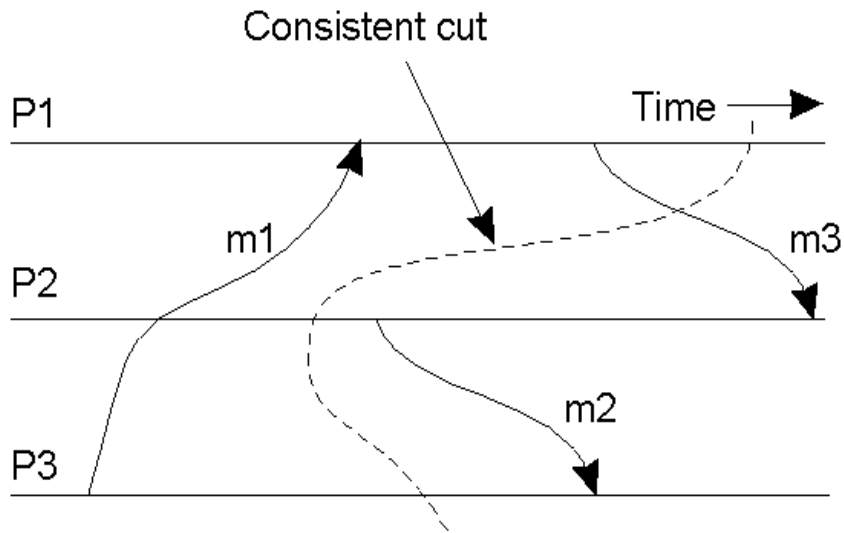


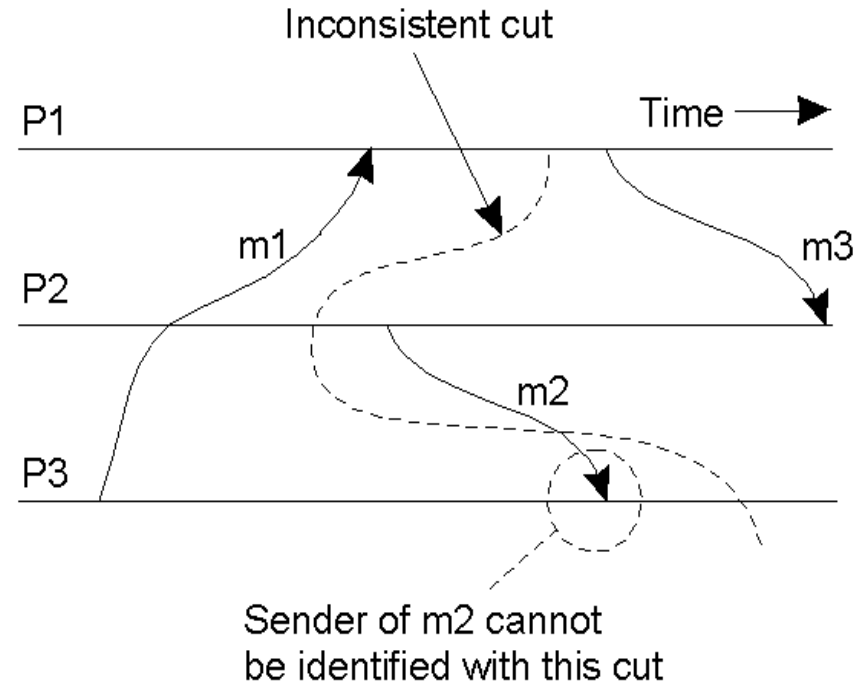
Szinkronizálás

2. rész

Globális állapot (1)



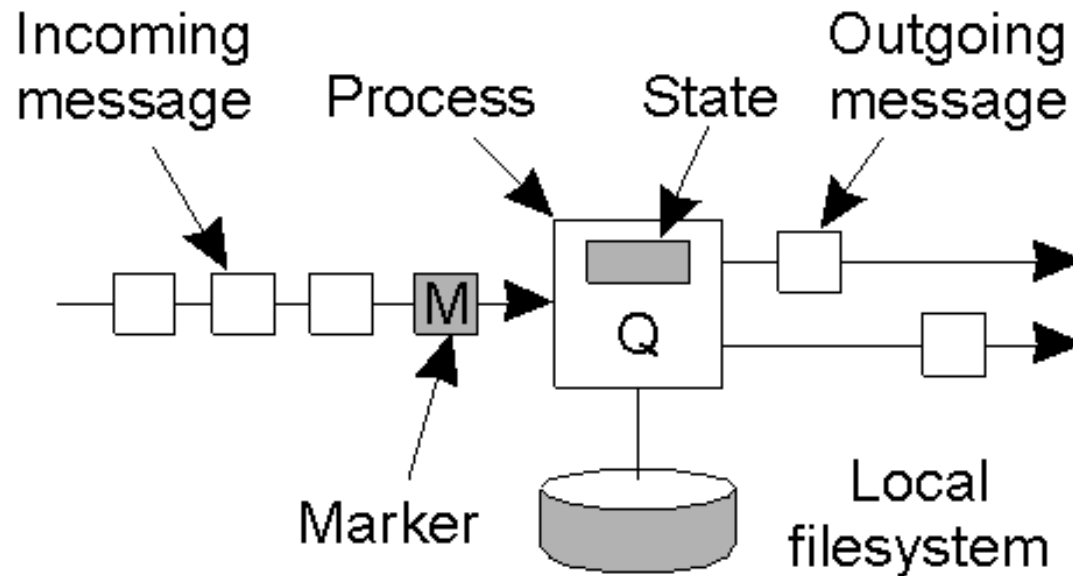
(a)



