# *Project Deliverable Report*

# D6.2 – Final Report

| | |
|---|---|
| **Security (distribution level)** | *Restricted* |
| **Contractual date of delivery** | *31 August, 2003* |
| **Actual date of delivery** | *24 September, 2003* |
| **Deliverable number** | D6.2 |
| **Deliverable name** | Final Report |
| **Type** | *Report* |
| **Status & version** | *Final, v9* |
| **Number of pages** | *63* |
| **WP contributing to the deliverable** | *WP6 Management* |
| **WP / Task responsible** | *SAGE (T6.2)* |
| **Other contributors** | |
| **Author(s)** | *Rachel Pick (RNIB), Paul Schmidt (IAI), Ilesh Dattani (IT Innovation), Cristina Arana, Ignacio Recio (SAGE)* |
| **EC Project Officer** | *Ms Evangelia Markidou* |
| **Keywords** | *Final report, IMAGINE summary, achievements,* |
| **Abstract (for dissemination)** | *Conclusions of the project. All the results achieved during the project are gathered and summarised in this report. A summary of the administrative data is also included.* |

# Executive Summary

IMAGINE project aimed to design and develop software technology that allows the interaction with electronic business applications by using a multi-lingual natural language interface from mobile devices and other appliances. The full potential given by electronic business-to-consumer markets can be made available to a broader audience by providing e-commerce on mobile phone.

The project is based on the results of a previous project that already delivered the basic technology, a sophisticated software architecture and an advanced natural language system for all the languages that will be dealt with in this project. The realisation requires improvements and adaptations of linguistic modules, adaptation of application modelling, integration with voice recognition, integration with e-business applications and testing and validation.

This report describes the results obtained at the end of the project, which are summarised next:

The project has created two products: A voice-enabled natural interface to two specific e-business applications, for an English and a Spanish e-business site. The interface includes an experimental German module. In addition, the project has developed some tools to facilitate the process of knowledge acquisition and adaptation of IMAGINE to new applications.

The project has been carried out with the participation of six European companies, four of them involved in the development (Software AG España, the University of Southampton/ T Innovation, Gesellschaft zur Förderung der Angewandten Informationsforshung e.V. and Teléfonica Investigación y Desarrollo) and two users (Royal National Institute of the Blind, Iniciativas Digital Media).

The report provides an overview of:

- General project results including prototypes, run time details and developer tools.
- A summary of the work completed per work-package and task.
- Administrative information including resources usage, results obtained in terms of deliverables, etc.

The report is intended for internal use within the Consortium and the European Commission.

# Table of Contents

# Figures and Tables

# 1. Introduction

## 1.1 *IMAGINE* Project

The major players in the field of information technology share the vision of using natural language to interact with computers and other telecommunication devices. Natural language access to the Internet, information systems, and computer software will become even more essential if the benefits of e-commerce and the internet are to become available to the whole society and not just to some privileged parts of it. The Internet is changing many areas of life. Set top boxes and other devices are taking the Internet into the homes of ordinary people via TV-cable, satellite and digital media and will enable them to experience the potential of e-commerce.

The ***IMAGINE*** project aims to provide a new means to interface with e-business applications, by providing a voice driven interface, and thus making them more accessible to the user. The solution provided by ***IMAGINE*** requires the correct integration of different key technologies e.g. speech recognition, natural language understanding, and a solid know-how of e-business architectures and solutions.

The project started from the technological basis established in a previous project, ***MELISSA*** (ESPRIT 22252), and built on this through the expertise of the key partners in the essential supporting technologies.

### Project Objectives

The ultimate objective is to exploit the full potential of electronic business-to-consumer markets through a broader audience by providing e-commerce through a speech driven interface on a variety of devices. This objective has been approached through the design and development of software technology that allows the interaction with electronic business applications by providing a multi-lingual natural language interface for mobile devices and other appliances.

### Expected Results

The project has created two products, a voice-enabled natural interface to two specific e-business applications for an English and a Spanish e-business site. In addition, the project developed some tools to facilitate the process of knowledge acquisition and adaptation of ***IMAGINE*** to new applications.

### Benefits

***IMAGINE***'s goal is to facilitate e-commerce by providing a voice driven interface and accessibility via a variety of devices e.g. mobile phones. It will thus enable large user groups not familiar with computers and computer technology to benefit from e-commerce opportunities.

