

# Análisis y Diseño de Sistemas Multi-Agente

Curso de doctorado 2001-02:  
Agentes Software

Jorge Gómez Sanz  
Juan Pavón Mestras  
Dep. Sistemas Informáticos y Programación  
Universidad Complutense Madrid

## Introducción

- Hemos visto
  - Qué son los agentes
  - Varios tipos de arquitecturas de agentes
  - Mecanismos de seguridad en agentes
  - Estándares de agentes
  - Aplicaciones de agentes
  
- Pero,  
*¿cómo se construye un sistema multiagente?*

## Tendencias

---

- En general, los sistemas de agentes aplican una mezcla de técnicas
  - Orientación a objetos:
    - estructura del sistema
    - asignación de responsabilidades
  - Sistemas expertos
    - definición del comportamiento
    - representación de conocimiento
  - Inteligencia artificial distribuida
    - organización
    - comunicación de conocimiento
    - coordinación

## Tendencias

---

- Las metodologías de ingeniería de agentes
  - Intentan aplicar métodos de IS convencionales a sistemas de agentes, extendiéndolos
    - Agent UML (AUML) [Odell et. al 99]: extiende UML con protocolos de interacción de agentes
    - MESSAGE Methodology for Engineering Systems of Software Agents [Message]
  - o intentan desarrollarse a partir de alguna metodología de IA o KBS
    - MAS-CommonKADS [Iglesias et al. 97]: define modelos de análisis y diseño de MAS extendiendo la metodología CommonKADS (para diseño de sistemas expertos)
    - KAOS [Bertrand et al. 98]: Análisis de requisitos basado en una estrategia orientada a objetivos

## Sistemas basados en agentes

---

- Consideraremos agentes
  - Con capacidad de *procesar conocimiento*
  - Que persiguen el cumplimiento de unos *objetivos*
  - Capaces de percibir *eventos* del entorno
  - Las interacciones entre los agentes se describen como *acciones comunicativas*
  - Capaces de influir en el entorno mediante acciones directas
- No vamos a tratar aquí aspectos tales como
  - Movilidad
  - Seguridad
  - Fiabilidad
    - aunque son altamente relevantes, especialmente en la implantación
    - y están bastante ligados a la plataforma

## Sistemas basados en agentes

---

- Cuándo utilizar agentes
  - Situaciones con tipos de comunicaciones complejos y diversos
  - Sistemas en los que no es práctico o posible especificar el comportamiento caso por caso
    - El comportamiento de los agentes está guiado por sus objetivos, y no se especifica como "entrada *produce salida*"
  - Sistemas de negociación, cooperación y competencia entre distintas entidades
  - Sistemas autónomos
  - Los sistemas de agentes son muy modulares
    - Distribución
    - Extensibilidad
    - Flexibilidad

## Aspectos del SMA

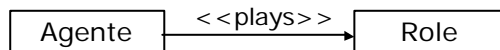
---

- Modelo de agente
  - Los agentes realizan tareas o persiguen objetivos
  - Responsabilidades, control y estado mental del agente
- Modelo de objetivos y tareas
  - Identificación de objetivos generales y descomposición en objetivos más concretos que se pueden asignar a agentes
  - Similarmente con tareas
  - Objetivos: motivación  $\bar{U}$  Tareas: actividad
- Modelo de interacción
  - Qué interacciones existen entre agentes/roles
- Modelo de organización
  - Estructura del SMA, roles, relaciones de poder, *workflows*
- Modelo de entorno
  - Entidades y relaciones con el entorno del SMA

## Modelo de agente

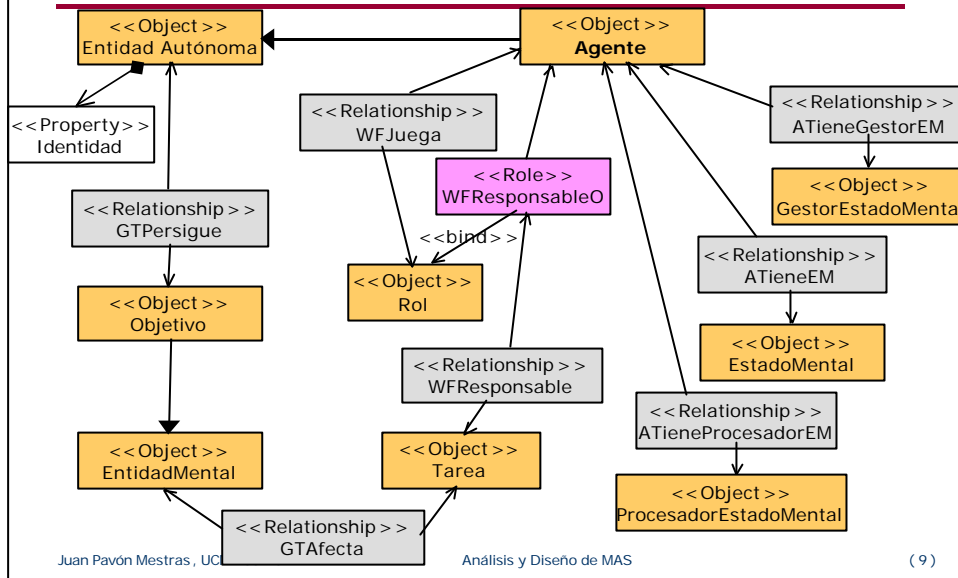
---

- Descripción de agentes particulares
  - Funcionalidad del agente: Responsabilidades
    - Qué tareas sabe ejecutar
    - Qué objetivos se compromete a alcanzar

