



FIPA Specifications and ACL



The Foundation for Intelligent Physical Agents



FIPA

“FIPA is an IEEE Computer Society standards organization that promotes agent-based technology and the interoperability of its standards with other technologies”.

FIPA è una organizzazione svizzera fondata nel 1996 e costituita per la produzione di specifiche e standard per sistemi ad agente.

Fin dalla sua fondazione, ha svolto un ruolo cruciale nello sviluppo di standard e ha promosso una serie di iniziative ed eventi che hanno contribuito allo sviluppo e alla diffusione delle tecnologia ad agenti.



FIPA & IEEE

Nel marzo 2005 FIPA entra a far parte dell' IEEE Computer Society.

Una descrizione completa delle specifiche FIPA ed il loro stato attuale possono essere trovati sul Web site di FIPA (<http://www.fipa.org>).

Il presidente di FIPA è James Odell, consulente indipendente specializzato in applicazioni object-oriented e agent-based. (ref: <http://www.jamesodell.com>).



FIPA specifications

Le FIPA specifications rappresentano un insieme di norme destinate a promuovere l'interazione tra agenti software eterogenei e servizi che questi possono rappresentare.

Un *agente FIPA* è considerato come una particolare entità software in grado di agire in modo autonomo prelevando informazioni dall'ambiente in cui si trova e agendo secondo le proprie conoscenze, di scambiare informazioni con altri agenti o con esseri umani, e di prendere l'iniziativa per soddisfare i propri goals.

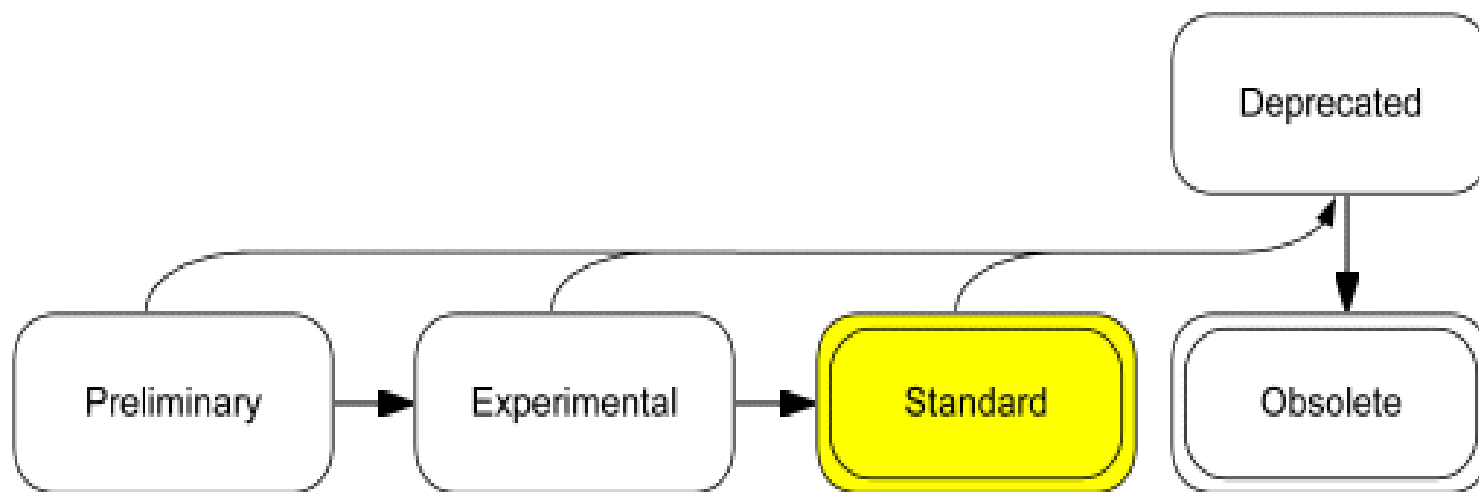


FIPA standard

Nel 2002, FIPA completò il processo di standardizzazione di un sottogruppo delle sue Specifiche.

Una specifica FIPA prima di diventare uno standard passa dallo stato di *Preliminary* e successivamente allo stato di *Experimental*.

Le specifiche diventate standard sono 25.



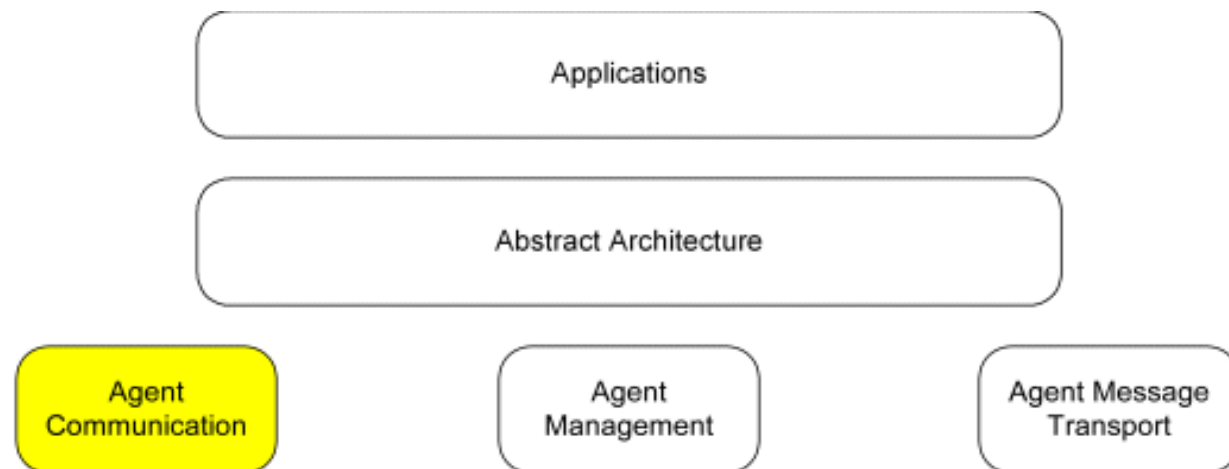


FIPA Specifications category

Le specifiche FIPA (standard e non) sono raggruppate in 5 categorie:

agent communication, agent transport, agent management, abstract architecture and applications.

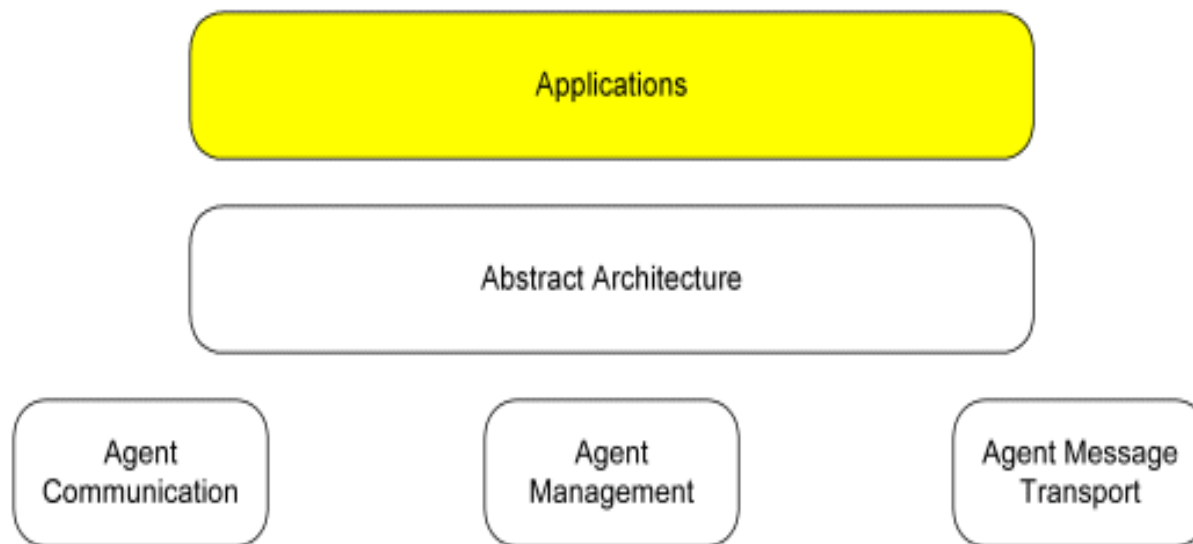
Agent Communication è la principale categoria del modello *FIPA multi-agent system*.





Applications

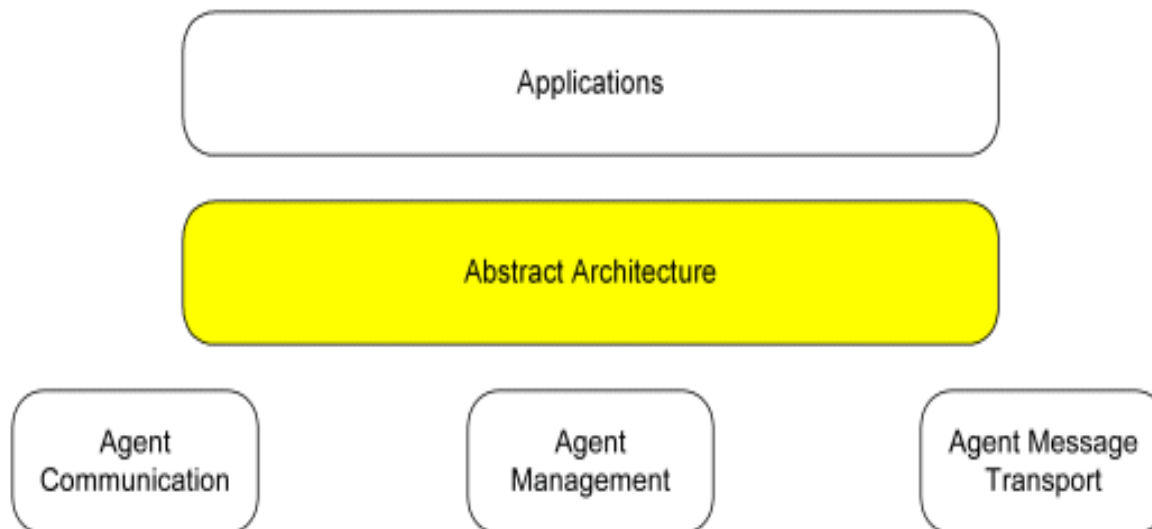
Le *FIPA Application Specifications* illustrano alcuni esempi di differenti campi di applicazione in cui può essere utilizzata la tecnologia ad agenti FIPA.





