



Metodologías para Sistemas Multi-agente

Curso Doctorado
Sistemas Multi-agente



Índice

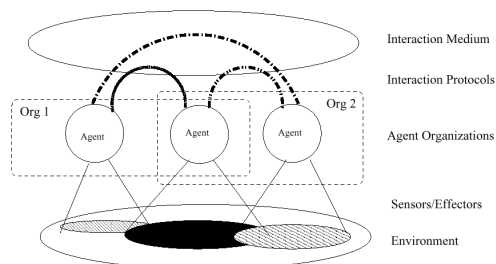
- Conceptos. Introducción
- Metodologías
 - BDI
 - GAIA
 - AUML
 - Message
- Conclusiones

Conceptos. Introducción

- Modelar sistemas reales complejos y con características claramente distribuidas
- Visión de un sistema como una organización computacional consistente de varios "roles" interactuando.

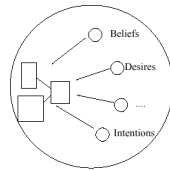
Conceptos. Introducción

- Identificar:
 - los diferentes subsistemas que forman parte del sistema global
 - las posibles interacciones y dependencias entre ellos

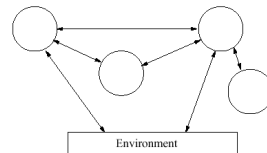


Conceptos. Introducción

- A tener en cuenta:
 - punto de vista interno → un agente
 - punto de vista externo → varios agentes



The Intra-Agent Viewpoint
(agent's own purpose, internal structure, technology)



The Inter-Agent Viewpoint
(interaction with the environment and with other agents)

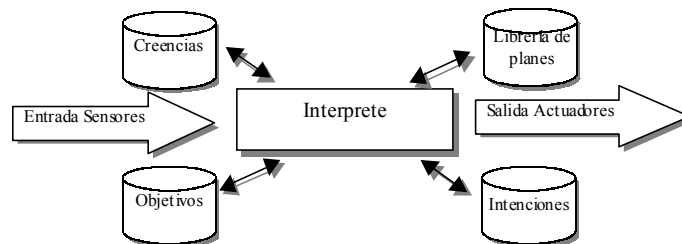
Metodologías existentes

- Diferentes aproximaciones de metodologías orientadas a agente pueden ser analizadas en la actualidad
- Podemos destacar los trabajos de:
 - Kinny (BDI),
 - Wooldridge y Jennings (GAIA),
 - Parunak y Odell (AUML),
 - Garijo et al. (MESSAGE)
 - Existen más: MaSE, MASSIVE, ...

Casos: BDI

- Idea:

- Trata de explorar como las técnicas de modelado OO se pueden extender para aplicarse a sistemas de agente basados en la arquitectura BDI.



Casos: BDI

- Se trabaja a dos niveles de abstracción.

- Un punto de vista *externo*:
 - el sistema es modelado como una jerarquía de clases de agente, los agentes individuales son instancias.
 - Las clases de agentes están caracterizadas por su propósito, sus responsabilidades, los servicios que desarrollan, la información acerca del mundo que requieran y las interacciones externas.
- Desde el punto de vista *interno*:
 - Conjunto de modelos los cuales permiten estructurar el estado de motivación y de información de los agentes y las estructuras de control que determinan sus conductas.



Casos: BDI

- El punto de vista externo puede ser capturado en dos modelos (independientes de la arquitectura BDI):
 - Modelo de agente:
 - describe la relación jerárquica entre diferentes clases abstractas y concretas de agente.
 - permite también identificar las instancias de agente que deben existir en el sistema.



Casos: BDI

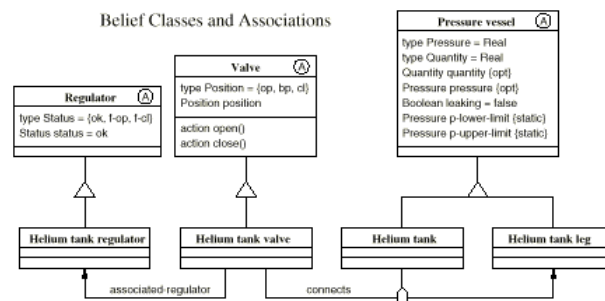
- Modelo de interacción:
 - describe las responsabilidades de una clase de agente, los servicios que provee, interacciones asociadas y relaciones de control entre clases de agente.
 - Incluye la descripción de los mensajes para la comunicación entre agentes y entre un agente y otros componentes del sistema.

