# Strukturális modellezés

Rendszermodellezés

2019.02.13.

**Budapesti Műszaki és Gazdaságtudományi Egyetem**
**Hibatűrő Rendszerek Kutatócsoport**

# Tudnivalók

- Gyakorlatra: színes tollak/ceruzák
- Jegyzet: https://inf.mit.bme.hu/edu/courses/remo/
  - Részletes definíciók, példák, szójegyzék
  - Mobiltelefonon és tableten is olvasható

RENDSZERMODELLEZÉS — Struktúra alapú modellezés

## 3.1. A struktúramodellezés alkalmazásai

Mind a természetben, mind a mesterséges rendszerekben fellelhetők bizonyos szabályszerűségek. Egyes szabályszerűségek a rendszer elemei közötti kapcsolatokat, míg mások magukat az elemeket jellemzik.

### 3.1.1. Hálózatok

Egy rendszert gyakran úgy jellemezhetünk a legjobban, ha bizonyos elemeit megkülönböztetjük és leírjuk az ezek közötti *kapcsolatot*.

**Példa.** Egy nagyváros közlekedési hálózata szövevényes rendszere az út- és sínhálózatnak, a több százezernyi járműnek és az ezeken utazó embereknek. A közlekedők mozgása mellett az infrastruktúra is folyamatosan változik a különböző fejlesztések, átalakítások és karbantartások miatt.

# Miről lesz szó?

- Struktúra modellezés célja, alkalmazásai
- Dekompozíció
- Modell elemek leírása gráfokkal
- Tulajdonságok

# STATIKUS MODELLALKOTÁS

*Hogyan modellezzük összetett rendszerek felépítését áttekinthető módon?*

Pl.

- Építészet – I épület
- Céges hálózat

# Definíció: Strukturális modell

A **strukturális modell** statikus. Kiindulópontja egy (rész)rendszer, amelyet a `része` reláció mentén alkotórészekre bont.

Az **alkotórészek** lehetnek:

- tovább bontott **részrendszerek** vagy
- tovább nem bontott (*elemi*) **komponensek**.

A strukturális modell a rendszer felépítését (struktúráját)

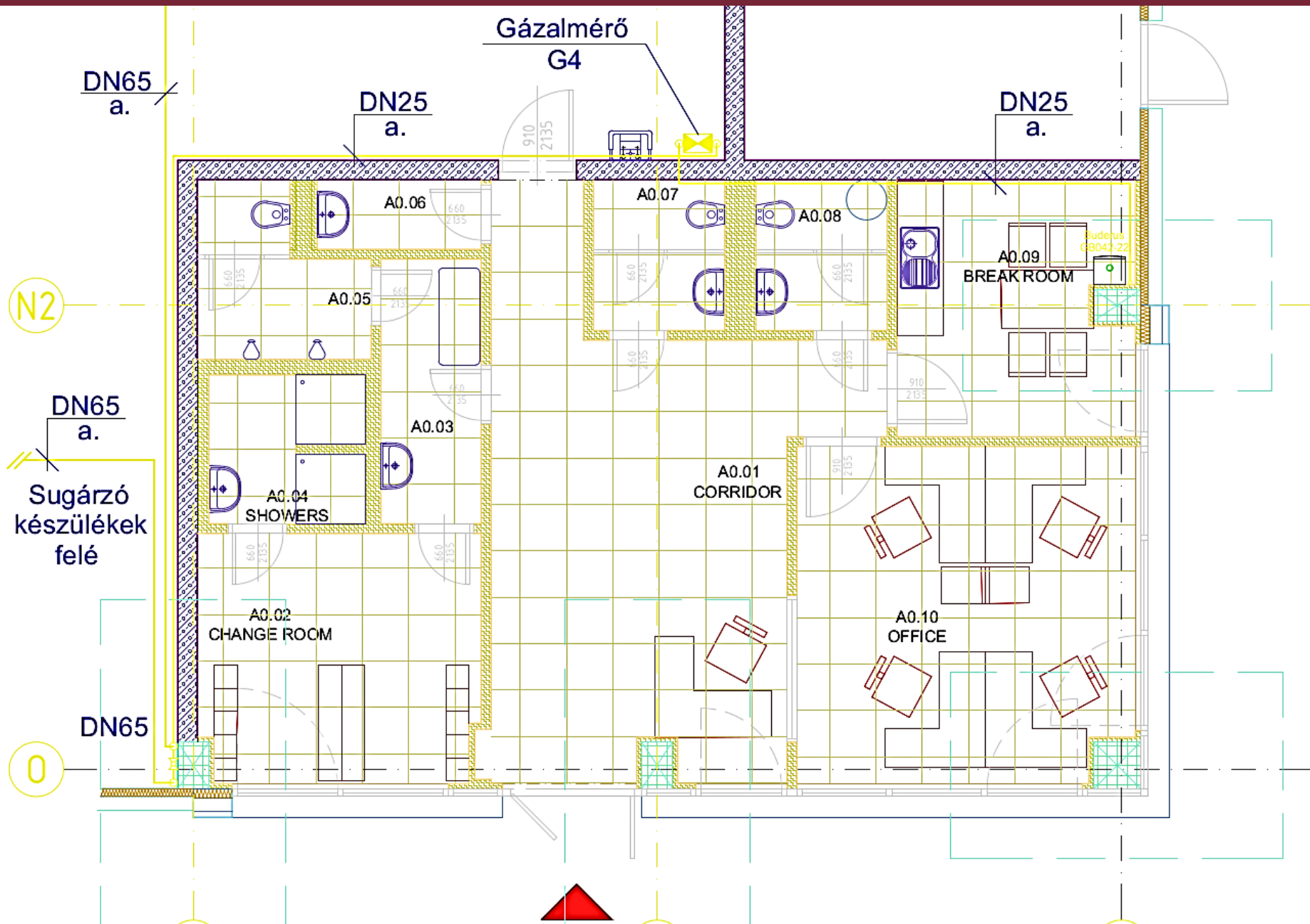
- az alkotórészei,
- azok **tulajdonságai** és
- egymással való **viszonya**

alapján reprezentálja.

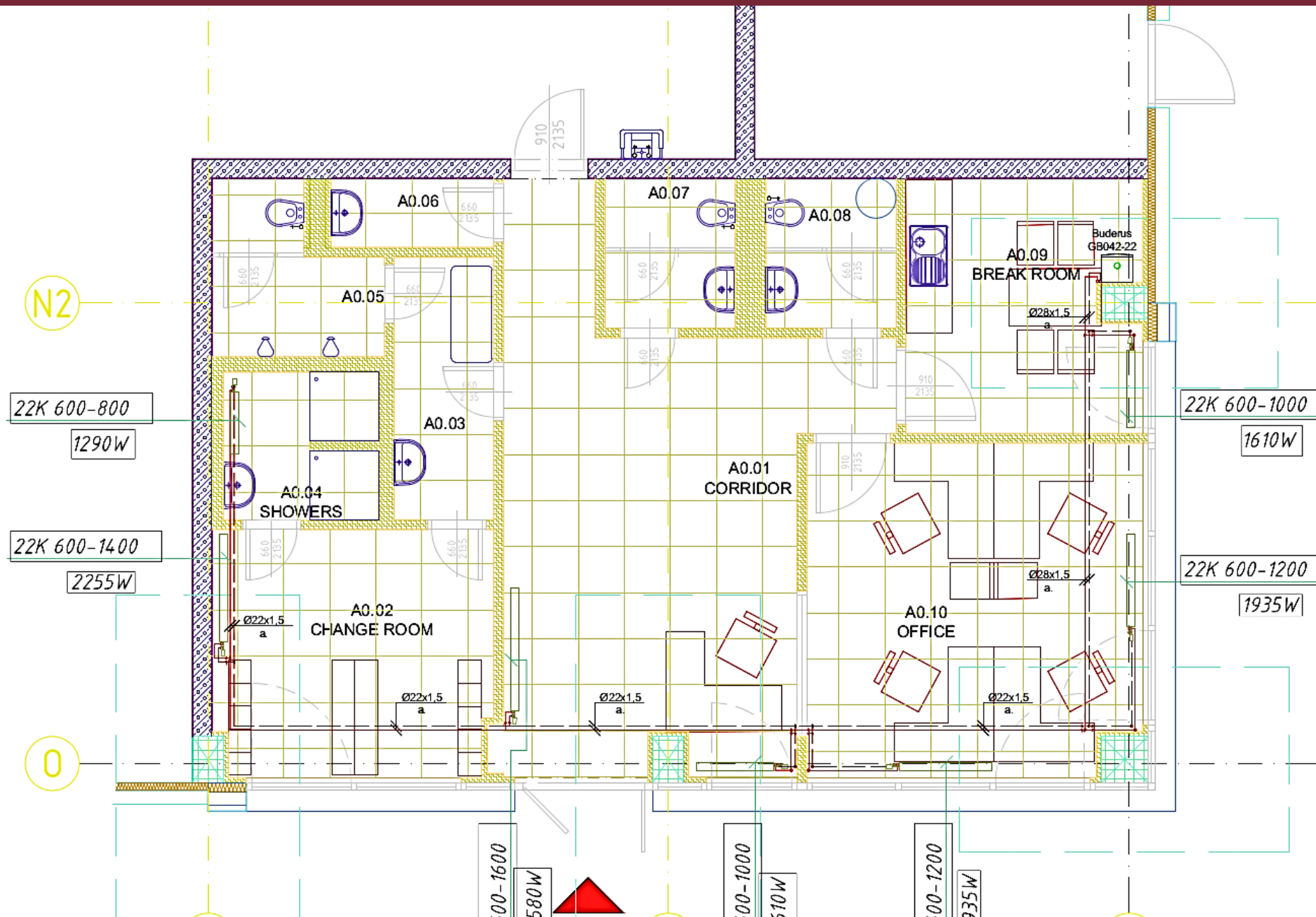
# Építészeti modellek

- BIM (Building Information Model)
- Épületek modellezése egy közös modellben
- Nézetek
  - Gázvezetékek
  - Szigetelés
  - Vízvezetékek és csatorna
  - Villamossági terv
  - Stb.

