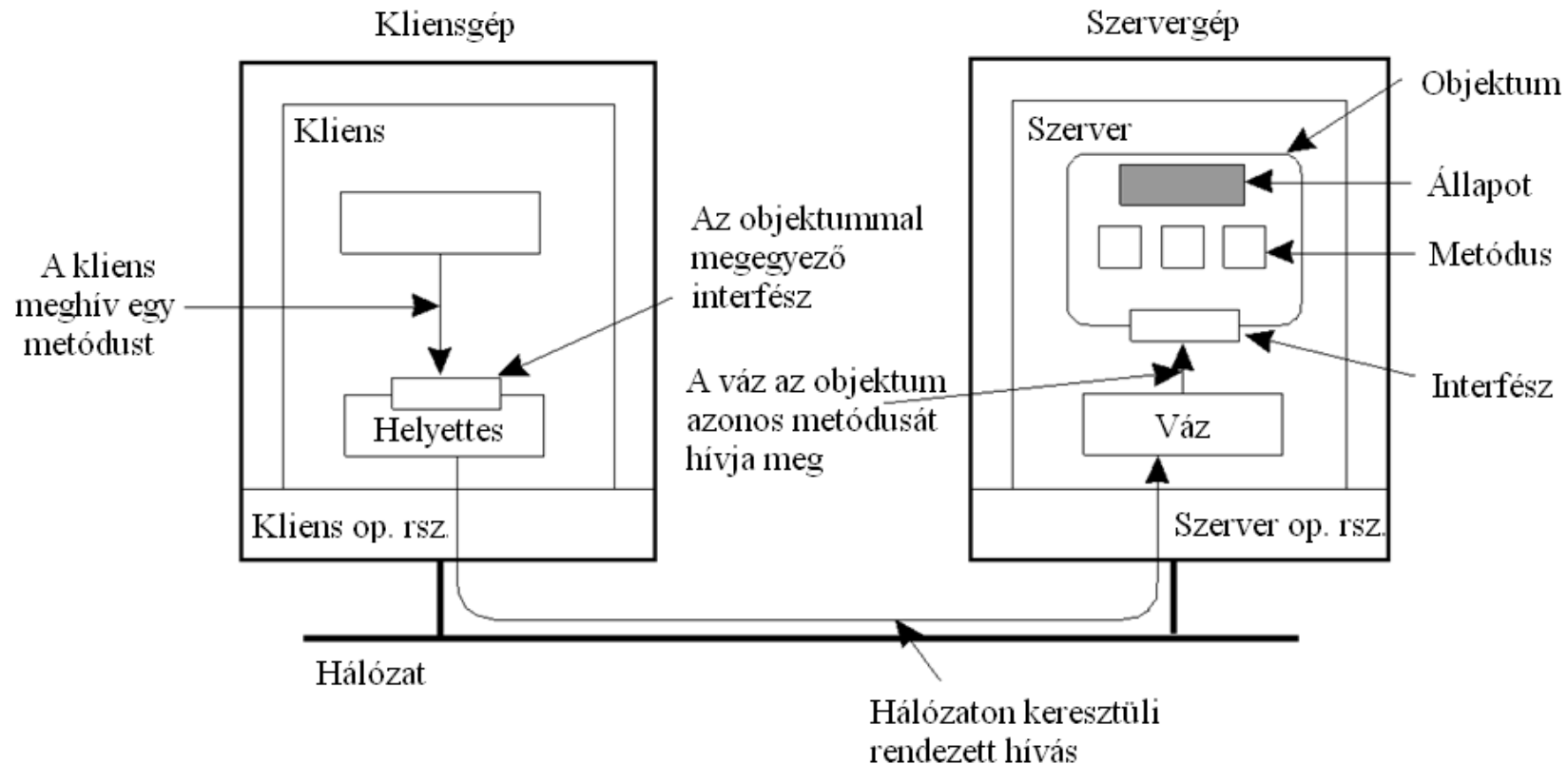


Kommunikáció

3. előadás

RMI – elosztott objektum alapú
kommunikáció

RMI – elosztott objektumok



A távoli objektum elterjedt kialakítása a kliensoldali helyettes (proxy) alkalmazásával

Modell – kliensoldali helyetttessel és szerveroldali vázzal.

RMI

- Elosztott objektum
- Távoli objektum
- Fordítási idejű objektum
- Futási idejű objektum
- Állandó objektum
- Nem állandó objektum

Kliens csatlakozása az objektumhoz

```
Distr_object* obj_ref; // Deklarálja az objektumhivatkozást  
obj_ref = ...; // Inicializálja az objektumhivatkozást  
obj_ref->do_something(); // Implicit csatlakozás és metódushívás
```

(a)

```
Distr_object objPref; // Deklarálja az objektumhivatkozást  
Local_object* obj_ptr; // Deklarálja a helyettes mutatóját  
obj_ref = ...; // Inicializálja az objektumhivatkozást  
obj_ptr = bind(obj_ref); // Explicit csatlakozás (helyettes mutatója)  
obj_ptr -> do_something(); // Helyettes metódusának meghívása
```

(b)

a) Implicit csatlakozás

b) Explicit csatlakozás

RMI – távoli művelet hívása

- Statikus hívás

- IDL -> csonk

`fobject.append(number);`

- Dinamikus hívás

- futás közbeni összeállítás

`invoke(fobject, id(append), number);`

[általánosan:

