Megbízhatósági modellezés és analízis: Mire jó ez egyáltalán?

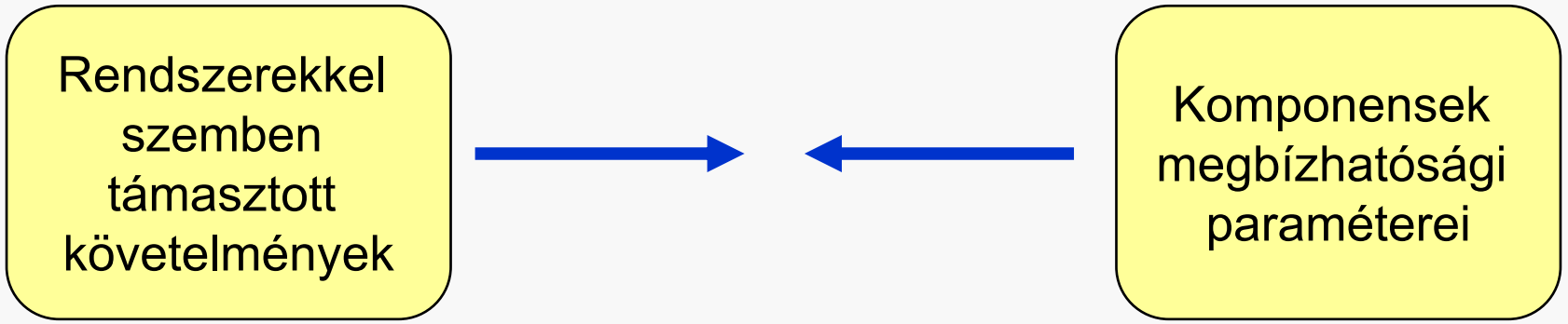
Majzik István
Méréstechnika és Információs Rendszerek Tanszék

2007. november 14.

Bevezetés: Mi a szolgáltatásbiztonság?

- Szolgáltatásbiztonság: Igazoltan bízni lehet a rendszer szolgáltatásában
  - Megbízhatóság: Folyamatos hibamentes működés
  - Rendelkezésre állás: Használatra kész rendszer
  - Biztonságosság: Katasztrofális következmények (baleset, káreset) nélküli működés
  - Bizalmasság: Nincs jogosulatlan információközlés
  - Integritás: Hibás változ(tat)ás elkerülése
  - Karbantarthatóság: Javítás és fejlesztés lehetősége
- Terjedő fogalom: Resilience
  - Szolgáltatásbiztonság + adatbiztonság + dinamikus (adaptív, mobil, ad-hoc) működés

Miért van az analízisre szükség?



Miért van az analízisre szükség?

Rendszerekkel szemben támasztott követelmények

Komponensek megbízhatósági paraméterei

Szolgáltatási szint szerződések (SLA):

- Ügyfél által elvárt jellemzők (pl. rendelkezésre állás)
- Telekom szolgáltatások szerver rendszerei:
  „Öt kilences”: 99,999% (5 perc/év kiesés)

Biztonságkritikus rendszerek:

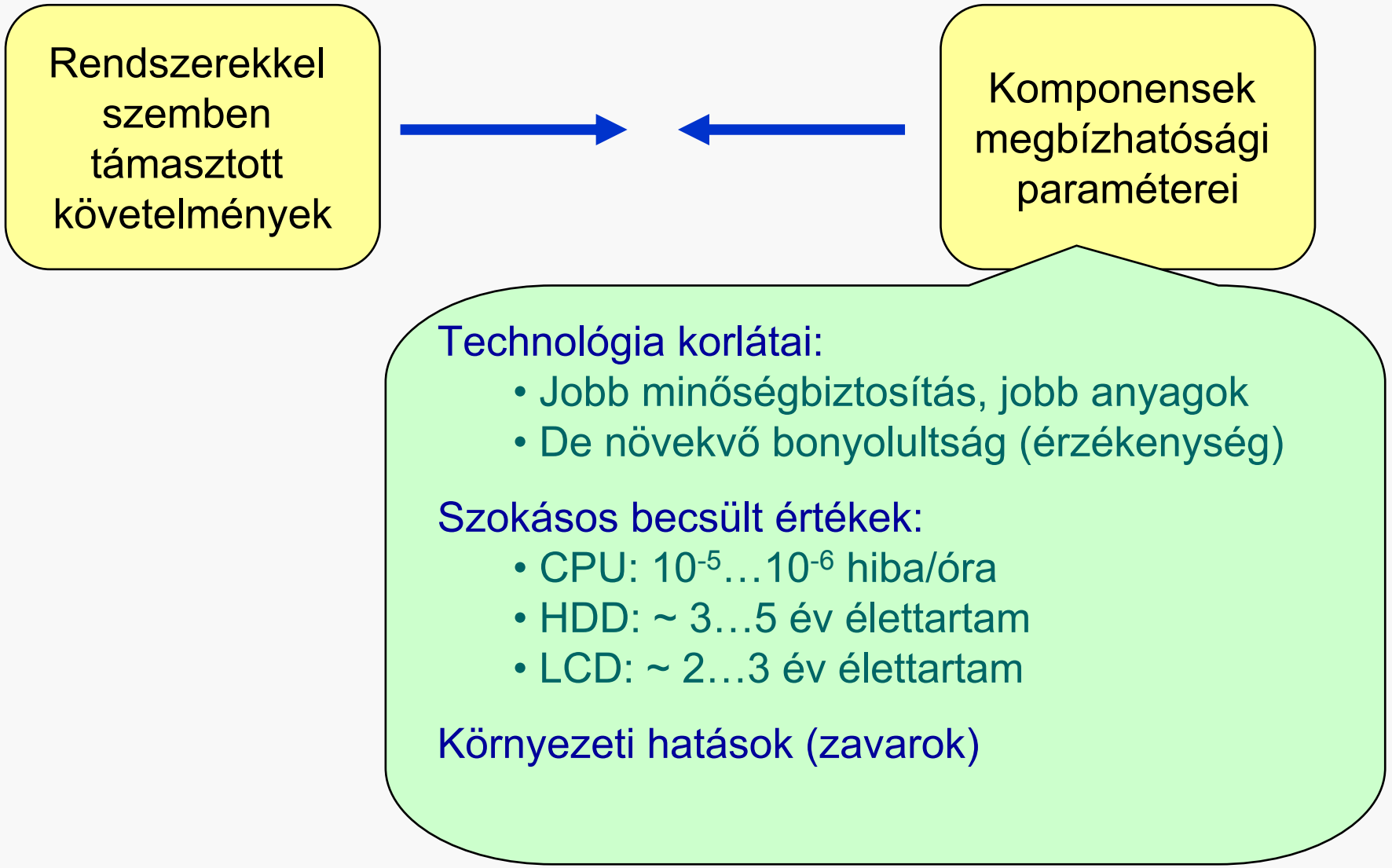
- Szabvány előírások a véletlen hibák gyakoriságára
- Biztonságintegritási szintek (SIL) szerint

SIL	Biztonságkritikus funkció hibája / óra
1	$10^{-6} \leq THR < 10^{-5}$
…	…
4	$10^{-9} \leq THR < 10^{-8}$