The full potential of e-commerce is a long way from being realised. In this context, ***IMAGINE*** will contribute to make e-commerce:

- *more accessible*, in the way that everyone can use the device that is more convenient,
- *easier to use*: users simply say what they want without having to learn to use new technologies,
- *affordable* as it becomes available through telephone, there is no need for buying a computer.

## 1.2 Situation

After two years working in IMAGINE project, this report summarises the work and achievements get from the project. Consequently, the report outlines the results obtained from the different project tasks, and special attention is paid to describe the overall results of the project as a whole, both in terms of software and supporting documentation.

All the partners responsible of one task have contributed to this document providing the summary of the task that they are responsible of. The coordinator, has translated and gathered the rest of the information to create the administrative and general section of the document.

## 1.3 Purpose

The objective of this report is to compile all the results of the IMAGINE project. It can serve also as index to find other documentation where concrete tasks have been more detailed or completely developed.

It has three different points of view according to the possible interest of the readers:

- From a non technical point of view, the first section **IMAGINE products**, gives an good idea of what is IMAGINE and what is the concrete stuff result of the project.
- A second section **Result and achievements**, is more devoted to technical and interested parties in the project, explaining task by task what are the concrete achievements and where to find the complete results within the project documentation.
- Finally, the **Management information** is more addressed to the commission readers or even to other technical readers, in order to know what are the more consuming tasks, what problems the consortium run into, etc.

## 1.4 Overview

The contents of this document are organised in three main sections: :

***Chapter 2. IMAGINE products*** gives a general overview of the achievements in ***IMAGINE.*** It is structured in the following way:

- Overall goals: summary of IMAGINE goals.
- The demonstrator: practical cases tackled during the IMAGINE project.
- How it works: high level description of the IMAGINE system.
- Developer subsystem: explanation of the maintenance tool for getting the information for a new application.
- Other results: partner general experience of the IMAGINE project.

***Chapter 3. Results and achievements*** details task by task what has been the works done during the project:

- WP1 Preparation: previous to the development phase, it is necessary to state the user requirements and review the MELISSA architecture for the adaptations required to e-commerce and speech input and output. Then, the architectural design can be tackled.
- WP2 Natural Language Interface: design and development of the components close of the natural language management, either written or spoken: linguistic processor, semantic analysis, dialog, English Speech tool and Spanish speech tool.
- WP3 e-commerce apps integration: design and development of the components related with the interfaced application: knowledge database and its maintenance, front end and answer adapter.
- WP4 Pilots validation: task related with the real application of IMAGINE to the two pilot sites: integration reports, validation activities and usability studies.

- WP5 Exploitation & dissemination: the final objective of IMAGINE is to have a pre-commercial product. In order to have this product well positioned for market adequate market studies, dissemination and exploitation plans have be made.
- WP6 Management: the correct administration of the project requires some material, document, communication areas (Web site), and in general resources that are developed in this work package

***Chapter 4. Management information*** aims to give a summary of the administrative details for the complete project, namely:

- Statement of progress: summary of the project evolution according to the plan.
- Resources usage: summary of resources consumed resources.
- Deliverables summary: complete list of deliverables.
- Problems encountered: summary of the main problems found during the project life cycle.
- Project meetings: list of consortium meetings.
- Updated Project Snapshot: make a short card with the definition of the project.

# 2. IMAGINE Products

## 2.1 Overall goals

From the technical annex, we recover the original objectives for ***IMAGINE***: "***IMAGINE*** *will provide easy access to e-commerce by mobile phone and other appliances through a NL interface*". The objectives are summed up in the following goals:

### 2.1.1 Integration of speech with a NL-Understanding System

The project expected to integrate speech with language understanding. The speech module for ***IMAGINE*** would have to satisfy a range of market and user requirements:

- Accept continuous natural language. Continuous natural language has well known and documented characteristics in comparison to for example isolated work speech. Most current front-end products support continuous speech with some greater error rate than for isolated words.
- Require minimal training to be operational, although training is possible for the mobile and Internet based applications (personal access or access devices). If many hours of training are required the users will not persevere with the device.
- Be operational in multiple languages. The languages of immediate choice are those represented in the consortium (Spanish and English).
- Be lightweight in the client device so that existing mobiles can accommodate the technology.

In the practice the IMAGINE project has integrated two speech systems with Natural Language processing mechanisms. An explanation of this integration is provided at section 2.3, and a more detailed explanation about how to integrate the speech machinery is detailed at section 2.4.3.

