

# Herramienta apropiada para el desarrollo de Sistemas Basados en Conocimiento “Shells Jess”

Gloria L. Quispe<sup>1</sup> & Nilda M. Pérez Otero<sup>1</sup>

(1) *Facultad de Ingeniería, Universidad Nacional de Jujuy.*

[glquispe@fi.unju.edu.ar](mailto:glquispe@fi.unju.edu.ar); [nilperez@fi.unju.edu.ar](mailto:nilperez@fi.unju.edu.ar)

**RESUMEN:** Los Sistemas Basados en Conocimiento fueron pensados para trabajar de forma aislada y ahora se conciben integrados con otros sistemas. Un Shell, inicialmente fue diseñado para crear sistemas que podían ser usados para nuevos Sistemas Expertos, agregando conocimiento al dominio del nuevo problema. Jess es un lenguaje escrito en Java y basado por completo en CLIPS. La gran diferencia radica en que Jess consigue ser más rápido que su antecesor; además, Java permite la integración entre ambos lenguajes para realizar aplicaciones de lenguajes orientados a objetos apoyadas por un motor de razonamiento. En este artículo, se realiza una comparación entre las herramientas CLIPS, Jess, y JessGui, y se busca la implementación de una interfaz gráfica mucho más sofisticada.

## 1 INTRODUCCIÓN

La Inteligencia Artificial (IA) es la parte de la Ciencia que se ocupa del diseño de sistemas de computación inteligentes, es decir, sistemas que exhiben las características que asociamos a la inteligencia en el comportamiento humano que se refiere a la comprensión del lenguaje, el aprendizaje, el razonamiento, la resolución de problemas, etc. (Arrúa & Meza, 2003)

Durante años la actividad de la IA, estuvo dedicada a las investigaciones teóricas y al desarrollo de experimentos a través de programas, cuando, en los años setenta, surgió un nuevo paradigma en la IA: los Sistemas Expertos (SSEE).

Un sistema experto es una clase especial de sistema basado en conocimiento (SBC), capaz de resolver un tipo específico de problema, por medio de la emulación de las capacidades de un experto humano, con una programación mínima (Giarratano, 2001).

Un SBC es un cuerpo de programas de ordenador que intenta imitar e incluso superar en algunas situaciones a un experto humano en un ámbito concreto de su actividad. La idea no es sustituir a los expertos, sino que estos sistemas sirvan de apoyo a los especialistas en un “dominio” de aplicación específico.

Se ha pasado de utilizar máquinas específicas (por ejemplo máquinas LISP) a usar ordenadores comunes (estaciones de trabajo, ordenadores personales, etc.). Las aplicaciones eran muy específicas y ahora el ámbito es mucho más amplio. Los SSBCC estaban pensados para trabajar de forma aislada y ahora se conciben integrados con otros sistemas.

A medida que el desarrollo de SSBCC iba aumentando en cantidad y complejidad, la comunidad científica comenzó a buscar formas de desarrollar los sistemas en menor tiempo y con menor esfuerzo. Esto dio lugar al surgimiento de otros productos de SE, a los que denominó shells, ya que ofrecen toda la arquitectura de un SBC.

Un shell es un grupo de paquetes y herramientas de software utilizados para diseñar, desarrollar, poner en operación y mantener SSEE (Carnota, 1987) (Bratko, 1990) (Walker et al., 1990).

Una herramienta para la construcción de SSBCC no es sólo un lenguaje de programación de alto nivel, sino que es un paquete de software que contiene: un motor de inferencias genérico, una interfaz de usuario, y una colección de herramientas que ayudan al desarrollo del SBC y a su uso posterior una vez finalizado (Gómez et al., 1997) (García Martínez y Britos, 2004).

Ya que los shells ofrecen toda la arquitectura de un sistema basado en conocimiento, en este trabajo se propone el desarrollo de la herramienta Jess, haciendo una comparación con JessGui y CLIPS.

