

# Engenharia Orientada a Agentes Open AEOLus

Daniela Maria Uez  
dani.uez@gmail.com

17 de outubro de 2017

- 1 Introdução
- 2 Sistemas Multiagentes Abertos
- 3 AOSE
- 4 Open AEOLus
- 5 Considerações Finais

# Sumário

- 1 Introdução
- 2 Sistemas Multiagentes Abertos
- 3 AOSE
- 4 Open AEOLus
- 5 Considerações Finais

# Sistemas Multiagentes (I)

O SMA pode ser composto por:

- Agente (A) - entidades autônomas que agem em prol dos seus objetivos
- Ambiente (E) - disponibiliza recursos e ferramentas para os agentes
- Organização (O) - define normas e protocolos que permitem a cooperação entre os agentes

# Sistemas Multiagentes (II)



# Sumário

- 1 Introdução
- 2 Sistemas Multiagentes Abertos**
- 3 AOSE
- 4 Open AEOLus
- 5 Considerações Finais

# Conceitos Iniciais

## Tempo de desenvolvimento

Todas as atividades desenvolvidas antes da execução do sistema: requisitos, análise, projeto, implementação ...

## Tempo de execução

Momento em que o sistema está sendo executado

# Definição clássica de abertura

- Um SMA aberto é aquele no qual os agentes podem entrar e sair livremente em tempo de execução
- Os agentes que farão parte do sistema em tempo de execução não são conhecidos durante o desenvolvimento do sistema



# Exemplo SMA aberto: Cenário MAPC

- Desenvolvido por um time na Alemanha
- Mas os agentes que fazem parte do sistema são desenvolvidos por diversos times ao redor do mundo
- Pessoal da Alemanha não sabe nada sobre o agentes durante o desenvolvimento do sistema (agentes desconhecidos)
- Somente durante a execução do sistema os agentes entrarão no sistema, para participar da competição

# SMA Aberto

- Artefatos, protocolos de interação, normas, etc, que não eram conhecidos durante o desenvolvimento do SMA também podem ser incluídos em tempo de execução
- Ou seja: outras dimensões do SMA também podem ser abertas

# Uma nova definição

## SMA aberto

- Um SMA é aberto quando qualquer uma de suas dimensões puder ser alterada em tempo de execução
- Ou seja, elementos de qualquer dimensão (agentes, artefatos, protocolos, normas...) que são desconhecidos durante o desenvolvimento do sistema são incluídos durante a execução
- Esses novos elementos precisam ser integrados ao sistema e trabalhar juntamente com os elementos já no sistema

# Tipos de abertura

- Para o time da Alemanha, o cenário do MAPC é um sistema:
  - Aberto na dimensão dos agentes  
Completamente desconhecido durante o desenvolvimento do sistema
  - Fechado na dimensão do ambiente  
Todos os elementos são conhecidos durante o desenvolvimento do sistema
  - Parcialmente aberto na dimensão da organização  
Alguns papéis e normas conhecidos durante o desenvolvimento, outros incluídos durante a execução do sistema

# Modelando SMA abertos

- A abertura não pode ser visualizada em durante o desenvolvimento do sistema  
Só pode ser visualizada durante a execução, quando novos elementos forem incluídos no sistema

# Então, como projetar SMA abertos?

- Durante o desenvolvimento do SMA alguns dos elementos não são conhecidos
- ~> Esses elementos não podem ser desenvolvidos
- Mas irão fazer parte do sistema em tempo de execução
- ~> O sistema precisa ser desenvolvido para que esses elementos sejam corretamente integrados em tempo de execução

# Sumário

- 1 Introdução
- 2 Sistemas Multiagentes Abertos
- 3 AOSE**
- 4 Open AEOLus
- 5 Considerações Finais

# Engenharia de software orientada a agentes

- Ainda sem a estrutura da engenharia de software OO
- Diversos métodos e linguagens de modelagens desenvolvidos
- Não existe um padrão (como a UML e o UP na OO)



# Alguns Métodos AOSE

- Adelfe (Bernon *et al.* [2002])  
<http://www.irit.fr/ADELFE/>
- ASPECS (Cossentino *et al.* [2010])  
<http://www.aspecs.org/Home>
- Ingenias (Pavón and Gómez-Sanz [2003])  
<http://ingenias.sourceforge.net/>
- O-MaSE (DeLoach and García-Ojeda [2010])  
<http://agenttool.cis.ksu.edu/>
- OperA+Environment (Dastani *et al.* [2004])
- Prometheus (Padgham and Winikoff [2004])  
<http://www.cs.rmit.edu.au/agents/prometheus/>
- Tropos (Cossentino and Seidita [2005])  
<http://www.troposproject.org/node/93>
- Prometheus AEOLus (Uez *et al.* [2013])