### 2.1.2 Naturally interfacing e-business applications

The main project objective was to facilitate the access and usage of e-business applications. From a technical perspective this is translated into the following goals:

- Analysis of the communication requirements for humans to get the data in terms of interactivity and dialogue acts analysed in the context in which humans move, and the knowledge that is tacitly assumed in contexts and thus be omitted in requests (elliptical and anaphoric interactions).
- On this basis, the system will be able to understand queries for given domains. It will cover the dialog and communication acts like: '*Give me the prices for carpets below 4000 $*', '*Upcoming special offers for next month*', '*Where can I find tea pots?*', '*Please, show me the price list for mobile phones*'.
- Provision of methods for accessing directly the right entry function-point of a specific application i.e. choosing the right entry point among several ones available in a Web site. This needs to be complemented with functionalities to manage the execution of several actions from a single sentence, simple standard disambiguation procedures, passing of parameters to the application, etc.

This goal has been approached through the cooperation and coordination of diverse modules (Linguistic processing, dialog handling, answer adapter module, etc) as it is explained in chapter 2.3 How it works.

### 2.1.3 Improvement and Extension of Available Language Processing Module

The improvement and the adaptation of the available language module were a major task.

- The linguistic resources had to be adapted to cover the sub-language relevant for doing e-commerce (such as ordering, asking for information, doing payment, seeking assurance, etc.). The NL system provides 'deep' understanding, i.e. it maps each utterance on a generic language-independent semantic representation that has been specified in a previous project in the wake of a trend towards 'flat' semantic representations.
- 'Robust' processing has to be provided by integrating a NL parser that is able to provide a shallow analysis for unlimited text into the foreseen deep understanding system. So, it is always possible to

provide a basic understanding of NL input even if the NL understanding system is not able to provide a full semantic representation.

- User acceptance had to be improved by introducing discourse-based elements:
- Covering sentences with ellipsis and anaphora referencing previous discourse elements.
- Base linguistic coverage on dialogue modelling.
- Base linguistic coverage on a concept of user sub-language.
- Integrate modelling of illocutionary force.
- Linguistic resources had to be adapted to e-commerce model by efficient integration of lexical material, conceptual ontological modelling of commercial domain.

As result, the project counts with specific Linguistic modules for English, Spanish and German, which after refinement in the different lines above explained, are prepared for the purpose of interfacing software applications, allowing a quicker adaptation to a new application.

### 2.1.4 Interfacing Specific Applications

In ***IMAGINE,*** it was expected to apply this technology to interface two pilot e-commerce applications for two different users, the Royal National Institute of the Blind (RNIB), and Viapolis (IDM). Both pilots would provide e-commerce with a natural language interface. Both users would take benefit from the integration of the voice-enabled Natural Language Interface to their e-business application.

The application chosen by the Royal National Institute of the Blind (RNIB) is the **RNIB Online Shop** – an e-commerce site that allows RNIB customers to browse for and purchase RNIB products via the Internet.

The **Viapolis** application is an Internet site that allows users to find out what's going on where and when in Spain. It is a valuable resource for residents of Spain and visitors alike to find restaurants, cinemas, etc.

The potential that this solution could bring to end users, citizens and disabled people, is enormous.

### 2.1.5 Generalisation

***IMAGINE*** has been designed with the aim of providing an environment allowing developers of e-commerce applications a general solution for creating Natural Language interfacing to e-business applications.

This objective is managed thanks to a modular architecture, relying on an ontology model, which makes the modules behaviour independent from the concrete data of an application. Additionally, the project provides tools to developers, allowing them an easy integration and configuration of ***IMAGINE*** with their own applications, by using different mechanisms (applets, JBeans, IIOP, DCOM, …)

### 2.1.6 Exploitation

From a market perspective, the objectives of ***IMAGINE*** were:

1. To validate the results at the pilot sites. A user-centred methodology has been applied for the various project stages. The pilot applications used as demonstration of the technology in dissemination activities. It was expected that a close relationship with users provide the best recipes to follow in order to deliver a useful and widely adopted tool.
2. To comply as much as possible with open software and non-proprietary tools, with the focus on the widest and simplest possible distribution. This will help open the target market for ***IMAGINE***.
3. To build a set of tools to facilitate the integration of further applications, that would open the market of third party products for ***IMAGINE***, to be included easily in new e-commerce application.

The project has concluded with an exploitation plan, which depicts the approach for a further exploitation of the project results.

## 2.2 The demonstrator

The project has concluded with the integration of the Voice Natural Language Interface with two e-Business applications:

- Access to the RNIB Online Shop Web site.

