1. **Overview of Inter-Agent Communication**

History suggests that agents and agent systems are typically implemented with a greater variety of interface mechanisms; existing example agents include those using TCP/IP sockets, HTTP, SMTP and GSM short messages.

Currently, the ACL imposes a minimal set of requirements on the message transport service, as shown below:

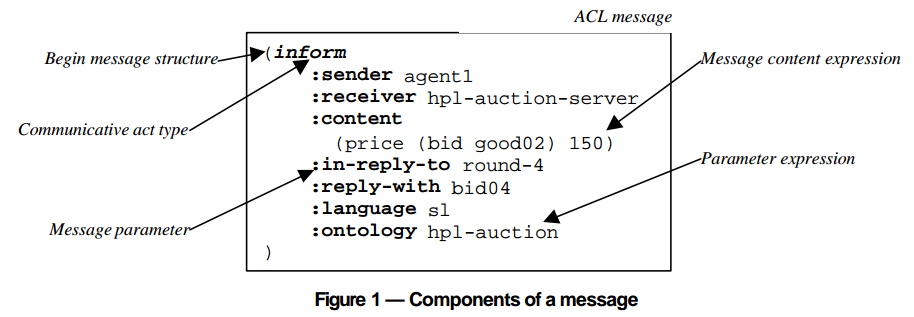
* The message service is able to deliver a message, encoded in the transport form below, to a destination as a sequence of bytes. The message service exposes through its interface whether it is able to cope reliably with 8-bit bytes whose high-order bit may be set.
* The normal case is that the message service is *reliable* (well-formed messages will arrive at the destination) *accurate* (the message is received in the form in which it was sent), and *orderly* (messages from agent a to agent b arrive at b in the order in which they were sent from a). Unless informed otherwise, an agent is entitled to assume that these properties hold.
* If the message delivery service is unable to guarantee any or all of the above properties, this fact is exposed in some way through the interface to the message delivery service
* An agent will have the option of selecting whether it suspends and waits for the result of a message (synchronous processing) or continues with other unrelated tasks while waiting for a message reply (asynchronous processing). The availability of this behaviour will be implementation specific, but it must be made explicit where either behaviour is not supported.
* Parameters of *the act of delivering a message*, such as time-out if no reply, are not codified at the message level but are part of the interface exposed by the message delivery service.
* The message delivery service will detect and report error conditions, such as: ill-formed message, undeliverable, unreachable agent, etc., back to the sending agent. Depending on the error condition, this may be returned either as a return value from the message sending interface, or through the delivery of an appropriate error message.
* An agent has a name which will allow the message delivery service to deliver the message to the correct destination. The message delivery service will be able to determine the correct transport mechanism (TCP/IP, SMTP, http, etc.), and will allow for changes in agent location, as necessary

