AOT LAB

Agent and **O**bject **T**echnology **Lab**
Dipartimento di Ingegneria dell’Informazione
Università degli Studi di Parma

Multi-Agent Systems

Communication

Prof. Agostino Poggi

AOT LAB

Methods of Agent Interaction

- Agent can coordinate its actions with the actions of the other agents without explicit communication
  - Blackboard systems are a mean to realize it
    - Shared memory among agents
    - Agents read and write information
- However, often explicit communication simplifies the interaction
  - Messages exchange
    - Need a common language
    - Allow complex conversation based on sequences of messages

- Most treatments of communication in multi-agent systems borrow their inspiration from speech act theory
- Speech act theories are pragmatic theories of language
  - they attempt to account for how language is used by people every day to achieve their goals and intentions
- The origins of speech act theories are usually traced to J.L. Austin's book "How to do things with words" (1962)
  - This is a philosophy book with a linguistic approach
  - Not a computer science book!

- Austin noticed that some utterances are rather like "physical actions", that appear to change the state of the world
- Paradigm examples would be:
  - Declaring war
  - "I now pronounce you man and wife"
- But more generally, everything we utter is uttered with the intention of satisfying some goal or intention
- A theory of how utterances are used to achieve intentions is a speech act theory

Searle Speech Acts Types (1969)

- Representatives
  - Such as informing, e.g., “It is raining”
- Directives
  - Attempts to get the hearer to do something, e.g., “please make the tea”
- Commisives
  - Which commit the speaker to doing something, e.g. “I promise to”
- Expressives
  - Whereby a speaker expresses a mental state, e.g. “Thank you!”
- Declarations
  - Such as declaring a war

Speech Acts Components

- In general, a speech act can be seen to have two components
  - A performative verb identifying the type of the speech act
    - E.g., request, inform, promise, …
  - Propositional content identifying the content of the speech act
    - E.g., ‘the door is closed’

AOT LAB

Speech Acts Components

- Speech act = “please close the door”
  - Performative = request
  - Content = “the door is closed”
- Speech act = “the door is closed!”
  - Performative = inform
  - Content = “the door is closed”
- Speech act = “is the door closed?”
  - Performative = inquire
  - Content = “the door is closed”

Semantics of speech acts

- How can one define the semantics of a speech act?
- What is going to be the effect of the speech act to the world and more exactly to the receiver?
  - The speaker can not (generally) force the hearer to accept some desired mental state
- Different formalisms were proposed, depending on the representation of the world

Plan based semantics

- Cohen & Perrault (1979) defined the semantics of speech acts using the precondition-delete-add list formalism of planning research
- For example the semantics of the speech act “John request Mary to open the door” is defined by:
  - Preconditions:
    - John believes that the Mary can open the door
      - John does not ask Mary to do something unless he think she can do it
    - John believes that Mary believes she can open the door
      - John does not ask Mary unless she believe she can do it
    - John believes he wants to open the door
      - John does not ask Mary unless he want it
  - Postconditions:
    - Mary believes John believes John wants to open the door
      - The effect is to make Mary aware of John’s desire

- Speech acts form messages in an Agent Communication Language (ACL)
- An ACL is a high-level language whose primitives and structures are expressly tailored to support the exchange of messages amongst multiple agents
- An ACL exists in a logical layer on top of existing infrastructures such as TCP/IP, HTTP, or IIOP

ACL Origins

- The first support for agent communication, was developed by the ARPA Knowledge Sharing Effort (KSE)
  - Central goal is to define means for knowledge sharing
  - Knowledge sharing needs common languages for representing knowledge and communication

- KSE defined two languages
  - Knowledge Interchange Format (KIF)
    - Language for expressing message content
  - Knowledge Query and Manipulation Language (KQML)
    - Language for both message formatting and message handling protocols

KIF

- KIF is a first-order logic based language created for expressing properties of a domain
  - Intended to express contents of a message
  - Not the message itself
- Using KIF, it is possible to express
  - Properties of things in a domain
    - E.g., Michael is a vegetarian – Michael has the property of being a vegetarian
  - Relationships between things in a domain
    - E.g., Michael and Janine are married – the relationship of marriage exists between Michael and Janine
  - General properties of a domain
    - E.g., Everybody has a mother

KIF Language Examples

- Relation between 2 objects
  - The temperature of m1 is 83 Celsius

    (= (temperature m1) (scalar 83 Celsius))
- Definition of new concept
  - An object is a bachelor if this object is a man and not married

    (defrelation bachelor (?x) :=

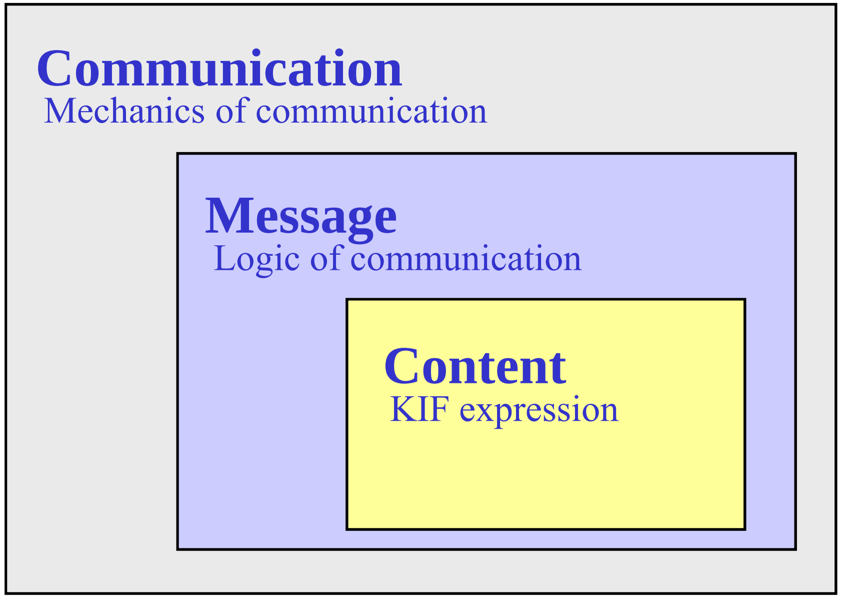
    (and (man ?x) (not (married ?x))))
- Relationship between individuals in the domain
  - A person with the property of being a person also has the property of being a mammal

