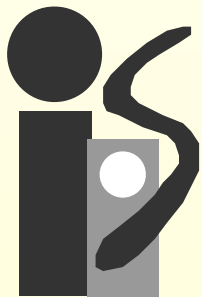
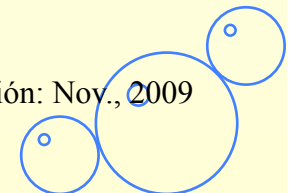
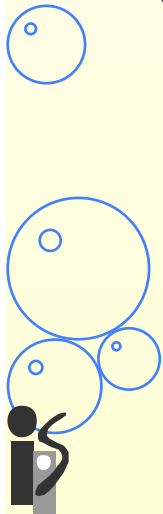


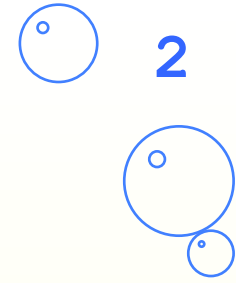
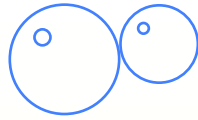
Comunicación entre agentes



J.A. Bañares Bañares

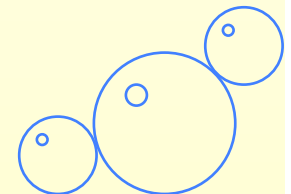
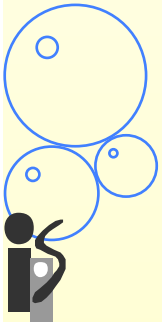
Departamento de Informática e Ingeniería de Sistemas
C.P.S. Universidad de Zaragoza

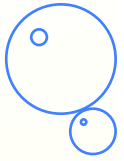
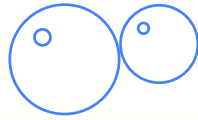




- **Esta presentación ha sido elaborada a partir del material docente del Curso de doctorado**

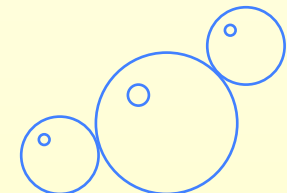
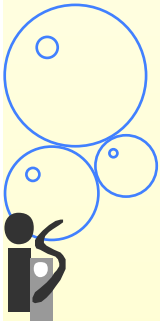
- Agentes Inteligentes de la Universidad Complutense de Madrid del profesor Juan Pavón Mestras
 - <http://www.fdi.ucm.es/profesor/jpavon/doctorado/>

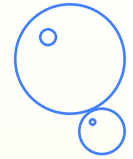
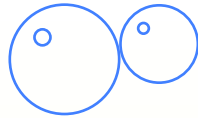




Conversaciones entre agentes

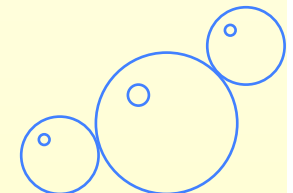
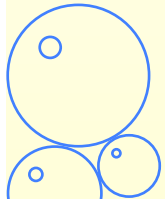
- **Coordinación**
 - Modelo de comunicación (ejemplo: Linda: modelo de comunicación ortogonal a los modelos de computación)
- **Lenguaje de comunicación**
 - Contenido del mensaje
- **Conversación**
 - Intercambio de mensajes
- **Protocolo**
 - Conversación válida
 - Se definen protocolos de negociación, subasta, ... que intercambian mensajes en un lenguaje común.

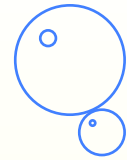
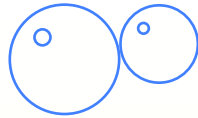




Comunicación entre agentes

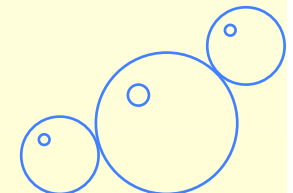
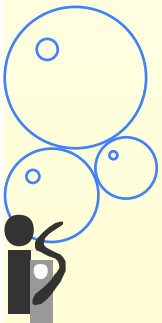
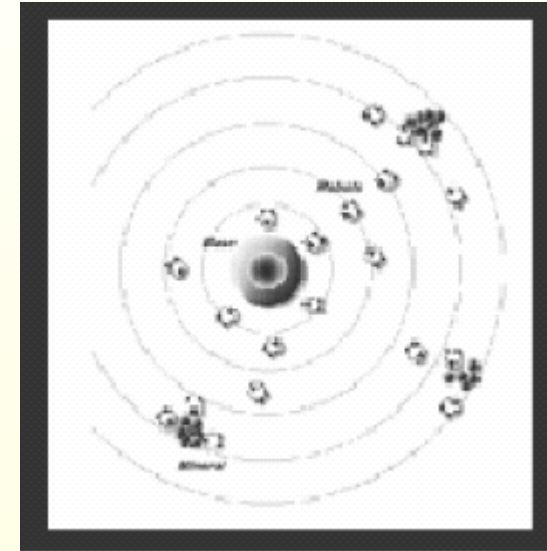
- La comunicación es la base para las interacciones y la organización social de los agentes
- Interacciones
 - Hay interacciones cuando la dinámica de un agente está perturbada por las influencias de otros [O. Boissier, 2001]
 - **Las interacciones son el motor de los SMA**
- Distintas **formas de interaccionar**
 - Acciones sobre el entorno
 - Pizarra compartida
 - Inferencias
 - Paso de mensajes
- ...

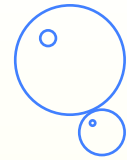
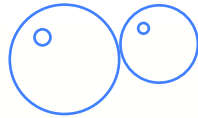




Comunicación mediante el entorno

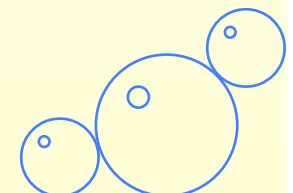
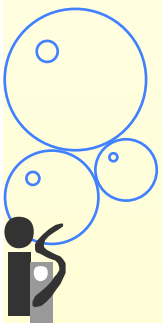
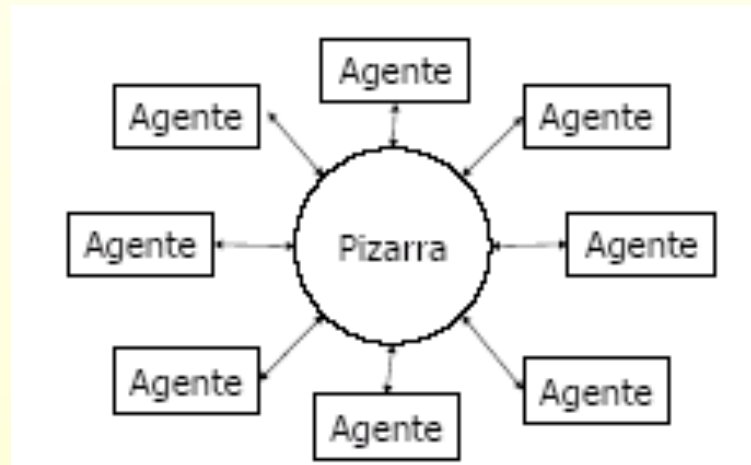
- Ejemplo: robots distribuidos [Steels 89]
Problema
Un conjunto de robots tienen que recoger piedras preciosas (cuya localización no se conoce de antemano) y llevarlas a una nave nodriza
- La comunicación se realiza a través del entorno:
 - Campo gradiente de la señal generada por la nave nodriza
 - Partículas radioemisoras que pueden recoger, echar y detectar los robots al pasar

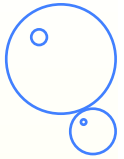
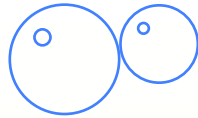




Sistema de pizarra

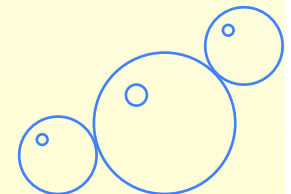
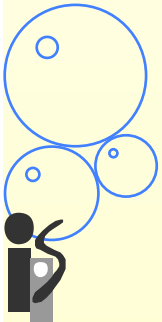
- Pizarra: Memoria compartida que permite a los agentes compartir todo tipo de información
- Un sistema multiagente puede tener varias pizarras con distintos agentes registrados en cada una
- **No hay comunicación directa entre agentes**

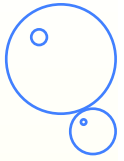
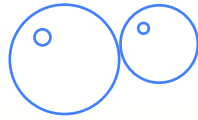




Sistema de pizarra

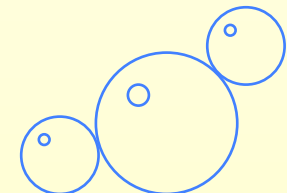
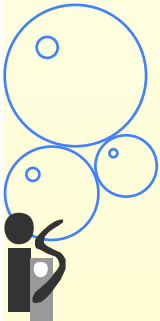
- Los sistemas más avanzados incorporan nuevos conceptos:
 - **Moderador.** Agente especializado con conocimiento de control y de evaluación que publica en la pizarra los subproblemas a resolver y decide a qué agentes se asignan de entre aquellos que se han ofrecido a resolverlos
 - **Despachador.** Agente que avisa a los agentes afectados por algún cambio producido en la pizarra

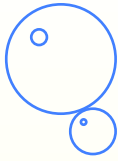
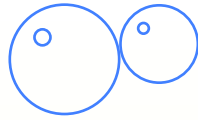




Sistema de pizarra

- **Pros:** *Método flexible de comunicación* para la resolución distribuida de problemas
 - **Son independientes de la estrategia de cooperación que se vaya a utilizar y no afectan a la arquitectura de los agentes individuales**
- **Contras:** la estructura central de la pizarra representa un cuello de botella, ya que todos los agentes distribuidos por una red se ven obligados a acceder al dispositivo central donde se encuentra la pizarra
 - Space based solutions => Pizarras/Espacios de tuplas distribuidos





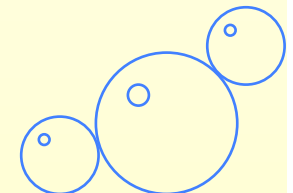
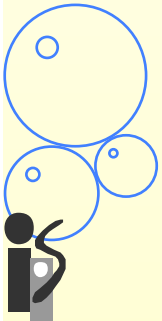
Interacción **sin comunicación!!!!**

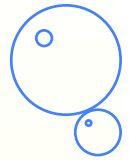
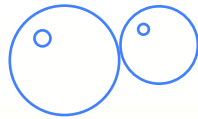
- **Inferencia de acciones** de otros agentes
 - Ejemplo: utilización de **la teoría de juegos con matrices de ganancia**

Dilema del prisionero

Dos hombres son acusados de un crimen y encerrados en celdas separadas. No pueden comunicarse ni llegar a acuerdos. A los dos hombres se les dice

- Si uno de los dos confiesa el crimen y el otro no, el confesor será liberado, y el otro condenado a tres años.
- Si los dos confiesan, ambos serán encarcelados durante dos años.
- Si ninguno confiesa, ambos serán encarcelados durante un año.