Se tomó como referencia el trabajo de Quispe y Pérez Otero (2012) en el que se realizó una tabla comparativa de las herramientas comerciales y gratuitas, donde se determinó que la mejor herramienta de entorno de desarrollo es CLIPS, y en el trabajo de investigación presentado para la beca de iniciación para la promoción en investigación para alumnos de la UNJu (2008).

Este trabajo se organiza de la siguiente forma. En la próxima sección se describe los Shell CLIPS y Jess. En la sección 3 se realiza el análisis de las herramientas apropiadas para un

SSBBCC y en la sección 4 se encuentra la conclusión del trabajo.

## 2 CLIPS Y JESS

### 2.1 CLIPS

Lenguaje C integrado para sistemas de producción (C Language Integrated Production System) es una herramienta productiva para el desarrollo y la distribución de Sistemas Expertos. CLIPS proporciona un ambiente completo del lenguaje para la construcción de sistemas basados en reglas o en reglas y objetivos y es el primer lenguaje en proporcionar una utilidad de verificación y validación para el desarrollo de sistemas expertos. Las características clave de CLIPS residen en la representación del conocimiento, la portabilidad, la integración, el desarrollo interactivo y orientado a objetos. CLIPS ha sido utilizado por cerca de 5.000 empresas tanto en el sector público como privado, incluyendo la NASA (donde se desarrolló inicialmente en 1985), divisiones militares de los Estados Unidos, entidades federales, contratistas del gobierno, universidades y empresas privadas.

### 2.2 Jess (Java Expert System Shell)

Es un motor de reglas y ambiente de desarrollo escrito en el lenguaje Java de Sun por Ernest Friedman-Hill en los Laboratorios Sandia National (Livermore, Estados Unidos). Originalmente la herramienta estuvo inspirada en CLIPS, pero ha crecido por su propia cuenta influenciada por el ambiente Java al cual pertenece. Utilizando Jess se pueden construir applets de Java y aplicaciones que tienen la capacidad de “razonar” utilizando el conocimiento que se les proporciona a través de reglas declaradas. Jess es bastante rápido y para algunos problemas posee un rendimiento que supera a CLIPS (especialmente si se utiliza un buen compilador Justo a Tiempo, Just In Time - JIT). Los aspectos fundamentales del lenguaje Jess son compatibles con CLIPS y muchos procedimientos Jess son válidos en CLIPS y viceversa. Jess adicionalmente es un ambiente de programación para Java en el cual se pueden crear objetos y métodos Java sin compilar código Java de manera directa. En la Fig. 1 se presenta la arquitectura del intérprete Jess, y en la Fig. 2 se presenta Jess: CLIPS reprogramado en java, con ligeras variaciones e integrado en el lenguaje.

### 2.3 JessGui (A Gui for Jess)

Desarrollada por Jovanovic (2004), JessGui es una interfaz gráfica de usuario desarrollada sobre el sistema experto Jess. La idea central de JessGui es hacer una interfaz gráfica de usuario mucho más fácil de usar y flexible a las pruebas. Hay muchas otras herramientas de SSBCC que proporcionan un entorno de desarrollo integrado y cómodo para los desarrolladores de sistemas expertos. Sin embargo, todos ellos son o bien productos comerciales o propietarios. Jess y JessGui son software gratuito de código abierto, muy adecuadas para la creación de aplicaciones de sistemas expertos mucho más complejos, tanto independientes como basados en la web.

Por lo tanto el uso de estas herramientas cobra importancia por el nivel de aplicaciones que tiene Jess.

