



SSADM OO nézőpontból

Molnár Bálint
Egyetemi docens,
Corvinus egyetem

Információrendszerek

Információrendszerek → *tartósan tárolt adatok*

adatbázis-központú

adatbázis-tervezés fontossága:

hatékonyság

stabilitás

módosíthatóság, továbbfejleszthetőség

hatékonyság elérése

relációs adatelemzés

lekérdezési út

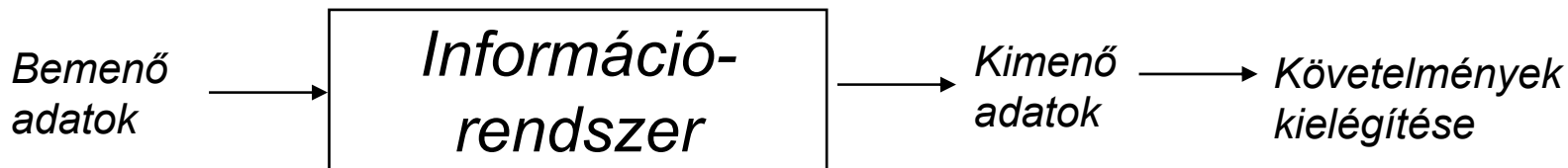
heurisztikus fizikai adatbázis-tervezési szabályok

Irodalom: Robinson, K., Berrisford, G., 'Object-oriented SSADM', Prentice Hall, 1994.