Interacción sin comunicación

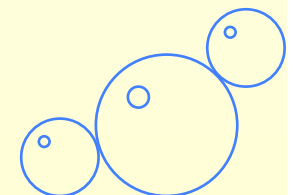
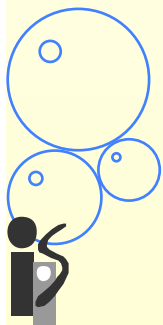
- **¿cuál es la mejor estrategia?**
 - Teorías de juegos....

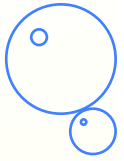
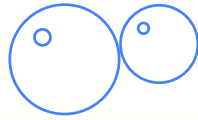
Matriz de ganancia

		<i>i</i>	
		Rechazo	coop
<i>j</i>	Rechazo	2 2	0 5
	coop	0 5	3 3

Consideramos **no confesar** como cooperar, y **confesar** como rechazar la cooperacion.

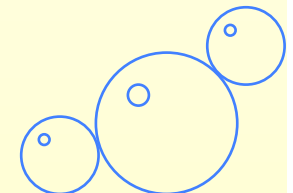
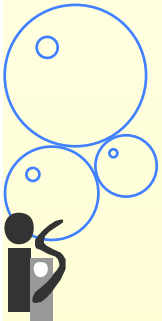
Los números no indican los años sino lo bueno que es para los agentes

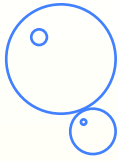
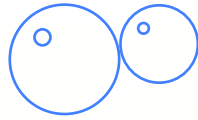




Comunicación a nivel del conocimiento

- Los agentes se comunican para
 - Mostrar a otros agentes su estado mental
 - Intentar modificar el estado mental de otros agentes



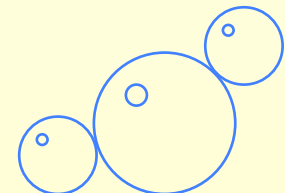
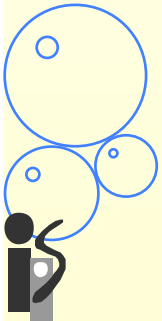


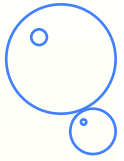
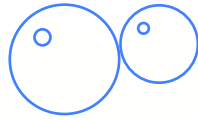
Actos del habla (Speech acts)

- Language as Action
 - **J.L. Austin (1962), How to do things with words, Clarendon Press**

La lingüística tradicional intentaba entender el significado de las frases indicando cómo es posible usar una combinación de palabras para hacer una declaración con significado

- ***Los actos del habla* hacen referencia a las acciones *intencionales* en el curso de una conversación**



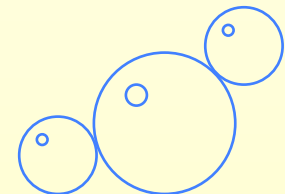
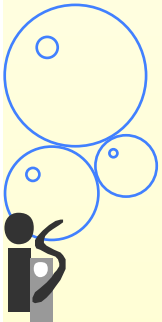


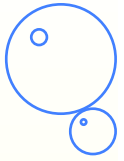
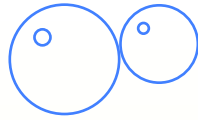
Actos del habla (Speech acts)

Language as Action

– **J.L. Austin (1962), How to do things with words, Clarendon Press**

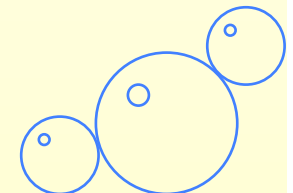
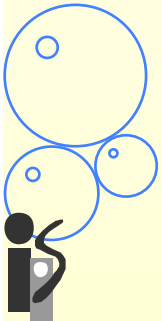
- Quien habla no declara solamente sentencias ciertas o falsas
- Quien habla realiza actos de habla:
peticiones, sugerencias, promesas, amenazas, etc.
- Cada declaración es un acto de habla

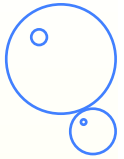
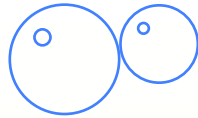




Tipos de actos del habla

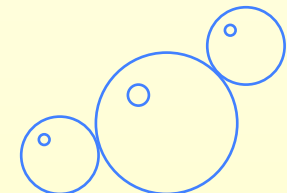
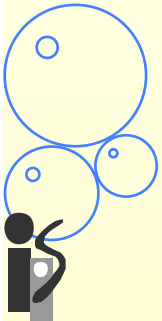
- J.R. Searle (1969), *Speech Acts*, Cambridge University Press
- **Actos asertivos**: dan información sobre el mundo
 - **Estoy de acuerdo** **2 y 2 son 4** **Estamos en clase**
- **Actos directivos**: para solicitar algo al destinatario
 - **Siéntate** **¿Cuántas pesetas son un euro?**
- **Actos de promesa**: comprometen al locutor a realizar ciertas acciones en el futuro
 - **Mañana vuelvo a las 8** **Te enviaré las fotos**
- **Actos expresivos**: dan indicaciones del estado mental del locutor
 - **Soy feliz** **Gracias** **Siento lo de tu perro**
- **Actos declarativos**: el mero hecho de la declaración es la realización de un acto
 - **Estás contratado** **Empezamos la clase**

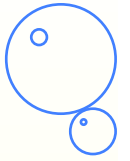
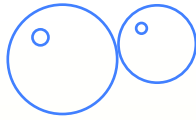




Componentes de los actos del habla

- **Locución**: modo de producción de frases utilizando una gramática y un léxico
- **Ilocución**: acto realizado por el locutor sobre el destinatario mediante la declaración (utterance)
 - **Fuerza ilocutoria (F): afirmación, pregunta, petición, promesa, orden => PERFORMATIVA**
 - **Contenido proposicional (P), objeto de la fuerza ilocutoria**
 - **Se puede representar como F(P) performativa (contenido)**
aserta(está nevando) responde(está nevando)
- **Perlocución**: efectos que pueden tener los actos ilocutorios en el estado del destinatario y en sus acciones, creencias y juicios
Ejemplos: convencer, inspirar, persuadir, atemorizar



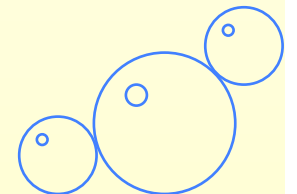
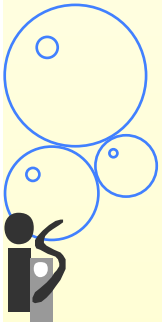


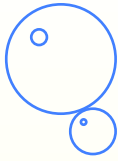
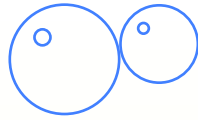
Componentes de los actos del habla

- Ejemplo:

Cierra la puerta

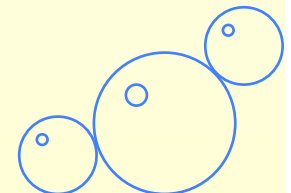
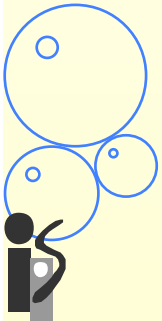
- locución: declaración física con contexto y referencia: quién habla y quién escucha, qué puerta, etc.
- ilocución: acto de llevar intenciones: el que habla quiere que el que escucha cierre la puerta
- perlocución: acciones que ocurren como resultado de la ilocución: el que escucha cierra la puerta





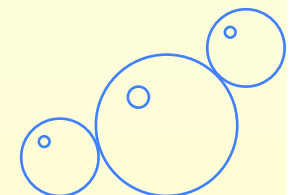
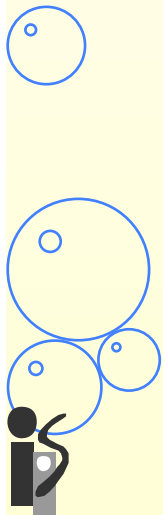
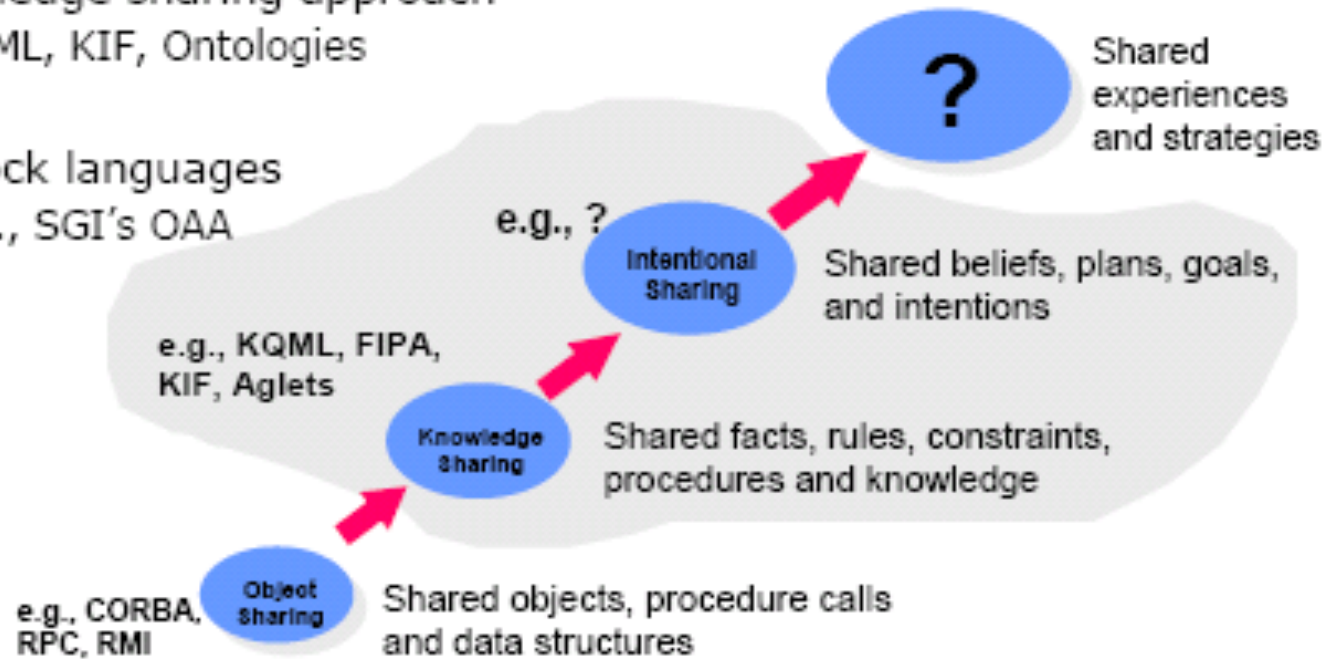
Éxitos de los actos del habla

