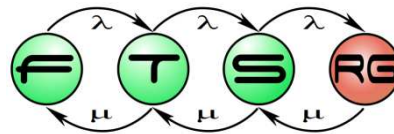


## Szolgáltatásorientált architektúra

Gönczy László  
gonczy@mit.bme.hu



# Mi a SOA?

- Service Oriented Architecture
  - Szolgáltatásorientált architektúra
- Rendszertervezési, integrációs paradigma
- Alapelv: a rendszer által nyújtott szolgáltatást nézzük
  - Ne a megvalósítás technikai részleteit
- A funkcionalitás leírására koncentrálnak
- Modulok: cserélhető elemek
- Heterogén rendszerek integrációját teszi lehetővé
- Cél: üzleti folyamatok megvalósítása

# IT architektúrák fejlődése



**Növekvő modularitás – nagyobb rugalmasság**

Várkonyi László (IBM Hungary):

ITCAM for SOA, Számítógépes infrastruktúra rendszertervezése, 2007.

# A rendszerintegráció szintjei

- Egyedi integráció
  - Pont-pont kapcsolatok
  - Költséges
  - Rugalmatlan
- Központi integráció
  - Egy rendszerbe csatoljuk be az alrendszereket
  - Sok pont-pont kapcsolat egységes keretben
- SOA
  - Rugalmas kapcsolat
  - Szabványos felületek, kommunikáció
  - „szolgáltatás sín” (ESB)

# A „Web” régi modellje

- Dokumentumok összessége
- Kliensek: böngészők
- Kézzel indított műveletek
- HTTP átvitel célja: megjelenítés

