

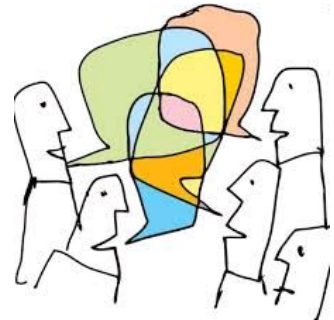


# Desarrollo Basado en Agentes

## Tema 3. Comunicación (y II)

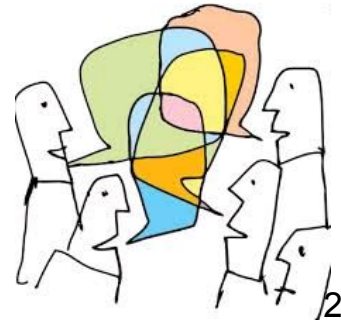
*Luis Castillo Vidal*

*Dpto Ciencias de la Computación e I.A.  
2016-2017*



# Contenido

- Comprensión
- Comunicación



# Comunicación

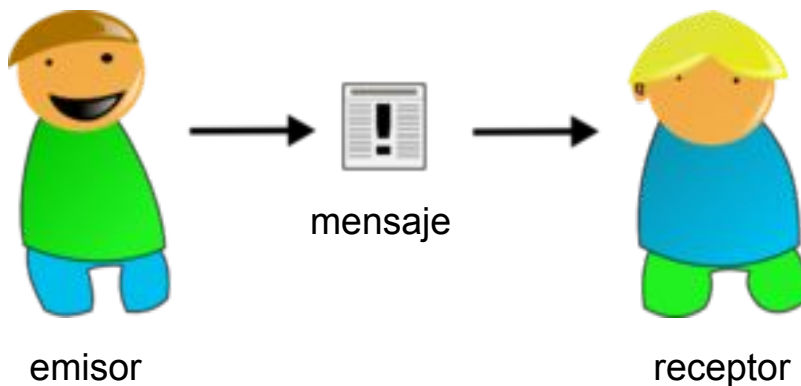
- Necesidad
  - Sincronizarse y evitar interferencias
  - Encargar una tarea a otros
  - Intercambiar información/conocimiento
- En la base de las capacidades de cualquier ser social

```
class objetito {  
  
    void objetito() { ... }  
  
    double cuadrado(double x) {  
        return x*x;  
        // return Math.pow(x, 2);  
    }  
}
```

```
objetito o = new objetito();  
double x;  
  
    x = o.cuadrado(2);  
...
```

# Comunicación

- Teoría de base
  - Semiótica (Ferdinand de Saussure):
    - “¿Por qué y cómo en una determinada sociedad algo —una imagen, un conjunto de palabras, un gesto, un objeto, un comportamiento, etc.— significa?”
  - La comunicación



# Comunicación

- Teoría de base

- Semiótica (Ferdinand de Saussure):

- “¿Por qué y cómo en una determinada sociedad algo —una imagen, un conjunto de palabras, un gesto, un objeto, un comportamiento, etc.— tiene un significado?”

- La comunicación

- Código
    - Canal
    - Emisor
    - Receptor
    - Contenido
    - Contexto
    - → FIPA ACL / KQML
      - Iniciador

# Comunicación

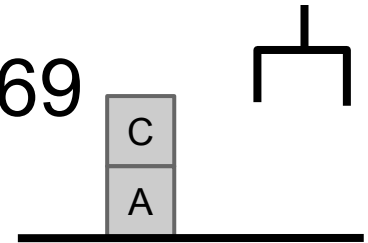
- Teoría de base: Austin '62 y Searle '69
  - **Speech act:** los procesos comunicativos son como una acción: pueden modificar el entorno (estado cognitivo, desires, commitments, emotions)
  - Ejemplo:
    - REQUEST → desires, commitments → actions
    - INFORM → knowledge base ↑
    - Declaración de guerra →
    - Declaración de matrimonio →

# Comunicación

- Teoría de base: Austin '62 y Searle '69
  - El lenguaje no sólo sirve para describir sino para hacer algo
  - **Performativas**: declaraciones con las que una persona realiza una acción → el verbo de una oración → FIPA ACL / KQML
    - Asertivas → describen algo
    - Comprometedoras → agendar una acción
    - Directiva → ordenan/piden una acción
    - Declarativa → en sí misma una acción
    - Expresiva → emociones

# Comunicación

- Teoría de base: Austin '62 y Searle '69
  - Dado un hecho “sobre(C,A)” el significado cambia según quién emite el mensaje y la performativa utilizada
    - REQUEST sobre(C,A)
    - INFORM sobre(C,A)
    - FAILURE sobre(C,A)
    - QUERY-IF sobre(C,A)
- ¿Son suficientes y/o necesarias estas performativas? → debate abierto





# Comunicación

- El acto comunicativo
  - Mensaje: Lenguajes de comunicación entre agentes
    - FIPA (EU) ACL
    - DARPA (US) KQML
    - Estructura y significado muy parecidos o equivalentes entre ambos → nos quedamos con ACL
  - Estructura del mensaje (parámetros) ← Semiótica
  - El contenido del mensaje es parte de la estructura y el “contenedor” principal.

