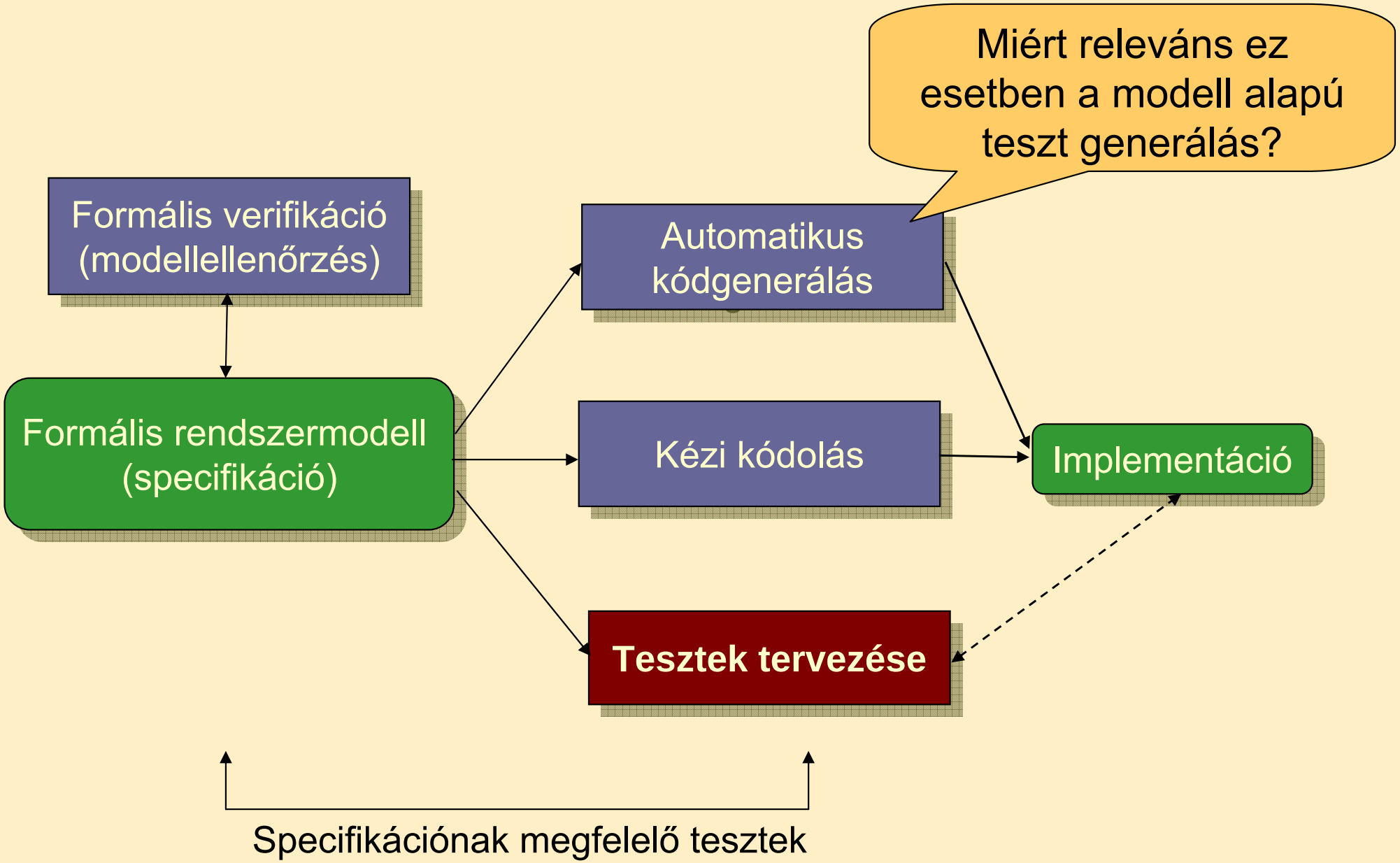
A modellellenőrzés érdekes alkalmazása: Tesztgenerálás modellellenőrzővel

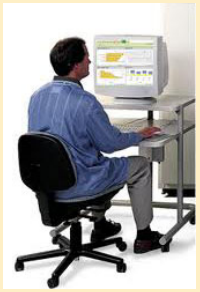
Majzik István és Micskei Zoltán
BME Méréstechnika és Információs Rendszerek Tanszék

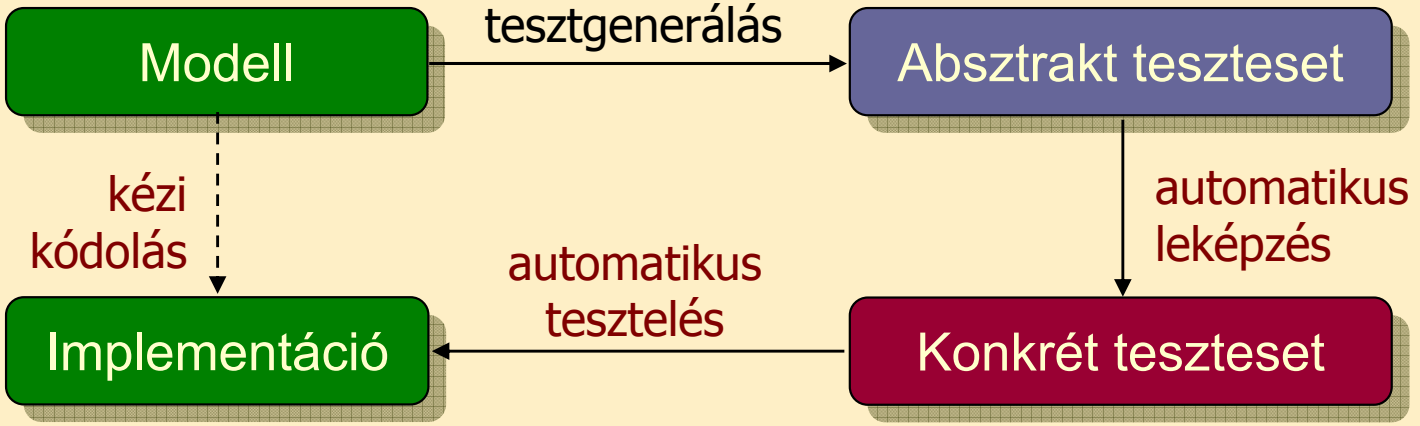
Modell alapú fejlesztési folyamat (részlet)



