

Adatbázis-kezelő rendszerek I.



Fogarassyné Vathy Ágnes

Rendszer- és Számítástudományi Tanszék

vathy@dcs.uni-pannon.hu

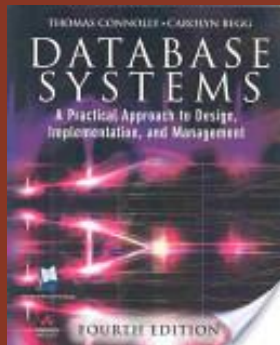
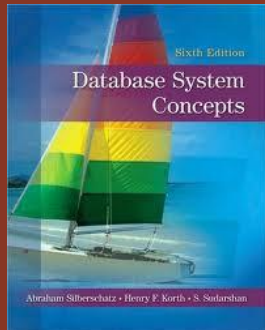
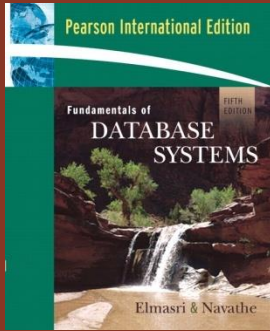
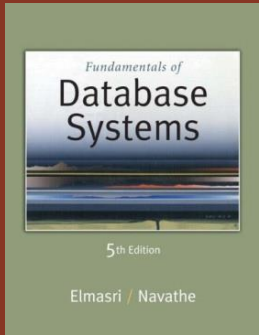
Iroda: I/914

Adatbázis-kezelő rendszerek I.



BEVEZETÉS

Irodalom



- Elmasri & Navathe: Fundamentals of Database Systems (ref as: EN)
- Silberschatz & Korth & Sudarshan: Database System Concepts (ref as: SKS)
- Thomas M. Connolly, Carolyn E. Begg: Database Systems: A Practical Approach to Design, Implementation, and Management (ref as: CB)

Irodalom

- Molina & Ullman & Widom:
Database Systems: The Complete Book
(ref as: UW)

Ullman & Widom: A First Course in Database Systems

+

Molina & Ullman & Widom: Database System
Implementation

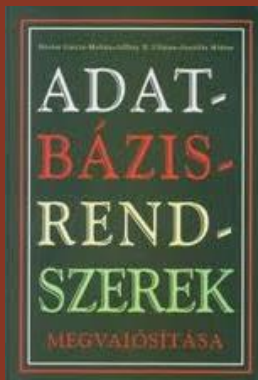
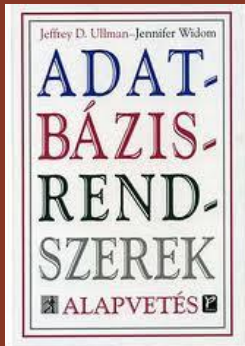
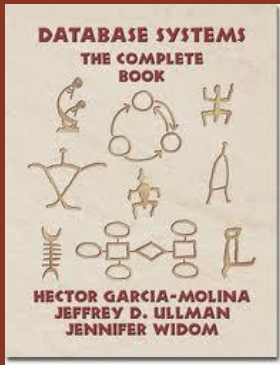
Magyarul:

Ullman & Widom: Adatbázisrendszerek - Alapvetés

+

Molina & Ullman & Widom: Adatbázisrendszerek
megvalósítása

http://www.inf.unideb.hu/kmitt/konvkmitt/fejezetek_az_adatbazisrendszerek_elmeletebol/book.xml.html



Irodalom



- Timár és Tsai: Építsünk könnyen és lassan adatmodellt!
- Kósa Márk, Pánovics János: Fejezetek az adatbázisrendszerek elméletéből
http://www.tankonyvtar.hu/hu/tartalom/tamop425/0046_fejezetek_az_adatbazisrendszerek_elmeletebol/index.html

Adatbázis alkalmazások



- Az adatbázisok jelen vannak életünk minden területén
 - Banki tranzakciók
 - Légitársaságok: foglalások, menetrendek
 - Oktatás: hallgatói adminisztráció, értékelés
 - e-kereskedelem: vásárlók, termékek, vásárlások, rendelések nyomon követése
 - Ipar: gyártási folyamatok, rendelések, ellátási lánc
 - Emberi erőforrások: alkalmazottak adatai, jövedelmek, adólevonások

Alapfogalmak



- **Adat**
 - A feladat szempontjából fontos, ismert tény, melyet tárolni szeretnénk
- **Adatbázis (DB, AB):**
 - Egymással összefüggő adatok gyűjteménye
- **Adatbázis-kezelő rendszer (DBMS – Database Management System):**
 - Adatbázisokhoz való hozzáférést, rendszeres és a felhasználói folyamatok zavartalan működést biztosító szoftveralkalmazás.
- **Adatbázisrendszer (Database System)**
 - Az adatbázis-kezelő rendszer és a tárolt adatok együtt. Az alkalmazásokat is magában foglalja.

Sémák és előfordulások



- **Séma:** az adatbázis struktúrája
 - Az adatbázis-tervezés folyamán kerül definiálásra.
 - Várhatóan nem változik gyakran.
- **Előfordulás:** az adatbázis aktuális tartalma egy adott időpontban
 - Az aktuális adat gyakran változhat.
 - Ugyanazon sémának sok-sok előfordulás feleltethető meg.

