



Zavod za telekomunikacije

Agentski orijentirano programsko inženjerstvo

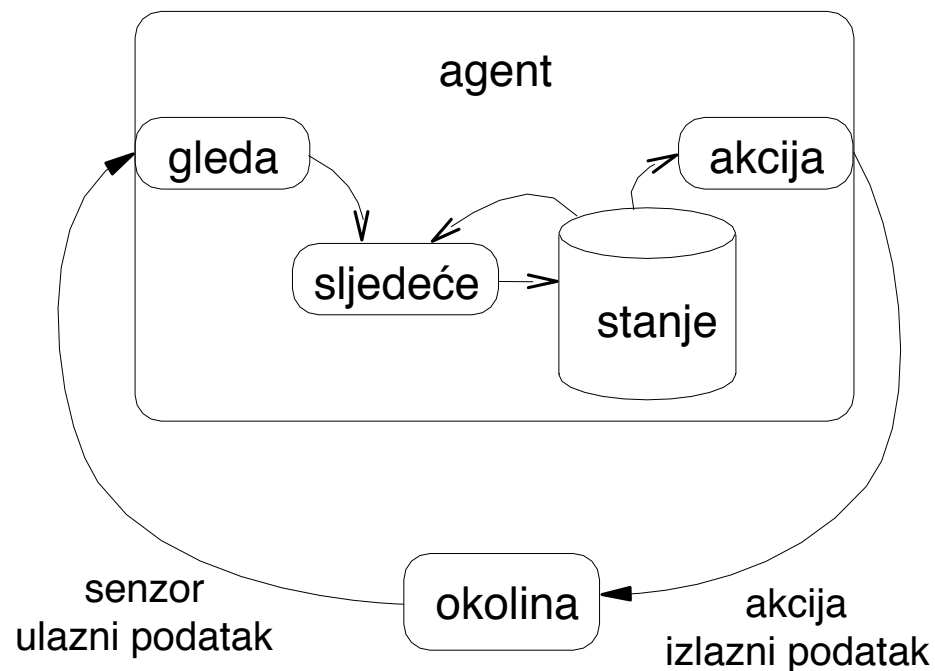
doc. dr. sc. Mario Kušek

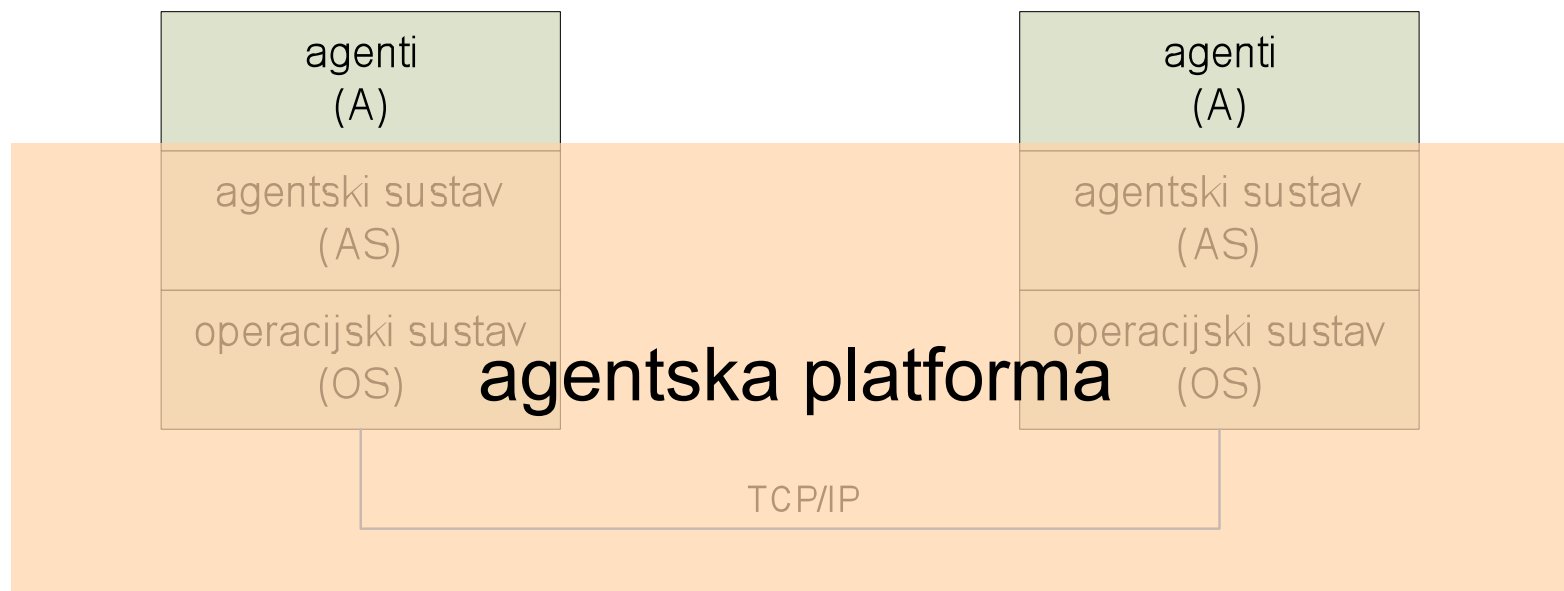
- ◆ Uvod u agente
- ◆ Višeagentski sustavi
- ◆ Pregled metodologija i procesa

- ◆ entitet koji predstavlja korisnika u nekoj okolini (mreža) i autonomno obavlja poslove umjesto korisnika
- ◆ svojstva:
 - **autonomnost**,
 - inteligencija,
 - pokretljivost,
 - reaktivnost,
 - proaktivnost,
 - komunikativnost i
 - društvenost.

Arhitekture agenata

- osnovni dijelovi agenta prema funkcijama koje obavljaju:
 - dio za opažanje (senzor) i
 - akcijski dio
- logički agenti
- reaktivni agenti
- arhitektura uvjerenje-želja-namjera (BDI)
- slojevite arhitekture





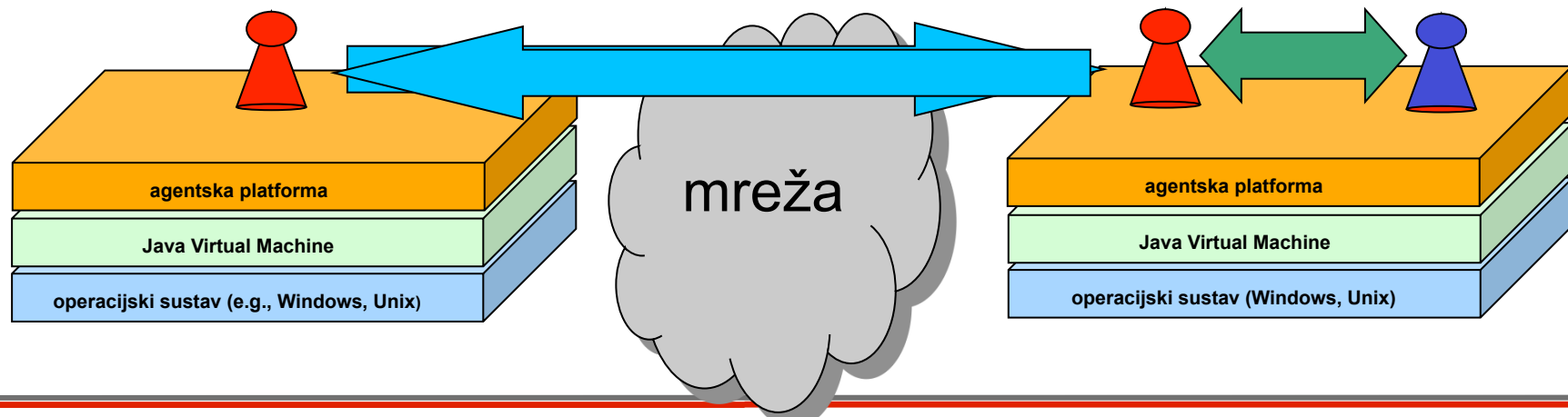
- agentske platforme:
 - Voyager (UK)
 - Grasshopper (DE) – pokretljivost, sigurnost
 - JADE (IT) – standardizacija, pokretljivost, otvoreni kod