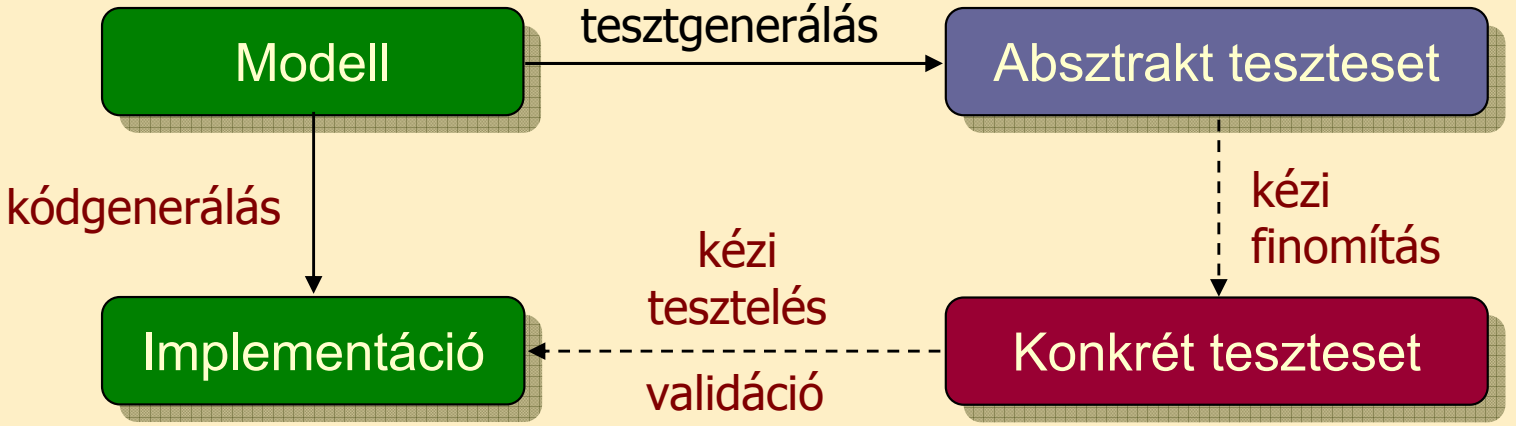
Használati esetek

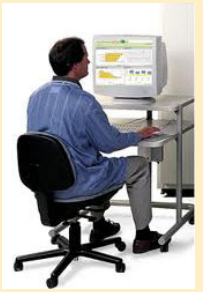
- Kézi kódolás esetén: Konformancia ellenőrzés





- Automatikus kódgenerálás esetén: Validáció



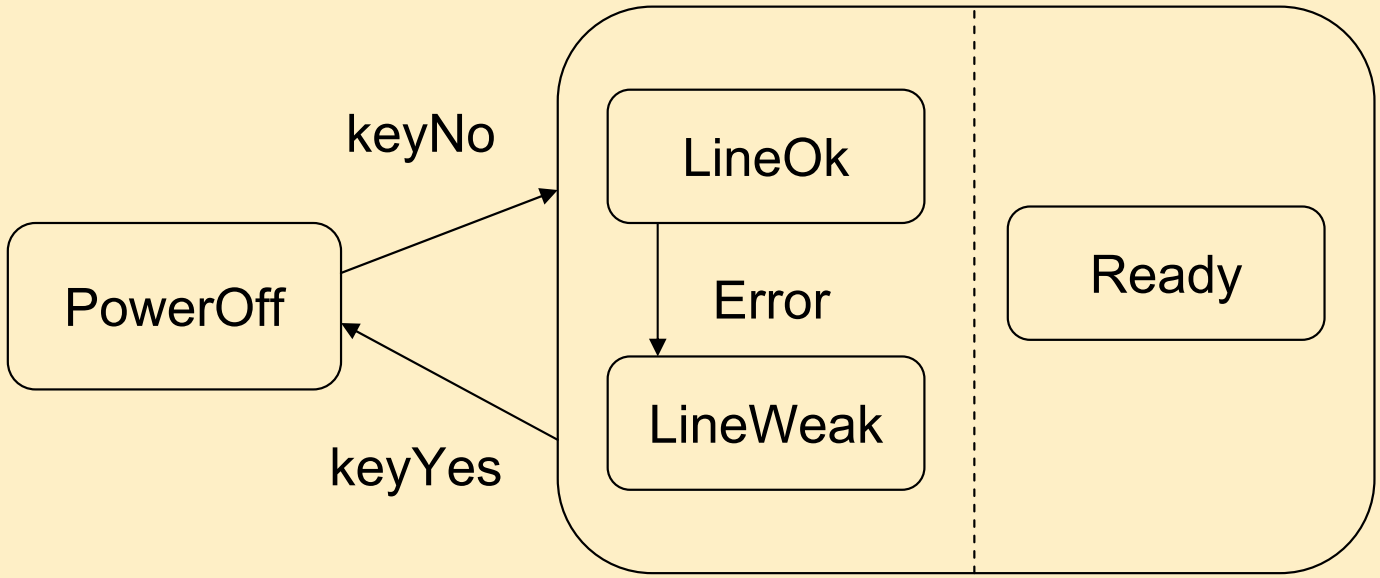


Előfeltételek

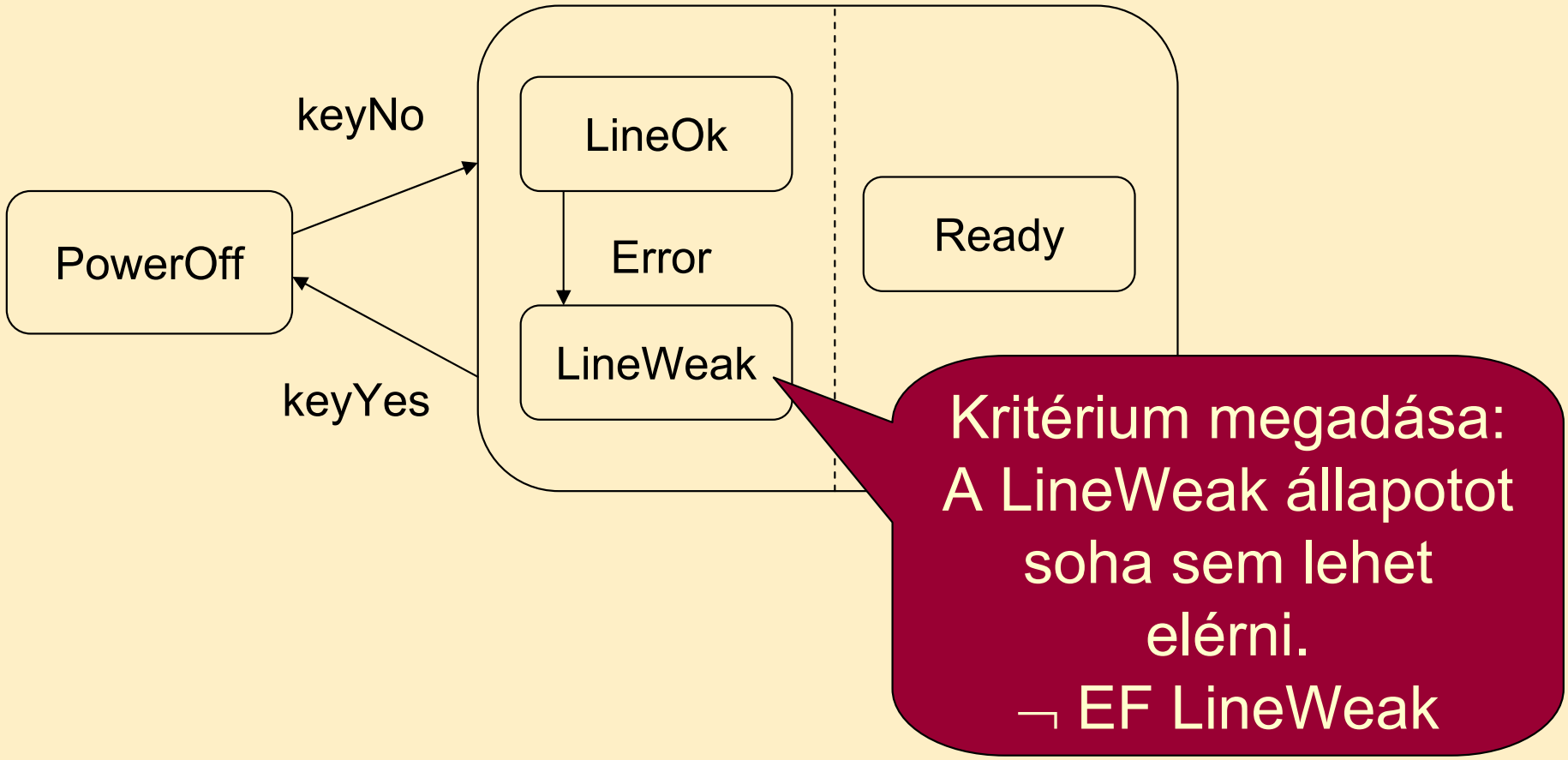
- Állapot alapú, eseményvezérelt működés
  - KS, LTS, KTS az alapszintű formalizmusok
- Fedési kritériumok szerinti tesztelés
  - Állapotfedés: A tesztekkel járjunk be minden állapotot
  - Átmenetfedés: A tesztekkel járjunk be minden átmenetet
- Alapötlet:
  - Egy teszt: Egy megfelelő állapottér bejárási szekvencia
  - Cél: A modellellenőrző járja be az állapotteret!
  - Irányítsuk úgy, hogy a modellellenőrző által generált ellenpélda legyen a teszteset
- Teszt elfogadhatósági kritérium
  - Modell mint referencia alapján származtatható
  - Kézi implementáció helyességének ellenőrzése

