35182

OI

Open Interface: an Open Source platform for developing multimodal interaction

SPECIFIC TARGETED RESEARCH OR INNOVATION PROJECT

Priority 2 - IST

D0.3: Final Activity Report

Due date of deliverable: August 31st 2009
Actual submission date: August 31st 2009

Period covered from September 1st 2006 to May 31st 2009

Date of preparation: June 4th 2009

Start date of project: September 1st 2006 Duration: 33 months

Project coordinator name: Laurence Nigay
Project coordinator organisation name: Université Joseph Fourier (UJF)

Revision: Final

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)		
Dissemination Level		
PU	Public	
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	X
CO	Confidential, only for members of the consortium (including the Commission Services)	

Deliverable Identification sheet

Project ref. no.	FP6 - 035182
Project acronym	OI
Contractual date of delivery	M36
Actual date of delivery	M36
Deliverable number	D0.3
Deliverable title	Final Activity Report
Nature / TYPE	Report
Status & version	Final
Number of pages	36
WP contributing to the deliverable	WP 0
WP / Task responsible	UJF
Editor	Laurence Nigay (UJF)
Author(s) (alphabetically)	All partners
EC Project Officer	Pierre-Paul Sondag
Keywords	OI, OpenInterface, Final Activity Report
ABSTRACT (FOR DISSEMINATION)	Not applicable

Table of contents

1 Executive Summary

Multimodal interaction is challenging to develop especially in mobile/ubiquitous settings with novel interaction devices and modalities. Addressing this challenge, the FP6-35182 OpenInterface project focuses, as its main objective, on the design, implementation and testing of an open source framework for the rapid development of multimodal interactive systems as a central tool for an iterative User-Centered Design (UCD) process.

The **open source OpenInterface framework** is downloadable and fully documented at www.oi-project.org. The OI framework is based on a theoretical understanding and characterisation of multimodality and is comprised of three key elements:

- **a repository of software components** that provide pure (mono)modalities (such as gesture and speech), fusion / fission mechanisms and multi-modalities (assemblies of pure modalities and fusion/fission components to create multimodal interaction techniques),
- **a development environment** (called the OpenInterface Development Environment, or OIDE for short) that includes development tools for assembling, combining, configuring, debugging and testing OI components and assemblies at multiple levels of abstraction, from the raw data to the semantics of users' events, and
- **a runtime kernel** providing management and communication among heterogeneous and distributed software components. It allows the integration of existing interaction modalities written in different languages.

Our work on the OI framework has been contextualised, driven and evaluated by testbed applications being co-developed with the tools. One testbed, exploiting ubiquitous computing technologies, provides multimodal interaction for Large Information Spaces or LISs (e.g. digital maps); the other testbed, mainly focusing on mobile technologies (phones & PDAs), is dedicated to multimodal and mobile games. As part of an iterative User-Centered Design (UCD) process, four versions of each testbed have been designed, developed and tested in the laboratory and in the field.

Based on the results coming out of the evaluations of these two testbeds, both developed with the framework, we identified extensions to multimodal standards (e.g., for data description and information exchange) and developed an interpreter for these extensions. In order to test the interpreter, two validators running on mobile phones for the same application domains as the testbeds have been developed and in-field evaluated.

Overall, the OpenInterface project provides an important set of theoretical results and a complete toolset for the rapid development of multimodal interactive systems using state of the art technologies and coupled with an explicit user-centred design development methodology. It is our ultimate aim by so doing to help close the gap between research, offering new modalities, new development methods and new tools, and industry with its demanding application requirements and a need for standards-based development.

2 Objectives

The goal of multimodal interaction is to make the computer and more generally speaking digital information easier to access and manipulate in various usage contexts. In OpenInterface, we studied multimodal interaction on mobile phones as well as multimodal interaction on an augmented table, at home for example navigating in a map looking for a restaurant. Clearly multimodal interaction has a strong impact on quality of life as well as working conditions (for example mobile workers) by making digital contents more accessible and manipulable. Finally multimodal interaction has a key role to play for handicapped users (see for example one of our publication - Investigating Touchscreen Accessibility for People with Visual Impairments – NordiCHI'08).

The multimodal interaction domain has expanded rapidly and significant achievements have been made in terms of interaction modalities. The advent of new modalities based on a variety of captors and effectors coupled with recognition/synthesis mechanisms, as well as the availability of affordable and commonly used devices, such as webcams and game devices (Nintendo Wii Remote) are rapidly enriching the interaction capabilities of desktop computers. As pointed out by Dan Olsen, "systems based on one screen, one keyboard and one mouse are the new equivalent of command-line interfaces". Nevertheless, available tools for development are still designed for a set of fixed standard input devices and interaction paradigms: Commonly used toolkits only consider graphical interaction with a mouse and a keyboard. Consequently, Multimodal interaction is hard to prototype and develop, especially in mobile/ubiquitous settings with novel interaction devices. Prototypes in this domain are difficult to build, to change and to monitor in order to analyse, interpret and evaluate interaction data, as part of an iterative user-centred design method.

Facing the vast world of possibilities for interaction modalities, tools for integrating and combining those modalities become a real challenge addressed in the OpenInterface project. Focusing on engineering multimodal interaction, the main objective of the OI project is to design, implement and test an open source framework for developing multimodal interaction:

- that handles a rich and extensible set of modalities,
- that enables quick replication,
- that enables a focus on innovation (new modalities or forms of multimodality),
- that supports dynamic selection and combination of modalities to fit the ongoing context of use,
- that enables iterative user-centred design.

To achieve this main objective and as described in the DoW (Description of Work or Annex I), we identified three objectives at the beginning of the project. We list those objectives and describe how they are achieved.

(1) First, in order to populate the platform with innovative pure or combined modalities, our objective is to gain understanding of multimodal interaction.

A conceptual model of multimodality has been established and published in the key conference on multimodal interaction (ACM-ICMI 2008). We also collected empirical results from the in-laboratory and in-field evaluation of our two testbeds (four versions of each testbed have been tested) as well as of our two validators running on mobile phones.

(2) Based on the theoretical results on multimodality, the second objective is to design,

development and test a platform for rapidly developing multimodal interaction. For testing the platform and understanding the impact of the platform on the user-centred design process, our objective is to design and develop two testbeds for different application domains. Two different application domains are considered for proving the unavoidable generic aspect of the platform. One testbed is dedicated to multimodal interaction with a large information space while the second one will enable us to study multimodal interaction in the context of a game. The two testbeds will consider multimodal interaction on a mobile device as well as with an augmented pervasive environment, e.g. home environment.

The open source OI framework is available and fully documented at www.oi-project.org. The OI framework is made of an underlying platform, namely OpenInterface platform, and the OpenInterface Interaction Development Environment (OIDE). A repository of software components is also provided. It provides pure (mono)modalities (such as gesture and speech), fusion / fission mechanisms and multi-modalities (assemblies of pure modalities and fusion/fission components to create multimodal interaction techniques).

Two testbeds, namely Large Information Space and Game, have been designed, developed using the OI framework and tested. As part of an iterative User-Centered Design (UCD) process, four versions of each testbed have been designed, developed as well as in-laboratory/in-field tested.

(3) Based on the iterative design of the two testbeds developed with the platform, our third objective is to identify standard extensions and to develop an interpreter for those standard extensions. For testing the interpreter and the coupling with the platform, two validators for the same application domains as the testbeds will be developed and evaluated.

The Multimodal Middleware Protocol was specified as a new standard. Going further than prototyping multimodal interaction on mobile phones using our OI framework (running on PC), the multimodal hub and browser implement our Multimodal Middleware Protocol and are running on mobile phones. There are J2ME and Android implementations as well as an Open Source Python implementation.

Two validators running on mobile phones, namely Large Information Space and Game, have been designed, developed using the multimodal hub and in-field tested.

3 Consortium

Participant name	Country	Role
Université Joseph Fourier Grenoble 1	France	Coordinator
Université Catholique de Louvain	Belgium	Contractor
University of Glasgow	UK	Contractor
Fraunhofer Institute for Applied Information technology	Germany	Contractor
Multitel ASBL	Belgium	Contractor
ArcadiaDesign	Italy	Contractor
PhonoClick	Turkey	Contractor
TXT e-Solutions Spa	Italy	Contractor
France Télécom	France	Contractor
Immersion SAS	France	Contractor

Immersion SAS entered the project on February 1st, 2008 (M18), replacing SOUTH WING partner who officially withdrew from the project on June, 18th 2007 (M10).

All the other partners were part of the consortium from the beginning of the project, on September 1st, 2006.

3.1 Université Joseph Fourier Grenoble 1

UJF is represented by the IIHM research group (Ingénierie de l'Interaction Homme-Machine). IIHM belongs to the LIG laboratory (Laboratoire d'Informatique de Grenoble). Laboratory LIG is a joint research unit (UMR 5517) affiliated with UJF, Centre National de la Recherche Scientifique (CNRS), Institut National Polytechnique de Grenoble (INPG), Institut National de Recherche en Informatique et Automatique (INRIA) and UPMF (Université Pierre Mendès France). IIHM includes eight senior professor-researchers, and 15 Post-Docs, PhDs, and Master's students. IIHM is primarily concerned with the software engineering aspects of Human Computer Interaction. This group has extensive experience in multimodal interaction, augmented environment, mixed reality, and software architecture modelling for interactive systems.

3.2 Université Catholique de Louvain

UCL is represented by the "visual communications" research group of the "Laboratoire de Télécommunications et Télédétection (TELE). The group, which comprises more than 40 people, concerns research into still image, video and 3D compression for streaming and communication applications, scene analysis and synthesis for immersive visual applications, processing and co-registration of 3-D medical imaging, content indexing and watermarking technologies, and the development of security architectures for multimedia communications. TELE develops technologies for augmented and pervasive visual communication and processing applications. It has been involved in scene analysis, image and video compression (TELE is managing the JPEG2000 open source initiative www.openjpeg.org), and 3D imaging, co-registration, mixed reality technologies and multimodal interfaces.

3.3 University of Glasgow

The Computing Science Department at UGLA was very highly rated (5) in the most the UK's recent Research Assessment Exercise (RAE). UGLA is represented by the GIST group (Glasgow

Interactive Systems Group) with the participation of the following GIST subgroups: the Multimodal Interaction Group and the User Interface Software Technology Group. GIST is one of the oldest HCI research groups in the UK, having begun in the early 1980s. Today it consists of 6 permanent academic staff plus around 20 research fellows, research assistants and PhD students. It is involved in a variety of externally funded research projects, including the EPSRC funded Equator Project (www.equator.ac.uk) in ubiquitous computing and MATCH, a Scottish collaborative project investigating the integration of homecare technologies involving the universities of Glasgow, Edinburgh, Stirling and Dundee.

3.4 Fraunhofer Institute for Applied Information technology

Fraunhofer Institute for applied information technology has more than 20 years experience in designing information technology, pioneering in many fields such as computer-supported cooperative work (CSCW), context awareness and human-computer interaction (HCI). Other core competences cover adaptive and adaptable systems, knowledge management, expertise sharing, and usability. In many projects FIT has already extended this expertise to the use of mobile technologies. One recurring success factor of FIT's work is the approach of accompanying technological innovations with the consistent development of organizational structures and processes.

3.5 Multitel ASBL

Multitel ASBL is a private, non-profit research centre, leading applied research and development activities for industry leaders, SMEs and spin-off companies. It was initiated by the "Faculté Polytechnique de Mons" and "Université Catholique de Louvain" under initiatives supported by the EC and the Walloon Region. Multitel is active in five interrelated R&D domains: Speech Technologies, Image processing and coding, Photonics, Local Networks, Signal Processing and Data Fusion. Multitel is a transfer technology medium, and many spin-off companies have already emerged from the centre: BABEL Technologies (now ACAPELA Group), IT-OPTICS, SMARTWEAR, ACIC, POLYMEDIS.

3.6 ArcadiaDesign

ArcadiaDesign was founded in 2001 following one of the first Italian multimedia experiences started since 1996. Their mission is to research, develop and utilise technological innovations in software applications. Our activity spans two main areas: entertainment software and games both for PC/console and for wireless platforms, and wireless application solutions. In 2003, Arcadia Design was awarded the prize for "best company start-up" at the Mobile Forum in Milan, Italy. ARC maintains close collaboration with the Electronic Engineering Dept. at the University of Cagliari, and is a business partner of COSMIC Group, one of the leading ICT Italian companies, for whom it plays the important role of preferred software developer and technology research laboratory.

3.7 PhonoClick

PhonoClick is a private software company in Istanbul, Turkey, specializing in research and development for telephony based voice interaction platforms, media platforms and solutions. The company was founded in 2002, by three managing partners who have a proven track record of successful technology companies. The company built up its own intellectual property base and has successfully made the transition to many commercial deployments. Using widely adopted industry standards including VoiceXML and VoIP (Voice over IP), the company builds service platforms and solutions for mobile and fixed-line telecom operators, financial institutions, call centers, government agencies and other businesses to handle multimodal business and consumer solutions.

3.8 TXT e-Solutions Spa

TXT e-Solutions is a private, mid-sized software vendor and system integration company, with its headquarters in Milan and growing branch offices in Genoa, Turin, Bari, Rome, Naples, Vicenza, Paris (F), Barcelona (E), Chemnitz, Halle (D), London (UK) and Cambridge (US). Thanks to an highly qualified and motivated staff of about 500 people, 90% of which are graduated, the 2005 turnover exceeds 54 million Euro (47 million in 2004). TXT is a technology intensive Company, offering to its customers complete solutions, based upon kernel products, high-level competence on enabling, innovative IT and specific know-how in a number of application domains.

3.9 France Télécom

France Télécom is the first leading R&D centre in Europe in the telecom sector. France Télécom R&D aims to deliver the best innovations to the intra-Group clients: 70% of the products and services marketed by the Group are born in FTR&D labs. France Télécom R&D develops its expertise and technology in the following fields:

- New generation networking: definition of concepts and architectures enabling networks to evolve progressively toward high bit rate, multi-service, easily operated solutions;
- ·Multimedia services and technology;
- Mobile networks and services including GRPS, UMTS and Beyond 3G ;
- Innovative End user interfaces: Multimodal interfaces, natural languages processing, advanced interfaces;
- Software engineering systems: real-time kernels, distributed systems, embedded systems.

3.10 Immersion SAS

Immersion SAS is a French Virtual Reality solutions provider (SME), since its inception in 1994 and the early ages of VR. Based on many years of experience and a newly created R&D department, Immersion distributes well-know VR input devices and visualisation solutions, studies and answers to specific configuration requirements and develops innovative products. Immersion long time clients include EADS, EUROCOPTER, FRANCE TELECOM, CEA, CS-SI, Dassault Aviation, DCN, Essilor, FIAT, ORANGE R&D, RENAULT, PSA Peugeot Citroen, CSSI, THALES, Labri, Irit, CNRS, … Thanks to more than 10 years of collaboration with all the major VR producers, Immersion has access to all the available VR hardware technologies. Those include visualisation solutions, from the various kinds of Head Mounted Displays to video projectors, and the many existing input devices solutions. Moreover, increasing workforce brings Immersion the necessary capacities for software support. Indeed, while Immersion first activity is the reselling of VR hardware, its constant growth pushed the company to expand around 3 departments: sales, projects, and research & development. The R&D activity is the one that would be the most involved in the OpenInterface project.

4 Project logo and website

Project logo

Project coordinator

Pr. Laurence Nigay (Laurence.Nigay@imag.fr)
UJF- Université Joseph Fourier, Grenoble 1,
LIG – Laboratoire d'Informatique de Grenoble
385 rue de la Bibliothèque, BP 53
38041 Grenoble, FRANCE
Tel. +33 4 76 51 44 40
Fax. +33 4 76 63 56 86

Project website

The OI web site is hosted at the following URL:

http://www.oi-project.org

The web site includes a download page for downloading the OI framework and Multimodal Hub. It also includes a page presenting the multimodal demonstrations made by the project (Showcase page). For each demonstration, a description of the multimodal application is provided along with the set of required devices, a video demonstration and the complete package of OI components to run the demonstration.

Example of the Slide Viewer demonstration on the Showcase page of the OI website

5 Approach and Methodology

The OpenInterface approach is an innovative combination of software engineering activities and standard extension definition based on a common theoretical understanding of multimodal interaction as well as empirical results from the evaluation of the developed testbeds and validators. Figure 1 summarizes the OI overall approach.

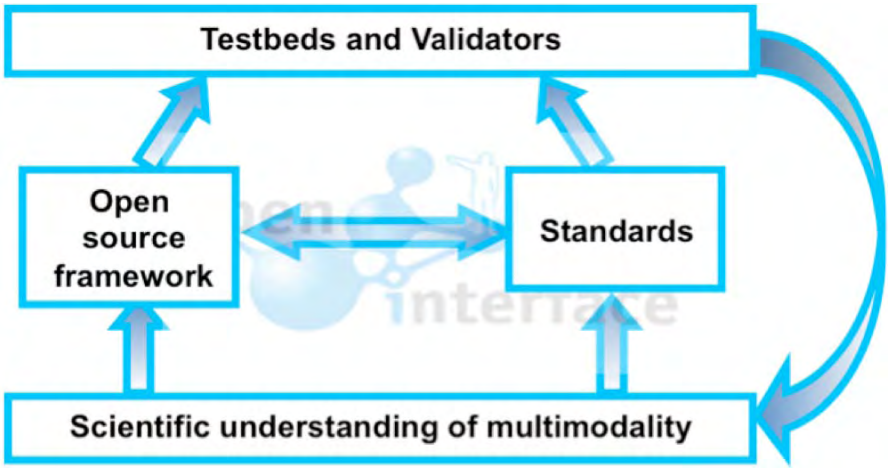


Figure 1: OI overall approach.

The key result of the OI project is the open source framework. Its design and development is based on expertise ranging from human-computer interaction, signal processing, software engineering to ergonomics and psychology.

- For designing the OI framework, we adopted an incremental approach and tested the OI framework by developing different versions of testbeds. For designing the testbeds, we applied an iterative User-Centered Design (UCD) process. Two application domains have been selected for proving the unavoidable generic aspect of the framework: Large Information Space (e.g., navigation in a map) and Game. Four complete iterations have been conducted during the project period as shown in Figure 2.

- For developing the OI framework, we adopted a Component-Based Development (CBD) approach that offers the established advantages of reducing the production costs, and of verifying the software engineering properties of reusability maintainability and evolution.

For standard related activities, the work has been based on the OI framework as well as on the experimental results of the testbed evaluations. The Multimodal Middleware Protocol was specified as a new standard. The multimodal hub (MMhub in Figure 2) and browser implement our Multimodal Middleware Protocol and are running on mobile phones. To test them, we designed and developed two validators running on mobile phones. Figure 2 explained the links between the testbeds developed using the OI framework and the validators developed using the multimodal hub.

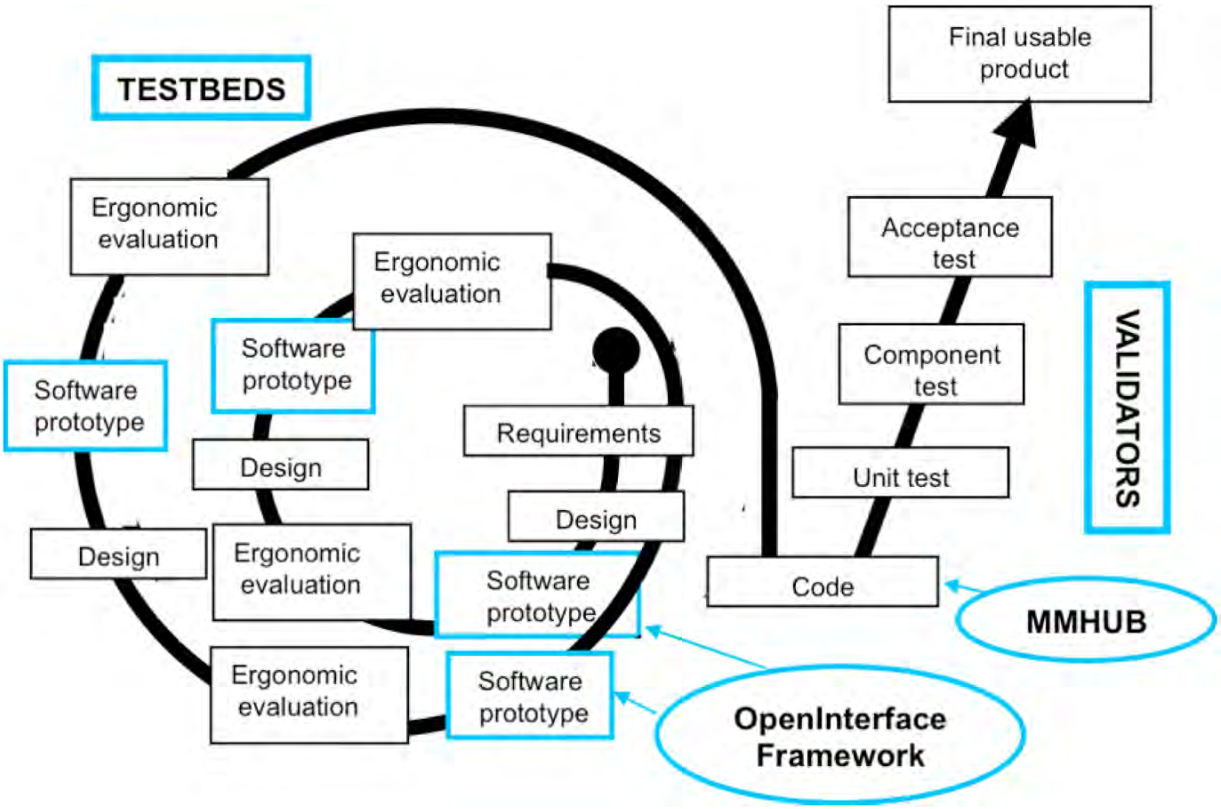


Figure 2: Iterative user-centred design process: OI testbeds and validators.

6 Results and Achievements

6.1 Overview

In OpenInterface, we studied multimodal interaction on mobile phones as well as multimodal interaction on an augmented table, at home for example navigating in a map looking for a restaurant. Facing the vast world of possibilities for interaction modalities, tools for integrating and combining those modalities become a real challenge addressed in the OpenInterface project. Focusing on engineering multimodal interaction, the key results of the project:

(1) The **OI framework** for the rapid development of multimodal interaction: The open source framework available and fully documented at www.oi-project.org is made of an underlying platform, namely OpenInterface platform, and the OpenInterface Interaction Development Environment (OIDE). The key aspect of the underlying component-based platform is to handle distributed heterogeneous components based on different technologies (Java, C++, Matlab, Python, .NET). It allows the integration of existing interaction modalities written in different languages. The OIDE adds development tools offering access to interaction capabilities at multiple levels of abstraction: it includes a component repository, a graphical construction tool, and debugging as well as logging tools. Some evaluations of the OIDE itself have been conducted.

(2) The **multimodal hub and multimodal browser**: going further than prototyping multimodal interaction on mobile phones using our OI framework (running on PC), the multimodal hub and browser implement **a new protocol (standard extension)** and are running on mobile phones. There are J2ME and Android implementations as well as an Open Source Python implementation. Such a tool enables us to directly reuse OI components for developing multimodal interaction on mobile phones.

These tools have been used to explore various multimodal interaction based on a large set of innovative devices in the context of **testbeds** (four versions of test beds for multimodal game and navigation in large information space) and of two **validators** fully running on mobile phones. Such an experience also leads us to collect **experimental data on multimodal interaction** during the in-laboratory and in-field evaluation that we carried out.

6.2 Open source OI framework

The designed and developed open source framework is dedicated to the development of multimodal interaction:

- that handles a rich and extensible set of modalities,
- that enables quick replication,
- that enables a focus on innovation (new modalities or forms of multimodality),
- that enables iterative user-centred design.

To fully understand the scope of the OpenInterface framework we show in Figure 3 where the automatically generated code is located within the complete code of the interactive system structured along the Arch model.

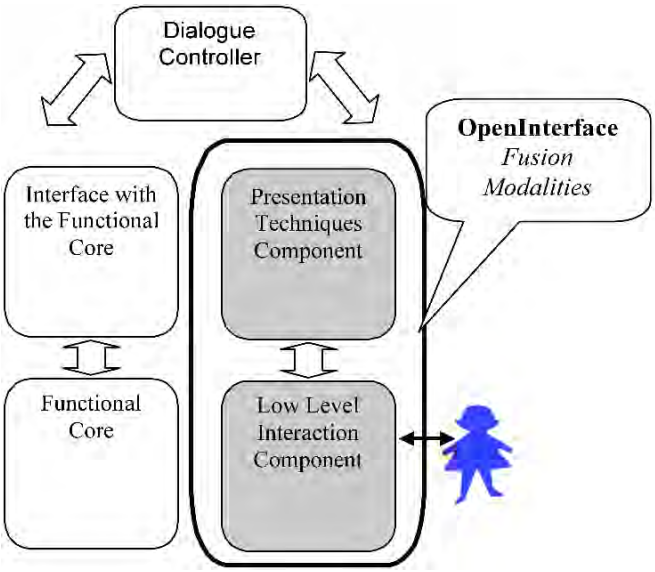


Figure 3: OpenInterface framework and the Arch software architecture.

The OpenInterface (OI) framework is made of a repository of modalities, an underlying platform, namely OpenInterface platform, and the OpenInterface Interaction Development Environment (OIDE).

• The OI repository embeds a set of pure and combined modalities as reusable components and generic mechanisms for combining modalities. It includes modalities both developed ”in-house” (e.g., speech recognition, video “finger tracker”, accelerometer-based gesture recognition) and also accessible via proxies to other component sets.

• The OI platform is a component-based platform that handles distributed heterogeneous components based on different technologies (Java, C++, Matlab, Python, .NET). It allows the integration of existing interaction techniques written in different languages. The platform includes a tool for creating new components (i.e., interaction techniques) from existing code without modifying the original code.

• The OIDE is a component-based system built on top of the OI platform. OIDE adds development tools offering access to interaction capabilities at multiple levels of abstraction. The OIDE includes a component repository as well as construction, debugging, and logging tools.

The diagram of Figure 4 shows a high level view of the relationships between the main components of the OpenInterface framework.

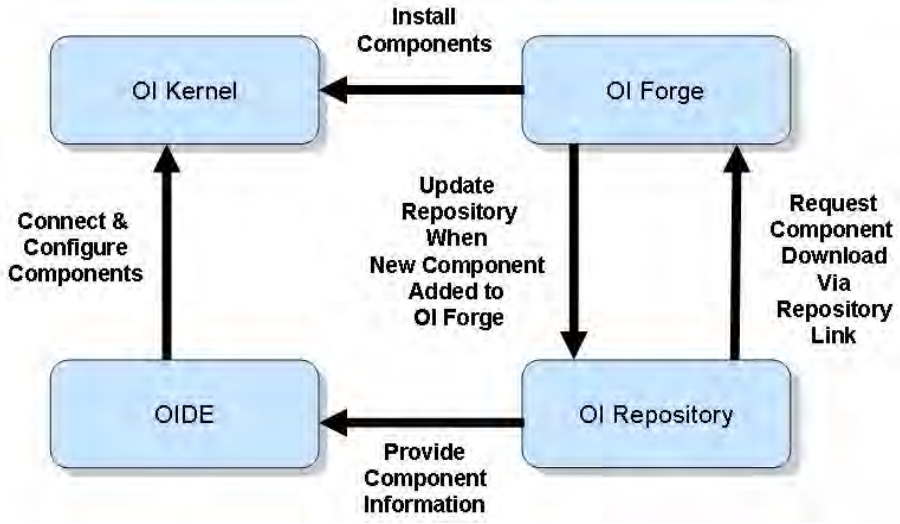


Figure 4: Main components of the OpenInterface framework.

6.2.1 OI repository

6.2.1.1 Component repository

The component repository includes modalities both developed ”in-house” (e.g., speech recognition, video “finger tracker”, accelerometer-based gesture recognition) and also accessible via proxies to other component sets (e.g., Phidgets, ARToolkit). Our components include device drivers, interaction techniques, multimodal fusion mechanisms, development services and developer-defined combinations. The repository supports descriptions, querying and access in an extensible set of description schemas.

Within the vast world of possibilities for input modalities (from the user to the system) as well as for outputs (from the system to the user), we distinguish two types of modalities: the active and passive modalities. For inputs, active modalities are used by the user to issue a command to the computer (e.g., a voice command or a gesture recognized by a camera). Passive modalities refer to information that is not explicitly expressed by the user, but automatically captured for enhancing the execution of a task. For example, in the “Put that there” seminal multimodal demonstrator of R. Bolt (MIT) that combines speech and gesture, eye tracking was used for disambiguating among multiple objects on screen. A huge variety of passive modalities can be used in interactive multimodal applications including emotion, bio-signal processing, location mechanism (e.g., GPS) and eye-movement.

At the end of the projects, 82 components were developed and available in the OI repository. They include device and transformation components (i.e., modality) as well as connectors and composition components. Several new innovative modalities have been designed, developed and experimented as part of testbeds or demonstrations. We give here three examples of designed devices (hardware and software) and modalities.

Examples of three innovative devices and modalities

SHAKE (Sensing Hardware Accessory for Kinaesthetic Expression) is a small Bluetooth interaction device (Figure 5) that contains a 3-axis accelerometer, gyroscope, 3-axis magnetometer, a vibrotactile actuator, 2 capacitive sensors (used as virtual buttons), and a physical button. The last SK7 variant of the SHAKE features a new hardware design compared to the previous SK6 versions, with the most visible improvement being the replacement of the 2 capacitive sensors on the upper surface of the case with a grid of 12 sensors, allowing basic “multitouch” interaction to take place. It also contains improved gyroscopes and has a slightly smaller casing than the SK6. Figure 5 presents an example of the Large Information Space (LIS) testbed using the SHAKE: Navigation in GoogleEarth is performed by tilting the SHAKE in the 4 main directions (left/right/forward/back) to move west/east/north/south, and by pressing the two capacitive sensors (zoom in and out).

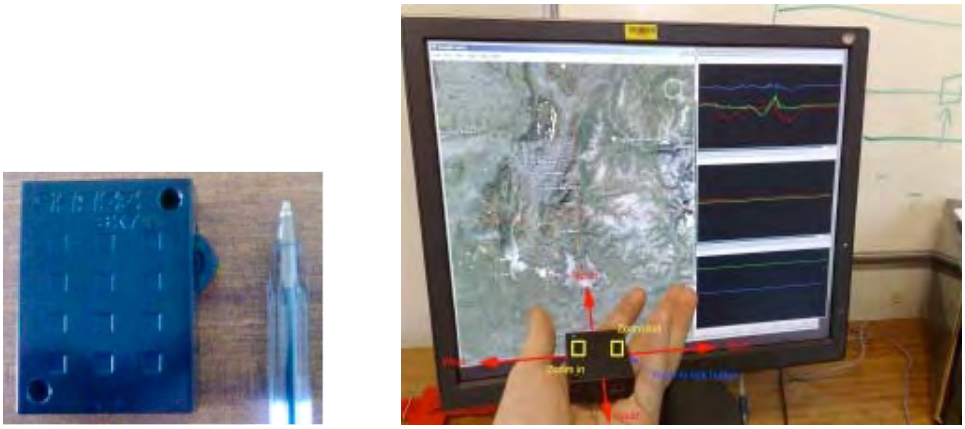
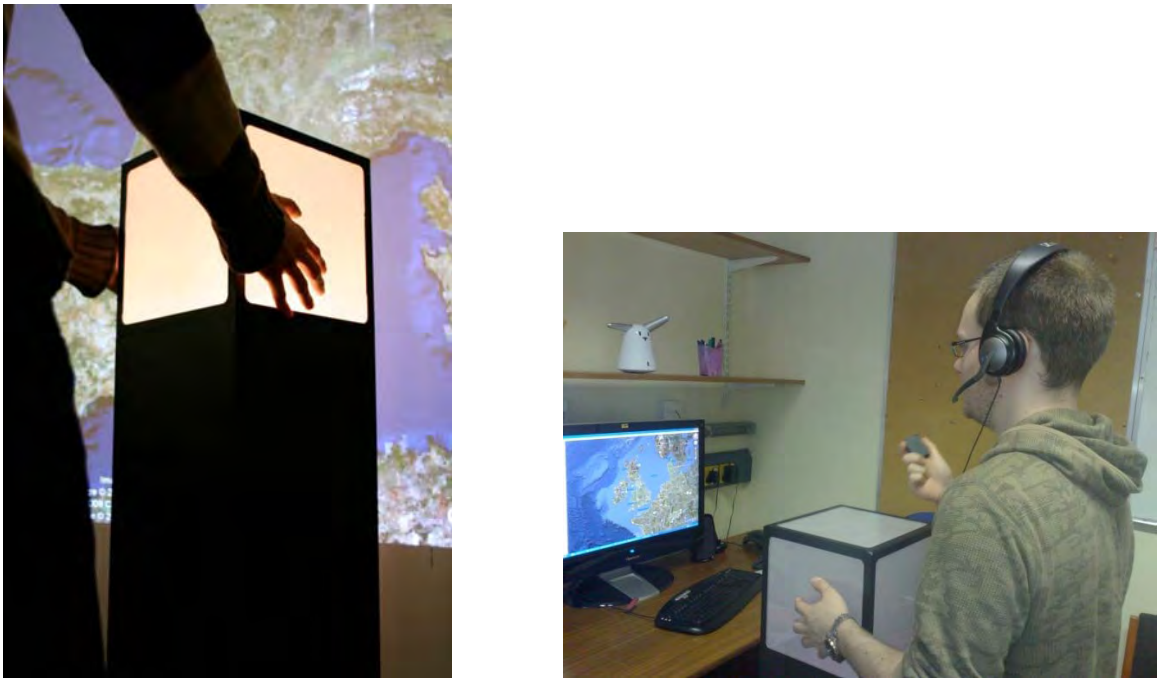


Figure 5: SHAKE SK7 (left) and LIS testbed (right): navigation by tilting the SHAKE SK6.

CUBTILE (Figure 6-a) is a 3D multitouch interface. The manipulated cube has five multitouch surfaces and is able to determine which fingers are touching which surfaces at the same time. For example it is possible to make gestures in parallel on two different faces. With this device, rotation, for example, can be done with a single hand on the top face or with two hands at the same time on two opposite sides of the cube, as if you were turning a real object thanks to its DoF redundancy. It offers an association to a 3D world reference frame thanks to having a 3D cube volume between the fingers. It also offers multiuser applications giving the opportunity for two people to interact at the same time with the cube on different faces. The cubtile is equipped with a high resolution camera calibrated according to the ambient light for the detection.

Using the cubtile, several demonstrations have been developed using the OI framework. For example a multimodal system for interacting with GoogleEarth has been developed (Figure 6-b). The globe in GoogleEarth is controlled using multi-touch finger gestures on the 5 surfaces of the Cubtile. Direct navigation to cities is possible through speech recognition e.g. “London”. Speaking the name of an interesting geographical landmark (e.g. “beach”) in isolation will take the user to the predefined location of such a landmark. However, saying that word at the same time as shaking the SHAKE will place a marker on the map with the given tag (“beach”) adding a point of interest.



(a) (b)

Figure 6: (a) Cubtile device. (b) Cubtile (centre) and SHAKE (in right hand) being used to interact with GoogleEarth.

AIRMICE (Figure 7) is a new interaction modality based on 3D finger gestures above the keyboard of a laptop. At a reasonably low cost, the technique can replace the traditional devices for pointing in two or three dimensions. We implemented a vision-based version of AirMice.

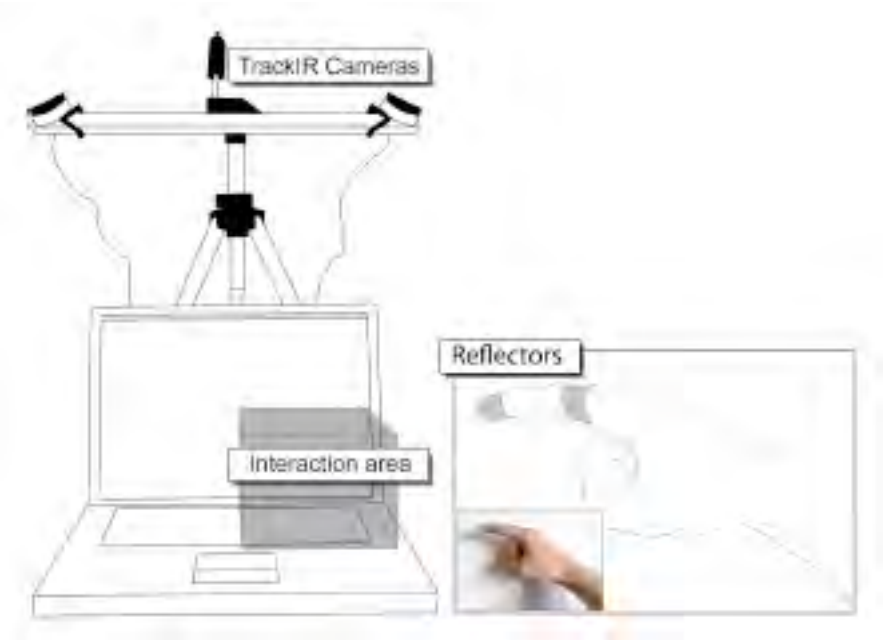


Figure 7: AirMice 2D/3D pointing modality.

In two controlled experiments, we evaluated the performance of the implemented technique of AirMice as a pointing device, using 2D and 3D Fitts' law studies. The two tasks have been performed by 15 subjects with no prior experience with 3D interaction devices. For the 3D Fitts' law study, the goal was to position the 3D pointing performances of AirMice in relation with other traditional device performances. We then compared our technique for 3D translation pointing with two well-known devices: the PHANTOM and the SpaceNavigator (figure 8).

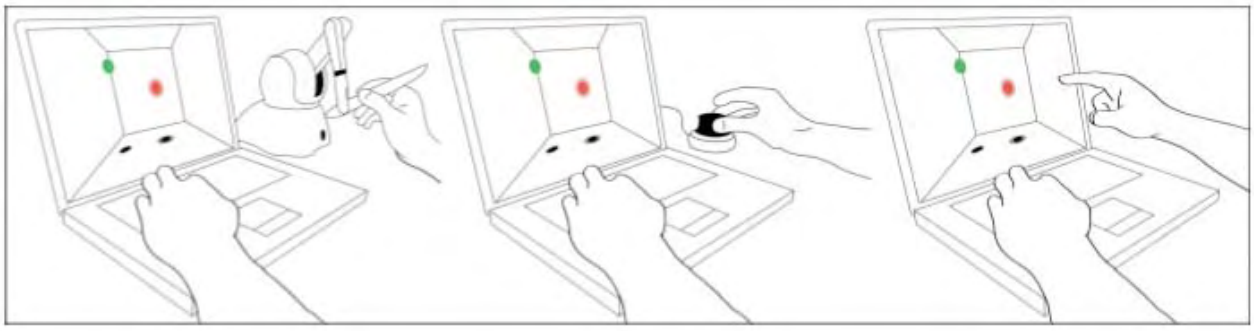


Figure 8: Experimental setting: Three configurations for the 3D pointing evaluation. Left: the PHANTOM, an arm-based device. Middle: the SpaceNavigator, a space-mouse. Right: AirMice technique, only fingers above the keyboard.

Results show that the performances of AirMice in 2D pointing are not better than the ones of the mouse, but are comparable to the ones of the touchpad, and are better than the ones of the key-joystick. In 3D pointing, AirMice is not so far from the PHANTOM and is much faster than the spacemouse. Finally, qualitative results show that the performance is not the most important criterion. Subjects prefer to use a device that is intuitive and easy to learn while providing correct performance. Based on these criteria, AirMice is appreciated by most of the users: they consider the technique as promising, and useful for laptop configuration.

6.2.1.2 Online Repository

The online OpenInterface Repository (http://dolak.dcs.gla.ac.uk:8080/OIRepository/) holds a collection of interaction elements for use in the development of interactive systems, especially those constructed using the OpenInterface framework. Repository users will find descriptions of a variety of interaction techniques using different interaction modalities, such as gesture and speech, plus the OpenInterface components and configurations needed to implement these techniques. Registered users can also edit these descriptions or add new ones.

Entries are intended to provide information that will help an interaction developer to understand what an interaction technique or component does, when it might be used in a design and how it can be implemented and integrated into an application. Illustrations and videos sometimes accompany the textual description.

The visual design of the online OI repository is based on users' experiments and focus groups. Figure 9 shows the visual design for individual entry while Figure 10 presents an example of search results.

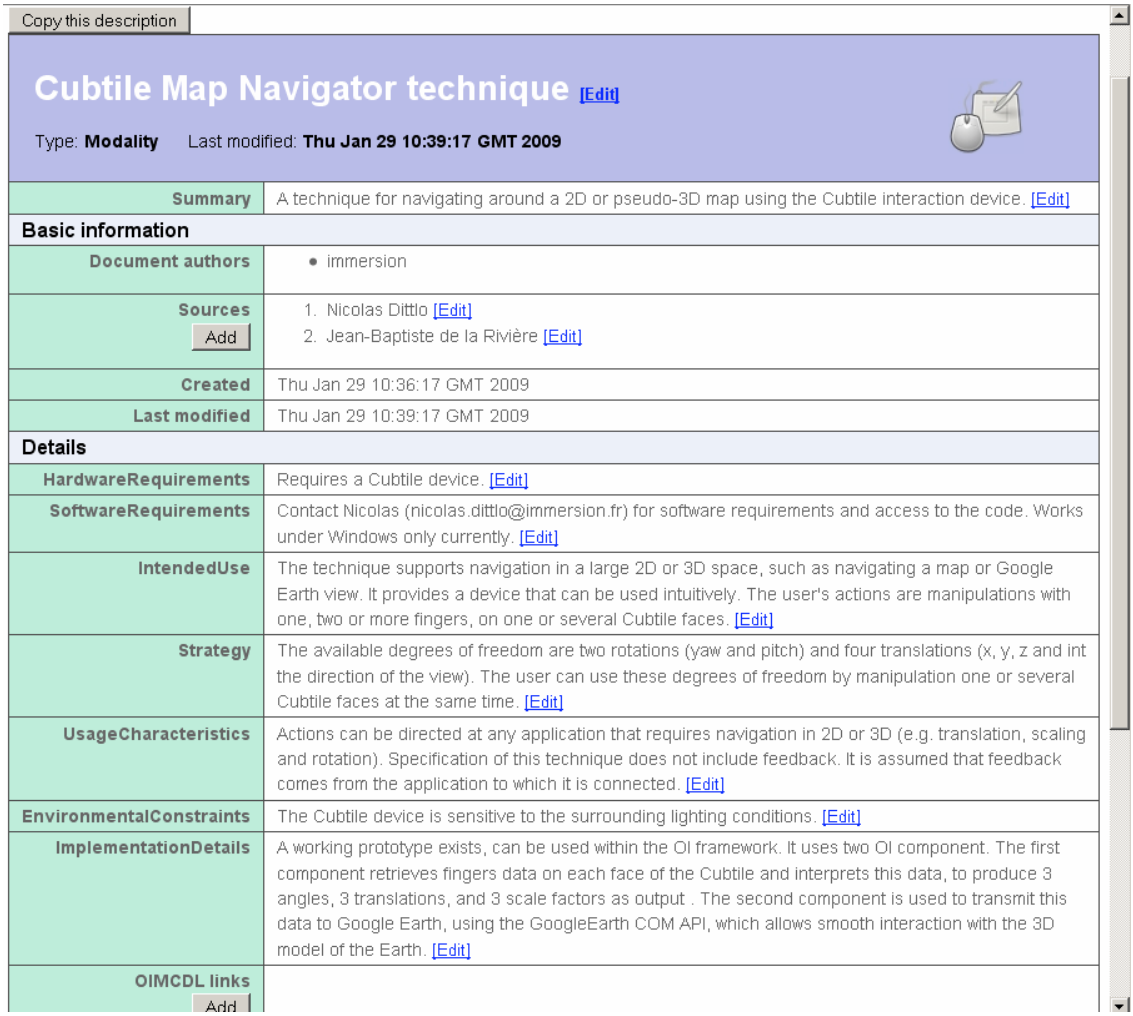


Figure 9: OI online repository: Visual design for individual entry. (Cubtile Map Navigateur)

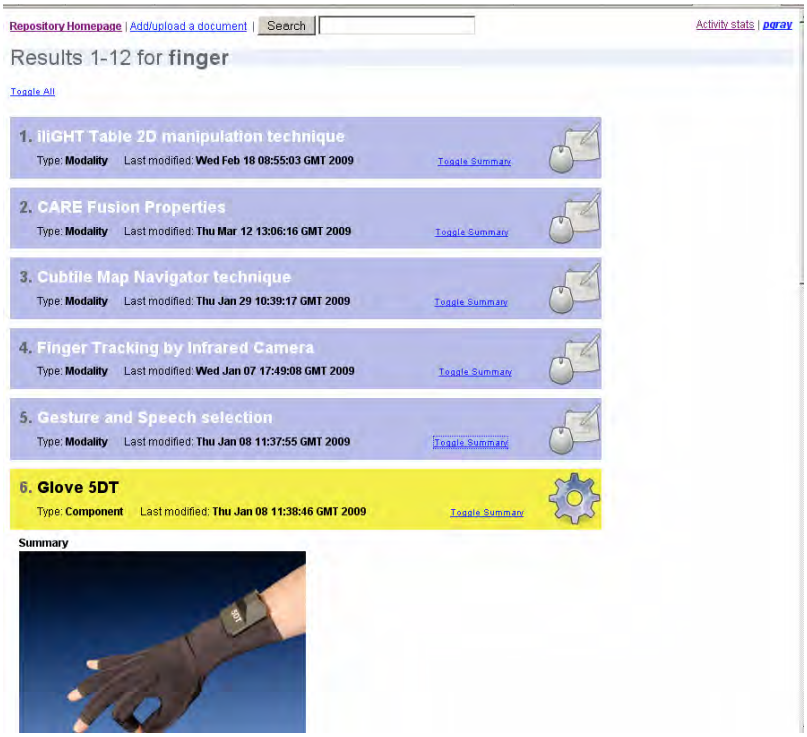


Figure 10: OI online repository: Visual design for search results.

6.2.2 OI kernel

OpenInterface Kernel is a generic runtime platform for integrating heterogeneous code (e.g. device drivers, applications, algorithms, etc.) by means of non-intrusive techniques and with minimal programming effort, while achieving exploitable runtime performances (e.g. low latency, low memory overhead). The OI kernel adopts an extensible modular architecture in which components are the basic manipulated objects. Components can be implemented in virtually any language: we do not constrain to the use of a component model, and we strive for minimal programming efforts when integrating new components. Components are unaware of the kernel in which they are running; therefore, programmers can use any preferred programming language and external tools, while only declaring interfaces. Within the kernel, a component is only characterized by its interface and is defined as a reusable and independent software unit with exported and imported Input/Output interfaces. Components to be used within the OI kernel are only specified by the following mandatory attributes:

1. API (Application Programming Interface): to communicate with the component services.

2. Installation/configuration: to facilitate the installation and configuration, the component should be packaged appropriately.

3. Documentation: the component must be well documented to enhance reusability.

4. No dependencies with other components: a component must not make assumptions about the platform or features of other components. All required features must either be declared as imported, or packaged within the component.

We used a code generation technique to provide a non-intrusive integration platform with minimum programming overhead, while achieving good runtime performance. The integration of heterogeneous software is done by (1) describing all components communication interfaces in a language and platform-independent manner, and (2) by generating proxies to encapsulate original components implementation.

Having components only declaring their communication interface enforces the requirement of “independence”. A component exports inputs and outputs to provide functionalities and services (e.g. image display, device status), while it imports inputs/outputs to request features provided by

other components. In order to declare interfaces, regardless of their implementation language, we define an XML-based CIDL description language (Component Interface Description Language). CIDL code is semi-automatically generated from source code, and is required by the OpenInterface kernel for manipulating components. Once the CIDL description of the component has been produced, the platform generates C++ code to encapsulate external binaries into a well defined programming interface, as illustrated in Figure 11. The encapsulated components can then be easily reused in any multimodal application in a plug and play fashion by using the OpenInterface Interactive Development Environment (OIDE) presented in the following section.

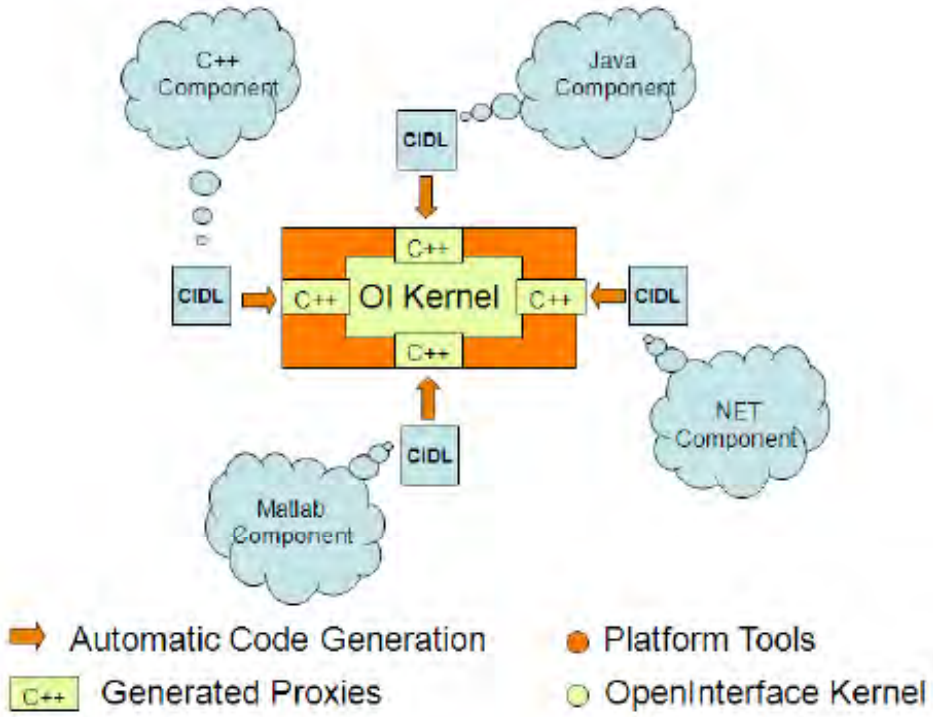


Figure 11: OI kernel: Heterogeneous Components Integration within OpenInterface.

6.2.3 OIDE: Construction and logging tools

The OIDE (OpenInterface Interactive Development Environment) is a component-based system built on top of the OI kernel. OIDE adds development tools offering access to interaction capabilities at multiple levels of abstraction.

Figure 12 illustrates the OIDE Graphical Editor in which OI components accessible via the repository (section 6.2.1) can be inspected, configured, linked to other components and to external services or to application functionality, in the assembly view. The graphical environment allows direct manipulation and assembling of components in order to specify a "pipeline" defining a multimodal interaction. The result can be either an individual technique or a fully functional user interface to an application.

In the example of Figure 13, we can see the representation of the assembly for specifying the multimodal interaction for a zoom task along with a screenshot of the same assembly implemented in the OIDE. The DiamondTouch is used for capturing the position of the finger (i.e., the point to zoom) while the speech recognition supports zooming commands such as “zoom here”. The two modalities are used in a complementary way.

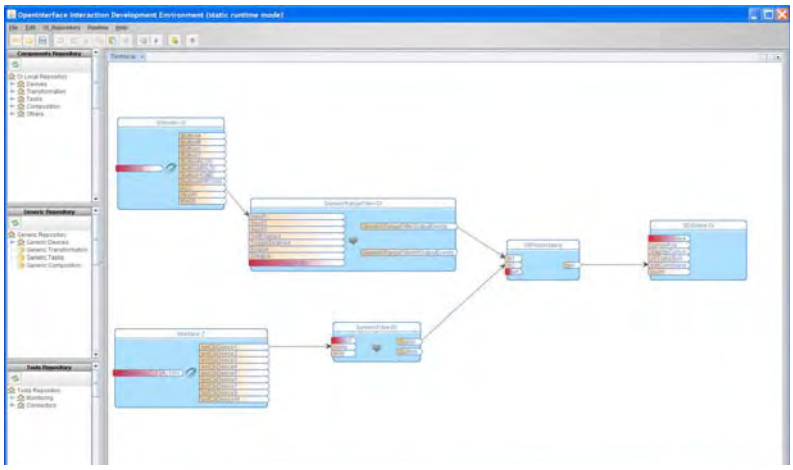


Figure 12: Overview of the OIDE graphical editor: 1-Palette of components (modalities, devices, fusion and transformation components). 2- Editing zone for assembling the components and defining an OI component pipeline.

Figure 13: OIDE: Zoom task using the DiamondTouch and the voice.

The logging tool is responsible for recording multiple component-specific formatted data streams while an OI application is executing. A generic “oscilloscope” display, that can be configured while in use, can be used to display data streams at runtime. In Figure 14, the oscilloscope shows data at the lowest level of abstraction, direct from the device itself (in the example, the SHAKE device) but it is also possible to display application-level data (as shown in Figure 5 right).

Figure 14: OIDE Debugging and Logging tools. In the example, the Shake accelerometer events are logged in order to tune the interaction with the mobile game.

6.2.4 Conclusion: OI toolset

To conclude the presentation of the OpenInterface (OI) framework, Figure 15 provides an overview of all the tools of the framework and their relationships. While using the OIDE, the user can access the OI repository and install new components from the OI forge. The assembly of components specified using the OIDE is executed using the OI kernel and can be tested by end-users.

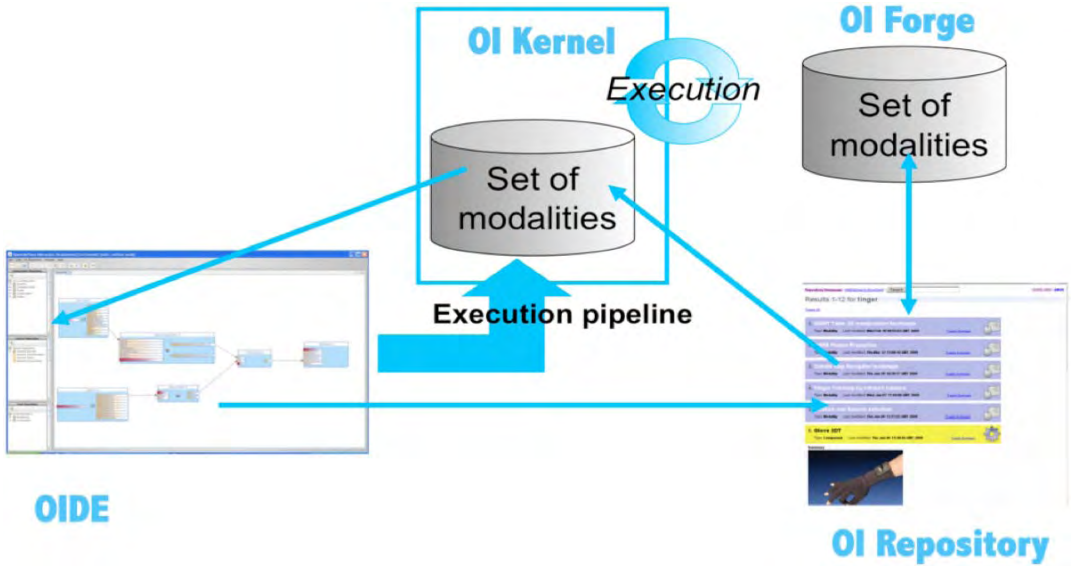


Figure 15: OI framework in use.

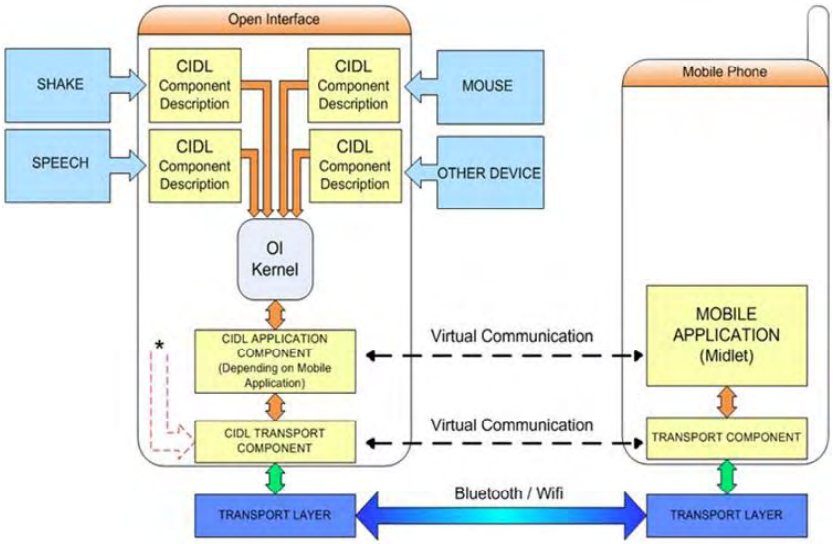
6.3 Testbeds

Testbeds have been used to explore the functionality of the OI framework. In the project, we considered two interaction contexts, namely a mobile and an ubiquitous setting, and two testbeds, namely multimodal interaction for large information spaces (e.g. maps), and multimodal interaction for games. During the project and as shown in Figure 2, we developed four versions of each testbed using the OI framework and embedding various modalities including speech, gesture with the SHAKE and head movement. Figure 16 presents the four versions of the large information space testbeds, while Figure 17 the four versions of the game testbeds.

Figure 16: Four versions of the OI Large Information Space testbeds.

Figure 17: Four versions of the OI Game testbeds.

The tesbeds has been used for testing the functionality of the OI framework. For the testbeds running on mobile devices, the testbeds demanded for an architecture that links the OI platform running on a PC with the functional core of mobile application running on devices such as PDAs or Smartphones. Testbed applications were expected to be flexible and open, to allow fast prototyping for experimentation, but also consistent and adequately performing, in order to address actual use cases. This tradeoff required careful architectural choices. The designed architecture is based on a layered protocol stack approach, as shown in Figure 18. We designed two additional OI components: the Application Component and the Transport Component. The Application Component acts as a proxy for the mobile application. It is connected with other OI components through the OI platform pipe. Whenever an event from any connected component occurs, the Application Component parses such an event and sends a corresponding message to the mobile application through the Transport Component. Figure 18 presents a setup that includes the SHAKE device, a Speech Recognition engine and a traditional mouse, all plugged into the OI kernel. Such devices and modalities have been selected and assembled using the OIDE editor of Figure 12. An additional sink/source pair in the Application Component is connected to the Transport Component. The Transport Component defines an interface for transferring messages from the OI platform to the mobile application and vice versa. Specific implementations of the Transport Component adopt different physical transport layers, such as Bluetooth or TCP/IP over WiFi. It is worth noting that the abstract transport layer interface also allows non physical connection, when the OI framework and the test-bed functional core are running on the same machine.



* Any other component can use the Transport Component to communicate with remote devices using the protocol described in the transport layer.

Figure 18: Testbed architecture connecting the OpenInterface framework running on a PC with an application running on a mobile device.

As part of the iterative design approach of Figure 2, each version of the two testbeds has been experimentally evaluated. That enabled us to collect empirical results on the usage of multimodal interaction for our two application domains: Large Information Space and Game. For example, for the Game testbed, we studied the usage of the SHAKE device for moving the 3D character. Users confirmed that this modality improved the gaming experience.

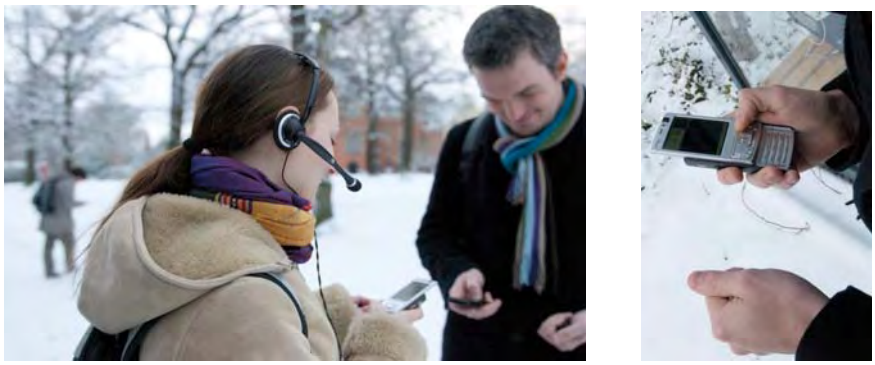


Figure 19 : In-field evaluation of the last version of the Game testbed.

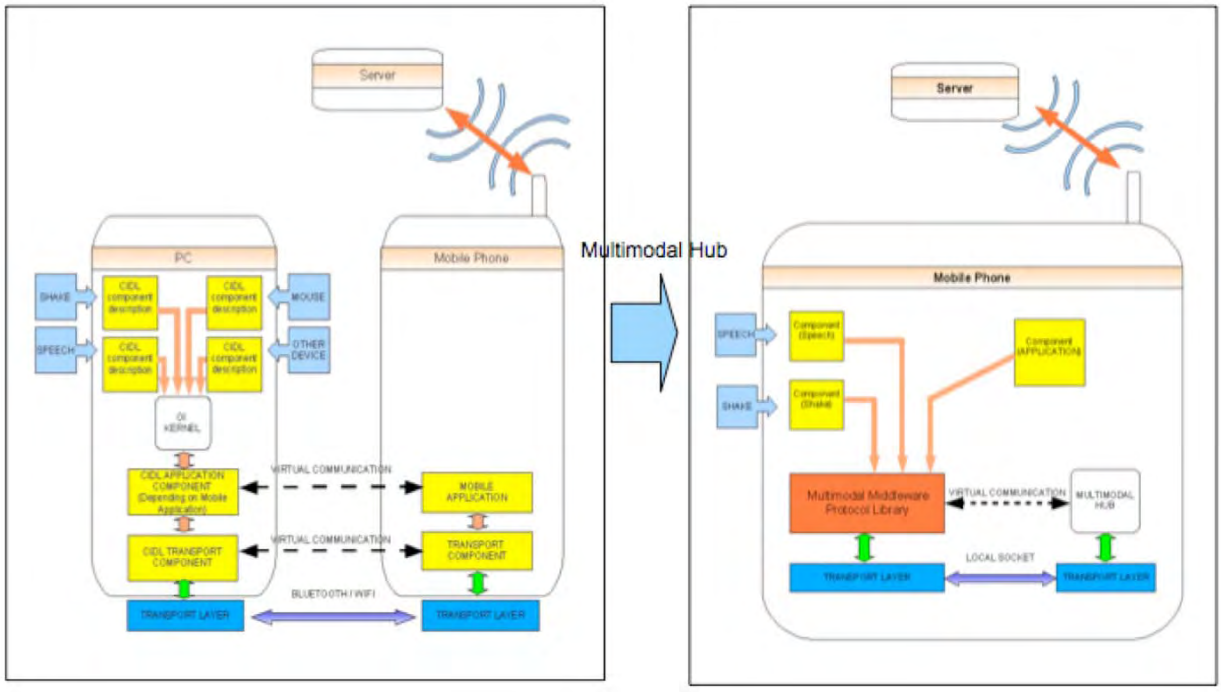
The knowledge gained from the testbed evaluation has been essential for the design of the two validators (section 6.5) as well as gaining much important information about using the OI framework in the field.

6.4 Standards extensions for multimodal mobile devices

The OI standardization process was based on the study of existing solutions and a new solution was designed, implemented, tested and submitted to external evaluation. The OI project produced a new protocol specification with an open source implementation on Source Forge and other implementations running on mobile devices. The full protocol specification was published to allow third party implementation of the Multimodal Middleware Protocol. A very important standardization activity started on ETSI where the creation of a new ISG (Industry Specification Group) is being carried out. The standardization activity will continue with ETSI and the creation of the new ISG.

The new protocol, called the Multimodal Middleware Protocol, allows the creation of a component network able to exchange messages between each other. The MultiModal Hub (MMH) is the central point of this network and all components connect to it. MMH acts like a network server, receiving connections using one or more network protocols like UDP, TCP or Bluetooth. Another feature of the MMH is to enable the connection of components using different network protocols, using the MMH as a translation engine between them. MMH is run first, before all other components. It manages the list of produced events and also consumed ones, looking for matches, forwarding messages and also updating all components in case of disconnection. The Multimodal Browser (MMB) is used as an interface to the core of the Multimodal Hub (MMH). The MMB is responsible for listing all connected components and also for allowing editing and application of new rules dynamically. These rules were also specified on the Multimodal Middleware Protocol in XML format. A rule set can be saved or loaded to store a user profile or configuration. The MMB can be used to manage the component network, enabling or disabling the whole network if necessary or selectively activating parts of it.

The Multimodal Hub was based on the OI Framework component based approach, but adapted to run on mobile devices. Figure 20 shows the architecture of a multimodal application with the OI framework and with the Multimodal Hub.



(a) **(b)**

Figure 20: Multimodal interaction on mobile devices: (a) Architecture with the OI framework (b) Architecture with the Multimodal Hub.

The open source implementation of the Multimodal Middleware Protocol and of the Multimodal Hub and Browser were released on SourceForge website with the name of Python Multimodal Hub. The Multimodal Hub and Browser have been used for developing demonstrations as well as the two OI validators running on mobile devices.

6.5 Validators

As part of our iterative user-centred design approach of Figure 2, the two validators have been designed based on the experimental evaluations of the two testbeds. They are developed using the Multimodal Hub and Browser (section 6.4) and are running on Nokia N95 and Nokia N93. Figure 21 presents the two designed and developed validators. The architecture of two validators are presented in Figure 20-b. The effort spent on defining an appropriate component architecture for the testbeds (Figure 18) greatly reduced the time and effort required to develop the validators by allowing component developers to work on the same interfaces as on the testbeds, to reuse significant amounts of existing code, and to allow the testbeds to continue to be a useful tool for testing new features throughout the whole period of the validators development.

(a) **(b)**

Figure 21: The two OI validators running on mobile devices:
(a) Large Information Space – Map (b) Game

The two validators have been experimentally evaluated (Figure 22). For example, for the Large Information Space validator, we investigated how multimodal interaction affects collaboration and map navigation on mobile devices. We conducted a user study in the field where pairs of users had to find and photograph points of interest on the map. The task for users was to find and photograph Points of Interest (POIs) inside a public park. Controlling the map was done either through the phone keypad or by tilting the SHAKE – tilting the device forward caused the map to scroll up, tilting it back caused the map to scroll down, and similarly for scrolling left and right. Points of interest were marked on the map as blue circles. To display information about a point of interest, users had to perform an "open" gesture using the SHAKE device, then perform a "close" gesture to return to the map. Taking a photograph of a POI was done by using the hardware button on the SHAKE as a shutter control for the camera in the N95. 18 users, (9 pairs) were involved in total. All participants completed the task successfully in that they could use the map, use the SHAKE to navigate, find POIs, use open and close gestures, and construct a tour of the park. Based on the collected data, navigation was achieved easily and used often with the tilt based control on a second device despite some participants finding it much easier and more natural to simply use the keypad on the mobile phone itself. When displays are shared, the designed interaction methods might offer additional benefits. Both the display and the control of the display can be shared and could lead to increased collaboration in some cases.

(a) **(b)**

Figure 22: In-field evaluation of the two OI validators:
(a) Large Information Space – Map (b) Game

7 Dissemination and use

For disseminating the work of the OpenInterface project, in addition to the dissemination materials that include the complete documentation of our toolset, the web site and two flyers, we organized or participated in several national and international events. We organized OpenInterface workshops, meetings, stands as well as demonstrated the OpenInterface project in numerous scientific international conferences.

We summarize here five key OpenInterface events that we organized and provide a complete list of published articles directly related to the OpenInterface project.

7.1 Key OpenInterface events: workshop and exhibit

7.1.1 OI workshop at DSV-IS 2008

As a collocated event to the DSV-IS (Design, Specification and Verification of Interactive Systems) 2008, hosted at Queen's University in Kingston, Ontario, Canada, from Wednesday July 16 to Friday July 18 2008; an OpenInterface workshop was set up on the first evening of the conference.

The objectives of this workshop were to disseminate widely the OpenInterface concept, attract high level developers to the open source community, and obtain feedback from experts concerning implementation and future standards proposals. The main aim was to present our results to the academic community and to receive user feedbacks. We have asked participants to submit requests on applications that they would like to have converted to multimodal application. We showed and explained how OpenInterface components can be integrated and reused. We presented common case studies to illustrate the capabilities and functionalities of the platform. Participants - motivated by the call for participation, 10 people had initially registered before the workshop. This figure increased up to a total of 23 the very day of the workshop, out of roughly 60-70 participants registered to the main conference. The workshop was held by Laurence Nigay, Marcos Serrano (UJF), Jean-Yves Lionel Lawson, Christian Frisson (UCL) and Phil Gray (UGLA). Laurence Nigay and Jean-Yves Lionel Lawson presented the OpenInterface Platform, the general concepts of the OpenInterface platform, and a focus on the kernel respectively. Marcos Serrano presented the OpenInterface Interaction Development Environment (OIDE) and some demonstrators, Phil Gray previewed the Open Interface Interaction Technique Repository and closed the event, inviting participants to fill-in a feedback form.

Before the DSV-IS conference, many USB keys had been prepared so that to have OpenInterface distributed to potential users. Many participants retrieved the USB key contents, curious to test the OpenInterface platform. This dissemination procedure was successfully more interactive and engaging than providing a single hyperlink on presentation slides.

7.1.2 eNTERFACE'08 summer workshop

The eNTERFACE summer workshops (www.enterface.net) are a new type of European workshops devoted to multimodal interfaces. They aim at establishing a tradition of collaborative, localized R&D work by gathering in a single place a group of senior project leaders, researchers, and (undergraduate) students, working together on a pre-specified list of challenges, for 4 weeks. Participants are organized in teams, attached to specific projects working on modular software. The ultimate goal of this workshop is to make this four weeks event a unique opportunity for students and experts to meet and work together, and to foster the development of tomorrow's multimodal research community.

The 4th in the series of annual workshops, this year eNTERFACE'08 was held at The Computer Sciences Laboratory for Mechanics and Engineering Sciences (LIMSI) in Orsay (Paris), France on

August 4-29, 2008. The event was supported by OpenInterface STREP and OpenInterface Association. 78 participants from 20 countries were working on and presented convincing final results achieved in 10 projects in various aspects of multimodal interfaces. One of the projects, led by an OI STREP member from UCL, was devoted to development of a multimodal high-level data fusion tooling to be integrated in the OI framework. The advantages of using the OI framework and interaction modalities in multimodal applications development were advertised in the presentations and OI flyers were distributed among the workshop participants.

7.1.3 OI exhibit at ICT '08 conference

The OI project has been accepted for a stand at ICT 2008 with 4000 participants in Lyon, a major event in the ICT community for both academic and industrial actors.

Our OI exhibit demonstrated the power and benefits of the OI framework by considering a simple and engaging interactive application: a multi-display and multi-user breakout game, shown in Figure 23. The game was highly interactive and intuitive enough to attract numerous visitors on the booth. Moreover the selected interactive application was generic enough (i.e., move a paddle to catch the balls, modify the size of the paddle) to help any representative of the other domains to understand the capabilities of the OI framework. For example, we had contacts with an industrial from the Netherlands working on museum exhibits.

Figure 23: OpenInterface Stand at ICT 2008.

7.1.4 OI and ETSI workshop on standardization of multimodal mobile devices

ETSI, in collaboration with the OpenInterface Project held a workshop on Standardization of Multimodal Interaction on Mobile Devices. The workshop has been organized at ETSI Headquarters, in Sophia Antipolis, France, on the 18th and 19th of November 2008. There is a high potential for new interaction modalities and multimodal interaction on mobile devices since it enables the user to access functions on the mobile devices while walking (on the move) as well as allows the user to choose the modalities according to her/his changing context of use (running in the street, sitting in a noisy environment, etc).

The presentation by UJF included an overview of the OI project as well as the presentation of a design space for multimodality including the CARE fusion properties and definitions of a modality. This presentation introduced some definitions towards a common terminology among the academic and industrial partners. MULT presented a general overview of the Multimodal Middleware Protocol and the reasons that led to a new protocol specification. The Multimodal Hub and Multimodal Browser were presented to show actual implementation of the protocol and also the mobile phone feasibility.

In the same workshop other companies like Nokia, Research in Motion, Swisscom, Telecom Italia, VLingo, Deutche Telecom Laboratories, IRIT (Toulouse Informatics Research Institute), CERTEC (Lund University – Sweden) and USTL Lille (Sciences and Technology University of Lille – France) made presentations with different approaches on multimodality and how to close the gap between end-users and multimodal applications.

MULT applied to be a member of ETSI on January 2009 and was accepted as a full member in April 2009. The standardization activities will be carried by MULT beyond the lifetime of OI with the support of other projects like “Skywin Telecom” and “Do It Yourself Smart Experiences”. FIT made efforts with MULT to officially become part of the ETSI group on mobile multimodal interaction and therefore collaborated in-house with a research team working for the research project HYDRA (EC IP project) that deals with construction of a software framework for context aware and mobile applications.

7.1.5 OI workshop on Challenges of Engineering Multimodal Interaction

OpenInterface organized a workshop on Challenges of Engineering Multimodal Interaction: Methods, Tools, Evaluation, at FIT institute in Sankt Augustin, on 13 May 2009. This workshop was intended both as a showcase for OI project’s work and as a forum for discussing the current state of the art in multimodal interaction development and its current and future research challenges. The call for participation is presented in Figure 24.

Figure 24: OI workshop on Challenges of Engineering Multimodal Interaction.

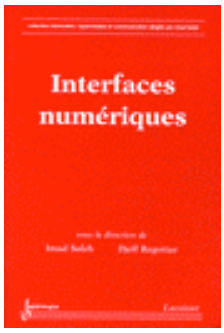
The objective of this workshop was to bring together researchers, designers and practitioners with a clear interest in the areas of human centered design and evaluation of multimodal interaction that can contribute to a vivid discussion leading to improved methods and tools to support such activity.

We received 10 position papers as well as demonstration proposals. 36 participants attended the workshop. 17 participants gave a presentation. The keynote speaker was Professor Luca Chittaro, Head of the HCI Lab at the University of Udine in Italy and the title of his talk was “Interaction in the mobile context: some challenges (and opportunities) for multimodality”.

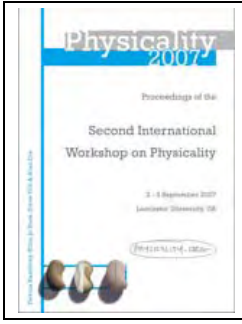
Based on the success of the workshop, we are currently preparing a special issue of the Springer journal on Multimodal User Interfaces (JMUI). The received papers are currently under review.

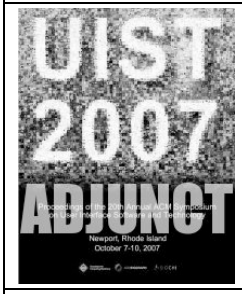
7.2 List of publications

Year 1:

	J. Bouchet (UJF), L. Madani, L. Nigay (UJF), C. Oriat, I. Parissis Formal Testing of Multimodal Interactive Systems Conference EIS 2007, Engineering Interactive Systems 2007, A joint conference IFIP WG2.7/13.4 10th Conference on Engineering Human Computer Interaction, IFIP WG 13.2 1st Conference on Human Centred Software Engineering and DSVIS - 14th Conference on Design Specification and Verification of Interactive Systems (DSVIS 2007), Springer-Verlag Berlin Heidelberg, LNCS (Lecture Notes in Computer Science), Volume 4940/2008, Spain, March 22-24, 2007, pp. 36-52.
	J. Williamson (UGLA), R. Murray-Smith (UGLA), S. Hughes (UGLA) Shoogle: Multimodal Excitatory Interaction on Mobile Devices Conference ACM CHI 2007, Conference on Human Factors in Computing Systems, ACM Press, USA, 28 April - 3 May, 2007, pp. 121-124.
	L. Nigay (UJF), C. Coutrix (UJF), P. Renevier Mixed Systems: Combining the physical and virtual worlds Systèmes interactifs mixtes: fusion des mondes physique et numérique In Interfaces numériques, Chapitre 3, Hermès-Lavoisier, ISBN 13 978-2-7462-1695-2, June 2007, 18 pages.

Year 2:

	E. Dubois, P. Gray (UGLA), A. Ramsay (UGLA) A model-based approach to describing and reasoning about the physicality of interaction Workshop Physicality 2007, The Second International Workshop on Physicality, UWIC Press, ISBN 978-1-905617-60-9, UK, September 2-3, 2007, pp. 77-82.

	G. Bailly (UJF), E. Lecolinet, L. Nigay (UJF) Wave Menus: Improving the Novice Mode of Hierarchical Marking Menus Conference INTERACT 2007, The eleventh IFIP TC13 International Conference on Human-Computer Interaction, Springer-Verlag Berlin Heidelberg, LNCS (Lecture Notes in Computer Science), Volume 4662/2008, Brasil, September 10-14, 2007, pp. 475-488.
	P. Gray (UGLA), A. Ramsay (UGLA), M. Serrano (UJF) A Demonstration of the OpenInterface Interaction Development Environment Conference ACM UIST 2007, The 20th annual ACM Symposium on User Interface Software and Technology, USA, October 7-10, 2007, Adjunct Proceedings, 2 pages. http://www.acm.org/uist/archive/html/proceedings/2007adjunct.html
	M. Horchani (UJF), B. Caron, L. Nigay (UJF), F. Panaget Natural Multimodal Dialogue Systems: A Configurable Dialogue and Presentation Strategies Component Conference ACM ICMI 2007, The Ninth International Conference on Multimodal Interfaces, ACM Press, Japan, November 12-15, 2007, pp 291-298.
	C. Coutrix (UJF), L. Nigay (UJF) Interacting with a Mixed Object: Physical and Digital properties Conference IHM 2007, The Nineteenth National French conference on Human-Computer-Interaction, ACM Press, France, November 13-15, 2007, 8 pages.
	G. Bailly (UJF), E. Lecolinet, L. Nigay (UJF) 15 Years of Research on Menus: Criteria and Properties of Menu Techniques Conference IHM 2007, The Nineteenth National French conference on Human-Computer-Interaction, ACM Press, France, November 13-15, 2007, 8 pages.
	M. Horchani (UJF), D. Fréard, B. Caron, E. Jamet, L. Nigay (UJF), F. Panaget Dialogue Strategies and Multimodal Presentation: a Dedicated Software Component and its Usage for Wizard of Oz Experiments Conference IHM 2007, The Nineteenth National French conference on Human-Computer-Interaction, ACM Press, France, November 13-15, 2007, 8 pages.

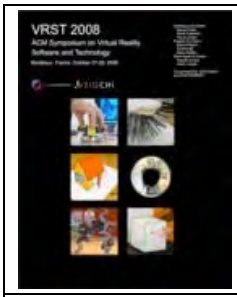
	T-J. Chin, Y. You, C. Coutrix (UJF), J-H. Lim, J-P. Chevallet and L. Nigay (UJF) Snap2Play: A Mixed-Reality Game based on Scene Identification Conference MMM 2008, The 14th International IEEE and ACM Multimedia Modeling Conference, Advances in Multimedia Modeling, Springer-Verlag Berlin Heidelberg, LNCS (Lecture Notes in Computer Science), Volume 4903/2008, Japan, January 9-11, 2008, pp. 220-229.
	M. Serrano (UJF), L. Nigay (UJF), J-Y L. Lawson (UCL), A. Ramsay (UGLA), R. Murray-Smith (UGLA), S. Denef (FIT) The OpenInterface framework: a tool for multimodal interaction Conference ACM CHI 2008, Conference on Human Factors in Computing Systems, ACM Press, Italy, April 5-10, 2008, Adjunct Proceedings, pp. 3501-3506.
	R. Murray-Smith (UGLA), J. Williamson (UGLA), S. Hughes (UGLA), T. Quaade Stane: Synthesized Textures for Tactile Input Conference ACM CHI 2008, Conference on Human Factors in Computing Systems, ACM Press, Italy, April 5-10, 2008, pp. 1299-1302.
	R. Murray-Smith (UGLA), J. Williamson (UGLA), S. Hughes (UGLA), T. Quaade (UGLA), S. Strachan Rub the Stane Conference ACM CHI 2008, Conference on Human Factors in Computing Systems, ACM Press, Italy, April 5-10, 2008, Extended Abstracts, pp. 2355-2360.
	C. Coutrix (UJF), L. Nigay (UJF) Balancing Physical and Digital Properties in Mixed Objects Conference ACM AVI 2008, Conference on Advanced visual interfaces, ACM Press, Italy, May 28-30, 2008, pp. 305-308.
	M. Serrano (UJF), D. Juras (UJF), M. Ortega (UJF), L. Nigay (UJF) OIDE: a tool for the design and development of multimodal interfaces Conference UBIMOB 2008, The 4th French-speaking Conference on Mobility and Ubiquity Computing, ACM Press, France, May 28-30, 2008, pp. 91-92.

	F. Jourde (UJF), Y. Laurillau, A. Moran, L. Nigay (UJF) Towards Specifying Multimodal Collaborative User Interfaces: A Comparison of Collaboration Notations Conference DSV-IS 2008, The XVth International Workshop on Design, Specification and Verification of Interactive Systems, Springer-Verlag Berlin Heidelberg, LNCS (Lecture Notes in Computer Science), Volume 5136/2008, Canada, July 16-18, 2008, pp. 281-286.

Year 3:

	F. Jourde (UJF), Y. Laurillau, A. Moran, L. Nigay (UJF) Specification of Multimodal Collaborative Systems: A comparative study of notations Conference IHM 2008, The 20th International Conference of the Association Francophone d'Interaction Homme-Machine, ACM Press, France, September 2-5, 2008, 8 pages.
HAID 08 Adjunct proceedings	D.K. McGookin (UGLA), S. A. Brewster (UGLA) Pressure Player: Combined Pressure and Audio Interaction HAID 2008, Haptic and Audio Interaction Design, Third International Workshop, Finland, September 15-16, 2008. Adjunct Proceedings.
	J.-Y. L. Lawson, (UCL), J. Vanderdonckt, B. Macq (UCL) Rapid Prototyping of Multimodal Interactive Applications Based on Off-The-Shelf Heterogeneous Components Conference ACM UIST 2008, The 21st annual ACM Symposium on User Interface Software and Technology, USA, October 19-22, 2008, Adjunct Proceedings, 2 pages. http://www.acm.org/uist/archive/
	M. Coiana (ARC), A. Conconi (TXT), L. Nigay (UJF) and M. Ortega (UJF) Test-Bed for Multimodal Games on Mobile Devices Conference Fun&Games 2008, International Conference on Fun and Games, Springer-Verlag Berlin Heidelberg, LNCS (Lecture Notes in Computer Science), Volume 5294/2008, the Netherlands, October 20-21, 2008, pp. 75-87.

	R. Murray-Smith (UGLA), S. Strachan Rotational Dynamics for Design of Bidirectional Feedback during Manual Interaction Conference Fun&Games 2008, International Conference on Fun and Games, Springer-Verlag Berlin Heidelberg, LNCS (Lecture Notes in Computer Science), Volume 5294/2008, the Netherlands, October 20-21, 2008, pp. 1-10.
	M. Serrano (UJF), D. Juras (UJF), L. Nigay (UJF) A Three Dimensional Characterization Space of Software Components for Rapidly Developing Multimodal Interfaces Conference ICMI 2008, The Tenth International Conference on Multimodal Interfaces, ACM Press, Greece, October 20-22, 2008, pp. 149-156.
	D. Juras (UJF), L. Nigay (UJF), M. Ortega (UJF), M. Serrano (UJF) Multimodal Slideshow: Demonstration of the OpenInterface Interaction Development Environment Conference ICMI 2008, The Tenth International Conference on Multimodal Interfaces, ACM Press, Greece, October 20-22, 2008, pp. 193-194.
	D. McGookin, (UGLA), S.A. Brewster (UGLA), W. Jiang (UGLA) Investigating Touchscreen Accessibility for People with Visual Impairments Conference NordiCHI 2008, The 5th Nordic Conference on Human-Computer Interaction: Building Bridges, ACM Press, Sweden, October 20-22, 2008, pp. 298-307.
	S. Denef (FIT), L. Ramirez (FIT), T. Dyrks (FIT), T. Schwartz (FIT), Ahmad-Amr Al-Akkad (FIT) Participatory Design Workshops to Evaluate Multimodal Applications Conference NordiCHI 2008, The 5th Nordic Conference on Human-Computer Interaction: Building Bridges, ACM Press, Sweden, October 20-22, 2008, pp. 459-462.
	S. Strachan, R. Murray-Smith (UGLA) GeoPoke: rotational mechanical systems metaphor for embodied geosocial interaction Conference NordiCHI 2008, The 5th Nordic Conference on Human-Computer Interaction: Building Bridges, ACM Press, Sweden, October 20-22, 2008, pp. 459-462.

	J-B de la Rivière (IMM), C. Kervégant (IMM), E. Orvain (IMM), N. Dittlo (IMM) CubTile: a multi-touch cubic interface Conference VRST 2008, the 2008 ACM symposium on Virtual reality software and technology, ACM Press, France, October 27-29, 2008, pp. 69-72.
	O. Vybornova, (UCL) H. Mendonça (UCL), L. Lawson (UCL), B. Macq (UCL) High level data fusion on a multimodal interactive applications platform Conference ISM 2008, the Tenth IEEE International Symposium on Multimedia, IEEE Computer Society, USA, December 15-17, 2008, pp. 493-494.
	S. Kieffer, J.-Y. L. Lawson (UCL), B. Macq (UCL) User-Centered Design and Fast Prototyping of an Ambient Assisted Living Solution for Elderly People Conference ITNG 2009, The 6th International Conference on Information Technology: New Generations, IEEE Computer Society, USA, April 27-29, 2009, pp. 1220-1225.
	S. Strachan, R. Murray-Smith (UGLA) Bearing-based selection in mobile spatial interaction Personal and Ubiquitous Computing Journal, Special Issue: Mobile Spatial Interaction, Springer-Verlag Berlin Heidelberg, Vol. 13, No. 4, May 2009.
	S. Strachan, R. Murray-Smith (UGLA) Nonvisual, distal tracking of mobile remote agents in geosocial interaction Conference LoCA 2009, The 4th International Symposium on Location and Context Awareness, Springer-Verlag Berlin Heidelberg, LNCS (Lecture Notes in Computer Science), Volume 55561/2009, Japan, May 11-14, 2009, pp. 88-102.
	E. Dubois , P. Gray (GLA) and L. Nigay (UJF) (co-editors): The engineering of Mixed Reality Systems Book, Human-Computer Interaction Series, Springer-Verlag. To appear Autumn 2009.

	C. Coutrix (UJF) and L. Nigay (UJF) An Integrating Framework for Mixed Systems Chapter 1 in The Engineering of Mixed Reality Systems, Human-Computer Interaction Series, Springer-Verlag. To appear Autumn 2009.
	J.-Y. L. Lawson (UCL), A. Al-Akkad, J. Vanderdonckt, B. Macq (UCL) An Open Source Workbench for Prototyping Multimodal Interactions Based on Off-The-Shelf Heterogeneous Components Conference EIS 2009, The First ACM SIGCHI Symposium on Engineering Interactive Computing Systems, ACM Press, USA, July 14–17, 2009, pp. 245-254.
	M. Ortega (UJF) and L. Nigay (UJF) AirMice: Finger Gesture for 2D and 3D Interaction Conference INTERACT 2009, The 12th IFIP TC13 Conference in Human-Computer Interaction, Springer-Verlag Berlin Heidelberg, LNCS (Lecture Notes in Computer Science), Sweden, August 26-28, 2009, 14 pages. To appear.
	R. Murray-Smith(UGLA) Empowering people rather than connecting them International Journal of Mobile Human Computer Interaction, IGI Publishing, Vol. 1, Issue 3, 2009, 18 pages.