

FOUNDATION FOR INTELLIGENT PHYSICAL AGENTS

FIPA Personal Travel Assistance Specification

Document title	FIPA Personal Travel Assistance Specification		
Document number	PC00080	Document source	FIPA Architecture Board
Document status	Preliminary	Date of this status	2000/06/15
Supersedes	OC00013		
Contact	fab@fipa.org		
Change history			
2000/06/15	Carried forward from FIPA 1997 Specification 4 V1.0		

© 2000 Foundation for Intelligent Physical Agents - <http://www.fipa.org/>

Geneva, Switzerland

Notice

Use of the technologies described in this specification may infringe patents, copyrights or other intellectual property rights of FIPA Members and non-members. Nothing in this specification should be construed as granting permission to use any of the technologies described. Anyone planning to make use of technology covered by the intellectual property rights of others should first obtain permission from the holder(s) of the rights. FIPA strongly encourages anyone implementing any part of this specification to determine first whether part(s) sought to be implemented are covered by the intellectual property of others, and, if so, to obtain appropriate licenses or other permission from the holder(s) of such intellectual property prior to implementation. This specification is subject to change without notice. Neither FIPA nor any of its Members accept any responsibility whatsoever for damages or liability, direct or consequential, which may result from the use of this specification.

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.

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 Procedures for Technical Work. A complete overview of the FIPA specifications and their current status may be found in the FIPA List of Specifications. A list of terms and abbreviations used in the FIPA specifications may be found in the FIPA Glossary.

FIPA is a non-profit association registered in Geneva, Switzerland. As of January 2000, the 56 members of FIPA represented 17 countries worldwide. Further information about FIPA as an organization, membership information, FIPA specifications and upcoming meetings may be found at <http://www.fipa.org/>.

Contents

1	Scope.....	1
2	General Analysis.....	2
2.1	Introduction.....	2
2.2	Problem Statements.....	3
2.3	Business Domain Analysis.....	4
2.4	Actors and Roles.....	5
2.4.1	Travel Service Agent.....	5
2.4.2	Travel Broker Agent.....	5
2.4.3	Personal Travel Assistant.....	6
2.4.4	Mini-Personal Travel Assistant.....	6
2.5	Scenario.....	6
2.6	External Software Integration.....	7
2.7	Internal Software Integration.....	7
2.8	Internal Capabilities.....	8
2.9	Human-Agent Interface.....	8
2.10	Agent Management.....	9
3	Architecture.....	10
3.1	Service Architectures and Protocols.....	10
3.2	Agent Definitions.....	10
3.2.1	Mini-Personal Travel Agent.....	10
3.2.2	Personal Travel Agent.....	11
3.2.3	Travel Broker.....	11
3.2.4	Tourist Office Broker.....	12
3.2.5	Flight Service Provider.....	12
3.3	Agent Platform Profiles.....	13
3.3.1	Small Company Agent Platform.....	13
3.3.2	Travel Broker Agent Platform.....	13
3.3.3	Agent Hotel Platform (On-Trip Execution).....	14
3.3.4	Domain Structures.....	14
4	Personal Travel Assistant Ontology.....	16
4.1	Object Descriptions.....	16
4.1.1	Trip Summary.....	17
4.1.2	Location.....	17
4.1.3	Travel Time.....	18
4.1.4	Budget.....	18
4.1.5	General Preferences.....	18
4.1.6	Common Travel Preferences.....	20
4.1.7	Individual Travel Preferences.....	20
4.1.8	Trip Details.....	21
4.1.9	Common Travel Segment.....	21
4.1.10	Individual Travel Segment.....	21
4.1.11	Service Point.....	22
4.1.12	Service Link.....	22
4.1.13	Geographic Data File Node.....	22
4.1.14	Geographic Data File Link.....	23
4.1.15	Plan Evaluation.....	23
4.2	Function Descriptions.....	23
4.2.1	Reserve a Segment.....	24
4.2.2	Cancel a Segment.....	24
4.2.3	Purchase a Segment.....	24
4.2.4	Modify a Segment.....	24
4.2.5	Evaluate a Plan.....	25

4.3	Elaboration of the User Profile	25
4.4	Exceptions	26
4.4.1	Failure Exception Propositions	26
5	Study Cases	27
5.1	Pre-Trip Planning.....	27
5.2	Elaboration of Pre-Trip Planning.....	31
5.3	Last Minute Auction for Lower Fare.....	33
5.4	On-Trip Execution	35
5.5	Travel Plan Monitoring.....	37
6	Agent/Software Integration.....	38
6.1	Web-Based Fare Wrapper.....	38
6.1.1	Registration of a Wrapper Agent	38
6.1.2	Request for a Price.....	38
6.1.3	Notification of Price Change.....	39
6.2	BAYERNInfo Service Wrapper	39
6.2.1	Agent Request for a Route	39
7	Future PTA Developments	41
7.1	Travelling Users.....	41
7.1.1	Agent Mobility in a Network: Travel Planning.....	41
7.1.2	Traveller Mobility: Travel Monitoring	41
7.1.3	Traveller Mobility: Travel Monitoring Via UMTS.....	41
7.2	Inter-Operation Between Agents and Workflow.....	42
8	References.....	43

1 Scope

This document extends the FIPA standard by providing an application specification for the travel industry. This specification provides:

- An overview of the current industry in regard to agents,
- A reference architecture for a multi-agent system in this industry,
- Examples of the agent management details such as domains and naming,
- Examples of agent communication details such as content ontologies and communication protocols, and,
- Examples of agent/software integration such as for accessing databases and mobile users

This specification is not complete, but the included examples help to illustrate the use of FIPA standard and thereby quicken the development and deployment of real systems. Some points of this architecture have been selected as normative in order to begin inter-operability tests of field trials. These requirements are noted throughout the specification as they arise.

In summary, this specification serves three purposes:

1. To continue testing the FIPA technical specifications. The context of a real application serves to determine the strengths and weaknesses of the specifications,
2. To demonstrate the real business value and requirement of a standard specification for such a large, distributed, multi-vendor application, and,
3. To define initial application architecture, object design and use case analysis for actual development of field trials.

2 General Analysis

2.1 Introduction

A wide variety of travel related services are becoming increasingly available through electronic means. There is a need for convenient and ready access to these services, in particular for travellers. This presents a prime example to showcase the benefits of agent technology. Agents operating on behalf of their users can provide assistance in the pre-trip planning phase, as well as during the on-trip execution phase of a trip. A system supporting these services is called a Personal Travel Assistant (PTA) system (see *Figure 1*).

In order to accomplish this assistance, these agents will interact with the user and with other agents representing the available travel services. The agent system is responsible for the configuration and delivery, including the right time, cost, quality of service and appropriate security and privacy measures, of trip planning and guidance services (for example, multi-modal route planning, hotel and parking-lot reservations, individualised traffic guidance, cartography services, tourism information, plane reservation, metro guidance, weather conditions, public transportation, special events, etc.). Further, there is interaction with other supporting agents such as media agents, directory services (yellow and white pages) and information brokers that seek, evaluate and deliberate on information.

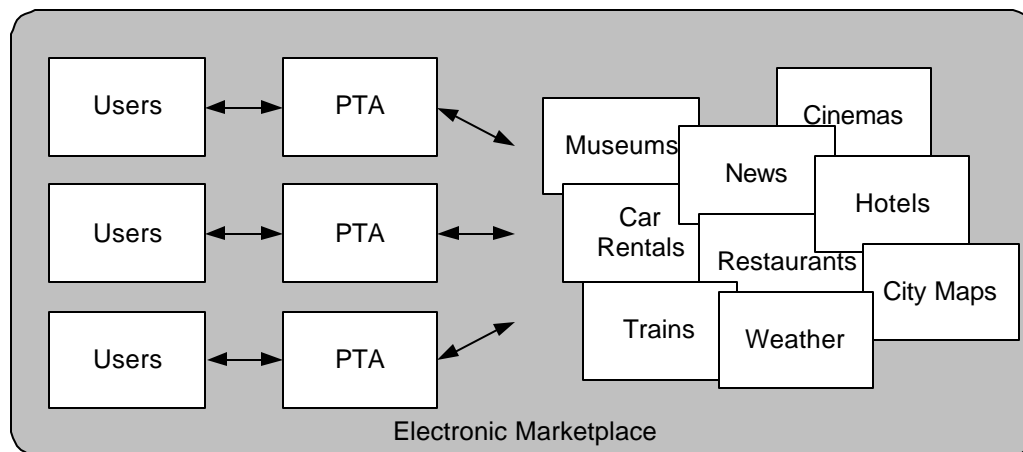


Figure 1: A Scene from FIPA Enabling Applications

The PTA system should support the following core functionalities:

- Different modes for request/response; the user does not need to be connected while a request completed,
- Composition of services; the system should provide an integrated experience even though the component services are disparate,
- Comparison of service offerings; the system should evaluate and provide the user with different service dimensions such as cost or other user's experience,
- Learning the user profile; the system should become more efficient toward the user's needs and habits with continued experience,
- Interoperability of communication means; the same underlying services should be available through many different media such as voice-phone, pager, e-mail and screen-phones,
- Administration of agents; the system and user will need the ability to follow-up agents or otherwise change their behaviour at any time,
- Alerts; the user should be notified of significant events, and,

- Negotiation and transactions; the system should act on behalf of the user to make deals and commit to purchases, for example.

This list of functions includes connectivity to basic services such as email as well as emerging services in e-commerce such as advertising and web casting. The PTA domain is rich with many basic and emerging possibilities, but for focus in this document, two test scenarios are developed, which represent the two basic phases of agent support:

- Pre-trip planning; the activities made in preparation for a trip, such as booking flights and hotels, and,
- On-trip execution; the activities required during a trip for successful execution, such as monitoring the schedule and making changes to bookings as required.

Focusing on these primary scenarios, this specification includes an overall outline of the agent types and roles and the software and devices required for both phases. For instance, on-trip execution introduces the potential use of PDAs and the agent's attachments to cellular or GSM-based phones and GPS services. Other secondary scenarios are included in this specification to demonstrate other aspects of the FIPA specifications; for instance, parts of an agent's lifecycle and special focus of mobility will be included.

