FOUNDATION FOR INTELLIGENT PHYSICAL AGENTS

FIPA Request Interaction Protocol Specification

Document title	FIPA Request Interaction Protocol Specification		
Document number	SC00026H	**Document source**	FIPA TC Communication
Document status	Standard	**Date of this status**	2002/12/03
Supersedes	None		
Contact	fab@fipa.org		
Change history	See *Informative Annex A — ChangeLog*		

http://www.fipa.org/
Geneva, Switzerland

Foreword

The Foundation for Intelligent Physical Agents (FIPA) is an international organization that is dedicated to promoting the industry of intelligent agents by openly developing specifications supporting interoperability among agents and agent-based applications. This occurs through open collaboration among its member organizations, which are companies and universities that are active in the field of agents. FIPA makes the results of its activities available to all interested parties and intends to contribute its results to the appropriate formal standards bodies where appropriate.

The members of FIPA are individually and collectively committed to open competition in the development of agent-based applications, services and equipment. Membership in FIPA is open to any corporation and individual firm, partnership, governmental body or international organization without restriction. In particular, members are not bound to implement or use specific agent-based standards, recommendations and FIPA specifications by virtue of their participation in FIPA.

The FIPA specifications are developed through direct involvement of the FIPA membership. The status of a specification can be either Preliminary, Experimental, Standard, Deprecated or Obsolete. More detail about the process of specification may be found in the FIPA Document Policy [f-out-00000] and the FIPA Specifications Policy [f-out-00003]. A complete overview of the FIPA specifications and their current status may be found on the FIPA Web site.

FIPA is a non-profit association registered in Geneva, Switzerland. As of June 2002, the 56 members of FIPA represented many countries worldwide. Further information about FIPA as an organization, membership information, FIPA specifications and upcoming meetings may be found on the FIPA Web site at http://www.fipa.org/.

Contents

1 FIPA Request Interaction Protocol

The FIPA Request Interaction Protocol (IP) allows one agent to request another to perform some action.

The representation of this protocol is given in *Figure 1* which is based on extensions to UML 1.x. [Odell2001]. This protocol is identified by the token `fipa-request` as the value of the `protocol` parameter of the ACL message.

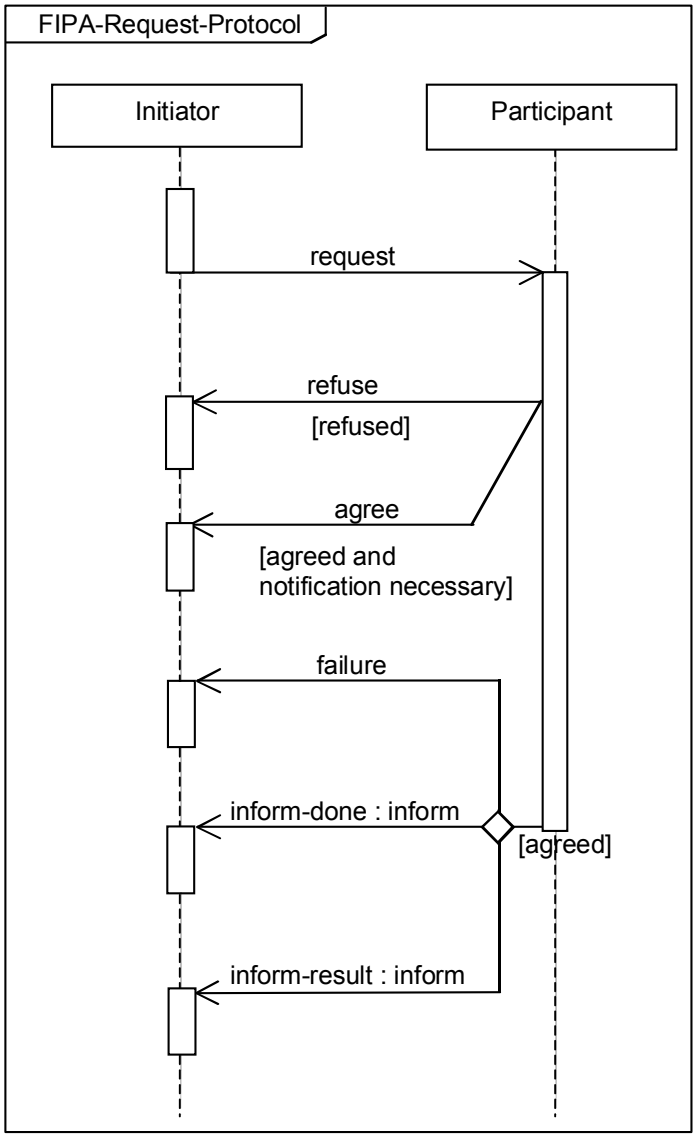

Figure 1: FIPA Request Interaction Protocol

1.1 Explanation of the Protocol Flow

The FIPA Request Interaction Protocol (IP) allows one agent to request another to perform some action. The Participant processes the request and makes a decision whether to accept or refuse the request. If a refuse decision is made, then "refused" becomes true and the Participant communicates a `refuse`. Otherwise, "agreed" becomes true.

If conditions indicate that an explicit agreement is required (that is, "notification necessary" is true), then the Participant communicates an `agree`. The `agree` may be optional depending on circumstances, for example, if the requested

action is very quick and can happen before a time specified in the `reply-by` parameter. Once the request has been agreed upon, then the Participant must communicate either:

- A `failure` if it fails in its attempt to fill the request,
- An `inform-done` if it successfully completes the request and only wishes to indicate that it is done, or,
- An `inform-result` if it wishes to indicate both that it is done and notify the initiator of the results.

Any interaction using this interaction protocol is identified by a globally unique, non-null `conversation-id` parameter, assigned by the Initiator. The agents involved in the interaction must tag all of its ACL messages with this conversation identifier. This enables each agent to manage its communication strategies and activities, for example, it allows an agent to identify individual conversations and to reason across historical records of conversations.

1.2 Exceptions to Protocol Flow

At *any* point in the IP, the receiver of a communication can inform the sender that it did not understand what was communicated. This is accomplished by returning a `not-understood` message. As such, *Figure 1* does not depict a `not-understood` communication as it can occur at any point in the IP. The communication of a `not-understood` within an interaction protocol may terminate the entire IP and termination of the interaction may imply that any commitments made during the interaction are null and void.

At any point in the IP, the initiator of the IP may cancel the interaction protocol by initiating the meta-protocol shown in *Figure 2*. The `conversation-id` parameter of the cancel interaction is identical to the `conversation-id` parameter of the interaction that the Initiator intends to cancel. The semantics of cancel should roughly be interpreted as meaning that the initiator is no longer interested in continuing the interaction and that it should be terminated in a manner acceptable to both the Initiator and the Participant. The Participant either informs the Initiator that the interaction is done using an `inform-done` or indicates the failure of the cancellation using a `failure`.

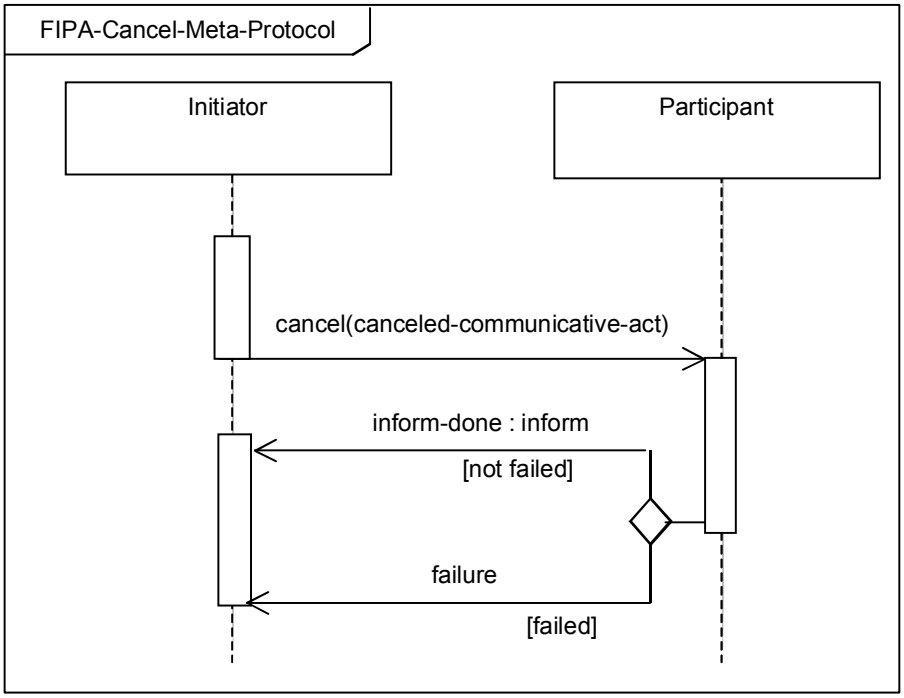

Figure 2: FIPA Cancel Meta-Protocol

This IP is a pattern for a simple interaction type. Elaboration on this pattern will almost certainly be necessary in order to specify all cases that might occur in an actual agent interaction. Real world issues such as the effects of cancelling actions, asynchrony, abnormal or unexpected IP termination, nested IPs, and the like, are explicitly not addressed here.

2 References

[Odell2001] Odell, James, Van Dyke Parunak, H. and Bauer, B., *Representing Agent Interaction Protocols in UML*. In: Agent-Oriented Software Engineering, Ciancarini, P. and Wooldridge, M., Eds., Springer, pp. 121-140, Berlin, 2001.
http://www.fipa.org/docs/input/f-in-00077/

3 Informative Annex A — ChangeLog

3.1 2002/11/01 - version G by TC X2S

Page 1, Figure 1:	The communication labeled `inform-ref` was changed to `inform-result` for clarity; the purpose of this communication is to inform the initiator of a result and `inform-result` implies `inform-done`
Page 1, Figure 1:	The `not-understood` communication was removed
Page 1, Figure 1:	Reworked the protocol flow to make the `agree` optional which also involved changing the exclusive-or with the `agree` to a different AUML notation
Page 1, Figure 1:	To conform to UML 2, the protocol name was placed in a boundary, x is removed from the diamonds (xor is now the default) and the template box was removed
Page 1, line 41:	Reworked and expanded the section description of the IP
Page 1, line 50:	Added a new section on Explanation of Protocol Flow
Page 1, line 50:	Reworked and expanded the section on Exceptions of Protocol Flow to incorporate a meta-protocol for cancel
Page 1, line 50:	Added a paragraph explaining the `not-understood` communication and its relationship with the IP

3.2 2002/12/03 - version H by FIPA Architecture Board

Entire document:	Promoted to Standard status

FOUNDATION FOR INTELLIGENT PHYSICAL AGENTS

FIPA Query Interaction Protocol Specification

Document title	FIPA Query Interaction Protocol Specification		
Document number	SC00027H	**Document source**	FIPA TC Communication
Document status	Standard	**Date of this status**	2002/12/03
Supersedes	None		
Contact	fab@fipa.org		
Change history	See *Informative Annex A — ChangeLog*		

http://www.fipa.org/
Geneva, Switzerland

Foreword

The Foundation for Intelligent Physical Agents (FIPA) is an international organization that is dedicated to promoting the industry of intelligent agents by openly developing specifications supporting interoperability among agents and agent-based applications. This occurs through open collaboration among its member organizations, which are companies and universities that are active in the field of agents. FIPA makes the results of its activities available to all interested parties and intends to contribute its results to the appropriate formal standards bodies where appropriate.

The members of FIPA are individually and collectively committed to open competition in the development of agent-based applications, services and equipment. Membership in FIPA is open to any corporation and individual firm, partnership, governmental body or international organization without restriction. In particular, members are not bound to implement or use specific agent-based standards, recommendations and FIPA specifications by virtue of their participation in FIPA.

The FIPA specifications are developed through direct involvement of the FIPA membership. The status of a specification can be either Preliminary, Experimental, Standard, Deprecated or Obsolete. More detail about the process of specification may be found in the FIPA Document Policy [f-out-00000] and the FIPA Specifications Policy [f-out-00003]. A complete overview of the FIPA specifications and their current status may be found on the FIPA Web site.

FIPA is a non-profit association registered in Geneva, Switzerland. As of June 2002, the 56 members of FIPA represented many countries worldwide. Further information about FIPA as an organization, membership information, FIPA specifications and upcoming meetings may be found on the FIPA Web site at http://www.fipa.org/.

Contents

1 FIPA Query Interaction Protocol

The FIPA Query Interaction Protocol (IP) allows one agent to request to perform some kind of action on another agent.

The representation of this IP is given in *Figure 1* which is based on extensions to UML1.x [Odell2001]. This protocol is identified by the token `fipa-query` as the value of the `protocol` parameter of the ACL message.

