

Linguagens de Modelação de Sistemas Multi-Agentes

Integrated Master's in Informatics Engineering

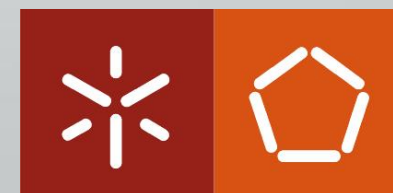
Intelligent Agents

2018/2019

Synthetic Intelligence Lab

Filipe Gonçalves

César Analide



Introdução

Agentes de Software:

- Entidade computacional situado num ambiente no qual efectua acções com autonomia e pró-actividade, de acordo com a sua percepção. Pode apresentar raciocínio e capacidade de adaptação (e.g. gestor de redes, gestor de processos, pesquisa de informação, etc.)

Sistema Multi-Agentes:

- Conjunto de agentes que interagem entendendo-se e coordenando-se em tarefas globais envolvendo cooperação ou competição

Agentes como extensões a **Objectos Activos**:

- Agentes passivos (aceitar / recusar pedidos)
- Agentes pró-activos (iniciar atividade sem invocação externa)

Introdução

Unified Modeling Language (UML) para modelação de **software orientado a objectos**
(adotado por OMG em Novembro de 1997)

UML não dá suporte para a modelação de SMA

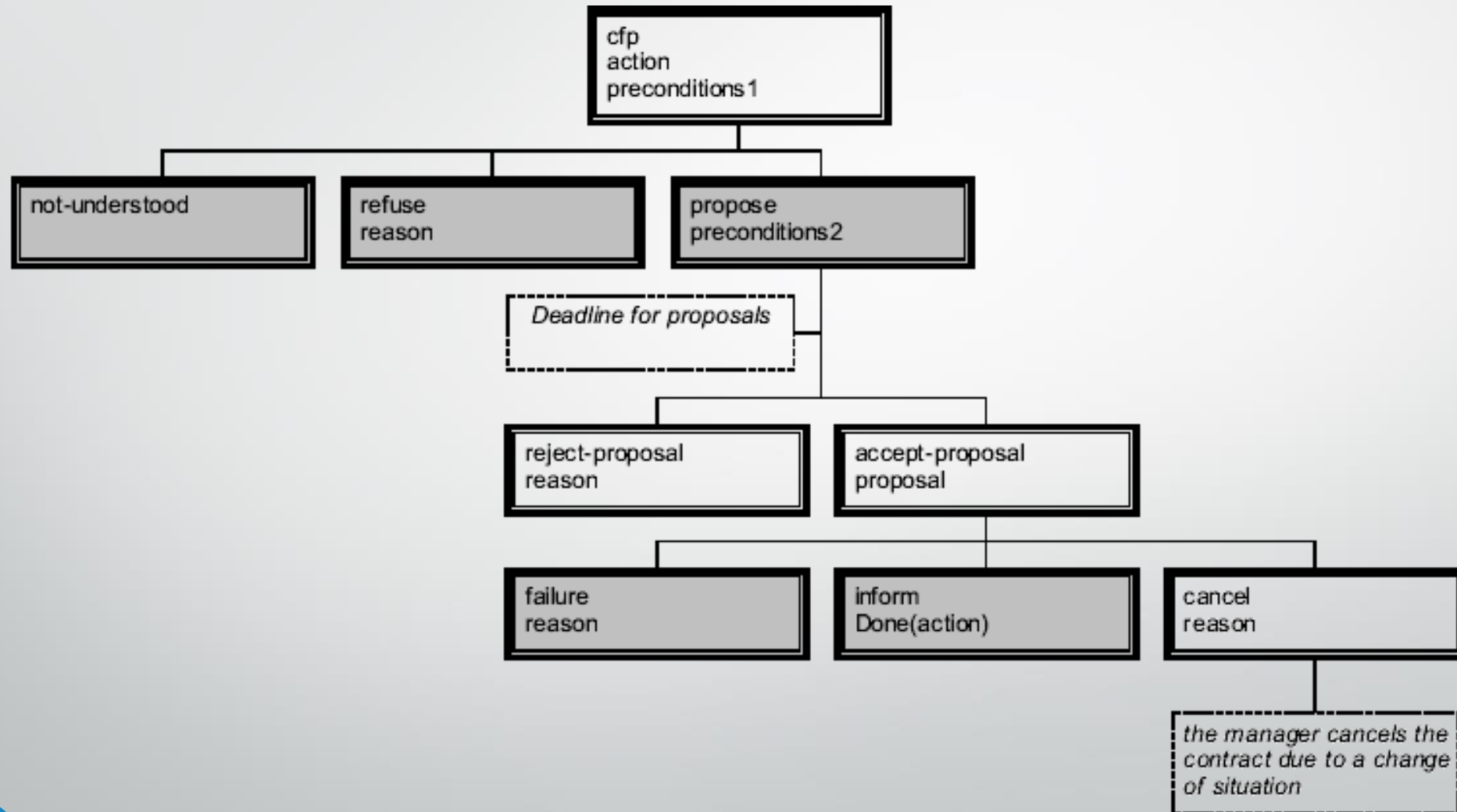
- Agentes e objetos possuem características / propriedades diferentes

Porquê não estender UML para modelar SMA?

AUML: Variações e Extensões a UML para modelação de actividades de Agentes

- FIPA (www.fipa.org) e
- OMG-AUML Agent Group (<http://aot.ce.unipr.it/auuml/>)
- **Representação de Protocolos de interação para Agentes**

FIPA Notation



AUML – Agent UML

- O objectivo do AUML passa pelo desenvolvimento de uma especificação formal dos protocolos de interação entre agentes (*AIP*)
- Adaptação dos diagrama de sequência de UML para modelar as interações entre agentes
- Seguiu-se a adaptação de outros diagramas

Extensões de representações UML:

- “packages” (pacotes)
- “templates” (moldes)
- Diagramas de Sequência
- Diagramas de Colaboração
- Diagramas de Actividades
- Diagramas de Estados
- Diagrama de Classes e Objetos

AUML – Agent UML

Aplicação de modelos AUML:

- Especificação dos Protocolos de Interação de Agentes (AIP)
- Especificação mais pormenorizada da invocação de acções
- Extensão do pacote
- Extensão do diagrama de implantação

AUML adota uma abordagem em camadas para protocolos:

- **Nível 1** - Representa o protocolo geral (diagramas de sequência, pacotes, modelos)
- **Nível 2**- Representar interações entre agentes (sequência, colaboração, actividade, diagramas de estado)
- **Nível 3**- Representar o processamento interno do agente (diagramas de actividade e de estado)

Nível 1: Protocolo Geral

Diagrama de Sequência

- Define o comportamento de grupos de objetos
- Interações básicas entre objetos ao nível da invocação de métodos
- No AUML, possibilitam demonstrar as interações / comunicações existentes entre Agentes do sistema