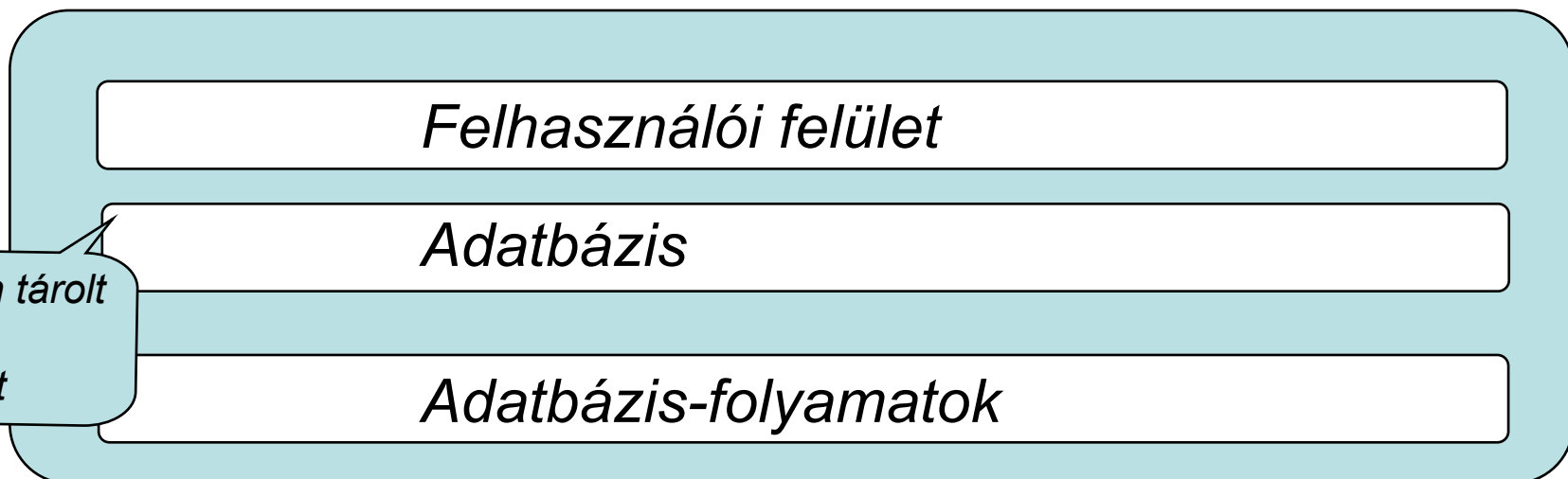
Információrendszerek

Modell készítés

a "valóságos világ" modellezése



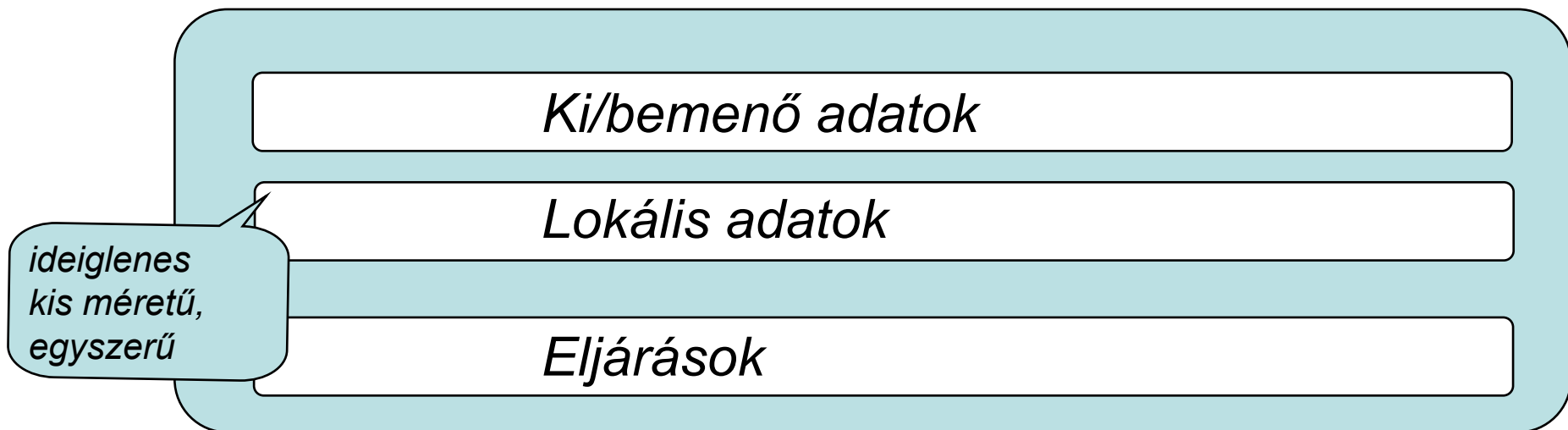
- *Információrendszer belső szerkezete*



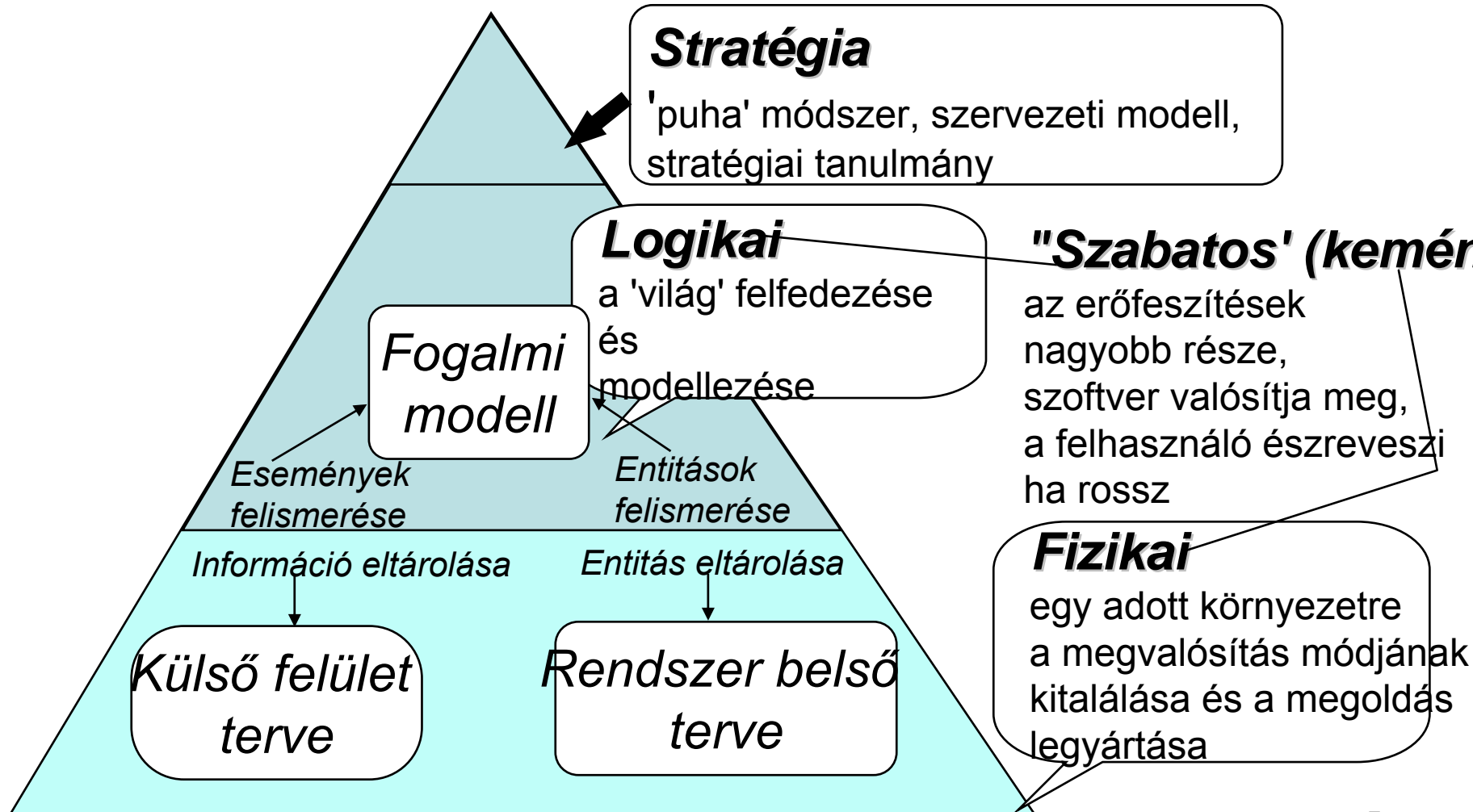
