

Dr. Mileff Péter

Integrált szoftver rendszerek és minőségbiztosításuk

**Bevezetés
Miskolci Egyetem
Általános Informatikai Tanszék**

Tantárgyi információk

- A tantárgy főleg elméleti jellegű.
 - 3 előadás + 1 gyakorlat
 - Előadásra járni nem kötelező, de ajánlott.
 - Nincs vezetett gyakorlat.
- Követelmények:
 - Előzetes egyeztetés lapján:
 - Előadás valamilyen tetszőleges, de a tárgyhoz kapcsolódó témakörből.
 - Kiadott féléves feladat sikeres teljesítése.
 - Zárthelyi nincs!
 - A tárgy vizsgával zárul:
 - Írásbeli és szóbeli

2

Miről is lesz szó?

- A tárgy alapvetően az **integrált rendszerekkel** és azok **minőségbiztosításával** foglalkozik.
 - Szoftverek, mint bonyolult rendszerek
 - Szolgáltatásorientált architektúra áttekintése
 - Szoftver minőségmenedzsment
 - Minőségbiztosítás
 - Minőség tervezés
 - Minőség ellenőrzés
 - Folyamat minősége és annak javítása
 - Szoftver metrikák
 - Projektmenedzsment alapvető kérdései feladati

3

A SZOFTVER, MINT BONYOLULT RENDSZER...

4

Szoftverek, mint bonyolult rendszerek

- **Mai szoftverek:**
 - komplex rendszernek tekinthetők,
 - modellezhetők rendszerelméleti fogalmakkal,
 - interfészekkel összekapcsolt rendszerekből, alrendszerekből, modulokból, és kisebb elemekből és végül komponensekből áll.
- **Napjainkban:** több különböző önmagában is bonyolult szoftvert integrálunk és kapcsolunk össze valamilyen globális cél érdekében.
 - Így új szolgáltatásokat nyújtó szoftverrendszer alakul ki.
- **Kialakulnak az Integrált szoftverrendszerek:**
 - napjainkban egyre inkább előtérbe kerülnek,
 - komplex szolgáltatásokat nyújtanak.

5

Integrált szoftverrendszerek

- **Részei lehetnek:**
 - már létező, a piacon rendelkezésre álló szoftverek,
 - akár saját tervezésű és fejlesztésű rendszerek is.
- Saját fejlesztés esetén:
 - az integráltságnak ebben az esetben is megfigyelhetőnek kell lennie!
- **Integráltság:**
 - Jól elkülöníthetők (funkcionálisan és logikailag) a különböző feladatokat ellátó alrendszerek és közöttük jól definiált interfészek biztosítják a kapcsolatot.

6

Példa integrált rendszerre

Online mobiltelefon feltöltő weboldal

- Két fontos nagy rendszer kerül integrálásra:
 - a banki fizetés és a mobilszolgáltató rendszere
- **Cél:** a vásárló online tudja készülékét feltölteni.
- Az integráló szerepet a megvalósító cég rendszere tölti be,
 - saját fejlesztésű szoftverével és adatbázismodelljével.

7

SZOCIOTECHNIKAI RENDSZEREK...

8

Szociotechnikai rendszerek

A szoftvereket tartalmazó rendszereket két kategóriájára:

- **Technikai számítógép-alapú rendszerek:**
 - hardver- és szoftverkomponensekből állnak, de nem eljárásokból és folyamatokból.
 - Jó példa ezekre a televízió, a mobiltelefon és a legtöbb személyi számítógépes szoftver.

9

Szociotechnikai rendszerek

- **Szociotechnikai rendszerek:**
 - tartalmaznak egy vagy több technikai rendszert,
 - tartalmaznak egy tudáshalmazt is arról, hogyan szolgálják ők egy tágabb cél elérését.
- Mit jelent mindez?
 - rendszerek jól definiált működési folyamattal rendelkeznek:
 - a működtető emberek, mint a rendszer részei,
 - a szervezeti szokások, szabályok, külső megszorítások, melyek a működést befolyásolják.

10

Szociotechnikai rendszerek jellemzői

- **Eredendő tulajdonságokkal rendelkeznek:**
 - melyek inkább a teljes rendszerre vonatkoznak, mintsem annak különálló részeire,
 - és nagyban függenek mind a rendszerkomponensektől, mind a köztük lévő kapcsolatoktól.
- **Gyakran nemdeterminisztikusak:**
 - nem biztos, hogy egy adott bemenetre mindig ugyanazt a kimenetet fogják szolgáltatni.
 - A rendszer viselkedése függ az emberi operátortól is, és az emberek nem mindig ugyanúgy reagálnak.

A rendszerek jellegzetessége tehát nem más, mint a rendszerkomponensek tulajdonságainak és viselkedésének keveréke.