Diagrama de Sequência UML

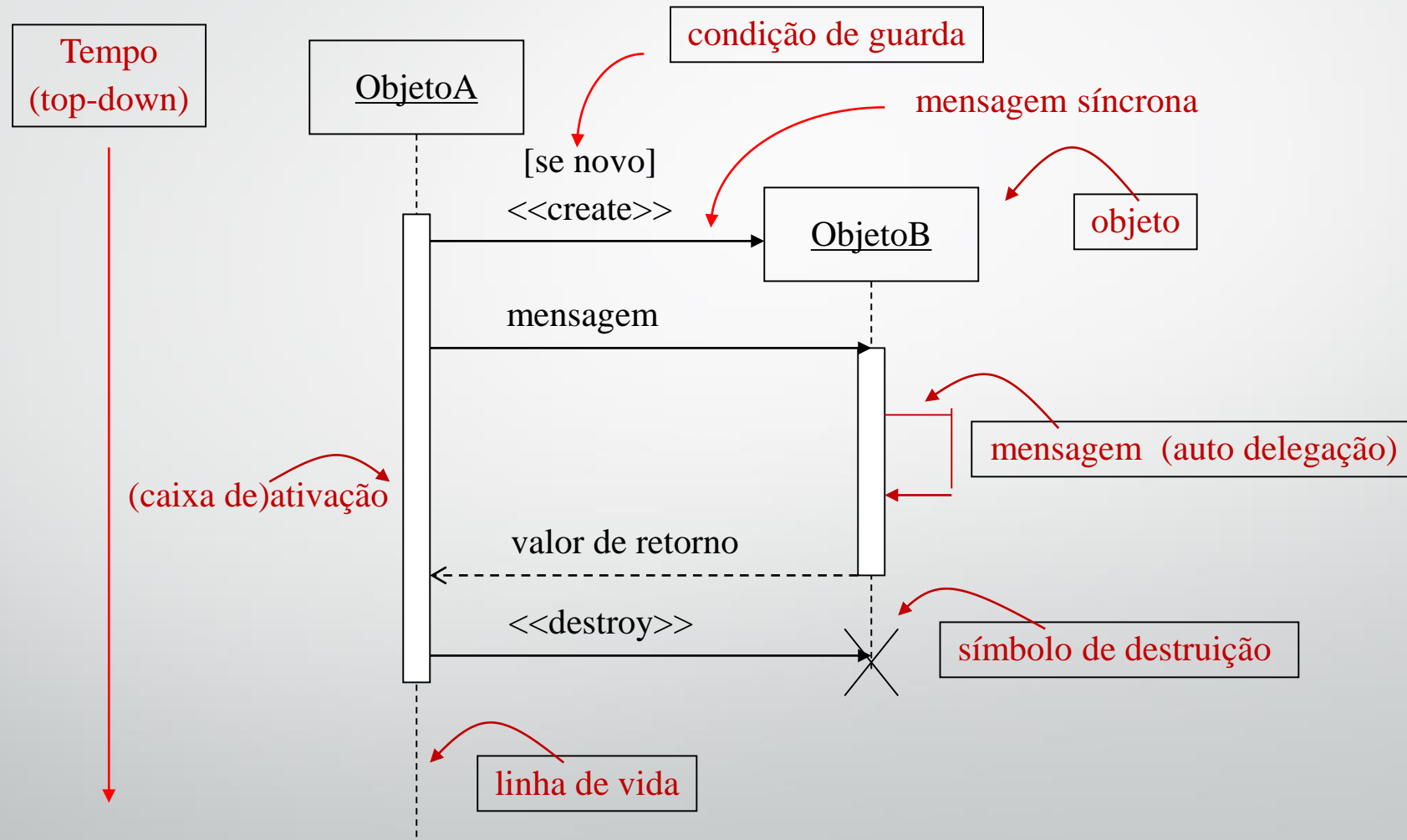


Diagrama de Sequência AUML

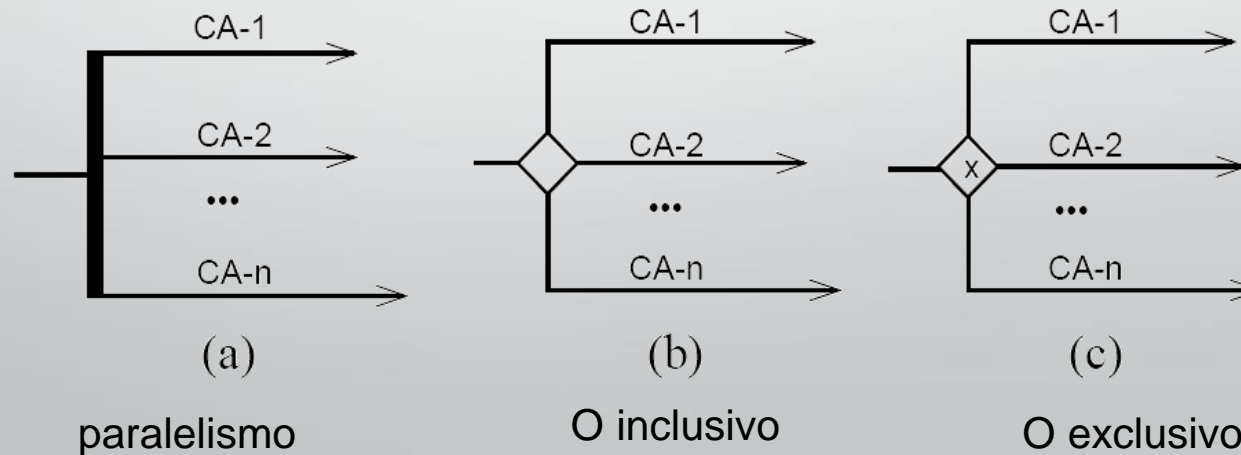
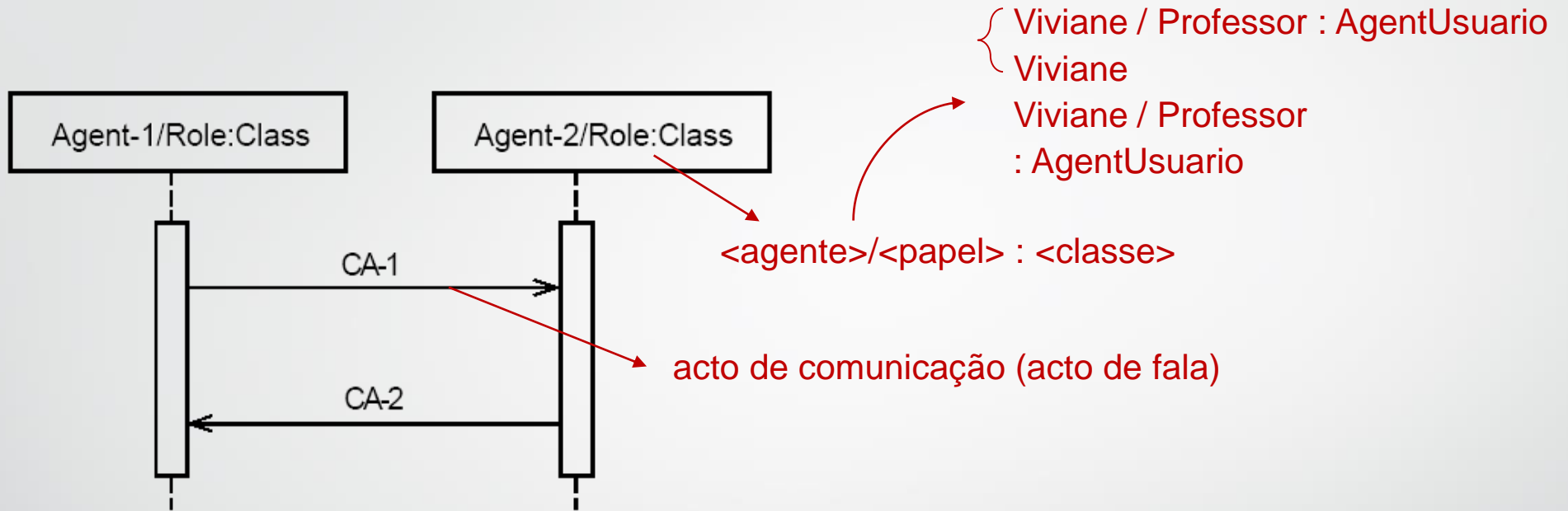
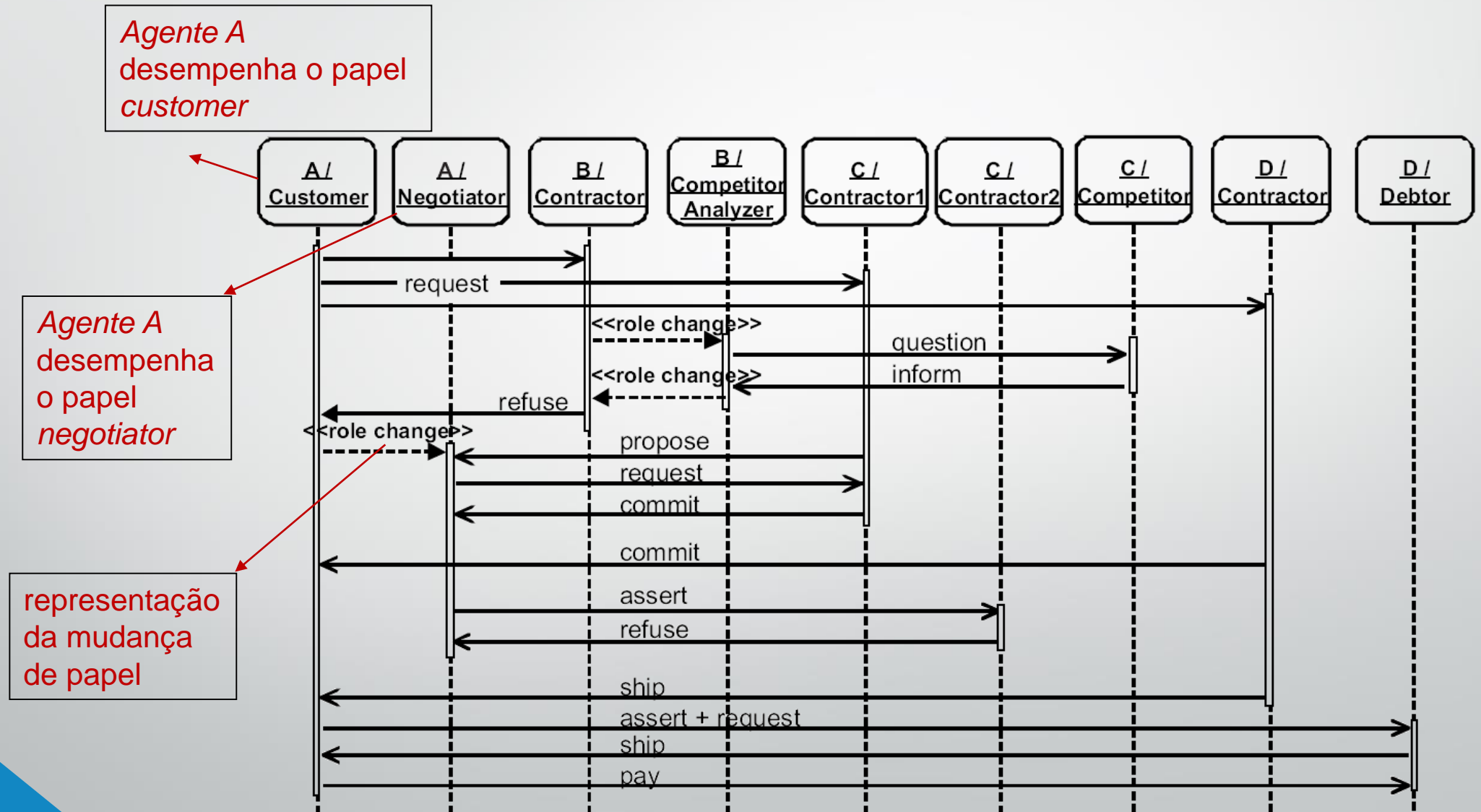
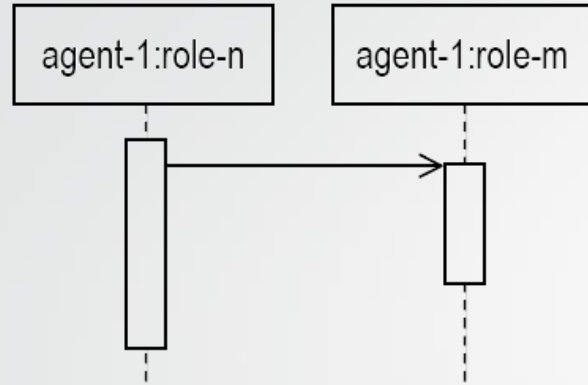