- Access to the Viapolis Wap city guide.

With the aim of proving the concept of an easy to integrate interface, and as an step to dissemination, the project has concluded with two additional prototypes:

- Access to the ***IMAGINE*** home site.
- The creation of a customisable package ready to be downloaded and adapted for easy Web applications similar to the ***IMAGINE*** Web site.

Following table shows a comparative summary of the prototypes characteristics:

| | Speech input | Written input | Access protocol | Functionalities covered | Model size (in LF) |
|---|---|---|---|---|---|
| **RNIB - Online Shop** | English | English | Http/HTML<br>Https/HTML | 26 | 65 |
| **Viapolis - City guide** | Spanish | Spanish | Http/WML | 11 | 20 |
| **IMAGINE Web site** | English | English<br>Spanish<br>German | Http/HTML | 9 | 18 |
| **Dummy application** | - | English<br>Spanish<br>German | Http/HTML<br>Https/HTML<br>Http/WML<br>Https/WML | - | 9[1] |

*Fig 2.2.1 - Comparative look up table between **IMAGINE** prototypes.*

### 2.2.1 The Online Shop

The application chosen by the Royal National Institute of the Blind (RNIB) is the **RNIB Online Shop**, an e-commerce site that allows RNIB customers to browse for and purchase RNIB products via the Internet. It is a very new site and was launched in March 2002.

The adaptation of ***IMAGINE*** to this site, is the main outcome of the ***IMAGINE*** project. Most of the efforts have been devoted to this prototype for various reasons:

- The Online Shop is a real working system;
- The Online Shop is a very good representative of an e-commerce application.
- The English spoken market is a very promising area in world wide market.
- The great interest from the very beginning in the project from RNIB partners.

As a result of these efforts, a very refined prototype has been delivered. Sixteen versions have culminated in the production of a very evolved prototype covering a high percentage of the Online Shop functionality. *D23 – Dialog and Interaction model* gives a detailed idea about the model supporting the Online Shop system.

The system can support most of the functionality of the visual Online Shop. In the next picture, a dialogue between the user and ***IMAGINE*** can be seen, along with the corresponding Online Shop screen.

**User >** my basket please

[1] Basic functionality: Help, Quit, Stop, etc

**IMAGINE >** What do you mean? 1 Add to Basket or 2 Show the basket or 3 Remove products from the basket Please choose one of them by saying its number or stop if you don't want any of them

**User >** two

**IMAGINE >** You have 1 type of product in your basket. Product code DK31 is a - Liquid level indicator, yellow which costs £8.04 each. Your basket contains 2 of them, therefore the cost for this is £16.07. To summarise, the products you have selected costs £16.07 altogether. A handling fee of £2.50 is added. This makes a total sum of £18.57 Would you like to update the quantities in your basket, remove products from your basket, browse for products, checkout or change your delivery address for this order?

*Fig 2.2.1.1 - Dialog to get the basket contents through IMAGINE.*


## 2.2.2 The city guide –Viapolis

The **Viapolis** application is an Internet site that allows users to find out what's going on where and when in Spain. It is a valuable resource for residents of Spain and visitors alike to find restaurants, cinemas, etc. and the WAP version can directly allow the user to get in contact with, say, the restaurant of their choice, for example, to book a table.

Given the wide scope of the entire Viapolis application, the objective of the project is to provide a voice enabled interface to retrieve information for a concrete selected city in Spain.

Unfortunately, during the project time, Viapolis has finished their commercial activities and the co-owner Iniciativas Digital Media take the role of Viapolis in the project. As the system is no longer active for external user but most of the work for model the system has been done, it was decided to maintain a test system, with non updated data.

The interest for ***IMAGINE*** is high in this system also, because it is representative of a huge set of entertainment, leisure, etc sites. The difficulties with a so volatile information, teach the consortium for managing this information through different mechanism that a previously known information.