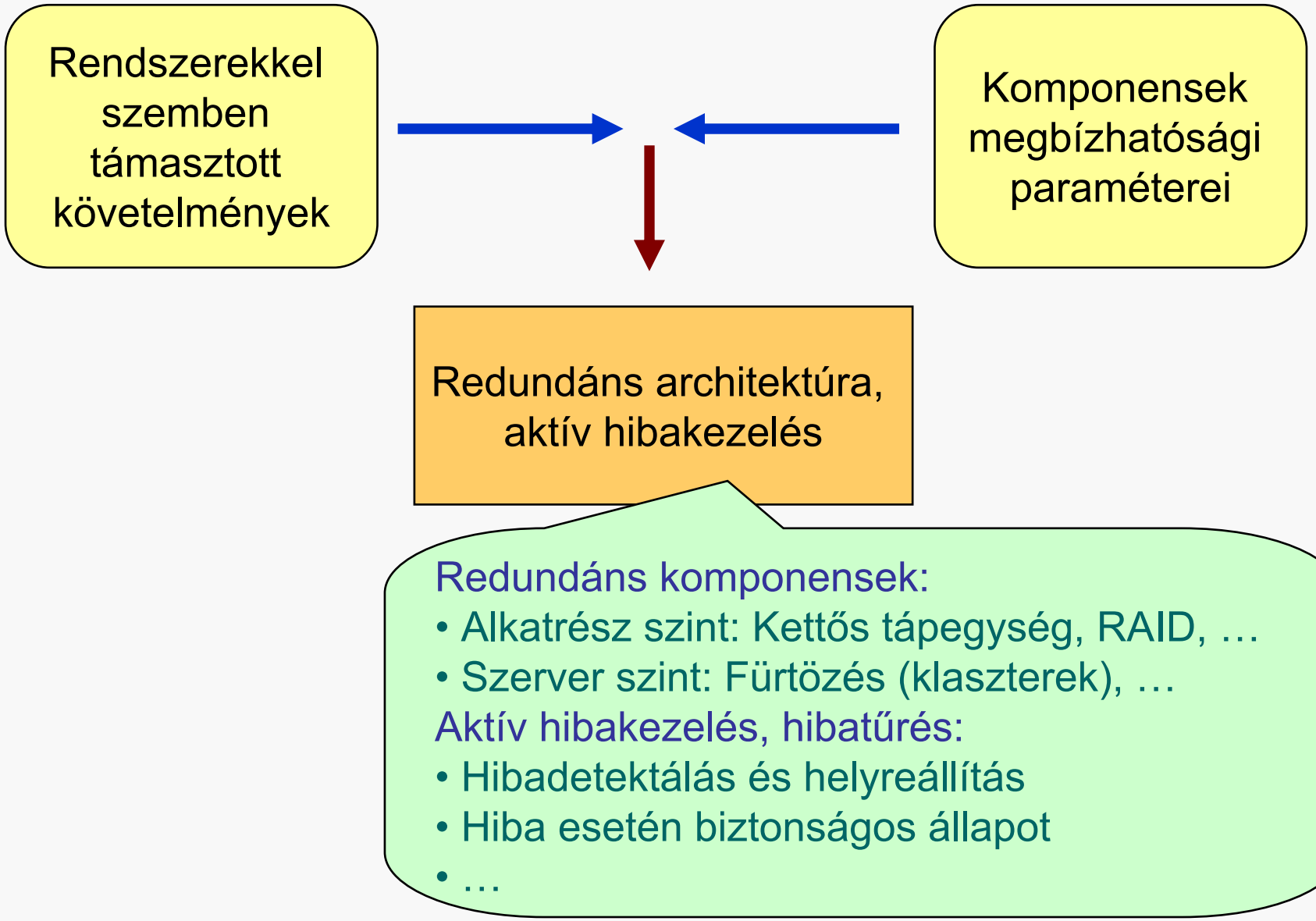
Ha 15 év az élettartam, akkor ez alatt kb. 750 berendezésből 1-ben lesz hiba

Hiba nélküli működés ~ 11.000 év?

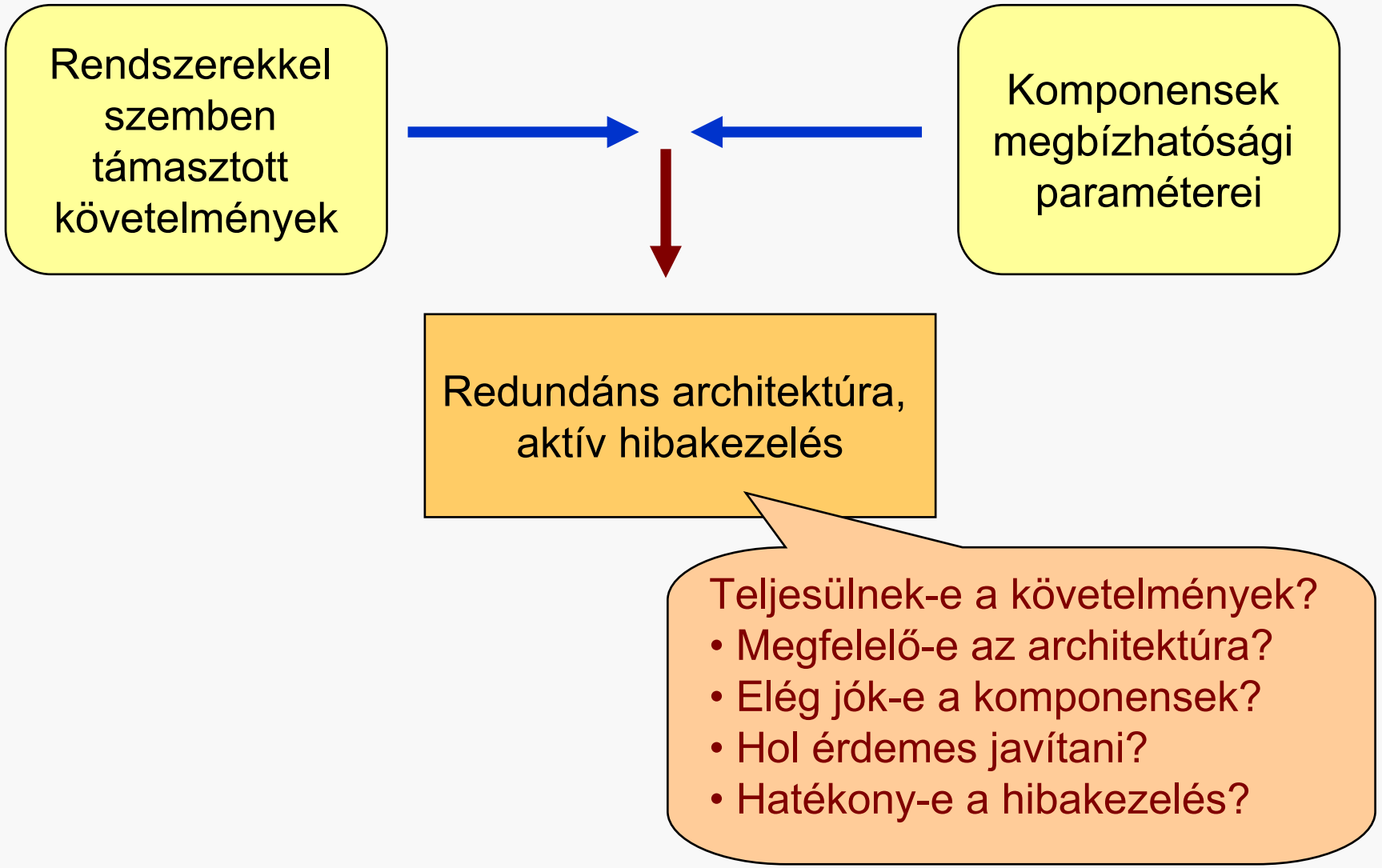
Miért van az analízisre szükség?



