

Implementación de agentes BDI en JADEX

Marcelo Errecalde[†], Guillermo Aguirre[†], Federico Schlesinger

[†]Laboratorio de Investigación y Desarrollo en Inteligencia Computacional (LIDIC)¹

Departamento de Informática

Universidad Nacional de San Luis

Ejército de los Andes 950 - Local 106

(D5700HHW) - San Luis - Argentina

Tel: (02652) 420823 / Fax: (02652) 430224

e-mail: {merreca, gaguirre}@unsl.edu.ar, fedest@gmail.com

Resumen

Este artículo describe, en forma resumida, parte de los trabajos de investigación y desarrollo que se están llevando a cabo en la línea “Agentes y Sistemas Multi-agente” del LIDIC. El objetivo de este trabajo es presentar las principales temáticas que están siendo abordadas actualmente en el área de modelos y arquitecturas de agentes cognitivos, para posibilitar un intercambio de experiencias con otros investigadores participantes del Workshop, que trabajen en líneas de investigación afines. Uno de los objetivos principales de esta línea, es el estudio y desarrollo de sistemas con agentes basados en el modelo BDI. Las arquitecturas (y modelos) BDI proponen a la trinidad **BDI (Beliefs, Desires e Intentions)** como los elementos claves del estado mental de un agente para tomar las decisiones acerca de cuándo y cómo actuar. Este tipo de enfoque ha demostrado una gran flexibilidad y efectividad en diversos problemas de gran complejidad del mundo real, lo que ha llevado a un creciente interés en la investigación de sus aspectos teóricos pero también de las plataformas que soportan el desarrollo de este tipo de agentes. En este, sentido, el objetivo general de este trabajo es realizar una breve descripción de las motivaciones y objetivos que perseguimos al implementar agentes BDI utilizando frameworks de agentes dedicados a tal fin. En particular, se propone el framework de distribución gratuita Jadex que ya ha sido utilizado en distintos problemas vinculados a la logística de hospitales en Alemania.

1. Introducción

Cada día es más frecuente la utilización de enfoques basados en *agentes inteligentes* [1, 2, 3] para abordar problemas de gran complejidad del mundo real. La principal fortaleza de este enfoque en estos casos reside en la capacidad de sus componentes (agentes) para exhibir un comportamiento *flexible*. La idea de flexibilidad en este contexto refiere a la capacidad de los agentes para:

- Percibir directamente un ambiente dinámico y reaccionar oportunamente a eventos y condiciones cambiantes (*reactividad*).
- Tomar la iniciativa cuando sea necesario e iniciar comportamientos dirigidos por un objetivo (*pro-actividad*).
- Interactuar y comunicarse, cuando es apropiado, con otros agentes artificiales o humanos (*socialidad*).

¹Las investigaciones realizadas en el LIDIC son financiadas por la Universidad Nacional de San Luis y por la Agencia Nacional de Promoción Científica y Tecnológica.

Proveer con estas capacidades a un agente no es una tarea sencilla. De hecho, una de las áreas de investigación más activa en el ámbito de agentes ha sido la definición de *arquitecturas de agentes* que intentan dar una respuesta a este problema.

Existen arquitecturas que se han concentrado en el aspecto de la reactividad [4, 5, 6] y otras en cambio que han privilegiado los mecanismos de deliberación y planning necesarios para proveer un comportamiento pro-activo[7]. Sin embargo, existe actualmente un consenso generalizado en que cualquier arquitectura realista de agente, debería proveer un soporte adecuado para todas estas capacidades. Las arquitecturas híbridas [8, 9] y las basadas en comportamientos [10, 11] han intentado lograr un adecuado balance entre reactividad y pro-actividad. Sin embargo, las arquitecturas que mayor atención han recibido para este propósito son las denominadas *arquitecturas BDI* [12].

El modelo BDI, al igual que la teoría de decisión clásica y la teoría de decisión cualitativa son modelos de *razonamiento práctico*. Razonamiento práctico es el razonamiento dirigido a la *acción*. Se diferencia del *razonamiento teórico* en que este último está dirigido a las *creencias*. Concluir que “Sócrates es mortal” es razonamiento teórico, dado que solamente afecta mis creencias sobre el mundo. Decidir si tomar un tren o un colectivo es razonamiento práctico, ya que es razonamiento dirigido a la acción.

Podemos encuadrar filosóficamente al modelo BDI dentro de lo que se suele referenciar como *postura intencional*. Este enfoque plantea esencialmente que un agente debe ser conceptualizado y/o implementado usando conceptos y nociones o estados mentales usualmente asociados a los humanos, como por ejemplo creencias, deseos, intenciones, obligaciones, compromisos, etc. El filósofo Daniel Dennet, ya en 1987 utiliza el término *sistema intencional* para describir entidades “cuyo comportamiento puede predecirse atribuyéndole creencias, deseos y perspicacia racional”. Básicamente, las nociones intencionales son herramientas de abstracción, que proveen una manera conveniente de describir, explicar y predecir el comportamiento de sistemas complejos.

