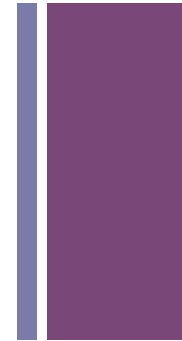


# Sistemas Multiagentes

Prof. Elder Rizzon Santos  
[ersantos@inf.ufsc.br](mailto:ersantos@inf.ufsc.br)

# + Motivação / tendências

- Ubiquidade
- Conectividade
  - Computação como um processo de interação
- Inteligência (autonomia, agentes inteligentes)
  - Complexidade das tarefas
- Delegação
  - Atribuição do controle para o sistema
  - Independência e interesses do usuário
- Abstração
  - Human-oriented programming

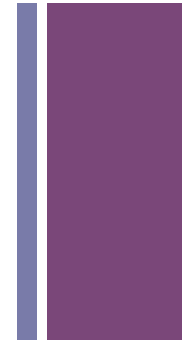


## + SMA (UMA definição)

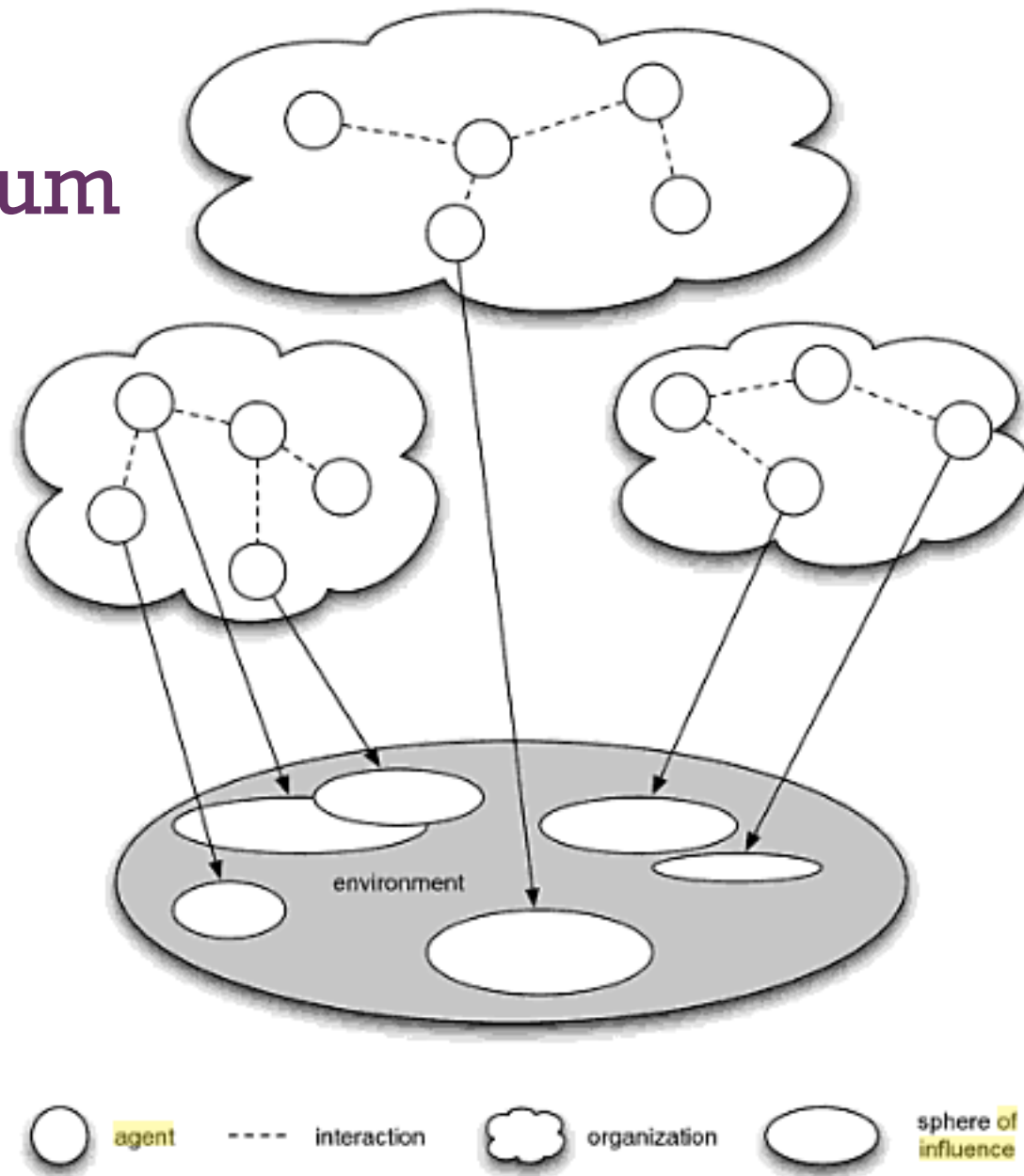
- Um sistema multiagente é um sistema constituído por mais do que um agentes, os quais **interagem** entre si.
- Generalizando, agentes atuam em prol de seus usuários com objetivos e motivações **distintas**.
- Para interagir com sucesso, os agentes precisam saber cooperar, coordenar, negociar e argumentar uns com os outros.
- Habilidade **social**