Miért van az analízisre szükség?



Miért van az analízisre szükség?



Milyen analízis módszerek vannak?

- Mérések működés közben:
  Hibanaplózás és az eredmények feldolgozása
  - Jellegzetes hibák illetve faktorok kiszűrése
  - Statisztikai jellemzők kinyerése
- Kísérleti kiértékelés: Hibainjektálás (várható hibák bevitele)
  - Hibakezelési eljárások, hibatűrés vizsgálata
  - Modellen vagy prototípuson is elvégezhető
  - Hibák lehetnek: hardver, szoftver, környezeti (sugárzás), …
- Megbízhatósági modellezés és modell alapú számítások:
  - Komponens meghibásodási, hibaterjesztési, javítási folyamatok modellezése
    - Adott előfeltételezések (pl. független hibák) és absztrakció mellett
  - Komponensek (ismert) paraméterei alapján rendszerszintű jellemzők számítása, architektúra-változatok összevetése

Mik a módszerek korlátai?

- Mérések működés közben:
  Hibanaplózás
  - Jellegzetes h
  - Statisztikai jellemzők kinyerése
- Kísérleti kiértékelés: Hibainjektálás (várható hibák bevitele)
  - Hibakezelési
  - Mo
  - Hibák
- Megbízhatósági modellezés és modell alapú számítások:
  - Kom

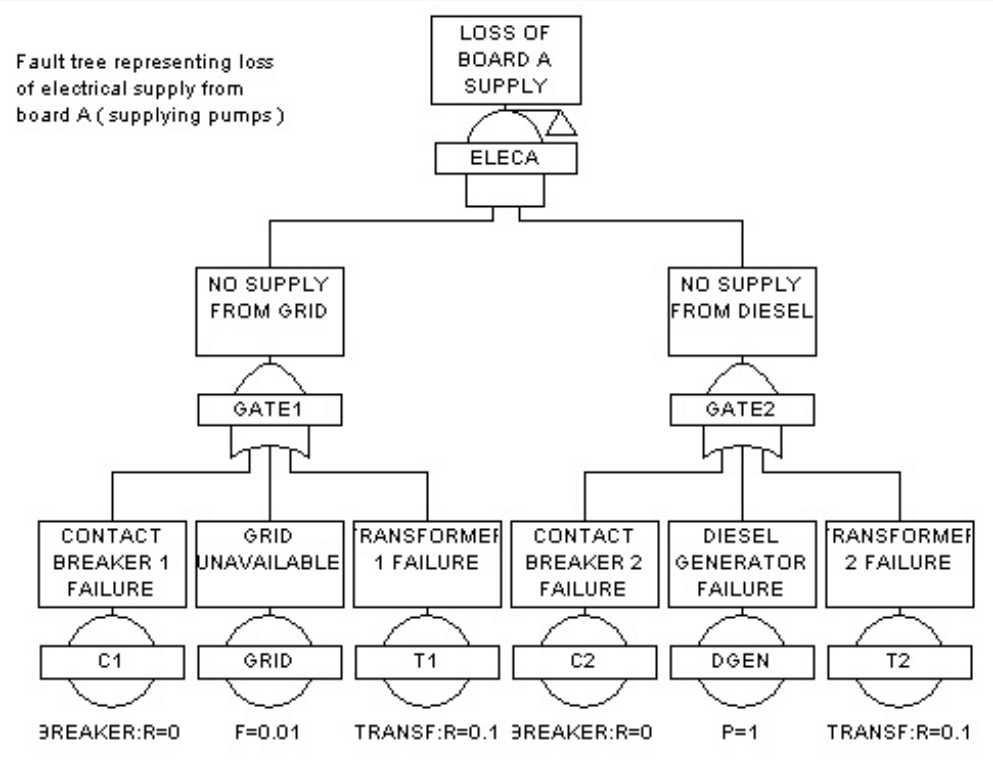
- Hosszú idejű ill. nagyszámú mérés kell
- Tervezési időben nem ad segítséget

- Kérdéses a magas szinten bevitt hibák valósághűsége
- Költséges az alacsony szintű hibák bevitele

- Mennyire valósághű a modellezés?
  - Jogosak-e az előfeltételezések és az absztrakció?
- Ismertek-e a komponensek paraméterei?
  - Szoftver esetén a tesztelés adhat támpontot
- Mennyire bonyolult lesz a modell, megoldható-e?
  - Analitikus megoldás nem mindig van (szimuláció kell)

Megbízhatósági modellezés

- Kombinatorikus modellek
  - Hibafa, eseményfa, hibamód- és hatás analízis
  - Rendszerhiba: Komponensek független hibáinak statikus kombinációjaként vagy időbeli forgatókönyveként felírva
  - A hibák és hatások szisztematikus áttekintését támogatják