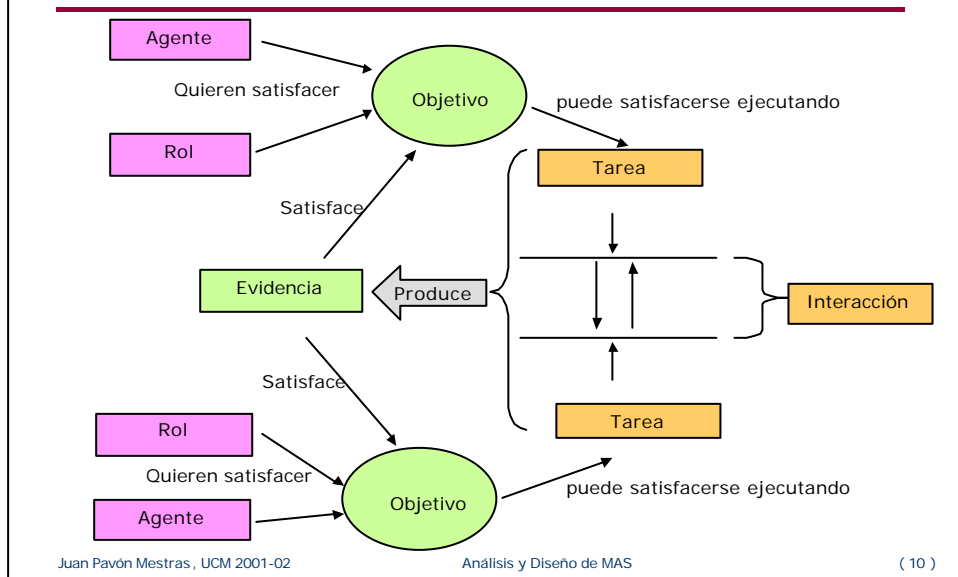


- Comportamiento: Control del agente
  - Estado mental
    - Agregación de entidades mentales: objetivos, creencias, compromisos, hechos
  - Gestión de estado mental
    - Creación, destrucción, modificación de las entidades del estado mental
  - Mecanismo de decisión: procesador de estado mental
    - Reglas, planificación, etc.

## Meta-modelo de agente

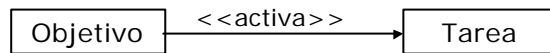


## Control del agente

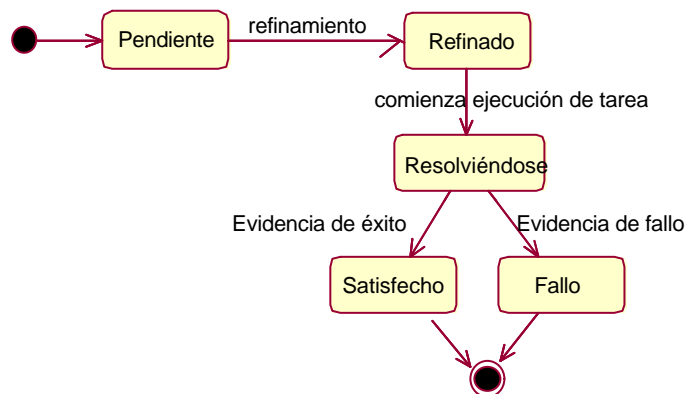


## Modelo de objetivos y tareas

- **Objetivo:** Situación deseada
  - Conjunto de estados que el agente quiere lograr, mantener, o evitar
  - Una función de utilidad que maximizar
  - Responde a *¿por qué?*
- **Tarea:** Transiciones de estado
  - Conduce a la consecución de objetivos
  - Responde a *¿cómo?*