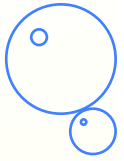
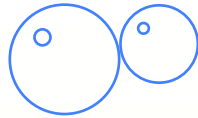
- Una declaración no es verdadera o falsa: tiene éxito o fracasa
- Un acto de habla puede fallar
 - en su enunciación, porque no llegue el mensaje o llegue corrompido o el destinatario no lo entiende
 - en su interpretación, por el destinatario
 - en su consecución final, por ejemplo porque el destinatario no sea capaz de llevar a cabo la acción solicitada



Lenguajes de comunicación de agentes (Y. Labrou 2000)

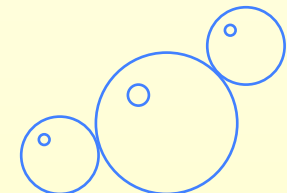
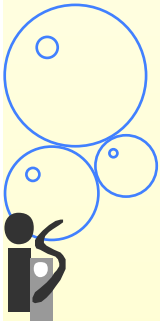
- Es CORBA un ACL?
- Knowledge sharing approach
 - KQML, KIF, Ontologies
- FIPA
- Ad hock languages
 - e.g., SGI's OAA





KSE (Knowledge Sharing Effort)

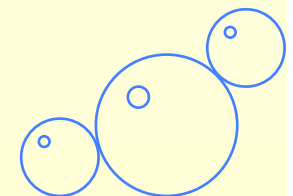
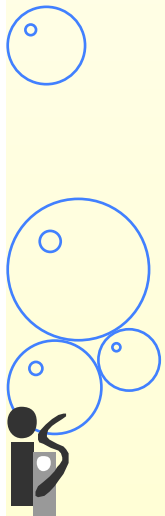
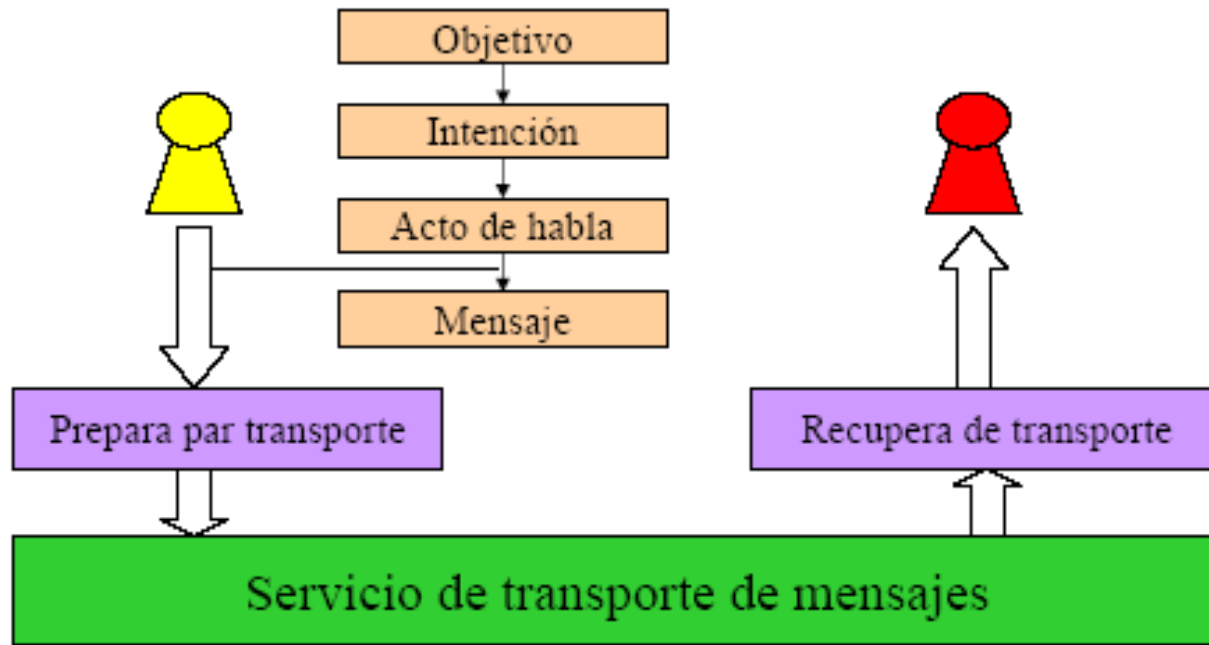
- Iniciado por ARPA hacia 1990, y apoyado por organismos norteamericanos de investigación (ASOFR, NSF, NRI)
- Propósito:
 - Desarrollo de técnicas, metodologías y herramientas software para la compartición y reutilización del conocimiento entre sistemas
- **El KSE produjo dos especificaciones**
 - **KQML** (knowledge Query Manipulation Language): Define la envoltura de un mensaje. El agente muestra explícitamente su intención (**ilocución**).
 - **KIF** (Knowledge Interchange Format): Representación del conocimiento. Contenido del mensaje.

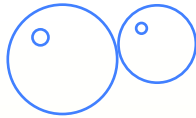


- ***KQML (Lenguaje de comunicación y consultas de conocimiento)***

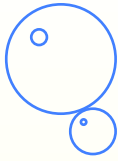
- *Lenguaje de comunicación y protocolo de alto nivel orientado a mensajes.*
- *Es independiente del protocolo de transporte, lenguajes de contenido y de la ontología.*
- *Los mensajes KQML comunican una actitud sobre el contenido que llevan (aserto, solicitud, pregunta)*
 - Las primitivas del lenguaje se llaman **performativas**
 - Cada mensaje KQML representa un acto de habla

KQML

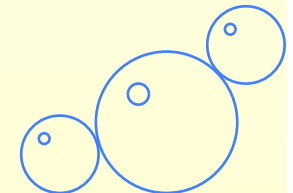
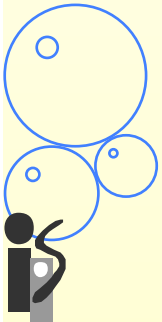




KQML



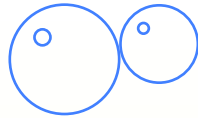
- Nivel de contenidos
 - **Lleva el contenido** del mensaje en el lenguaje de representación propio de los agentes
 - No es procesado por las implementaciones de KQML (se deja al agente)
- Nivel de mensajes
 - **Núcleo del lenguaje KQML**, determina los tipos de interacción que pueden tenerse con un agente que hable KQML
 - Identifica el **protocolo** que se utiliza para enviar un mensaje y proporciona una **performativa** al emisor que añade el contenido
 - **Indica** también el lenguaje de contenidos, **ontología** asumida, y algún descriptor sobre el contenido (esto permite a algunas aplicaciones procesar los mensajes sin acceder al contenido)
- Nivel de comunicaciones
 - Trata los parámetros de comunicación de más bajo nivel: identidad de emisor y receptor, identificación de la comunicación



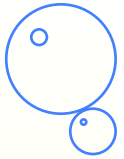
KQML

- Mensajes KQML
 - **Representa un acto de habla o performativa**
 - **Consta de una lista de pares atributo-valor**





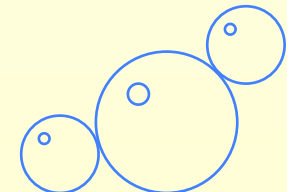
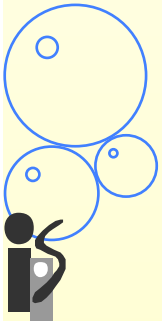
KQML

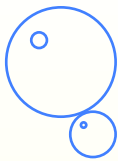
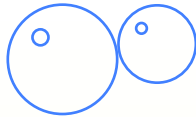


- Mensajes KQML

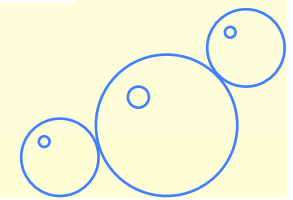
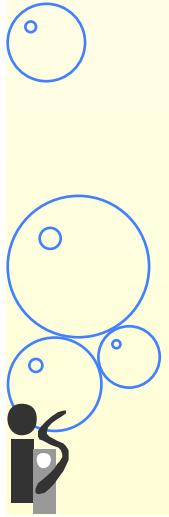
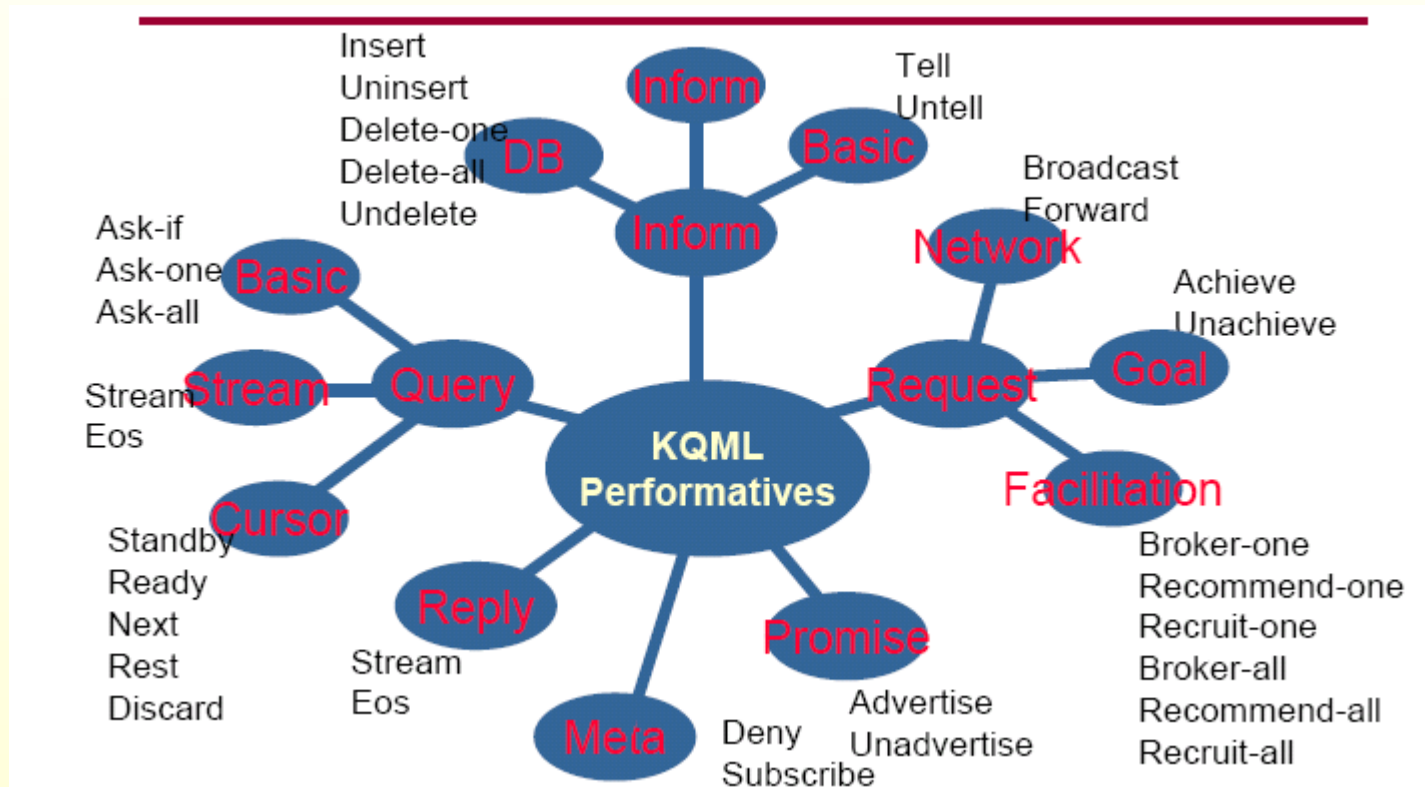
La respuesta al anterior:

```
(tell
  :sender servidor-bolsa
  :content (PRECIO TELEFONICA 19)
  :receiver pepe
  :in-reply-to accion-telefonica
  :language LPROLOG
  :ontology IBEX
)
```