## + SMA (outra definição)

- É uma rede fracamente acoplada de solucionadores de problemas que **trabalham em conjunto** para resolver problemas que **vão além** da sua capacidade individual.
- Estes solucionadores de problemas são essencialmente autônomos, distribuídos e, muitas vezes heterogêneos em sua natureza.



# Estrutura Típica de um MAS

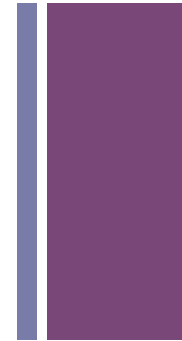


# + Comunicação

Cooperação	Estratégias			
	Protocolos			
Comunicação	Linguagens de comunicação			
	Quadro de avisos	Difusão de mensagens	Federação	Mensagens diretas
	Transporte			

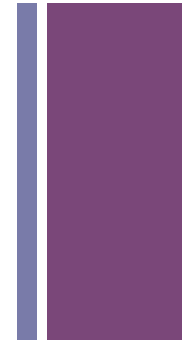
# + Aspectos (sub-áreas de pesquisa) de um SMA

- Comunicação
- Estratégias:
- Negociação
  - Argumentação
- Cooperação
  - Coordenação
- Resolução de Conflitos
- Comprometimento



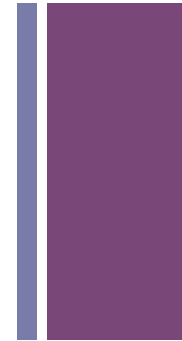
# + Cooperação

- Um agente sozinho em geral não possui conhecimento e recursos suficientes para resolver um problema.
  - Uma abordagem é fragmentar o problema em sub-problemas ou sub-tarefas
- Como dividir em sub-tarefas?
- Como unir e sincronizar os resultados das sub-tarefas
- Como maximizar a eficiência (evitando interações desnecessárias)?
  - Agentes podem ter objetivos distintos/contraditórios
  - Atividades de cooperação, coordenação e negociação devem ser realizadas dinamicamente (em tempo real).
  - Teoria de jogos





# + Comunicação



## ■ ComunicAÇÃO

- Atos comunicativos: visam influenciar outros agentes de acordo com as necessidades
- Ag1: está chovendo em Floripa! (o que ag1 deseja com isto?)
- Ag2: acredito no ag1 e mudo minhas crenças?

## ■ Teoria dos atos de fala (John Austin, 1962)

- Comunicação como ação
- Toda ação que é realizada através da fala, do dizer.
- Ato locucionário: comunicar, dizer algo
- Ato ilocucionário: a ação realizada através do locucionário (informar, reclamar, avisar, advertir, etc.)
- Ato perlocucionário: os efeitos que deseja-se provocar nos ouvintes – intenção (convencer, levar a uma decisão etc.)

