

**IST Amigo Project
Deliverable D4.7**

**Intelligent User Services
1 – Introduction**

IST-2004-004182
Public



Project Number	:	IST-004182
Project Title	:	Amigo
Deliverable Type	:	Report

Deliverable Number	:	D4.7
Title of Deliverable	:	1 - Introduction
Nature of Deliverable	:	Public
Internal Document Number	:	amigo_1_d4.7_final
Contractual Delivery Date	:	30 November 2007
Actual Delivery Date	:	14 January 2008
Contributing WPs	:	WP4
Author(s)	:	Jürgen Baum (Editor, Fraunhofer Institute SIT), All

Abstract

This document is the introduction to the deliverable D4.7, final programmer's guides for the Amigo Intelligent User Services IUS. The overall goal of this work package was to develop the scientific basis and to implement a number of Intelligent User Services offered by the Amigo environment. The results of this work package serve as the building blocks for the realization of the major prototypes in the area of "home care and safety", "home information and entertainment" and "extended home environment" in the corresponding Amigo work packages (WP5, WP6 and WP7).

Keyword list

IUS, Intelligent User Services, Introduction, Connected home, ambient intelligence, service orientation, context awareness, context management, user modeling and profiling, personalization, awareness and notification, user interface, speech recognition, gesture recognition, multimodal dialogues, home information and entertainment applications, home care and safety applications, extended home environment, trust, privacy, awareness systems, personal security.

Table of Contents

Table of Contents.....	2
1. Introduction to Intelligent User Services (IUS)	3
1.1 Overall Amigo Architecture	3
1.2 Intelligent User Services.....	4
1.3 Structure of the Deliverable.....	6
1.4 References	7

1. Introduction to Intelligent User Services (IUS)

1.1 Overall Amigo Architecture

The Amigo service-oriented architecture consists of a Base Middleware layer, an Intelligent User Services layer, and a programming and deployment framework (Figure 1). The Base Middleware contains the functionality that is needed to facilitate a networked environment. The Intelligent User Services contain the functionality that is needed to facilitate an ambient in-house network. The programming and deployment framework contains modules that facilitate developers in creating an Amigo aware service by providing support for interoperability, security and finding and exporting context information.

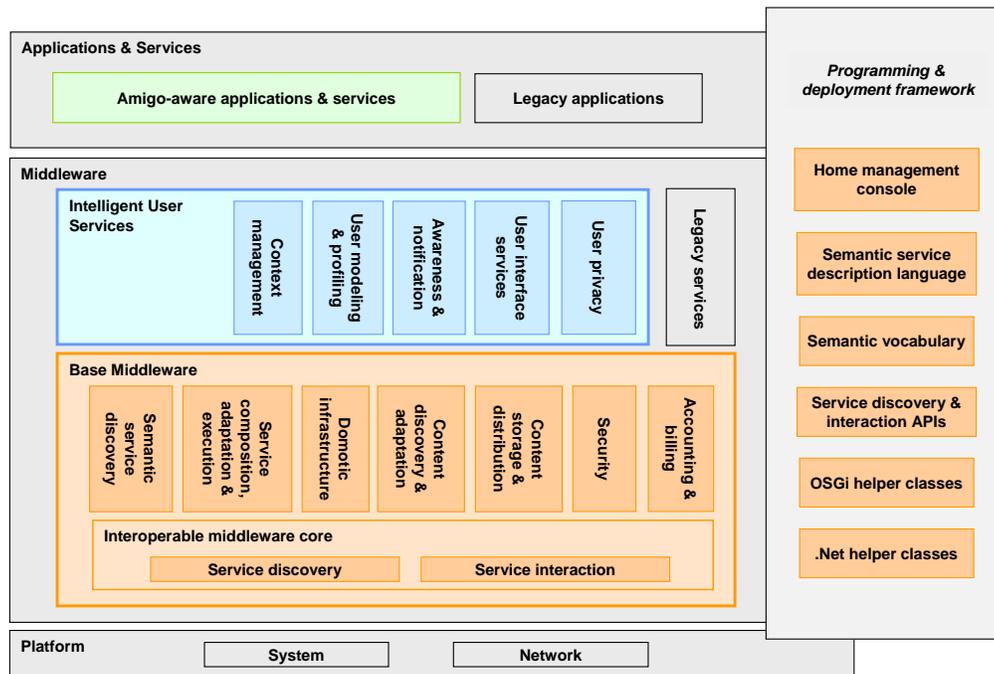


Figure 1: Amigo architecture

The Amigo Base Middleware is a comprehensive and flexible middleware solution for the networked home that can integrate the most important existing technologies in terms of service platforms, middleware protocols, and programming paradigms. This solution is based on the semantics that is used to communicate and discover available services and devices in the network, including the ones that are based on existing communication and discovery standards, such as UPnP, WS, or SLP. Existing hardware and software and new services can be discovered and composed independently. The Base Middleware enables interoperability and integrates heterogeneous service platforms. It provides a generalized use of semantics to represent functional, non-functional and architectural features. Semantic technologies allow for automated reasoning on represented concepts, thus providing the basis for resolution of device and service heterogeneity, service discovery and composition, context-awareness, content discovery and distribution. Advanced mechanisms for ad hoc composition of heterogeneous home resources towards complex applications are integrated. The security mechanisms for authentication, authorisation, and encryption are also provided by the Base Middleware.