En el caso particular de las arquitecturas (y modelos) BDI se proponen a la trinidad **BDI (Beliefs, Desires e Intentions)** como los elementos claves del estado mental de un agente para tomar las decisiones acerca de cuándo y cómo actuar. A continuación describimos brevemente cada una de estas componentes:

- *Beliefs* (Creencias): Son sentencias que un agente toma como verdaderas (que a diferencia del *conocimiento* pueden ser falsas) acerca de propiedades de su mundo (y de sí mismo). Las creencias intentan capturar el *estado de información* (“informational state”) del agente.
- *Desires* (Deseos): Son acciones que un agente desea realizar o situaciones que prefiere y quiere lograr. Los deseos intentan capturar el *estado de motivación* (“motivational state”) del agente. Los *Deseos* se asemejan a los **objetivos (goals)**, pero los objetivos involucran cierto grado de compromiso del agente en su realización y que el conjunto de objetivos perseguidos sea consistente.
- *Intentions* (*Intenciones*): Acciones factibles, planes o situaciones deseadas que el agente ha seleccionado y se ha comprometido a realizar o lograr. Las intenciones intentan capturar el *estado deliberativo* (“deliberative state”) del agente.

El modelo BDI, puede dar una respuesta adecuada a los requerimientos que deberán enfrentar los sistemas de software en el futuro. En particular en [13] se reconoce que los ambientes complejos (dinámicos, inciertos, limitados en recursos y parcialmente observables) tienden a ser la norma y se requieren enfoques alternativos al desarrollo de software tradicional. En este tipo de problemas, el aporte de cada una de las componentes del modelo BDI se torna evidente:

- **Beliefs**: el mundo cambia y por consiguiente debo recordar eventos pasados. La percepción es incompleta por lo que debo recordar lo que no percibo actualmente. Además el sistema es acotado en recursos computacionales por lo que no conviene recomputar toda la información relevante a partir de la entrada perceptual.

- **Desires (y Goals):** el software tradicional (orientado a tareas) no tiene ningún registro de cual es el motivo por el cual está siendo ejecutado. Cuando el estado motivacional es explícitamente representado, el sistema puede recuperarse automáticamente ante las fallas. Puede además, aprovechar oportunidades que surgen dinámicamente.
- **Intentions:** el agente necesita reconsiderar los planes que ha adoptado y está ejecutando en el contexto de un mundo cambiante. Esta posibilidad de razonar sobre los planes adoptados para atender a necesidades más urgentes o determinar que una intención ha perdido razón de ser, no está disponible en el software tradicional donde no se reconsidera nunca.

Más allá de los fundamentos filosóficos [14] y formales [15] del modelo BDI, su efectividad en problemas concretos de gran complejidad ha quedado demostrada en los últimos años. Algunas de las aplicaciones más conocidas son:

- Sistema de control de tráfico aéreo OASIS (aeropuerto de Sydney)
- Simulador para la fuerza aérea australiana (SWARMM)
- Sistema de propulsión de una nave espacial de la NASA (RCS)
- Sistema administrador de procesos de negocios (SPOC)
- Sistema para el diagnóstico, control y monitoreo de una red de telecomunicaciones de Telecom Australia (IRTNMS)

Sin embargo, a pesar de la relevancia de las arquitecturas BDI para el desarrollo de software para dominios complejos, se podría decir que en nuestro país es un área prácticamente inexplorada tanto en el sector industrial como universitario. Si bien se han realizado algunos trabajos teóricos de agentes BDI abordados con enfoques argumentativos [16], hasta donde sabemos, no existen trabajos concretos con plataformas y herramientas específicos para el desarrollo de este tipo de agentes. En este contexto, dentro de nuestro grupo de trabajo nos proponemos hacer una experiencia de desarrollo de agentes BDI utilizando la plataforma JADDEX.

2. Antecedentes

Las arquitecturas BDI comienzan a recibir una atención creciente a partir de la aplicación de las ideas del filósofo Bratman en la arquitectura *IRMA* (Intelligent Resource-bounded Machine)[12]. En esencia, Bratman argumenta que un agente racional tenderá a enfocar su razonamiento práctico sobre las intenciones que ya ha adoptado y tenderá a “bypasar” aquellas opciones que entran en conflicto con estas intenciones. La característica distintiva de la propuesta de Bratman es el énfasis en el **rol de las intenciones** para ayudar a enfocar los procesos de deliberación y el razonamiento medios-fines (planning).