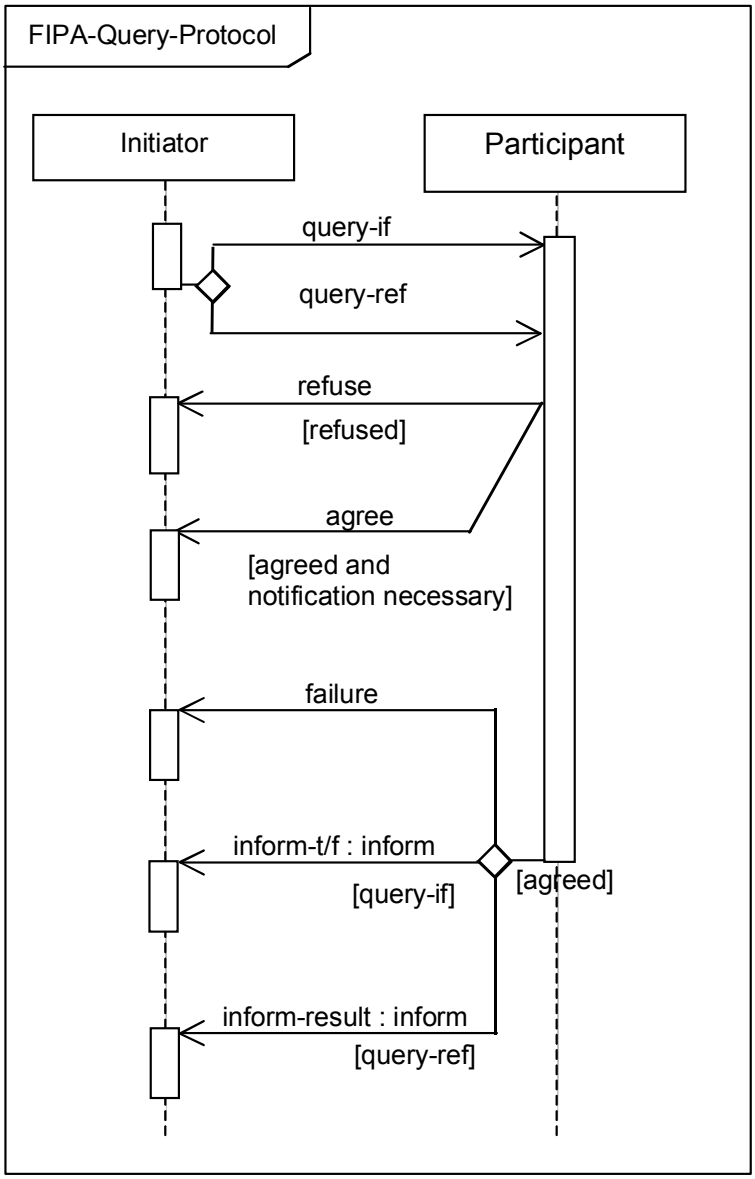

Figure 1: FIPA Query Interaction Protocol

1.1 Explanation of the Protocol Flow

The Initiator requests the Participant to perform some kind of `inform` action using one of two query communicative acts, `query-if` or `query-ref` (see [FIPA00037]). The `query-if` communication is used when the Initiator wants to query whether a particular proposition is true or false and the `query-ref` communication is used when the Initiator wants to query for some identified objects. The Participant processes the `query-if` or `query-ref` and makes a decision whether to accept or refuse the query request. If the Participant makes a refuse decision, then “refused” becomes true and the Participant communicates a `refuse`. Otherwise, “agreed” becomes true.

If conditions indicate that an explicit agreement is required (that is, “notification necessary” is true), then the Participant communicates an `agree`. The `agree` may be optional depending on circumstances, for example, if the requested action is very quick and can happen before a time specified in the `reply-by` parameter. If the Participant fails, then it communicates a `failure`.

In a successful response, the Participant replies with one of two versions of `inform`:

- The Participant uses an `inform-t/f` communication in response to a `query-if` where the content of the `inform-t/f` asserts the truth or falsehood of the proposition, or,

- The Participant returns an `inform-result` communication in response to a `query-ref` and the content of the `inform-result` contains a referring expression to the objects for which the query was specified.

Any interaction using this interaction protocol is identified by a globally unique, non-null `conversation-id` parameter, assigned by the Initiator. The agents involved in the interaction must tag all of its ACL messages with this conversation identifier. This enables each agent to manage its communication strategies and activities, for example, it allows an agent to identify individual conversations and to reason across historical records of conversations.

1.2 Exceptions to Interaction Protocol Flow

At *any* point in the IP, the receiver of a communication can inform the sender that it did not understand what was communicated. This is accomplished by returning a `not-understood` message. As such, *Figure 1* does not depict a `not-understood` communication as it can occur at any point in the IP. The communication of a `not-understood` within an interaction protocol may terminate the entire IP and termination of the interaction may imply that any commitments made during the interaction are null and void.

At any point in the IP, the initiator of the IP may cancel the interaction protocol by initiating the meta-protocol shown in *Figure 2*. The `conversation-id` parameter of the cancel interaction is identical to the `conversation-id` parameter of the interaction that the Initiator intends to cancel. The semantics of cancel should roughly be interpreted as meaning that the initiator is no longer interested in continuing the interaction and that it should be terminated in a manner acceptable to both the Initiator and the Participant. The Participant either informs the Initiator that the interaction is done using an `inform-done` or indicates the failure of the cancellation using a `failure`.

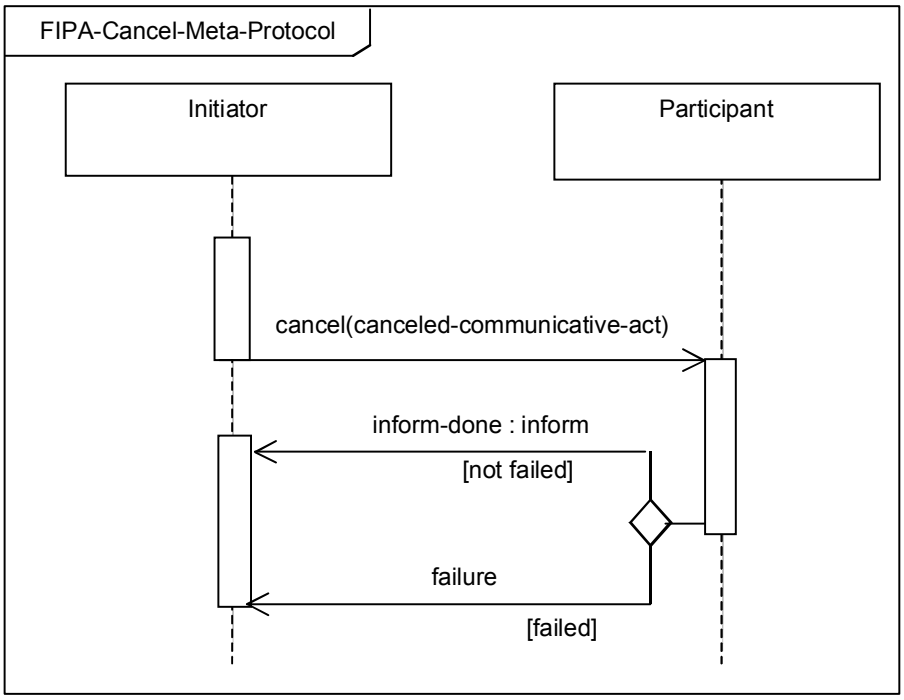

Figure 2: FIPA Cancel Meta-Protocol

This IP is a pattern for a simple interaction type. Elaboration on this pattern will almost certainly be necessary in order to specify all cases that might occur in an actual agent interaction. Real world issues such as the effects of cancelling actions, asynchrony, abnormal or unexpected IP termination, nested IPs, and the like, are explicitly not addressed here.

2 References

[FIPA00037] FIPA Communicative Act Library Specification. Foundation for Intelligent Physical Agents, 2000. http://www.fipa.org/specs/fipa00037/

[Odell2001] Odell, James, Van Dyke Parunak, H. and Bauer, B., *Representing Agent Interaction Protocols in UML*. In: Agent-Oriented Software Engineering, Ciancarini, P. and Wooldridge, M., Eds., Springer, pp. 121-140, Berlin, 2001. http://www.fipa.org/docs/input/f-in-00077/

3 Informative Annex A — ChangeLog

3.1 2002/11/01 - version G by TC X2S

Page 1, Figure 1: The `not-understood` communication was removed

Page 1, Figure 1: Reworked the protocol flow to make the agree optional and made explicit the different inform response content expected for a `query-if` as opposed to a `query-ref`

Page 1, Figure 1: To conform to UML 2, the protocol name was placed in a boundary, x is removed from the diamonds (xor is now the default) and the template box was removed

Page 1, line 42: Reworked and expanded the section description of the IP

Page 1, line 54: Added a new section on Explanation of Protocol Flow

Page 1, line 54: Reworked and expanded the section on Exceptions of Protocol Flow to incorporate a meta-protocol for cancel

Page 1, line 54: Added a paragraph explaining the `not-understood` communication and its relationship with the IP

3.2 2002/12/03 - version H by FIPA Architecture Board

Entire document: Promoted to Standard status

FOUNDATION FOR INTELLIGENT PHYSICAL AGENTS

FIPA Request When Interaction Protocol Specification

Document title	FIPA Request When Interaction Protocol Specification		
Document number	SC00028H	**Document source**	FIPA TC Communication
Document status	Standard	**Date of this status**	2002/12/03
Supersedes	None		
Contact	fab@fipa.org		
Change history	See *Informative Annex A — ChangeLog*		

http://www.fipa.org/
Geneva, Switzerland

Foreword

The Foundation for Intelligent Physical Agents (FIPA) is an international organization that is dedicated to promoting the industry of intelligent agents by openly developing specifications supporting interoperability among agents and agent-based applications. This occurs through open collaboration among its member organizations, which are companies and universities that are active in the field of agents. FIPA makes the results of its activities available to all interested parties and intends to contribute its results to the appropriate formal standards bodies where appropriate.

The members of FIPA are individually and collectively committed to open competition in the development of agent-based applications, services and equipment. Membership in FIPA is open to any corporation and individual firm, partnership, governmental body or international organization without restriction. In particular, members are not bound to implement or use specific agent-based standards, recommendations and FIPA specifications by virtue of their participation in FIPA.

The FIPA specifications are developed through direct involvement of the FIPA membership. The status of a specification can be either Preliminary, Experimental, Standard, Deprecated or Obsolete. More detail about the process of specification may be found in the FIPA Document Policy [f-out-00000] and the FIPA Specifications Policy [f-out-00003]. A complete overview of the FIPA specifications and their current status may be found on the FIPA Web site.

FIPA is a non-profit association registered in Geneva, Switzerland. As of June 2002, the 56 members of FIPA represented many countries worldwide. Further information about FIPA as an organization, membership information, FIPA specifications and upcoming meetings may be found on the FIPA Web site at http://www.fipa.org/.

Contents

1 FIPA Request When Interaction Protocol

The FIPA Request When Interaction Protocol (IP) allows an agent to request that the receiver perform some action at the time a given precondition becomes true. This IP provides a framework for the `request-when` communicative act (see [FIPA00037]).

The representation of this IP is given in *Figure 1* which is based on extensions to UML1.x. [Odell2001]. This protocol is identified by the token `fipa-request-when` as the value of the `protocol` parameter of the ACL message.

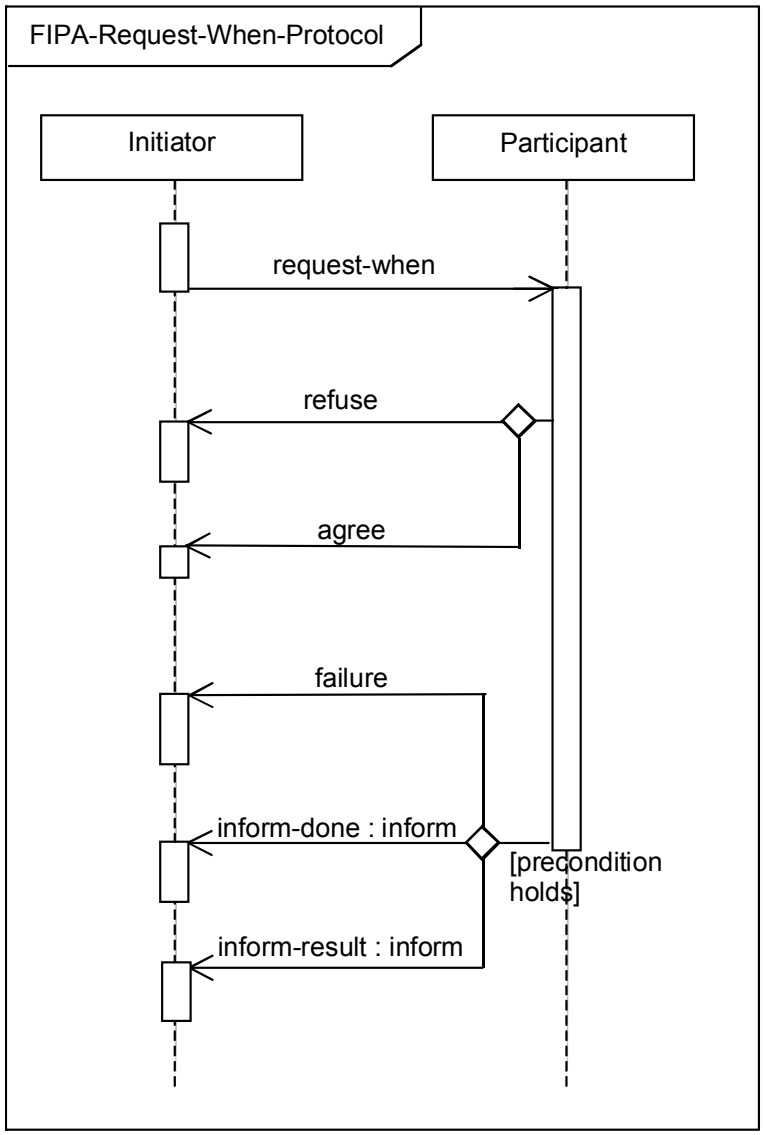

Figure 1: FIPA Request When Interaction Protocol

1.1 Explanation of the Protocol Flow

The initiator uses the `request-when` action to request that the participant do some action once a given precondition becomes true. If the requested agent understands the request and does not initially refuse, it will `agree` (see [FIPA00037]) and wait until the precondition occurs. Then, it will attempt to perform the action and notify the requester accordingly.

If after the initial agreement the participant is no longer able to perform the action, then it will send a `failure` action (see [FIPA00037]) to the initiator. Once the action has completed and the `failure`, `inform-done`, or `inform-result` has been sent, the conversation ends.