# Comunicación

- Estructura del mensaje ACL (parámetros)
  - **performative** Tipo de acto comunicativo
  - **sender** ID del agente emisor
  - **receiver** ID(s) del agente(s) receptor(es)
  - **reply-to** ID del agente que recibirá la(s) respuesta(s)
  - **content** Contenido del mensaje u objeto de la acción
  - **language** Idioma
  - **encoding** Formalismo de representación
  - **ontology** Significado de los objetos del contenido
  - **conversation-id** Expresión (string) como ID de la conversación. Único.
  - **reply-by** Expresión (date/time) timeout para recibir respuesta
  - **protocol** Procotolo de interacción (requiere los anteriores)
  - **reply-with** ID de la respuesta a este mensaje
  - **in-reply-to** ID de la respuesta enviada

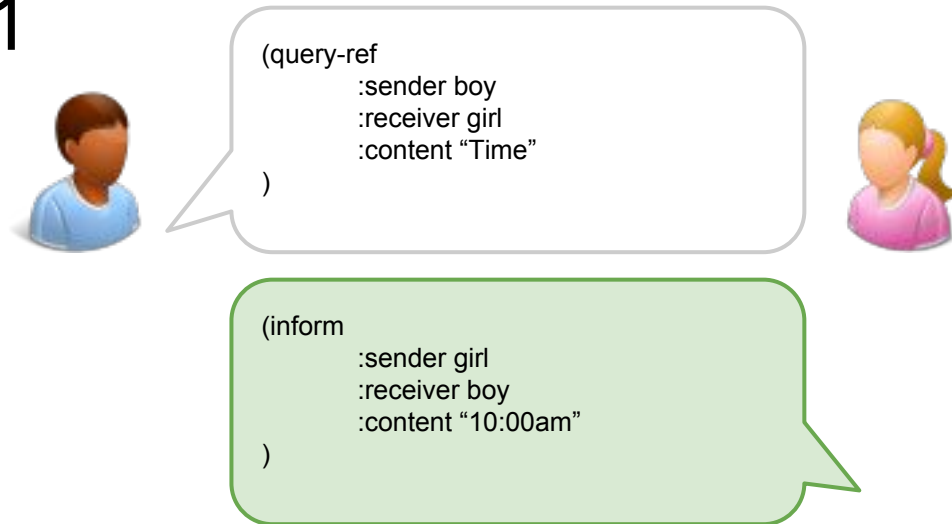
# Comunicación

- Performativas ACL (tipo de mensaje)

performative	passing info	requesting info	negotiation	performing actions	error handling
accept-proposal			x		
agree				x	
cancel		x		x	
cfp			x		
confirm	x				
disconfirm	x				
failure					x
inform	x				
inform-if	x				
inform-ref	x				
not-understood					x
propose			x		
query-if		x			
query-ref		x			
refuse				x	
reject-proposal			x		
request				x	
request-when				x	
request-whenever				x	
subscribe		x			