# + Linguagens de Comunicação entre Agentes

- KQML (Knowledge Query Manipulation Language)  
desenvolvida pelo KSE (Knowledge Sharing Effort), 1990
- FIPA-ACL (Foundation for Intelligent Physical Agents – Agent Communication Language) - 1999
- A maioria é baseada na teoria dos atos de fala

■ Ex. Msg. KQML:

```
(ask-one
  :content (PRICE IBM ?price)
  :receiver stock-server
  :language LPROLOG
  :ontology NYSE-TICKS
)
```

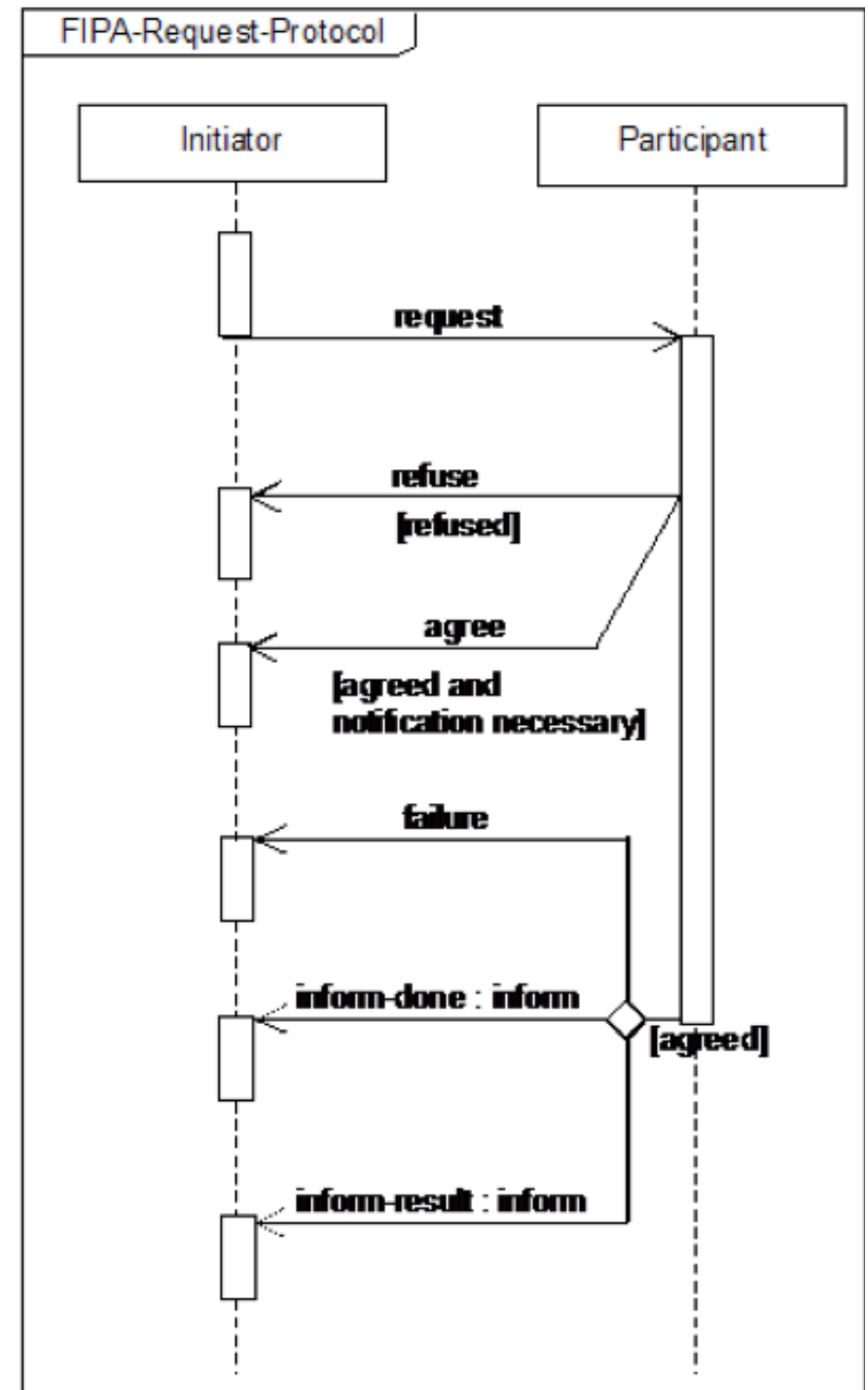
# + Algumas performativas KQML

Performative	Meaning
achieve	<i>S</i> wants <i>R</i> to make something true of their environment
advertise	<i>S</i> claims to be suited to processing a performative
ask-about	<i>S</i> wants all relevant sentences in <i>R</i> 's VKB
ask-all	<i>S</i> wants all of <i>R</i> 's answers to a question <i>C</i>
ask-if	<i>S</i> wants to know whether the answer to <i>C</i> is in <i>R</i> 's VKB
ask-one	<i>S</i> wants one of <i>R</i> 's answers to question <i>C</i>
break	<i>S</i> wants <i>R</i> to break an established pipe
broadcast	<i>S</i> wants <i>R</i> to send a performative over all connections
broker-all	<i>S</i> wants <i>R</i> to collect all responses to a performative
broker-one	<i>S</i> wants <i>R</i> to get help in responding to a performative
deny	the embedded performative does not apply to <i>S</i> (anymore)
delete-all	<i>S</i> wants <i>R</i> to remove all sentences matching <i>C</i> from its VKB
delete-one	<i>S</i> wants <i>R</i> to remove one sentence matching <i>C</i> from its VKB
discard	<i>S</i> will not want <i>R</i> 's remaining responses to a query
eos	end of a stream response to an earlier query
error	<i>S</i> considers <i>R</i> 's earlier message to be malformed
evaluate	<i>S</i> wants <i>R</i> to evaluate (simplify) <i>C</i>
forward	<i>S</i> wants <i>R</i> to forward a message to another agent
generator	same as a standby of a <code>stream-all</code>
insert	<i>S</i> asks <i>R</i> to add content to its VKB
monitor	<i>S</i> wants updates to <i>R</i> 's response to a <code>stream-all</code>
next	wants <i>R</i> 's next response to a previously streamed performative