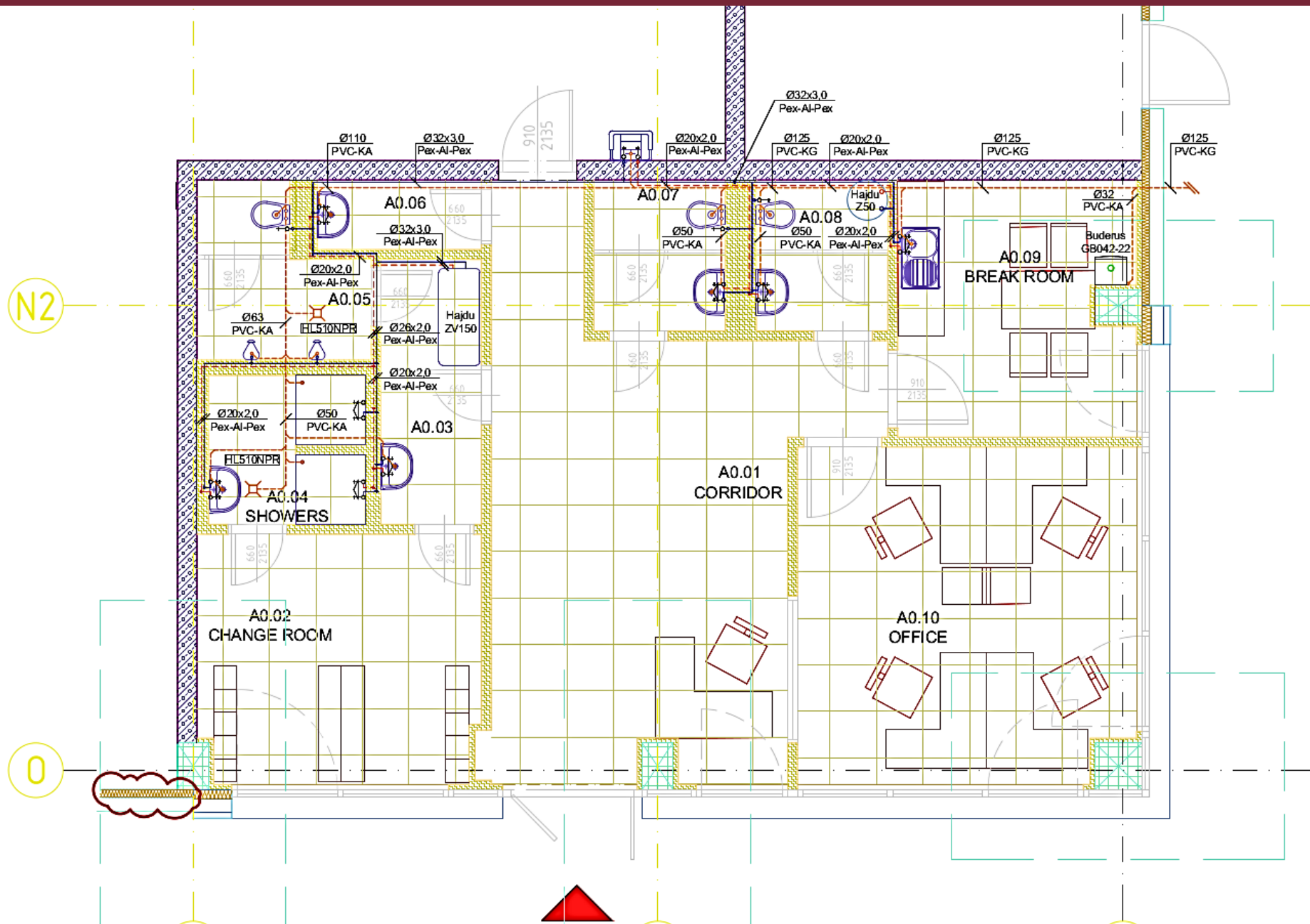
# Gázvezetékek



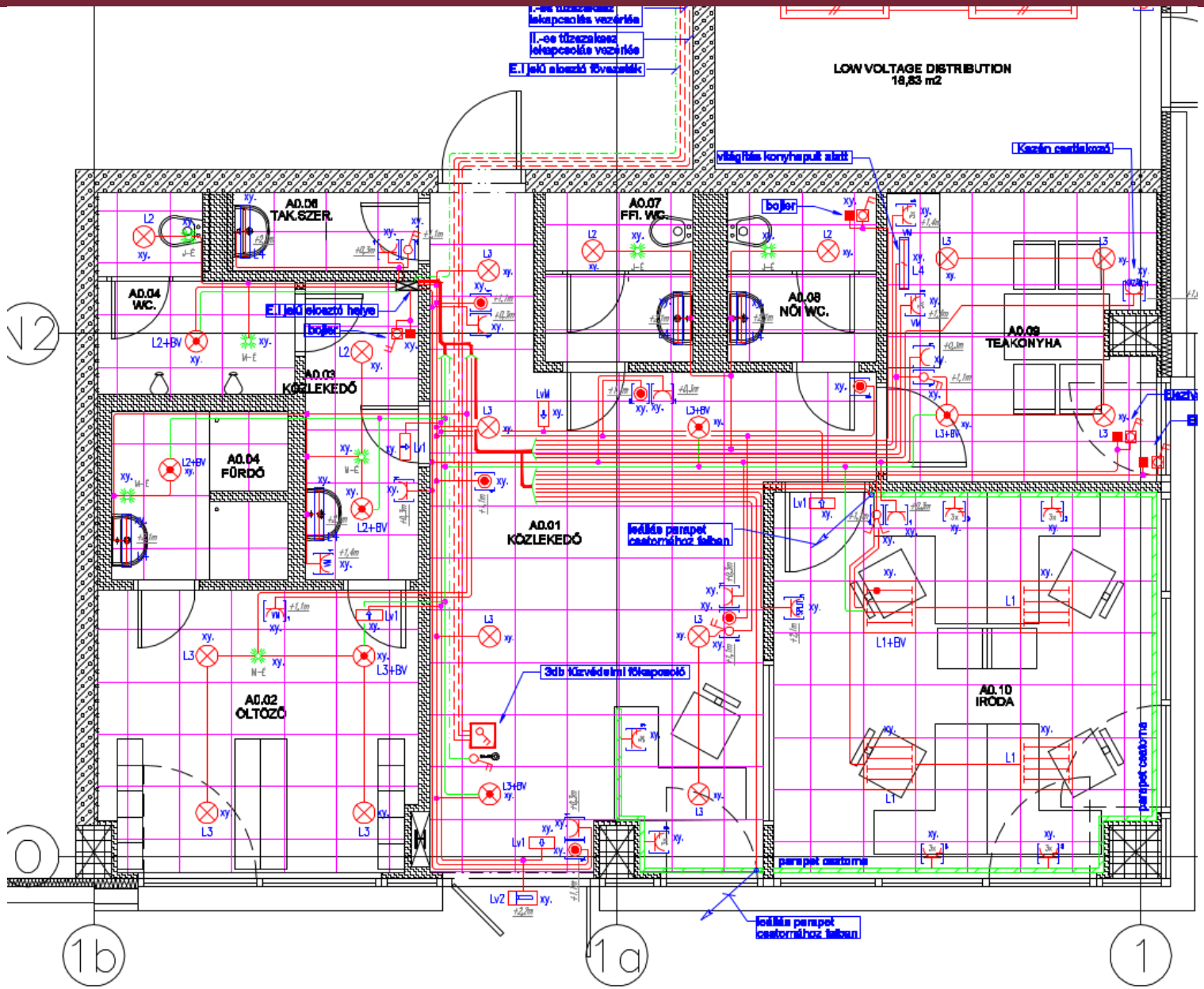
# Szigetelés


A0.06
A0.07
A0.08
A0.09
BREAK ROOM
Buderus
GB042-22
A0.05
A0.03
A0.04
SHOWERS
A0.01
CORRIDOR
A0.02
CHANGE ROOM
A0.10
OFFICE
N2
0
22K 600-800
1290W
22K 600-1400
2255W
22K 600-1000
1610W
22K 600-1200
1935W
Ø28x1,5
a.
Ø22x1,5
a
Ø22x1,5
a.
Ø22x1,5
a.
Ø22x1,5
a.
Ø28x1,5
a.
910
2135
660
2135

# Vízvezeték és csatorna



# Villamossági terv



# Strukturális modellezés célja

- Rendszer részekre bontása
  - Kisebb egységeket könnyebb megtervezni
  - Részegységek felhasználása
  - Általános célú komponensek használata
- Létező rendszer dokumentálása
  - „Rendszertérkép”
- Adatszerkezet megalkotása
  - Milyen információt kezelünk?

# Robotporszívó

# Felépítési és viselkedési modellezés

- Felépítési (*structural*)
  - Statikus
  - Rész és egész, összetevők
  - Kapcsolatok, összeköttetések
- Viselkedési (*behavioral*)
  - Dinamikus
  - Időbeli lefolyás
  - Állapot, folyamat
  - Reakciók a külvilágra
- Nem fed le mindent, nem válik élesen szét…

A robotporszívó főbb részei a vezérlő, a hajtómű, és a porszívó.

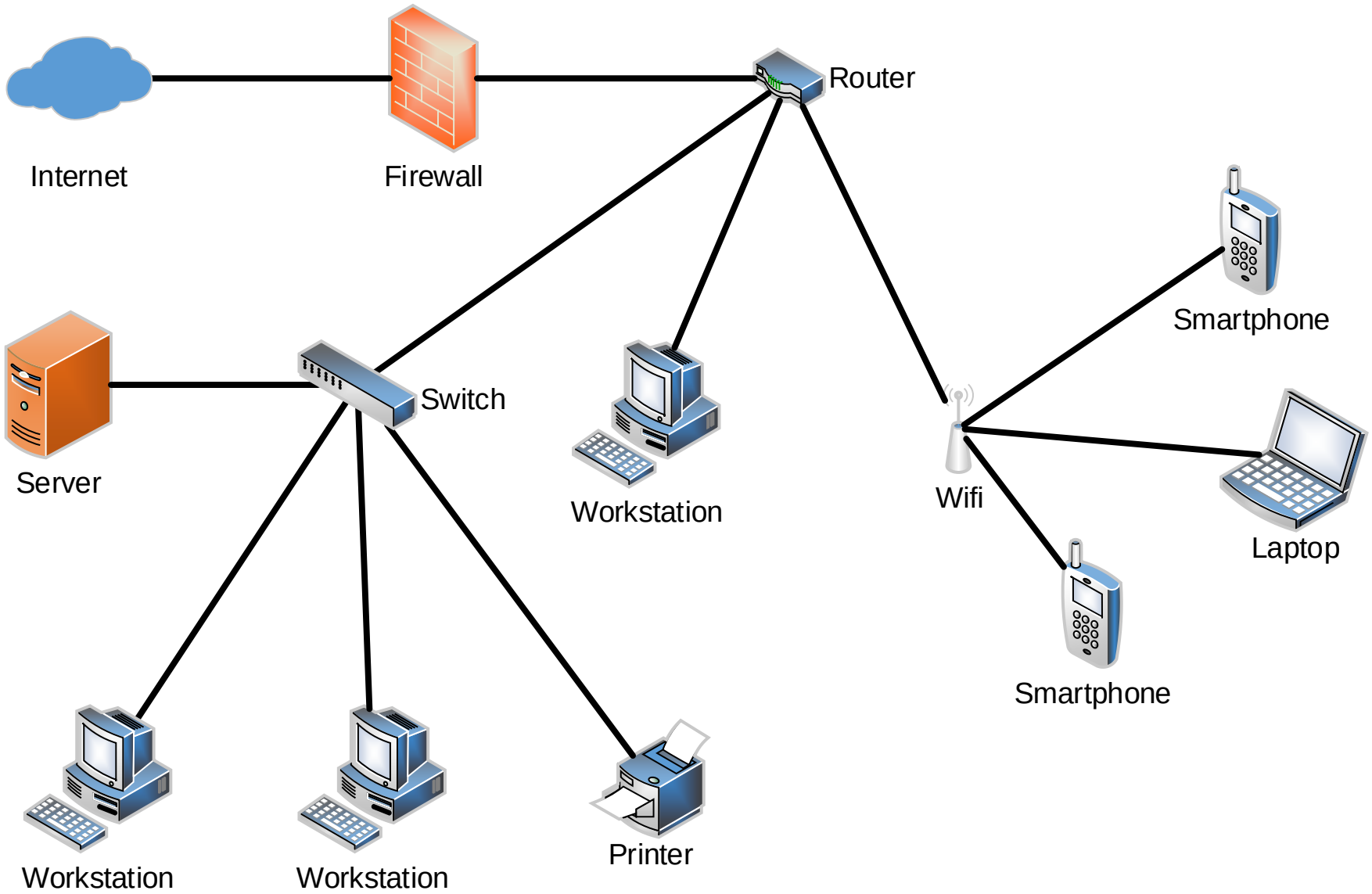
A hajtómű „jobbra” parancs hatására átvált „kanyarodás” üzemmódba

A robotporszívóban a szenzorok jeleit fogadva a hajtómű beavatkozik (mikor/hogyan?)

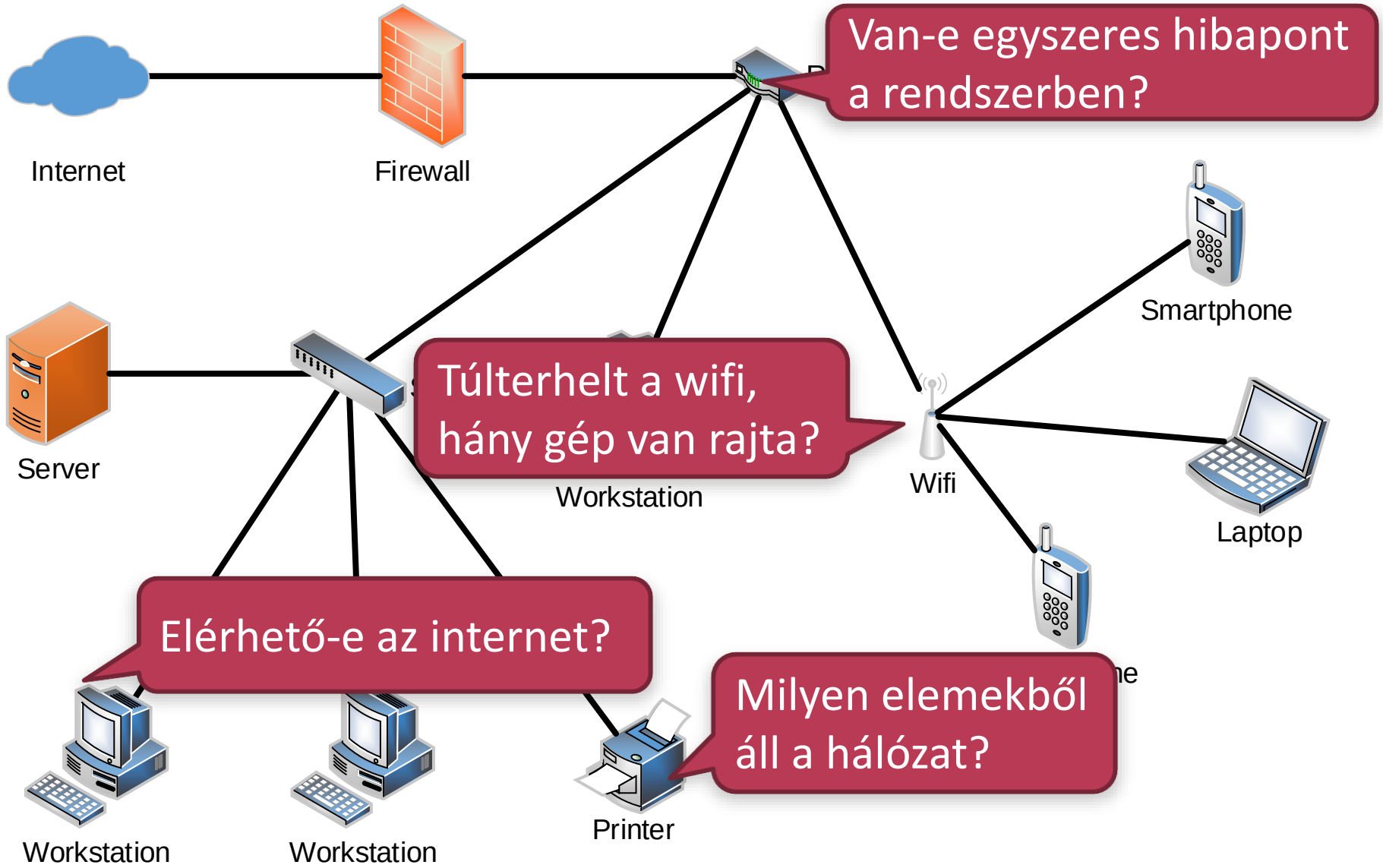
# Strukturális modell

- A rendszer felépítésére vonatkozó tudás
  - Milyen elemekből áll a rendszer?
  - Hogyan kapcsolódnak egymáshoz az elemek?
  - Milyen tulajdonságúak az elemek?

