

Adatbázis-kezelő rendszerek I.



KITERJESZTETT EGYED-KAPCSOLAT MODELL

(Enhanced Entity-Relationship model)

Elmasri & Navathe: Fundamentals of Database Systems

Enhanced Entity-Relationship (EER) Model

Kiterjesztett egyed-kapcsolat modell



- EER: Enhanced (Extended) Entity-Relationship Model
- Az EER modell fogalmai:
 - Tartalmazza az egyed-kapcsolat modell minden elemét
 - További fogalmak:
 - ✦ alosztály/főosztály
 - ✦ specializáció/általánosítás
 - ✦ kategória (UNIO típus)
 - ✦ attribútum és kapcsolat öröklődése
- Az EER modell fogalmaival bizonyos problémakörök pontosabban, komplexebben modellezhetőek.
 - Az EER modell tartalmaz néhány objektum-orientált alapelvet, mint például az öröklődés.

Alosztályok és főosztályok



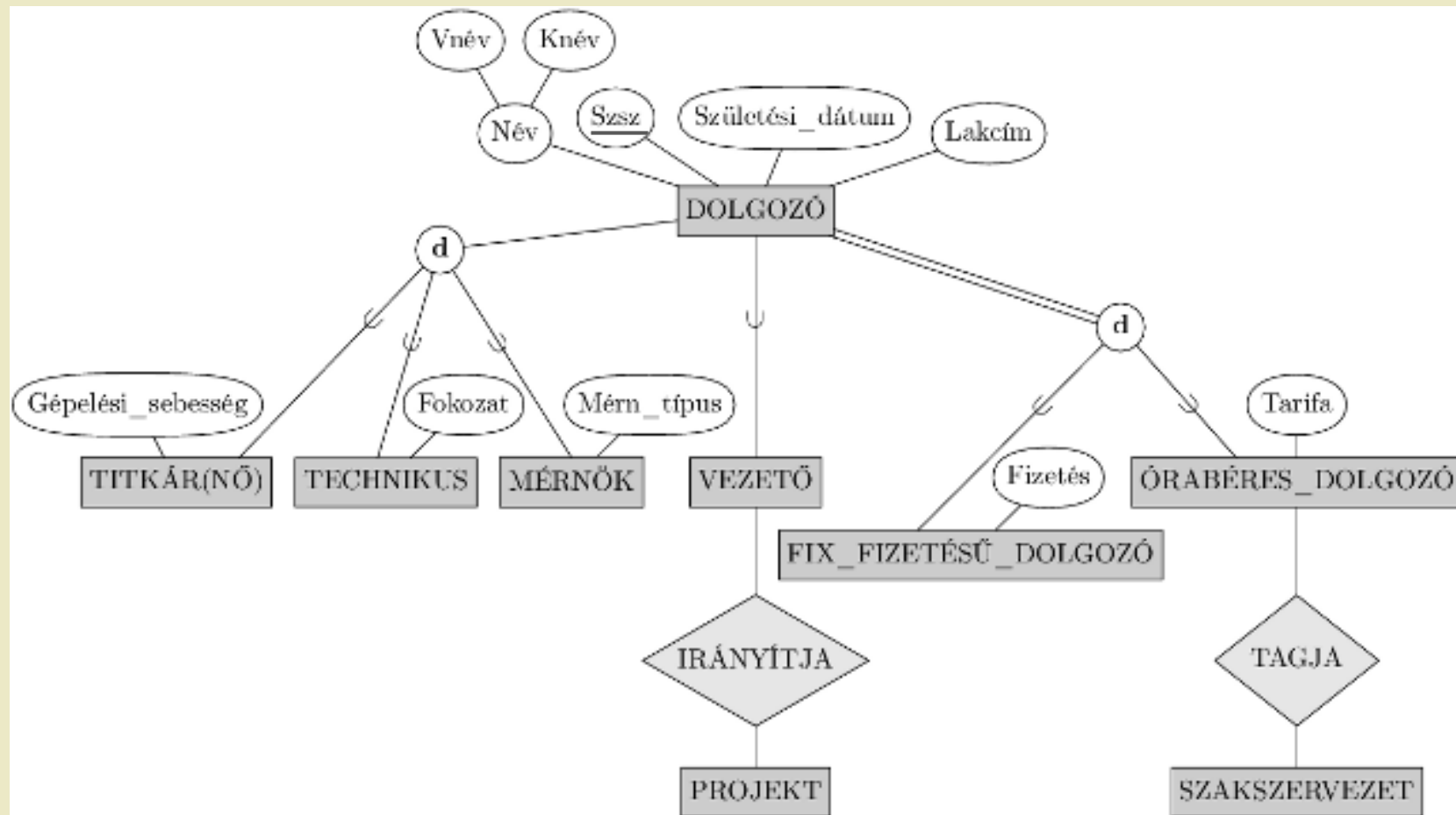
- Az egyedtípus egyedei esetenként további *alcsoportokra* oszthatóak
 - Például: a DOLGOZÓ tovább csoportosítható:
 - ✦ TITKÁRNŐ, MÉRNÖK, TECHNIKUS
 - az alkalmazott munkáján alapulva
 - ✦ VEZETŐ
 - azon alkalmazottak, akik valamely projektet irányítanak
 - ✦ FIX_FIZETÉSŰ_DOLGOZÓ, ÓRABÉRES_DOLGOZÓ
 - a fizetési mód alapján
- A kiterjesztett egyed-kapcsolat diagram kiterjeszti az egyed-kapcsolat diagramot, úgy hogy további alcsoportokat is megjelenít. Ezek az *alosztályok*, *altípusok*.

Alosztályok és főosztályok (2)




- **Alosztály:** Az egyedtípus egymástól elkülönülő egyedeinek halmaza.
- **Főosztály:** egy egyedtípus, melynek egyedei alcsoportokba sorolhatók.
- **Főosztály/alosztály kapcsolat:** kapcsolat egy főosztály és annak bármely alosztálya között.
- Például:
 - Az előző dián említett csoportok a DOLGOZÓ egyedtípus egyedeinek részhalmazai lehetnek.
 - Mindegyik **alosztálya** a DOLGOZÓ főosztálynak
 - A DOLGOZÓ **főosztálya** ezen alosztályoknak
 - **Főosztály/alosztály kapcsolatok:**
 - ✦ DOLGOZÓ/TITKÁRNŐ
 - ✦ DOLGOZÓ /TECHNIKUS
 - ✦ DOLGOZÓ /VEZETŐ
 - ✦ ...

Alosztályok és főosztályok az EER diagramon



Alosztályok és főosztályok (3)



- Ezen kapcsolatokat szokás **IS-A kapcsolatoknak** nevezni. 
 - SECRETARY IS-A EMPLOYEE, TECHNICIAN IS-A EMPLOYEE,
- Megj: egy egyed, amely tagja valamely alosztálynak és a főosztálynak is, ugyanazt a valós világbeli egyedet jelöli mind a két helyen.
 - Az alosztály tagja ugyanaz az egyed egy *megkülönböztetett speciális szerepben*.
 - Egy egyed nem létezhet úgy, hogy az csak valamely alosztály tagja, mert a főosztálynak is tagja kell, hogy legyen.
 - A főosztálybeli egyed, nem szükségszerűen van besorolva az alosztály(ok)ba (opcionális).
- Az alosztályok speciális kapcsolattípusokban vehetnek részt.

Attribútum öröklődés a Főosztály/Alosztály kapcsolatokban



- Egy egyed, amely az alosztály tagja örökli:
 - főosztálybeli egyedként a főosztály összes *attribútumát*
 - főosztálybeli egyedként a főosztály összes *kapcsolatát*
- Példa:
 - A TITKÁRNŐ (úgy mint a TECHNIKUS és a MÉRNÖK is) örökli a Név, a Szsz, ... attribútumokat a DOLGOZÓ egyedtípustól.

Specializáció



- A **specializáció** egy **folyamat**, melynek során egy főosztály alosztályainak halmazát határozzuk meg.
 - *Fentről lefelé (top down) haladó koncepcionális finomítás (szétbontás)*
- Az alosztályok halmazának kialakítása a főosztálybeli egyedek néhány megkülönböztető jellemvonalán alapul.
 - Példa: a {TITKÁRNŐ, MÉRNÖK, TECHNIKUS} egy specializációja a DOLGOZÓ egyedtípusnak a *foglalkozás típusa* alapján.
- Egy főosztálynak számos specializációja is lehetséges
 - Példa: A DOLGOZÓ egyedtípus másik lehetséges specializációja a *fizetési mód* alapján: {FIX_FIZETÉSŰ_DOLGOZÓ, ÓRABÉRES_DOLGOZÓ}

Általánosítás

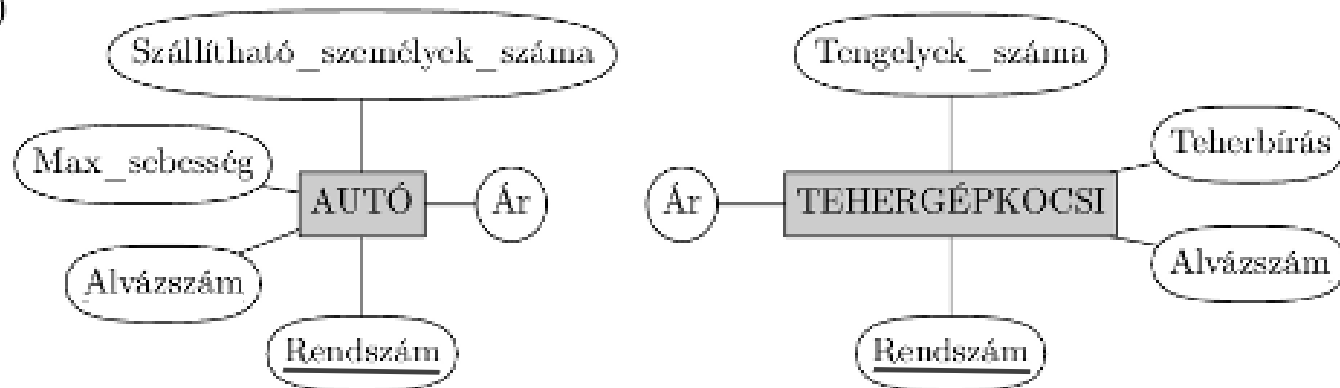


- **Az általánosítás** a specializációval *ellentétes* folyamat
 - Alulról felfelé haladó (*bottom up*) koncepcionális *egyesítő* eljárás
- Közös jellemzőkkel rendelkező osztályok egy főosztályba történő általánosítása
 - az eredeti osztályok a főosztály alosztályaivá válnak
- Példa: Az AUTÓ és TEHERGÉPKOCSI általánosítása a JÁRMŰ egyedtypusba
 - Mind az AUTÓ, mind a TEHERGÉPKOCSI egyedtypusok a JÁRMŰ főosztály alosztályaivá válnak.
 - Az {AUTÓ, TEHERGÉPKOCSI} -t tekinthetjük a JÁRMŰ specializációjának is (felülről lefelé).
 - Analóg módon, tekinthetjük a JÁRMŰ-t az AUTÓ és a TEHERGÉPKOCSI egyedtypusok általánosításának is (alulról felfelé).