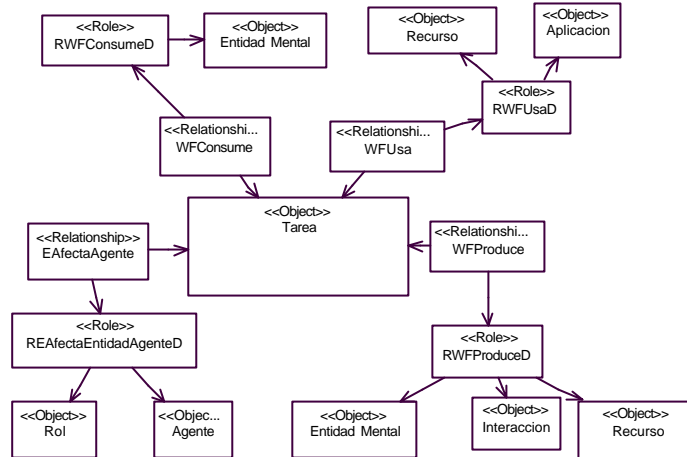


## Ciclo de vida de un Objetivo





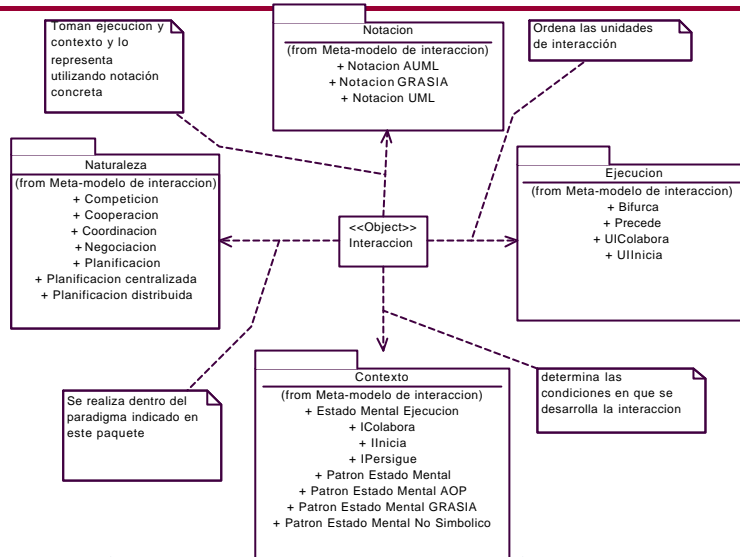
## Meta-modelo de objetivos y tareas: Descripción de tareas



## Modelo de interacción

- Intercambio de conocimiento o peticiones (intencionalidad) entre agentes
- Define las interacciones entre los agentes o entre agentes y humanos
  - Se definen a alto nivel, en diseño se detalla el protocolo de interacción
  - Se puede usar el concepto de protocolo de interacción de Agent UML o los protocolos de Gaia

## Aspectos de modelado de interacciones

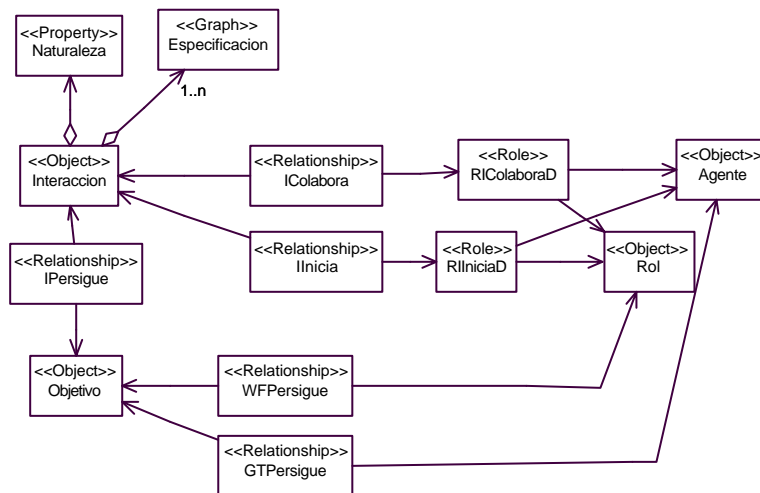


Juan Pavón Mestras, UCM 2001-02

Análisis y Diseño de MAS

( 17 )

## Meta-modelo de interacción

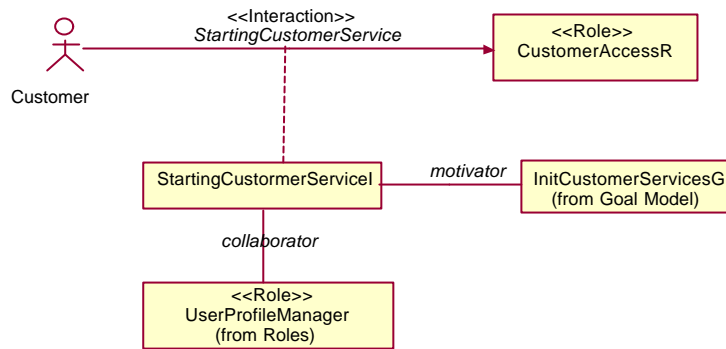


Juan Pavón Mestras, UCM 2001-02

Análisis y Diseño de MAS

( 18 )