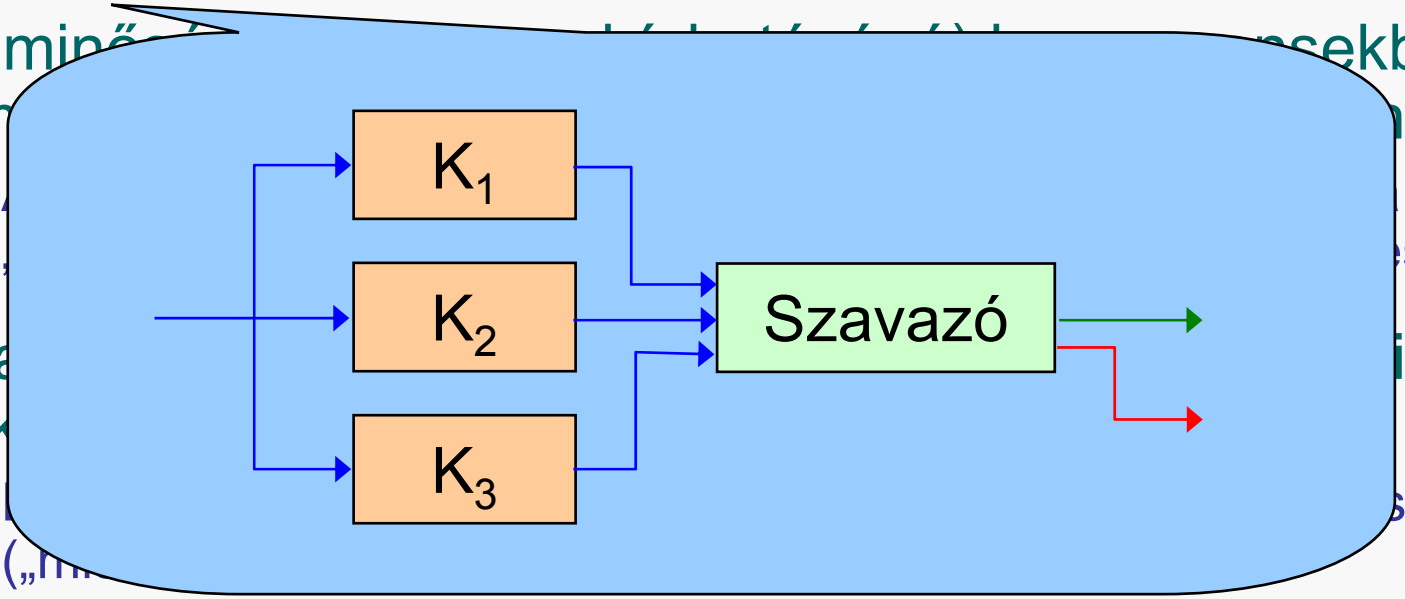


Megbízhatósági modellezés

- Kombinatorikus modellek
  - Hibafa, eseményfa, hibamód- és hatás analízis
  - Rendszerhiba: Komponensek független hibáinak statikus kombinációjaként vagy időbeli forgatókönyveként felírva
  - A hibák és hatások szisztematikus áttekintését támogatják
- Sztochasztikus állapot-alapú modellek
  - Markov-láncok, sztochasztikus Petri-hálók
    - Feltételezés: Meghibásodási, javítási folyamat: exp. eloszlás
  - Rendszerhiba: Adott állapot-partíció eléréséhez kötve, ennek bekövetkezési valószínűségét számítva
  - Finomabb modellezésre ad lehetőséget:
    - Degradált működési állapotok
    - Összefüggő (közös módusú) hibák
    - Javítási függőségek, javítási illetve helyreállítási stratégiák

Érdekes analízis eredmények

- Redundáns architektúrák analízise
  - Rossz minőségű (kis megbízhatóságú) komponenseket nem lehet replikációval és szavazással „feljavítani”
    - Az eredő megbízhatóság csökken az egyszereshez képest
  - Jó min
    - A
  - Sza
    - Rendszeres karbantartással fenntartható magas rendelkezésre állás (pl. repülőgép fedélzeti rendszerek, erőművi rendszerek)

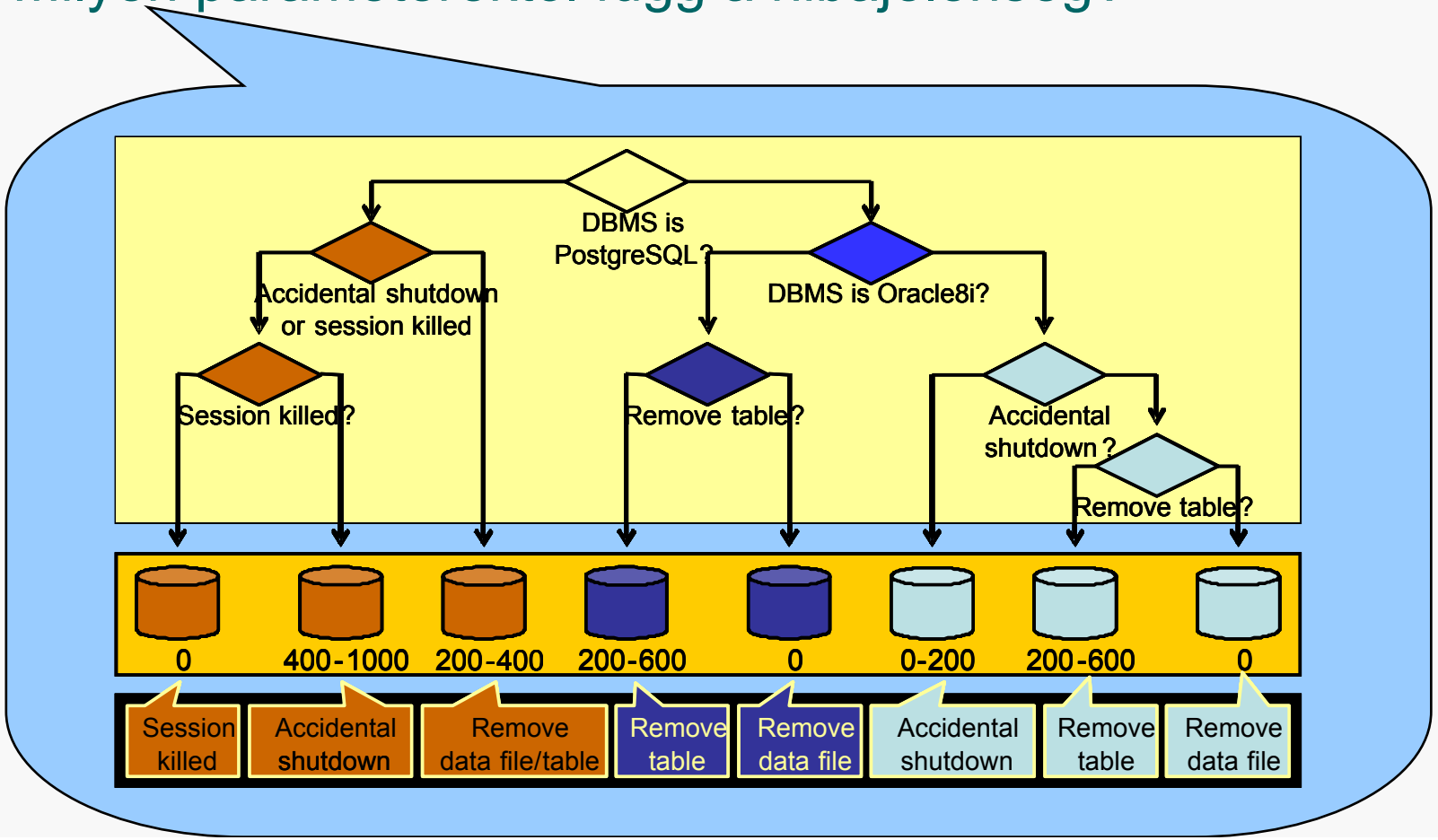


Érdekes analízis eredmények

- Redundáns architektúrák analízise
  - Rossz minőségű (kis megbízhatóságú) komponenseket nem lehet replikációval és szavazással „feljavítani”
    - Az eredő megbízhatóság csökken az egyszereshez képest
  - Jó minőségű (nagy megbízhatóságú) komponensekből sem mindig érdemes redundáns rendszert kialakítani
    - A redundancia kezelés (pl. átkapcsoló logika) bejövő hibája „leronthatja” a rendszert az egyszeres komponenshez képest
  - Szavazásos redundáns rendszer: az első rendszerhiba bekövetkezésének várható ideje kisebb lesz
    - De ennél rövidebb időtartam túlélési valószínűsége jobb lesz („misszió” idejének túlélése hiba nélkül)
    - Rendszeres karbantartással fenntartható magas rendelkezésre állás (pl. repülőgép fedélzeti rendszerek, erőművi rendszerek)

Aktuális kutatási területeink I.