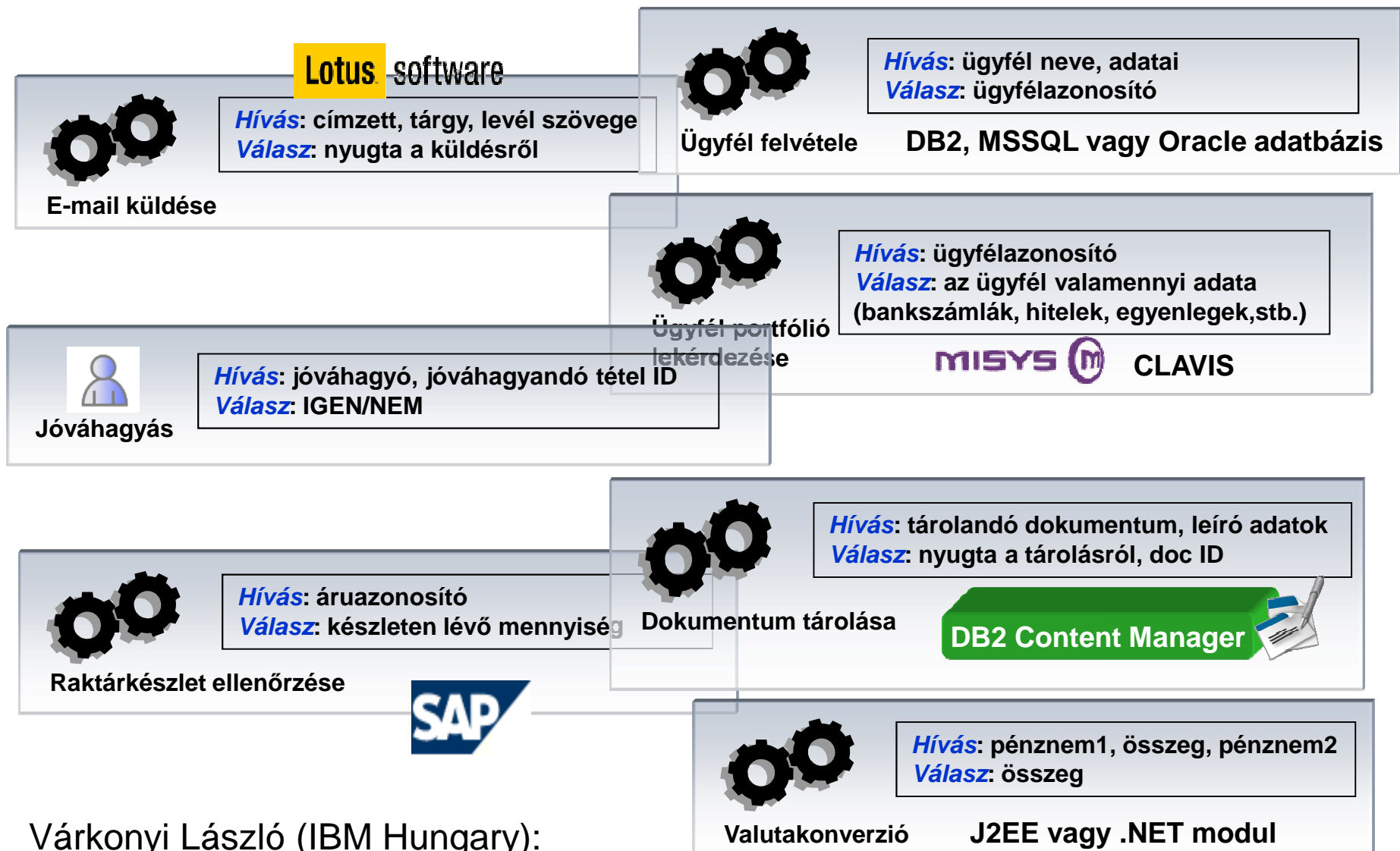
# Új paradigma

- A Web egy elosztott számítógép
  - „Global computer”
- Automatikusan indított műveletek
  - Tranzakciókezelés
- Intelligens ügynökök
- Dinamikus rendszerek
- Cél: nem rendszerek, hanem szolgáltatások integrációja
- XML alapú szabványok
  - Szolgáltatások, adatsémák, konfiguráció, folyamatleírás, stb.
- (Web 2.0?)

# Szolgáltatás fogalma

- Önleíró
- Önhordó
- Moduláris
- Kombinálható
  - Pl. termelési lánc, outsourcing
- Szabvány: XML alapúak
- Szolgáltatás lehet pl.
  - Egyszerű Java osztály
  - Nagyvállalati mainframe
  - SAP alrendszer
  - Adatbázis tárolt eljárás
  - Üzenetküldés
  - Szenzorhálózat mérési eredménye

# Példák a szolgáltatásokra



Várkonyi László (IBM Hungary):

ITCAM for SOA, Számítógépes infrastruktúra rendszertervezése, 2007.



# Üzleti/logikai modell

- Top-down megvalósítás
  - Funkció → megvalósítás
  - Logikai szolgáltatás → konkrét alkalmazás
  - Műveletek: pl. számlázás, egyenleg lekérdezés, keresés
  - Alkalmazási területek: pl. banki rendszerek, biztosítók... ma már minden szektorban
  - Üzleti motiváció: outsourcing
- Technológiai szemléletváltás
  - Alapvetően meglévő technológiákra építve

# Alkalmazási területek

- Logisztikai rendszerek
  - Biztosítók, bankok
  - Állami szektor
    - Pl. EU szintű személyi igazolvány nyilvántartás
  - Beszállítói lánc
  - e-Health
- iparági szabványok betartása  
(Az interfész (WSDL) sok esetben szabványos)

# A szolgáltatás két oldala

- A publikus oldal
  - nyilvánosan hozzáférhető
  - a vevő számára ez az érdekes
  - Technológiafüggetlen
  - az állandósága érték
- A privát oldal
  - üzleti titok
  - a vevőt nem érdekli
  - technológiafüggő
  - rugalmasan fejleszhető
- Mint OO rendszereknél az enkapszuláció

# Példa: MNB árfolyam

- <http://www.mnb.hu/arfolyamok.asmx?WSDL>
- Metódusok
  - string GetInfo(), string GetCurrentExchangeRates()
  - string GetExchangeRates(string startDate, string endDate, string currencyNames)
- Lehetséges hibák definiálása
- További elérhető webszolgáltatások
  - [http://www.xmethods.net/ve2/Directory.po;jsessionid=pbAWv12wFOSCrkWYZ0\\_NvNN4](http://www.xmethods.net/ve2/Directory.po;jsessionid=pbAWv12wFOSCrkWYZ0_NvNN4)
  - <https://www.thedacs.com/databases/url/key/5440/5443/5537>