Hogyan használható a modell ellenőrző?

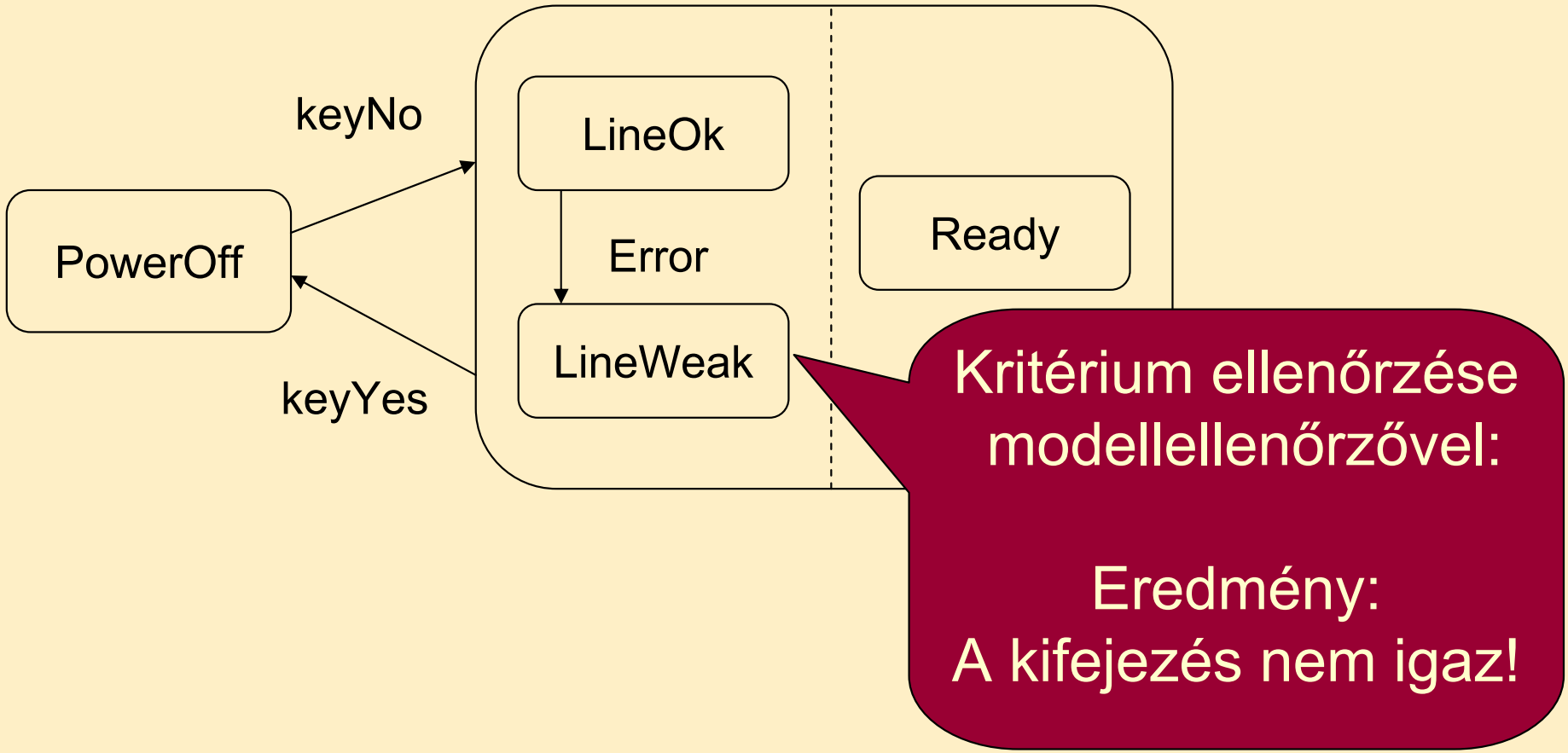
- Állapot fedettség: LineWeak állapotra



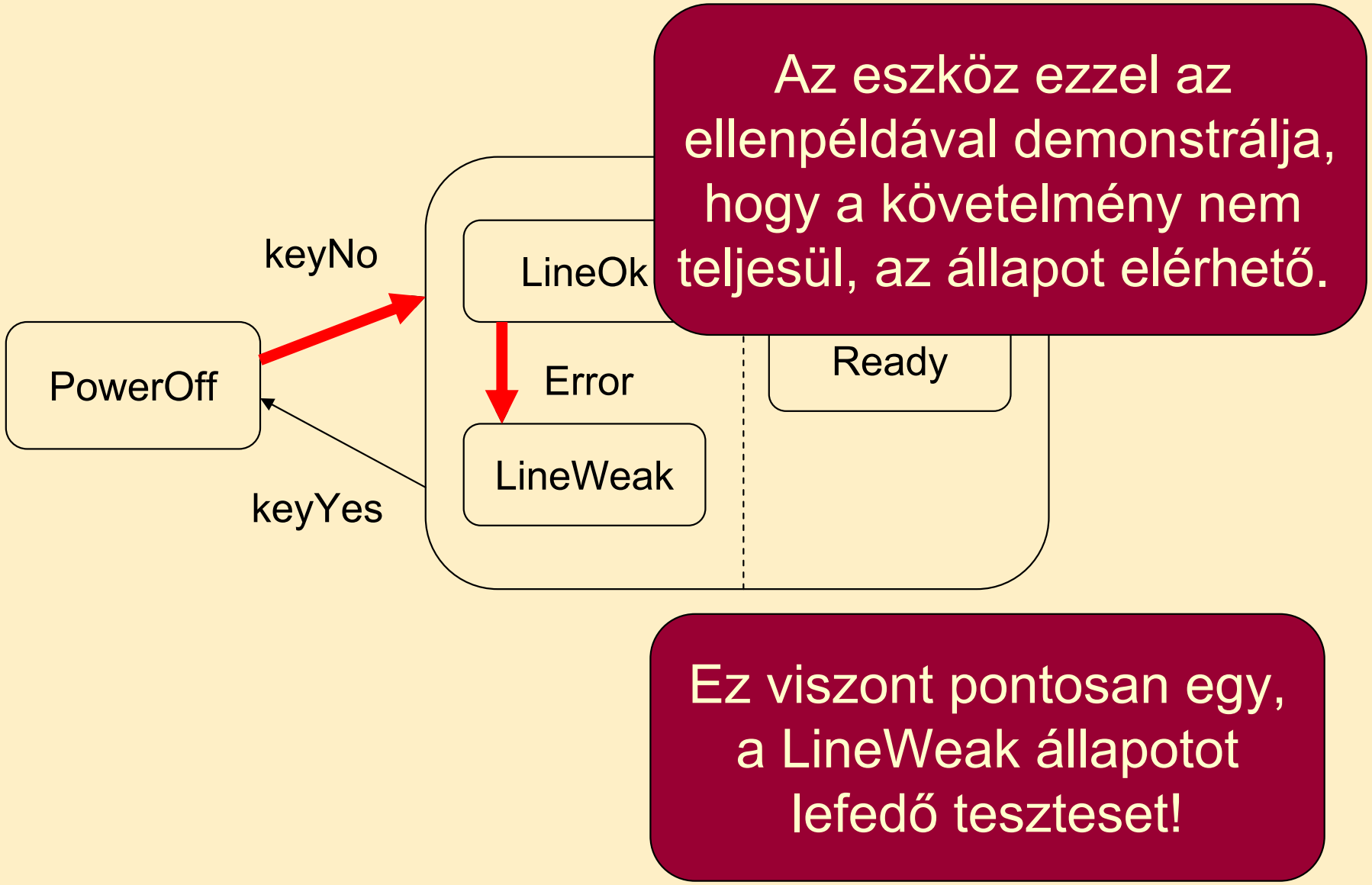