Programski model pokretnog agenta



Zavod za telekomunikacije

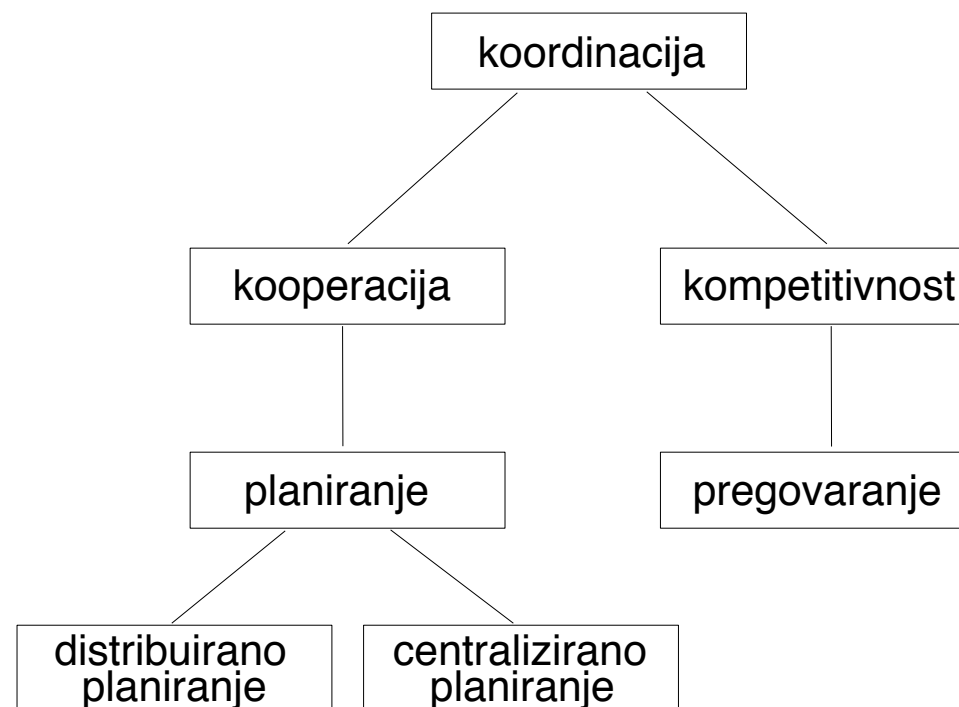
- ◆ programski modeli:
 - klijent/poslužitelj
 - pozivanje udaljene procedure (RPC)
 - kod na zahtjev (java applet)
 - udaljeno programiranje – kao kod na zahtjev
 - pokretni programski agenti



- ◆ sustav u kojem više agenata međudjeluju komunikacijom i interakcijom
- ◆ komunikacija
 - kako agenti razmjenjuju i razumiju poruke
- ◆ interakcija
 - omogućuje razgovor tj. strukturnu razmjenu poruka
- ◆ sustav koji ima toliko agenata da ih se ne može pratiti čine društva agenata koja imaju ugrađenu organizaciju

- zasniva se na:
 - opažanju (primanju poruka) i
 - izvođenju akcija (slanju poruka)
- komunikacija omogućuje koordinaciju akcija i ponašanja agenata

Taksonomija koordinacije



- ◆ Model nadređeni/podređeni
 - Dobar za paralelizaciju poslova
 - Loše je da podređeni agenti ne mogu reagirati u problematičnim situacijama
- ◆ Model tima
 - Agenti u timu zajednički rješavaju probleme
 - Jedan agent može tražiti drugog agenta za pomoć
- ◆ Modeli iz prirode obično samoorganizirajući
 - Mravinjak (mravi) – praćenje traga
 - Jata (ribe, ptice) – relativna udaljenost
 - Pčele – traženje peluda (smjer i udaljenost)
 - ...

- ◆ teorija združenih namjera (*joint intentions theory*)
 - članovi tima djeluju u skladu s ciljem tima
 - prilikom odustajanja od namjere agent mora obavijestiti sve agente u timu
- ◆ teorija dijeljenog plana (*shared plans theory*)
 - članovi tima imaju zajednički plan kojega se pridržavaju
 - komunikacija kod odustajanja je samo jedan način rješavanja problema zajedničkih uvjerenja

Prednosti i nedostaci agenata



Zavod za telekomunikacije

- ◆ prednosti:
 - manje mrežno opterećenje
 - asinkronost – samostalnost
 - manje opterećenje klijenta
 - raspodijeljeno izvođenje – brže vrijeme izvođenja
- ◆ nedostaci:
 - pokretanje agenta troši resurse
 - veće opterećenje poslužitelja
 - standardizacija
 - kompatibilnost platformi
 - sigurnost

- Područja gdje agenti mogu pomoći:
 - Elektroničko poslovanje
 - Osobni asistenti
 - Raspodijeljeno dohvaćanje informacija
 - Usluge u telekomunikacijskim mrežama
 - Aplikacije grupnog rada (tijek posla)
 - Nadzor i obavještanje
 - Dostava informacija
 - Paralelno procesiranje
- Potencijalna područja primjene se dijele na 3 dijela:
 - Osobni agenti (prikupljanje informacija i djelovanje u ime vlasnika)
 - Višeagentski sustavi za odlučivanje (npr. pregovaranje u telekomunikacijskoj mreži)
 - Višeagentski simulacijski sustavi (npr. simulacije ljudskih ili životinjskih zajednica, računalne mreže, igre, upravljanje prometom na cestama)
- Potencijalne industrijske primjene
 - Produkcija, kontrola procesa, telekomunikacijski sustavi, kontrola zračnog prometa, elektroničko poslovanje, prometni i transportni sustavi, prikupljanje i filtriranje informacija, zabava, medicinska njega ...

- ◆ Agent-Oriented Software Engineering (AOSE)
 - nova paradigma u programskom inženjerstvu
 - proširenje objektno-orijentiranog (OO) koncepta
 - agenti – apstrakcije na visokoj razini *aktivnih* entiteta u programskom sustavu
 - autonomni su, bez kontrole izvana (*blackbox control*)
 - visoka razina interakcije - jezik za komunikaciju između agenata
 - cilj:
 - stvoriti metodologije i alate za razvoj, ponovnu iskoristivost i održavanje agentskih sustava
 - unapređivanje koordinacije, komunikacije, suradnje i umjetne agencije među agentima i agentskim sustavima