Más allá del impacto que tuvieron las ideas planteadas en *IRMA*, se la puede considerar todavía una arquitectura abstracta donde existen distintas componentes cuya implementación no está totalmente especificada. En este sentido, el primer sistema de uso industrial basado en el modelo BDI es el *Sistema de Razonamiento Procedural (PRS)* [17], el cual contaba con estructuras de datos explícitas que correspondían a los estados mentales BDI.

A partir del suceso del sistema PRS en varios problemas de envergadura, surgieron distintas variantes de PRS que en muchos casos fueron simples reimplementaciones en otros lenguajes, o bien extensiones para cubrir aspectos no considerados en el sistema original. Así, comienza un período donde se implementan y difunden distintas plataformas para el desarrollo de agentes BDI, algunos

de índole académico y otras pensadas para el uso industrial. Entre las más conocidas podemos citar a JAM [18], Jack [19], AgentSpeak(L) [20], dMars [21, 22] y Jadex [23].

De todas estas plataformas, la más relevante para nuestro trabajo es la plataforma Jadex. Jadex es una extensión al poderoso “middleware” de agentes Jade [24]. Al igual que otros “middleware” de agentes, Jade provee funcionalidades genéricas para facilitar el desarrollo de agentes, que incluyen la administración de agentes, servicios de directorio y distribución de mensajes confiable entre los agentes. Todas estas facilidades están implementadas siguiendo el modelo de referencia de FIPA ². Como antecedentes más cercanos en el uso de Jade podemos mencionar distintos trabajos finales y trabajos de investigación desarrollados en el LIDIC de la Universidad Nacional de San Luis [25, 26, 27].

Jadex surge en el contexto del proyecto MedPage (“Medical Path Agents”), donde se plantea la necesidad de contar con una plataforma de agentes que soporte comunicación conforme a los requerimientos de FIPA y además provea una arquitectura de agente de alto nivel del tipo BDI. El proyecto MedPage es parte del siguiente programa de investigación prioritario de Alemania: “1013 *Intelligent Agents in Real World Business Applications*. Surge a partir de la cooperación del departamento de administración de negocios de la Universidad de Mannheim y el departamento de ciencias de la computación de la Universidad de Hamburgo, en un trabajo conjunto para investigar las ventajas de usar tecnología de agentes en el contexto de la logística de los hospitales [28]. El proyecto Jadex comenzó en Diciembre de 2002 para proveer el sustento técnico a los prototipos de software de MedPage desarrollados en Hamburgo.

3. Objetivos

El objetivo general de nuestro trabajo es realizar una primera aproximación al problema de implementar agentes BDI utilizando frameworks de agentes dedicados a tal fin. En particular, se utilizará el framework de distribución gratuita Jadex que ya ha sido utilizado en distintos problemas vinculados a la logística de hospitales en Alemania. En este contexto, un objetivo parcial a cumplir será el estudio y análisis de otros frameworks para el desarrollo de agentes BDI, a los fines de individualizar las similitudes y diferencias de los mismos con Jadex. También se experimentará con los ejemplos introductorios provistos con Jadex para adquirir experiencia en las herramientas para el desarrollo, visualización y depuración de aplicaciones BDI que provee este framework.

En base al trabajo de investigación y experimentación previo, se propondrá un problema donde claramente se visualice la potencialidad y flexibilidad del enfoque BDI. Este problema involucrará la ejecución de comportamientos dirigidos por el objetivo que con frecuencia deberán ser reconsiderados debido a la ocurrencia inesperada de eventos (por ejemplo generada desde dispositivos móviles) que pueden requerir de un tratamiento urgente. Todas estas componentes, al igual que las capacidades de meta-razonamiento para la reconsideración de intenciones serán implementadas en Jadex.

Referencias

- [1] Michael Wooldridge. *An Introduction to MultiAgent Systems*. John Wiley & Sons, Chichester, England, 2002.
- [2] Gerhard Weiss, editor. *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*. The MIT Press, 1999.
- [3] Jacques Ferber. *Multi-Agent Systems - An Introduction to Distributed Artificial Intelligence*. Addison-Wesley, 1999.
- [4] Rodney A. Brooks. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 1986.
- [5] R. C. Arkin. *Behaviour-Based Robotics*. The MIT Press, 1998.

