

Követelménykezelés Specifikáció készítés A specifikáció ellenőrzése

Majzik István

Egyes ábrák: Pap Zsigmond, Polgár Balázs

<http://www.inf.mit.bme.hu/>

1

Tartalomjegyzék

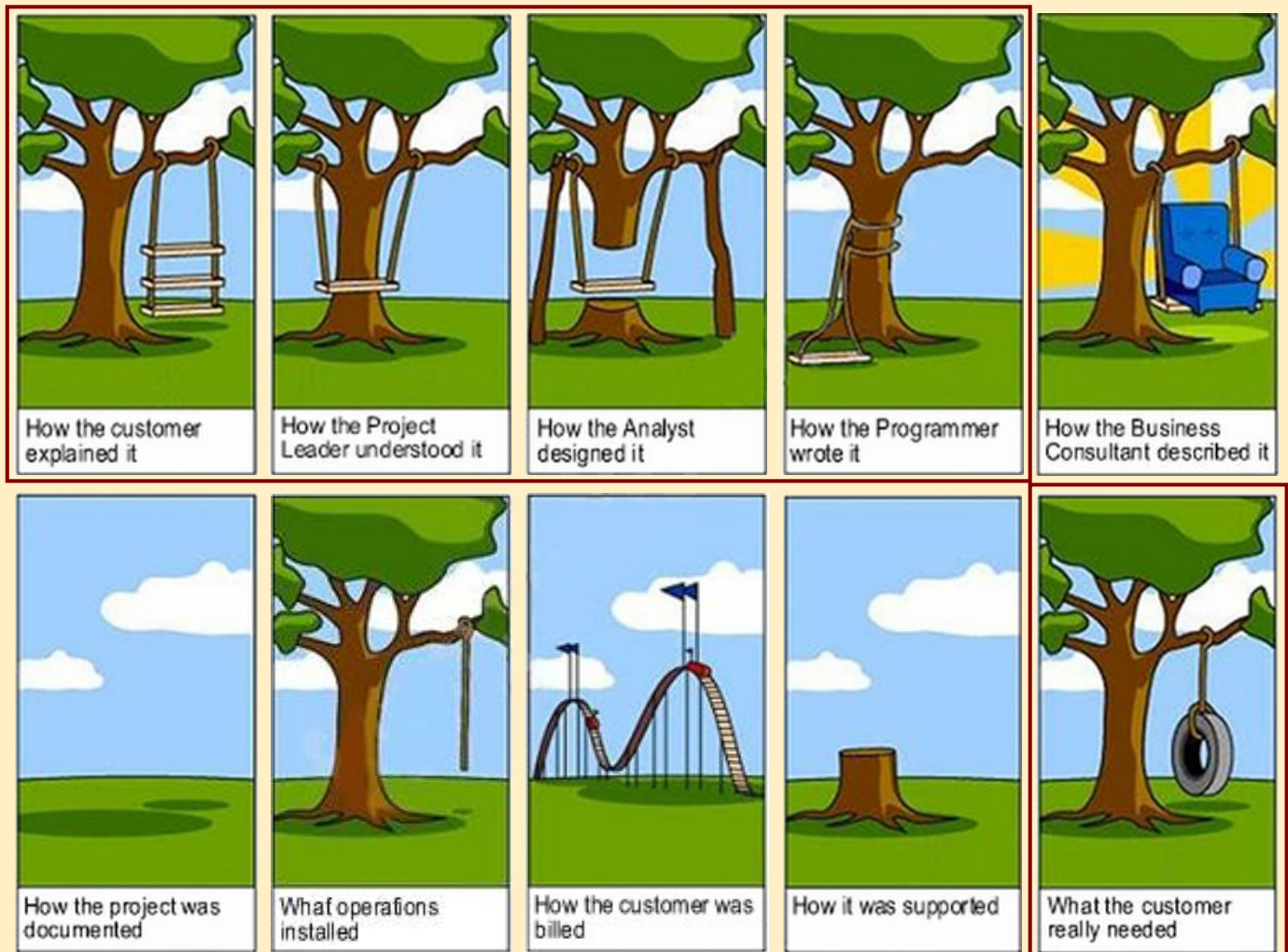
- **Motiváció**
 - Miért fontosak a tervezési folyamat ezen szakaszai?
 - Milyen elvárások vannak a specifikációval szemben?
 - Milyen módszerei vannak a specifikáció készítésnek?
- **Az általános követelménykezelés feladatai**
 - Követelmények nyilvántartása
 - Követhetőség a verifikációhoz
- **Félformális specifikáció**
 - Specifikus technikák: SysML
- **A követelményspecifikáció verifikációja**
 - Általános kritériumok
 - Specifikus kritériumok UML állapottérképekre (mintapélda)

2

Motiváció

- **Tapasztalat: Sok hiba visszavezethető hiányos vagy ellentmondásos specifikációra**
 - Példa: Meta Group felmérés, 2003:
 - Az IT projekt kudarcok 60%-70%-a a nem kielégítő követelményelemzésre vezethető vissza
 - Példa: 203 szoftverfejlesztési projekt utólagos felülvizsgálata „An analysis of defect densities found during software inspections” (Journal on Systems Software)
 - Gyakoribbak a hibák a **specifikáció elkészítésének fázisában**, mint a későbbi, implementációs fázisokban
 - Példa: Voyager és Galileo űrszondák szoftver tesztelése során felfedezett hibák okainak elemzése
 - **78% (149/192) specifikációs hiányosság**, ebből
 - 23% veszélyes állapotban ragadás (nincs kilépés)
 - 16% időzíteni kényszerek megadásának hiánya
 - 12% nincs specifikált reakció külső eseményre
 - 10% bemeneti érték ellenőrzésének hiánya

3



4

Elvárások, követelmények és specifikáció

- **Követelmény (requirement):**
 - Bejövő igény, vízió, elvárás
 - Felhasználóktól (user)
 - Érdekeltektől (stakeholder: hatóság, vezetőség, operátor, ...)
 - Validáció alapja
- **Specifikáció (specification, requirement specification, system specification):**
 - Tervezők, fejlesztők felé átalakított elvárások
 - A követelményelemzés (absztrakció, strukturálás, szűrés) eredménye
 - Sokféle típus
 - Rendszerspecifikáció, architektúra specifikáció, tervspecifikáció
 - Verifikáció alapja

5

Elvárások a specifikációval szemben

- A követelmények teljes lefedése
 - Funkcionális követelmények
 - Extra-funkcionális követelmények
- Megfogalmazás: Egyértelmű, igazolható, megvalósítható
- Javasolt megoldások:
 - Szigorú specifikációs nyelv (pl. formális nyelvek)
 - Ellenőrzött (tervezési illetve specifikációs) minták használata
 - Utólagos ellenőrzés
- Példa: EN 50128 szabvány által adott lehetőségek
 - Formális módszerek (VDM, Z, B, TL, PN, ...)
 - Félformális módszerek (diagram alapú technikák, UML)
 - Strukturált metodika (JSD, SADT, SSADM)
 - Emellett természetes nyelvű megadás is szükséges!

6

Tartalomjegyzék

- Motiváció
 - Miért fontosak a tervezési folyamat ezen szakaszai?
 - Milyen elvárások vannak a specifikációval szemben?
 - Milyen módszerei vannak a specifikáció készítésnek?
- **Az általános követelménykezelés feladatai**
 - Követelmények nyilvántartása
 - Követhetőség a verifikációhoz
- Félformális specifikáció
 - Specifikus technikák: SysML
- A követelményspecifikáció verifikációja
 - Általános kritériumok
 - Specifikus kritériumok UML állapottérképekre (mintapélda)

7

A követelménykezelés feladatai (áttekintés)

- Követelmények hatékony, strukturált tárolása
 - Hierarchikus elrendezés, tulajdonságokkal
- Követelmények finomítása és kapcsolódása
 - Specifikáció -> Rendszerterv -> Modulterv -> Forráskód -> Teszt
- Követelmény életciklus támogatása
 - Felvétel, törlés, változás, finomítás, kapcsolatok megjelenése
- **Analízis lehetőségek**
 - **Hatás** analízis (impact analysis): változáskezelés
 - Mit befolyásol, ha a követelmény megváltozik?
 - **Eredet** analízis (derivation analysis): költség-haszon elemzés
 - Milyen követelményre vezethető vissza? Miért van itt, szükséges-e?
 - **Fedettség** analízis (coverage analysis): projekt követés
 - Mely követelmények nincsenek implementálva?
 - Mely követelmények nincsenek tesztelve?

Követhetőség
(traceability)
szükséges

8

A követelménykezelés „kézi” módszerei

- Természetes nyelvű követelmény dokumentum
 - Strukturálás adott (fejezetek, alfejezetek)
 - Követelményazonosítók felvétele
- Követelményfinomítás: Táblázatos nyilvántartás
 - Követelményazonosítók szerepelnek
 - Különböző dokumentumokból (SRS, SA, MDS, MTS, ...)
 - Analízis makrókkal támogatható
 - Üres, többszörös, ... mezők kikeresése
- Követhetőségi mátrix: Táblázatos forma
 - Követelmények azonosítói
 - Kódrészlet azonosítók (funkció szint tipikus)
 - Teszt azonosítók
 - Sikeres/sikertelen teszteredmény bejelölése

9

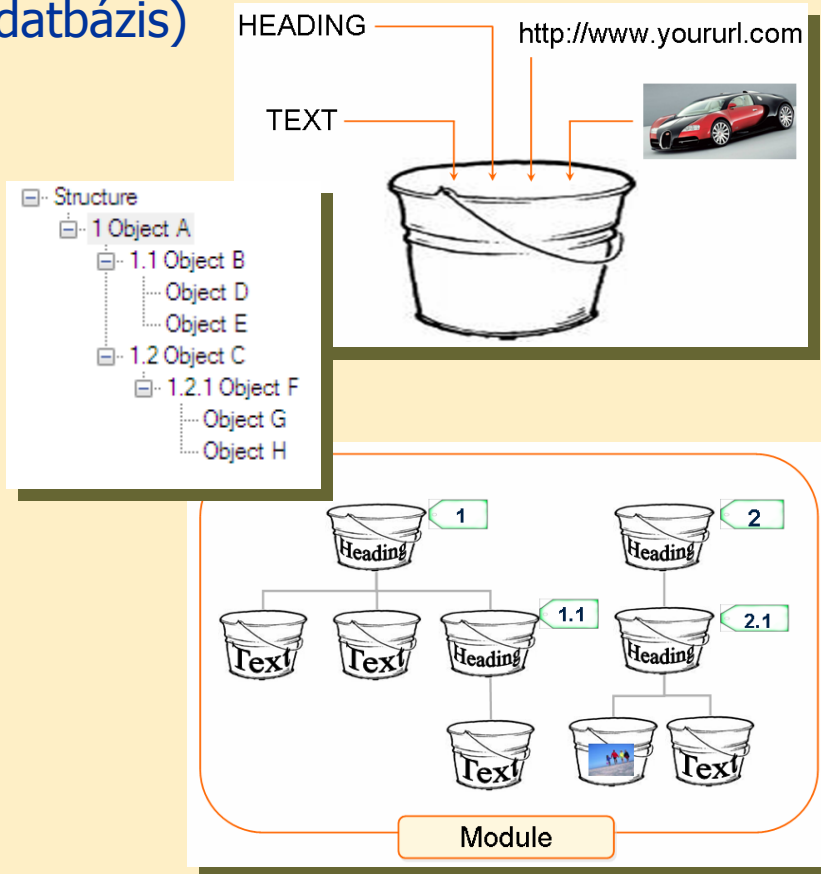
Automatikus követelménykezelők feladatai

| | |
|--|---|
| Követelmények nyilvántartása | Hierarchikus felépítés |
| Kapcsolatok nyilvántartása | Sokféle reláció: Kompozíció, származtatás, finomítás, bizonyítás, .. Követelmény – Modell – Kód – Teszt – Teszteredmény között |
| Követelmény változások kezelése | Időbeli struktúra, triggerek |
| Navigáció a kapcsolatokon | Előre: pl. hatás analízishez Vissza: pl. motiváció analízishez |
| Fedettségi listák készítése | Lefedetlen követelményekhez Indokolatlan megvalósításhoz |
| Jogosultságok kezelése | Hozzáférés szerepek |
| Értesítési rendszer | Változások |
| Biztonsági megoldások | Sértetlenség |

10

Megvalósítás: Strukturált tárolás

- Objektum modell (adatbázis)
 - Általános „tároló”
 - Egyedi azonosító
 - Modulok
- Hierarchia
- Tulajdonságok
 - Fejléc / szöveg
 - Hozzáférési jogok
 - Történet
 - Attribútumok
 - Prioritás
 - Státusz
 - Költség
 - ...
 - Linkek