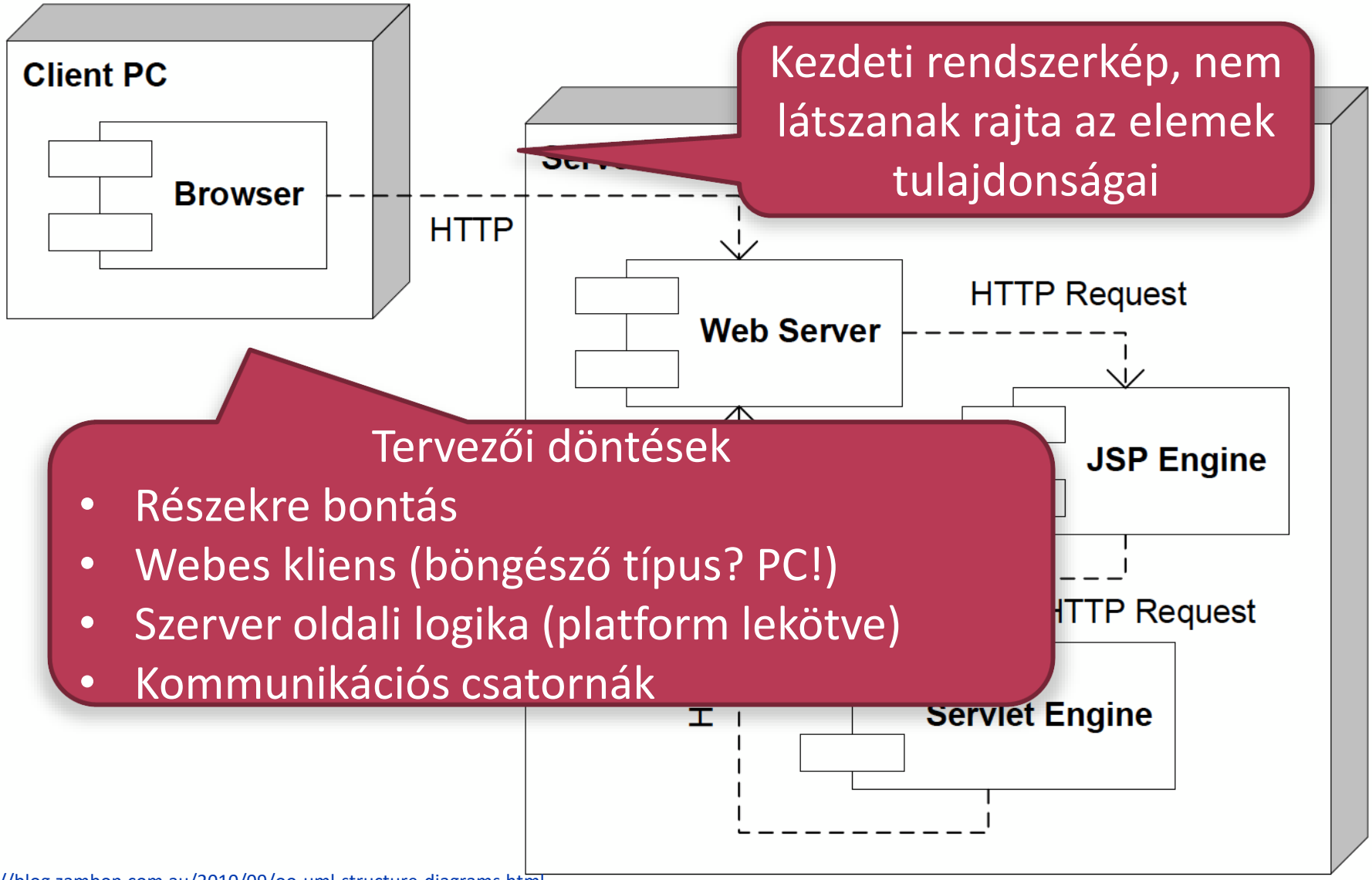
# Példa: (céges) hálózat



# Példa: (céges) hálózat



# Hogyan épül fel a rendszerem?

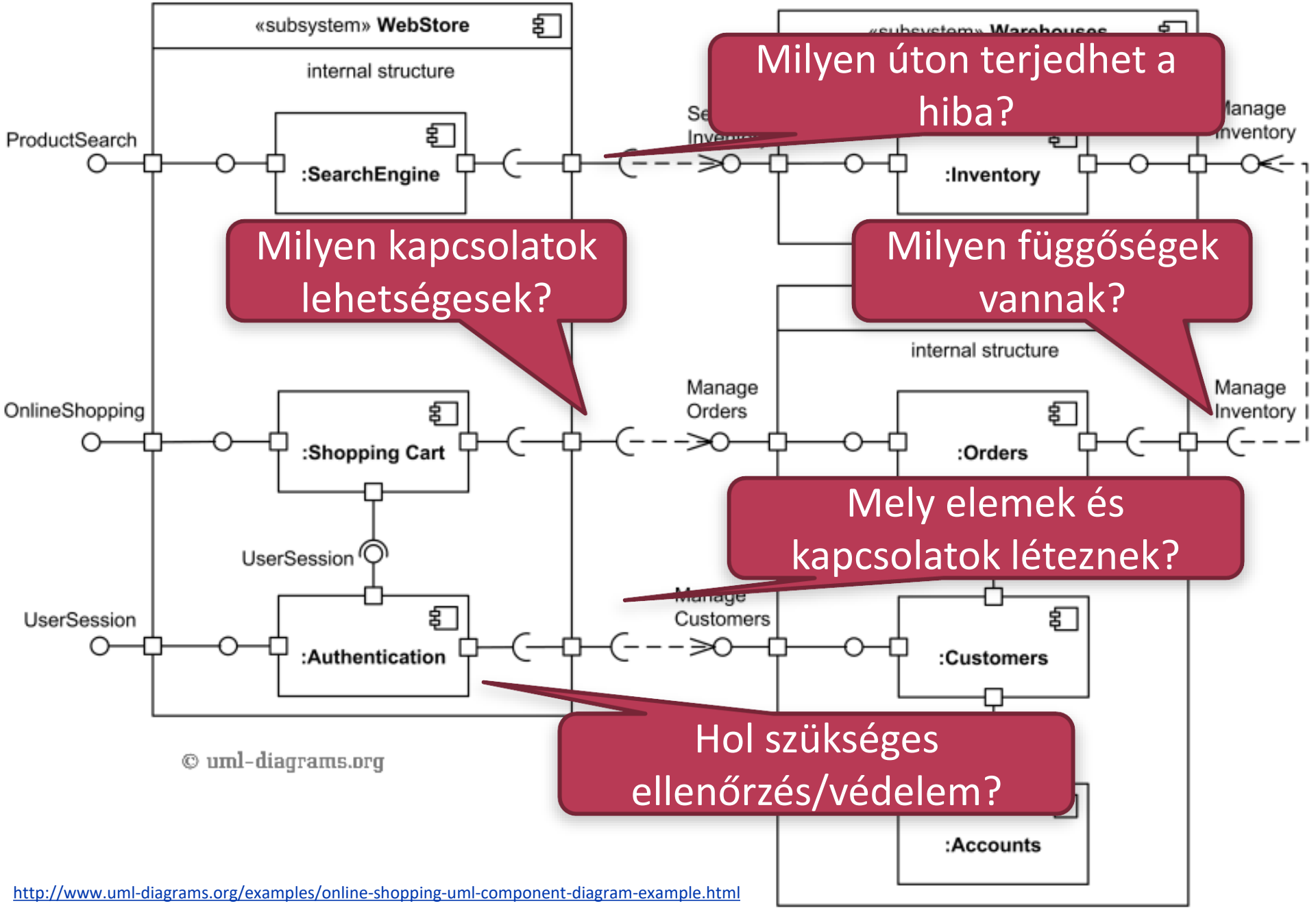



Kezdeti rendszerkép, nem látszanak rajta az elemek tulajdonságai

Tervezői döntések

- Részekre bontás
- Webes kliens (böngésző típus? PC!)
- Szerver oldali logika (platform lekötve)
- Kommunikációs csatornák

http://blog.zambon.com.au/2010/09/oo-uml-structure-diagrams.html




http://www.uml-diagrams.org/examples/online-shopping-uml-component-diagram-example.html

# STRUKTURÁLIS DEKOMPOZÍCIÓ

*Milyen tartalmazási viszonyok vannak az egyes részrendszerek között?*

Pl.

- Robotporszívó
- Batcave

# Definíció: Dekompozíció

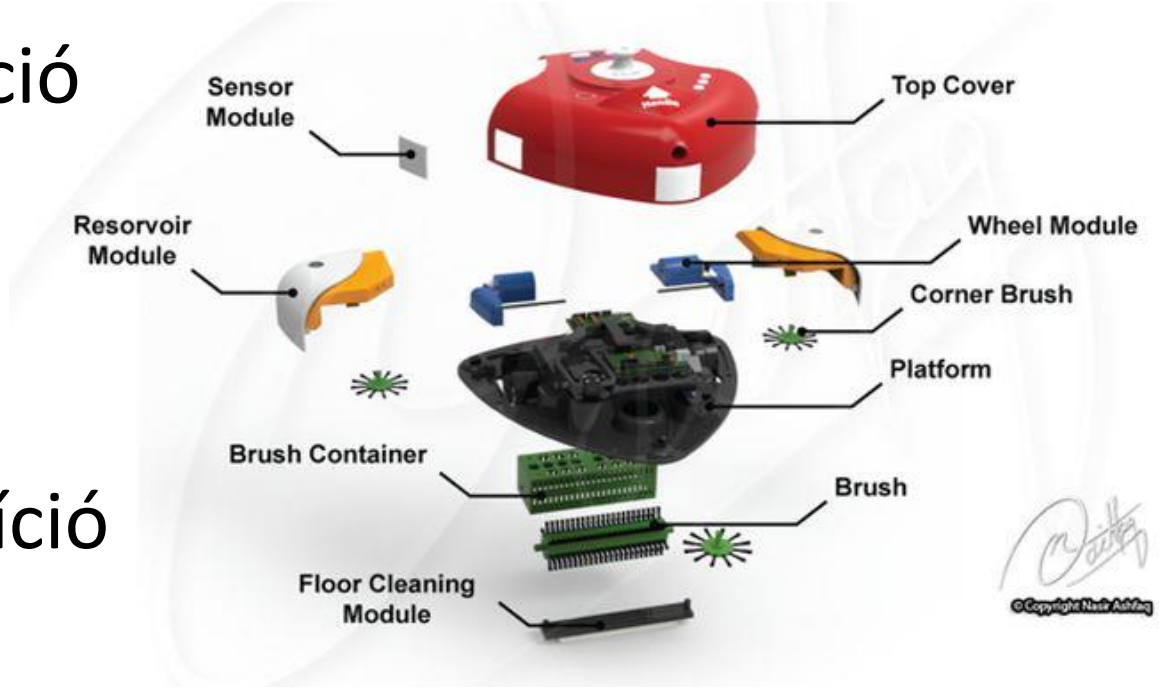
- *„A **strukturális modell** statikus. Kiindulópontja egy (rész)rendszer, amelyet a* `része` *reláció mentén alkotórészekre bont.”*

A **dekompozíció** („faktoring”) egy összetett probléma vagy rendszer kisebb részekre bontása, amelyek könnyebben érthetők, fejleszthetők és karbantarthatók.

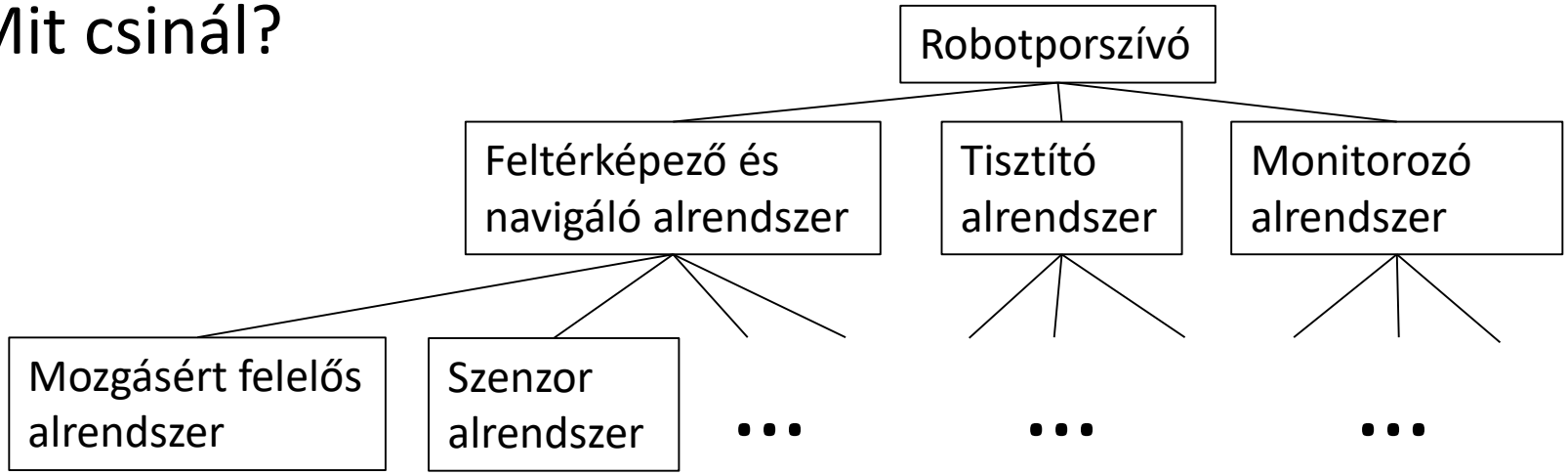
- Hierarchikus dekompozíció:
  - Többlépcsős rész-egész viszony
- Része reláció tartalma szerint:
  - Fizikai dekompozíció: `része` = térbeli tartalmazás
  - Logikai dekompozíció: pl. `része` = funkció

# Dekompozíció típusai

- Fizikai dekompozíció
  - Milyen alkatrész?




- Logikai dekompozíció *funkciók* szerint
  - Mit csinál?

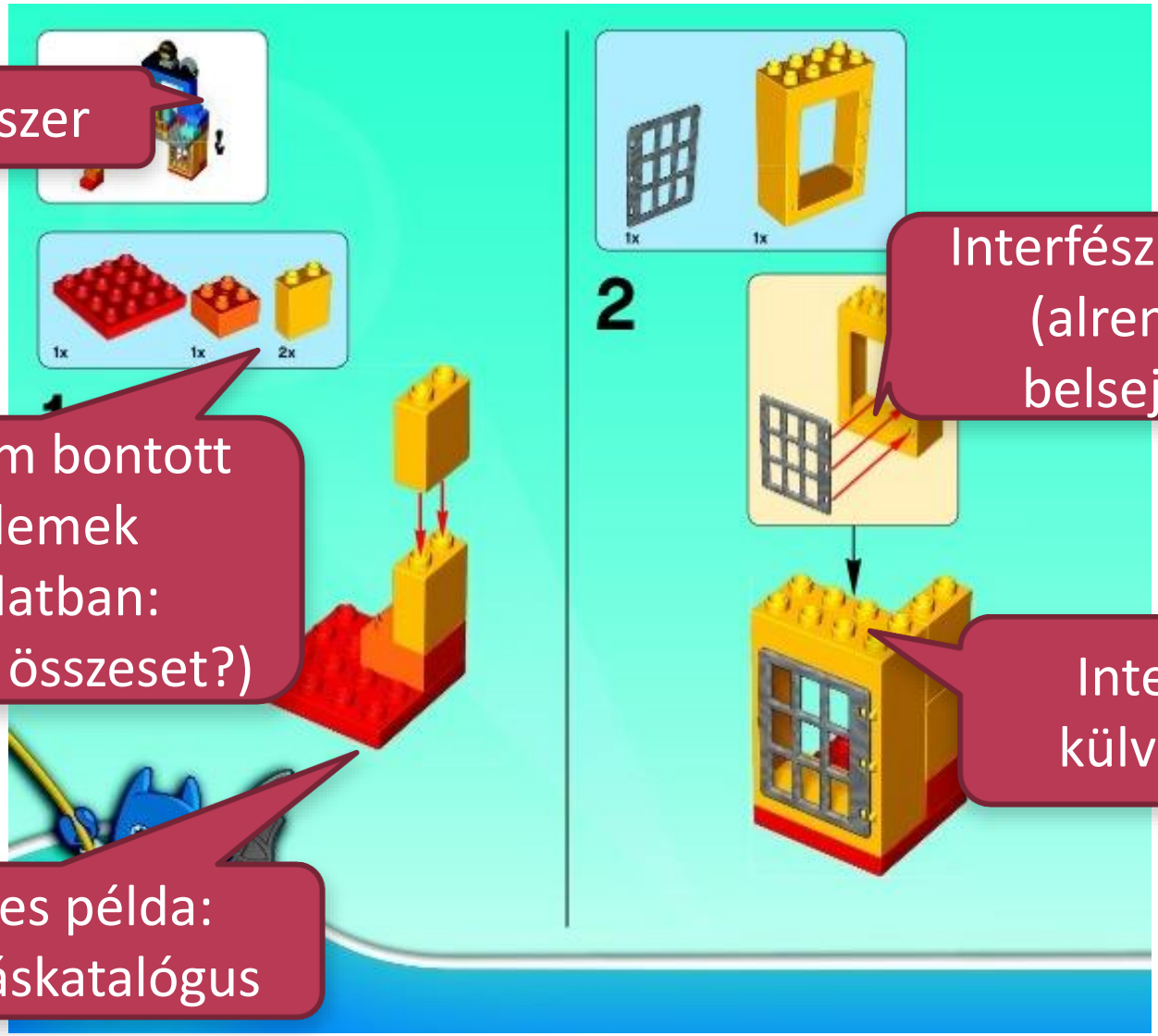


# Definíció: Dekompozíció helyessége

Egy dekompozíció **helyes**, ha

- a dekompozícióval kapott rendszer **minden elemének** megfeleltethető az eredeti rendszer **valamelyik eleme**, és
- az eredeti rendszer **minden eleméhez** hozzárendelhető a dekompozícióval kapott rendszer **legalább egy eleme**.