    (defrelation (person ?x) :=> (mammal ?x))

KQML

- KQML is an “outer language” that defines a set of performatives (communicative acts) used for exchanges of knowledge

- Knowledge is expressed in KIF language



KQML Performatives

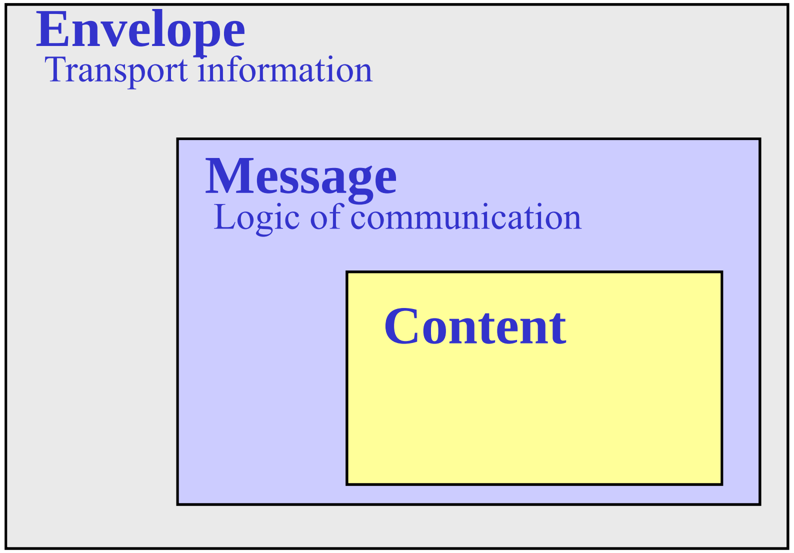
Category	Performatives
Basic Query	evaluate, ask-if, ask-one, ask-all, ask-about
Multi-response Query	stream-about, stream-all
Response	reply, sorry
Generic informational	tell, achieve, cancel, untell
Generator	standby, ready, next,rest, discard, generator
Capability-definition	advertise, subscribe, monitor, import, export
Networking	register, unregister, forward, broadcast, route

KQML Examples

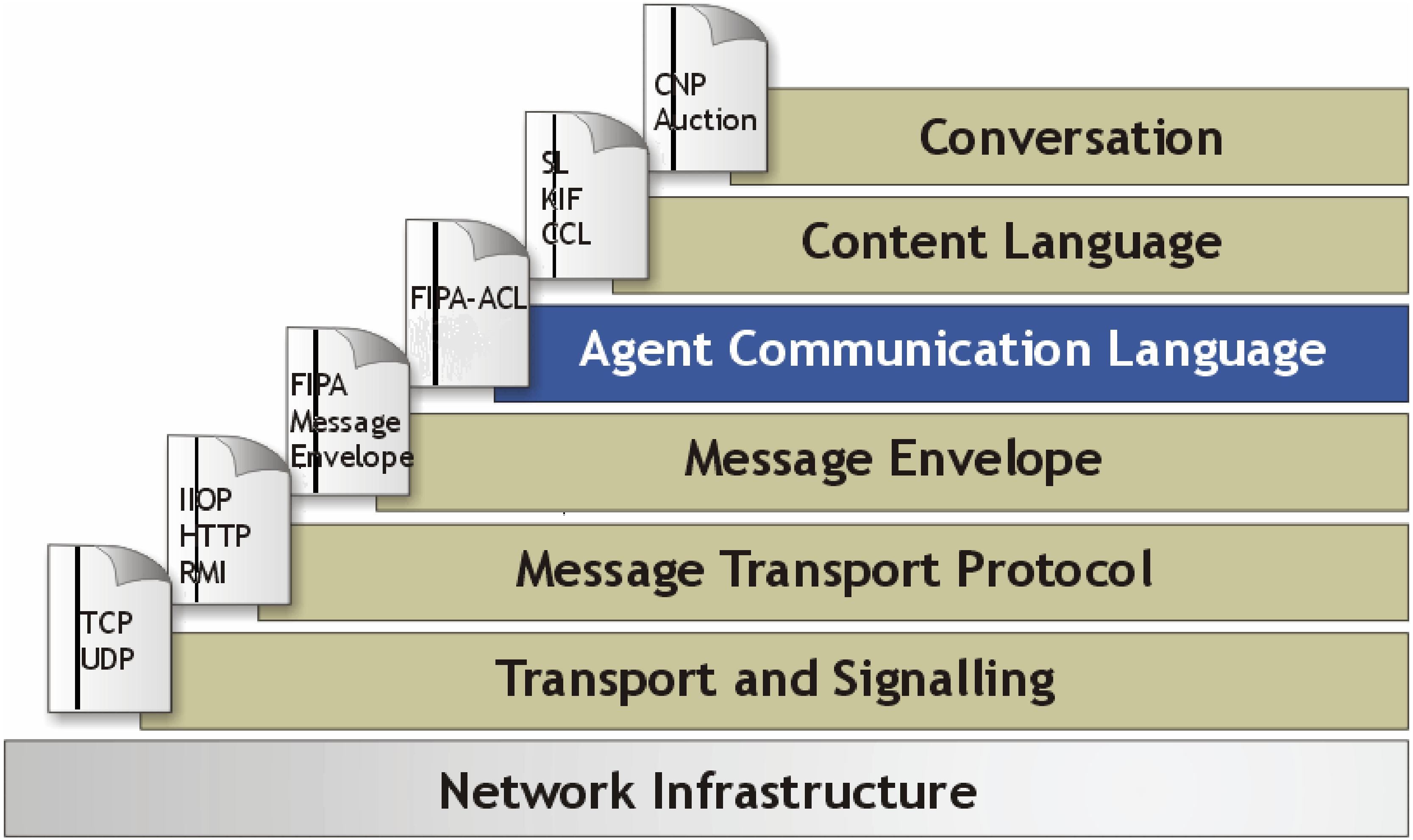
```
(evaluate
  :sender A : receiver B
  :language KIF: ontology motors
  :reply-with q1
  :content (val (torque m1)))

(reply
  :sender B : receiver A
  :language KIF: ontology motors
  :in-reply-to q1
  :content
    (=  (torque m1) (scalar 12 kgf)))

(stream-about
     :sender A : receiver B
     :language KIF: ontology motors
     :reply-with q2 :content (m1)

(tell
     :sender B : receiver A
     :in-reply-to q2
     :content
    (=(torque m1)
      (scalar 12 kgf)))

(tell
     :sender B : receiver A
     :in-reply-to q2
     :content
    (=(status m1)(normal)))

(eos
     :sender B : receiver A
     :in-reply-to q2)
```

FIPA ACL

- FIPA ACL structure is quite similar to KQML
- FIPA ACL is an "outer language" that defines a set of performatives (communicative acts) used for exchanges of knowledge
- But knowledge can be expressed in different languages (e.g., SL, Prolog, KIF, etc.)