Példa: DOORS

| ID | Last Modified By | Car user requirements | Priority | Percentage cost | Comments |
|-----------|------------------|---|-----------|-----------------|-------------------|
| TRN-CSR-1 | Bill Young | 1 Introduction | Mandatory | 0.172835 | |
| TRN-CSR-2 | Bill Young | This module contains the user requirements for a new car to be commercially available by 1 August 2006. | Mandatory | | ... form text |
| TRN-CSR-3 | Bill Young | 2 User types | Desirable | 1.370889 | ... a text field. |
| TRN-CSR-4 | Bill Young | 2.1 Nationalities | Mandatory | 0.642687 | |
| TRN-CSR-5 | Bill Young | The car will be used in the following countries: UK, USA, Northern Europe, Eastern Europe, Japan, Russia, Australia. | Mandatory | 0.769025 | |
| TRN-CSR-6 | Bill Young | 2.2 User sizes | Mand | | |
| TRN-CSR-7 | Bill Young | People come in all shapes and sizes. The car must be suitable for people maximum and minimum sizes. ffgfg to 2 m weighing 25 kilograms to | Mand | | |

Objektum azonosító

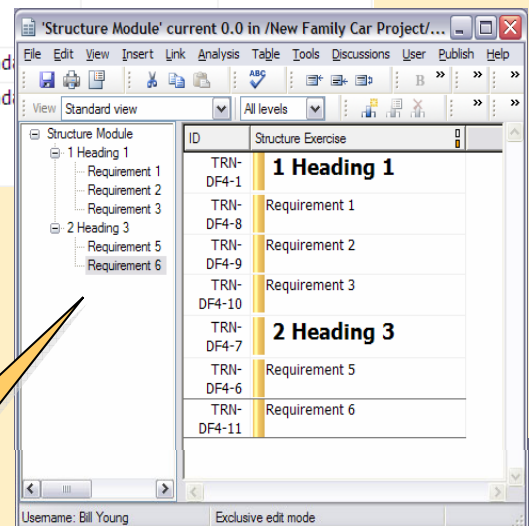
Változás-jelző

Fejléc objektum

Szöveges objektum

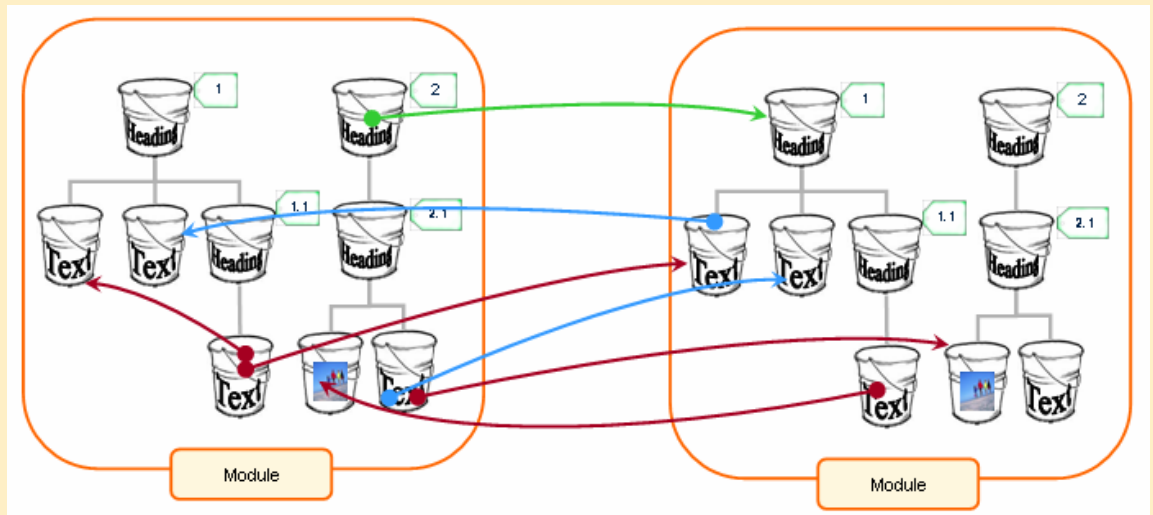
Hierarchia

Tulajdonságok



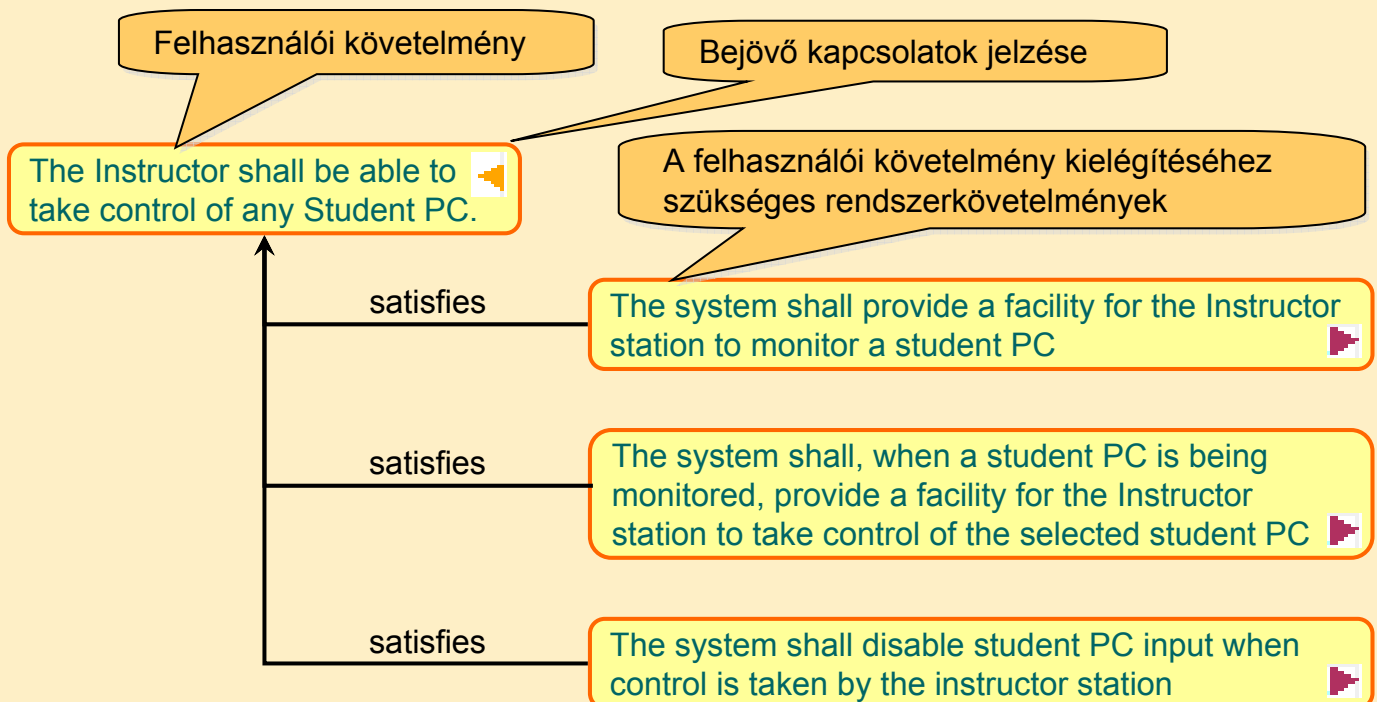
Megvalósítás: Kapcsolatok (linkek)

- Relációk: Rendezett párok
 - Objektumok között
 - Külső kapcsolatok
- Típusok
 - Finomítja
 - Kielégíti
 - Teszteli
 - ...



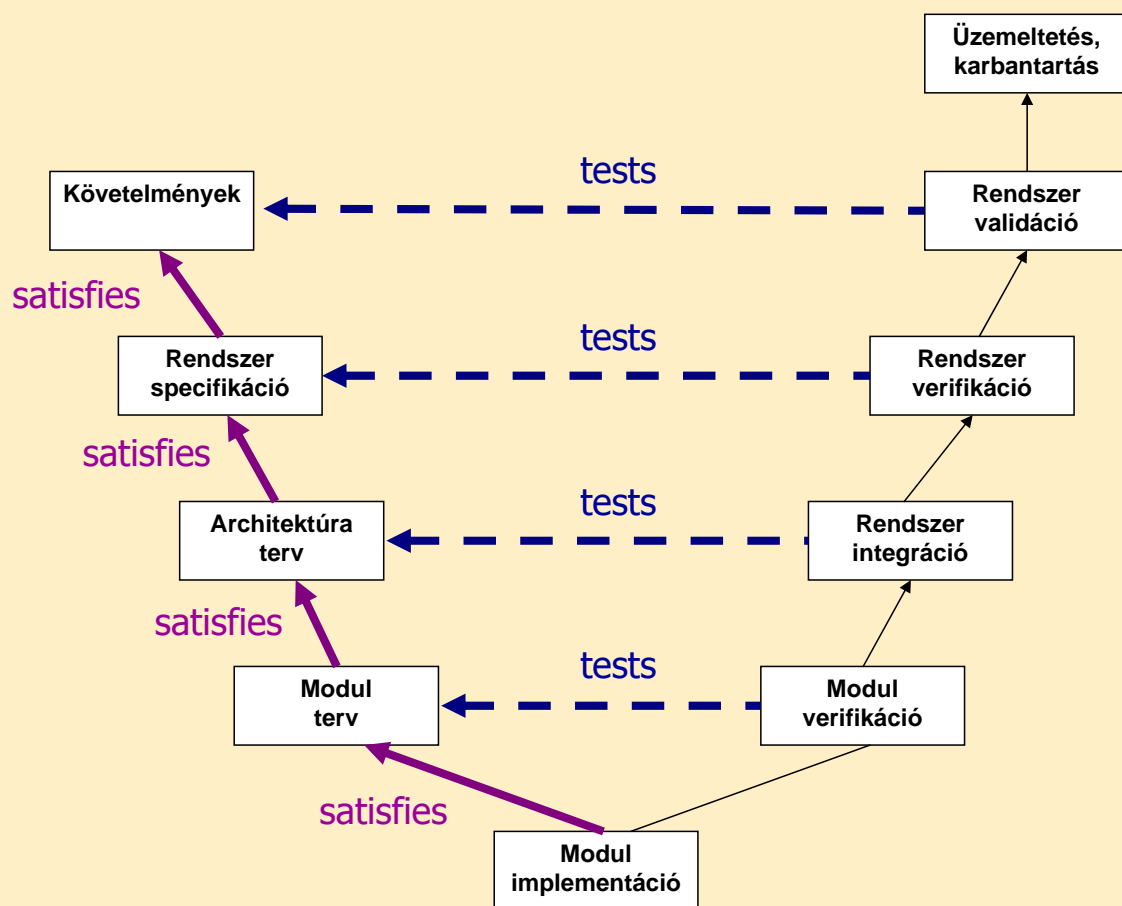
15

Példa: DOORS



14

Relációk a V-modellben



15

Megvalósítás: Követelmény életciklus

- **Objektum szintű változások**
 - Közvetlen szerkesztés (létrehozás, módosítás, törlés)
 - Megváltozott objektum kapcsolatok jelzése (suspect link)
 - **Változtatási javaslatok** menedzselése (elkészítés, csoportosítás, felülvizsgálat, érvényre juttatás)
- **Változások mint események**
 - Scriptek indítására használhatók
- **Nagyléptékű változások**
 - **Baseline** definiálás (állapot rögzítése)
 - Inkrementális változások vizsgálhatók
 - Összehasonlítás lehetséges
 - Követelmény-partíciók **kiajánlása**, távoli szerkesztés, szinkronizálás, visszavétel

16

Megvalósítás: Analízisek

- **Követhetőség alapja:**
Navigálás a kapcsolatokon keresztül
 - Kiterjedés (scope) és mélység kijelölhető
 - Irány kijelölhető (előre, hátra)
- **Script nyelv használható**
 - Bejárás, kigyűjtés
 - Tulajdonságok megváltoztatása
- **Jelentések készítése**
 - **Hatás** analízis: Előre navigálás alapján
 - **Eredet** analízis: Hátra navigálás alapján
 - **Fedettség** analízis
 - Szűrés: Navigálás kód objektumokig, teszt objektumokig
 - Objektumok kigyűjtése: Nincs kapcsolat adott célhalmazig
 - Pl. nincs megvalósítás, nincs sikeres teszt