Példa adatbázis séma



DOLGOZÓ

Vnév	Knév	<u>Szsz</u>	Szdátum	Lakcím	Nem	Fizetés	Főnök_szsz	Osz
------	------	-------------	---------	--------	-----	---------	------------	-----

OSZTÁLY

Onév	<u>Oszám</u>	Vez_szsz	Vez_kezdő_dátum
------	--------------	----------	-----------------

OSZT_HELYSZÍNEK

<u>Oszám</u>	<u>Ohelyszín</u>
--------------	------------------

PROJEKT

Pnév	<u>Pszám</u>	Phelyszín	Osz
------	--------------	-----------	-----

DOLGOZIK_RAJTA

<u>Dszsz</u>	<u>Psz</u>	Órák
--------------	------------	------

HOZZÁTARTOZÓ

<u>Dszsz</u>	<u>Hozzá tartozó_név</u>	Nem	Szdátum	Kapcsolat
--------------	--------------------------	-----	---------	-----------

Példa adatbázis állapot (előfordulás)

DOLGOZÓ

Vnév	Knév	Szsz	Szdátum	Lakcím	Nem	Fizetés	Főnök_szsz	Osz
Kovács	László	1 650109 0812	1965. január 9.	4033 Debrecen	F	390000	2 551208 2219	5
Szabó	Mária	2 551208 2219	1955. december 8.	1097 Budapest	N	520000	1 371110 4519	5
Kiss	István	1 680119 6749	1968. január 19.	1172 Budapest	F	325000	1 410620 4902	4
Takács	József	1 410620 4902	1941. június 20.	4027 Debrecen	F	559000	1 371110 4519	4
Horváth	Erzsébet	2 620915 3134	1962. szeptember 15.	1092 Budapest	N	494000	2 551208 2219	5
Tóth	János	1 720731 2985	1972. július 31.	6726 Szeged	F	325000	2 551208 2219	5
Fazekas	Ilona	2 690329 1099	1969. március 29.	3535 Miskolc	N	325000	1 410620 4902	4
Nagy	Zoltán	1 371110 4519	1937. november 10.	1061 Budapest	F	715000	NULL	1

OSZTÁLY

Onév	Oszám	Vez_szsz	Vez_kezdő_dátum
Kutatás	5	2 551208 2219	1988. május 22.
Humán erőforrás	4	2 690329 1099	1995. január 1.
Központ	1	1 371110 4519	1981. június 19.

DOLGOZIK_RAJTA

Dszsz	Psz	Órák
1 650109 0812	1	32.5
1 650109 0812	2	7.5
2 620915 3134	3	40.0
1 720731 2985	1	20.0
1 720731 2985	2	20.0
2 551208 2219	2	10.0
2 551208 2219	3	10.0
2 551208 2219	10	10.0
2 551208 2219	20	10.0
1 680119 6749	30	30.0
1 680119 6749	10	10.0
2 690329 1099	10	35.0
2 690329 1099	30	5.0
1 410620 4902	30	20.0
1 410620 4902	20	15.0
1 371110 4519	20	NULL

OSZT_HELYSZÍNEK

Oszám	Ohelyszín
1	Budapest
4	Kecskemét
5	Vác
5	Tiszafüred
5	Budapest

PROJEKT

Pnév	Pszám	Phelyszín	Osz
X termék	1	Vác	5
Y termék	2	Tiszafüred	5
Z termék	3	Budapest	5
Komputerizáció	10	Kecskemét	4
Rcorganizáció	20	Budapest	1
Új fejlesztések	30	Kecskemét	4

HOZZÁTARTOZÓ

Dszsz	Hozzá tartozó_név	Nem	Szdátum	Kapcsolat
2 551208 2219	Anna	N	1986. április 5.	lánya
2 551208 2219	Bence	F	1983. október 25.	fia
2 551208 2219	Máté	F	1958. május 3.	házastársa
1 410620 4902	Viktória	N	1942. február 28.	házastársa
1 650109 0812	Balázs	F	1988. január 4.	fia
1 650109 0812	Anna	N	1988. december 30.	lánya
1 650109 0812	Réka	N	1967. május 5.	házastársa

Adat és Metaadat



Adatbázis megközelítés vs. Hagyományos programozás
(fájl-feldolgozó megközelítés)

Az adatbázisrendszerek önleíró tulajdonsága:

- A DBMS **katalógus** tárolja az egyes adatbázisok leírásait is
- Ez a leírás a **metaadat**, tartalmazza pl:
 - adatstruktúrák,
 - típusok,
 - szabályok, korlátozások
 - ...
- Ez teszi lehetővé, hogy a DBMS különböző adatbázisokkal dolgozzon.
- A hagyományos fájl-feldolgozó megközelítésű programok csak meghatározott adatbázissal tudnak dolgozni.

Egyszerű (relációs) adatbázis



HALLGATÓ

Név	Hallgatói_azonosító	Évfolyam	Szak
Kovács	17	1	PTI
Szabó	8	2	PTI

TÁRGY

Tárgynév	Tárgykód	Kredit	Tanszék
Bevezetés az informatikába	INDK201	5	IT
Adatszerkezetek	INDK421	5	IT
Diszkrét matematika	INDK101	4	AM
Adatbázisrendszerek	INDK501	4	IT

KURZUS

Kurzus kód	Tárgykód	Félév	Tanév	Oktató
85	INDK101	ősz	2009/10	Bácsó
92	INDK201	ősz	2009/10	Terdik
102	INDK421	tavaszi	2010/11	Kósa
112	INDK101	ősz	2010/11	Bácsó
119	INDK201	ősz	2010/11	Csernoch
135	INDK501	ősz	2010/11	Adamkó