(b)

- a) Konzisztens metszet
- b) Inkonzisztens metszet

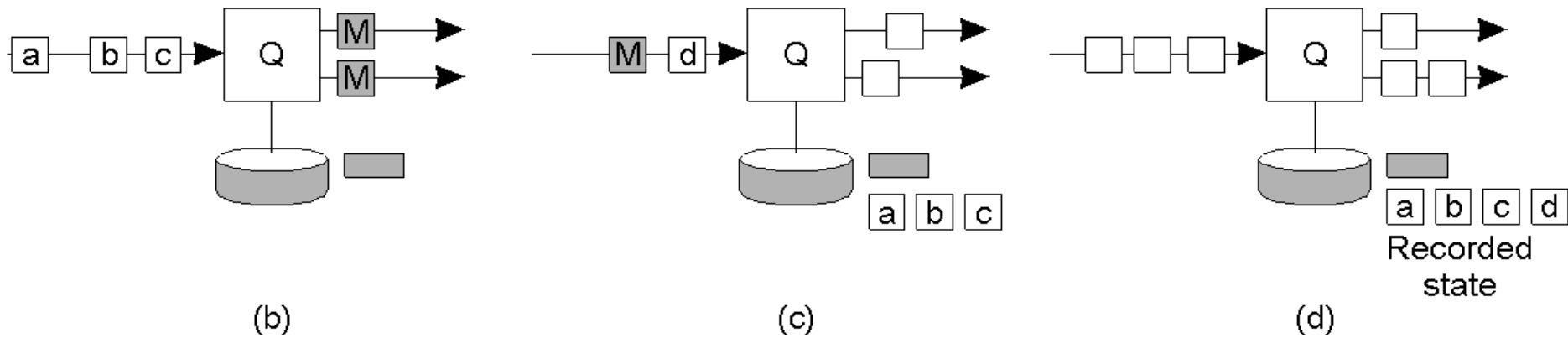
Globális állapot (2)



(a)

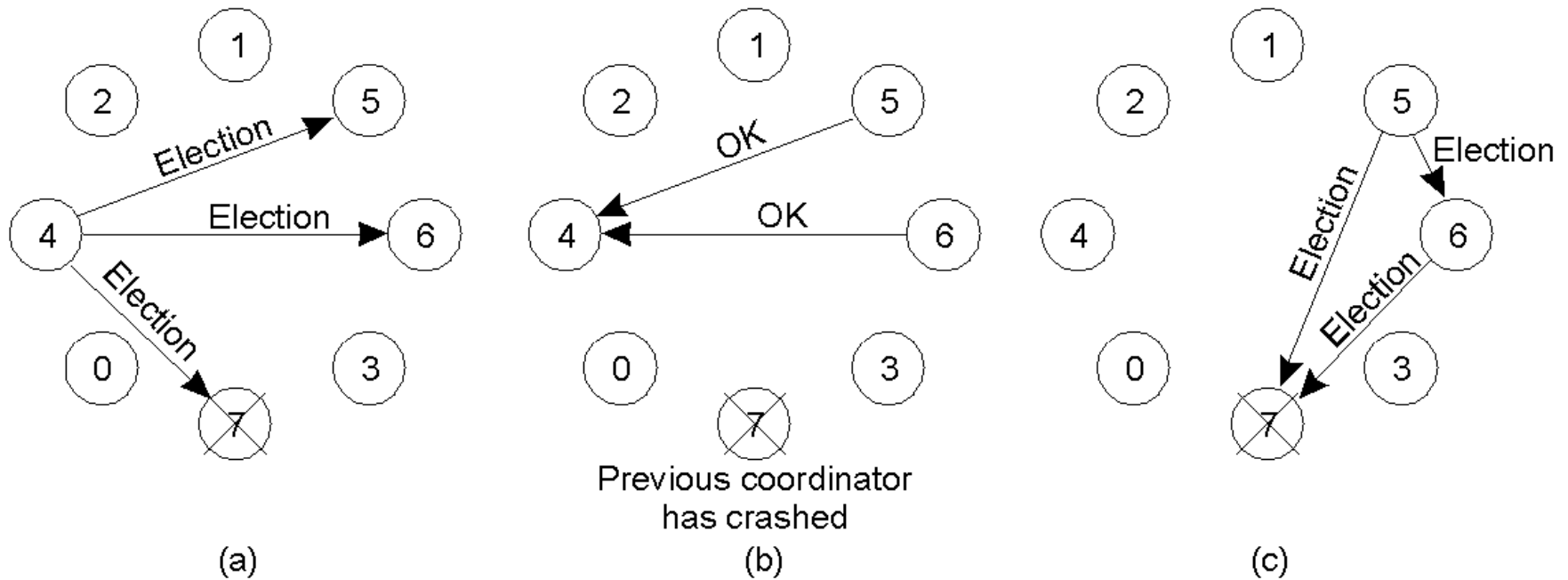
- a) A folyamat és csatornáinak szerkezete az elosztott pillanatfelvétel elkészítéséhez.