Program, objektum, automata

egy program kimenete közvetlenül megfeleltethető a bemenetének, meghatározza a kimenetét;

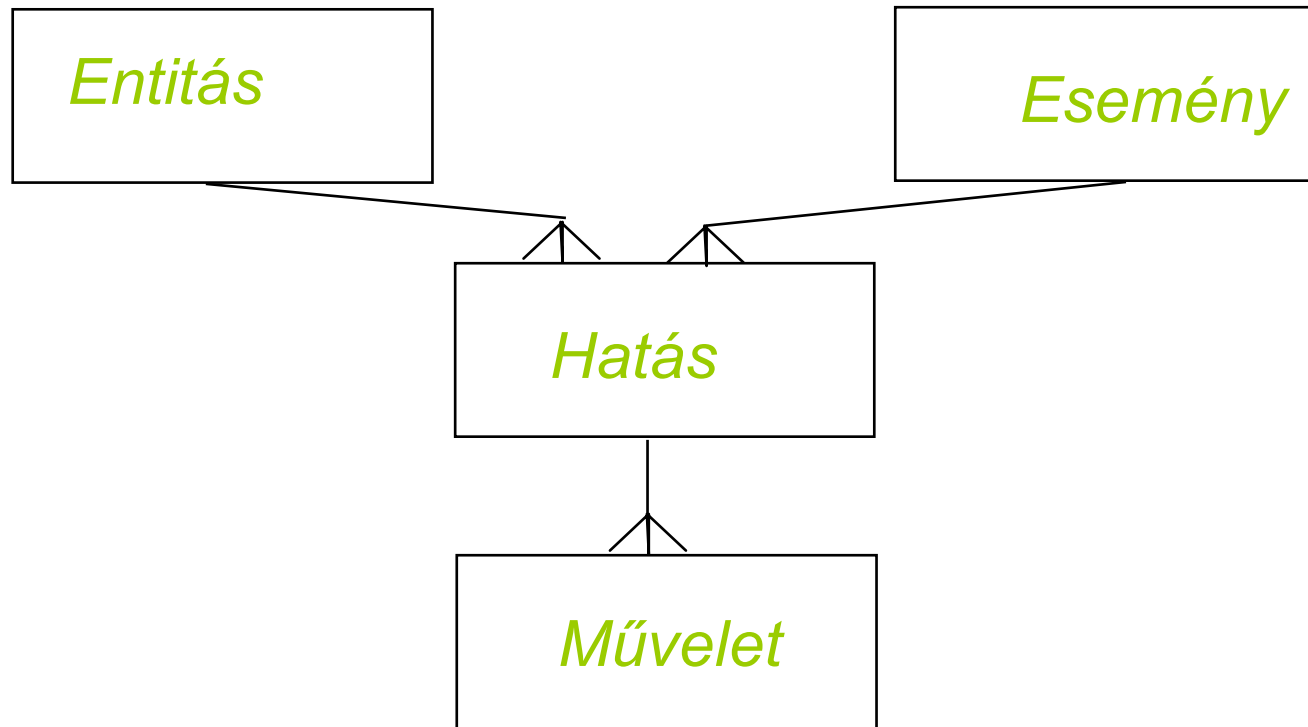
információrendszer kimenetét csak közvetve határozza meg a bemenete.