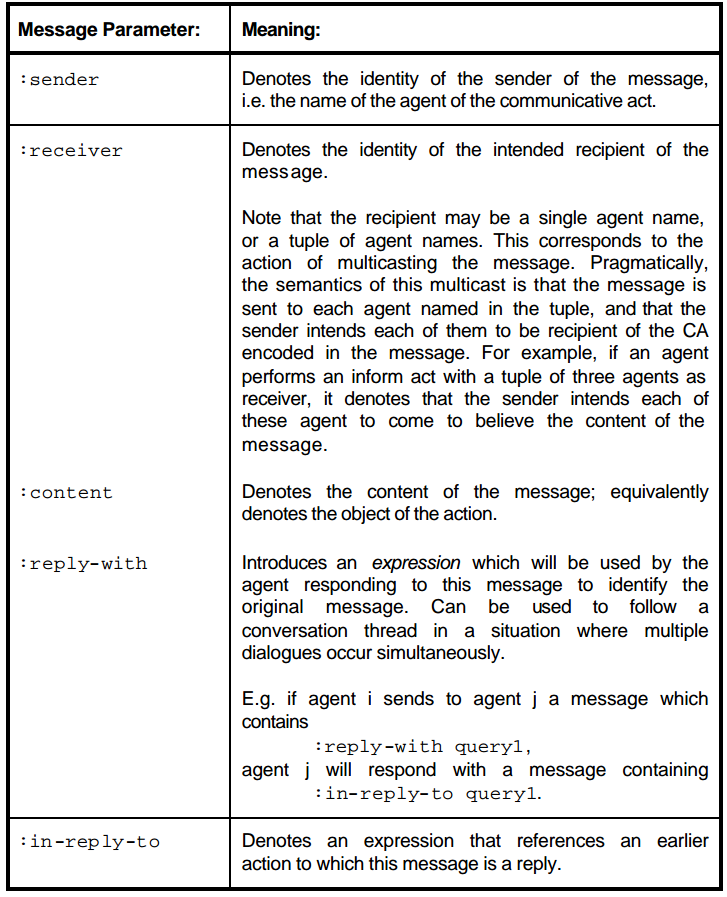
1. **FIPA ACL Messages**

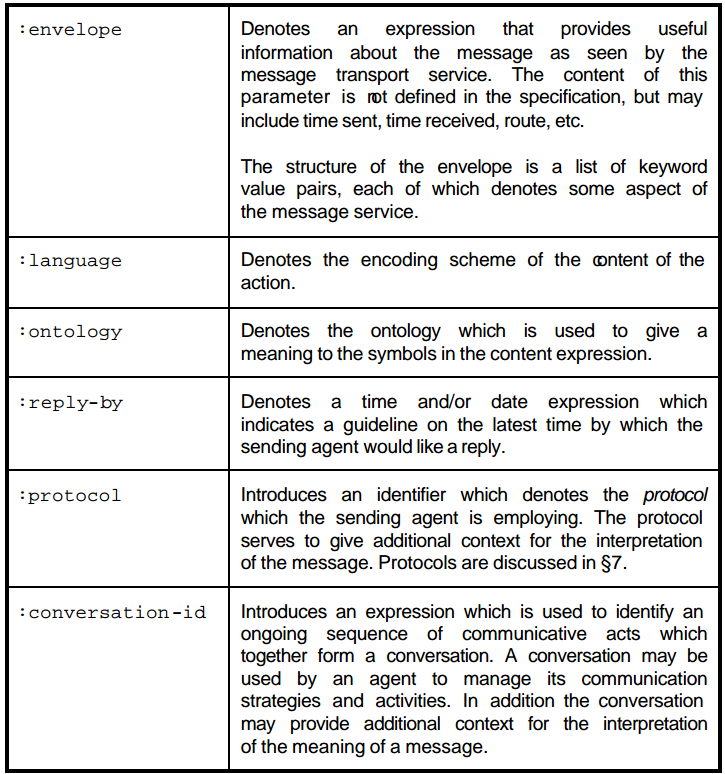
**Requirements on agents**

* Requirement 1: Agents should send not-understood if they receive a message that they do not recognise or they are unable to process the content of the message. Agents must be prepared to receive and properly handle a not-understood message from other agents.
* Requirement 2: An ACL compliant agent may choose to implement any subset (including all, though this is unlikely) of the predefined message types and protocols. The implementation of these messages must be correct with respect to the referenced act's semantic definition.
* Requirement 3: An ACL compliant agent which uses the communicative acts whose names are defined in this specification must implement them correctly with respect to their definition.
* Requirement 4: Agents may use communicative acts with other names, not defined in this document, and are responsible for ensuring that the receiving agent will understand the meaning of the act. However, agents should not define new acts with a meaning that matches a pre-defined standard act.
* Requirement 5: An ACL compliant agent must be able to correctly generate a syntactically well formed message in the transport form that corresponds to the message it wishes to send. Symmetrically, it must be able to translate a character sequence that is well-formed in the transport syntax to the corresponding message.