Globális állapot (3)



- b)* Q folyamat a jelzést először kapja meg, ezért feljegyzi saját állapotát.
- c)* Q valamennyi érkező üzenetét feljegyzi.
- d)* Q megkapja a jelzést a bejövő csatornán és befejezi a csatorna állapotának rögzítését.

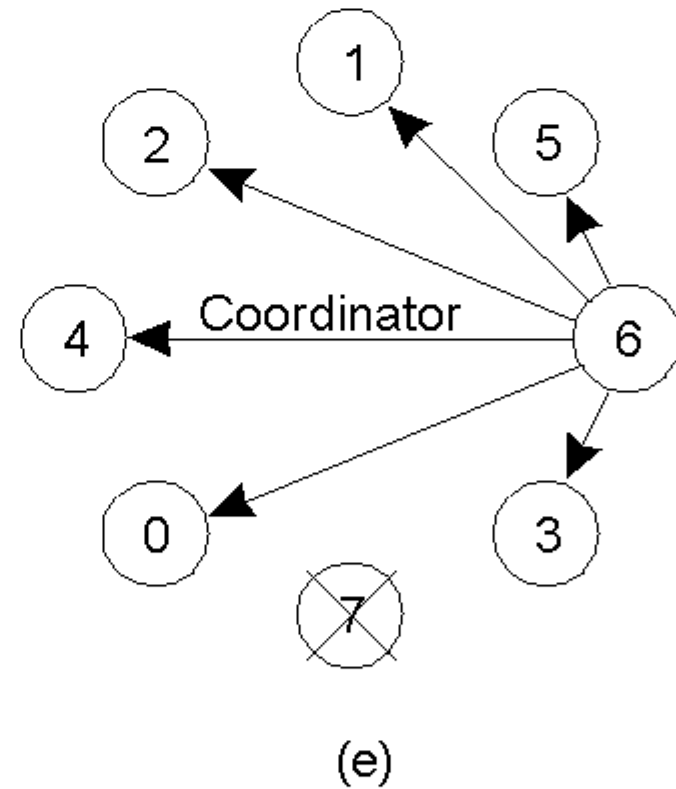
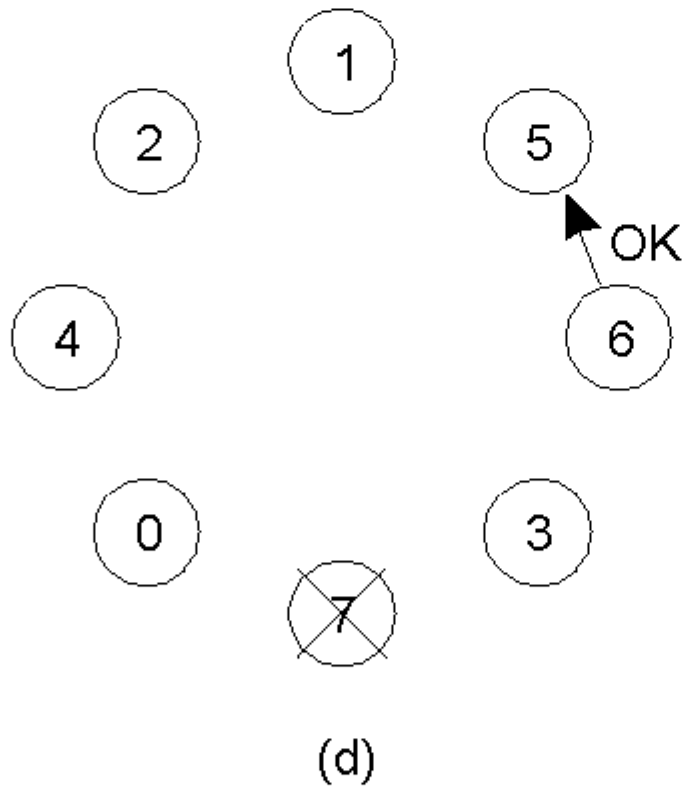
Szavazó algoritmusok - zsarnok (1)