Any interaction using this interaction protocol is identified by a globally unique, non-null `conversation-id` parameter, assigned by the Initiator. The agents involved in the interaction must tag all of its ACL messages with this conversation identifier. This enables each agent to manage its communication strategies and activities, for example, it allows an agent to identify individual conversations and to reason across historical records of conversations.

1.2 Exceptions to Interaction Protocol Flow

At *any* point in the IP, the receiver of a communication can inform the sender that it did not understand what was communicated. This is accomplished by returning a `not-understood` message. As such, *Figure 1* does not depict a `not-understood` communication as it can occur at any point in the IP. The communication of a `not-understood` within an interaction protocol may terminate the entire IP and termination of the interaction may imply that any commitments made during the interaction are null and void.

At any point in the IP, the initiator of the IP may cancel the interaction protocol by initiating the meta-protocol shown in *Figure 2*. The `conversation-id` parameter of the cancel interaction is identical to the `conversation-id` parameter of the interaction that the Initiator intends to cancel. The semantics of cancel should roughly be interpreted as meaning that the initiator is no longer interested in continuing the interaction and that it should be terminated in a manner acceptable to both the Initiator and the Participant. The Participant either informs the Initiator that the interaction is done using an `inform-done` or indicates the failure of the cancellation using a `failure`.

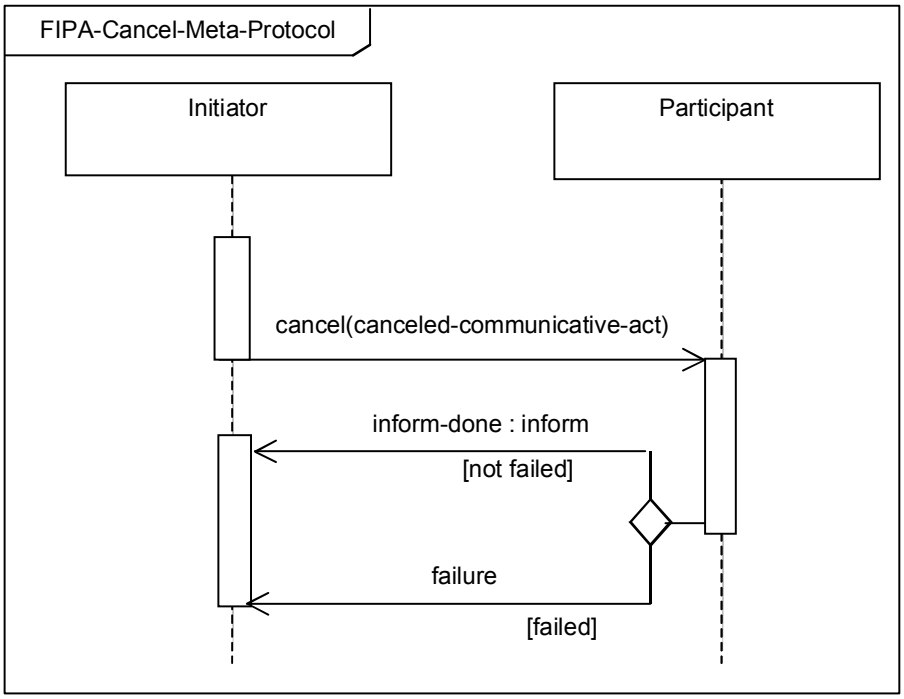

Figure 2: FIPA Cancel Meta-Protocol

This IP is a pattern for a simple interaction type. Elaboration on this pattern will almost certainly be necessary in order to specify all cases that might occur in an actual agent interaction. Real world issues such as the effects of cancelling actions, asynchrony, abnormal or unexpected IP termination, nested IPs, and the like, are explicitly not addressed here.

2 References

[FIPA00037] FIPA Communicative Act Library Specification. Foundation for Intelligent Physical Agents, 2000. http://www.fipa.org/specs/fipa00037/

[Odell2001] Odell, James, Van Dyke Parunak, H. and Bauer, B., *Representing Agent Interaction Protocols in UML*. In: Agent-Oriented Software Engineering, Ciancarini, P. and Wooldridge, M., Eds., Springer, pp. 121-140, Berlin, 2001. http://www.fipa.org/docs/input/f-in-00077/

3 Informative Annex A — ChangeLog

3.1 2002/11/01 - version G by TC X2S

Page 1, Figure 1: The communication labeled `inform-ref` was changed to `inform-result` for clarity; the purpose of this communication is to inform the initiator of a result and `inform-result` implies `inform-done`

Page 1, Figure 1: The `not-understood` communication was removed

Page 1, Figure 1: To conform to UML 2, the protocol name was placed in a boundary, x is removed from the diamonds (xor is now the default) and the template box was removed

Page 1, line 42: Reworked and expanded the section description of the IP

Page 1, line 56: Added a new section on Explanation of Protocol Flow

Page 1, line 56: Reworked and expanded the section on Exceptions of Protocol Flow to incorporate a meta-protocol for cancel

Page 1, line 56: Added a paragraph explaining the `not-understood` communication and its relationship with the IP

3.2 2002/12/03 - version H by FIPA Architecture Board

Entire document: Promoted to Standard status

FOUNDATION FOR INTELLIGENT PHYSICAL AGENTS

FIPA Contract Net Interaction Protocol Specification

Document title	FIPA Contract Net Interaction Protocol Specification		
Document number	SC00029H	**Document source**	FIPA TC Communication
Document status	Standard	**Date of this status**	2002/12/03
Supersedes	None		
Contact	fab@fipa.org		
Change history	See *Informative Annex A — ChangeLog*		

http://www.fipa.org/
Geneva, Switzerland

Foreword

The Foundation for Intelligent Physical Agents (FIPA) is an international organization that is dedicated to promoting the industry of intelligent agents by openly developing specifications supporting interoperability among agents and agent-based applications. This occurs through open collaboration among its member organizations, which are companies and universities that are active in the field of agents. FIPA makes the results of its activities available to all interested parties and intends to contribute its results to the appropriate formal standards bodies where appropriate.

The members of FIPA are individually and collectively committed to open competition in the development of agent-based applications, services and equipment. Membership in FIPA is open to any corporation and individual firm, partnership, governmental body or international organization without restriction. In particular, members are not bound to implement or use specific agent-based standards, recommendations and FIPA specifications by virtue of their participation in FIPA.

The FIPA specifications are developed through direct involvement of the FIPA membership. The status of a specification can be either Preliminary, Experimental, Standard, Deprecated or Obsolete. More detail about the process of specification may be found in the FIPA Document Policy [f-out-00000] and the FIPA Specifications Policy [f-out-00003]. A complete overview of the FIPA specifications and their current status may be found on the FIPA Web site.

FIPA is a non-profit association registered in Geneva, Switzerland. As of June 2002, the 56 members of FIPA represented many countries worldwide. Further information about FIPA as an organization, membership information, FIPA specifications and upcoming meetings may be found on the FIPA Web site at http://www.fipa.org/.

Contents

1 FIPA Contract Net Interaction Protocol

The FIPA Contract Net Interaction Protocol (IP) is a minor modification of the original contract net IP pattern[1] in that it adds rejection and confirmation communicative acts. In the contract net IP, one agent (the Initiator) takes the role of manager which wishes to have some task performed by one or more other agents (the Participants) and further wishes to optimise a function that characterizes the task. This characteristic is commonly expressed as the price, in some domain specific way, but could also be soonest time to completion, fair distribution of tasks, etc. For a given task, any number of the Participants may respond with a proposal; the rest must refuse. Negotiations then continue with the Participants that proposed.

The representation of this IP is given in *Figure 1* which is based on extensions to UML1.x. [Odell2001]. This protocol is identified by the token `fipa-contract-net` as the value of the `protocol` parameter of the ACL message.

[1] Originally developed by Smith and Davis.

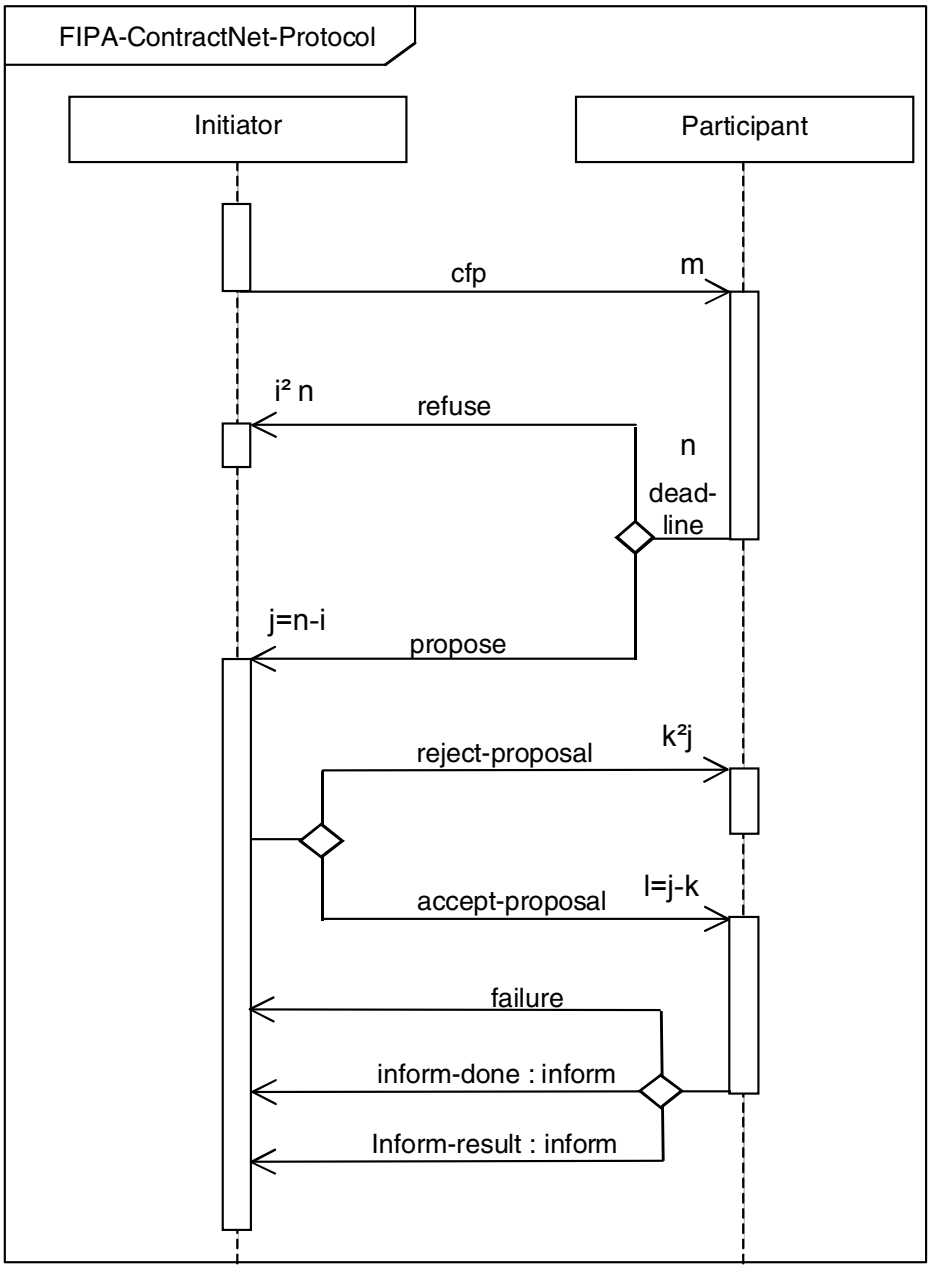

Figure 1: FIPA Contract Net Interaction Protocol

1.1 Explanation of the Protocol Flow

The Initiator solicits *m* proposals from other agents by issuing a call for proposals (`cfp`) act (see [FIPA00037]), which specifies the task, as well any conditions the Initiator is placing upon the execution of the task. Participants receiving the call for proposals are viewed as potential contractors and are able to generate *n* responses. Of these, *j* are proposals to perform the task, specified as `propose` acts (see [FIPA00037]).

The Participant's proposal includes the preconditions that the Participant is setting out for the task, which may be the price, time when the task will be done, etc. Alternatively, the *i=n-j* Participants may `refuse` (see [FIPA00037]) to propose. Once the deadline passes, the Initiator evaluates the received *j* proposals and selects agents to perform the task; one, several or no agents may be chosen. The *l* agents of the selected proposal(s) will be sent an `accept-`

`proposal` act (see [FIPA00037]) and the remaining *k* agents will receive a `reject-proposal` act (see [FIPA00037]). The proposals are binding on the Participant, so that once the Initiator accepts the proposal, the Participant acquires a commitment to perform the task. Once the Participant has completed the task, it sends a completion message to the Initiator in the form of an `inform-done` or a more explanatory version in the form of an `inform-result`. However, if the Participant fails to complete the task, a `failure` message is sent.

Note that this IP requires the Initiator to know when it has received all replies. In the case that a Participant fails to reply with either a `propose` or a `refuse` act, the Initiator may potentially be left waiting indefinitely. To guard against this, the `cfp` act includes a deadline by which replies should be received by the Initiator. Proposals received after the deadline are automatically rejected with the given reason that the proposal was late. The deadline is specified by the `reply-by` parameter in the ACL message.

Any interaction using this interaction protocol is identified by a globally unique, non-null `conversation-id` parameter, assigned by the Initiator. The agents involved in the interaction must tag all of its ACL messages with this conversation identifier. This enables each agent to manage its communication strategies and activities, for example, it allows an agent to identify individual conversations and to reason across historical records of conversations.

In the case of 1:N interaction protocols or sub-protocols the Initiator is free to decide if the same `conversation-id` parameter should be used or a new one should be issued. Additionally, the messages may specify other interaction-related information such as a timeout in the `reply-by` parameter that denotes the latest time by which the sending agent would like to have received the next message in the protocol flow.

