



**Implementação de uma livraria
virtual utilizando agentes BDI
através da linguagem
AgentSpeak(L)**

Daniel Dalcastagne
Orientando

Prof. Dr. Jomi Fred Hübner
Orientador

Roteiro

- Introdução e Objetivos
- Fundamentação Teórica
- Especificação
- Implementação
- Apresentação do Sistema
- Conclusões
- Extensões

Introdução

Introdução e Objetivos

■ Paradigmas:

■ Imperativo

(Comandos e procedimentos)

■ Orientado a Objetos

(Representa o mundo em classes)

■ Funcional

(Funções)

■ Lógico

(Interferência lógica)

Introdução

Introdução e Objetivos

- **Sistemas Multiagentes (SMA)**
 - Novo paradigma
 - Surgiu no final da década de 90
 - Linguagem: AgentSpeak(L)
 - Poucas aplicações desenvolvidas

Objetivo

Introdução e Objetivos

- Realizar um estudo da viabilidade da implementação de um sistema computacional convencional
- Orientada a agentes
- Utilizando a linguagem AgentSpeak(L), através do interpretador Jason

Motivação

Introdução e Objetivos

- Poucas aplicações foram desenvolvidas em AgentSpeak(L)
- Requer um trabalho mais aprofundado de experimentação
- Possibilitar se torne uma linguagem de programação de uso mais amplo

Sistemas Multiagentes (SMA)

Fundamentação Teórica

- Conceber problemas de maneira que se aproximem da realidade
- Surge da observação de comportamentos sociais
- Organização a partir da interação de seus elementos
- Exemplo da Colônia de Formigas



Sistemas Multiagentes (SMA)

Fundamentação Teórica

- Componente indispensável: “O AGENTE”

“...pode-se definir um agente como uma entidade, que está inserida em um ambiente, podendo perceber, agir, ser persistente nas suas metas e comunicar-se com outros agentes assumindo comportamentos autônomos (PADGHAM; WINIKOFF, 2004, p. 3).”

Arquitetura BDI

Fundamentação Teórica

- Baseado em três atitudes (*beliefs, desires e intentions*)
- Crença: (“A porta esta fechada”) é uma crença sobre a porta
- Desejo: (“Entrar na sala”) é um desejo, do agente para o mundo
- Intenções: (Processo de raciocínio) determinam as ações a serem realizadas

A linguagem AgentSpeak(L)

Fundamentação Teórica

- Projetada para a programação de agentes BDI
- Sintaxe fortemente baseada em programas escritos utilizando o paradigma de lógica (Prolog)

```
?- [X|Y] = [4].  
X = 4  
Y = []
```

Código Prolog

A linguagem AgentSpeak(L)

Fundamentação Teórica

Um plano é formado por um conjunto de condições de ativador (denotadas por Φ)

Conjunto de crenças (devem ser verdadeiras para o plano ser considerado aplicável)

```
+!userLogon(Username, Password)
:   userPassword(Username, Password, UserCode) &
  clientBookstore(UserCode, Name, Address) &
  userProfile(UserCode, ProfileCode)
<-  welcome(Name);
     !showBookProfile(UserCode).
```

Seqüência de ações básicas ou subobjetivos que o agente deve atingir

O Interpretador Jason

Fundamentação Teórica

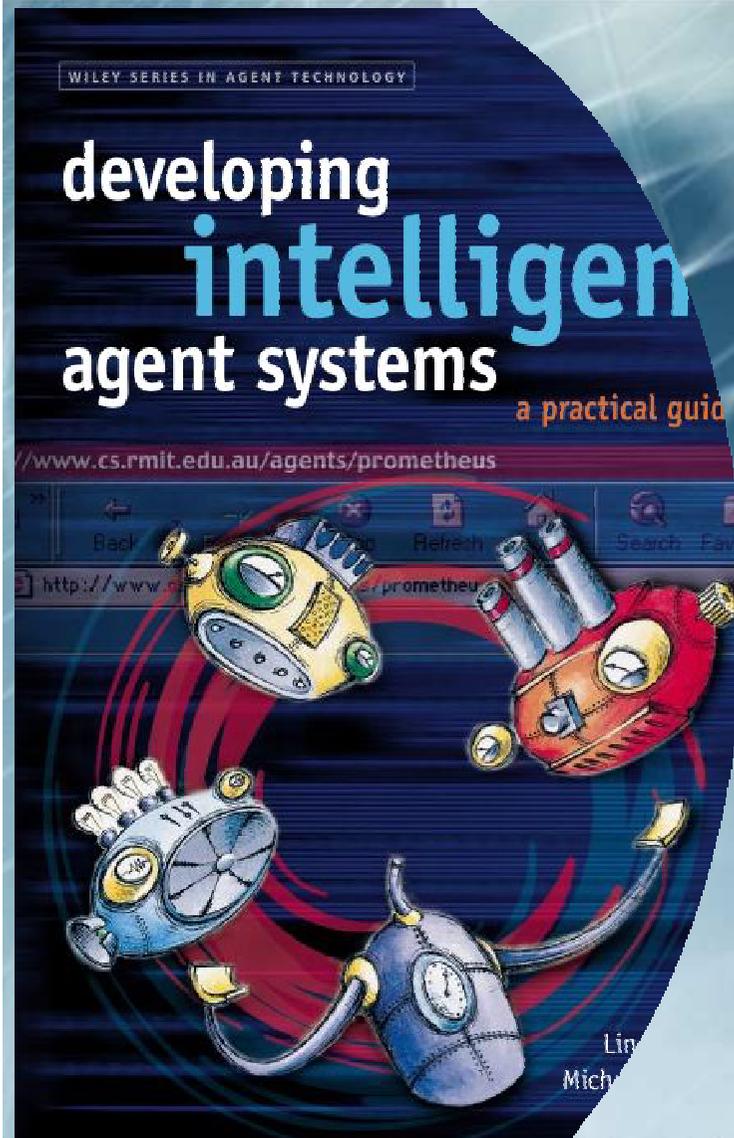


Jason
Oil on canvas,
Musée d'Orsay, Paris

- Interpretador multi-plataforma para a linguagem AgentSpeak(L)
- Suporte para o desenvolvimento de ambientes (normalmente não é programada em AgentSpeak(L))
- Executar o SMA distribuídamente (SACI)
- Possibilitar a extensão da biblioteca de ações internas

A Metodologia Prometheus

Fundamentação Teórica



- ❑ Metodologia para especificação de SMA
- ❑ Apresentada na obra *Developing intelligent agent systems: a practical guide*
- ❑ Modelagem de uma livraria virtual (implementação em Jack)