`invoke(objektum, metódus,`

`input_paraméterek, output_paraméterek); ]`

Java RMI

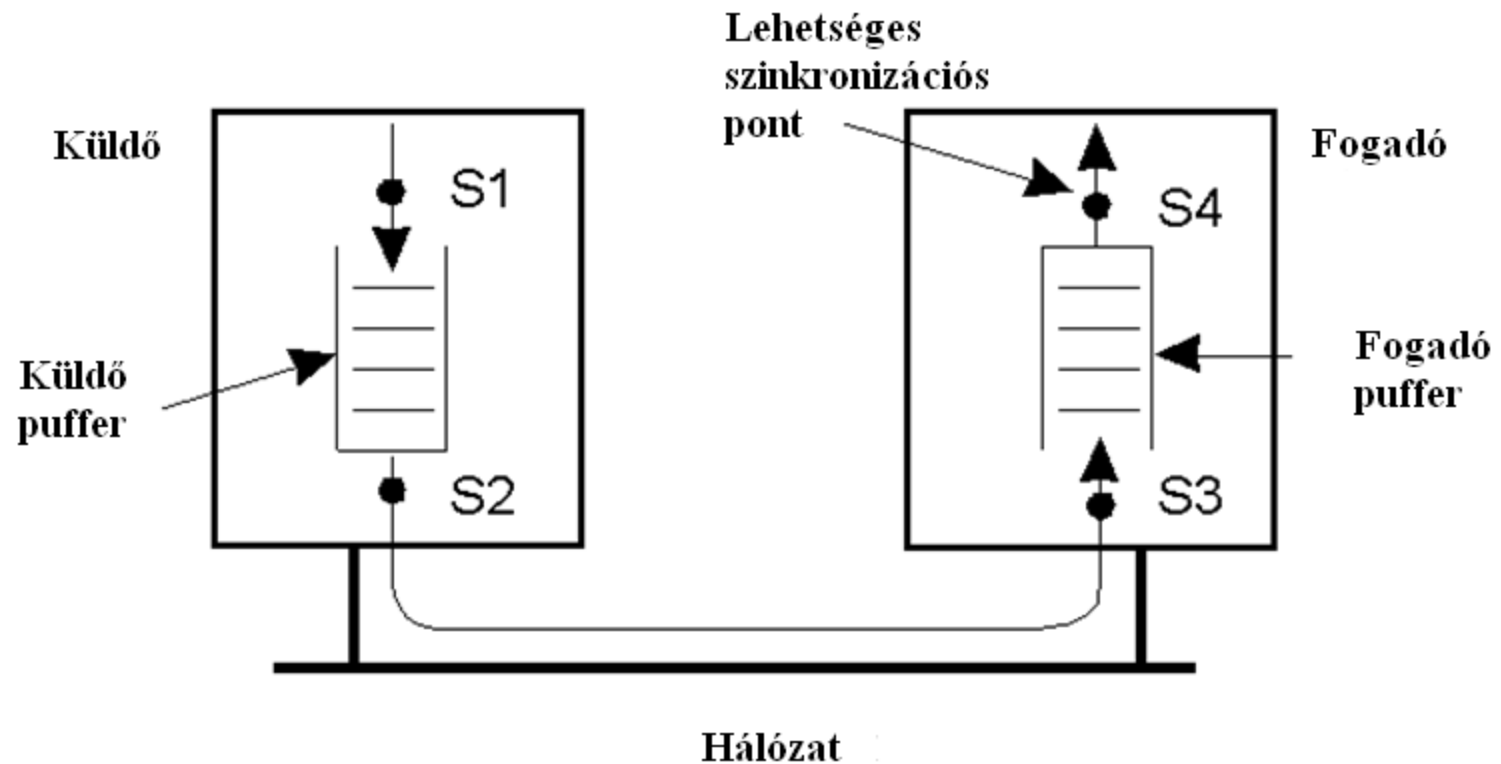
- Csak távoli objektum
- Objektumok zárolása – synchronized
 - kliensen blokkol
- Helyettes serializálható

Üzenetorientált kommunikáció

Üzenetorientált kommunikáció

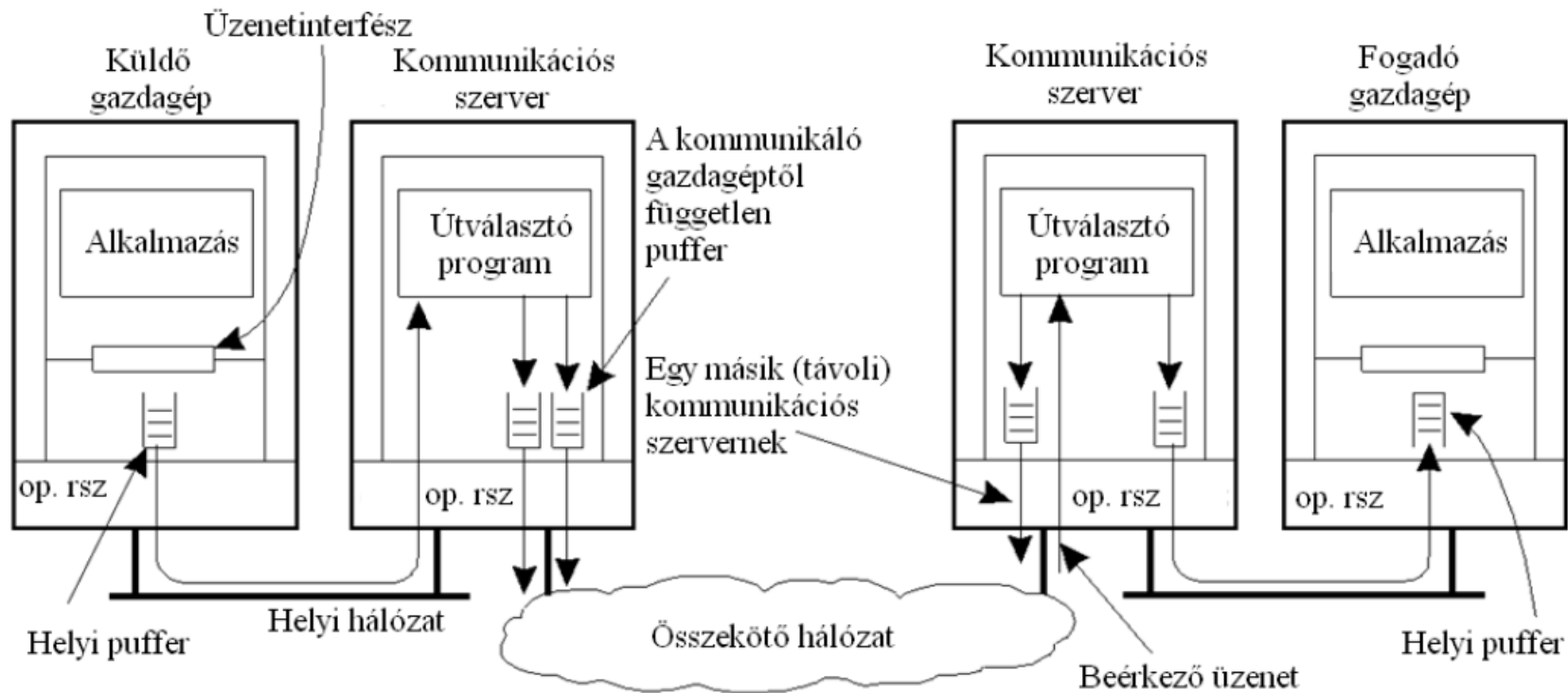
- RPC, RMI nem mindig használható:
- Szerver nem biztos, hogy üzemel a kérés elküldésekor
- Nem mindig hatékony a nagyfokú szinkronitás miatt
- => üzenetcsere