# SOA szerepe - IT

- Integrációs technológia
- Régi technológiák felhasználhatók
  - legacy
- Plusz feladatok
  - Szolgáltatás leírók (WSDL, registry)
  - „Szótár” (pl. XSD)
  - Konverziós műveletek
  - Komponensek (folyamatvégrehajtás, kommunikációs/mediációs komponens, ...)
- Plusz technológiák
- Hosszú távú projekteknel rövidebb fejlesztési idő
  - Technológiai „belépési szint”



# SOA szerepe - üzlet

- Innovatív ötleteket könnyebb megvalósítani
- Beszállítókkal könnyebb kommunikálni
  - Pl. Coca Cola, 11.000 ember foglalkozik a beszállítókkal...
- Rövidebb time-to-market
- Kisebb üzemeltetési költségek
- DE

Üzlet-IT kommunikáció nélkül rosszabb, mint bevezetés előtt

Pl. a cégek 40%-a nem méri, mikor térül meg egy SOA projekt! (Gartner)

<http://www.devshed.com/c/a/Web-Services/Introduction-to-Service-Oriented-Architecture-SOA/>

<http://apsblog.burtongroup.com/2009/05/soa-its-dead-jim.html>

# Mi kellett ahhoz, hogy sikeres legyen a SOA?

- Egyszerű integrálhatóság üzleti folyamatokba
  - Pl. BPEL
- Könnyű illesztés tetszőleges platformhoz
  - Pl. Web service kiegészítés COBOLhoz
- Rugalmas bővíthetőség
  - XML szabványok
- Hordozható (adat)leírás
- Megoldatlan kérdés: menedzsment (governance)

# „Szolgáltatás” koncepciók


- XML webszolgáltatások
  - „Távoli eljáráshívás”
  - B2B integráció
  - Felület definíciója
  - Rengeteg kiegészítő szabvány
- REST
  - Állapottal rendelkező erőforrások elérése HTTP felett
  - A’la URL, RSS....
- OSGI
  - Szolgáltatásokat nyújtó komponensek (bundle) integrációja
  - Életciklus kezelés
  - Pl. Eclipse

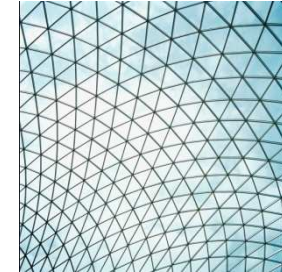


# Új technológiák

- Üzleti szabályrendszerek (business rules)
  - Logika különválasztása
- Komplex eseményfeldolgozó rendszerek (CEP)
  - Nagymennyiségű üzenet szűrése és feldolgozása
- Web 2.0 technológiák
  - Blog, wiki, mashup, social networks...
  - AJAX