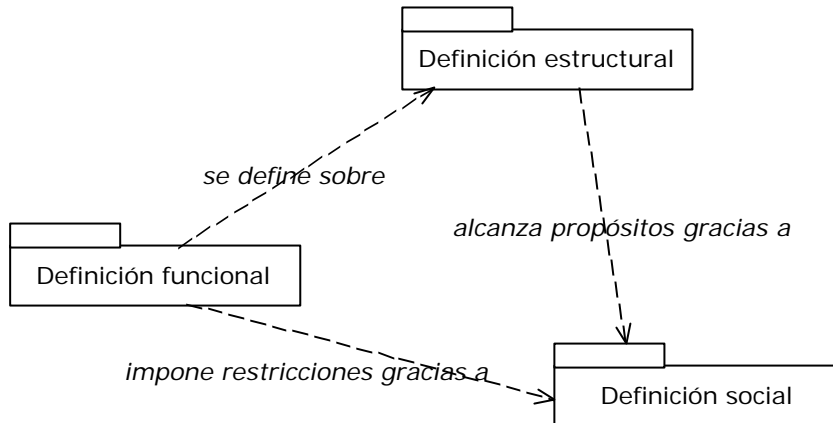
## Modelo de interacción



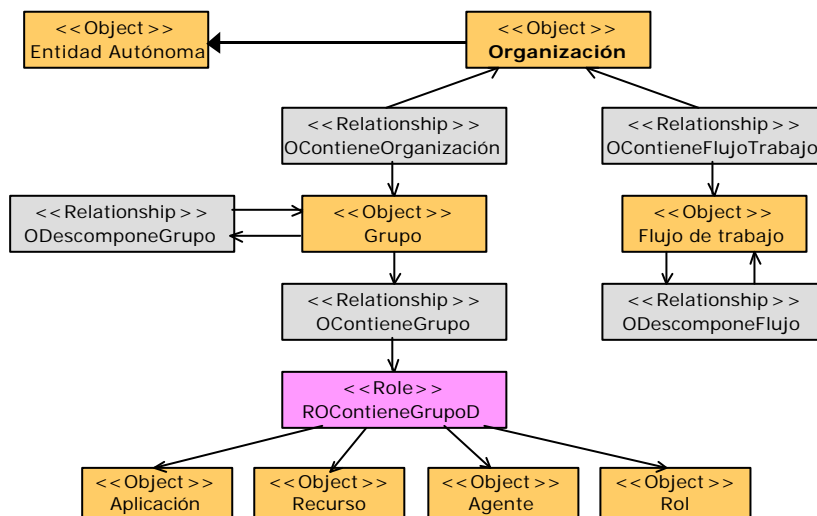
## Modelo de organización

- La organización define una estructura donde van a existir los agentes, recursos, tareas y objetivos
  - Estructura. Descomposición de la organización en:
    - Grupos
    - Flujos de trabajo
      - Interrelación de tareas en flujos de trabajo
      - Relaciones entre agentes respecto a las tareas
      - Recursos disponibles y asignación
  - Relaciones sociales
    - Relaciones de poder (p.ej. subordinación) y cliente/servidor entre agentes
    - Relaciones entre grupos
    - Relaciones entre organizaciones
  - Funcionalidad
    - Propósito
    - Tareas que debe realizar