Korábbi kommunikációs modell



Pufferelés és a küldő blokkolásának lehetséges helyei

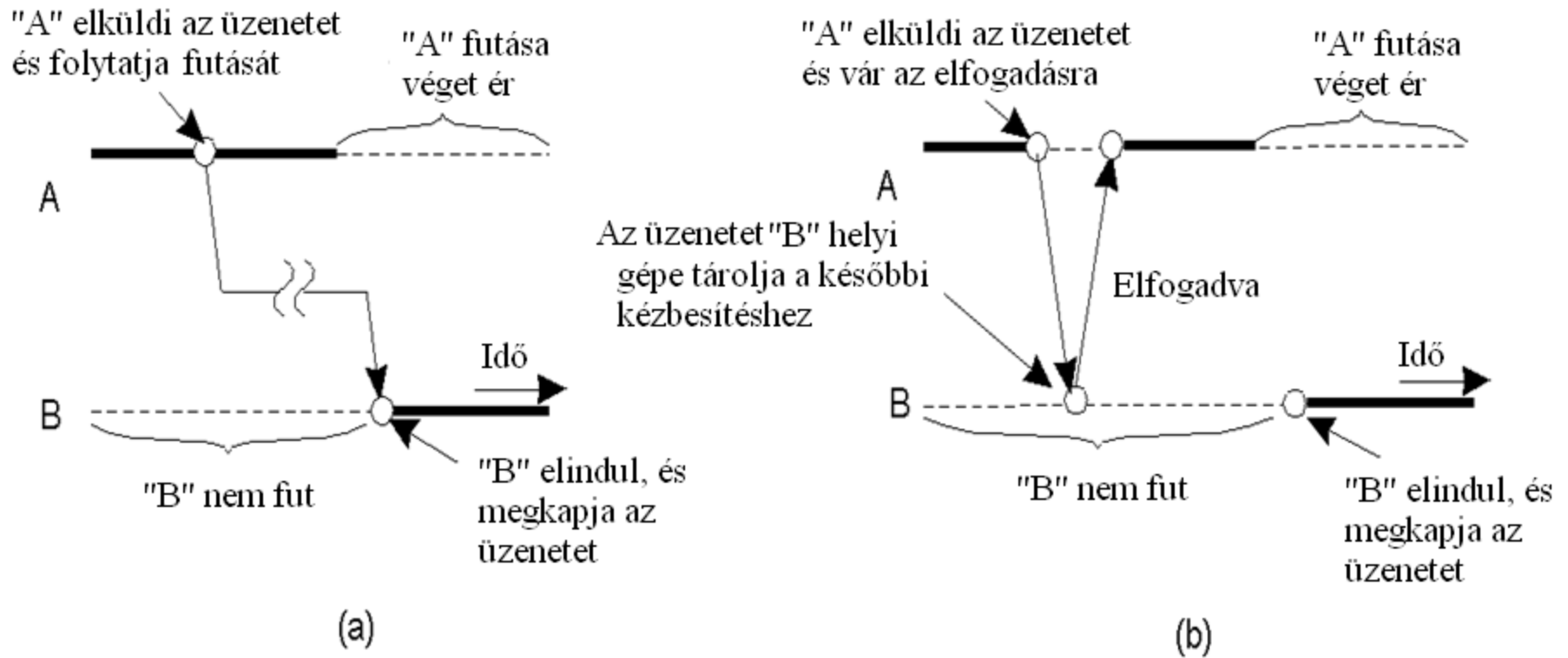
Új kommunikációs modell



Üzenetküldés

- Megtartó kommunikáció
- Időleges kommunikáció
- Szinkron kommunikáció
- Aszinkron kommunikáció
- Ezek kombinációja is lehetséges