Layered Model



FIPA Performatives

Category	Performatives
Informing	confirm, disconfirm, inform, inform-if, inform-ref
Requiring information	cancel, query-if, query-ref, subscribe
Negotiating	accept-proposal, cfp, propose, reject-proposal
Performing action	agree, cancel, refuse, request, request-when, request.whenever
Error handling	failure, not-understood

FIPA Performatives

Speech act	Performative
Is the door open?	query
Open the door (for me)	request
OK! I'll open the door	agree
The door is open	inform
I am unable to open the door	failure
I will not open the door	refuse
Say when the door becomes open	subscribe
Anyone want to open the door?	cfp
I can open the door for you ... at a price	propose
Door? What's that? Don't understand ...	not-understood

AOT LAB

Performative Parameters

Scope	Parameters
Participants in communication	sender, receiver, reply-to
Content value	content
Content description	language, encoding, ontology
Control of conversation	protocol, conversation-id, reply-with, in-reply-to, reply-by

FIPA Examples

```
(inform
  :sender (agent-identifier :name i)
  :receiver (agent-identifier :name j)
  :content ”door( now, open )"
  :language Prolog)

(request
  :sender (agent-identifier :name i)
  :receiver (agent-identifier :name j)
  :content
     (action (agent-identifier :name j)
                  open_the_door)
  :language fipa-sl)

(confirm
  :sender i
  :receiver j
  :content "weather( today, snowing )"
  :language Prolog)
```

```
(query-ref
  :sender i
  :receiver j
  :content (iota ?x (available-services ?x))
```

```
(cfp
  :sender (agent-identifier :name j)
  :receiver (set (agent-identifier :name i))
  :content
   "((action (agent-identifier :name i)
     (sell plum 50))
    (any ?x (and (= (price plum) ?x)
                      (< ?x 10))))"
  :ontology fruit-market
  :language fipa-sl)
```

FIPA ACL Semantics Properties

- Property 1
  - An agent's intention to achieve a given goal generates an intention that one of the acts known to the agent be done
  - Further, the act is such that its rational effect corresponds to the agent's goal, and that agent has no reason for not doing it
- Property 2:
  - Whenever an agent chooses to perform some act, the agent intends to seek the satisfiability of its feasibility preconditions

FIPA ACL Semantics Properties

- Property 3:
  - if an agent has the intention that a communicative act be performed, it necessarily has the intention to bring about the rational effect of the act
- Property 4 (Intentional Effect)
  - When an agent observes a communicative act, it should believe that the agent performing the act has the intention to achieve the rational effect of the act
- Property 5
  - When an agent observes a given communicative act, it is entitled to believe that the persistent feasibility preconditions hold

FIPA ACL Semantics

- The two basic performatives in FIPA ACL are "inform" and "request"
- The meaning of inform and request is defined in two parts
  - Pre-condition (FP = *feasibility precondition*)
    - What must be true in order for the speech act to succeed
  - Rational effect
    - What the sender of the message hopes to bring about

Inform Semantics

- The sender informs the receiver that a given proposition is true
- The sending agent:
  - Holds that some proposition is true
  - Intends that the receiving agent also comes to believe that the proposition is true
  - Does not already believe that the receiver has any knowledge of the truth of the proposition

<*i*, inform(*j*, ϕ)>

FP: Biϕ $\wedge$ $\neg$Bi(Bifj ϕ $\vee$ Uifj ϕ)

RE: Bjϕ

Request Semantics

- The sender requests the receiver to perform an action
- The sending agent:
  - Believes that the other agent is able to do the action
  - Does not believe that the receiver has already the intention to perform the action

<i, request(j,a)>

FP: FP(a) [i/j] $\wedge$ agent(j, a) $\wedge$ $\neg$Bi Ij Done(a)

RE: Done(a)

AOT LAB

Other Performatives Semantics

- All others performatives are macro definitions, defined in terms of these two performatives

- For example, the query-if performative can be defined as a request to be informed about the veracity of some proposition

$<i, \text{query-if}(j, \phi) \equiv <i, \text{request}(j, <j, \text{inform-if}(i, \phi)>)>$

FP: $\neg Bifj\phi \wedge \neg Uifj\phi \wedge \neg Bi\ Ij\ \text{Done}(<j, \text{inform-if}(i, \phi)>)$

RE: $\text{Done}(<j, \text{inform}(i, \phi)>|<j, \text{inform}(i, \neg\phi)>)$

FIPA ACL Content Languages

- FIPA-SL
  - Used in most of the FIPA specs
    - Mandatory for example for agent management
  - Three subsets
    - SL0 (Minimal Subset of SL)
    - SL1 (Proposition Form)
    - SL2 (Restrictions for decidability)
- FIPA-CCL
  - Constraint Choice Language
- FIPA-RDF0
  - RDF based content language
- FIPA-KIF

FIPA compliant Content Languages Properties

- The utility of such a new language should be made clear
- A good level of syntactical development is required
- A clear and intuitive (although not necessarily formal) semantics
- Substantial and clear documentation must be provided
  - Examples of the usage of such a language are recommended
- In general, a content language must be able to express propositions, objects and actions
  - No other properties are required, though any given content language may be much more expressive than this