Casos: BDI

- En el punto de vista interno cada clase de agente se especifica en tres modelos:
 - Modelo de creencias: describe la información acerca del entorno y el estado interno que un agente de una clase puede tener y las acciones que puede realizar.
 - Modelo de objetivos: describe los objetivos que un agente puede adoptar, y los eventos a los que debe responder.
 - Modelo de planes: describe los planes que un agente puede emplear para alcanzar sus objetivos o responder a eventos que percibe.

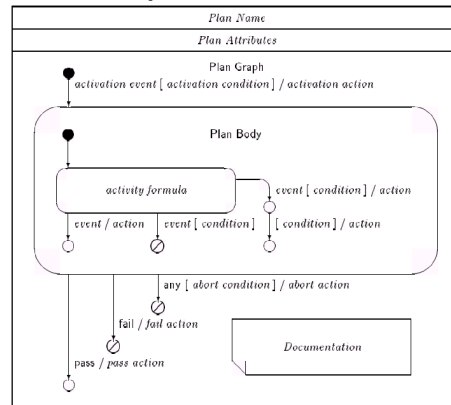
Casos: BDI

- Ej. modelo de creencias:



Casos: BDI

■ Ej. modelo de planes:



Casos: BDI

■ Conclusiones:

- Provee elementos para modelar y especificar sistemas multi-agente.
- Se proponen técnicas de modelado para describir las perspectivas internas y externas de sistemas multi-agente basados en la arquitectura BDI.
- su construcción se basa en modelos OO existentes.
- La metodología se centra en la parte interna dejando un poco oscura la forma de diseñar externamente un agente, esto es, sus interacciones y organizaciones.



Casos: GAIA

- Se centra en la idea de que la construcción de sistemas basados en agente es un proceso de diseño organizacional.
- Los principales conceptos que aparecen en la metodología se dividen en dos: *abstractos* y *concretos*.
- Las entidades *abstractas* son aquellas que son empleadas durante el análisis para la conceptualización del sistema.
- Las entidades *concretas* son empleadas en el proceso de diseño.



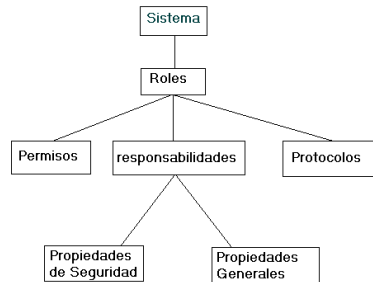
Casos: GAIA

- La entidad más abstracta de un sistema en la *jerarquía de conceptos* que se presenta es el *sistema* relacionado con la idea de “sociedad” u “organización”.
- Idea: ver un sistema informático definiéndolo por un conjunto de roles teniendo en cuenta una visión organizacional del mundo.
- Un rol es definido a su vez por tres atributos: *responsabilidades*, *permisos* y *protocolos*.

Casos: GAIA

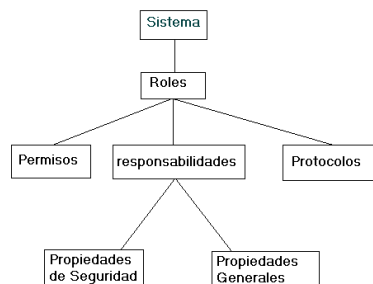
Un rol es asociado a:

- un **conjunto de permisos**, derechos asociados al rol, identifican los recursos que están disponibles al rol para realizar sus responsabilidades.
- un **conjunto de protocolos**, los cuales definen la manera de interactuar del rol con otros roles.