# Trendek

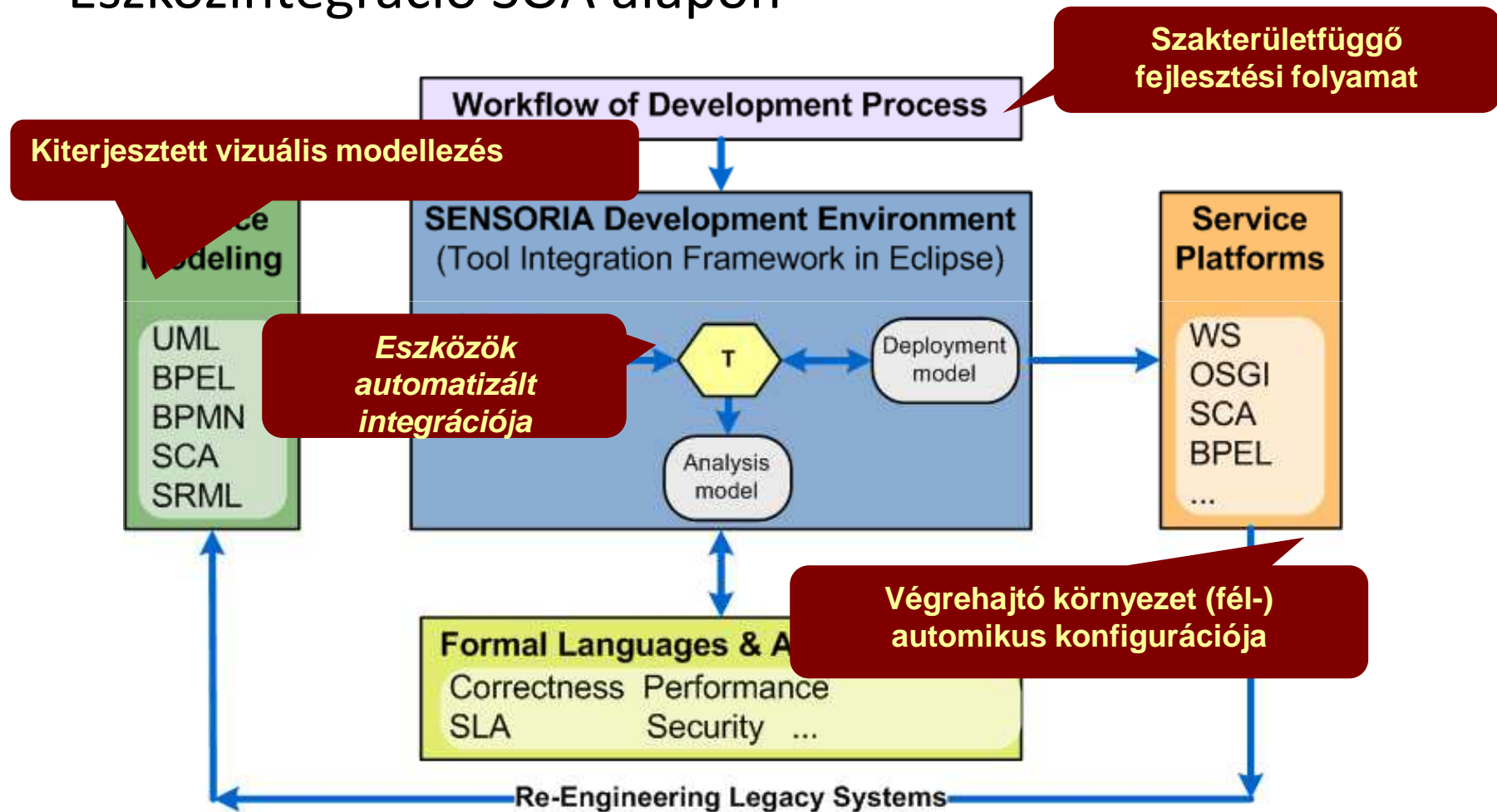
- Grid computing
  - Nagy számítások elosztottan (tudományos célok)
  - „Használjuk ki a szabad számítógépkapacitást”
- Cloud computing 
  - Google, amazon, ..
  - Jobb erőforráskihasználás
- \* as a service
  - Software/Application...
  - Infrastructure..
  - PI. CRM rendszerek
- Virtualizációs technológiák



Gartner: <http://vcap.star-digital.co.uk/?userpath=00000013/00001135/00038972/&page=7>