1.2 Exceptions to Interaction Protocol Flow

At *any* point in the IP, the receiver of a communication can inform the sender that it did not understand what was communicated. This is accomplished by returning a `not-understood` message. As such, *Figure 1* does not depict a `not-understood` communication as it can occur at any point in the IP. The communication of a `not-understood` within an interaction protocol may terminate the entire IP and termination of the interaction may imply that any commitments made during the interaction are null and void. However, since this IP broadcasts to more than one Participant, multiple responses are also possible. Each response, then, must be evaluated separately – and some of these responses might be `not-understood`. However, terminating the entire IP in this case might not be appropriate, as other Participants may be continuing with their sub-protocols.

At any point in the IP, the initiator of the IP may cancel the interaction protocol by initiating the meta-protocol shown in *Figure 2*. The `conversation-id` parameter of the cancel interaction is identical to the `conversation-id` parameter of the interaction that the Initiator intends to cancel. The semantics of cancel should roughly be interpreted as meaning that the initiator is no longer interested in continuing the interaction and that it should be terminated in a manner acceptable to both the Initiator and the Participant. The Participant either informs the Initiator that the interaction is done using an `inform-done` or indicates the failure of the cancellation using a `failure`.

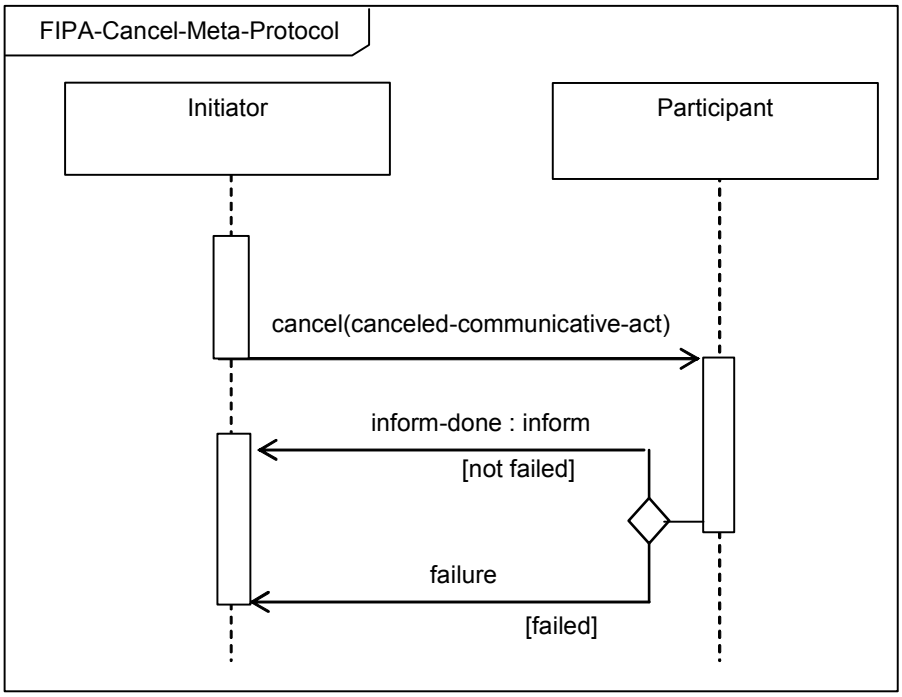

Figure 2: FIPA Cancel Meta-Protocol

This IP is a pattern for a simple interaction type. Elaboration on this pattern will almost certainly be necessary in order to specify all cases that might occur in an actual agent interaction. Real world issues such as the effects of cancelling actions, asynchrony, abnormal or unexpected IP termination, nested IPs, and the like, are explicitly not addressed here.

2 References

[FIPA00037] FIPA Communicative Act Library Specification. Foundation for Intelligent Physical Agents, 2000. http://www.fipa.org/specs/fipa00037/

[Odell2001] Odell, James, Van Dyke Parunak, H. and Bauer, B., *Representing Agent Interaction Protocols in UML*. In: Agent-Oriented Software Engineering, Ciancarini, P. and Wooldridge, M., Eds., Springer, pp. 121-140, Berlin, 2001. http://www.fipa.org/docs/input/f-in-00077/

3 Informative Annex A — ChangeLog

3.1 2002/11/01 - version G by TC X2S

Page 1, line 42:	Reworked and expanded the section description of the IP
Page 2, Figure 1:	The communication labeled `inform-ref` was changed to `inform-result` for clarity; the purpose of this communication is to inform the initiator of a result and `inform-result` implies `inform-done`
Page 2, Figure 1:	The `not-understood` communication was removed
Page 2, Figure 1:	To conform to UML 2, the protocol name was placed in a boundary, x is removed from the diamonds (xor is now the default) and the template box was removed
Page 2, line 72:	Added a new section on Explanation of Protocol Flow
Page 2, line 72:	Reworked and expanded the section on Exceptions of Protocol Flow to incorporate a meta-protocol for cancel
Page 2, line 72:	Added a paragraph explaining the `not-understood` communication and its relationship with the IP

3.2 2002/12/03 - version H by FIPA Architecture Board

Entire document:	Promoted to Standard status

FOUNDATION FOR INTELLIGENT PHYSICAL AGENTS

FIPA Iterated Contract Net Interaction Protocol Specification

Document title	FIPA Iterated Contract Net Interaction Protocol Specification		
Document number	SC00030H	**Document source**	FIPA TC Communication
Document status	Standard	**Date of this status**	2002/12/03
Supersedes	None		
Contact	fab@fipa.org		
Change history	See *Informative Annex A — ChangeLog*		

http://www.fipa.org/
Geneva, Switzerland

Foreword

The Foundation for Intelligent Physical Agents (FIPA) is an international organization that is dedicated to promoting the industry of intelligent agents by openly developing specifications supporting interoperability among agents and agent-based applications. This occurs through open collaboration among its member organizations, which are companies and universities that are active in the field of agents. FIPA makes the results of its activities available to all interested parties and intends to contribute its results to the appropriate formal standards bodies where appropriate.

The members of FIPA are individually and collectively committed to open competition in the development of agent-based applications, services and equipment. Membership in FIPA is open to any corporation and individual firm, partnership, governmental body or international organization without restriction. In particular, members are not bound to implement or use specific agent-based standards, recommendations and FIPA specifications by virtue of their participation in FIPA.

The FIPA specifications are developed through direct involvement of the FIPA membership. The status of a specification can be either Preliminary, Experimental, Standard, Deprecated or Obsolete. More detail about the process of specification may be found in the FIPA Document Policy [f-out-00000] and the FIPA Specifications Policy [f-out-00003]. A complete overview of the FIPA specifications and their current status may be found on the FIPA Web site.

FIPA is a non-profit association registered in Geneva, Switzerland. As of June 2002, the 56 members of FIPA represented many countries worldwide. Further information about FIPA as an organization, membership information, FIPA specifications and upcoming meetings may be found on the FIPA Web site at http://www.fipa.org/.

Contents

1 FIPA Iterated Contract Net Interaction Protocol

The FIPA Iterated Contract Net Interaction Protocol (IP) is an extension of the basic FIPA Contract Net IP (see [FIPA00029]), but it differs by allowing multi-round iterative bidding.

The representation of this IP is given in *Figure 1* which is based on extensions to UML1.x. [Odell2001]. This protocol is identified by the token `fipa-iterated-contract-net` as the value of the `protocol` parameter of the ACL message.

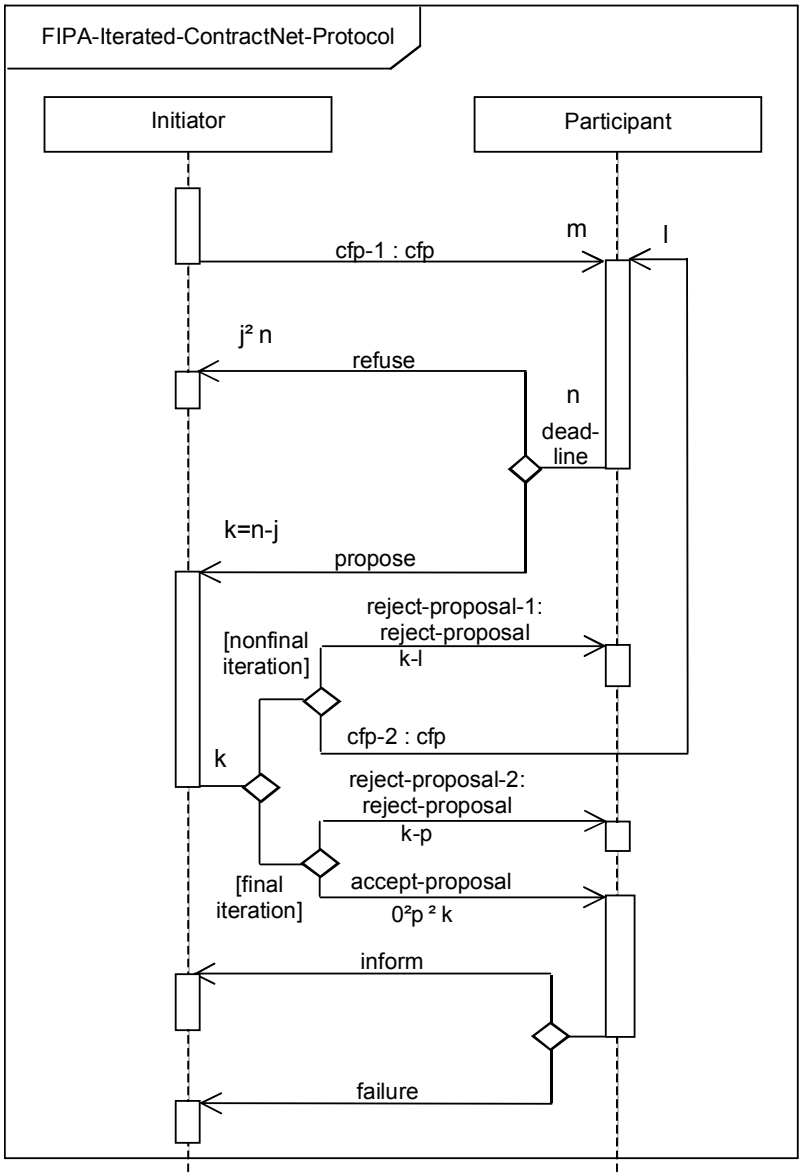

Figure 1: FIPA Iterated Contract Net Interaction Protocol

1.1 Explanation of the Protocol Flow

As with the FIPA Contract Net IP, the Initiator issues *m* initial call for proposals with the `cfp` act (see [FIPA00037]). Of the *n* Participants that respond, *k* are `propose` messages (see [FIPA00037]) from Participants that are willing and able to do the task under the proposed conditions and the remaining *j* are from Participants that `refuse`.

Of the *k* proposals, the Initiator may decide this is the final iteration and accept *p* of the bids *(0 p k)*, and reject the others. Alternatively the Initiator may decide to iterate the process by issuing *a* revised `cfp` to *l* of the Participants and rejecting the remaining *k-l* Participants. The intent is that the Initiator seeks to get better bids from the Participants by modifying the call and requesting new (equivalently, revised) bids. The process terminates when the Initiator refuses all proposals and does not issue a new `cfp`, the Initiator accepts one or more of the bids or the Participants all refuse to bid.

Any interaction using this interaction protocol is identified by a globally unique, non-null `conversation-id` parameter, assigned by the Initiator. The agents involved in the interaction must tag all of its ACL messages with this conversation identifier. This enables each agent to manage its communication strategies and activities, for example, it allows an agent to identify individual conversations and to reason across historical records of conversations.

In the case of 1:N interaction protocols or sub-protocols the Initiator is free to decide if the same `conversation-id` parameter should be used or a new one should be issued. Additionally, the messages may specify other interaction-related information such as a timeout in the `reply-by` parameter that denotes the latest time by which the sending agent would like to have received the next message in the protocol flow.

1.2 Exceptions to Interaction Protocol Flow

At *any* point in the IP, the receiver of a communication can inform the sender that it did not understand what was communicated. This is accomplished by returning a `not-understood` message. As such, *Figure 1* does not depict a `not-understood` communication as it can occur at any point in the IP. The communication of a `not-understood` within an interaction protocol may terminate the entire IP and termination of the interaction may imply that any commitments made during the interaction are null and void. However, since this IP broadcasts to more than one Participant, multiple responses are also possible. Each response, then, must be evaluated separately – and some of these responses might be `not-understood`. However, terminating the entire IP in this case might not be appropriate, as other Participants may be continuing with their sub-protocols.

At any point in the IP, the initiator of the IP may cancel the interaction protocol by initiating the meta-protocol shown in *Figure 2*. The `conversation-id` parameter of the cancel interaction is identical to the `conversation-id` parameter of the interaction that the Initiator intends to cancel. The semantics of cancel should roughly be interpreted as meaning that the initiator is no longer interested in continuing the interaction and that it should be terminated in a manner acceptable to both the Initiator and the Participant. The Participant either informs the Initiator that the interaction is done using an `inform-done` or indicates the failure of the cancellation using a `failure`.