Pregled agentski orijentiranih metodologija

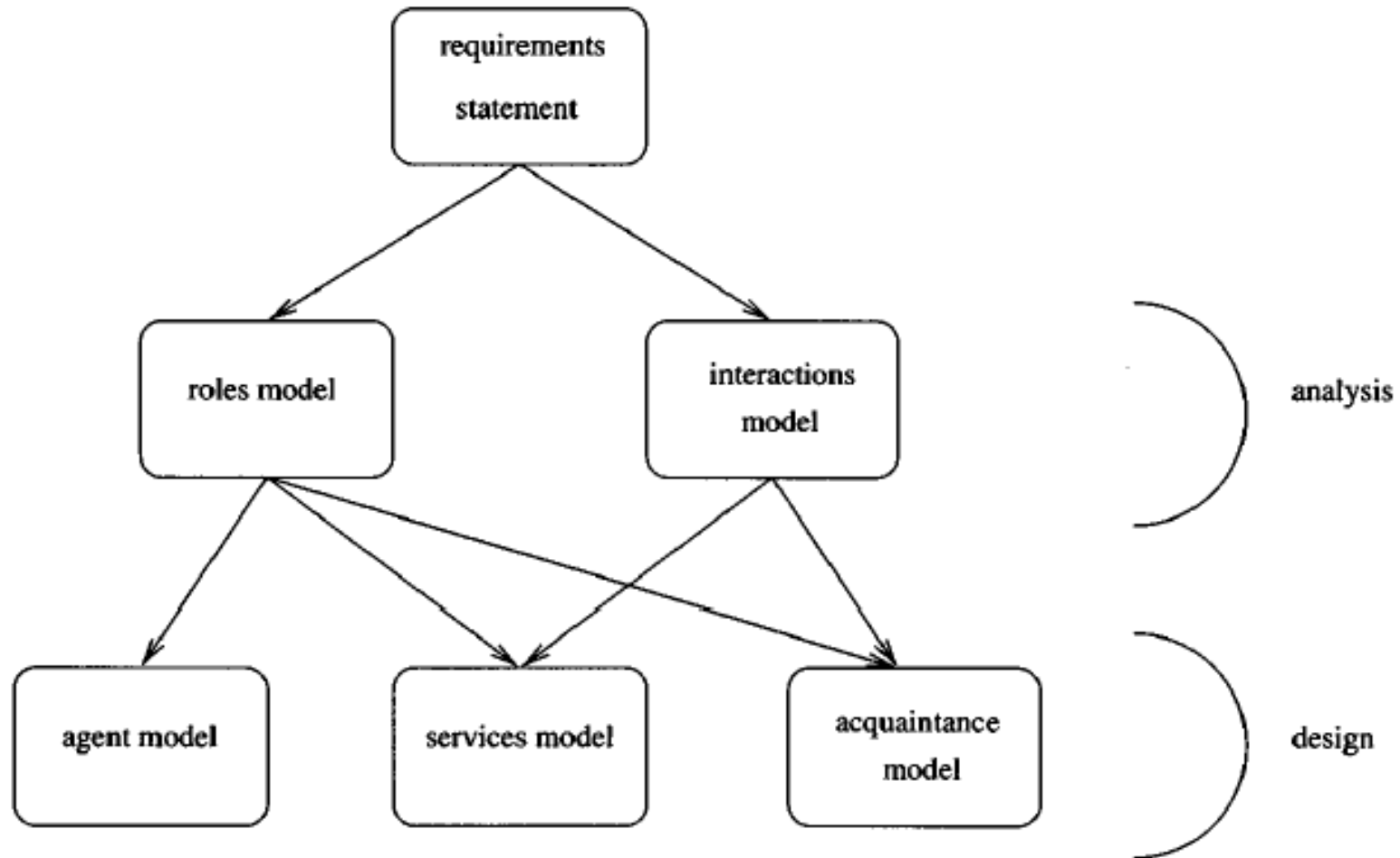


Zavod za telekomunikacije

- ◆ Opće AOSE metodologije
 - Gaia (<http://citeseer.ist.psu.edu/wooldridge00gaia.html>)
 - Tropos (<http://www.troposproject.org/>)
 - MaSE – Multiagent Systems Engineering
(<http://en.afit.af.mil/ai/publications/Conference/aois-99/MaSE-AOIS99.htm>)
 - PASSI (<http://mozart.csai.unipa.it/passi/>)
- ◆ Posebne metodologije
 - ADELFE (<http://www.irit.fr/SMAC/PROJETS/ADELFE/AdelfeMethodology.html>)
 - MESSAGE (<http://www.eurescom.de/~public-webSPACE/P900-series/P907/D1/>)
 - Prometheus (<http://www.cs.rmit.edu.au/agents/Papers/aose02.pdf>)
- ◆ Alati i infrastrukture
 - AUML – jezik za specifikaciju (<http://www.auml.org/>)
 - AML – jezik za specifikaciju
(<http://www.whitestein.com/pages/solutions/meth.html>)
 - FIPA – Compliant Agent Infrastructure (<http://www.fipa.org/>)

- ◆ služi za analizu i dizajn agentski orijentiranih sustava
- ◆ područja primjene:
 - agenti intenzivno koriste procesni kapacitet
 - nije za sustave s konfliktima
 - neovisna o implementaciji i platformi
 - statička organizacija sustava
 - veze između agenata
 - usluge koje pružaju
 - mogućnosti agenata
 - za mali broj različitih agenata (<100)

Gaia – model



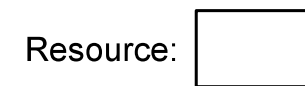
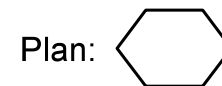
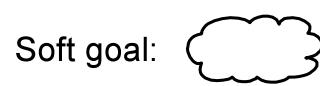
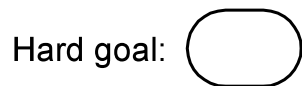
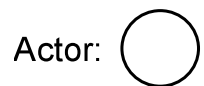
- ◆ zasniva se na konceptima ciljno temeljenim zahtjevima (*goal-based requirements*)
- ◆ fokus je na BDI agentima
- ◆ pokriva:
 - nefunkcionalne zahtjeve
 - organizacijske aspekte MAS-a
 - za ostale aspekte koristi:
 - UML (dijagram klasa, aktivnosti, interakcija) ili
 - AUML (slijedni dijagram)

Tropos

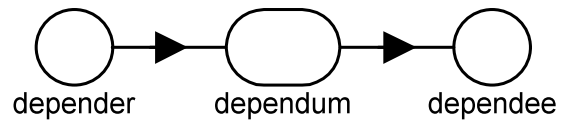


Zavod za telekomunikacije

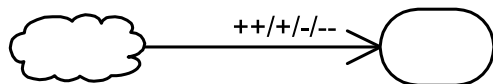
- primjer notacije (dolje)
- dijagrami:
 - aktera (*actor*)
 - ciljeva (*goal*)
 - mogućnosti (*capability*)
 - plana (*plan*)
 - interakcije agenata



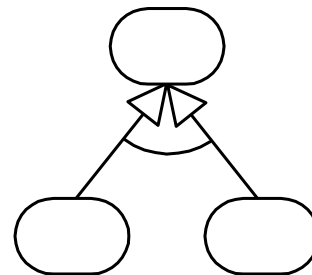
Goal dependency:



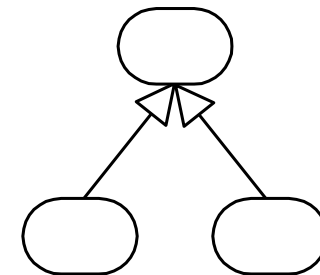
Contribution:



AND decomposition:

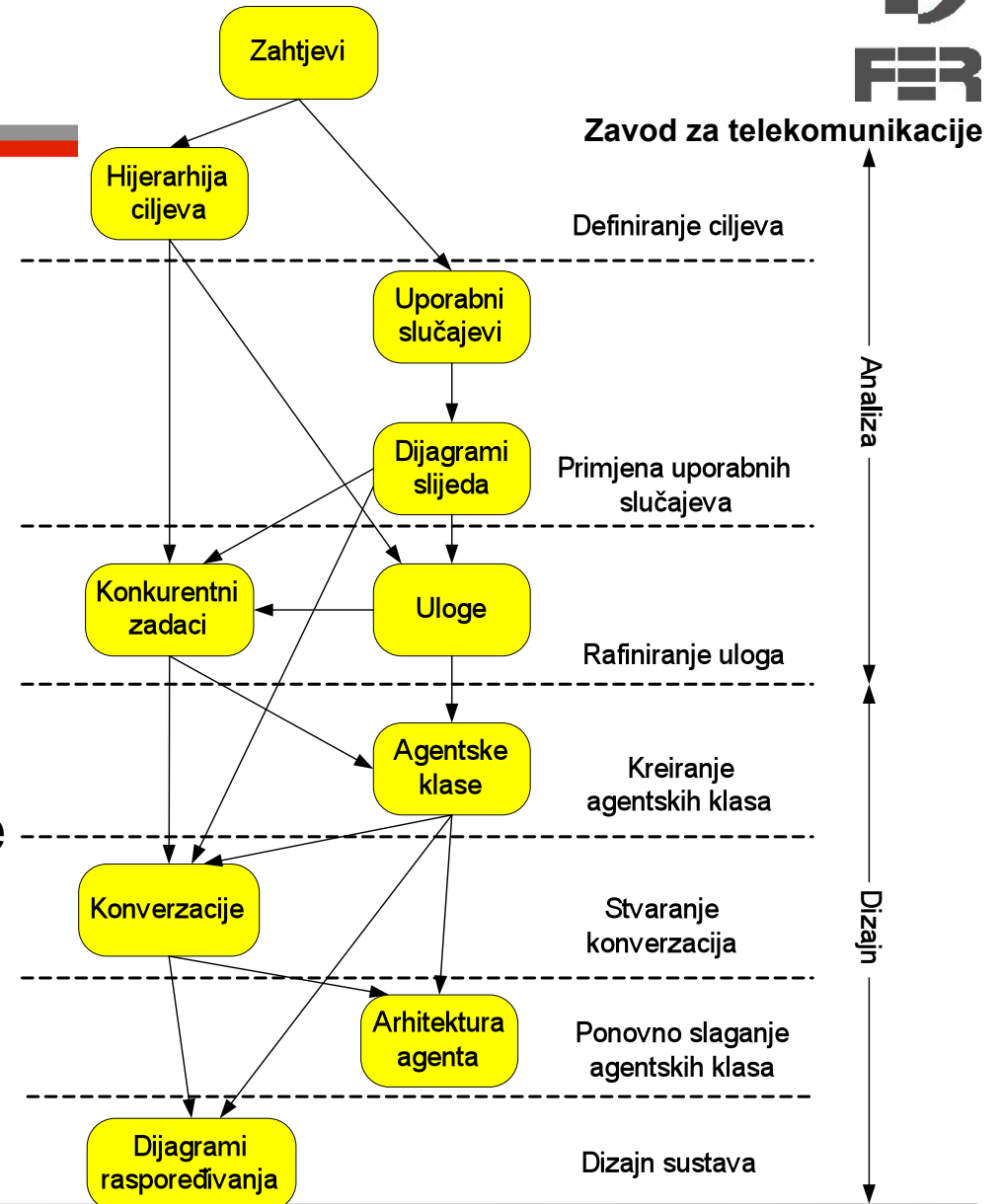


OR decomposition:

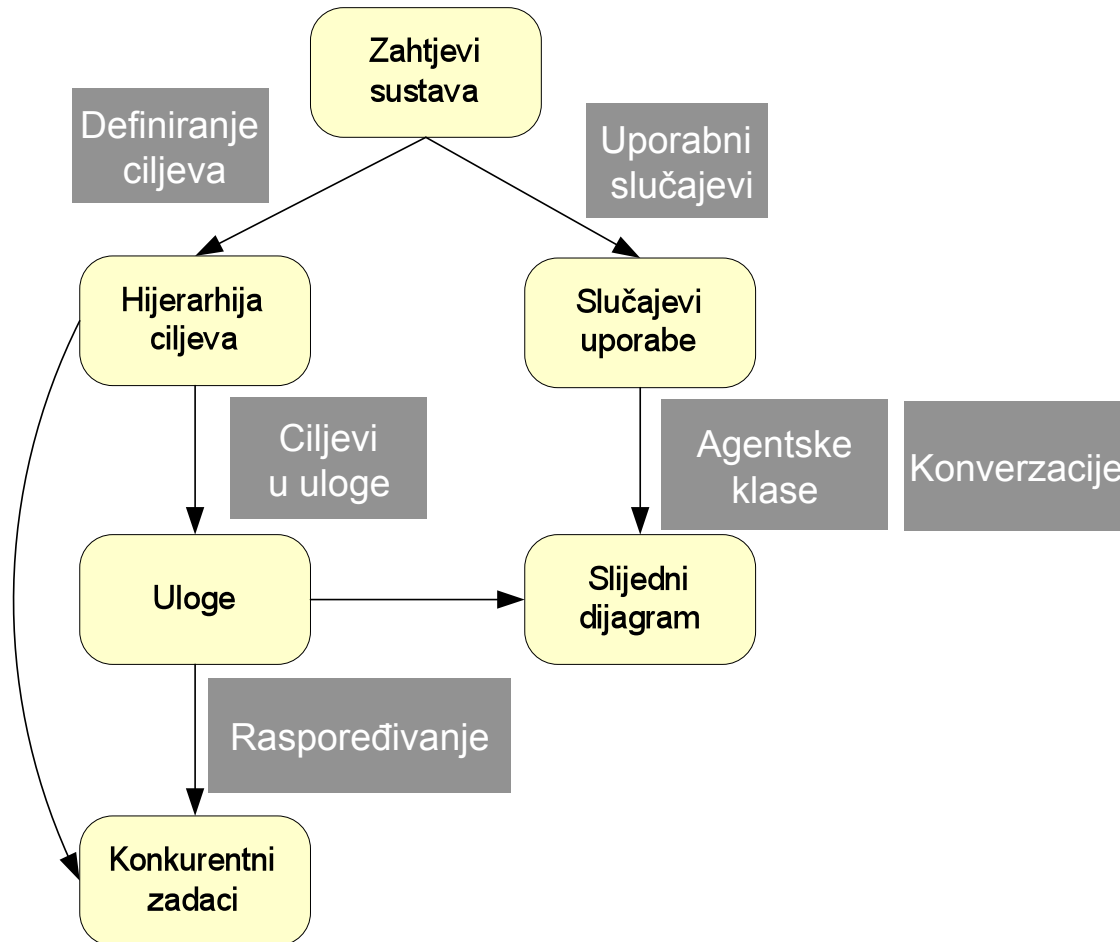


MaSE

- pokriva čitav životni ciklus razvoja i implementacije agentskog sustava
- faze: analiza i dizajn
- grafički modeli se koriste u svakom koraku kao pomoć dizajneru sustava u procesu
- napravljen alat koji potpomaže razvoj



MaSE modeli u fazi analize



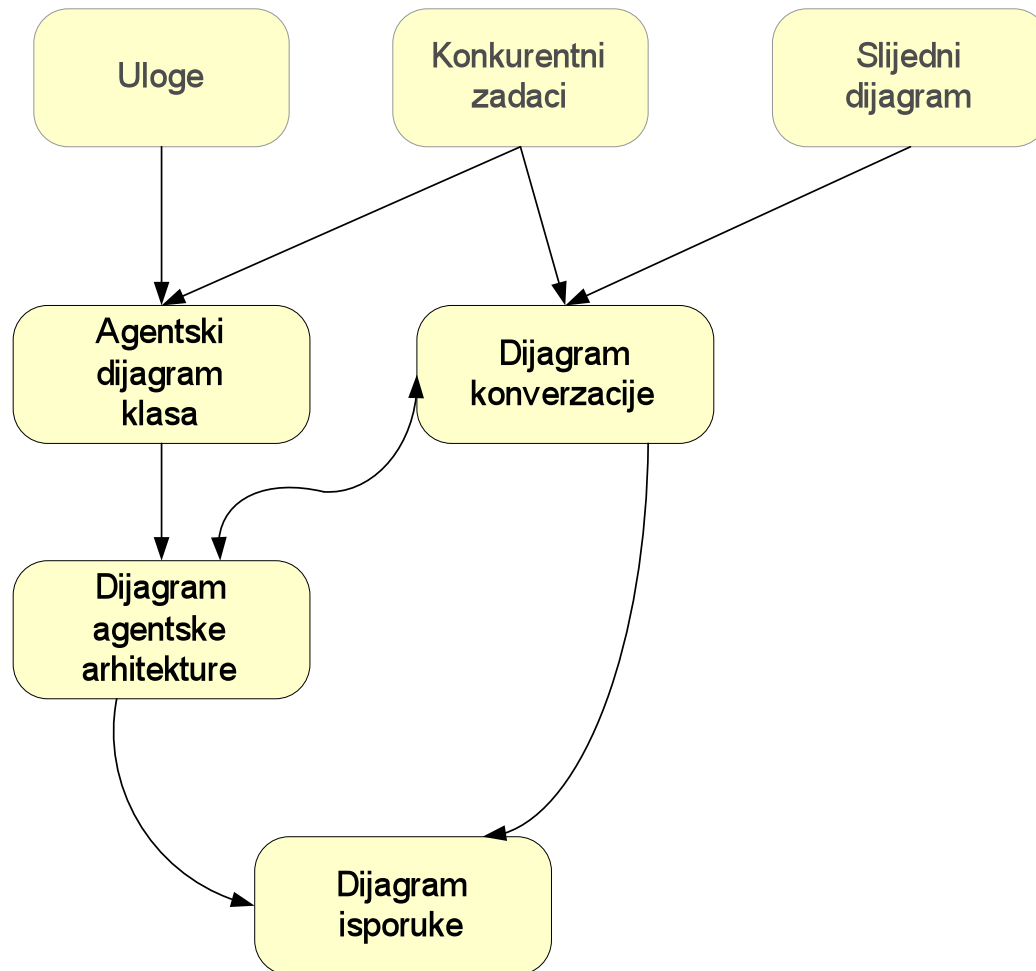
- dijagrami:

- hijerarhija ciljeva (*Goal hierarchy diagram*)
- dijagram uloga (*Role Model*)
- dijagram konkurentnih zadataka (*Concurrent Task Diagram*) – dijagram stanja
- dijagram slučaja uporabe (*Use Cases*)
- slijedni dijagram (*Sequence Diagram*)

MaSE modeli u fazi dizajna



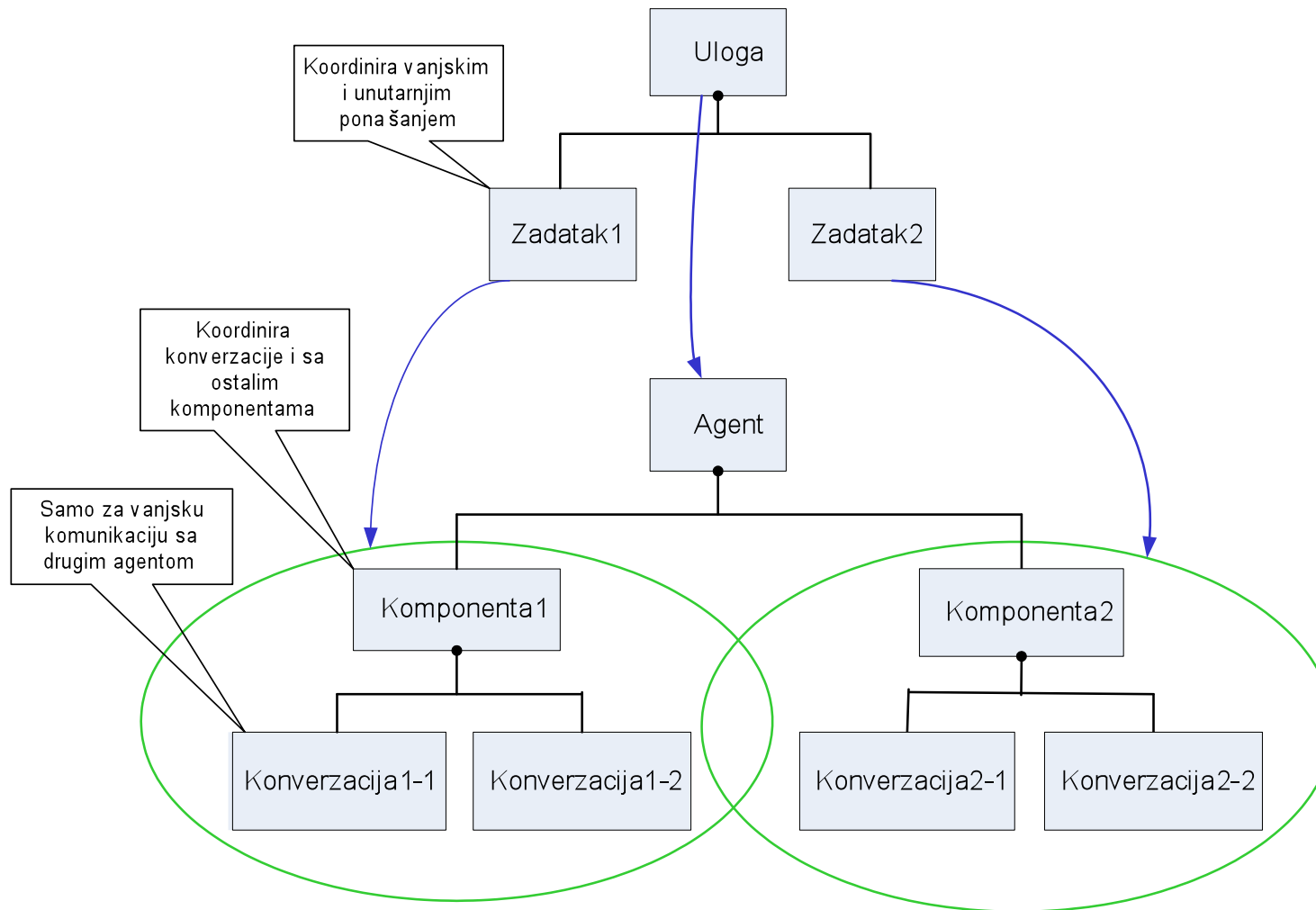
Zavod za telekomunikacije



- dijagrami:

- agentski dijagram klasa (*Agent Class Diagram*) – dijagram klasa
- dijagram konverzacije (*Conversation Diagram*) – dijagram stanja
- dijagram agentske arhitekture (*Agent Architecture Diagram*) – dijagram klasa
- Dijagram isporuke (*Deployment Diagram*)

Veza između modela analize i dizajna



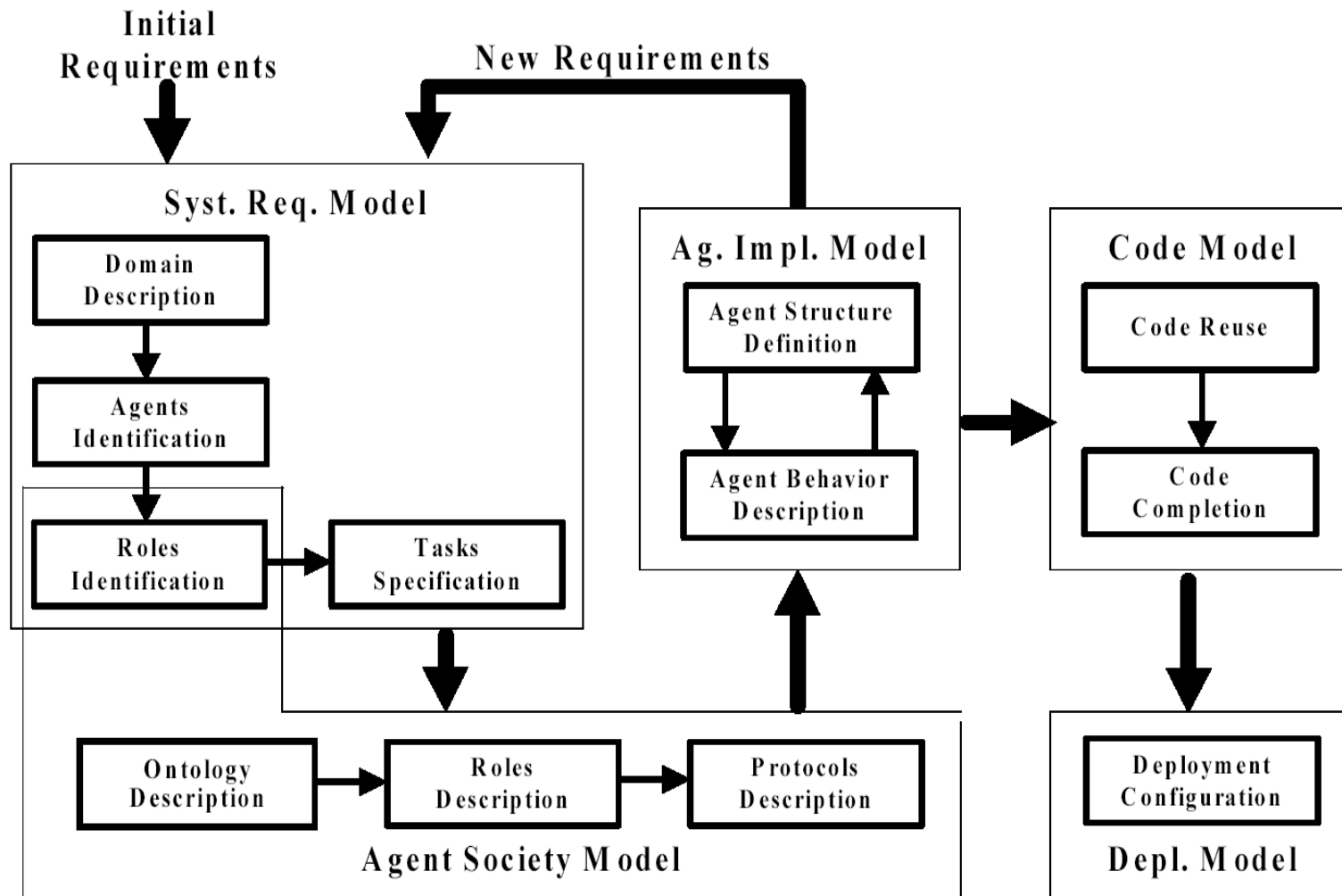
PASSI (*Process for Agent Societies Specification and Implementation*)



Zavod za telekomunikacije

- ◆ detaljna metodologija
- ◆ iz modela se može generirati kod
- ◆ za izradu višeagentskih sustava i društava
- ◆ koristi UML za modeliranje
- ◆ povezuje OO razvoj prog. proizvoda s konceptima umjetne inteligencije
- ◆ razvijen je alat PASSI ToolKit koji uključuje:
 - add-in za Rational Rose
 - alat za korištenje agentskih uzoraka

PASSI – metodologija



- ◆ *System Requirements Model*
 - *Domain Description* – funkcijski opis sustava (dij. slučaja uporabe)
 - *Agent Identification* – raspodjela odgovornosti po agentima (dij. paketa)
 - *Role Identification* – razrada odgovornosti agenata kroz scenarije uloga (slijedni dij.)
 - *Task Specification* – prikaz mogućnosti agenata (dij. aktivnosti)
- ◆ *Agent Society Model*
 - *Role Identification* – vidi gore
 - *Ontology Description* – opisuje znanje koje treba znati svaki agent (dij. klasa)
 - *Role Description* – prikazuje agenta, njihove uloge te poslove koje trebaju izvršavati (dij. klasa)
 - *Protocol Description* – opis gramatike, protokola te poruka koje agenti razmjenjuju (slijedni dij.)

PASSI – modeli (2)



Zavod za telekomunikacije

- ◆ *Agent Implementation Model*
 - *Agent Structure Definition* – opis strukture i rješenja agenta (dij. klasa)
 - *Agent Behavior Description* – opis ponašanja svakog agenta (dij. aktivnosti ili stanja)
- ◆ *Code Model*
 - *Code Reuse Library* – skup dijagrama klasa i aktivnosti koji predstavljaju uzorak koji se može ponovno iskoristiti
 - *Code Completion Baseline* – izvorni kod sustava
- ◆ *Deployment Model*
 - *Deployment Configuration* – opis kako će agenti biti raspoređeni u sustavu te koje će resurse koristiti i koja ograničenja imati u pogledu pokretnosti (dij. isporuke)
- ◆ *Testing*
 - testiranje jednog agenta – usporedba ponašanja u odnosu na zahtjeve
 - društveno testiranje – provjera interakcija između agenata u rješavajući zadatke kooperirajući

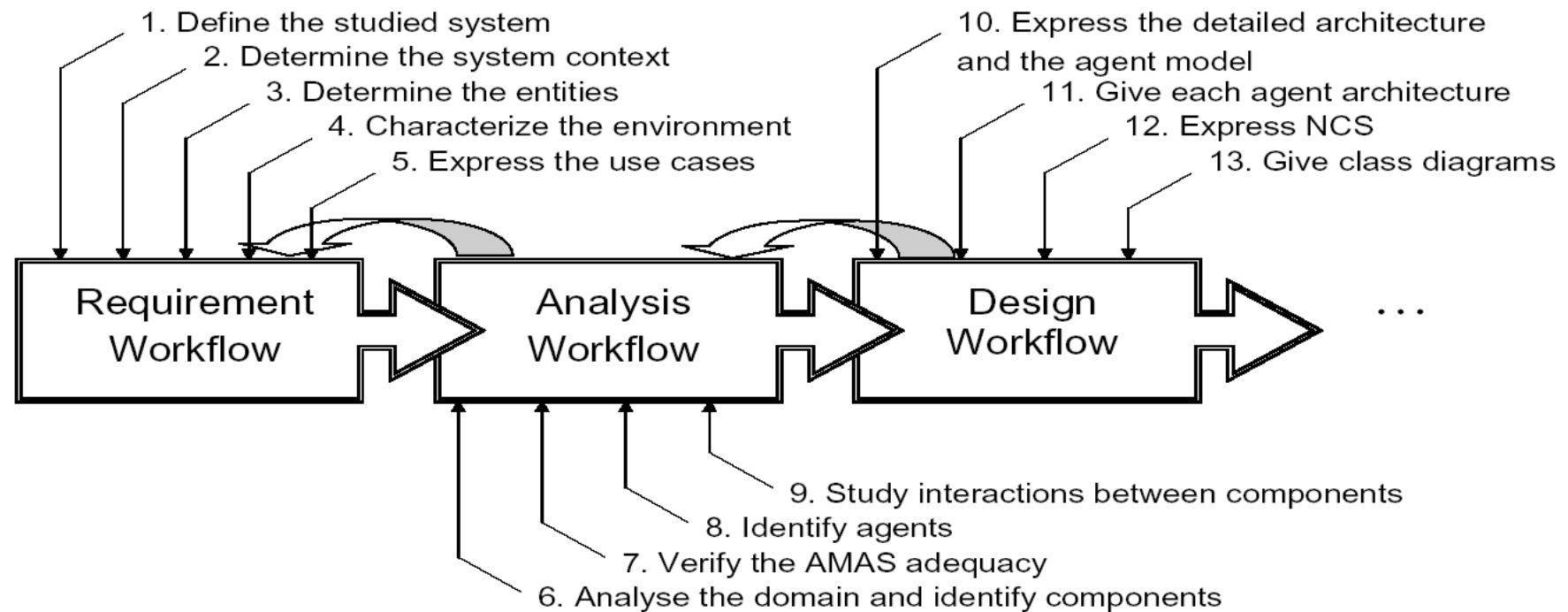
ADELFE (Atelier de Développement de Logiciels à Fonctionnalité Emergente)