Hogyan használható a modell ellenőrző?



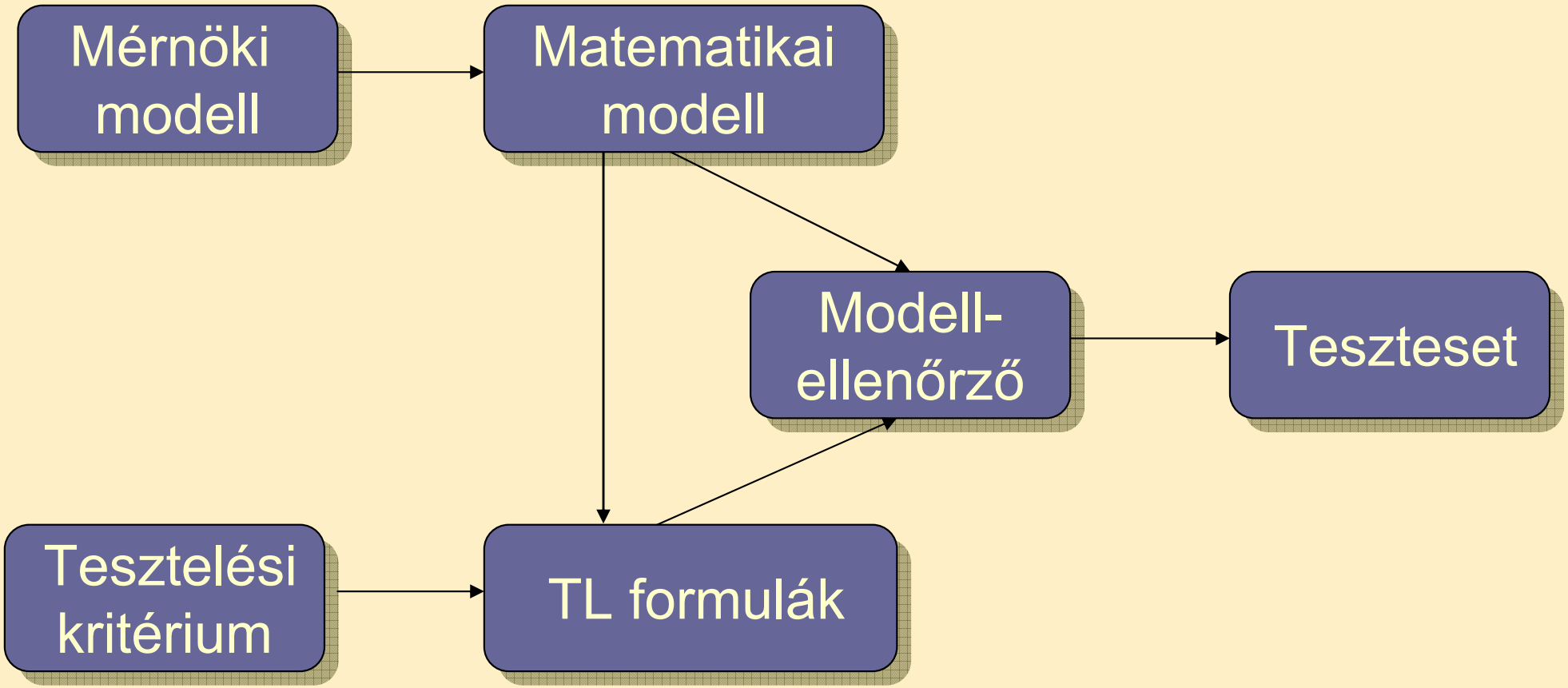
Hogyan használható a modell ellenőrző?