A zsarnok algoritmus

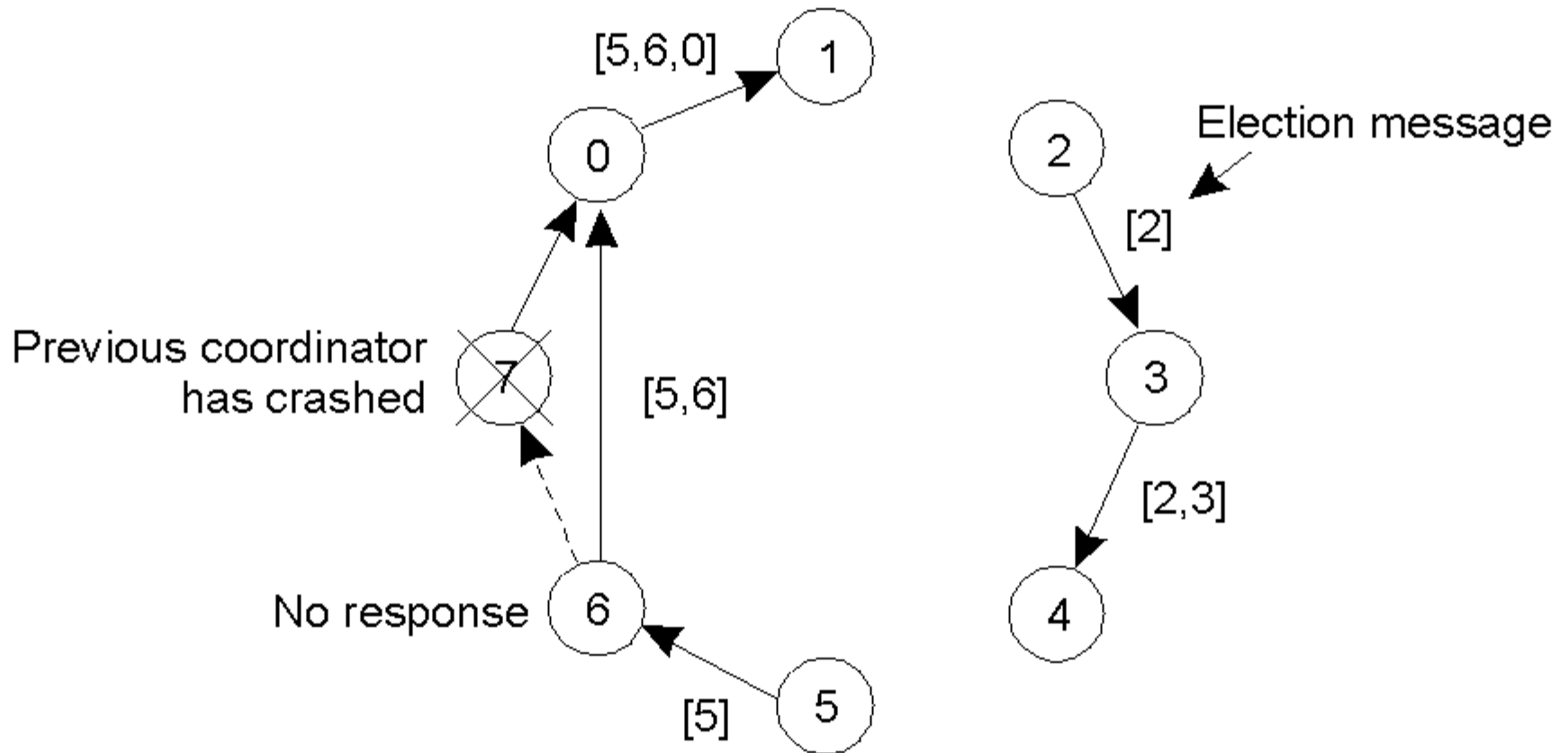
- A 4-es folyamat választást tart
- 5-ös és 6-os válaszol, megmondva 4-esnek, hogy álljon le a választással
- 5-ös és 6-os választást kezdeményez

Szavazó algoritmusok - zsarnok (2)



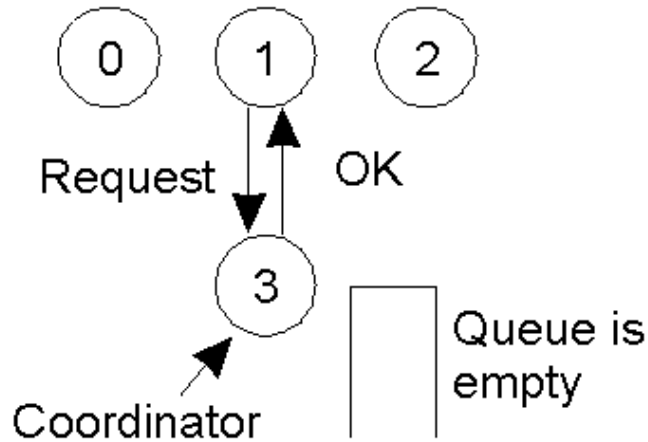
- d) 6-os szól 5-ösnek, hogy álljon le a választással
- e) 6-os megnyeri a választást és kihirdeti a győzelmét

Szavazó algoritmusok - gyűrű

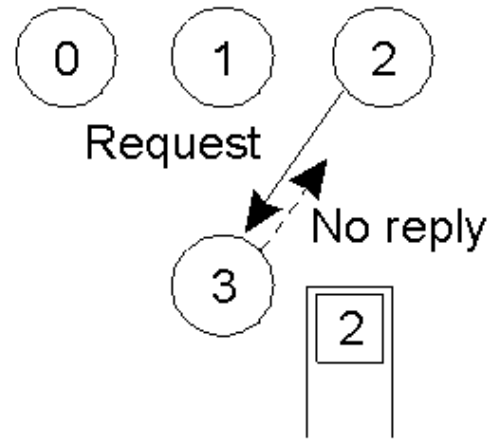


Kört használó szavazó algoritmus.

