A szolgáltatásbiztonság alapfogalmai

Majzik István
majzik@mit.bme.hu

http://www.mit.bme.hu/oktatas/targyak/vimim146/

Tartalomjegyzék

Motiváció: Hibamentes működés

- Szolgáltatási szint szerződések (SLA):
  - Ügyfél által elvárt jellemzők (pl. rendelkezésre állás)
  - Telekom szolgáltatások szerver rendszerei („carrier grade”):
    „Öt kilences”: 99,999% (5 perc/év kiesés)
- Biztonságkritikus rendszerek:
  - Szabvány előírások a hibák gyakoriságára
  - Biztonságintegritási szintek (Safety Integrity Level)

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

SIL	Biztonságkritikus funkció hibája / óra
1	$10^{-6} \leq THR < 10^{-5}$
2	$10^{-7} \leq THR < 10^{-6}$
3	$10^{-8} \leq THR < 10^{-7}$
4	$10^{-9} \leq THR < 10^{-8}$

Hiba nélküli működés ~ 11.000 év??

Elkerülhetetlen: Hibahatások

Fejlesztési folyamat → **Működő termék**

- Tervezési hibák
- Implementációs hibák

- Hardver hibák
- Konfigurációs hibák
- Kezelői hibák

Elkerülhetetlen: Hibahatások

Fejlesztési folyamat → **Működő termék**

- Tervezési hibák
- Implementációs hibák

- Hardver hibák
- Konfigurációs hibák
- Kezelői hibák

Fejlesztési folyamat jellemzői:
- Jobb minőségbiztosítás, jobb módszertanok
- De növekvő bonyolultság, nehezebb ellenőrzés

Szokásos becsült értékek 1000 kódsorra:
- Jó kézi fejlesztés és tesztelés: <10 hiba marad
- Automatizált fejlesztés: ~1-2 hiba marad
- Formális módszerek használata: <1 hiba marad

Elkerülhetetlen: Hibahatások

Fejlesztési folyamat → **Működő termék**

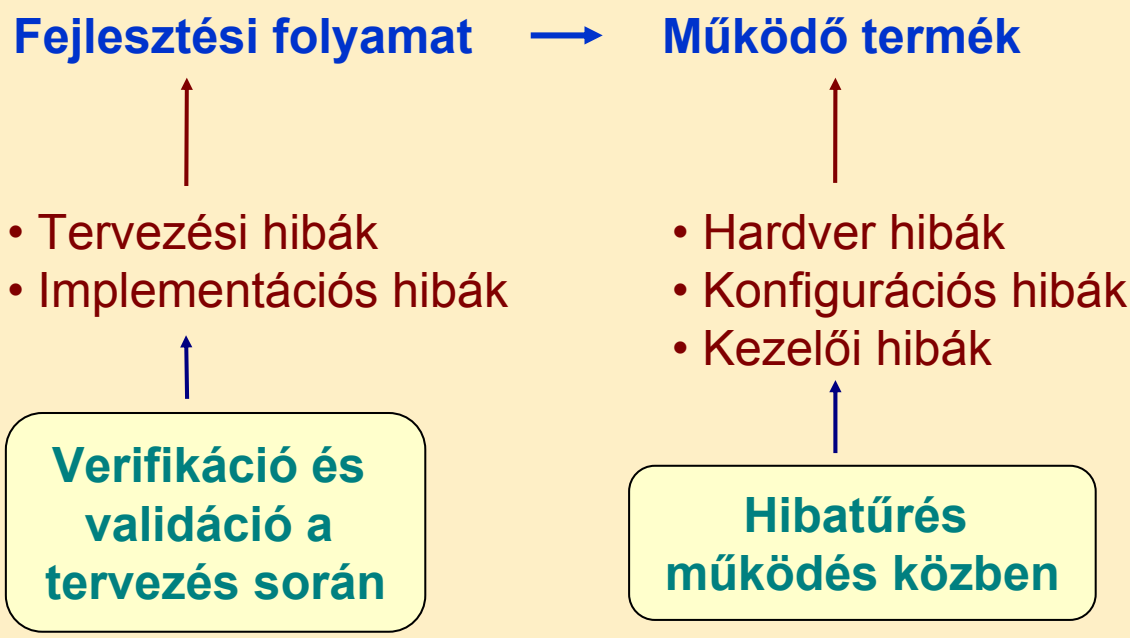
- Tervezési hibák
- Implementációs hibák

