

Modelos y herramientas de software libre en el campo de la robótica

Pablo Barrera, Francisco Martín, Gregorio Robles, José M. Cañas and Vicente Matellán

Grupo de Sistemas y Comunicaciones (GSyC), Universidad Rey Juan Carlos,
C/ Tulipán s/n CP. 28933 Móstoles (Madrid), Spain.
{barrera,grex,jmplaza,fmartin,vmo}@gsyc.escet.urjc.es

Abstract. El software es uno de los principales componentes de los robots. Realmente es el principal cuello de botella para la proliferación de la robótica en nuestra vida diaria. En los últimos años el campo de la robótica ha sido un área en el que el fenómeno del software libre ha crecido de manera extraordinaria.

En este artículo presentamos brevemente la situación del software libre en tres subáreas: industria, enseñanza e investigación. Las herramientas libres han sido muy populares entre la comunidad docente e investigadora en el campo de la robótica. Incluso compañías cuyo principal modelo de negocio es la venta de robots, han creído conveniente compartir el software con el fin de promover comunidades alrededor de sus productos.

Todos estos casos muestran que el software libre puede actuar como un catalizador en la industria robótica, un sector aún en fase temprana de desarrollo, en el mundo investigador e industrial.

Introducción La robótica es un campo poco conocido por los desarrolladores de software libre, pero está fuertemente influenciado por el software libre, sus modelos de desarrollo y herramientas.

Muchos de los kits robóticos que se venden actualmente incluyen algún tipo de software para controlar o interactuar con el robot. Estos programas puede adaptarse, mejorarse o incluso reemplazarse para añadir nuevas funciones que no fueron incluidas por el fabricante. Las comunidades que se forman alrededor de estos robots no comparten solo el interés por ellos, sino también mejoras, soluciones, código y nuevas ideas gracias a Internet.

En este artículo revisaremos la situación actual del software libre en el campo de la robótica en las áreas de industria, educación e investigación. Para cada una hemos elegido algunos casos de estudio conocidos por los autores, donde el software libre ha sido la estrategia adoptada.

Software libre en la industria, enseñanza e investigación robótica Aunque la industria se muestra reacia a abrir sus desarrollos, se ha demostrado claramente que existen buenos argumentos para impulsar a una compañía de robots a entrar en un modelo caracterizado por la compartición del código fuente que hace funcionar a sus robots. Un ejemplo es el robot *Cye*¹, desarrollado y distribuido por Probotics Inc. Inicialmente esta compañía vendía su software *Map-N-Zap* independientemente de sus robots. Pero pronto se dieron cuenta que los robots no eran útiles sin ese software, así que decidieron incluirlo en el kit básico de éstos. En enero del año 2000 finalmente decidieron distribuir el software bajo licencia GNU GPL (GNU General Public License). El anuncio de esta estrategia tenía como objetivo claro la comunidad de software libre².

Otro ejemplo lo encontramos en el robot Pioneer de Activmedia se vende como una herramienta de investigación, y se usa por una gran cantidad de universidades como principal plataforma de investigación en robots móviles. En el momento en que Activmedia comenzó a vender este robot existían otras alternativas en el mercado con otras estrategias, pero actualmente Activmedia es conocido como el mayor vendedor de robots para fines investigadores. Probablemente la diferencia determinante para esta evolución fue que Activmedia comenzó a distribuir su software de manera abierta con licencia GPL mientras que las otras empresas eran más restrictivas.

En la enseñanza también tenemos ejemplos del uso del software libre en robótica, como es el caso de los LEGO Mindstorms³. Este producto fue originalmente concebido para que los niños aprendieran, pero hoy en día es sin duda el kit de robots más distribuido en cualquier entorno docente. El éxito de este producto es, en parte, por la comunidad

¹ <http://personalrobots.com>

² el anuncio de la versión libre se puede encontrar en <http://slashdot.org/article.pl?sid=00/01/24/110219>

³ <http://www.legomindstorms.com>

que se ha formado alrededor de él y que ha desarrollado por completo un nuevo sistema operativo (BrickOS [4]) y un lenguaje de programación (NQC), desplazando las herramientas cerradas proporcionadas por el fabricante, al tener mejores prestaciones. Los desarrollos de esta comunidad han sobrepasado los límites de los kits que vende LEGO, llevando Mindstorms a su posición privilegiada en el mercado.

