

Capítulo 3. Tecnología de Agentes Inteligentes y la

Metodología Prometheus

3.1 Transición de inteligencia artificial a agentes inteligentes

Durante las décadas pasadas el surgimiento de la inteligencia artificial (IA) tuvo muchos adeptos y críticos como cualquier otra tecnología en desarrollo. Es interesante mencionar que durante algún tiempo se esperaron cosas extraordinarias de la inteligencia artificial por ejemplo, que las computadoras pudieran adquirir la habilidad de pensar como un ser humano. Sin embargo al paso del tiempo se vinieron abajo muchas expectativas acerca de la IA y su interés decayó [Bigus, 01], pero la expectativa de poder crear software que se comporte de forma más inteligente jamás ha dejado de estar latente. La IA trajo consigo un gran número de tecnologías y otras que se complementaban, como procesamiento de símbolos, algoritmos más eficientes de búsqueda, algoritmos genéticos, representación del conocimiento, sistemas expertos, sistemas de razonamiento y sistemas de aprendizaje.

El los años 90s el tema de agentes comenzó a adentrarse muy de lleno en la mayoría de los desarrollos, impulsados por la implantación de Internet. Características como autonomía, movilidad y colaboración los hacían demasiado atractivos para ser implantados en ambientes donde estas aptitudes se necesitaban. El paradigma de agentes tuvo sus fundamentos en algunas tecnologías conceptuales como *frames* y el paradigma orientado a objetos [Bigus, 01].

La transición de la inteligencia artificial (IA) al desarrollo de agentes inteligentes (AI) se ha dado poco a poco durante finales de los 90s y principios de este nuevo siglo.

Esta transición como tal, le podríamos llamar una reactivación de la IA, se dio a partir de implementar algoritmos o técnicas de aprendizaje en agentes; anteriormente los agentes estaban diseñados para procedimientos específicos o ambientes colaborativos en el cual las reglas y protocolos estaban detallados en su totalidad. En la actualidad los agentes que tienen pequeñas tareas pueden llegar a evaluar si están optimizando cierta tarea, con lo cual pueden modificar sus propias reglas de comportamiento. Esto representa el principio en el camino hacia el anhelo de desarrollar software con un comportamiento más inteligente. La IA como tal no ha muerto, más bien se ha integrado con otras tecnologías, para crear aplicaciones que cada vez nos sorprenden más. Algunos de estos detalles los podemos encontrar en robocup.org [Robocup, 06].

3.2 Agentes inteligentes

De forma inicial definimos que un agente es un componente de software o hardware que es capaz de actuar de forma autónoma para completar tareas de parte del usuario [Jenings, 98]. Muchas personas probablemente sientan que esta definición no cubre en su totalidad la idea que tengan de agentes, la razón es que existen bastantes definiciones de agentes dependiendo del contexto o aplicación. La autonomía, por ejemplo, es una de las características principales, pero hay aplicaciones en las cuales el agente no posee esta característica y solo reacciona a estímulos por parte del usuario o ambiente.

Como se mencionó anteriormente existe una variedad de agentes los cuales se pueden agrupar en estas categorías:

- Agentes de colaboración
- Agentes de interfaz
- Agentes móviles
- Agentes de información o Internet
- Agentes reactivos
- Agentes híbridos
- Agentes inteligentes

Por otra parte, un agente puede tener los siguientes atributos:

- **Autonomía:** Un agente puede operar sin la intervención directa de un humano o alguna especie de control sobre sus acciones y estado interno.
- **Movilidad:** Es la capacidad de una entidad de software de poder migrar de ambiente en caso de ser necesario. Esto se ve aplicado cuando el agente tiene acceso a Internet y puede moverse en distintos ambientes.
- **Inteligencia:** Los agentes con esta habilidad deben ser capaces de reconocer eventos, determinar su significado y de ahí llevar a cabo acciones.
- **BDI :** (*belief, desire, intentions*) Esto se refiere a creencias, deseos e intenciones del agente. Las creencias son todo aquello que se tiene definido como conocimiento inicial de su entorno y responsabilidades, los deseos son los objetivos que deba de cumplir el agente y por último las intenciones es el plan que se llevará a cabo para cumplir esos objetivos.

- **Reactividad:** Los agentes perciben su entorno (el cual puede ser el mundo físico o por medio de la interfaz de un usuario, Internet o tal vez todo combinado) y responden a los cambios que perciben [Wooldridge, 95].
- **Proactividad:** Los agentes no solo deben responder hacia un estímulo del ambiente sino también ser capaces de demostrar iniciativa [Wooldridge, 95].
- **Sociabilidad:** Un agente puede tener una comunicación con otro agente o un ser humano.

Durante el desarrollo de alguna solución dentro de la cual usemos agentes inteligentes siempre existen preguntas que surgen, como que tanta inteligencia o capacidades de razonamiento queremos dotar a nuestro agente?, ya que esto reduce la capacidad de movilidad del agente, el concepto de movilidad lo tenemos definido en la página anterior. Los deseos, creencias e intenciones del agente pueden ir de forma explícita en la misma programación del agente o pueden tener su base de conocimiento, por ejemplo en una base de datos que pueda ser modificada constantemente.

La organización FIPA (*Foundation for intelligent physical agents*) [FIPA, 05] que se ha dedicado a establecer los distintos estándares en el desarrollo de agentes inteligentes. Algunas organizaciones como, por ejemplo AUML [AUML, 05] ofrecen técnicas de modelado de agentes en lenguajes basados en el estándar UML de *Rational*. Para el desarrollo del sistema de selección de personal, se apoyó en una metodología de diseño

de agentes inteligentes, en este caso la seleccionada fue *Prometheus* la cual se explica en la sección 3.3.

3.3 Metodología Prometheus

Prometheus es una metodología para el desarrollo de agentes inteligentes creada por *Lin Padgham* y *Michael Winikoff*, en la cual se define un lenguaje de modelado relativamente sencillo, que forma parte de los fundamentos de AUML junto con otras metodologías. *Prometheus* ha sido desarrollado de manera conjunta con *Agent Oriented Software*, su ambiente de desarrollo PDT (*Prometheus Design Tool*) actualmente está en constante desarrollo. En la figura 3.1 se puede apreciar una imagen de la herramienta.

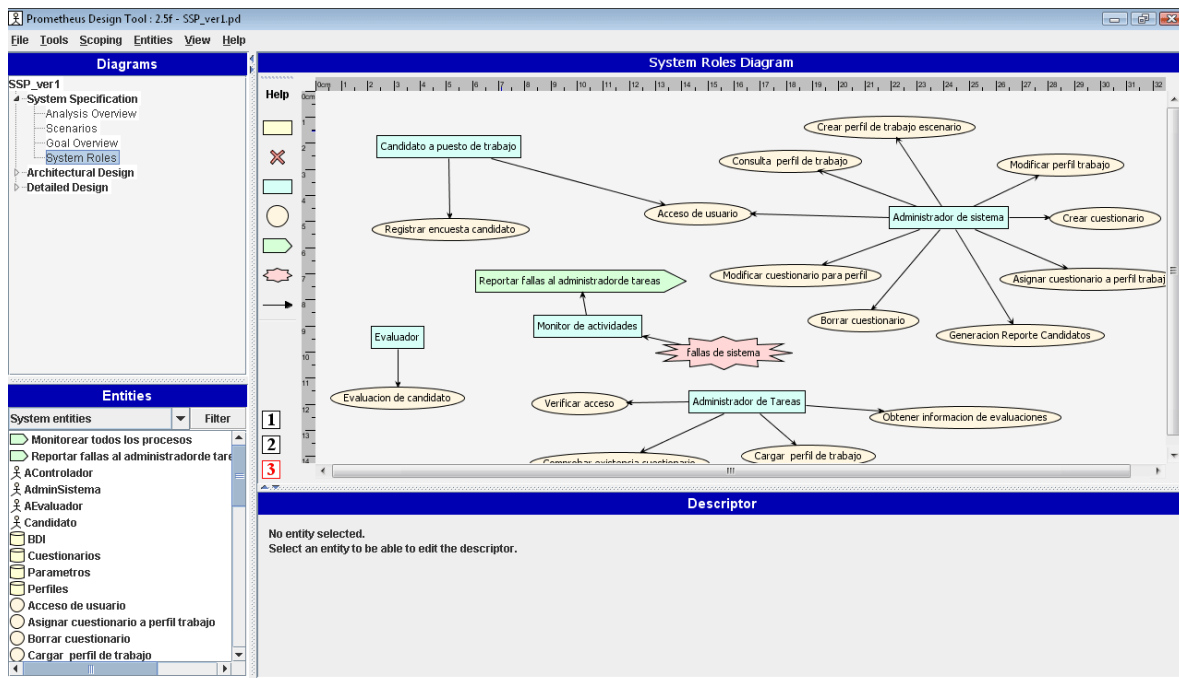


Figura 3.1 *Prometheus Design Tool* [Padgham, 02]

Algunas de las características de *Prometheus* son:

- Ofrece soporte para desarrollo de agentes inteligentes que contengan creencias, objetivos, planes y eventos.
- Cuenta con distintos artefactos de diseño para representar esquemas.
- Tiene una estructura jerárquica que permite diseñar múltiples niveles de abstracción.
- En vez de utilizar el diseño lineal en cascada para el proceso de ingeniería de software, usa procesos iterativos.
- Ha sido utilizada tanto en el ambiente académico como en la industria.

Un diagrama que nos ofrece una visión global de lo que es la metodología lo podemos consultar en la figura 3.2.

La metodología *Prometheus* se puede simplificar en 3 etapas básicas:

- Especificaciones del sistema
- Arquitectura del sistema
- Diseño detallado del sistema

Dentro de las especificaciones del sistema se deben desarrollar casos de uso, identificar totalmente los objetivos primordiales, diagramas de escenarios, objetivos, roles, identificar la interface del agente con su ambiente (percepciones, acciones e información externa) y describir las funcionalidades básicas.

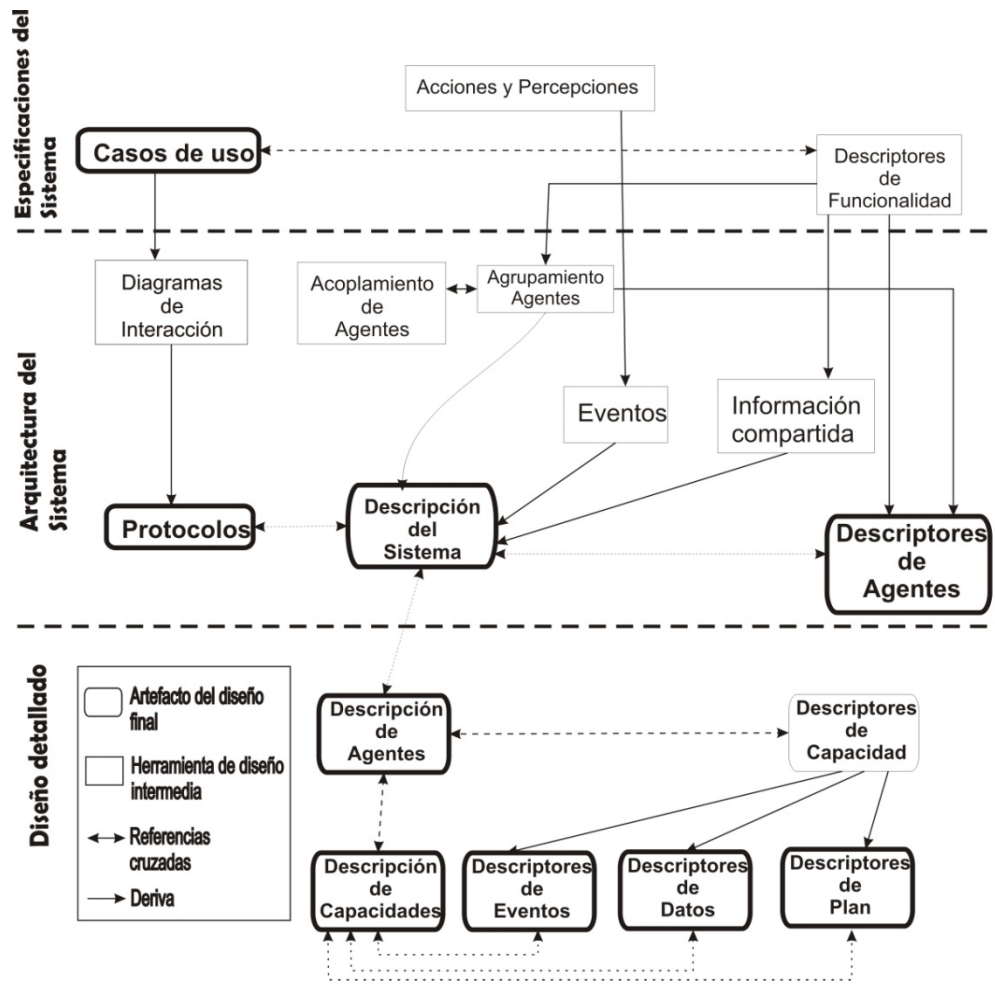


Figura 3.2 Metodología de diseño *Prometheus* [Padgham, 02]

En la etapa de especificaciones del sistema y arquitectura es muy importante conocer ciertos conceptos que utiliza esta metodología:

- **Percepción:** Es toda aquella información que provenga del entorno.
- **Evento:** Es aquella información que se percibe por medio de la percepción pero tiene relevancia importante.
- **Acciones:** Es la forma en que un agente interactúa con su entorno.
- **Actores:** Son cualquier persona o rol que interactúan con el sistema, estas entidades pueden ser humanas o entidades del sistema.

- **Escenarios:** Son una secuencia de pasos que describen una operación del sistema.
- **Protocolo:** Son aquellas funciones o interacciones bien identificadas, por ejemplo la solicitud de datos por parte de cierto agente a otro.
- **BDI:** El término BDI es referente a creencias, deseos e intenciones. Las creencias son el conocimiento que el agente tiene de su ambiente, los deseos son los objetivos que tiene y por último las intenciones representa que es lo el agente decide por llevar a cabo [Wikipedia, 06].

Respecto a la arquitectura del sistema se deben definir funcionalidades para determinar los tipos de agentes que se necesitarán desarrollar, el desarrollo de descriptores de agentes que son aquellos que describen la funcionalidad y especificaciones de un agente, el diseño de un diagrama general describiendo toda la estructura del sistema y diagramas de agrupamiento agente-rol.

En el diseño detallado se desarrollan diagramas de procesos, diagramas de los agentes para conocer sus funcionalidades internas, y la definición de las creencias y conocimiento. Como se puede apreciar es una metodología muy sencilla de aplicar y debido a esto ha sido utilizada con frecuencia en algunas universidades. Por ejemplo, los estudiantes de segundo semestre de la universidad de RMIT en Australia pueden lograr crear sistemas basados en agentes inteligentes de una manera eficiente [Padgham, 02].

Para el desarrollo de esquemas y diagramas de *Prometheus* los diseñadores de la metodología recomiendan el software PDT [PDT, 05] pues posee un ambiente de desarrollo muy completo y sencillo de utilizar.

La simbología utilizada para los distintos esquemas se muestra en la figura 3.3:



Figura 3.3 Simbología utilizada en metodología *Prometheus* [PDT, 05].

El sistema presentado en este proyecto ha sido diseñado apoyándose en esta metodología.

El siguiente capítulo explica con detalle su funcionamiento.

3.4 Resumen

En el capítulo 3 se introdujo a la base tecnológica que se implementó en el sistema SSPi, que son los agentes inteligentes, también se explica la metodología de diseño *Prometheus*, la cual fue seleccionada para el desarrollo del sistema.

Ésta metodología forma parte de los fundamentos del estándar AUML, consta con una herramienta de diseño llamada PDT (*Prometheus Design Tool*) [Padgham, 02] y se divide en 3 grandes fases: especificaciones del sistema, arquitectura del sistema y diseño del sistema.