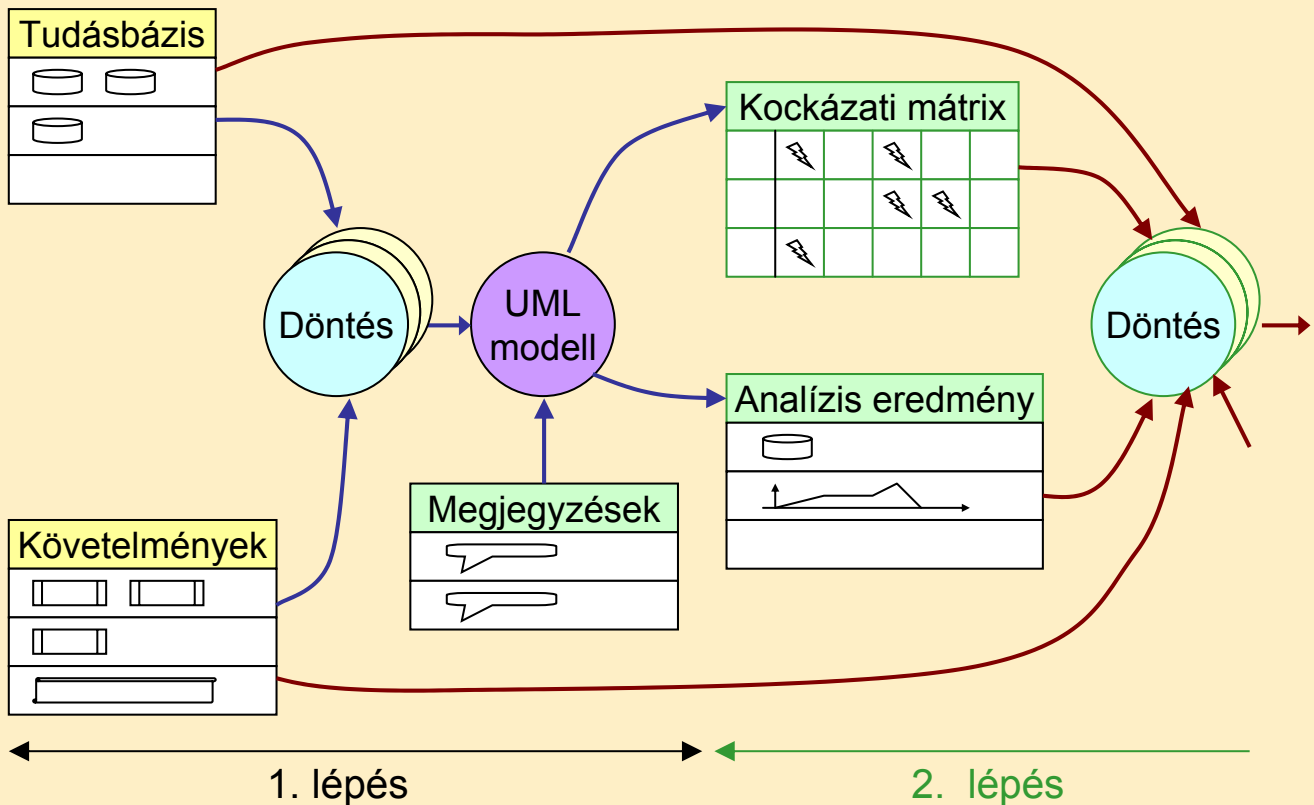
17

Megvalósítás: Járulékos funkciók

- **Nézetek az objektum listában**
 - Szűrés hierarchiaszintekre, tulajdonságokra, ...
- **Űrlapok a tulajdonságok szerkesztésére**
- **Megosztott szerkesztés (csoportmunka)**
 - Objektum (követelmény) szintű zárolás
- **Dokumentáció generálás**
 - Hierarchia (=> fejezetek) alapján rendezett szöveges és egyéb objektumok
 - **Exportálás:** Strukturált külső dokumentumban (pl. Word) történt változtatások vissza is olvashatók (struktúra megőrzése)
 - **Importálás:** Előkészítés szükséges
- **Webes hivatkozások (pl. e-mailben küldhető)**
 - Adatbázis, projekt, objektum hivatkozás

18

Egy más aspektus: Tervezői döntés adatbázis



19

Követelmény alapú eszközláncok: DECOS Test Bench

- **Verifikációs tevékenység felvétele a követelmények mellé**
 - Tervezett és rögzített verifikáció (pl. biztonsági ügy)
 - Ellenőrzések a követelmények és szabványok alapján
 - Többféle tevékenység kombinálható
- **Verifikációs eszközláncok kialakítása (általában külső)**
 - **Analízis:** analízis modell generálása, analízis végrehajtása, eredmények visszacsatolása
 - **Tesztelés:** modell alapú tesztgenerálás, teszt végrehajtás, teszt eredmény kiértékelés
 - **Mérések:** mérési konfigurálása, végrehajtása, eredmények értékelése
- **Verifikációs eszközláncok indítása a követelménykezelőből**
 - Triggerek alapján; script nyelven programozható
- **Verifikáció státuszának rögzítése**
 - Ellenőrzött modell, sikeresen tesztelt követelmény

20

Verifikációs terv

Módszerek kiválasztása:

- V-plan SIL4 rendszerhez a generikus IEC 61508-3 V-plan alapján
- Referenciákat tartalmazhat V&V eszközökre és módszerekre

| Generic Template for Software | SIL1Rec | SIL1Ef | SIL2Rec | SIL2Ef | SIL3Ef | SIL3Rec | SIL4Rec | SIL4Ef |
|--------------------------------------|---------|--------|---------|--------|--------|---------|---------|--------|
| 1 Generic V Plan IEC 61508-03 | | | | | | | | |
| Activity VP-A1 | HR | High | HR | High | High | HR | HR | High |
| Activity VP-A2 | ... | Low | R | Medium | Medium | R | HR | High |
| Activity VP-A3 | R | Medium | R | Medium | Medium | R | R | Medium |
| Activity VP-A4 | HR | High | HR | High | High | HR | HR | High |
| Activity VP-A5 | ... | Low | R | Medium | Medium | R | HR | High |
| Activity VP-A6 | R | Medium | R | Medium | Medium | R | R | Medium |
| Activity VP-A7 | ... | Low | ... | Low | Medium | R | R | Medium |
| Activity VP-A8 | ... | Low | R | Medium | Medium | R | HR | High |
| Activity VP-A9 | R | Medium | HR | High | High | HR | HR | High |

| ID | Generic Template for Software | Recommendation | Effectiveness |
|---------------------------|-------------------------------|----------------|---------------|
| 1 Test-IEC61508-03 | | | |
| 4 | Activity VP-A1 | HR | High |
| 29 | Activity VP-A2 | HR | High |
| 28 | Activity VP-A3 | R | Medium |
| 27 | Activity VP-A4 | HR | High |
| 26 | Activity VP-A5 | HR | High |
| 25 | Activity VP-A6 | R | Medium |
| 24 | Activity VP-A7 | R | Medium |
| 30 | Activity VP-A8 | HR | High |
| 31 | Activity VP-A9 | HR | High |

21

| ID | DECOS Artifacts | Version | Status | Data Directory Remote |
|------|------------------------------|---------|-----------|---|
| AUT1 | 1 PIMs | | | |
| AUT2 | PIM for Tiny Example | 1.0 | Failed | ftp://FTPDecos@ftp.sp.se/private/DECOS/PIM/1.0 |
| AUT3 | PIM Updated for Tiny Example | 1.1 | Completed | ftp://FTPDecos@ftp.sp.se/private/DECOS/PIM/1.0 |
| AUT5 | 2 SCADE Model | | | |
| AUT6 | SCADE model for Roll Control | 1.0 | | |

Egy példa a Test Bench alkalmazására

| ID | VWStatus | Tool Integration Status | Type |
|-------------------------|-----------|-------------------------|------------|
| 1 PIM Validation | | | |
| 1 | Completed | | Compound |
| 51 | Completed | Not processing | Elementary |

PIM validation

| ID | VWStatus | Tool Integration Status | Type |
|---------------------------------|-----------|-------------------------|------------|
| 1 SCADE Model Validation | | | |
| 1 | Completed | | Compound |
| 51 | Completed | Not processing | Elementary |

V-Plan for SCADE model validation

22

Eszközláncok indítása a DOORS-ból

Eszközök végrehajtása:

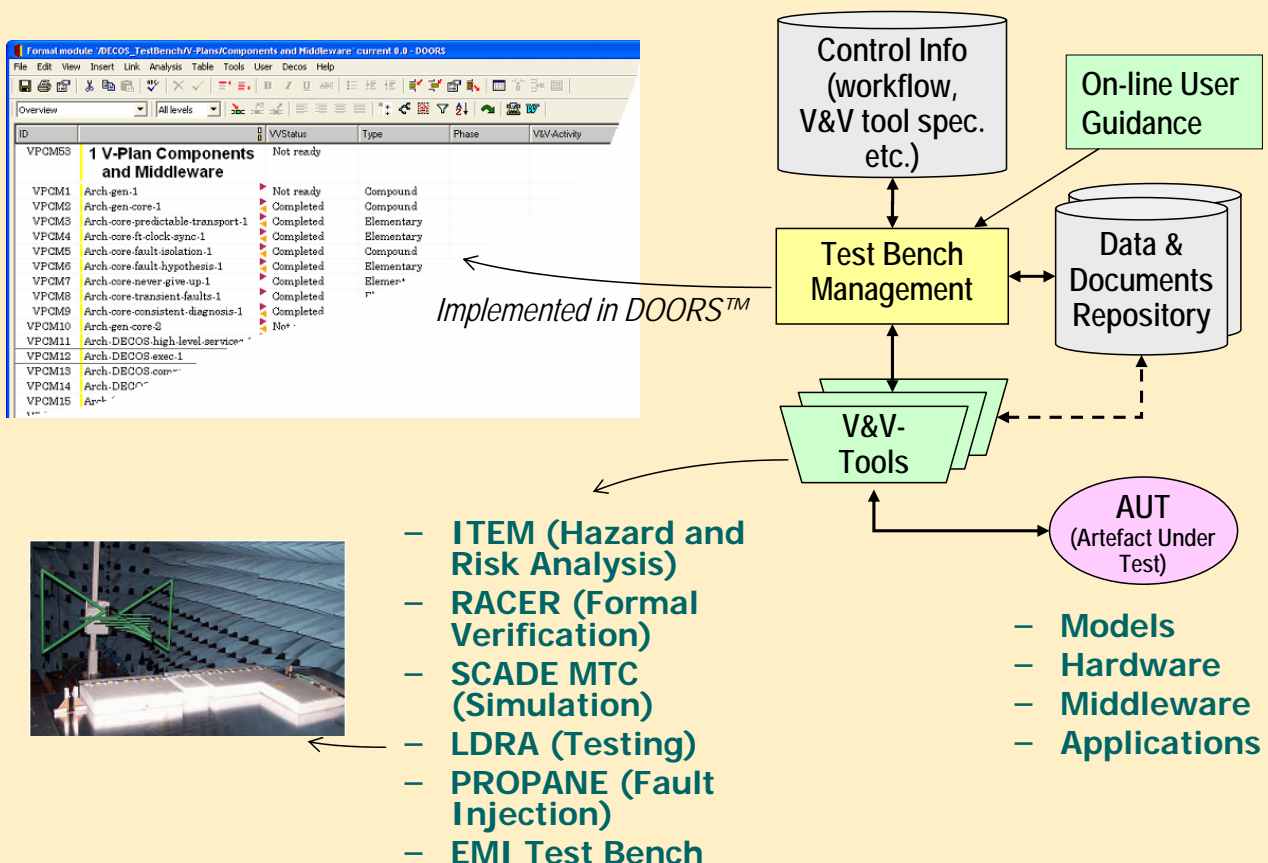
- **Belső** eszköz: Pl. egyszerű ellenőrző lista
 - Megvalósítható a DOORS keretein belül
- **Automatikus helyi** eszközvégrehajtás: pl. RACER (ontológia alapú ellentmondásmentesség vizsgálat)
 - Automatikus indítás (scriptből) és eredmény becsatolás
- **Távoli** eszközvégrehajtás: pl. PROPANE (SWIFI)
 - Indítás távoli hozzáféréssel (pl. üzenet alapú)
- **Kézi** eszközvégrehajtás: pl. EMI Hardware Test Bench
 - Eszköz indítási és eredmény beviteli dialógus

Eredmények tárolása:

- Távolról is elérhető **adattár** (repository)