Travel is an excellent application to demonstrate because it includes so many external attachments that are of interest to many other applications. For instance, the Travel scenario will include:

- **Information Retrieval**
Travel services provide both database and Web-based access and search.
- **Scheduling**
Travel not only includes scheduling within its own domain, travel schedules must also interact with personal calendars and schedules. Calendar tools, e-mail, and other general office applications are required.
- **End-user Mobility**
Not to be confused with agent mobility, travel implies several mobile device modalities and problems of communication in connected/disconnected states.
- **Agent Mobility**
Because of user mobility, agent mobility is often indicated for the transfer of binary or script code through the network.

Moreover, the travel scenario includes very strong testing of agent-to-agent attachment and the internal capacities to support different agent roles. For instance, the following agent-based technologies are also of very general interest:

- **Combined or Competitive Services**
Compare attributes, negotiate cost and time .
- **User Profiling**
Personal preferences and adaptive user modelling.

The latter issue is not directly addressed by the FIPA standard, but is critical to travel and several other end-user driven applications.

2.2 Problem Statements

The application of agents to the Travel industry exposes some very important problems now being faced by agent developers and applications in many other industries as well:

- **Web-based and Database-Based Publication**
As the travel service providers move from database to web-based pricing, for instance, agent developers are faced with the problems of HTML parsing. While this method is workable, it is very sensitive to minor and peripheral format

changes. All agents of all vendors must spend a great deal of effort to maintain the agents' proper attachment. Both the database-based and Web-based content can include "agentised" mediation. Aside from some re-publication issues, one or a few agent-based services can parse and otherwise "logicise" the raw data, offering this service to other agents. Other solutions, such as XML tags for ontology and content are very sympathetic to agent development, and future Web-based service providers might directly provide the agent-based service as well, but in any case, other agents from other vendors should rely on a well-founded communication standard at the level of agents.

- **Complexity of Market (De)Regulations**

Travel policy (especially in world-wide travel) is complex and often un-known to human travel agents. These policies are highly distributed, from corporate policy to agency policy to national and international law. The representation and use of such policies is a fairly straight-forward knowledge engineering task. A distributed agent approach seems required to partition the problem and allow different vendors to provide different parts of the solution so that every agent in the system needs not carry all the responsibility.

- **Complexity of Real World Transactions**

Travel planning is really a "super-transaction" of many negotiations. A service cannot merely find low fare, because lower fare is only one of many hard and soft constraints. A transaction cannot be based or concluded only for flight arrangements, because hotel, car, and many personal arrangements must also be established. To provide real value, a service should also be suggestive and beyond the direct travel needs and the PTA services should collectively provide the end user with a complete travel package, not just the minimal travel documents. It should contribute for market expansion into other segments.

This last problem suggests the need to co-ordinate the transactions using agent-based protocols such as Contract Net and internal technologies such as incremental scheduling. Because these are very specialised techniques, the FIPA design philosophies for agent software integration and agent interaction provide a solution by distributing the responsibilities; PTA is a very large and difficult problem, best solved by vendor specialists in internal agent technologies, external software domains, and agent-to-agent protocols that can work together.

To summarise, the PTA services should provide an effective test bed of the technology-oriented normative parts of the FIPA standard.

2.3 Business Domain Analysis

The viewpoint on business domain analysis of the Travel industry is based on a system focus; on the purpose, scope and policies for the system. It can be modelled in terms of objects representing user roles, business and management policies. This viewpoint is concerned with the overall environment in which a system is to operate. In the case of this specification, it spans co-operating organisations. In general, *Figure 2* represents the separate business domains.

This model can be used as a framework for:

- **Analysing the organisational environment**

This mainly includes network operators, service providers and customers. Which actors are involved and how do they relate to each other, that is, their roles, their domains of activity, the inter-domain policies (security, billing) and what are the interactions between the system and the environment in which it is placed?

- **Defining the requirements of actors**

For instance, what are the requirements between customers with respect to providers, that is, contractual relationships properties such as security aspects, payment and quality of service?

In each role, an actor performs different types of provisioning activities. Identifying these helps distinguish between different parts of an organisation and can indicate the types business and management support required.

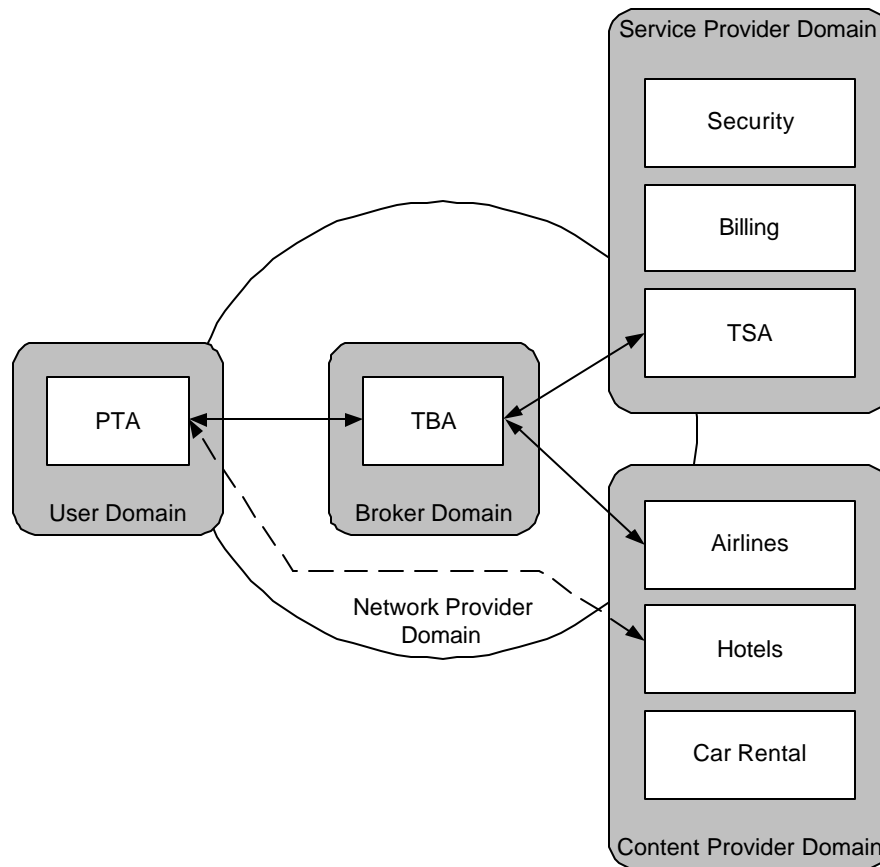


Figure 2: Relationships Between Business Domains

2.4 Actors and Roles

2.4.1 Travel Service Agent

The Travel Service Agent (TSA) is responsible for attachment to the data of their domain. The scope of each domain is arbitrary, but each such agent would tend to specialise in global flight plans and hotel arrangements or local hotel, car and restaurant information, for example. Other services might specialise in tourism or restaurants, for example, but globally. In either case, providing such "soft" added value about museums, theme parks and special events/offers should be a strong part of agent co-operation to build a more complete travel plan for the user.

In all cases, this agent type is responsible for maintaining the data access, interpretation and delivery to other agents. Such agents would typically use search services, too, in order to keep themselves up to date or to provide integrated searches within the a travel domain to other agents. Any such agent service might be implemented as a "wrapper" around legacy databases or WWW page content (see [FIPA00079]). New services can be directly wrapped by an agent, but this distinction would be transparent to other agents.

2.4.2 Travel Broker Agent

The Travel Broker Agent (TBA) is responsible to locating and contracting with TSAs. It can obtain the travel options from several services, filter and select from the alternatives, and legally bind a contract and travel documents based on a final selection. It can schedule and incrementally reschedule the entire travel plan across several service types such as flight, train, hotel and special events.

This agent type provides its service to any "anonymous" user. In other words, its service connection with the user is only for the life of the super transaction; it does not serve as the personal agent to any one user and does not keep any persistent information about particular users, aside from its own auditing/logging needs.

2.4.3 Personal Travel Assistant

The PTA acts on behalf of a user and is legally authorised to do so, up to the level allowed by the user. While conceptually seen as one Personal Assistant (PA) for each user, the implementation should be assumed to use a multi-user, server-based design. This agent type has many similarities to a personal assistant and might simply be a cast in a similar role. This agent is responsible for remembering and following the user's instructions and learning the user's preferences based on choices or feedback after the trip.

2.4.4 Mini-Personal Travel Assistant

The Mini-Personal Travel Assistant (MPTA) is a lightweight agent that is typically device-dependent, such as an agent operating on a PDA or laptop computer. For instance, bandwidth and modality become special issues. Although this tends to cause a restriction of functionality, many additional functions such as GPS and GSM can be provided in this context.

Some assumptions about these responsibilities might be changed or elaborated. For instance, the TBA might maintain some of the personal information of users, such as simple travel preferences (airline seating, smoking or non-smoking). Also, value-added services can be provided by many different arrangements. For instance, the communication of the MPTA can be various. Does the user/MPTA contact the TBA directly on the road or always through the PTA? Can the user directly contact the TBA? Is the PTA really a sub-function of a PA (like a personal secretary)?

Each project will determine the answers to these questions, but for initial field trials of FIPA standards, this specification will assume that TBA will interact with PAs (see [FIPA00083]) and that the PA will take the role of PTA. In either case, the following scenario is primary for such field trials.

2.5 Scenario

The typical dialogue between real users and travel agencies will be used as a guiding metaphor:

1. The user asks their secretary to make travel reservations for the next day. The user delegates the task to the agent. The agent is generally autonomous and bothers the user only for confirmation or in exception conditions. Time constraints for completion of this task might be explicitly stated or assumed according to the travel attributes or personal preferences (for example, past history).
2. The secretary calls a travel agency. In the simplest case, the user's company might be pre-contracted with only one agency or the secretary might have some choice, but only within a list of approved and registered agencies. The assumption here is that there is some sort of accreditation or professional membership that ensures and suggests competency of the agency.
3. The travel agency contacts several providers of services to build a complete plan. The travel agent maintains a dialogue with the secretary, who has a better sense of the user, validates how the travel documents should be delivered, etc.
4. The secretary reports back to the user with a plan, options and additional information. The secretary places the schedule with some travel information on the user's calendar, perhaps also setting reminders for when the user should leave to catch the flight.

2.6 External Software Integration

These different agent types have varying levels of integration to external software and/or other agents. For instance, the TSA's responsibilities are most for attachment to data sources, whereas a TBA's function is more abstract and more responsible to managing agent interactions. The *Table 1* lists external software attachments.