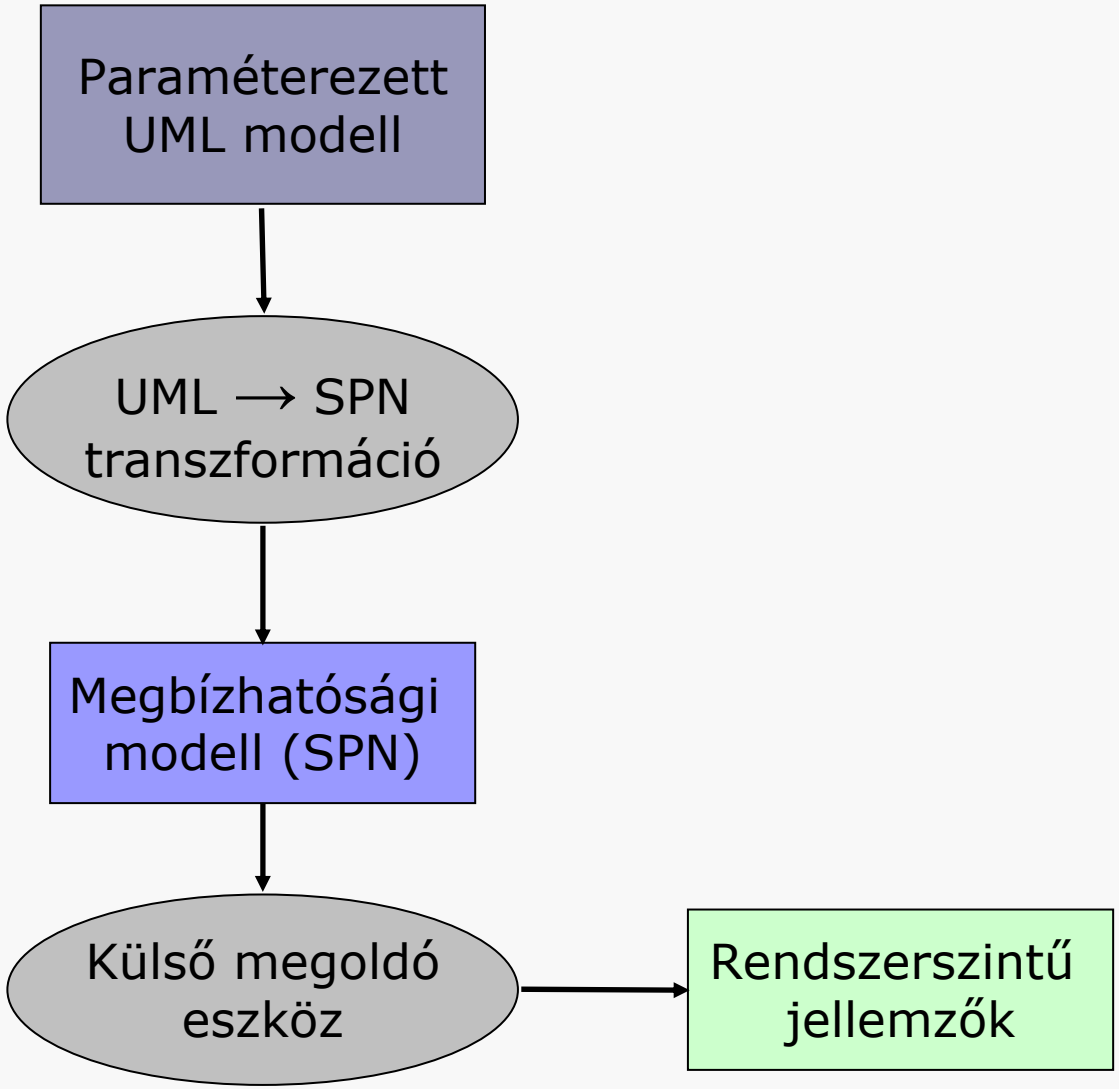
- Hibainjektálás: Szoftver alapú hibák
- Hibanaplók feldolgozása adatbányászattal
  - Milyen paraméterektől függ a hibajelenség?



Aktuális kutatási területeink II.

- Megbízhatósági modellek automatikus generálása mérnöki modellek alapján
- Általános séma:
  1. Rendszer modellje (pl. UML, AADL)
  2. Komponensek felparaméterezése
     - Típus (szoftver, hardver, állapota van-e)
     - Meghibásodási gyakoriság, hiba lappangási idő, javítási idő
     - Hibaterjedési valószínűség komponensek között
     - Redundancia típusa
  3. Megbízhatósági modell generálása
  4. Rendszerparaméterek számítása a modell megoldásával (pl. Markov-lánc állapotvalószínűség)

Eszközfejlesztés



Eszközfejlesztés

Paraméterezett UML modell

UML ⟶ SPN transzformáció

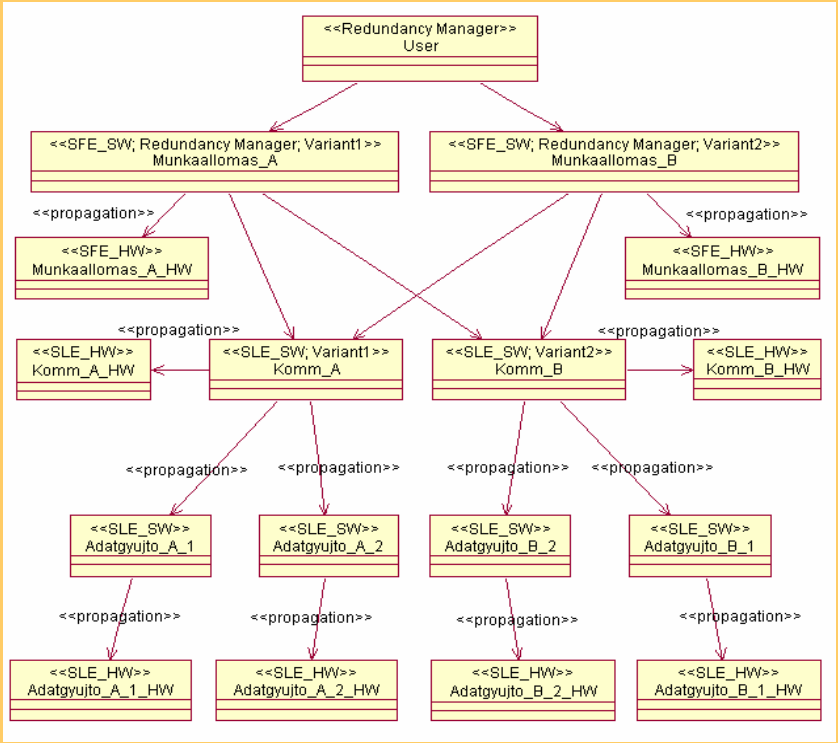
Megbízhatósági modell (SPN)

Külső megoldó eszköz

Re… jellemzők

Rendszermodell

UML osztály/objektum diagram



Eszközfejlesztés

Paraméterezett UML modell

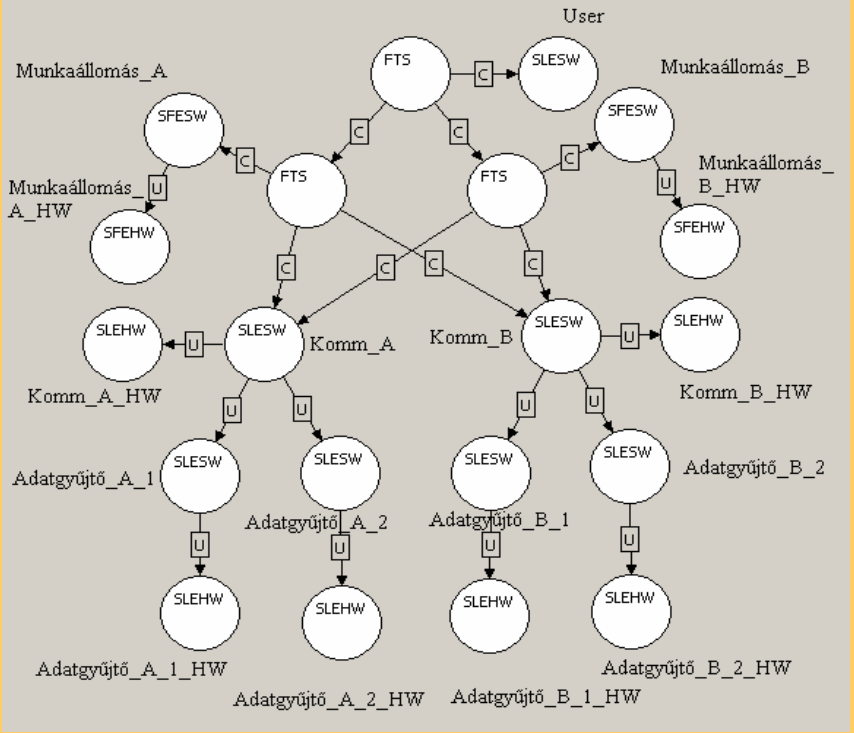
UML ⟶ SPN transzformáció

Megbízhatósági modell (SPN)

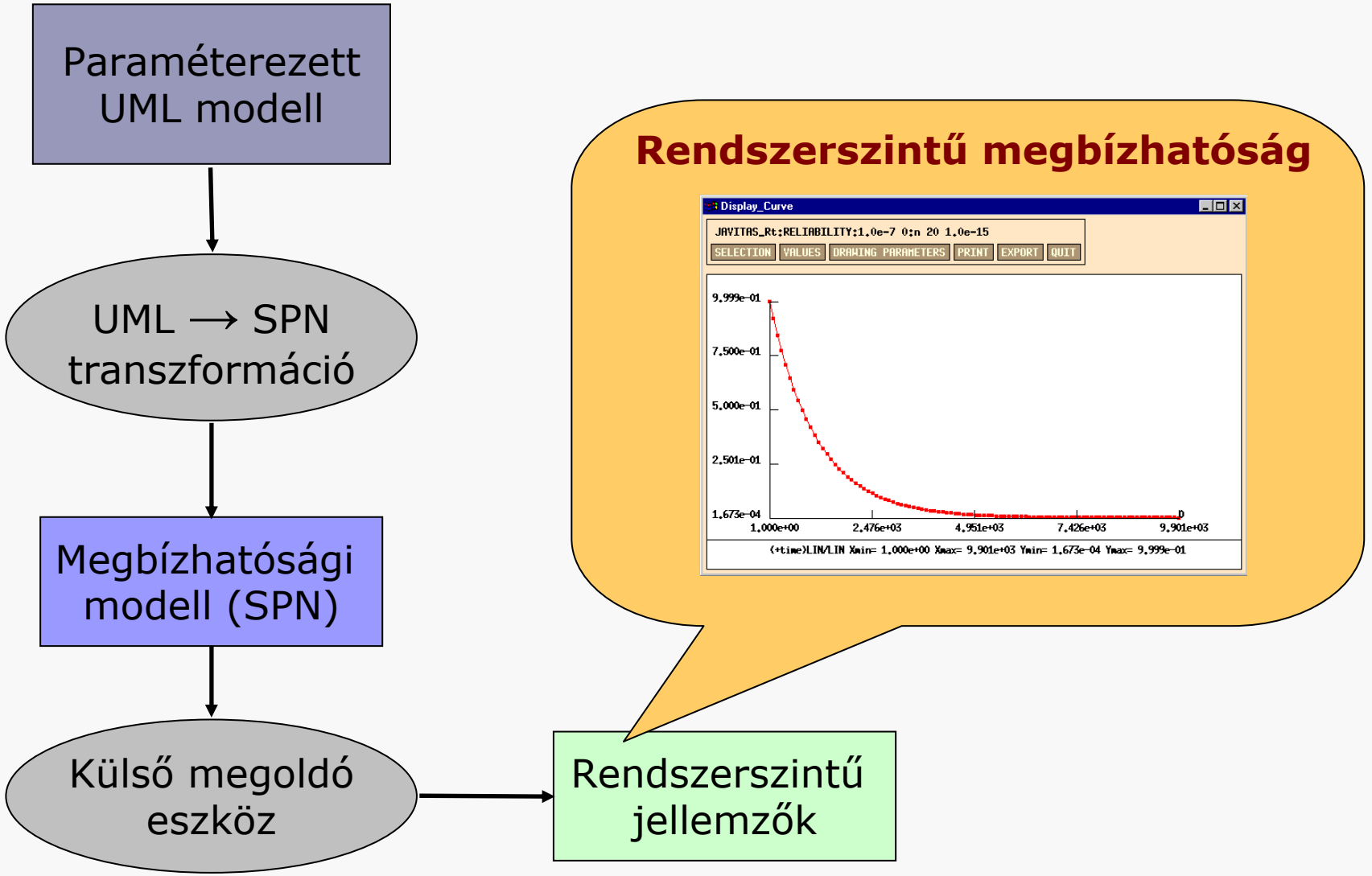
Külső megoldó eszköz

Rendszerszintű jellemzők

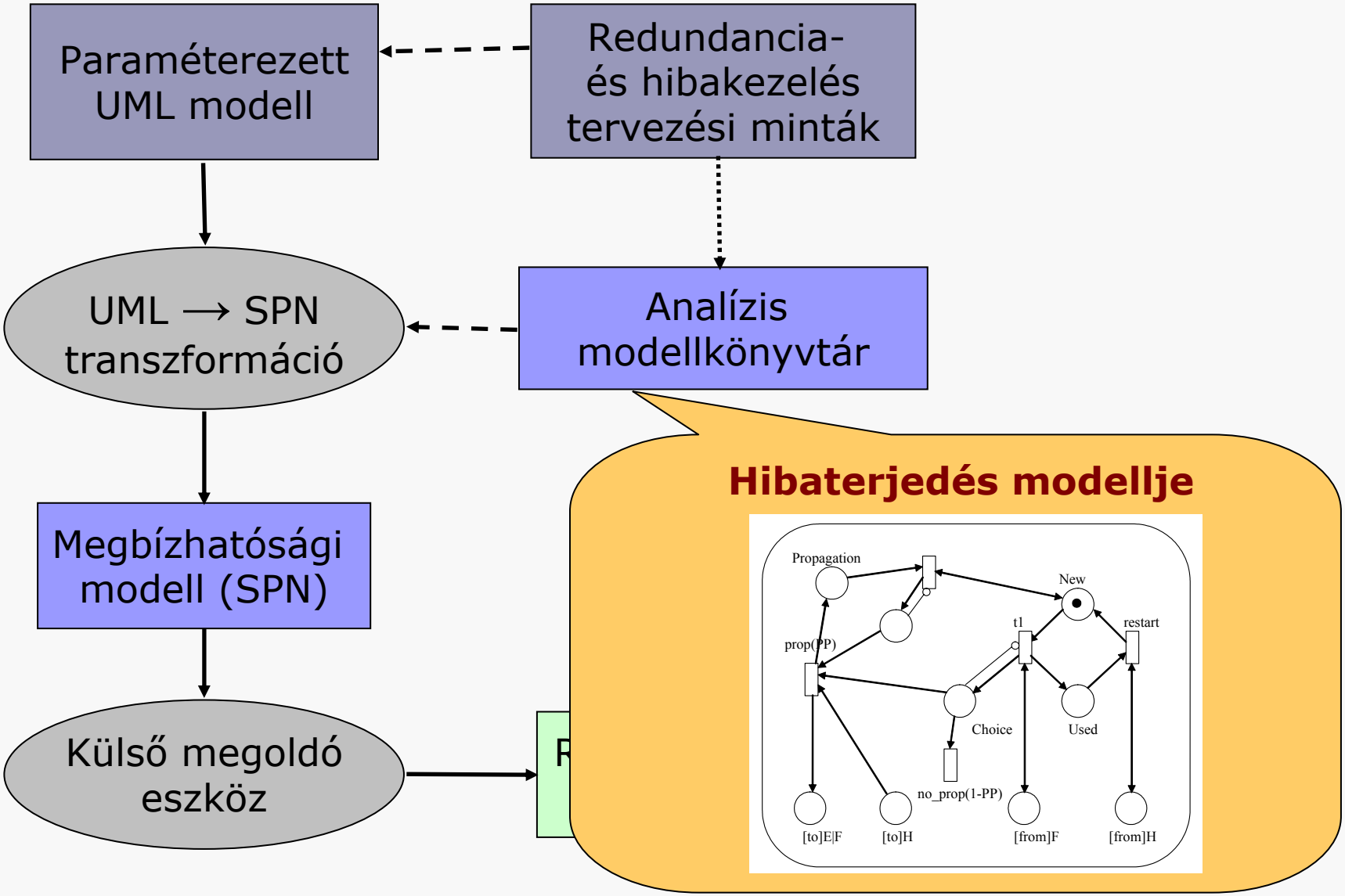
Megbízhatósági modell



Eszközfejlesztés



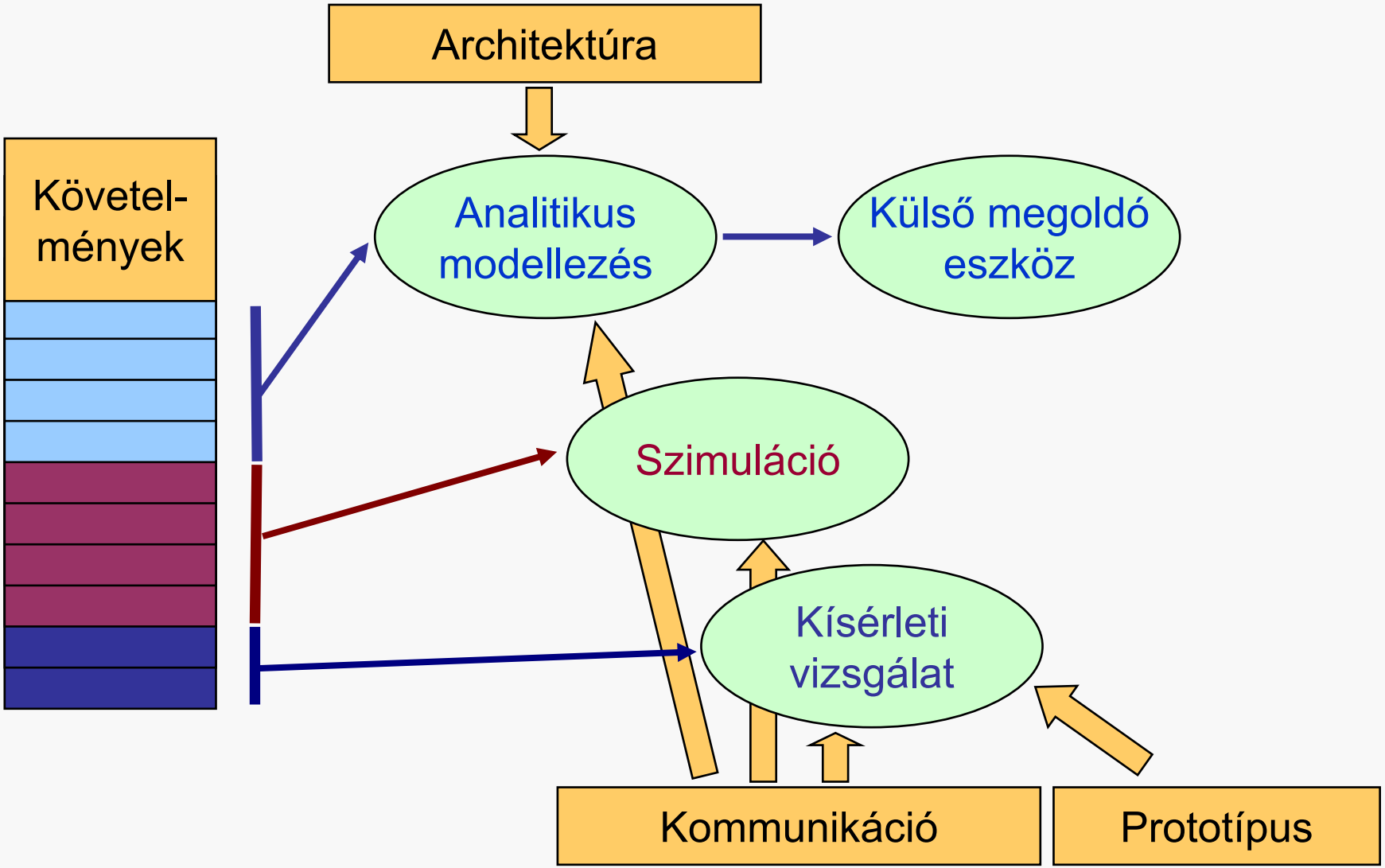
Eszközfejlesztés



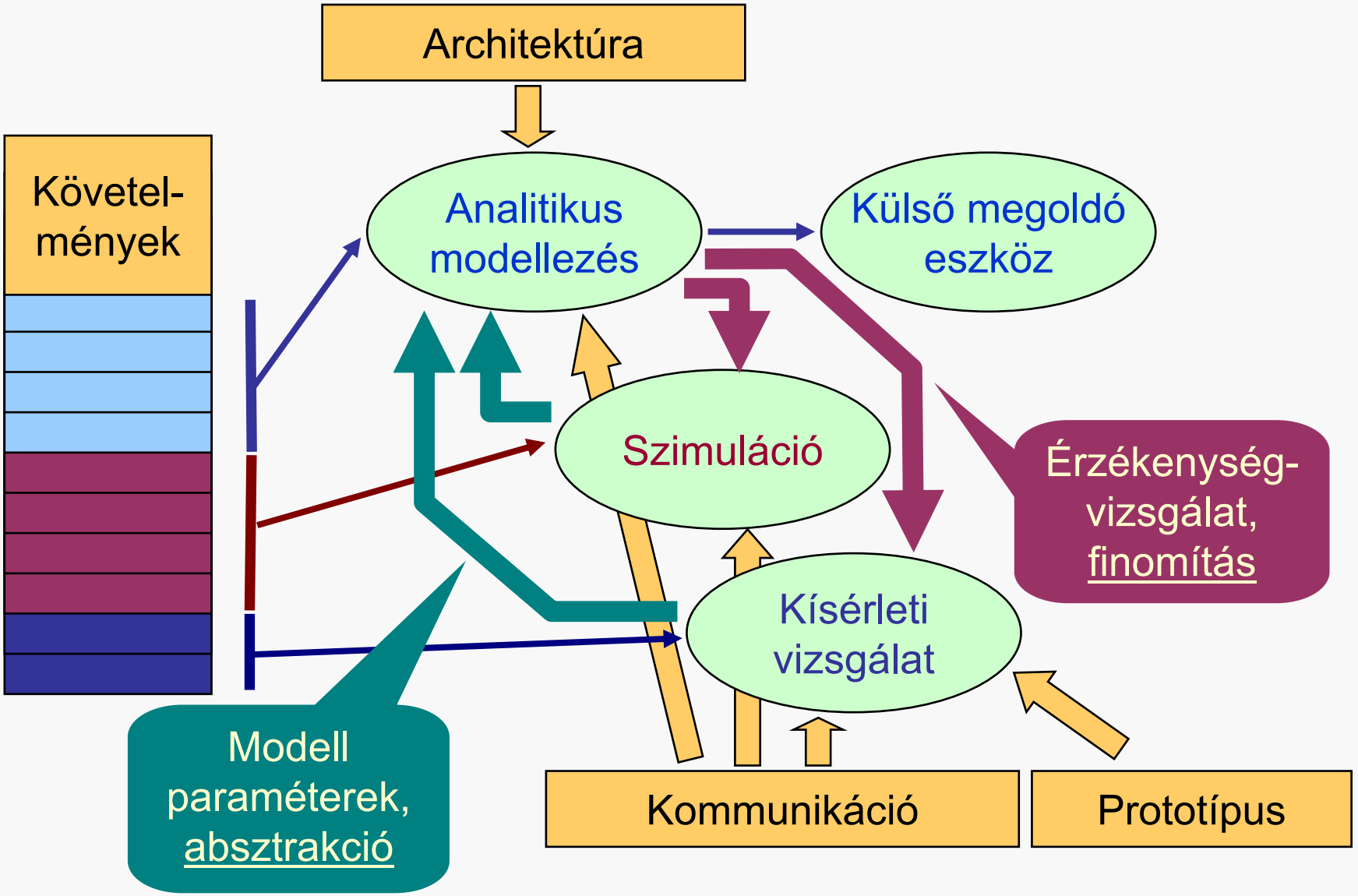
Aktuális projekt: SAFEDMI

- Biztonságkritikus mozdonyvezetői kezelőfelület fejlesztése, analízise és tesztelése
  - Bemenet: Rendszerarchitektúra UML modellje (komponens paraméterekkel kiegészítve)
  - Hierarchikus megbízhatósági modellezés
    - Üzemmódok, taszkok szerinti rész-modellek
    - Rész-modell megoldása magasabb szintű modell paramétere
  - Számítás: Rendszer megbízhatóság (veszélyes hiba gyakoriság megfelel-e a SIL-nek)

Aktuális projekt: SAFEDMI



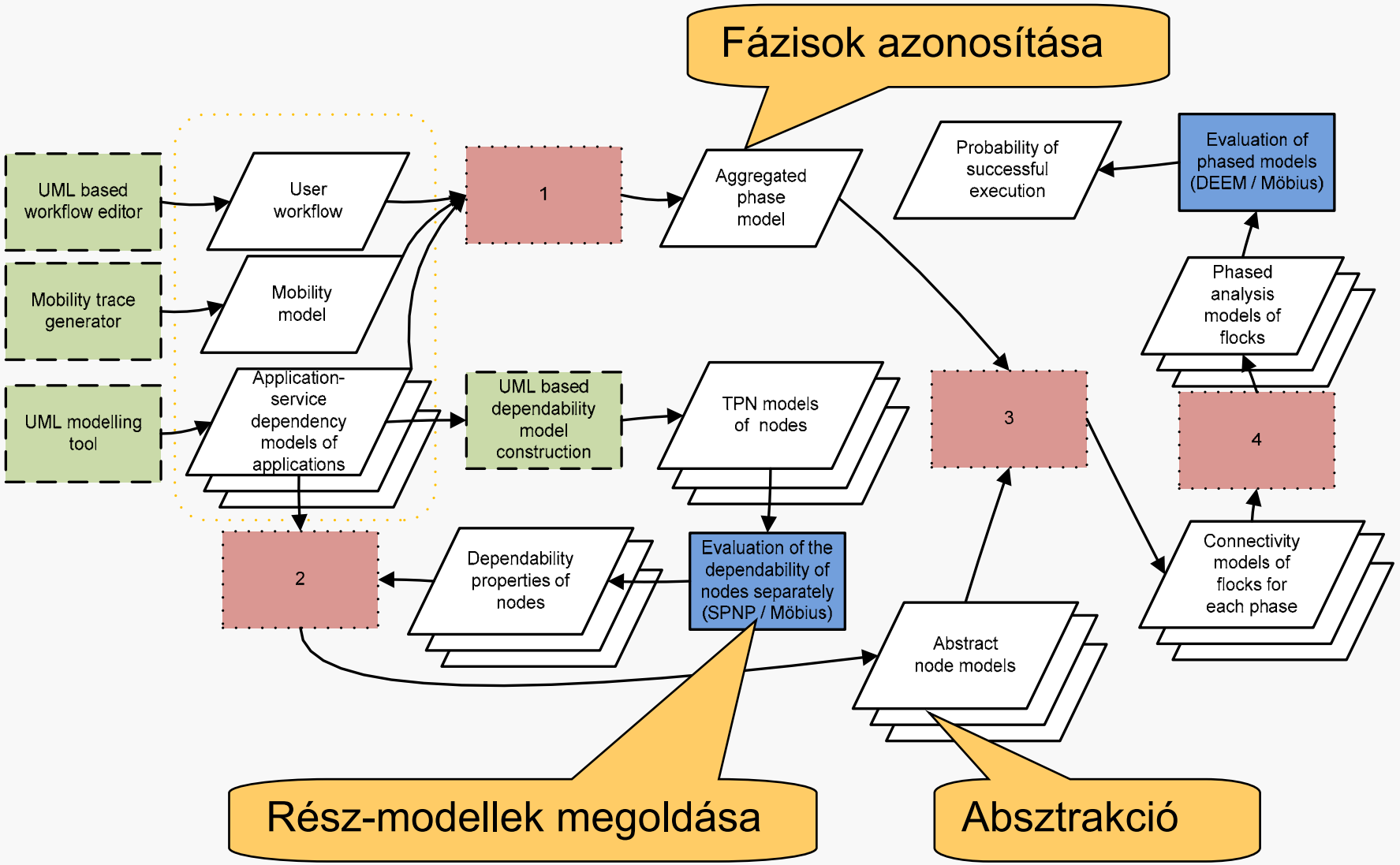
Aktuális projekt: SAFEDMI



Aktuális projekt: HIDENETS

- IP alapú hálózati szolgáltatások mobil ad-hoc környezetben (mozgó autók között)
  - Bemenetek:
    - Felhasználói aktivitások modellje (workflow)
    - Rendszerarchitektúra UML modellje
    - Mobil ad-hoc környezet modellje (hálózati topológia)
  - Időbeli fázisok modellezése
    - Mobil környezet, aktivitás változása, hibák bekövetkezése
  - Számítás: Felhasználói munkafolyamat hibamentes végrehajtásának valószínűsége

Aktuális projekt: HIDENETS



Összefoglalás: A megbízhatósági analízis technikái

- Tervezési fázisban: Becsült (nem pontos) paraméterek alapján végzett modell alapú analízis
  - Architektúra változatok összehasonlítása
  - Szűk keresztmetszetek keresése, érzékenységvizsgálat
- Prototípus fázisban: Hibainjektálás
  - Hibakezelési módszerek tesztelése
  - Paraméterek származtatása
- Működési fázisban: Paraméterek mérése, finomított modell alapú analízis
  - Tervezési fázis feltételezéseinek, döntéseinek igazolása
  - Előrejelzés, karbantartás optimalizálás, hatásvizsgálat