Alkalmazási rendszer (információrendszer)



Esemény, hatás, metódu



Esemény, hatás, metódus

Esemény (informatikai)

a szervezetben, annak környezetében bekövetkező események
a rendszer entitásaira gyakorolnak hatást
felismerhetőek
a rendszer számára bemeneti adatként jelenhet meg
bemenő adatcsoport
adatbázis aktualizáló adateljárás

Esemény

a sikeresen végrehajtandó feldolgozási egység (commit unit)
valós világbeli megfelelője

Szervezeti esemény

(felhasználói) funkció kezdeményezése:
funkció tervezés, GUI, felhasználói felület tervezés

Esemény és lekérdezés

Esemény

megváltoztatja a rendszer állapotát

Lekérdezés

a rendszert változatlan állapotban hagyja

Hatás

Aktualizálási hatás

attribútum, kapcsolat, állapot változtatás

Ellenőrzési hatás

állapot jelző értékének ellenőrzése a
feldolgozás előtt

Lekérdezési hatás

egy entitásból információt nyer ki egy másik
entitás aktualizálása végett

Metódus

Egy esemény egy entitásra gyakorolt hatását jeleníti meg, típusai:

egy esemény hatása

opcionális hatások közötti választás (amelyet egy esemény kezdeményez)

szuper-esemény (több esemény kezdeményezheti)

Mi nem a metódus

nem fizikai adatbázis művelet (a megvalósítás részleteihez tartoznak ezek a műveletek)

hanem a probléma terület szótárában használt 'objektum' manipuláció

Jelentés háromszög

Fogalom

intenzió

hivatkozik

Szimbólum

extenzió

Valós világ eleme

Entitás viselkedés modellezése

Célja:

az események hatásának elemzése

az entitásokra gyakorolt hatás feljegyzése

Leírja:

egy entitás (típus) viselkedését, amely érvényes minden egyes példányára

akkortól, amikor a rendszer látókörébe kerül addig, míg ki nem kerül onnan

mi történik az entitással és az mit csinál, megmutatja azokat az eseményeket, amelyek hatnak az entitásra, esetleg, amire az entitás hat

hatásokat:

sorrend, szelekció, iteráció (reguláris kifejezések)

Entitás viselkedés modellezése

Elemzés

az entitás mélyebb megértése

logikai adatmodell változtatása

a felhasználó által elvárt rendszer működés megértése

Tervezés

az eseményekhez kötődő adatfeldolgozás

megfogalmazására ad egy formális módszert

OO terminológiában

az objektum viselkedését írja le az idő dimenzióban,
azaz a metódusok hívási sorrendjét, valamint az
erre vonatkozó kényszerfeltételeket

a metódus hívások fölött definiált nyelvtan
(grammatika)

OO kontra vezérlési struktúra

OO megközelítés előnyei

nincs vezérlési struktúra, amely a metódus hívásra vonatkozó kényszerfeltételeket adná meg
helyette: állapotjelzők, változók