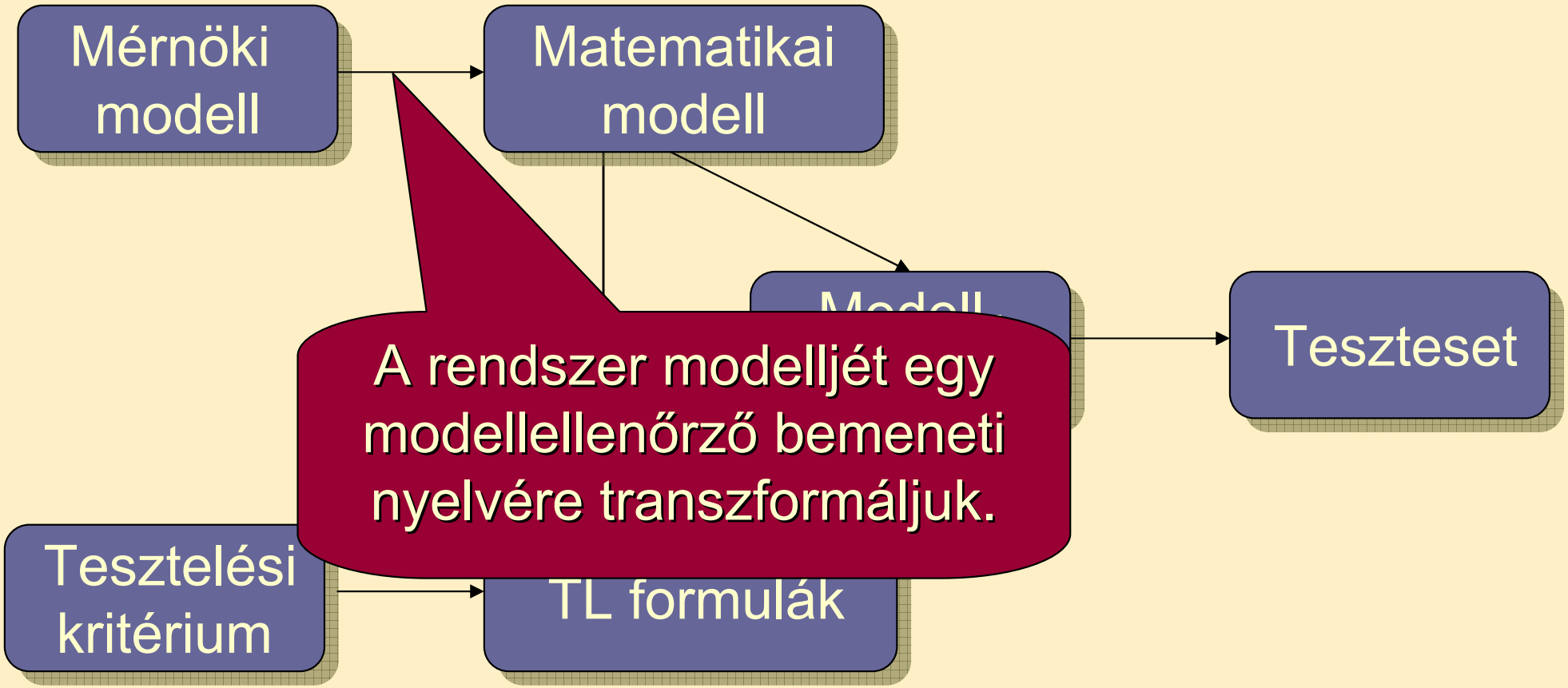
Hogyan használható a modell ellenőrző?



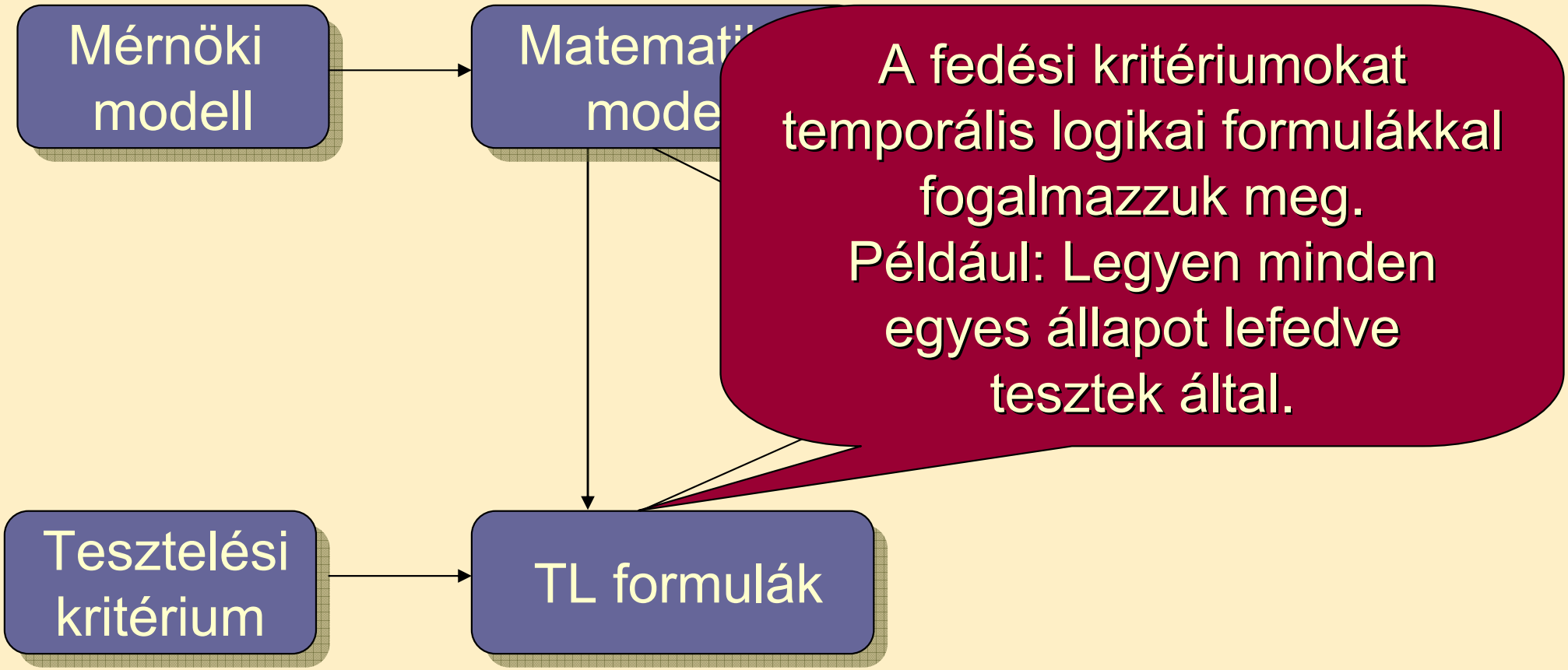
Automatikus tesztgenerálás



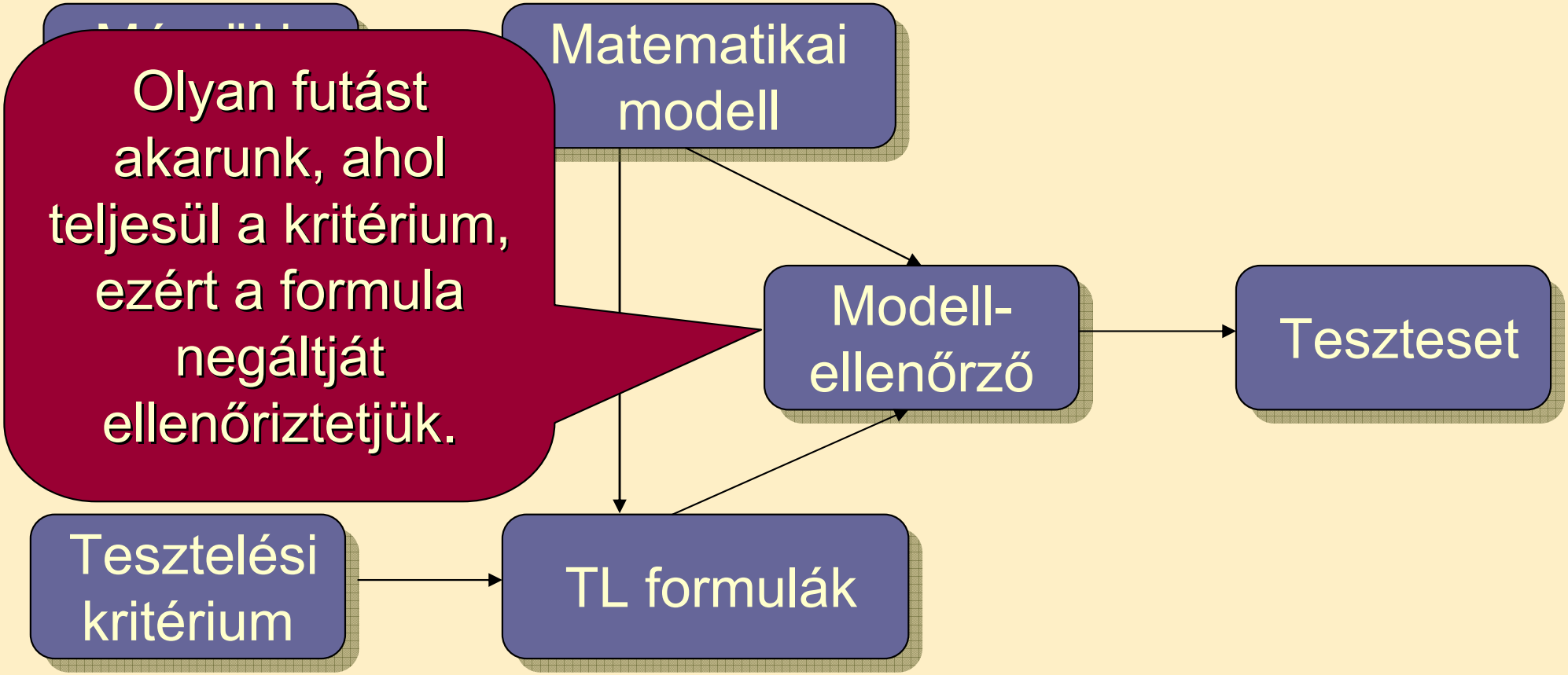
A működés menete – I.



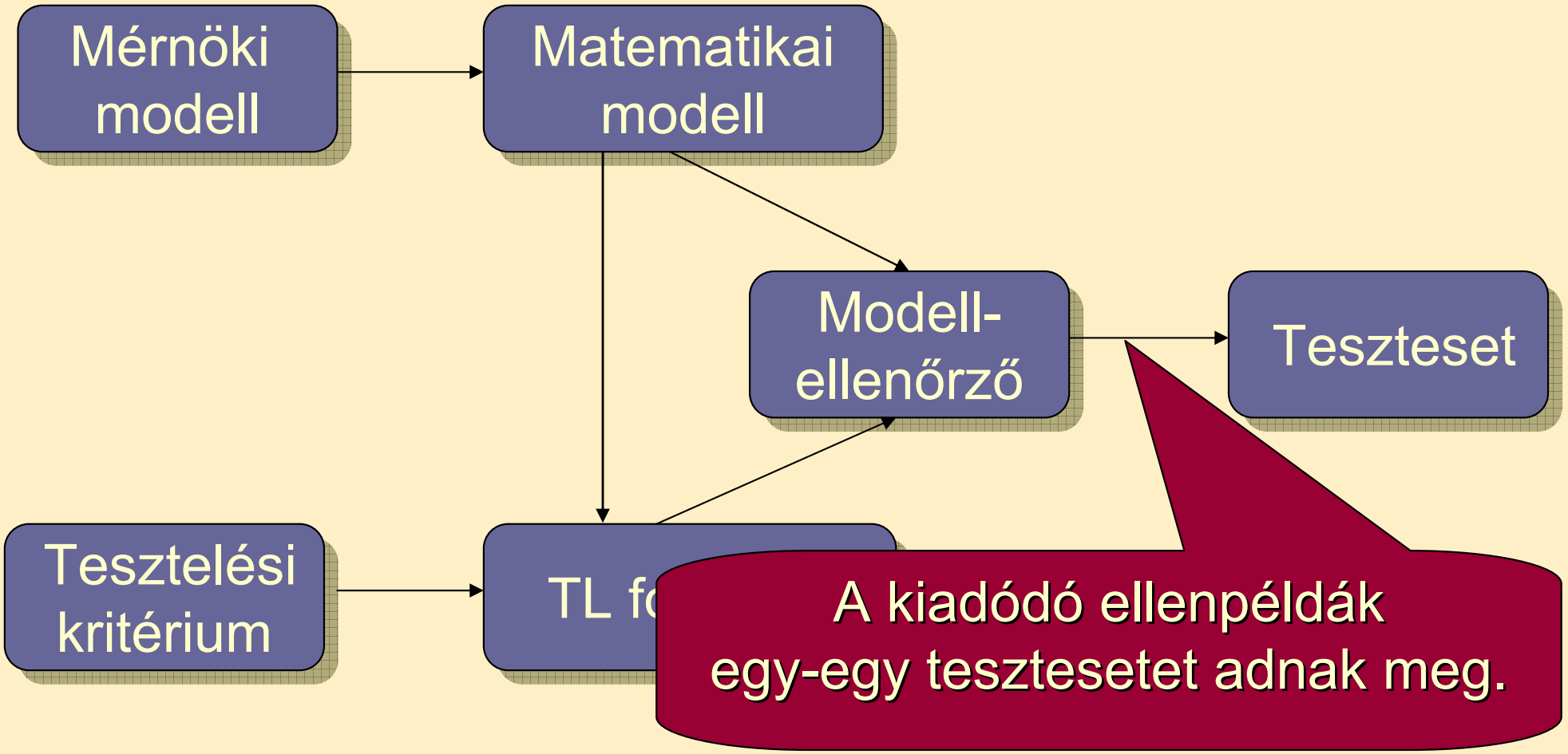
A működés menete – II.



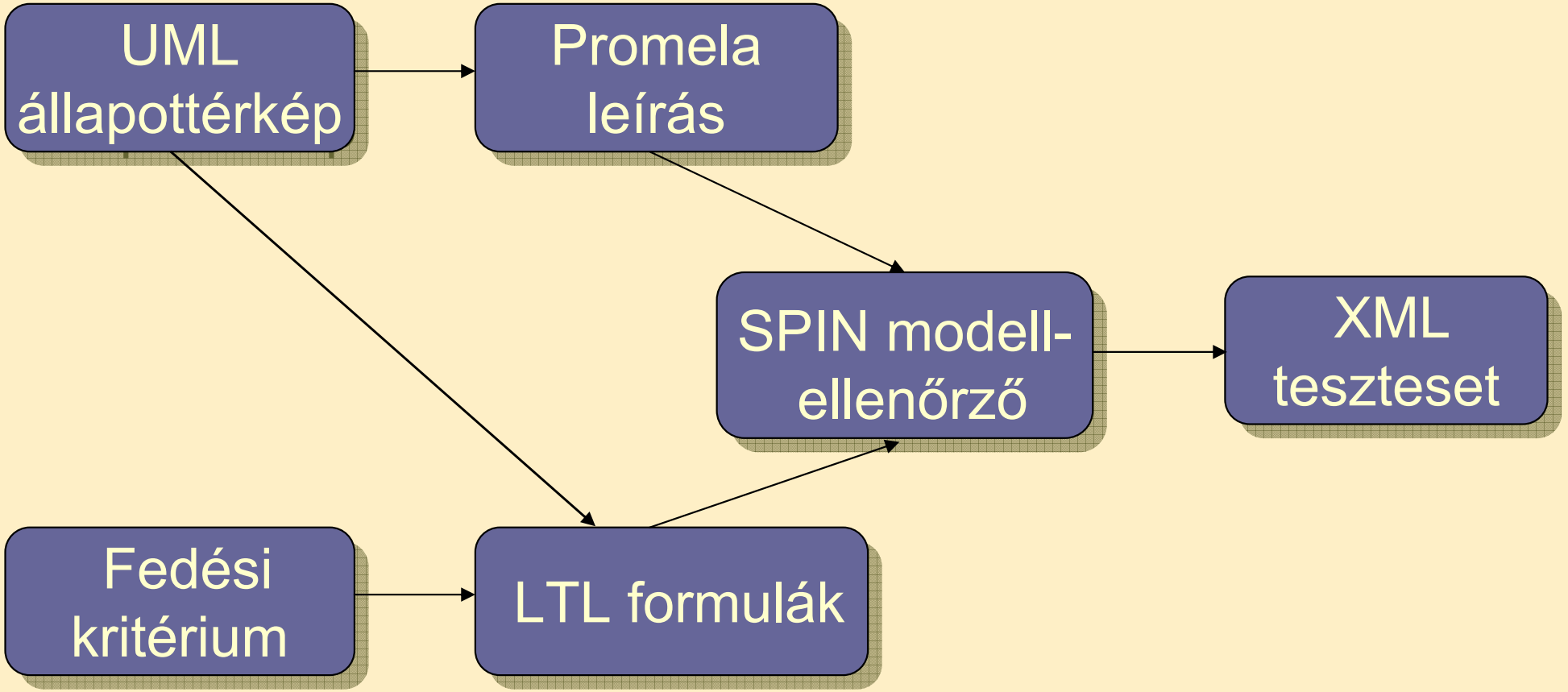
A működés menete – III.



A működés menete – IV.



Egy megvalósítás



Vezérlés alapú fedettségi kritériumok

Minden állapot és átmenet egyértelműen azonosítva:
L(s) egyedi címke illetve (a) akció

- Állapotfedés: KS vagy KTS modellen
  {¬EF (L(s)) | minden s állapotra}
  {¬EF (L(s) ∧ EF exit) | minden s állapotra}

exit egy stabil állapot címkéje
a következő teszt indításához

- Átmenet fedés: LTS vagy KTS modellen
  {¬EF (a) | minden t átmenetre, ahol (s,a,s')∈→}

Korlátozások

- Modell ellenőrző:
  - Csak egy ellenpéldát generál
  - Célja a hatékony állapottér bejárás:
    Nem feltétlenül a legrövidebb teszt esetet kapjuk!
  - Sok modell ellenőrző konfigurálható:
    - Szélességi bejárás kérhető
    - Mélységi bejáráshoz mélységkorlát megadható
    - Rövidebb ellenpélda iteratívan kereshető
  - Legrövidebb/legkisebb tesztkészlet kiválasztása:
    NP-teljes probléma!
- Absztrakt és konkrét teszt esetek:
  - Ellenpélda: Csak a bejárás adott (absztrakt teszt eset)
  - Le kell képezni bemeneti-kimeneti szekvenciára
- Nemdeterminisztikus modellek esetén nehézségek

Tesztgenerálási eredmények (állapotfedés)

Options (compile or run-time)	Time required for test generation	Length of the test sequences	Longest test sequence
-i	22m 32.46s	17	3
-dBFS	11m 48.83s	17	3
-i -m1000	4m 47.23s	17	3
-I	2m 48.78s	25	6
default	2m 04.86s	385	94
-I -m1000	1m 46.64s	22	4
-m1000+	1m 25.48s	97	16
-m200 -w24	46.7s	17	3

Paraméterek:

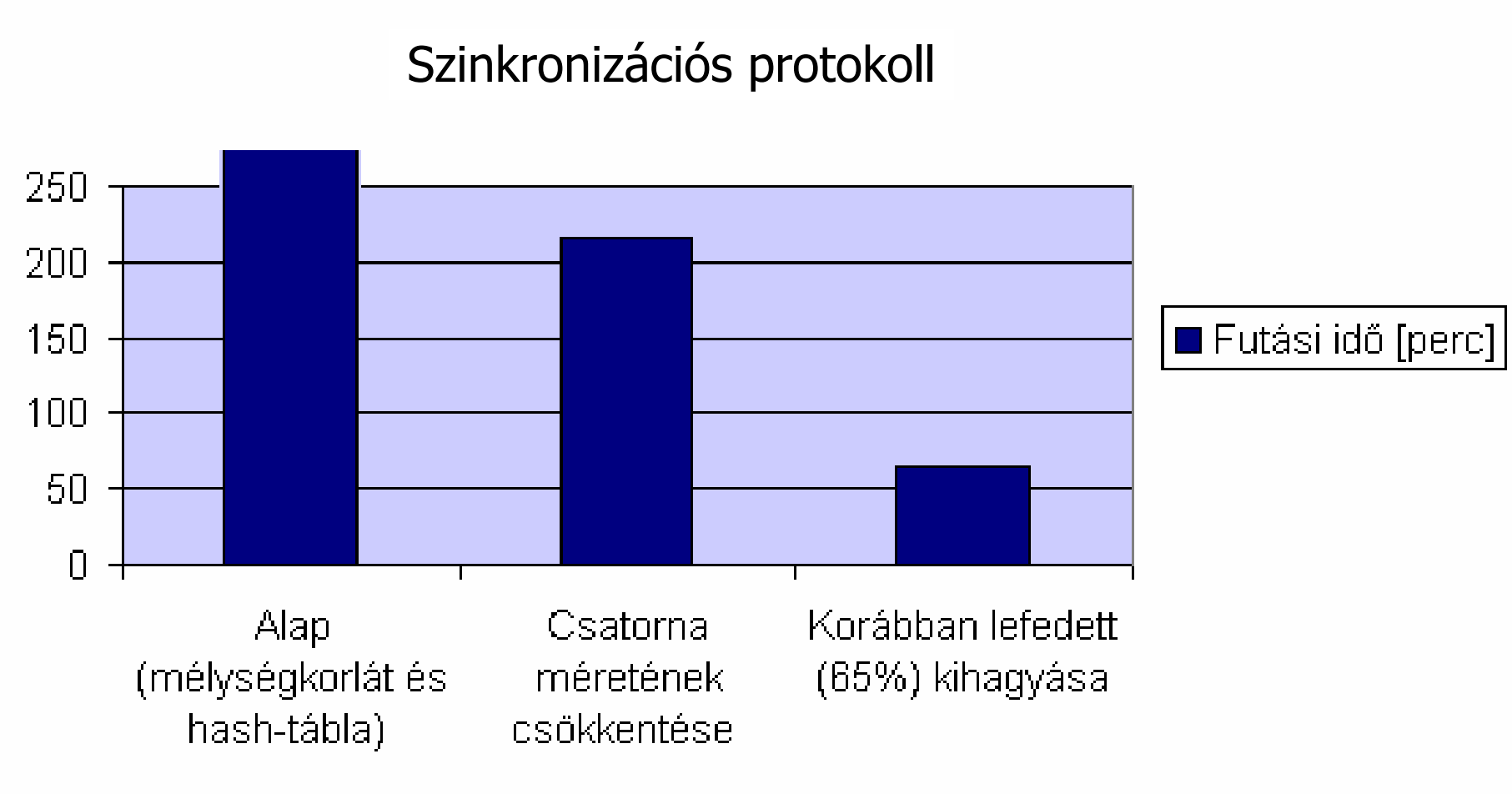
- -i iteratív, -I közelítő iteratív
- -dBFS: szélességi keresés
- -m mélységi keresés korlátja
- -w hash tábla korlátja