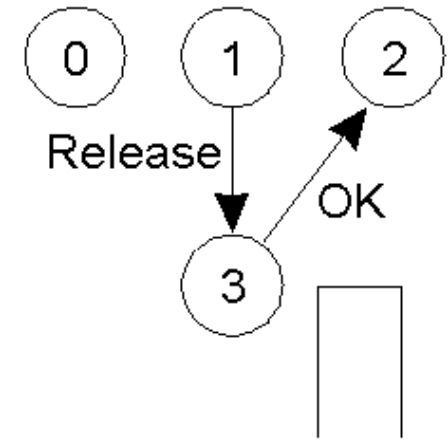
Kölcsönös kizárás: A központosított algoritmus



(a)



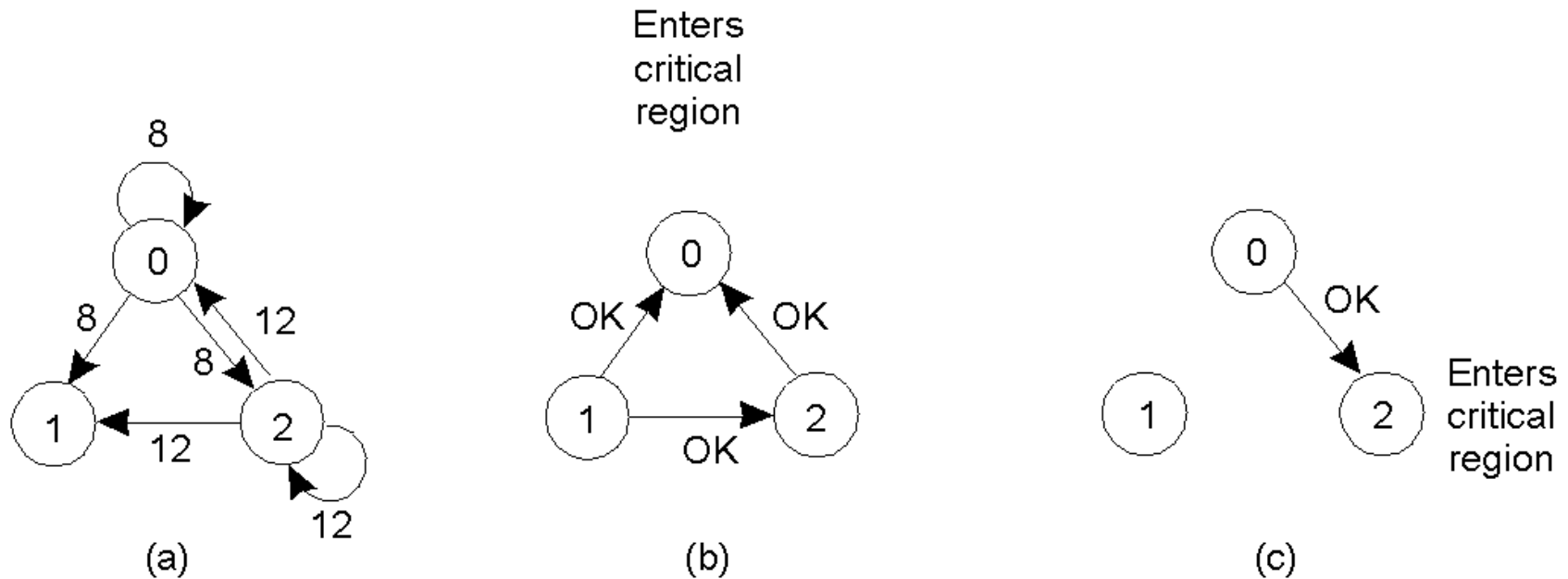
(b)



(c)

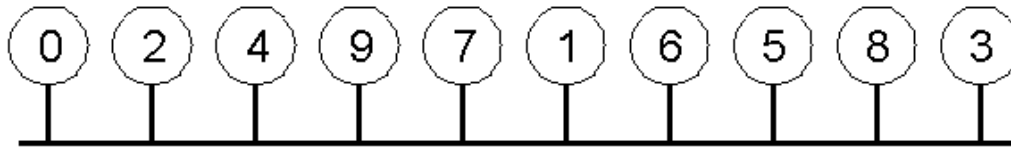
- a) Az 1-es folyamat engedélyt kér a koordinátortól, hogy beléphessen a kritikus területre. Az engedély megadva.
- b) A 2-es folyamat engedélyt kér ugyanerre a kritikus területre való belépéshez. A koordinátor nem válaszol.
- c) Amikor az 1-es folyamat elhagyja a kritikus területet, értesíti erről a koordinátort, aki ezután válaszol a 2-es folyamatnak.