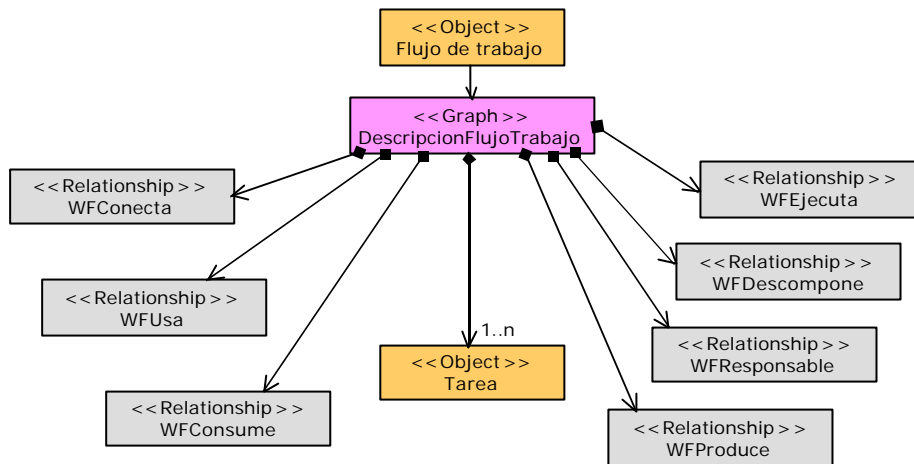
## Meta-modelo de organización



## Meta-modelo de organización: Descripción estructural



## Meta-modelo de organización: Descripción funcional



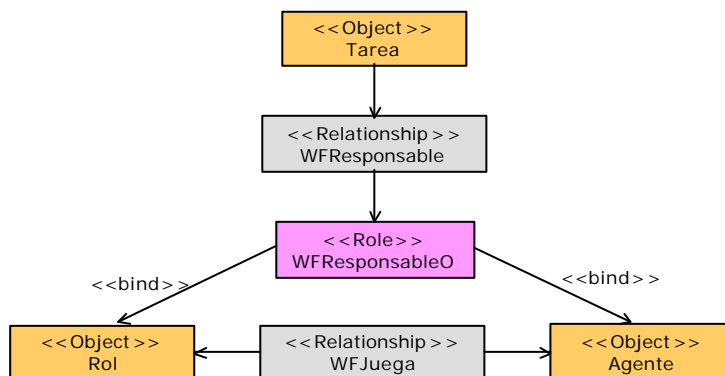
Juan Pavón Mestras, UCM 2001-02

Análisis y Diseño de MAS

( 23 )

## Meta-modelo de organización: Descripción funcional

### Asignación de tareas

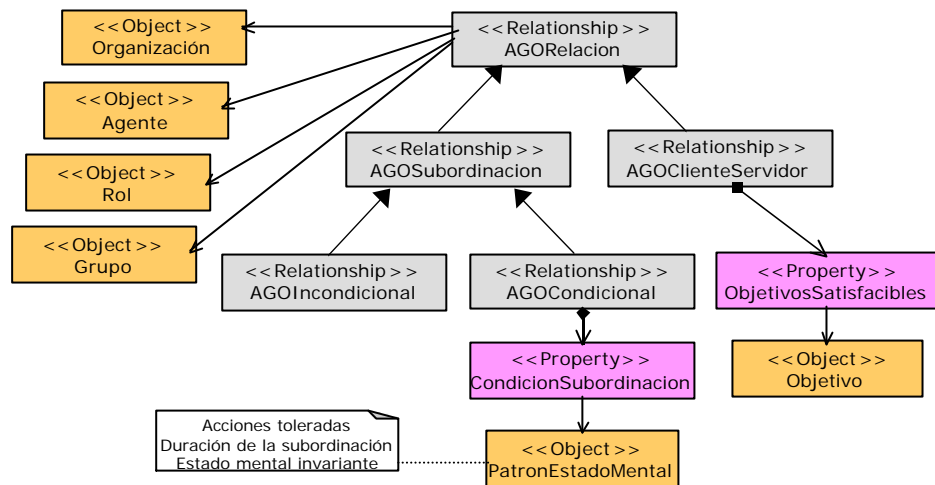


Juan Pavón Mestras, UCM 2001-02

Análisis y Diseño de MAS

( 24 )

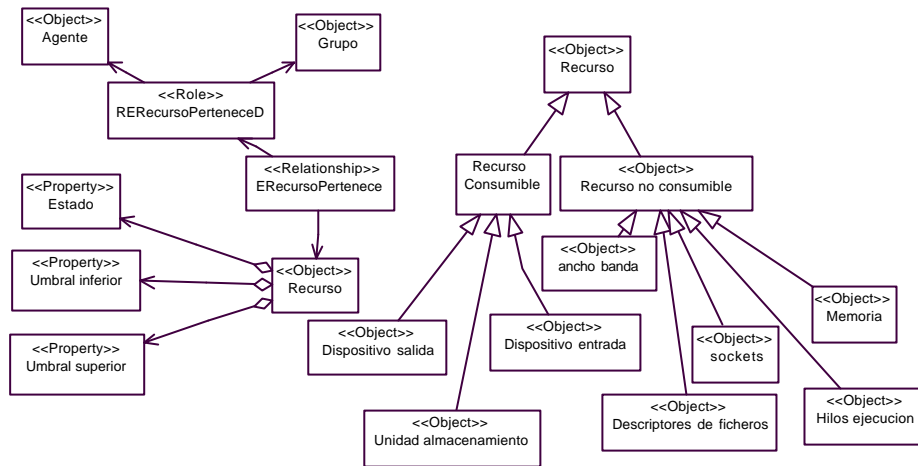
## Meta-modelo de organización: Descripción social



## Modelo de entorno

- Define las entidades del entorno del SMA con las que vaya a interactuar
  - Recursos
    - Elementos consumibles o no consumibles:
      - Descriptores de ficheros, hilos de ejecución, memoria, dispositivos de E/S, sockets, ancho de banda, etc.
  - Aplicaciones
    - Uso más complejo, por medio de alguna interfaz
  - Agentes
    - Satisface el principio de racionalidad