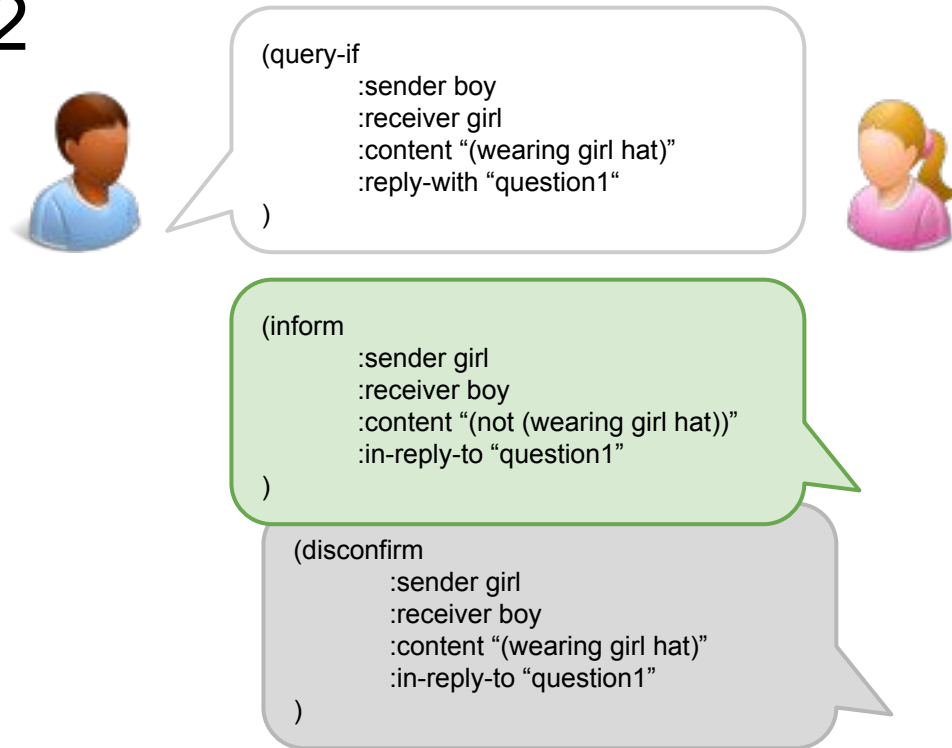
# Comunicación

- Ejemplo 1



# Comunicación

- Ejemplo 2



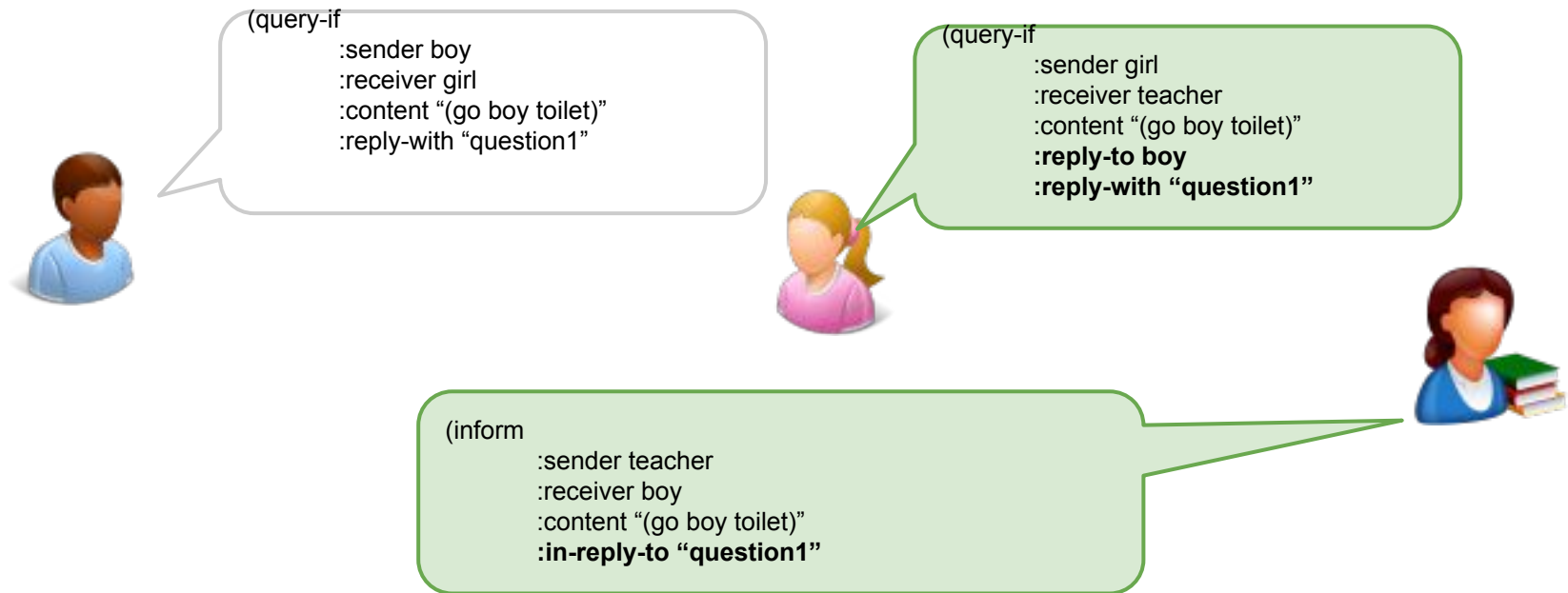
# Comunicación

## ● Ejemplo 3



# Comunicación

## ● Ejemplo 4



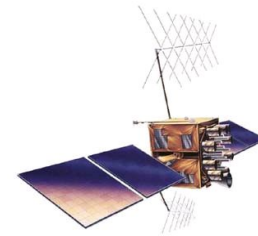
# Comunicación

## ● Ejemplo 5



```
(request  
  :sender bot  
  :receiver gps  
  :content "status"  
  :encoding "JSon")
```

```
(inform  
  :sender gps  
  :receiver bot  
  :content "{ \"connected\": \"YES\", \"ready\": \"YES\",  
    \"gps\": { \"x\": 10, \"y\": 5 }, \"goal\": \"No\", \"battery\": 100,  
    \"radar\": [0,0,0,0,0,0,0,1,1] }"  
  :encoding "JSon")
```





