ESPRIT Project No. 25 338

Work package H

User Access

Requirements

ID:	Requirements	Date:	9.12.97
Author(s):	Elcha Triep, Fast e. V.	Status:	
Reviewer(s):		Distribution:	Internal

Change History

Document Code	Change Description	Author	Date
Document-Name	First version of document. No changes.	Triep	9.Dec.97

1 Requirements

The aim of this document is to analyse and describe the requirements to the User Access work package. Most of the issues addressed in this document are the result of the numerous meetings between partners and requirements description done by the different work packages, specially pilot application I and II.

1.1 Overview of the User Access

As often mentioned, one of the Follow Me key objective is to enable users to access the system through a variety of different media without losing quality of interaction. The intention of having as far as possible a generic approach , enabling additional devices to join the system without the need to modify the existing interfaces, drove us to the use of Java enabled devices. Unfortunately the nowadays presence of Java enabled devices in the market is still very poor. Therefore, our approach is now based on dealing with two types of devices , interactive and output only devices. Within the last ones, we consider fax, telephones and SMS (not Java enabled devices). The interactive devices are input/output devices (all Java enabled).

This way the User is mobile and can access Follow Me trough the different devices. User Access is the contact point between Follow Me and the User.

When the User wants to contact Follow Me, he access the Follow Me environment through an entry point. This entry point links the User to the Agent. They communicate each other through a *delivery*. I.e. the Agent shows the User what services are available in the "system". The *delivery* generated by the Agent contain also the expected response from the User and the way the Agent will be notified about this response. Once this response is communicated back to the Agent, he generates the next delivery. The User Access acts as the communication bus between Agents and Users. UA receives the *delivery* requirement from the Agent, adapts the data to the corresponding device format where the info will be delivered and send the User the data. UA will also get the answer from the User and give it back to the Agent.

Within Follow Me, the Agents need an environment to live in, this environment is a place[1].

The UA always will be located at a place. If the User wants interactively to join Follow Me, he has to run a place. (See 1.2 General Requirements).

The Agent is the main object within Follow Me environment interacting with the User Access. The Agent will require UA the desired delivery and UA will provide transparency with respect to the place where the delivery will take place, e.g. the Agent wants to deliver the User a fax, the local UA to where the Agent is located realises that this host(place) hat no fax-modem card attaches to it. The local UA contact (through a UA trader or at the beginning through an indication in the personal profile) another UA located at a place which hat a fax-modem card attached to it.

---

[1] Architecture internal 1.1 deliverable, Place is defined by the work package A- Architecture.

1.2 General requirements

This part of the document addresses general requirements of the system where the UA seems to be involve, requirements from others work packages to the User Access and requirements from User Access to other work packages.

The following list was composed with the collaboration of the Follow Me partners and the solutions will be placed on the top of the architecture of the system developed by work package A.

General requirements of Follow Me:

- Users are able to run a place only if they have JVM 1.1 [2]. Java enabled devices will be JVM 1.1 compilant by themselves or by whether the user of the *Cage*[3] or *Java Activator*[4].

- Support of SMS gateways.

- Follow Me allows the user to be mobile providing users access from different end devices . UA will develop such access , enabling interaction between users and Agents through Java enabled devices and only one-way communication (output features) for not Java enabled devices such as fax and phones.

- UA is always located at a place. The User needs to run a place to join Follow Me.

- UA is capable to handle applets, XML and forms.

- UA will provide transparency of location, i.e. an Agent will always require a local UA the delivery of a document. UA will contact another UA(e.g. located at a place which hat attached the necessary device features) if necessary to carry on with the delivery.

Requirements from UA to the system- other work package

- Existence of a User Access Trader. The UA Trader will provide addresses from UA at places which devices with the desired features are attached.

- Mark up language to be used by Agents and UA in order to deal with the information retrieved by the Agent. It could also describe the input required from the user and the way to give back this info to the Agent - XML .Clear interfaces between Agents and UA:

    - Messages/methods types/event notification

    - Content of the messages/parameters of the methods

- If the documents given to the UA by the Agent contains references to others documents, e.g. URL, references to documents stored in IS, etc., UA needs eventually to access to the IS and also a way to understand and resolve these references.

- UA receives deliveries from Agents. This delivery contains : data itself and desired layout of the data according to the device where the info will be displayed.

---

[2] Users need JVM 1.1 to run a place. Architecture of follow Me.

[3] Cage is an APM product which enable the secure running of an applet in a remote machine instead of running in the user clients. As a side effect, it led users using JVM 1.0 (or any other version) feel like running JVM 1.1.

[4] User Access Survey .

www.javasoft.com/products/activator

1.3 Pilot applications requirements

- Enable users of Netscape 3.0 (not JVM 1.1 capable) [5] to join Follow Me.

- The UA has to provide the Agents with feed back mechanisms:

    - Error catching and communicating back the Agent

    - Interactive communication : handling user response, etc and giving that back to the Agent.

- Conversion from text into fax.
- Support for rules that are triggered when UA adapts the layout of what has to be displayed to the actual device. These rules are provided by Etel++ for Pilot Application II.
- UA getting characteristics of a terminal:
    - Type (phone, fax, etc.)
    - Direction of use (input, output, both)
    - If the device is Java enabled or not.
    - Resolution of the screen : number of colours, number of lines, length
    - If the device supports sound, stereo, pictures
    - Memory
    - Mips rate
    - If it has a secondary storage, etc.

1.4 Proposal for needed features not covered by the User Access

The need of using additional tools e.g. graphic tools, etc., become obvious when , for example, we deal with the display of the same image in two different size displays.
The following list of requirements are desired features of the Follow Me system, not covered by the User Access.
We propose to have a server or libraries including those features within Follow Me. Thus, each library or service can be reached by every task agent that needs these services. Such libraries can be composed by existing specific goal oriented tools, like Photoshop for Photos. We have to choose the closest to the needs of our pilot applications.

    - Graphic tool
    - Photo/Image design and production tool:
        - For a colour image : reduce its size by a factor of x%, reduce its definition by factor x%, make it black and white, crop it (keep x% of the image around its centre, kill the edges), provide an algo that gives temporal improvement: the more the time you spend sending the image, the better the resolution.
    - Conversion from text into speech

1.5 Use cases

Users to be reached through non Java enabled devices
The first case to be consider in this paper addresses the problem of providing a user with no Java enabled device with the information collected by the Agent. The information may include a very short message or a large description of theme user interest areas. The communication is only one-way, the devices are output only devices.


[5] Netscape 3.0 is not capable of running JVM 1.1.
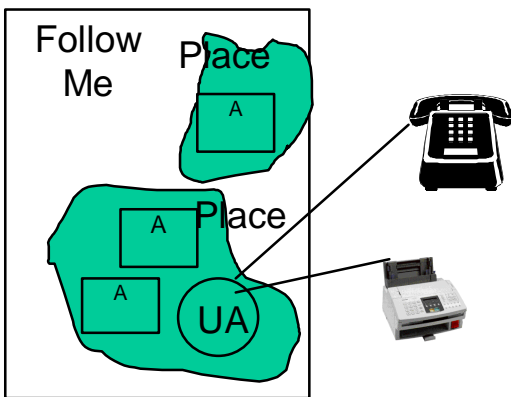
1. The Agent realises that he has to deliver the user some info.
2. The Agent prepares a delivery containing, the data to be deliver, e.g. A message indicating a interesting activity in the banking shares and the new values of the shares, the desired layout to display the info and info about how and where to contact the user, e.g. telephone number. This delivery is a XML document indicating also different layouts styles according to different devices types (what to display when). For the first version of UA, the Agent has to provide audio files if the device to be address is a telephone. The same has to be supposed by different photos sizes, etc.
3. The Agent contacts the local UA.
4. The local UA will contact , when necessary another UA to carry on with the delivery
5. The local or remote UA parses the document delivered by the Agent, retrieves data when necessary, e.g. the document contains URL's, and display it into the corresponding device.
6. Message to the Agent : delivery OK, Error, etc.