# Példa helyes kompozícióra




Teljes rendszer

Interfész definíció (alrendszer belsejében)

Tovább nem bontott építőelemek (gyakorlatban: ismerjük az összeset?)

Interfész a külvilág felé

Szoftveres példa: szolgáltatáskatalógus

http://lego.brickinstructions.com/en/lego_instructions/set/10545/batcave_adventure

# Példa helyes kompozícióra

Később:
hogyan határozzuk meg
a kompozíciót?
(elemek/rendszer felől?)

http://lego.brickinstructions.com/en/lego_instructions/set/10545/batcave_adventure

# STRUKTURÁLIS MODELLEK

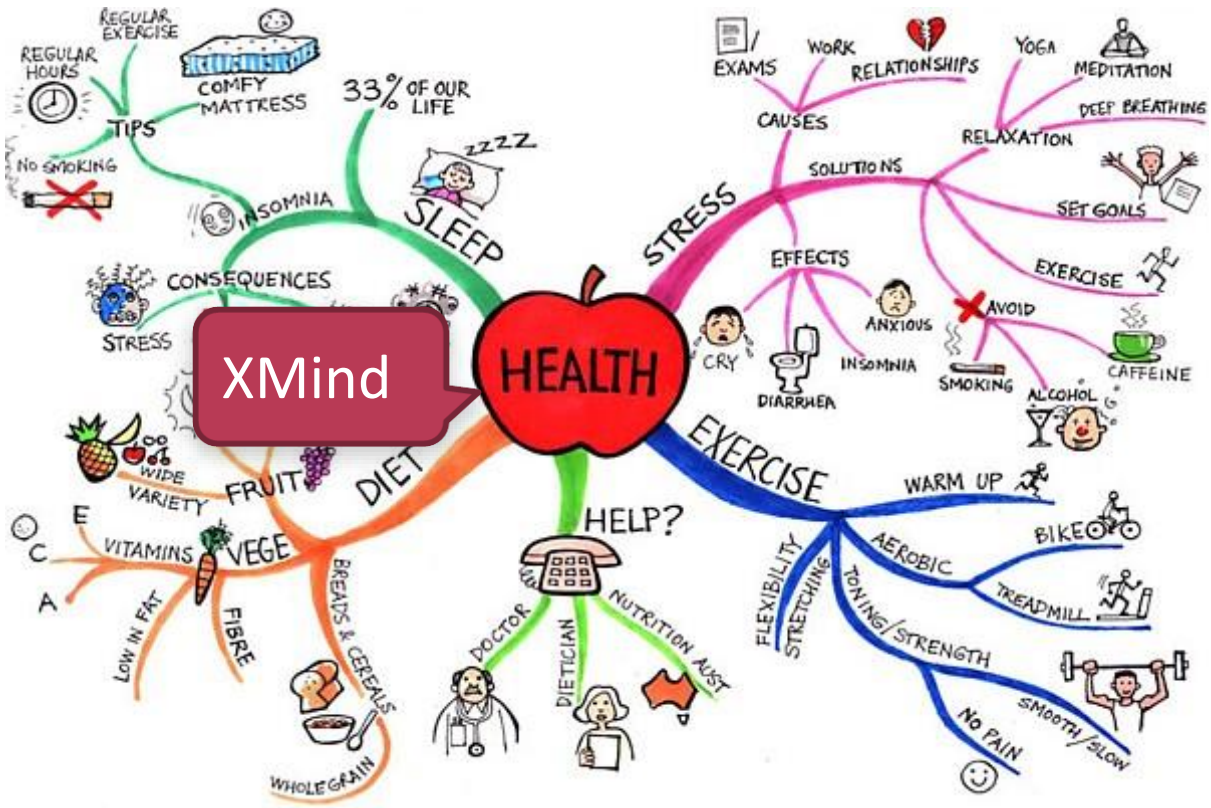
*Hogyan viszonyulnak egymáshoz az egyes részrendszerek, komponensek?*

Pl.

- Úthálózat
- Fájlrendszer

# Gráf reprezentáció

- Emberi gondolkodásmód: „dolgok kapcsolata”
  - **dolgok:** személyek, repülőterek, területek
  - **kapcsolata:** függőség, ismeretség, repülőjárat, része
- Matematikai formalizmus: **gráf**
  - csomópontok
  - élek
  - (tulajdonságok)



# Definíciók az informatikában

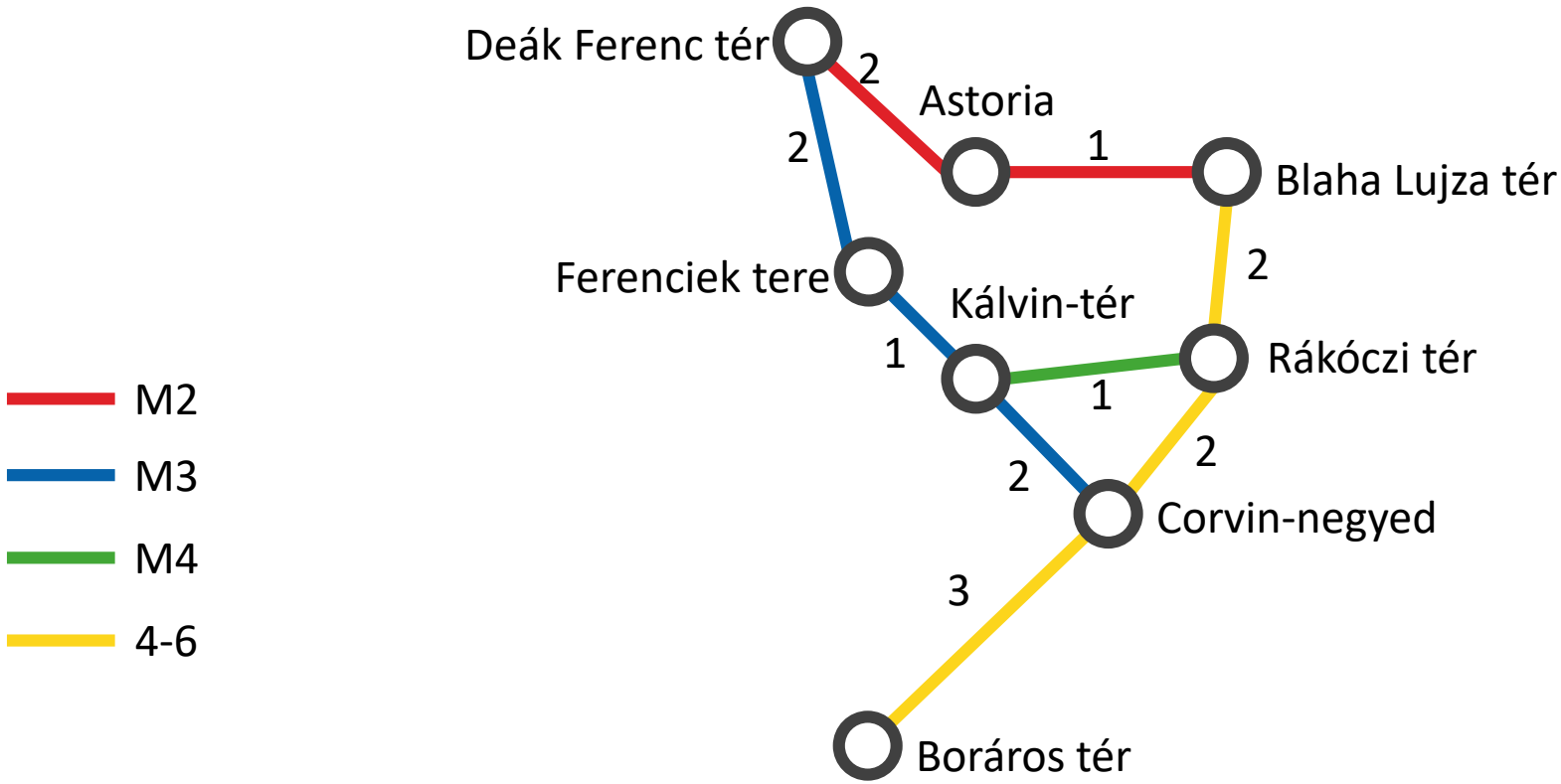
- Az informatikában mindennek több, gyakran egymásnak ellentmondó definíciója van
  - rendszer = ?
  - modell = ?
- Gyakran ugyanarra a fogalomra több nevével is hivatkozunk
  - csomópont, csúcs, pont, objektum (*node, vertex, object, concept*)
  - él, összeköttetés, kapcsolat (*edge, link, arc, connection, relationship*)

Egyet válasszunk!

# Budapest – kötöttpályás közlekedés

M2
M3
M4
4-6

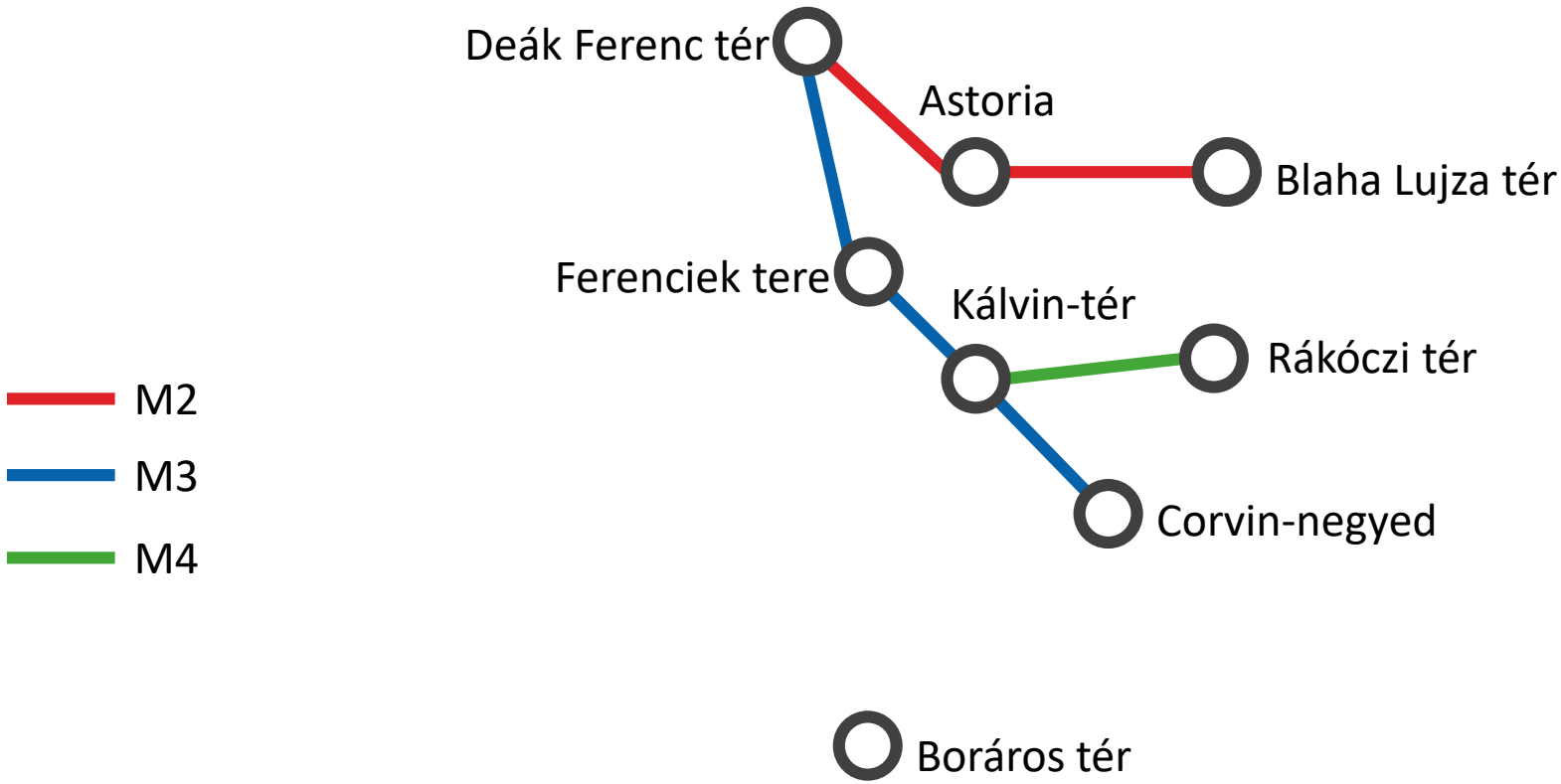
# Budapest – kötöttpályás közlekedés




legrövidebb út?

