä tavalla T1:n tulos tapauksessa (a) voi poiketa siitä, mikä se on tapauksessa (b)?

Harjoitustehtäviä (käsitellään torstain luennolla):

1. Käytetään ankaraa kaksivaiheista lukitusta. Transaktio T korottaa kaikkien osaston 5 työntekijöiden palkkaa 500 markalla. Selvitä lukkojen käyttö T:n suorituksessa, kun a) relaatioon EMPLOYEE ei ole hakemistoa dno:lla, b) dno:lle on hajautusrakenteinen oheishakemisto.

2. Transaktio T' korottaa kaikkien Smith-nimisten työntekijöiden palkkaa 500 markalla. Selvitä lukkojen käyttö, kun käytössä on sukunimen mukainen ISAM-hakemisto.