Zavod za telekomunikacije

- ◆ *Toolkit to develop software with emergent functionality*
- ◆ Temelji se na RUP-u
- ◆ Koristi UML notaciju
- ◆ Temelji se na 4 faze:
 - Zahtjevi (*Requirements*)
 - Analiza (*Analysis*)
 - Dizajn (*Design*)
 - Implementacija (*Implementation*) – isti kao u RUP-u
 - Testiranje (*Test*) – isti kao u RUP-u

ADELFE – tijek izvođenja



MESSAGE



Zavod za telekomunikacije

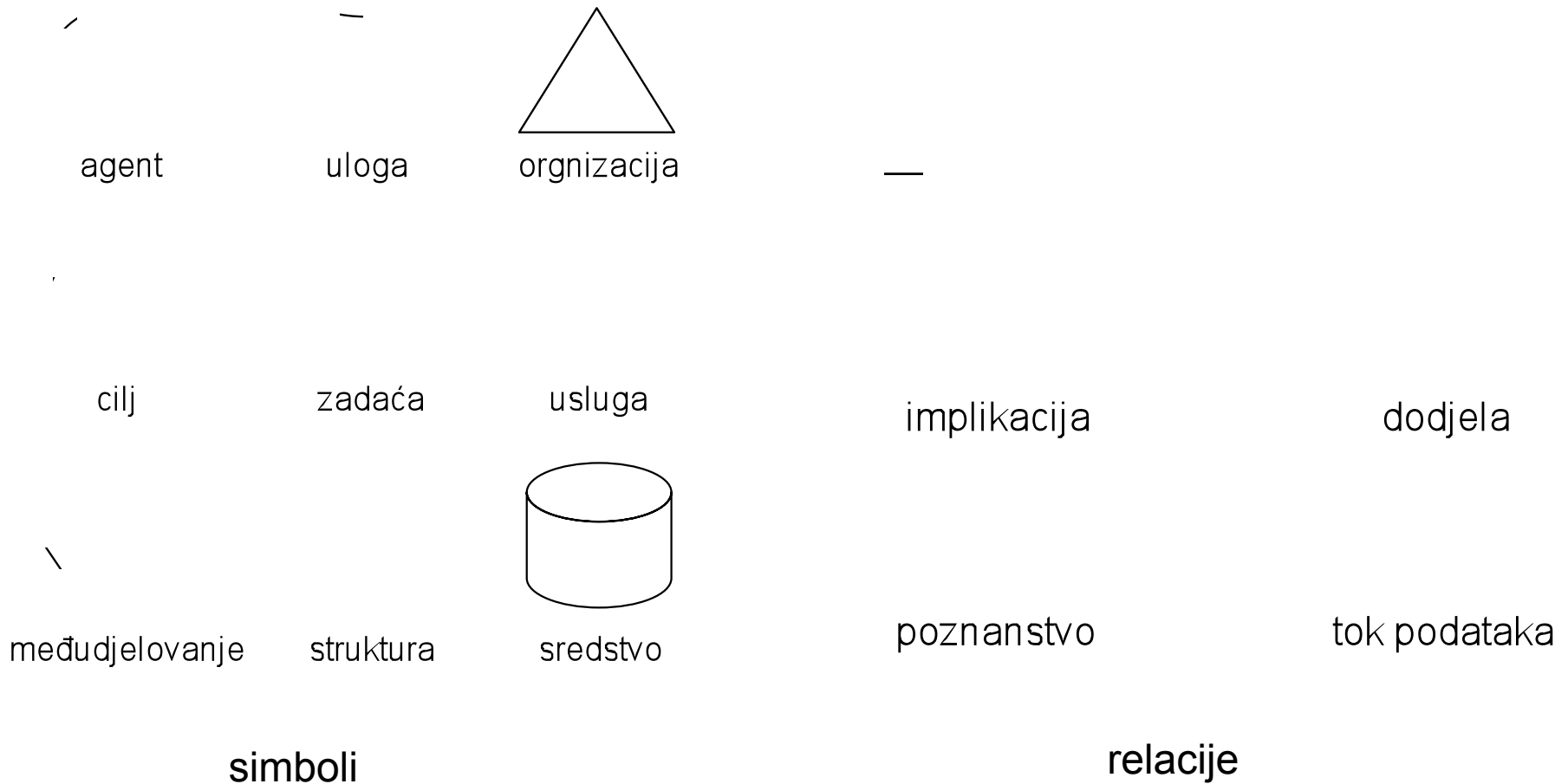
- ◆ EURESCOM projekt
- ◆ 2 godine je trajao (završen 7. mj. 2001.)

- ◆ fokus je na:
 - analizi,
 - meta-modelu,
 - notaciji i
 - procesu.
- ◆ proširuje UML meta-model
- ◆ dodatna notacija