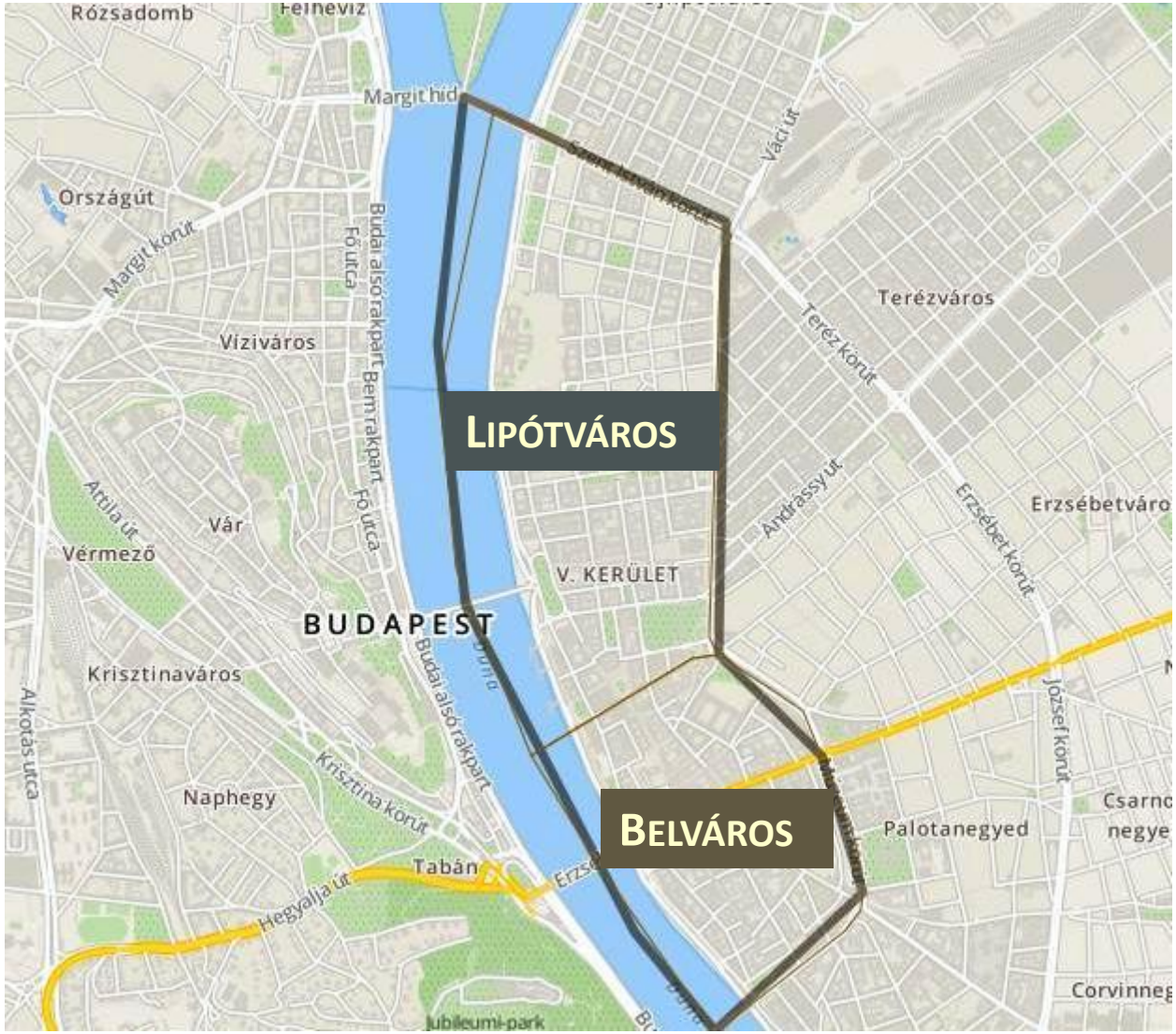
# Szűrés: élcímke szerint (részgráf)

- Pl. csak a metróhálózatot szeretnénk nézni

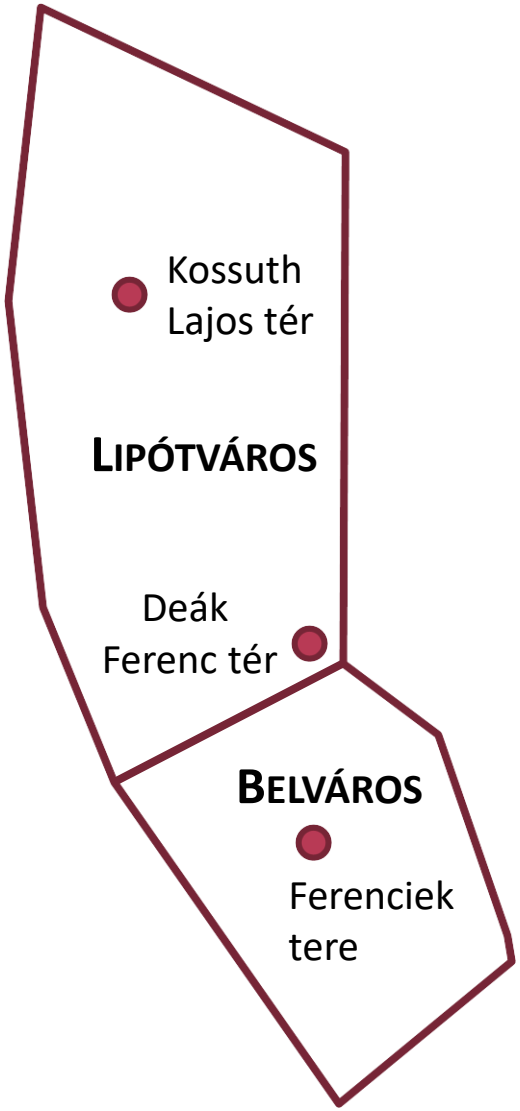



- Hova lehet eljutni metróval?

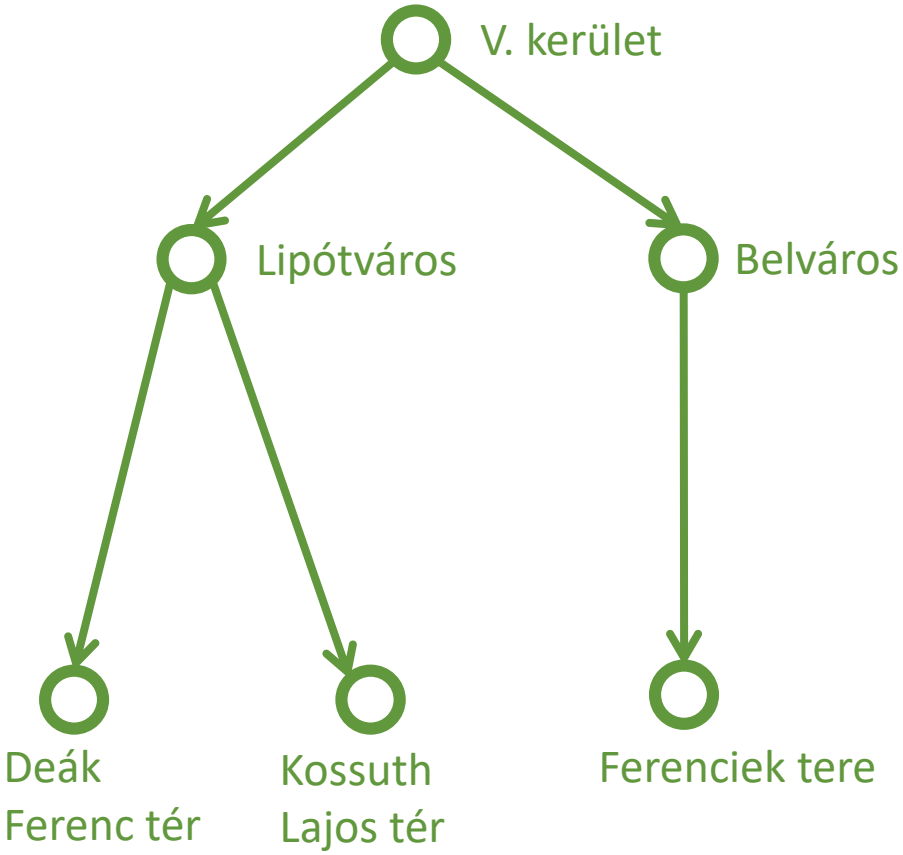
# Budapest V. kerülete



# Budapest V. kerülete – hierarchikus modell




≡

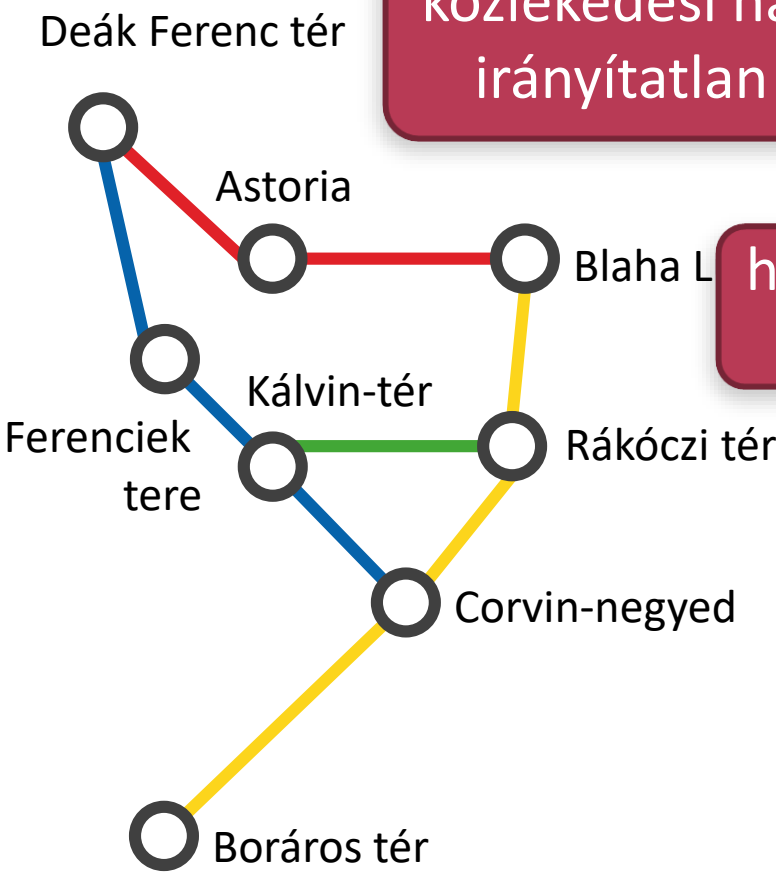


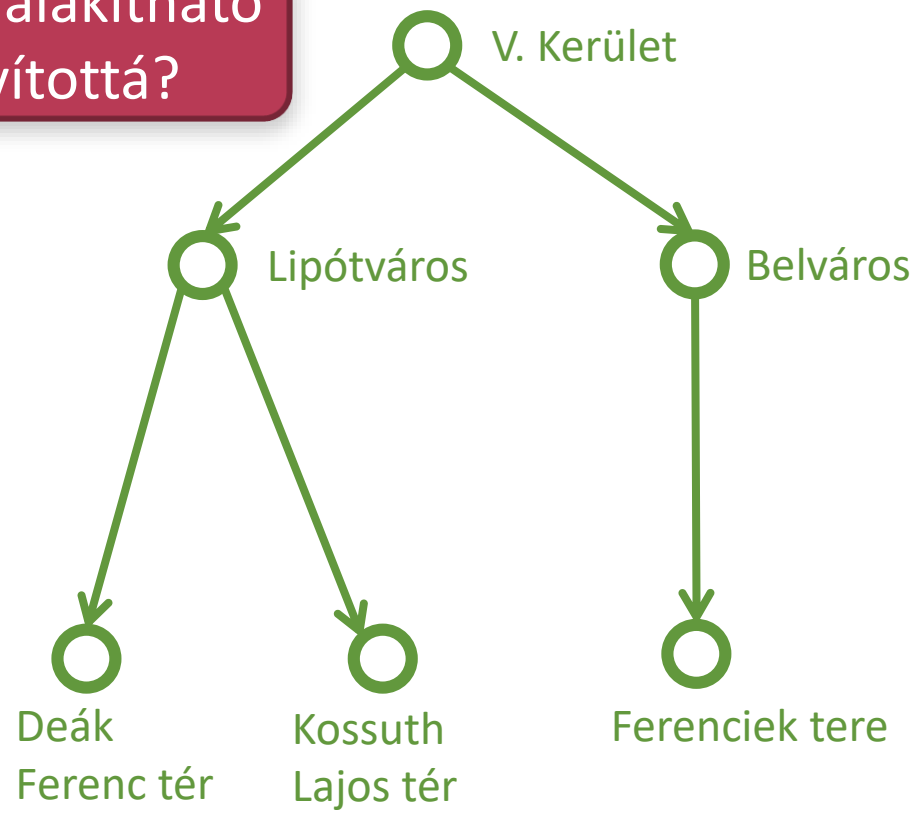
# Egyszerű gráfok

közlekedési hálózat: irányítatlan gráf

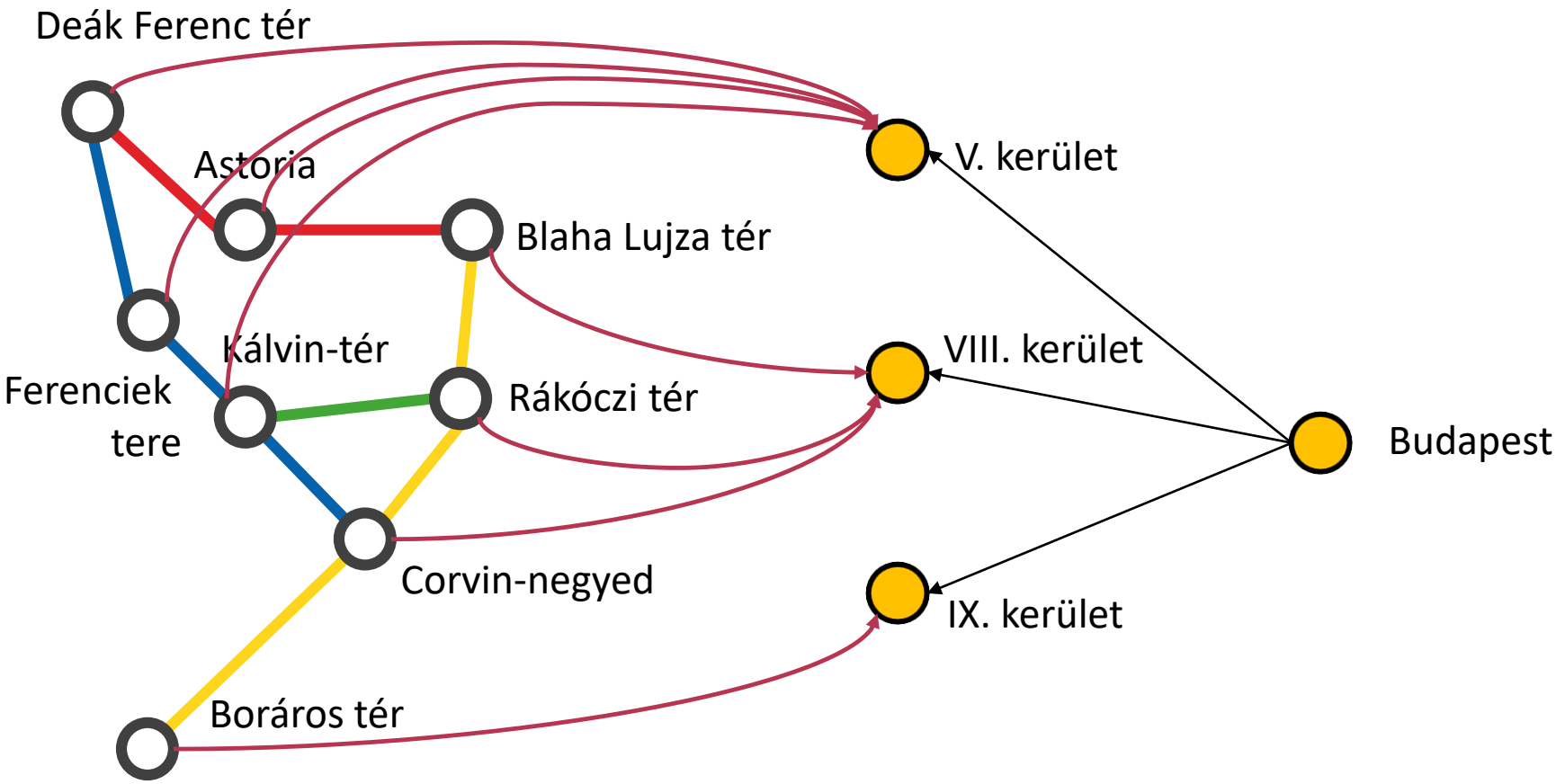
hogyan alakítható irányítottá?

területek: irányított gráf

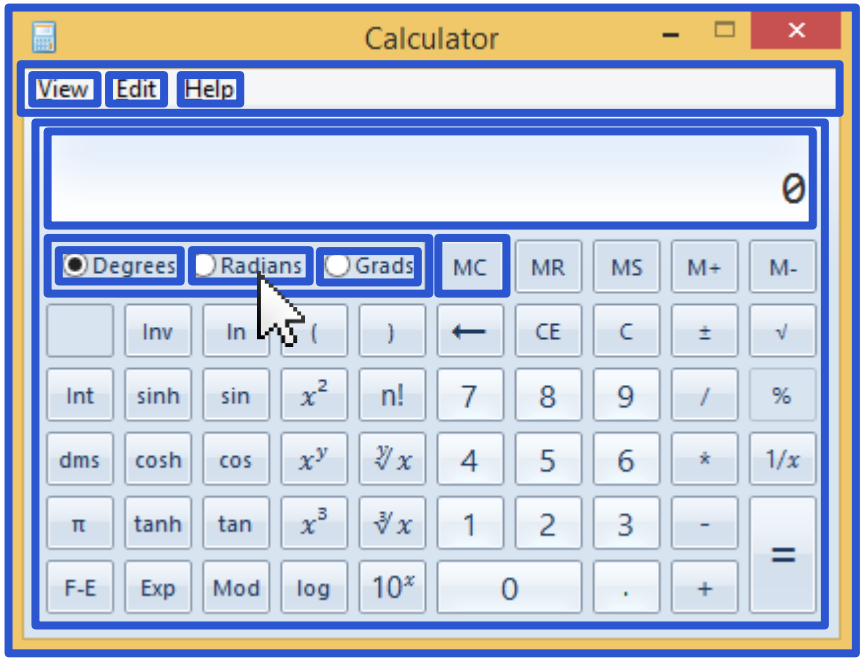


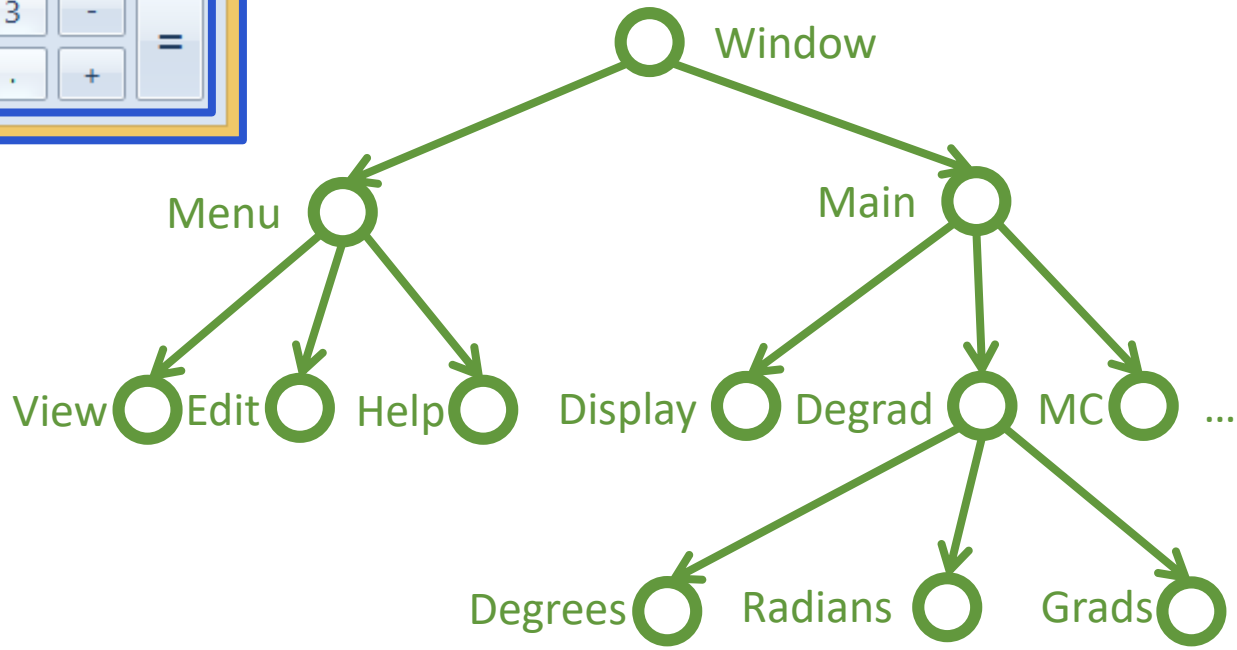
# Típusos gráfok



# Grafikus felhasználó felület




- Fa struktúra ábrázolása
  - Gráfként
  - Bennfoglaló ábrázolás



# Fastruktúra ábrázolása – fájlrendszer

C:

\Dokumentumok

\Kepek

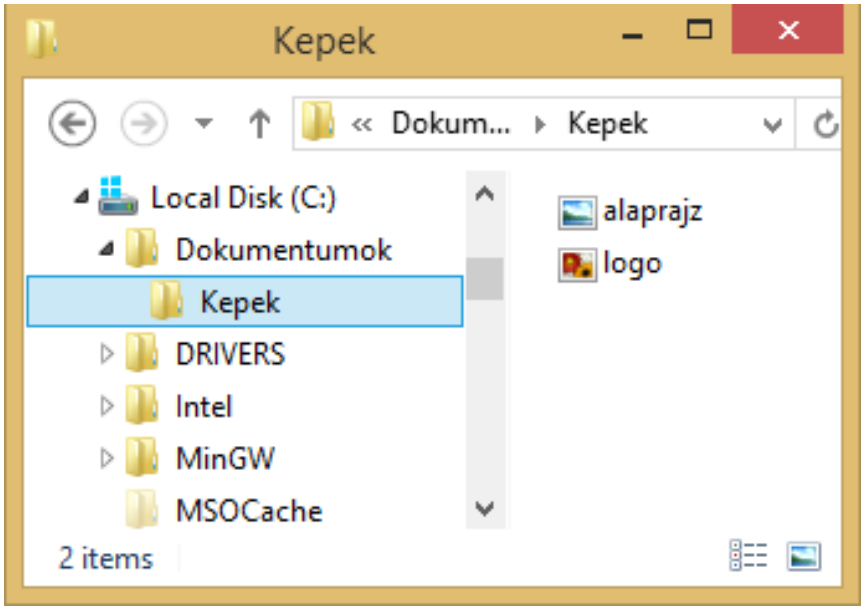
\logo.png

\alaprajz.jpg

\szerzodes.pdf

\Drivers

…




Mennyi tárhely szabadul fel a „Dokumentumok” könyvtár törlésével?

# Fájlrendszer jellemzői

- Mennyi tárhely szabadul fel a „Dokumentumok” könyvtár törlésével?
- Ehhez szükségünk van a fájlok egyes leíróinak pontos ismeretére.

| Név | Típus | Méret (kB) | Utolsó módosítás |
|---|---|---|---|
| Dokumentumok | könyvtár | | 2/2/2016 |
| szerződés.pdf | fájl | 569 | 11/9/2015 |
| Képek | könyvtár | | 2/2/2016 |
| logó.png | fájl | 92 | 3/6/2015 |
| alaprajz.jpg | fájl | 1226 | 2/2/2016 |

- Ezen információk, és a fájlhierarchia együttes ismeretével megválaszolható a kérdés.

# TULAJDONSÁGOK MODELLEZÉSE

*Milyen jellemzői vannak az egyes komponenseknek?*

Pl.

- Táblázatok
- Adatbázisok

# Definíció: Tulajdonság

- A modell elemeinek **tulajdonságait** is jellemezhetjük.

A **tulajdonság** egy a modell elemein értelmezett **függvény**, amelyet

- a **modellelemeken** értelmezünk,
- **parciális** (lehet kitöltetlen),
- **értékkészlete** a tulajdonságok értékeinek halmaza.

# Táblázatos ábrázolás

- **Táblázat sora** = modellelem
- **Táblázat oszlopa** = tulajdonság

| Név | Típus | Méret (kB) | Utolsó módosítás |
|---|---|---|---|
| Dokumentumok | könyvtár | | 2016.02.02 |
| szerződés.pdf | fájl | 569 | 2015.11.09 |
| Képek | könyvtár | | 2016.02.02 |
| logó.png | fájl | 92 | 2015.03.06 |
| alaprajz.jpg | fájl | 1226 | 2016.02.02 |

NULL / NA attribútumok

# Definíció: Szűrés

A **szűrés** művelet során

- a modell elemein kiértékelünk egy **szűrési feltételt**,
- azt a részmodellt tartjuk meg, amelynek elemei megfelelnek a feltételnek.

- `Típus = "fájl"`

| Név | Típus | Méret (kB) | Utolsó módosítás |
|---|---|---|---|
| szerződés.pdf | fájl | 569 | 2015.11.09 |
| logó.png | fájl | 92 | 2015.03.06 |
| alaprajz.jpg | fájl | 1226 | 2016.02.02 |

- `Méret > 1000 kB`

| Név | Típus | Méret (kB) | Utolsó módosítás |
|---|---|---|---|
| alaprajz.jpg | fájl | 1226 | 2016.02.02 |

# Definíció: Vetítés

**Vetítés** során a modell egyes **jellemzőit** kiválasztjuk és a többit elhagyjuk

| Név | Típus | Méret (kB) | Utolsó módosítás |
|---|---|---|---|
| Dokumentumok | könyvtár | | 2016.02.02 |
| szerződés.pdf | fájl | 569 | 2015.11.09 |
| Képek | könyvtár | | 2016.02.02 |
| logó.png | fájl | 92 | 2015.03.06 |
| alaprajz.jpg | fájl | 1226 | 2016.02.02 |

Vetítés:

{Név, Méret}

| Név | Méret (kB) |
|---|---|
| Dokumentumok | |
| szerződés.pdf | 569 |
| Képek | |
| logó.png | 92 |
| alaprajz.jpg | 1226 |

# Példa lekérdezés: szűrés és vetítés

SELECT név, méret

FROM Állományok;

WHERE típus = ‚fájl’ AND méret> 1000

vetítés

adatforrás

szűrés

- SQL: Structured Query Language
  - Adatlekérdezés (és -manipuláció)
  - A relációs adatbáziskezelés alapja
  - Itt: minta (szűrés, vetítés, származtatás)
  - Részletesebben: Adatbázisok

https://portal.vik.bme.hu/kepzes/targyak/VITMAB00/

# TÍPUSOK

# Táblázatos ábrázolás

- **Sor** = modellelem
- **Oszlop** = jellemző (gráf elemeinek tulajdonsága)

| Név | Típus | Méret (KB) | Utolsó módosítás |
|---|---|---|---|
| Dokumentumok | könyvtár | | .02 |
| szerződés.pdf | fájl | | .09 |
| Képek | könyvtár | | 2016.02.02 |
| logó.png | fájl | 92 | 2015.03.06 |
| alaprajz.jpg | fájl | 1226 | 2016.02.02 |

parciális függvény

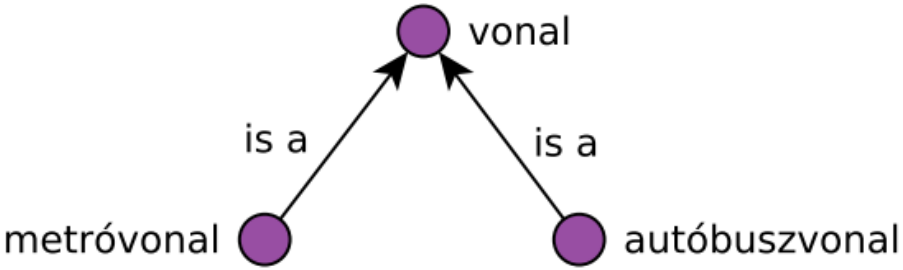
# Típus és tulajdonságok

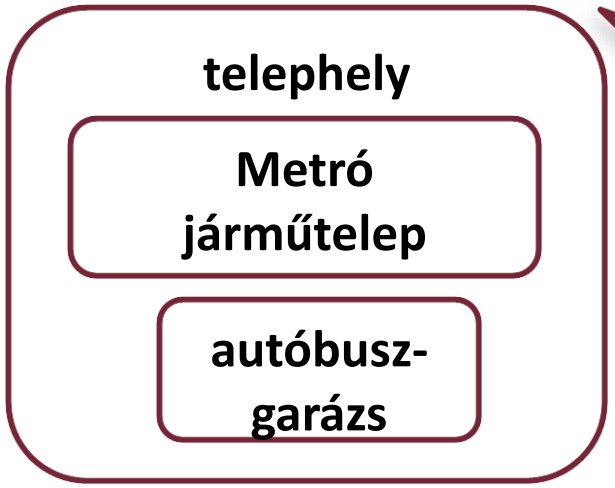
- A típus a címkében kifejezett tudás része
  - Tehát a *típus* egy kitüntetett attribútum
  - A többi jellemző: a *tulajdonságok*

- Gyakori konvenció:
  - tulajdonságok akár változhatnak is
  - típus egy elemre tipikusan állandó