11

Eredendő rendszertulajdonságok

- **Egy rendszer több mint a részek pusztá együttese:**
 - Erre utalnak a rendszer komponensei közötti komplex kapcsolatok.
 - Olyan tulajdonságokkal is rendelkezik, amely csak a rendszer egészére jellemző.
- **Elnevezésük: eredendő rendszertulajdonságok**
- **Jellemzőik:**
 - Nem köthetjük egyik különálló rendszerrészhez sem.
 - Elsősorban akkor jelennek meg, amikor a rendszer komponenseit integráljuk.
 - Néha származtatható valamelyik alrendszer tulajdonságából.

12

Gyakori eredendő rendszertulajdonságok

- **Kiterjedés:** egy rendszer kiterjedése a komponensek összeállításától és összekapcsolásától függ.
- **Megbízhatóság:** a rendszer megbízhatósága a komponensek megbízhatóságától függ,
 - de nem várt kölcsönhatások új típusú hibákat eredményezhetnek.
- **Biztonság:** a rendszer biztonsága olyan összetett tulajdonság, amit nehéz mérni.
 - A támadások ott következhetnek be, ahol a rendszertervező nem várja őket, és így a beépített védelem csődöt mondhat.

13

Gyakori eredendő rendszertulajdonságok

- **Javíthatóság:** azt tükrözi, milyen könnyen lehet behatárolni egy felfedezett problémát.
 - Függ: probléma észlelésétől, hibázó komponens azonosításától, a komponens lecserélésétől vagy módosításától.
- **Használhatóság:** ez a tulajdonság arra vonatkozik, hogy milyen könnyű használni a rendszert.
 - Függ a technikai rendszerkomponensektől, az operátoroktól és a működési környezettől.

14

RENDSZERINTEGRÁCIÓ...

15

Rendszerintegráció

- **Az egymástól függetlenül fejlesztett alrendszerek összeillesztése egy teljes rendszerré.**
- **Megvalósítása:**
 - a „nagy-bumm” megközelítési elv szerint:
 - minden alrendszert egy adott időben integrálunk. Nem szerencsés.
 - Tervezéssel, mint a szoftverfejlesztés minden fázisát.
 - Sokkal célravezetőbb!
- **Célszerű megoldás:**
 - Technikai és vezetési okokból az inkrementális megközelítési elv sokkal hasznosabb:
 - az alrendszereket egyesével integráljuk a rendszerbe.

16

Rendszerintegráció

- **Az inkrementális megközelítés érvei:**
 - Szinte lehetetlen úgy ütemezni a különböző alrendszerek fejlesztését, azok ugyanarra az időpillanatra rendelkezésre álljanak.
 - Az inkrementális integráció csökkenti a hibák behatárolásának költségeit:
 - Ha egy időben több alrendszert integrálunk a rendszerbe, akkor a fellépő hiba ezek közül bármelyikben előfordulhat.
 - Egyszerre egy alrendszert integrálva a hiba valószínűleg az újonnan integrált alrendszerben keresendő.
- **Egy komponens integrálására után a rendszer tesztelésére kerül a sor**
 - Ezt a folyamatot **integrációs tesztelésnek** nevezzük.
 - Ellenőrzi, hogy a komponensek valóban képesek-e együttműködni:
 - megfelelően vannak-e meghívva és interfészeiken keresztül a megfelelő adatokat a megfelelő időpontban küldik-e át.

17

INTEGRÁLT RENDSZEREK ARCHITEKTÚRÁI...

18

Integrált rendszerek architektúrái

- Nincs megkötés az architektúrára!
 - Tetszőleges, akár egyedi megközelítések is alkalmazhatók.
 - Javaslatok vannak, és a célszerűséget is érdemes figyelembe venni.
- **Kialakítás szempontjai:**
 - Fontos szempont a továbbfejleszhetőség és a karbantarthatóság.
 - Megfelelő programozási nyelv választása
 - rugalmasan és hatékonyan valósítható meg az integráció, vagy az alrendszerek kidolgozása.
 - Szabványos, vagy „kvázi” szabványos kommunikációs interfész kialakítása.
 - megkönnyíti a későbbi karbantartást, vagy a fejlesztést.

19

Integrált rendszerek architektúrái

- **Kialakítás szempontjai:**
 - Megfelelő eszközöket alkalmazása a fejlesztéshez és az integrációhoz
 - Legyenek jól ismertek, és napjainkban élő fejlesztések.
 - Pl.: Eclipse, Maven, Ant, Icefaces, Apache Tomcat, stb.
- **Hibás elgondolás példa:**
 - nem okos dolog egy régi DOS-os programot alkalmazni egy olyan architektúra kialakítására, ahol internetes szolgáltatások kapják a fő szerepet.

20

Népszerűbb architektúrák

- Kliens-szerver arch.
- Rétegzett / Hierarchikus rendszerek
- P2P alapú osztott rendszer
- Szolgáltatás orientált architektúra
 - (SOA – Service Oriented Architecture)
 - egyre inkább előtérbe kerül,
 - több mai rendszer ezen az elven, ilyen alapú eszközöket használ a szolgáltatásainak közzé tételére
- Pl.: A mobilszolgáltató rendszerek.

21

Köszönöm a figyelmet!

22