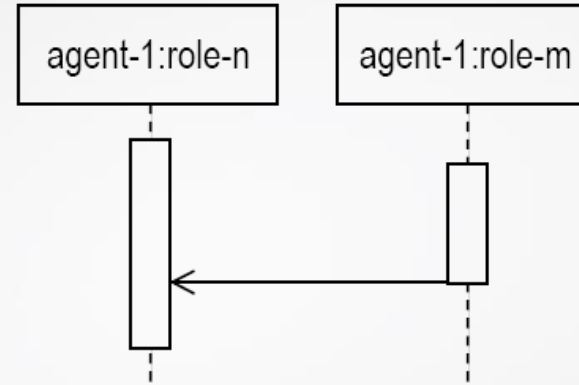
Diagrama de Sequência AUML (Exemplo)



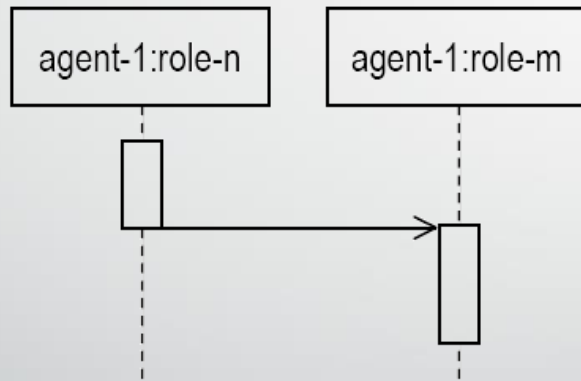
Diferentes Estados de Agentes



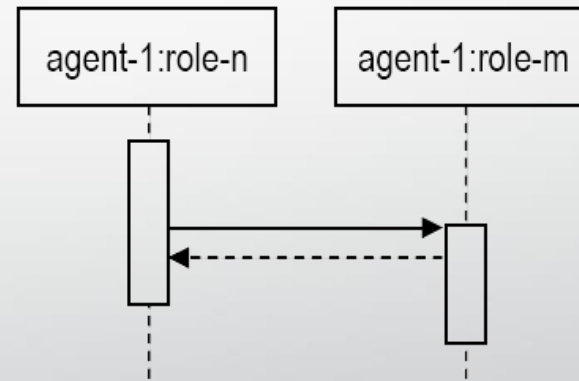
(a) Activate



(b) Suspend

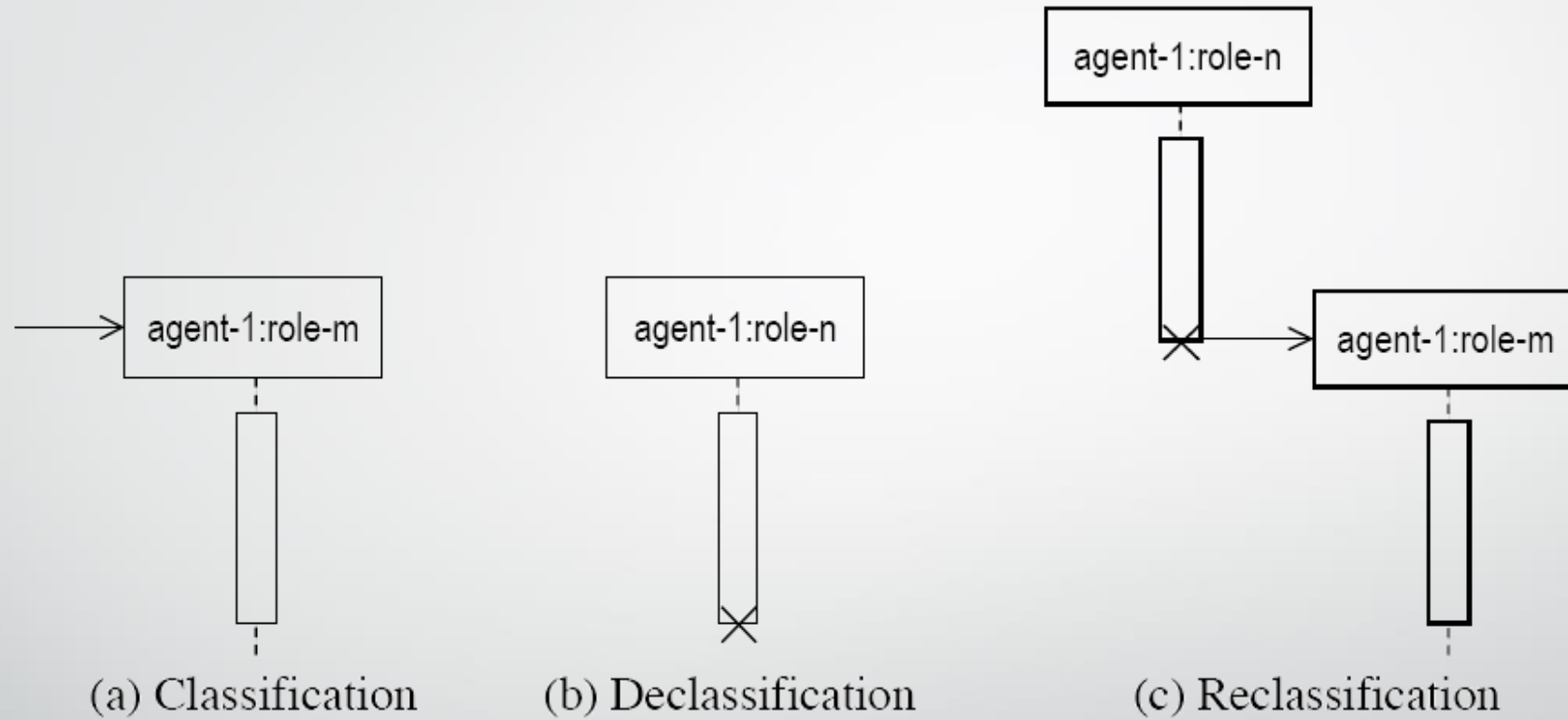


(c) Shift (asynchronous)