INDEXSOR

Hallgatói_azonosító	Kurzus kód	Érdemjegy
17	112	4
17	119	3
8	85	5
8	92	5
8	102	4
8	135	5

ELŐFELTÉTEL

Tárgykód	Előfeltétel_kód
INDK501	INDK421
INDK501	INDK101
INDK421	INDK201

Egyszerűsített adatbázis katalógus - példa



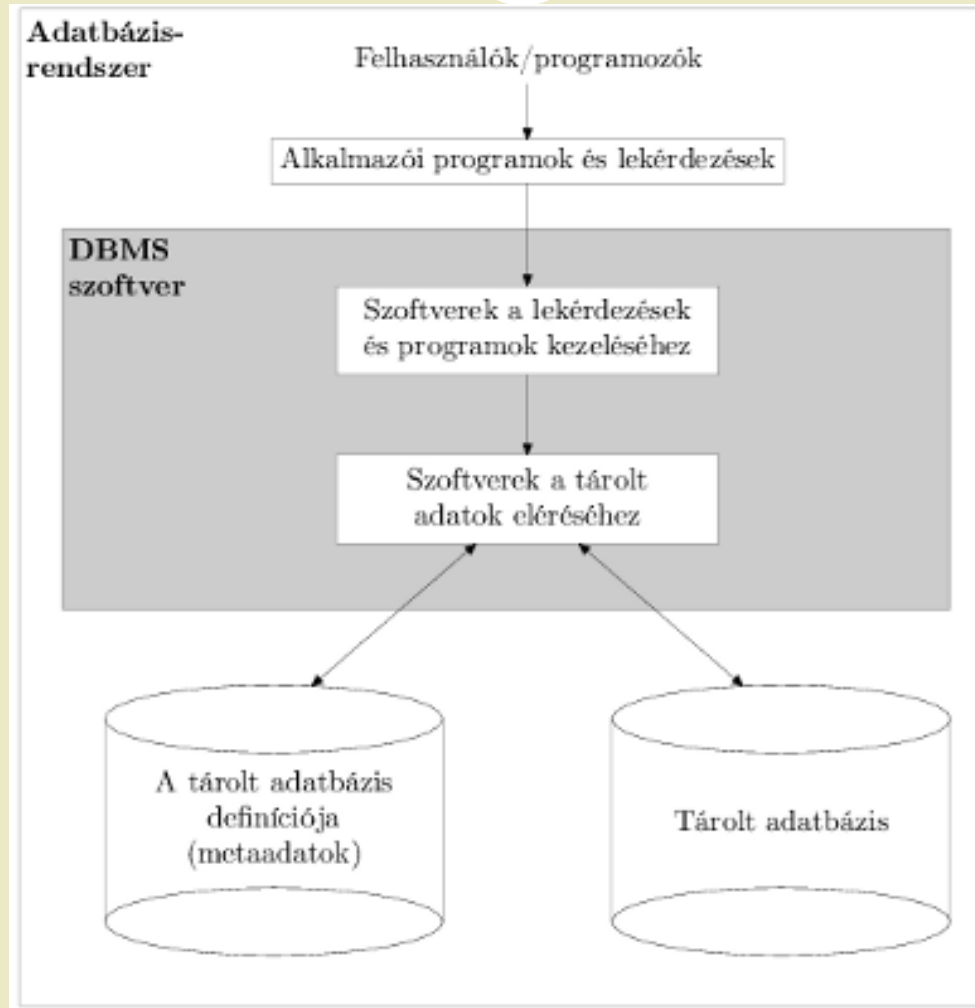
TÁBLÁZATOK

Táblanév	Oszlopszám
HALLGATÓ	4
TÁRGY	4
KURZUS	5
INDEXSOR	3
ELŐFELTÉT EL	2

OSZLOPOK

Oszlopnév	Adattípus	Táblázat
Név	Character(30)	HALLGATÓ
Hallgatói_azonosító	Character(4)	HALLGATÓ
Évfolyam	Integer(1)	HALLGATÓ
Szak	Szaktípus	HALLGATÓ
Tárgynév	Character(30)	TÁRGY
Tárgykód	XXXXNNN	TÁRGY
...
Előfeltétel_kód	XXXXNNN	ELŐFELTÉTEL

Egyszerűsített adatbázisrendszer



Az adatbázis-kezelő rendszerek szolgáltatásai



- **Az adatbázis definiálása:** az adatok típusának, szerkezetének, és rájuk vonatkozó korlátozásoknak a meghatározása
 - **Metaadat:** leíró jellegű információk az adatbázisról (a tárolandó adatról) – az adatokat leíró adatok
- **Az adatbázis létrehozása:** A DBMS által kontrollált folyamat, melynek során az adatbázis fizikailag is létrejön valamely adattároló eszközön
- **Az adatbázis manipulálása:**
 - **Információ kinyerése:** lekérdezések, riportok
 - **Adatmanipuláció:** Beszúrás, törlés és módosítás
- **Az adatbázis megosztása és tranzakciók:** Lehetővé teszi, hogy az adatbázist egyszerre több felhasználó érje el és tudja használni oly módon, hogy az adatok mindvégig konzisztensek maradjanak

Az adatbázis-kezelő rendszerek szolgáltatásai (folyt.)



