

Métodos, Técnicas y Herramientas para la Ingeniería de Software Orientada a Agentes

Diana Palliotto, Gregorio N. Tkachuk, Carlos E. Vega Ugozzoli
Departamento de Informática, Facultad de Ciencias Exactas y Tecnologías
Universidad Nacional de Santiago del Estero (UNSE)
Av. Belgrano (S) 1912, Santiago del Estero
Teléfono: +54 385 4509500 interno 1817
e-mails: {dpalliot, gregorio}@unse.edu.ar; cvegaugozzoli@hotmail.com

Resumen

La Ingeniería de Software Orientada a Agentes es un enfoque que ha demostrado ser conveniente para construir sistemas complejos en comparación con el desarrollo estándar.

Indudablemente, el éxito de la realización de los sistemas orientados a agentes sólo puede garantizarse si se puede reducir la brecha entre el análisis y la implementación, y así desarrollar herramientas para que los conceptos y las técnicas de los sistemas multiagentes puedan implementarse fácil y directamente.

Pero, para desarrollar sistemas basados en agentes, se necesita una metodología que soporte el proceso de desarrollo como es común en otras disciplinas. En años recientes, han surgido muchos métodos y técnicas de modelado, aunque ninguna de las metodologías existentes puede considerarse completa. Se podría decir que las metodologías orientadas a agentes son el resultado de una transferencia de conocimiento desde la ingeniería del software. Seguramente, estas metodologías evolucionarán hacia el establecimiento de nuevas actividades de desarrollo lo que incrementará su alcance, provocando una mejora general de las herramientas de soporte.

Este proyecto se propone aportar al mejoramiento tanto del proceso de desarrollo como de las aplicaciones basadas en agentes, revisando, modificando y realizando propuestas metodológicas que se aplicarán y evaluarán en

sistemas basados en agentes, propiciando el avance del conocimiento científico-tecnológico.

Palabras clave: Ingeniería de Software, Metodología de Desarrollo de Software, Sistemas Basados en Agentes, Ingeniería de Software Orientada a Agentes.

Contexto

Este trabajo presenta una de las líneas de investigación del programa "Sistemas de Información Web Basados en Tecnología de Agentes", aprobado por el Consejo de Ciencia y Técnica de la Universidad Nacional de Santiago del Estero (CICyT, UNSE) para el período 2012-2015. La propuesta es continuación de líneas de investigación iniciadas en el año 2005 en el proyecto "Herramientas Conceptuales, Metodológicas y Técnicas de la Informática Teórica y Aplicada" (Código 23/C062), aprobado y financiado por CICYT, UNSE.

La línea de investigación que se plantea originó el proyecto denominado "Métodos, Técnicas y Herramientas para la Ingeniería de Software Orientada a Agentes" (Código 23/C096), y está orientada a revisar, modificar y/o realizar propuestas metodológicas en el área de la Ingeniería de Software Orientada a Agentes, dirigidas particularmente al desarrollo de sistemas de información web que soporten el aprendizaje ubicuo y colaborativo.

Introducción

Con las aplicaciones cada vez más sofisticadas exigidas por los negocios que apuntan a tomar ventajas competitivas, las tecnologías de objetos están complementándose y suplementándose con las tecnologías de agentes. Esto es especialmente cierto en áreas tales como inteligencia ambiental, e-business, servicios Web, redes y bioinformática. Estas áreas exigen software que sea robusto, que pueda operar dentro de una amplia gama de ambientes, que pueda evolucionar para cumplir con los requerimientos cambiantes, que pueda personalizarse para satisfacer las necesidades de un amplio rango de usuarios, y que sea suficientemente seguro para proteger datos personales y otros recursos. Para cumplir estos requerimientos, los desarrolladores de sistemas necesitan una metodología adecuada con el enfoque orientado a agentes. [1]