- Hardver hibák
- Konfigurációs hibák
- Kezelői hibák

Technológia korlátai:
- Jobb paraméterek, jobb anyagok
- De növekvő bonyolultság (érzékenység)

Szokásos becsült értékek:
- CPU: 10^{-5}…10^{-6} hiba/óra
- RAM: 10^{-4}…10^{-5} hiba/óra
- LCD: ~ 2…3 év élettartam

Elkerülhetetlen: Hibahatások



A hibamentesség jellemzése

- Felhasználó: Szolgáltatás jellemzői érdeklik
  - Szolgáltatásminőség:
    - használhatóság, rendelkezésre állás, javíthatóság,...
  - Termékminőség: ezt nyújtja a tervező
    - előállítási folyamat (ISO 9000)
- Szolgáltatásbiztonság (dependability):
  Milyen biztonsággal képes ellátni feladatait a rendszer?

  Képesség: igazoltan bízni lehet a szolgáltatásban
  - *igazoltan*: elemzésen, méréseken alapul
  - *bizalom*: szolgáltatás az igényeket kielégíti

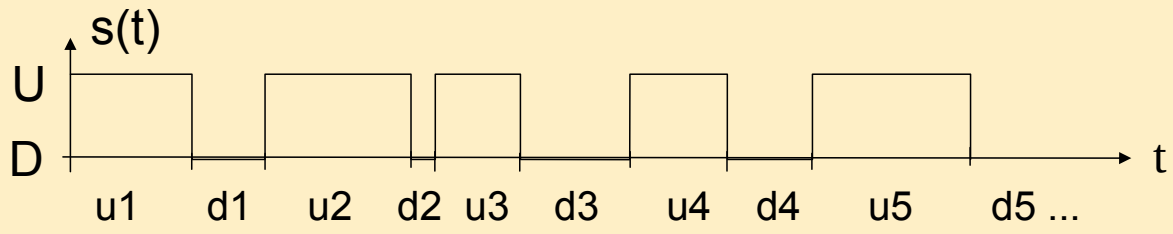
  Összetett fogalom

Szolgáltatásbiztonság jellemzői

- Alapjellemzők:
  - Megbízhatóság: folyamatos hibamentes szolgáltatás
  - Rendelkezésre állás: (javítva) használatra kész szolg.
  - Biztonságosság: katasztrofális következmények (baleset, káreset) nélküli szolgáltatás
  - 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