MESSAGE – simboli i relacije

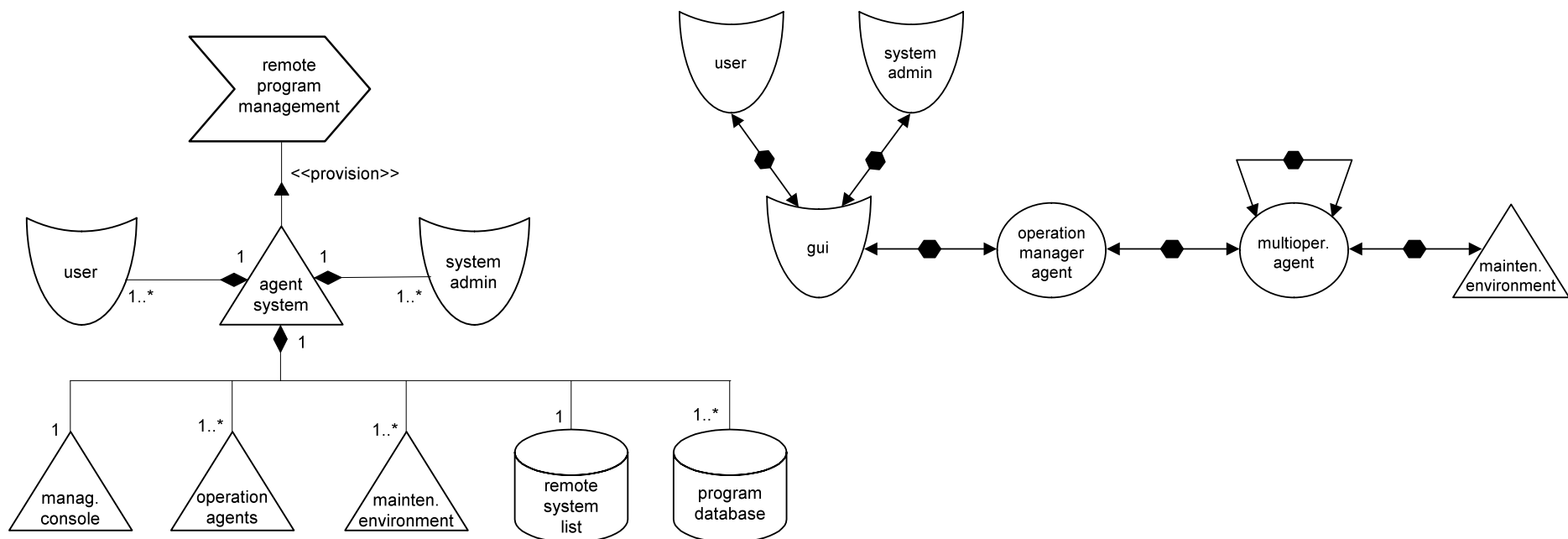


Zavod za telekomunikacije



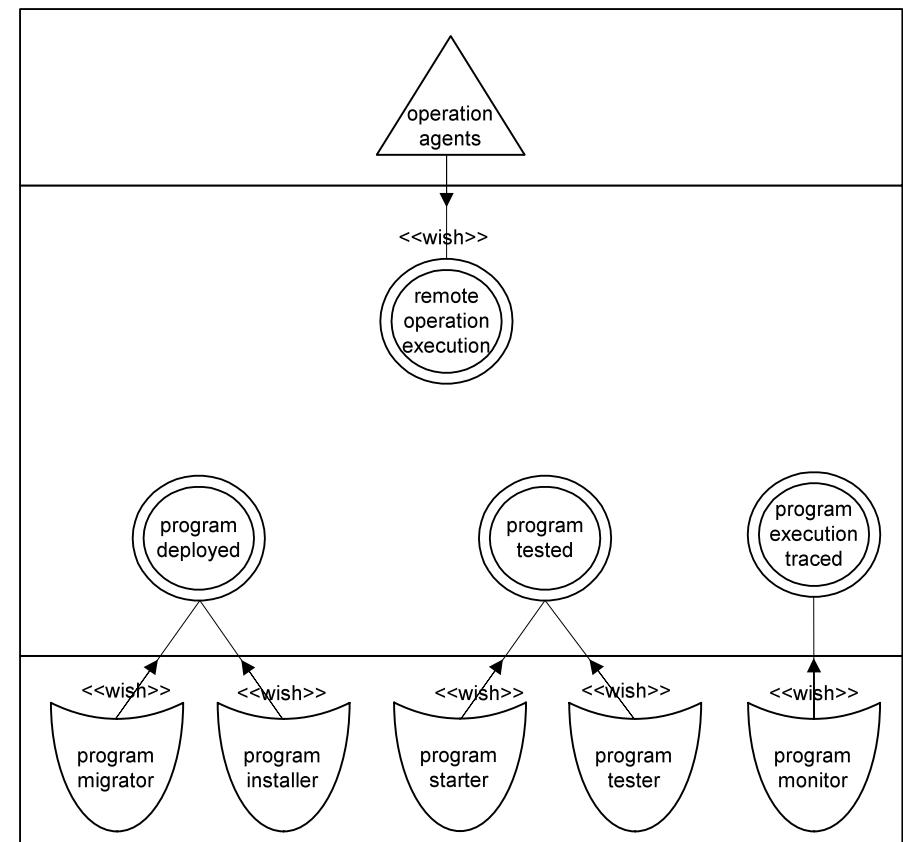
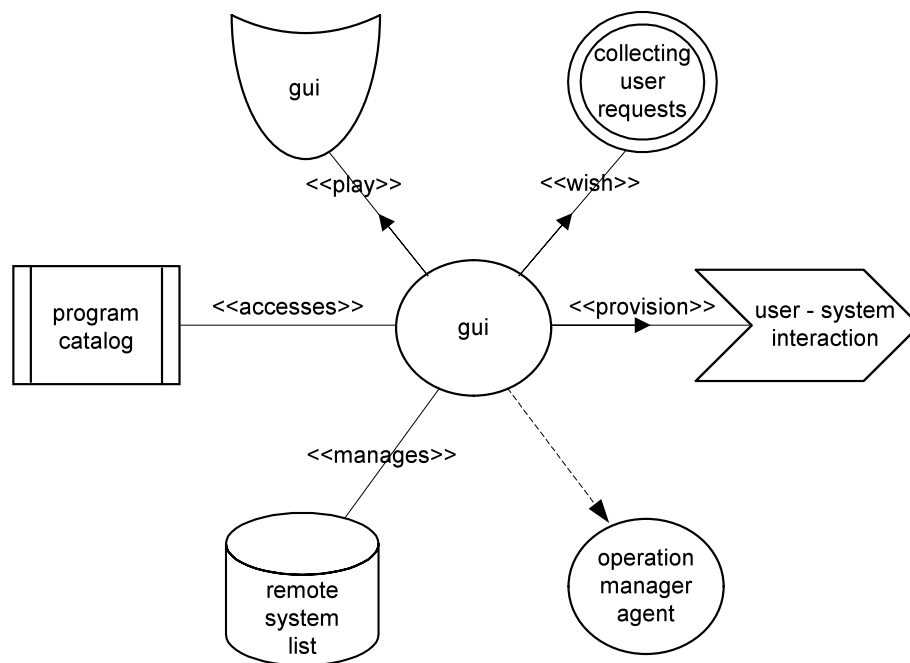
MESSAGE – pogled organizacije

- agenti, organizacije, uloge i resursi
- dijagrami strukturnih odnosa i organizacije - odnosi poznanstva



MESSAGE – pogled agent/uloga

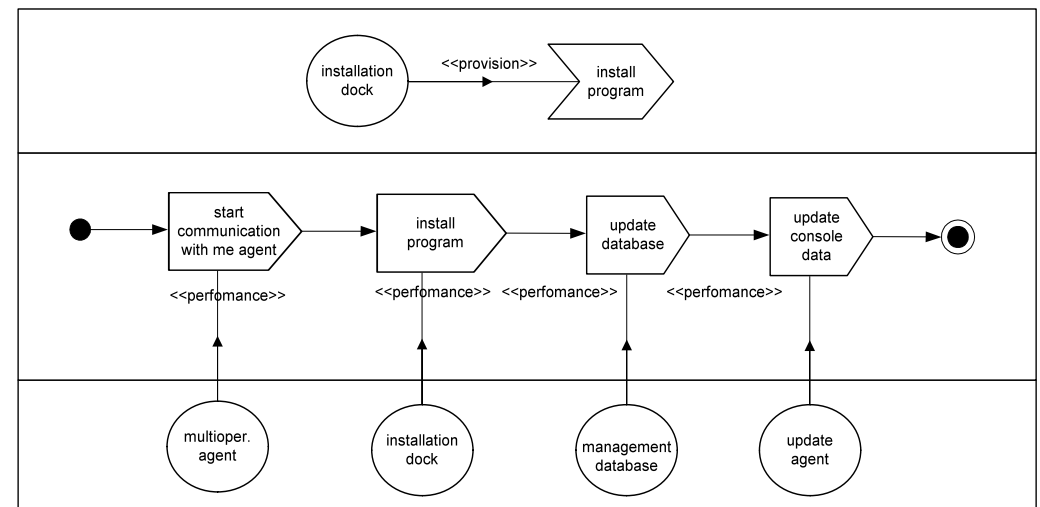
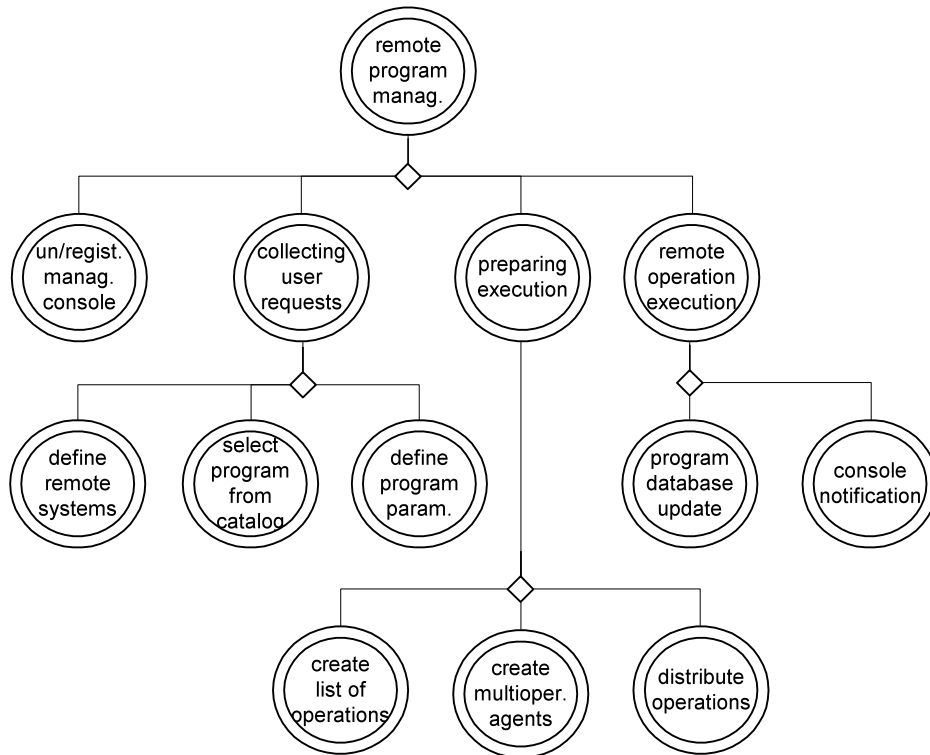
- agentski dijagram – agentske uloge, ciljevi, resursi, ...
- dijagram strukture delegiranja