Ahora bien, existen numerosas discusiones acerca de lo que es una metodología orientada a agentes, y es sorprendente que todavía no haya ninguna definición estándar. Esto puede deberse a un abuso del término en muchos trabajos de investigación que afirman ser una metodología. Muchos investigadores han detectado este problema, aunque eso no ha provocado una autocrítica por parte de la comunidad de la ingeniería de software orientada a agentes. Como resultado, muchos artículos dicen presentar una metodología y sólo introducen un lenguaje de modelado o enumeran unas pocas actividades de desarrollo. [2]

Se podría decir que las metodologías orientadas a agentes son el resultado de una transferencia de conocimiento desde la ingeniería del software. Los investigadores de agentes son los expertos para decir si una metodología captura lo esencial del concepto de agente; sin embargo, para evaluar la capacidad de una metodología los expertos son los ingenieros de software.

Actualmente, conocer sobre ingeniería de software significa que, por lo menos, se está familiarizado con el Cuerpo de Conocimiento de la Ingeniería de Software (Software Engineering Body of Knowledge - SWEBOK) [3].

SWEBOK tiene un área denominada Métodos y Herramientas de la Ingeniería de Software y plantea que "un método puede ser heurístico, formal o de prototipado" [3]. De esta categorización de los métodos, un método orientado a agentes podría rotularse como un método heurístico donde el sistema se ve como una colección de agentes. Esto tendría sentido en la mayoría de los casos, aunque la integración con los métodos formales y/o las técnicas de prototipado puede complicar la situación. En cualquier caso, el método debe apuntar a hacer la actividad sistemática para que su éxito sea más probable, y debe proporcionar notación y vocabulario, procedimientos para realizar tareas identificables y pautas para verificar tanto el proceso como el producto.

El alcance del método puede ir desde una única fase al ciclo de vida completo. Este alcance permite hablar sobre análisis orientado a agentes el cual se concentraría en cómo determinar qué agentes son necesarios para proporcionar la funcionalidad del sistema; o bien sobre implementación orientada a agentes que dirigiría la especificación del sistema, sin tener en cuenta su orientación a agentes, para luego implementarse usando soluciones orientadas a agentes (por ejemplo, un lenguaje de programación orientado a agentes o un framework orientado a agentes). La fase o el proceso concreto se define separadamente y puede ser ad hoc o derivarse de algún modelo de ciclo de vida del software. Claramente, el proceso o la fase afectados se inclinarán por el uso de conceptos relacionados a agentes, utilizando los productos de ingeniería necesarios para referirse a esos conceptos. Finalmente, debería haber herramientas de soporte que

ayuden en la ejecución de las diferentes actividades dirigidas por el método.

Actualmente, pueden encontrarse dos grandes grupos de propuestas metodológicas orientadas a agentes [2]. En el primero de ellos, existe una metodología que primero intenta construir una notación y un vocabulario para luego ocuparse de algunas fases de desarrollo específicas. En el segundo, la notación y el vocabulario están incluidos en una herramienta que permite moverse desde el análisis y el diseño hacia la implementación.

En el primer grupo, hay trabajos que se introdujeron como metodologías en su momento, pero deberían introducirse como métodos específicos de algunas fases; GAIA [4, 5] y MESSAGE/UML [6] son ejemplos de este grupo.

En el segundo grupo, los trabajos se desarrollaron alrededor de un soporte poderoso de herramientas que cubren una parte importante del ciclo de vida. ZEUS [7] y MaSE [8] son representativas de este grupo.

Con el tiempo estos dos grupos se han unido y, como resultado, se ha incrementado el número de propuestas metodológicas que consideran más partes del ciclo de vida de software. Las metodologías como INGENIAS, PASSI, ADELFE o Prometheus consideran fases como requerimientos, implementación y prueba [1]. Cuantas más fases incorporen, están más cercanas a una metodología de calidad competitiva.