## Meta-modelo de entorno: Recursos

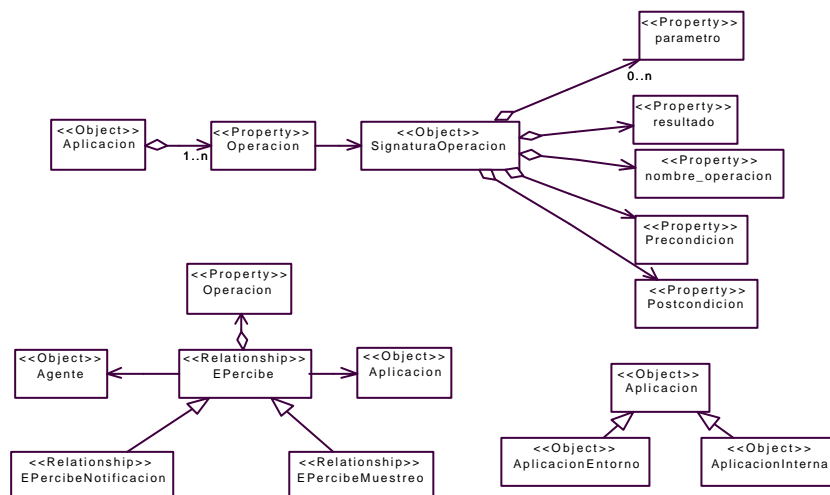


Juan Pavón Mestras, UCM 2001-02

Análisis y Diseño de MAS

( 27 )

## Meta-modelo de entorno: Aplicaciones y percepción



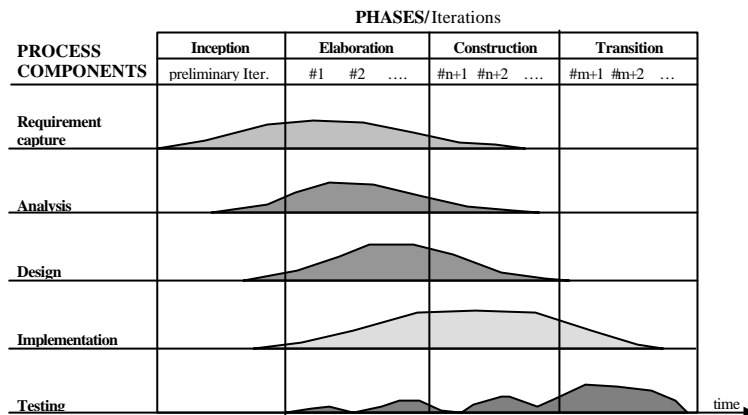
Juan Pavón Mestras, UCM 2001-02

Análisis y Diseño de MAS

( 28 )

## Ciclo de vida de desarrollo de un MAS

- Se puede asemejar al Proceso Unificado de Rational: **iterativo e incremental**, y **dirigido por riesgos** (casos)



Juan Pavón Mestras, UCM 2001-02

Análisis y Diseño de MAS

( 29 )

## Análisis del MAS

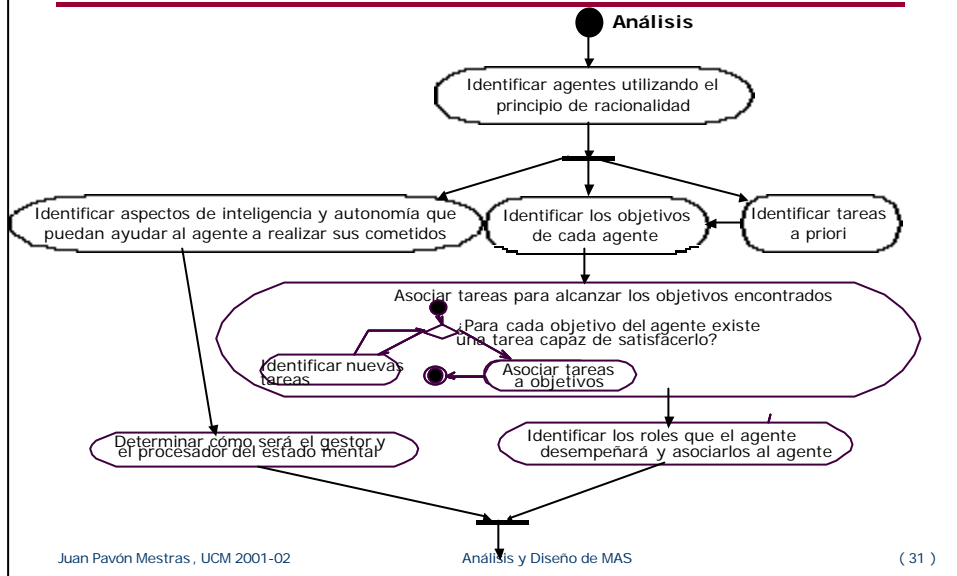
- Propósito: obtener un modelo abstracto que represente los requisitos del sistema para
  - entender mejor el problema
  - confirmar que se trata del problema a resolver (validación)
  - facilitar el diseño de la solución
- Define *qué* hace el sistema
  - Requisitos funcionales
    - Qué servicios proporciona el sistema
  - Requisitos no funcionales
    - Parámetros de calidad de servicio