# Métodos AOSE x Plataformas

Method	Tool	Platform
ADELFE	Adelfe Toolkit	-
ASPECS	Janeiro	Janus
Ingenias	IDK	JADE
O-MaSe	AgentTool III	JADE
Prometheus	PDT	Jack
Tropos	TAM4E	Jack Jadex

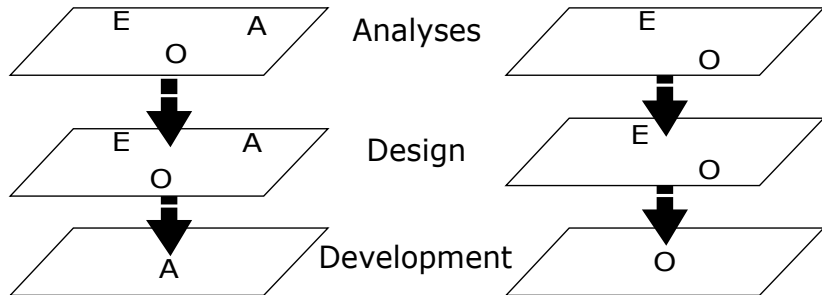
# Engenharia de Métodos Situacionais (SME)

- Visa criar métodos específicos para uma situação
- Usa fragmentos de métodos existentes
  - Medee Framework (Casare [2011])  
<http://medee.poli.usp.br/>

# Engenharia de Métodos Situacionais (SME)

- Visa criar métodos específicos para uma situação
- Usa fragmentos de métodos existentes
  - Medee Framework (Casare [2011])  
<http://medee.poli.usp.br/>
- Desvantagens do uso do SME
  - Exige conhecimento do repositório de métodos e dos fragmentos
  - Necessita grande experiência para desenvolvimento dos métodos
  - Pode não existir fragmento que atenda as necessidades

# AOSE e SMA abertos



# AOSE e SMA abertos (II)

- Esses métodos costumam depender de uma única dimensão  
Conceitos de outras dimensões costumam ser traduzidos em conceitos de agente ou organização
  - As plataformas mais comuns de desenvolvimento lidam somente com conceitos de agentes
- Os métodos não oferecem *guidelines* específicos para inclusão de novos elementos em tempo de execução

## AOSE e SMA abertos (III)

- **Isso não impede a especificação de SMA abertos**

Mas dificulta a definição de questões específicas da abertura

- Mas não é uma tarefa fácil determinar o que é preciso para lidar com abertura durante o desenvolvimento do SMA

# Sumário

- 1 Introdução
- 2 Sistemas Multiagentes Abertos
- 3 AOSE
- 4 Open AEOLus**
- 5 Considerações Finais



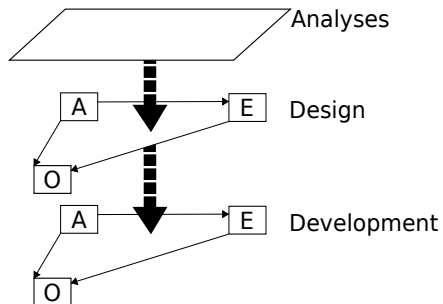
# Open AEOLus



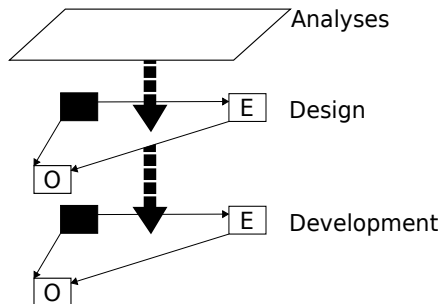
## **Open AEOLus**

- Método para modelagem de SMA abertos
- Dimensões independentes - cada uma definida com abstrações e conceitos próprios
- Conceitos de borda - conceitos especiais que visam auxiliar a inclusão dos novos elementos em tempo de

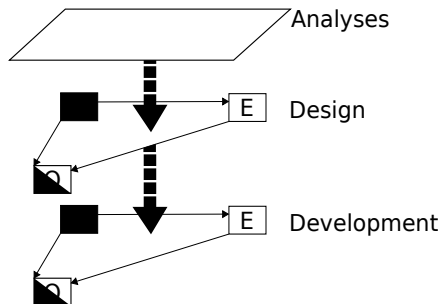
# Visão Geral



# Visão Geral (II)



# Visão Geral (III)





# Conceitos de borda

- Usados para definir, em tempo de desenvolvimento, informações específicas sobre como as dimensões são ligadas em tempo de execução

# Fases

- **Análise:** o que o sistema deve fazer?
- **Projeto:** define cada dimensão independentemente e os conceitos de borda que permitem a integração das dimensões em tempo de execução
- **Implementação:** refina os conceitos da fase anterior levando em conta uma plataforma de implementação específica (JaCaMo) - geração de código