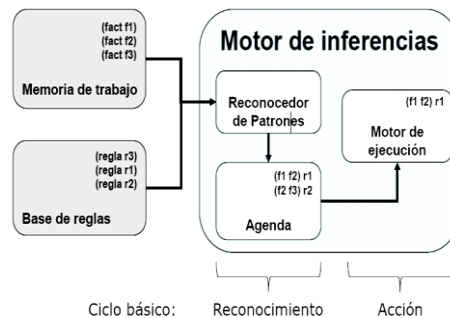


Figura 1. Arquitectura del Intérprete Jess

## 3 ANÁLISIS DE LAS HERRAMIENTAS APROPIADAS PARA UN SSBCC

El análisis comparativo de Herramientas, presentado por Quispe y Pérez Otero (2012), ha permitido aportar a los desarrolladores de SSBCC una eficaz elección de las herramientas apropiadas. A partir de ese análisis comparativo, se ha mejorado e incorporado otra herramienta, JessGui, citada anteriormente. En la Tabla 1, se presentan los resultados de la nueva evaluación de las herramientas.

Se puede decir que la mejor herramienta de entorno de desarrollo sigue siendo CLIPS, a continuación se encuentra la herramienta JessGui. CLIPS es de libre distribución, y sus características son: potencia de representación del conocimiento, flexibilidad, comodidad, sencillez de la interfaz del usuario, eficiencia, robustez, traza, y depuración e interfaces. Según el proceso de evaluación y selección, fijados con sus respectivos pesos, Tabla 1, se ha realizado un aporte de conocimiento en este trabajo de investigación, para todos aquellos interesados en el desarrollo de SSBCC.

Hay que destacar que, a partir de las ampliaciones realizadas en Jess, se rediseña la GUI de la aplicación para facilitar el seguimiento de la ejecución de una forma más intuitiva, lo que nos permite utilizar la interfaz gráfica de forma más sofisticada. Jess se puede

integrar en Java y acceder a sus clases y librerías. Y viceversa, Java se puede integrar en Jess. Por tal razón Jess se puede integrar a un agente inteligente como por ejemplo Jade, como así también se hace uso de Jess en la interfaz web para validación de documentos XML, etc.

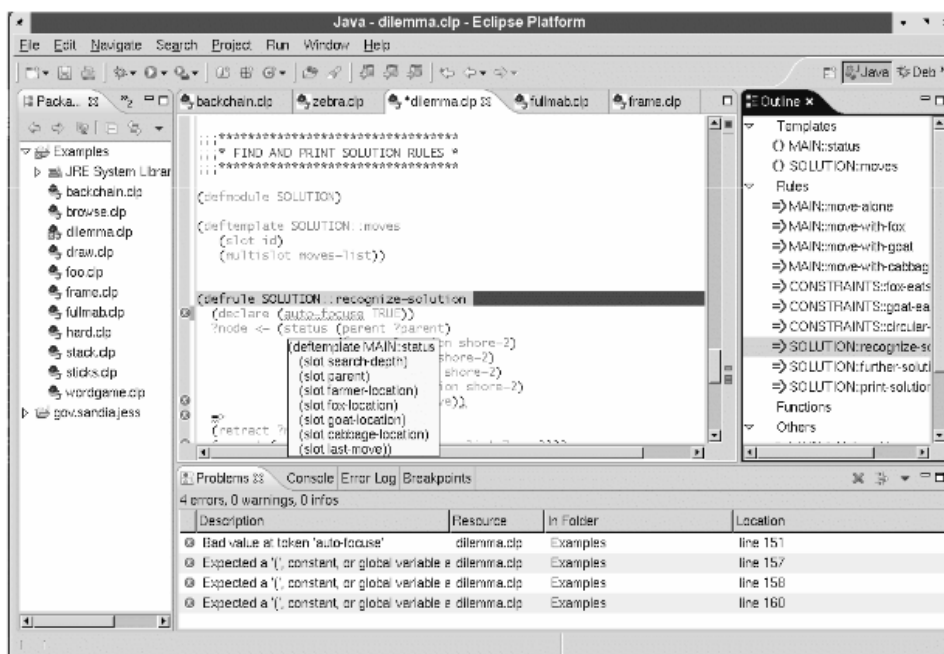


Figura 2. Jess: CLIPS reprogramado en java

El uso de Jess ha permitido utilizar potentes funcionalidades completas de sistemas expertos (por ejemplo, la apertura, la flexibilidad), lo cual permite continuar este trabajo de

investigación a los fines de determinar cual resulta mejor en la implementación de una interfaz Java basado en Jade o en la web.