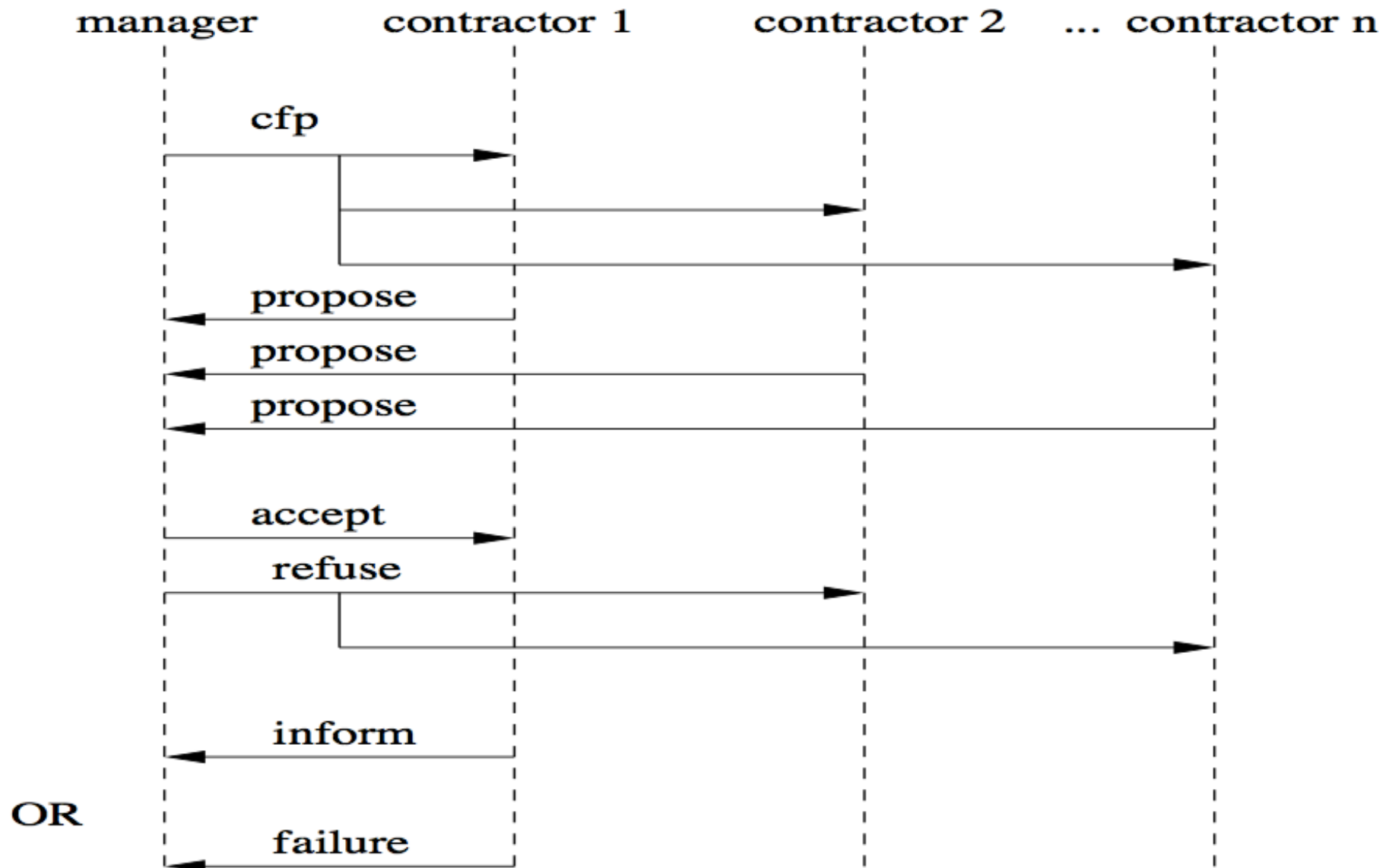
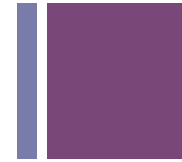
# + Exemplo Ato de Fala FIPA-ACL: Request

<b>Summary</b>	<p>The sender requests the receiver to perform some action.</p> <p>One important class of uses of the request act is to request the receiver to perform another communicative act.</p>
<b>Message Content</b>	An action expression.
<b>Description</b>	<p>The sender is requesting the receiver to perform some action. The content of the message is a description of the action to be performed, in some language the receiver understands. The action can be any action the receiver is capable of performing, for example, pick up a box, book a plane flight, change a password, etc.</p> <p>An important use of the request act is to build composite conversations between agents, where the actions that are the object of the request act are themselves communicative acts such as inform.</p>
<b>Formal Model</b>	$\langle i, \text{request}(j, a) \rangle$ $\text{FP: FP}(a) [i \setminus j] \dot{\cup} B_i \text{ Agent}(j, a) \dot{\cup} \emptyset B_i I_j \text{ Done}(a)$ $\text{RE: Done}(a)$ <p><math>\text{FP}(a) [i \setminus j]</math> denotes the part of the FPs of <math>a</math> which are mental attitudes of <math>i</math>.</p>
<b>Examples</b>	<p>Agent / requests / to open a file.</p> <pre>(request :sender (agent-identifier :name i) :receiver (set (agent-identifier :name j)) :content   "open \"db.txt\" for input" :language vb)</pre>

