

IST Amigo Project
Deliverable D10.4

WP10 Training

IST-2004-004182
Public



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

Deliverable Number	:	D10.4
Title of Deliverable	:	Training
Nature of Deliverable	:	Public
Internal Document Number	:	amigo_d10.4_final
Contractual Delivery Date	:	31-12-2007
Actual Delivery Date	:	10-01-2008
Contributing WPs	:	WP10
Authors	:	Edwin Naroska (Fraunhofer IMS) <i>editor</i> Ioanna Roussaki (ICCS-NTUA) Maddy Janse (PHI) Peter Vink (PHI) Jörg Schmalenströer (PAE) Volker Leutnant (PAE) Reinhold Hüb-Umbach (PAE) Álvaro Ramos (TID)

Abstract

This deliverable presents the material that has been developed within WP10. It includes description of the seminar as well as of the training modules. While the seminar content mainly targets at presenting the features and benefits of the Amigo middleware, the training parts aims at getting people in touch with the actual Amigo software. To this end, the training parts include extensive programming sessions where the user can gain hand-on experience with the software.

Keyword list

training activities, training modules, tutorials, seminars

Table of Contents

Table of Contents	2
Table of Figures	3
1 Introduction	4
2 Goals and approach of the Training Efforts	5
2.1 Goals the Training Efforts.....	5
2.2 Approach to reach the goals.....	7
3 Training module framework	8
3.1 Training modules.....	9
4 Seminar	11
4.1 Didactic approach	11
4.2 Seminar blocks.....	11
4.3 Seminar handouts	16
4.4 Development process and feedback	16
5 Training	19
5.1 Didactic approach	19
5.2 Training blocks.....	20
5.3 Training documents.....	25
5.4 Development process and feedback	25
6 Sample tutorial walkthrough	29
6.1 Developing a service	29
6.2 Integrating a device into an Amigo network	30
7 Feedback results	32
Appendix	37
Appendix A: Training document template.....	37

Table of Figures

<i>Figure 1: Training module framework</i>	8
<i>Figure 2: Directory structure of a typical training component</i>	28
<i>Figure 3: “Did you plan to use middleware technology before attending the seminar?”</i>	32
<i>Figure 4: “Will you now use middleware technology?”</i>	33
<i>Figure 5: “Do you plan to use Amigo software in the future?”</i>	33
<i>Figure 6: “Are you interested in further information about Amigo?”</i>	34
<i>Figure 7: “Did the Seminar address your current needs?”</i>	34
<i>Figure 8: Evaluation of the course content</i>	35
<i>Figure 9: Evaluation of the course slides</i>	35
<i>Figure 10: Evaluation of the course handouts</i>	36

1 Introduction

The training activities aim at industrial and post-graduate learners. Especially SME participation will be encouraged by coordinating work in this work package with the dissemination activities. Courses are organized in a modular way. Modules addressing the Amigo related topics will be developed in cooperation with the different partners. This modular approach offers the opportunity to combine different modules according to the special needs of the target audience. Among Amigo specific topics, the course also addresses key technologies on which the Amigo middleware is based. This is required as SMEs are not necessarily familiar with modern software paradigms, e.g. ontologies.

To address people all over Europe, the courses will be held at different locations and in different countries. A first successful course took place in Duisburg (Germany) on May 2006. The seminar addressed decision makers who were responsible for the development of new products and applications in the area of smart homes. It introduced complex technical issues in a concise and easy to understand manner. A second seminar was given in November 2006 in Eindhoven (The Netherlands).

This document describes the seminar and training materials that has been created within WP10. The material will be used to run training courses. Moreover, during creation of the training documents a focus has been put on their comprehensiveness. The document shall not only be used to help people during a training course but were also designed to be used at home. That is, the material have been developed so that it can be also used by people that download them from the internet to get used to the Amigo middleware in minimal time. In order to streamline course development WP10 developed a training plan (D10.3) which outlines the training objectives, training outputs, the didactic approach as well as the test and feedback plans.

2 Goals and approach of the Training Efforts

2.1 Goals the Training Efforts

The training efforts target at bringing the Amigo middleware and technology into real products at a wide scale. To achieve this goal, an approach has been developed to create some “distribution channels” for promoting the Amigo technology. A key channel is comprised of the training and dissemination courses which target at SMEs.

In order to reach this goal, SMEs must be aware of the deficiencies of current middleware technology and must understand the benefits that may arise from using modern technology. Further, in order to use the Amigo middleware for their products, developers have to get used to the Amigo system. Finally, the SMEs have to get familiar with the amount of extra effort that is needed to incorporate the Amigo technology in their products.

The following sub-goals are addressed:

- The target audience must be aware of the problems and shortcoming of current technology and how they are going to be addressed by Amigo. Here, the decision makers within the SMEs are especially targeted. This is perhaps not an obvious goal but it must be noted that especially the SMEs don't typically have big development and research teams that are addressing future issues within the field of smart homes. One of the reasons for this is that the current gap between the technology that is available on the market right now and the technology that is currently developed by the research community is quite large.
- The target audience shall become familiar with the user benefits of a modern middleware technology. Benefits are the actual driver of the technology from the user's perspective. Hence, it is important to realize what benefits can be obtained from modern middleware technology and why it is not possible to achieve the same benefits with current approaches.
- The audience must be familiar with the technology on which the Amigo middleware is based.
Modern software modelling techniques, e.g. description logics, are typically unknown to the SMEs that operate in the domotic field. As the majority of Amigo middleware components rely on such techniques, appropriate knowledge must be available at the target audience. The level of knowledge of course depends on the actual attendees. While decision makers only need a high level understanding of these technologies, developers must have a thorough insight and experience in using such technologies.
- The target audience must understand how Amigo technology is going to solve the problems and how to use it in order to create Amigo devices or services. In order to foster application of Amigo technology in real products, the target audience shall learn how to apply the technology for building Amigo aware or compatible devices and services. To help people get started, they shall not only become familiar with Amigo on a theoretical level but also get used to it in practical sessions. To this end, people shall be provided with hands-on training material that allows them to build devices and service by their own. This of course targets mainly developers and to a lower extend decision makers as it requires some programming skills.
- The audience must become aware of the development efforts that are needed to apply Amigo technology to their products. In order to gain the additional functionality offered by the Amigo middleware some efforts to make devices Amigo compatible or Amigo-aware is required. An important aspect is to point out the extent of such efforts as well as how these additional efforts integrate into the typically development chain. It is important for the decision makers to know about this as the additional efforts of course effect the development costs and time. Further, they need to know about the skills that are needed in

order to properly staff the development team. Finally, the developers need information about the additional time and man power that may be required.

As can be seen from the enumeration, the actual goals slightly depend on whether decision makers or developers are addressed. Table 1 summarizes the various goals with respect to the target audience. Among the specific goals, this table also briefly gives a list of prerequisites for the specific topics. Note that for the decision makers the intention was to keep the number of prerequisites as low as possible.

However, it should be noted that in order to successfully bring Amigo to the market, both, the management level as well as the leading developers must be convinced by the advantages of the approach. The training efforts must show that

- Amigo provides a comprehensive set of functions that allow creation of new services and devices and hence provide significant improvement in terms of user benefits
- Amigo requires only a limited amount of additional resources to make devices Amigo-aware or compatible
- The Amigo results and software will be available as open source beyond of the actual project runtime

Topic	Decision makers	Developers
Shortcomings of current technology	<p>Prerequisites: Overview of state-of-the-art home automation technologies.</p> <p>Goals: After the training the decision makers shall have a high level understanding of the problems that arise from integrating current devices using currently available solutions.</p>	<p>Prerequisites: Overview of state-of-the-art home automation technologies.</p> <p>Goals: After the training developers shall have a high level understanding of the problems that arise from integrating devices using currently available solutions. Further, they know technical details about issues that are addressed by the Amigo middleware.</p>
User benefits	<p>Prerequisites: Aware of current domotic technology.</p> <p>Goals: Decision makers shall have a high level view on the benefits of Amigo in contrast to other (simple) approaches.</p>	<p>Prerequisites: Aware of current domotic technology.</p> <p>Goals: Developers shall have a high level view on the benefits of Amigo in contrast to other (simple) approaches.</p>
Middleware basics	<p>Prerequisites: None.</p> <p>Goals: Decision makers shall have an overview on domotic middleware approaches, their advantages and disadvantages.</p>	<p>Prerequisites: None.</p> <p>Goals: Developers shall learn what middleware is good for and how it is typically used in domotic environments.</p>
Amigo technology	<p>Prerequisites: None.</p> <p>Goals: Further, they get an overview on the architecture, technology and components that are used in Amigo. Finally, they shall know how the architecture contributes to reach the overall</p>	<p>Prerequisites: Programming knowledge in Java or .NET.</p> <p>Goals: Developers shall get an overview on key technologies and concepts that are exploited in Amigo. They shall get an overview on the architecture and the components of Amigo. They shall</p>