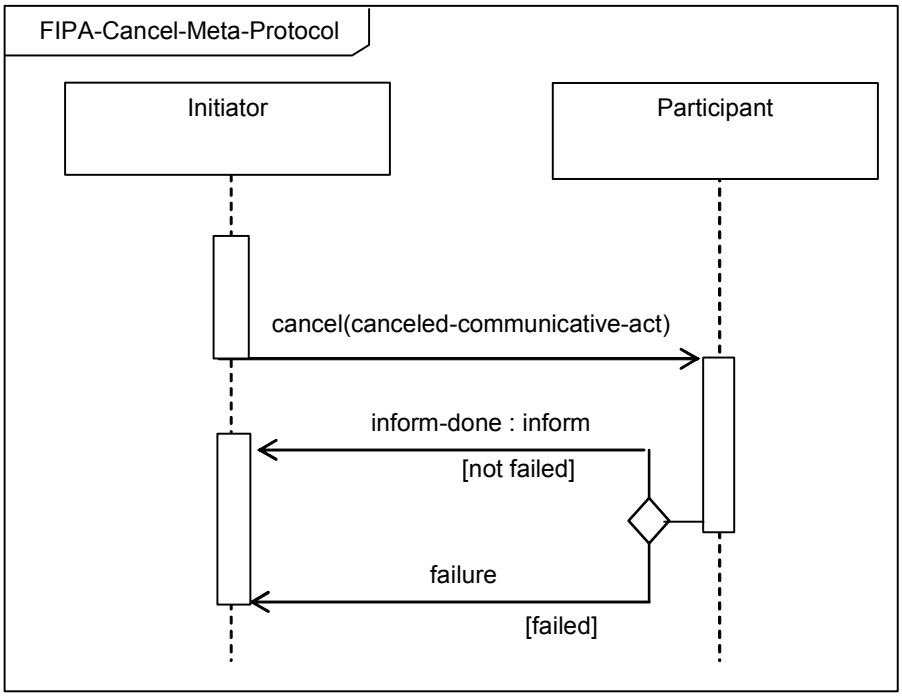

Figure 2: FIPA Cancel Meta-Protocol

This IP is a pattern for a simple interaction type. Elaboration on this pattern will almost certainly be necessary in order to specify all cases that might occur in an actual agent interaction. Real world issues such as the effects of cancelling actions, asynchrony, abnormal or unexpected IP termination, nested IPs, and the like, are explicitly not addressed here.

2 References

[FIPA00029] FIPA Contract Net Interaction Protocol Specification. Foundation for Intelligent Physical Agents, 2000. http://www.fipa.org/specs/fipa00029/

[FIPA00037] FIPA Communicative Act Library Specification. Foundation for Intelligent Physical Agents, 2000. http://www.fipa.org/specs/fipa00037/

[Odell2001] Odell, James, Van Dyke Parunak, H. and Bauer, B., *Representing Agent Interaction Protocols in UML*. In: Agent-Oriented Software Engineering, Ciancarini, P. and Wooldridge, M., Eds., Springer, pp. 121-140, Berlin, 2001. http://www.fipa.org/docs/input/f-in-00077/

3 Informative Annex A — ChangeLog

3.1 2002/11/01 - version G by TC X2S

Page 1, Figure 1:	The `not-understood` communication was removed
Page 1, Figure 1:	To conform to UML 2, the protocol name was placed in a boundary, x is removed from the diamonds (xor is now the default) and the template box was removed
Page 1, line 42:	Reworked and expanded the section description of the IP
Page 1, line 57:	Added a new section on Explanation of Protocol Flow
Page 1, line 57:	Reworked and expanded the section on Exceptions of Protocol Flow to incorporate a meta-protocol for cancel
Page 1, line 57:	Added a paragraph explaining the `not-understood` communication and its relationship with the IP

3.2 2002/12/03 - version H by FIPA Architecture Board

Entire document:	Promoted to Standard status

FOUNDATION FOR INTELLIGENT PHYSICAL AGENTS

FIPA Brokering Interaction Protocol Specification

Document title	FIPA Brokering Interaction Protocol Specification		
Document number	SC00033H	**Document source**	FIPA TC Communication
Document status	Standard	**Date of this status**	2002/12/03
Supersedes	None		
Contact	fab@fipa.org		
Change history	See *Informative Annex A — ChangeLog*		

http://www.fipa.org/
Geneva, Switzerland

Foreword

The Foundation for Intelligent Physical Agents (FIPA) is an international organization that is dedicated to promoting the industry of intelligent agents by openly developing specifications supporting interoperability among agents and agent-based applications. This occurs through open collaboration among its member organizations, which are companies and universities that are active in the field of agents. FIPA makes the results of its activities available to all interested parties and intends to contribute its results to the appropriate formal standards bodies where appropriate.

The members of FIPA are individually and collectively committed to open competition in the development of agent-based applications, services and equipment. Membership in FIPA is open to any corporation and individual firm, partnership, governmental body or international organization without restriction. In particular, members are not bound to implement or use specific agent-based standards, recommendations and FIPA specifications by virtue of their participation in FIPA.

The FIPA specifications are developed through direct involvement of the FIPA membership. The status of a specification can be either Preliminary, Experimental, Standard, Deprecated or Obsolete. More detail about the process of specification may be found in the FIPA Document Policy [f-out-00000] and the FIPA Specifications Policy [f-out-00003]. A complete overview of the FIPA specifications and their current status may be found at the FIPA Web site.

FIPA is a non-profit association registered in Geneva, Switzerland. As of June 2002, the 56 members of FIPA represented many countries worldwide. Further information about FIPA as an organization, membership information, FIPA specifications and upcoming meetings may be found on the FIPA Web site at http://www.fipa.org/.

Contents

1 FIPA Brokering Interaction Protocol

The FIPA Brokering Interaction Protocol (IP) is designed to support brokerage interactions in mediated systems and in multi-agent systems, for example, [Finin97].

Generally speaking, a broker is an agent that offers a set of communication facilitation services to other agents using some knowledge about the requirements and capabilities of those agents. A typical example of brokering is one in which an agent can request a broker to find one or more agents who can answer a query. The broker then determines a set of appropriate agents to which to forward the query, sends the query to those agents and relays their answers back to the original requestor. The use of brokerage agents can significantly simplify the task of interaction with agents in a multi-agent system. Additionally, brokering agents also enable a system to be adaptable and robust in dynamic situations, supporting scalability and security control at the brokering agent.

The representation of this IP is given in *Figure 1* which is based on an extension of UML 1.x. [Odell2001]. This protocol is identified by the token `fipa-brokering` as the value of the `protocol` parameter of the ACL message.

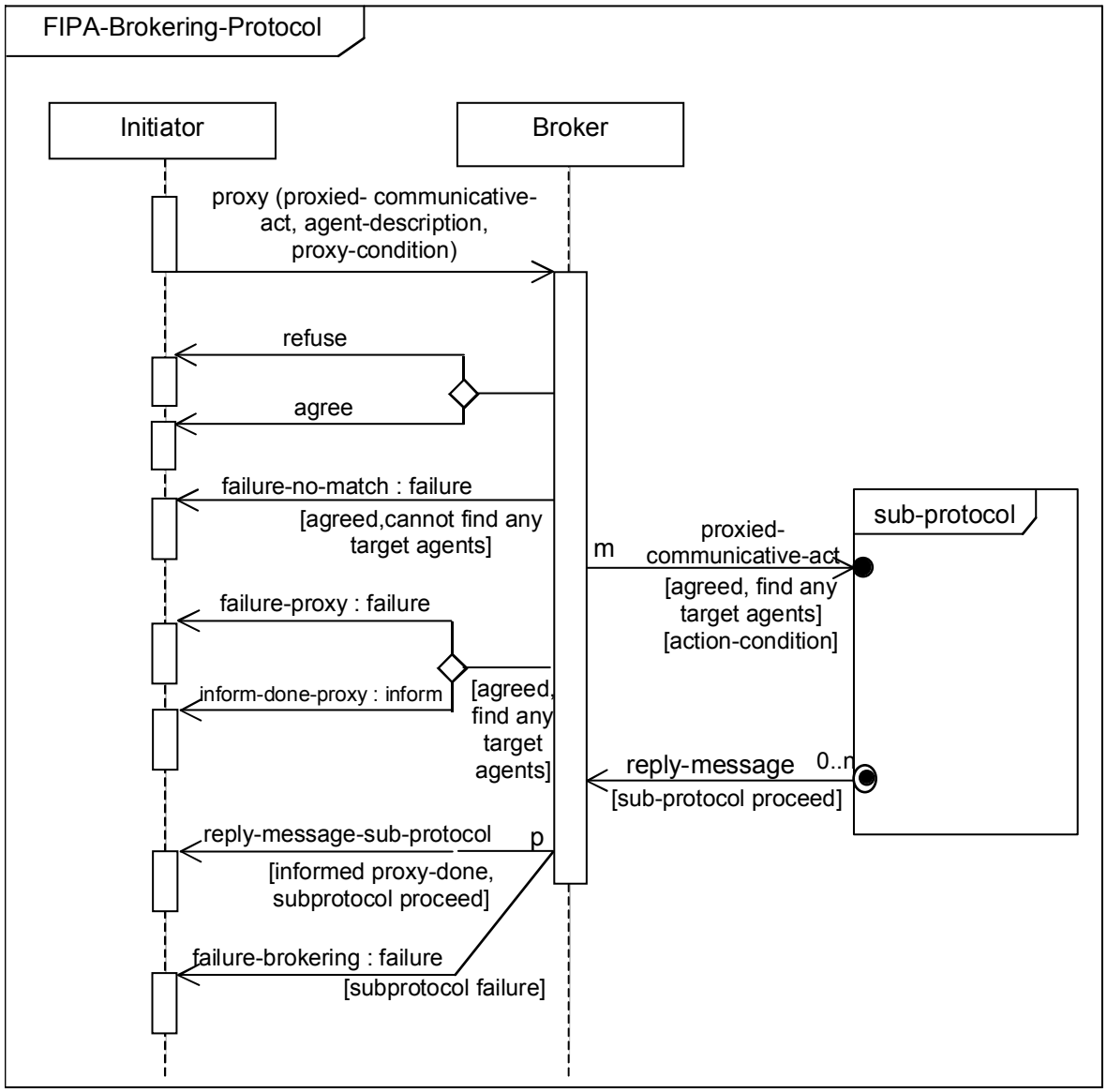

Figure 1: FIPA Brokering Interaction Protocol

1.1 Explanation of the Interaction Protocol Flow

The FIPA Brokering Interaction Protocol (IP) is a macro IP since the `proxy` communicative act (see [FIPA00037]) for brokerage embeds a communicative act as its argument and so the IP for the embedded communicative act is also embedded in this IP. This embedded IP guides some parts of the remainder of the interaction, thus parts of this protocol are written very generically.

The Initiator of the brokering interaction begins the interaction with a `proxy` message which contains the following: a referential expression denoting the target agents to which the broker should forward the communicative act, the communicative act to forward and a set of proxy conditions such as the maximum number of agents to which the message should be forwarded. The Broker processes the request and makes a decision whether to agree to or refuse the request and communicates either an `agree` or a `refuse` communicative act accordingly. Communication of a `refuse` terminates the interaction.

Once the Broker has agreed to be a proxy, it then locates agents per the description from the `proxy` message. If no such agents can be found, the Broker returns a `failure-no-match` and the interaction terminates. Otherwise, the Broker may modify the list of matching agents based on the `proxy-condition` parameter. It then begins *m* interactions with the resulting list of *n* agents with each interaction in its own separate sub-protocol. At this point, the Broker should record some of the ACL parameters (see [FIPA00061]), for example, `conversation-id, reply-with` and `sender`, of the received `proxy` message to return in the *r* replies to the Initiator.

Note that the nature of the sub-protocol and the nature of the replies are driven by the interaction protocols specified in the communicative act from the `proxy` message. As the sub-protocol progresses, the Broker forwards the responses that it receives from the sub-protocol to the Initiator. These messages are defined as the `reply-message-sub-protocol` communications, and may be either successful replies as defined by the sub-protocol or `failure`. If the initial proxy was an `inform`, there may in fact be no replies from the sub-protocol (and in fact means that the interaction is identical to a recruited `inform`). When the sub-protocol completes, the Broker forwards the final `reply-message` from the sub-protocol and the brokering IP terminates. However, there can be other failures that are not explicitly returned from the sub-protocol, for example, the agent that is executing the sub-protocol has failed. If the Broker detects such problems, it returns a `failure-brokering`, which terminates the IP.

A second issue to address occurs because multiple agents may match and therefore multiple sub-protocols (*m* of them) may be initiated by the Broker within the brokering IP. In this case, the Broker may collect the *n* received responses and combine them into a single `reply-message-sub-protocol`, or may forward the `reply-message-sub-protocol` messages from the separate sub-protocols individually (*1 p n.*). This is complicated by situations such as one agent responding with a `failure` while a second agent returns a `reply-message`, or the situation where results are inconsistent. The Broker must determine whether to resolve such situations internally or forward the responses to the Initiator. In doing this, the Broker must also be careful to avoid disruptive acts such as directly forwarding a `failure` from a sub-protocol, which would have the inadvertent effect of ending the brokering IP.

Any interaction using this interaction protocol is identified by a globally unique, non-null `conversation-id` parameter, assigned by the Initiator. The agents involved in the interaction must tag all of its ACL messages with this conversation identifier. This enables each agent to manage its communication strategies and activities, for example, it allows an agent to identify individual conversations and to reason across historical records of conversations.

In the case of 1:N interaction protocols or sub-protocols the Initiator is free to decide if the same `conversation-id` parameter should be used or a new one should be issued. Additionally, the messages may specify other interaction-related information such as a timeout in the `reply-by` parameter that denotes the latest time by which the sending agent would like to have received the next message in the protocol flow.

1.2 Exceptions to Interaction Protocol Flow

At *any* point in the IP, the receiver of a communication can inform the sender that it did not understand what was communicated. This is accomplished by returning a `not-understood` message. As such, *Figure 1* does not depict a `not-understood` communication as it can occur at any point in the IP. The communication of a `not-understood` within an interaction protocol may terminate the entire IP and termination of the interaction may imply that any

commitments made during the interaction are null and void. However, since this IP broadcasts to more than one Participant, multiple responses are also possible. Each response, then, must be evaluated separately – and some of these responses might be `not-understood`. However, terminating the entire IP in this case might not be appropriate, as other Participants may be continuing with their sub-protocols.