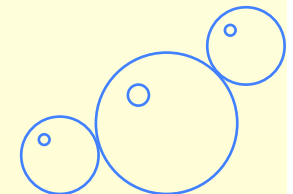
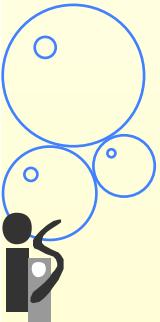


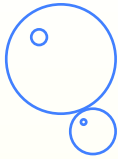
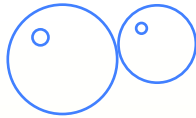


Performativas de KQML



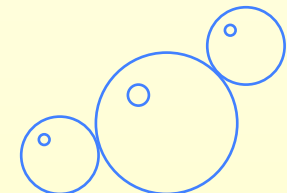
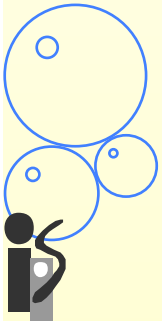
- ***KIF (Lenguaje para el intercambio de conocimiento)
Lenguaje de contenido***
 - *Crea ontologías*
 - *Diccionarios abiertos*
 - *Permite comunicar el conocimiento entre agentes*
 - *Estándar de los mensajes KQML.*

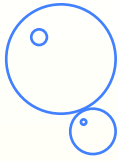
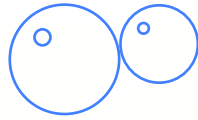




KIF

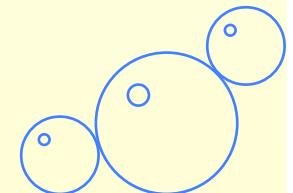
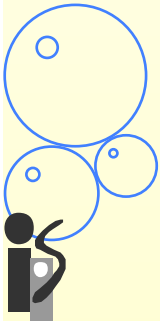
- Ejemplos de expresiones
 - **Datos**
 - (empleado pepe ventas)
 - tupla que indica que pepe es un empleado en el departamento de ventas
 - **Operaciones lógicas**
 - (> (* (ancho chip1) (largo chip1)) (* (ancho chip2) (largo chip2)))
 - establece que un chip es de mayor tamaño que el otro
 - **Representación conocimiento**
 - (interesado pepe '(empleado ,?x ,?y))
 - el agente pepe está interesado en recibir tuplas de la relación de empleados
 - **Procedimientos (con sintaxis parecida a LISP)**
 - (progn (frase-tipica f)
 - (print "Hola mundo.")
 - (frase-tipica f))

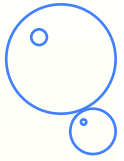
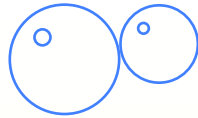




Ontologías

- Ontología: un vocabulario común en el que se han acordado significados para describir un dominio
 - **Es una conceptualización del mundo, en función de objetos, cualidades, distinciones, relaciones**
- Una ontología define un conjunto de clases, funciones y constantes para un dominio de discurso, e incluye una axiomatización para restringir su interpretación (reglas de inferencia)
- Ejemplos de ontologías:
 - **Cyc (<http://www.cyc.com/>), ontología de propósito general**
 - **WordNet, sistema de referencia léxica accesible por internet**
 - **CIA World Fact Book ([http://www.cia.gov/cia/publications Servicios Web /factbook/](http://www.cia.gov/cia/publications_Servicios_Web/factbook/)), 5 megas de KIF...**
 - **UMLS (Unified Medical Language System)**

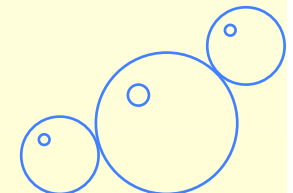
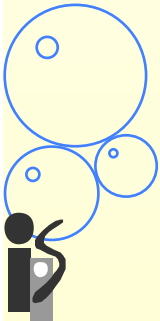


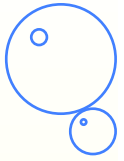
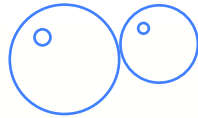


Representaciones del conocimiento

- **Esquemas de clasificación de conceptos**

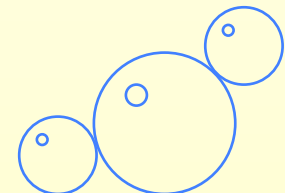
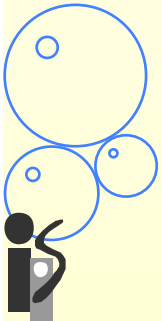
- **Keywords:** Son la orma más rápida y sencilla de localizar información potencialmente útil. Metadatos básicos.
- **Tesauros:** Ofrecen una aproximación estructurada al manejo de las palabras clave, estableciendo entre los conceptos las relaciones broader, narrower, and related.
- **Taxonomías:** Estructuras de clasificación que añaden la potencia de la herencia.
- **Ontologías:** Mayor riqueza en las relaciones.
 - La mayoría de ontologías incorporan las relaciones:
 - Generalización y Herencia
 - Agregación.
 - Instanciación

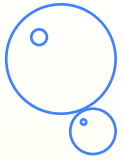
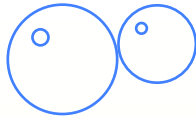




Ontololingua

- Ontolingua es un lenguaje permite construir, publicar y compartir ontologías
 - **Una interfaz a un servidor de edición/navegación**
 - **Las ontologías pueden traducirse automáticamente a distintos lenguajes de contenidos, como KIF, LOOM, Prolog, CLIPS, etc.**
- El lenguaje incluye primitivas para combinar ontologías
- Otras herramientas para trabajar con ontologías:
 - **Ontosaurus (<http://www.isi.edu/isd/ontosaurus.html>), navegador para Loom**
 - **WebOnto (<http://kmi.open.ac.uk/projects/webonto/>), Java applet**





Convergencia de intereses con los Servicios Web

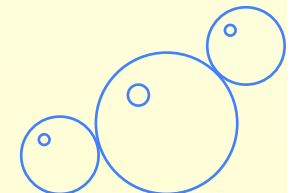
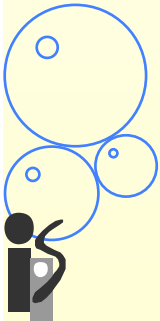
- Web Semántica
Weners-Lee, el padre del World-Wide Web (HTTP+HTML), ha propuesto su siguiente visión: **La Web Semántica**

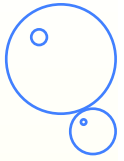
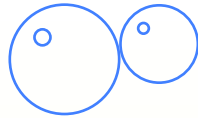
La Web :

- Compartir entre humanos conocimiento en forma de documentos hiperenlazados
- La interpretación es posible a que los humanos compartimos conocimientos.

La Web Semántica:

- Compartir entre aplicaciones conocimientos. Se trata de etiquetar las páginas no sólo con detalles de presentación (HTML), sino con detalles de significado.

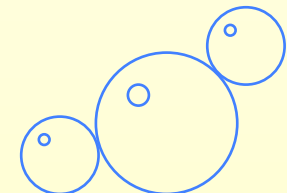
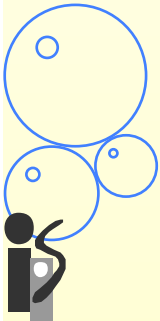


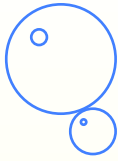
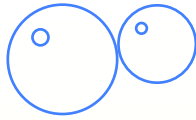


Frames vs Descriptions

- **Dos aproximaciones detrás de las ontologías:**

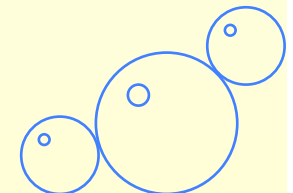
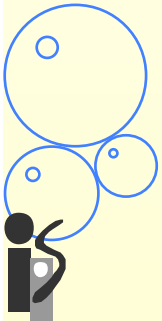
- **Frames:** Representación del conocimiento directamente en terminos de grafos. Se definen los frames (que representan clases o instancias) y se relacionan entre ellos. *Los frames se definen directamente en terminos de un conjunto de propiedades (slots) y propiedades (facets).*
- **Description Logics** es una familia de lenguajes que formalmente expresan ciertas restricciones sobre la representación del conocimiento. Tienen una *semántica precisa y se puede automatizar su procesamiento.* Description logics comienza con conceptos primitivos y sigue con la definición de conceptos en términos de descripciones formales. La **Subsumption**, esto es la jerarquía de especialización entre conceptos, se puede determinar de la descripción de los conceptos.
 - **Description logic determina si un concepto es redundante o está relacionado con otros conceptos.**