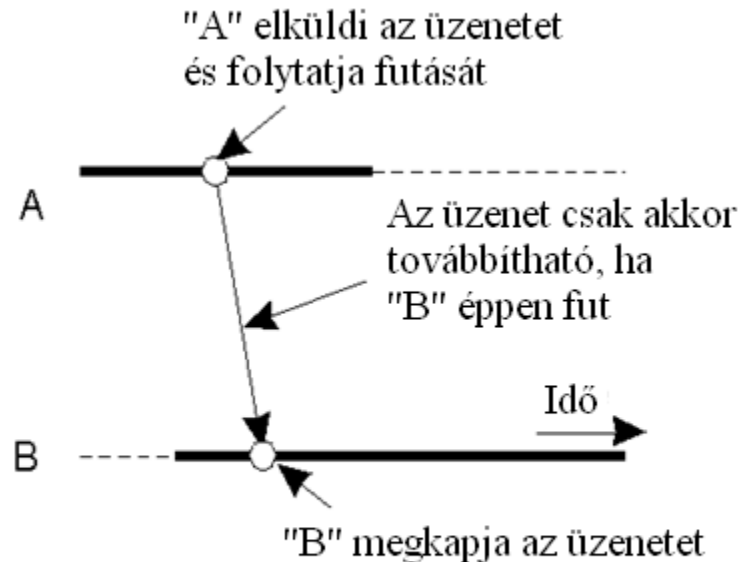
Kommunikáció fajtái – 1.



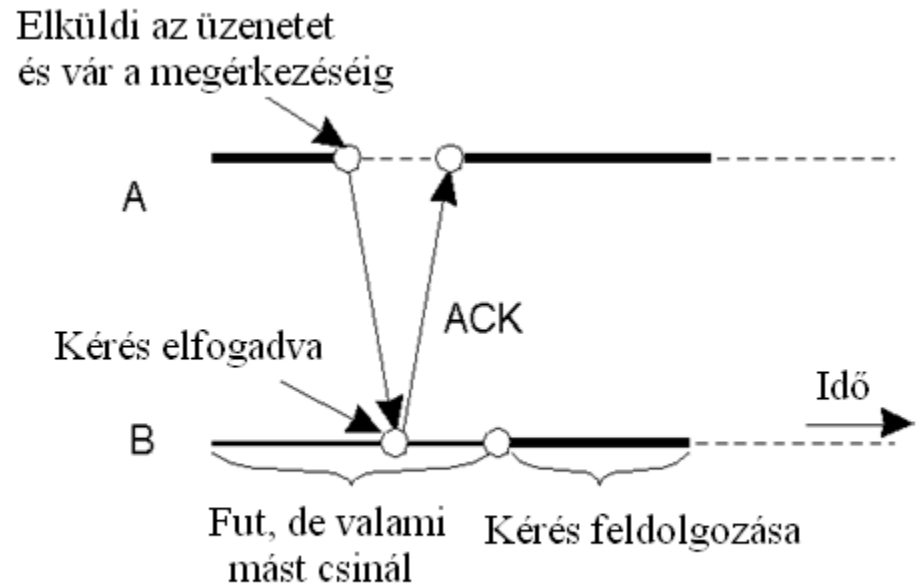
a) Megtartó aszinkron kommunikáció

b) Megtartó szinkron kommunikáció

Kommunikáció fajtái – 2.



(c)

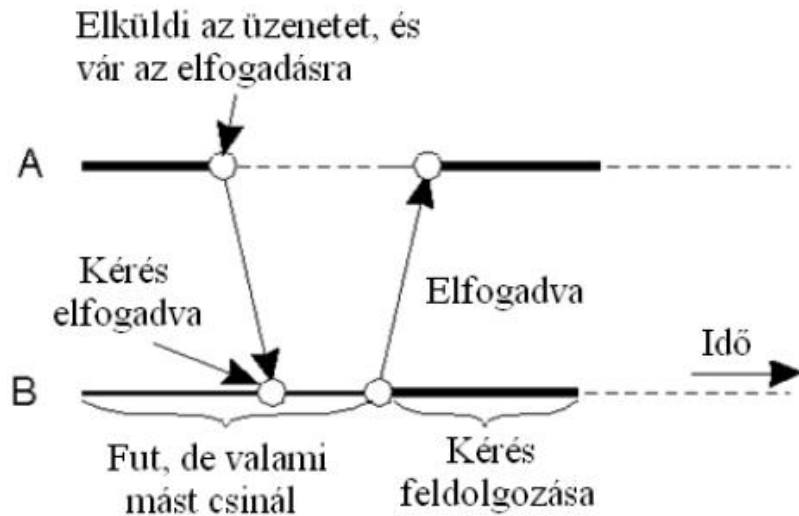


(d)

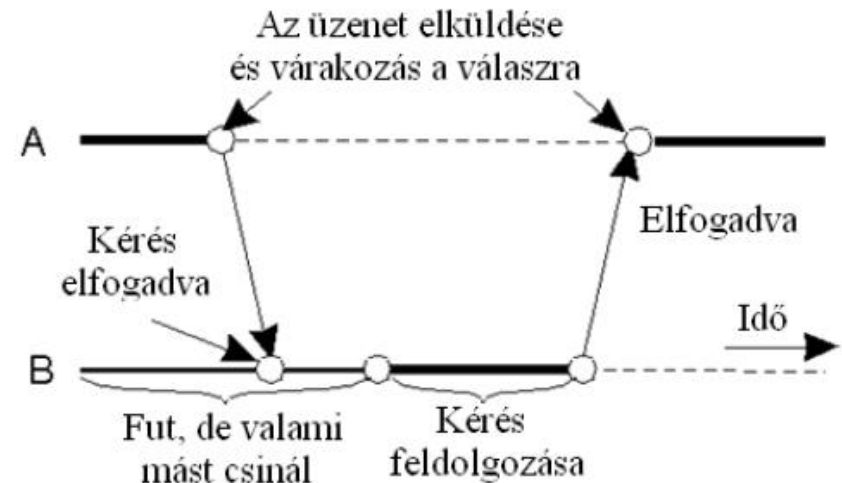
c) Időleges aszinkron kommunikáció

d) Nyugta alapú időleges szinkron kommunikáció

Kommunikáció fajtái – 3.