Through its Programming and Deployment Framework, the Base Middleware offers modules that facilitate the development of Amigo-aware services in .NET or Java by providing support for interoperability, security and service description to service developers. Amigo supports and abstracts over several important protocols used for discovery and communication. Therefore, heterogeneous services can be integrated into the networked home independently of their underlying software and hardware technologies. A number of the important protocols for discovery and communication are supported in an interoperable way. This allows programmers to select the protocol of their choice while they can still access functionality of services by using different methods. Developers can work, for example, with an 'AmigoService class' which is provided in both C# as well as java. Target networked services come from the four application domains of the Amigo home, i.e., from the Personal Computing (PC), Mobile Computing, Consumer Electronics (CE) and Home Automation domains. The middleware protocols being bridged include: (i) service-oriented interaction protocols, such as those offered by Web Services (SOAP), UPnP (SOAP), and RMI-based infrastructures; and (ii) service discovery protocols such as SLP, UPnP and WS-Discovery. Additional middleware-related protocols being integrated embrace the control/command protocols from the Home Automation domain. This implies that the Amigo system wraps domotic protocols in such a way that they are discovered and controlled by using any of the above methods. For example, an X10 protocol that is used to control a lighting system can be wrapped and offered as a service that runs on an Amigo gateway.

1.2 Intelligent User Services

The Intelligent User Services contain the functionality that is needed to facilitate an ambient in-home network. They broker between users and service providers, and provide context information, combine multiple sources of information and make pattern-based predictions. Information is tailored to user profiles and adapts to the user's situation and changes in the context.

Through their **Context Management Service (CMS)**, the Intelligent User Services deal with the collection of data for establishing context information and transform them into appropriate formats that can be used for further processing. Context sources are, for example, basic sensor data on location, emotional states, environmental parameters, but also data on the presence of people (local and remote) that are derived from the existence of voices (speech detection and recognition front end) and acoustic scene analysis. This also includes detecting gestures of people and movements of objects. In all cases, it is necessary to recognize multiple users and multiple objects and being able to differentiate between them. CMS takes input from a range of different sources, transforms and aggregates them so that it can be used in a higher-level format by other services. Major challenges are to find the appropriate units of aggregation and the mechanism for dealing with missing data. Transformation mechanisms, context representation formalisms, and context ontologies are major components of the context management service, which together with semantic service descriptions allow inferring service parameters by reasoning from this context knowledge. Aggregated context data are also used to derive interaction histories and can be used – in combination with the User Modeling and Profiling Service (UMPS) – to offer predictions about anticipated contexts.

Through their **User Modeling and Profiling Service (UMPS)**, the Intelligent User Services provide personalization and the corresponding personalized services by combining the necessary context information with information about the users. This is done by exploiting profile and preference information that is explicitly provided by the users and, in addition, by deriving user models on the basis of previous interaction, location, and activity histories. User interaction in Amigo is multimodal and includes speech and gestures. This service takes into account the guidelines developed in the task on Privacy and Personal Security Issues (PPS) and addresses the needed “privacy enhancing technology” to be used by other services.

Through their **Awareness and Notification Service (ANS)**, the Intelligent User Services enable to develop applications that allow users to stay aware of any significant change in context with minimal effort. From the system viewpoint, ANS makes application layer services aware of context changes by notifying them. The benefit of using ANS is that applications do not have to care about subscribing to and monitoring of context data. These tasks are handled by ANS. In order to be notified, applications have to register monitoring rules. Once an application has set its monitoring rules, ANS constantly checks them. If one rule evaluates to true, ANS notifies the application accordingly. From the user perspective ANS provides notifications based on the user's preferences and their current context. In order to be notified appropriately, users create an individual user notification profile by using UMPS. The profile describes how and when a user wants to be notified. Before ANS sends a notification to an application, the service checks the notification profile of the user that is to be notified. Based on this profile, ANS sends a notification with an appropriate rendering of intensity. The application receiving the ANS notification implements the notification of the user according to the intensity, for example, using an "ambi-light" for an ambient notification.

Finally, through their **User Interface Service (UIS)**, the Intelligent User Services handle the devices to present their contents and interaction modalities, and cope with explicit as well as implicit user interactions. UIS is informed by UMPS with its privacy enhancing. In the UIS context, an interaction framework was developed that allows for flexible combination and dynamic coupling of multiple devices so that services can be offered in a context-aware fashion, i.e., selection of appropriate devices depending on their "display" and multimodal presentation capabilities. This way, the devices can be adaptive and assume the role of 'smart artifacts' acting in an ambient intelligent environment.