Abstract Architecture

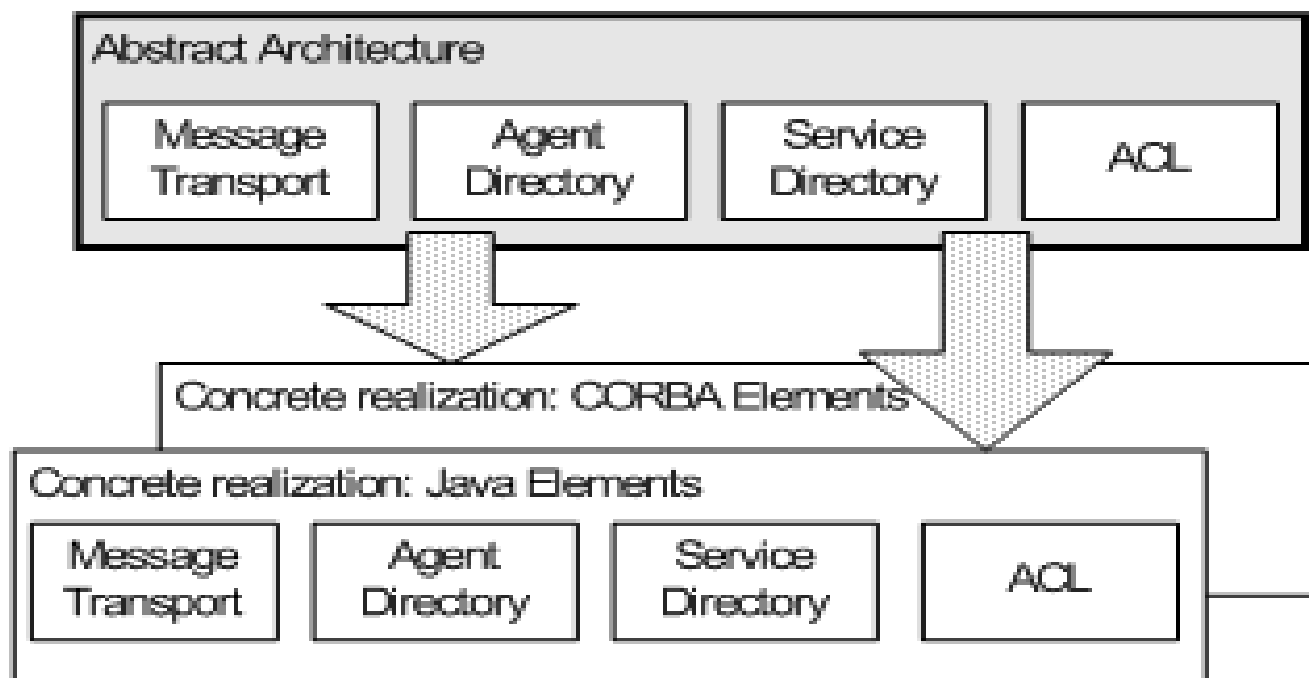
Le specifiche *FIPA Abstract Architecture* descrivono le entità astratte che sono richieste per costruire un sistema ad agenti. Viene descritto l'ambiente, gli agenti, i servizi a disposizione degli agenti e le relazioni tra gli elementi del sistema.





Abstract Architecture 2

Obiettivo principale è di fornire un supporto per lo scambio di messaggi tra agenti i quali possono usare differenti *agent communication language* (ACL) o differenti *content language*.





Abstract Architecture 3

Questa architettura include un modello per :

- la descrizione dei servizi a disposizione per gli agenti
- la ricerca dei servizi da parte degli agenti e degli altri servizi,
- l'interoperabilità nel trasporto dei messaggi,
- il supporto a varie forme di *Communication Language*,
- la rappresentazione di *directory services*.



Abstract Elements

Agent, Agent Name and Agent Attribute,

Agent Directory Service and Agent Directory Entry and Agent Locator,

Agent Communication Language, Content, Content Language, Ontology, and Encoding Representation

Transport Message, Envelope Message, Transport Description, Payload

Transport Service, Encoding Service, Service Address, Service Attributes, Service Directory, Service Directory Entry, Services , Service Identifier, Service Location Description, Service Locator, Service Root, Service Signature, Service Type, Signature Type, Transport, Transport Specific Address, Transport Specific Property, Transport Type.



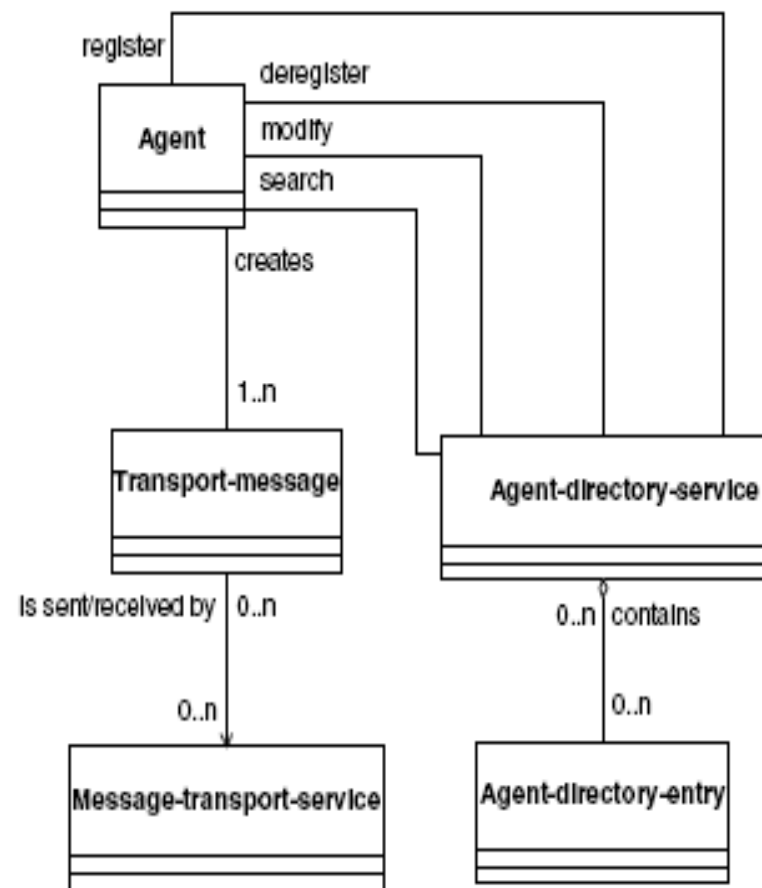
Agent

Entità software in grado di agire in modo autonomo e comunicare usando un Agent Communication Language.

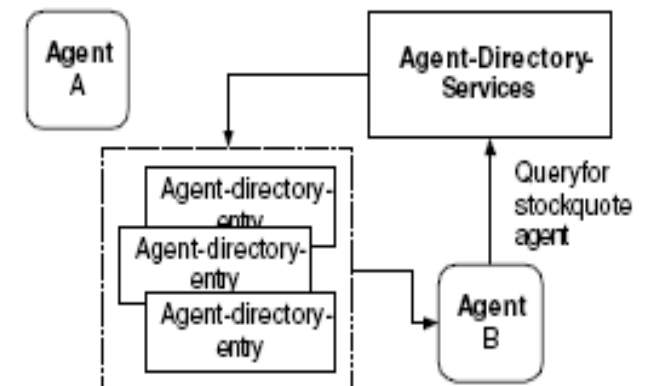
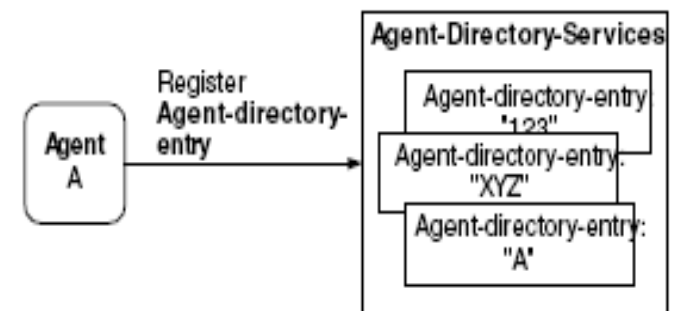
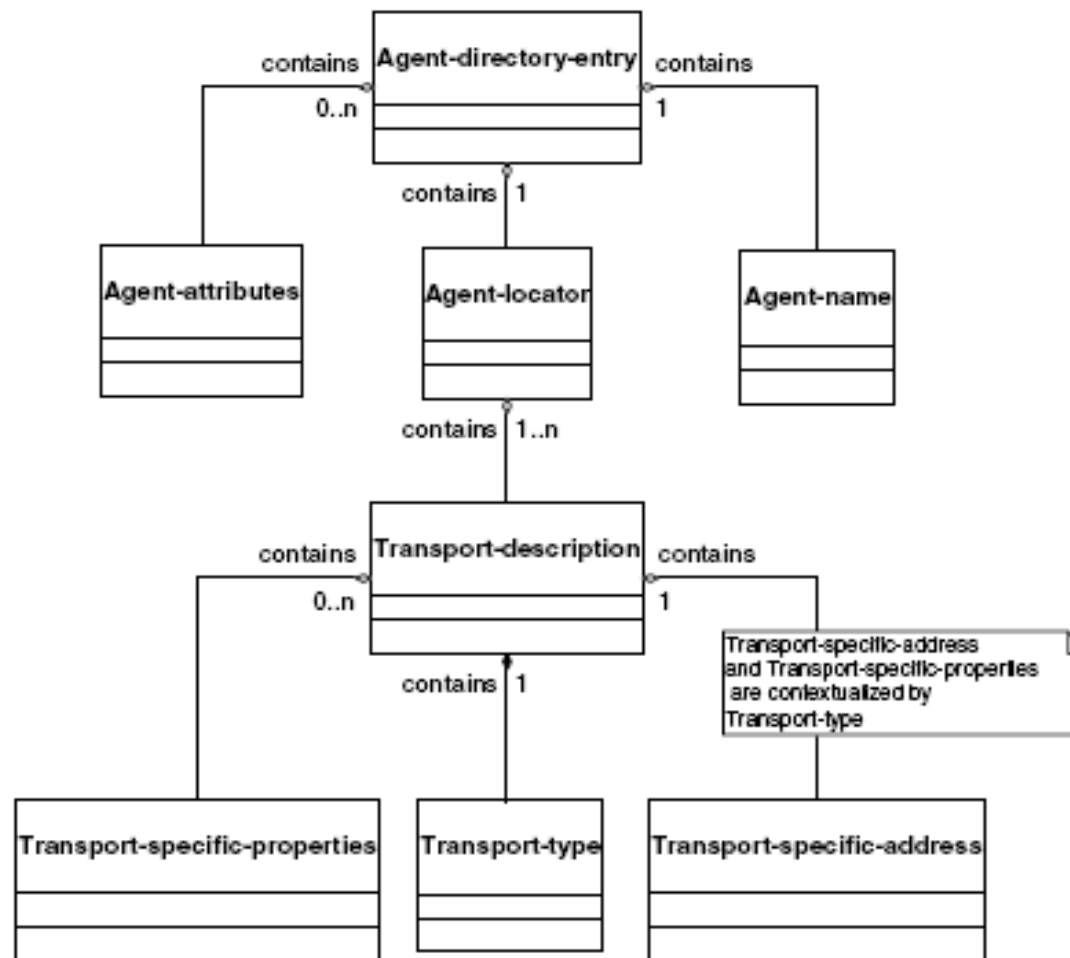
Ha un agent-name, agent-attributes e può inviare un transport-message agli altri agenti.

Può essere registrato in una o più agent-directory-services attraverso una agent-directory-entry.

E' indirizzabile tramite la transport-descriptions inclusa nella agent-directory-entry.