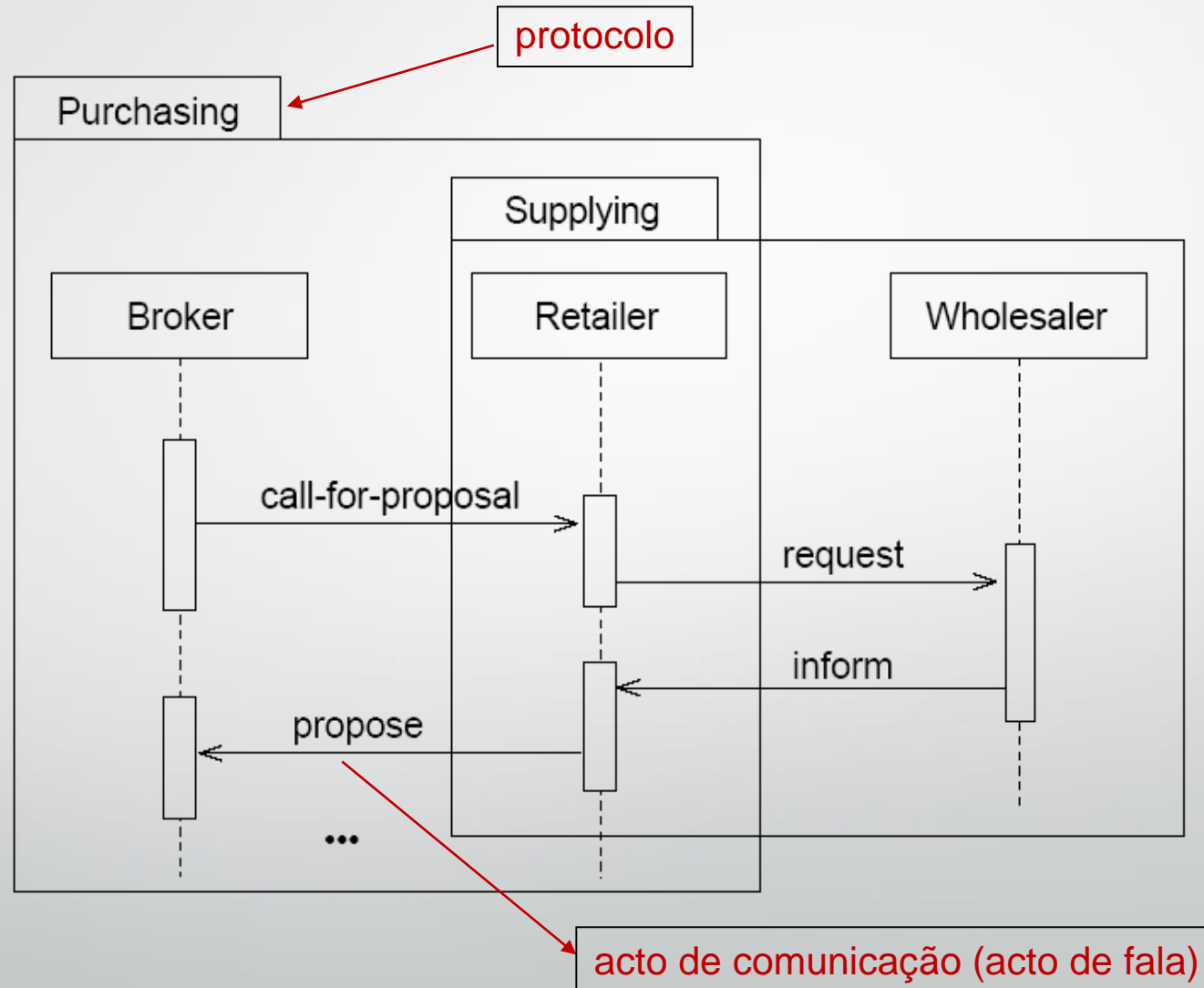


(d) Shift (synchronous)

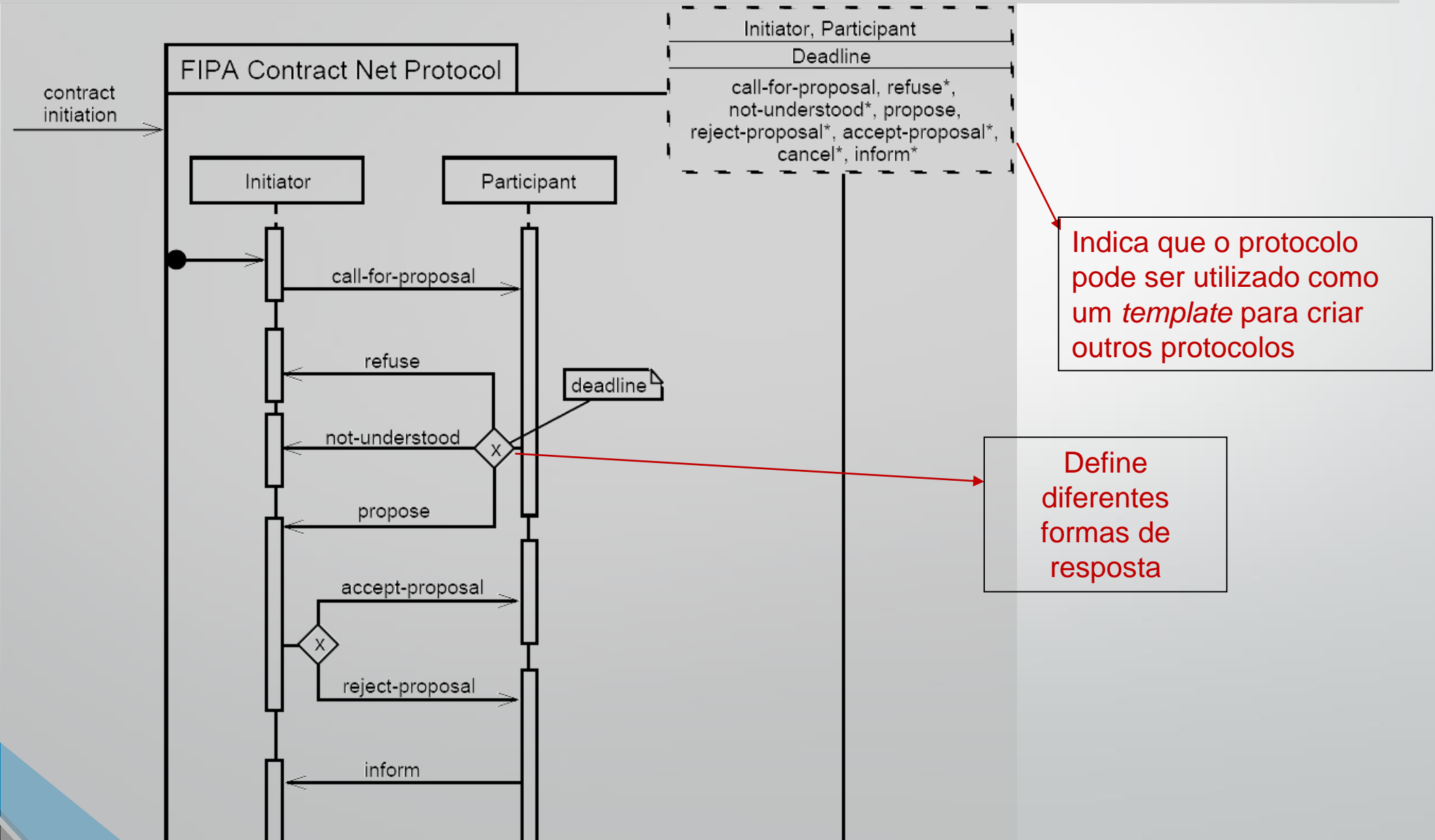
Diferentes Estados de Agentes



Modelação de Protocolos



Modelação de Protocolos (Packages)



Nível 2: Interações entre Agentes

Diagramas de sequência estendidos (segmentos simultâneos de interação)

Diagramas de colaboração

Diagramas de atividades

No entanto, maior complexidade do sistema exige uma apresentação gráfica mais complexa:

- Muitas vezes, precisamos de expressar o papel que um agente desempenha no decorrer de sua interação com outros agentes
- Se o número de agentes e funções aumenta, os diagramas UML tornam-se graficamente complexos
- UML não possui capacidades para representar as funções dos agentes nas linhas de interação. **Solução: Mensagens identificam transição de role**

Diagrama de Colaboração (Exemplo)