Kölcsönös kizárás: Az elosztott algoritmus

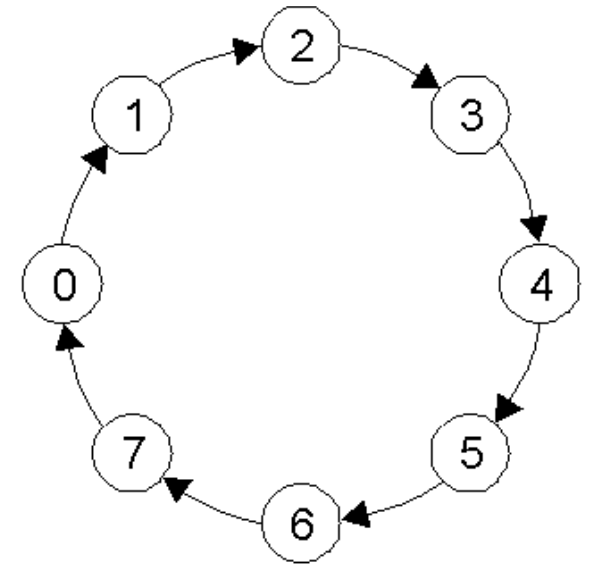


- Két folyamat akar ugyanabban a pillanatban belépni ugyanarra a kritikus területre.
- A 0-ás folyamatnak alacsonyabb az időbélyege, így ő nyer.
- Amikor a 0-ás folyamat végzett, ő is elküldi az OK választ, így a 2-es folyamat már beléphet a kritikus területre.

Kölcsönös kizárás: A zsetongyűrű algoritmus



(a)



(b)

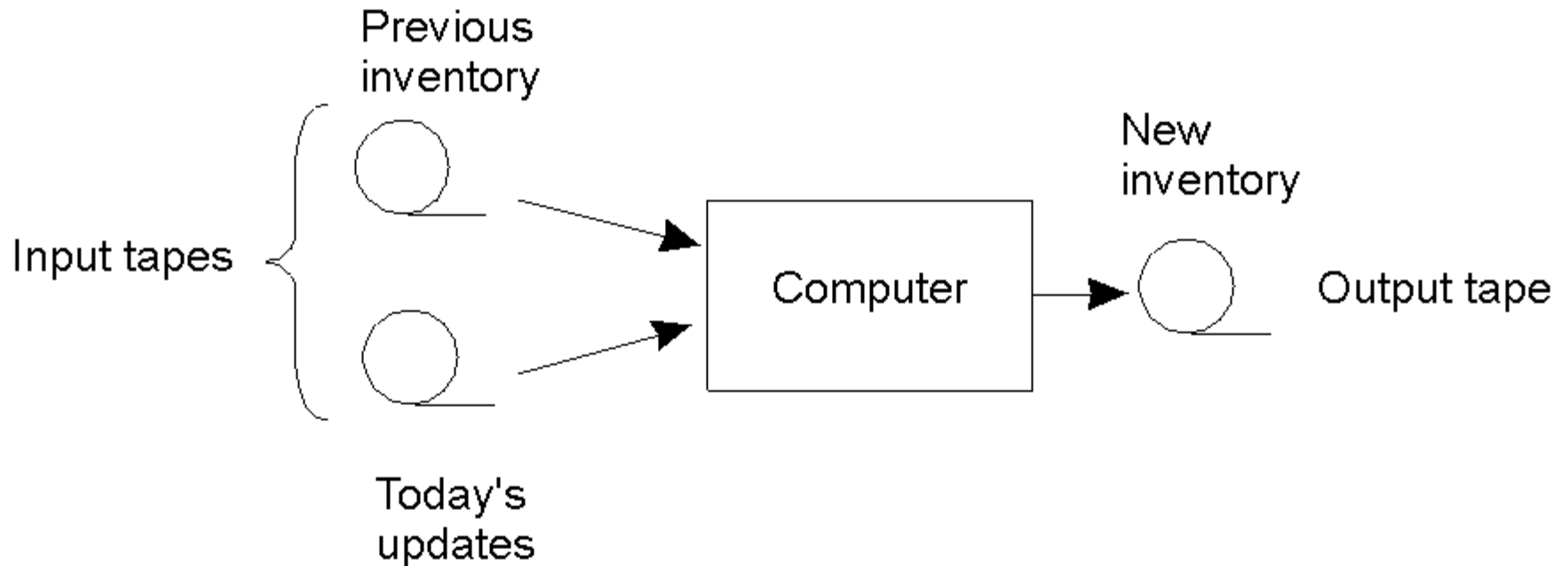
- a) A folyamatok rendezetlen csoportja a hálózaton.
- b) A szoftver által alkotott logikai gyűrű.