Como el área de agentes autónomos y de sistemas multiagentes se ha convertido en una tecnología prometedora, que ofrece alternativas razonables para el diseño de sistemas distribuidos inteligentes, se han realizado numerosos esfuerzos (académicos, industriales y de consorcios de estandarización) para proporcionar nuevas herramientas, con la intención de establecer las normas necesarias para lograr un uso extendido de las técnicas de los sistemas multiagentes. Para esto, es esencial que tal tecnología pueda incorporarse en las prácticas existentes en la industria del software, y no que

se vea simplemente como un nuevo paradigma prometedor. El éxito está irrevocablemente asociado con el desarrollo de lenguajes y herramientas que soporten una programación eficaz, incluso la implementación de nociones importantes en un framework unificado.

Con respecto a los lenguajes de programación, se pueden mencionar los siguientes: Jason [9, 10], 3APL [9, 11], IMPACT [9, 12], el lenguaje CLAIM y la plataforma distribuida SyMPA [9].

En relación a lenguajes y plataformas que son extensiones o están basados en Java, entre otros, se pueden indicar los siguientes: JADE [9, 13], Jadex [9, 14] y JACK [9, 15].

Descripción de la línea de investigación

En nuestro país existen, en el ambiente académico, varios grupos de investigación cuya línea central son los sistemas basados en agentes, pero se concentran principalmente en el desarrollo de aplicaciones basadas en agentes y en la construcción de herramientas de soporte para el desarrollo de agentes.

Esta línea de investigación se propone contribuir al perfeccionamiento del proceso de desarrollo y, por ende, de las aplicaciones basadas en agentes, a través de la revisión, la modificación y/o la realización de propuestas metodológicas que luego se aplicarán y evaluarán en sistemas basados en agentes, propiciando el avance del conocimiento científico-tecnológico.

Consecuentemente, el objetivo general que guía esta investigación es:

Aportar al mejoramiento de la calidad tanto del proceso de desarrollo como de las aplicaciones basadas en agentes.

Considerando este objetivo general, se plantean los siguientes objetivos específicos:

- a. Alcanzar un mayor entendimiento del paradigma orientado a agentes.

- b. Evaluar las técnicas de modelado existentes, considerando tanto criterios de la ingeniería de software así como características de los sistemas basados en agentes, en especial aquellos que se utilizan como apoyo al aprendizaje ubicuo y colaborativo.
- c. Facilitar el modelado, el desarrollo, el mantenimiento y la reutilización de los sistemas basados en agentes.
- d. Realizar aplicaciones y derivaciones metodológicas y técnicas de la Ingeniería de Software Orientada a Agentes.
- e. Contribuir al mejoramiento de la calidad (portabilidad, interoperabilidad, etc.) de los sistemas basados en agentes.

Metodología

Con el fin de dar consecución al objetivo específico **a**, se realizarán el estudio y el análisis de los siguientes temas:

- Paradigmas de la Ingeniería de Software y de la Ingeniería de Software Orientada a Agentes.
- Metodologías de desarrollo de sistemas basados en agentes, particularmente de sistemas de información web que soportan el aprendizaje ubicuo y colaborativo.
- Lenguajes y plataformas de desarrollo de sistemas basados en agentes.

Luego, para alcanzar el objetivo específico **b**, se proponen las siguientes actividades:

- Establecer criterios de evaluación provenientes de la ingeniería de software.
- Establecer las características a evaluar de los sistemas basados en agentes.
- Evaluar las técnicas en base a los criterios y las características.
- Revisar los resultados de las evaluaciones.

Con el propósito de dar cumplimiento al objetivo específico **c**, las actividades son:

- En base a las evaluaciones anteriores, determinar las ventajas y desventajas de

cada método y herramienta, indicando los problemas que se detectaron.

- Producir informes sobre la aplicación de los métodos y herramientas.
- Validar empíricamente la aplicación de los métodos y herramientas.

Para el objetivo específico **d**, las actividades son las que siguen:

- Seleccionar los métodos y modelos de agentes.
- Analizar y formular mejoras.
- Proponer modelos, métodos y/o técnicas.
- Validar e interpretar los resultados y aplicarlos.