# SOA rendszerek modell alapú fejlesztése

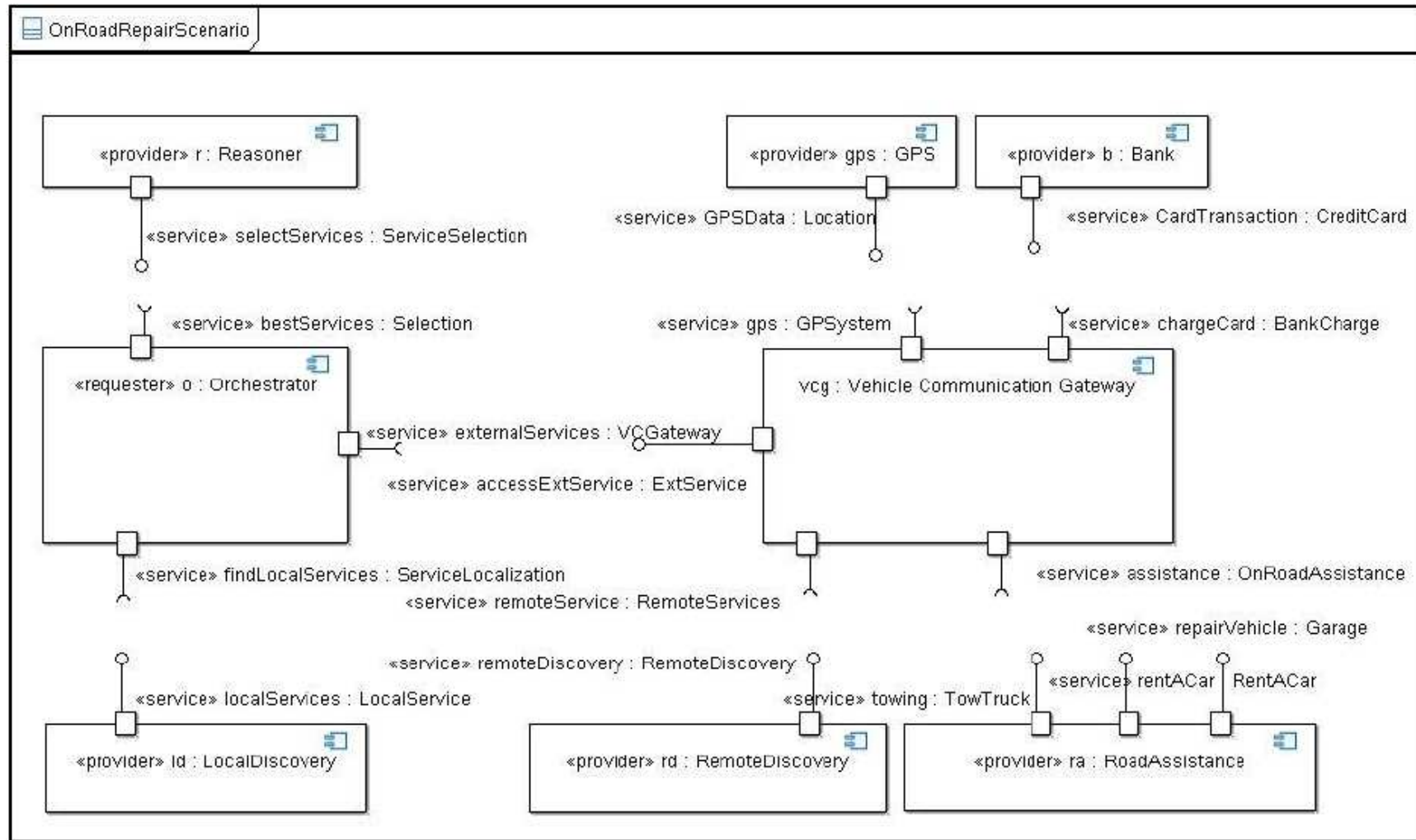
- PI. SENSORIA EU projekt
- Eszközintegráció SOA alapon