One of the characteristics of the Amigo environment is its multiple-user aspect and the distributed applications for sharing information within and between different locations. This includes the remote presence and ambient sharing, thus linking spaces at different locations, the detection of multiple users in one and different locations. One has to observe that there are different notions of distribution: the distribution within the home, e.g., in different rooms of the same building (in this case the home), the distribution between different homes, and distribution between the home and being outside the home in a mobile context. In all situations, users are able to operate their distributed devices and get in touch with each other's distributed services. This is accomplished by the communication framework **CHES (Community CE-HTML based Experience Sharing Service)**, which provides the support for remote user interfaces and handles the gateways, the communities, the link to UIS, the service discovery, and integration of the services.

Designing for **Privacy and Personal Security (PPS)** right from the beginning is a basic concern of the Amigo project. As a consequence, in the PPS guidelines were produced for developing and integrating privacy enhancing features in the different services, including exploitation of the middleware services for security and privacy. The Amigo approach towards perceived privacy and personal security consists of 3 distinct phases. The objective of the first phase was to gain insight and an overall view of the problem. Here we took a user-centered approach and conducted user studies that resulted in design guidelines. In the second phase, an application prototype was developed to model the privacy for an Amigo environment and to validate and refine the design guidelines. In the third phase a privacy framework was set-up for handling user's privacy in context aware environments.

These services are the building blocks for the realization of the application prototypes in the three application domains of the Amigo project - namely: Home Care and Safety, Home Information and Entertainment, and Extended Home Environment. The role of an application or application service is to select from the pool of the available services those that meet the

required capabilities, combine and orchestrate them to execute the desired work flow. The following paragraph presents a selected, simplified example illustrating how Intelligent User Services should be combined and orchestrated by an application for handling a particular task. A selected, simplified example is provided in the following paragraph: the Shopping List Manager application:

The Shopping List Manager application [Ami06b] is aware of the goods that are stored in the fridge so it can manage the elements that are missing and need to be added to the shopping list. It contacts the User Modeling and Profiling Service to gather information about user's preferences. To get the actual refrigerator content it makes use of the Base Middleware functionality which connects to the refrigerator service to prepare a shopping list. Through the User Interface Service this list is proposed to the user and a negotiation procedure takes place which allows the user to add/delete some goods. The Awareness and Notification Service – configured to be aware of specific products' discounts - supports the user to select special offers and add these products to the shopping list, too. When the shopping list is finalized, the list can be downloaded via the User Interface Service. This application contacts the Context Management Service to get the user device that is currently available and streams the shopping list to the specified device.

Furthermore for developing mock-ups and for evaluating the initial results, an **Interaction Design and User-based Evaluation** task was established. This task consisted of developing mock-ups in order to investigate issues of interaction design and evaluation with users in the context of different "labs" provided by several Amigo partners.

1.3 Structure of the Deliverable

This deliverable consists of the developer's guides for the Amigo Intelligent User Services. The developer's guides introduce the Intelligent User Service, the components and their functionality, and provide the guides and tutorials for setting up and using the service. The following documents comprise deliverable D4.7:

1. Deliverable D4.7 Intelligent User Services - 1: Introduction
2. Deliverable D4.7 Intelligent User Services - 2: Context Management Service (CMS)
3. Deliverable D4.7 Intelligent User Services - 3: User Modeling and Profiling Service (UMPS)
4. Deliverable D4.7 Intelligent User Services - 4: Awareness and Notification Service (ANS)
5. Deliverable D4.7 Intelligent User Services - 5: User Interface Service (UIS)
6. Deliverable D4.7 Intelligent User Services - 6: Community CE-HTML based Experience Sharing Service (CHESS)
7. Deliverable D4.7 Intelligent User Services - 7: Privacy and Personal Security (PPS)
8. Deliverable D4.7 Intelligent User Services - 8: Interaction Design and User-based Evaluation

1.4 References

- [Amigo-D4.2 & D4.3] Amigo Consortium. Deliverable D4.2 and D4.3: Report on detailed Intelligent User Services design and first version of software services. March 2006.
- [Amigo-D4.5] Amigo Consortium. Deliverable D4.5: Amigo Intelligent User Services: Initial prototype implementation and documentation. October 2006.
- [Amigo-D4.6] Amigo Consortium. Deliverable D4.6: Amigo Intelligent User Services: Second prototype implementation and documentation. May 2007.
- [Amigo-D9.5] Amigo Consortium. Deliverable D9.5: Web site for sharing open source software developed within Amigo. March 2006.
- [Amigo-OSS] Amigo Consortium. Amigo OSS Repository - Public Web Site. <https://gforge.inria.fr/projects/amigo/>