(e)



(f)

e) Kézbesítés alapú időleges szinkron kommunikáció

f) Válasz alapú időleges szinkron kommunikáció

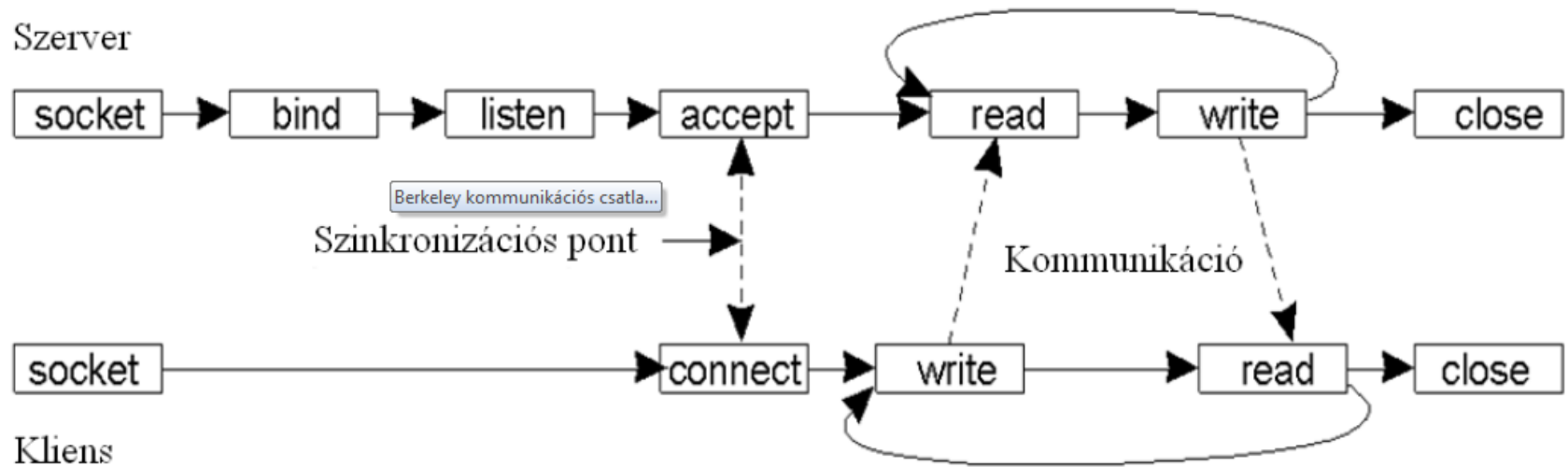
Üzenetorientált időleges kommunikáció

- Szállítási réteg által biztosított modell
- Kommunikációs csatlakozópont (socket)
- TCP primitívek

Berkeley kommunikációs csatlakozópontok

Primitív	Jelentése
Csatlakozópont (Socket)	Új kommunikációs végpont létrehozása
Kötés (Bind)	A csatlakozóponthoz hozzárendel egy helyi címet
Figyel (Listen)	Meghirdeti a kommunikációra való képességet
Elfogad (Accept)	Blokkolja a hívót a kapcsolatfelvételi kérés megérkezéséig
Csatlakozik (Connect)	Aktívan megkísérli a kapcsolat felépítését
Küld (Send)	Adatot küld a kapcsolaton keresztül
Fogad (Receive)	Adatot fogad a kapcsolaton keresztül
Lezár (Close)	Bontja a kapcsolatot

Berkeley kommunikációs csatlakozópontok



MPI – Üzenetátadó interfész

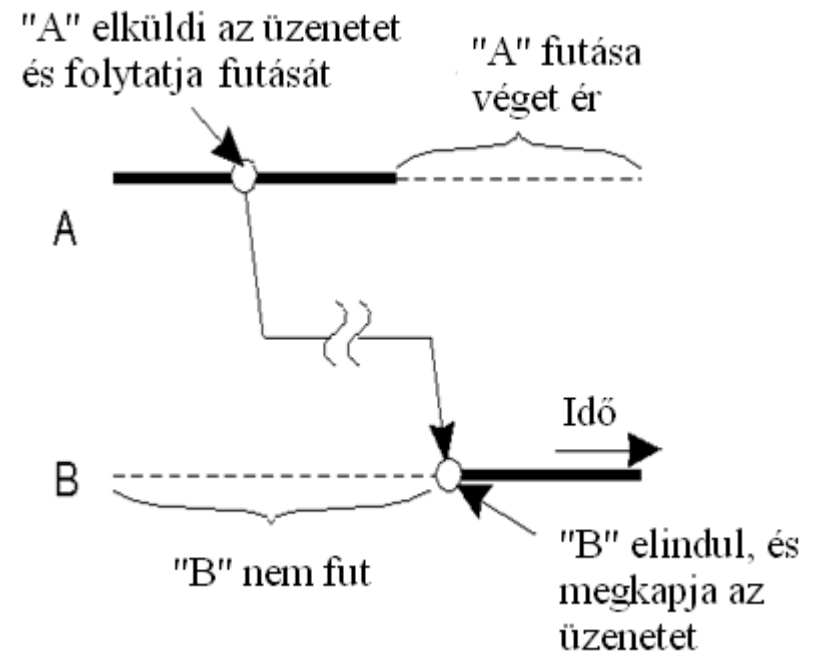
- Message-Passing Interface
- Csatlakozóponos megoldás nem eléggé magas szintű
 - > kommunikációs könyvtárak – inkompatibilitás
 - > MPI (szabványosítás)
- Időleges kommunikáció
- Azonosító (csoport, folyamat)

MPI – Üzenetátadó interfész

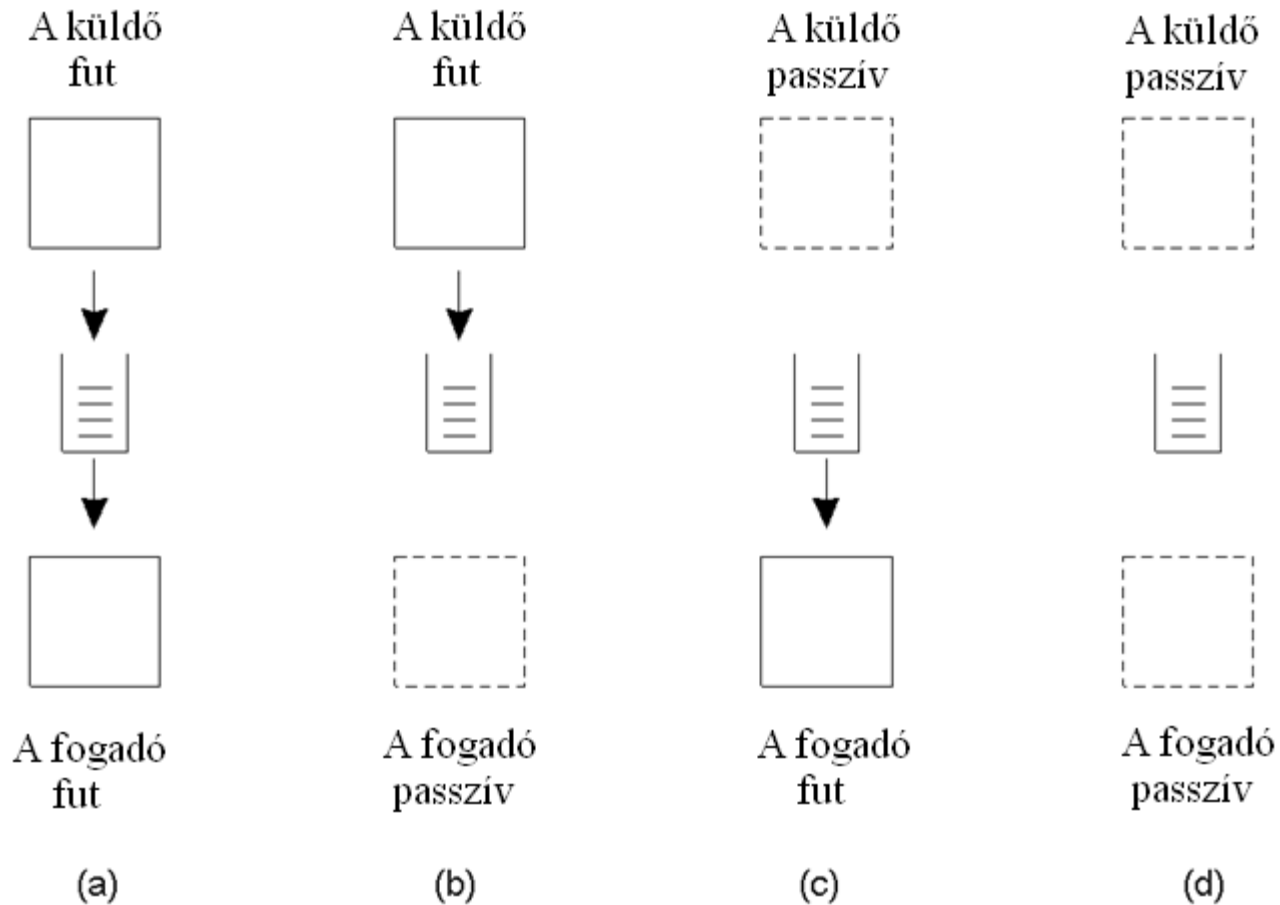
Primitív	Jelentése
MPI_bsend	A helyi küldőpuffer végéhez hozzáadja a kimenő üzenetet
MPI_send	Üzenetet küld, és vár annak helyi vagy távoli pufferbe történő másolásáig
MPI_ssend	Üzenetet küld, és vár a nyugta megérkezéséig
MPI_sendrecv	Üzenetet küld, és vár a válasz megérkezéséig
MPI_isead	Átad egy kimenő üzenetre mutató hivatkozást, és továbbmegy
MPI_issend	Átad egy kimenő üzenetre mutató hivatkozást, és várja ennek nyugtázását
MPI_recv	Üzenet fogadása; blokkolódik, ha nincs üzenet
MPI_irecv	Ellenőrzi, hogy van-e beérkező üzenet, de nem blokkolja a hívót

Üzenetorientált megtartó kommunikáció

- Üzenetorientált köztesréteg – MOM (Message Oriented Middleware)
- Megtartó aszinkron kommunikáció
- Üzenetek köztes szinten való tárolása (várakozási sorok)