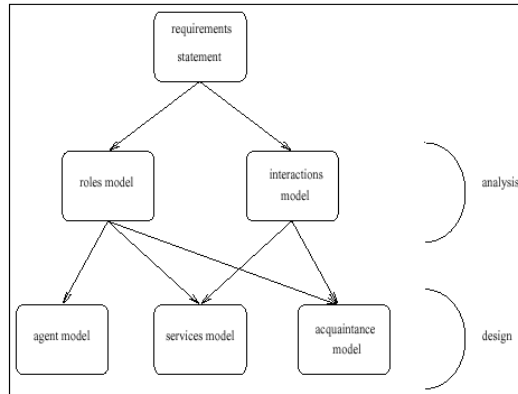
Casos: GAIA

- Las **responsabilidades** determinan la funcionalidad, podemos ver dos tipos:
 - **propiedades de viveza**: describen aquellos estados de los asuntos que un agente debe efectuar. Decir que “algo será hecho”
 - **propiedades de seguridad**: son invariantes. condiciones de seguridad



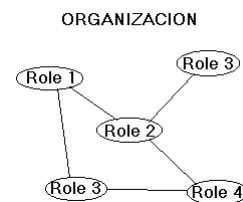
Casos: GAIA

■ Modelos presentados



Casos: GAIA

■ El proceso de análisis



En resumen el proceso de análisis se compone de:

- Identificar los roles del sistema
- Para cada papel identificar y documentar los protocolos asociados
- Empleando el modelo de protocolos como base, elaborar con más detalle el modelo de roles.



Casos: GAIA

- **El proceso de diseño**

En resumen el proceso de diseño se compone de las siguientes fases:

- Creación de un modelo de agente, agregando roles a los tipos de agentes y documentando las instancias de cada tipo de agente.
- Desarrollo de un modelo de servicios, examinando protocolos y propiedades de viveza y seguridad.
- Desarrollo de un modelo de conocimiento, por medio del modelo de interacción y el modelo de agente.



Casos: AUML

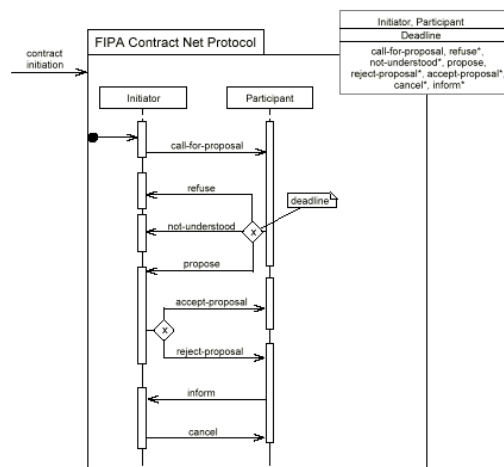
- Se centra en intentar emplear herramientas de desarrollo ya existentes
 - como puede ser el caso de UML
- Orientándolas hacia el campo de los agentes.
- La visión que se presenta de un agente es como el siguiente paso a partir del concepto de objeto
- AUML sintetiza el interés por disponer de metodologías de desarrollo orientadas a agentes con la aceptación de UML.

Casos: AUML

- UML es insuficiente para modelar agentes y sistemas basados en agentes.
 - Si los comparamos con los objetos, los agentes son activos.
 - Sus actividades incluyen objetivos y condiciones que guían la ejecución de las tareas definidas.
 - Toman la responsabilidad de sus necesidades.
 - Los agentes actúan de igual forma solos o con otros agentes.
 - Forman una comunidad social de miembros interdependientes que actúan de forma autónoma.

Casos: AUML

- Ejemplo de empleo de notación UML para definir un protocolo de interacción entre agentes





Casos: AUML

- Presenta en la actualidad un conjunto de extensiones de UML para:
 - la especificación de protocolos de interacción de agentes
 - la representación de estructuras sociales y organizativas entre agentes



Casos: AUML

- Conclusiones sobre AUML:
 - UML está lo suficientemente aceptado y es evidente que alguna de sus herramientas pueden ser aplicadas directamente a sistemas basados en agentes adoptando algunas convenciones.
 - Actualmente se está trabajando en esta aproximación, sugiriéndose extensiones a UML para que se de soporte a la funcionalidad adicional que aportan los agentes.