At any point in the IP, the initiator of the IP may cancel the interaction protocol by initiating the meta-protocol shown in *Figure 2*. The `conversation-id` parameter of the cancel interaction is identical to the `conversation-id` parameter of the interaction that the Initiator intends to cancel. The semantics of cancel should roughly be interpreted as meaning that the initiator is no longer interested in continuing the interaction and that it should be terminated in a manner acceptable to both the Initiator and the Participant. The Participant either informs the Initiator that the interaction is done using an `inform-done` or indicates the failure of the cancellation using a `failure`.

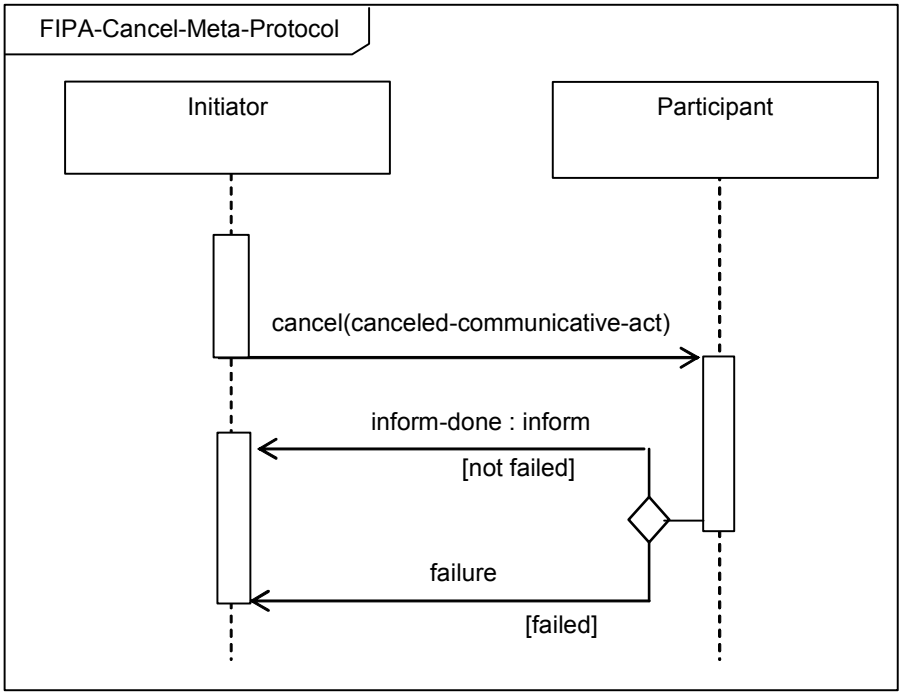

Figure 2: FIPA Cancel Meta-Protocol

This IP is a pattern for a simple interaction type. Elaboration on this pattern will almost certainly be necessary in order to specify all cases that might occur in an actual agent interaction. Real world issues such as the effects of cancelling actions, asynchrony, abnormal or unexpected IP termination, nested IPs, and the like, are explicitly not addressed here.

2 References

[Finin97] Finin, T. Labrou, Y. and Mayfield, J., *KQML as an Agent Communication Language*. In: Software Agents, Bradshaw, J., Ed., MIT Press, 1997.

[FIPA00037] FIPA Communicative Act Library Specification. Foundation for Intelligent Physical Agents, 2000. http://www.fipa.org/specs/fipa00037/

[FIPA00061] FIPA ACL Message Structure Specification. Foundation for Intelligent Physical Agents, 2000. http://www.fipa.org/specs/fipa00061/

[Odell2001] Odell, James, Van Dyke Parunak, H. and Bauer, B., *Representing Agent Interaction Protocols in UML*. In: Agent-Oriented Software Engineering, Ciancarini, P. and Wooldridge, M., Eds., Springer, pp. 121-140, Berlin, 2001. http://www.fipa.org/docs/input/f-in-00077/

3 Informative Annex A — ChangeLog

3.1 2002/11/01 - version G by TC X2S

Page 1, line 42: Reworked and expanded the section description of the IP

Page 2, Figure 1: The `not-understood` communication was removed

Page 2, Figure 1: Used a more generic set of communicative acts which the Broker is going to forward the responses it received from the sub-protocol and if the Broker notices some failure, such as no response at all from the sub-protocol after a given time period, then the Broker may send the Initiator a failure of its own

Page 2, Figure 1: Multiple sub-protocols indicated by inserting *m*, *n* and *p* respectively on three arcs; *m* sub-protocols can be started, resulting in *n* responses that the Broker can consolidate into *p* responses to the Initiator

Page 2, Figure 1: To conform to UML 2, the protocol name was placed in a boundary, x is removed from the diamonds (xor is now the default) and the template box was removed

Page 2, line 70: Added a new section on Explanation of Protocol Flow

Page 2, line 70: Reworked and expanded the section on Exceptions of Protocol Flow to incorporate a meta-protocol for cancel

Page 2, line 70: Added a paragraph explaining the `not-understood` communication and its relationship with the IP

3.2 2002/12/03 - version H by FIPA Architecture Board

Entire document: Promoted to Standard status

FOUNDATION FOR INTELLIGENT PHYSICAL AGENTS

FIPA Recruiting Interaction Protocol Specification

Document title	FIPA Recruiting Interaction Protocol Specification		
Document number	SC00034H	**Document source**	FIPA TC Communication
Document status	Standard	**Date of this status**	2002/12/03
Supersedes	None		
Contact	fab@fipa.org		
Change history	See *Informative Annex A — ChangeLog*		

http://www.fipa.org/
Geneva, Switzerland

Foreword

The Foundation for Intelligent Physical Agents (FIPA) is an international organization that is dedicated to promoting the industry of intelligent agents by openly developing specifications supporting interoperability among agents and agent-based applications. This occurs through open collaboration among its member organizations, which are companies and universities that are active in the field of agents. FIPA makes the results of its activities available to all interested parties and intends to contribute its results to the appropriate formal standards bodies where appropriate.

The members of FIPA are individually and collectively committed to open competition in the development of agent-based applications, services and equipment. Membership in FIPA is open to any corporation and individual firm, partnership, governmental body or international organization without restriction. In particular, members are not bound to implement or use specific agent-based standards, recommendations and FIPA specifications by virtue of their participation in FIPA.

The FIPA specifications are developed through direct involvement of the FIPA membership. The status of a specification can be either Preliminary, Experimental, Standard, Deprecated or Obsolete. More detail about the process of specification may be found in the FIPA Document Policy [f-out-00000] and the FIPA Specifications Policy [f-out-00003]. A complete overview of the FIPA specifications and their current status may be found on the FIPA Web site.

FIPA is a non-profit association registered in Geneva, Switzerland. As of June 2002, the 56 members of FIPA represented many countries worldwide. Further information about FIPA as an organization, membership information, FIPA specifications and upcoming meetings may be found on the FIPA Web site at http://www.fipa.org/.

Contents

1 FIPA Recruiting Interaction Protocol

The FIPA Recruiting Interaction Protocol (IP) is designed to support recruiting interactions in mediated systems and in multi-agent systems, for example, [Finin97].

A recruiter agent is a form of broker, which, generally speaking, is an agent that offers a set of communication facilitation services to other agents using some knowledge about the requirements and capabilities of those agents. A typical example of brokering is one in which an agent can request a broker to find one or more agents who can answer a query. The broker then determines a set of appropriate agents to which to forward the query and sends the query to those agents.

In the case of recruiting (as opposed to brokering), the answers from the selected target agents either go directly back to the original requestor or to some designated receivers. The use of recruiter agents can significantly simplify the task of interaction with agents in a multi-agent system. Recruiter agents also enable a system to be adaptable and robust in dynamic situations, supporting scalability and security control at the recruiting agent.

The representation of this IP is given in *Figure 1* which is based on an extension of UML 1.x. [Odell2001]. This protocol is identified by the token `fipa-recruiting` as the value of the `protocol` parameter of the ACL message.

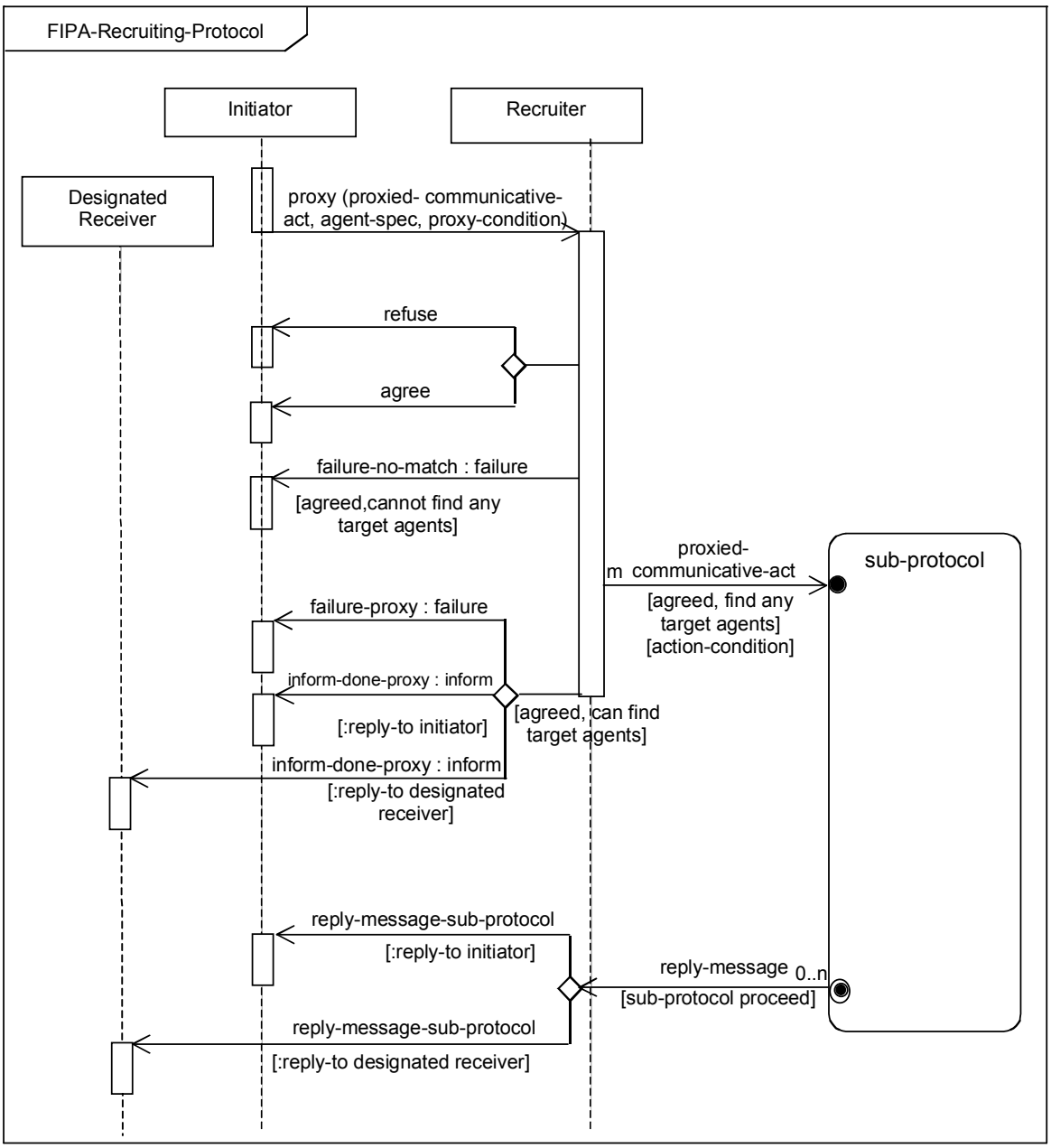

Figure 1: FIPA Recruiting Interaction Protocol

1.1 Explanation of the Interaction Protocol Flow

The FIPA Recruiting Interaction Protocol (IP) is a macro IP since the `proxy` communicative act (see [FIPA00037]) for recruiting embeds a communicative act as its argument and so the IP for the embedded communicative act is also embedded in this IP. This embedded IP guides some parts of the remainder of the interaction, thus parts of this protocol are written very generically.

The Initiator of the recruiting interaction begins the interaction with a `proxy` message which contains the following: a referential expression denoting the target agents to which the recruiter should forward the communicative act, the communicative act to forward and a set of proxy conditions such as the maximum number of agents to be forwarded. The Recruiter processes the request and makes a decision whether to agree to or refuse the request, and communicates either an `agree` or a `refuse` communicative act accordingly. Communication of a `refuse` terminates the interaction.

Once the Recruiter has agreed to be a proxy, it then locates agents per the description from the `proxy` message. If no such agents can be found, the Recruiter returns a `failure-no-match` and the interaction terminates. Otherwise, the Recruiter may modify the list of matching agents based on the `proxy-condition` parameter. It then begins *m* interactions with the resulting list of *n* agents with each interaction in its own separate sub-protocol. The initiation of the sub-protocol should be done with care, using the ACL parameters (see [FIPA00061]) to correlate the responses to the request. If the Recruiter has been given a message containing a separate `designated-receiver` parameter from the interaction Initiator, it needs to start each sub-protocol with a `reply-to` parameter containing the Designated Receiver and the `conversation-id` of the original conversation. If the Recruiter instead is to indicate that the Initiator should receive the replies, then the `reply-to` parameter should designate the Initiator and the `conversation-id` of the recruiting conversation. Other ACL parameters may also need to be propagated.