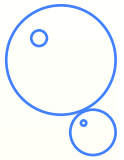
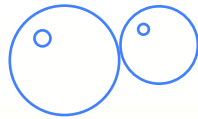




RDF (Resource Description Framework)

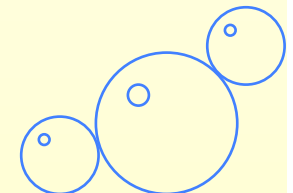
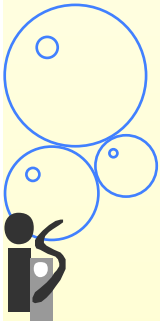
- **RDF es un lenguaje que permite en XML definir estructuras tipo grafo. De esta forma cualquier conocimiento estructurado se puede representar en RDF**
 - Un documento RDF es una colección de sentencias expresadas como tripletas. El conocimiento se expresa en forma de tripletas subject, predicate, object p.e . (estudiante, subclase, persona)





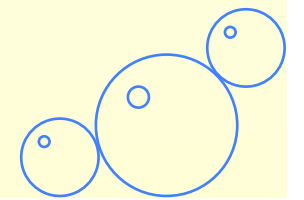
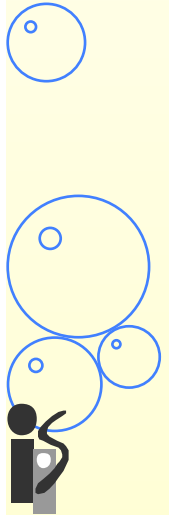
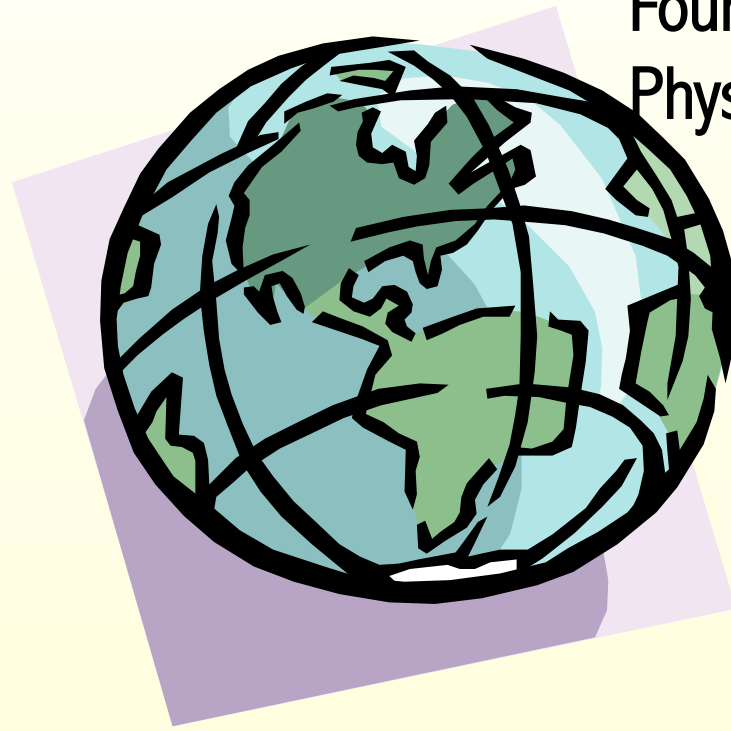
Web Ontology Language

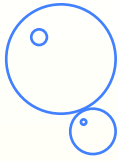
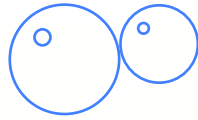
- **Web Ontology Language (OWL) permite especificar clases y propiedades en forma de sentencias description logic.**
- **OWL-S: Una ontología para la descripción de los servicios Web.**
- **SWRL: Semantic Web Rule Language. Extiende OWL con reglas (RuleML).**
- **Otras propuestas: .**



FIPA

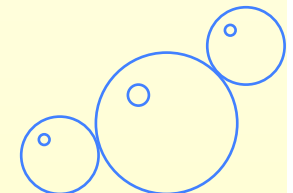
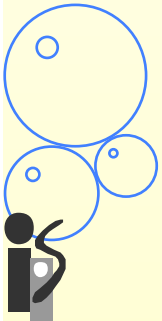
Foundation for Intelligent
Physical Agents (FIPA)

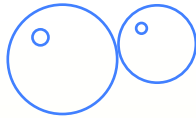




Arquitectura Abstracta FIPA

- **Foundation for Intelligent Physical Agents**
 - **Consortio industrial fundado en 1996**
 - **Varias decenas de compañías de telecomunicaciones e informática**
<http://www.fipa.org>
- **Objetivos**
 - **Acelerar el desarrollo de tecnologías de agentes inteligentes mediante la producción de especificaciones acordadas internacionalmente**
- **Especificación del comportamiento y capacidades externas de subsistemas genéricos: recursos de agentes (para migración, ejecución, etc.), interacción y cognitivos**
 - **las especificaciones estarán basadas en casos prácticos concretos**
 - **Agentes, multi-agentes, y sociedades de agente**
 - **Selección y adaptación de tecnologías existentes**





Arquitectura Abstracta FIPA

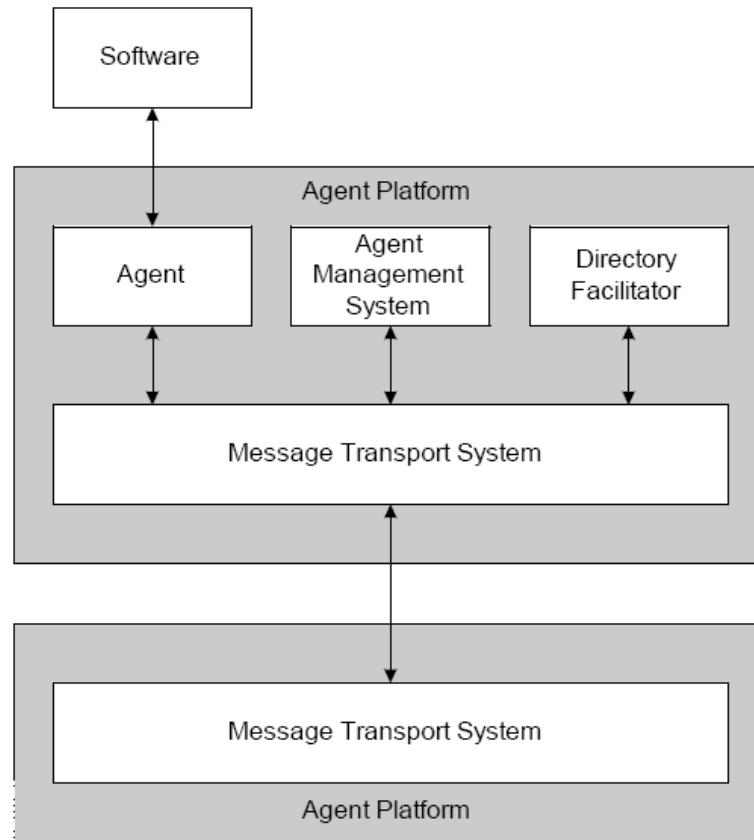
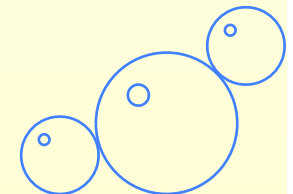
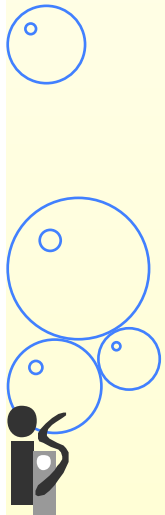
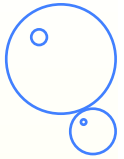
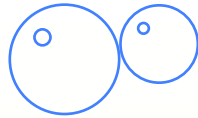


Figure 1: Agent Management Reference Model



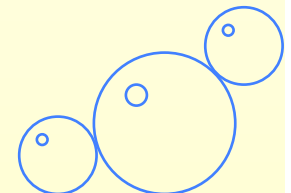
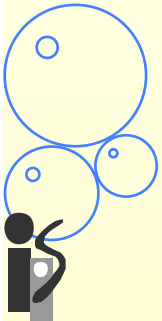
Arquitectura Abstracta FIPA

- Un **Agente** es un proceso computacional que implementa la funcionalidad de **comunicación** y **autonomía** de una aplicación. Los agentes comunican utilizando el "*Agent Communication Language*". El agente es el actor principal de una plataforma de agentes (AP).
- La Plataforma de agentes (AP) ofrece servicios de **publicación**, y **descripción de servicio**. Un agente debe tener al menos un propietario y una noción de identidad única. La identidad viene dada por el Agent Identifier (AID). El agente puede ser registrado con distintos protocolos (transport addresses) de transporte.
 - El **directorio** (Directory Facilitator, DF) es una componente opcional. Ofrece servicios de páginas amarillas en las que registrar o preguntar por servicios de otros agentes. Pueden existir varios DF en una plataforma y estar federadas.
- Un sistema de gestión de agentes (**Agent Management System, AMS**) es una componente obligatoria de la AP. Ejerce el control sobre el acceso y uso de la AP. Solo puede existir un AMS en la AP y mantiene un directorio de AIDs que contiene las direcciones de transporte (entre otras cosas) de los agentes registrados .