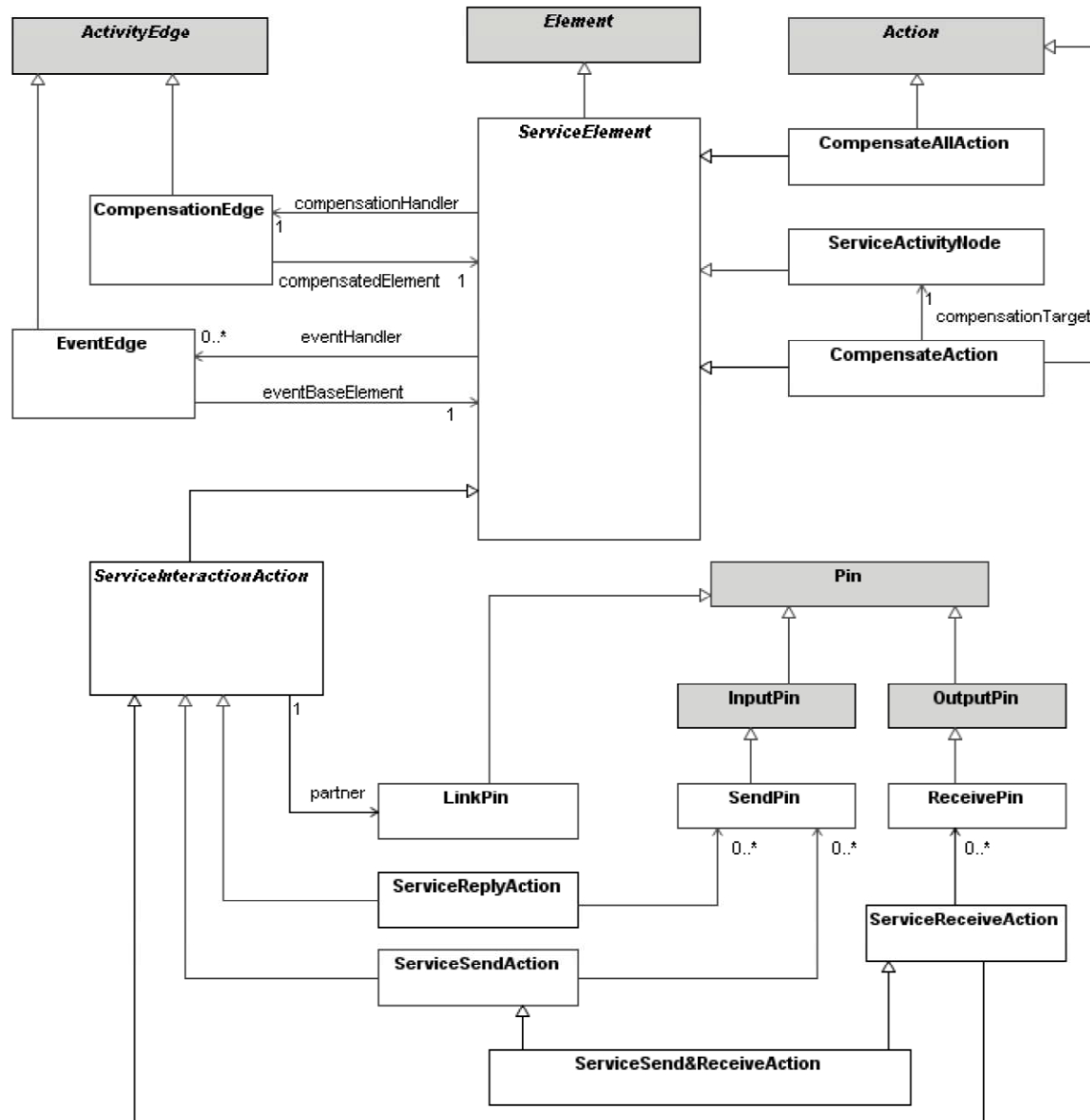
# Szolgáltatások modellezése: UML4SOA

- Hogyan terjesszük ki az UML-t szolgáltatás-orientált rendszerekhez?
- Struktúra
  - Milyen elemek (szolgáltatások) és összeköttetések vannak
- Dinamikus viselkedés
  - Hogyan működik egy összetett szolgáltatás
  - Végrehajtó platformok: ld. későbbi ea.
- „Nemfunkcionális” tulajdonságok (ld. legközelebb)
- Szabványos UML profile
  - Sztereotípiák
  - Felsorolt értékek
  - Grafikus jelölések
- Példa: car2infrastructure