Kölcsönös kizárás: összehasonlítás

| Algoritmus | Belépési / kilépési üzenetszám | Belépés előtti késleltetés (üzenetidőben) | Problémák |
|-------------------|---|--|--|
| Központosított | 3 | 2 | A koordinátor összeomlása |
| Elosztott | $2(n - 1)$ | $2(n - 1)$ | Bármely folyamat összeomlása |
| Zsetongyűrű | 1 to ∞ | 0 to $n - 1$ | Elveszett zseton, a folyamat összeomlása |

A kölcsönös kizárást biztosító algoritmusok összehasonlítása.

Elosztott tranzakciók - tranzakciómodell (1)



A mesterszalag frissítése hibatűrő folyamat.

Elosztott tranzakciók - tranzakciómodell (2)

| Primitív | Leírása |
|-------------------|--|
| BEGIN_TRANSACTION | Tranzakció kezdetének jelzése |
| END_TRANSACTION | Tranzakció végének jelzése |
| ABORT_TRANSACTION | A tranzakció félbeszakítása és az eredeti állapot visszaállítása |
| READ | Adat beolvasása állományból, táblából vagy bárhonnán |
| WRITE | Adat kiírása állományba, táblába vagy bárhová |

Néhány példa a tranzakció-primitívekre.

Elosztott tranzakciók - tranzakciómodell (3)

```
BEGIN_TRANSACTION  
reserve WP -> JFK;  
reserve JFK -> Nairobi;  
reserve Nairobi -> Malindi;  
END_TRANSACTION
```

(a)

```
BEGIN_TRANSACTION  
reserve WP -> JFK;  
reserve JFK -> Nairobi;  
reserve Nairobi -> Malindi full =>  
ABORT_TRANSACTION
```

(b)

- a) 4 járat helyfoglalása egy tranzakcióként
- b) A tranzakció félbeszakad, amikor a negyedik járat nem elérhető

A tranzakciók tulajdonságai (ACID)

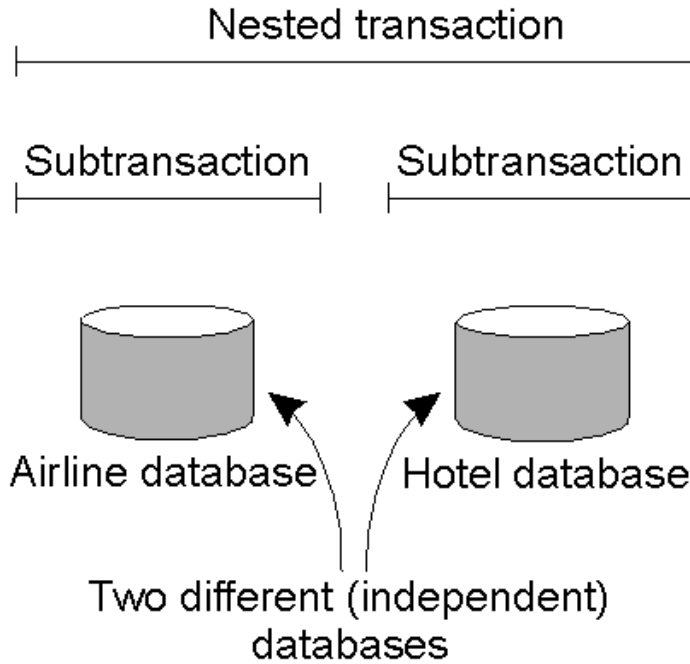
1. Oszthatatlan (elemi) - Atomic
2. Konzisztens - Consistent
3. Elkülönülő - Isolated
4. Maradandó - Durable