Interaction Protocols

- Ongoing conversation between agents often fall in
- typical patterns

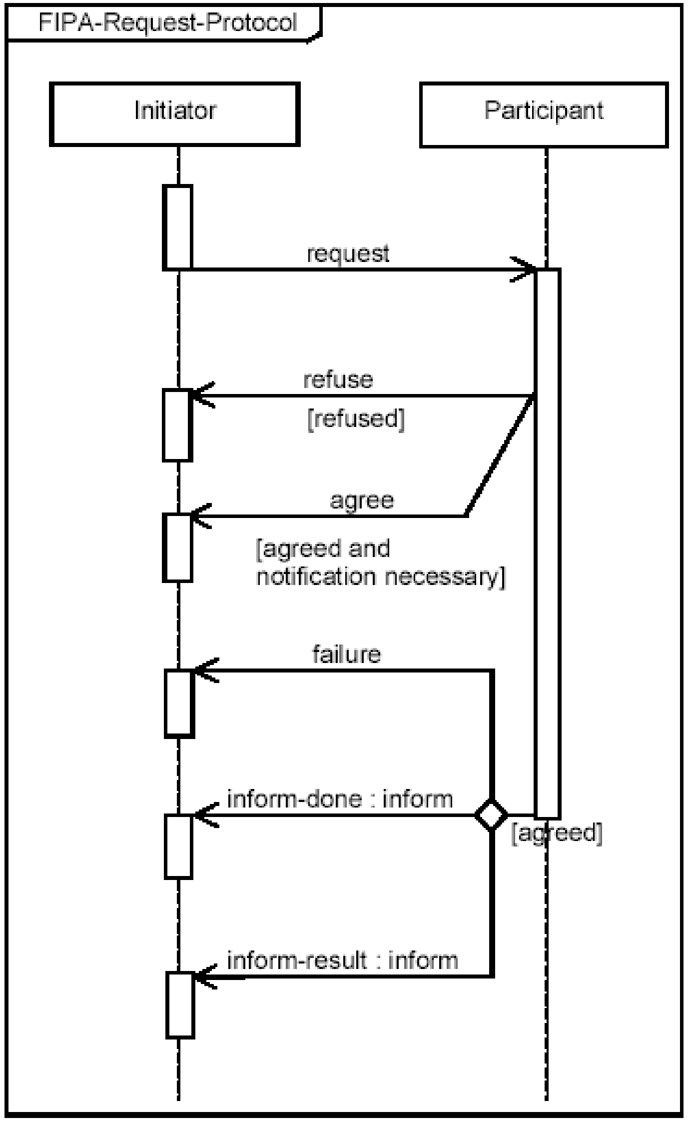
- Typical patterns of message exchange are called interaction protocols
  - FIPA pre-specifies a number of protocols

- The use of interaction protocols eases the implementation
  - Agent can engage in meaningful interactions simply by following an interaction protocol

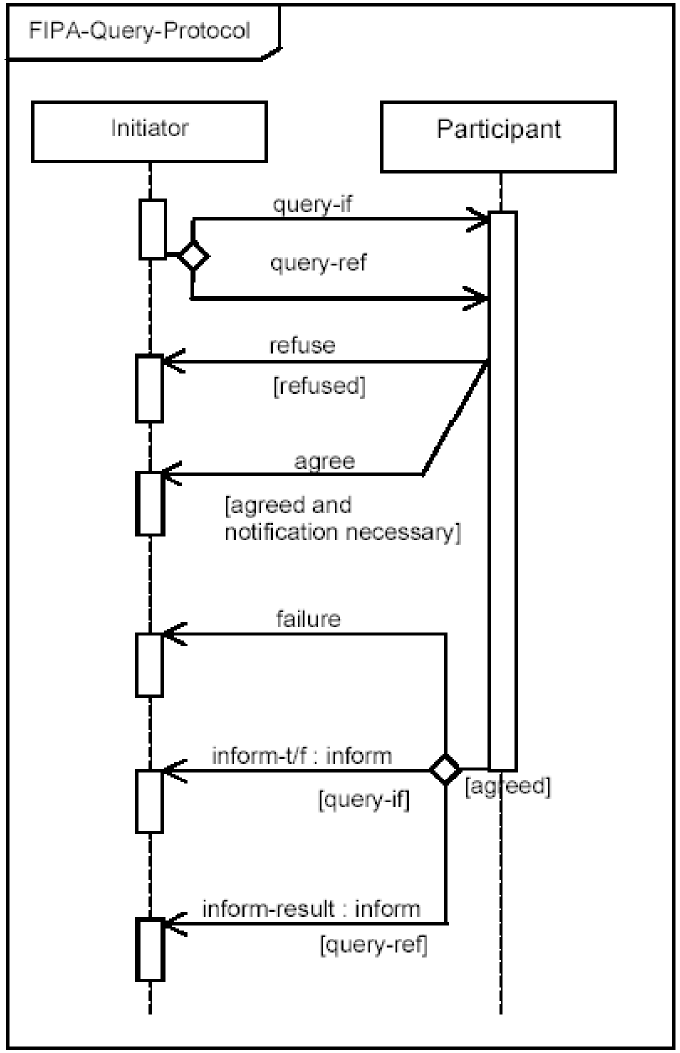
FIPA Interaction Protocols

- FIPA Request
- FIPA Query
- FIPA Request When
- FIPA Contract Net
- FIPA Iterated Contract Net
- FIPA English Auction
- FIPA Dutch Auction
- FIPA Brokering
- FIPA Recruiting
- FIPA Subscribe
- FIPA Propose Interaction