- Egyéb jellemzők:
 - *Az adatbázisok védelme:*
 - ✦ *Rendszervédelem:* hibás hardware, vagy software működése esetén
 - ✦ *Biztonsági védelem:* jogosulatlan felhasználás vagy rosszindulatú támadással szemben
 - *Az adatbázis* és a kapcsolódó programok *fenntartása* az adatbázis alkalmazás teljes élettartama során
 - *Az adatok megjelenítése* (karakteresen és grafikusan)

Tranzakciókezelés



- A DBMS támogatja a több felhasználós környezetet, és az adat megosztását:
 - **OLTP** (Online Transaction Processing) **Tranzakciókezelő**: Az adatbázis alkalmazások fontos része, mely lehetővé teszi egymással konkuráló tranzakciók százainak egyidejű végrehajtását.
 - A DBMS támogatja, hogy az egymással **versengő felhasználók** az adatbázist elérjék és manipulálják.
 - A DBMS biztosítja hogy minden egyes **tranzakció** vagy tökéletesen végrehajtsódjon, vagy az eredeti állapot állítsódjon vissza.
 - A *tárkezelő* biztosítja, hogy minden egyes teljesített tranzakció esetén a végrehajtott módosítások az adatbázisban tartósan rögzüljenek.

Adatabsztrakció



- **Adatabsztrakció**

- A DBMS a felhasználók számára biztosítja az adatok olyan absztrakt ábrázolását, amely nem tartalmaz sok részletet (pl: hogyan tárolják az adatot, hogyan ágyazzák be a műveletet).
- Az absztrakció szintjei:
 - ✦ Fizikai (belső) szint
 - ✦ Logikai (fogalmi, koncepcionális) szint
 - ✦ Felhasználó nézetek szintje - (külső) szint

Külső szint



- **A külső szint** a legmagasabb szint, melyen az alkalmazó programok elrejtik a technikai részleteket (pl. adattípusok). - felhasználói nézetek
 - Alkalmasak az információk biztonsági okokból történő elrejtésére is (pl. az alkalmazottak fizetése).
 - **Külső séma:** a különféle felhasználói nézetek
- A különféle jogosultsággal rendelkező felhasználók különféle felhasználói felületeket látnak.
 - A külső szint számos különböző külső nézetből áll.
 - Az egyes felhasználók számára a valós világ egy nézete olyan formában jelenik meg, amely a felhasználó számára ismert, fontos.
 - A különböző nézeteknek ugyanazon adatok különböző megjelenési formái lehetnek.

Logikai szint



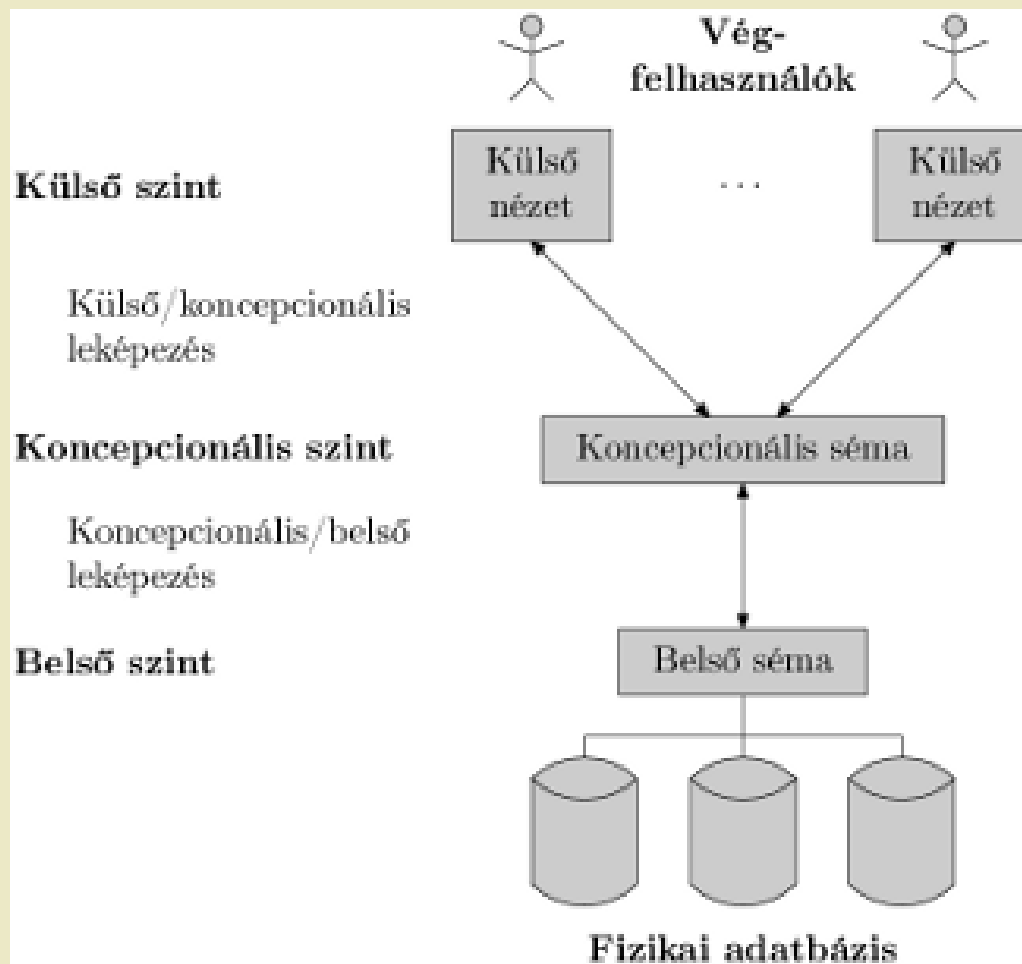
- A **logikai szint** a következő legmagasabb szint, amely leírja, hogy az adatbázisban milyen adatokat tárolunk, s az adatok milyen kapcsolatban állnak egymással.
 - **Logikai séma:** az adatbázis logikai struktúrája
 - Ez a szint az egész adatbázis logikai szerkezetét tartalmazza.
 - Tartalmazza:
 - ✦ Egyedek, attribútumok, kapcsolatok
 - ✦ Korlátozások
 - ✦ Szemantikus információk
 - ✦ Biztonsági információk
 - A logikai szint *támogatja a külső felhasználói nézeteket és független bármely adattárolási megfontolástól.*