Note that the nature of the sub-protocol and the nature of the replies are driven by the interaction protocols specified in the communicative act from the proxy message. As the sub-protocol progresses, it forwards its responses back either to the Designated Receiver or to the Initiator, depending on the value of the `reply-to` parameter in the `proxy` message. These messages are defined as `reply-message-sub-protocol` communications and may be either successful replies as defined by the sub-protocol or `failure`. If the initial proxy was an `inform`, there may in fact be no replies from the sub-protocol (and in fact means that the interaction is identical to a brokered `inform`). When the sub-protocol completes, the Recruiter forwards the final `reply-message-sub-protocol` from the sub-protocol and the recruiting IP terminates.

A second issue to address occurs because multiple agents may match and therefore multiple sub-protocols may be initiated by the Recruiter within the recruiting IP. In this case, the sub-protocols may be communicating multiple `reply-message-sub-protocol` communications from the different agents involved in the IP (for a total of *m* responses). This is complicated by such situations as one sub-protocol responding with a `failure` while a second sub-protocol returns a `reply-message-sub-protocol`, or the situation where results are inconsistent. The agent that receives the messages must determine how to detect and resolve such situations internally.

Any interaction using this interaction protocol is identified by a globally unique, non-null `conversation-id` parameter, assigned by the Initiator. The agents involved in the interaction must tag all of its ACL messages with this conversation identifier. This enables each agent to manage its communication strategies and activities, for example, it allows an agent to identify individual conversations and to reason across historical records of conversations.

In the case of 1:N interaction protocols or sub-protocols the Initiator is free to decide if the same `conversation-id` parameter should be used or a new one should be issued. Additionally, the messages may specify other interaction-related information such as a timeout in the `reply-by` parameter that denotes the latest time by which the sending agent would like to have received the next message in the protocol flow.

1.2 Exceptions to Interaction Protocol Flow

At *any* point in the IP, the receiver of a communication can inform the sender that it did not understand what was communicated. This is accomplished by returning a `not-understood` message. As such, *Figure 1* does not depict a `not-understood` communication as it can occur at any point in the IP. The communication of a `not-understood` within an interaction protocol may terminate the entire IP and termination of the interaction may imply that any commitments made during the interaction are null and void. However, since this IP broadcasts to more than one Participant, multiple responses are also possible. Each response, then, must be evaluated separately – and some of these responses might be `not-understood`. However, terminating the entire IP in this case might not be appropriate, as other Participants may be continuing with their sub-protocols.

At any point in the IP, the initiator of the IP may cancel the interaction protocol by initiating the meta-protocol shown in *Figure 2*. The `conversation-id` parameter of the cancel interaction is identical to the `conversation-id` parameter of the interaction that the Initiator intends to cancel. The semantics of cancel should roughly be interpreted as meaning that the initiator is no longer interested in continuing the interaction and that it should be terminated in a manner acceptable to both the Initiator and the Participant. The Participant either informs the Initiator that the interaction is done using an `inform-done` or indicates the failure of the cancellation using a `failure`.

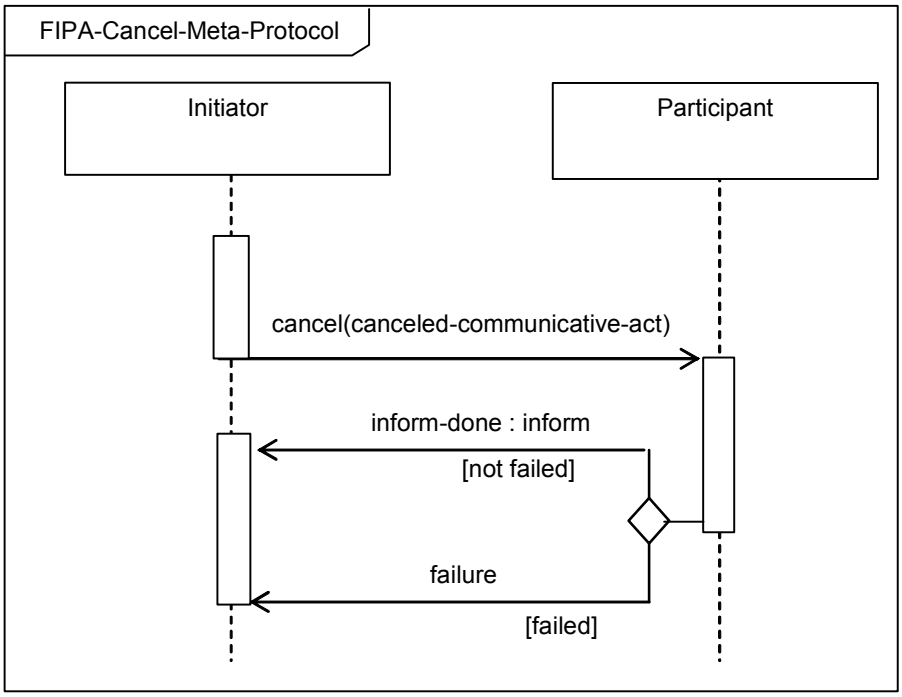

Figure 2: FIPA Cancel Meta-Protocol

This IP is a pattern for a simple interaction type. Elaboration on this pattern will almost certainly be necessary in order to specify all cases that might occur in an actual agent interaction. Real world issues such as the effects of cancelling actions, asynchrony, abnormal or unexpected IP termination, nested IPs, and the like, are explicitly not addressed here.

2 References

[Finin97] Finin, T. Labrou, Y. and Mayfield, J., *KQML as an Agent Communication Language*. In: Software Agents, Bradshaw, J., Ed., MIT Press, 1997.

[FIPA00037] FIPA Communicative Act Library Specification. Foundation for Intelligent Physical Agents, 2000. http://www.fipa.org/specs/fipa00037/

[FIPA00061] FIPA ACL Message Structure Specification. Foundation for Intelligent Physical Agents, 2000. http://www.fipa.org/specs/fipa00061/

[Odell2001] Odell, James, Van Dyke Parunak, H. and Bauer, B., *Representing Agent Interaction Protocols in UML*. In: Agent-Oriented Software Engineering, Ciancarini, P. and Wooldridge, M., Eds., Springer, pp. 121-140, Berlin, 2001. http://www.fipa.org/docs/input/f-in-00077/

3 Informative Annex A — ChangeLog

3.1 2002/11/01 - version G by TC X2S

Entire document:	Changed the name Destinator to Designated Receiver
Page 1, line 42:	Reworked and expanded the section description of the IP
Page 1, Figure 1:	The `not-understood` communication was removed
Page 2, Figure 1:	Used a more generic set of communicative acts which indicates that the sub-protocols are going to forward their responses (failure or references) to either the Initiator or the Designated Receiver
Page 2, Figure 1:	Multiple sub-protocols indicated by inserting *m* and *n* respectively on two arcs; *m* sub-protocols can be started, resulting in *n* responses
Page 2, Figure 1:	To conform to UML 2, the protocol name was placed in a boundary, x is removed from the diamonds (xor is now the default) and the template box was removed
Page 2, line 69:	Added a new section on Explanation of Protocol Flow
Page 2, line 69:	Reworked and expanded the section on Exceptions of Protocol Flow to incorporate a meta-protocol for cancel
Page 2, line 69:	Added a paragraph explaining the `not-understood` communication and its relationship with the IP

3.2 2002/12/03 - version H by FIPA Architecture Board

Entire document:	Promoted to Standard status

FOUNDATION FOR INTELLIGENT PHYSICAL AGENTS

FIPA Subscribe Interaction Protocol Specification

Document title	FIPA Subscribe Interaction Protocol Specification		
Document number	SC00035H	**Document source**	FIPA TC Communication
Document status	Standard	**Date of this status**	2002/12/03
Supersedes	None		
Contact	fab@fipa.org		
Change history	See *Informative Annex A — ChangeLog*		

http://www.fipa.org/
Geneva, Switzerland

Foreword

The Foundation for Intelligent Physical Agents (FIPA) is an international organization that is dedicated to promoting the industry of intelligent agents by openly developing specifications supporting interoperability among agents and agent-based applications. This occurs through open collaboration among its member organizations, which are companies and universities that are active in the field of agents. FIPA makes the results of its activities available to all interested parties and intends to contribute its results to the appropriate formal standards bodies where appropriate.

The members of FIPA are individually and collectively committed to open competition in the development of agent-based applications, services and equipment. Membership in FIPA is open to any corporation and individual firm, partnership, governmental body or international organization without restriction. In particular, members are not bound to implement or use specific agent-based standards, recommendations and FIPA specifications by virtue of their participation in FIPA.

The FIPA specifications are developed through direct involvement of the FIPA membership. The status of a specification can be either Preliminary, Experimental, Standard, Deprecated or Obsolete. More detail about the process of specification may be found in the FIPA Document Policy [f-out-00000] and the FIPA Specifications Policy [f-out-00003]. A complete overview of the FIPA specifications and their current status may be found on the FIPA Web site.

FIPA is a non-profit association registered in Geneva, Switzerland. As of June 2002, the 56 members of FIPA represented many countries worldwide. Further information about FIPA as an organization, membership information, FIPA specifications and upcoming meetings may be found on the FIPA Web site at http://www.fipa.org/.

Contents

1 FIPA Subscribe Interaction Protocol

The FIPA Subscribe Interaction Protocol (IP) allows an agent to request a receiving agent to perform an action on subscription and subsequently when the referenced object changes.

The representation of this IP is given in *Figure 1* which is based on an extension of UML 1.x. [Odell2001]. This protocol is identified by the token `fipa-subscribe` as the value of the `protocol` parameter of the ACL message.

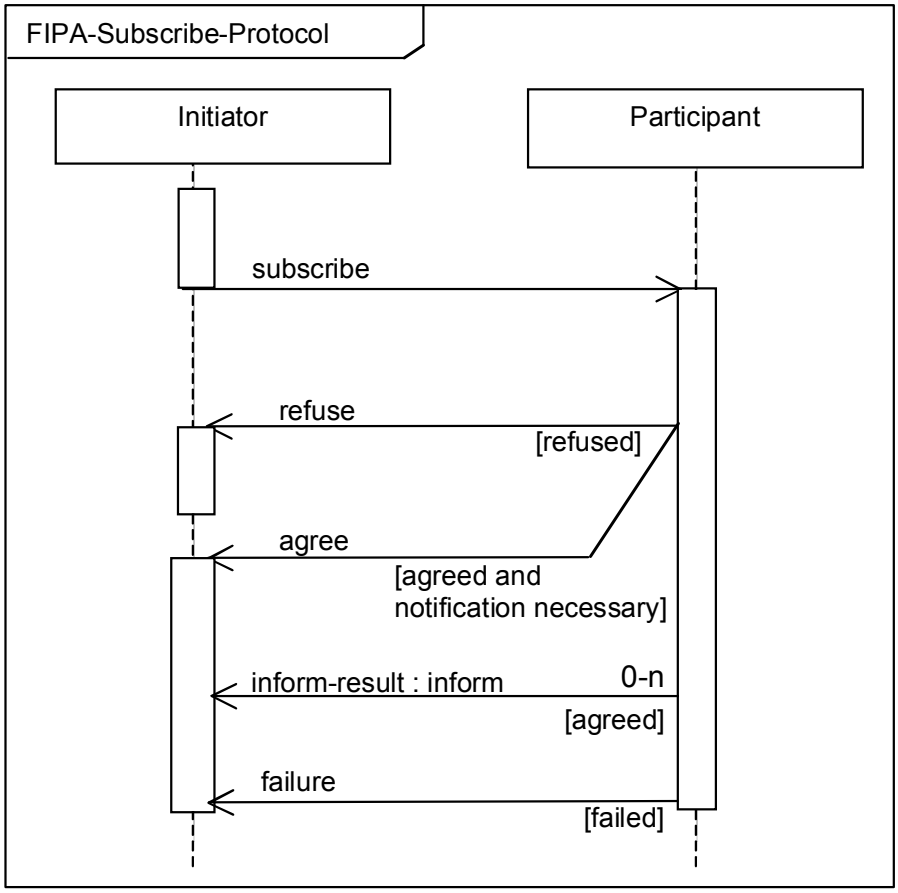

Figure 1: FIPA Subscribe Interaction Protocol

1.1 Explanation of the Protocol Flow

The Initiator begins the interaction with a subscribe message containing the reference of the objects in which they are interested. The Participant processes the `subscribe` message and makes a decision whether to accept or refuse the query request. If the Participant makes a refuse decision, then “refused” becomes true and the Participant communicates a `refuse`. Otherwise, ”agreed” becomes true.

If conditions indicate that an explicit agreement is required (that is, “notification necessary” is true), then the Participant communicates an `agree`. The `agree` may be optional depending on circumstances, for example, if the requested action is very quick and can happen before a time specified in the `reply-by` parameter.

In a successful response, the Participant replies with an `inform-result` communication with the content being a referring expression to the subscribed objects. The Participant continues to send `inform-result` messages as the objects denoted by the referring expression change. If at some point after the Participant agrees, it experiences a

failure, then it communicates this with a `failure` message, which also terminates the interaction. Otherwise, the interaction may be terminated by the Initiator using the cancel meta-protocol as described in Section 1.2.

Any interaction using this interaction protocol is identified by a globally unique, non-null `conversation-id` parameter, assigned by the Initiator. The agents involved in the interaction must tag all of its ACL messages with this conversation identifier. This enables each agent to manage its communication strategies and activities, for example, it allows an agent to identify individual conversations and to reason across historical records of conversations. Additionally, because it may be important to preserve the sequence of the `inform-result` messages, it is important that the message transport used for this IP preserve the ordering of messages.

1.2 Exceptions to Interaction Protocol Flow

At *any* point in the IP, the receiver of a communication can inform the sender that it did not understand what was communicated. This is accomplished by returning a `not-understood` message. As such, *Figure 1* does not depict a `not-understood` communication as it can occur at any point in the IP. The communication of a `not-understood` within an interaction protocol may terminate the entire IP and termination of the interaction may imply that any commitments made during the interaction are null and void.