User interacts with Follow Me through Java enabled devices (JVM 1.1 enabled)
The second case to be address in the paper considers a user interacting with Follow Me through a Java enabled device Thus, the user is capable to run a place and he is within the Follow Me environment.
Using the same example as for the previous case and we suppose the user is using a browser.
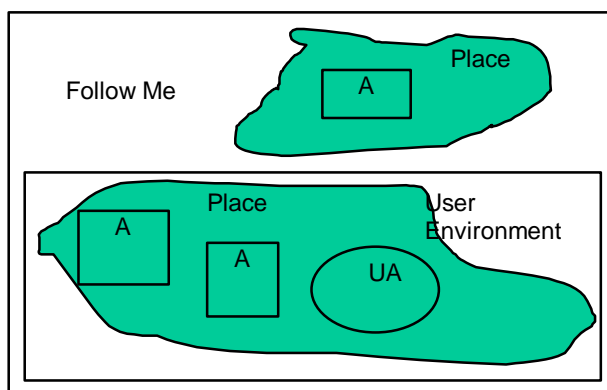
1. The user runs a place within the Browser (Perhaps, to run a place means opening the Browser with a URL from or html/xml form).
2. The User completes the form, giving his name and the name of his agent.
3. UA call the Agent, one has to remember, the user is within Follow Me environment, with the message "User getting in contact with Follow Me @IP address, UA address" .
4. The Agents prepares a delivery to show the user. The delivery contains as data, the available services within F.M. the user can "work" with. The delivery contains also, the way the data has to be presented to the user and for each possible response from the user, the way UA will inform the Agent about this. All these things are described also within a XML document.
5. Agent contact the UA at the specified address and requires it to make a delivery. UA send the delivery to the User.

6. According to the XML document received from the Agent, the UA generates the desired event or message to be received by the agent [6] - the exact method has to be agreed with Autonomous Agents Work package.- indicating the event produced by the user, e.g. look the banking shares values. Example AgentName.show(banking shares values)

7. The Agent prepares a new delivery consisting of the values of the banking shares with the corresponding layout to each device and send the UA a new delivery. The steps from 4 to 6 will be repeated as long as the user and the agents interact each other.

8. We can imagine that each applet or form will support an Exit command. When the user presses Exit, the communication will be broken down. UA will generate an event to be received by the Agent , telling him "End of Session with User".

User that join Follow Me with Netscape 3.0 (non JMV 1.1 enabled)
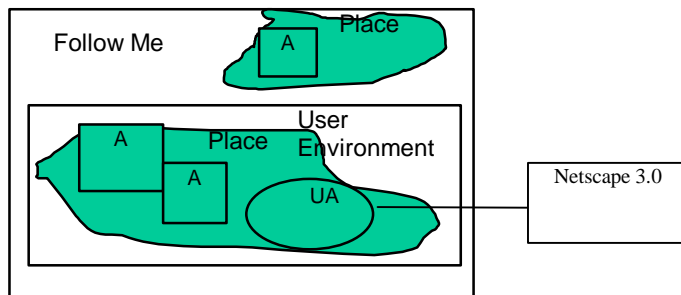
Follow Me system handles Java enabled devices without making a difference between JVM 1.0 and JVM 1.1. This transparency will be solved by using the Cage APM product or the Java Activator.

Cage

The user access the applets trough the cage. He does not need to add software to his system.

Java Applicator

The user needs to install a plug in order to run a JVM 1.1 on a previous JVM.

Taking these special consideration, the user will proceed like in the previous case.

---

[6] We are considering generating events or messages or just giving the agent the completed form back. The disadvantage of the last method is that the agent need to have a parse and that takes more much time than parsing form at UA. The final decision will be taken with AA work package. But we are considering also supporting more than one possibility.

1. The Agent realises that he has to deliver the user some info.
2. The Agent contacts the UA Trader in order to discover which UA can attend the delivery of info.
3. The Agent prepares a delivery containing, the data to be deliver, e.g. A message indicating a interesting activity in the banking shares and the new values of the shares, the desired layout to display the info and info about how and where to contact the user, e.g. telephone number. This delivery is a XML document indicating also different layouts styles according to different devices types (what to display when). For the first version of UA, the Agent has to provide audio files if the device to be address is a telephone. The same has to be supposed by different photos sizes, etc.
4. UA parses this document, retrieves data when necessary , e.g. the document contains URL's, and display it into the corresponding device.

```
┌─────────────────┐        ┌─────────────────┐
│      Agent      │        │      User       │
│                 │        │     Profile     │
│                 │        │                 │
└─────────────────┘        └─────────────────┘
```