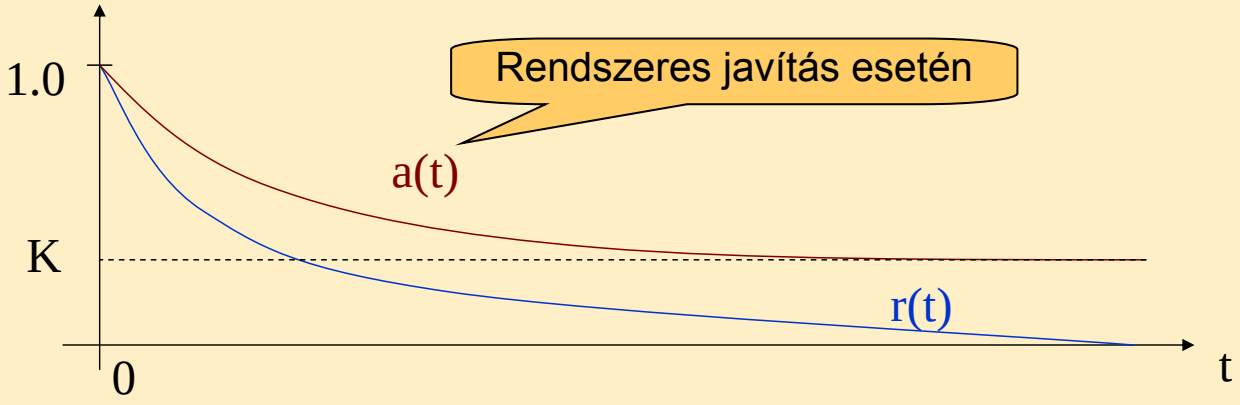
Megbízhatósági mértékek

- Állapot particionálás: s(t) rendszerállapot
  - Hibás (D) - Hibamentes (U) állapotpartíció



- Várható értékek:
  - Első hiba bekövetkezése: MTFF = E{u1} (mean time to first failure)
  - Hibamentes működési idő: MUT = E{ui}
  - Hibás állapot ideje: MDT = E{di}
  - Hibák közötti idő: MTBF = MUT + MDT (mean time between failures)

- Valószínűség időfüggvények
  - Rendelkezésre állás:
    $a(t) = P\{ s(t) \in U \}$ (közben meghibásodhat)
  - Készenlét:
    $K=\lim_{t\to\infty} a(t)$ (aszimptotikus)
  - Megbízhatóság:
    $r(t) = P\{\forall t'<t : s(t')\in U \}$ (nem hibásodhat meg)



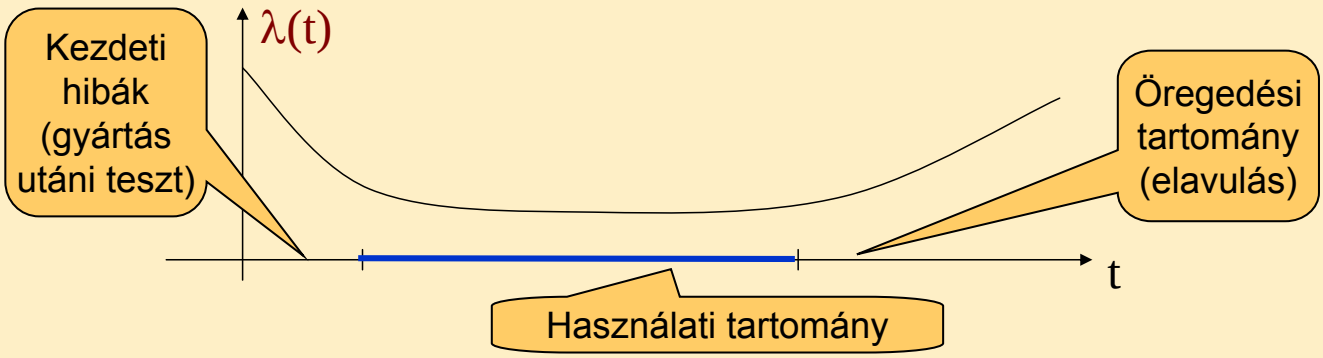
- Meghibásodási tényező (gyakoriság):
  - A rendszer mekkora valószínűséggel fog éppen t-ben elromlani, feltéve, hogy t-ig jól működött

$$\lambda(t) = \frac{P\{s(t+\Delta t) \in D \mid s(t) \in U\}}{\Delta t}, \quad \text{miközben } \Delta t \to 0$$

másként

$$\lambda(t) = -\frac{1}{r(t)}\frac{dr(t)}{dt}, \quad \text{így } r(t) = e^{-\int_0^t \lambda(t)dt}$$

  - Elektronikai alkatrészekre:



Elektronikai komponensek használati tartományában:

- Konstans a meghibásodási tényező:

$$\lambda(t) = \lambda$$

- Megbízhatóság:

$$r(t) = e^{-\lambda t}$$

- Első hiba bekövetkezése:

$$\mathrm{MTFF} = \frac{1}{\lambda}$$

Rendelkezésre állás követelményei

Rendelkezésre állás	Max. kiesés egy év alatt
99%	~ 3,5 nap
99,9%	~ 9 óra
99,99% („4 kilences”)	~ 1 óra
99,999% („5 kilences”)	~ 5 perc
99,9999% („6 kilences”)	~ 32 másodperc
99,99999%	~ 3 másodperc