Üzenetsor-kezelő modell



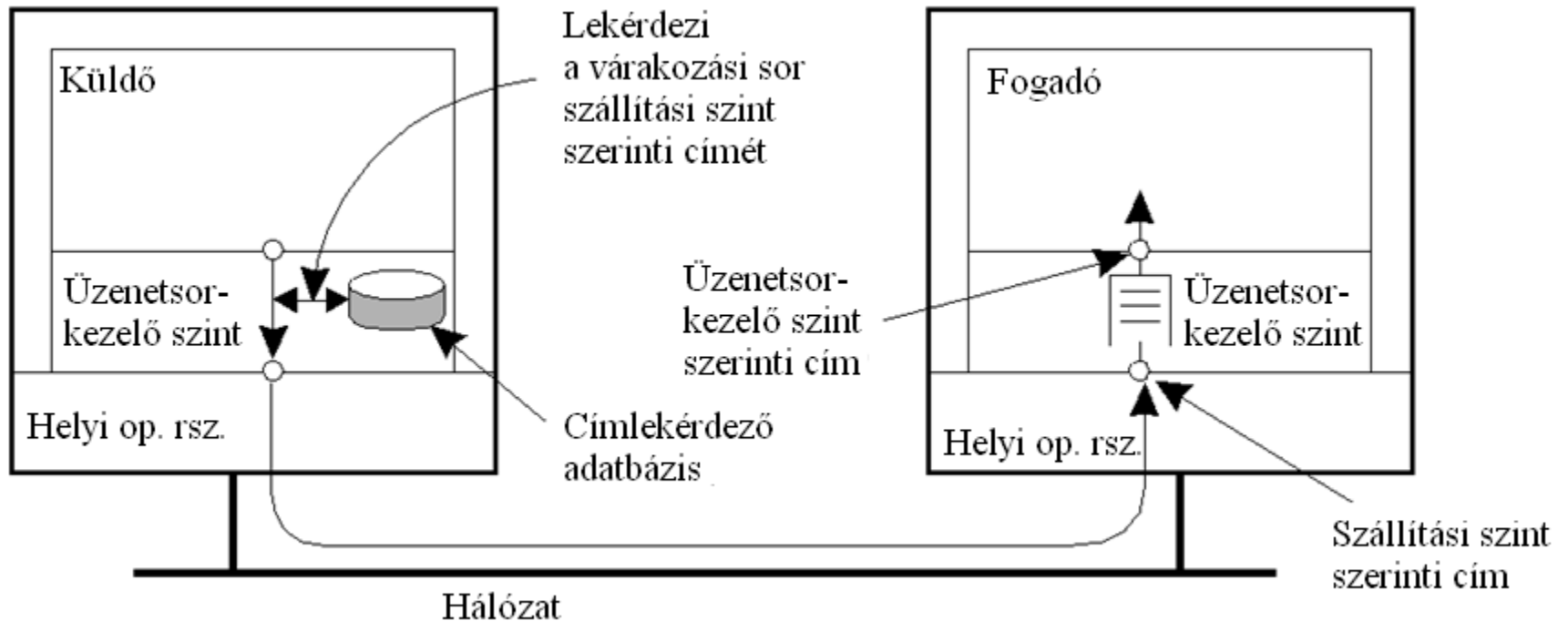
Várakozási sort használó gyengén csatolt kommunikáció 4 lehetősége

Üzenetsor-kezelő modell

Primitív	Jelentése
Betesz (Put)	A megadott várakozási sor végére helyez egy üzenetet
Kivesz (Get)	Mindaddig blokkol, amíg üres a várakozási sor, majd kiveszi a legelső üzenetet
Lekérdez (Poll)	Megvizsgálja, a megadott várakozási sort, és kiveszi a legelső üzenetet, nem blokkol
Értesít (Notify)	Egy kezelő rutint installál, amely mindannyiszor meghívásra kerül, ha üzenet kerül a várakozási sorba

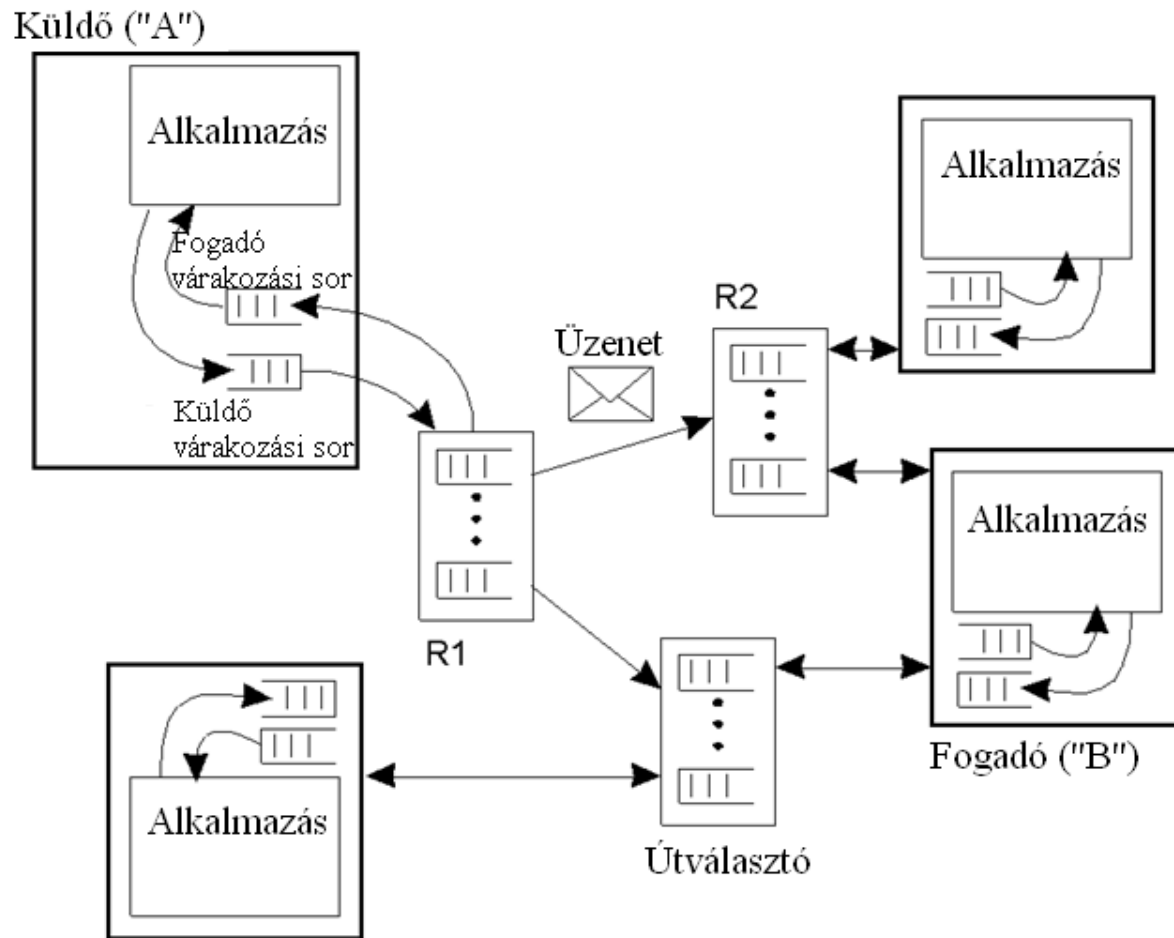
Az üzenetsor-kezelő rendszer várakozási sorát kezelő
alapinterfész