Tabla 1. Evaluación de las herramientas

	Peso	JessGui	CLIPS	Jess
<i>Inferencia</i>				
Encadenamiento hacia delante	7.6	8 (Si)	8 (Si)	5 (No)
Encadenamiento hacia atrás	7.8	7 (Si)	9 (Si)	7 (Si)
Herencia	7	8 (Si)	9 (Si)	7 (Si)
<i>Representación</i>				
Reglas	8	8 (Si)	9 (Si)	7 (Si)
Marcos	6.2	8 (Si)	8 (Si)	8 (Si)
Lógica	7	8 (Si)	8 (Si)	6 (Si)
<i>Interfaz con el usuario</i>				
Editor gráfico	6.4	8 (Si)	9 (Si)	6 (Si)
Menús	7.2	8 (Si)	8 (Si)	4 (Si)
Ventanas múltiples	7.2	8 (Si)	10 (Si)	0 (No)
Seguimiento	8	8 (Si)	10 (Si)	0 (No)
Explicaciones incorporadas	8	8 (Si)	10 (Si)	0 (No)
Total		635.4	718.8	356.4

#### 4 CONCLUSIÓN

Con el presente trabajo, Herramientas apropiadas para el desarrollo de SSBCC “Shell Jess”, se espera lograr:

- Que Jess y JessGui se utilicen como herramientas prácticas para apoyar y ayudar a todos aquellos profesionales, investigadores y estudiantes interesados en el desarrollo de SSBCC.

- Una interfaz gráfica de usuario integradora y rica, con el fin de proporcionar características inteligentes originalmente no compatibles con Jess.

El análisis de estas herramientas es fundamentalmente para ayudar al Ingeniero de conocimiento y todas las personas que trabajan en el desarrollo de SSBCC. Por ello, la comparación de las herramientas Jess, JessGui, y CLIPS, brinda un aporte de conocimiento la cual da como resultado la herramienta que mejor se adecua para el desarrollo de un determinado SBC.

## 5 REFERENCIAS

- Arrúa, L. & Meza F. M., *Sistemas Expertos*. 2003.
- Jovanović, J. *Expert Systems Shells Jess and JessGUI*, Faculty of Organizational Sciences & TEMPUS, Belgrade, 2004.
- Bratko, Ivan, *Prolog programming for Artificial Intelligence*, Addison Wesley, Segunda Edición, New York, 1990.
- Carnota, R. J. & A. D. Teszkiewicz, *Sistemas Expertos y Representación del Conocimiento*, Edición - EBAI, Octubre de 1987.
- García, Martínez, R. & P. V. Britos, *Ingeniería de Sistemas Expertos*, Ed. Nueva Librería, 2004.
- Giarratano, J. y G. Riley, *Sistemas Expertos – Principios y Programación*, 3° Ed. International Thomson Editores, México, 2001.
- Gómez, A., Juristo, N., Montes, C. y Pazos J. *Ingeniería del Conocimiento*. Ed. Centro de Estudios Ramón Areces S.A, 1997.
- Quispe, G. L. & N. M. Pérez Otero, Análisis comparativo de herramientas apropiadas para el desarrollo de Sistemas Basados en Conocimiento. *Investigaciones en Facultades de Ingeniería del NOA*. Editorial científica universitaria Tucumán pp. Octubre 2012.
- Walker, A, Michael, M. C., John, S. & K. Walter Wilson, *Based Systems and Prolog*, Addison Wesley, Segunda Edición, New York, 1990.