# Comunicación

## ● Ejemplo 6

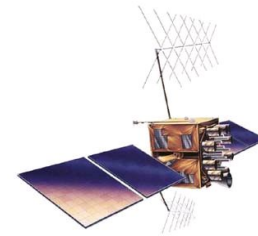


```
(request
:sender bot
:receiver gps
:content "status"
:encoding "JSon"
:reply-with "request104")
```

Ø

```
(request
:sender bot
:receiver gps
:content "moveN"
:encoding "JSon"
:reply-with "request105")
```

```
(inform
:sender gps
:receiver bot
:content "{ \"connected\": \"YES\", \"ready\": \"YES\",
  \"gps\": { \"x\": 10, \"y\": 5 }, \"goal\": \"No\",
  \"gonio\": { \"alpha\": 0, \"dist\": 4.0 }, \"battery\": 100,
  \"radar\": [0, 0, 0, 0, 0, 0, 0, 1, 1] }"
:encoding "JSon"
:in-reply-to "request104")
```



# Comunicación

- La semántica de las performativas
  - Lo que el agente receptor interpretaría al recibir un mensaje con esa performativa
    - En general, por acuerdo de la comunidad, tienen una semántica predefinida
      - Protocolos de comunicación “estándar”
      - Agentes mediadores
    - Revisable en cada plataforma
      - Protocolos de comunicación ad-hoc
      - Posiblemente los agent protocolos ad-hoc no p entenderse bien con oti  
fuera de su plataforma
  - + Info FIPA Communicative / Library Specification



## ● REQUEST

- Significado: El emisor le pide al receptor que ejecute una acción
- Contenido: La expresión de una acción → formato??
- Condiciones:
  - Emisor  $\neq$  Receptor
  - El emisor sabe que el receptor puede hacerlo
    - Agentes predeterminados para servicios de directorio y páginas amarillas (según plataforma)
  - El emisor sabe que la acción no está hecha aún

```
(request :sender gugelcar :receiver controller :content "moveN")
```

```
(request :sender GUI :receiver robot :content "(:action Desapilar :parameters (?X ?Y - Block)  
:condition (and (manovacia) (libre ?X)(sobre ?X ?Y)) :duration (= ?duration 25):effects (and  
(not (manovacia))(not (libre ?X))(not (sobre ?X ?Y))(cogido ?X) (libre ?Y)))" :encoding  
"PDDL")
```

## ● REQUEST

### ○ Posibles respuestas (reply-with)

#### ■ INFORM

- Significado:
  - Receptor ejecutó la acción con éxito y/o
  - (opc) Resultado de la ejecución de la acción
- Contenido: La expresión de la acción realizada

#### ■ REFUSE

- Significado: Receptor rechaza ejecutar la acción
- Contenido: La expresión de la acción y una condición que indica la razón del rechazo (true)
  - Agentes cooperativos

```
(inform :sender gps :receiver bot :content {"done":{"action":"Move N"}})
```

```
(inform :sender gps :receiver bot :content {"action":"Move N","result":"OK"})
```

```
(refuse :sender gps :receiver bot :content {"action":"Move N","condition":"Bad map"})
```

- **REQUEST**

- Posibles respuestas (reply-with)

- **AGREE**

- Significado: Receptor acepta ejecutar la acción (futuro)
        - Receptor puede subcontratar a otro para que lo haga
      - Contenido: Expresión de la acción y condición a cumplir

- **FAILURE**

- Significado: Receptor intentó la acción y falló
      - Contenido: Expresión de la acción y una expresión que indica el error

```
(agree :sender gps :receiver bot :content {"action":"Move N","when":"10:00"})
```

```
(failure :sender gps :receiver bot :content {"action":"Move N","reason":"crashed"})
```

- **REQUEST**

- Posibles respuestas (reply-with)

- **NOT-UNDERSTOOD**

- Significado: Receptor no entiende el mensaje recibido (múltiples razones)
      - Contenido: copia del mensaje y, opcionalmente, alguna condición que explique por qué no se ha comprendido

```
(not-understood :sender gps :receiver bot :content {"action":":action Desapilar  
:parameters (?X ?Y - Block) ... ","reason":"PDDL encoding unknown"})
```

## ● CANCEL

- Significado: Emisor cancela la ejecución de la acción por el receptor. Debe haber un compromiso previo entre el emisor y el receptor.
- Contenido: Expresión de la acción cancelada

```
(request :sender bot :receiver gps :content {"action":"Move N","when":"10:00"})
```

```
(agree :sender gps :receiver bot :content {"action":"Move N","when":"10:00"})
```

```
(cancel :sender bot :receiver gps :content {"action":"Move N","when":"10:00"})
```