Hátránya

nincs vezérlési struktúra

a metódus hívási környezete nehezen azonosítható
akadályozza az elemzést

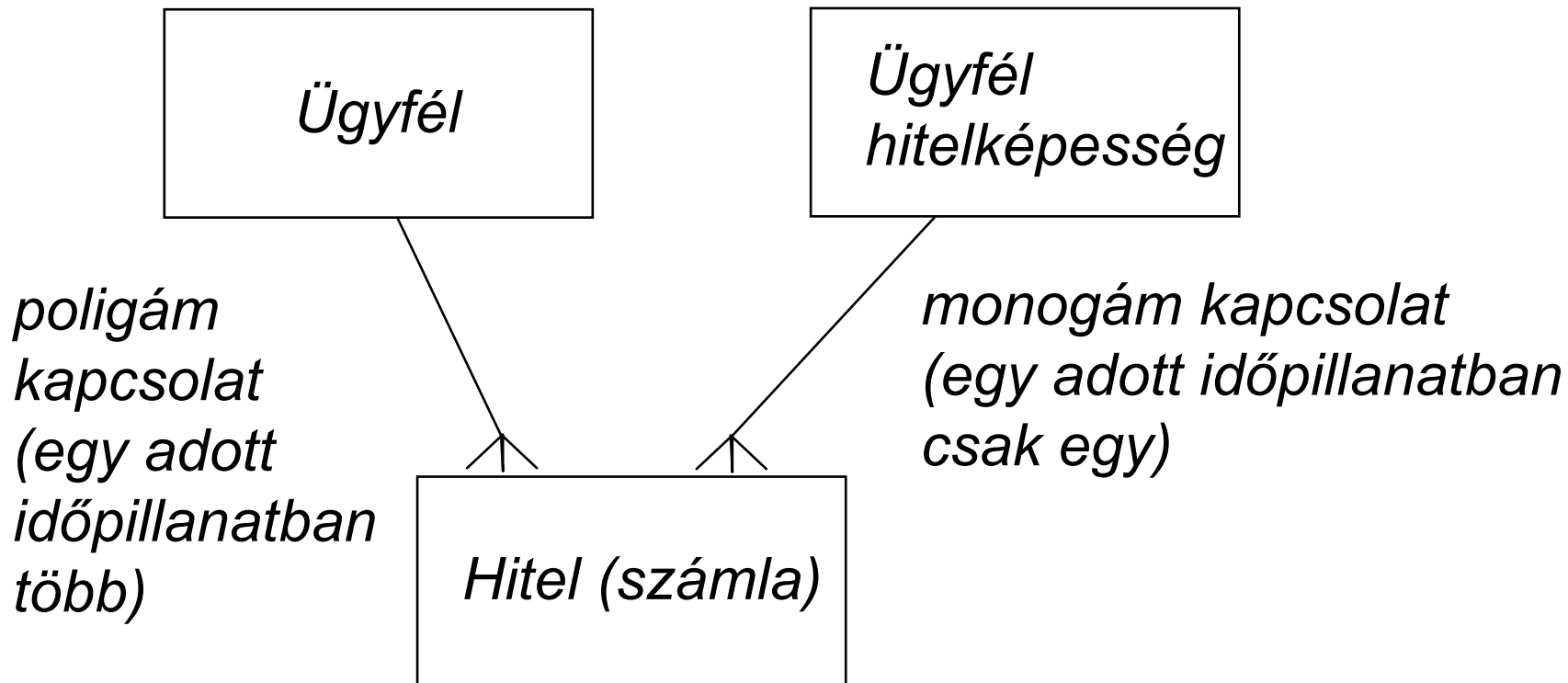
a kényszerfeltételek, korlátok

előfeltételek

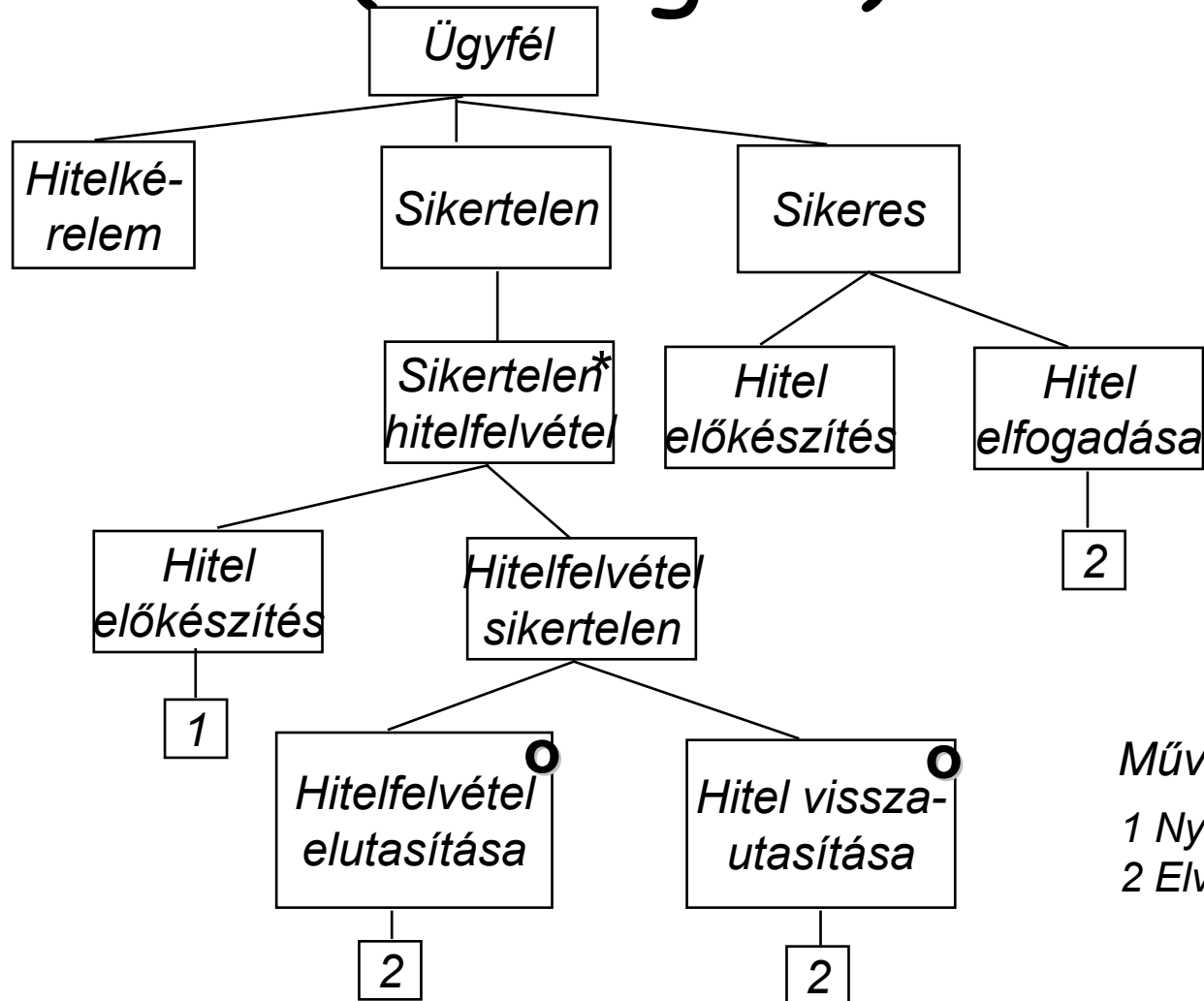
szabályok

felismerésében

Adatszerkezet részlet



Entitás viselkedés modell (monogám)



Műveletek listája

- 1 Nyer (Gain) Hitel-t
- 2 Elveszít (Lose) Hitel-t

Műveletek listája

I:	INSERT	(B, beszúrás)
M:	MODIFY	(M, módosítás)
D:	DEATH	(H, halál)
B:	BURIED	(T, delete) (törlés)
G:	GAIN DETAIL	(N, alentitás nyerése)
L:	LOSE DETAIL	(V, alentitás elvesztése)
T:	TIE	(K, főentitáshoz kapcsolás)
