



# Collaboration and Coordination Infrastructure for Personal Agents

Smart Reminder - Personal Assistance in a Mobile Computing Environment  
Department of Multimedia Computing - University of Ulm - Germany

In today's rapidly changing world, people spend more and more time with the coordination of information. In spite of the large amount of technical tools available, like phone, fax, email or the WWW, there are research studies showing that for many people this overload of information and communication activities is simply too much. In contrast to the introduction of ever new technology, productivity of professionals often even diminished. So today's computers often fail to deliver their promise of being helpful productivity tools.

Personal Agent Technology offers a promising way out of this dilemma. Like a good secretary they allow their users to delegate routine tasks and get them finished without any further user intervention.

Think of organizing a meeting in your company. Instead of sending a dozen mails back and forth or phoning all the participants a number of times to fix a date, reserve a suitable meeting room etc., you just tell your personal diary agent to do all this for you. All you have to do is to specify who should join the meeting, a time frame when the meeting should happen and if a meeting room is required and -if so- which properties (e.g. a beamer) this room should have. The rest is up to the personal agents of the participants. After a few minutes you will get a notification with the arranged date and room.

In order to make this happen, the agents need to

cooperate in a complex way. They have to agree on a time slot for the meeting and contact a room reservation service to reserve a suitable room. It is reasonable to encapsulate most of the basic functionality in a middleware component that releases the agent programmer from dealing with details of the underlying communication system, database engine or application server. It is the aim of the CIA project to realize such a framework for personal agents that provides:

- ♦ Easy implementation and integration of software agents.
- ♦ Powerful but still easy-to-use communication features.
- ♦ Easy and location transparent access to agents of other users and to external services.
- ♦ Integrated support for commonly needed features like persistency, security and mobility.
- ♦ Support for flexible and device independent user interaction.

All personal agents of one person are concentrated in a virtual construct called Agent Cluster that can be spread out over a multitude of different physical devices.

## Smart-Reminder - a personal agent in the CIA framework

The Smart-Reminder is a personal agent in the CIA framework which offers a smart reminder service based on the current context of its user. Whenever a user Alice meets another Smart-Reminder user Bob e.g. on a corridor or in an office the two Smart-Reminder agents involved will detect this encounter and exchange identity information. This information is then presented to Alice and Bob e.g. via a head-mounted display or as speech output.

In a next step, Alice's Smart-Reminder agent contacts her other personal agents searching for dates and tasks related to Bob. That way Alice has all the relevant information regarding the person met immediately available. She will be informed about upcoming dates like a planned lunch with Bob the next day and may then ask him if this date is still valid. By displaying tasks related to the encountered persons she will never forget to ask Bob important things like the status of the development progress of a joint project.

### Technical details

All components of the CIA framework and the personal agents are written in Java ensuring easy portability to a

large range of devices. Encounters are detected using Bluetooth short range radios. The Bluetooth adapters are constantly inquiring their neighborhood. If any new devices are detected they are trying to establish a connection which is then used to transmit ID data etc. We are using Linux OS on wearable PCs from Xybernaut. As Bluetooth stack we chose Affix. The Bluetooth functions are accessed from Java using a self-written JNI adaptor.



For more information visit <http://cia.informatik.uni-ulm.de/>