## ● REQUEST-WHEN

- Significado: El emisor pide al receptor que ejecute una acción cuando se de una condición, sólo la primera vez.
- Contenido: Descripción de la acción y de la condición
- Implica mecanismos complejos para **acciones diferidas y estados mentales complejos** (comprobar la condición)

```
(request-when :sender bot :receiver gps :content {"action":"Move N","when":"at 12:00"})
```

```
(request-when :sender bot :receiver gps :content {"action":"Move N","when":"(has fuel)"}))
```



- **REQUEST-WHEN**
  - Posibles respuestas
    - REFUSE
    - AGREE
    - INFORM
    - FAILURE
    - NOT-UNDERSTOOD

## ● REQUEST-WHENEVER

- Significado: El emisor pide al receptor que ejecute una acción cuando se de una condición, todas veces.
- El compromiso es para siempre o hasta que se recibe un CANCEL.
- Contenido: Descripción de la acción y de la condición

```
(request-whenEVER :sender user :receiver alarm :content {"action":"Wake me up",  
"when":"at 06:00am"})
```

- **REQUEST-WHENEVER**
  - Posibles respuestas
    - REFUSE
    - AGREE
    - INFORM
    - FAILURE
    - NOT-UNDERSTOOD

## ● QUERY-IF

- Significado: El emisor desconoce el valor de verdad de una fórmula y se la pregunta al receptor
- Contenido: una fórmula
- Posibles respuestas: REFUSE, INFORM, CONFIRM, DISCONFIRM

```
(query-if :sender bot :receiver gps :content "(isgoal bot)" )
```

```
(inform :sender gps :receiver bot :content "(isgoal bot)" )
```

```
(confirm :sender gps :receiver bot :content "(isgoal bot)" )
```

```
(inform :sender gps :receiver bot :content "(not (isgoal bot))" )
```

```
(disconfirm :sender gps :receiver bot :content "(isgoal bot)" )
```

## ● QUERY-REF

- Significado: El emisor le pregunta al receptor por el valor de un objeto, el cual debe serializarse para poder ser transmitido
- Contenido: una referencia a un objeto
  - Serializado
  - Servicio web
- Posibles respuestas: REFUSE, INFORM

```
(query-ref :sender bot :receiver gps :reply-with "q01" :content "radar" )
```

```
(inform :sender gps :receiver bot :content {"radar":[0,0,0,0,0,0,0,1,1]}  
encoding JSon :in-reply-to "q01")
```

## ● SUBSCRIBE

- Significado: El emisor le pide al receptor el valor de un objeto. Cada vez que se el objeto cambie, se debe volver a informar al emisor. Es una versión persistente de QUERY-REF.
- La suscripción persiste hasta que se recibe CANCEL
- Contenido: una referencia a un objeto
- Posibles respuestas: INFORM, REFUSE

```
(subscribe :sender bot :receiver gps :reply-with "q01" :content "radar" )
```

```
(inform :sender gps :receiver bot :content {"radar": [0,0,0,0,0,0,0,1,1]}  
encoding JSon :in-reply-to "q01")
```

```
(inform :sender gps :receiver bot :content {"radar": [1,1,1,1,0,0,0,0,0]}  
encoding JSon :in-reply-to "q01")
```

```
(cancel :sender bot :receiver gps :content "radar" )
```

- **INFORM**

- Significado: pues ya se ha ido viendo
- Contenido: una fórmula

- **INFORM-IF / INFORM-REF**

- Significado: Macro asociada a REQUEST
- Contenido: una fórmula o una referencia
- $\text{QUERY-IF} \equiv \text{REQUEST} \supseteq \text{INFORM-IF}$
- $\text{QUERY-REF} \equiv \text{REQUEST} \supseteq \text{INFORM-REF}$



- **CFP**

- Significado: (Call For Proposal) petición para ver quién puede realizar una acción. Inicia un proceso de negociación con los agentes que responden
  - Subastas
  - Regateos
  - Coaliciones
- Contenido: La acción que se solicita y las precondiciones de la acción

- **PROPOSE**

- Significado: El emisor le propone al receptor que el emisor realice una acción, dadas una condiciones iniciales (SLA). Inversa a REQUEST.
- Contenido: la acción que se solicita y las condiciones para la prestación del servicio.