Belső szint



- Az adatbázis fizikai megjelenése a számítógépen.
- **Fizikai szint:** a legalsó szint, amely azt mondja meg, hogy hogyan tároljuk az adatot.
 - **Fizikai séma:** az adatbázis szerkezete a fizikai szinten
 - Ez a szint lefedi az adatbázis fizikai megvalósítását, abból a célból, hogy az optimális futási időt és a tárhely kihasználás megvalósuljon.
 - Tartalmazza
 - ✦ a tárhely elosztását az adatok és indexek számára
 - ✦ a tárolandó rekordok leírását
 - ✦ a rekord elhelyezését
 - ✦ adattömörítés és kódolás (titkosítás)
 - A DBMS által vezérelve az operációs rendszer kezeli.

ANSI-SPARC 3-rétegű architektúra

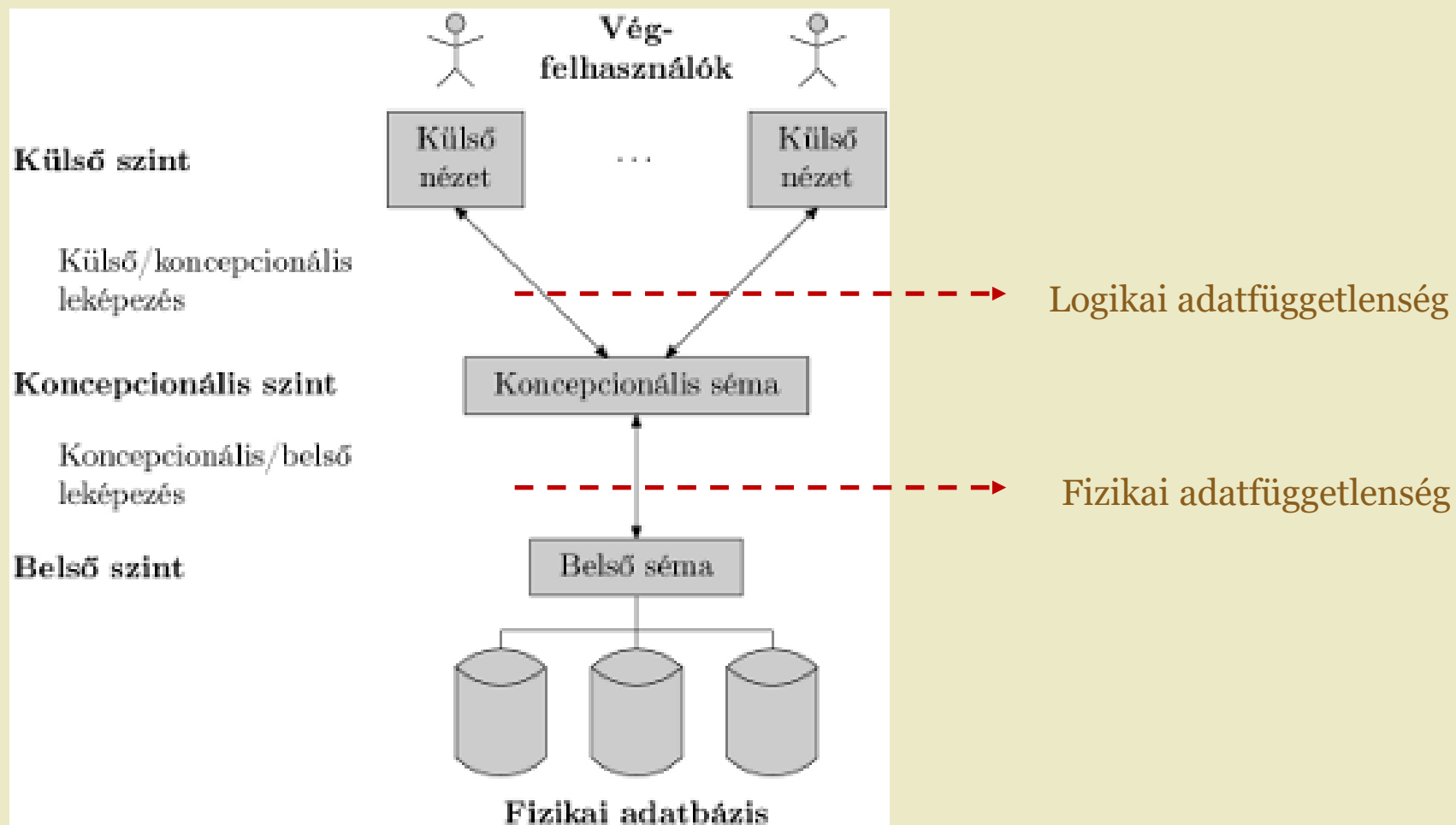


Adatfüggetlenség



- A 3-rétegű architektúra fő előnye az **adatfüggetlenség**:
Az architektúra *minden egyes magasabb szintje független az alatta elhelyezkedő szinttől.*
 - Logikai adatfüggetlenség
 - Fizikai adatfüggetlenség

ANSI-SPARC 3-rétegű architektúra



Adatfüggetlenség (folyt.)