Agent Type	Possible Software Attachments
Travel Service Agent	Existing Travel Database Services HTTP/HTML Broadcast Protocols (for example, RDS, DAB, ...) Search Service
Travel Broker Agent	Yellow Pages Directory (for example, LDAP) White Pages Directory (for example, LDAP)
Personal Travel Assistant	GSM Email Calendar/Scheduling Fax E-Commerce Video Server
Mini-Personal Travel Assistant	GSM GPS/Cartography Pager

Table 1: External Software Attachments

Note that the TBA uses directory services but provides much more. More than a directory service alone, a TBA is itself an agent and can provide the negotiation and consolidation of services as an added-value. Also note how the PTA might provide travelogue video services; although a PA can also talk directly to a TBA, this is the kind of added value within a particular industry focus that a PTA can uniquely provide. This list is by no means exhaustive, but gives some idea of the integration components required and how these components might be reusable in other domains aside from travel.

2.7 Internal Software Integration

For instance, there are two approaches to integrating internal software. First, special internal engines such as for scheduling or learning can use this specification to attach such components to the agent. The internal reasoning of the agent can control other external and internal components equally. Applications can test whether or not the external wrapper interface can be used to attach internal capabilities of the agent to each other as well.

Second, any special intelligence function can be made into a first class agent that provides such scheduling or translation of learning services. This approach too should be tested with different applications and compared with the first approach.

In some regards, the two approaches are very internal components of intelligence to be viewed recursively and a large-grained agent's internal composition is a "society of minds" based on smaller, semantically simpler agents. Wrappers are much like very simple agents using a subset of communicative acts.

2.8 Internal Capabilities

Internal capabilities of agents are not the subject of FIPA standards but are important considerations for application design. *Table 2* lists the types of technology the agents are likely to require to serve each of their purposes.

Agent Type	Possible Internal Capability
Travel Service Agent	Rule-Based Inference Procedural Scripting
Travel Broker Agent	Rule-Based Policy and Planning Contract Net Protocol Rationality Acquaintance Modelling
Personal Travel Assistant	Rule Sets Preference Facts Based on User Instruction Adaptive User Model Learning
Mini-Personal Travel Assistant	Micro-Kernel capabilities Server-loadable Procedures

Table 2: Internal Agent Capabilities

TSA's have simple requirements; they typically will respond to requests for information. Simple rule based or even scripting systems for the most basic services will be typical.

TBA's are probably the most complex agents; they must adhere to industry and owner policies. They should follow a number of co-operation and negotiation protocols. This is the most appropriate place for rational agents that can understand and respond very flexibly to any number of different situations. The TBA's should maintain an acquaintance model, such as for management of long-term associations with other agents.

As for the Personal Agents, basic inference is probably appropriate, but the addition of end-user modelling (learning) will be of increasing importance in such agents. The MPTA is more peculiar since it should act much like the PTA, but given the small size of the devices it must live on, the MPTA per se needs to be minimal and rely on networking with other agents to provide its intelligence as perceived by the user. Some core capabilities will need to be installed, but aside from communications with other agents, alternative architectures employing mobile code can dynamically load the MPTA as needed.

2.9 Human-Agent Interface

While the fundamentals of human-agent and agent-agent interaction should be based on the same underlying formal dialogue model, the FIPA standard does not seem to support full application development. Particularly, there are neither standard interfaces and component definitions for supporting the graphical/text and/or voice/speech interfaces directly at the end-user, nor are there translation tools from these "natural" representations to a formal model. To compensate, the above scenario assumes a highly restrictive end-user input form, which would have to be tightly coupled to a dialogue representation.

A very important issue to consider is the "just necessary level" of user interaction. How is this established? By standard user interface controls and techniques? This problem requires specialised studies to define just necessary level: how are user preferences established and how do preferences interact with task complexity. Acceptability of the PTA and all other assistants will be based largely on matters of trust and control.

Even though human-agent dialogue tools are not specified by FIPA, this specification includes a Dialogue Wrapper which translates any software user-interface events and media applications into FIPA-compliant communicative acts and content within the agent.

2.10 Agent Management

Life cycle management (see FIPA00023) is the first concern of the PTA system, even before the system is deployed. The domain definitions, agent naming and registrations must be handled first.

PTA requirements for ecommerce and personal profiling give great need to addressing security. Basic services for ensuring the financial transaction and certification of documents are required. Much of this can be assumed by appropriate use of the underlying protocol, such as SSL.

3 Architecture

3.1 Service Architectures and Protocols

The PTA architecture (see *Figure 3*) should act as a reference model which identifies and characterises the components, interfaces and protocols.

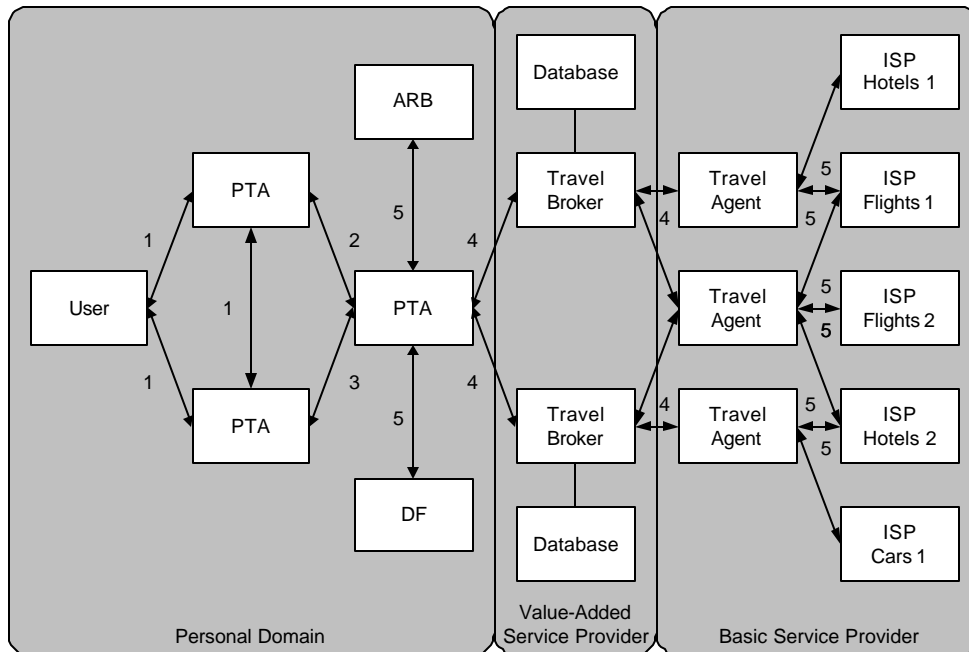


Figure 3: Personal Travel Assistant Architecture

The diagram represents the various agent types and the communication types between them. This section provides a description of representative agents, some representative APs and then the protocols between them. Conventions such as for agent naming will be followed as they are developed by [FIPA00023], but note that much of what is given in this specification is deliberately inconsistent (when consistency is not required) to demonstrate the probable state of multi-vendor vagaries.

3.2 Agent Definitions

Assume that a small company, *CompanyXYZ*, has installed an AP in which a multi-user implementation of a PTA is added. Each employee also is given a PDA with an MPTA. *CompanyXYZ* has agreements and policies to use World Travel Agency business travel and as an added value to its employees, *CompanyXYZ* has also developed its PTA to look-up value-added brokers to arrange for their personal interests, as well. These agencies are associated with various basic service providers.

3.2.1 Mini-Personal Travel Agent

```

:name
  (agent-identifier
   :name MPTA.JoeSmith@companyxyz.com
   :addresses (sequence gsm://011235551234/acc))
:type fipa-mpta
:services (set
  (service-description
   :properties (set

```

```

    (property
      :name Actions
      :value (set notify available)))
  :ontology (set User))
(service-description
  :properties (set
    (property
      :name Location
      :value self))
  :ontology (set FIPA-PTA)))
:protocol (set FIPA-Request)
:ontology (set User FIPA-PTA)

```

Joe Smith is given an MPTA because he travels a lot for Company XYZ. Due to its limited capacity, it understands only the `request` protocol (see [FIPA00026]), but can provide a unique service to the entire PTA system of agents. Assuming an ontology called `User`, it can handle the operation of notifying the user, if he/she is available. For on-trip monitoring, it can provide location of itself, through its GPS attachment for example.

3.2.2 Personal Travel Agent

```

:name
  (agent-identifier
    :name PTA@companyxyz.com
    :addresses (sequence iiop://agents.companyxyz.com/acc))
:type fipa-pta
:services (set
  (service-description
    :properties (set
      (property
        :name Profile
        :value PersonalInterests))
    :ontology (set User))
  (service-description
    :ontology (set FIPA-PTA)))
:protocol (set FIPA-Contract-Net FIPA-Auction-Dutch FIPA-Request)
:ontology (set FIPA-PTA)

```

Assume that a small company such as CompanyXYZ would have only one personal travel agent as a multi-user system to service its entire staff. CompanyXYZ allows any flights with any carrier in order to get the cheapest fare and therefore, this PTA can follow Dutch auctions as well as contract net for conversation, either with brokers or with service providers directly. The company itself owns this PTA in order to control it in regard to corporate travel policies for example. Not only does the PTA handle the `FIPA-PTA` ontology for making regular travel arrangements, but it only understands user profiling. Residing on a server, the PTA is responsible for holding such personal profiling information (common travel preferences as well recreational interests, perhaps).

3.2.3 Travel Broker

```

:name
  (agent-identifier
    :name TravelAgent76@worldtravel.com
    :addresses (sequence iiop://broker.worldtravel.com/acc))
:type fipa-pta-broker
:services ...
:protocol (set FIPA-Contract-Net FIPA-Request-When)
:ontology (set FIPA-PTA)

```

As a large travel company, *WorldTravel* has a bank of several agents and this is number 76. As a TBA, this agent understands contract-net for negotiating basic travel arrangements, but also provides monitoring functions for its customers by using the `request-when` protocol (see [FIPA00028]) with its service providers. For instance, when a certain condition occurs concerning a reservation or the availability of a resource, the TBA is notified and can in turn notify other acquaintances.

3.2.4 Tourist Office Broker

```
:name
  (agent-identifier
   :name TourAgent@tokyotourism.com
   :addresses (sequence iiop://broker.toyko.tourism/acc))
:type fipa-pta-broker
:services ...
:protocol (set FIPA-Request)
:ontology (set User-Personal-Interest)
```

A tourist office in Tokyo with a small budget wants to participate in the PTA system by registering its agent with several brokers as a free value-added source of information. It is itself of broker of other agents in its geography, but it is informational only. For instance, given a user's personal interests, it can connect a PTA to an appropriate soft-service agent. It might also provide information about these soft services but does no transaction itself; it only needs the `request` protocol (see [FIPA00026]).

3.2.5 Flight Service Provider

```
:name
  (agent-identifier
   :name Domestic389@foil.com
   :addresses (sequence iiop://flightplanners.foil.com/acc))
:type fipa-pta-server
:services (set
  (service-description
   :ontology (set FIPA-PTA)
   :properties (set
    (property
     :name Actions
     :value (set reserve purchase))
    (property
     :name PTA-MeanType
     :value plane))))))
:language (set FIPA-KIF))
:protocol (set FIPA-Contract-Net)
:ontology (set FIPA-PTA)
```

A very large flight reservation company maintains a number of agents, some for domestic travel and some for international. It can make reservations or accept purchase for flights, but for flights only.

6.2.6 Web Service Provider

```
:name
  (agent-identifier
   :name GardenGuide@kewgardens.com
   :addresses (sequence http://agents.kewgardens.com/acc))
:type fipa-pta-server
:services (set
```



```
(service-description
 :ontology FIPA-PTA
 :properties (set
  (property
   :name PointOfInterest
   :value Gardening))))
:protocol (set FIPA-Request)
:ontology (set Yahoo FIPA-PTA)
```

A public garden that has a Web site for itself and links to other points of similar interest could register with a broker to provide information in this recreational domain. Although IOP was initially required to register with the brokers, it then changes its preferred address to use HTTP, perhaps to use a future HTTP user profiling standard. Note also that the ontology assumes Yahoo-based classification as a de-facto standard for specifying a user's interests.

3.3 Agent Platform Profiles

The following descriptions provide a list of examples using the AP description definition from [FIPA00023].

3.3.1 Small Company Agent Platform

```
(ap-description
 :name companyxyz.com
 :dynamic false
 :mobility false
 :properties (set
  (property
   :name Change-Environment
   :value Administrator)
  (property
   :name Delegation-Allowed
   :value (set (PTA.User) (PTA.Administrator)))
  (property
   :name Grant-Services
   :value Within-Platform)
  (property
   :name Access-DF
   :value within-platform))
 :transport-profile ...)
```

CompanyXYZ knows and provides all agents to its employees and so the agent system design is tightly controlled; the broker agents that the company has decided to use are known and static. Therefore, it does not allow dynamic registration nor support mobility. Authority is given to the administrator only and all permissions for accessing services and the DF are limited to agents within this AP. If any broker wants to contact the PTA, it must be based on its acquaintance model developed from the PTA's initial contact with it.

3.3.2 Travel Broker Agent Platform

```
(ap-description
 :name worldtravel.com
 :dynamic true
 :mobility false
 :properties (set
  (property
   :name Change-Environment
   :value Administrator)
  (property
```

```

    :name Delegation-Allowed
    :value No)
(property
  :name Grant-Services
  :value (set (Within-Platform PTA.CompanyXYZ)))
(property
  :name Access-DF
  :value Within-Platform)
:transport-profile ...)

```

The *Travel Service* company wants to allow outside agents to use its services, but no delegation is allowed.

3.3.3 Agent Hotel Platform (On-Trip Execution)

```

(ap-description
  :name travelservice.com
  :dynamic true
  :mobility true
  :properties (set
    (property
      :name Change-Environment
      :value Administrator)
    (property
      :name Delegation-Allowed
      :value No)
    (property
      :name Grant-Services
      :value (set (Within-Platform (ServiceProvider.Guest ContentProvider.Guest))))
    (property
      :name Access-DF
      :value Yes)
    :transport-profile ...)

```

Here, the metaphor of travelling agents as entourage to the human traveller is entertained by giving mobile agents a temporary home. The requirement is obviously not to rest; indeed, the agent can be continuously very active, but such an AP and the availability of a local DF provides a natural metaphor for many agent-based services.

The AP grants the agent access to all the services and content granted to guest authority. Many such services can be provided by the hotel itself or by surrounding partner agents in the local area. For instance, the hotel can provide its services to a human guest to the agent; the agent can request room service to deliver the user's preferred breakfast at the preferred time, for example. Note that such an AP can also be hosted by a company other than the hotel itself.

3.3.4 Domain Structures

Table 3 provides the list of DFs and the agents registered to them (and DFs registered to other DFs) for the pre-trip planning architecture and illustrates the agent-to-agent relationships that are most likely. For instance, a corporation is usually responsible for software distribution to its employees, in this case providing the directory of PTAs, MPTAs within its own domain, as well as contracted relationships to one or two travel brokers.

Directory Facilitator	Registered Agent
df@companyxyz.com	PTA@companyxyz.com MPTA.JoeSmith@companyxyz.com broker@worldtravel.com TravelGuideBroker@travelservice.com

df@worldtravel.com	Planner@foil.com TravelAgent76@worldtravel.com
df@foil.com	international@foil.com us@foil.com Domestic389@foil.com
df@travelguide.com	GardenGuide@kewgardens.com TicketSeller@worldsoccerfederation.com

Table 3: Example Agent Registrations

The TBAs maintain a directory of TSAs which are usually associated with well known, large service providers in the case of corporate travel agents, but generally, TBAs might also keep web-based TSAs in their directory.

Large service providers might keep their own directory of service agents and associate different agents to different requests as a method of call handling. For instance, some service agents in a larger agency might handle international travel, while others handle local arrangements. These sorts of service differences would be registered in the directory.

4 Personal Travel Assistant Ontology

Ontologies are needed to serve as a medium of common understanding among the collaborating agents. The FIPA-PTA ontology should be defined in a precise and consistent way to ensure an unambiguous interaction model between the disparate agents. More specifically, it is a significant part of the protocol that collaborating agents necessarily communicate the same terms or vocabularies to mean the same concepts or ideas for the same context. There are already several methods for building ontologies and languages to express them (Prolog, LILog, Ontolingua, Loom, Back++, etc.). However, there is not a well-known ontology built for travel.

The FIPA-PTA ontology does not exist by itself, neither is it self-sufficient to represent the PTA. Separation and cross-references to other ontologies is necessary as indicated in *Figure 4*.

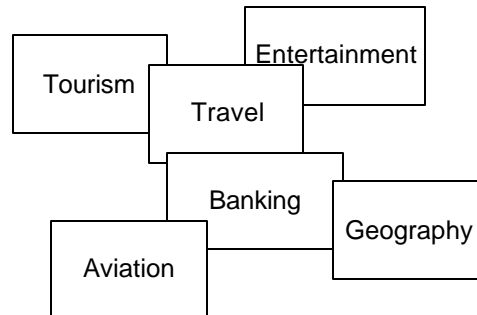


Figure 4: Potential Ontologies for Travel-Related Domains

As FIPA moves to support ontology definition and publication, these various ontologies will in fact become better separated. However, because the development and publication of ontologies per se is still evolving, the FIPA-PTA ontology will be defined here. For other possible relationships to travel, consider [FIPA00081], [FIPA00082] and [FIPA00083]. For instance, the entertainment domain is applicable for referencing video travelogues as a special case for video-on-demand.

Non-FIPA standards such as for Geographic Data Files (see [CENgdf]) will be referenced whenever they exist. ISO standards, such as for Language and Country codes, are also available (see [ISO639], [ISO4217] and [ISO3166]).

The FIPA-PTA ontology is a starting point to help the interoperability of early field trials of this application.

4.1 Object Descriptions

This section describes a set of frames, that represent the classes of objects in the domain of discourse within the framework of the FIPA-PTA ontology. The type definitions that are not specified here can be found in [FIPA00023].

The following terms are used to describe the objects of the domain:

- **Frame.** This is the mandatory name of this entity, that must be used to represent each instance of this class.
- **Ontology.** This is the name of the ontology, whose domain of discourse includes the parameters described in the table.
- **Parameter.** This is the mandatory name of a parameter of this frame.
- **Description.** This is a natural language description of the semantics of each parameter.
- **Presence.** This indicates whether each parameter is mandatory or optional.
- **Type.** This is the type of the values of the parameter: Integer, Word, String, URL, Term, Set or Sequence.

- **Reserved Values.** This is a list of FIPA-defined constants that can assume values for this parameter.

4.1.1 Trip Summary

The sender will provide `trip-summary` object as part of a query for travel arrangements, passing its parameters as a set of constraints. The receiver will reply with `trip-details` object or an exception object.

Frame Ontology	trip-summary FIPA-PTA	Parameter	Description	Presence	Type	Reserved Values
origin	The original location of the trip.	Mandatory	location			
via	A list of via locations of the trip.		Sequence of location			
destination	The destination location of the trip.	Mandatory	Sequence of location			
time	A list of start dates and times of the trip	Mandatory	Sequence of travel-time			
return-travel-time	A list of return dates and times of the trip.		Sequence of travel-time			
budget	The budget for the trip.	Optional	budget			
g-prefs	The general preferences of the trip.	Optional	g-prefs			
ct-prefs	The common travel preferences of the trip.	Optional	ct-prefs			
it-prefs	The individual travel preferences of the trip.	Optional	it-prefs			

4.1.2 Location

Frame Ontology	location FIPA-PTA	Parameter	Description	Presence	Type	Reserved Values
type	The type of the location	Optional	String	Address ParkAndRide PointOfInterest TextLocation ServicePoint TaxiStand GDFNode ResolvedCity		
address	The street of the location.	Optional	String			
city	The city of the location.	Optional	String			
zip-code	The zip code of the location.	Optional	String			
country	The country of the location.	Optional	String	See [ISO3166]		
text	The description of the location.	Optional	String			
service-point	The service point of the location.	Optional	service-point			
gdf-node	The GDF node of the location.	Optional	gdf-node			

4.1.3 Travel Time

Frame Ontology	travel-time FIPA-PTA	Parameter	Description	Presence	Type	Reserved Values
type	The type of the travel time point.	Mandatory	String	Departure Arrival		
at	The date and time of travel.	Mandatory	DateTime	See [FIPA00070]		
before	The time required before travelling.	Optional	DateTime			
after	The time required after travelling.	Optional	DateTime			

4.1.4 Budget

The sender can establish a budget range by specifying an upper spending limit for example. The receiver can reply with the exact amount using the `:at` parameter. Such a budget can also be used in other scenarios, such as for a Dutch auction. The budget can be used to trigger the automatic purchase by an agent when the price meets the constraints.

Frame Ontology	budget FIPA-PTA	Parameter	Description	Presence	Type	Reserved Values
currency	The currency in which the currency is expressed.	Mandatory	String	See [ISO4217]		
at	The exact value of the budget.	Mandatory	Integer			
upper	The upper limit of the budget.	Optional	Integer			
lower	The lower limit of the budget.	Optional	Integer			

4.1.5 General Preferences

This object indicates the preferred means of travel such as train versus car. The `:selection` parameter allows quality of service requirements to be expressed. The receiver should be expected to use this parameter to both clip and to order the results.

Frame Ontology	g-prefs FIPA-PTA	Parameter	Description	Presence	Type	Reserved Values
selection	The selection preference.	Mandatory	String	ByCost ByTime ByComfort		
preferred	The transport mechanisms preferred in this preference.	Mandatory	Set of String	CollectiveTransport IndividualCar IndividualTransport InterCityCollectiveTransport Taxi UrbanPublicTransport ...		
exclude	The transport mechanisms excluded in this preference.	Mandatory	Set of String	CollectiveTransport IndividualCar IndividualTransport InterCityCollectiveTransport Taxi UrbanPublicTransport ...		
language	A list of languages associated	Mandatory	Set of	See [ISO3166]		

	with this preference.		String	
map	The map type preference.	Mandatory	String	ForRoute ForOrigin ForDestination

4.1.6 Common Travel Preferences

This object represents common travel requirements that a user might wish to express to influence the type of transportation selected.

Frame Ontology	ct-prefs FIPA-PTA	Parameter	Description	Presence	Type	Reserved Values
class	The class of travel for this preference.	Mandatory	String	First Second Business Economy LastMinute ...		
fare	The fare type for this preference.	Mandatory	String	Child Adult Senior WeeklyPass MonthlyPass ...		
other	A list of other preferences.	Mandatory	Set of String	FootPathKnow EscalatorRequested HandicapForEntry HeavyLuggage ...		

4.1.7 Individual Travel Preferences

This object represents individual travel preferences to allow the user to include transport services, such as buses, but which allow the user an anonymous and individual means of transport.

Frame Ontology	it-prefs FIPA-PTA	Parameter	Description	Presence	Type	Reserved Values
preferred-speed	The preferred speed of transport.	Optional	String	Bus CarHurried CarRelaxed Lorry ...		
other	A list of other preferences.	Optional	Set of String	ParkingAtDestination WeatherInformation ...		

4.1.8 Trip Details

Given a `trip-summary` object as a query, the receiver will reply with a `trip-details` object. This object will include the original `trip-summary`. The constraints passed by the sender are replaced by the specific values or the trip-plan. For instance, the exact time and budget of the trip are provided. Additional information relating to the trip is appended, typically of travel documents for providing contact numbers and emergency procedures. Most importantly, the details of the trip are provided as `ct-segments` or `it-segments`.

Frame Ontology	trip-details FIPA-PTA	Parameter	Description	Presence	Type	Reserved Values
		trip-summary	The original trip summary query.	Mandatory	trip-summary	
		ct-segments	A list of common travel segments.	Optional	Sequence of ct-segment	
		it-segments	A list of individual travel segments.	Optional	Sequence of it-segment	
		information	Additional information about the trip.	Optional	String	

4.1.9 Common Travel Segment

A `ct-segment` is composed of service-links. This level of detail might not always be presented to the user except in summary form, but formally, a common travel segment often includes plane hops or train stops. These links are important to construct and monitor a trip.

Frame Ontology	ct-segment FIPA-PTA	Parameter	Description	Presence	Type	Reserved Values
		service-point	The service point of the segment.	Mandatory	service-point	
		trip-summary	A summary of the segment.	Mandatory	trip-summary	
		service-links	A list of links relating to the segment.	Optional	Sequence of service-links	

4.1.10 Individual Travel Segment

A `it-segment` has a similar structure to a `ct-segment`. Both include a `trip-summary` to provide location, time, budget and preference information for each segment and both indicate service points. However, an `it-segment` might include an unresolved service point, as well. For instance, car transportation might require a rental car (from a resolved service point) or simply a personal car (an unresolved service point).

Frame Ontology	it-segment FIPA-PTA	Parameter	Description	Presence	Type	Reserved Values
		service-point	The service point of the segment.	Mandatory	service-point	
		trip-summary	A summary of the segment.	Mandatory	trip-summary	
		gdf-links	A list of GDF links relating to the segment.	Optional	Sequence of gdf-links	

4.1.11 Service Point

Resolved and unresolved service points distinguish between well known locations of service providers versus general locations that are less well defined.

Frame Ontology	service-point FIPA-PTA			
Parameter	Description	Presence	Type	Reserved Values
type	The type of the service point.	Mandatory	String	Resolved Unresolved
identifier	The identifier of the service point.	Optional	String	
name	The name of the service point.	Mandatory	String	
mean	The type of the service point.	Optional	String	Bus CableRailway ChainTrain CommuterTrain Foot LowFloorBus MagneticTrain Plane Ship SuspensionRailway Train Tram Underground
city	The city of the service point.	Mandatory	String	
country	The country of the service point.	Mandatory	String	See [ISO3166]
co-ordinates	The co-ordinates of the service point.	Optional	Double Double	

4.1.12 Service Link

This object forms a travel connection within a *ct*-segment.

Frame Ontology	service-link FIPA-PTA			
Parameter	Description	Presence	Type	Reserved Values
identifier	The identifier of the transport.	Mandatory	String	
origin	The start point of the link.	Mandatory	service-point	
departure-time	The departure date and time of the link.	Mandatory	Date	
destination	The destination of the link.	Mandatory	service-point	
arrival-time	The arrival date and time of the link.	Mandatory	Date	
delay	The delay associated with the link.	Optional	Integer	

4.1.13 Geographic Data File Node

This object forms a particular geographical point.

Frame Ontology	gdf-node FIPA-PTA			
Parameter	Description	Presence	Type	Reserved Values
identifier	The identifier of the node.	Mandatory	Integer	
name	The name of the node.	Mandatory	String	

4.1.14 Geographic Data File Link

This object forms a travel connection which is based upon geographical points for navigation.

Frame Ontology	gdf-link FIPA-PTA			
Parameter	Description	Presence	Type	Reserved Values
identifier	The identifier of the link.	Mandatory	Integer	
name	The name of the link	Mandatory	String	
location-start	The start co-ordinates of the link.	Mandatory	Double Double	
location-end	The end co-ordinates of the link.	Mandatory	Double Double	
turn-instruction	The instruction associated with the link.	Mandatory	String	GoStraight TurnLeft TurnRight
length	The length of the link.	Mandatory	Integer	
information	Travel information associated with the link.	Optional	String	

4.1.15 Plan Evaluation

This object forms a pair of references for evaluating new plans based on existing relevant plans or trash plans.

Frame Ontology	plan-evaluation FIPA-PTA			
Parameter	Description	Presence	Type	Reserved Values
relevant	A list of identifiers of relevant plans.	Optional	Integer Integer	
trash	A list of identifier of irrelevant plans.	Optional	Integer Integer	

4.2 Function Descriptions

The following tables define usage and semantics of the functions that are part of the FIPA-PTA ontology.

The following terms are used to describe the functions of the FIPA-PTA domain:

- **Function.** This is the symbol that identifies the function in the ontology.
- **Ontology.** This is the name of the ontology, whose domain of discourse includes the function described in the table.
- **Supported by.** This is the type of agent that supports this function.
- **Description.** This is a natural language description of the semantics of the function.
- **Domain.** This indicates the domain over which the function is defined. The arguments passed to the function must belong to the set identified by the domain.
- **Range.** This indicates the range to which the function maps the symbols of the domain. The result of the function is a symbol belonging to the set identified by the range.
- **Arity.** This indicates the number of arguments that a function takes. If a function can take an arbitrary number of arguments, then its arity is undefined.

4.2.1 Reserve a Segment

Function	reserve
Ontology	FIPA-PTA
Supported by	TBA/Service Provider
Description	The execution of this function has the effect of reserving a segment or group of segments.
Domain	Sequence of <i>ct-segment</i> / Sequence of <i>it-segment</i>
Range	The execution of this function results in a change of the state, but it has no explicit result. Therefore there is no range set.
Arity	1

4.2.2 Cancel a Segment

Function	cancel
Ontology	FIPA-PTA
Supported by	TBA/Service Provider
Description	The execution of this function has the effect of cancelling a segment.
Domain	<i>ct-segment</i> / <i>it-segment</i>
Range	The execution of this function results in a change of the state, but it has no explicit result. Therefore there is no range set.
Arity	1

4.2.3 Purchase a Segment

Function	purchase
Ontology	FIPA-PTA
Supported by	TBA/Service Provider
Description	The execution of this function has the effect of purchasing a segment or group of segments.
Domain	Sequence of <i>ct-segment</i> / Sequence of <i>it-segment</i>
Range	The execution of this function results in a change of the state, but it has no explicit result. Therefore there is no range set.
Arity	1

4.2.4 Modify a Segment

Function	modify
Ontology	FIPA-PTA
Supported by	TBA/Service Provider
Description	An agent may make a modification in order to change the details of a segment. The argument of a <i>modify</i> function will replace the existing object description stored within the executing agent.
Domain	<i>ct-segment</i> / <i>it-segment</i>
Range	The execution of this function results in a change of the state, but it has no explicit result. Therefore there is no range set.
Arity	1

4.2.5 Evaluate a Plan

Function	evaluate
Ontology	FIPA-PTA
Supported by	TBA/Service Provider
Description	An agent may ask a for future plans based upon the evaluation of given plans as either relevant or trash.
Domain	plan-evaluation
Range	The execution of this function results in a change of the state, but it has no explicit result. Therefore there is no range set.
Arity	1

4.3 Elaboration of the User Profile

The purpose of the user profile is to improve the PTA service to the user as well as to the broker or service/content providers. Personalisation means ease of filling the request since many personal data items are constant and also means service modifications and propositions according to the accuracy of the user profile. From the user's point of view, personalisation affects the search process, assistance and the presentation of results and from the service/content provider's perspective it helps in better matching user needs.