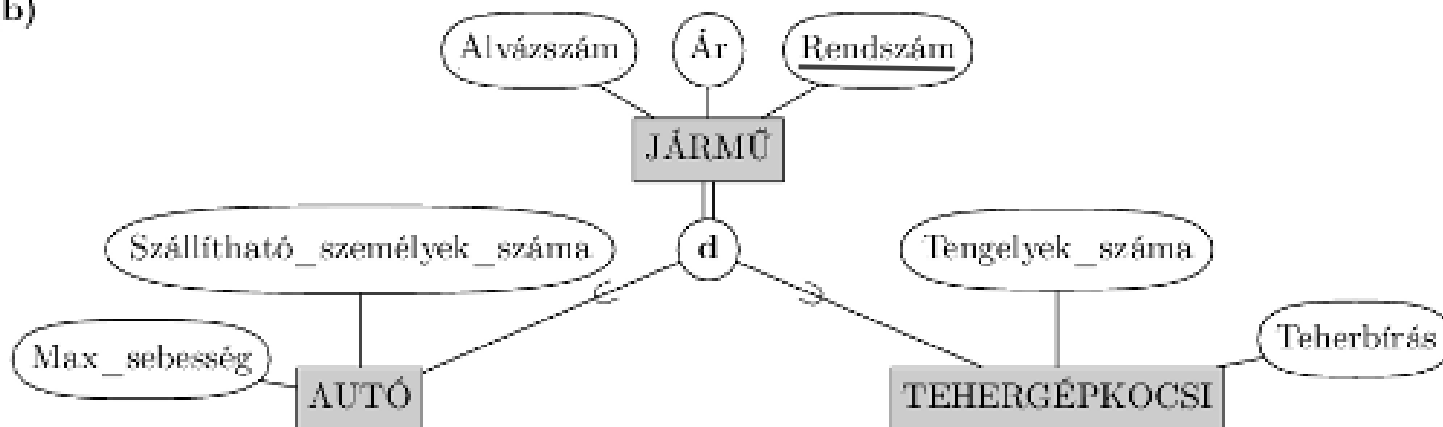
Általánosítás – Példa



(a)



(b)



Főosztály/Alosztály kapcsolatok korlátozásai



- Két alapvető korlátozás létezik:
 - **Elkülönülő korlátozás (Disjointness Constraint)**
 - ✦ Azt határozza meg, hogy egy főosztálybeli egyed legfeljebb hány alosztálynak lehet a tagja (egy vagy több).
 - **Teljességi korlátozás (Completeness Constraint)**
 - ✦ Azt határozza meg, hogy a főosztály minden egyes egyedének tartoznia kell-e legalább egy alosztályhoz, vagy sem.

Főosztály/Alosztály kapcsolatok korlátozásai (folyt.)



- **Elkülönülő korlátozás (Disjointness Constraint):**

- **Elkülönülő (disjoint):**

- ✦ egy főosztálybeli egyed legfeljebb egy alosztály tagja lehet
- ✦ jelölése az EER diagramon: ***d***

- **Átfedő (overlapping):**

- ✦ ugyanazon egyed tagja lehet több alosztálynak is
- ✦ jelölése az EER diagramon: ***o***

Főosztály/Alosztály kapcsolatok korlátozásai (folyt.)



- **Teljességi korlátozás (Completeness Constraint):**
 - **Teljes (totális, total):** a főosztály minden egyes egyedének tagja kell, hogy legyen legalább 1 alosztálynak is.
 - ✦ Jelölése az EER diagramon: **dupla vonal**
 - **Részleges (parciális, partial)** megengedi, hogy egy főosztálybeli egyed ne tartozzon egyetlen alosztályhoz sem.
 - ✦ Jelölése az EER diagramon: **szimpla vonal**

Főosztály/Alosztály kapcsolatok korlátozásai (folyt.)

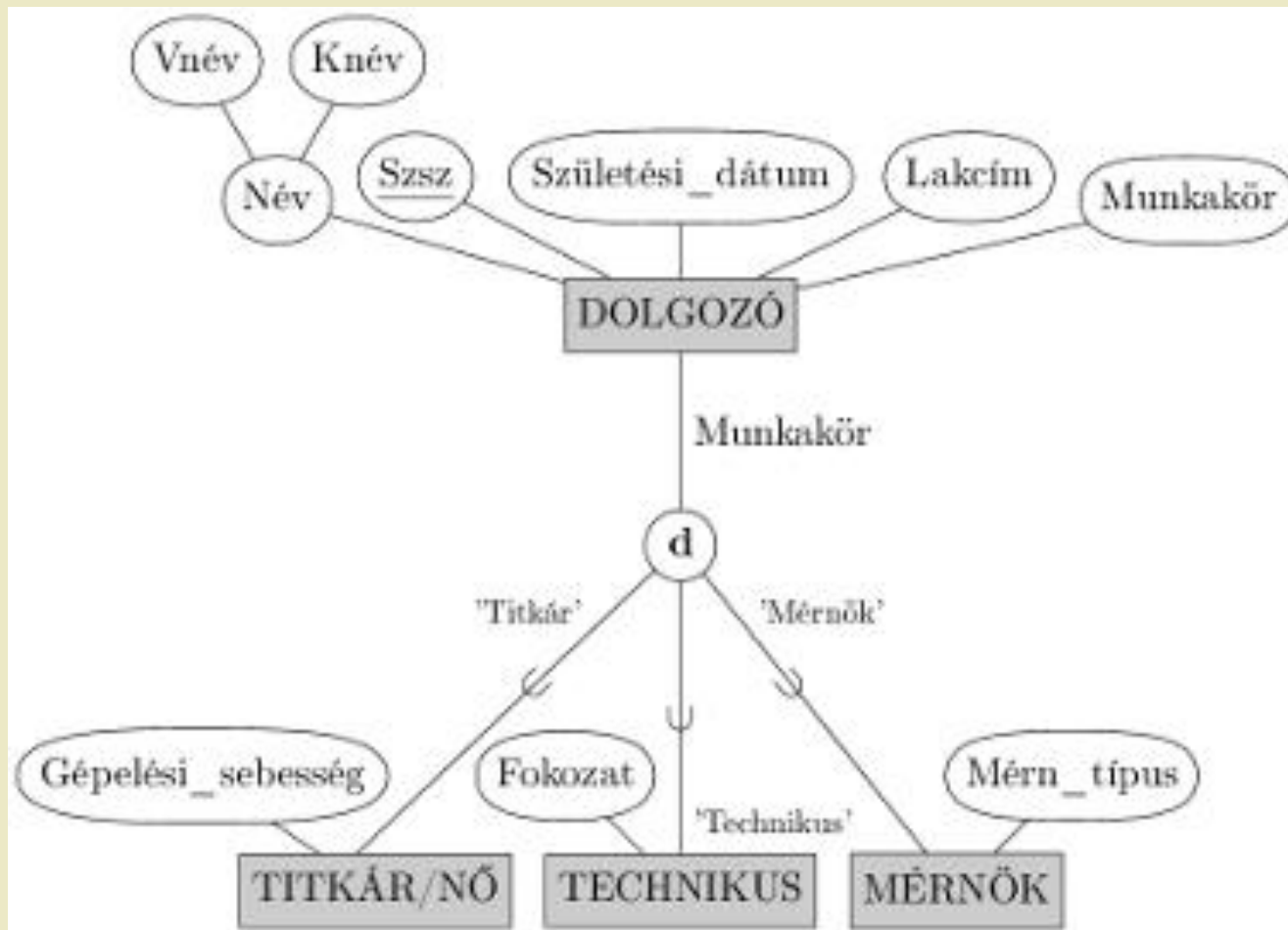


Ennek alapján 4-féle specializációnk/általánosításunk lehet:

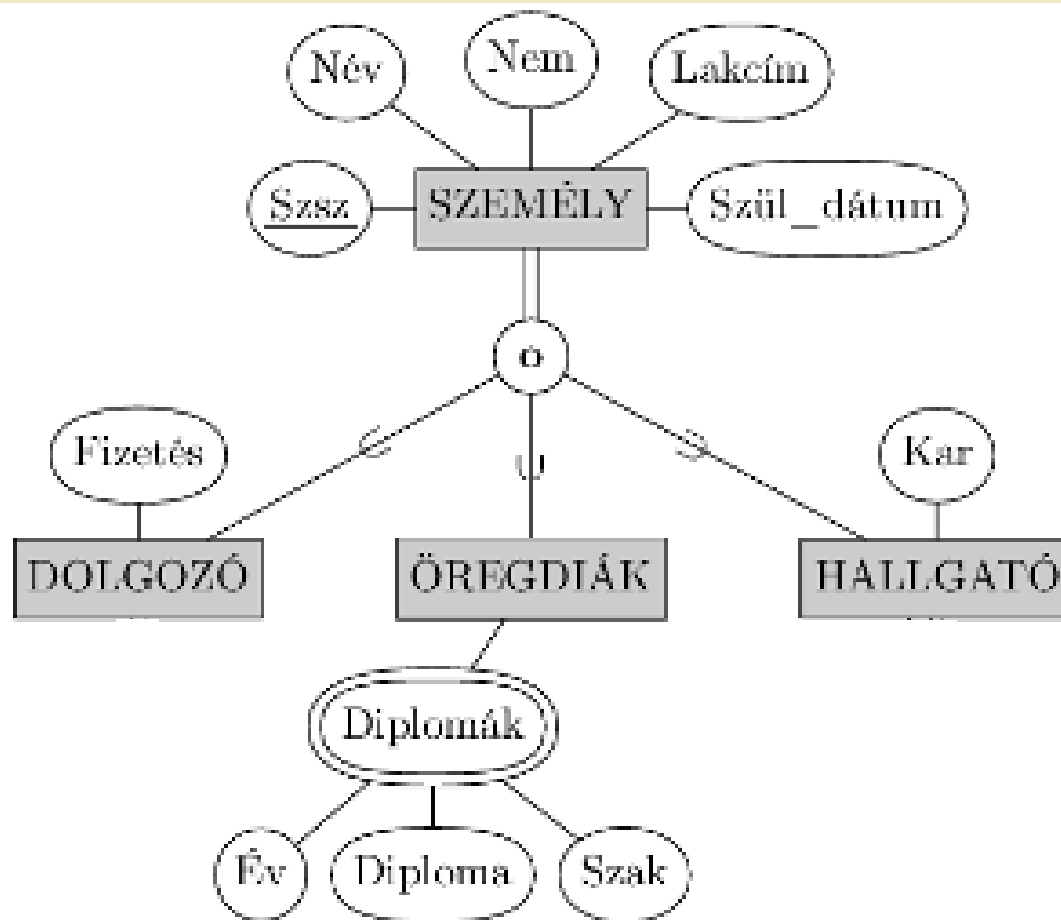
- Elkülönülő teljes (disjoint, total)
- Elkülönülő részleges (disjoint, partial)
- Átfedő teljes (overlapping, total)
- Átfedő részleges (overlapping, partial)

Megjegyzés: Az általánosítás általában teljes, mert a főosztály az alosztálytól származik.