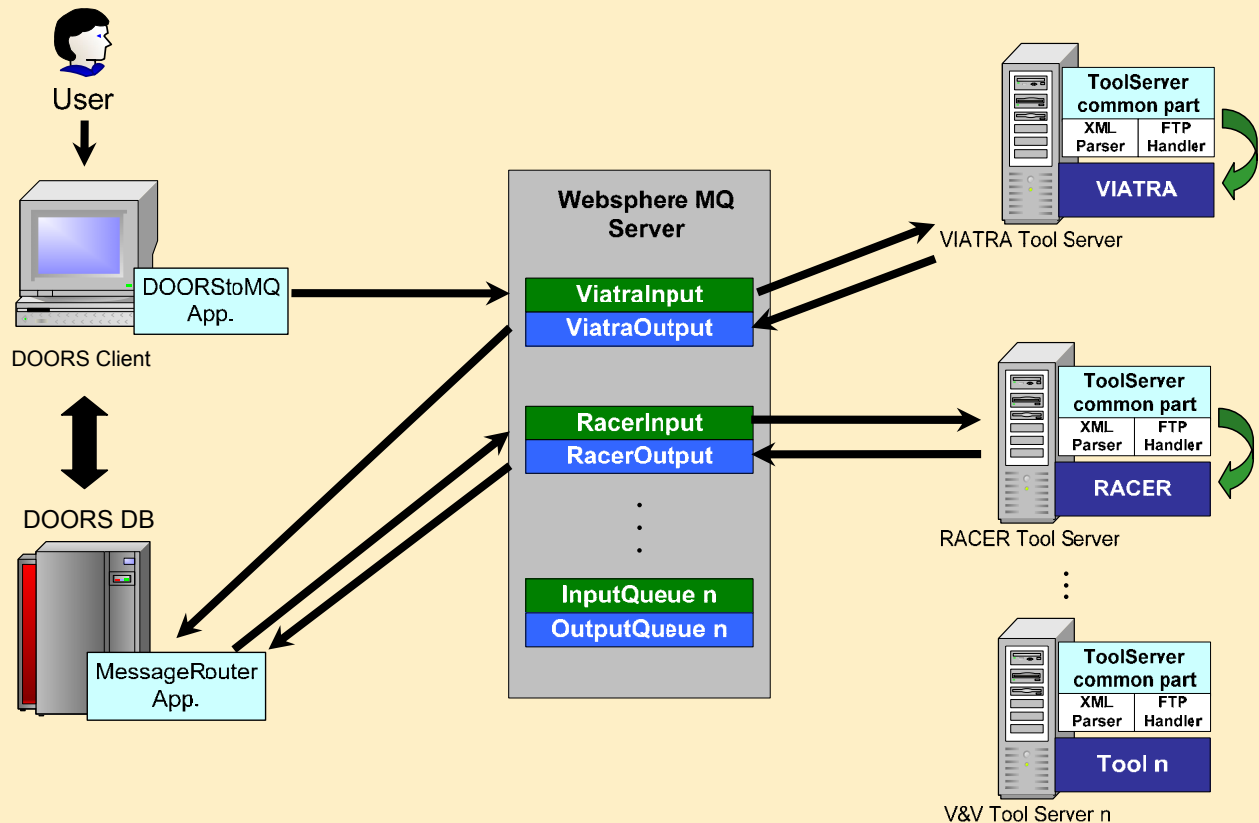
23

Verifikáció „menedzselése” a követelménykezelővel



24

Példa végrehajtás: PIM ellenőrés RACER-rel



25

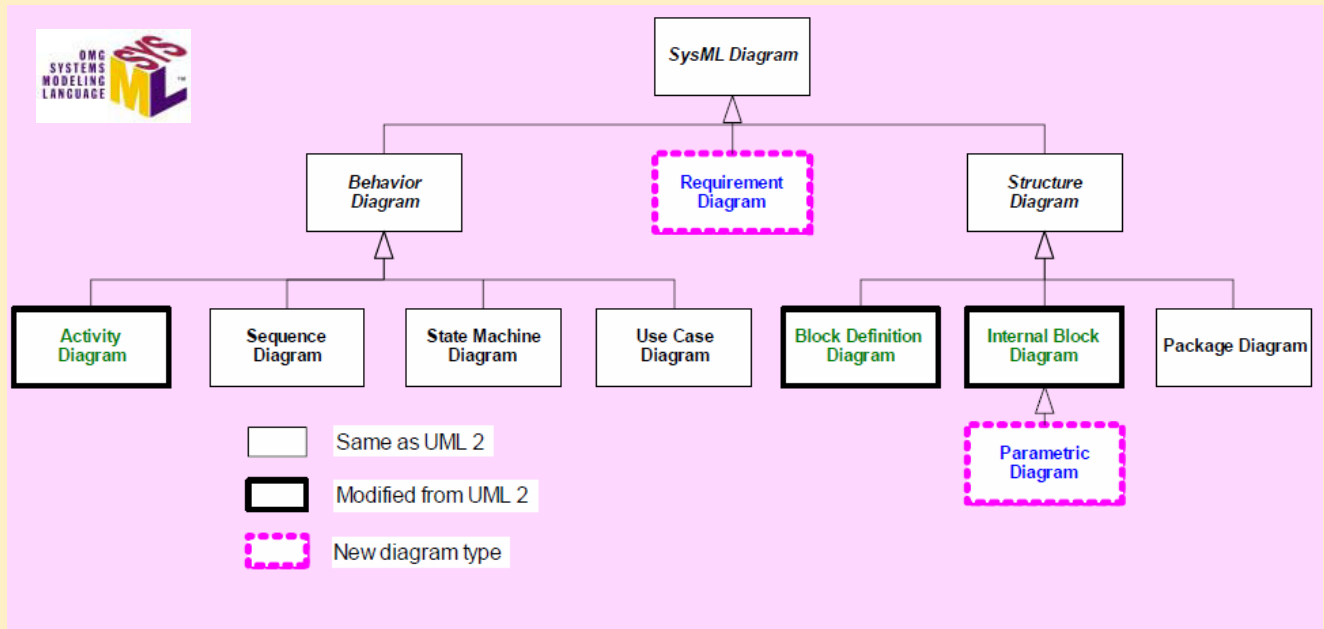
Tartalomjegyzék

- **Motiváció**
 - Miért fontosak a tervezési folyamat ezen szakaszai?
 - Milyen elvárások vannak a specifikációval szemben?
 - Milyen módszerei vannak a specifikáció készítésnek?
- **Az általános követelménykezelés feladatai**
 - Követelmények nyilvántartása
 - Követhetőség a verifikációhoz
- **Félformális specifikáció**
 - Specifikus technikák: SysML
- **A követelményspecifikáció verifikációja**
 - Általános kritériumok
 - Specifikus kritériumok UML állapotterképekre (mintapélda)

26

Félformális követelményspecifikáció: SysML

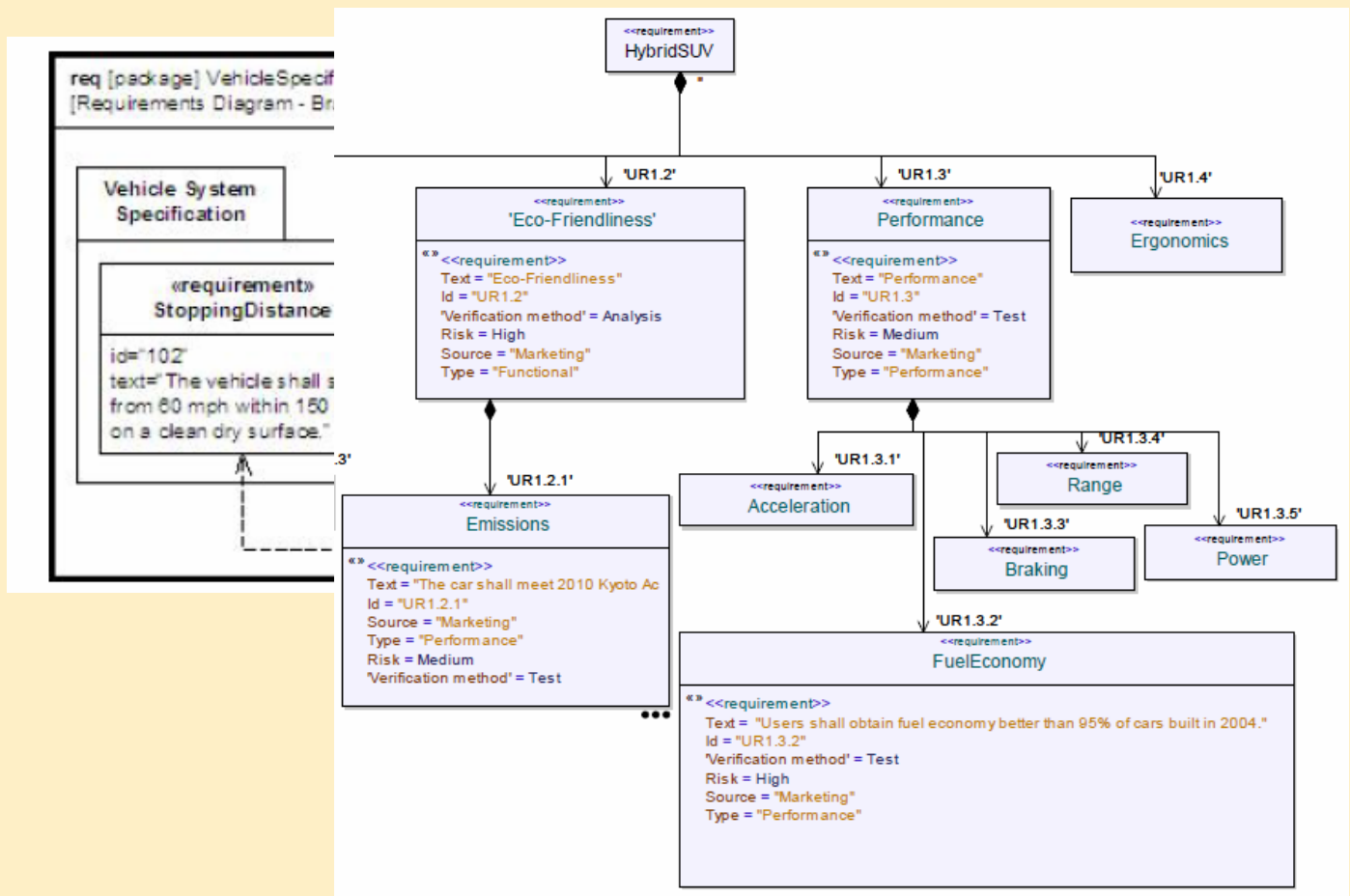
- Systems Modeling Language
 - UML részhalmoz egy kiterjesztése rendszertervezéshez
 - Fő újdonságok: Requirements és Parametric diagram



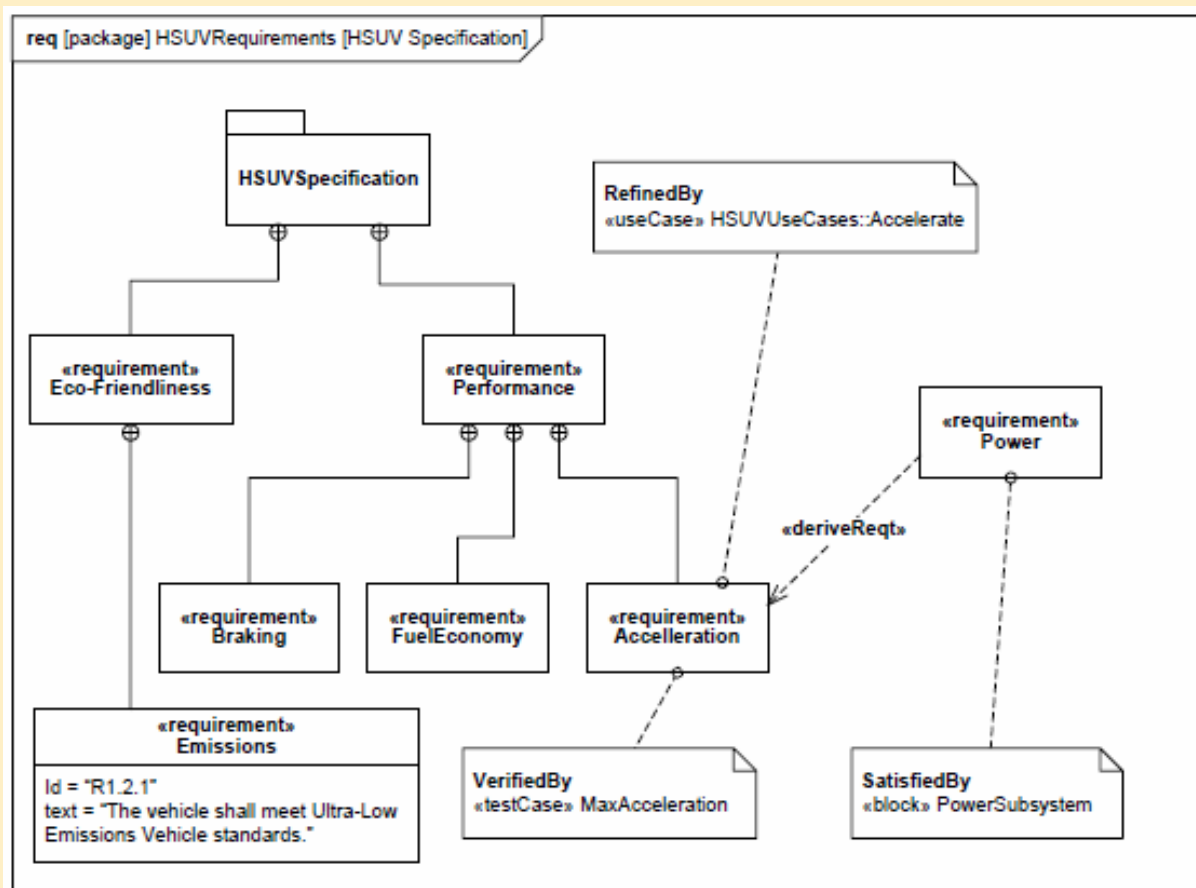
Requirements diagram

- Követelmények (szöveges is) tárolása azonosítóval
 - <<requirement>> stereotype
 - Id és text mezők
 - Felhasználói attribútumok: pl type, source, risk, ...
 - Táblázatos forma is támogatott
- Követelmények hierarchikus csomagokba rendezhetők
 - Funkcionális, teljesítmény, ... kategóriák
- Követelmények közötti finomítás (~ subclass), kompozíció
- Relációk használhatók (callout: megjegyzésekben):
 - Copy: követelmények között (master – slave)
 - Trace: követelmények között (client – supplier)
 - DeriveReq: követelmények között (forrás – származtatott)
 - Refine: követelmények és terv elemek között (pl. szövegeshez)
 - Satisfy: követelmények és terv vagy implementáció elemek között
 - Verify: követelmények és teszt elemek között