Some items of preference were included in the FIPA-PTA ontology, but much more is possible in this special domain. Even most simply, the requirements for e-commerce should include the user's preferred method of payment in an object such as:

Frame	payment			
Ontology	FIPA-PTA			
Parameter	Description	Presence	Type	Reserved Values
method	The payment method.	Mandatory	String	Visa MasterCard AmericanExpress ...
balance	The balance on the payment method.	Optional	Integer	
limit	The limit on the payment method.	Optional	Integer	

A hotel might also like to know whether a smoking or non-smoking room is preferred in a booking and this is a property of the user that might be granted to the hotel for this need, but in the FIPA-PTA ontology, explicit reference to this type of information is not made.

There are also many other complexities to what is generally called a user profile. Aside from the more static and clear parameters of the user such as name, telephone and email addresses, we need to differentiate such a user profile into three separate structures:

1. The ontology of domains such as travel, recreation, sports, entertainment and music,
2. An explicit preference structure mapped onto this ontology (`:preference-carrier AirFrance`), and,
3. An implicit preference structure, also mapped onto this ontology, such as learned patterns of the user's behaviour within a given ontology.

In other words, the ontology description of virtually all items should first exist separately from the user profile as already emphasised in the previous section. Moreover, the functions "preference" and "interest" can be applied. If it is of value, a distinction between these two might be:

- Preferences that are reserved for the user's probable selection from a short, well defined list (forced choice situations), and,
- Interests which describe personal strength of like-dislike on a single item (rating situations).

In summary, the FIPA-PTA ontology is a beginning towards the definitions of trip segments, especially in multi-modal travel. It highlights some inclusion of soft services and the important application of position and way finding technologies. It is still inadequate for the definition of node-based resources such as hotels and attractions and its reference to electronic commerce standards such as SET still need development for real business transactions to take place. Towards integration with other standards, issues of user profiling and privacy, such as the Open Profiling Standard [OPS], much more can be done to make such an application available.

4.4 Exceptions

The exceptions for the FIPA-PTA ontology follow the same form and rules as specified in [FIPA00023].

4.4.1 Failure Exception Propositions

Communicative Act Ontology	failure FIPA-PTA	
Predicate symbol	Arguments	Description
location-ambiguous	String	The location cannot be determined precisely; the string identifies the location.
location-not-found	String	The specified location cannot be found; the string identifies the location.
no-ct-connection	String String	A common traveller connection cannot be establish from a specified start location to a specified end location; the first string identifies the start location and the second string identifies the end location.
service-not-available	String	A service is not available; the string identifies the service name.
no-city-info	String	No information is available for the specified city; the string identifies the city.

5 Study Cases

5.1 Pre-Trip Planning

This scenario is focused exclusively on the details of agent interaction. As such, the interaction diagram in *Figure 5* shows the four agents involved and the communicative acts exchanged between them.

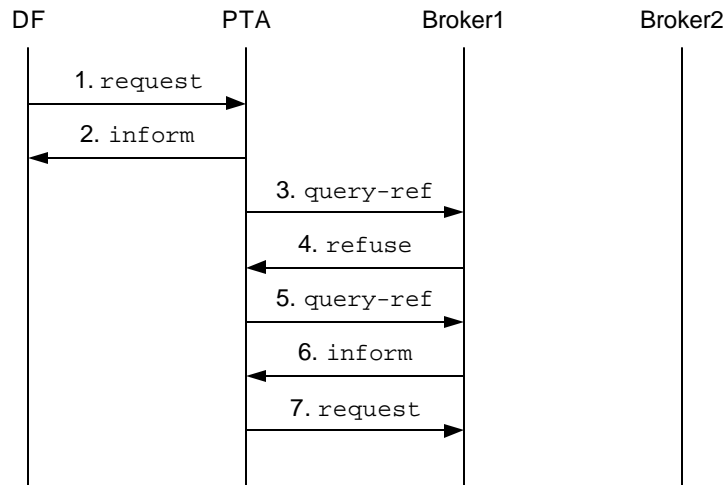


Figure 5: Agent Interaction for Pre-trip Planning

A formal description of intentions and some of the important content description is described as follows¹:

1. Request the DF to find more than one broker. Message content requires some rough description of service offerings/capabilities.

```

(request
  :sender
    (agent-identifier
      :name PTA@foo.com
      :addresses (sequence iiop://foo.com/acc))
  :receiver (set
    (agent-identifier
      :name DF@bar.com
      :addresses (sequence iiop://bar.com/acc)))
  :language FIPA-SL0
  :protocol FIPA-Request
  :ontology FIPA-Agent-Management
  :content
    (action
      (agent-identifier
        :name DF@bar.com
        :addresses (sequence iiop://bar.com/acc))
      (search
        (df-agent-description
          :services (set
            (service-description
              :type fipa-tba))))))
  :reply-with KarlsTrip)

```

¹ The intermediate messages of the FIPA-Request and FIPA-Query-Ref interaction protocol have been omitted for clarity.

- The DF searches through its own directory and possible that of other DFs. It informs the PTA with list of two TBAs meeting the specified service requirements. Note that the DF has not been required to open communication to the brokers or to ensure their current existence after registration.

```
(inform
  :sender
    (agent-identifier
      :name DF@foobarcom
      :addresses (sequence iiop://bar.com/acc))
  :receiver (set
    (agent-identifier
      :name PTA@foo.com
      :addresses (sequence iiop://foo.com/acc)))
  :language FIPA-SL0
  :protocol FIPA-Request
  :ontology FIPA-Agent-Management
  :content
    (result
      (action
        (agent-identifier
          :name DF@bar.com
          :addresses (sequence iiop://bar.com/acc))
        (search
          (df-agent-description
            :services (set
              (service-description
                :type fipa-tba))))))
      (set
        (df-agent-description
          :name
            (agent-identifier
              :name Broker1@foo.com
              :addresses (sequence iiop://foo.com/acc))
          :services (set
            (service-description
              :type fipa-tba)))
        (df-agent-description
          :name
            (agent-identifier
              :name Broker2@bar.com
              :addresses (sequence iiop://bar.com/acc))
          :services (set
            (service-description
              :type fipa-tba))))))
  :in-reply-to KarlsTrip)
```

- The PTA ask one of the brokers for information (no contractual obligation) for a possible trip. Note that the PTA uses the `iota` operator when communicating with the broker, which requires the `FIPA-SL2` language rather than `FIPA-SL0`. This does not imply that `SL` is required for field trials; this content language in this scenario is provided only as an example.

```
(query-ref
  :sender
    (agent-identifier
      :name PTA@foo.com
```



```

      :addresses (sequence iiop://foo.com/acc))
:receiver (set
  (agent-identifier
    :name Broker1@bar.com
    :addresses (sequence iiop://bar.com/acc)))
:language FIPA-SL2
:protocol FIPA-Query-Ref
:ontology FIPA-PTA
:content (iota ?trip-details
  (available Broker1@bar.com ?trip-details
    (trip-summary
      :origin (location
        :country GE
        :city Frankfurt)
      :destination (location
        :country FR
        :city Dublin)
      :time (sequence
        (travel-time
          :type Departure
          :after 19971010T170000Z
          :before 19971919T240000Z))))))

```

4. Broker1@bar.com refuses because it knows about two cities with similar names (Frankfurt am Main and Frankfurt a.d. Oder).

```

(refuse
  :sender
    (agent-identifier
      :name Broker1@bar.com
      :addresses (sequence iiop://bar.com/acc))
  :receiver (set
    (agent-identifier
      :name PTA@foo.com
      :addresses (sequence iiop://foo.com/acc)))
  :language FIPA-SL0
  :ontology FIPA-PTA
  :content ((iota ?trip-details
    (available Broker1@bar.com ?trip-details
      (trip-summary
        :origin (location
          :country GE
          :city Frankfurt)
        :destination (location
          :country FR
          :city Dublin)
        :time (sequence
          (travel-time
            :type Departure
            :after 19971010T170000Z
            :before 19971919T240000Z))))))
    (location-ambiguous Frankfurt)))

```

5. The PTA corrects this problem using the full name of the city.

```

(query-ref

```

```

:sender
  (agent-identifier
   :name PTA@foo.com
   :addresses (sequence iiop://foo.com/acc))
:receiver (set
  (agent-identifier
   :name Broker1@bar.com
   :addresses (sequence iiop://bar.com/acc)))
:language FIPA-SL2
:protocol FIPA-Query-Ref
:ontology FIPA-PTA
:content (iota ?trip-details
  (available Broker1@bar.com ?trip-details
   (trip-summary
    :origin (location
     :country GE
     :city "Frankfurt am Main")
    :destination (location
     :country FR
     :city Dublin)
    :time (sequence
     (travel-time
      :type Departure
      :after 19971010T170000Z
      :before 19971919T240000Z))))))

```

6. Broker1@bar.com can now reply with the details of the trip.

```

(inform
 :sender
  (agent-identifier
   :name Broker1@bar.com
   :addresses (sequence iiop://bar.com/acc))
:receiver (set
  (agent-identifier
   :name PTA@foo.com
   :addresses (sequence iiop://foo.com/acc)))
:ontology FIPA-PTA
:content
  (tripDetails
   :summary
    (trip-summary
     :origin (location
      :country GE
      :city "Frankfurt am Main")
     :destination (location
      :country FR
      :city Dublin)
     :time (sequence
      (travel-time
       :type Departure
       :after 19971010T170000Z
       :before 19971919T240000Z)))
   :ct-segments (sequence
    (ct-segment
     :service-point (service-point

```

```

      :type Resolved
      :identifier LH
      :name "Lufthansa Airlines"
      :country DE
      :city "Frankfurt am Main")
    :trip-summary ...
    :service-links ...))
  :information ...))

```

7. The PTA is satisfied with this plan and proceeds to reserve the suggested common travel segment.

```

(request
  :sender
    (agent-identifier
      :name PTA@foo.com
      :addresses (sequence iiop://foo.com/acc))
  :receiver (set
    (agent-identifier
      :name Broker1@bar.com
      :addresses (sequence iiop://bar.com/acc)))
  :language FIPA-SL0
  :protocol FIPA-Request
  :ontology FIPA-PTA
  :content
    (action
      (agent-identifier
        :name Broker1@bar.com
        :addresses (sequence iiop://bar.com/acc))
      (reserve
        (ct-segment
          :service-point (service-point
            :type Resolved
            :identifier LH
            :name "Lufthansa Airlines"
            :country DE
            :city "Frankfurt am Main")
          :trip-summary ...
          :service-links ...))
        :information ...))