# Típusgráf

- Minden csomóponttípushoz egy típuscsomópont
- Minden éltípushoz egy típusél

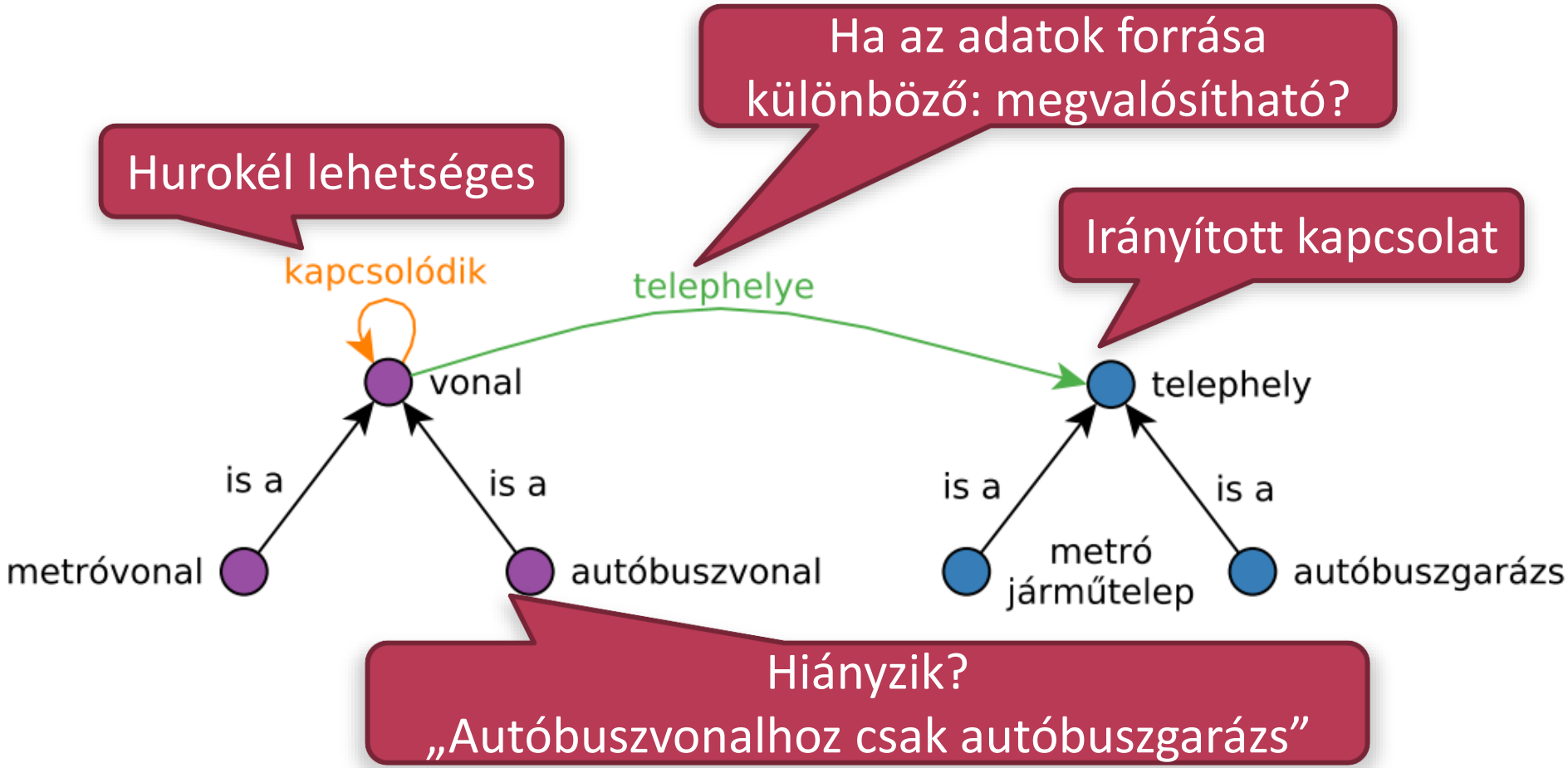


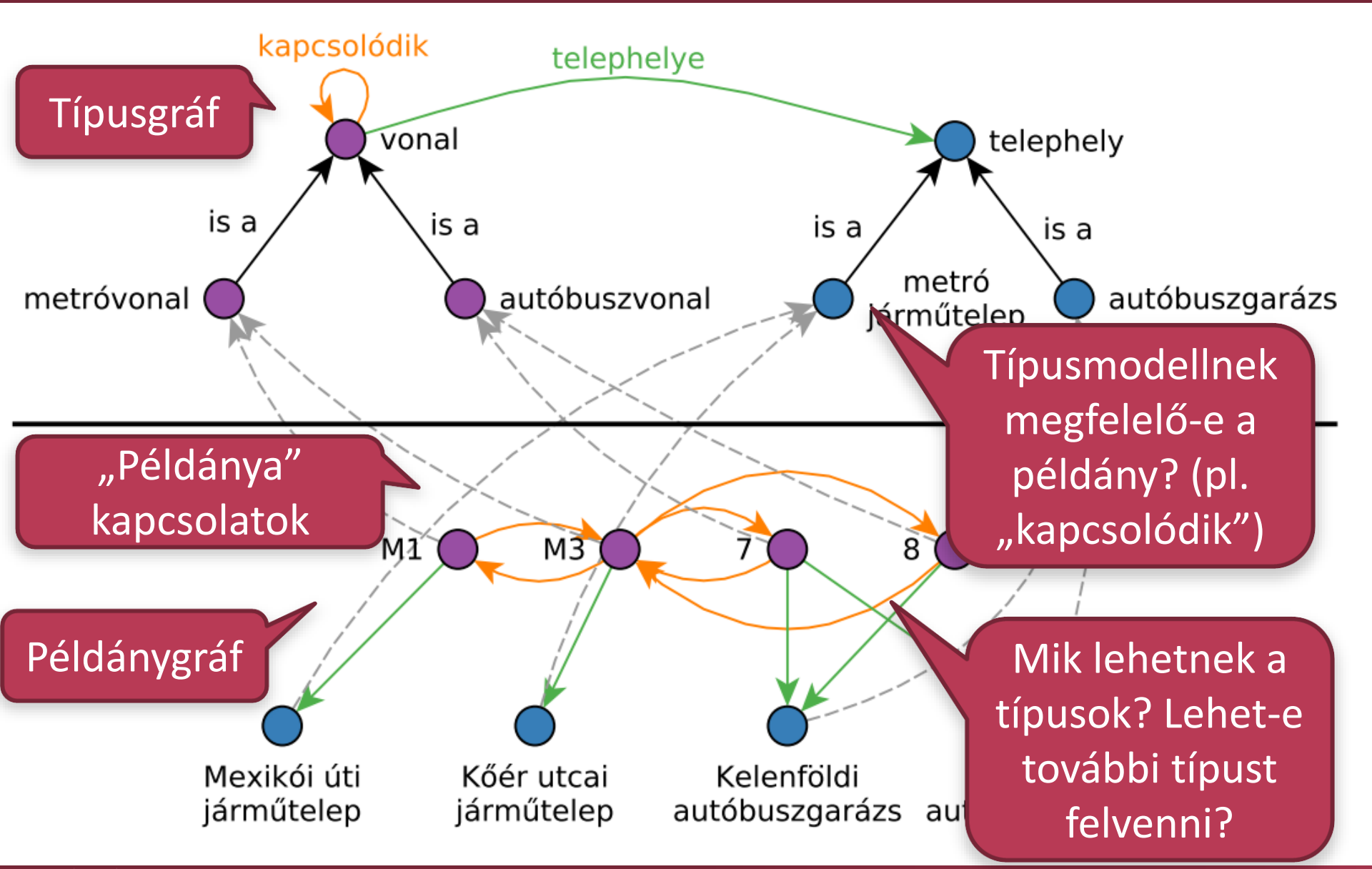

Vajon van-e közös vonás? (pl. „helyszín”, „közlekedési hálózat eleme”)

# Típusgráf

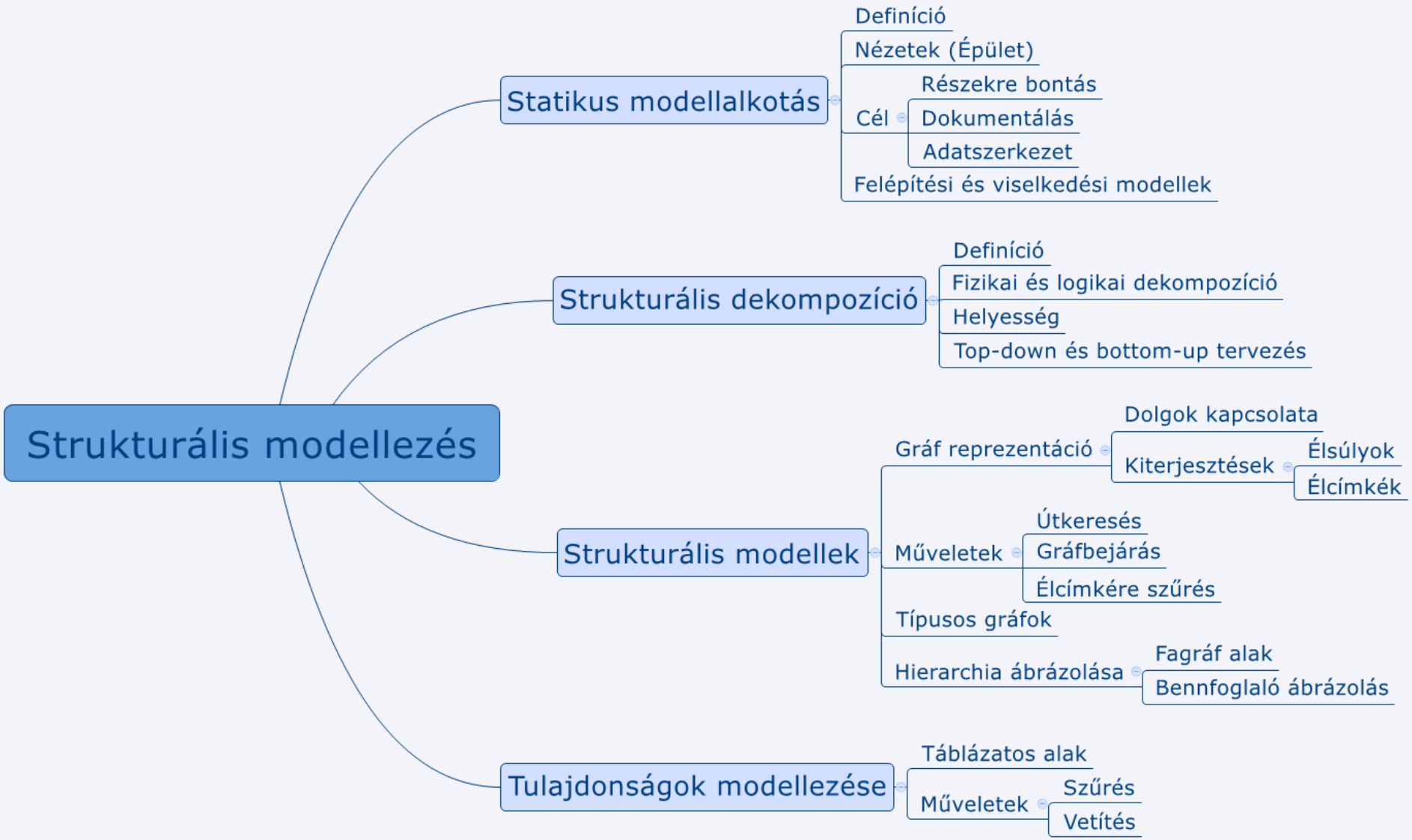
- Címkézett (típusos) kapcsolatok a típus csomópontok közt