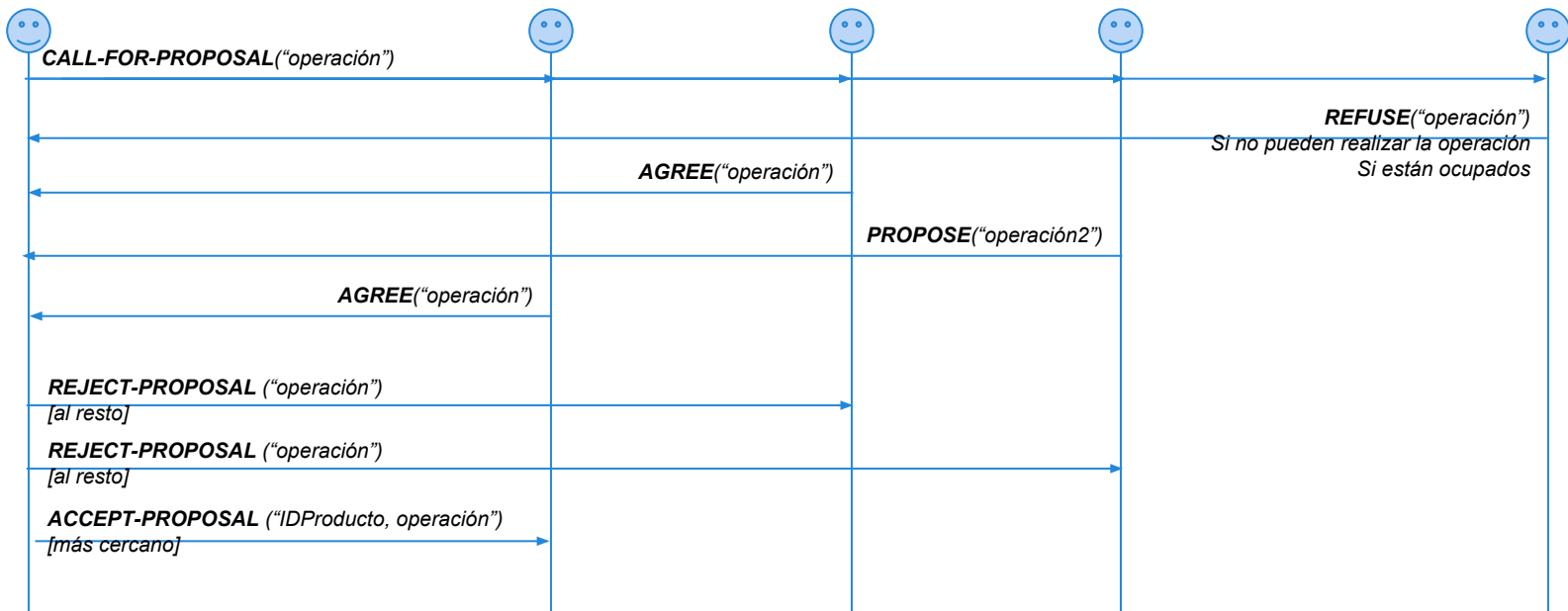
- **ACCEPT-PROPOSAL**

- Significado: aceptar una propuesta anterior para realizar una acción
- Contenido: la acción que se solicitó y las condiciones de la aceptación impuestas por el receptor de la propuesta (emisor del accept-proposal)

- **REJECT-PROPOSAL**

- Significado: rechazar una propuesta anterior para realizar una acción
- Contenido: la acción que se solicitó y las condiciones del rechazo (opcional).

Un agente pide realizar una operación a otros agentes y elige el más cercano a él de entre los que pueden hacerla



# Comunicación

- Performativas ACL (tipo de mensaje)

performative	passing info	requesting info	negotiation	performing actions	error handling
accept-proposal			x		
agree				x	
cancel		x		x	
cfp			x		
confirm	x				
disconfirm	x				
failure					x
inform	x				
inform-if	x				
inform-ref	x				
not-understood					x
propose			x		
query-if		x			
query-ref		x			
refuse				x	
reject-proposal			x		
request				x	
request-when				x	
request-whenever				x	
subscribe		x			

# Comunicación

- **El lenguaje para el contenido**
  - Hasta ahora el contenido de un mensaje sigue un formato libre, pero se pueden usar otras estructuras conocidas
    - Formales
      - KIF
      - FIPA-SL ([Content Language specification](#))
    - Informales
      - XML
      - JSon

# Comunicación

- **KQML**
  - Equivalente a ACL

<i>Category</i>	<i>Name</i>	<i>advertise</i>	<i>subscribe</i>	<i>standby</i>	<i>forward broadcast</i>	<i>Facilitation performatives</i>
<b>Discourse</b>	ask-if	X	X		X	X
	ask-all	X	X		X	X
	ask-one	X	X	X	X	X
	stream-all	X	X	X	X	X
	eos			X	X	
	tell			X	X	
	untell				X	
	deny				X	
	insert	X			X	X
	uninsert				X	
	delete-one	X			X	X
	delete-all	X			X	X
	undelete				X	
	achieve	X			X	X
	unachieve				X	
	advertise				X	
	unadvertise				X	
	subscribe	X		X	X	X
<b>Intervention and Mechanics</b>	error				X	
	sorry				X	
	standby				X	
	ready				X	
	next				X	
	rest				X	
	discard				X	
<b>Facilitation and Networking</b>	register					
	unregister					
	forward					
	broadcast					
	transport-address					
	broker-one				X	
	broker-all				X	
	recommend-one		X	X	X	
	recommend-all		X	X	X	
	recruit-one				X	
	recruit-all				X	