- Mobiltelefon vezérlését leíró modell
- 10 állapot, 21 átmenet

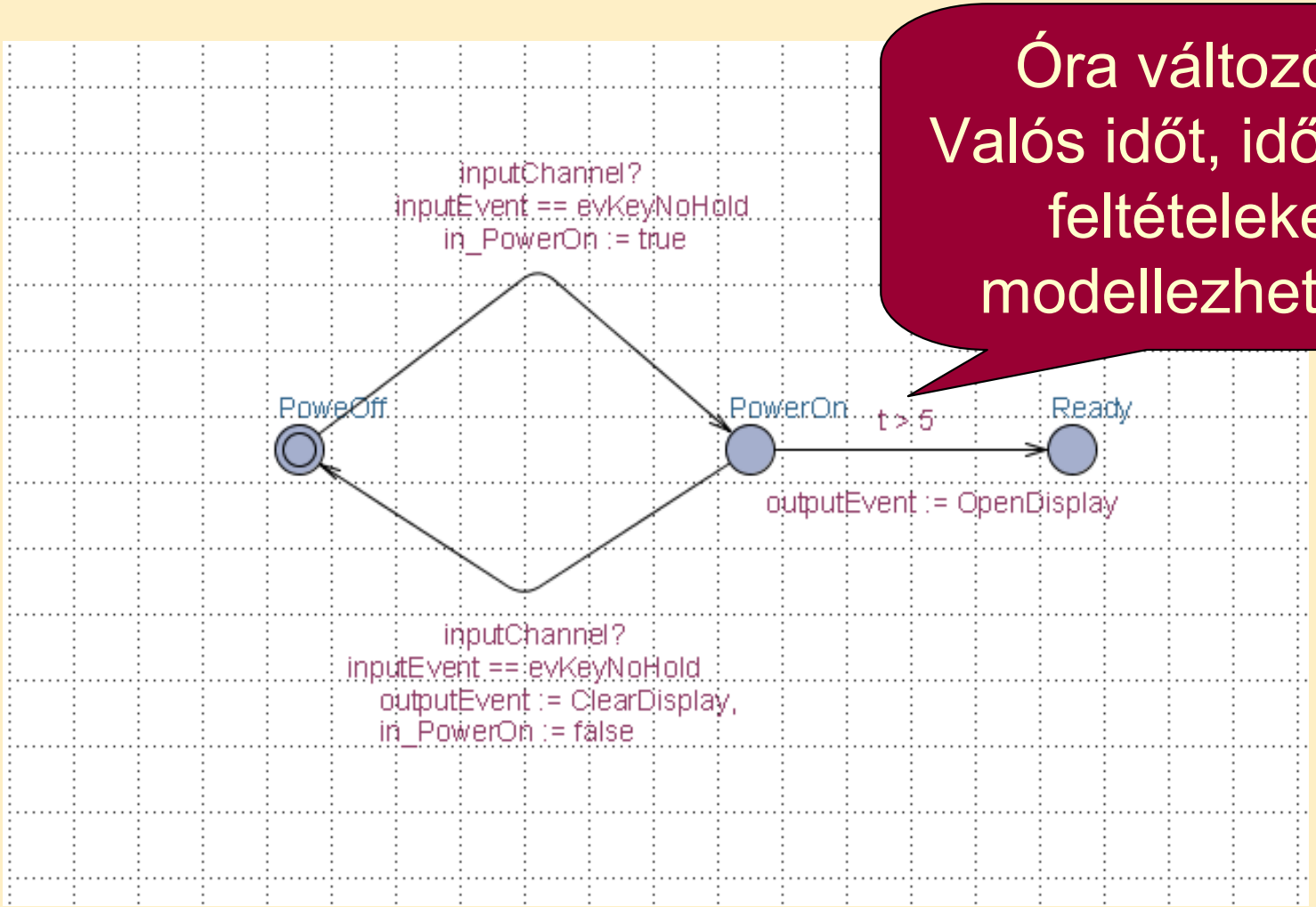
Szinkronizációs protokoll példa

- Bitek szinkronizálása egy elosztott rendszerben
  - 5 objektum, 31 állapot, 174 átmenet
  - 2e+08 bejárandó állapot
- Más technikák is kellenek:
  - Állapottér tárolás erősen tömörítve (bitstate hashing)
  - Szűkítések a modellben: kommunikáció megkötése (csatorna méret csökkentés)
  - Korábban lefedett kritériumok kihagyása
- További heurisztikák alkalmazása:
  - Mélyen fekvő állapotok előbb
  - Állapottér levágása

Teljes állapotfedésű tesztek



Kiterjesztés valósidejű rendszerekre



- Időzített automaták használata
- Modellellenőrző: UPPAAL

Generált tesztek

```
State:
( input.sending mobile.PowerOn mobile1.LineOK
  mobile2.CallWait )
t=0 inputEvent=28 outp                          #depth=5

Delay: 6

State:
( input.sending mobile.PowerOn mobile1.LineOK
  mobile2.CallWait )
t=6 inputEvent=28 outputEvent=14 in_PowerOn=1 #depth=5

Transitions:
input.sending->input.sendInput { 1, inputChannel!, 1 }
mobile2.CallWait->mobile2.VoiceMail { inputEvent ==
  evKeyYes && t > 5 && in_PowerOn, inputChannel?, 1 }
```

A teszt időzítési viszonyok is szerepelnek a generált tesztesetben