At any point in the IP, the initiator of the IP may cancel the interaction protocol by initiating the meta-protocol shown in *Figure 2*. The `conversation-id` parameter of the cancel interaction is identical to the `conversation-id` parameter of the interaction that the Initiator intends to cancel. The semantics of cancel should roughly be interpreted as meaning that the initiator is no longer interested in continuing the interaction and that it should be terminated in a manner acceptable to both the Initiator and the Participant. The Participant either informs the Initiator that the interaction is done using an `inform-done` or indicates the failure of the cancellation using a `failure`.

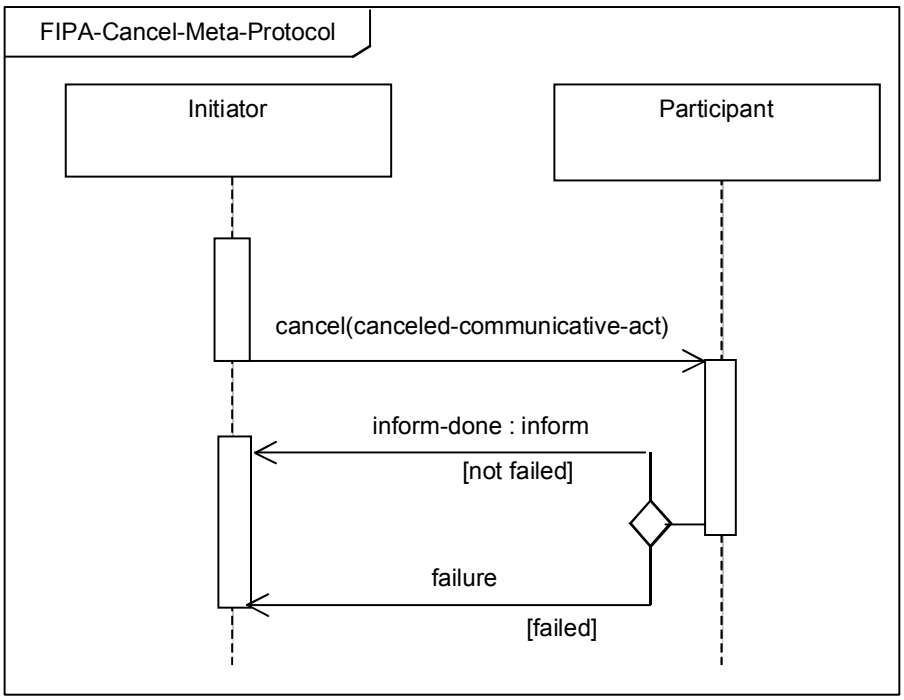

Figure 2: FIPA Cancel Meta-Protocol

This IP is a pattern for a simple interaction type. Elaboration on this pattern will almost certainly be necessary in order to specify all cases that might occur in an actual agent interaction. Real world issues such as the effects of cancelling actions, asynchrony, abnormal or unexpected IP termination, nested IPs, and the like, are explicitly not addressed here.

2 References

[FIPA00037] FIPA Communicative Act Library Specification. Foundation for Intelligent Physical Agents, 2000. http://www.fipa.org/specs/fipa00037/

[Odell2001] Odell, James, Van Dyke Parunak, H. and Bauer, B., *Representing Agent Interaction Protocols in UML*. In: Agent-Oriented Software Engineering, Ciancarini, P. and Wooldridge, M., Eds., Springer, pp. 121-140, Berlin, 2001. http://www.fipa.org/docs/input/f-in-00077/

3 Informative Annex A — ChangeLog

3.1 2002/11/01 - version G by TC X2S

Page 1, Figure 1:	The `not-understood` communication was removed
Page 1, Figure 1:	Reworked the protocol to insert an optional `agree`
Page 1, Figure 1:	Deleted the explicit cancel from the protocol diagram because it has been moved to the meta-protocol section
Page 1, Figure 1:	Added guards to the diagram to indicate that the protocol may be terminated by reaching the end of the conversation-length
Page 1, Figure 1:	To conform to UML 2, the protocol name was placed in a boundary, x is removed from the diamonds (xor is now the default) and the template box was removed
Page 1, line 42:	Reworked and expanded the section description of the IP
Page 1, line 54:	Added a new section on Explanation of Protocol Flow
Page 1, line 54:	Reworked and expanded the section on Exceptions of Protocol Flow to incorporate a meta-protocol for cancel
Page 1, line 54:	Added a paragraph explaining the `not-understood` communication and its relationship with the IP

3.2 2002/12/03 - version H by FIPA Architecture Board

Entire document:	Promoted to Standard status

FOUNDATION FOR INTELLIGENT PHYSICAL AGENTS

FIPA Propose Interaction Protocol Specification

Document title	FIPA Propose Interaction Protocol Specification		
Document number	SC00036H	**Document source**	FIPA TC Communication
Document status	Standard	**Date of this status**	2002/12/03
Supersedes	None		
Contact	fab@fipa.org		
Change history	See *Informative Annex A — ChangeLog*		

http://www.fipa.org/
Geneva, Switzerland

Foreword

The Foundation for Intelligent Physical Agents (FIPA) is an international organization that is dedicated to promoting the industry of intelligent agents by openly developing specifications supporting interoperability among agents and agent-based applications. This occurs through open collaboration among its member organizations, which are companies and universities that are active in the field of agents. FIPA makes the results of its activities available to all interested parties and intends to contribute its results to the appropriate formal standards bodies where appropriate.

The members of FIPA are individually and collectively committed to open competition in the development of agent-based applications, services and equipment. Membership in FIPA is open to any corporation and individual firm, partnership, governmental body or international organization without restriction. In particular, members are not bound to implement or use specific agent-based standards, recommendations and FIPA specifications by virtue of their participation in FIPA.

The FIPA specifications are developed through direct involvement of the FIPA membership. The status of a specification can be either Preliminary, Experimental, Standard, Deprecated or Obsolete. More detail about the process of specification may be found in the FIPA Document Policy [f-out-00000] and the FIPA Specifications Policy [f-out-00003]. A complete overview of the FIPA specifications and their current status may be found on the FIPA Web site.

FIPA is a non-profit association registered in Geneva, Switzerland. As of June 2002, the 56 members of FIPA represented many countries worldwide. Further information about FIPA as an organization, membership information, FIPA specifications and upcoming meetings may be found on the FIPA Web site at http://www.fipa.org/.

Contents

1 FIPA Propose Interaction Protocol

The FIPA Propose Interaction Protocol (IP) allows an agent to propose to receiving agents that the initiator will do the actions described in the `propose` communicative act (see [FIPA00037]) when the receiving agent accepts the proposal.

The representation of this IP is given in *Figure 1* which is based on an extension of UML 1.x. [Odell2001]. This protocol is identified by the token `fipa-propose` as the value of the `protocol` parameter of the ACL message.

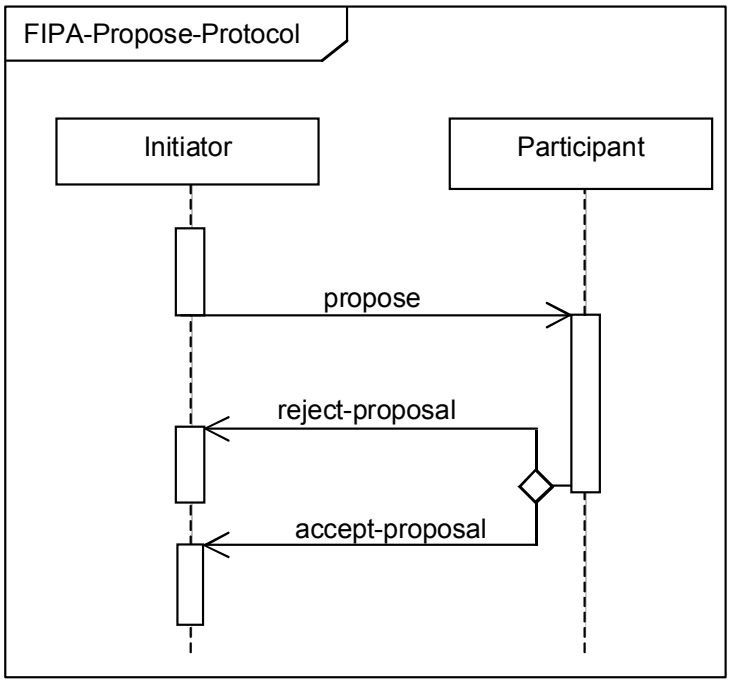

Figure 1: FIPA Propose Interaction Protocol

1.1 Explanation of the Interaction Protocol Flow

The Initiator sends a `propose` message to the Participant indicating that it will perform some action if the Participant agrees. The Participant responds by either accepting or rejecting the proposal, communicating this with the `accept-proposal` or `reject-proposal` communicative act, accordingly. Completion of this IP with an `accept-proposal` act (see [FIPA00037]) would typically be followed by the performance by the Initiator of the proposed action and then the return of a status response.

Any interaction using this interaction protocol is identified by a globally unique, non-null `conversation-id` parameter, assigned by the Initiator. The agents involved in the interaction must tag all of its ACL messages with this conversation identifier. This enables each agent to manage its communication strategies and activities, for example, it allows an agent to identify individual conversations and to reason across historical records of conversations.

1.2 Exceptions to Interaction Protocol Flow

At *any* point in the IP, the receiver of a communication can inform the sender that it did not understand what was communicated. This is accomplished by returning a `not-understood` message. As such, *Figure 1* does not depict a `not-understood` communication as it can occur at any point in the IP. The communication of a `not-understood` within an interaction protocol may terminate the entire IP and termination of the interaction may imply that any commitments made during the interaction are null and void.

At any point in the IP, the initiator of the IP may cancel the interaction protocol by initiating the meta-protocol shown in *Figure 2*. The `conversation-id` parameter of the cancel interaction is identical to the `conversation-id` parameter of the interaction that the Initiator intends to cancel. The semantics of cancel should roughly be interpreted as meaning that the initiator is no longer interested in continuing the interaction and that it should be terminated in a manner acceptable to both the Initiator and the Participant. The Participant either informs the Initiator that the interaction is done using an `inform-done` or indicates the failure of the cancellation using a `failure`.

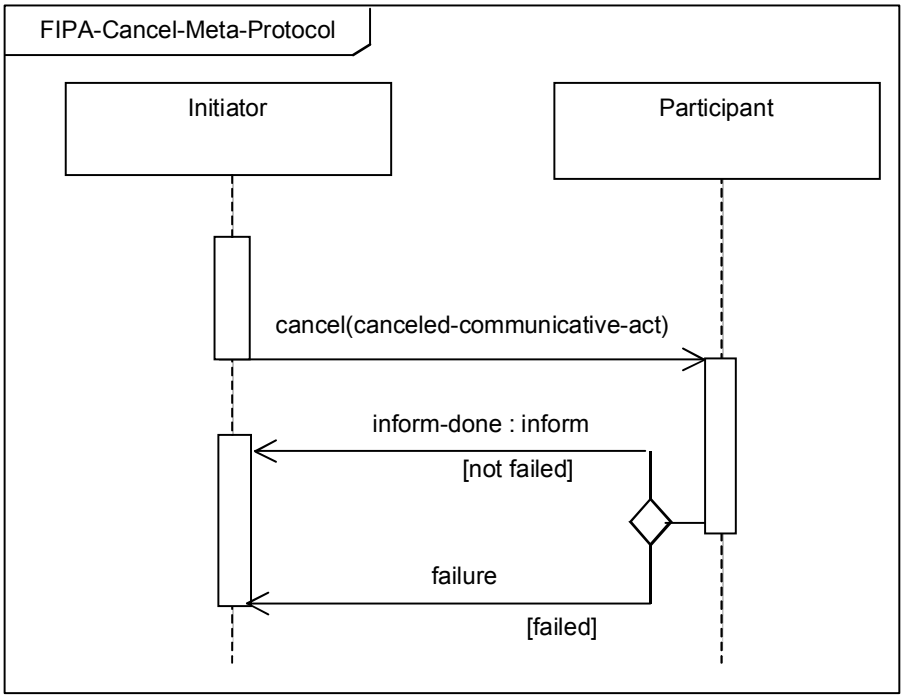

Figure 2: FIPA Cancel Meta-Protocol

This IP is a pattern for a simple interaction type. Elaboration on this pattern will almost certainly be necessary in order to specify all cases that might occur in an actual agent interaction. Real world issues such as the effects of cancelling actions, asynchrony, abnormal or unexpected IP termination, nested IPs, and the like, are explicitly not addressed here.

2 References

[FIPA00037] FIPA Communicative Act Library Specification. Foundation for Intelligent Physical Agents, 2000. http://www.fipa.org/specs/fipa00037/

[Odell2001] Odell, James, Van Dyke Parunak, H. and Bauer, B., *Representing Agent Interaction Protocols in UML*. In: Agent-Oriented Software Engineering, Ciancarini, P. and Wooldridge, M., Eds., Springer, pp. 121-140, Berlin, 2001. http://www.fipa.org/docs/input/f-in-00077/

3 Informative Annex A — ChangeLog

3.1 2002/11/01 - version G by TC X2S

Page 1, Figure 1: The `not-understood` communication was removed
Page 1, line 42: Reworked and expanded the section description of the IP
Page 1, line 54: Added a new section on Explanation of Protocol Flow
Page 1, line 54: Reworked and expanded the section on Exceptions of Protocol Flow to incorporate a meta-protocol for cancel
Page 1, line 54: Added a paragraph explaining the `not-understood` communication and its relationship with the IP

3.2 2002/12/03 - version H by FIPA Architecture Board

Entire document: Promoted to Standard status