```

5.2 Elaboration of Pre-Trip Planning

While pre-trip planning is mostly a matter of reserving or purchasing hard travel documents, the full PTA system is intended to include the added value of soft services. This scenario demonstrates such an elaboration of pre-trip planning. As per the FIPA-PTA ontology, the profiling ontology is not ready for field trial usage. However, this elaboration assumes such an ontology will at least include an object named `PersonalInterest`, which is used in this scenario which continues where the last scenario ended.

1. `Broker1@bar.com` asks the PTA whether it can access to the user's preference profile in order to add additional entertainment items to the travel plans.

```

(query-ref
  :sender
    (agent-identifier
      :name Broker1@bar.com
      :addresses (sequence iiop://bar.com/acc))

```

```

:receiver (set
  (agent-identifier
    :name PTA@foo.com
    :addresses (sequence iiop://foo.com/acc)))
:language FIPA-SL2
:protocol FIPA-Query-Ref
:ontology FIPA-Profile
:content (iota ?profile
  (access PTA@foo.com ?profile)))

```

2. The PTA decides to provide Broker1@bar.com with a subset of the user's profile. It provides three interest items, defined by the item itself and the item's ontology.

```

(inform
  :sender
    (agent-identifier
      :name PTA@foo.com
      :addresses (sequence iiop://foo.com/acc))
  :receiver (set
    (agent-identifier
      :name Broker1@bar.com
      :addresses (sequence iiop://bar.com/acc)))
  :ontology FIPA-Profile
  :content (profile
    :personal-interests
      (interests (sequence
        (:interest football
          :ontology (set sport))
        (:interest ballet
          :ontology (set culture))
        (:interest gardening
          :ontology (set hobby)))))))

```

3. Broker1@bar.com replies with a Botanic Garden in Dublin as a potential point of interest for the user.

```

(inform
  :sender
    (agent-identifier
      :name Broker1@bar.com
      :addresses (sequence iiop://bar.com/acc))
  :receiver (set
    (agent-identifier
      :name PTA@foo.com
      :addresses (sequence iiop://foo.com/acc)))
  :ontology FIPA-Profile
  :content (location
    :country IR
    :city Dublin
    :name "Botanic Gardens"))

```

The PTA ontology does not yet extend to node items such as hotels, much less to soft travel items such as entertainment events. However, with such extensions a similar conversation could also provide a means for the broker to suggest ballet or football tickets and the PTA to reserve or purchase them in which case they would become part of the complete travel package.

5.3 Last Minute Auction for Lower Fare

Another airline provider notices a large number of open seats on one of its flights (which happens to satisfy the flight plans in the above scenario). The airline provider agent contacts several brokers, one of which is the broker in the above scenario.

1. The broker contacts the PTA that owns the travel documents to see if it (or the PTA's user) would be interested in a possibly cheaper fare.

```
(inform
  :sender
    (agent-identifier
      :name Flights@bar.com
      :addresses (sequence iiop://bar.com/acc))
  :receiver (set list-of-acquaintances)
  :ontology FIPA-PTA
  :protocol FIPA-Auction-Dutch
  :content ((sell seats 100)
    (trip-summary
      :origin (location
        :country GE
        :city "Frankfurt am Main")
      :destination (location
        :country FR
        :city Dublin)
      :time (sequence
        (travel-time
          :type Departure
          :after 19971010T170000Z
          :before 19971919T240000Z))))
```

2. The auctioneer agent opens the auction at some starting price and invites takers for that price from the audience. The auctioneer in this case is assumed to be the `Flights@bar.com` but this is not necessary. Additionally, assume that the PTA has registered itself with the auctioneer and is one of the agents participating in the audience.

```
(cfp
  :sender
    (agent-identifier
      :name Auctioneer@bar.com
      :addresses (sequence iiop://bar.com/acc))
  :receiver (set audience)
  :content ((buy ticket) ((max-no 20) (cost 100)))
  :reply-with cfp0
  :context FIPA-Auction-Dutch)
```

3. If no audience takes bid, the auctioneer counter-proposes with a lower price.

```
(cfp
  :sender
    (agent-identifier
      :name Auctioneer@bar.com
      :addresses (sequence iiop://bar.com/acc))
  :receiver (set audience)
  :content ((buy ticket) ((max-no 20) (cost 90)))
  :reply-with cfp1
  :context FIPA-Auction-Dutch)
```

4. Audiencel@foo.com takes a bid.

```
(bid
  :sender
    (agent-identifier
      :name Audiencel@foo.com
      :addresses (sequence iiop://foo.com/acc))
  :receiver (set
    (agent-identifier
      :name Auctioneer@bar.com
      :addresses (sequence iiop://bar.com/acc)))
  :content ((buy ticket) ((no 20) (cost 90)))
  :in-reply-to cfp1)
```

5. The auctioneer accepts this bid.

```
(accept-offer
  :sender
    (agent-identifier
      :name Auctioneer@bar.com
      :addresses (sequence iiop://bar.com/acc))
  :receiver (set
    (agent-identifier
      :name Audiencel@foo.com
      :addresses (sequence iiop://foo.com/acc)))
  :content ((buy ticket) ((no 20) (cost 90)))
  :in-reply-to cfp1)
```

6. The auctioneer continues to invite takers with a lower price.

```
(cfp
  :sender
    (agent-identifier
      :name Auctioneer@bar.com
      :addresses (sequence iiop://bar.com/acc))
  :receiver (set audience)
  :content ((buy ticket) ((max-no 15) (cost 85)))
  :reply-with cfp2
  :context FIPA-Auction-Dutch)
```

7. This propose, bid and accept-offer cycle continues until the number of seats becomes zero or it arrives at minimum price. If the number of goods offered is insufficient, the auctioneer may reject a bid as follows.

```
(reject-offer
  :sender
    (agent-identifier
      :name Auctioneer@bar.com
      :addresses (sequence iiop://bar.com/acc))
  :receiver (set
    (agent-identifier
      :name Audience2@foo.com
      :addresses (sequence iiop://foo.com/acc)))
  :content ((buy ticket) ((no 20) (cost 85)))
  :in-reply-to cfp2)
```

- At last the auctioneer tells the audience that the auction is finished.

```
(inform
  :sender
    (agent-identifier
      :name Auctioneer@bar.com
      :addresses (sequence iiop://bar.com/acc))
  :receiver (set audience)
  :content (done auction))
```

5.4 On-Trip Execution

This scenario focuses more on the required software attachments rather than agent interaction. This scenario description is still incomplete, but the following diagram shows the Inform-Request performative within the simple client/server protocol between an agent core and its wrappers.

- DialogWrapper@foo.com asks MPTA@foo.com "Where am I?" This is not a performative between user and agent; it is simply an event from a piece of software. DialogWrapper@foo.com informs the agent core of this event, but it is now in terms of dialogue semantics and content.

```
(inform
  :sender
    (agent-identifier
      :name DialogWrapper@foo.com
      :addresses (sequence iiop://foo.com/acc))
  :receiver (set
    (agent-identifier
      :name MPTA@foo.com
      :addresses (sequence iiop://foo.com/acc)))
  :ontology UserDialog
  :content (gui-event
    :event WhereAmI))
```

- MPTA@foo.com queries its GPS co-ordinates.

```
(query-ref
  :sender
    (agent-identifier
      :name MPTA@foo.com
      :addresses (sequence iiop://foo.com/acc))
  :receiver (set
    (agent-identifier
      :name MapAgent@foo.com
      :addresses (sequence iiop://foo.com/acc)))
  :ontology GPS
  :content (iota ?x (city-list ?x
    (gps-position
      :x 135
      :y 35))))
```

- MapAgent@foo.com returns the list of nearby cities.

```
(inform
  :sender
    (agent-identifier
```

```

        :name MapAgent@foo.com
        :addresses (sequence iiop://foo.com/acc))
:receiver (set
  (agent-identifier
    :name MPTA@foo.com
    :addresses (sequence iiop://foo.com/acc)))
:ontology GPS
:content (Akashi))

```

4. MPTA@foo.com requests DialogWrapper@foo.com to display the some information about the city at the current position.

```

(request
  :sender
    (agent-identifier
      :name MPTA@foo.com
      :addresses (sequence iiop://foo.com/acc))
  :receiver (set
    (agent-identifier
      :name DialogWrapper@foo.com
      :addresses (sequence iiop://foo.com/acc)))
  :ontology UserDialog
  :content (gui-action
    :display "The city at the current position is Akashi"))

```

The following is another scenario where the MiniPTA migrates on the network.

1. MPTA@foo.com migrates bar.com and requests GPSWrapper@bar.com to notify it when the GPS co-ordinates of the user change.

```

(subscribe
  :sender
    (agent-identifier
      :name MPTA@bar.com
      :addresses (sequence iiop://bar.com/acc))
  :receiver (set
    (agent-identifier
      :name GPSWrapper@bar.com
      :addresses (sequence iiop://bar.com/acc)))
  :ontology GPS
  :content (iota ?x (gps-position ?x)))

```

2. GPSWrapper@bar.com informs MPTA@foo.com of its CPS co-ordinates when they change.

```

(inform
  :sender
    (agent-identifier
      :name GPSWrapper@bar.com
      :addresses (sequence iiop://bar.com/acc))
  :receiver (set
    (agent-identifier
      :name MPTA@bar.com
      :addresses (sequence iiop://bar.com/acc)))
  :ontology GPS
  :content (gps-position
    :x 135

```



```
:y 35))
```

3. MPTA@foo.com requests a translation of the GPS co-ordinates into a list of nearby cities.

```
(query-ref
  :sender
    (agent-identifier
      :name MPTA@bar.com
      :addresses (sequence iiop://bar.com/acc))
  :receiver (set
    (agent-identifier
      :name MapAgent@foo.com
      :addresses (sequence iiop://foo.com/acc)))
  :ontology GPS
  :content (iota ?x (city-list ?x
    (gps-position
      :x 135
      :y 35))))))
```

4. MapAgent@foo.com returns the list of nearby cities.

```
(inform
  :sender
    (agent-identifier
      :name MapAgent@foo.com
      :addresses (sequence iiop://foo.com/acc))
  :receiver (set
    (agent-identifier
      :name MPTA@bar.com
      :addresses (sequence iiop://bar.com/acc)))
  :ontology GPS
  :content (Akashi))
```

5.5 Travel Plan Monitoring

The following notations provide some initial definition of agent planning, plan decomposition and communication in the context of plan monitoring. These steps are assumed to tie pre-trip planning with on-trip execution. For instance, pre-trip planning should include the distribution of the plan to multiple agents, such as between a PTA and an MPTA.