	Amigo goals.	learn how to use them, how to add new devices and services so that they become able to develop their own Amigo services and devices
Development efforts	<p>Prerequisites: None.</p> <p>Goals: Decision makers shall gain knowledge about the skills that are required to develop Amigo devices and services. Further, they shall receive a rough overview of the resources that are needed to develop Amigo compatible devices and services.</p>	<p>Prerequisites: None.</p> <p>Goals: Developers shall get knowledge about the additional efforts that are needed to build Amigo devices or services.</p>

Table 1: Goals that shall be achieved with respect to different topics and target audience type

2.2 Approach to reach the goals

As it has been pointed out in the previous sections, there are three major means to reach the goal of establishing Amigo technology in the market. To this end, the SME managers as well as leading developers have to be convinced that

- Amigo provides a comprehensive set of functions that allow creation of new services and devices and hence provide significant improvement in terms of user benefits
- Amigo requires only a limited amount of additional resources to make devices Amigo-aware or compatible
- The Amigo results and software will be available as open source beyond the actual project runtime.

As managers and developers have quite different needs and require information at a different level of granularity, it was decided to split the training package into two parts. The first part started before any actual software was available to the public. It is intended to provide high level information to managers. The second part actually consists of the detailed presentation of the Amigo middleware. By applying this approach both target audiences can be provided with the appropriate information without risking to overburden or to bore someone.

Section 4 explains the seminar blocks that have been developed and that are going to be developed while Section 0 gives a description for the hands-on training parts of Amigo.

3 Training module framework

In order to obtain a reusable flexible training concept, training modules have been arranged into a two-day structure as shown in the figure below. The framework consists of two parts which are associated with a complete day. The first part (day) gives a theoretical introduction into the problem domains and how the Amigo solutions are designed.

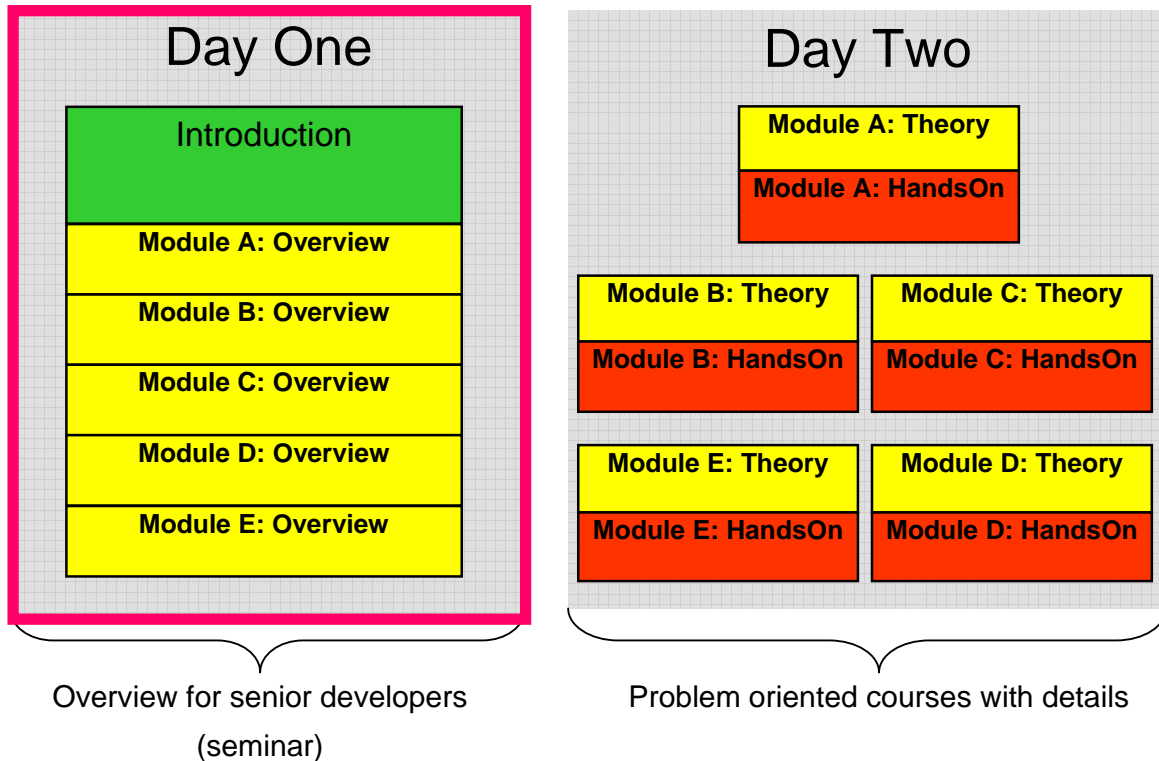


Figure 1: Training module framework

As the typical SME manager or developer is not aware of modern technologies like semantic based approaches to knowledge representation, during the first day the focus has been put on the presentation of the core software and modeling paradigms that are applied within Amigo. Knowledge about these technologies is mandatory for both, decision makers as well as developers. For the management level it is important to notice what kind of benefits can be gained from integrating such technologies. Note that the application of these technologies in real-world products requires a certain amount of additional resources in terms of man power. Further, it requires that the developers are educated accordingly which in turn also puts some additional pressure to the total development budget. Hence, in order to convince the management levels to use these technologies they must be clearly aware of their benefits.

Beside of giving an introduction to the core technologies, the main Amigo architecture and solutions are also introduced during the first day. As a result, the audience will learn the basic concepts applied within Amigo, how the core components are laid out and how they interact with each other at the higher levels. This provides the audience with the first knowledge of how to use the Amigo middleware in order to build an Aml system.

During the second day, the basic Amigo components are explained and trained. To this end, each module consists of two parts: a theoretical part, providing sufficient details in order to apply the

knowledge within the following practical part. Note that the second day addresses developers as some programming skills are needed during the practical parts. Note further, that the attendees of the training parts may choose to use either Java or .NET as deployment framework. Hence, the hands-on training blocks will be available for both platforms.

Depending on the knowledge of the audience, it may be sufficient to attend only the training part. All practical parts are accompanied with appropriate documentation material that briefly introduces the basics. Hence, if appropriate programming skills are present, users may visit only the second day and learn about the details of the Amigo middleware from the documentation.

The actual content of both days is divided into five modules A to E which are explained in the following.

3.1 Training modules

In our opinion, a successful Aml middleware must address issues concerning interoperability, context awareness, user interaction, and security. Due to the importance of these topics, each of the them is devoted a separate training module. Additionally, a module to introduce the problem domain along with the concepts of the Amigo middleware is provided. In detail the following aspects are targeted:

- **Module A: New procedures and innovative approaches in multimedia and home automation integration:** This module covers the introduction to the problem domain as well as an introduction to the Amigo architecture. Hence, Module A gives an overview what is currently available, how future home environments may look like. Further, it explains how Amigo targets the challenges by explaining its major features, components and how they interact with each other in order to provide a smart environment.
- **Module B: Interoperability in managed home networks:** Interoperability is actually one of the most important topics in heterogeneous environments like a domestic home. As Amigo's goal is to spread into real products, a major requirement for the Amigo middleware is to support various existing communication standards and protocols. Further, Amigo does not only provide mechanisms that allow devices to talk to each other but also to get a machine based "understanding" of the functionality of devices. This is a major requirement for real (automatic) interoperability. Due to the importance of this topic, an entire module is devoted to this issue. It introduces the concepts that are applied by Amigo to achieve the interoperability goal.
- **Module C: Interaction design:** User interfaces are what a user immediately gets in touch with. Hence, it is very important for the acceptance of a technology or product. As a result, a special focus in the training program is put on user interfaces and how the user interacts with the system. To this end, UI concepts of Amigo are shown and explained.
- **Module D: Context awareness in Aml-systems:** In order to make a Aml-system smart it is required to make the system aware to the context the user is currently in. This of course requires some kind of mean to describe context information in a human and machine readable form. Further, appropriate mechanisms to collect or extract this kind of information from the available data is required. As a result, Module D focuses on techniques to describe context information using ontologies and how to extract context information from sensor data.
- **Module E: Addressing security and privacy issues in Aml-systems:** If in the future, major parts of the domotic environment are controlled by the home network, security will become one of the major concerns. Actually, if the security issues is not addressed and solved sufficiently, success of the Amigo middleware to become a major software basis for domotic environments is endangered. As a result, the major security problems of modern networks are introduced along with standard approaches applied to establish security.

Further, appropriate means to provide a sufficient security level within Amigo are shown as well.

Table 2 provides the goals for the various modules with respect to the target audience. Note that decision makers (managers) will typically only attend the first day while developers will normally attend both days or only the second if sufficient programming skills and knowledge are present.

Module	Decision makers	Developers
A: New procedures and innovative approaches in multimedia and home automation integration	Goals: After the training the decision makers shall have a high level understanding of the problems that arise from integrating current devices using currently available solutions. Further, they know the benefits of applying Amigo technology for integration.	Goals: Same as decision makers plus providing hands-on experiences with the Amigo middleware. Especially, training on how to use the framework and how to create context sources and develop services, is provided.
B: Interoperability in managed home networks	Goals: Decision makers shall have a high level view on the interoperability issues and how they are addressed in Amigo.	Goals: Same as decision makers. Additionally, developers shall gain experience with implementing and using plain and semantic based service discovery within Amigo.
C: Interaction design	Goals: Decision makers shall have an overview on the UI concepts that are supported by Amigo.	Goals: Same as decision makers. Additionally, developers will gain experience with creating user interfaces based on the Amigo middleware.
D: Context awareness in Aml-systems	Goals: Decision makers shall learn about the importance of context-awareness for future products.	Goals: Same as decision makers. In addition, developers will learn how to semantically model context information using ontology tools and languages.
E: Addressing security and privacy issues in Aml-systems	Goals: Decision makers will become sensible to the security threats within a connected home. Further, they get a brief introduction on technologies that are common to establish security and how these technologies are applied in Amigo.	Goals: Same as decision makers. In addition, developers will gain hands on experience by designing and implementing security using the concepts and mechanisms provided by Amigo.

Table 2: Goals that shall be achieved by the various training modules with respect to the target audience

4 Seminar

There are eight seminar blocks available. All target at the providing sufficient information to the management level without giving too much detailed information. The pursued goals are:

- Showing that there are significant gaps and deficiencies that hamper development of a real smart ambient system.
- Showing that Amigo provides the solution to these problems
- Briefly explaining the Amigo technology as well as the underlying technology.
- Showing that Amigo software and solutions will remain beyond of the actual project end.

4.1 Didactic approach

The seminar blocks were implemented in form of a lecture / presentation using Microsoft PowerPoint. While the actual presentations typically did not provide significant room for interaction with the audience, it was taken care during the actual seminars, that a sufficient amount of time was available for discussions.

4.2 Seminar blocks

Six blocks have been chosen as seminars in order to provide a sufficient amount of information starting from a general introduction to the problem domain. Thereafter, the Amigo architecture is explained at a higher level.

The blocks has been chosen to gradually start from the basic problems of Aml home systems and what kind of challenges arise when designing such systems. An overview on the blocks is given in Table 3.

Lecture name	Goals	Component
Ambient Intelligence Environments	After the presentation the audience knows the benefits of modern Aml home systems as well as the challenges that arise when building such systems. The presentation is intended to give a motivation for the Amigo approach and architecture.	A
Amigo Architecture	After the presentation the audience knows the core elements of Amigo. They know the basic vocabulary that is used within Amigo in order to be able to follow additional detailed explanations of various Amigo components.	B
Service Discovery for Context Aware Applications	After the presentation the audience knows how services within the Amigo environment are discovered and how context-aware service discovery is done.	B
Opportunities and Challenges of Speech Processing in a Networked Home	The audience gains knowledge about the major components of a speech processing system as well as what can be achieved with speech processing.	C

Building adaptable user interfaces	Attendees will learn how to build adaptable user interfaces with the Amigo middleware.	C
Context-Awareness Concepts and Principles: Introduction	After the presentation the audience knows the benefits of context awareness as well as how it is described within Amigo. Further, a short introduction to ontologies is given.	D
Semantic Modelling of Context Information	Attendees will gain knowledge on the ontology tools and languages and how they can be used to represent context information in Aml applications.	D
Security and Privacy	The lecture makes the audience sensitive to the security problems and shows the core techniques that are used to implement secure systems nowadays. The audience learns about the used approaches within Amigo.	E

Table 3: Overview on seminar blocks

4.2.1 Ambient Intelligence Environments

Type: Lecture.

Module: A

Author: Viktor Grinewitschus.

Goals: After the presentation the audience knows the benefits of modern Aml home systems as well as the challenges that arise when building such systems. The presentation is intended to give a motivation for the Amigo approach and architecture.

Description: During this block the participants will get an idea on the look and feel of an Aml home system. The focus is put on outlining the benefits of such systems as well as the challenges that arise when such systems are designed. In special, the course will introduce the user requirements and the problems that must be solved by a middleware of an Aml home system. The discovered results will be mapped to existing approaches and show why they fail. Next, a set of interesting scenarios will be given that show how a future smart home will present itself to the user and which kind of services can be expected. These examples will serve as a motivation for the various components and solutions that are provided by the Amigo middleware.

Versions: There is a 20 minutes short version as well as a 45 minutes long version available. For both versions the appropriate slides in PowerPoint format are available. For the long version also a video has been recorded.

4.2.2 Amigo Architecture

Type: Lecture.

Module: B

Author: Peter Vink.

Goals: After the presentation the audience knows the core elements of Amigo. They know the basic vocabulary that is used within Amigo in order to be able to follow additional detailed explanations of various Amigo components.

Description: The lecture will provide an overall introduction to the Amigo architecture explaining the Amigo middleware and its components. Major issues addressed on this lecture are service discovery, interoperability, service composition, management of context information, as well user profiles and user models. Further, an outline will be given on how to setup an Amigo network and how to write Amigo services.

Versions: There is a 20 minutes short version as well as a 45 minutes long version available. For both versions the appropriate slides in PowerPoint format are available. For the long version also a video has been recorded.

4.2.3 Context-Awareness Concepts and Principles: Introduction

Type: Lecture.

Module: D

Author: Ioanna Roussaki.

Goals: After the presentation the audience knows the benefits of context awareness as well as how this is established in the Amigo middleware solution.

Description: During this lecture we will point out the advantages of deploying context-aware services and the necessary issues that need to be addressed in order to reach the pervasive computing vision. The concepts involved in the context-awareness (CA) and pervasive computing domains will be identified and properly defined, while the strong interdependency of these domains will be highlighted. Initially, definitions of context and context-awareness concepts will be provided, while subsequently elaboration on the context characteristics and nature, as well as details on the sensors required to support context-aware systems. Then, the most popular context-aware applications and prototypes will be presented. Finally, the framework built to support CA in Amigo will be described. Thus, the CA architecture implemented and the CA functionality provided will be presented, while an overview of the contexts sources used will be given. Finally, the critical context-awareness aspects involved in service provision within Aml home environments will be highlighted.

Versions: There is a 30 minutes short version as well as a 45 minutes long version available. For both versions the appropriate slides in PowerPoint format are available. For the long version also a video has been recorded.

4.2.4 Service Discovery for Context Aware Applications

Type: Lecture.

Module: B and D

Authors: Maarten Wegdam and Aart van Halteren.

Goals: After the presentation the audience knows how services within the Amigo environment are discovered and how context-aware service discovery is done.

Description: The lecture will give an overview of service discovery protocols, including the general principles behind service discovery and a high-level overview of the popular service discovery protocols Jini, UPnP, Service Location Protocol (SLP) and UDDI. The second part of the lecture extends service discovery to context-aware service discovery. It discusses context-aware lookup of services and context-aware registration of services.

Versions: There is a 20 minutes short version as well as a 45 minutes long version available. For both versions the appropriate slides in PowerPoint format are available. For the long version also a video has been recorded.

4.2.5 Opportunities and Challenges of Speech Processing in a Networked Home

Type: Lecture.

Module: C

Author: Reinhold Häb-Umbach.

Goals: The audience gains knowledge about the major components of a speech processing system as well as what can be achieved with speech processing. Further, it shows how to exploit the speech signal for physical user tracking.

Description: This lecture consists of two parts. The first part gives an overview of the components of a speech signal processing system, e.g. microphones, loudspeakers, multi-channel audio cards, etc. Next an introduction to typical speech signal processing tasks, their performance, resource requirements and limitations is given. This includes a discussion of single- and multi-channel speech enhancement methods, acoustic echo compensation techniques, hands-free communication etc. The section is concluded with pointers to related internet sites and literature on the topic.

The second part discusses the state-of-the-art in automatic speech recognition for human-machine communication. The performance, resource requirements and limitations of natural language speech user interfaces are discussed. Further, it is shown how context information can be gleaned from speech signals, since speech conveys more than its verbal content. Examples include speaker identification, speaker change detection and speaker position estimation and tracking. Again this part concludes with references to further information.

Versions: There is a 20 minutes short version as well as a 45 minutes long version available. For both versions the appropriate slides in PowerPoint format are available. For the long version also an video has been recorded.

4.2.6 Security and Privacy

Type: Lecture.

Module: E

Author: Edwin Naroska.

Goals: The lecture makes the audience sensitive to the security problems and shows the core elements techniques that are used to implement secure systems nowadays. The audience shall learn about the key approach that is used within Amigo to enable security.

Description: This course part will give an overview over today's security threats and introduces threats to a connected home. Beside of the example driven introduction the participants will get first ideas of defending strategies like data encryption using private and public keys. Further, it will be briefly explained how messages that are send across an unreliable channel can be authenticated using public key techniques. Finally, the Kerberos protocol can be used to authenticate clients to multiple servers in a secure manner.

Versions: There is a 20 minutes short version as well as a 45 minutes long version available. For both versions the appropriate slides in PowerPoint format are available. For the long version also an video has been recorded.

4.2.7 Semantic Modeling of Context Information

Type: Lecture.

Module: D

Author: Ioanna Roussaki.

Goals: Attendees will gain knowledge on the ontology tools and languages and how they can be used to represent context information in Aml applications.

Description: An introduction on ontologies will initially be provided. This will focus on their advantages and shortcomings as well as on the languages and tools used for ontology modelling. Then, various popular context ontology solutions that have been proposed by the research community will be presented. Subsequently, the Amigo semantic framework will be presented, focusing on the context ontologies introduced for the Aml home. The context semantics range from a context ontology vocabulary, which provides a taxonomy of the various context parameters involved in pervasive service provision, to a context ontology language that is adequate for the representation of all the distinct and heterogeneous contextual parameters identified in the vocabulary. Additionally, the most important ontology-based context management systems paradigms will be briefly presented, while the advantages and disadvantages of these various approaches will be identified. Finally, a short demonstration on how a context ontology can be built using a popular ontology editor will be provided.

Versions: A 45 minutes long version is available.

4.2.8 Building adaptable user interfaces

Type: Lecture.

Module: C

Author: Christian Ressel.

Goals: Attendees will learn how to build adaptable user interfaces using the Amigo middleware.

Description: The lecture explains how user interfaces that adapt to the user's needs can be build using the Amigo middleware. Beside of the feature to modify the behaviour and structure of the graphical user interface according to the user's preferences, another important aspect of the Amigo user interface is its ability to seamlessly integrated various kinds of devices and services into a single homogeneous UI. In this lecture, the means and techniques to implement these features are explained. Further, the information that must be provided by the devices or services so that they can be integrated automatically are discussed.

Versions: A 45 minutes long version is available.

Status: Under development.

4.3 Seminar handouts

Beside of the actual presentations the attendees are provided with handouts. The handout material consists of the slides augmented with additional information and a more in-depth explanation of the key topics. As a result, the handouts provide a valuable extension to the oral presentations.

4.4 Development process and feedback

The development process of each seminar block includes several feedback steps where content as well as the performance of the presenters are checked and improved. In detail, the following steps are done:

1. Determining block content. In order to develop a comprehensive set of seminar blocks that are well aligned, the content of each block is determined in collaboration with the partners from WP10. This ensures that all important topics are covered and there is no unintended overlap in the contents between the blocks.
2. Slide and content layout guidelines. In order to provide a consistent look and feel among the seminar blocks, a template for the slides have been created. This template is used by all the presenters which defines layout, font style, size and color. Further, a guideline for developing the slides and the content has been designed. The rules are presented in Section 4.4.1.
3. Early style and content alignment. In order to fine tune alignment between the seminar blocks, preliminary versions of the slides are reviewed at the meetings and suggestions to improve the presentations are generated. Further, the partner check the content against their own presentation in order to find overlapping parts, inconsistencies or gaps. This process ensures that the content as well as the style is consistent among the different blocks which are typically coming from different partners.
4. Dry run. After the seminar blocks are developed, the close to final seminar versions are presented to the partners in order to obtain some final feedback information. Feedback information covers slides as well as presentation style.
5. User feedback. At the end of each seminar users are requested to fill out a feedback form in order to provide improvement comments. The feedback forms are collected, analyzed and the results are send to the partners in order to help improve the seminar blocks. If required, the feedback data is discussed at the regular partner meetings.

Note that some of the steps listed above are done only once (e.g., steps 1 and 2) and others must be repeated for each seminar block that is developed.

4.4.1 Presentation / slide guidelines

In the following a set of guidelines shall be used during block development to align the style of the presentations accordingly:

Slide style:

- Use the seminar PPT file available from Quickplace as a template and stick to the font style and font size requirements. Ensure that the font size is always big enough to be readable even at a larger distance.
- Do not use complete long sentences (if there are no good reasons to do so). Text should be short and to the point. Any further explanation may be provided in the notes section of the PPT slides.
- Do not cram too much information into the slides. Use the slides as a guide to your talks rather than a complete picture of the content.
- Use a meaningful outline for each slide. The outlines help the audience to navigate through the presentation.
- Use bullets to structure the slides. Do not use more than 2 lines per bullet and more than 4 to 5 bullets per page.
- Use colours and font style variations *carefully* to point out some important statements.
- Do not use "All Caps" for body text.

Slide content:

- Do not forget to add a title page to the presentation. See template file for further information.
- Add a outline page at the beginning of the presentation. It is not required to repeat (this holds especially for short presentation) the outline page at each topic change during the talk.
- Use graphics to illustrate complex topics. Use examples to explain theoretical content. This will help the audience grasp what you are talking about.
- Keep in mind what the goal of the seminar is. Note that the seminar addresses both: decision makers as well as developers.
- Conclude with a summary. A summary slide is recommended but not required.

Presentation:

- Face the audience, do not talk to your screen or note cards. Speak slowly and project your voice.
- If needed, use laser pointers to emphasis specific points on the slides.
- Make sure that you stick to the time limit. As a rule of thumb each slide takes about 1.5 minutes.
- Leave enough time for discussion (typically 5 minutes per presentation). Repeat questions that are coming from the audience so that everybody can listen to them.
- Present from memory, do not read text from cards.

4.4.2 Seminar feedback form template

The feedback from template for seminar sessions are given in **Error! Reference source not found.**

5 Training

In contrast to the seminar blocks which consists of purely lecture type parts, the training blocks contain both, theoretical parts and practical parts. However, the only purpose of the theoretical parts is to provide the attendants with the knowledge needed for the following practical parts.

Nevertheless, the documentation for the training parts have been designed so that people do not necessarily need to attend the training sessions. Instead, major efforts were devoted to develop documentation material that can be used by people to run the practical parts at home.

5.1 Didactic approach

For the training course a traditional didactic approach is used. It comprises the following stages (steps): awareness, acquisition, application, utilization and maintenance. The first five steps are applied during the training course while the last step is intended to be done outside of the training session (e.g. at home). Further, where possible and useful a spiral approach is taken. Each training block consists of several phases:

1. **Awareness and acquisition:** Training starts with an traditional passive lecture based part where the knowledge required for the following practical parts are explained. This lecture part is based on the appropriate seminar blocks where the general problem and solution technologies have been introduced. The lecture part in the training approach just adds some basic information about the Amigo approach that is required in order to start with the practical part. After the theoretical part is finished, training continues with the practical part.
2. **Application:** In the next phase the participants use a computer to implement a set of simple tasks. Task assignment and the steps required to fulfil the task are explained in the documentation. The task in the second phase are just designed to help people get familiar with the Amigo solution. Hence, each step needed to fulfil the assignment is detailed described in the documentation so that participants can easily find out how to perform a specific operation.
3. **Utilization:** Once people became familiar with the Amigo framework and the libraries that are needed for the specific training block, the next training phase starts. During this phase participants practice their knowledge and techniques learned in the previous phase. To this end, they are assigned a set of medium complex task. Note that for these assignment not detailed procedural steps are provided (in contrast to phase 2). Instead, participants are provided with a reference solution so that they can compare their approach or solution with the reference implementation.
4. **Maintenance:** The final phase doesn't actually belong to the training course. It is a kind of homework that is assigned to the participants. The goal of this trainings phase is to help people get used to the framework and the corresponding libraries. The assignments that are provided during this phase are intended to cover a wide variety of different features and components of the corresponding library.

Another didactic component that is used is the spiral approach. That is, training cases start out with a simple test case. This test case is revisited from time to time making it more and more complex. Due to this approach, participants do not have to deal with complex interfaces and functionalities of the framework and the libraries in the first place. Instead, their full functionality is discovered step by step giving people enough time and opportunity to get used to it.

A summary of the training phases is provided in Table 4.

Phase	Purpose	Goals	Type	Training approach
1	Awareness and acquisition	Obtain all required information in order to start with the practical parts.	Theoretical	Lecture
2	Application	Get used to the Amigo framework and the libraries that are needed for the specific training block.	Practical, on site	Solution is explained step by step in the documentation. Trainer gives direct feedback and help.
3	Utilization	Practice the knowledge and techniques learned in the previous phase by developing more complex solutions.	Practical, on site	Medium complex assignments are given to participants. Only a complete reference solution is provided for a final check / comparison. Trainer gives direct feedback and help.
4	Maintenance	Widen knowledge about Amigo technology	Practical, off site	Complex assignment are given to the participants. These assignment use advanced features of the specific component or technology. These assignments are typically not done during the training but are intended for homework. No trainer support is provided for this phase.

Table 4: Training phases

5.2 Training blocks

As Amigo supports two deployment frameworks (one .NET based and another based on OSGi), the training courses also contain two versions of the assignment and explanations. Examples for .NET and OSGi follow each other. The first approach to mix both parts with each other has been abandoned. Instead, a complete solution of the one framework is followed with the appropriate counterpart of the other. The reason for this is that we discovered that it is difficult to follow the explanations if C# and Java implementations are mixed with each other in the same section. Nevertheless, having examples for both deployment framework clearly shows the interoperability of the approach, which is one of the key features of Amigo. Another advantage of providing two versions is that attendees which are only familiar with either .NET or Java may be interested to see how a specific operation or solution is done within the counterpart framework giving some additional motivation to follow the training course.

The training blocks are summarized in Table 5.

Titel	Goals	Time	Prerequisites	Components	Module
Making devices Amigo compatible	<ul style="list-style-type: none"> Use Amigo's deployment frameworks Understand and design simple ontology based knowledge bases Make a device Amigo aware or compatible. 	Theory: 45 min Practice: 135 min		<ul style="list-style-type: none"> Semantic modelling using ontologies Using Web Services 	B
Creating context sources and context clients	<ul style="list-style-type: none"> Teach how to create Amigo context sources Teach how to create context-aware applications 	Theory: 45 min Practice: 135 min	Training block "Making devices Amigo compatible"	<ul style="list-style-type: none"> CMS-Tutorial 	D
Developing Amigo services	<ul style="list-style-type: none"> Teach how to create Amigo based services Show how to use ANS Show how to use UMPS 	Theory: 45 min Practice: 135 min	Training block "Making devices Amigo compatible"	<ul style="list-style-type: none"> ANS-Tutorial UMPS-Tutorial 	A, C
Using Amigo security services	<ul style="list-style-type: none"> Teach how to secure services by applying the Amigo security services 	Theory: 30 min Practice: 90 min	Training blocks "Making devices Amigo compatible", "Developing Amigo services"	<ul style="list-style-type: none"> Using Amigo security services 	E
Multimedia in Amigo	<ul style="list-style-type: none"> Teach how to use Amigo for creating multimedia application Teach how to use Amigo data storage services Teach how to use Amigo's content distribution 	Theory: 30 min Practice: 90 min	Training blocks "Making devices Amigo compatible", "Developing Amigo services" and "Creating context sources"	<ul style="list-style-type: none"> Multimedia in Amigo 	B, C

Table 5: Training blocks

Each training block consists of one or two components. These components are separate training units that can be for their own. That is, each component as a theoretical part as well as a practical part. The reason for this is to provide a fine grained set of training materials that can be individually composed to match some specific training needs. For example, if a user wants to know about ANS she or he can directly choose the appropriate component instead of extracting the corresponding material from a larger document.

In the following sections the various training blocks are explained in more detail.

5.2.1 Making devices Amigo compatible

Type: Training.

Module: B.

Prerequisites: None

Authors: Edwin Naroska, Todor Dimitrov, *Ioannis Papaioannou*, *Ioanna Roussaki*, *Miltiades Anagnostou*, *Dimitris Tsesmetzis*

Goals: Attendees will learn how to apply the basic technologies that are used within Amigo. After the training session attendees will be able to use Amigo's deployment frameworks. Further, they will be able to understand and design simple ontology based knowledge bases. Finally, they will know how to make a device Amigo-aware or compatible.

Description: This training block will teach attendees about the basic technologies that are used in the Amigo middleware, e.g. web services, .Net, OSGi as well as ontologies. In order to use the Amigo system, people must know these core technologies and learn how to apply them in the Amigo context. In detail the training block will introduce the .NET and OSGi based deployment frameworks among some general introduction into C# and Java based programming. Both frameworks base on Web Services as a platform independent way to do remote calls from one computer to another. Web Services exploit web technology in order to ensure interoperability between different machine types. As a result, it does not matter which kind of machine is used at server or client side. Further, appropriate support to offer and consume Web Services are available for many programming languages, making this approach nearly language neutral.

Another interesting feature that can be used within Amigo is eventing based on Web Services. For eventing, a server informs a client about a specific event by calling a remote event handler on the client. Note that this is an asynchronous mechanism, i.e., the client does not know when an event may occurs. In order to be notified by server, the client must register to receive events.

During the practical parts attendees will develop simple Web Service based applications. This includes designing a web service as well as the corresponding client part. Moreover, attendees learn how to web service eventing within their applications. As a result, they learn what is needed to build a basic Amigo compatible device or service. This includes designing and implementing all required descriptions and services based on the deployment frameworks.

Additionally, this training block aims to teach how to create and use ontologies. After finishing this part, the user will be able to use the most popular web ontology editor, Protégé-OWL, in modelling simple and more complex ontologies, import external ones and use multiple complex semantic structures. Protégé is a tool for rapid ontology prototyping and application development, based on Java. This tool exports the ontology models in .owl and .rdf files, rendering their usage appropriate for programming languages and other semantic-related tools. More specifically, users will be able to create all the various classes, properties and rules of the ontology they intend to model for a specific concept. Note that major parts of the Amigo middleware use ontologies for semantic modelling of information.

This training block is a prerequisite for all other blocks. It consist of two components: "Ontology Creation for AMI services" and "Using Web Services",

Versions: The theoretical part is 45 minutes, the practical part runs approximately for 135 minutes.

5.2.2 Creating context sources and context clients

Type: Training.

Module: D.

Prerequisites: “Making devices Amigo compatible”

Authors: Jörg Schmalenströer

Goals: After the training course attendees will be able to create Amigo compatible context sources and context clients.

Description: Context awareness is a major building block of Amigo. Hence, a separate training block has been chosen to cover this topic. The training block will teach attendees what is needed to implement a context source as well as a context client using the aforementioned source. They will learn which methods have to be implemented in order to use the Amigo context management system. To this end, they will practically design a context source and a context client using the Amigo deployment framework and services. Further, they will be learn about the CMS Broker which is the main component for the context management service, as it provides all required functionality for discovering and using context sources. This block is based on the block “Making devices Amigo compatible”.

The training block consist of one component: “CMS-Tutorial”,

Versions: The theoretical part is 45 minutes long, the practical part runs for 135 minutes.

5.2.3 Developing Amigo services

Type: Training.

Module: A and C.

Prerequisites: “Making devices Amigo compatible” and “Creating context sources and context clients”

Authors: Edwin Naroska, Todor Dimitrov, Otilia Kocsis

Goals: After the training course attendees will be able to create Amigo based services. To this end, they learn how to apply ANS as well as UMPS in order to make their applications dependent to user and context information.

Description: The training block will teach attendees what is needed to setup a Amigo service. Similar to context sources, services are at the heart of a Amigo system. Hence, a separate training block has been devoted towards this topic. One element of the block is about using the Awareness and Notification System (ANS) of Amigo. ANS is part of the intelligent user services and provides an easy way to get access to changes in the context of the home environment. ANS provides functionality to allow people/applications to stay aware of context changes. For example, it can inform applications about changes of the current location of a specific user. As a result, the application does not have to query all the various context sources that may provide appropriate location information. Instead, it subscribes to ANS and inject an appropriate rule that is automatically checked by ANS each time context information changes During this block. ANS and how to write ANS clients are explained.

As another important element of Amigo this training block addresses the User modelling and profiling service (UMPS). UMPS provides the methodology to enhance the effectiveness and usability of services and interfaces in order to (a) tailor information presentation to user and context, (b) reason about user's future behaviour, (c) help the user to find relevant information, (d) adapt interface features to the user and the context in which it is used, (e) indicate interface features and information presentation features for their adaptation to a multi-user environment. The training block teaches how to use the tools that come along with UMPS and how to access user information via the UMPS API.

This training blocks includes two components: the "ANS-Tutorial" and "UMPS-Tutorial".

Versions: The theoretical runs for 45 minutes, the practical part runs for 135 minutes.

5.2.4 Using Amigo security services

Type: Training.

Module: E.

Prerequisites: "Making devices Amigo compatible", "Developing Amigo services" and "Creating context sources"

Authors: Edwin Naroska, Todor Dimitrov

Goals: After the training course attendees will be able to secure services by applying the Amigo security mechanisms.

Description: The Amigo security framework is responsible of securing access to the services in the home. The Amigo security model is based on the Kerberos protocol, with shared secrets to establish mutual authentication. The security service is the mutually trusted Kerberos authority and acts as a trust broker between users and services inside the home and grants or denies access to services. The employed access control scheme in Amigo is role based with few, well-known user, device, and service roles, thus enabling non-technical users to setup a secure system easily. As usual with Kerberos, access is granted or denied centrally by the security service and is relayed to users and services with encrypted tickets.

The training block will teach attendees how to use the Amigo Security Services in order to setup secure communication channels between client and services. In detail the tutorial will show how to write a secured service, how to write a client that makes use of a secured service. Further, it will explain how to authenticate a user/device within the Amigo network. The training block is based on the training blocks "Making devices Amigo compatible" and "Developing Amigo services".

This training block contains of the component: "Using Amigo security services".

Versions: The theoretical part runs 30 minutes, the practical part runs for 90 minutes.

5.2.5 Multimedia in Amigo

Type: Training.

Module: B and C.

Prerequisites: “Making devices Amigo compatible”, Developing Amigo services” and “Creating context sources”

Authors: José María Miranda, Alvaro Ramos, Ricard Jiménez,, David Roldan, David Cordón

Goals: After the training course attendees know how to use Amigo for creating multimedia application, how to use Amigo data storage services and how content distribution is handled.

Description: This training block is a tutorial for building multimedia applications using the Amigo framework. The tutorial is designed to be used in combination with java, but web services can also be implemented in any other language. It shows examples of how to implement a content distribution client to interact with the multimedia server and shows the capabilities and the power of the functionalities provided by content Distribution regarding multimedia content. During training the user will be guided through the process of writing his or her own code to interact with the content distribution and be able to develop a multimedia application..

This training blocks includes the component: “Multimedia Training”.

Versions: The theoretical runs for 45 minutes, the practical runs for 135 minutes.

5.3 Training documents

For each theoretical part of a training block there is an appropriate document to be delivered as handout which introduces the basic concepts and explain the prerequisites to run the training. Further, the document contains the assignment as well as a step-by-step description of the solution(s). Note that if available the description of the .Net and OSGi based solutions will be shown side by side.

Further, a source code package is associated with each training block. This package contains supporting source code for the assignments as well as the final solutions. In the supporting source code some parts of the code are preset in order to save some tedious typing. For example, the source files already include all the references to the external package that must be included. As a result, attendees can focus on the actual code of the assignment instead of struggling with many implementation side effects.

In order to ensure that the documents are having the same structure as well as design, the following procedure has been applied for training block development.

5.4 Development process and feedback

The development process of each training block includes several internal feedback steps where content as well as the performance of the presenters are checked and improved. Further, a feedback step where the training blocks are rated from the attendees are included as well. In detail, the following steps are done:

1. Determining block content. In order to develop a comprehensive set of training blocks that are well aligned, the content of each block has been determined in collaboration with the partners from WP10. This ensures that all important topics are covered and there is no unintended overlap in the contents between the blocks. Further, the experience with internal training sessions that were given from Amigo partners to Amigo partners are

incorporated. While these sessions are not representative for designing training courses that target at SMEs, valuable information can be still derived from it.

2. Content and layout guidelines. In order to provide a consistent look and feel among the training blocks, templates for the slides of the theoretical parts as well as for the actual supporting document have been created. These templates define layout, font style, size and color (in case of the slides). Further, for the coding document it also defines the overall structure of the document. It can be downloaded from the Amigo Quickplace project web site. For the slides of the theoretical part, the rules from section 4.4.1 apply. For the coding part, the rules are shown in section 5.4.1.
3. Early style and content alignment. In order to fine tune alignment between the training blocks, preliminary versions of the theoretical slides as well as the actual coding documents were reviewed at the meetings and suggestions to improve the presentation were generated. Further, the partners checked the content against their own training blocks and their assigned seminar blocks in order to find overlapping parts, inconsistencies or gaps. This process ensured that the content as well as the style is consistent among the different blocks.
4. User feedback. At the end of each training block users are requested to fill out a short feedback form in order to provide hints about what can or should be improved. The feedback forms are collected, analyzed and the results are sent to the partners in order to help improve the training blocks. If required, the feedback data is discussed at the regular partner meetings.

Note that some of the steps listed above were done only once (e.g., steps 1 and 2) and others were repeated for each training block that has been developed or course that is given.

5.4.1 Training document guidelines

In the following a set of guidelines were used during coding block document development to align the style of the documents accordingly:

Document style:

- Build a document title using the following template: "Amigo Training Document: <Topic of the block>".
- Use the "coding template file" available from Quickplace as a template and stick to the font style and font size suggestions.

Document content:

- Use the structure of the document template document and fill in the content to each section. The coding document contains the following information
 1. A short description on what topic is addressed by the training. For example: "The training block teaches how to use UPMS. After finishing the course, users will be able to add new items to the UPMS service and store/retrieve user profile information to/from the UPMS service."
 2. A list of common prerequisites for this training block. Here, general (non Amigo specific) requirements are listed, e.g. "Requires basic programming skills".
 3. A list of Amigo related prerequisites that are needed in order to process the training block. Along with the requirement, a recommendation is given how to gain the corresponding knowledge. Typically this will be a link to another seminar or training

block. Example: “The training block requires that the user is familiar with ANS. Information about ANS is available from document X. Further, a training block on how to use ANS can be downloaded from Y”.

4. A list of recommended training blocks to continue with. This shall provide the user with information on how to continue in a most efficient way. By following the recommendation the user will eventually have a complete overview over the Amigo middleware.
 5. A list of software components that are needed to do the training block. Along with the list, a link where to obtain the appropriate components is given.
 6. A link to the source code package that is associated to the training blocks (e.g., predefined blocks of code that shall be modified by the user).
 7. An introduction to the training block explaining the what is going to be done. Further, the introduction explains the features of the Amigo components that are introduced in this training block. Please note that not a complete description of a Amigo component is provided here. Instead, only the basic information is provided that is needed to proceed with the training block. Additional information is given when they are needed to fulfil the assignment.
 8. The actual training block that contains the assignments along with a step-by-step explanation of the solution for the basic assignments. Note that a detailed step-by-step solution is only provided for the basic assignments. For the extended ones only the final solution is explained and also provided as source code.
- The actual practical part of each training block shall be separated into several steps. Each step should be started by providing additional information about the Amigo component that is going to be used next. Instead of providing a complete description of the component, the knowledge about it is extended by revisiting it, i.e. iteratively extending the feature set that is used (spiral approach).
 - Do not dump long sections of source code to the document. Instead show only the relevant sections and point to the source code package that is associated with each training block for further reference.
 - The tutorial document shall contain sufficient material so that users can run it by their own. As a result, the documents can be used as home leaning material that is of use beyond of the actual runtime of the project.
 - Make sure that the files in the source code package are commented sufficiently.

5.4.2 Directory structure

A typical training component consist of a documentation an a set of source code. The documentation and source code are packed into a single distribution file which can be downloaded and installed. The typical directory structure is shown in *Figure 2*. Within the installation directory there are three elements. The documentation in form of a PDF file and two sub-directories “Code” and “Code_Solution”. The first one contain the template files for the training. During the training, these files are edited. The second sub-directory stores the actual reference solutions.

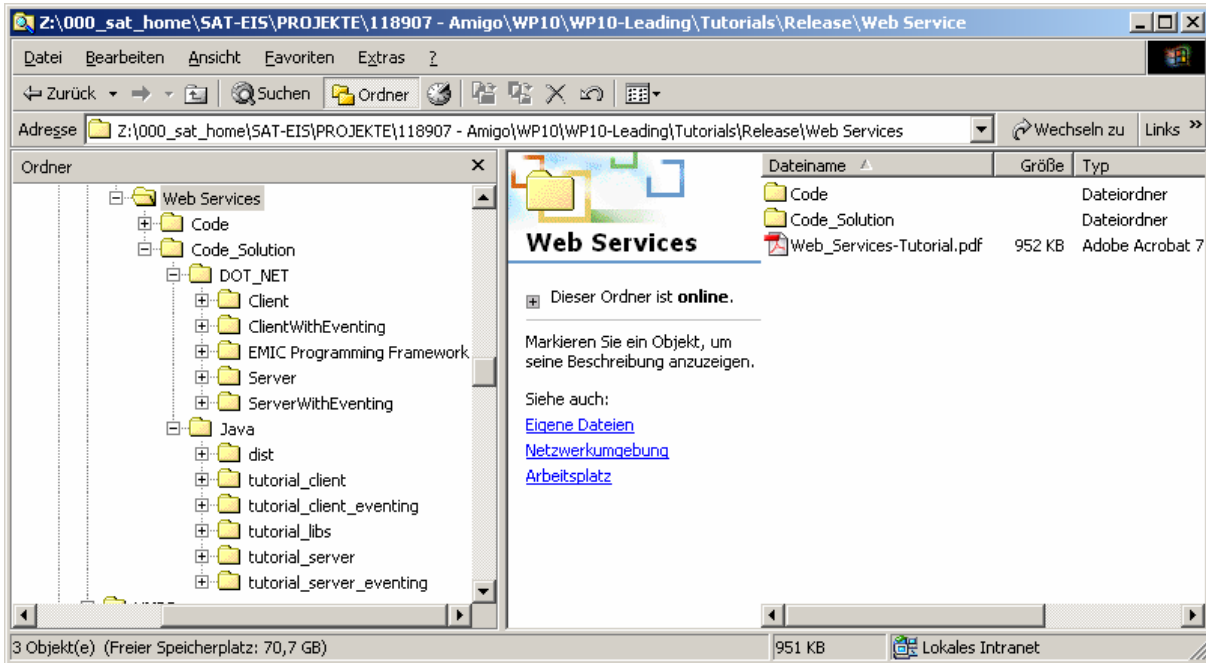


Figure 2: Directory structure of a typical training component

5.4.3 Training Feedback from template

The feedback from template for training sessions are given in **Error! Reference source not found.**

6 Sample tutorial walkthrough

In the following sections two examples are presented who the training courses can be used to efficiently gather information needed to implement a Amigo service and to integrate a device into an Amigo network.

6.1 Developing a service

In this section we will exemplarily show how to use the training documents in order to obtain the knowledge to work with the Amigo middleware. To this end, we assume that there is an user who wants to develop an Amigo service. The Amigo service shall have the following properties:

- The service shall be dependent on the preferences of a specific user and
- it shall also dependent on the location of the user.
- Further, it shall deal with user related information that shall be kept as a secret.

In order to write a corresponding service, the developer must know the Amigo basics, how to access user preferences and how to get information about the users current location. Further, the developer must learn how to protect sensitive communication between service and client.

6.1.1 1st Lesson: How Amigo components communicate

The first step in our walkthrough is to learn about the basics of Amigo. That is, the developer must learn how Amigo components talks to each other. To this end, the training block “Making devices Amigo compatible” is used. During this block, developers learn who to use Amigo based Web Services for communication.

In detail, our sample developer will learn how to write a server that is amigo compliant and how to talks to Amigo components. Moreover, the training explains how to look for services that match a specific property and what is needed so that a service can be discovered by other components on the network.

At the end of this training block, the developer has implemented two simple Amigo compatible server as well as corresponding clients. He or she further designed a service that fires events in order to notify clients about a specific condition.

In order to write the service, the developers still needs the information about how to retrieve user preference data and the location of a user. This is addressed in the next two lessons.

6.1.2 2nd Lesson: Retrieving user location

The best method to retrieve the current user’s location in an Amigo network is to search for an appropriate context source. Hence, the next lesson named “Creating context sources and context clients” addresses how to use context sources. He or she learns how to look for a specific context source using the Amigo context broker. Further, the developer learns how context sources behave and what kind of information they provide. In detail, developers learn how to parse the information packets that are generated by a context source component.

At the end of the lesson, the developer succeeded in designing a context client that is capable of finding an appropriate context source that provides the needed localization information. Further, the developer wrote code to interpret the context data in order to extract the location information. As a result, know all knowledge is present to extract the location information from an Amigo network.

6.1.3 3rd Lesson: Retrieving user preference information

As the service that shall be developed will be dependent on user preference information, the next step is to get used to UMPS which is discussed in training block “Developing Amigo Services”. In order to retrieve the appropriate user preference information, the UMPS database must of course be aware of the type of information that shall be retrieved. In fact, the predefined UMPS database comes along with a predefined set of user preferences that are directly read to use. Nevertheless, in order to add new preferences, the training block also provides appropriate training material.

Last but not least, during the training the developer will practice to use the UMPS API in order to retrieve the preference values for a specific user or to write new values into the preference database. As a result, at the end of the training session, the developer will be able to extract the needed preferences information from the UMPS service via the Amigo UMPS API.

6.1.4 4th Lesson: Securing the service

At this point, the developer collected all information to develop the Amigo compatible service. However, what still need to be addressed is security. To this end, the training block “Using Amigo security services” provides the information on how to encrypt communication between service and client so that nobody other than the intended communication partners can read the information as clear text.

During this lesson, the developer learns how to authenticate Amigo devices. This is needed in order to connect them together using secure communication channels. Further, one will learn how to modify the un-secure services and clients to make them secure. In fact, not too much changes must be done to reach this goal. The major efforts are handled by the underlying middleware.

At the end of the last lesson, the developer has now all information that is needed to develop the example service.

6.2 Integrating a device into an Amigo network

In the next example. The developer has the intention to integrate an existing device into an Amigo network. We assume that the developer uses a proxy approach, that is, the device is not directly integrated into the network but represented by a proxy instance. As a result, the proxy is visible in the network, receives requests and forwards them to the actual device. Results are transmitted in reverse direction via the proxy as well. We further assume that the device shall operate as an context source providing some kind of information. The actual type of information is not of interest for this document.

Hence, the actual goal of the developer is to build the proxy instance. Note that we do not address communication between proxy and device as this is device dependent,

6.2.1 1st Lesson: How Amigo components communicate

Same as for the other example, the first step in our walkthrough is to learn about the basics of Amigo. That is, the developer must learn how Amigo components talks to each other. To this end, the training block “Making devices Amigo compatible” is used. During this block, developers learn who to use Amigo based Web Services for communication.

In detail, our sample developer will learn how to write a server that is amigo compliant and how to talks to Amigo components. Moreover, the training explains how to look for services that match a specific property and what is needed so that a service can be discovered by other components on the network.

In addition, the developer will learn how semantic information is encoded in Amigo so that it can be automatically processed. To this end, he or she learns how to develop simple ontology knowledge bases. This will enable the developer to extend the knowledge base so that the information of ones context source can be encoded appropriately.

At the end of this training block, the developer has implemented two simple Amigo compatible server as well as corresponding clients. He or she further designed a service that fires events in order to notify clients about a specific condition. Finally, the developer also extended a knowledge base in order to include some new concepts.

6.2.2 2nd Lesson: Introduction to context sources

The best method to retrieve the current user's location in an Amigo network is to search for an appropriate context source. Hence, the next lesson named "Creating context sources and context clients" addresses how to use context sources. He or she learns how to look for a specific context source using the Amigo context broker. Further, the developer learns how context sources behave and what kind of information they provide. In detail, developers learn how to generate the information packets that are send by a context source.

At the end of the lesson, the developer succeeded in designing a simple context source. This will be a base for the proxy code.

7 Feedback results

In the following some preliminary feedback results gathered from the Amigo seminars are presented.

Figure 3 shows whether the people thought about using middleware technology before attending the course. Please note that about 67% intended to use middleware.

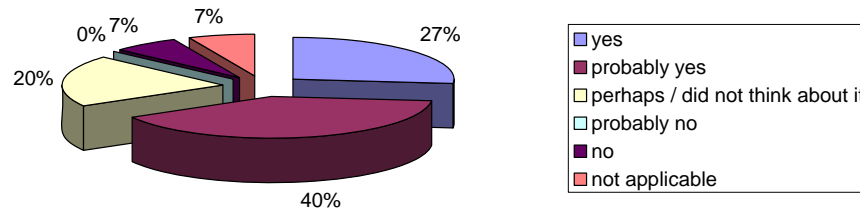


Figure 3: "Did you plan to use middleware technology before attending the seminar?"

Further, we also asked whether those people still intended to use middleware now that they learned something about its features and properties. It can be seen that the number of people that answered "yes" or "probably yes" dropped to 60% (compared to 67%). The reason for this may be that there were some knowledge missing about the purpose of a middleware. This knowledge gap has probably been closed during the course. As a result, some people perhaps discovered that middleware does not solve the problems they are currently facing.

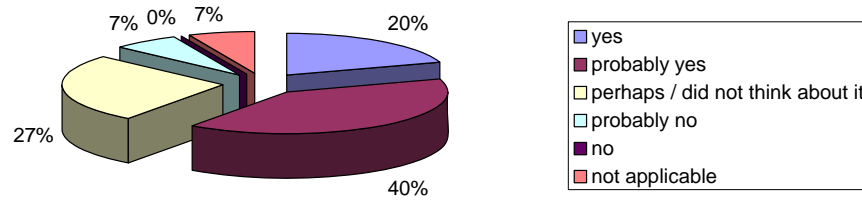


Figure 4: "Will you now use middleware technology?"

Figure 5 shows the answers people gave when asked about their intention to use Amigo software. Please note that already 35% answered "yes" or "probably yes". Obviously, a significant amount of people are successfully addressed by the Amigo solutions.

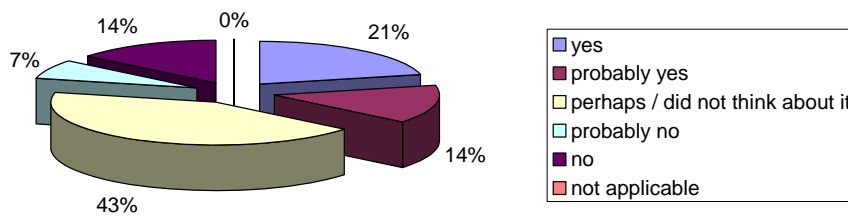


Figure 5: "Do you plan to use Amigo software in the future?"

Further, the interest in Amigo is quite high as can be seen from Figure 6. Note that 64% of the people clearly stated "yes" and nobody answered "no" or "probably no". Obviously, the course were able to keep the audience interested in Amigo even if no direct application of the software is intended.

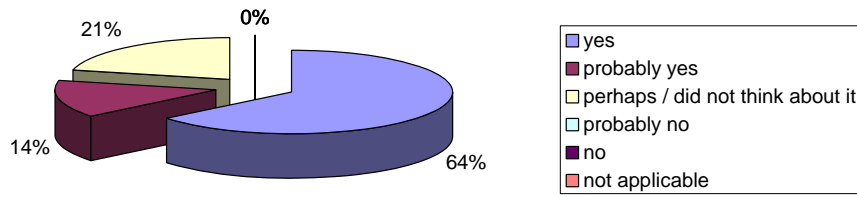


Figure 6: “Are you interested in further information about Amigo?”

People were also asked whether the course addresses their current needs. Note that 47% stated “good” or better. Hence, about 50% of the attendees could find their current problems being addressed by the course. Nevertheless, there is still room for improvement here.

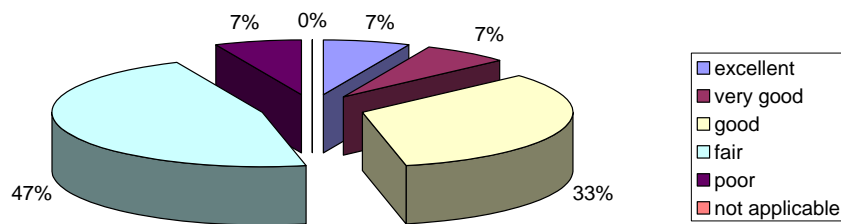


Figure 7: “Did the Seminar address your current needs?”

The attendees were also asked to rate the content of the seminar. Obviously, the content was reasonably chosen as about 72% of the people rated “good” or better (see Figure 8).

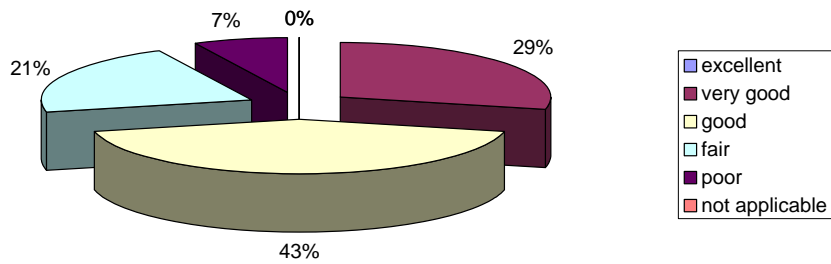


Figure 8: Evaluation of the course content

Further, people were also asked about the quality of the content and the handouts. As can be seen from Figure 9 and Figure 10 they were quite satisfied. Especially, people liked the course handouts.

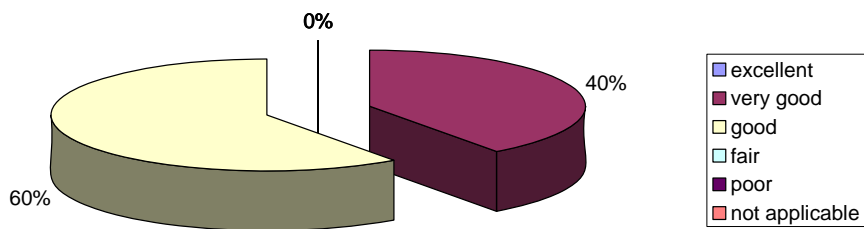


Figure 9: Evaluation of the course slides

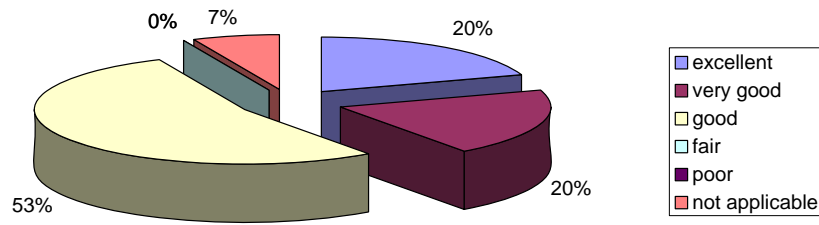


Figure 10: Evaluation of the course handouts

To summarize the evaluation results, it is clearly shown that the interest in middleware technology in general and Amigo especially is quite high. Further, people liked the presentation and the handout. As a result, the course material seems to be suitable as download material on the Amigo web site.

Appendix

In the following some additional material relevant to the training blocks are shown.

Appendix A: Training document template

The training documents are created according to the following template pages. Basically a training document consists of four main sections. In the first section, a introduction to the appropriate topic is given. This includes providing knowledge that is required to run the training part. Note that the documents were created so that people can use them in order to run the training blocks for their own. That is, the training parts do not necessarily require to be controlled or guided by lecturers. To this end, in the first part also explains what knowledge is needed by the user in order to successfully run the training block.

In the following part the system and software requirements are explained. This includes the hardware that is needed as well as the software that must be installed on the system. Further, the document lists where to obtain the software modules.

The third part contains the actual training part. Here, the assignments are described and the corresponding solutions are explained step by step. Of course the source code here corresponds with the actual source code package that is provided with the training block.

Finally, in the last part recommendations on how to proceed are provided. As a result, the complete training block set forms a chain of recommended steps to get into the Amigo middleware.