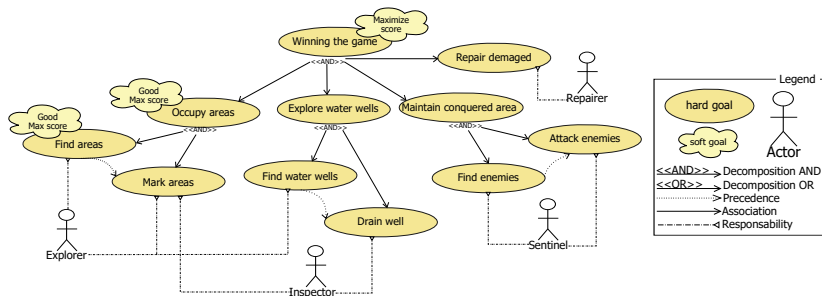
# Análise

Parte de uma definição inicial do sistema

- 1 Define *use cases*
- 2 Define objetivos do sistema
- 3 Define atores envolvidos
- 4 Liga atores e objetivos
- 5 Caracterização das dimensões



# Atores - Objetivos



# Caracterização das dimensões

## MAS Dimensinal Review

### Agents

- ☐ Open ☐ Closed  
☐ Half-Open

### Organization

- ☐ Open ☐ Closed  
☐ Half-Open ☐ Inexistent

### Environment

- ☐ Open ☐ Closed  
☐ Half-Open ☐ Inexistent

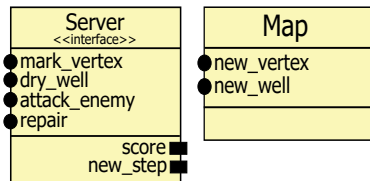
# Projeto

Com base nos artefatos criados na fase anterior

- Projeto das dimensões: define cada dimensão (fechada ou parcialmente aberta) independentemente, usando conceitos e abstrações próprias
- Projeto de abertura: define os conceitos de borda para auxiliar na integração dos elementos das dimensões abertas

# Projeto das dimensões - Ambiente

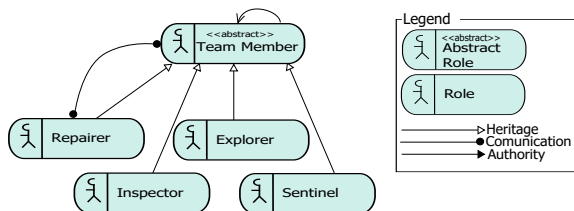
- Com base nos *use cases* e no diagrama de atores e objetivos, define os artefatos do ambiente (atores externos ou recursos)
- Refina os artefatos para inclusão de eventos e operações



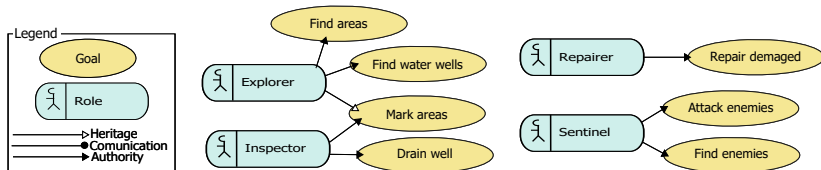
# Projeto das dimensões - Organização

- Identifica os papéis que farão parte da organização
- Refina os papéis definindo relações entre eles
- Define as responsabilidades de cada papel, ligando-o ao seu objetivo

# Refinamento de papéis



# Responsabilidade dos papéis

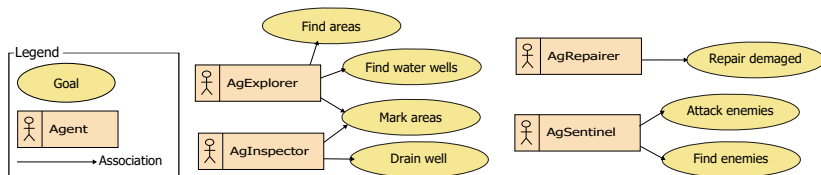


# Projeto das dimensões - Agentes

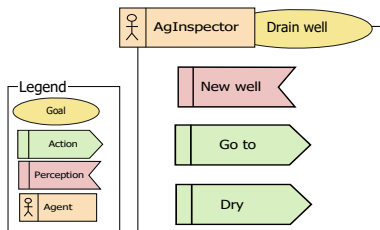
- Identifica agentes
- Define responsabilidades dos agente ligando-os com os objetivos
- Define planos do agente para atingir os objetivos



# Responsabilidade dos agentes



# Planos dos agentes

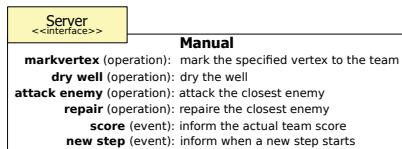
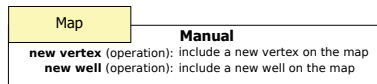


# Projeto de abertura

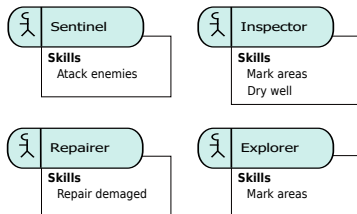
- Para cada dimensão aberta, um conjunto diferente de conceitos de borda deve ser levado em conta:

	A	E	O
A	–	Manual	Skills
E	Manual	–	Brute Facts
O	Skill	Institutional Facts	–

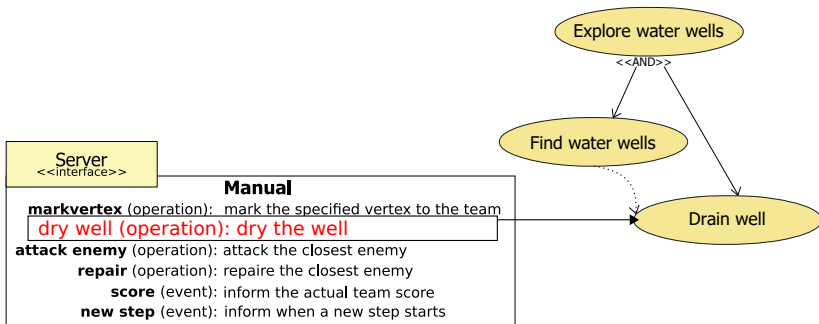
# Manual



# Skills



# Count-As



# Implementação

Refina os elementos definidos no projeto para geração de código do JaCaMo

# Sumário

- 1 Introdução
- 2 Sistemas Multiagentes Abertos
- 3 AOSE
- 4 Open AEOLus
- 5 Considerações Finais**



# Para saber mais...

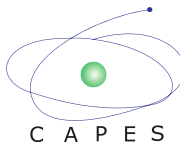
- Site (em construção constante):  
<http://www.uez.com.br/aeolus>
- Ferramenta AEOLia
  - Ferramenta para uso do método AEOLus
  - Em desenvolvimento (plug-in pra o Eclipse)

# That's all folks!

Obrigada!  
dani.uez@gmail.com



DAS



# Bibliografia I

- Federico Bergenti and Michael N. Huhns. On the use of agents as components of software systems. In Federico Bergenti, Marie-Pierre Gleizes, and Franco Zambonelli, editors, *Methodologies and Software Engineering for Agent Systems: The Agent-Oriented Software Engineering Handbook*, volume 11 of *Multiagent Systems, Artificial Societies, and Simulated Organizations*, chapter 2, pages 273–296. Kluwer Academic, 2004.
- Carole Bernon, Pierre Glize, Gauthier Picard, and Pierre Glize. The ADELFE methodology for an intranet system design. In *AOIS 2002*, pages 27–28. 2002.
- Sara Jane Casare. *Medee: a method framework for multiagent systems*. Tese de doutorado em sistemas digitais, Universidade de São Paulo, 2011.

## Bibliografia II

- M. Cossentino and V. Seidita. Tropos: Processo e frammenti, 2005.
- Massimo Cossentino, Nicolas Gaud, Vincent Hilaire, Stéphane Galland, and Abderrafiâa Koukam. ASPECS: an agent-oriented software process for engineering complex systems. *JAAMAS*, 20(2):260–304, 2010.
- Mehdi Dastani, Joris Hulstijn, Frank Dignum, and John-Jules Ch. Meyer. Issues in multiagent system development. In *AAMAS '04: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 922–929, Washington, DC, USA, 2004. IEEE Computer Society.
- Scott A. DeLoach and Juan Carlos García-Ojeda. O-mase: a customisable approach to designing and building complex, adaptive multi-agent systems. *Int. J. Agent-Oriented Softw. Eng.*, 4(3):244–280, 2010.

## Bibliografia III

- Brian Henderson-Sellers. Agent-oriented methodologies: an introduction. In Paolo Giorgini and Brian Henderson-Sellers, editors, *Agent-Oriented Methodologies*. Idea Group Publishing, 2005.
- Lin Padgham and Michael Winikoff. *Developing Intelligent Agent Systems: A Practical Guide*. Halsted Press, New York, NY, USA, 2004.
- Juan Pavón and Jorge J. Gómez-Sanz. Agent oriented software engineering with INGENIAS. In *CEEMAS 2003*, volume 2691 of *LNAI*, pages 394–403. Springer, 2003.
- Daniela Maria Uez, Jomi Fred Hübner, and Carine G. Webber. Método para modelagem de agentes, ambiente e organização de sistemas multiagentes. In *Anais IV Workshop Brasileiro sobre Sistemas de Software Autônomos - Autosoft 2013*, pages 41–50, 2013.