²Foundation for Intelligent Physical Agents. <http://www.fipa.org/>

- [6] R. C. Arkin. Integrating behavioral, perceptual and world knowledge in reactive navigation. *Robotics and Autonomous Systems*, 1990.
- [7] M. E. Pollack. Planning technology for intelligent cognitive orthotics. In *Proceedings of 6th International Conference on AI Planning and Scheduling*, 2002.
- [8] E. Gat. On three-layer architectures. In *Artificial Intelligence and Mobile Robots*, 1998.
- [9] E. Gat. Integrating planning and reacting in a heterogeneous asynchronous architecture for mobile robots. *SIGART Bulletin*, 1991.
- [10] Julio K. Rosenblatt. DAMN: A distributed architecture for mobile navigation. In *Proc. AAAI Spring Symposium on Lessons Learned from Implemented Software Architectures for Physical Agents*. AAAI Press, 1995.
- [11] Paolo Pirjanian. *Multiple Objective Action Selection and Behaviour Fusion using Voting*. PhD thesis, Department of Medical Informatics and Image Analysis, Institute of Electronic Systems, Aalborg University, Denmark, August 1998.
- [12] M. E. Bratman, D. J. Israel, and M. E. Pollack. Plans and resource-bounded practical reasoning. *Computational Intelligence*, 4:349–355, 1988.
- [13] Mike Georgeff, Barney Pell, Martha Pollack, Milind Tambe, and Mike Wooldridge. The belief-desire-intention model of agency. In Jörg Müller, Munindar P. Singh, and Anand S. Rao, editors, *Proceedings of the 5th International Workshop on Intelligent Agents V : Agent Theories, Architectures, and Languages (ATAL-98)*, volume 1555, pages 1–10. Springer-Verlag: Heidelberg, Germany, 1999.
- [14] Michael E. Bratman. *Intention, Plans, and Practical Reason*. CSLI Publications, 1999.
- [15] Klaus Schild. On the relationship between bdi logics and standard logics of concurrency. *Autonomous Agents and Multi-Agent Systems*, 3(3):259–283, 2000.
- [16] Sonia Rueda, Alejandro J. García, and Guillermo R. Simari. Argument-based negotiation among BDI agents. *Journal of Computer Science and Technology*, 2(7):1–8, October 2002.
- [17] M. P. Georgeff and A. L. Lansky. Reactive reasoning and planning. In *Proceedings of the Sixth National Conference on Artificial Intelligence (AAAI-87)*, pages 677–682, 1987.
- [18] Marcus J. Huber. Jam: a bdi-theoretic mobile agent architecture. In *AGENTS '99: Proceedings of the third annual conference on Autonomous Agents*, pages 236–243, New York, NY, USA, 1999. ACM.
- [19] Nick Howden, Ralph Rönquist, Andrew Hodgson, and Andrew Lucas. Jack intelligent agents summary of an agent infrastructure. In *Proceedings of the Fifth International Conference on Autonomous Agents*, 2001.
- [20] Anand S. Rao. AgentSpeak(L): BDI agents speak out in a logical computable language. In Rudy van Hoe, editor, *Seventh European Workshop on Modelling Autonomous Agents in a Multi-Agent World*, Eindhoven, The Netherlands, 1996.
- [21] Mark d'Inverno, David Kinny, Michael Luck, and Michael Wooldridge. A formal specification of dmars. In Munindar P. Singh, Anand S. Rao, and Michael Wooldridge, editors, *ATAL*, volume 1365 of *Lecture Notes in Computer Science*, pages 155–176. Springer, 1997.
- [22] Mark D'Inverno, Michael Luck, Michael Georgeff, David Kinny, and Michael Wooldridge. The dMARS architecture: A specification of the distributed multi-agent reasoning system. *Autonomous Agents and Multi-Agent Systems*, 9:5–53, 2004.
- [23] Alexander Pokahr, Lars Braubach, and Winfried Lamersdorf. Jadex: Implementing a bdi infrastructure for jade agents. *EXP In Search of Innovation (Special Issue on JADE)*, 3(3):76–85, 2003. Telecom Italia Lab.
- [24] Fabio Bellifemine, Agostino Poggi, and Giovanni Rimassa. Jade a fipa compliant agent framework. In *In Proceedings of PAAM99*, pages 97–108, 1999.
- [25] Fernando González. Análisis teórico – práctico del desarrollo de un sistema multiagente. Trabajo Final de Licenciatura en Ciencias de la Computación (UNSL), 2003.
- [26] Cristina Sáez. SMALL: Sistema multiagente para la administración de links en una lan. Trabajo Final de Licenciatura en Ciencias de la Computación (UNSL), 2003.
- [27] Errecalde M., G. Aguirre, and F Gonzalez. Agentes y mecanismos de votación. In *Anales del 10mo Congreso Argentino de Ciencias de la Computación*, pages 1474–1485. Universidad Nacional de La Matanza, 2004.
- [28] T. O. Paulussen, N. R. Jennings, K. S. Decker, and A. Heinzl. Distributed patient scheduling in hospitals. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence (IJCAI-03)*, 2003.