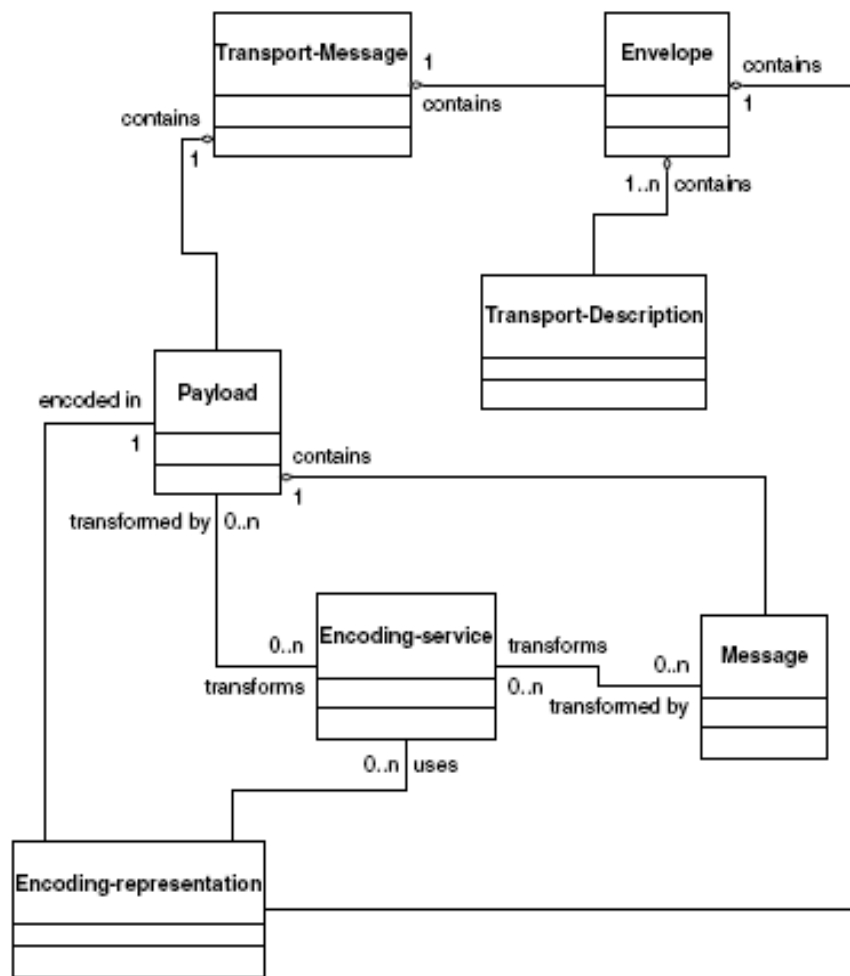
Agent-Directory-Entry





Transport-Message

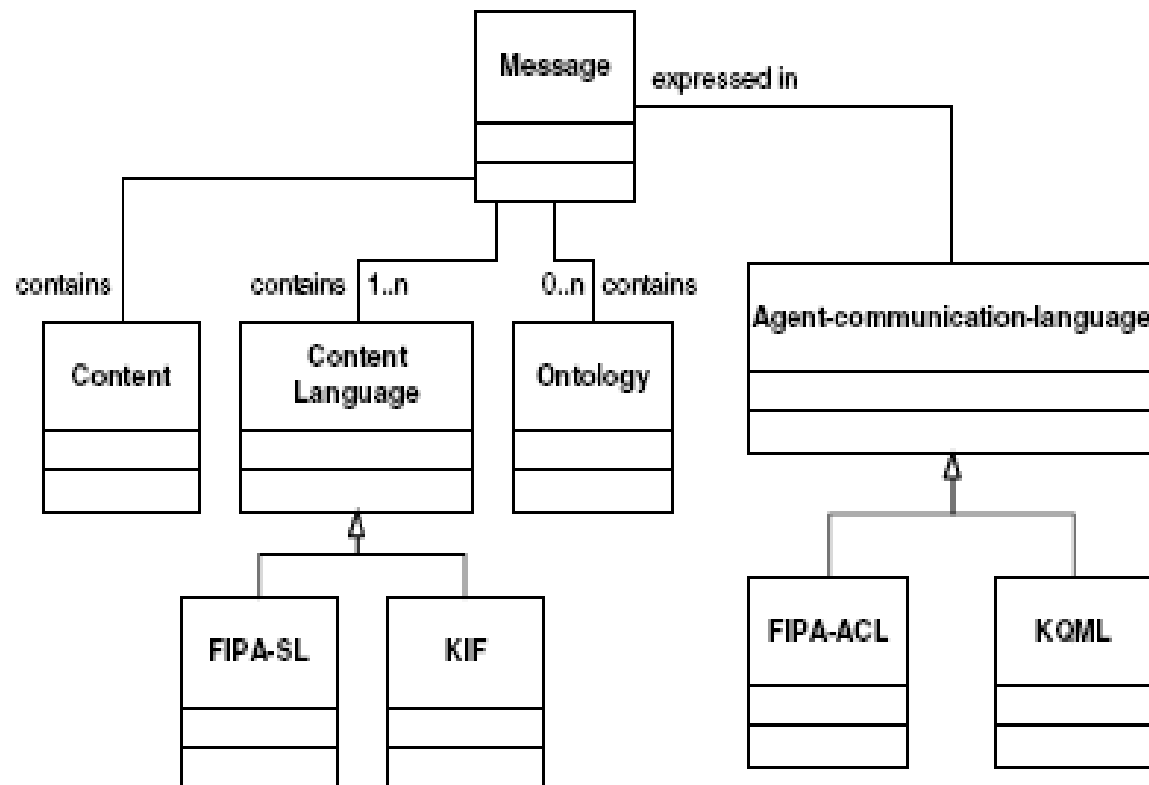
L'oggetto che viene spedito fra gli agenti e che contiene la Transport-description e il Payload che a sua volta contiene il messaggio ACL





Message

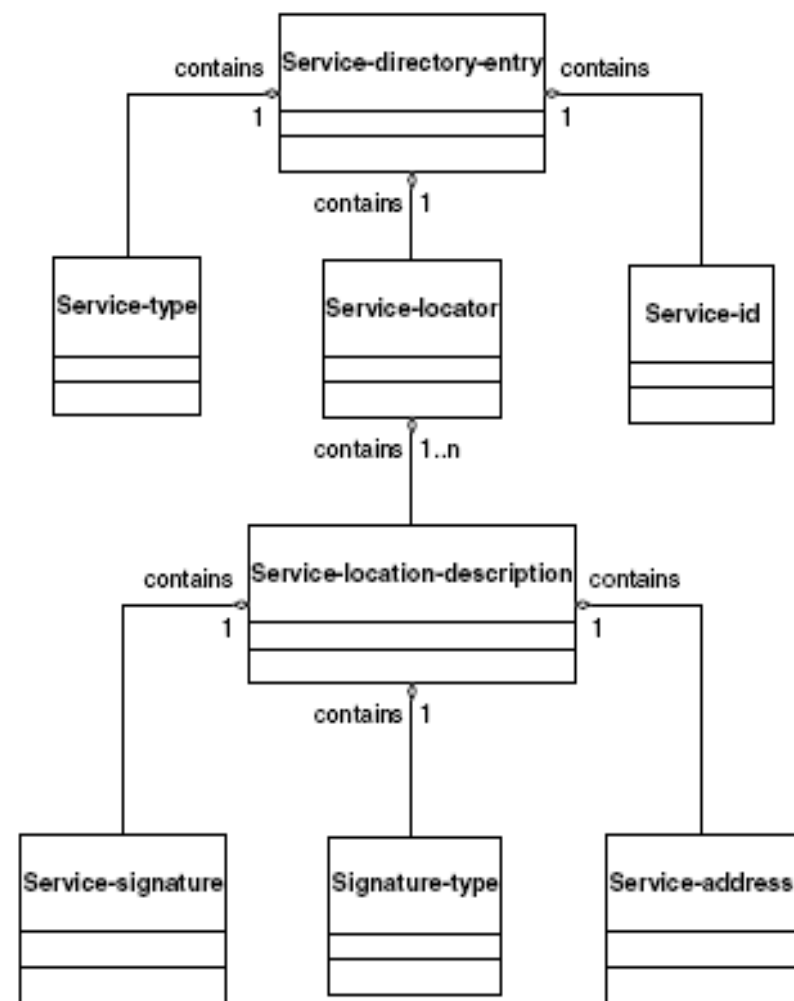
Il Message è contenuto nel payload del Transport-Message.





Services

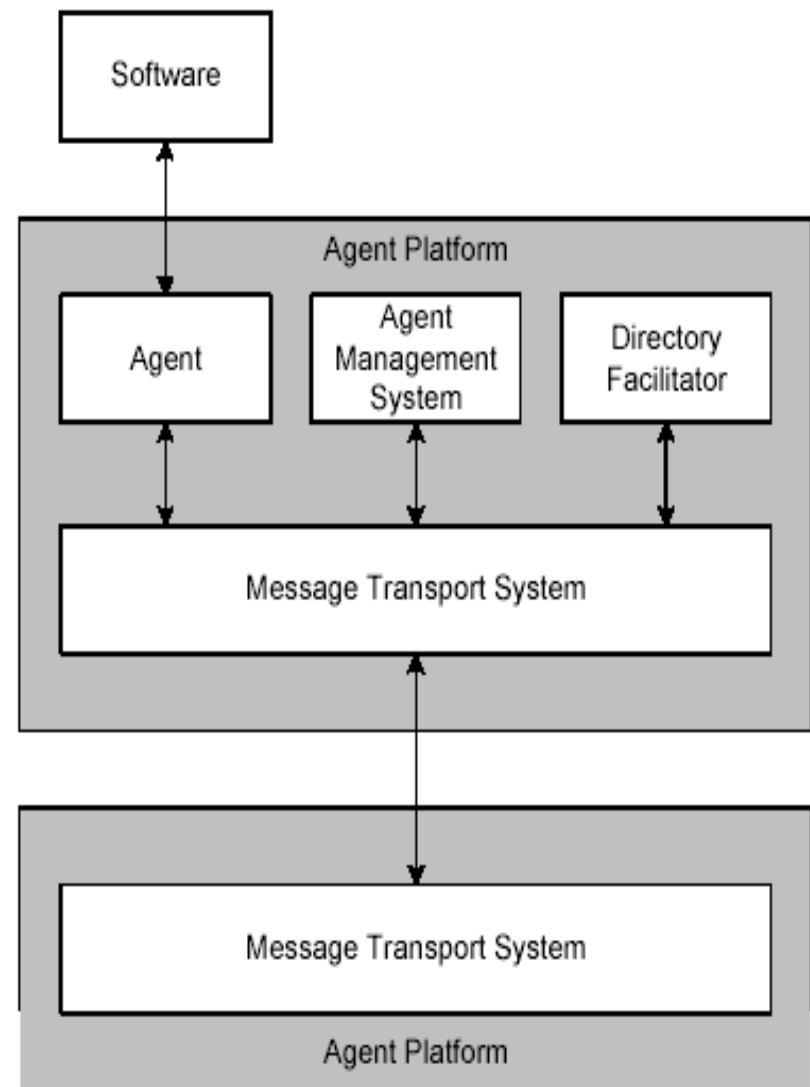
Nella *seivices-directory-service* vengono registrati i servizi dell'infreastruttura FIPA, questi non sono limitati al *message-transport-services*, *agent-directory-services*, *gateway services*, *message e buffering services*, ma può essere usata per descrivere servizi application oriented e quindi riferiti ad una concreta implementazione dell'architettura.





Agent Managment

Le Agent Management specifications discutono il controllo e la gestione degli agenti nella piattaforma o attraverso la piattaforma e descrivono la struttura dell'ambiente nel quale gli agenti FIPA esistono ed operano.



Agent
Communication

Agent
Management

Agent Message
Transport



Agent Managment Components

Un agente è un processo computazionale che comunica con altri agenti usando l'ACL e appoggiandosi su un *message transport system*.

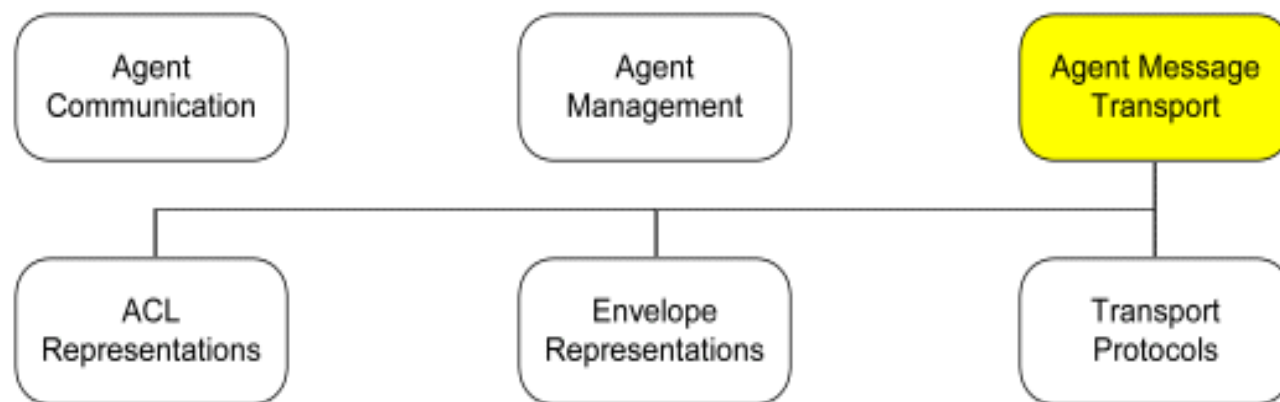
Un *Directory Facilitator (DF)* offre un servizio di di pagine gialle agli agenti. Almeno un *DF* deve esistere su ogni *Agent Platform*.

Un *Agent Management System (AMS)* è un componente che deve essere obbligatoriamente presente nella piattaforma ed è responsabile della creazione e eliminazione di agenti e della migrazione di agenti da e verso altre *AP*.



Agent Message Transport

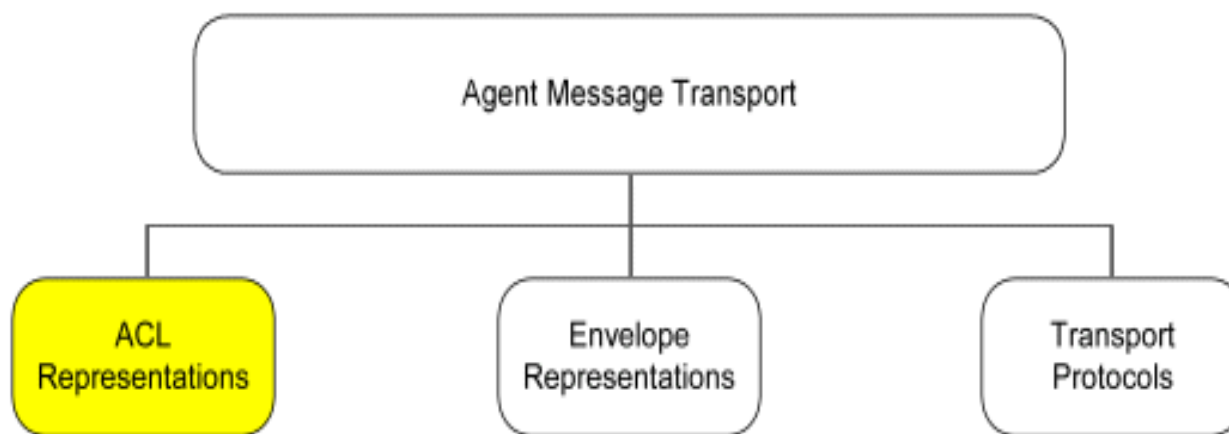
Le *FIPA Agent Message Transport specifications* trattano il trasporto e la rappresentazione dei messaggi attraverso differenti protocolli di trasporto, compresi gli ambienti di rete wireline e wireless.





ACL Representation

Le FIPA ACL Message Representation specifications si occupano delle diverse forme di rappresentazione di un messaggio ACL, come ad esempio la sua rappresentazione in XML, String o Bit.





ACL Representation 2

Il messaggio ACL può essere rappresentato in 3 differenti formati quali:

Bit-Efficient,

XML,

String.

Le regole grammaticali e lessicali possono essere reperite nelle specifiche [FIPA00069], [FIPA00070], [FIPA00071].



FIPA ACL Message Representation in Bit-Efficient Specification

Identifier: 00069

Title: FIPA ACL Message Representation in
Bit-Efficient Specification

Subject: ACL Representations

Summary: A description of an ACL message
representation in a bit-efficient
encoding.

Date: 2002-12-06

Status: **Standard**



Bit-Efficient ACL Representation

Questo documento tratta il trasporto dei messaggi tra gli agenti e fa anche parte della FIPA Agent Management Specification.

Contiene le specifiche per la rappresentazione sintattica di un messaggio ACL in forma di bit-efficient.

La sintassi di un bit-efficient ACL message è espressa nel formato standard EBNF.

EBNF Rules:

Grammar rule component	Example
Terminal tokens are enclosed in double quotes	"("
Non-terminals are written as capitalised identifiers	Expression
Square brackets denote an optional construct	[" , " OptionalArg]
Vertical bars denote an alternative between choices	Integer Float
Asterisk denotes zero or more repetitions of the preceding expression	Digit*
Plus denotes one or more repetitions of the preceding expression	Alpha+
Parentheses are used to group expansions	(A B)*
Productions are written with the non-terminal name on the left-hand side, expansion on the right-hand side and terminated by a full stop	ANonTerminal = "terminal".
0x?? is a hexadecimal byte	0x00



Bit-Efficient ACL Representation

Per la sintassi del messaggio ACL rappresentato in bit occorre fare riferimento al documento della specifica, qui ne è riportata una piccola parte:

ACLCommunicativeAct = Message.

Message = Header MessageType MessageParameter* EndofMsg.

Header = MessageId Version.

MessageId = 0xFA | 0xFB | 0xFC. /* See comment 1 below */

Version = Byte. /* See comment 2 below */

EndofMsg = EndOfCollection.

EndOfCollection = 0x01.

MessageType = PredefinedMsgType | UserDefinedMsgType. /* See comment 3 below */

UserDefinedMsgType = 0x00 MsgTypeName.

MsgTypeName = BinWord.

MessageParameter = PredefinedParam | UserDefinedMsgParam. /* See comment 4 below */

UserDefinedMsgParam = 0x00 ParameterName ParameterValue.

ParameterName = BinWord.

ParameterValue = BinExpression.

PredefinedMsgType = 0x01 /* accept-proposal */

 | 0x02 /* agree */

 | 0x03 /* cancel */

 | 0x04 /* cfp */

 | 0x05 /* confirm */

 | 0x06 /* disconfirm



FIPA ACL Message Representation in String Specification

Identifier: 00070

Title: FIPA ACL Message Representation in
String Specification

Subject: ACL Representations

Summary: A description of an ACL message
representation in a String encoding.

Date: 2002-12-06

Status: **Standard**



String ACL Representation

Contiene le specifiche per la rappresentazione sintattica di un messaggio ACL in forma di stringhe.

Descrive la sintassi di uno String ACL message è espressa nel formato standard EBNF.

Grammar rule component	Example
Terminal tokens are enclosed in double quotes	" ("
Non-terminals are written as capitalised identifiers	Expression
Square brackets denote an optional construct	[", " OptionalArg]
Vertical bars denote an alternative between choices	Integer Float
Asterisk denotes zero or more repetitions of the preceding expression	Digit*
Plus denotes one or more repetitions of the preceding expression	Alpha+
Parentheses are used to group expansions	(A B)*
Productions are written with the non-terminal name on the left-hand side, expansion on the right-hand side and terminated by a full stop	ANonTerminal = "terminal".

Per approfondimenti sulla grammatica, i Token, la sintassi occorre fare riferimento al documento.



Syntax (example)

```
CLCommunicativeAct = Message.  
Message = "(" MessageType  
MessageParameter* ")".  
MessageType = See [FIPA00037]  
MessageParameter = ":sender"  
AgentIdentifier  
| ":receiver" AgentIdentifierSet  
| ":content" String  
| ":reply-with" Expression  
| ":reply-by" DateTime  
| ":in-reply-to" Expression  
| ":reply-to" AgentIdentifierSet  
| ":language" Expression  
| ":encoding" Expression  
| ":ontology" Expression  
| ":protocol" Word  
| ":conversation-id" Expression  
| UserDefinedParameter Expression.  
sion.
```

```
UserDefinedParameter = Word .  
Expression = Word  
| String  
| Number  
| DateTime  
| "(" Expression* ")".  
AgentIdentifier = "(" "agent-identifier"  
":name" word  
[ ":addresses" URLSequence ]  
[ ":resolvers"  
AgentIdentifierSequence ]  
( UserDefinedParameter  
Expression )* ")".  
AgentIdentifierSequence = "("  
"sequence" AgentIdentifier* ")".  
AgentIdentifierSet = "(" "set"  
AgentIdentifier* ")".  
URLSequence = "(" "sequence" URL*  
")".  
DateTime = DateTimeToken.  
URL = See [RFC2396]
```



FIPA ACL Message Representation in XML Specification

Identifier 00071

Title FIPA ACL Message Representation in
XML Specification

Subject ACL Representations

Summary A description of an ACL message
representation in an XML encoding.

Date 2002-12-06

Status **Standard**



XML ACL Representation

Questo documento definisce la sintassi dei messaggi con rappresentazione XML di un ACL message.