Példa – elkülönülő részleges specializáció



Példa – átfedő teljes specializáció

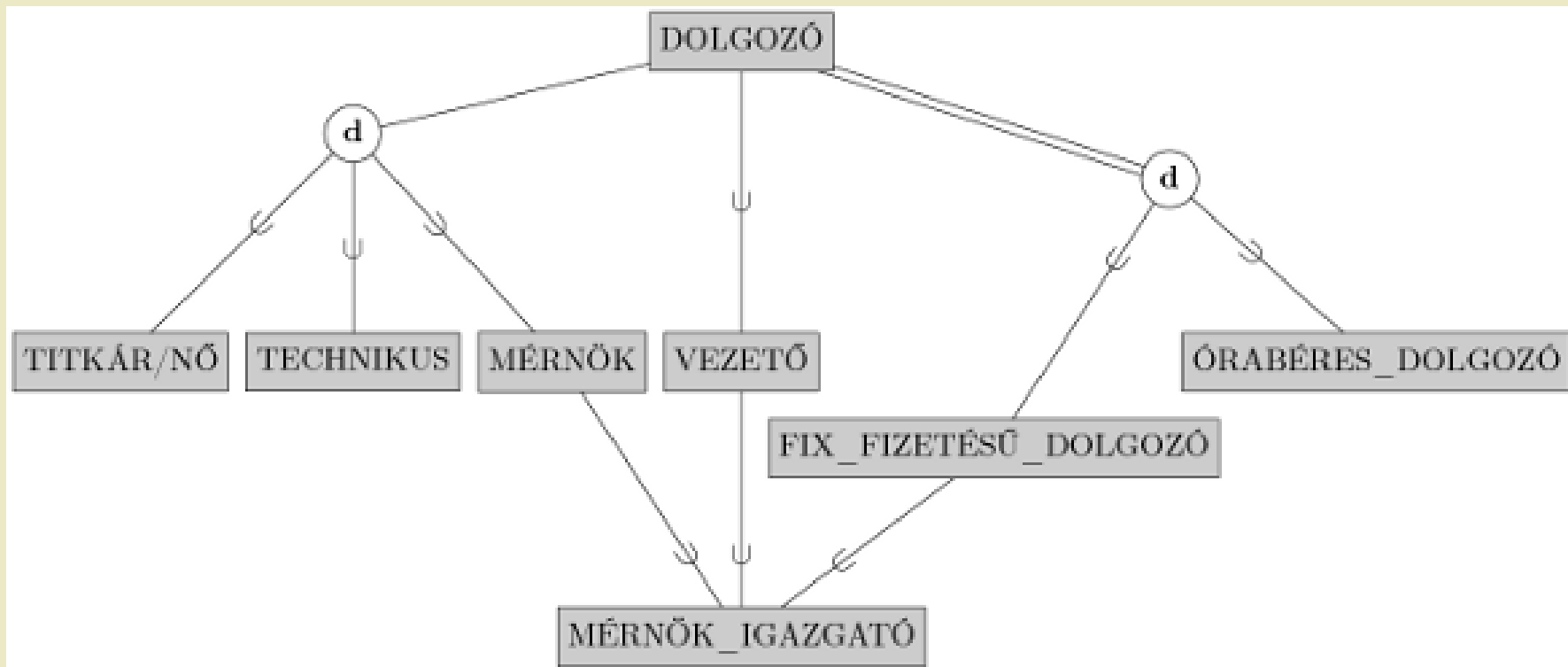


Hierarchia, osztott alosztályok



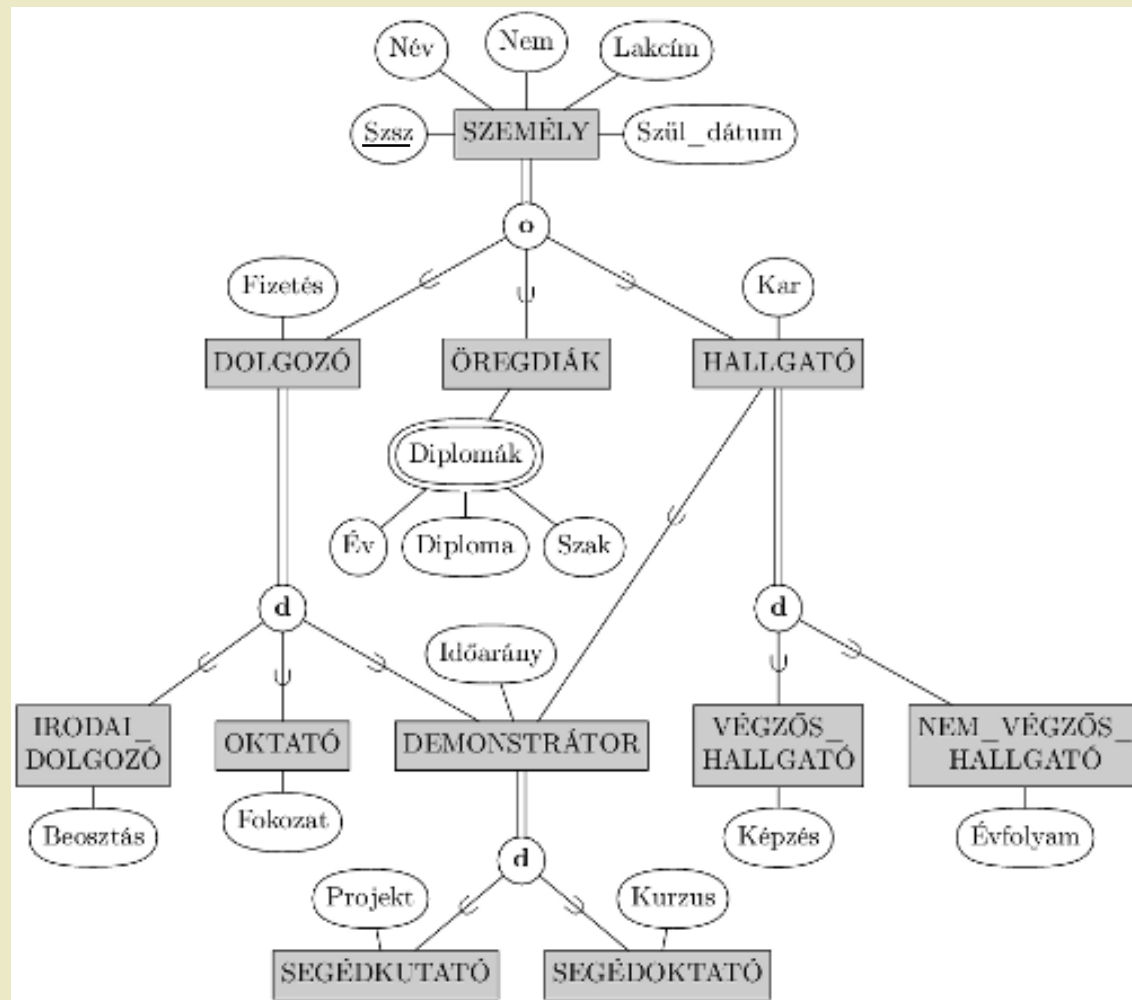
- Egy alosztálynak további alosztályai lehetnek, *hierarchiát* vagy *rácsot* alkotva.
- **Hierarchia:** korlátozza, hogy minden egyes alosztálynak csak egy szülő osztálya lehet (*egyszerű öröklődés*)
- Egy **rácsban**, egy alosztálynak több szülő osztálya is lehet (*többszörös öröklődés*) (**osztott alosztályok**)

Példa



Osztott alosztály: MÉRNÖK_IGAZGATÓ

Példa – rács (UNIVERSITY)



Kategória (UNIO típus)



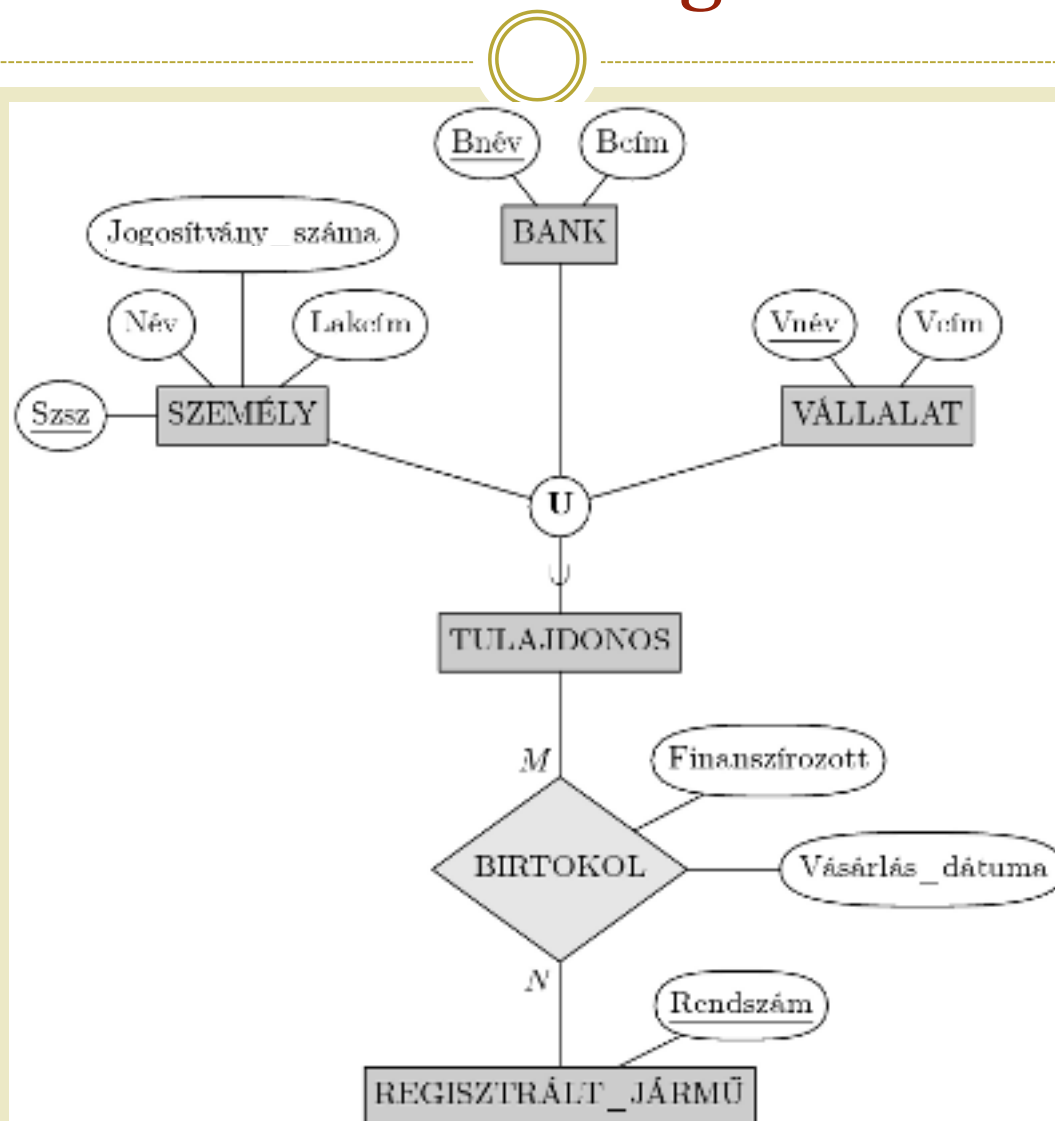
- Néhány esetben, olyan főosztály/alosztály kapcsolatot szükséges modellezni amelyben **több mint egy főosztály** van.
 - A főosztályok különböző egyedtípusokat jelölnek.
- **Unio**: egy főosztály/alosztály kapcsolat amelyben több főosztály van.
- **Unio típus** vagy **kategória**: egy olyan alosztály, melynek különböző főosztályai vannak.

Kategória (UNIO típus) (folyt.)



- Példa: egy jármű regisztrációs adatbázisban egy járműtulajdonos lehet egy SZEMÉLY, egy BANK (pl. zálogjog) vagy egy VÁLLALAT.
 - *Kategóriának* (UNIO típusnak) nevezzük a létrehozott TULAJDONOS egyedtípust, amely a három főosztálynak (SZEMÉLY, BANK, és VÁLLALAT) az *uniója*.
 - A kategória egyedeinek **legalább egy** főosztályhoz is tartozniuk kell.
- Eltérés a megosztott alosztálytól:
 - A megosztott alosztály *részhalmaza* a főosztályok metszetének.
 - Egy megosztott alosztálybeli egyednek **minden** főosztályban léteznie kell.