Diagrama UML

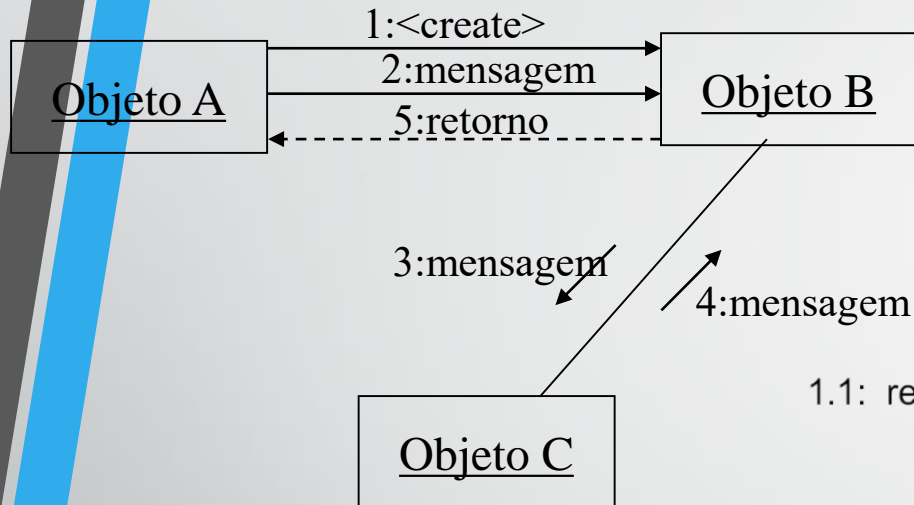


Diagrama AUML

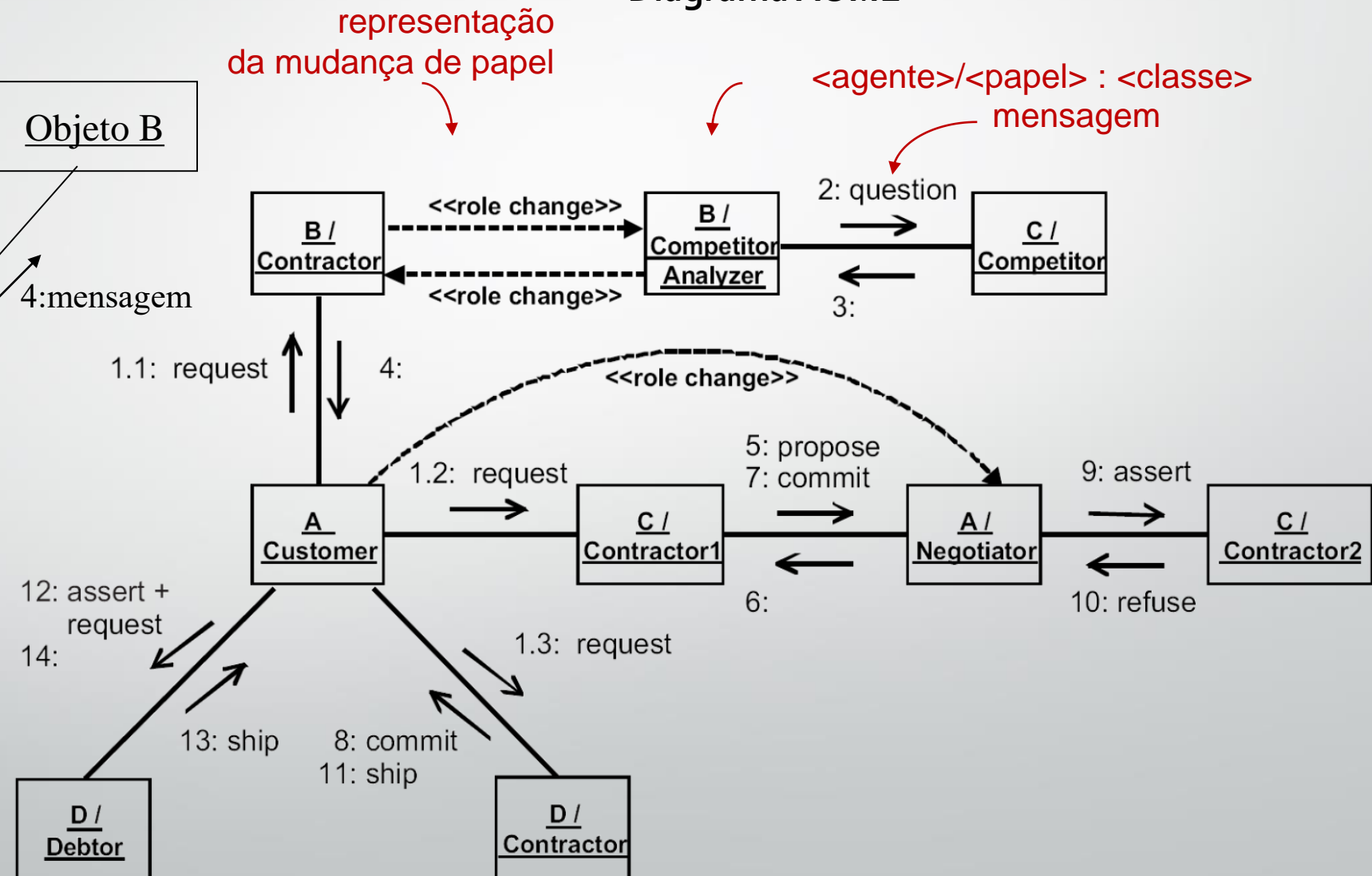


Diagrama de Actividades UML

- Utilizado para representar as atividades associadas a um protocolo ou às atividades de um papel / role
- Útil para protocolos de interação complexos que envolvem processamento simultâneo

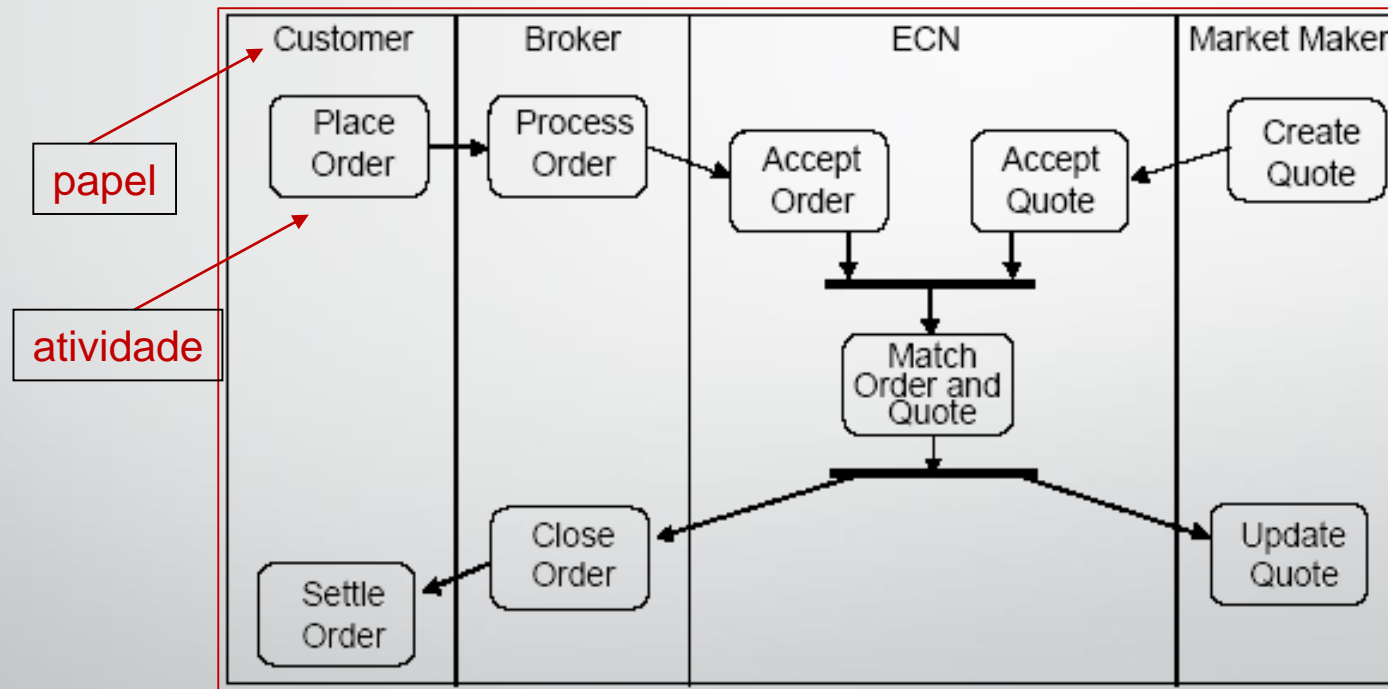


Diagrama de Actividades AUML (Exemplo)