A Metodologia Prometheus

Fundamentação Teórica

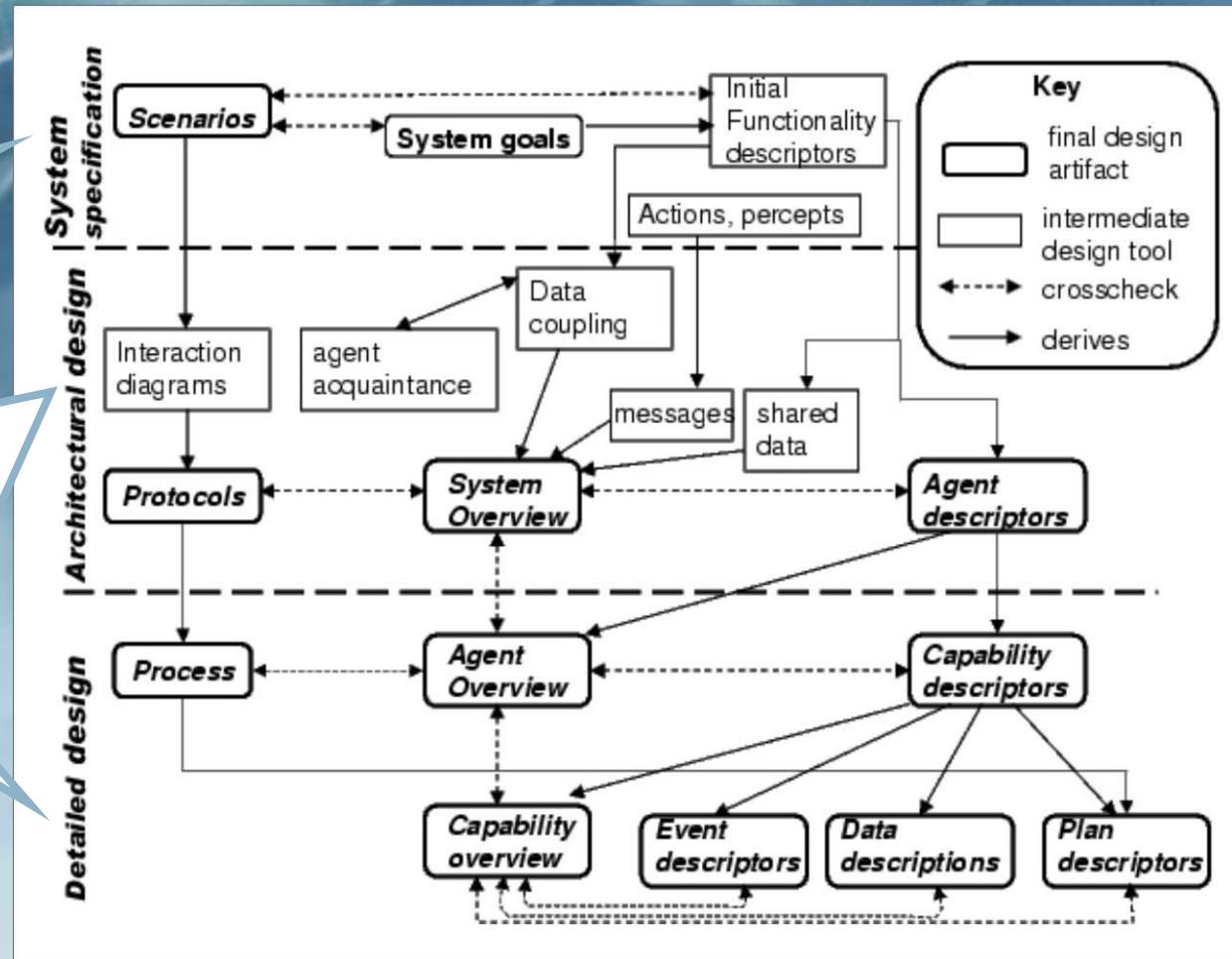
- Consiste de três fases

Determinar quais

Foca o desenvolvimento da estrutura interna de cada agente e como ele realizará sua tarefa dentro do sistema

os agentes

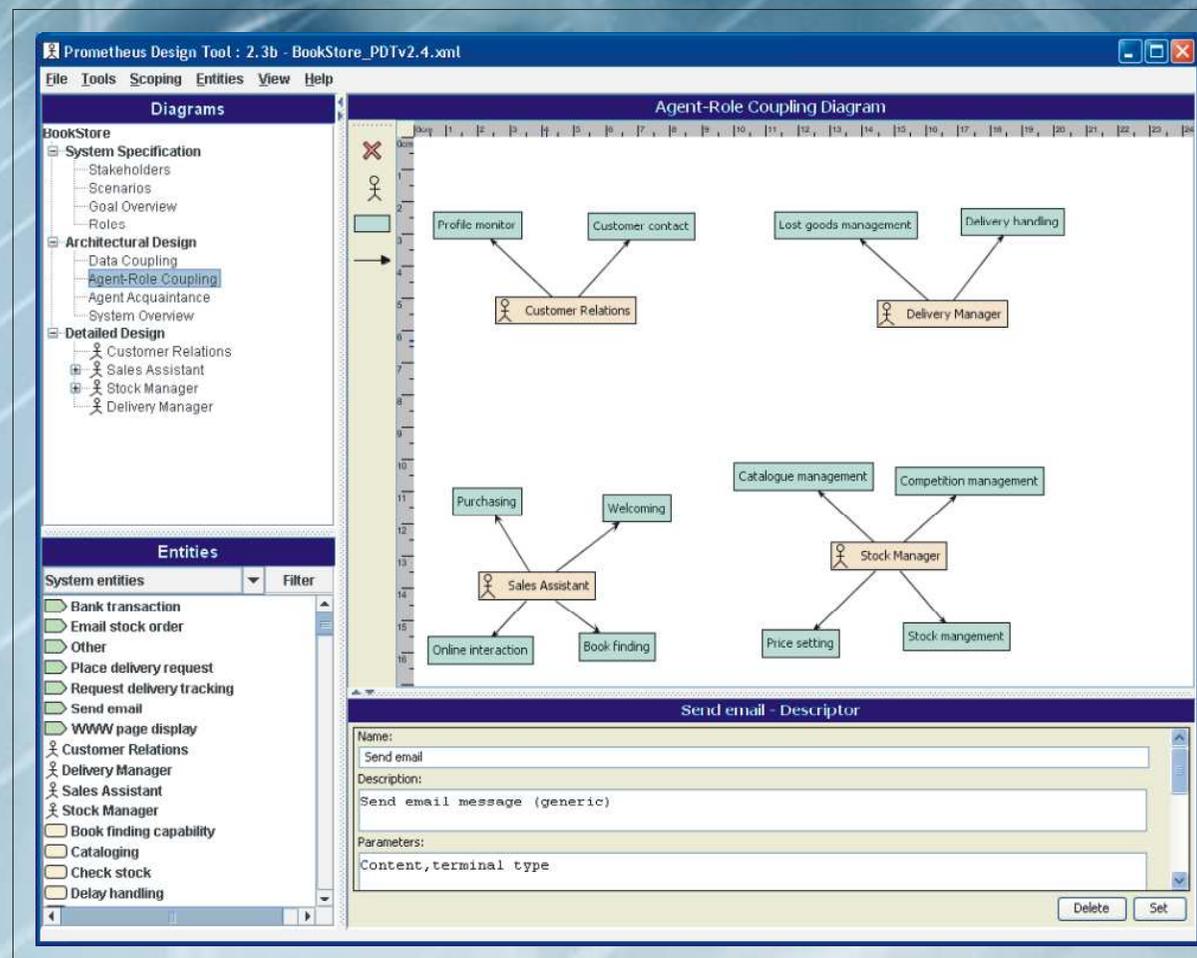
as funcionalidades e especificar as ações e percepções



A Metodologia Prometheus

Fundamentação Teórica

■ Prometheus Design Tool (PDT)



Especificação do Sistema

Especificação

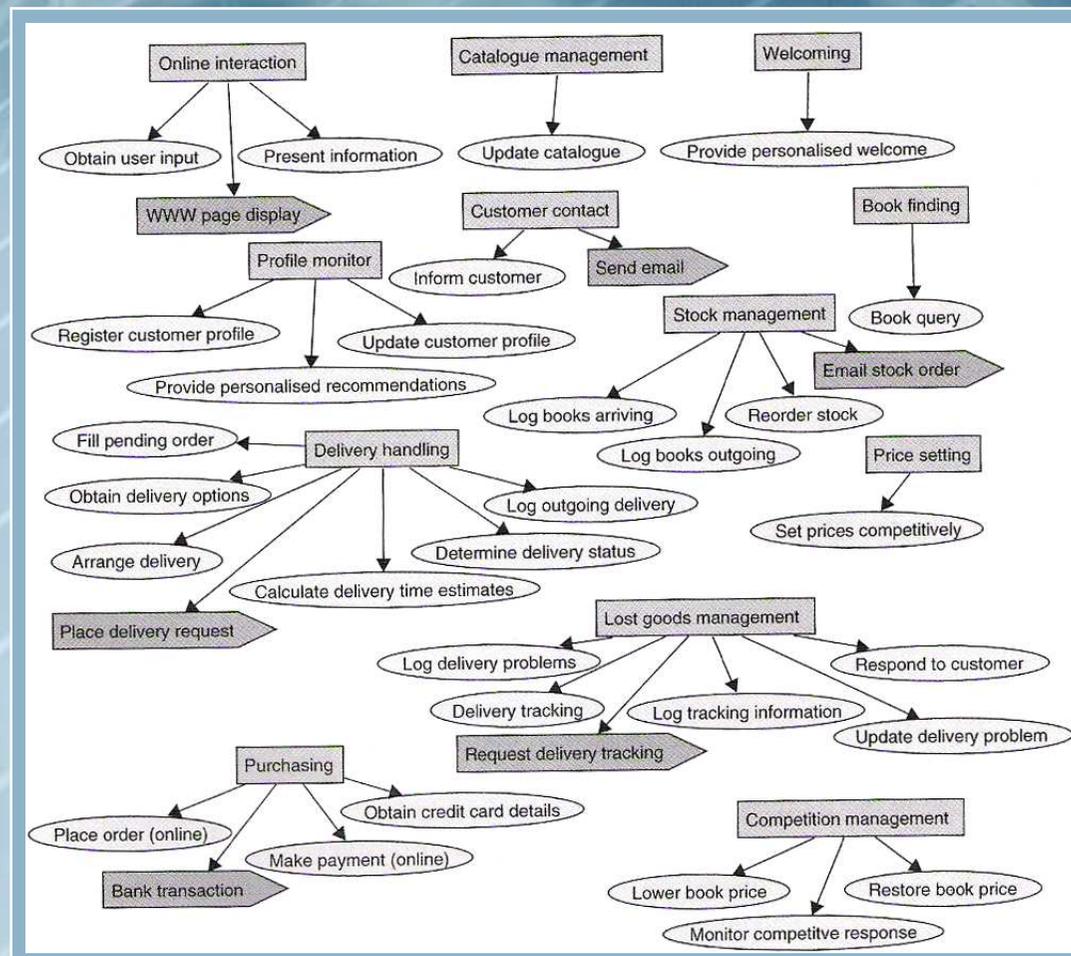
■ Funcionalidades

- Interação on-line
- Boas-vindas
- Gerenciar estoque
- Localizar livros
- Gerenciar entregas
- Ser competitivo

Especificação do Sistema

Especificação

■ Funcionalidades



Especificação do Sistema

Especificação

■ Cenários

Name	Scenario Book finding scenario						
Description	Finds book(s) as requested by the user and displays the result.						
Priority	Not Specified						
Stakeholders							
Initiated by	System						
Trigger							
Steps	#	Type	Name	Role	Descrip	Data used	Data prod
	1	Goal	Book query	Book finding		Stock DB	
	2	Goal	Present information	Online interaction		Books DB	
	3	Goal	WWW page display	Online interaction			
Variation	No books found that match request. Provide message and suggest changes to the user.						

Especificação do Sistema

Especificação

Name	Order book scenario					
Description	An order is received from WWW page interface (goal Place order).Information is obtained in order to place the order and order is placed.					
Priority	Not Specified					
Stakeholders						
Initiated by	System					
Trigger						
Steps	#	Type	Name	Role	Data used	Data prod
	1	Goal	Obtain Delivery options	Delivery handling	Courier DB, Postal DB	
	2	Goal	Calculate delivery time	Delivery handling	Courier DB, Postal DB	
	3	Goal	Present information	Online interaction		
	4	Percept	User input	Online interaction		

Especificação do Sistema

Especificação

	5	Goal	Arrange delivery	Delivery handling		
	6	Action	Place delivery request	Delivery handling		
	7	Goal	Log outgoing delivery	Delivery handling		Customer Orders
	8	Goal	Log books outgoing	Stock mangement		Stock DB
	9	Goal	Update customer record	Profile monitor		Customer DB
Variation						

Agentes

Especificação

- **Agente assistente de vendas**
(*Agent Sales Assistant*)

“Este agente negocia com o cliente através de interações on-line, de maneira análoga ao assistente real de uma loja qualquer. Isto inclui a ajuda ao cliente a procurar um livro adequado ou se necessário apurar as devidas informações.”

Agentes

Especificação

- **Agente gerente de entrega**
(*Agent Delivery Manager*)

“Organiza todos os aspectos relacionados à entrega do produto ao cliente”

Agentes

Especificação

- **Agente relações com cliente**
(*Agent Customer Relations*)

“Este agente é responsável por todas as transações off-line com o cliente .”

Agentes

Especificação

- **Agente gerente de estoque**
(*Agent Stock Manager*)

“É responsável pelos livros disponíveis na loja, assegurando que os livros estão disponíveis. Avaliar, reordenar, monitorar entregas, etc...”

Percepções

Especificação

- Chegada no website
- Entradas do usuário
- Reposição de estoque
- Preços dos competidores

Ações

Especificação

“A principal ação inerente ao website é apresentar a página com o conteúdo pertinente. Esta ação é gerada de diversas maneiras, como por exemplo, quando o agente percebe que o usuário efetuou *login* no website, uma possível ação é apresentar os livros de sua preferência.”

Projeto Detalhado - Capacidades

Especificação

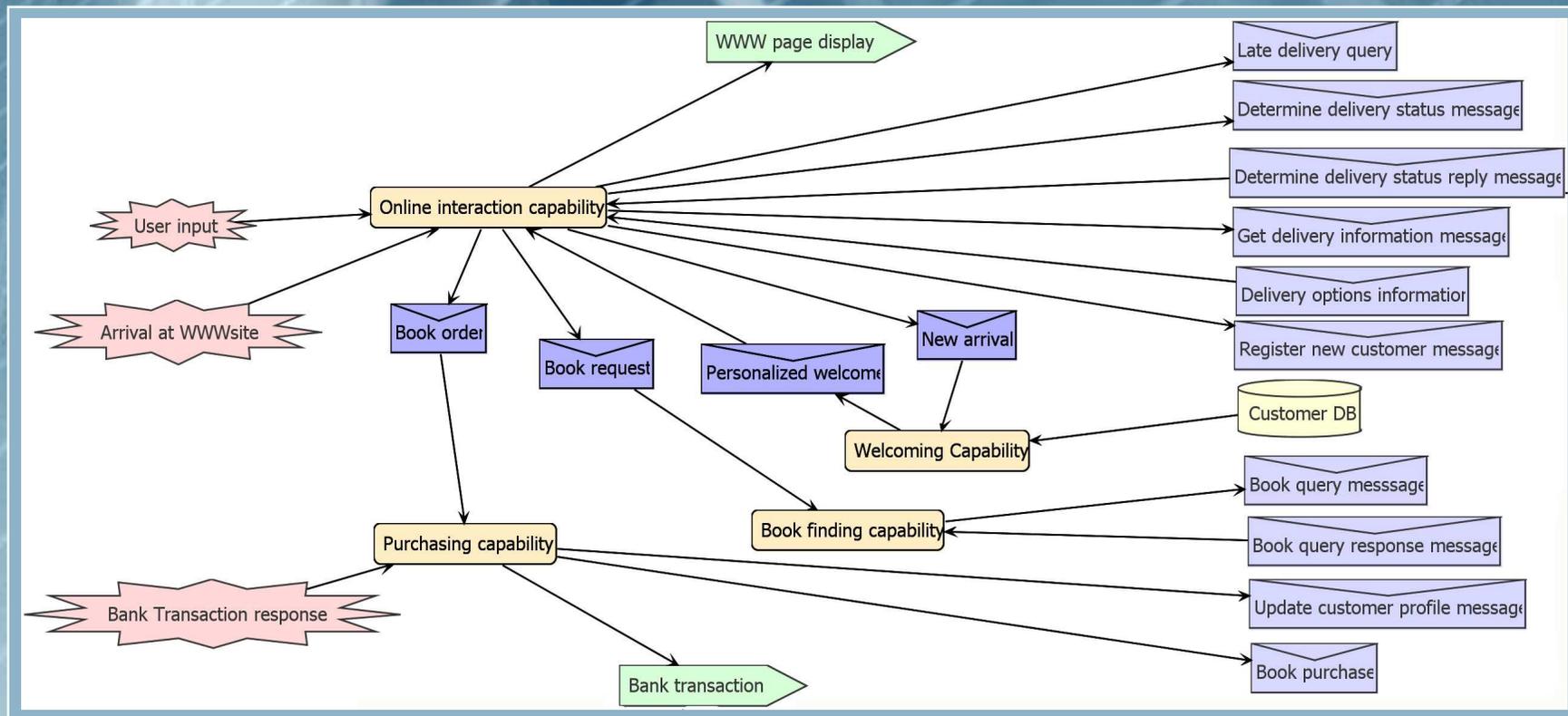
- **Agente assistente de vendas**
(*Agent Sales Assistant*)

- Boas vindas
- Interação on-line
- Localizar livros
- Compra

Projeto Detalhado - Capacidades

Especificação

■ Agente assistente de vendas (*Agent Sales Assistant*)



Projeto Detalhado - Capacidades

Especificação

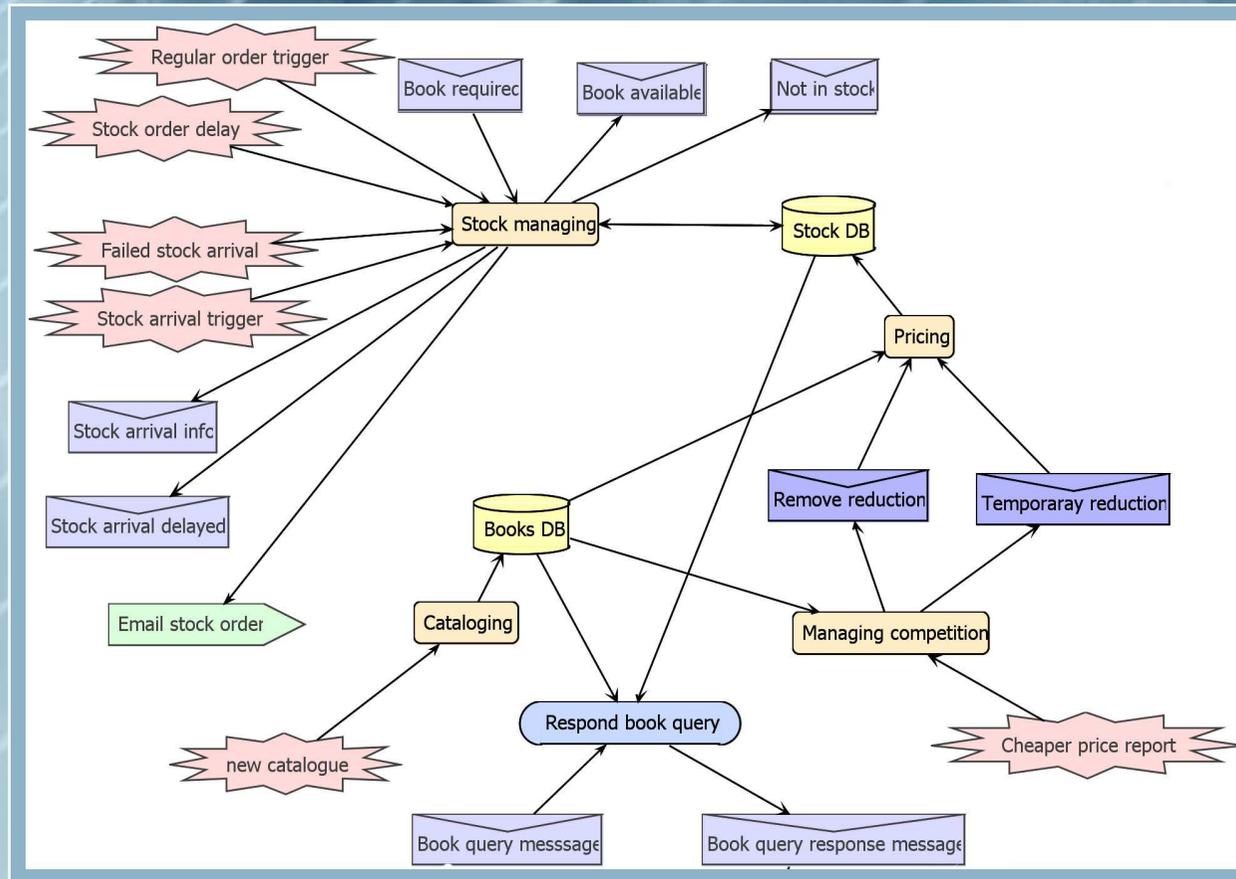
- **Agente gerente de estoque**
(*Agent Stock Manager*)

- Gerenciar o estoque
- Catalogar
- Gerenciar preços

Projeto Detalhado - Capacidades

Especificação

■ Agente gerente de estoque (*Agent Stock Manager*)



Técnicas e ferramentas utilizadas

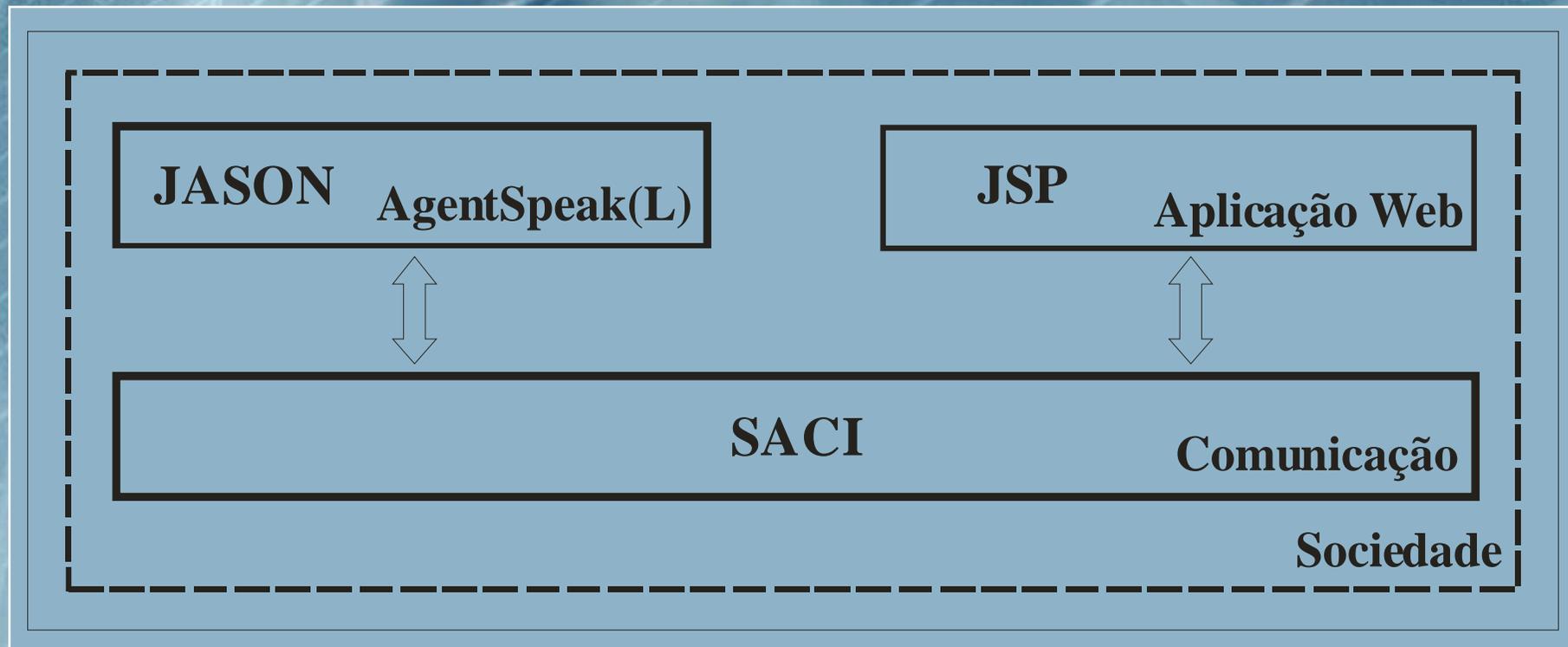
Implementação

- Jason
- Saci
- NetBeans



Camadas da Aplicação

Implementação



Implementação dos Agentes (Jason / AgentSpeak(L))

Implementação

■ Projeto em Jason

```
MAS eletronicBookstore {  
  infrastructure: Saci  
  agents:  
    deliveryManager;  
    customerRelations;  
    salesAssistant;  
    stockManager;
```

Implementação dos Agentes (Jason / AgentSpeak(L))

Implementação

■ Adição de uma crença

```
// Adicionado um novo cliente
+received(InterfaceAg, askOne, addClient(Username, Password, Name,
    Address, Uptown, City, State, Phone, EMail, RG), M)
    :
    not clientBookstore(_,Username,_,_,_,_,_,_,_)
    <-
    !generateSeqCust(Sequence);
    +clientBookstore(Sequence, Username, Password, Name,Address,
        Uptown, City, State,Phone, EMail, RG);
    .send(InterfaceAg, tell, "Congratulations! Register OK", M);
    .print("Add client: ", Name, " code: ", Sequence).

// Nome de usuario já existe. Cliente não adiciondo
+received(InterfaceAg, askOne, addClient(Username,_,_,_,_,_,_,_,_), M)
    :
    true
    <-
    .send(InterfaceAg, tell, "Invalid username", M);
    .print("Invalid username").
```

Implementação dos Agentes (Jason / AgentSpeak(L))

Implementação

■ Exemplo: Ação interna (.findall / .send)

```
+!findBook( Author, Title, Publisher, Gender)
: true
<- .findall( CodeBook, book(CodeBook, Title,
    Author, Publisher, Gender),
    ListBookRet).
```

```
+received(S, askOne, sale(CodeBook, CodeClient, QtBook, ZipCode, TotPurchase,
    Name, Address, Uptown, City, State), M)
: true
<- !generateCodeSale(Code);
    ?dayDelivery(ZipCode, Days);
    +sale(Code, CodeBook, CodeClient, QtBook, ZipCode, TotPurchase,
        Name, Address, Uptown, City, State, "Processed");
    .send(stockManager, tell, reduceStock(CodeBook,QtBook));
    .print("Sale made with success. Code ", Code, ". User code
        ",CodeClient);
    .send(S, tell, Days, M).
```

Implementação do Agente Web (Netbeans / KQML)

Implementação

■ Criação do MBox

```
<%!  
saci.MBoxSAg mbox;  
  
public void jspInit() {  
  
    System.out.println("Start page");  
    Config c = new Config();  
    c.set("society.name", "eletronicBookstore");  
    try {  
        mbox = new MBoxSAg("web",c);  
        mbox.init();  
        System.out.println("MBox created!");  
        getServletContext().setAttribute("mbox", mbox);  
    } catch (Exception e) {  
        System.err.println("Error jsp "+e);  
    }  
}  
%>
```

Implementação do Agente Web (Netbeans / KQML)

Implementação

```
(...) while (iterCB.hasNext()) {
    codeBookLis = iterCB.next().toString();
    msg         = new Message("(askOne :receiver stockManager)");
    cont        = Literal.parseLiteral("showShopWindow("+codeBookLis+)");
    msg.put("content", cont);
    answer      = mbox.ask(msg);
    content     = answer.get("content").toString();

    terms       = Term.parse(content);

    String codeBook    = terms.getTerm(0).toString().replaceAll("\\\"", "");
    String image       = "image/" + terms.getTerm(1).toString().replaceAll("\\\"", "");
    String title       = terms.getTerm(2).toString().replaceAll("\\\"", "");
    String author      = terms.getTerm(3).toString().replaceAll("\\\"", "");
    String publisher   = terms.getTerm(4).toString().replaceAll("\\\"", "");
    String price       = terms.getTerm(5).toString().replaceAll("\\\"", "");
    String gender      = terms.getTerm(6).toString().replaceAll("\\\"", "");

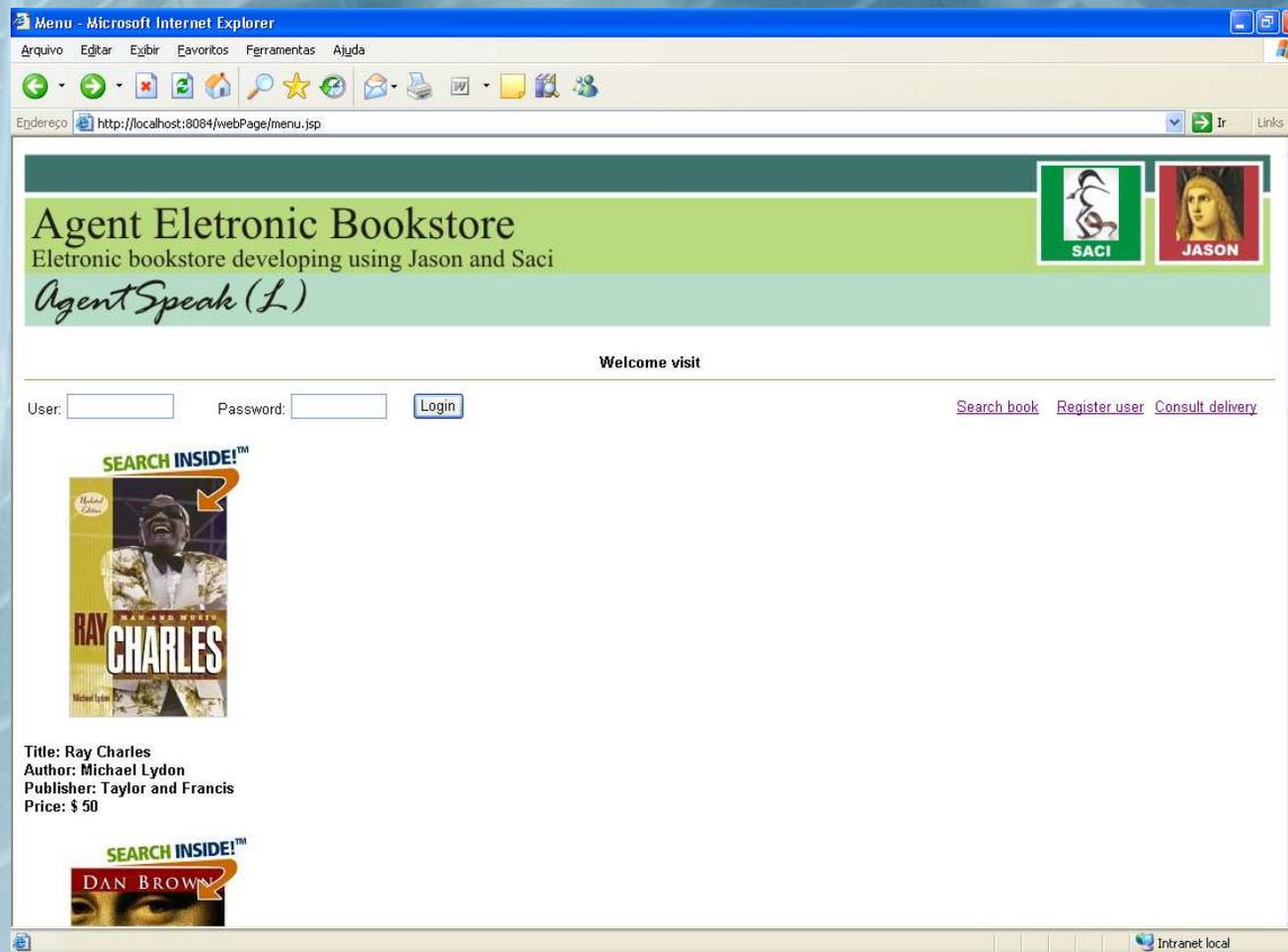
    if (!title.equals("Book not found")) { %>
        <span class="style1">
            <img src=         <%= image %>>         <p>
            <font color="#FF0000">
                <b> Code:         <%= codeBook %>         </b><br></font>
                <b> Title:        <%= title %>             </b><br>
                <b> Author:       <%= author %>            </b><br>
                <b> Publisher:    <%= publisher %>         </b><br>
                <b> Price: $      <%= price %>             </b><br>
                <b> Gender:      <%= gender %>             </b><br></p>
        }
    }
}
```

(...)

Apresentação do Sistema

Apresentação do Sistema

■ Menu principal



The screenshot shows a web browser window titled "Menu - Microsoft Internet Explorer". The address bar displays "http://localhost:8084/webPage/menu.jsp". The main content area features a green header with the text "Agent Eletronic Bookstore" and "Eletronic bookstore developing using Jason and Saci". Below this, there is a logo for "Agent Speak (L)" and two small images labeled "SACI" and "JASON". A "Welcome visit" message is centered below the header. A login form is present with fields for "User:" and "Password:" and a "Login" button. To the right of the login form are links for "Search book", "Register user", and "Consult delivery". Below the login form, there are two book covers. The first is "SEARCH INSIDE!™" by Michael Lydon, featuring a portrait of Ray Charles. Below the cover, the following information is displayed: "Title: Ray Charles", "Author: Michael Lydon", "Publisher: Taylor and Francis", and "Price: \$ 50". The second book cover is "SEARCH INSIDE!™" by Dan Brown, featuring a close-up of a face.

Apresentação do Sistema

Apresentação do Sistema

■ Comprando livro

Buy book - Microsoft Internet Explorer

Arquivo Editar Exibir Favoritos Ferramentas Ajuda

Endereço <http://localhost:8084/webPage/buyBook.jsp?codeBook=6&codeClient=1>

Agent Eletronic Bookstore
Eletronic bookstore developing using Jason and Saci
Agent Speak (L)

SACI JASON

Price

Description	Quantity	Unt price	Price
Ray Charles	1	50	50
Zip code 88320000			2.484
Total (\$)			52.484

Calculate

Delivery place

Name: Daniel Dalcastagne
Address: Rua 21 de Junho, 535
Uptown: Centro
City: Ilhota
State: SC

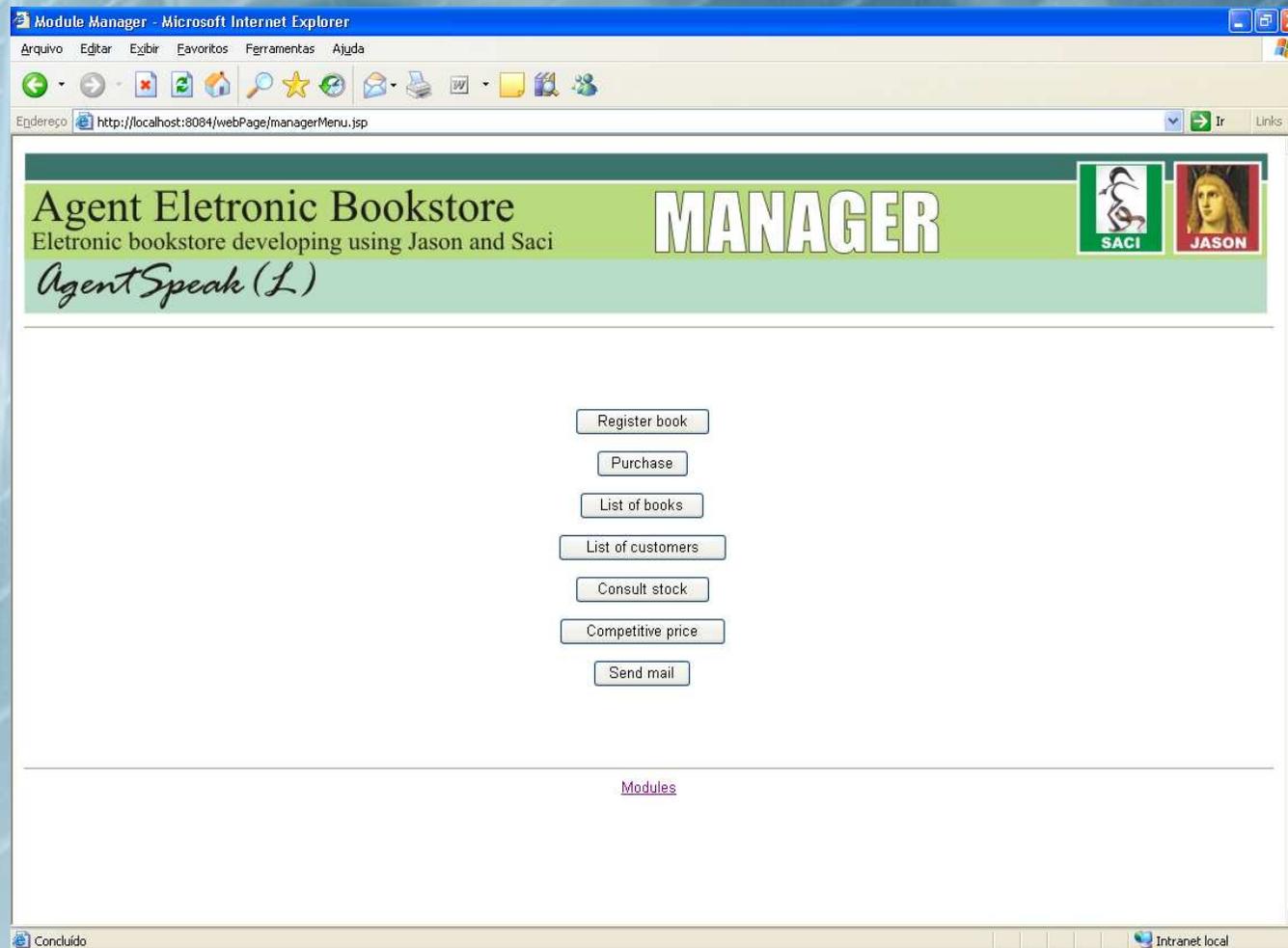
Confirm

Concluído Intranet local

Apresentação do Sistema

Apresentação do Sistema

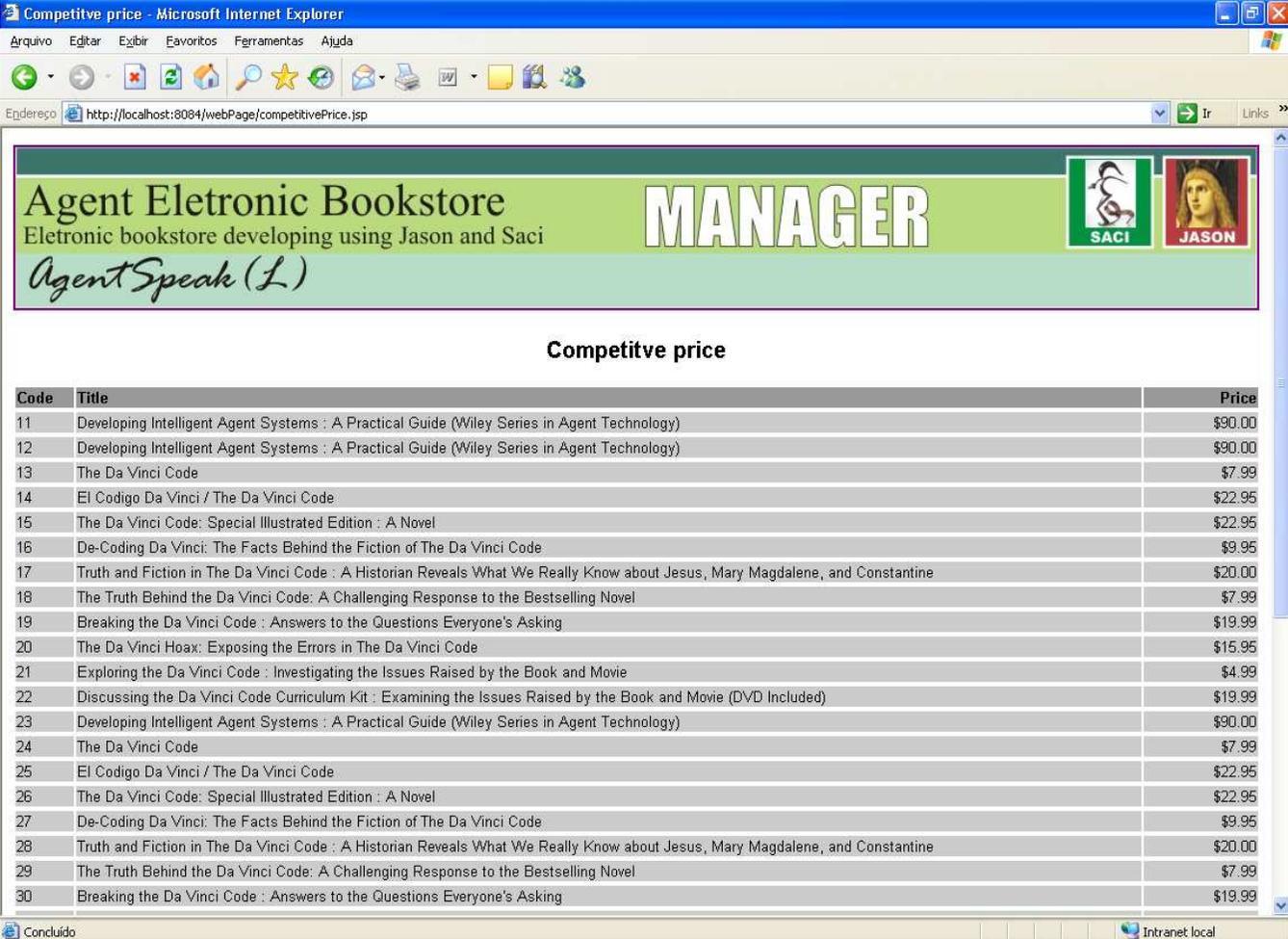
■ Menu de Gerência



Apresentação do Sistema

Apresentação do Sistema

■ Lista de preços da Amazon



The screenshot shows a Microsoft Internet Explorer browser window displaying a web page titled "Competitive price - Microsoft Internet Explorer". The address bar shows the URL "http://localhost:8084/webPage/competitivePrice.jsp". The page content includes a header for "Agent Eletronic Bookstore" with the text "Eletronic bookstore developing using Jason and Saci" and "AgentSpeak (L)". There are two logos: "SACI" and "JASON". The main content is a table titled "Competitive price" with three columns: "Code", "Title", and "Price". The table lists 30 items, including books like "Developing Intelligent Agent Systems" and "The Da Vinci Code", with prices ranging from \$4.99 to \$90.00.

Code	Title	Price
11	Developing Intelligent Agent Systems : A Practical Guide (Wiley Series in Agent Technology)	\$90.00
12	Developing Intelligent Agent Systems : A Practical Guide (Wiley Series in Agent Technology)	\$90.00
13	The Da Vinci Code	\$7.99
14	El Codigo Da Vinci / The Da Vinci Code	\$22.95
15	The Da Vinci Code: Special Illustrated Edition : A Novel	\$22.95
16	De-Coding Da Vinci: The Facts Behind the Fiction of The Da Vinci Code	\$9.95
17	Truth and Fiction in The Da Vinci Code : A Historian Reveals What We Really Know about Jesus, Mary Magdalene, and Constantine	\$20.00
18	The Truth Behind the Da Vinci Code: A Challenging Response to the Bestselling Novel	\$7.99
19	Breaking the Da Vinci Code : Answers to the Questions Everyone's Asking	\$19.99
20	The Da Vinci Hoax: Exposing the Errors in The Da Vinci Code	\$15.95
21	Exploring the Da Vinci Code : Investigating the Issues Raised by the Book and Movie	\$4.99
22	Discussing the Da Vinci Code Curriculum Kit : Examining the Issues Raised by the Book and Movie (DVD Included)	\$19.99
23	Developing Intelligent Agent Systems : A Practical Guide (Wiley Series in Agent Technology)	\$90.00
24	The Da Vinci Code	\$7.99
25	El Codigo Da Vinci / The Da Vinci Code	\$22.95
26	The Da Vinci Code: Special Illustrated Edition : A Novel	\$22.95
27	De-Coding Da Vinci: The Facts Behind the Fiction of The Da Vinci Code	\$9.95
28	Truth and Fiction in The Da Vinci Code : A Historian Reveals What We Really Know about Jesus, Mary Magdalene, and Constantine	\$20.00
29	The Truth Behind the Da Vinci Code: A Challenging Response to the Bestselling Novel	\$7.99
30	Breaking the Da Vinci Code : Answers to the Questions Everyone's Asking	\$19.99

Conclusões

Conclusões

- Metodologia: Prometheus
- AgentSpeak(L) : Abstração
- Saci: Comunicação entre o ambiente desenvolvido em NetBeans e os agentes de maneira transparente

Conclusões

Conclusões

■ Tecnologias / Recursos

- Prometheus
- Jason -> AgentSpeak(L)
- Saci -> KQML
- JSP
- HTML

Conclusões

Conclusões

- Agentes mais proativos
- Crenças compostas por muitos termos
- Substring
- Persistência?

Conclusões

Conclusões

“De maneira geral conclui-se que o trabalho possibilitou que um primeiro passo fosse dado para que a linguagem AgentSpeak(L) através do interpretador Jason e o SACI se tornem uma linguagem de uso mais amplo e acessível.”

Extensões

Extensões

- Melhoramento da qualidade gráfica
- Criação de mais funcionalidades (módulo cliente)
- Aperfeiçoamento do Módulo de Gerência
- Implementação de um comportamento mais sofisticado para os agentes

OBRIGADO!
dalcastagne@gmail.com