FIPA-Request

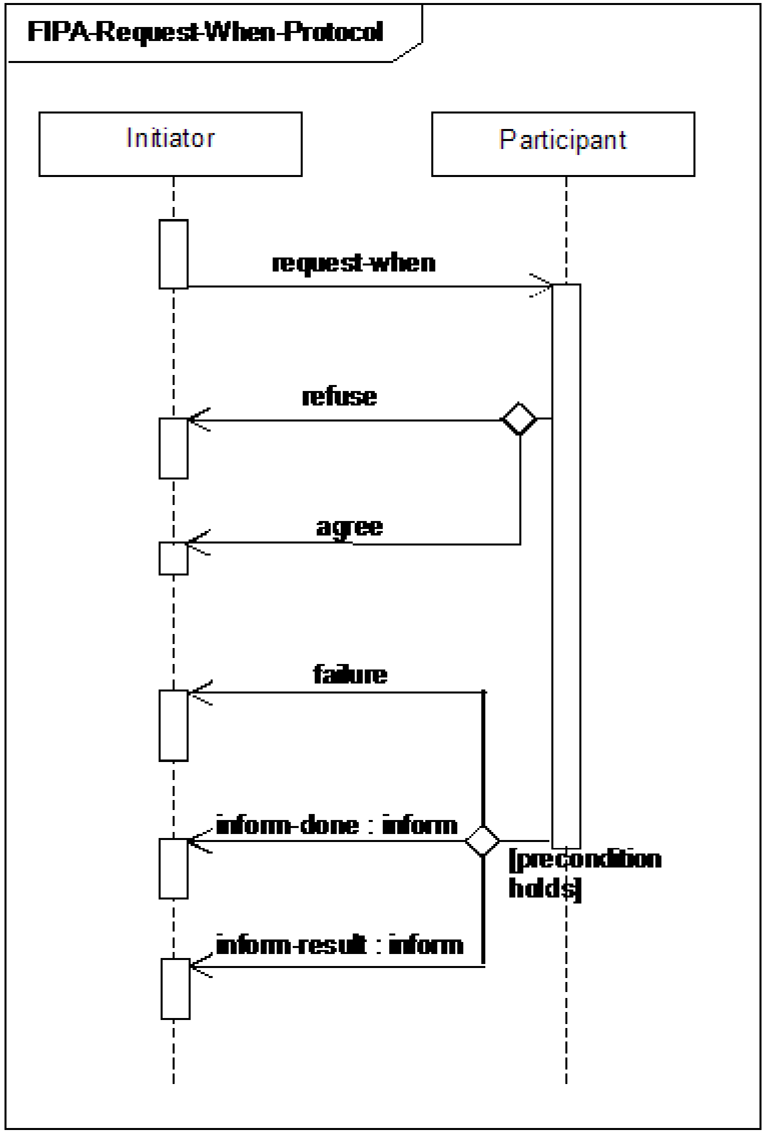


FIPA-Query

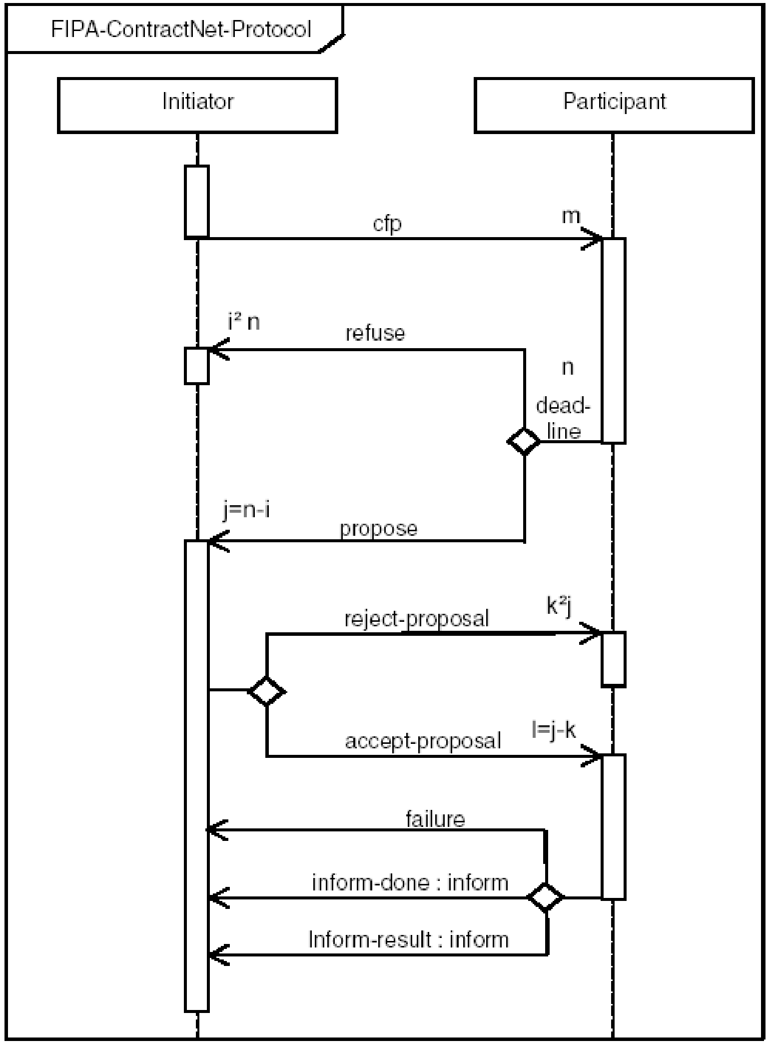


FIPA-Query-Protocol
Initiator
Participant
query-if
query-ref
refuse
[refused]
agree
[agreed and
notification necessary]
failure
inform-t/f : inform
[query-if]
[agreed]
inform-result : inform
[query-ref]