Finalmente, para alcanzar el objetivo específico **e**, se plantean las actividades siguientes:

- Realizar un juicio técnico de los sistemas basados en agentes desarrollados.
- Detallar clara y exhaustivamente todas y cada una de las actividades completas y los productos generados.

Resultados esperados

Los resultados que se esperan obtener de la realización de este proyecto son:

- Evaluación de métodos, técnicas y herramientas existentes para la ingeniería de software basada en agentes.
- Propuestas metodológicas (modelos, métodos y/o técnicas) para el desarrollo de sistemas basados en agentes, que deberán validarse adecuadamente aplicándolas en desarrollos concretos.

El proyecto impulsa la apertura de una nueva línea de investigación sobre ingeniería de software. Además, el desarrollo de este proyecto permitirá afianzar al grupo de investigación en la temática del proyecto.

Formación de Recursos Humanos

El proyecto facilitará la formación de recursos humanos de la FCEyT, brindando la

posibilidad a investigadores noveles de realizar sus primeras experiencias en investigación, y permitiendo que los integrantes estudiantes realicen sus Trabajos Finales de Grado de la carrera Licenciatura en Sistemas de Información en el marco del proyecto.

Referencias

- [1] Henderson-Sellers, Brian; Giorgini, Paolo (Editors). "Agent-oriented methodologies". Idea Group Publishing. London, UK, 2005.
- [2] Gomez-Sanz, Jorge J.; Fuentes-Fernández, Rubén; Pavón, Juan. "Understanding Agent Oriented Software Engineering Methodologies". 10 AAMAS Conference, Workshop AOSE 2011. Taipei, Taiwan, 2011.
- [3] Abran, Alain; Moore, James W. (executive editors); Bourque, Pierre; Dupuis, Robert (editors). "Guide to the software engineering body of knowledge: 2004 version (SWEBOK)". The Institute of Electrical and Electronics Engineers, 2004.
- [4] Wooldridge, M., Jennings, N. R. and Kinny, D. "The Gaia Methodology for Agent-Oriented Analysis and Design". Journal of Autonomous Agents and Multi-Agent Systems, 3 (3). pp. 285-312. 2000.
- [5] Zambonelli, F.; Jennings, N. R.; Wooldridge, M. "Developing multiagent systems: The Gaia methodology". ACM Transactions on Software Engineering and Methodology, 12(3):317-370, 2003.
- [6] Caire, Giovanni et al. "Agent Oriented Analysis using MESSAGE/UML". Proceeding AOSE '01 Revised Papers and Invited Contributions from the Second International Workshop on Agent-Oriented Software Engineering II. Springer-Verlag, London, UK, 2002.
- [7] Nwana, Hyacinth S.; Ndumu, Divine T.; Lee, Lyndon C. "ZEUS: An Advanced Tool-Kit for Engineering Distributed Multi-Agent Systems". Proceedings Agents 1999, pp. 360-361. 1999.
- [8] Deloach, Scott A.; Wood, Mark F.; Sparkman, Clint H. "Multiagent Systems Engineering". International Journal of Software Engineering and Knowledge Engineering, Vol. 11, No. 3, pp. 231-258. World Scientific Publishing Company, 2001.
- [9] Bordini, Rafael H.; Dix, Jiirgen; Dastani, Mehdi; El Fallah Seghrouchni, Amal (Editors). "Multi-Agent Programming. Languages, Platforms and Applications". Springer Science+Business Media, Inc. 2005.
- [10] Sitio Web de Jason. Disponible en: <http://jason.sourceforge.net/Jason/Jason.html>
- [11] Sitio Web de 3APL. Disponible en: <http://www.cs.uu.nl/3apl>
- [12] Sitio Web de IMPACT. Disponible en: <http://www.cs.umd.edu/projects/impact>
- [13] Sitio Web de JADE. Disponible en: <http://jade.tilab.com>
- [14] Sitio Web de Jadex. Disponible en: <http://jadex-agents.informatik.uni-hamburg.de>
- [15] Sitio Web de JACK. Disponible en: <http://www.agent-oftware.com.au/products/jack>