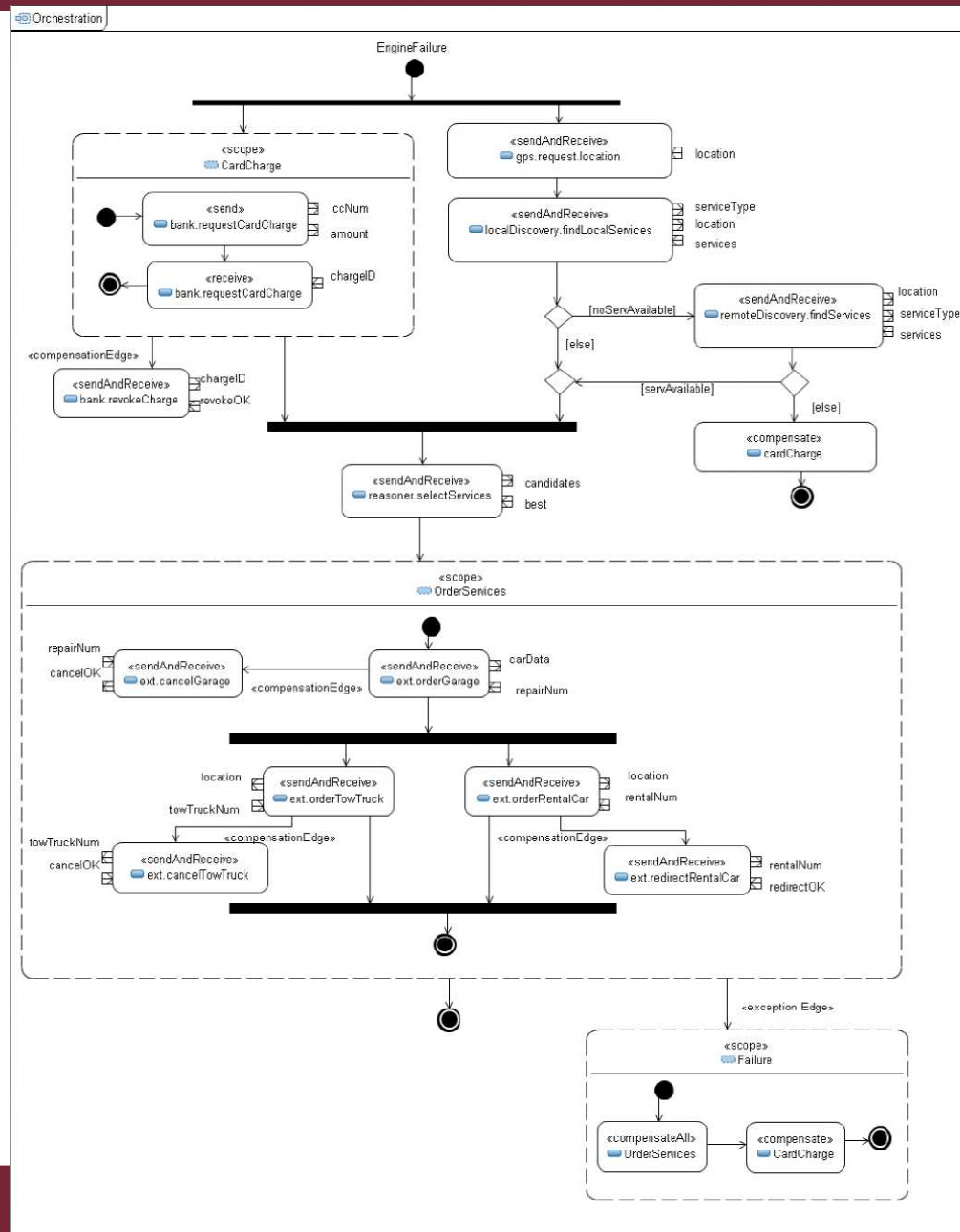
# Strukturális modellek: komponensek



# UML4SOA: viselkedés leírása

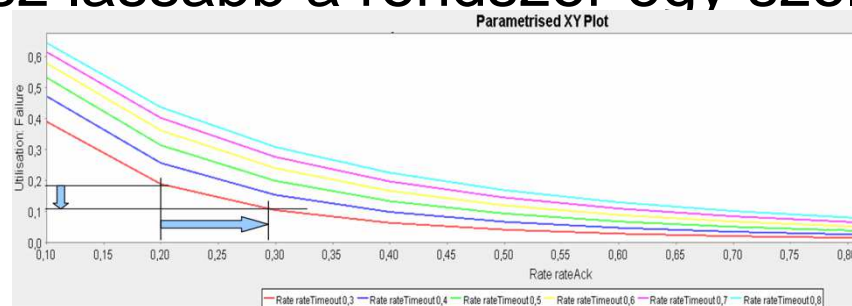


# Dinamikus modellek: viselkedés



# Mire jó egy ilyen modell?

- „Deployment”
  - Kódgenerálás (pl. Apache AXIS)
  - Folyamatok kódjának előállítása (pl. BPEL)
- Modell helyessége ellenőrizhető
  - Nincs „összekötetlen” port
- Modell viselkedése ellenőrizhető
- Rendszer teljesítménye becsülhető
  - Modell kiegészítve mennyiségi paraméterekkel
  - Pl. „mennyivel lesz lassabb a rendszer egy szolgáltató hibája miatt”





# Összefoglalás

- SOA
  - Rendszerintegráció
  - Heterogén környezet
  - Implementációs technológiák
  - Párhuzamos szabványok
  - Folyamatok maguk is szolgáltatások
- Új irányok
  - Pl. webszolgáltatások szenzorhálózatokban
- Kitekintés: SOA + MDA