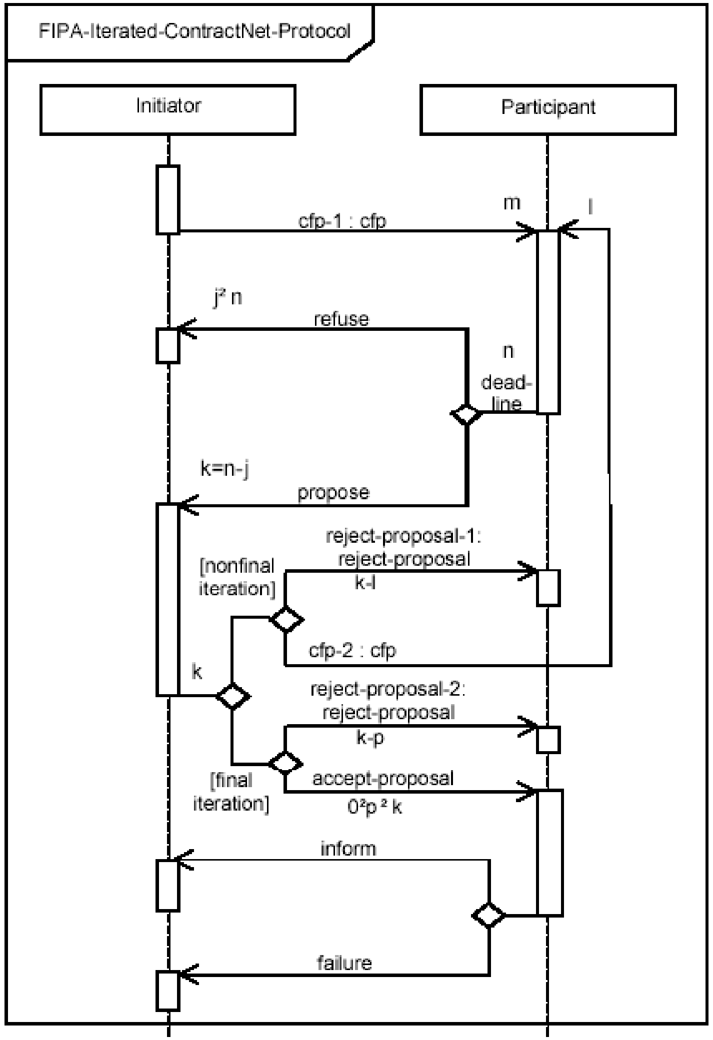
FIPA-Request-when



FIPA-Contract-net

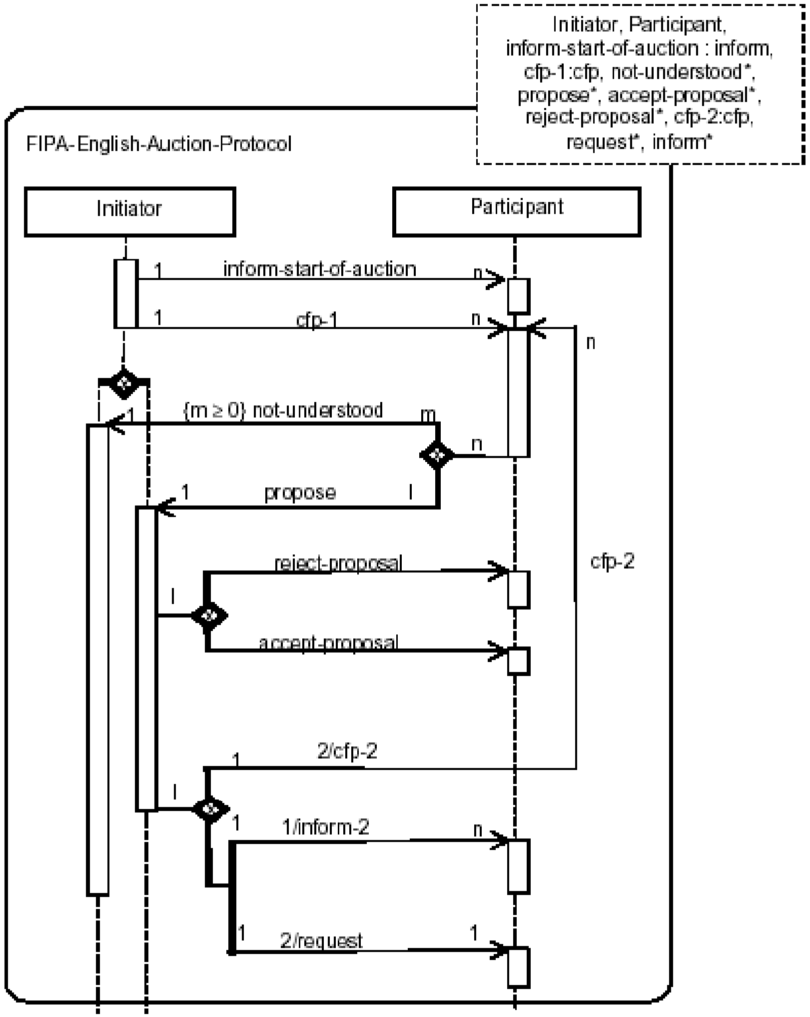


FIPA-Iterated-Contract-net