- **Logikai adatfüggetlenség:** A logikai szint változtatása nem vonja maga után a külső szint változtatását.
 - például: új egyedek, attribútumok, korlátozások vagy kapcsolatok hozzáadása a logikai sémához, vagy eltávolítása a logikai sémából anélkül is lehetséges, hogy a külső sémában, vagy alkalmazói programban változás történjen.
- **Fizikai adatfüggetlenség:** A fizikai séma megváltoztatása nem vonja maga után a logikai séma megváltoztatását.
 - Például: a belső sémát érintő változások, mint például új fájlrendszer kialakítása, vagy tárolási struktúra, tárterület, indexelés változtatása anélkül megoldható, hogy a logikai sémát változtatni kellene.

2-rétegű és 3-rétegű architektúrák

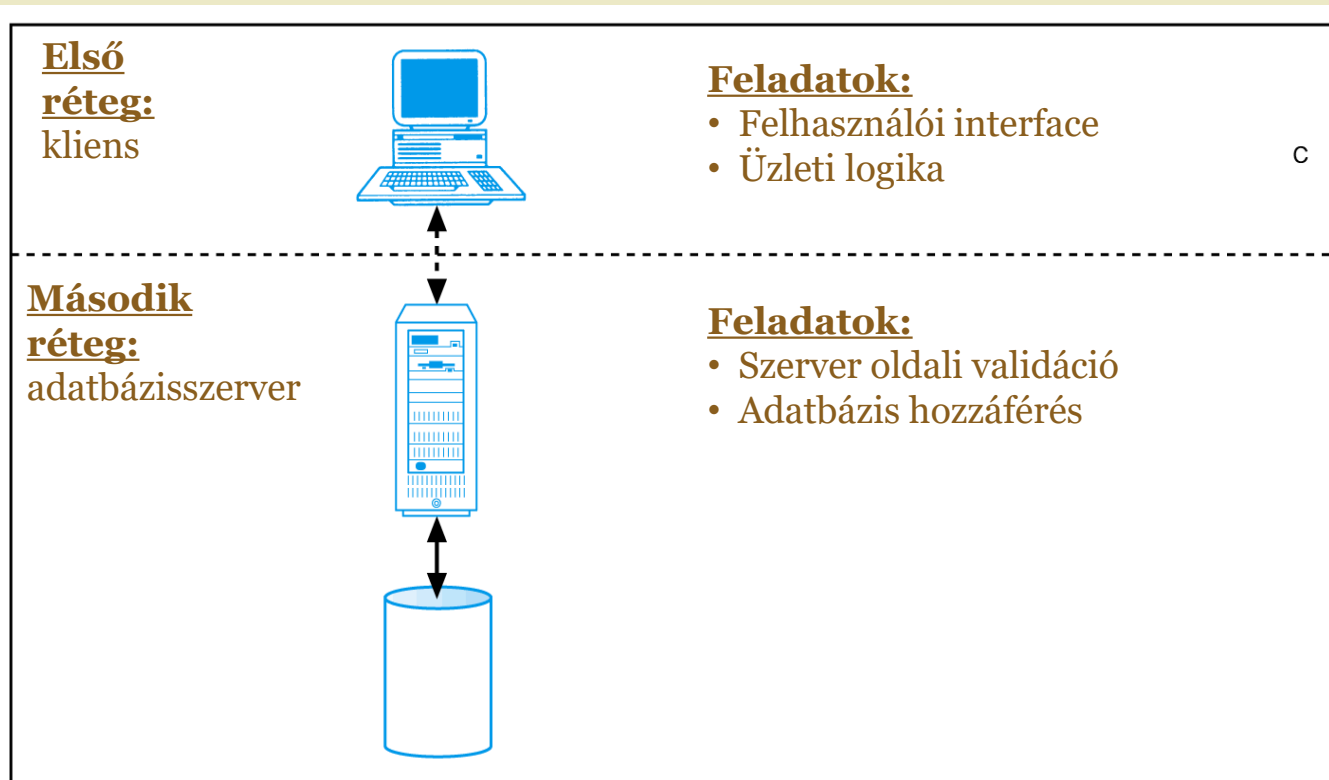


Bővebben: CB

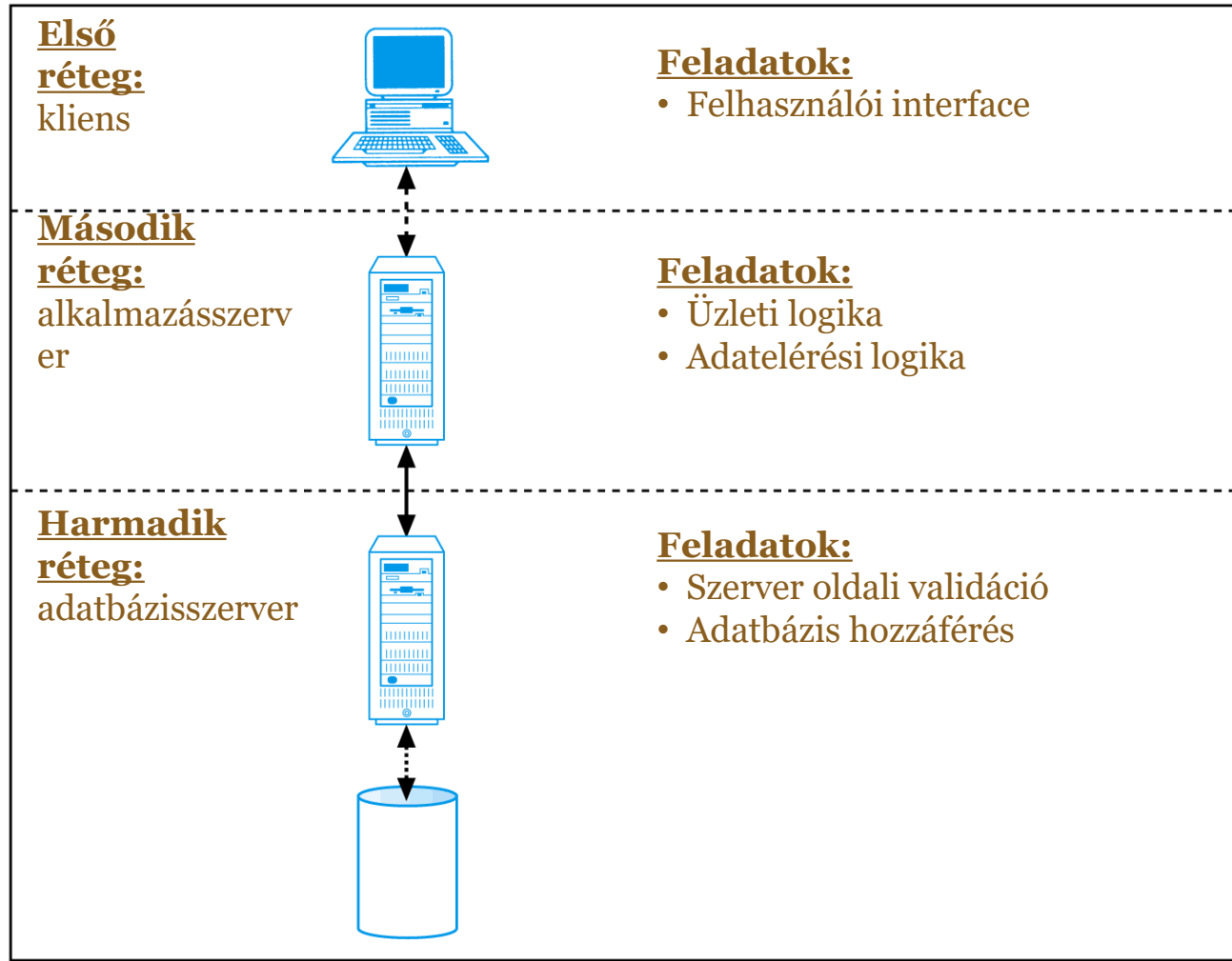
2-rétegű architektúra



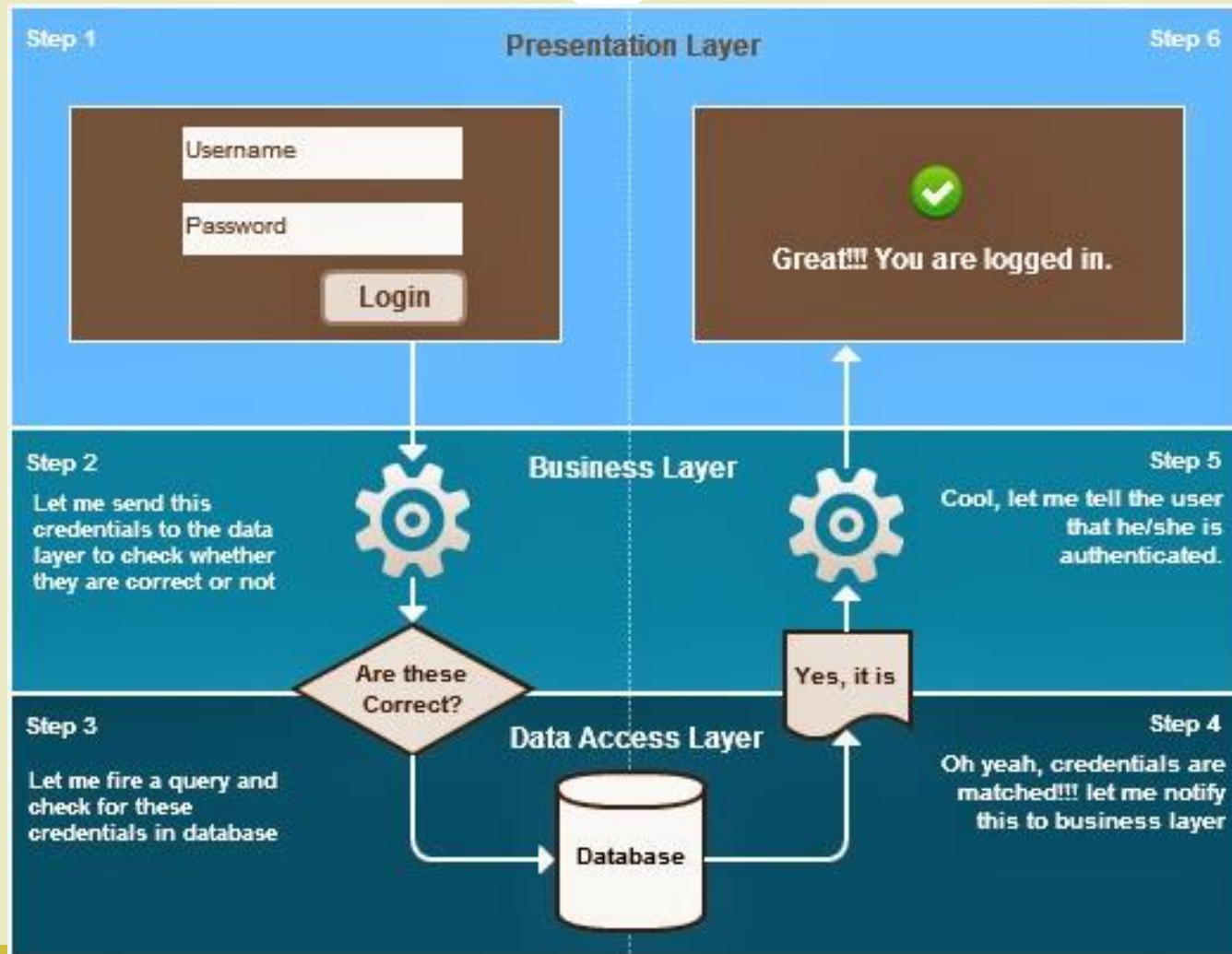
- Hagyományos 2-szintű kliens-szerver architektúra
 - Kliens (1. szint): a felhasználó interfészeket kezeli és futtatja az alkalmazásokat
 - Szerver (2. szint): tartalmazza az adatbázist és a DBMS-t