Juan Pavón Mestras, UCM 2001-02

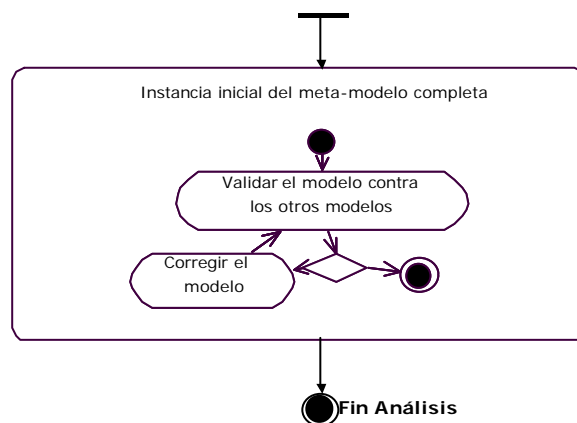
Análisis y Diseño de MAS

( 30 )

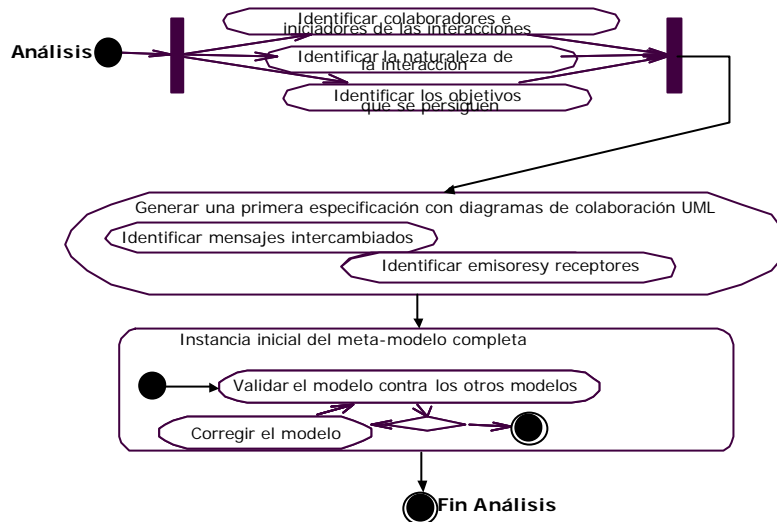
## Generación de modelo de agente



## Generación de modelo de agente



## Generación del modelo de interacción

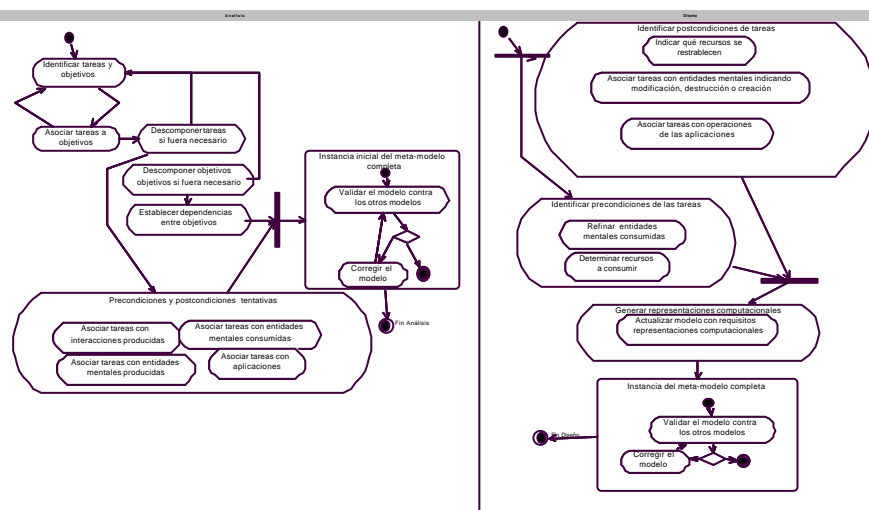


Juan Pavón Mestras, UCM 2001-02

Análisis y Diseño de MAS

( 33 )

## Generación del modelo de tareas y objetivos

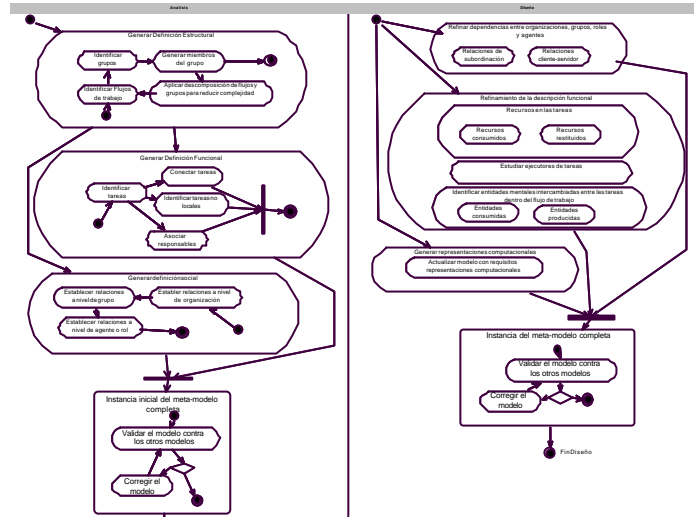


Juan Pavón Mestras, UCM 2001-02

Análisis y Diseño de MAS

( 34 )