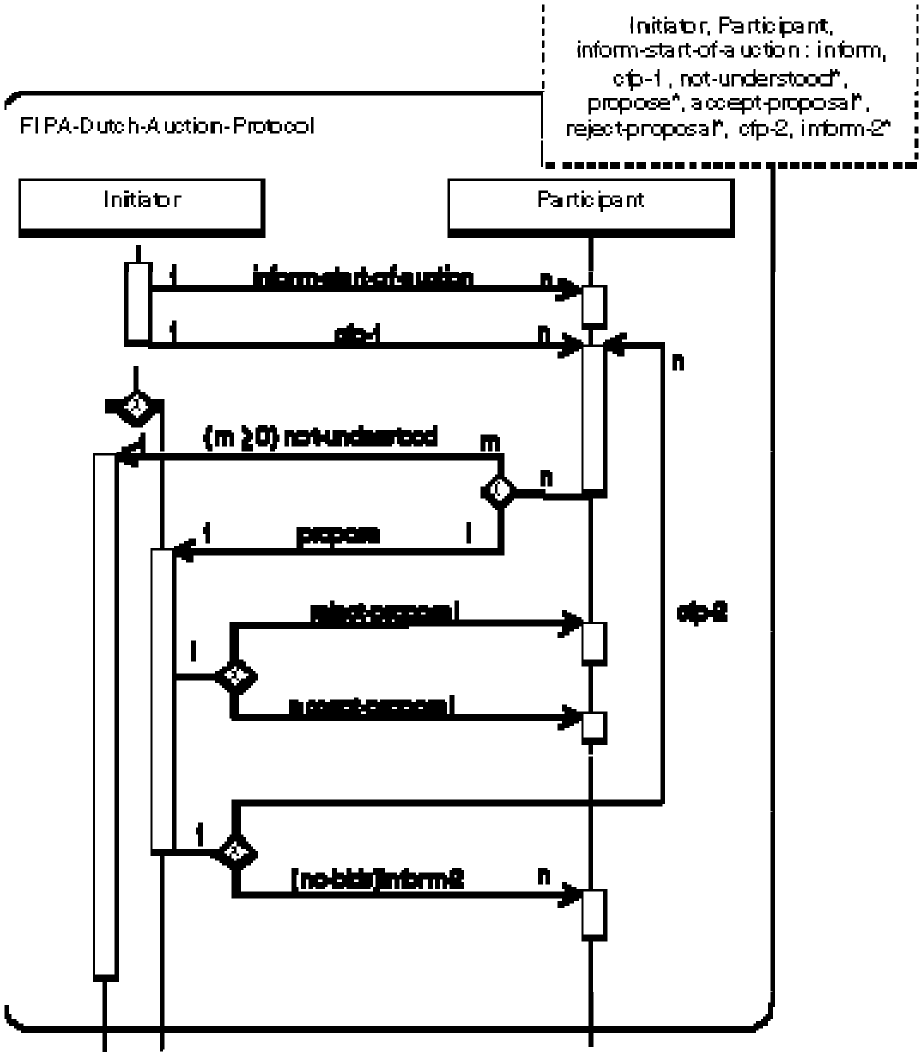
AOT LAB

FIPA-English-Auction

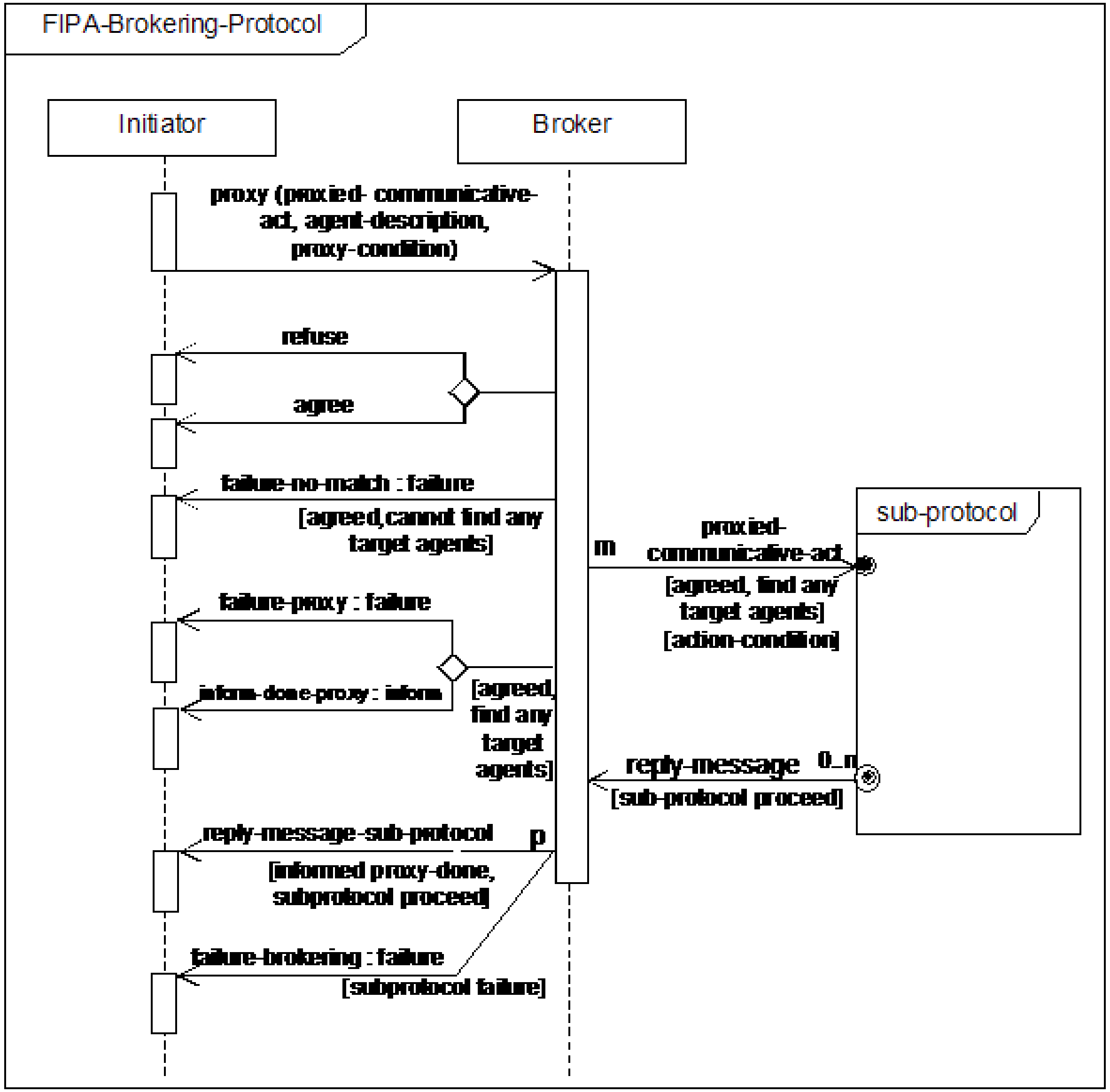


AOT LAB

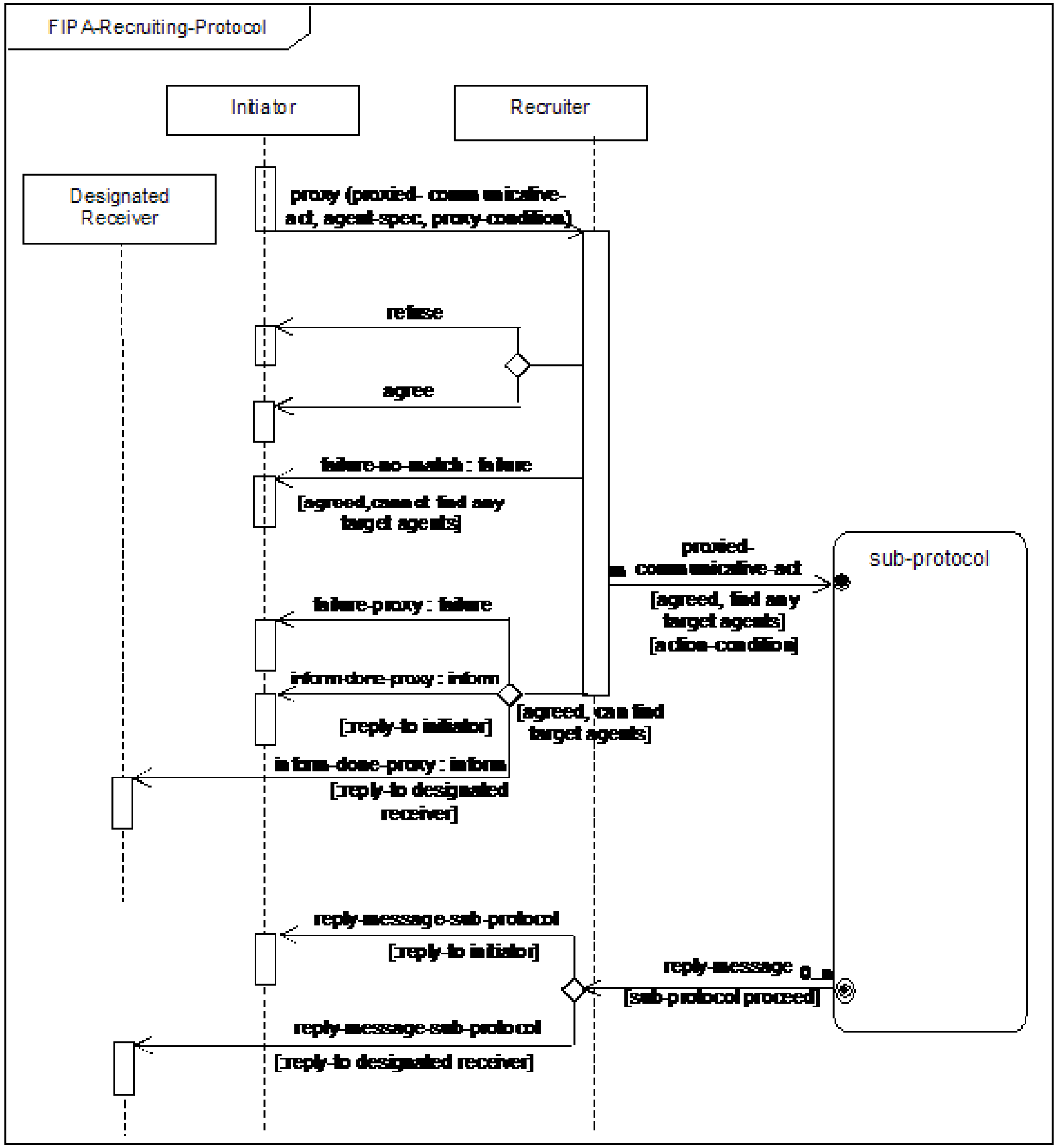