Requirement diagram példa: Struktúra

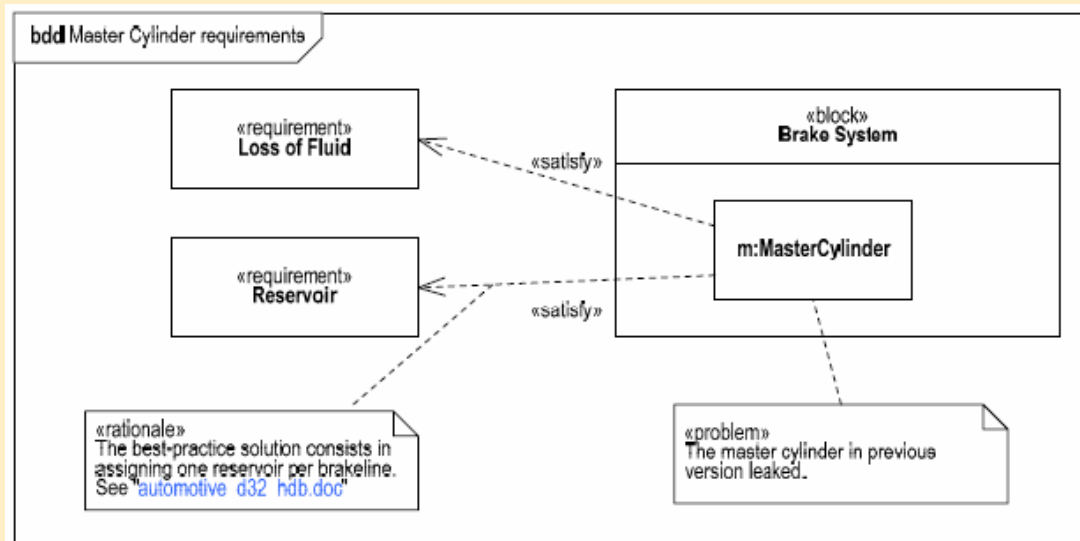


Requirement diagram példa: Relációk



Requirement diagram példa: Tervezői döntések

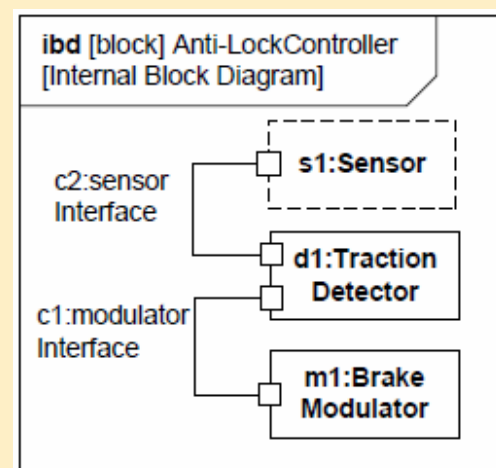
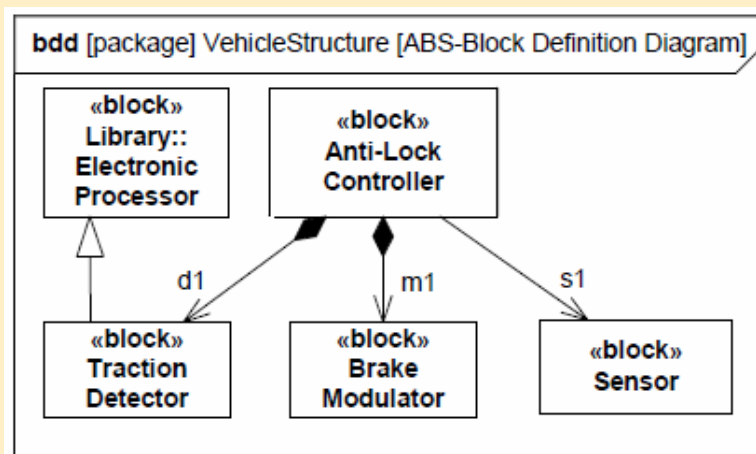
- Tetszőleges modell elemhez köthető megjegyzések (előredefiniált stereotype):
 - <<problem>>: Probléma, döntést igénylő felvetés
 - <<rationale>>: Megoldás, magyarázat



31

Block diagram

- Block: A struktúra eleme (fekete / üveg doboz)
 - Komponens (nem csak szoftver)
 - A SysML-ben az UML 2.0 osztályokon alapul
- Internal block diagram:
 - Konkrét szerepek; típust a Block adja meg



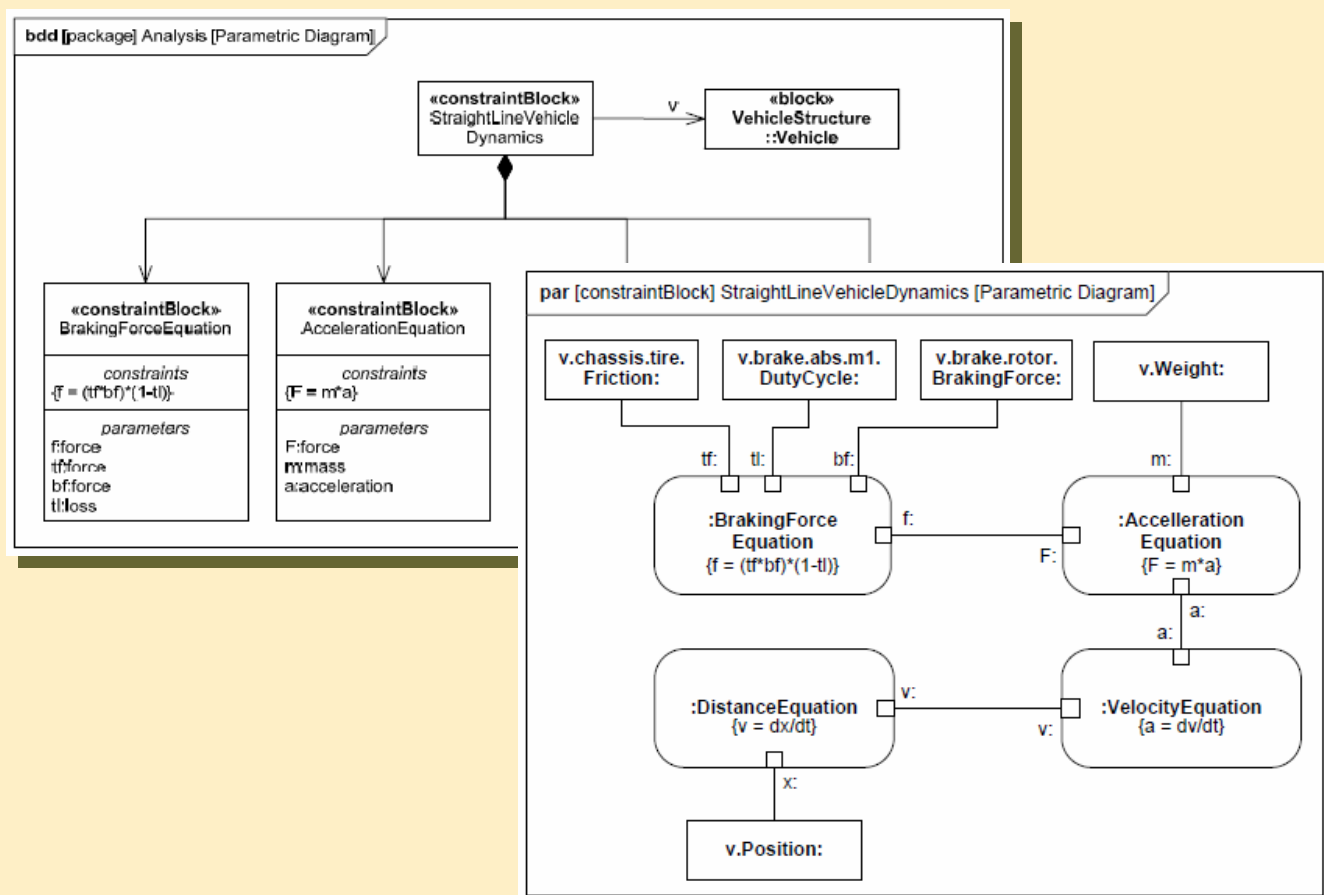
32

Parametric diagram

- Cél: Ellenőrizhető számszerű követelmények (kényszerek) megfogalmazása tulajdonságokra
 - Nem-funkcionális követelmények aspektusa
 - **Analízis** (pl. teljesítmény, megbízhatóság) támogatása
- ConstraintBlock: Összefüggések megadása
 - **Formális** (pl. MathML, OCL), vagy **informális** alakban
 - **Analízis eszközhöz igazítható** (nem SysML specifikus)
- Parametric diagram: Alkalmazás
 - Az összefüggések (Constraint block) **alkalmazása** egy adott környezetben
 - Kötések értékek között