3-rétegű kliens-szerver



Példa



Hová helyezzük az üzleti logikát?



- Különféle lehetőségek:
 - **ORM: Object Relational Mapper (pl. Entity Framework)**
 - ✦ **LINQ (Language Integrated Query)** adatbázis elérés SQL nélkül, paraméteres kérdésekkel
 - Kevesebb kézzel írt kód
 - Gyors alkalmazásfejlesztés
 - ORM által generált objektumokhoz kötött üzleti logika
 - **Tárolt eljárások**
 - ✦ **Teljesítmény:** gyorsabb futás – teljesítményoptimalizáció a DBMS-ben
 - ✦ Biztonság
 - ✦ Kód újrafelhasználhatóság

Adatbázis szereplők



DB felhasználók



- Az alábbi felhasználók különböztethetők meg:
 - azok, akik használják és ellenőrzik az adatbázis tartalmát, és azok, akik tervezik, fejlesztik és fenntartják az adatbázis alkalmazásokat, és
 - azok, akik tervezik és fejlesztik a DBMS szoftvert és a kapcsolódó eszközöket (tools).

Felhasználók



- **DBMS fejlesztők:**

- ✦ Az adatbázis-kezelő rendszerek implementálását végzik.

- **Adatbázis-tervezők:**

- ✦ Felelősek a konkrét adatbázisok tartalmi, strukturális kialakításáért, továbbá korlátozásokat, eljárásokat és tranzakciókat definiálnak.
- ✦ Kommunikálnak a végfelhasználókkal, hogy felmérjék az igényeiket.

- **Adatbázis programozók:**

- ✦ Számítógépes szakemberek, akik az alkalmazói programokat írják.

Felhasználók (folyt.)



○ **Adatbázis adminisztrátorok (DBA):**

- ✦ Központi szereppel rendelkeznek az adatok, programok és hozzáférés tekintetében.
- ✦ Felelős/feladatai:
 - az adatbázishoz való hozzáférések biztosítása
 - koordinálja és monitorozza a működést
 - hatékony működés
 - az adatbázis periodikus mentése és visszaállítása
 - figyelemmel kíséri, hogy a normál működéshez elegendő tárhely áll-e rendelkezésre
 - ...

Felhasználók (folyt.)



○ **Végfelhasználók:**

- ✦ Az adatokat lekérdezésekre, jelentésekre használják, néhányan közülük az adatbázis tartalmát is frissítik.

DBMS használatának előnyei és hátrányai



Az adatbázis-kezelő rendszerek használatának előnyei



- Logikai és fizikai adatfüggetlenség
- Korlátozott jogtalan hozzáférés
- Tartós tárolás biztosított a program és objektumok számára
- Hatékony tárolási struktúrák (pl. indexek) – hatékony lekérdező folyamatok
- Adatok mentésének és visszaállításának támogatása
- Különböző felhasználói csoportok – különböző felhasználói nézetek
- Az adatok komplex kapcsolatrendszerének modellezése
- Hivatkozási korlátozások kezelése
- A tárolt adatokból levonható következtetések deduktív és aktív szabályok által.
 - Aktív adatbázisok, tárolt eljárások, triggerek

Mikor NE használjuk DBMS-t?



- Főbb ellenérvek:
 - Magas kezdő beruházási költség és további esetleges hardware igények
 - Alapköltségek a rendszer üzemeltetéséhez
- Amikor a DBMS nem szükséges:
 - Ha az adatbázis *nagyon egyszerű*, jól meghatározott és ebben változás nem várható
 - Amikor a DBMS is kevés:
 - ✦ Ha az adatbázis rendszer nem képes kezelni az adatok összetettségét az alkalmazható modellek határai miatt.
 - ✦ Amikor az adatbázis felhasználók speciális igényeit nem tudja a DBMS kiszolgálni.

Kiterjesztett lehetőségek



- Hagyományos alkalmazások
 - Numerikus és szöveges adatbázisok
- Kapcsolódó területek, témák:
 - Tudományos területek
 - XML (eXtensible Markup Language), JSON
 - Képfeldolgozás
 - Audio és Video adatok tárolása és feldolgozása
 - Adattárházak és adatbányászat
 - Térbeli adatbázisok
 - Idősor és hisztorikus adatok
 - Big Data technológiák
 - RDF