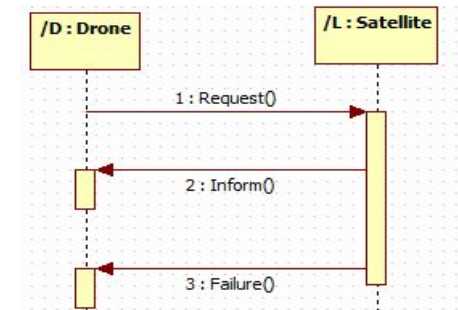
# Comunicación

## ● Conversaciones

- Las performativas por sí solas no son suficiente, sino que se espera un diálogo con ellas: un protocolo de conversación
- Usar los campos apropiados del mensaje ACL
  - **performative** Tipo de acto comunicativo
  - **sender** ID del agente emisor
  - **receiver** ID(s) del agente(s) receptor(es)
  - **reply-to** ID del agente que recibirá la(s) respuesta(s)
  - **content** Contenido del mensaje u objeto de la acción
  - **language** Idioma
  - **encoding** Formalismo de representación
  - **ontology** Significado de los objetos del contenido
  - **conversation-id** Expresión (string) como ID de la conversación. Único.
  - **reply-by** Expresión (date/time) timeout para recibir respuesta
  - **protocol** Procotolo de interacción (requiere los anteriores)
  - **reply-with** ID de la respuesta a este mensaje
  - **in-reply-to** ID de la respuesta

# Comunicación

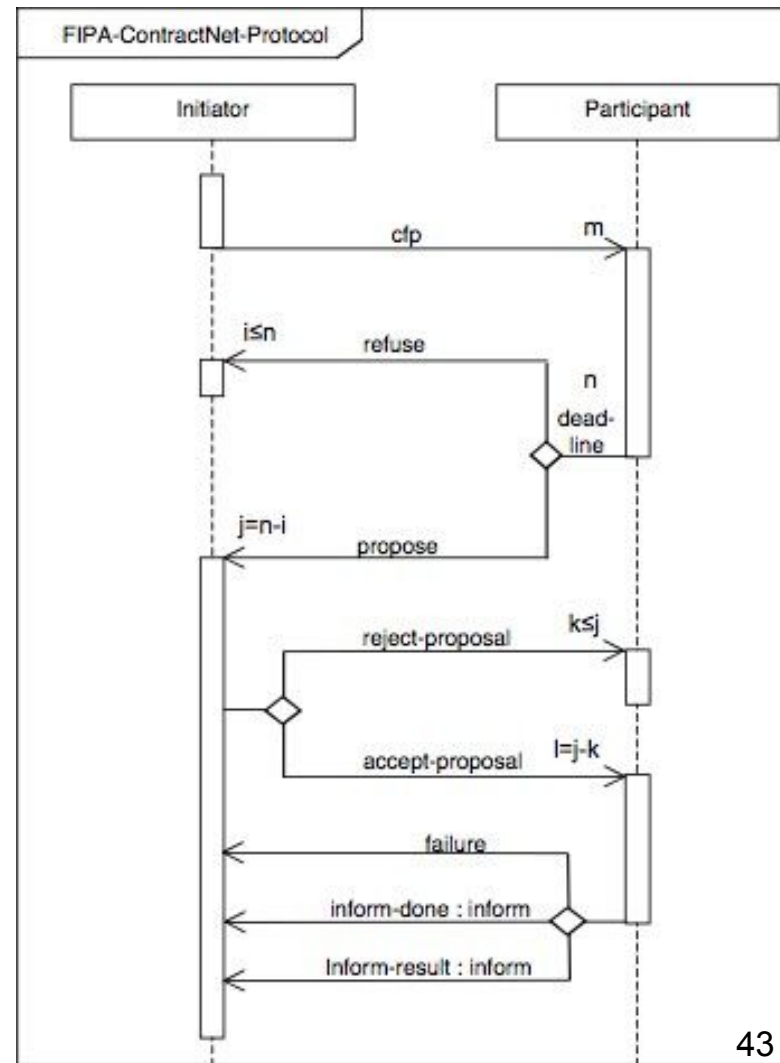
- Conversaciones
  - Un **protocolo** es una conversación entre agentes, la cual sigue una serie de reglas que describen qué performativa usar y cuándo usarla para completar la conversación correctamente