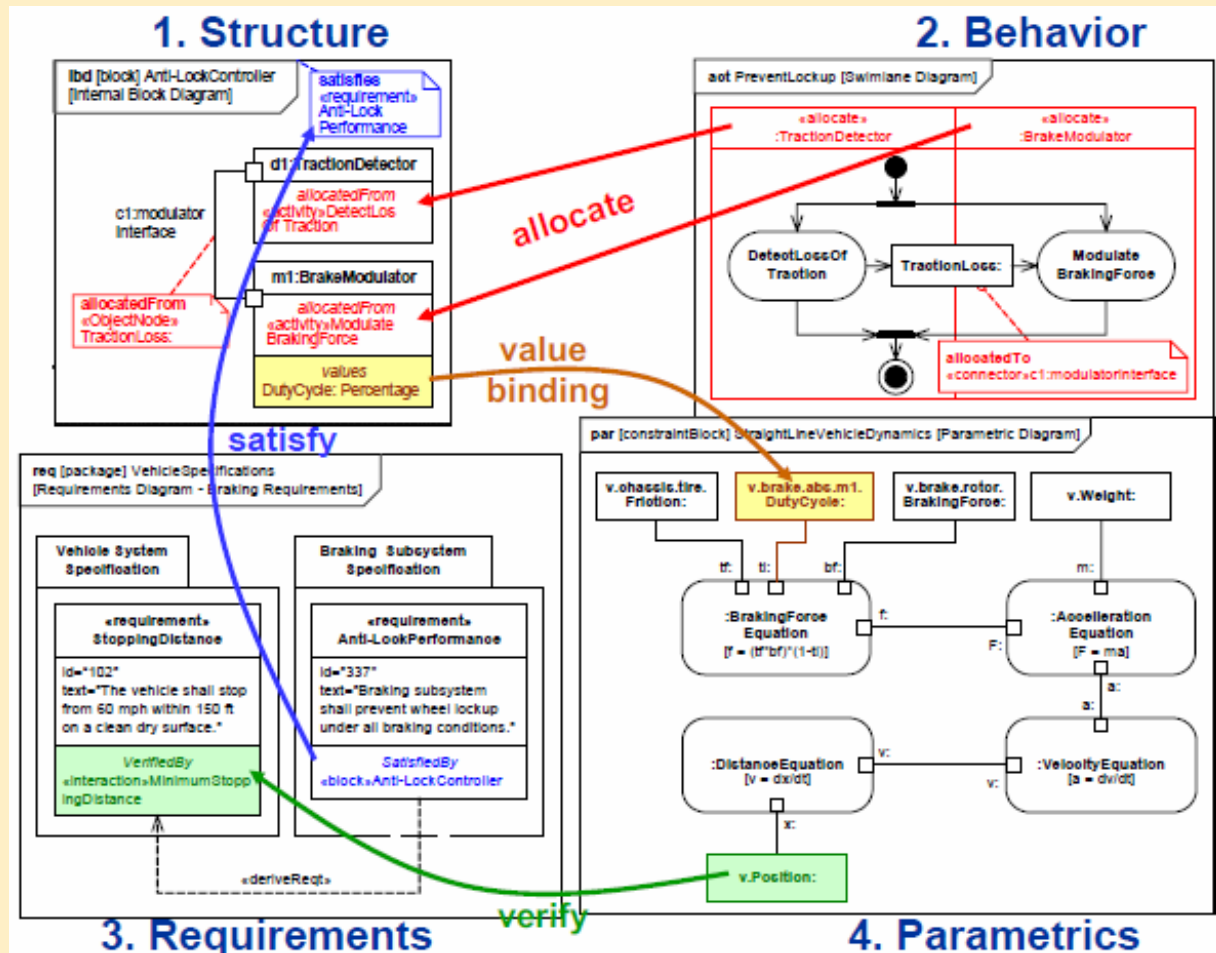
33

Parametric diagram példa



4

Relációk diagramok között: Követhetőség



36

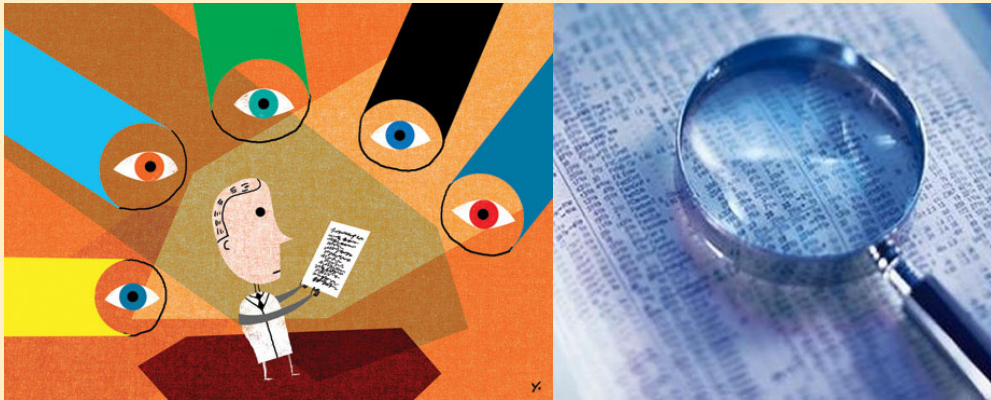
Tartalomjegyzék

- Motiváció
 - Miért fontosak a tervezési folyamat ezen szakaszai?
 - Milyen elvárások vannak a specifikációval szemben?
 - Milyen módszerei vannak a specifikáció készítésnek?
- A követelménykezelés általános feladatai
 - Követelmények nyilvántartása
 - Követhetőség a verifikációhoz
- Félformális specifikáció
 - Specifikus technikák: SysML
- A követelményspecifikáció verifikációja
 - Általános kritériumok
 - Specifikus kritériumok UML állapotterképekre (mintapélda)

37

Ellenőrzési módszerek

- **Ellenőrző listák**
 - Tipikus hibák esetén hatékony (újra ne kövessük el)
 - Teljességet nem várhatunk
- **Statikus analízis**
 - Hiányosságok, ellentmondások kiszűrése a specifikáció (ill. terv, kód) végrehajtása nélkül
 - Analógia: Átolvasás „sorról sorra”
 - Szerepek: Szerző, átvizsgáló, tesztelő



38

Vizsgálati szempontok biztonságkritikus rendszerekben

- **Teljesség**
 - Funkciók, hivatkozások, eszközök
- **Konzisztencia (ellentmondásmentesség)**
 - Külső és belső
 - Követhetőség
- **Megvalósíthatóság**
 - Erőforrások
 - Használhatóság
 - Karbantarthatóság
 - Kockázatok: költségbeli, technikai, környezeti
- **Tesztelhetőség**
 - Specifikus
 - Egyértelmű
 - Számszerűsíthető

39

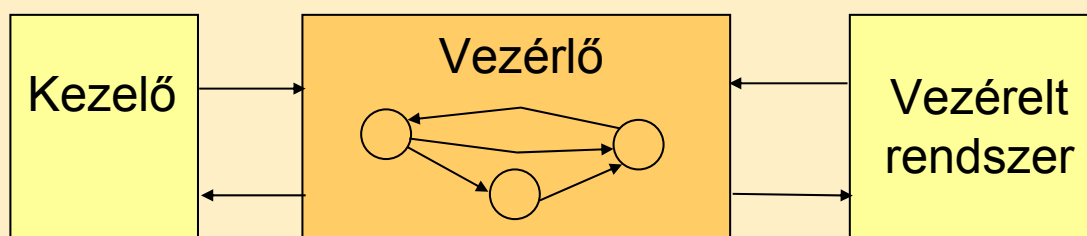
Vizsgálati szempontok IEEE Std 830-1998 alapján

- Helyes
 - A szoftverre vonatkozó követelményeknek (elvárásoknak) megfelelő
 - Konzisztens a külső forrásokkal (pl. szabványok)
- Egyértelmű
 - Nem félreérthető, egy jelentése van
 - Hasznosak a formális, félformális specifikációs nyelvek
- Teljes
 - Minden (érvényes, érvénytelen) bemenetre van specifikált viselkedés
 - TBD csak indoklással és a feloldás módjával
- Konzisztens
 - Nincs belső ellentmondás, egységes a terminológia
- Fontosság és stabilitás szempontjából rendezett
 - Követelmények szükségessége, változatlansága felmérve
- Ellenőrizhető
 - Megállapítható egyértelműen, ha nem teljesül egy követelmény
- Módosítható
 - Nem redundáns, jól strukturált, jól elválasztott követelmények
- Követhető
 - Eredet becsatolható, további hatások hivatkozhatók

40

Vizsgálati szempontok reaktív rendszerekre (Leveson)

- Állapotdefiníció
- Bemenetek (események)
- Kimenetek
- Kimenetek és trigger kapcsolata
- Állapotátmenetek
- Ember-gép interfész



41

Példa: Vizsgálati szempontok reaktív rendszerekre

- Állapotdefiníció

- Bemenetek (események)

- Biztonságos a kezdőállapot

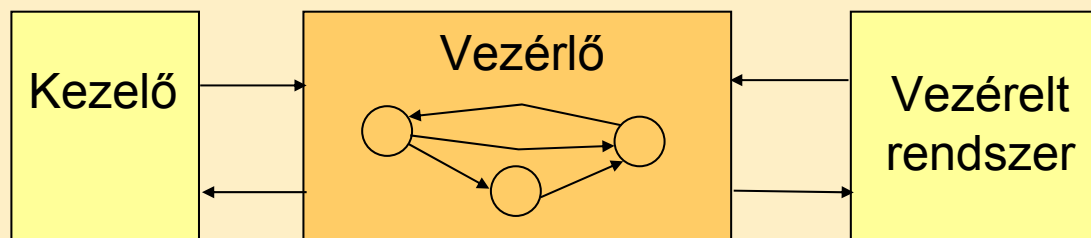
- Kimenetek (akciók)
- Belső modell aktualizálva van a környezettel

- Kimenetek (akciók)
- (kimaradó bemeneti események esetén

- time-out van, és nincs a kimeneten akció)

- Állapotát

- Ember-gép interfész



42

Példa: Vizsgálati szempontok reaktív rendszerekre

- Állapotdefiníció

- Bemenetek (események)

- Kimenetek (akciók)
- Minden bemenetre, minden állapotban van

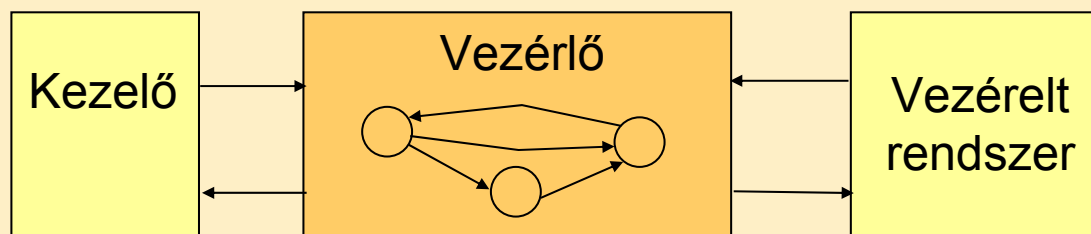
- Kimenetek (akciók)
- specifikált viselkedés (reakció)

- Állapotát
- Egyértelműek (determinisztikusak) a reakciók

- Van bemeneti ellenőrzés (értékbeli, időbeli)

- Ember-gép interfész
- Hibás bemenet kezelése specifikálva

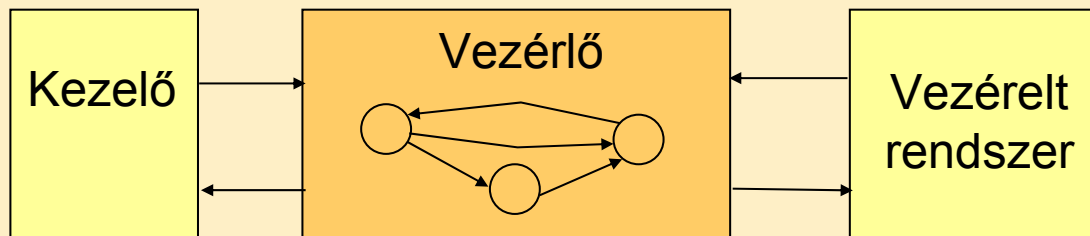
- Megszakítások gyakorisága korlátozva



43

Példa: Vizsgálati szempontok reaktív rendszerekre

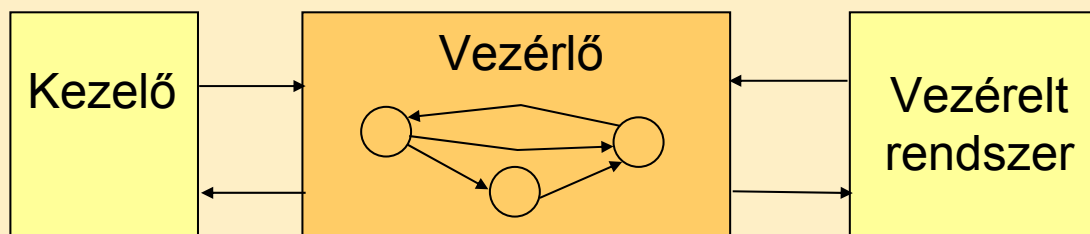
- Állapotdefiníció
 - Bemenetek (események)
 - Kimenetek
 - Kimenetek és trigger kapcsolata
 - Állapotátmenetek
 - Ember-gép interfész
- Hihetőségvizsgálat kritériumai specifikáltak
- Nincsenek fel nem használt kimenetek
- Környezeti feldolgozókéesség be van tartva



44

Példa: Vizsgálati szempontok reaktív rendszerekre

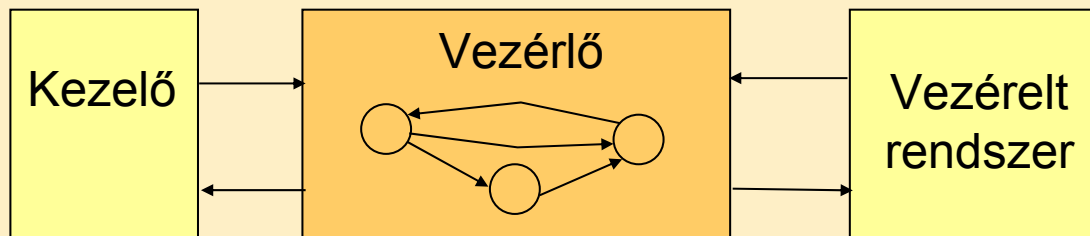
- Állapotdefiníció
 - Bemenetek
 - Kimenetek
 - Kimenetek és trigger kapcsolata
 - Állapotátmenetek
 - Ember-gép interfész
- Kimenetek hatása a bemeneteken keresztül ellenőrizve
- A szabályzási kör stabil