**Message structure**







**Message content**

* **Requirement 6:** 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. More specifically, the content of a message must express the data type of the action: propositions for inform, actions for request, etc.
* **Requirement 7:** A compliant agent is required to exercise the standard agent management capabilities through the use of messages using the agent management content language and ontology. The language and ontology are each denoted by the reserved term fipa-agent-management in their respective parameters.

**Notes on grammar rules**

* The standard definitions for integers and floating point numbers are assumed.
* All keywords are case-insensitive.
* A length encoded string is a context sensitive lexical token. Its meaning is as follows: the header of the token is everything from the leading "#" to the separator " inclusive. Between the markers of the header is a decimal number with at least one digit. This digit then determines that exactly that number of 8-bit bytes are to be consumed as part of the token, without restriction. It is a lexical error for less than that number of bytes to be available.

Note that not all implementations of the agent communication channel (ACC) [see Part One of the FIPA 97 specification] will support the transparent transmission of 8-bit characters. It is the responsibility of the agent to ensure, by reference to the API provided for the ACC, that a given channel is able to faithfully transmit the chosen message encoding.

* A well-formed message will obey the grammar, and in addition, will have at most one of each of the parameters. It is an error to attempt to send a message which is not well formed. Further rules on well-formed messages may be stated or implied the operational definitions of the values of parameters as these are further developed.
* Strings encoded in accordance with ISO/IEC 2022 may contain characters which are otherwise not permitted in the definition of Word. These characters are ESC (0x1B), SO (0x0E) and SI (0x0F). This is due to the complexity that would result from including the full ISO/IEC 2022 grammar in the above EBNF description. Hence, despite the basic description above, a word may contain any well-formed ISO/IEC 2022 encoded character, other (representations of) parentheses, spaces, or the “#” character. Note that parentheses may legitimately occur as part of a well formed escape sequence; the preceding restriction on characters in a word refers only to the encoded characters, not the form of the encoding.
* Time tokens are based on the ISO 8601 format, with extensions for relative time and millisecond durations. Time expressions may be absolute, or relative to the current time. Relative times are distinguished by the character "+" appearing as the first character in the construct. If no type designator is given, the local timezone is used. The type designator for UTC is the character "Z". UTC is preferred to prevent timezone ambiguities. Note that years must be encoded in four digits. As examples, 8:30 am on April 15th 1996 local time would be encoded as:

19960415T083000000

the sametime in UTC would be:

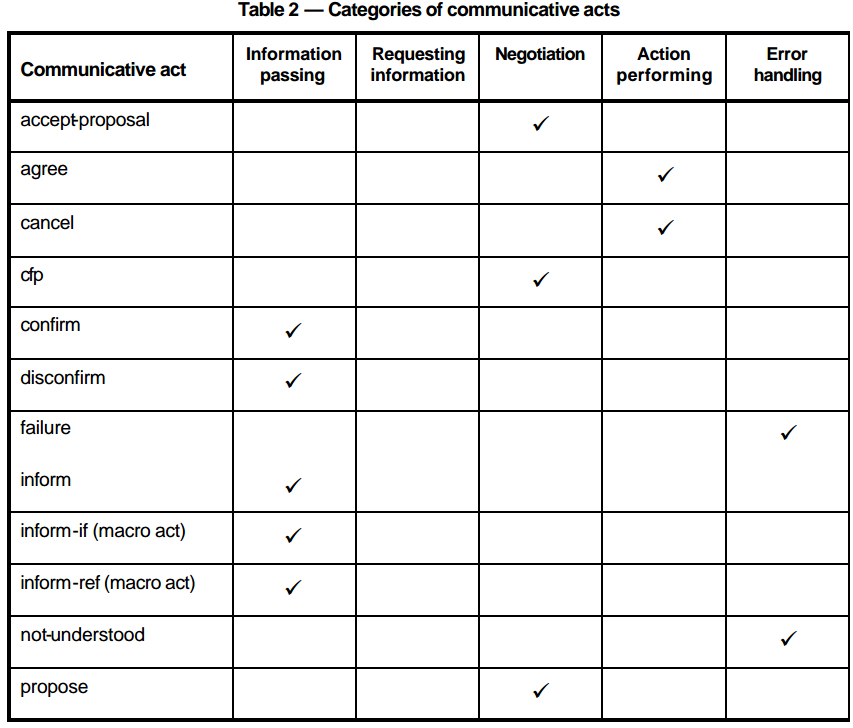
19960415T083000000Z

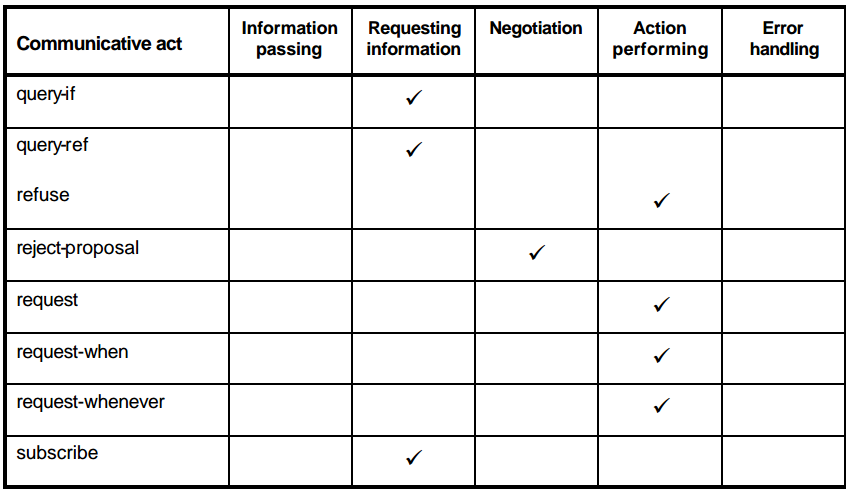
while one hour, 15 minutes and 35 milliseconds from now would be:

+00000000T011500035.

* The format defined for agent names is taken from part one of the FIPA 97 standard. The option of simply using a word as the agent name is only valid where that word can be unambiguously resolved to an full agent name in the format given.

**Catalogue of Communicative Acts**





1. **Interaction Protocols**

**Requirement 8:** An ACL compliant agent need not implement any of the standard protocols, nor is it restricted from using other protocol names. However, if one of the standard protocol names is used, the agent must behave consistently with the protocol specification given here.

**Defined protocols**

* Failure to understand a response during a protocol: a protocol, by convention an agent which is expecting a certain set of responses in a protocol, and which receives another message not in that set, should respond with a not-understood message
* FIPA-request Protocol: The FIPA-request protocol simply allows one agent to request another to perform some action, and the receiving agent to perform the action or reply, in some way, that it cannot.
* FIPA-query Protocol: In the FIPA-query protocol, the receiving agent is requested to perform some kind of inform action. Requesting to inform is a query, and there are two query-acts: query-if and query-ref.
* FIPA-request-when Protocol: The FIPA-request-when protocol is simply an expression of the full intended meaning of the request-when action. The requesting agent uses the request-when action to seek from the requested agent that it performs some action in the future once a given precondition becomes true.
* FIPA-contract-net Protocol: IPA-Contract-Net is a minor modification of the original contract net protocol in that it adds rejection and confirmation communicative acts. In the contract net protocol, one agent takes the role of manager. The manager wishes to have some task performed by one or more other agents, and further wishes to optimise a function that characterises 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.
* FIPA-Iterated-Contract-Net Protocol: The iterated contract net protocol is an extension of the basic contract net as described above. It differs from the basic version of the contract net by allowing multi-round iterative bidding.
* FIPA-Auction-English Protocol:
* FIPA-Auction-Dutch Protocol: