

JASON

URKO SANCHEZ ORTIZ

Presentación

- ⦿ Conocimientos necesarios:
 - BDI
 - AGENTSPEAK
- ⦿ JASON

BDI (arquitectura deliverativa)

- ⦿ Arquitecturas BDI se basan en la suposición de que la mente (estado mental) de los agentes consiste en:
 - Creencias (Beliefs): lo que el agente cree que es verdad sobre el mundo (información).
 - Deseos (Desires): estado(s) del mundo que los agentes desean establecer (motivación).
 - Intenciones (Intentions): lo que el agente realmente tiene la intención de hacer y cómo hacerlo (deliberación).
- ⦿ El mundo para un agente son los otros agentes, el entorno, y el propio agente.

AGENTSPEAK

- Originalmente creado por Anand Rao (1996)
- Originalmente llamado AgentSpeak (L)
- AgentSpeak es un lenguaje abstracto que se utiliza para describir y programar agentes BDI
- Notación elegante, basado en la programación lógica
- Inspirado por PRS, dMARS (Kinny) y Lógica BDI (Rao y George)
(TIPOS DE ESTRUCTURAS PARA AGENTES)
- La principal estructura del lenguaje de AgentSpeak es:
 - Beliefs: Creencia del estado actual del agente, la información sobre medio ambiente y otros agentes
 - Goals: Objetivos que declaran el deseo del agente para lograr el objetivo
 - Plans: Plan que ha de cumplir el agente para cumplir su objetivo

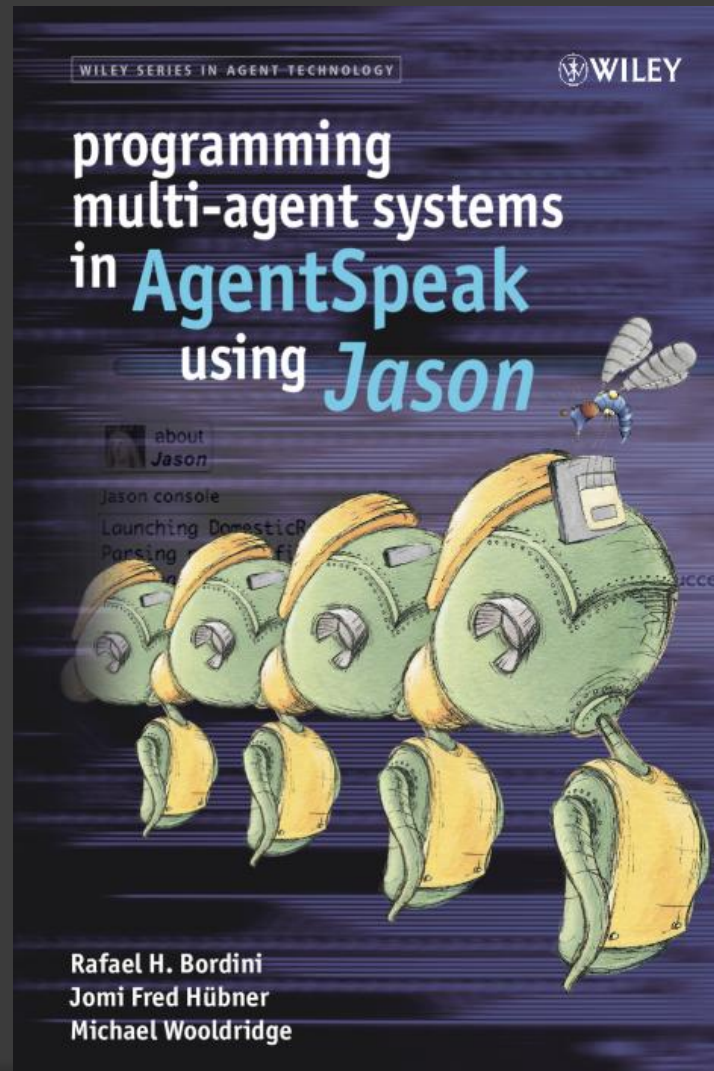
AGENTSPEAK

- La arquitectura de un agente AgentSpeak tiene cuatro componentes principales:
 - Belief Base: Base de creencias.
 - Plan Library: Librería de planes
 - Set of Events: Conjunto de eventos
 - Set of Intentions: Conjunto de intenciones

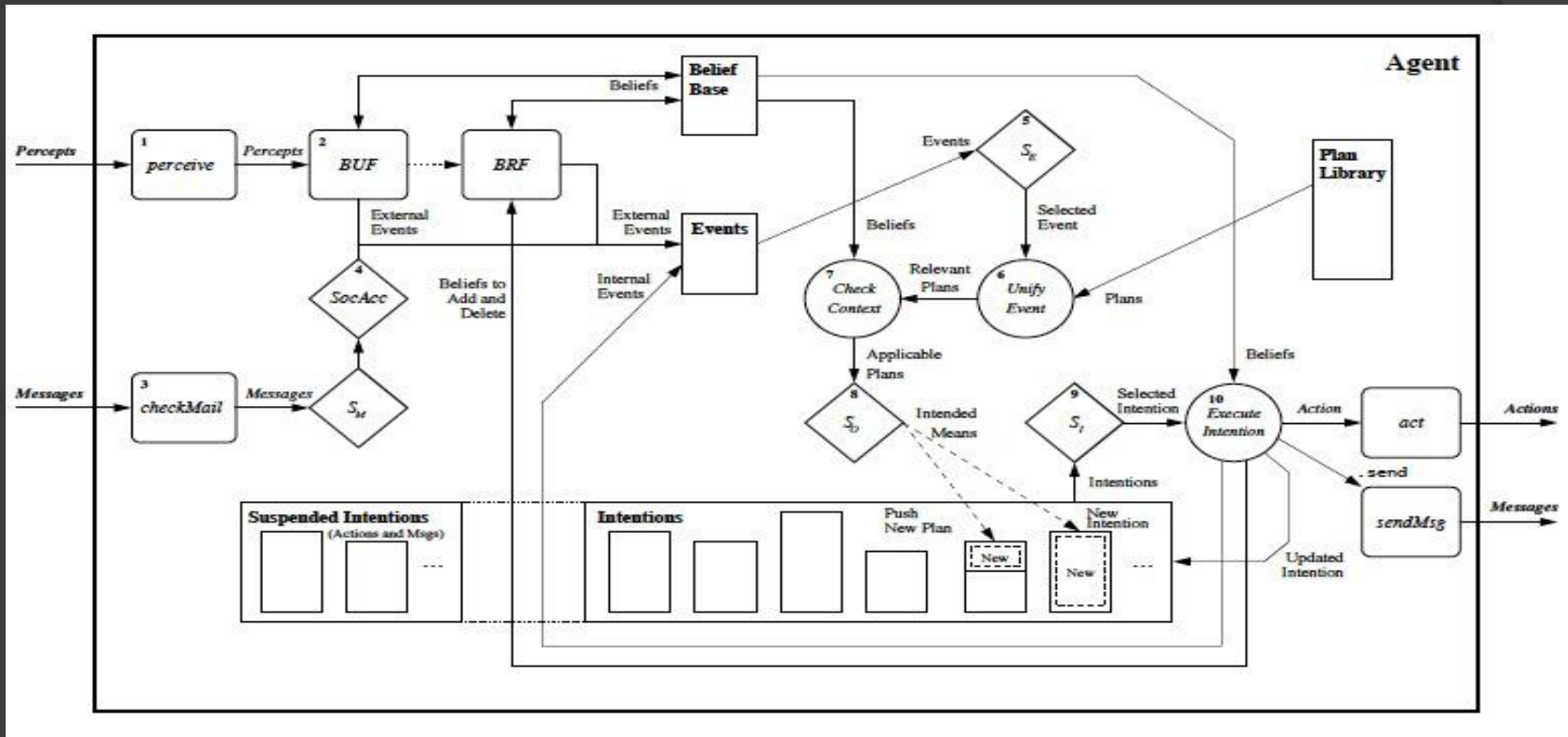
JASON

- ⦿ Desarrollado por Jomi F. Hubner y Rafael H. Bordini
- ⦿ Es una extensión de AgentSpeak, que está destinado a ser el idioma de los agentes
- ⦿ Extensiones dirigidas a un lenguaje de programación más práctico
 - Lenguaje de alto nivel para definir el comportamiento de los agentes
 - Utiliza Java como lenguaje de bajo nivel para darse cuenta de los mecanismos (por ejemplo, las funciones internas del agente) y personalizar la arquitectura
- ⦿ Agrega un conjunto de «mecanismos poderosos» para mejorar las habilidades del agente
- ⦿ Viene con un framework para el desarrollo de sistemas multi-agente
- ⦿ Sistemas multi-agentes distribuidos sobre una red, utilizando SACI, Jade o algún middleware.
- ⦿ Licencia: LGPL (**Lesser General Public License**)

JASON (libro)



JASON (arquitectura)



- 1 Percepción del Medio Ambiente
- 2 Actualización de la Base de Creencia
- 3 Recepción de Comunicación de otros agentes
- 4 Selección Mensajes 'socialmente aceptable'
- 5 Selección de un evento

- 6 Recuperación de todos los planes pertinentes
- 7 Determinación de los Planes Aplicables
- 8 Selección de un plan aplicable
- 9 Selección de una intención para la ejecución
- 10 Ejecución de un paso de la intención

JASON (interface)

The screenshot displays the jEdit IDE interface with the following components:

- Title Bar:** jEdit - proba.mas2j (modified)
- Menu Bar:** File, Edit, Search, Markers, Folding, View, Utilities, Macros, Plugins, Help
- Toolbar:** Standard editing and development icons.
- Structure Browser:** Shows the file 'proba.mas2j' with a sub-entry 'Not parsed'.
- Main Editor:** Contains the following JSON code:

```
1 /* Jason Project */
2
3 MAS proba {
4
5     infrastructure: Centralised
6
7     agents:
8 }
9
10
```
- Footer Area:**
 - about Jason:** A small icon and text.
 - Jason console:** Displays the error message: 'Further Jason functionalities are available in the menu Plugin->Jason.java.net.UnknownHostException: jason.sf.net'
 - Project agents:** An empty panel.
 - Buttons:** 'Error List' and 'Jason IDE'.

Bibliografia

- ◉ German Rigau, Auntes
- ◉ Miguel Lozano, universidad de Valencia
- ◉ <http://jason.sourceforge.net/wp/>
- ◉ <http://dtai.cs.kuleuven.be/projects/ALP/newsletter/aug06/nav/articles/article5/article.html>
- ◉ <http://ppginf.ucpel.tche.br/visitantes/Bordini-UCPel.pdf>
- ◉ <https://cw.felk.cvut.cz/wiki/media/courses/ae4m36mas/slidesjason.pdf>