# Típusgráf és példánygráf



# Összefoglalás

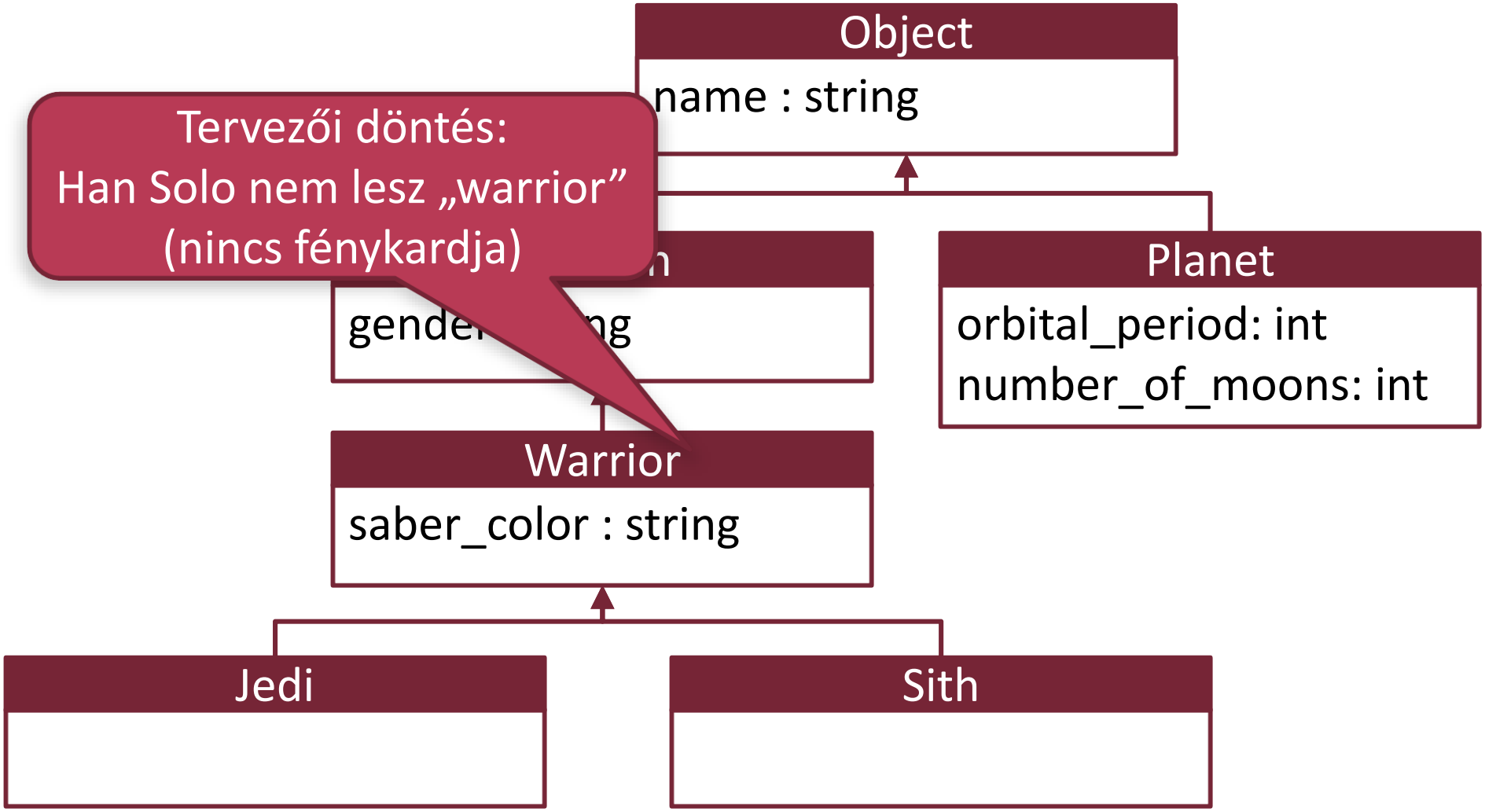


# STRUKTURÁLIS MODELL REPREZENTÁLÁSA PROGRAMBAN

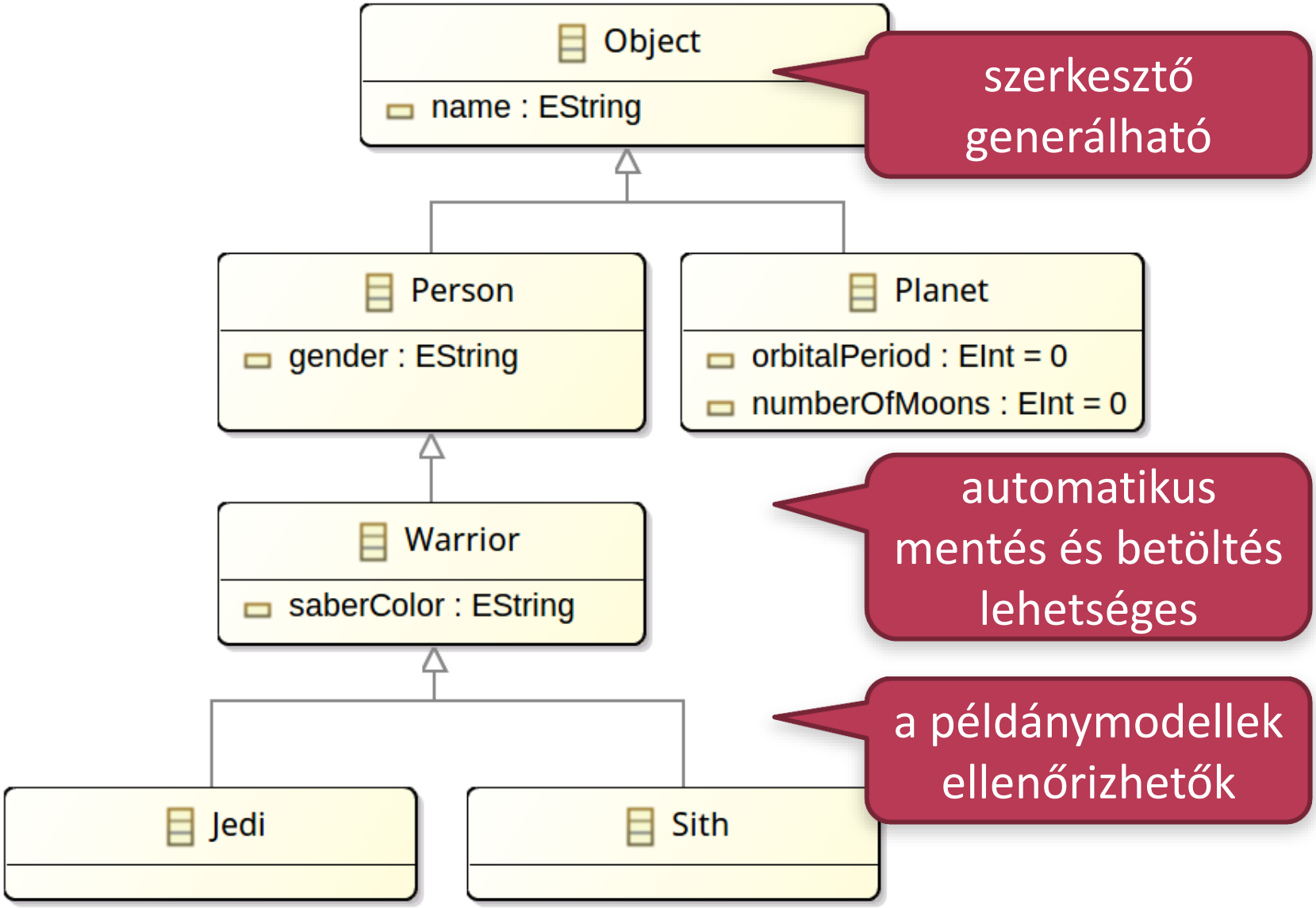
# Programozási paradigmák

- **Programozási paradigma:** egy programozási nyelv elméleti modellje
- **Strukturált programozás** (C)
  - változók struktúrába szervezése: struct
- **Objektum-orientált programozás**, OOP (C++, Java)
  - típus: *osztály*
  - példány: *objektum*
  - attribútum: *mező*
  - műveletek: *metódusok*
  - attribútumok/metódusok láthatósága, egységbe zárás stb. – bővebben ld. a Programozás alapjai 2. tárgyban

# OOP: öröklés


Object
name : string
Tervezői döntés:
Han Solo nem lesz „warrior”
(nincs fénykardja)
Planet
orbital_period: int
number_of_moons: int
Warrior
saber_color : string
Jedi
Sith

# Eclipse Modeling Framework – Ecore diagram



# Szöveges reprezentációk

- XML (Extensible Markup Language)
  - szabványos, általános célú nyelv leírónyelvek definiálására
  - ember számára (elvileg…) olvasható
- JSON (JavaScript Object Notation)
  - szöveg alapú szabvány ember által olvasható adatcserére

# XML példa: időjárás webszolgáltatás


```
▼<current>
  ▼<city id="3054643" name="Budapest">
      <coord lon="19.04" lat="47.5"/>
      <country>HU</country>
      <sun rise="2015-02-17T05:45:24" set="2015-02-17T16:10:12"/>
    </city>
    <temperature value="268.061" min="268.061" max="268.061" unit="kelvin"/>
    <humidity value="83" unit="%"/>
    <pressure value="1034.42" unit="hPa"/>
  ▼<wind>
      <speed value="2.12" name="Light breeze"/>
      <direction value="52.0001" code="NE" name="NorthEast"/>
    </wind>
    <clouds value="0" name="clear sky"/>
    <visibility/>
    <precipitation mode="no"/>
    <weather number="800" value="Sky is Clear" icon="01n"/>
    <lastupdate value="2015-02-17T20:11:20"/>
  </current>
```

# JSON példa: Google Maps API


```
{ ⊟
  "results":[ ⊟
    { ⊟
      "address_components":[ ⊞ ],
      "formatted_address":"1600 Amphitheatre Pkwy, Mountain View, CA 94043, USA",
      "geometry":{ ⊟
        "location":{ ⊟
          "lat":37.42291810,
          "lng":-122.08542120
        },
        "location_type":"ROOFTOP",
        "viewport":{ ⊞ }
      },
      "types":[ ⊞ ]
    }
  ],
  "status":"OK"
}
```

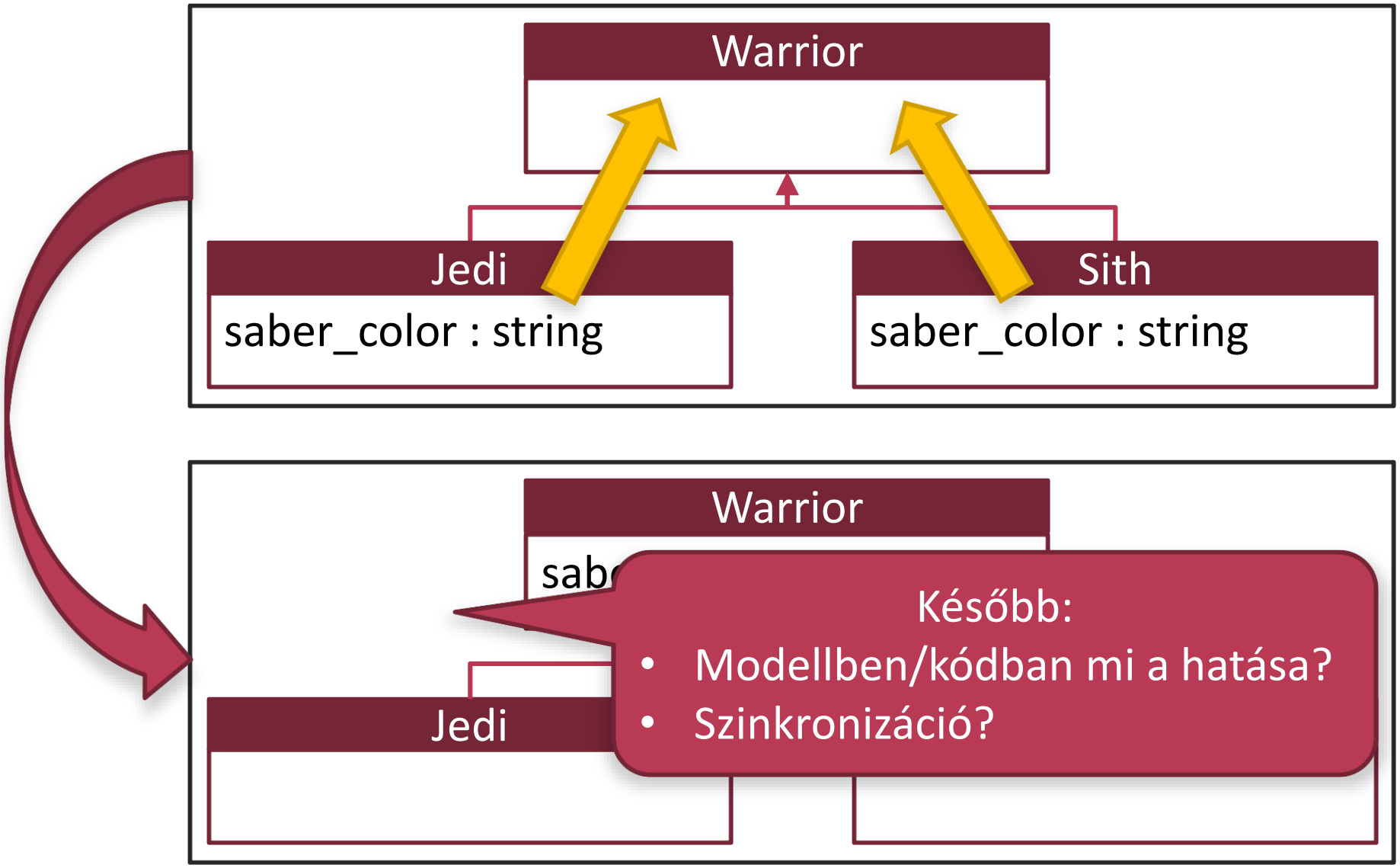
# Modellezési nyelvek

- UML (Unified Modeling Language)
  - általános célú modellezési nyelv
- AADL (Architecture Analysis & Design Language)
  - architektúraleíró nyelv
- SysML (Systems Modeling Language)
  - UML-alapú általános modellezési nyelv rendszertervezési célokra
- EMF (Eclipse Modeling Framework, Ecore)
  - modellezési nyelv készítésére alkalmas
- (Adatközpontú modellezés, szakterületi modellek)

# Kitekintés: refaktoring

- **Code refactoring is the process of restructuring existing computer code** – changing the **factoring** – without changing its external behavior. Refactoring improves nonfunctional attributes of the software. Advantages include
  - improved code readability and
  - reduced complexity to improve source code maintainability, and
  - create a more expressive internal architecture or object model to improve extensibility.
- Bővebben: felsőbbéves szoftveres tantárgyak

# Refaktoring példa: „pull up”



# ÖSSZEFOGLALÁS

# Definíció: Strukturális modell

A **strukturális modell** statikus. Kiindulópontja egy (rész)rendszer, amelyet a `része` reláció mentén alkotórészekre bont.

Az **alkotórészek** lehetnek:

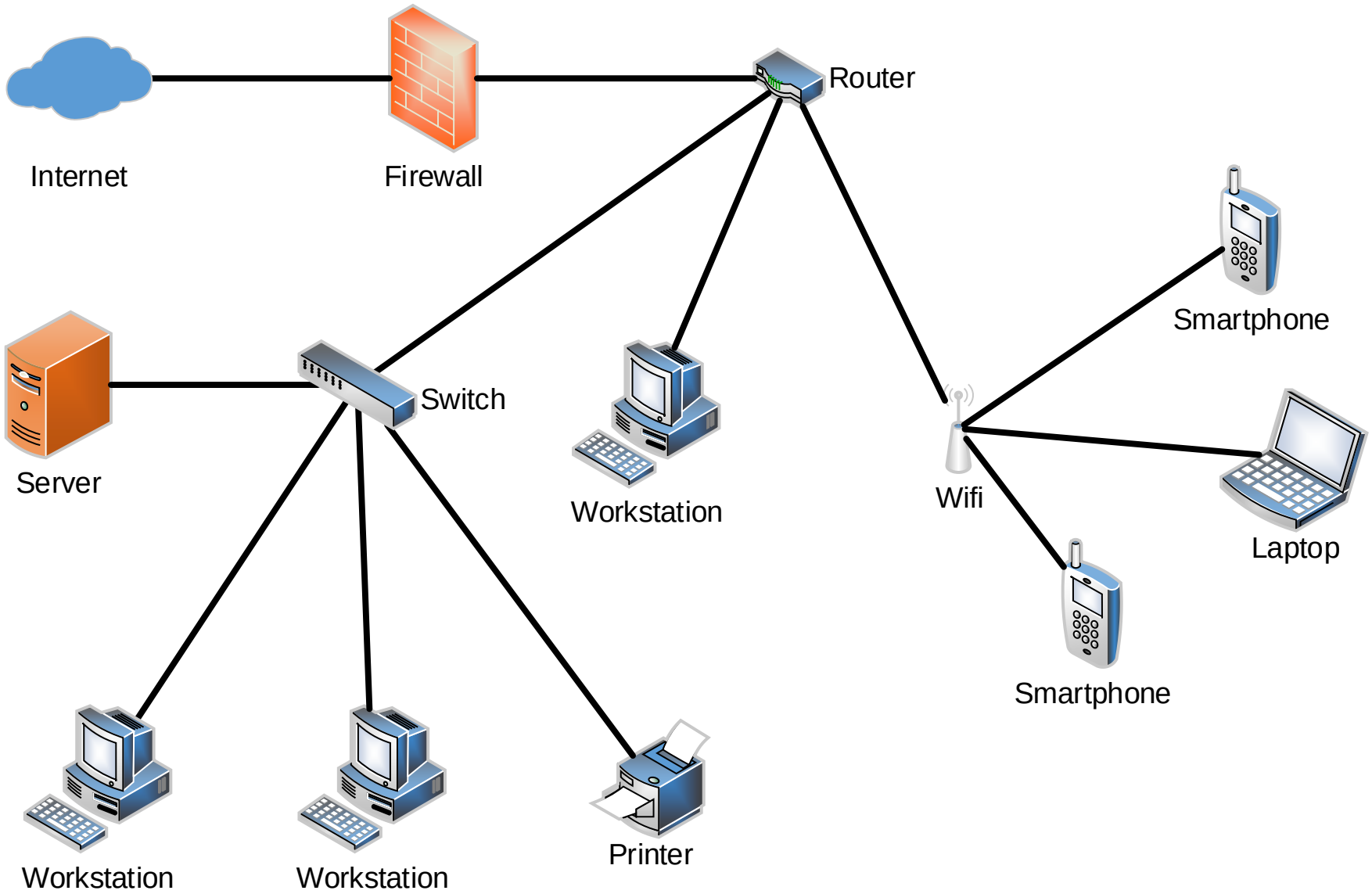
- tovább bontott **részrendszerek** vagy
- tovább nem bontott (*elemi*) **komponensek**.

A strukturális modell a rendszer felépítését (struktúráját)

- az alkotórészei,
- azok **tulajdonságai** és
- egymással való **viszonya**

alapján reprezentálja.

# Példa: (céges) hálózat



# Definíció: Dekompozíció

- *„A **strukturális modell** statikus. Kiindulópontja egy (rész)rendszer, amelyet a `része` reláció mentén alkotórészekre bont.”*

> A **dekompozíció** („faktoring”) egy összetett probléma vagy rendszer kisebb részekre bontása, amelyek könnyebben érthetők, fejleszthetők és karbantarthatók.

- Hierarchikus dekompozíció:
  - Többlépcsős rész-egész viszony
- Része reláció tartalma szerint:
  - Fizikai dekompozíció: `része`=térbeli tartalmazás
  - Logikai dekompozíció: pl. `része`=funkció