Az üzenetsor-kezelő rendszer általános felépítése



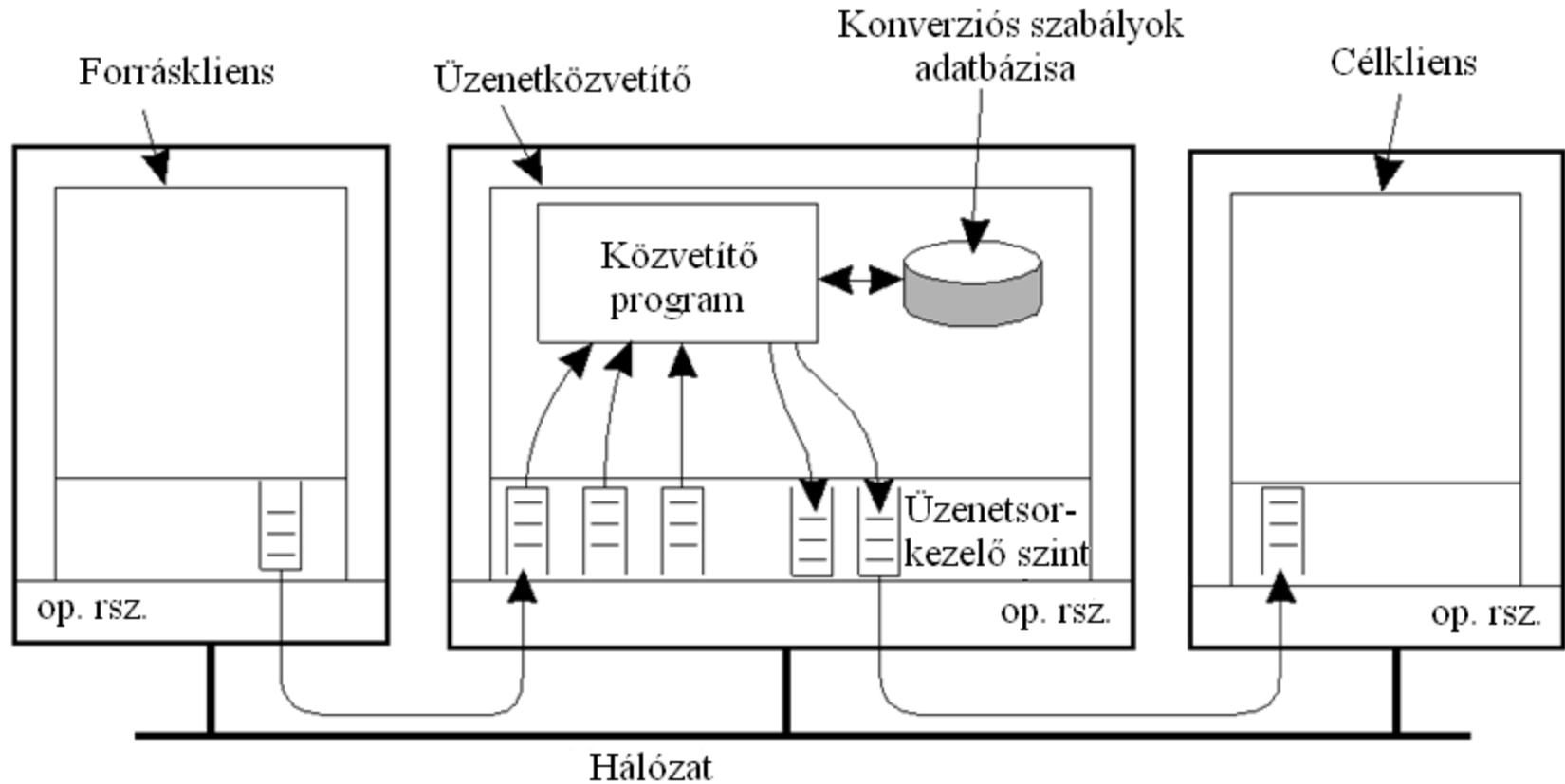
Az üzenetsor-kezelő szint szerinti és szállítási szint szerinti cím

Az üzenetsor-kezelő rendszer általános felépítése



Az útválasztókkal kiépített üzenetsor-kezelő rendszer általános felépítése

Üzenetközvetítők



Az üzenetközvetítőt használó üzenetsor-kezelő rendszer általános felépítése