A plan is composed of plan items such as:

$$P = P_1 \cdot P_2 \cdot P_3 \cdot \dots \cdot P_N$$

which can be decomposed for the purposes of parallel execution of the monitoring as:

$$\text{monitor}(P) = \text{monitor}(P_1) \mid \text{monitor}(P_2) \mid \dots \mid \text{monitor}(P_n)$$

Given this parallel execution, the task of monitor can be distributed to many agents at many places (at the GPS input point, at the flight database, etc.).

A PTA owns the entire composite plan at the pre-trip phase. Given the registered capabilities of other agents to accept the monitor performative, the PTA can request other agents to monitor parts of the plan. For instance, the PTA can distribute some elements to the MPTA or to the service provider agents. In the latter case, the PTA can request a TSA to notify it if schedule or other conditions change.

6 Agent/Software Integration

6.1 Web-Based Fare Wrapper

This example (which uses definitions from [FIPA00079]) shows how a wrapper to web-based content hosting can be provided by a third-party vendor. Parsing HTML is never a clean solution, but is the only recourse available for an agent if it needs to access web-based content. To assist in this parsing, wrapper agents can be used to provide a mapping between the raw content and its representation to the level of an ontology and an agent-based representation.

This example shows how such a third-party vendor can provide added-value to the PTA community of agents, so that every agent in the system does not have to re-implement such lower level attachments. The content structure is likely to change, but this wrapper provider can monitor and moderate such changes for several agents.

This example also assumes that the web-based content provider offers a Dutch auction to human participants from time to time. The Great Deal Web site publishes this event on its site such that the `GreatDealParser` can determine this event automatically.

6.1.1 Registration of a Wrapper Agent

```
(request
  :sender
    (agent-identifier
      :name GreatDealWrapper@foo.com
      :addresses (sequence iiop://foo.com/acc))
  :receiver (set
    (agent-identifier
      :name ARB@bar.com
      :addresses (sequence iiop://bar.com/acc)))
  :ontology FIPA-ARB)
:content (action
  (agent-identifier
    :name ARB@bar.com
    :addresses (sequence iiop://bar.com/acc))
  (register
    (service-description
      :name GreatDealParser
      :type fipa-wrapper
      :ontology (set Market)
      :properties (set
        (property
          :name events
          :value (set price-change great-deal-auction))
        (property
          :name sensors
          :value (set current-price carrier flight-number))))
    :communication (set
      (:protocol HTTP
        :address http://www.greatdeal.com/pricetable
        :body-format FIPA-String-ACL
        :body-encoding XDR))))))
```

6.1.2 Request for a Price

```
(query-ref
```

```

:sender
  (agent-identifier
   :name FlightServiceAgent@bar.com
   :addresses (sequence iiop://bar.com/acc))
:receiver (set
  (agent-identifier
   :name GreatDealWrapper@foo.com
   :addresses (sequence iiop://foo.com/acc)))
:content (current-price
  :carrier AA
  :flight 712))

(inform
 :sender
  (agent-identifier
   :name GreatDealWrapper@foo.com
   :addresses (sequence iiop://foo.com/acc))
:receiver (set
  (agent-identifier
   :name FlightServiceAgent@bar.com
   :addresses (sequence iiop://bar.com/acc)))
:content (price
  :currency USD
  :value 400))

```

6.1.3 Notification of Price Change

The wrapper might support a subscription method to receiving such notification, but in the simplest case, consider that the wrapper will trigger the following message when any published price changes on the price table page.

```

(inform
 :sender
  (agent-identifier
   :name GreatDealWrapper@foo.com
   :addresses (sequence iiop://foo.com/acc))
:receiver (set
  (agent-identifier
   :name FlightServiceAgent@bar.com
   :addresses (sequence iiop://bar.com/acc)))
:content (price-change
  :carrier AA
  :flight 712
  :price (price
   :currency USD
   :value 250)))

```

6.2 BAYERNInfo Service Wrapper

This is an example of a specific existing service for very high level inter-modal route planning which is restricted to Bavaria.

6.2.1 Agent Request for a Route

```

(query-ref
 :sender

```

```
(agent-identifier
  :name MPTA.JoeSmith@companyxyz.com
  :addresses (sequence gsm://011235551234/acc))
:receiver (set
  (agent-identifier
    :name BAYERNInfoWrapper@bar.com
    :addresses (sequence iiop://bar.com/acc)))
:content (street-route
  :start-location ...
  :end-location ...
  :start-time ...))
```

7 Future PTA Developments

7.1 Travelling Users

Mobile end-users are a major driver toward mobile agent technology. The applications of mobile agent technology to personal travel assistance as a natural abstraction design seems clear.

7.1.1 Agent Mobility in a Network: Travel Planning

The traveller is based in Germany and organises a business trip to Korea and Japan. The costs of communications and bandwidth have to be minimised and long distance calls should be avoided. While in Germany, the PTA checks for flight facilities. Then, it moves into the Korean domain which contains information on local arrangements as well as entertainment facilities. The organisation of meetings with partners requests the use of negotiations to find the best schedules for all parties. In case of drastic time constraints, such negotiations require a lot of effort and the hotel reservations may be performed by auction to find the best conditions. Due to its autonomy, the PTA successfully makes the bookings and collects only the required information according to the flight schedule possibilities. For example, it will provide a list of concerts and other events the traveller may wish to attend during their stay. It moves to Japan to carry the same work out and to finalise the trip possibilities. Finally, the PTA returns to Germany with the schedules of the meetings, entertainment, hotel and car reservations, etc.

This scenario shows benefits for the traveller in terms of the quality of planning and the lower travelling costs. In particular, the mobility of the agents provides shorter response times, minimises the cost of the transmissions and lowers the bandwidth required by the application.

7.1.2 Traveller Mobility: Travel Monitoring

The traveller packs an MPTA in their luggage so they are able to connect to their virtual office environment in a transparent manner, for example, email, ongoing work, Internet access, etc. Agent migration reduces the connection costs by moving some agents in fixed network to gain efficiency and lower bandwidth requirements.

Another function of the mobile MPTA is to monitor the progress of the traveller. For instance, while staying in Korea, a typhoon hits the country and the flight out of the country is cancelled. As such, the traveller will spend one extra day in Korea, but has to reschedule their meetings in Japan. The MPTA will provide access to the requested data, propose to reschedule the journey and the meetings and contact the Japanese partners involved. Finally, the MPTA will also inquire for entertainment possibilities in Korea for the extra day and inform German colleagues and family of the traveller's new arrangements.

In this case, the MPTA has to access the local entertainment resources in Korea, but needs some agent mobility to minimise the connection costs to Japan and Germany.

7.1.3 Traveller Mobility: Travel Monitoring Via UMTS

The mobile telecommunication world permits access to everyone anywhere at any time. As such, the service offered by the UMTS (Universal Mobile Telecommunications System) MPTA is greatly enhanced. Using the earlier example, the traveller gets the weather forecast as soon as it is published and the MPTA may reschedule the trip in time to finish business in Korea before the arrival to the typhoon. In such a case, the traveller benefits from the pro-activity of the agent approach and anticipates and avoids a potential problem.

Additionally, the UMTS MPTA may need to move their agents into the fixed infrastructures to reach computer resources that cannot be integrated into the UMTS MPTA.

7.2 Inter-Operation Between Agents and Workflow

The agent design model was born from a blending of roots from artificial intelligence and transaction systems. In regard to the latter, other models such as workflow have come to mature and are closely related to agent applications.

Relationships between workflow and agent models is becoming very important to several application domains. In the case of personal travel assistance, the relationship between travel agents and corporate approval procedures should be considered. On one hand, the practical matter of agent application (as in this PTA example) indicates a need to understand and inter-operate with other such technologies that are already established. On the other hand, understanding and comparing both underlying models can be explored and tested within the context of FIPA directions and its relationships to other evolving standards.

8 References

- [CENgdf] Geographical Data Files. European Committee for Standardisation for Geo Points.
- [FIPA00023] FIPA Agent Management Specification. Foundation for Intelligent Physical Agents, 2000.
<http://www.fipa.org/specs/fipa00023/>
- [FIPA00026] FIPA Request Interaction Protocol Specification. Foundation for Intelligent Physical Agents, 2000.
<http://www.fipa.org/specs/fipa00026/>
- [FIPA00027] FIPA Query Interaction Protocol Specification. Foundation for Intelligent Physical Agents, 2000.
<http://www.fipa.org/specs/fipa00027/>
- [FIPA00028] FIPA Request When Interaction Protocol Specification. Foundation for Intelligent Physical Agents, 2000.
<http://www.fipa.org/specs/fipa00028/>
- [FIPA00032] FIPA Dutch Auction Interaction Protocol Specification. Foundation for Intelligent Physical Agents, 2000.
<http://www.fipa.org/specs/fipa00032/>
- [FIPA00054] FIPA Query Ref Communicative Act Specification. Foundation for Intelligent Physical Agents, 2000.
<http://www.fipa.org/specs/fipa00054/>
- [FIPA00070] FIPA ACL Message Representation in String Specification. Foundation for Intelligent Physical Agents, 2000.
<http://www.fipa.org/specs/fipa00070/>
- [FIPA00079] FIPA Agent Software Integration Specification. Foundation for Intelligent Physical Agents, 2000.
<http://www.fipa.org/specs/fipa00079/>
- [FIPA00081] FIPA Audio-Visual Entertainment and Broadcasting Specification. Foundation for Intelligent Physical Agents, 2000. <http://www.fipa.org/specs/fipa00081/>
- [FIPA00082] FIPA Network Management and Provision Specification. Foundation for Intelligent Physical Agents, 2000.
<http://www.fipa.org/specs/fipa00082/>
- [FIPA00083] FIPA Personal Assistant Specification. Foundation for Intelligent Physical Agents, 2000.
<http://www.fipa.org/specs/fipa00083/>
- [ISO639] Codes for the Representation of Names of Languages. International Standards Organisation, 1988.
<http://www.iso.ch/cate/d4766.html>
- [ISO3166] Codes for the Representation of Names of Countries and Their Subdivisions, Part 1: Country Codes. International Standards Organisation, 1997.
<http://www.iso.ch/cate/d24591.html>
- [ISO4217] Codes for the Representation of Currencies and Funds. International Standards Organisation, 1995.
<http://www.iso.ch/cate/d23132.html>
- [OPS] Open Profiling Standard. Netscape, Inc. et al., 1999.
<http://developer.netscape.com/ops/ops.html>