Munkaállomásokból álló elosztott rendszer:

- 1 szgép: 95% rendelkezésre állással
- 2 szgép: 90%
- 5 szgép: 77%
- 10 szgép: 60%

Tartalomjegyzék

- A szolgáltatásbiztonság fogalma
- A szolgáltatásbiztonságot befolyásoló tényezők
- A szolgáltatásbiztonság eszközei

Befolyásoló tényezők

- Hibajelenség (failure):
  A specifikációnak nem megfelelő szolgáltatás
  - értékbeli / időzítésbeli, katasztrofális / „jóindulatú”
- Hiba (error):
  Hibajelenséghez vezető rendszerállapot
  - lappangó → detektált
- Meghibásodás (fault):
  A hiba feltételezett oka
  - hatás: alvó → aktív
  - fajta: véletlen vagy szándékos, időleges vagy állandósult
  - eredet: fizikai/emberi, belső/külső, tervezési/működési

Tipikus meghibásodások

Fajta	Eredet	Hely	Fázis	Idő	Példa
Véletlen	Fizikai	Belső	Működés	Állandó-sult	Komponens hiba
Véletlen	Fizikai	Külső	Működés	Időleges	Tranziens hiba
Véletlen	Emberi	Belső	Tervezés	Állandó-sult	Tervezési hiba
Véletlen	Emberi	Külső	Működés	Időleges	Kezelési hiba
Szándé-kos	Emberi	Külső	Működés	Állandó-sult	Rongálás
Szándé-kos	Emberi	Külső	Működés	Időleges	Behatolás

Szoftver hibák

- Szoftver hiba: Állandósult, tervezési hiba
- Aktiválás a működési profil függvénye
  - Adott bemeneti tartomány, trajektória aktivál
- Becslési módszerek:
  - Megbízhatóság arányos:
    Tesztelés után bennmaradó hibák számával
  - Bennmaradó hibák száma arányos:
    Időegység alatt detektált hibák számával a tesztelés végén
  - Statisztikai módszerekkel becsülhető,
    meddig kell a tesztelést folytatni
    adott megbízhatóság eléréséhez

Hatáslánc

- Meghibásodás → Hiba → Hibajelenség
  - pl. szoftver:
    - meghibásodás: progr. hiba: csökkentés helyett növel
    - hiba: vezérlés ráfut, változó értéke hibás lesz
    - hibajelenség: számítás végeredménye rossz
  - pl. hardver:
    - meghibásodás: kozmikus sugárzás egy bitet átbillent
    - hiba: hibás memóriacella olvasása
    - hibajelenség: robotkar a falnak ütközik
- Rendszer hierarchiaszint függvénye
  - alsó szintű hibajelenség felsőbb szinten meghibásodás
    - kimenet beragadás egy chip szintjén hibajelenség
    - rendszer szintjén meghibásodás (chip a cserélhető komponens)