MESSAGE – pogled cilj/zadaća



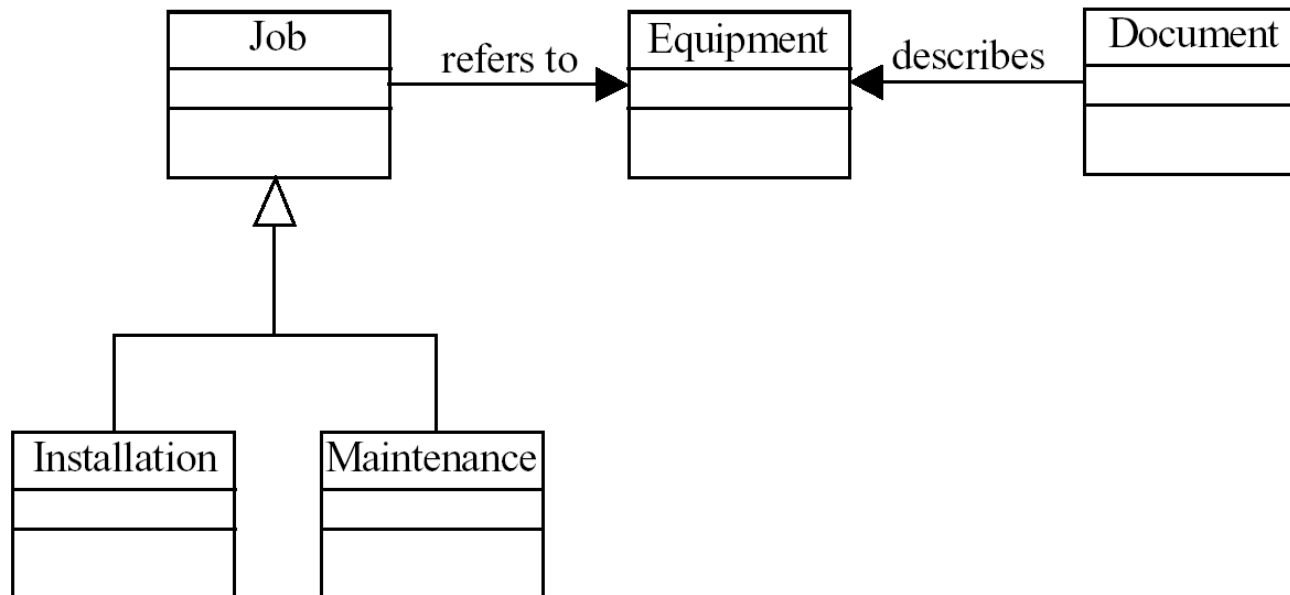
- dijagram ciljeva – implikacije i ovisnosti
- dijagram tijeka poslova



MESSAGE – pogled domene

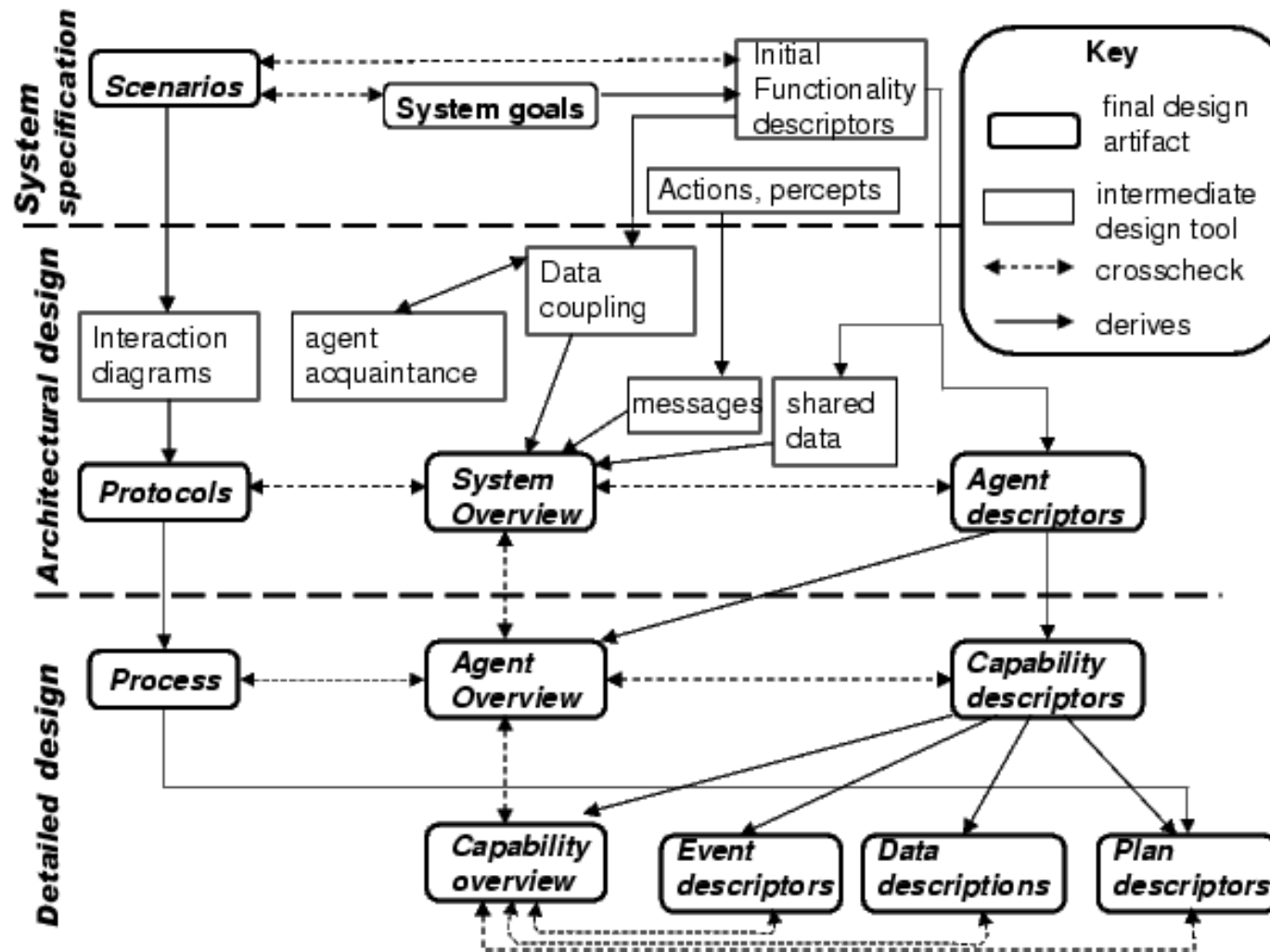


- veza koncepata iz domene i njihove relacije
- koristi UML dijagrame klasa



- ◆ namijenjen razvoju inteligentnih agenata (posebno BDI agenata)
- ◆ podržavaju sve faze razvoja (uključujući kodiranje, testiranje i održavanje)
- ◆ detaljni dizajn iz kojeg alat generira kostur koda za JACK i JADEx agentski programski okvir
- ◆ 3 faze:
 - specifikacija sustava
 - dizajn arhitekture
 - detaljni dizajn

Prometheus – pregled metodologije



- ◆ agentski specifično proširenje UML-a 2.0
- ◆ namijenjen za:
 - modeliranje domene,
 - analiza i specifikacija zahtjeva
 - dizajn sustava temeljenih na agentima te njihovim konceptima i principima
- ◆ dobro specificiran i dokumentiran
- ◆ kombinira OOSE i AOSE
- ◆ neovisan o teoriji, procesima ili implementaciji sustava
- ◆ proširiv
- ◆ implementacija alata kao dodatak Rational Rose-a
- ◆ nema sigurnosti, preslikavanje koncepata u koncepte agentskih platformi

AML (2)



Zavod za telekomunikacije

- ◆ definira metamodel
- ◆ nema dijagrame već definira koncepte:
 - ontologija
 - MAS entiteti (agenti, resursi, okolina)
 - socijalni aspekti
 - apstrakcija ponašanja i njihova dekompozicija
 - komunikativna interakcija
 - usluge
 - promatranje
 - mentalni koncepti (uvjerenje, stav, želja, namjera, ...)
 - MAS isporuka (kako će sustav izgledati isporučen)
 - pokretnost agenata

AUML – Agent UML



Zavod za telekomunikacije

- FIPA Modeling Technical Committee (Modeling TC)
- AUML – zajednička semantika, meta-model, apstraktna sintaksa za agentsko temeljene metodologije
- temelji se na UML 2.0
- plan - 12/2004 – završni prijedlog integracije metoda

izvori:

- | | |
|--|--------------|
| – UML 2.0 | – Gaia |
| – AOR | – BRIC |
| – PASSI | – Styx |
| – MESSAGE | – Prometheus |
| – Tropos (uključujući <i>i*</i> i GRL) | – MADkit |
| – ADELFE | – OPM |

- ◆ potpuno novi dijagrami (npr. ontologijski)
- ◆ novi koncepti:
 - hijerarhija dijagrama
 - grananja u slijednim dij. i dij. međudjelovanja
 - kombiniranje dijagrama (paketi i slijed)
 - predlošci
 - promjena uloge jednog agenta
- ◆ većina prijedloga je ugrađena u UML 2.0