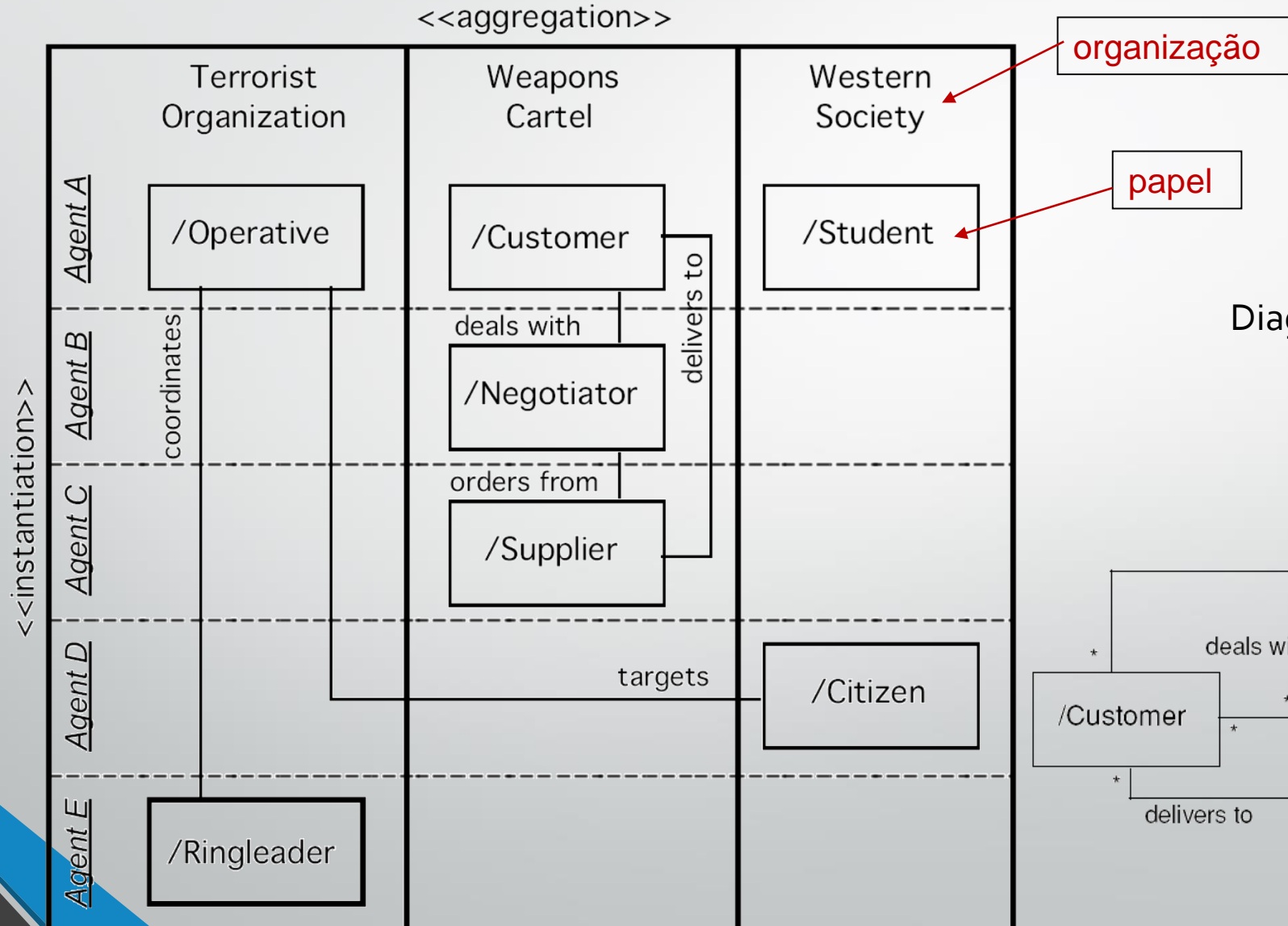


Diagrama de Classes

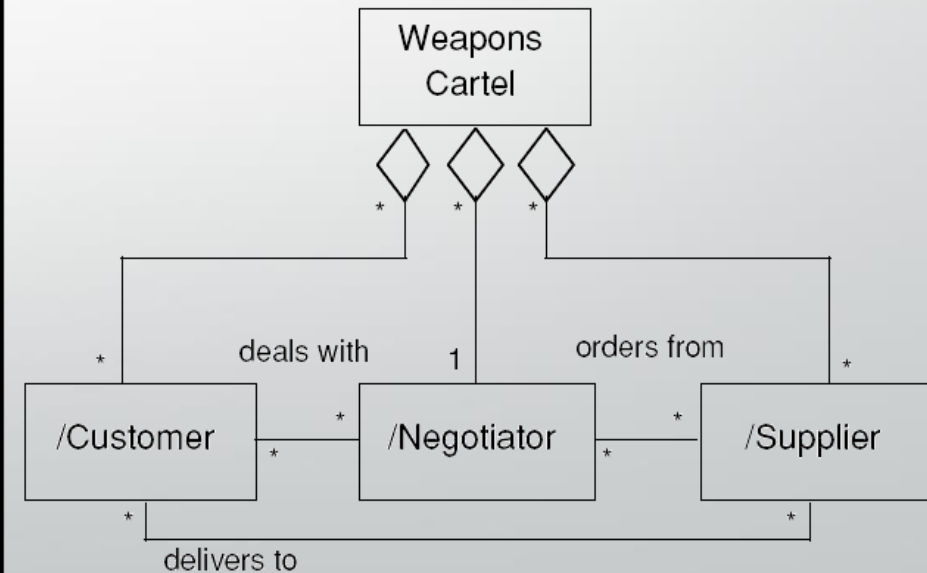


Diagrama de Estado UML

- Usado para representar os estados do sistema e as suas transições

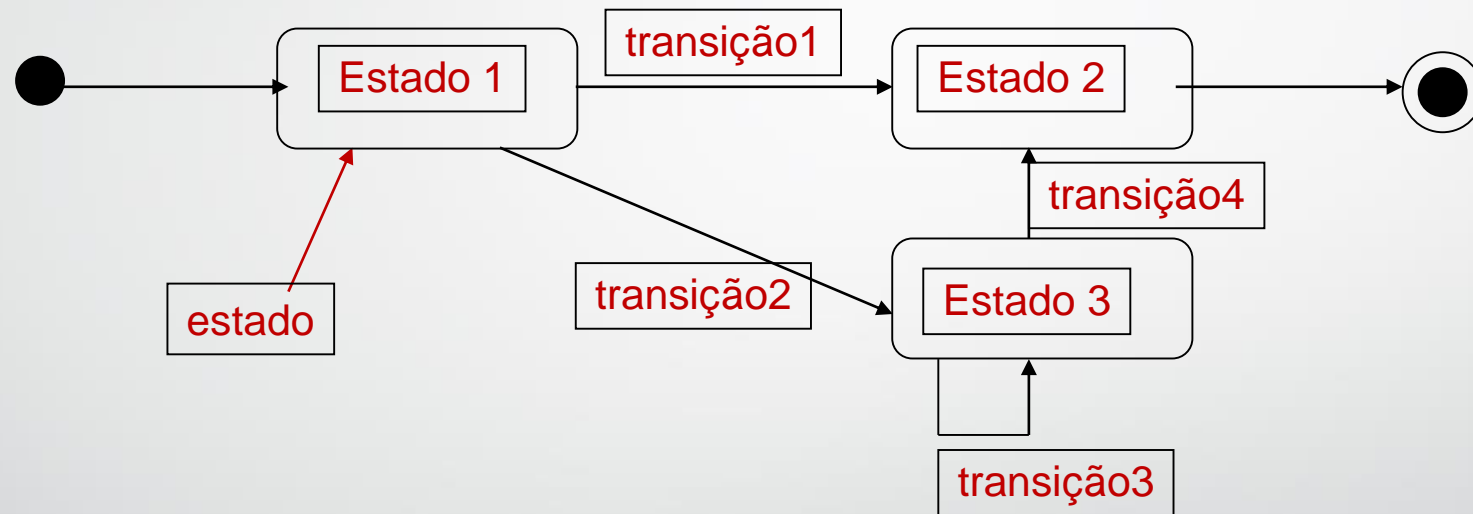
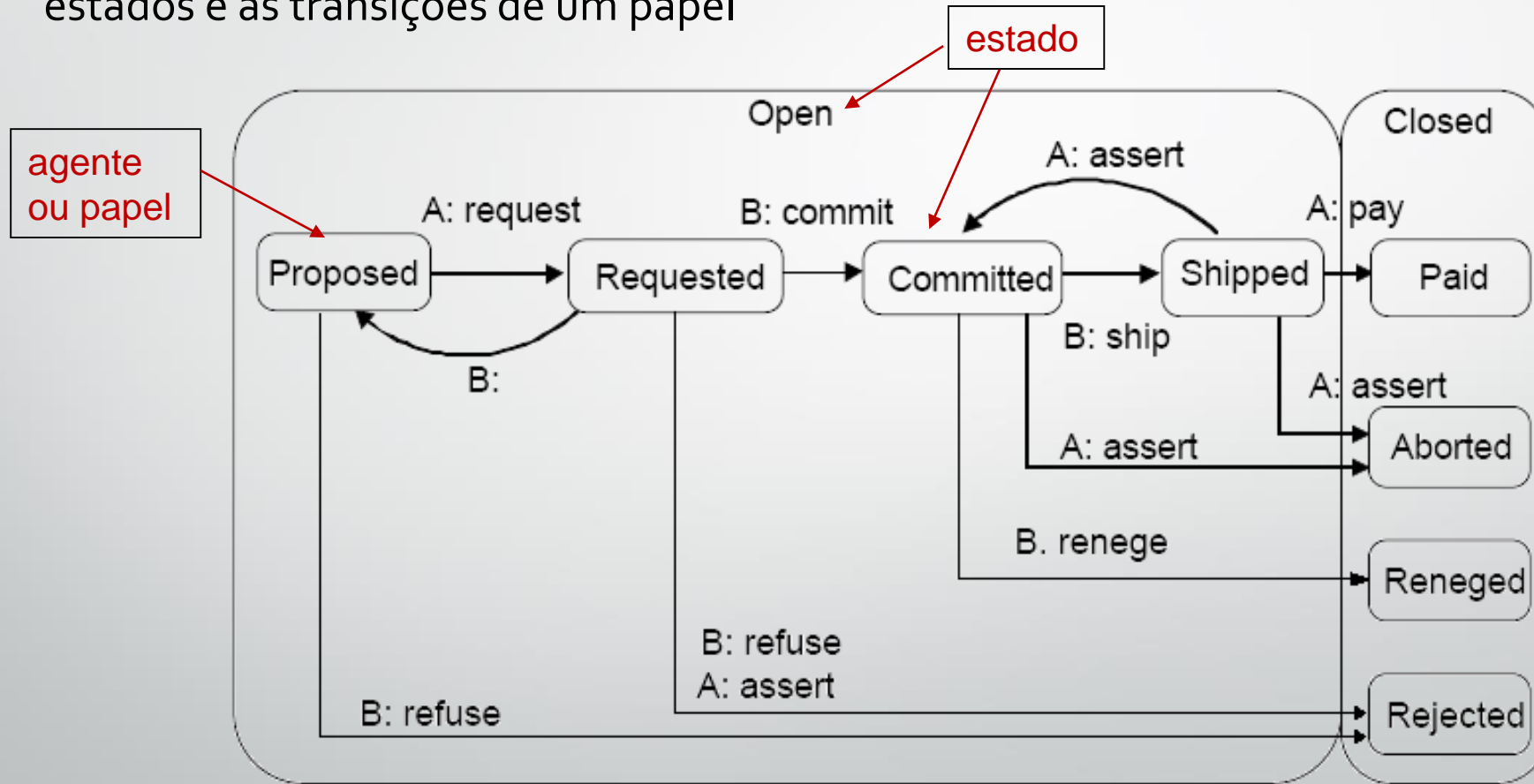


Diagrama de Estado AUML

- Usado para representar os estados e as transições associados a um protocolo ou só os estados e as transições de um papel



Nível 3: Processamento interno do Agente

Nível 3: Representação do processamento interno do agente

- Exemplo:** processamento dentro do Agente "Order Processor"

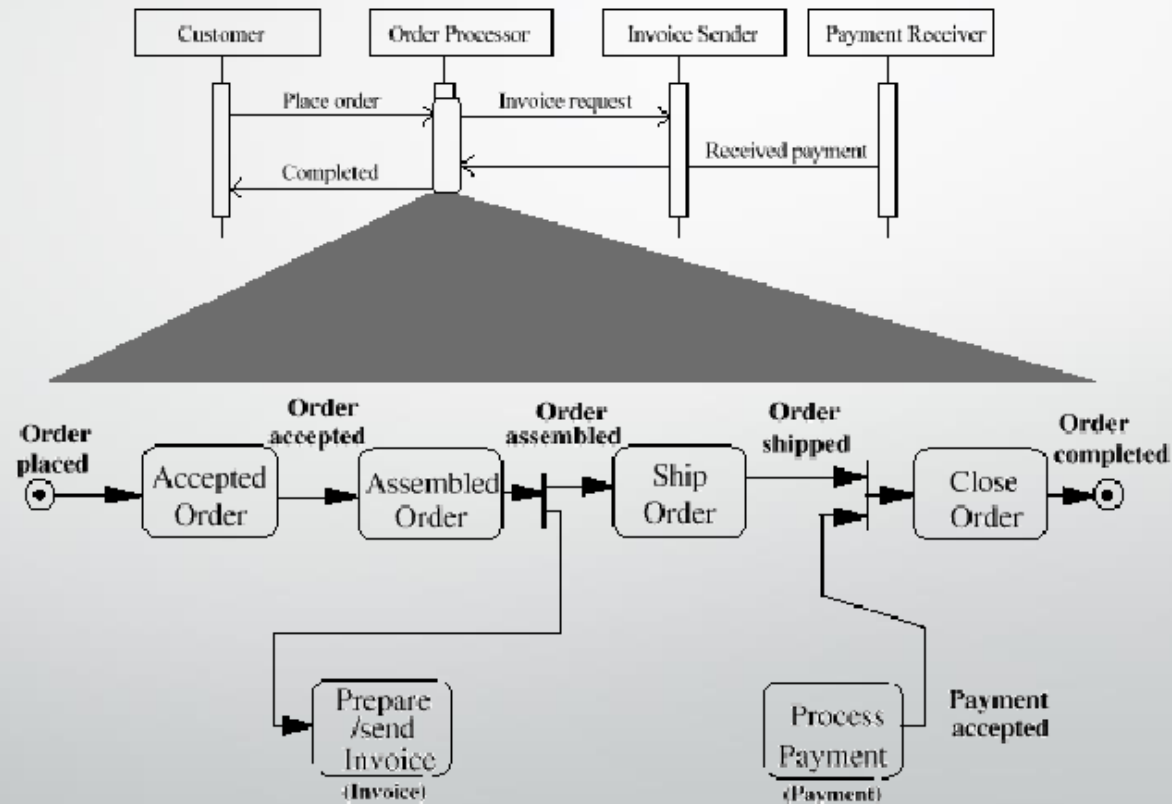


Diagrama de Classes

Diagramas de Classes usados para:

- Modelar o domínio do Problema
- Modelar a implementação das Classes

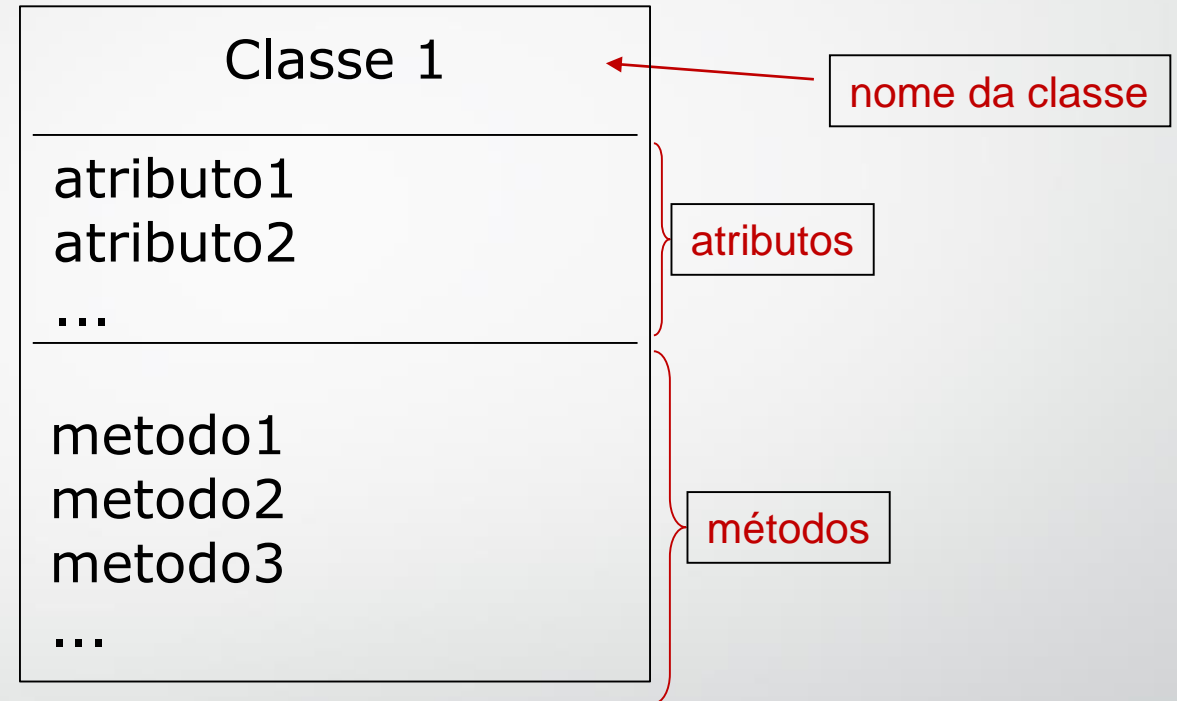


Diagrama de classes de AUML

Nome da classe de agente e os papéis que os agentes podem desempenhar

Descrição do que pode fazer

a organização onde o agente executa e as condições para entrar e sair da organização

atos de fala que podem ser utilizados nas mensagens que recebe e nas mensagens que envia

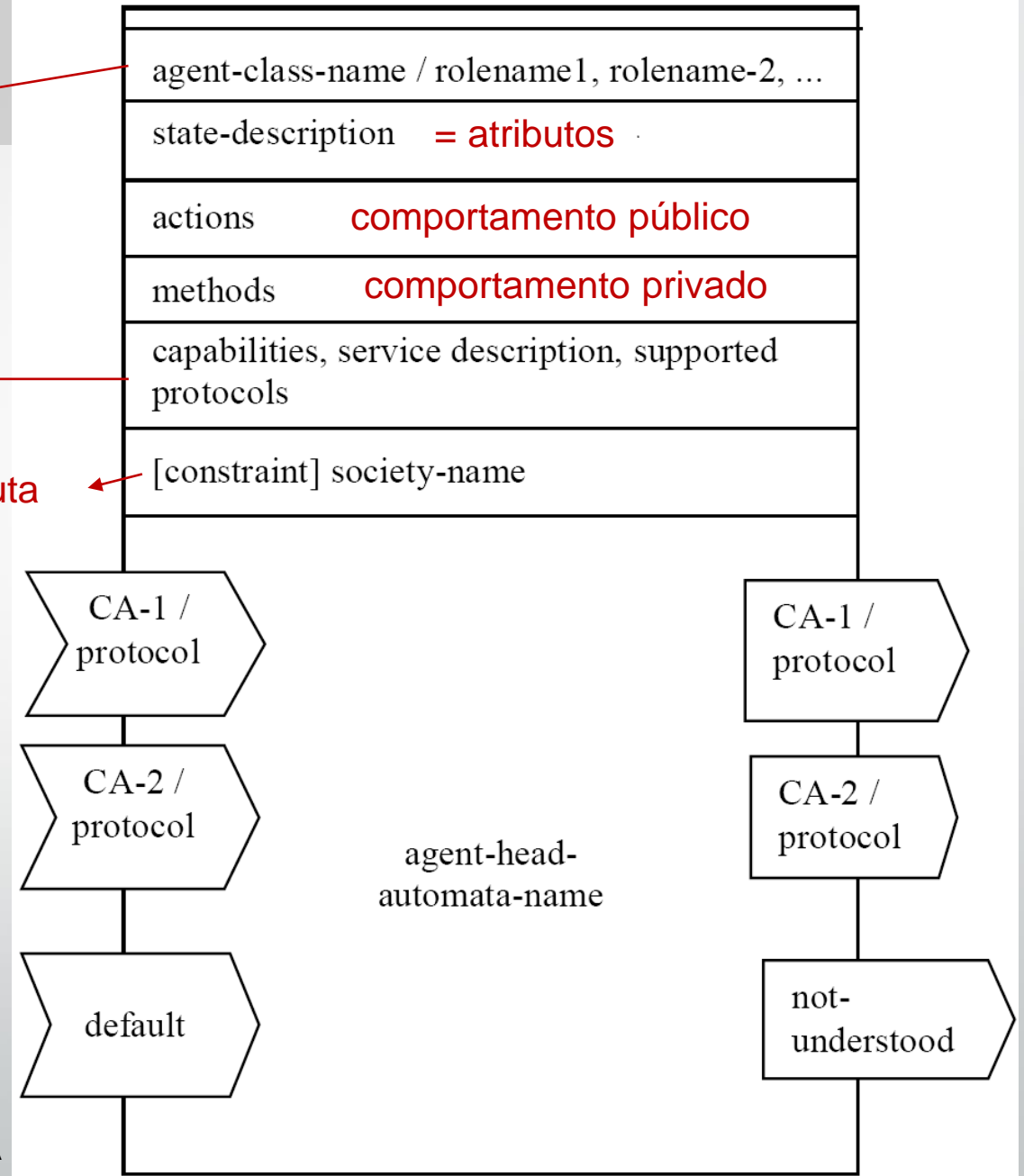


Diagrama de classes de AUML

visão de agente

visão da organização

visão do ambiente

visão de interação

visão da organização

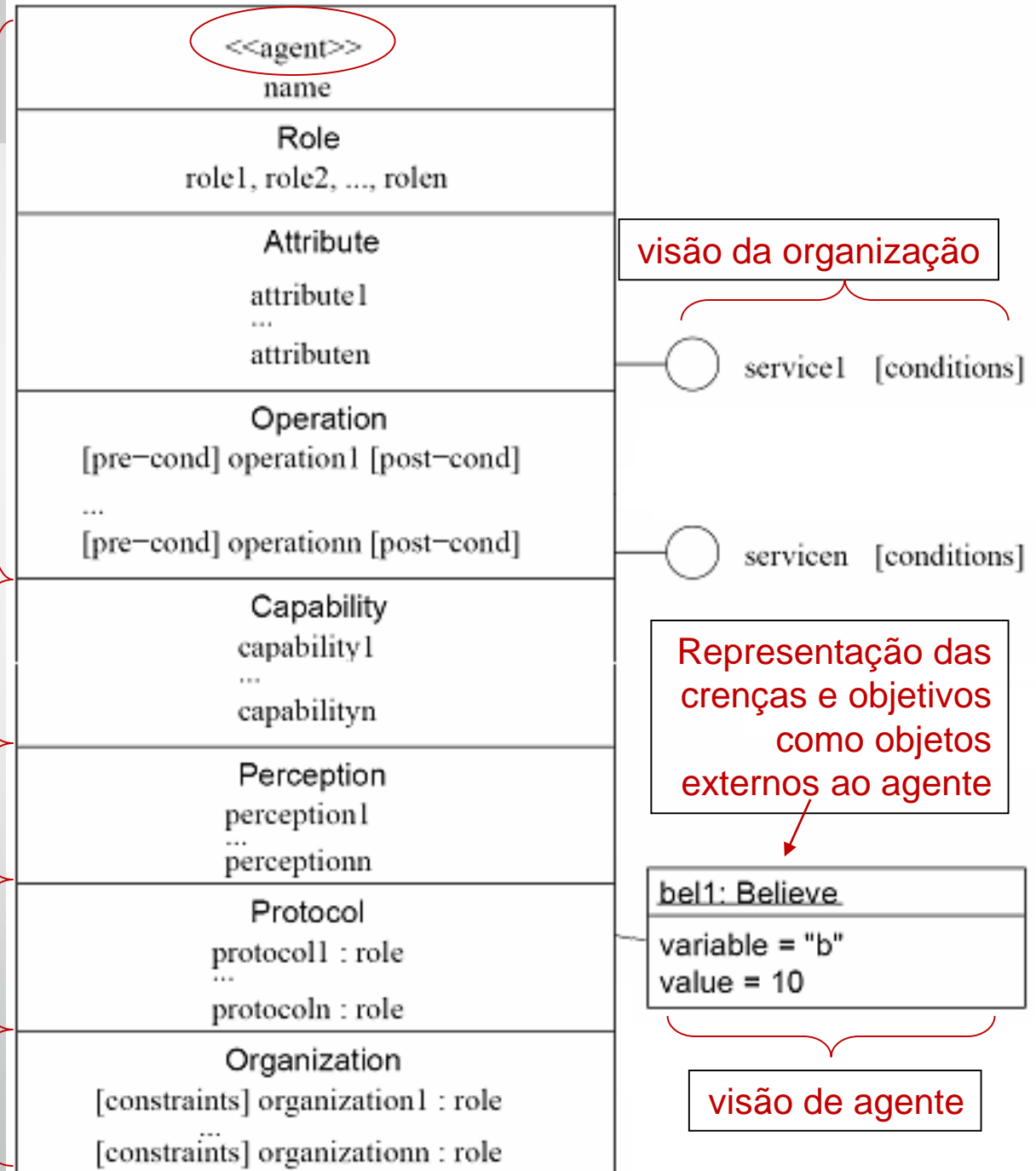
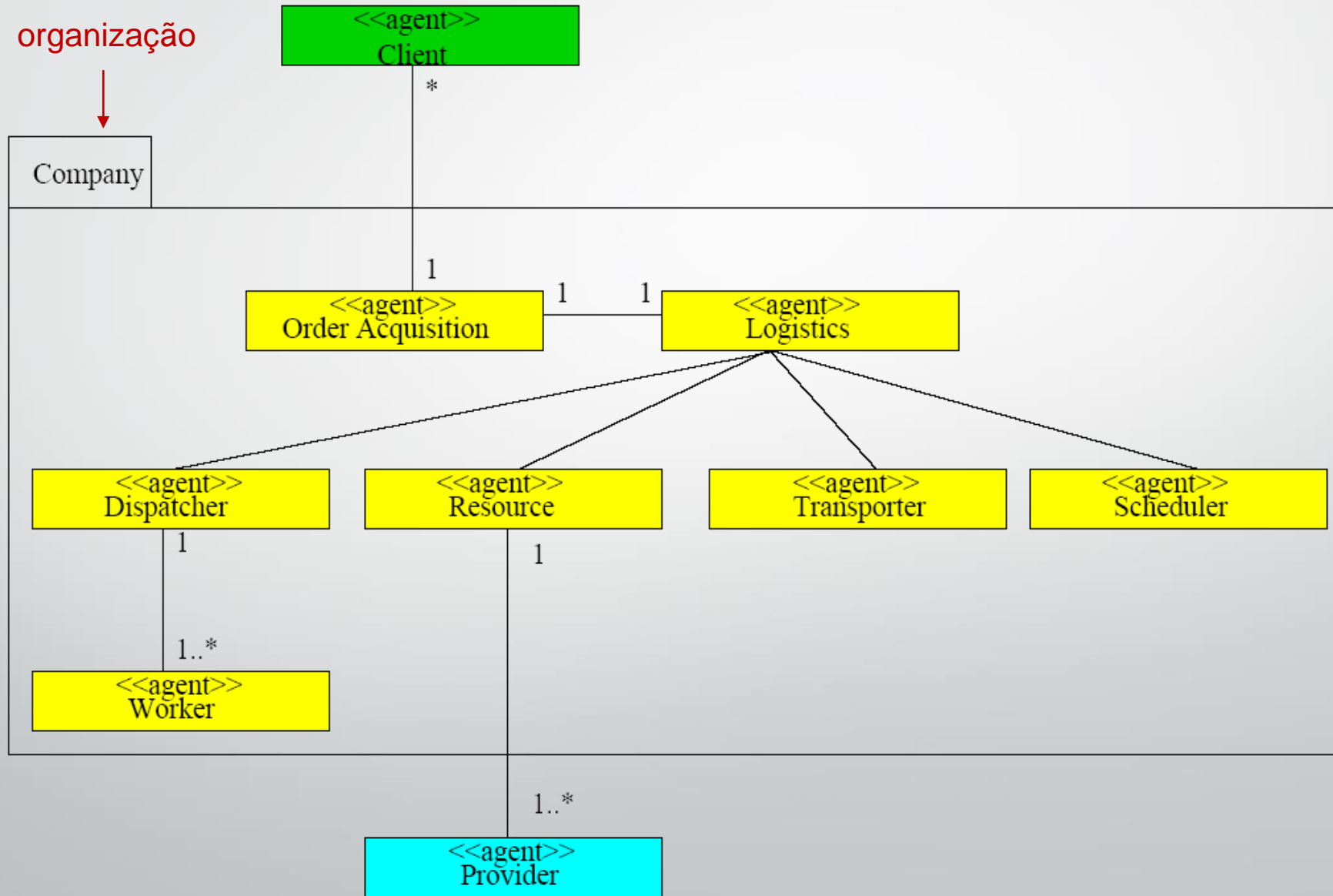


Diagrama de classes de AUML



Conclusões

Mecanismos de extensão baseados em UML fornecem formalismos para especificar interação de agentes a vários níveis:

- Especificar protocolos como um todo
- Exprimir padrões de interação entre Agentes por Protocolo
- Expressar o comportamento interno a um Agente
- Formalização da especificação dos Agentes e AIPs importante para especificação e implementação de **Sistemas Multi-agentes**

Bibliografia

- James Odell, H. Van Dyke Parunak, Bernhard Bauer. *Representing Agent Interaction Protocols in UML*, 1999.
- James Odell, H. Van Dyke Parunak, Bernhard Bauer. *Extending UML for Agents*, 2000.
- James Odell, Conrad Bock. OMG document ad/99-12-01. *Suggested UML Extensions for Agents*. Submitted to the OMG's Analysis and Design Task Force (ADTF) in response to the Request of Information (RFI) entitled "UML2.0 RFI". Dec. 1999.
- Bernhard Bauer. OMG document ad/99-12-03. *Extending UML for the Specification of Agent Interaction Protocols*. FIPA submission to the OMG's Analysis and Design Task Force (ADTF) in response to the Request of Information (RFI) entitled "UML2.0 RFI". Dec. 1999.

Linguagens de Modelação de Sistemas Multi-Agentes

Integrated Master's in Informatics Engineering

Intelligent Agents

2018/2019

Synthetic Intelligence Lab

Filipe Gonçalves

César Analide