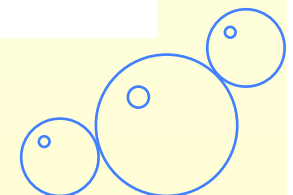
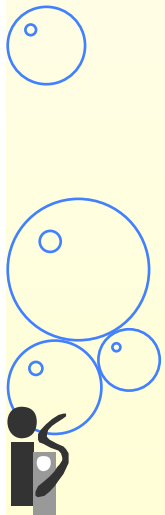
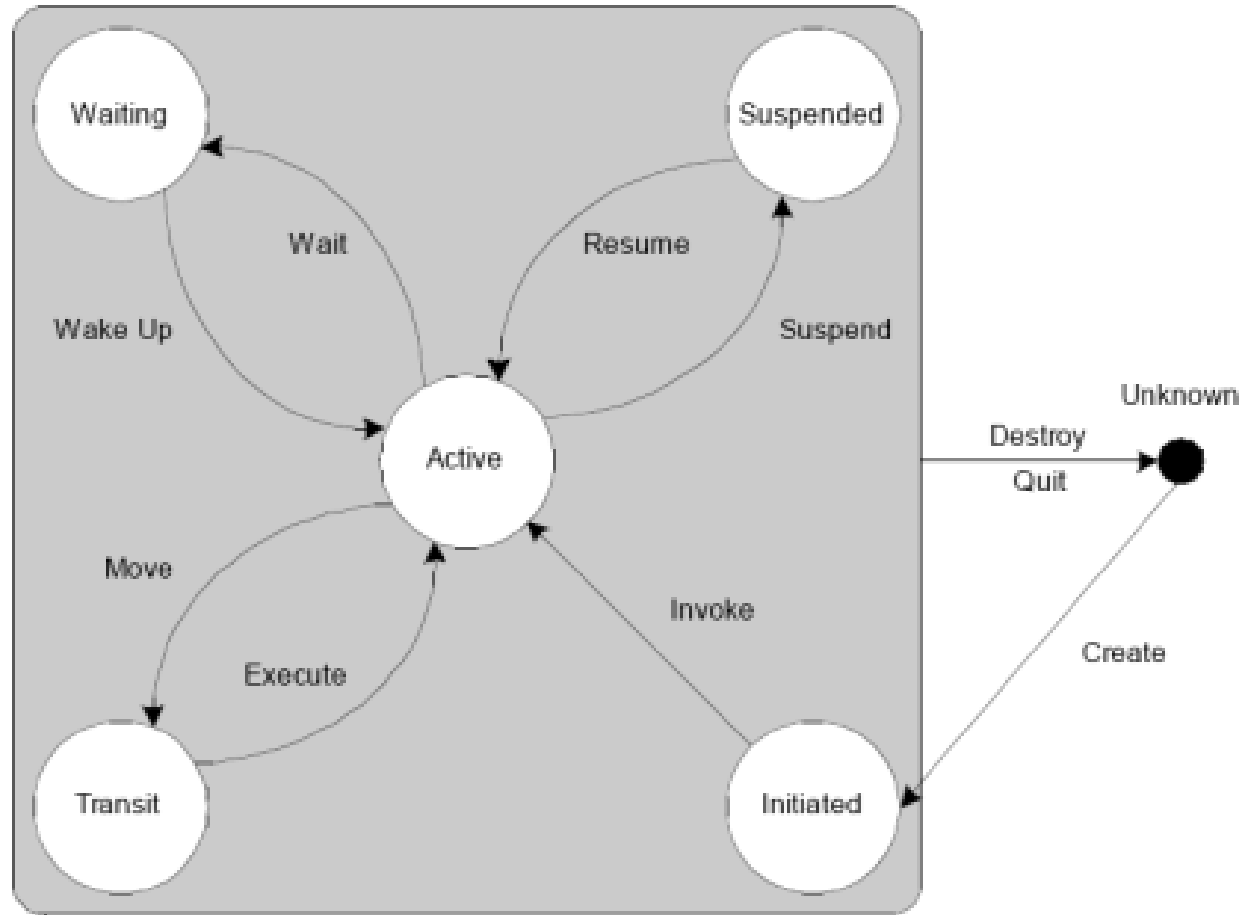


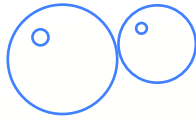
Arquitectura Abstracta FIPA

- Una plataforma de agentes (**Agent Platform, AP**) suministra la infraestructura física en la que se despliegan los agentes. Consiste en sistema operativo, soporte para ejecutar los agente, y los componentes FIPA (DF, AMS, MTS).
 - Un servicio de transporte de mensajes (Message Transport Service , MTS) es el método de comunicación por defecto entre los diferentes agentes.
 - Se asume el patrón IOC (Inversion of Control) . La infraestructura se encarga de la gestión del ciclo de vida de los servicios. => Mayor nivel de desacoplo.

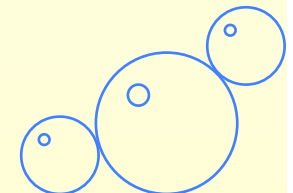
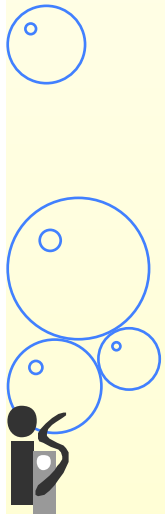
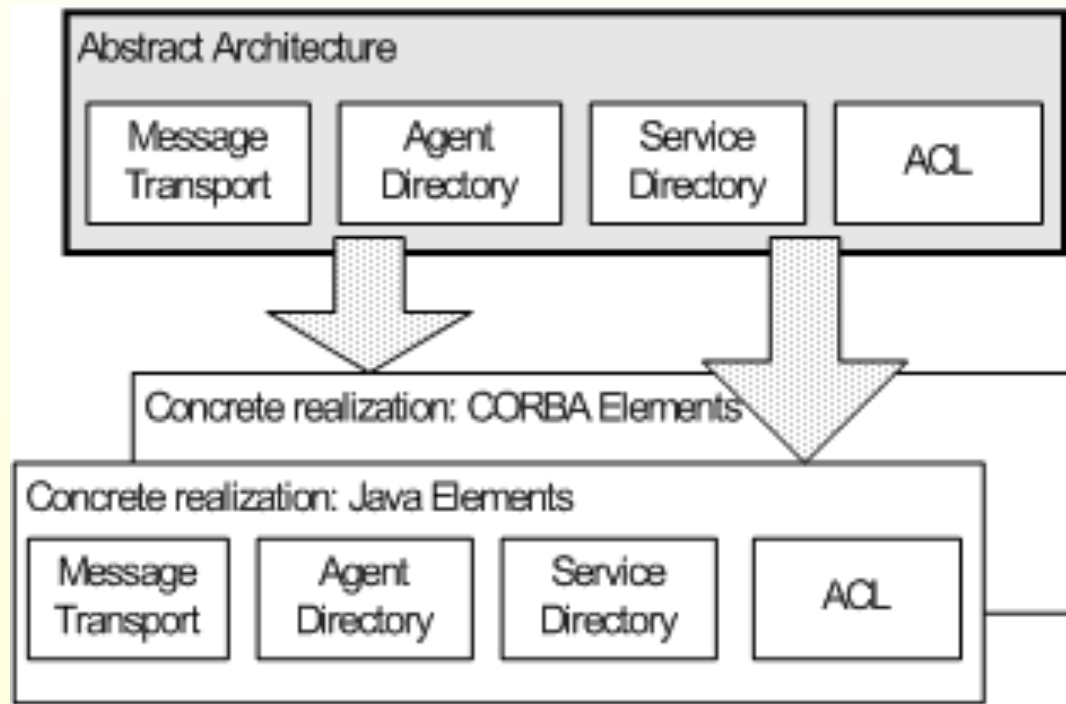


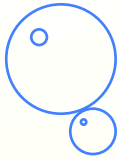
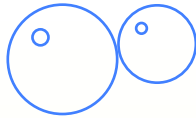
Ciclo de vida de un agente gestionado por el AP





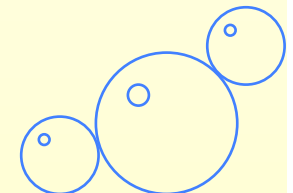
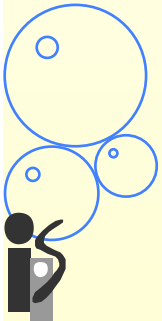
Arquitectura Abstracta FIPA

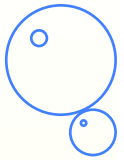
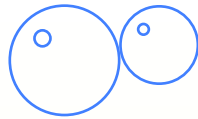




Arquitectura Abstracta FIPA

- Los agentes se comunican intercambiando mensajes
 - que representan actos de habla
 - codificados en un lenguaje de comunicación de agentes
- Servicios de soporte a los agentes:
 - Servicios de directorio
 - Servicios de transporte de mensajes
- Los servicios se pueden implementar
 - como agentes
 - como software que se accede invocando métodos:
 - C++, Java, IDL





Arquitectura Abstracta FIPA

Servicios de Directorio

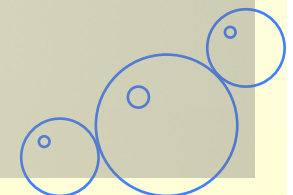
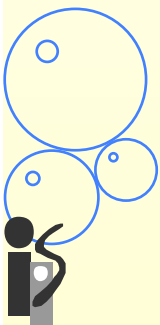
- Soporta un conjunto de entradas formadas por varias tuplas cada una con dos pares clave-valor:
 - **Nombre de agente:** único globalmente
 - **Localizador:** Uno o más descriptores de transporte que describen el tipo de transporte y la dirección de transporte específica para comunicarse con el agente
- Los agentes registran entradas de directorio para que otros agentes puedan buscarlas con el propósito de encontrarles para poder interactuar
 - **Ejemplo de entrada de directorio:**

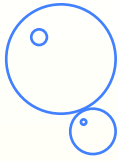
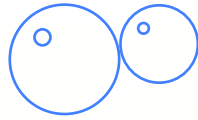
Nombre de agente: AgenteEjemplo

Localizador:	tipo de transporte	dirección de transporte
	HTTP	http://www.agentes.com/AgenteEjemplo
	SMTP	AgenteEjemplo @agentes.com

Atributos del agente: Idiomas: English, Español

Negociación preferida: contract-net

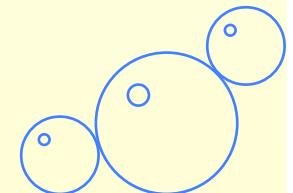
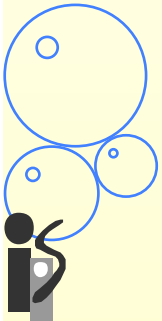


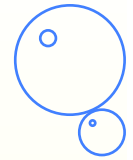
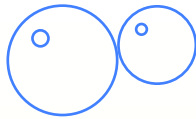