45

Példa: Vizsgálati szempontok reaktív rendszerekre

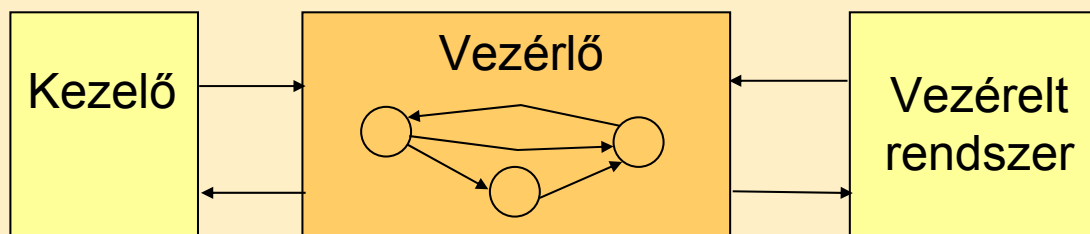
- **Állapot** - Minden állapot elérhető statikusan
- **Bevitel** - Állapotátmenetek visszafordíthatók (visszaút van)
- **Kimenet** - Több átmenet van veszélyes állapotból biztonságosba
- **Kimenet** - Megerősített átmenet van biztonságos állapotból veszélyes állapotba
- **Állapotátmenetek**
- **Ember-gép interfész**



46

Példa: Vizsgálati szempontok reaktív rendszerekre

- **Állapot** - Kezelő felé kimenő események specifikációja:
- **Bevitel** - Sorrendezés előírt (prioritással)
- **Kimenet** - Frissítés előírt
- **Kimenet** - Gyakoriság korlátozott (kezelő terhelhetősége)
- **Állapotátmenetek**
- **Ember-gép interfész**



47

Tartalomjegyzék

- Motiváció
 - Miért fontosak a tervezési folyamat ezen szakaszai?
 - Milyen elvárások vannak a specifikációval szemben?
 - Milyen módszerei vannak a specifikáció készítésnek?
- A követelménykezelés általános feladatai
 - Követelmények nyilvántartása
 - Követhetőség a verifikációhoz
- Félformális specifikáció
 - Specifikus technikák: SysML
- A specifikáció verifikációja
 - Általános kritériumok
 - Specifikus kritériumok UML állapottérképekre (mintapélda)

48

Példa: IAR VisualState eszköz

A statikus ellenőrzés UML állapottérképeken:

- Reset funkcionalitás
- Események, akciók felhasználása
- Állapotátmenetek statikus engedélyezettsége
- Állapotátmenet konfliktusok (azonos trigger)
- Állapotok statikus elérhetősége
- Nyelő állapot keresése (nincs kimenet)

49

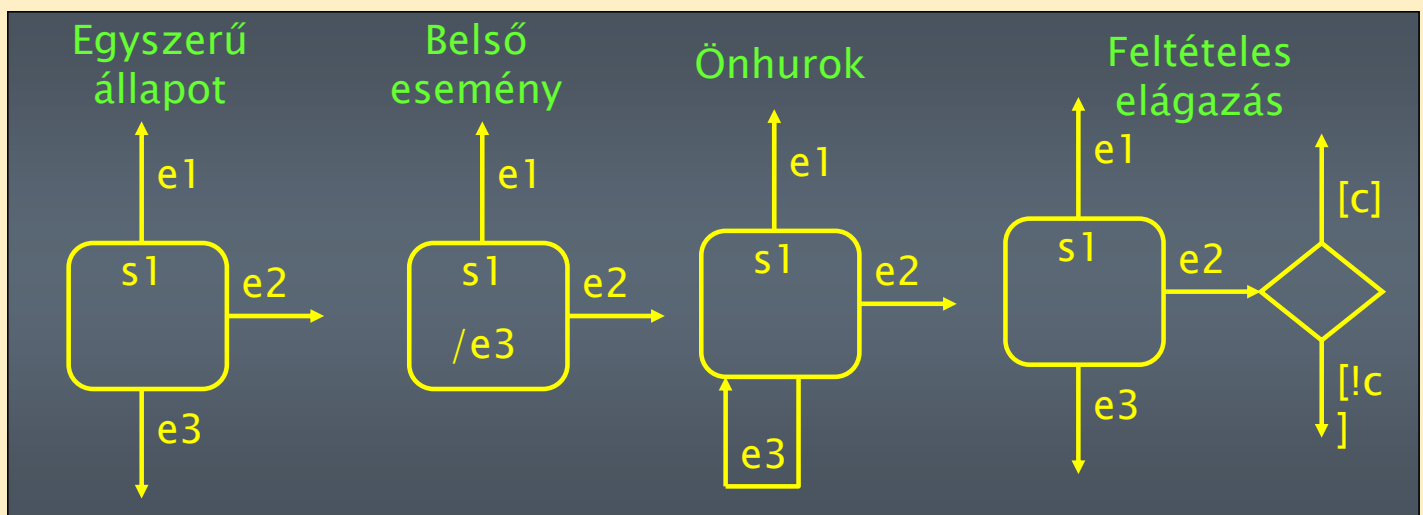
Példa: Nehézségek UML állapottérképek vizsgálata esetén

- Hierarchikus állapot-definíció:
 - (1) Állapot helyett **állapot-konfiguráció** ellenőrzése kell
 - (2) A **prioritásokat** is figyelembe kell venni
 - Teljesség: Minden eseményre, minden állapot-konfigurációban van specifikált viselkedés (tranzíció vagy helybenmaradás)
 - Ellentmondásmentesség: Egy eseményre egy állapot-konfigurációban csak egy tranzíció lehet engedélyezett
- Konkurens régiók: **Konkurens tranzíciók**
 - Akciók determinisztikus végrehajtása szükséges
- Örfeltételek használata: **Kiértékelés**
 - Teljesség: Bármely állapot-konfigurációban egy esemény által triggerelt tranzíciók örfeltételeinek VAGY kapcsolata igaz értéket ad (tautológia)
 - Ellentmondásmentesség: Bármely állapot-konfigurációban egy esemény által triggerelt tranzíciók örfeltételei közül csak egy lehet igaz

50

Teljesség

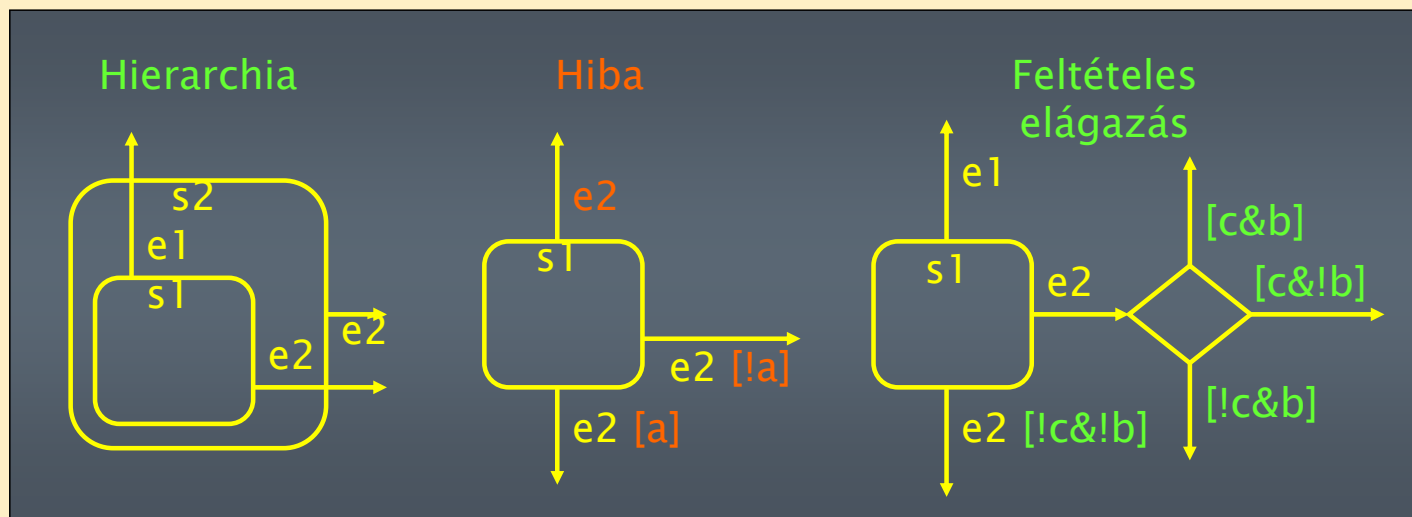
- Minden állapotkonfigurációból, minden eseményre vonatkozóan, minden örfeltétel-kiértékelés esetén kell lennie definiált tranzíciónak



51

Egyértelműség I.

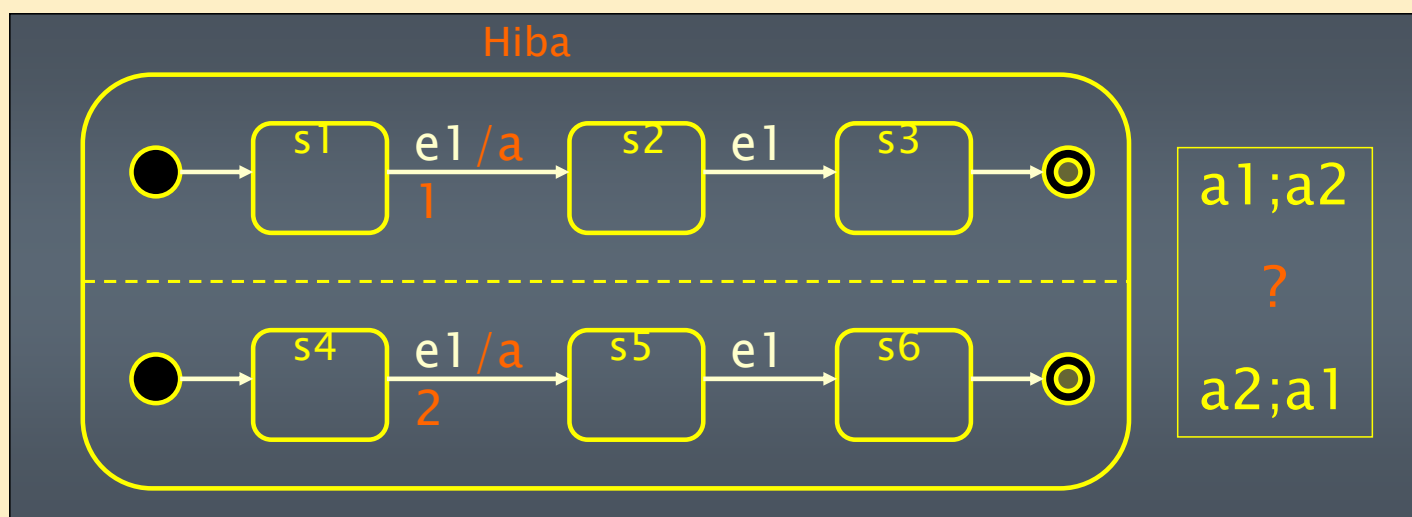
- Minden állapotkonfiguráció és minden esemény esetében az összes őrfeltétel-kiértékelés mellett egy hierarchia szinten belül egy időben csak egy tranzíció lehet aktív



32

Egyértelműség II.

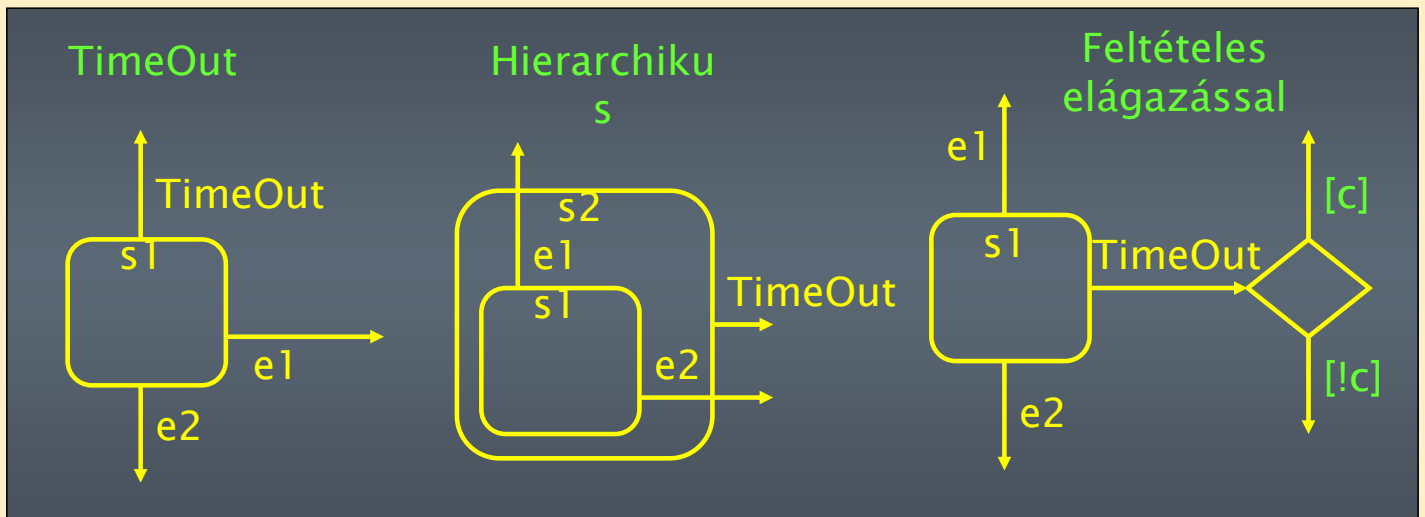
- Konkurens állapotgépeken belül egyazon eseményre csak az egyik gépben lehet akció definiálva