FIPA-Dutch-Auction



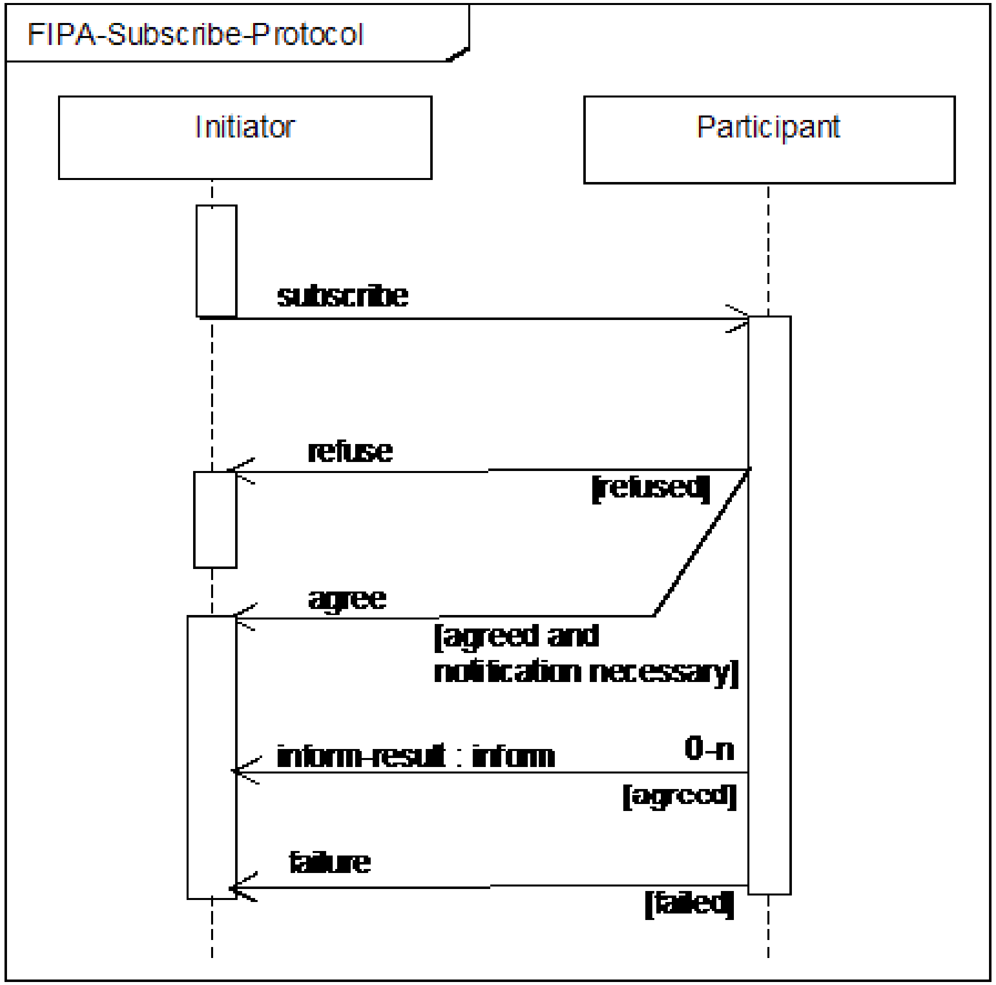
FIPA-Brokering



FIPA-Recruiting



FIPA-Subscribe



FIPA-Propose