C:	CUT	(L, főentitásról leválasztás)
X:	SWAP DETAIL(S)	(X, alentítások cseréje)
S:	SWAP MASTER(S)	(C, főentítások cseréje)
R:	READ	(O, olvasás, eseményekben/lekérdezésekben)

Állapotjelzőkkel bővítés

Megelőző és végállapot feljegyzése

az entitás állapotait írja le, nem tesz hozzá többet
nirvánából jön (-), nirvánába megy (-), egyesével növelgeti,
betartandóak a sorrendből, szelekcióból, iterációból adódó
szabályok

Állapotok optimalizálása

Állapotok számának csökkentése (ekvivalencia osztályok)

szelekció ágainak végállapota ugyanaz

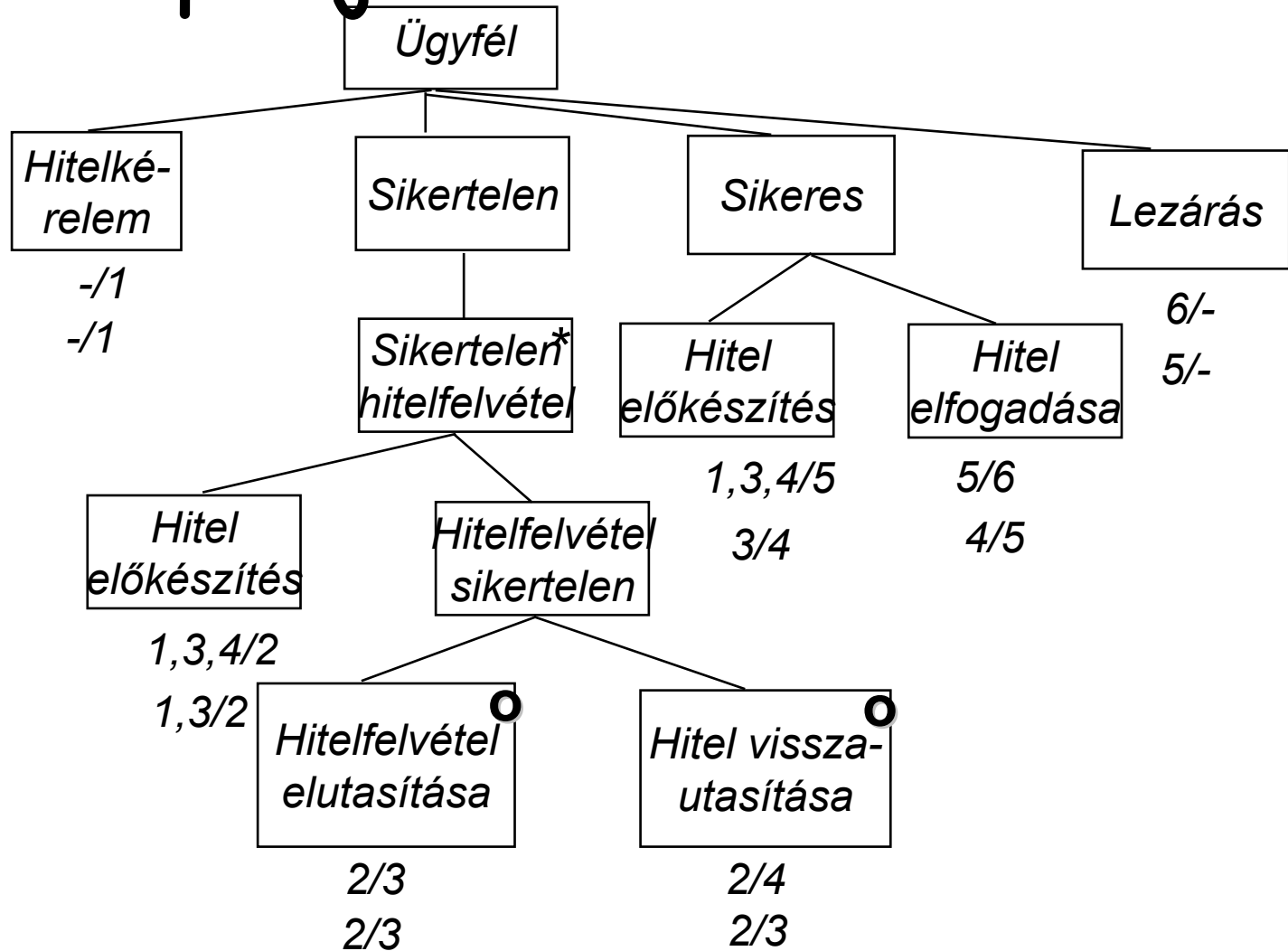
iteráció végállapota megegyezik a bemenő, megelőző állapottal