# Comunicación

- Conversaciones

- Un **protocolo** es una conversación entre agentes, la cual sigue una serie de reglas que describen qué performativa usar y cuándo usarla para completar la conversación correctamente → **es un FSA asociado a la conversación**
- Initiator y Responder
- **El contenido del mensaje?**
- Diagramas de secuencia
- **Verbosity**

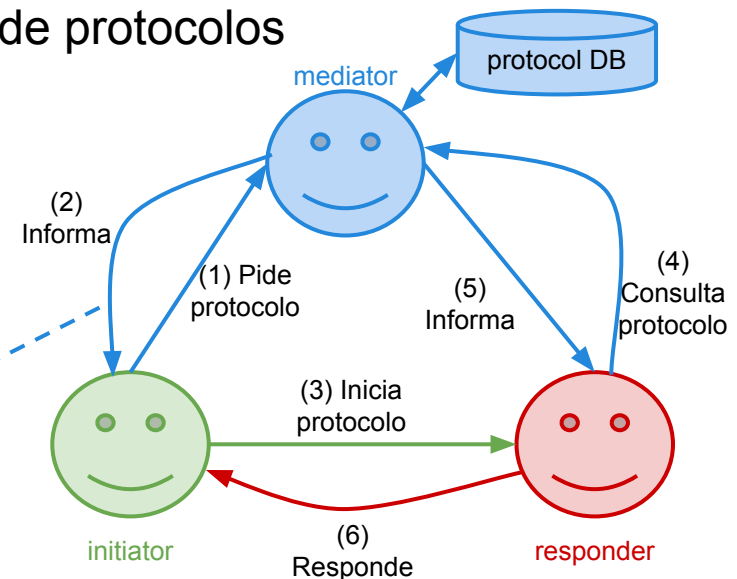


# Comunicación

- Conversaciones

- El protocolo de la conversación debe ser conocido a priori entre el emisor y el receptor (subrogación)
  - Es un protocolo estándar (como contract-net)
  - Es un protocolo existente que se puede consultar
    - Agentes mediadores y librería de protocolos

```
{ "protocol-name": "contract-net",  
  "states": [  
    "s_init", "s_end", "s_receive", "s_propose", "s_reject", "s_cancel"  
  ],  
  "transitions": {  
    "s_init": { "call-for-proposal": "s_receive",  
    "s_receive": { "refuse": "s_end", "not-understood": "s_end", "propose": "s_propose",  
    "s_propose": { "accept-proposal": "s_end", "reject-proposal": "s_reject",  
    "s_reject": { "inform": "s_cancel",  
    "s_cancel": { "cancel": "s_end"  
  }  
}
```

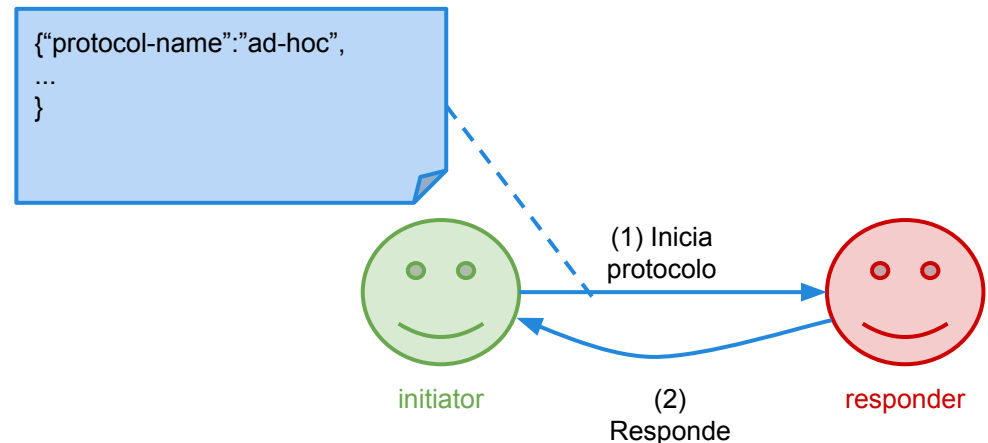


# Comunicación

- Conversaciones

- Conversación dinámica

- Definir protocolo dinámico en el content



- Ya se requieren sociedades más complejas

- Agentes mediadores
  - Service Directory

# Comunicación

- Conversaciones

- De hecho FIPA define los protocolos más importantes
  - REQUEST
  - REQUEST-WHEN
  - QUERY
  - CONTRACT-NET
  - BROKERING
  - RECRUITING
  - SUBSCRIBE
  - PROPOSE
- Verbosity y tráfico en la red
- → TEMA 4