El uso de software libre ha tenido también un gran impacto en en la investigación en el campo de la robótica. El software libre ofrece a la comunidad científica la posibilidad de compartir entre ellos las soluciones a diferentes problemas comunes. Esto, por supuesto, es básico para la evolución de la ciencia; pero el software libre va más allá y hace que el proceso suceda de una manera asequible [3]. Como ejemplo de prácticas de software libre en la investigación de la robótica, podemos echar una ojeada a la competición de la Robocup [2]. Robocup⁴ es una iniciativa internacional de investigación y docencia. Su meta es fomentar la investigación en el campo de la inteligencia artificial y de la robótica, proporcionando un problema desafiante (principalmente jugar al fútbol), donde una amplia gama de tecnologías pueden ser examinadas e integradas. La organización del Robocup fomenta el uso del software del libre como manera de mejorar el nivel de la competición. Todo el software producido por la organización se libera bajo licencia de software libre, y la mayoría de los equipos comparten su código, principalmente bajo licencia GPL o LGPL. Otro ejemplo es la plataforma Player/Stage. El software libre presenta también algunas ventajas aquí. Compartiendo sus herramientas, diferentes grupos introducen cambios pequeños en el sistema completo, beneficiándose del trabajo hecho por el resto de la comunidad. Como un ejemplo tenemos el proyecto Player/Stage [1]. Desde el principio el desarrollo ha estado abierto a cualquier contribución, ya que proporcionaban su software bajo licencia GPL. Debido a esto, el proyecto fue utilizado por más investigadores por todo el mundo, construyendo una gran comunidad. Sobre las plataformas citadas anteriormente (Aria y Player/Stage), se construyen arquitecturas de más alto nivel como JDE, que ha sido desarrollado íntegramente por nosotros. JDE es una aproximación jerárquica a una arquitectura robótica de control, inspirada en la etología y basada en la activación selectiva de esquemas. Todo el código se ha desarrollado en el lenguaje C para máquinas GNU/Linux, y está liberado bajo licencia GPL.

Conclusiones La robótica es un sector emergente de la industria de las computadoras que todavía se encuentra en etapas tempranas de investigación y uso. En este trabajo hemos presentado algunos ejemplos de porqué el software libre puede actuar como catalizador para el desarrollo de esta industria.

Hemos mostrado ejemplos donde el uso del software libre ha sido el factor clave para hacer de un producto la referencia de un mercado completo, tanto en la investigación como en la educación. En ambos casos, el software libre es más fácil de adaptar a las necesidades particulares de usuarios y reutilizarlo en otras situaciones. Las comunidades del software libre han demostrado que herramientas libres pueden llegar a ser más populares incluso que las proporcionadas por los fabricantes originales.

Agradecimientos El trabajo de Pablo Barrera ha sido financiado por la Universidad Rey Juan Carlos. El trabajo de Jose M. Cañas, Francisco Martín y Vicente Matellán ha sido financiado parcialmente por el proyecto español ACRACE del MEC, (Dpi2004-07993-c03-01). El trabajo de Gregorio Robles ha sido financiado en parte por la Comisión Europea, bajo CALIBRE CA, el programa de IST, contrato número 004337, y en parte por el Universidad Rey Juan Carlos bajo el proyecto PPR-2004-42.

References

1. **Brian Gerkey, Richard T. Vaughan and Andrew Howard** *The Player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems* Proceedings of the 11th International Conference on Advanced Robotics, pages 317-323, Coimbra, Portugal, June 2003 (ICAR'03).
2. **Hiroaki Kitano, Minoru Asada, Yasuo Kuniyoshi, Itsuki Noda and Eiichi Osawa** *RoboCup: The Robot World Cup Initiative*. Proceedings of the first international conference on Autonomous agents table of contents, 1997. Marina del Rey, California, United States. Pages 340-347.
3. **Tony Meyer**. *Building Cost-effective Research Platforms: Utilising Free — Open-source Software in Research Projects*. Res. Lett. Inf. Math. Sci. (2003) 4, 91-99. <http://iims.massey.ac.nz/research/letters/volume4/10meyer.pdf>.
4. **Markus L. Noga**. *Open-source embedded operating system for the LEGO Mindstorms*. <http://www.noga.de/legOS/>.
5. **Keith O'Hara and Jennifer S.Kay**. *Investigating Open Source Software and Educational Robotics*. The Journal of Computing Sciences in Colleges, Volume 18 , Issue 3 (February 2003), Pages 8-16. <http://elvis.rowan.edu/~kay/papers/OSSEduRob.pdf>.

⁴ <http://www.robocup.org>