egyszerűbb események érvényességének ellenőrzése

adatfeldolgozó folyamatok újra felhasználhatósága

az állapotok megnevezése

Állapotjelzőkkel bővítés



Mik lennének a metódusok ?

Hitelkérelem

Hitel elfogadása

Hitel visszautasítása

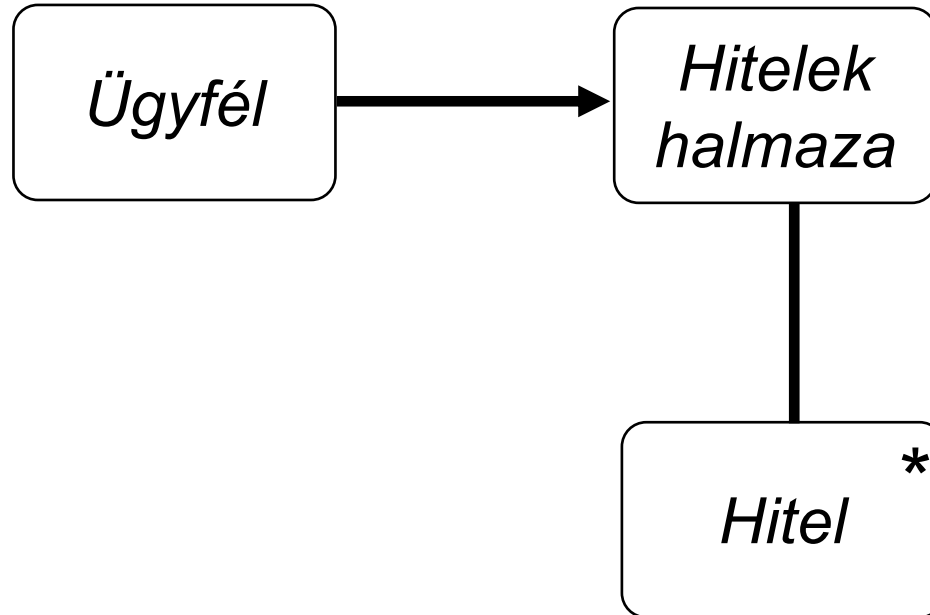
Hitelfelvétel elutasítása

Szuper esemény

szelekció levelei utalhatnak arra, hogy két eseménynek ugyanaz a hatása

OO: metódus, amelyet két különböző esemény kezdeményez

Esemény-hatás diagramok



Esemény-hatás diagramok

ECD (Effect Correspondence Diagram)

megvalósítástól független specifikáció

hagyományos és OO környezetben is használható programok specifikálására

SSADM feltételezi:

Adatbázis kezelő rendszer alkalmazását (DBMS)

ECD átalakítható egy sikeresen végrehajtandó egységgé (commit unit)

OODBMS vagy OO környezetben:

meg kell oldani a tranzakció kezelés hasonló módját

Újra felhasználhatóság

szuper-események: különböző események kezdeményezik, de ugyanaz a "hatás", az adatfeldolgozás, a műveletek listája