Arquitectura Abstracta FIPA

Mensajes de los agentes

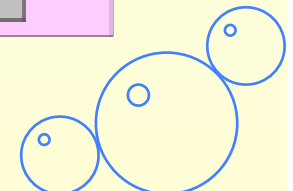
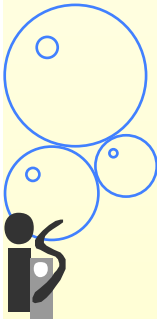
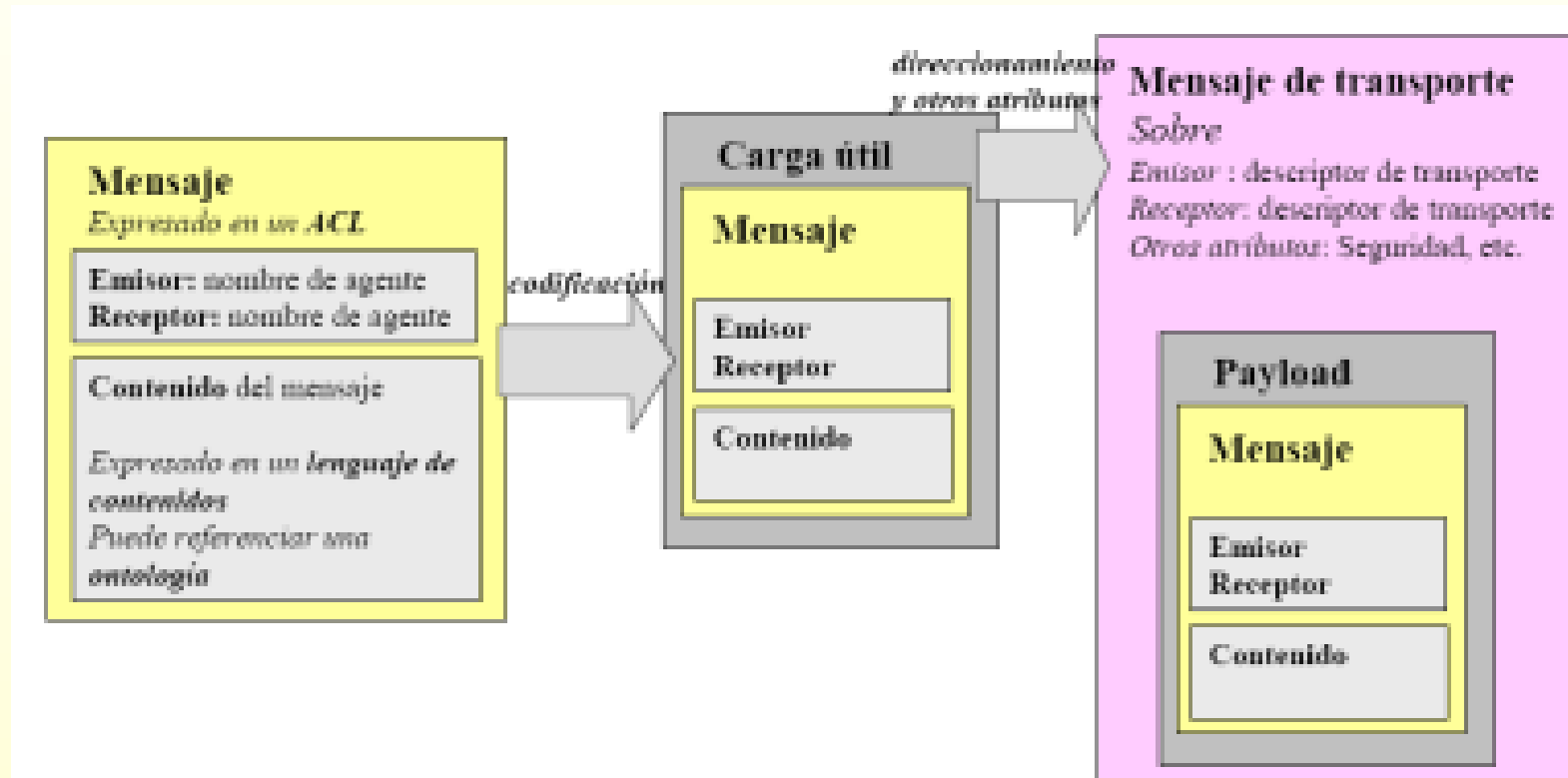
- Estructura de los mensajes
 - Los mensajes son tuplas clave-valor
 - Escritos en un lenguaje de comunicación de agentes (p.ej. FIPA ACL)
 - El contenido expresado con un lenguaje de contenidos (KIF, SL, ...)
 - El lenguaje de contenidos puede hacer referencia a una ontología
 - Incluyen los nombres de emisor y receptor
 - Un mensaje puede contener recursivamente otros mensajes
- Transporte de los mensajes
 - El mensaje de transporte consta de un campo de carga útil (payload) y un sobre (envelope)
 - El mensaje se transporta en la carga útil de un mensaje

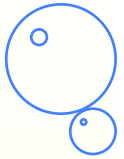
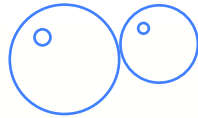




Arquitectura Abstracta FIPA

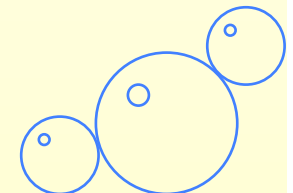
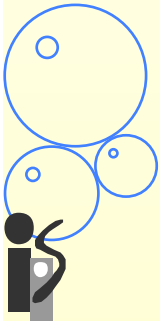
Mensajes de los agentes

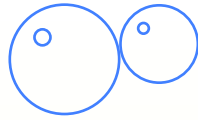




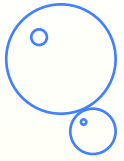
FIPA ACL

- Basado en actos del habla
 - La semántica se basa en aptitudes mentales (creencias, intenciones, etc.)
- Los mensajes son acciones comunicativas
 - Sintaxis similar a KQML
 - Semántica formal definida con lógica modal
- Además se definen protocolos de interacción de alto nivel, llamados conversaciones
- Es posible definir nuevas primitivas a partir de un núcleo de primitivas mediante composición
- El estado de los agentes se describe con el lenguaje SL (Semantic Language)



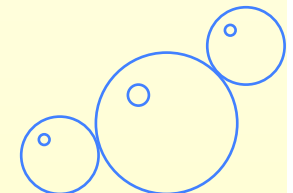
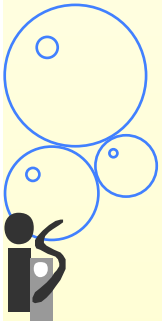


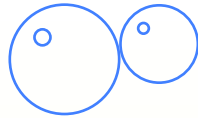
FIPA ACL



Ejemplo de mensaje ACL

```
( inform
  :sender agente1
  :receiver agente2
  :content
    (precio libro 1000)
  :in-reply-to round-4
  :reply-with bid04
  :language sl
  :ontology ontolibro
)
```





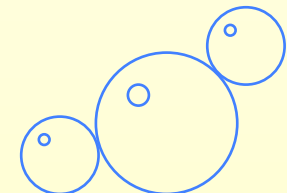
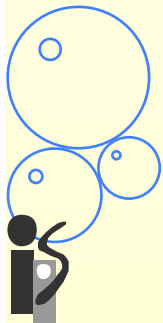
FIPA ACL

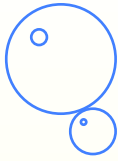
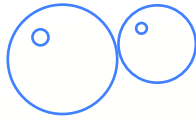
Parámetros del mensaje

:sender	:language
:receiver	:ontology
:content	:reply_by
:reply_with	:protocol
:in_reply_to	:conversation-id
:envelope	

Tipos de mensaje

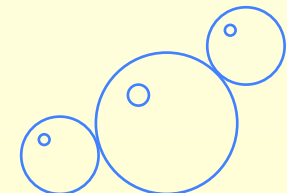
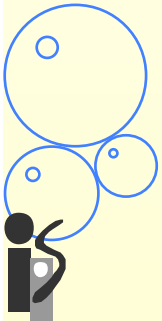
accept-proposal	inform	refuse
agree	inform-if	reject-proposal
cancel	inform-ref	request
cfp	not-understood	request-when
confirm	propose	request-whenever
disconfirm	query-if	subscribe
failure	query-re	

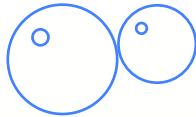




FIPA ACL

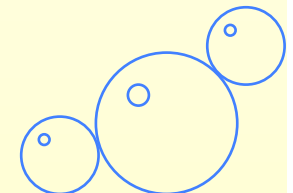
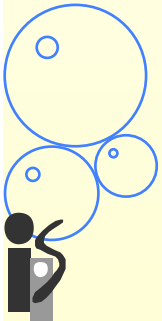
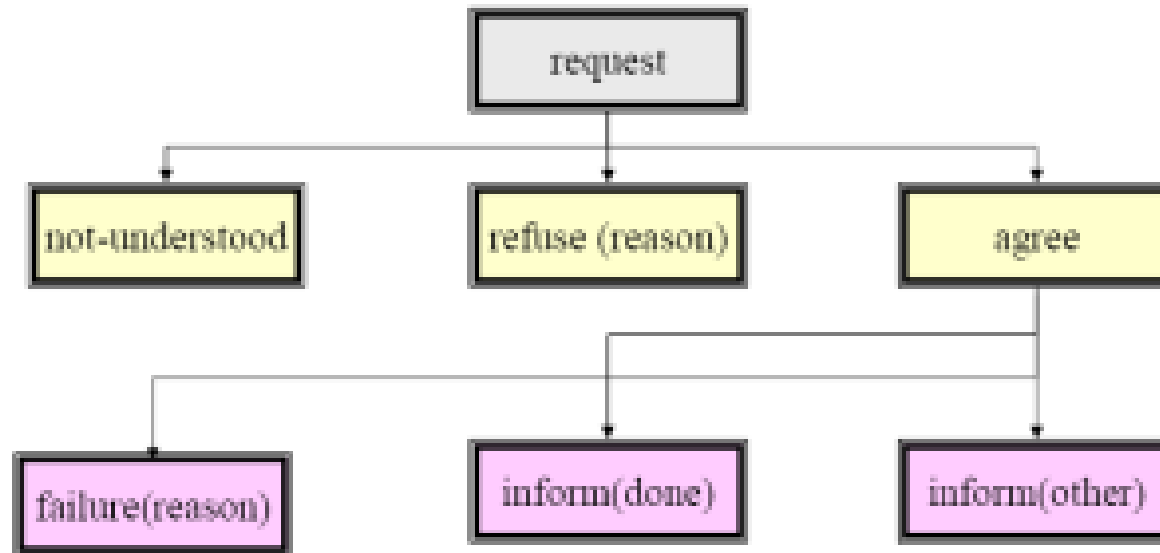
- Las conversaciones entre agentes suelen seguir unos ciertos patrones, secuencia típicas de mensajes: protocolos de conversación
- Un agente informa del protocolo que quiere usar mediante el parámetro :protocol
- Protocolos básicos definidos por FIPA:
 - **FIPA-request**
 - **FIPA-query**
 - **FIPA-request-when**
 - **FIPA-contract-net**
 - **FIPA-iteraterated-contract-net**
 - **FIPA-auction-english**
 - **FIPA-auction-ducth**