Casos: MESSAGE

(*Methodology for Engineering Systems of Software Agents*)

- Cubre análisis y diseño de MAS.
- Realiza inicialmente una definición de conceptos
- Presenta una notación basada en UML siempre que es posible.
 - Los diagramas son extensiones sobre diagramas de clase y de actividad.
 - Incorpora las extensiones de AUML para diagramas de interacción.

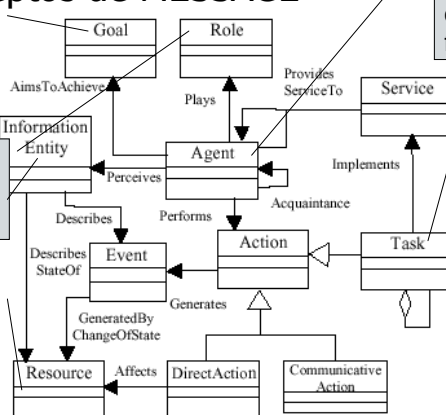
Casos: MESSAGE

■ Conceptos de MESSAGE

Asocia un agente con una situación, el agente intentará alcanzar la situación

Características externas de un agente
Un objeto encapsulando un trozo de información

Entidad no autónoma como bases de datos o programas externos



Entidad atómica autónoma capaz de desarrollar alguna función útil

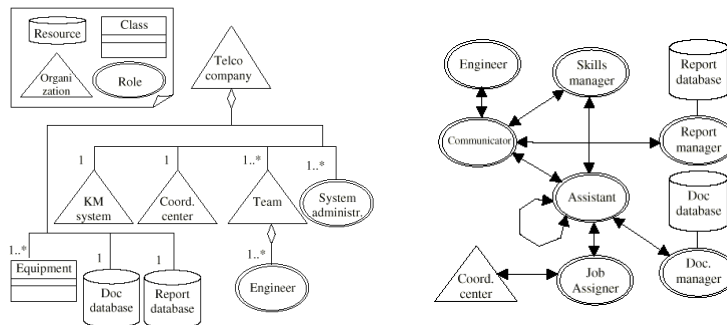
Unidad de actividad del nivel de conocimiento

Casos: MESSAGE

- Fase de análisis:
 - Su propósito es producir un conjunto de modelos en los que estén de acuerdo el analista y el usuario.
 - Incorpora diferentes modelos.
 - Se realiza por refinamientos:
 - level 0: definir el sistema con respecto su entorno. El sistema es visto como un conjunto de organizaciones que interactúan con recursos, actores u otras organizaciones.
 - level 1: la estructura y conducta de entidades como organización, agentes, tareas, objetivos son definidas
 - Niveles adicionales pueden ser definidos para analizar aspectos específicos del sistema.

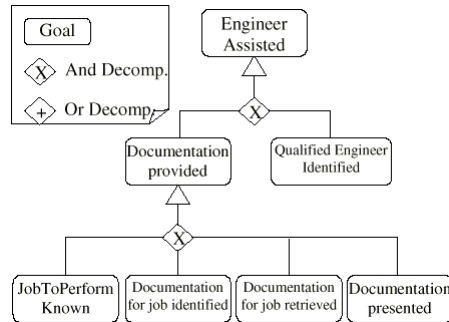
Casos: MESSAGE

- Modelos en la fase de análisis:
 - **Organisation view:** muestra entidades concretas (agentes, roles, recursos) en el sistema y en el entorno. Indica relaciones entre las entidades.



Casos: MESSAGE

- Modelos en la fase de análisis:
 - **Goal/Task view:** muestra objetivos, tareas, situaciones y las dependencias entre estos conceptos. Se emplea un diagrama de actividad.



Casos: MESSAGE

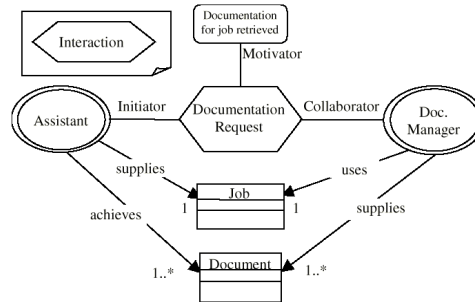
- Modelos en la fase de análisis:
 - **Agent/Role view:** se centra en los roles y agentes individuales. Para cada agente/rol se desarrolla un esquema de la forma:

Role Schema	Assistant
Goals	JobToPerformKnown, DocumentationForJobRetrieved
Capability	Some learning capability is required to keep the profile of the engineer updated on the basis of the completed job.
Knowledge, Beliefs	A profile of the skills of the engineer to be used to evaluate if and how the engineer can provide help to a colleague requesting assistance A profile
Agent requirements	This role will be played by the agent that actually assists the Engineer.