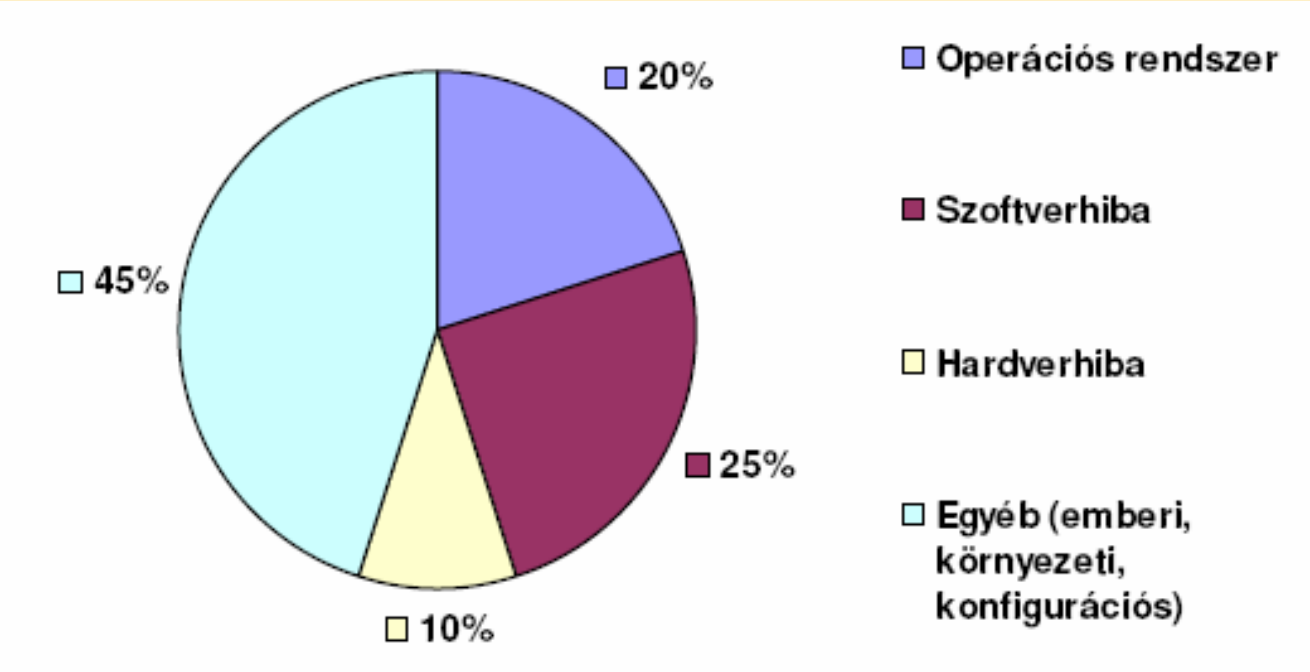
A hatáslánc befolyásolása

- Meghibásodási tényező csökkentése
  - jobb minőségű komponensek
  - szigorúbb fejlesztési folyamat (ellenőrzés, tesztelés)

  A meghibásodás-mentesség nem garantálható (csökkenő chipméretek, bonyolultabb programok)
- Hibajelenség kialakulásának megakadályozása
  - rendszerstruktúra kialakítása: redundancia
- Hibatípusok:
  - Előre figyelembe vehető hibák: optimális kezelés a tervezési folyamat során
  - Előre figyelembe nem vehető hibák: megfelelő rendszerstruktúra kialakítása szükséges

A hibajelenségek okai (egy felmérés)

A hibajelenségek okai munkaállomás szoftverekben:



A hibajelenségek okai (másik felmérés)

A hibajelenségek okai kliens-szerver rendszerekben:

- Hardver hiba: 10%
- Szoftver hiba: 40%
  - Szerver szoftver: 30%
  - Kliens szoftver: 5%
  - Hálózati szoftver: 5%
- Emberi hiba: 15%
- Környezeti hatás: 5%
- Tervezett leállás: 30%

Tartalomjegyzék

- A szolgáltatásbiztonság fogalma
- A szolgáltatásbiztonságot befolyásoló tényezők
- A szolgáltatásbiztonság eszközei

Eszközök

- Hiba megelőzés: Meghibásodás megakadályozása
  - Fizikai hibák: jó minőségű alkatrészek, árnyékolás,...
  - Tervezési hibák: verifikáció, jól meghatározott folyamatok
- Hiba megszüntetés:
  - Prototípus fázis: tesztelés, diagnosztika, javítás
  - Működés közben: monitorozás, javítás
- Hibatűrés: Szolgáltatást nyújtani hiba esetén is
  - Működés közben: hibakezelés, redundancia
- Hiba előrejelzés: Hibák és hatásuk becslése
  - Mérés és „jóslás”, megelőző karbantartás