+ FIPA-ACL:  
Protocolos de  
Interação  
ex: FIPA-Request



## + FIPA-Contract-Net

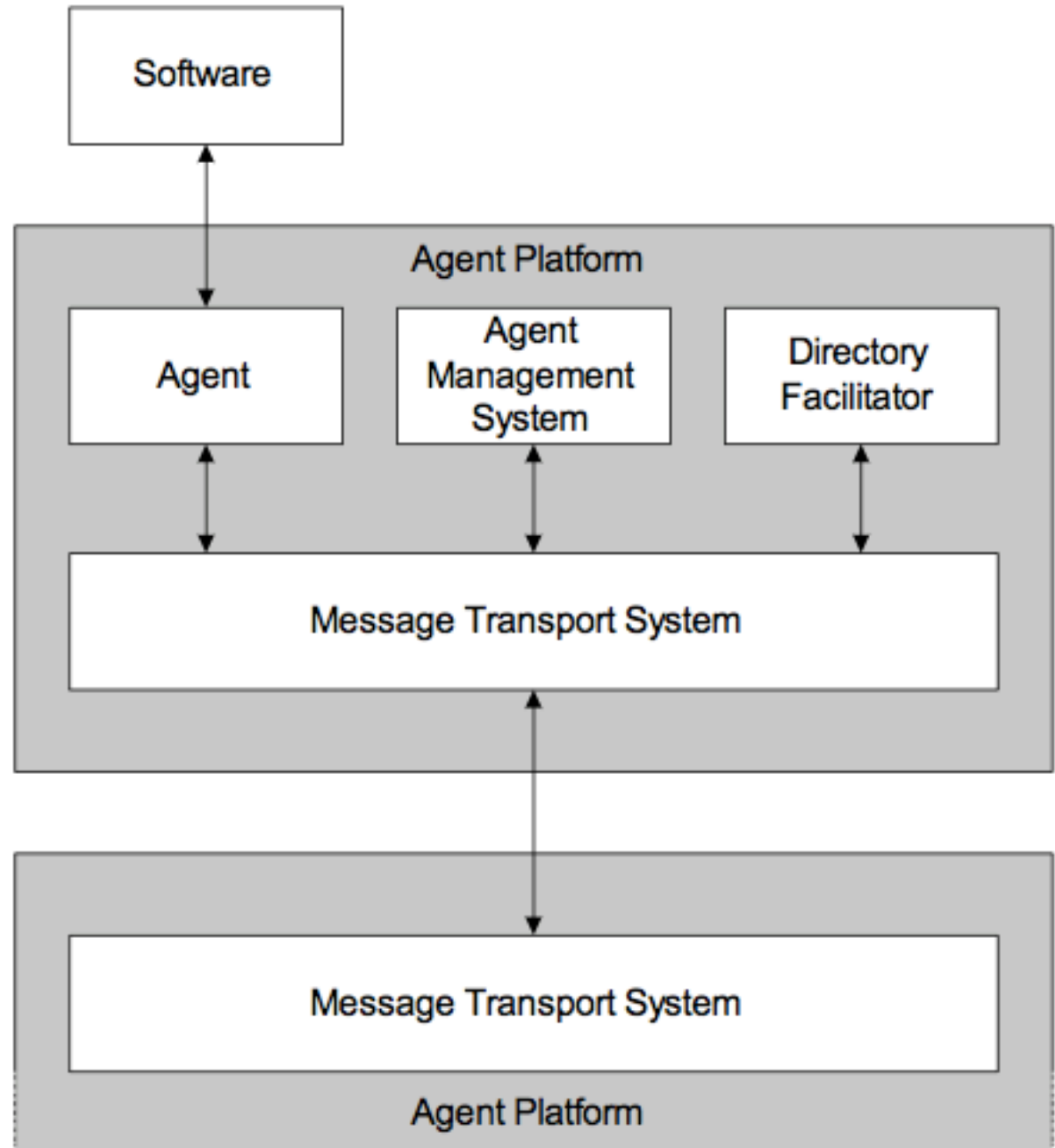


# + Elementos de uma mensagem FIPA-ACL

Parameter	Category of Parameters
performative	Type of communicative acts
sender	Participant in communication
receiver	Participant in communication
reply-to	Participant in communication
content	Content of message
language	Description of Content
encoding	Description of Content
ontology	Description of Content
protocol	Control of conversation
conversation-id	Control of conversation
reply-with	Control of conversation
in-reply-to	Control of conversation
reply-by	Control of conversation

## + FIPA: Um conjunto de padrões

- Interoperabilidade entre agentes e plataformas heterogêneas
- [fipa.org](http://fipa.org)
- Implementação: [jade.tilab.com](http://jade.tilab.com)





# + FIPA: Ciclo de vida