Casos: MESSAGE

- Modelos en la fase de análisis:

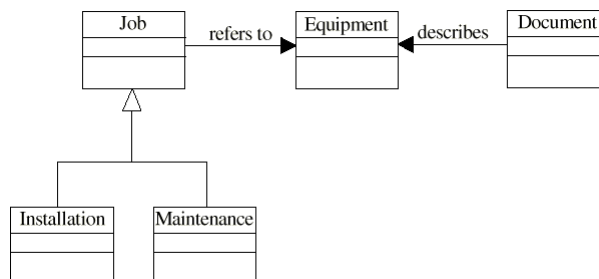
- Interaction view:** para cada interacción entre agentes/roles se indica el iniciador, los colaboradores, la motivación (un objetivo), la información suministrada, los eventos que la disparan.



Casos: MESSAGE

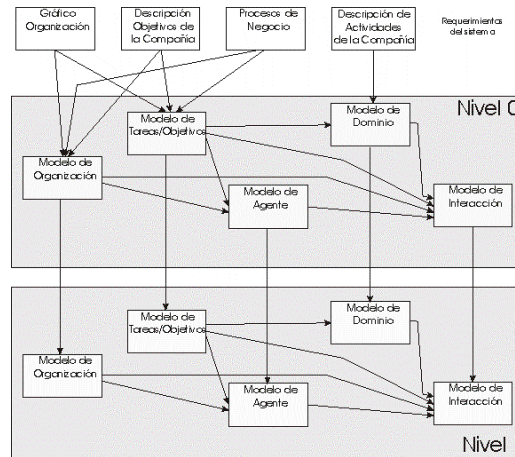
- Modelos en la fase de análisis:

- Domain view:** muestra los conceptos específicos del dominio y sus relaciones.



Casos: MESSAGE

■ fase de análisis:



Casos: MESSAGE

Diseño

- En el modelo de diseño realmente no presentan un único proceso sino que plantean dos posibles aproximaciones:
 - Aproximación 1: Una primera aproximación donde el diseño es dirigido por la organización del sistema multiagente y la arquitectura.
 - Aproximación 2: La segunda aproximación está orientada por una plataforma de agente concreta.

Casos: MESSAGE

Diseño (Aproximación 1)

- Podría verse como una aproximación de alto nivel, donde un agente es un subsistema formado por componentes.
- El modelo de organización obtenido en el análisis dirigirá el proceso de diseño.
- Esta aproximación trata de ser lo más independiente posible de la arquitectura de agente que se elija para un agente concreto.
- El diseño en principio está basado en el empleo de patrones arquitectónicos.

Casos: MESSAGE

■ Diseño (Aproximación 1)





Casos: MESSAGE

Diseño (Aproximación 2)

- Esta aproximación podría verse como un proceso de diseño a bajo nivel, ya que es específico de una plataforma de agente
- Una plataforma de agente provee ya de ciertos servicios: entorno de ejecución, protocolos de coordinación, ontologías, herramientas de desarrollo ...
- Los autores de MESSAGE emplean una plataforma FIPA y en concreto la implementación JADE, basada en un modelo de
- Programación OO y empleando JAVA como lenguaje de programación.
- Por tanto, en este caso, los artefactos a desarrollar deben estar en términos OO.



Conclusiones

- A nivel conceptual la mayoría de propuestas introducen, a lo largo de su desarrollo, términos muy similares.
- La mayoría de propuestas intentan cubrir fundamentalmente las etapas de análisis y diseño de sistemas multiagente.
- La existencia de herramientas de desarrollo asociadas no es ni mucho menos un aspecto generalizado.
- En lo que se refiere al empleo de una arquitectura de agente concreta, destacar que existen distintas alternativas.