Példa – Kategória



Alternatív jelölések

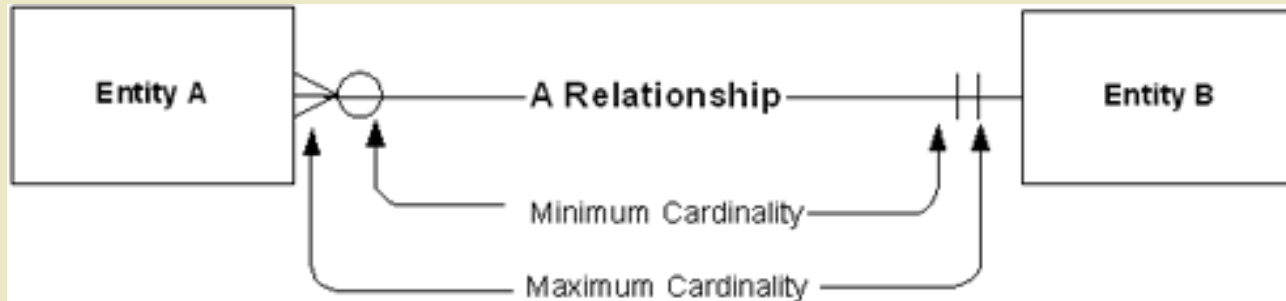


Alternatív jelölések

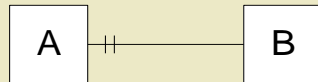


- Az (E)ER diagram az adatbázis sémák egy közkedvelt megjelenítési módszere.
- A szakirodalomban és az adatbázis-tervező és modellező szoftvereszközökben sok más jelöléssel is találkozhatunk.
- Például:
 - **Chen jelölés:** alap jelölés
 - **Varjúlábas jelölés**
 - ✦ Az egyedeket dobozokként jelöli, a kapcsolatokat a dobozokat összekötő vonalakként. A vonalak végén különböző formák jelölik a kapcsolattípusok kardinalitását.
 - **UML**
 - ✦ Az UML osztály diagramja hasonlít az az ER diagramhoz, de van számos eltérés is felfedezhető. Számos kereskedelmi forgalomban lévő tervezőszoftver használja.

Varjúlábas jelölés



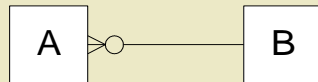
Az A egyedtípus minden egyes előfordulása legalább 0 legfeljebb 1 B-beli egyeddel kapcsolatban áll.



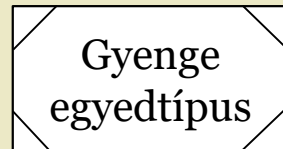
A B egyedtípus minden egyes előfordulása pontosan 1 A-beli egyeddel áll kapcsolatban.



Az A egyedtípus minden egyes előfordulása legalább 1 legfeljebb sok B-beli egyeddel áll kapcsolatban.



A B egyedtípus minden egyes előfordulása legalább 0 legfeljebb sok A-beli egyeddel áll kapcsolatban.



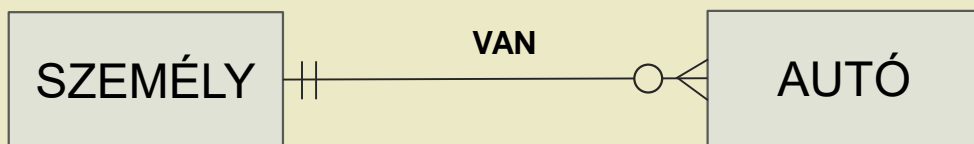
Ekvivalens jelölések



Chen

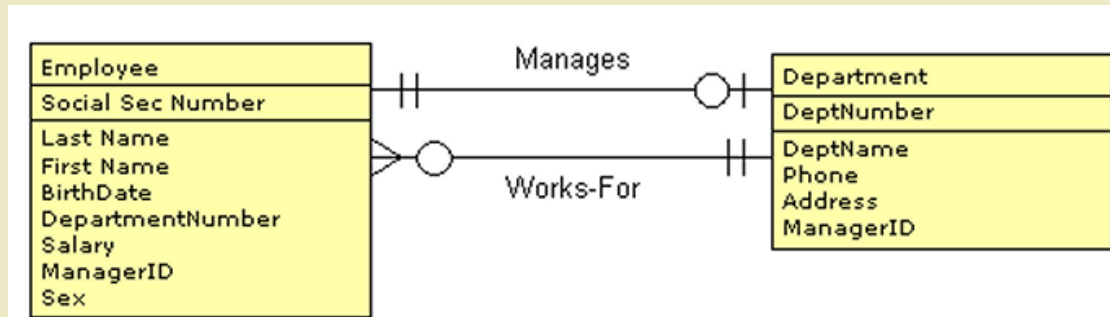
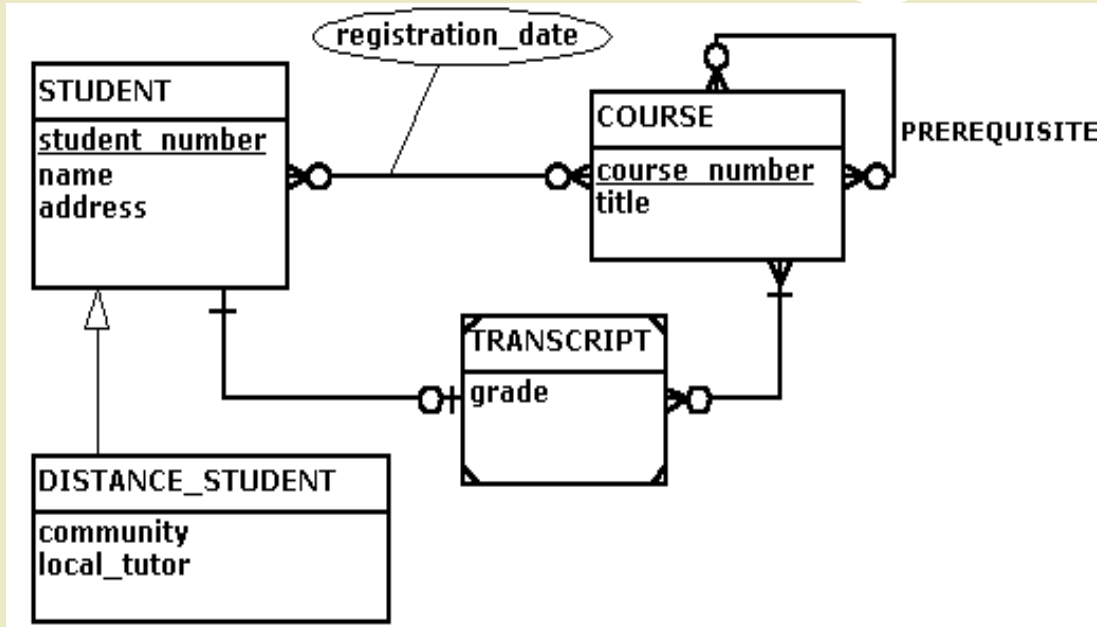


Min-Max jelölés



Varjúláb jelölés

Varjúlábás jelölés – Példa



UML Osztály Diagramok

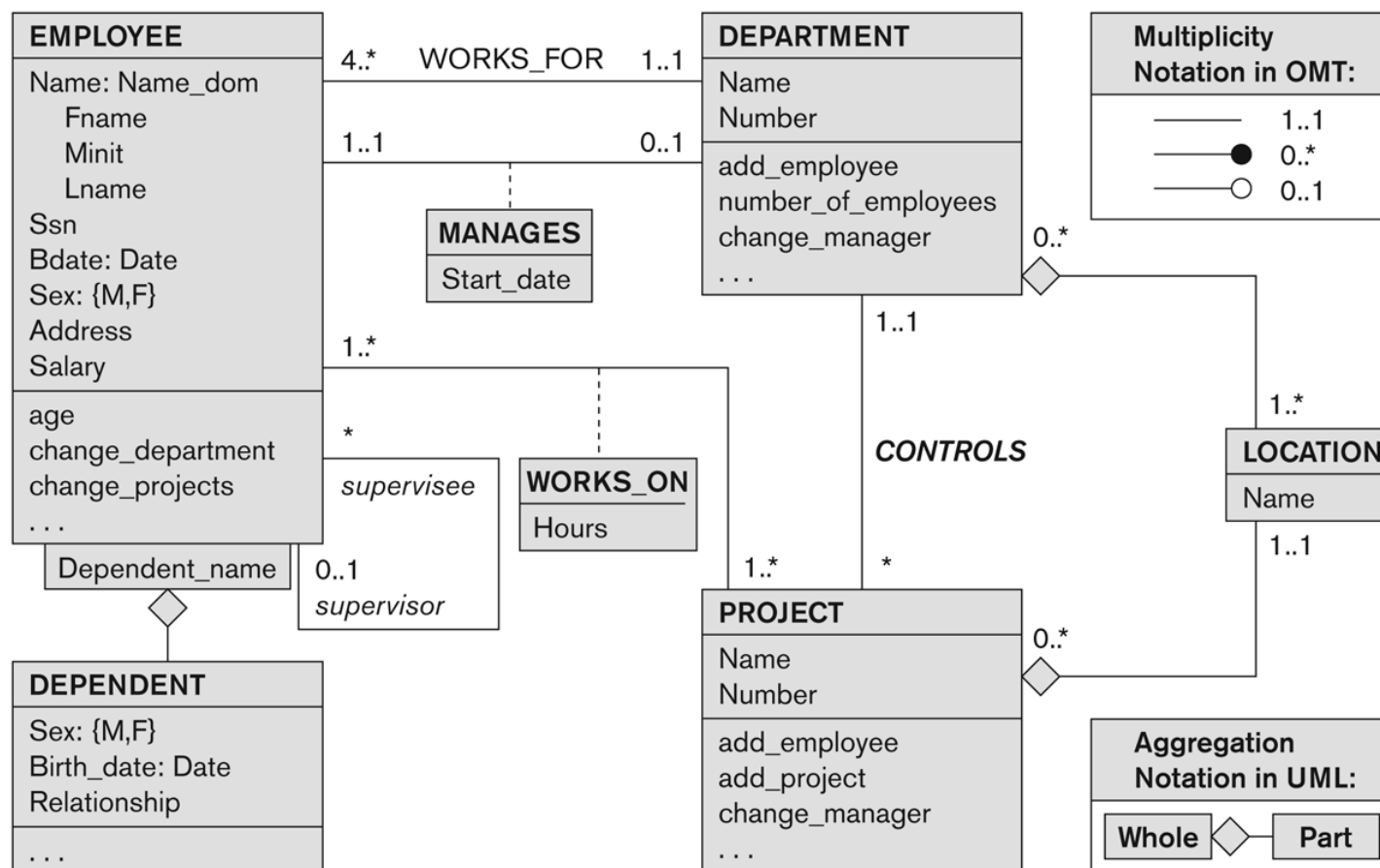


- Az UML osztály diagramok az ER modell fogalmait más módon ábrázolják – számos kereskedelmi tervezőszoftver használja
 - Adatbázis tervezésben és objektum-orientált szoftvertervezésben használatos
- Az UML az osztályokat (hasonlóan az egyedtípusokhoz) lekerekített dobozokban ábrázolja, melyeket 3 részre oszt:
 - Felső rész: egyedtípus (osztály) neve
 - Középső rész: attribútumok
 - Alsó rész: osztályokon végezhető műveletek (a műveletek nem részei az ER modellnek)
- A kapcsolatok (asszociációk) az osztályokat összekapcsoló vonalak

UML osztály diagram a COMPANY adatbázis sémához

Figure 3.16

The COMPANY conceptual schema in UML class diagram notation.



UML Osztály Diagramok (folyt.)



- Az *egyedtypusok* jelölése téglalap, *attribútumok* a téglalapon belül.
- *Bináris kapcsolattypusok*: az egyedhalmazokat összekötő vonalak. A kapcsolattypus neve a vonal mellé írva.
 - Alternatív mód: a *kapcsolattypus* neve írható egy téglalapba is, a kapcsolattypus attribútumaival együtt, és az a téglalap szaggatott vonallal kapcsolódik a kapcsolattypus vonalához.
- *Nem bináris kapcsolattypusok*: rombusz, mint az ER modellben
- Az egyedtypus kapcsolattypusban betöltött *szerepe*: az egyedtypus mellé, a vonalra írva.

UML Osztály Diagramok (folyt.)

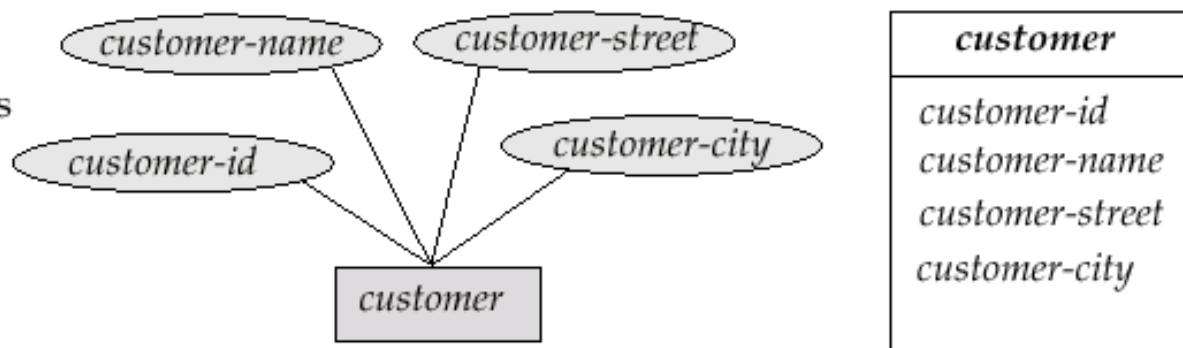


- *Kardinalitás jelölése: $l..h$ formában, ahol l jelöli a minimális és h maximális kardinalitást.*
- **Figyelem: a korlátozások elhelyezése pontosan fordított, mint az ER diagramon!.**
 - A $0..*$ korlátozás az $E2$ oldalon és a $0..1$ korlátozás az $E1$ oldalon azt jelenti hogy minden $E2$ -beli egyed legfeljebb egy kapcsolatban vehet részt, míg minden $E1$ -beli egyed több kapcsolatban is részt vehet.
 - Egyetlen érték, mint például 1 vagy * szintén írható a vonalak mellé. Az 1 érték ekvivalens az $1..1$, a * pedig ekvivalens a $0..*$ jelöléssel.

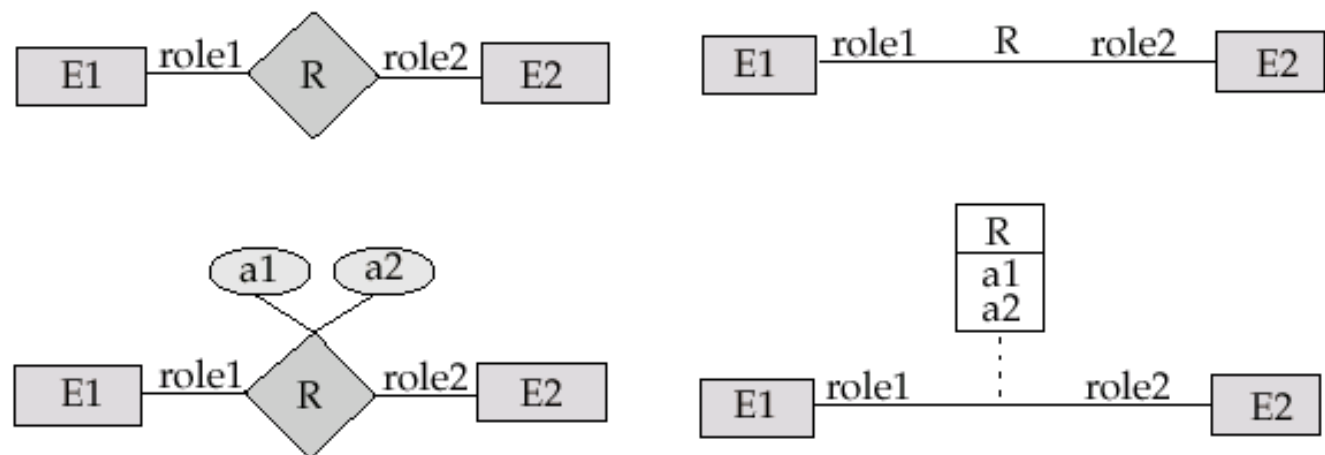
UML Osztály Diagram jelölése



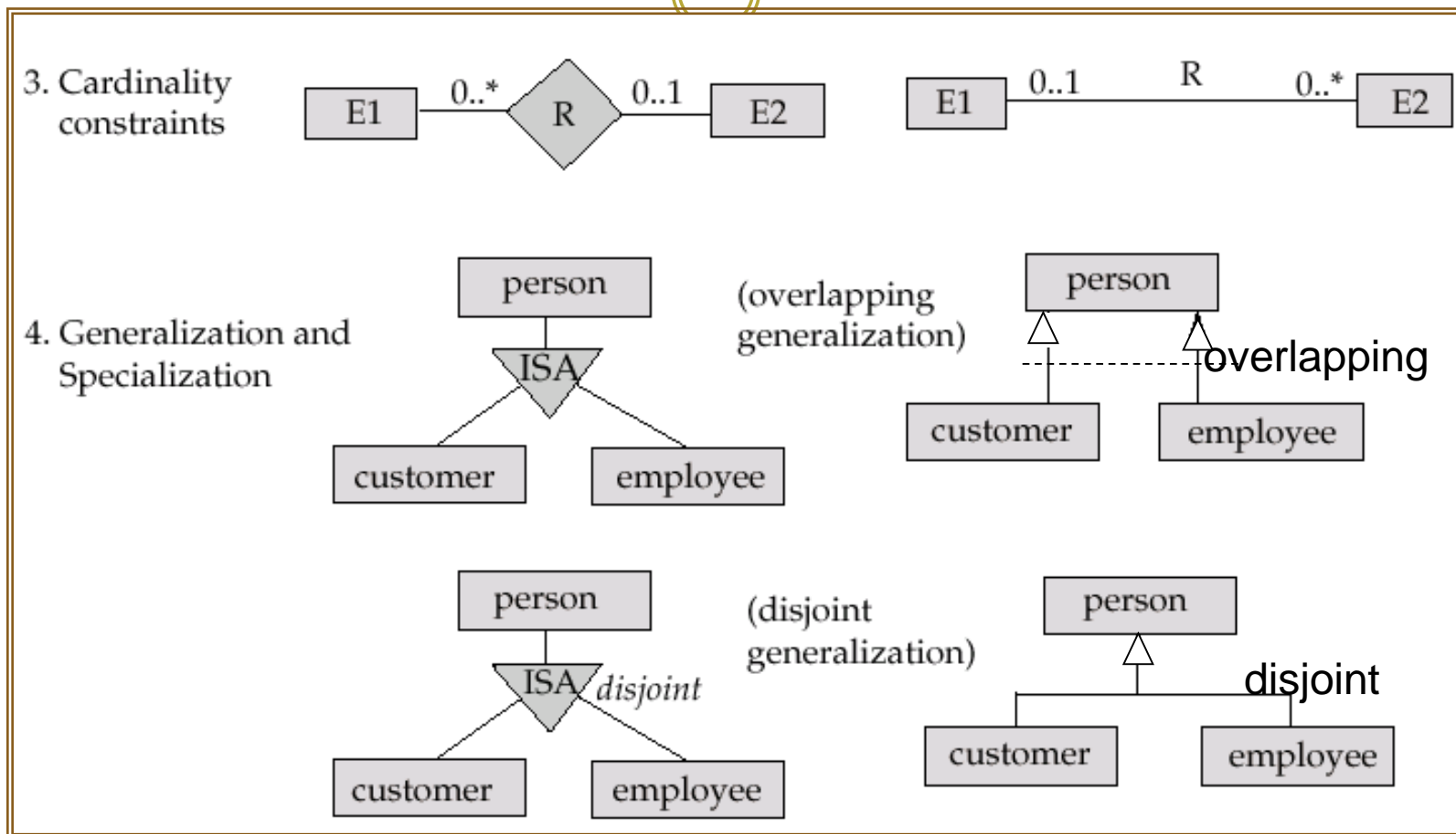
1. Entity sets and attributes



2. Relationships



UML Osztály Diagram jelölése (folyt.)

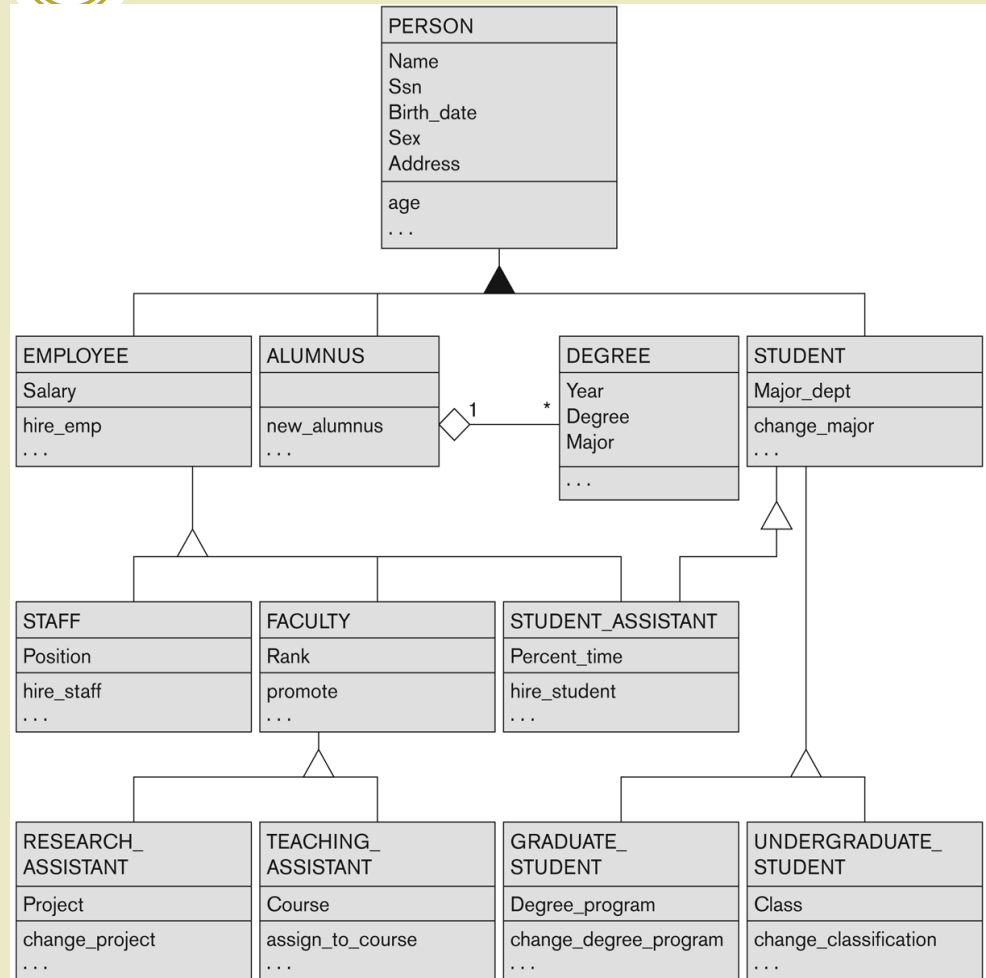
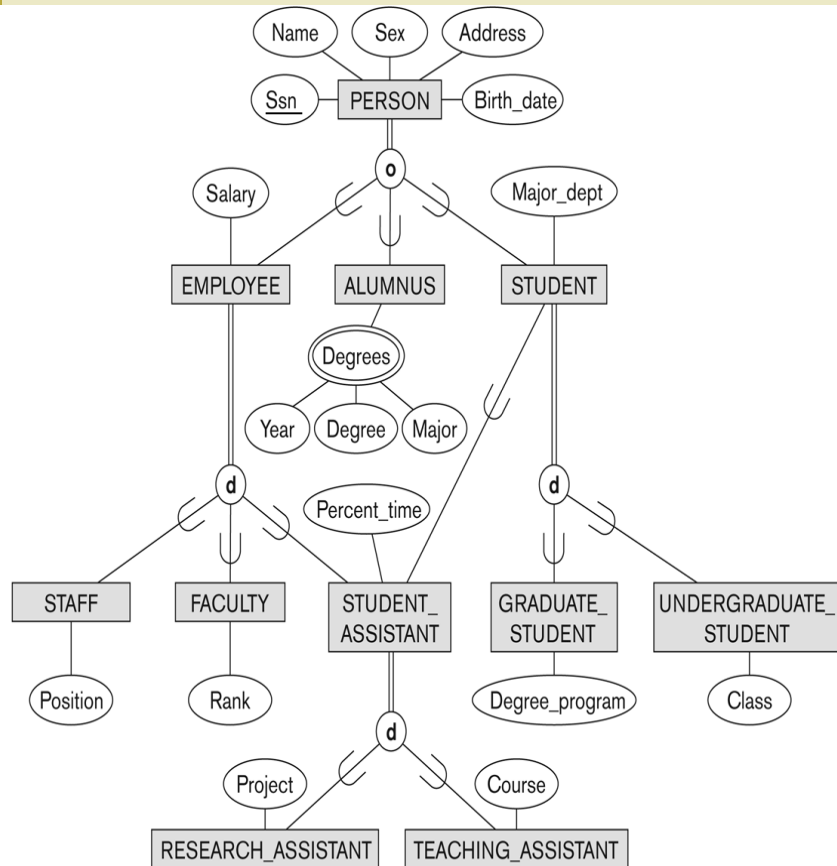


?!

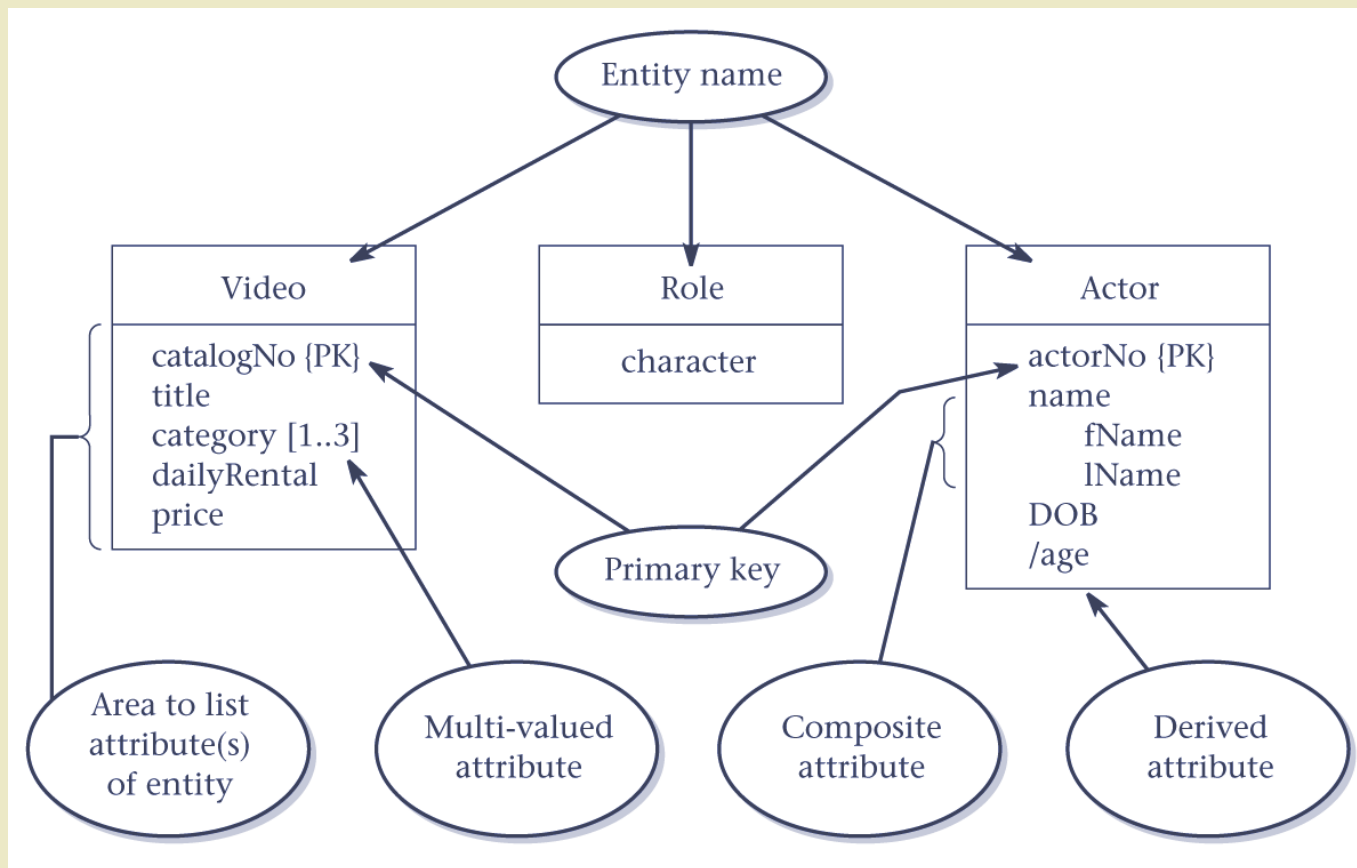
*a kardinalitási korlátozás fordított jelölése

*Az általánosítás lehet összevont vagy elkülönülő nyíl is az elkülönülő/átfedő korlátozástól függetlenül.

UML példa a Specializáció/Általánosítás jelölésére



Egyéb alternatív jelölések



Egyéb alternatív jelölések



Alternative ways to represent multiplicity constraints	Meaning
0..1	Zero or one entity occurrence
1..1 (or just 1)	Exactly one entity occurrence
0..* (or just *)	Zero or many entity occurrences
1..*	One or many entity occurrences
5..10	Minimum of 5 up to a maximum of 10 entity occurrences
0, 3, 6–8	Zero or three or six, seven, or eight entity occurrences

Egyéb alternatív jelölések

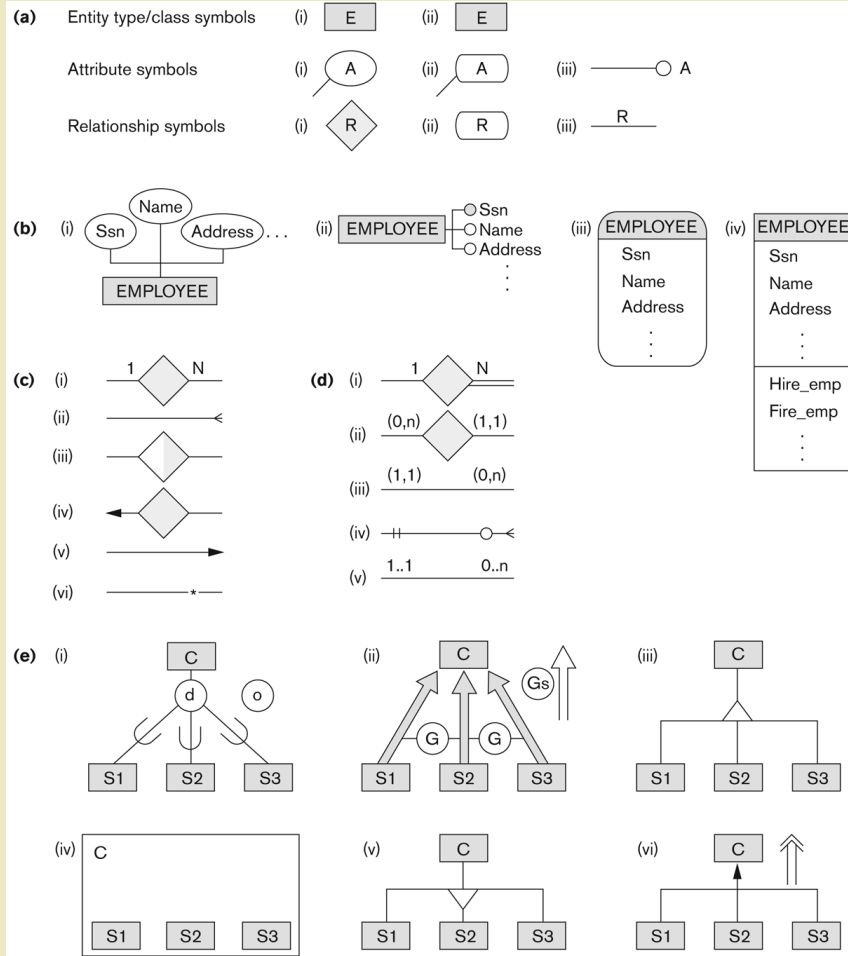
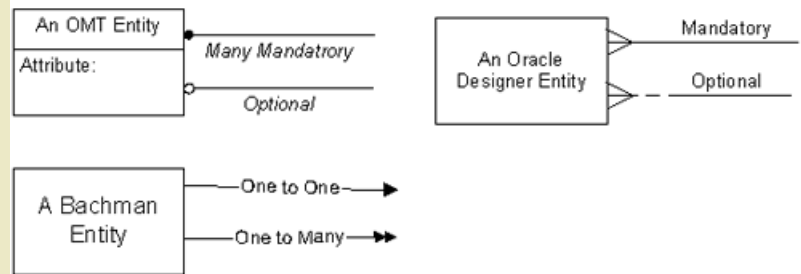


Figure A.1

Alternative notations. (a) Symbols for entity type/class, attribute, and relationship. (b) Displaying attributes. (c) Displaying cardinality ratios. (d) Various (min, max) notations. (e) Notations for displaying specialization/generalization.

Other Notations are less intuitive and cluttered



Notation	Information Engineering	Barker Notation	IDEF1X	UML
Multiplicities:				
- Zero or one				
- One only				
- Zero or more				
- One or more				
- Specific range	N/A	N/A	N/A	
Attributes:				
Names	N/A	Attribute Name: Type	attribute-name: Type	attributeName: Type
Primary key/unique identifier	N/A	# Attribute Name		attributeName <<PK>> {order=#}
Foreign key	N/A	N/A	attribute-name (FK)	attributeName <<FK>> {to=tablename}
Associations:				
Labels				
Entity roles	N/A	N/A	N/A	
Subtyping				
Aggregation				
Composition				
Or Constraint		N/A	N/A	
Exclusive Or (XOR) Constraint			N/A	

Problémák az ER modellel



Csapdák



- Az adatmodell tervezése közben számos probléma felmerülhet. Ezeket **kapcsolati csapdák**ként ismerhetjük.
- A kapcsolati csapdáknak 2 fő típusa van:
 - **Legyező csapda**
 - **Szakadék csapda (űr)**

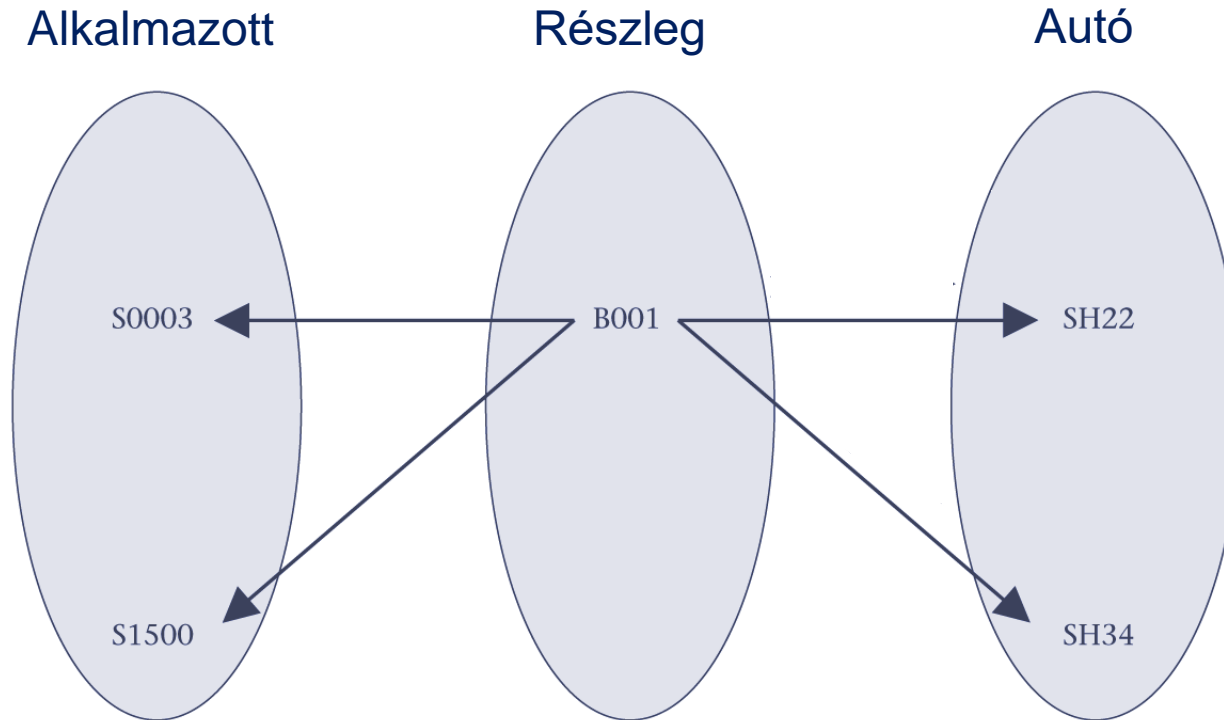
Legyező csapda



- **Legyező csapda:** Akkor következik be, amikor a modell kapcsolatot jelöl egyed típusok között, de az út az egyes egyedek között nem egyértelmű.
- Akkor fordul elő, ha 1:N ágaznak ki egy egyed típusból.
- Példa:
 - nem tudjuk, hogy ki melyik autót használja

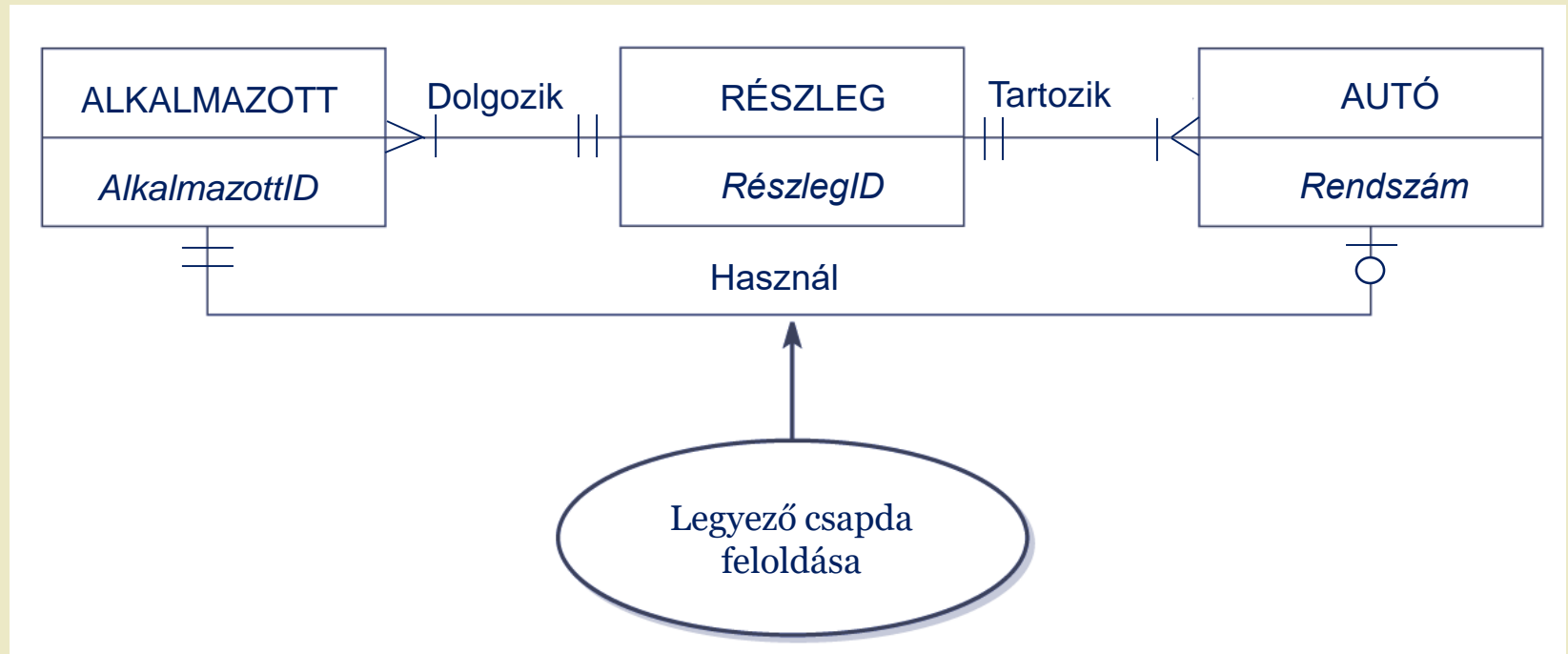


Legyező csapda – példa

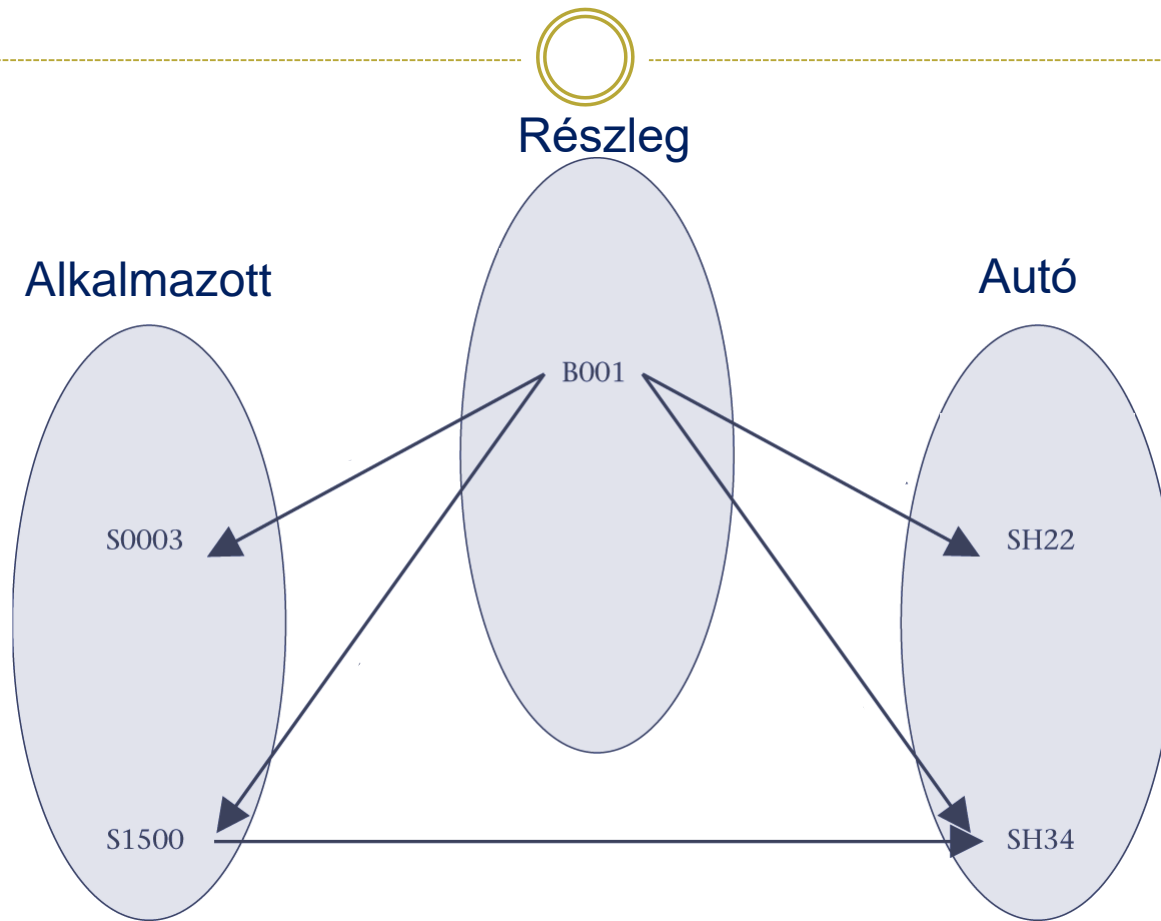


- Nem tudjuk megmondani, hogy melyik munkatárs használj az SH34-es autót.

A legyező csapda feloldása



Feloldott legyező csapda – példa



- Most már meg tudjuk mondani.

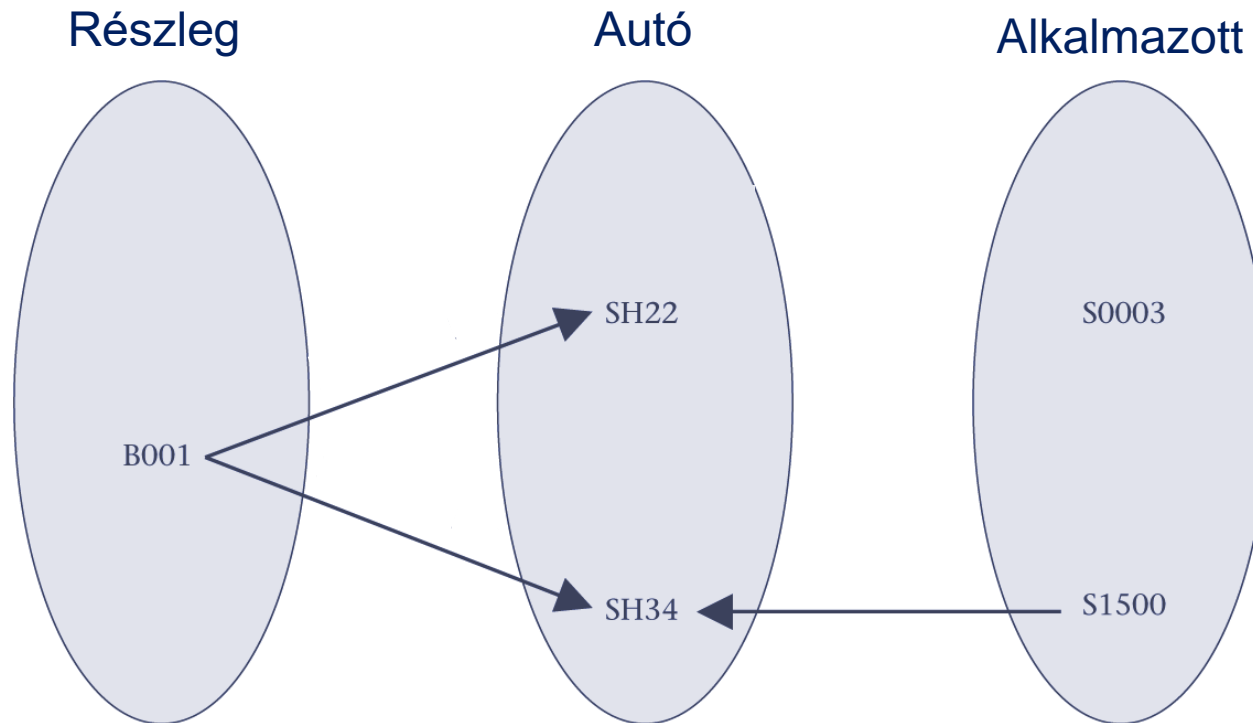
Szakadék csapda



- **Szakadék csapda:** A modell arra utal, hogy az egyedek között van kapcsolat, de az útvonal nem létezik bizonyos egyed előfordulások esetében.
- Ott fordul elő, ahol az egyedeket összekötő útvonal részleges részvételű egyed típusokat tartalmaz.
- Példa:

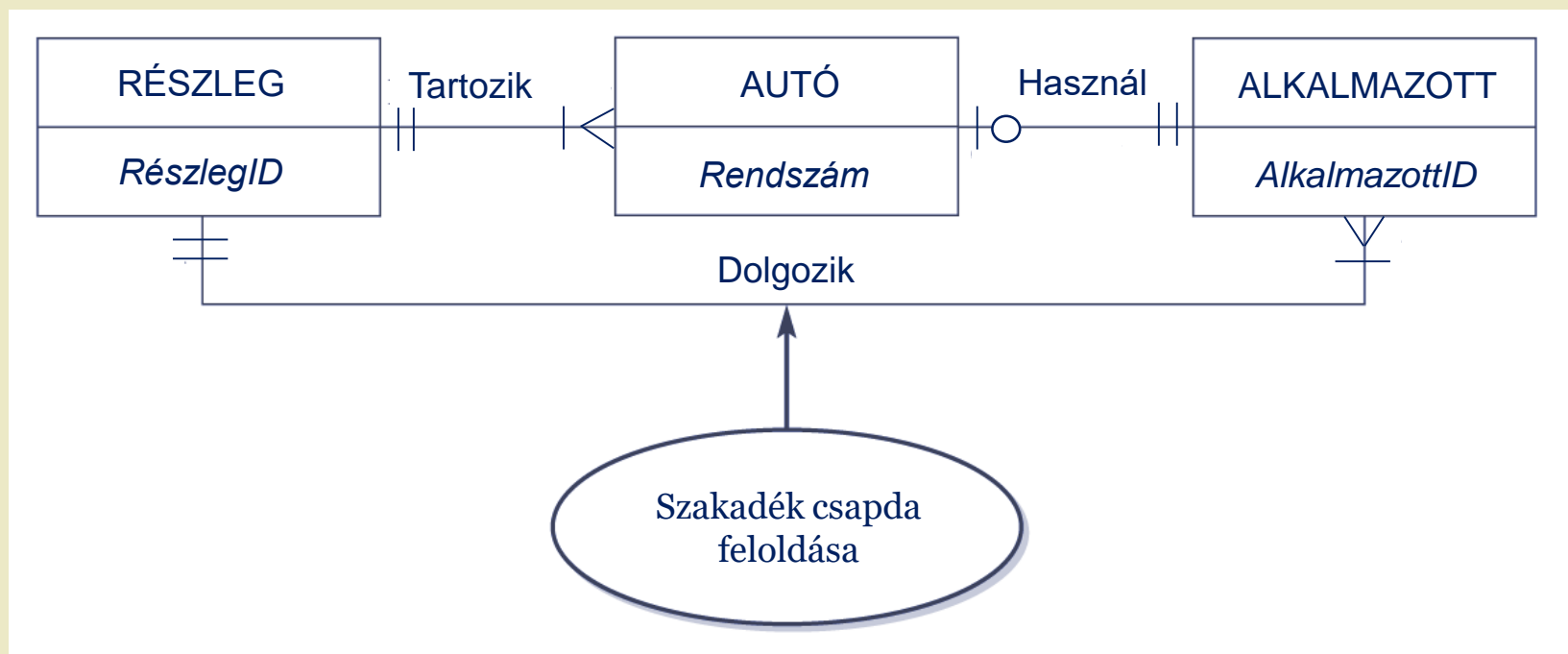


Szakadék csapda – példa

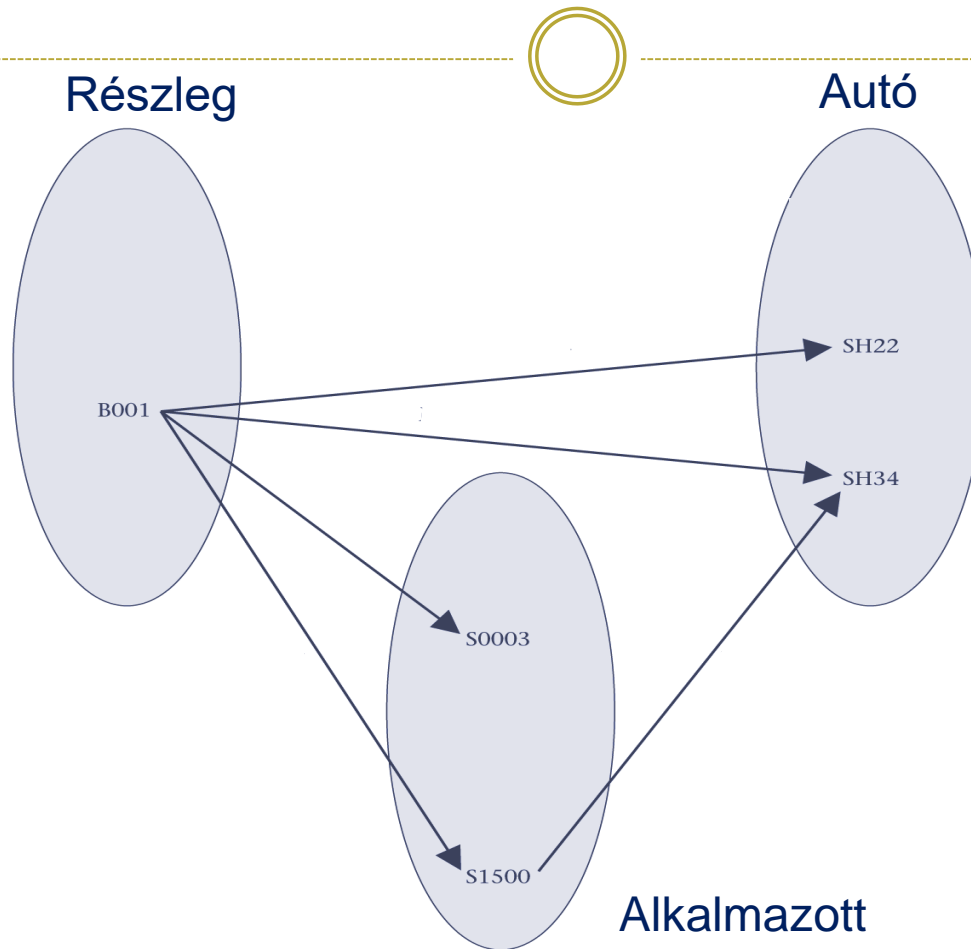


- Nem tudjuk megmondani, hogy az S0003 jelű alkalmazott melyik részlegen dolgozik.

A szakadék csapda feloldása

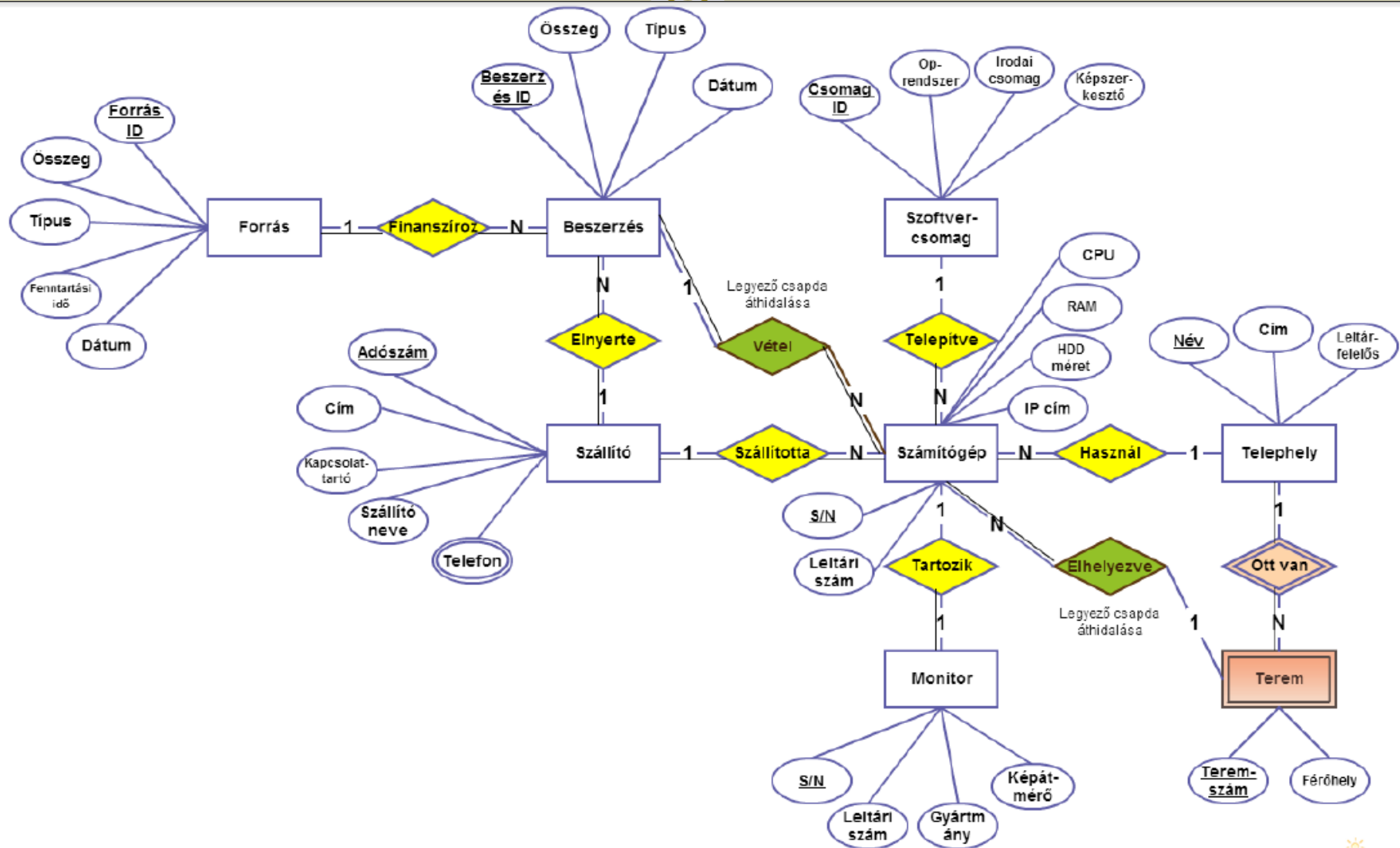


Feloldott szakadék csapda – példa



- Most már meg tudjuk mondani minden alkalmazott esetén, hogy melyik részlegen dolgozik.

Legyező csapda?



Adatmodellező case-tool eszközök



- Számos közkedvelt adatmodellező és relációs sémába leképező szoftver létezik
 - Például: ERWin, S-Designer (Enterprise Application Suite), ER-Studio, (MySQLWorkbench), OpenModelSphere (open-source), RISE Editor (can generate database and application layer code), Gliffy, Creately, draw.io, stb.
- POZITÍVUMOK:
 - Követelmények dokumentálását segíti elő, felhasználóbarát felületek – főként grafikus támogatás