Hiba megelőzés és megszüntetés

A tervezés során végzett ellenőrzések

- Verifikáció (igazolás): „Jól tervezem-e a rendszert?”
  A rendszer(modell) megfelel-e a specifikációnak
  - formális verifikáció: a rendszermodell és a követelmények matematikai objektumok
  - tesztelés: végrehajtható modell vagy kód
  - szimuláció: rendszer és környezet modellje alapján
- Validáció (érvényesítés): „Jó rendszert terveztem-e?”
  A rendszer megfelel-e az elvárásoknak
  - validációs tesztelés: prototípus, termék alapján
  - szimuláció valós környezetben
  - mérések

Hibatűrés

- Akármilyen jó is az ellenőrzés a tervezés során, a szolgáltatásbiztonság nem garantálható:
  - időleges hardver hibák (ld. zavarérzékenység)
  - fel nem derített (teszteletlen) szoftver hibák
  - figyelembe nem vett komplex interakciók

  → Fel kell készülni a működés közbeni hibákra!
- Hibatűrés: Szolgáltatást nyújtani hiba esetén is
  - működés közbeni autonóm hibakezelés
  - beavatkozás a meghibásodás → hibajelenség láncba
  - rendszertechnikai megoldások (+ megbízható alkatrész)
- Alapfeltétel: Redundancia (tartalékolás)
  - többlet erőforrások a hibás komponensek kiváltására

A redundancia megjelenése

1. Hardver redundancia
   - többlet hardver erőforrások
2. Szoftver redundancia
   - többlet szoftver modulok
3. Információ redundancia
   - többlet információ a hibajavítás érdekében
4. Idő redundancia
   - ismételt végrehajtás, hibakezelés többlet ideje

Együttes megjelenés!

A redundancia típusainak összehasonlítása

Redundancia / tulajdonság	Hideg tartalék (passzív redundancia)	Langyos tartalék (másodlagos funkciók)	Meleg tartalék (aktív redundancia)
Alapelv	Csak hiba esetén aktiválva	Csökkentett terheléssel működik	Ugyanúgy működik, mint az elsődleges
Előnye	Nem hibásodik meg a passzív komponens	Kisebb meghibásodási tényező	Gyorsan átveheti az elsődleges helyét
Hátránya	Lassan veszi át az elsődleges helyét	Közepes sebességű feladat átvétel	Azonos meghibásodási tényező
Példa	Kikapcsolt tartalék számítógép	Naplózó számítógép belép elsődlegesként	Árnyék számítógép

1. Hardver redundancia

- Hardver állandósult hibák esetén
  - Cél az egyszeres hibapont (single point of failure, SPOF) elkerülése
- Megjelenése:
  - Eleve a rendszerben lévő redundáns komponensek
    - Elosztott rendszer, adaptív átkonfigurálás
  - Hibatűréshez betervezett redundancia (tartalékolás)
    - kettőzés
    - TMR: Triple-modular redundancy
    - NMR: N-modular redundancy

2. Szoftver redundancia

Használat:

1. Szoftver tervezési hibák esetén:
   - ismételt végrehajtás nem segít...
   - redundáns modulok: eltérő tervezés szükséges variánsok: azonos specifikáció, de
     - eltérő algoritmus, adatstruktúrák
     - más fejlesztési környezet, programnyelv
     - elszigetelt fejlesztés
2. Időleges (hardver) hibák esetén:
   - ismételt végrehajtás esetén a hiba nem jelentkezik
   - hibahatások kiküszöbölése a fontos