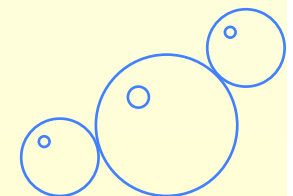
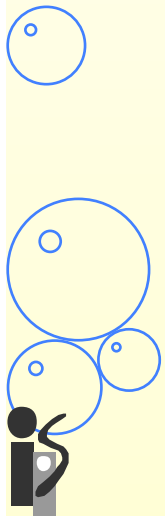
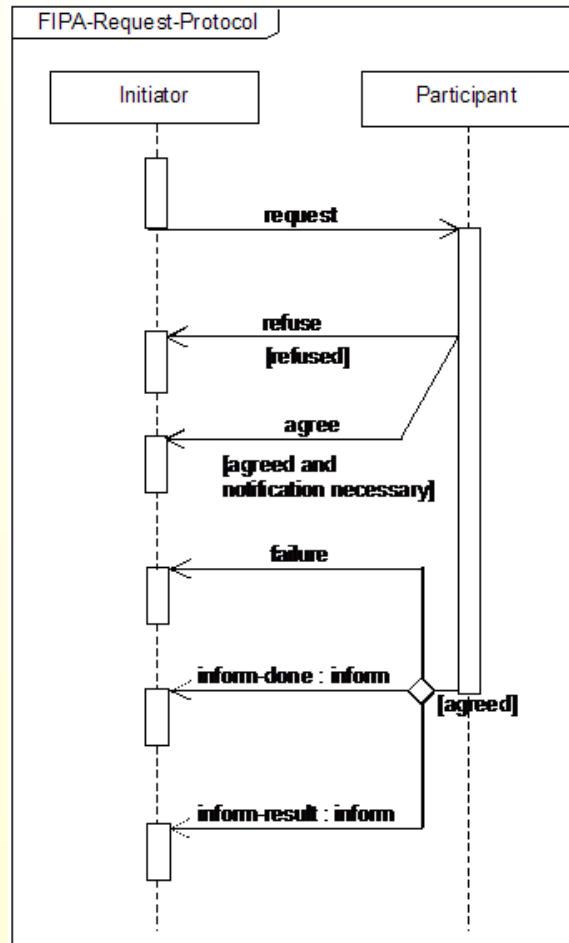


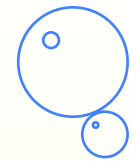
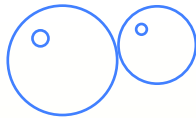
FIPA ACL

Protocolo FIPA request

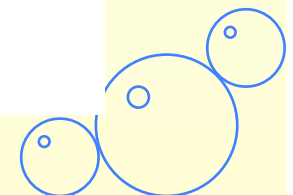
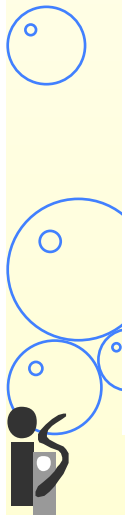
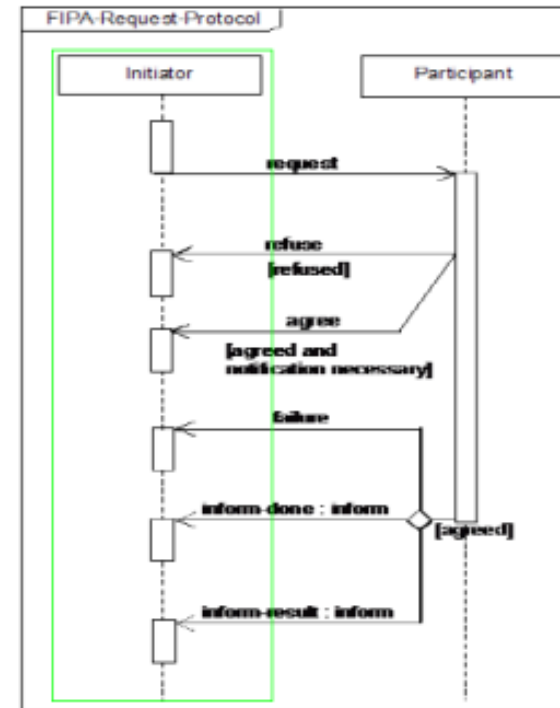
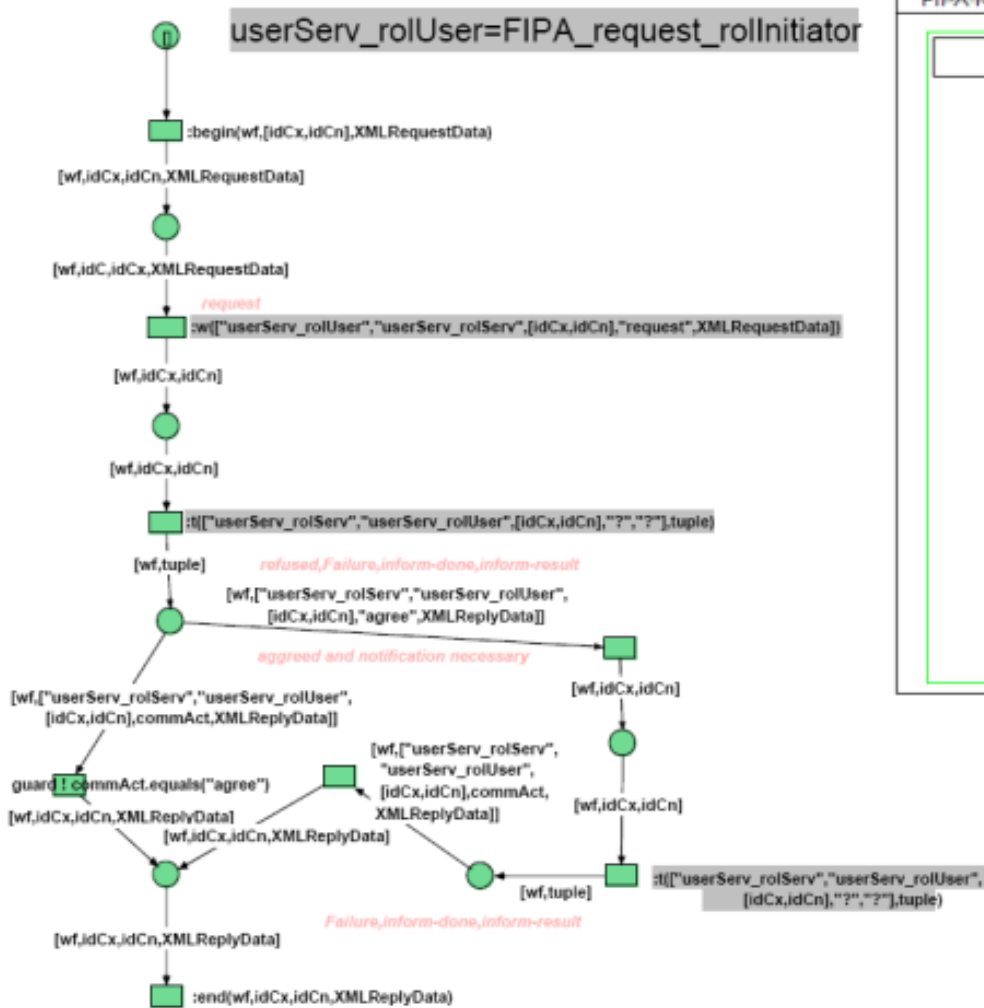


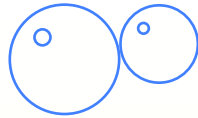
FIPA ACL





FIPA ACL (Renew/Linda Implementation)

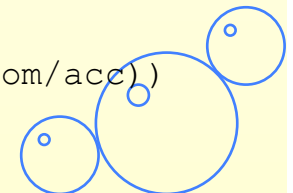
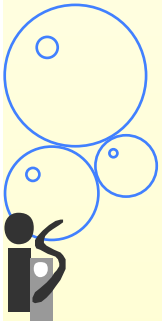


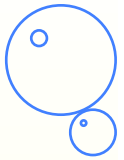
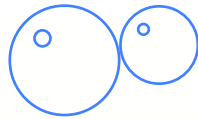


Ejemplos de diálogos

The agent dummy is created and it registers with the AMS of its home AP:

```
(request
  :sender
    (agent-identifier
      :name dummy@foo.com
      :addresses (sequence iiop://foo.com/acc))
  :receiver (set
    (agent-identifier
      :name ams@foo.com
      :addresses (sequence iiop://foo.com/acc)))
  :language fipa-sl0
  :protocol fipa-request
  :ontology fipa-agent-management
  :content
    ((action
      (agent-identifier
        :name ams@foo.com
        :addresses (sequence iiop://foo.com/acc))
      (register
        (ams-agent-description
          :name
            (agent-identifier
              :name dummy@foo.com
              :addresses (sequence iiop://foo.com/acc))
          :state active))))))
```

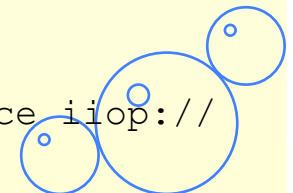
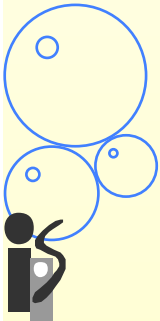


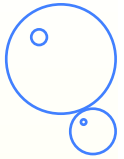
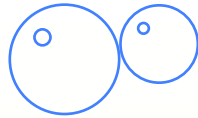


Ejemplos de diálogos

The AMS agrees and then informs dummy of the successful execution of the action:

```
(agree
  :sender
    (agent-identifier
      :name ams@foo.com
      :addresses (sequence iiop://foo.com/acc))
  :receiver (set
    (agent-identifier
      :name dummy@foo.com
      :addresses (sequence iiop://foo.com/acc)))
    :language fipa-sl0
    :protocol fipa-request
    :ontology fipa-agent-management
  :content
    "((action
      (agent-identifier
        :name ams@foo.com
        :addresses (sequence iiop://foo.com/acc))
      (register
        (ams-agent-description
          :name
            (agent-identifier :name dummy@foo.com :addresses (sequence iiop://
foo.com/acc)) :state active))) true)")
```





Ejemplos de diálogos

```
(inform
  :sender
    (agent-identifier
      :name ams@foo.com
      :addresses (sequence iiop://foo.com/acc))
  :receiver (set
    (agent-identifier
      :name dummy@foo.com
      :addresses (sequence iiop://foo.com/acc)))
  :language fipa-sl0
  :protocol fipa-request
  :ontology fipa-agent-management
  :content
    "((done
      (action
        (agent-identifier
          :name ams@foo.com
          :addresses (sequence iiop://foo.com/acc))
        (register
          (ams-agent-description
            :name
              (agent-identifier
                :name dummy@foo.com
                :addresses (sequence iiop://foo.com/acc))
                :state active))))))")
```