# Generación del modelo de organización

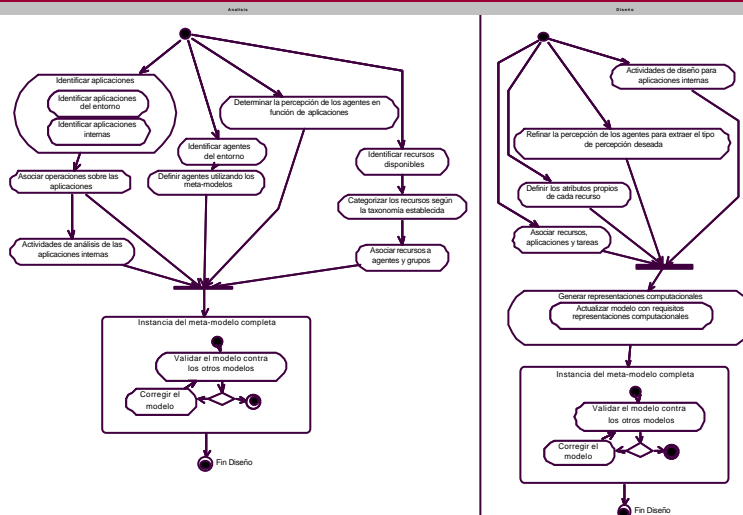


Juan Pavón Mestras, UCM 2001-02

Análisis y Diseño de MAS

( 35 )

# Generación del modelo de entorno



Juan Pavón Mestras, UCM 2001-02

Análisis y Diseño de MAS

( 36 )

## Diseño del MAS

---

- Propósito: Transformación de las entidades de análisis en entidades computacionales
  - Se asume una plataforma de agentes que establece una arquitectura y restricciones en las alternativas de diseño
    - Por ejemplo, BDI, capas, FIPA, etc.
- Define *cómo* se realizan los requisitos del sistema
  - Definición de la organización de los agentes del sistema
  - Especificación de las interfaces de los agentes y componentes del sistema
  - Estructura interna de los agentes y su comportamiento

## Diseño del agente

---

- 4 niveles:
  - Control y gestión
    - Sincronización y coordinación de las actividades de los otros componentes
  - Percepción y comunicación
    - Gestión de las sesiones de comunicación del agente con
      - Otros agentes
      - Usuarios
      - Sistemas propietarios (legacy systems)
  - Dominio
    - Recursos y elementos de información propios del dominio de la aplicación
  - Recursos
    - Recursos genéricos de la plataforma de agentes

## Bibliografía

---

- [Odell et al. 00] *Extending UML for Agents*, James Odell, H. Van Dyke Parunak, Bernhard Bauer, submitted paper, 2000. <http://www.jamesodell.com/ExtendingUML.pdf>
- [UML] OMG Unified Modeling Language Specification Version 1.3, June 1999, Object Management Group, Inc. <http://www.rational.com/uml/resources/documentation/index.jttml>
- [MOF] OMG Meta Object Facility (MOF) Specification: <ftp://ftp.omg.org/pub/docs/ad/99-09-04.pdf>
- [Wooldrige et al. ] *The Gaia Methodology for Agent-Oriented Analysis and Design*, Wooldridge, M., Jennings, N. R., Kinny, D., Kluwer Academic Press
- [Shoham] Shoham, *Agent Oriented Programming*
- [Iglesias et al. 97] *Analysis and Design of Multiagent Systems using MAS-CommonKADS*, Carlos Iglesias, Mercedes Garrijo, Jose Gonzalez and Juan R. Velasco, *Intelligent Agents IV: Agent Theories, Architectures and Languages*, 1997, M. P. Singh, Anand Rao and M. J. Wooldridge, eds. *Lecture Notes in Computer Science* 1365
- [RUP] *The Rational Unified Process: An introduction*, Philippe Kruchten, pub. Addison Wesley.
- [Bertrand et al. 98] P. Bertrand, R. Darimont, E. Delor, P. Massonet, A. van Lamsweerde *GRAIL/KAOS: an environment for goal driven requirements engineering* Proceedings ICSE'98 - 20th International Conference on Software Engineering, IEEE-ACM, Kyoto, April 98
- [Message] *Methodology for Engineering Systems of Software Agents, Deliverable 1, Initial Methodology*, Julio 2000 <http://www.eurescom.de/~public-webspace/P900-series/P907/index.htm>