3. Információ redundancia

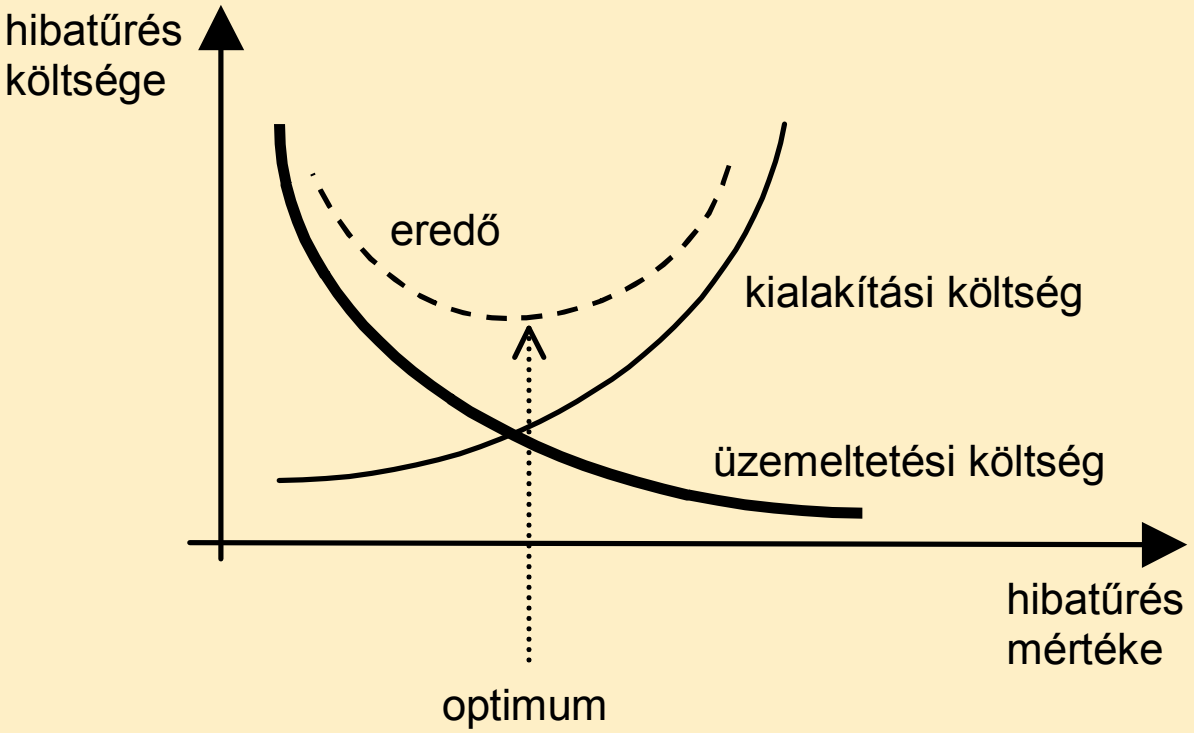
- Hibajavító kódolás
  - memóriák. háttértárak, adatátvitel
  - pl. Hamming-kód, Reed-Solomon kódok

  Korlátozott hibajavító képesség
  - hosszú idejű adatstabilitás rossz lehet ("felgyűlnek" a hibák)
  - háttértárak: "memory scrubbing" folyamatos olvasás és javítva visszaírás
- Többpéldányos (elosztott) adattárolás
  - hozzáférések konzisztenciájának biztosítása
  - egypéldányos sorosíthatóság

4. Idő redundancia

- Tiszta eset: utasítás újrapróbálás (retry)
  - alacsony hardver szinten: processzor utasítás
  - időleges hibák esetén hatásos
- Idő redundancia "velejárója" a többi típusnak
  - Valósidejű rendszerek: tervezési szempont, hogy mennyire garantálható a hibakezelés ideje
    - állandósult hardver hibák: maszkolás, meleg tartalék
    - időleges hardver hibák: előrelépő helyreállítás
    - szoftver tervezési hibák: N-verziós programozás

Költségoptimalizálás



Összefoglalás

- Szolgáltatásbiztonság
  - Jellemzők: Megbízhatóság, rendelkezésre állás, biztonság, bizalmasság, integritás, karbantarthatóság
  - Hatáslánc: Meghibásodás → hiba → hibajelenség
  - Eszközök: Hiba megelőzés, hiba megszüntetés, hibatűrés, hiba előrejelzés
- Hibatűrés
  - Redundancia megjelenése: Szoftver, hardver, idő, információ redundancia
  - Redundancia típusa: Meleg, langyos, hideg tartalék