Esempio parziale

della sintassi :

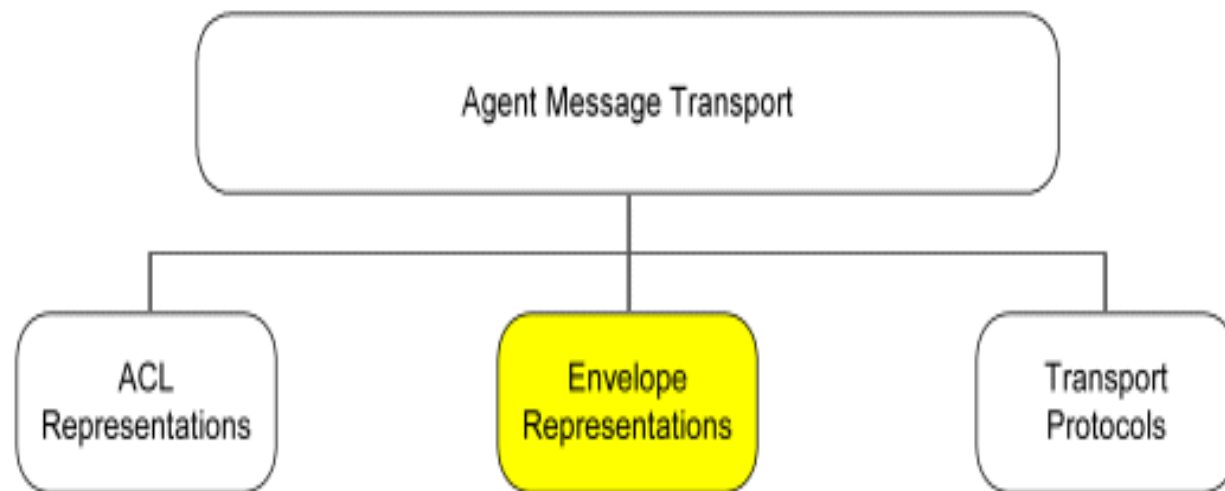
```
<!-- Document Type: XML DTD
Document Purpose: Encoding of FIPA ACL messages in XML
(see [FIPA00067]) and http://www.fipa.org/)
Last Revised: 2002/05/10 -->

<!-- Possible FIPA Communicative Acts. See [FIPA00037] for a
full list of valid performatives. -->
<!ENTITY    %communicative-acts      "accept-proposal
| agree
| cancel
| cfp
| confirm
| disconfirm
| failure
| inform
| not-understood
| propose
| query-if
| query-ref
| refuse
| reject-proposal
| request
| request-when
| request-whenever
| subscribe
| inform-if
| inform-ref
| proxy
| propagate">
```



Envelope Representation

Le FIPA Envelope Representation Specifications descrivono le differenti forme di rappresentazione di una busta del messaggio ACL.





Envelope Representation in XML

La rappresentazione dell'*envelope* di un messaggio in XML segue le specifiche dettate dal *World Wide Web Consortium (W3C)*.

Il set di caratteri che viene usato deve essere conforme al [W3Cxml].

La sintassi è consultabile presso [FIPA00085].

La rappresentazione del tempo sfrutta i Tokens che sono basati su [ISO8601].



Envelope Representation in Bit-efficient

La sintassi dell'*envelope* è espressa nel formato standard EBNF.

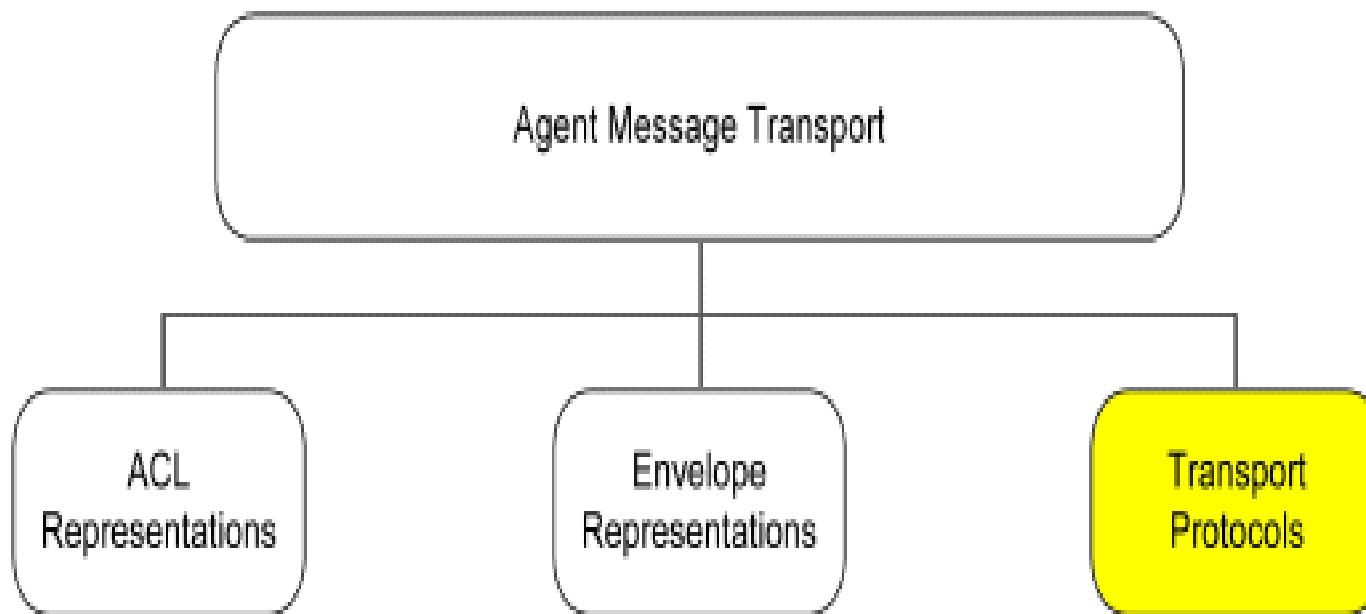
In accordo a [FIPA00067], a un agente non è permesso modificare alcun elemento dell'*envelope* che riceve.

Grammar rule component	Example
Terminal tokens are enclosed in double quotes	" ("
Non-terminals are written as capitalised identifiers	Expression
Square brackets denote an optional construct	[", " OptionalArg]
Vertical bars denote an alternative between choices	Integer Float
Asterisk denotes zero or more repetitions of the preceding expression	Digit*
Plus denotes one or more repetitions of the preceding expression	Alpha+
Parentheses are used to group expansions	(A B)*
Productions are written with the non-terminal name on the left-hand side, expansion on the right-hand side and terminated by a full stop	ANonTerminal = "terminal".
0x?? is a hexadecimal byte	0x00



Transport Protocol

Le FIPA Agent Message Transport Protocol (MTP) specifications trattano i differenti protocolli di trasporto per i messaggi ACL.





Transport Protocol 2

Attualmente esistono specifiche per tre diversi tipi di Message Transport Protocol (MTP): WAP, HTTP e IIOP.

Il MTP basato su WAP (Wireless Application Protocol) rappresenta l'intero messaggio inclusa l'envelope in un messaggio WAP.

HTTP lo rappresenta come una HTTP Request.

L'IIOP (Internet Inter-Orb Protocol) MTP è basato sul trasferimento di una struttura OMG IDL contenente il *message envelope* e una sequenza che rappresenta il corpo del messaggio ACL.

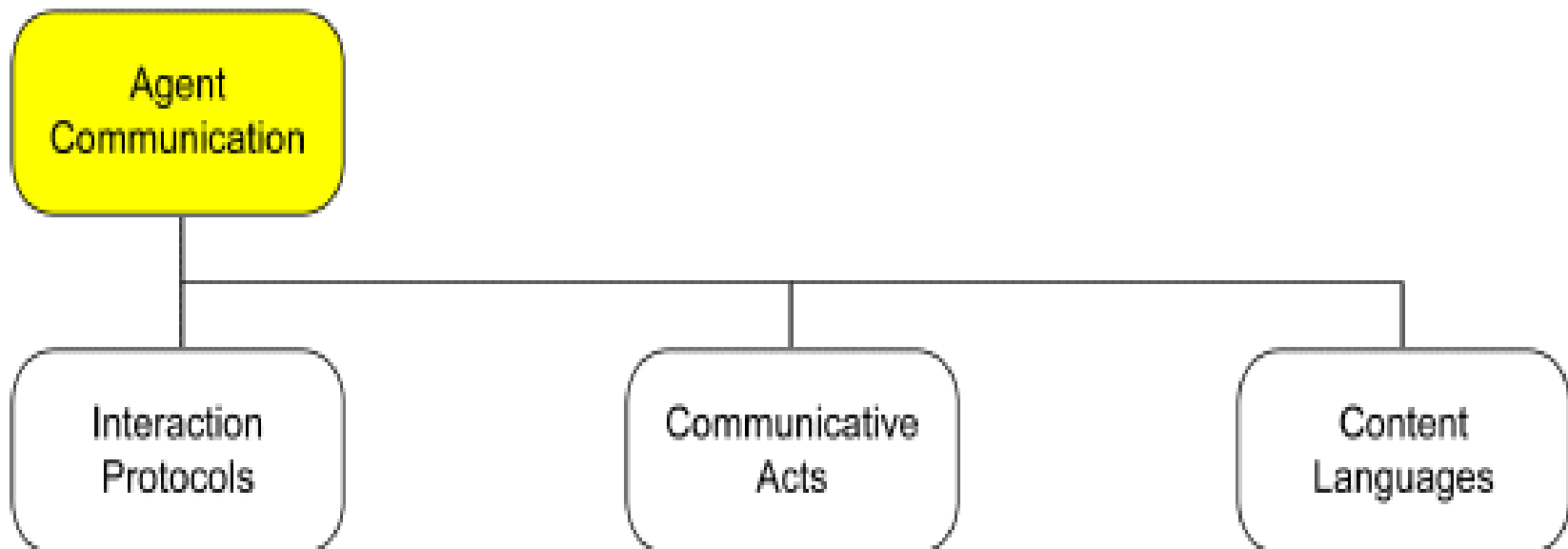


Agent Communication Language Specifications

Le FIPA Agent Communication specifications trattano la standardizzazione dei messaggi ACL, di questo gruppo fanno parte le specifiche:

② FIPA ACL Message Structure specification SC00061 (standard)

② FIPA Ontology Service Specification XC00086





FIPA ACL

Message Structure Specification

Identifier: 00061

Title: FIPA ACL Message Structure
Specification

Subject: ACL

Summary: A description of the structure of FIPA
ACL.

Date: 2002-12-06

Status: **Standard**



Scope

FIPA ACL Message Structure Specification tratta le specifiche dei parametri utilizzati nel messaggio FIPA ACL.

L'obiettivo è quello di standardizzare la forma dei messaggi inviati dagli agenti fornendo una struttura standard per il messaggio ACL in modo da garantire l'interoperabilità.

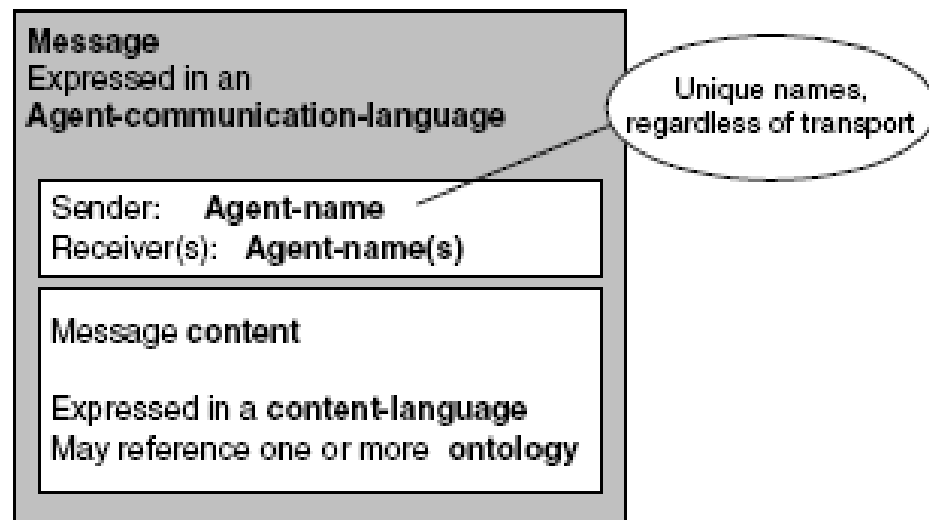


Agent Message from FIPA [SC00001L]

Gli agenti comunicano mediante l'invio di messaggi. Tre aspetti fondamentali del messaggio sono la struttura , la rappresentazione e il trasporto.

La struttura di un messaggio è una tupla scritta nell'Agent Communication Language.

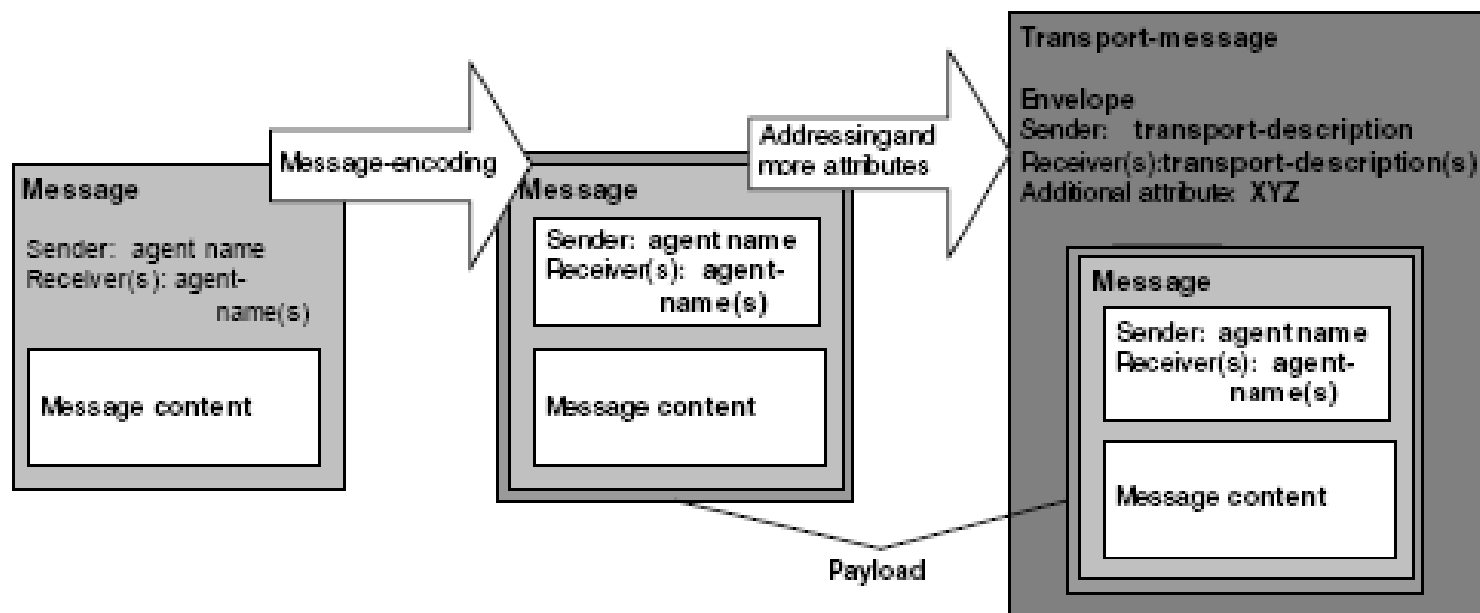
Il contenuto del messaggio è espresso in un content language, come KIF o SL.





Agent Message 2

Quando viene inviato un messaggio è codificato in un payload, e incluso in un transport message. Il payload è codificato utilizzando una codifica adeguata per il trasporto. Ad esempio, se il messaggio sta per essere inviato tramite un mezzo di trasporto con poca risorsa di banda si può codificare in un modo più efficiente.





Agent Message 3

Il transport message è di per sé il payload codificato più la busta. La busta comprende il mittente e il destinatario del messaggio.

La transport description contiene le informazioni su come inviare il messaggio (via quale mezzo di trasporto, a quale indirizzo, i dettagli su come utilizzare il trasporto). La busta può contenere anche ulteriori informazioni, come ad esempio la codifica, i dati relativi alla sicurezza, etc..

Un messaggio può essere codificato in XML, ma il payload che contiene il messaggio deve essere codificato in base al tipo di trasporto che si utilizza.

Il payload non aggiunge nulla al messaggio, ma codifica solo in un'altra rappresentazione.



Message Structure 1

Un messaggio FIPA ACL contiene un set di uno o più message parameters.

La specifica indica quali parametri sono necessari per la comunicazione tra agenti a seconda della situazione.

L'unico parametro che è obbligatorio in tutti i messaggi ACL è *performative*, anche se si prevede che i messaggi ACL conterranno probabilmente anche i parametri *sender*, *receiver* e *content*.

Se un agente non riconosce o non è in grado di elaborare uno o più dei parametri può rispondere comunque con un *not-understood message*.



Message Structure 2

La specifica permette di definire propri *message parameters* oltre quelli compresi nella specifica.

In questo caso il nome di questi parametri deve iniziare con la stringa “X-”.

Il set completo dei parametri dei messaggi FIPA ACL sono descritti nel documento, ma senza riguardo per la codifica dell'implementazione.

Le codifiche approvate da FIPA sono espresse nelle specifiche FIPA ACL Message Representation che contengono precise descrizioni della sintassi usando XML, stringhe, descritte precedentemente.



FIPA ACL Message Parameters

Parameter	Category of Parameters
performative	Type of communicative acts
sender	Participant in communication
receiver	Participant in communication
reply-to	Participant in communication
content	Content of message
language	Description of Content
encoding	Description of Content
ontology	Description of Content
protocol	Control of conversation
conversation-id	Control of conversation
reply-with	Control of conversation
in-reply-to	Control of conversation
reply-by	Control of conversation



Type of Communicative Act

performative

Denota il tipo di atto comunicativo del messaggio ACL.

Il significato è descritto nelle Communicative Act (CAs) specifications descritte più avanti.



Participants in Communication 1

sender

Denota l'identità del mittente del messaggio, quindi il nome dell'agente .

Sarà un parametro compreso in quasi tutti i messaggi ACL.

E' possibile ometterlo se per esempio l'agente che invia il messaggio vuole rimanere anonimo.



Participants in Communication 2

receiver

Denota l'identità del ricevente.

Di solito presente in tutti i messaggi ACL.

E' possibile ometterlo se il ricevente può essere ricavato dal contesto o in altri casi particolari.

Come destinatario si può avere il nome di un solo agente ma anche di più agenti nel caso di un messaggio multicast.



Participants in Communication 3

reply-to

Questo parametro indica che i messaggi successivi della conversazione devono essere indirizzati all'agente indicato in *reply-to* invece che all'agente presente nel parametro mittente.



Content of Message

content

Denota il contenuto del messaggio.

Il significato del contenuto di qualsiasi messaggio ACL è destinato ad essere interpretato dal destinatario del messaggio.



Description of Content 1

language

Denota il linguaggio in cui è espresso il contenuto (content parameter).

Il linguaggio è descritto nelle specifiche FIPA

Può essere omesso se chi riceve il messaggio si può assumere che conosca tale linguaggio.

Più avanti sono descritti i *Content Languages*



Description of Content 2

encoding

Denota la specifica codifica del linguaggio.

Descritto nella specifica [FIPA00007]

L'espressione dei contenuti può essere codificata in modi diversi. Il parametro encoding è facoltativamente utilizzato per specificare questa codifica per l'agente destinatario.

Se il parametro di codifica non è presente sarà comunque specificato nella busta che racchiude il messaggio ACL.



Description of Content 3

ontology

Denota l'ontologia , utilizzata per dare un significato ai simboli della content expression.

Il parametro ontology è usato in congiunzione con il parametro language per l'interpretazione della content expression da parte dell'agente ricevente. In molte situazioni, il parametro ontologia sarà comunque compreso dalla comunità di agenti del sistema e quindi può essere omesso.



Control of Conversation 1

protocol

Indica il protocollo di interazione che è utilizzato nel messaggio ACL.

Descritto nello standard [FIPA00025]



Control of Conversation 2

conversation-id

Viene utilizzato per identificare la sequenza continua di atti comunicativi, che insieme formano una conversazione.

Permette ad un agente di identificare i colloqui individuali con agenti multipli.



Control of Conversation 3

reply-with

Introduce un'espressione che sarà utilizzata da parte dell'agente che deve rispondere per identificare il messaggio.

Il parametro è stato progettato per essere usato per seguire un thread di conversazione nel caso di dialoghi multipli. Se l'agente invia un messaggio con : “reply-con <expr>” l'agente risponderà con un messaggio contenente : “In-reply-to <expr>”.



Control of Conversation 4

in-reply-to

Denota un'espressione che fa riferimento a un precedente intervento a cui il messaggio è una risposta.

reply-by

Denota una espressione di tempo e / o di data che indica il tempo entro il quale l'agente che ha inviato il messaggio desidera ricevere una risposta.



Message Example

Example: Agent i did not understand a query-if message because it did not recognize the ontology.

```
(not-understood
  :sender (agent-identifier :name i)
  :receiver (set (agent-identifier :name j))
  :content
    "((action (agent-identifier :name j)
      (query-if
        :sender (agent-identifier :name j)
        :receiver (set (agent-identifier :name i))
        :content
          \"<fipa-ccl content expression>\"
        :ontology www
        :language fipa-ccl))
      (unknown (ontology \"www\"))))"
  :language fipa-sl)
```



FIPA ACL

Ontology Service Specification

Identifier: 00086

Title: FIPA Ontology Service Specification

Subject: ACL

Summary: The description of an Ontology Agent or registering and serving ontologies to agents

Date: 2001-08-15

Status: **Experimental**



Scope

Il modello di comunicazione FIPA si basa sul presupposto che due agenti, che vogliono dialogare, devono condividere non solo il linguaggio e protocolli di comunicazione, ma anche un comune vocabolario di termini utilizzati nei messaggi scambiati; una ontologia comune per il dominio del discorso.

Ontology Service Specification assicura che gli agenti attribuiscano lo stesso significato per gli elementi utilizzati nel messaggio.

E' previsto un ontology service che viene fornito da un agente dedicato, chiamato Ontology Agent (OA).



Ontology Agent

Il ruolo dell'Ontology Agent nella comunità è quello di fornire alcuni o tutti i seguenti servizi:

- scoperta di ontologie pubbliche, al fine di accedervi,
- tradurre le espressioni tra diverse ontologie e / o diversi linguaggi,
- rispondere alle query su relazioni tra due termini o tra ontologie,
- facilitare l'individuazione di una ontologia comune nella comunicazione tra due agenti.



Ontology Agent

La specifica tratta solo l'interfaccia del servizio mentre l'attuazione e funzionalità sono lasciate agli sviluppatori.

Ogni OA non deve essere in grado di eseguire tutti i compiti ma ogni OA deve essere in grado di partecipare in una comunicazione relativa a tali mansioni.

La specifica definisce il protocollo di interazione, gli atti comunicativi e, in generale, il vocabolario che gli agenti devono adottare quando usano questo servizio.



Ontology Service Specification

Al fine di mantenere l'applicabilità della specifica più libera possibile è stato utilizzato un approccio platform independent.

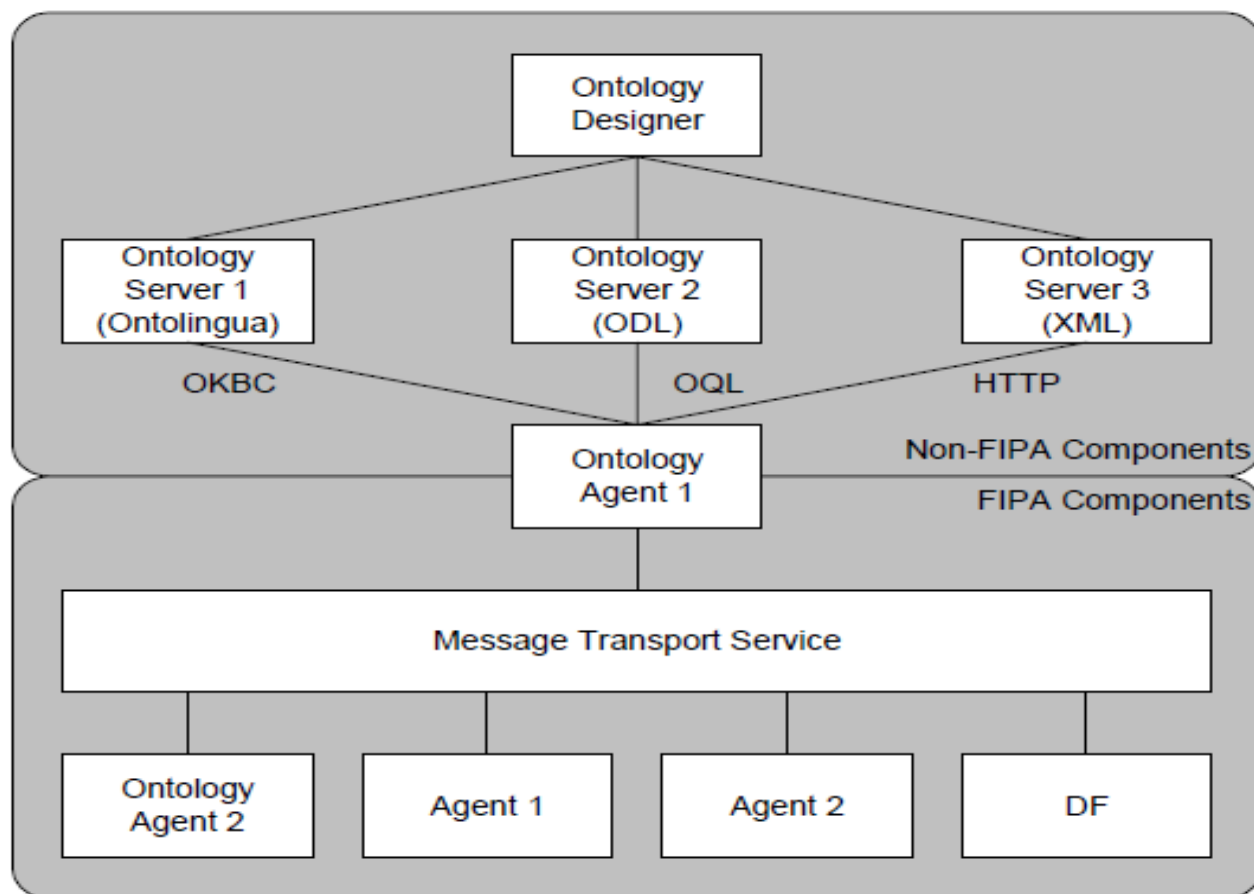
Le specifiche non indicano il formato di memorizzazione delle ontologie, ma solo il modo in cui gli agenti accedono al servizio.

Tuttavia, al fine di specificare il servizio, un certo formalismo è stato specificato (FIPA-Meta-Ontology).



Ontology Service Reference Model

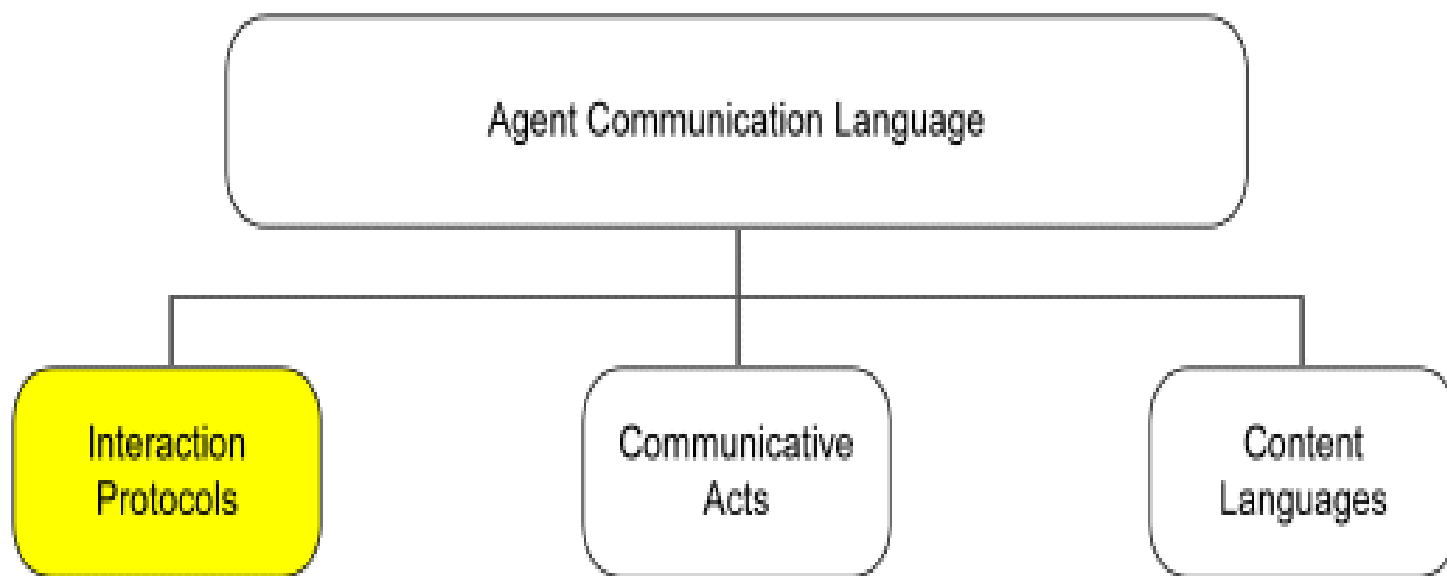
Le ontologie sono memorizzate in uno o più server ontology. L'OA consente agli agenti di accedere ai propri servizi in una maniera unica.





Interaction Protocols Specifications

FIPA Interaction Protocols (IPs) specifications trattano i protocolli per lo scambio di messaggi (gli ACL messages).





Interaction Protocols Specifications

- ② FIPA Request Interaction Protocol Specification SC00026
- ② FIPA Query Interaction Protocol Specification SC00027
- ② FIPA Request When Interaction Protocol Specification SC00028
- ② FIPA Contract Net Interaction Protocol Specification SC00029
- ② FIPA Iterated Contract Net Interaction Protocol Specification SC00030
- ② FIPA English Auction Interaction Protocol Specification XC00031
- ② FIPA Dutch Auction Interaction Protocol Specification XC00032
- ② FIPA Brokering Interaction Protocol Specification SC00033
- ② FIPA Recruiting Interaction Protocol Specification SC00034
- ② FIPA Subscribe Interaction Protocol Specification SC00035
- ② FIPA Propose Interaction Protocol Specification SC00036



Protocolli di interazione

Nella comunicazione tra agenti vengono usati diversi *protocolli* tra cui abbiamo:

Protocollo di richiesta

Protocollo di consultazione

Protocollo di richiesta “Quando”

Protocollo di contract net

Protocollo di interazione di contract net

Protocollo inglese di vendita all'asta

Protocollo olandese di vendita all'asta

Protocollo di mediazione

Protocollo di reclutamento

Protocollo di sottoscrizione

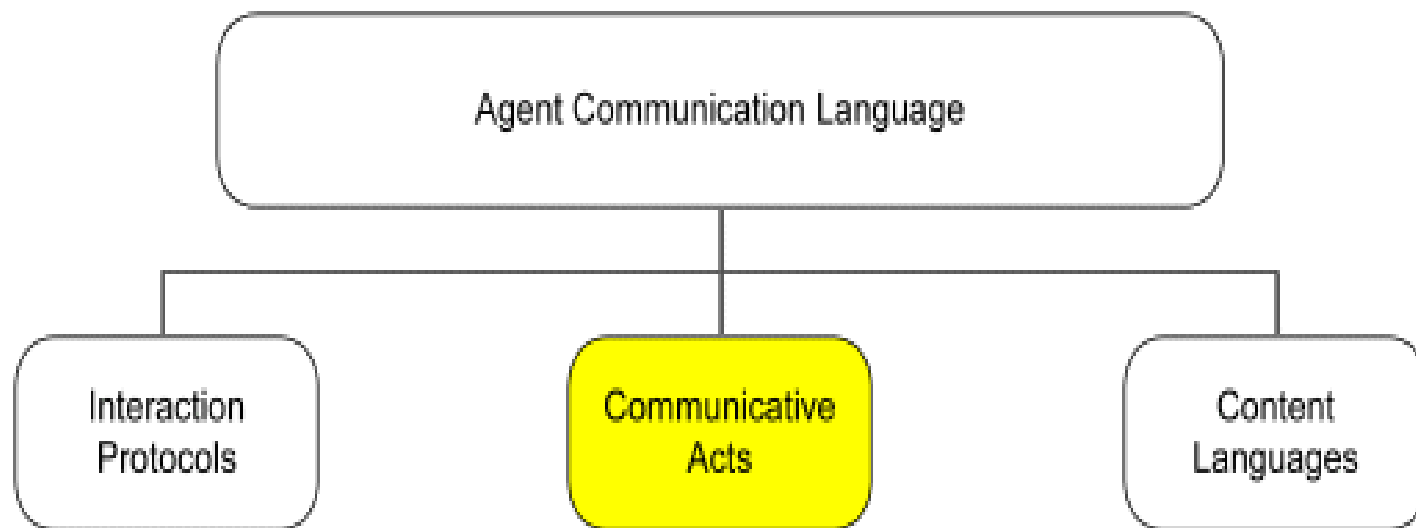
Protocollo di proposta



Commutative Acts Specifications

FIPA Communicative Act (CAs) specifications tratta alcuni enunciati riguardanti gli ACL messages.

 FIPA Communicative Act Library Specification SC00037 (standard)





FIPA Communicative Act Library Specification

Identifier: 00037

Title: FIPA Communicative Act Library
Specification

Subject: Communicative Acts

Summary: A library of FIPA communicative acts
and requirements for new
communicative acts.

Date: 2002-12-06

Status: **Standard**



Communicative Act Library

Il documento FIPA Communicative Act Library (CAL) contiene le specifiche per definire la struttura del CAL.

Gli obiettivi della standardizzazione e definizione di una libreria di atti comunicativi sono garantire l'interoperabilità, fornendo un set standard di atti comunicativi e facilitare il riutilizzo di materiale

Gli agenti conformi ACL non sono tenuti ad implementare tutte le specifiche CAL, tranne il *not-understood act*.

E' possibile definire i propri *Communicative Act*.



FIPA Communicative Acts 1

request

Il mittente invia al ricevente la richiesta di eseguire un'azione. Un importante uso del *request act* è di chiedere al ricevente di effettuare un altro *communicative act*.

Example: Agent i requests j to open a file.

(request

 :sender (agent-identifier :name i)

 :receiver (set (agent-identifier :name j))

 :content

 "open \"db.txt\" for input"

:language vb)



FIPA Communicative Acts 2

Not-understood

Chi lo invia informa chi lo riceve che ha in precedenza ricevuto un communication act ma non ha capito cosa deve fare

Example: Agent i did not understand a query-if message because it did not recognize the ontology.

```
(not-understood
  :sender (agent-identifier :name i)
  :receiver (set (agent-identifier :name j))
  :content
    ((action (agent-identifier :name j)
      (query-if
        :sender (agent-identifier :name j)
        :receiver (set (agent-identifier :name i))
        :content
          \"<fipa-ccl content expression>\"
        :ontology www
        :language fipa-ccl))
      (unknown (ontology \"www\"))))
  :language fipa-sl)
```



FIPA Communicative Acts 3

Altri Communicative Acts:

Accept-Proposal

Agree

Cancel

Call-for-Proposal

Confirm

Disconfirm

Failure

Inform

Inform-If

Inform-Ref

Not-Understood

Propagate

Propose

Proxy

Query-Ref

Refuse

Request

Request-When

Request-Whenever

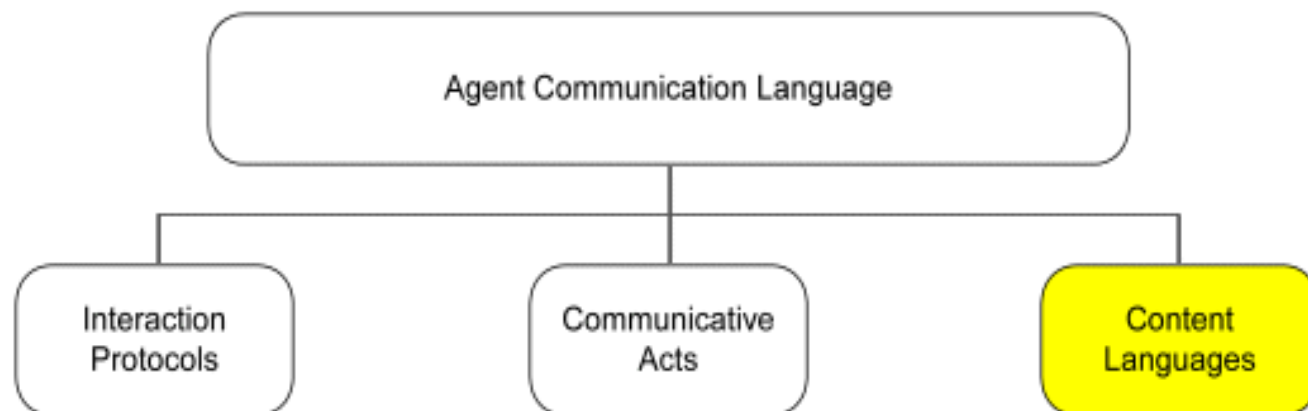
Subscribe



Content Language Specifications

FIPA Content Language (CL) Specifications discute i differenti linguaggi con cui è possibile rappresentare il contenuto del messaggio ACL.

- 🌀 FIPA SL Content Language Specification SC00008 (standard)
- 🌀 FIPA CCL Content Language Specification XC00009
- 🌀 FIPA KIF Content Language Specification XC00010
- 🌀 FIPA RDF Content Language Specification XC00011





FIPA SL Content Language Specification

Identifier: 00008

Title: FIPA SL Content Language Specification

Subject: Content Language

Summary: A description of the FIPA Semantic Language content language.

Date: 2002-12-06

Status: **Standard**



Scope

Questa specifica definisce una sintassi concreta per il FIPA *Semantic Language* (SL), il linguaggio dei contenuti.

Questa sintassi e la sua semantica associati sono proposti come *content language* nei messaggi ACL FIPA.

In particolare, la sintassi è definita come un sub-grammatica della s-expression syntax.

S-expression: convenzione per la rappresentazione di dati semi-strutturati in forma testuale.



SL Content Language

Una espressione del linguaggio FIPA SL può essere usata come il contenuto di un messaggio ACL. Si distinguono tre casi:

- ① Una proposizione a cui può essere assegnato un valore di verità in un determinato contesto. Più precisamente, è una formula well-formed (Wff) che usa le regole descritte nella produzione di Wff.
- ② Un'azione che può essere compiuta; un'azione è usata come una *content expression* quando l'atto è **request**.
- ③ Una *identifying reference expression* (IRE) che identifica un oggetto nel dominio. Questo è il Referential operator ed è usato nell' *inform-ref* act ed in altri che ne derivano.

Altre valide expression content possono essere il risultato della composizione di questi casi.



Grammar FIPA SL Concrete Syntax

Esempio parziale della grammatica:

```
Content                = "(" ContentExpression+ ")".

ContentExpression      = IdentifyingExpression
                        | ActionExpression
                        | Proposition.

Proposition             = Wff.

Wff                     = AtomicFormula
                        | "(" UnaryLogicalOp  Wff ")"
                        | "(" BinaryLogicalOp Wff Wff ")"
                        | "(" Quantifier      Variable Wff ")"
                        | "(" ModalOp        Agent Wff ")"
                        | "(" ActionOp       ActionExpression ")"
                        | "(" ActionOp       ActionExpression Wff ")"

UnaryLogicalOp          = "not".

BinaryLogicalOp         = "and"
                        | "or"
                        | "implies"
                        | "equiv".

AtomicFormula           = PropositionSymbol
                        | "(" BinaryTermOp    TermOrIE TermOrIE ")"
                        | "(" PredicateSymbol TermOrIE+ ")"
                        | "true"
                        | "false".

BinaryTermOp            = "="
                        | "result".

Quantifier              = "forall"
                        | "exists".

ModalOp                 = "B"
                        | "U"
                        | "PG"
                        | "I".

ActionOp                = "feasible"
                        | "done".
```



Lexical Definitions

Esempio parziale:

```
String          = Word
                  | ByteLengthEncodedString
                  | StringLiteral.

ByteLengthEncodedString = "#" DecimalLiteral+ "\" <byte sequence>.

Word            = [~ "\0x00" - "\0x20", "(", ")", "*", "#", "0" - "9", ":", "-", "?"]
                  [~ "\0x00" - "\0x20", "(", ")", "*"].

ParameterName   = ":" String.

VariableIdentifier = "?" String.

Sign            = [ "+", "-" ].

Integer         = Sign? DecimalLiteral+
                  | Sign? "0" ["x", "X"] HexLiteral+.

Dot             = "."

Float           = Sign? FloatMantissa FloatExponent?
                  | Sign? DecimalLiteral+ FloatExponent.

FloatMantissa   = DecimalLiteral+ Dot DecimalLiteral*
                  | DecimalLiteral* Dot DecimalLiteral+.

FloatExponent   = Exponent Sign? DecimalLiteral+.

Exponent        = ["e", "E"].

DecimalLiteral  = ["0" - "9"].

HexLiteral      = ["0" - "9", "A" - "F", "a" - "f"].
```



Grammar Elements

- ② **Well-Formed Formulas:** (not <Wff>), (and <Wff0> <Wff1>), (exists <variable> <Wff>).
- ② **Atomic Formula:** true, false, (= Term1 Term2).
- ② **Terms.**
- ② **Referential Operators:** (Iota, Any, All).
- ② **Functional Terms.**
- ② **Result.**
- ② **Predicate.**

ACL message

example :

```
(query-ref
  :sender (agent-identifier :name B)
  :receiver (set (agent-identifier :name A))
  :content
    "((any (sequence ?x ?y) (q ?x ?y)))"
  :language fipa-sl
  :reply-with query1)

(inform
  :sender (agent-identifier :name A)
  :receiver (set (agent-identifier :name B))
  :content
    "((= (any (sequence ?x ?y) (q ?x ?y)) (sequence 1 a)))"
  :language fipa-sl
  :in-reply-to query1)
```



FIPA RDF Content Language Specification

La struttura **RDF** (*Resource Description Framework*) è basata sul modello ER(entità-relazione), la sua peculiarità risiede nella sua estensibilità, riusabilità e semplicità.

Abbiamo il modello Data RDF che è descritto attraverso risorse, proprietà e rispettivi valori e lo schema RDF che definisce concetti come classi, sottoclassi, proprietà o sottoproprietà, e permette anche l'espressione di vincoli.

Entrambi (Data e Schema RDF) propongono XML come sintassi per la codifica.

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:s="http://description.org/schema/">

  <rdf:Description ID="TCP/IP Illustrated">
    <s:author>W. Richard Stevens</s:author>
  </rdf:Description>
</rdf:RDF>
```



Figure 1: A Proposition as an RDF Statement



FIPA KIF Content Language Specification

KIF (Knowledge Interchange Format) è un formato per lo scambio di conoscenze tra sistemi differenti.

Chi riceve una KB in FIPA KIF la converte per poterla elaborare nel suo formato e dopo la riconverte nel formato KIF per poterla condividere con altri .

Le seguenti caratteristiche sono essenziali per utilizzare FIPA KIF:

Il linguaggio deve avere una semantica dichiarativa.

Il linguaggio deve essere logicamente comprensibile.

Il linguaggio deve dare la possibilità all'utente di prendere esplicite decisioni e permette inoltre all'utente di introdurre nuovi costrutti.

```
:language FIPA-KIF
:ontology ec-ontology
:content
  (<= (specialist agent1 '(price ,?x ,?y))
      (constant ?x)
      (number ?y)))
```



FIPA CCL Content Language Specification

FIPA CCL è basato sulla rappresentazione di problemi di scelta quali i CSPs (Constraints Satisfaction Problems).

Il linguaggio CCL è utilizzato per:

- ① modellare il problema,
- ② raccogliere le informazioni,
- ③ fondere le informazioni,
- ④ Risolvere il problema.



Bibliografia

FIPA (Foundation for Intelligent Physical Agents)

<http://www.fipa.org>



Tutte le specifiche FIPA e in particolare:

-  [FIPA Abstract Architecture Specification SC00001](#)
-  [FIPA ACL Message Representation in Bit-Efficient Specification SC00069](#)
-  [FIPA ACL Message Representation in String Specification SC00070](#)
-  [FIPA ACL Message Representation in XML Specification SC00071](#)
-  [FIPA ACL Message Structure specification SC00061 \(standard\)](#)
-  [FIPA Ontology Service Specification XC00086](#)
-  [FIPA Communicative Act Library Specification SC00037 \(standard\)](#)
-  [FIPA SL Content Language Specification SC00008 \(standard\)](#)
-  [FIPA CCL Content Language Specification XC00009](#)
-  [FIPA KIF Content Language Specification XC00010](#)
-  [FIPA RDF Content Language Specification XC00011](#)