33

Időtűllépés

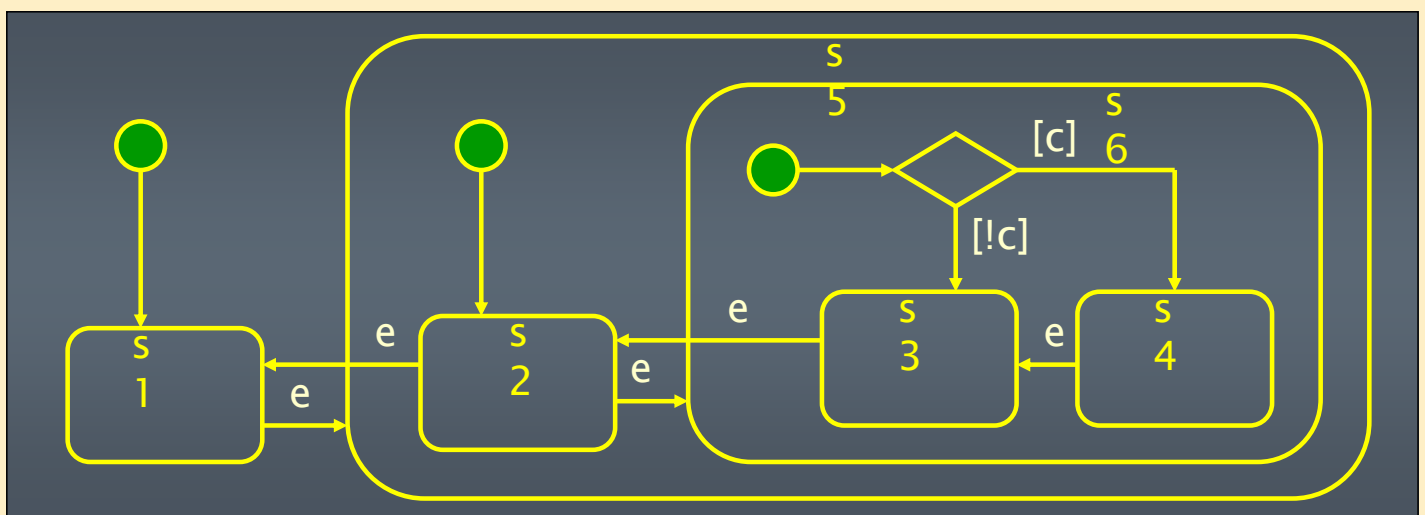
- Minden állapotkonfigurációra vonatkozóan definiálva kell lennie olyan tranzíciónak, mely a Timeout nevű eseményre van triggerelve (lehet örökölt tranzíció is)



34

Indulási állapot I.

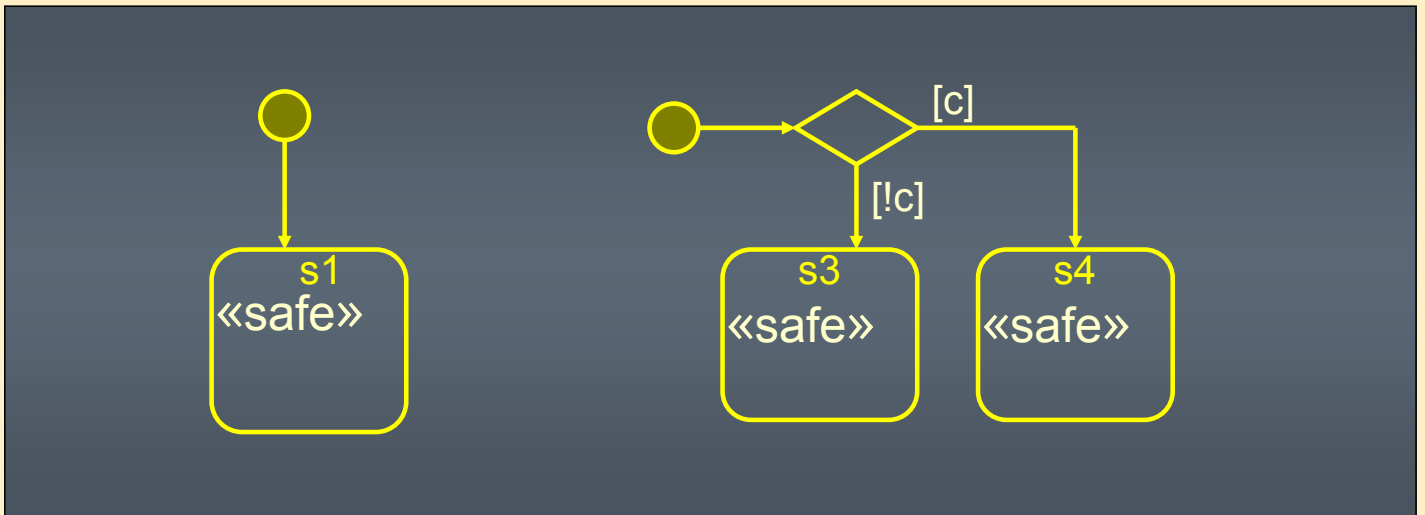
- Minden rész-automatában szerepelnie kell Start állapotnak, beleértve a legfelső szintű régiót is



35

Indulási állapot II.

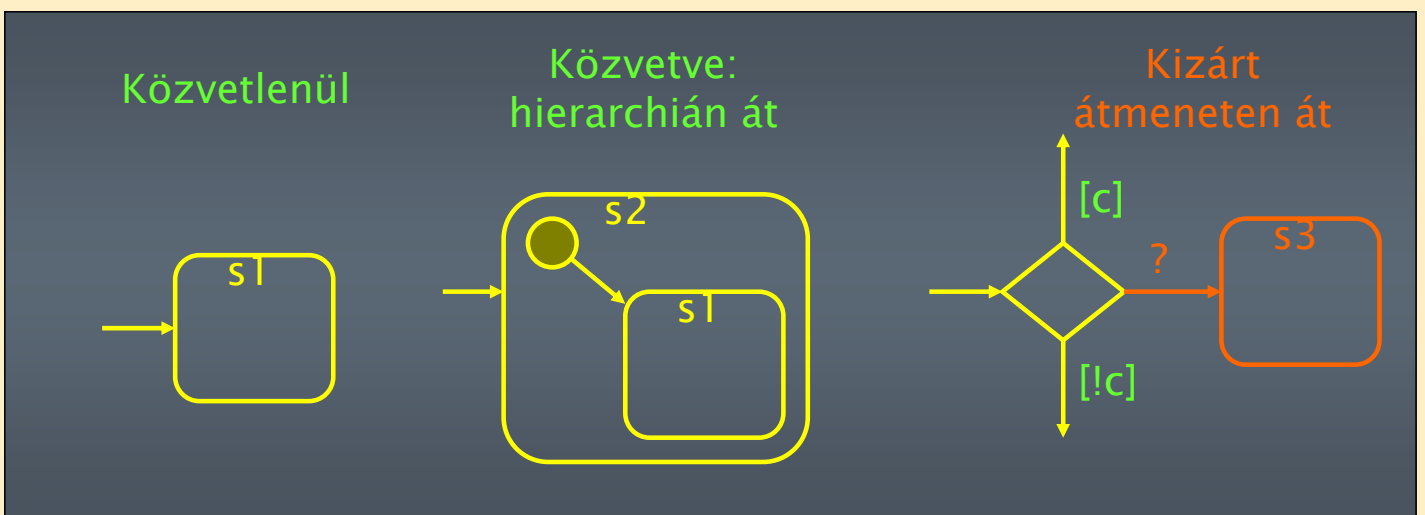
- A legfelső szintű régió induló állapotának biztonságosnak kell lennie: a Start-ból közvetlenül elérhető állapotoknak «safe» sztereotípiával jelöltnek kell lennie



30

Elérhetőség

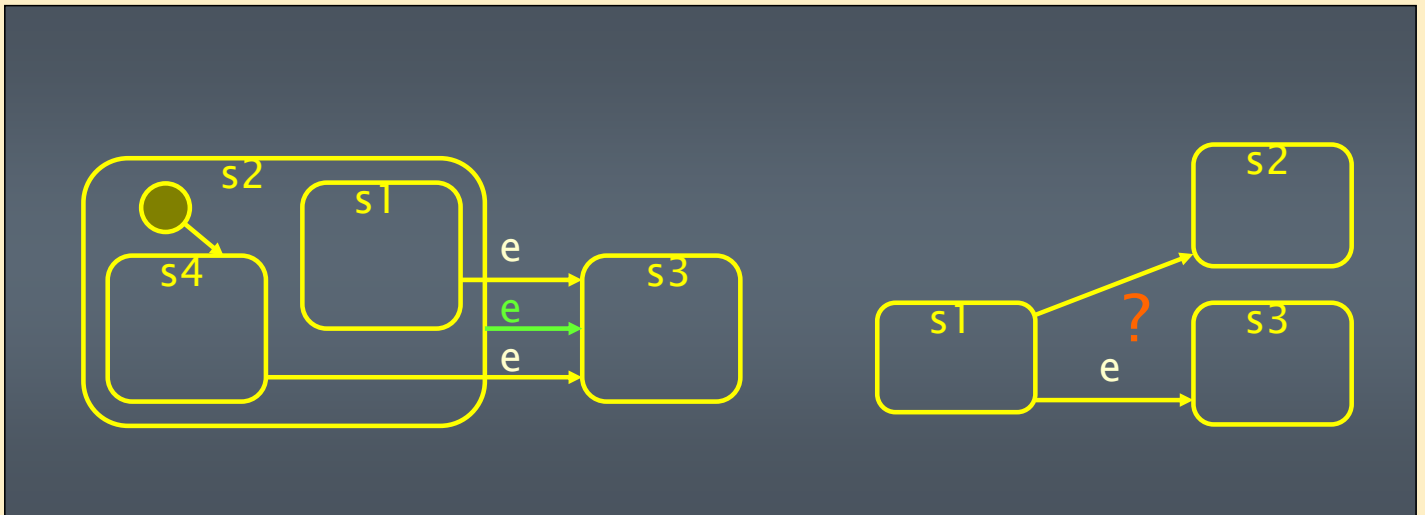
- A rendszer minden egyszerű állapotának elérhetőnek kell lennie vagy közvetlenül, vagy közvetve



31

Takarás, completion

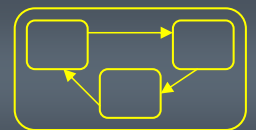
- A hierarchia miatt tranzíciók takartak lehetnek
- Completion és eseménnyel triggerelt tranzíció nem keverhető



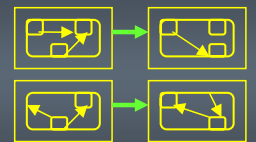
58

Az ellenőrzés menete

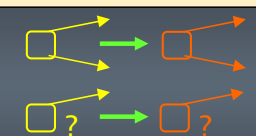
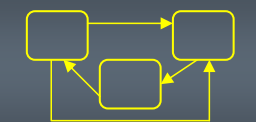
Eredeti modell



Transzformáció-sorozat



1. Események kigyűjtése
2. Átmeneti állapotok megszüntetése
3. Párhuzamos állapotok összerendelése
4. Hierarchia felbontása
5. Entry/exit áthelyezése
6. Belső akciók konvertálása önhurokká
7. Pseudoállapotok, őrfeltételek konvertálása



Hiba: definiálatlan
Hiba: kértélmű...
Hiba: felesleges...
Hiba: hiányzik...
Hiba: elérhetetlen

59

Miről volt szó?

- Motiváció
 - Miért fontosak a tervezési folyamat ezen szakaszai?
 - Milyen elvárások vannak a specifikációval szemben?
 - Milyen módszerei vannak a specifikáció készítésnek?
- A követelménykezelés általános feladatai
 - Követelmények nyilvántartása
 - Követhetőség a verifikációhoz
- Félformális specifikáció
 - Specifikus technikák: SysML
- A specifikáció verifikációja
 - Általános kritériumok
 - Specifikus kritériumok UML állapottérképekre (mintapélda)

60

Hogyan dokumentálható az ellenőrzés eredménye?

- Szoftverkövetelmények igazolójelentése
 - Megvalósítás dokumentálása
 - Felülvizsgálat
 - Egyenrangú átvizsgálás
 - Vizsgálati szempontok szerinti eredmények dokumentálása
 - Ellenőrző lista
 - Követhetőségi vizsgálatok
 - Statikus analízis
 - Formális verifikáció
 - ...
 - Összefoglaló vélemény
 - Minőségi értékelés
 - Szükséges javítások előírása

61