Egyszerűsít

Entitás viselkedés modellezés OO fogalmakban

Objektum = entitás + viselkedési modell

A rendszer megismerésének elemzési eszköze

az objektumok felismerésének biztosabb módja mint az entitás-kapcsolat modellezés, vagy a relációs adatelemzés

lépésenként segíti feltárni az események entitásokra gyakorolt hatását

vezérlési struktúrát tartalmaz

fogalmi modell elemzésének előírásos módja

a fogalmi modell inkrementális elemzése

a rendszer megismerésére, felfedezésére ad egy keretet

Entitás viselkedés modellezés OO fogalmakban

A metódusok specifikálása

a hívási sorrendet az entitás viselkedési modell tartalmazza

kikényszeríti a szuper-események felismerését

egy objektum összes metódusát specifikálja:

műveletek, állapotváltozások

meghatározza az elő- és utófeltételeket:

állapotváltozó értékei

integritási szabályok és kényszerfeltételek:

az állapotváltozó értékének ellenőrzése

a teljes entitás esemény viselkedés modellből
levezethető az adatbázis aktualizálás pszeudó
kódja

Entitás viselkedés modell mint módszer

1. menet

Az adatszerkezet hierarchiában

alulról felfelé: először az objektum alentitásainak viselkedését modellezzük

létrehozási és 'halál' események összefüggése a fő- és az alentitás között

a szabályokat és a kényszerfeltételeket itt írjuk le (nem az adatszerkezetben)

párhuzamos nézőpontok felfedése (parallel aspects)

Metódusok felismerése

szuper-események hatása az álentítésra

opcionális hatások (esemény altípusok) -> ECD
ortogonális nézet

Entitás viselkedés modell mint módszer

A diagram levelei: metódus hívás, amely egy bizonyos végállapotot eredményez

A metódust:

az entitás viselkedés modell (EBM) és az ECD együtt, ortogonális szemszögből reprezentálja

Elnevezés

ECD-n: esemény név + objektum név

Entitás viselkedés modell mint módszer

2. menet

Az adatszerkezet hierarchiában

felülről lefelé : a kivételes helyzeteket keressük,
amit eddig nem modelleztünk

alternatív és a korai entitás halál

backtracking (Posit, Admit, Quit)

OO implementációs lehetőségek

Objektumok, EBM, ECD

esemény adat-feldolgozási eljárása egy optimalizált
objektum-orientált specifikáció

objektumok közötti kommunikáció "munkaterületen"
keresztül

Események átküldésének módja?

körmérközésszerűen egyik entitástól a másikig
esemény-kezelő objektum bevezetése

Milyen legyen az üzenetközvetítési protokoll (message passing mechanism) ?

OO implementációs lehetőségek

Iteráció kezelése (ECD)

A főentitás egy metódusán belüli ciklus

Az alentitások osztályának definiálása és ebben egy osztály szintű metódus:

az összes alentításra

az adott főentitás összes alentítására

Az alentitás 'osztály szintű' metódusaként

Sikeres végrehajtás kezelése

két fázisú commit megvalósítása

Esemény feldolgozás előtti kommunikáció kezelése

ad-hoc üzenetközvetítési protokoll

két fázisú commit

OO implementációs lehetőségek

Egy esemény többször is érinti ugyanazt az entitás példányt

rekord zárási probléma

Üzenetek előválogatása

ez az esemény érintette-e már ezt az entitás példányt?

ez-e az utolsó alkalom?

Megvalósítás a PDI-ban (folyamat adatkapcsolat)

Eseménykezelő (esemény diszpécser)

adatbázis-kezelést, adatfeldolgozást helyezünk el az eseménykezelőben

hagyományos programozási megoldáshoz vezet

egy adott objektum "munkaterületén" keresztül kommunikálnak az objektumok

Információrendszer egy OO környezetben

OO program

viszonylag kevés objektum

különböző szinteken jelennek meg - fogalmi és megvalósítás - egyes objektumok másokat megvalósítása

objektumok közötti kölcsönhatás különböző szintek között (fogalmi és megvalósítás), a fogalmi objektum belügye

Információrendszer

sok objektum (esetleg több millió)

kölcsönhatás azonos (fogalmi) szinten történik, kommunikáció nyilvános

a felhasználók által kezdeményezett és csak a szervezet biztonsági politikája korlátozza

a kommunikációt kézben kell tartani

Információrendszer egy OO környezetben

Fogalmi szint: Esemény

sikeresen végrehajtandó egység

Implementációs szint: DBMS commit unit

OO környezet: Esemény kezelő objektum

Hatékony háttértárolási mechanizmus

objektumok lokális változóinak értékét meg kell őrizni

hatékony beolvasás, csoportosítás, indexelés,
fizikai szintű megoldások (DBMS cluster)