A tranzakciók osztályozása

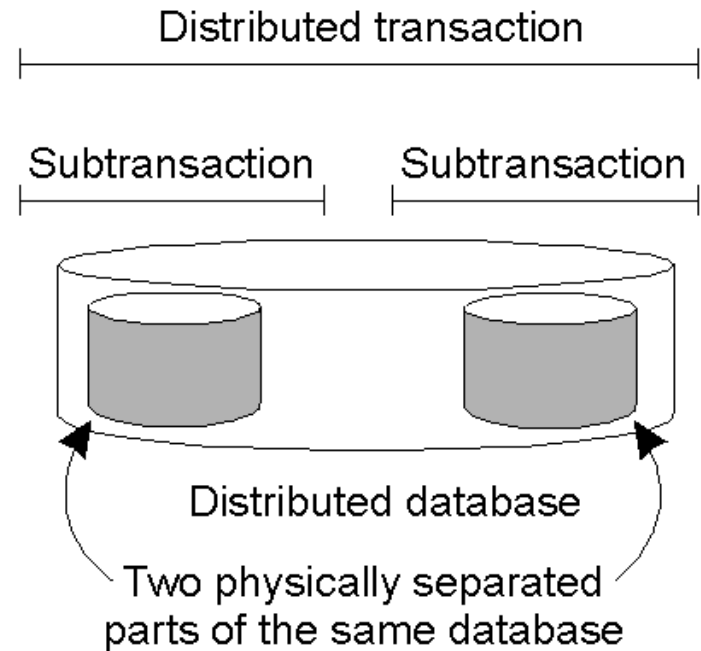
Az egységes tranzakciók korlátai

- Beágyazott tranzakciók
- Elosztott tranzakciók

Elosztott tranzakciók



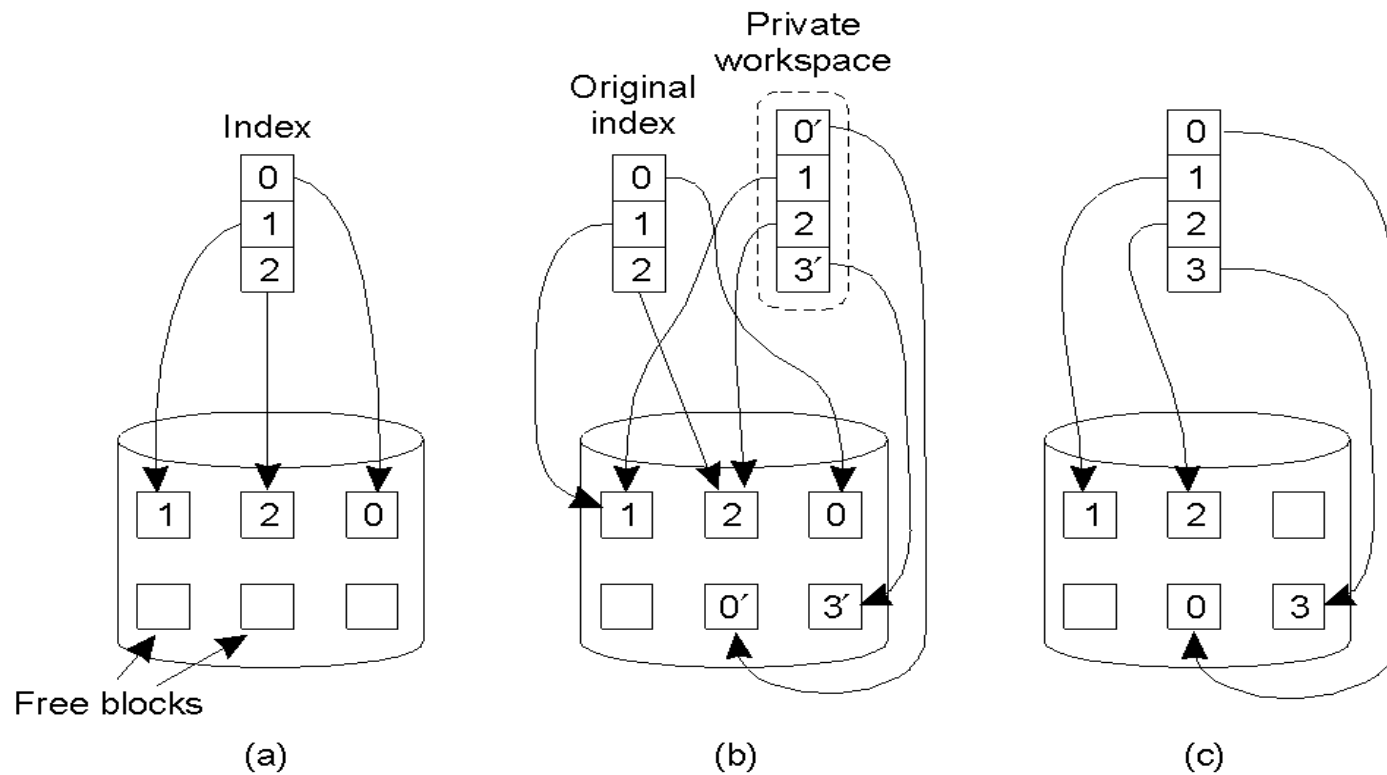
(a)



(b)

- a) Beágyazott tranzakció
- b) Elosztott tranzakció

Implementáció – saját munkaterület



- a) Egy három blokk hosszúságú állomány indexei és lemezblokkjai
- b) Az állomány 0-ás blokkjának módosítása a tranzakció során, és egy negyedik blokk hozzáfűzése után kialakult helyzet
- c) A tranzakció lezárása utáni helyzet

Implementáció – munkatervnapló

| | | | |
|---------------------------------|-------------|-------------|-------------|
| <code>x = 0;</code> | Log | Log | Log |
| <code>y = 0;</code> | | | |
| <code>BEGIN_TRANSACTION;</code> | | | |
| <code> x = x + 1;</code> | [x = 0 / 1] | [x = 0 / 1] | [x = 0 / 1] |
| <code> y = y + 2</code> | | [y = 0/2] | [y = 0/2] |
| <code> x = y * y;</code> | | | [x = 1/4] |
| <code>END_TRANSACTION;</code> | | | |
| (a) | (b) | (c) | (d) |

a) Egy tranzakció

b) – d) Az egyes utasítások végrehajtása előtt kiírt naplóbejegyzés