**User >** Actividades para hoy

**IMAGINE >** de entre las siguientes opciones elija una de ellas diciendo sunumero: 1 Eventos por fecha y categoria o 2 Eventos por fecha

**User >** 2

**IMAGINE >** 1 - Sbarro, cocina rápida italiana; 2 - Opaqo, sabores exóticos en Ciutat Vella; 3 - ¿Dónde quedamos?; 4 - Glamour y modernidad en el Schilling; 5 - Solidaridad y compromiso en La Cerería; 6 - Más;

*Fig 2.2.2.1 - Activities for today through IMAGINE.*

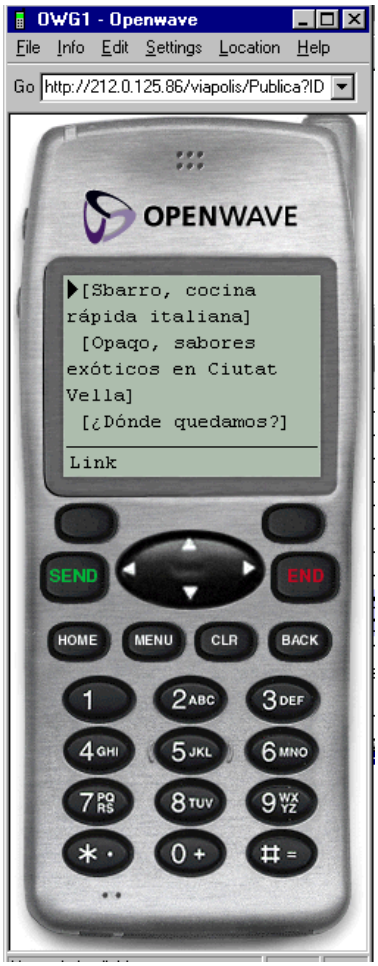



*Fig 2.2.2.2 - Viapolis Wap access.*

### 2.2.3 IMAGINE Web Site

The access to the ***IMAGINE*** Web site is part of the dissemination activities, although it has not been until the end of the project, when it has been included as a third prototype. The learning during the previous months with the other prototypes, make that the model of this prototype has been very quick designed and supported. Only the English will support the speech access to the site.

In the example, a German query and response is got from the ***IMAGINE*** site. Also the ***IMAGINE*** site hard copy shows the corresponding German version of the site

**IMAGINE >** Dieses System kann Ihnen über die folgenden Projekte erklären: IMAGINE, OPAL, ALFANET, COVAX, MELISSA and EFIS. Welches Projekt werden Ihnen an interessiert?

**User >** Ich moechte gerne etwas ueber die Projektziele IMAGINE [2]

**IMAGINE >** IMAGINE will das gesamte Potential des elektronischen Handels (Business-to-Consumer, B2C) ausschöpfen durch die Erschliessung neuer Käuferschichten. Dieses Ziel soll durch die Entwicklung neuer Software erreicht werden, die die Kommunikation mit elektronischen Handelsplattformen mit Hilfe eines multilingualen natürlich-sprachlichen Interfaces ermöglicht, das auf anderen Geräten als dem Computer, also z.B. Handys, zur Verfügung gestellt wird. Dieses Ziel soll erreicht werde, indem ein multilinguales natürlichsprachliches Interface geschaffen wird, das es dem Nutzer erlaubt, per Telefon auf E-Commerce-Seiten zuzugreifen.

*Fig 2.2.3.1 - IMAGINE Project objectives asked through IMAGINE.*


*Fig 2.2.3.2 - German version of the IMAGINE Web site.*

[2] The default project is IMAGINE. Also the previous prompt, the so-called Welcome prompt, can be easily changed to conduct the user inputs to known functionalities.

### 2.2.4 Dummy application

As an extra result of the third prototype, the model for it has been solved in a general way in such a way that it can be applied to other static sites with similar structure to ***IMAGINE***. The structure is a simple page with concrete sections that can be directly addressed by the user through their name as part of a natural language sentence.

The basic stuff in the free package includes all the software required to make an access in different languages (English, Spanish or German) to a site with the stated characteristics. No speech input mode is provided with this but written mode. The complexity of the hardware and software required for a new application with ***IMAGINE*** make very difficult to give by free anything really useful.

Anyhow, the detailed interfaces for the connection of external speech tools is provided also for making easy the adaptation to ***IMAGINE*** of another speech tool.

The summary of updates required in this free package to access to a new Web suite are summarised in the following table:

| | **Provided** | **Adaptation** |
|---|---|---|
| **Speech tool** | No | The detailed RMI interface is published. The access is easy through RMI implementing the interfaces included in the free package. |
| **Dialog** | Yes | Not required |
| **Linguistic Parser** | Spanish, English | It requires to update a concrete set of files with the keywords to state in the sentences referring to the different pages and section in the pages. The example for ***IMAGINE*** Web site is included |
| **Semantic Module** | Yes | Requires the update of the KNM. This will require to know how to model and application. A detailed explanation will be included in the package and also the example for accessing ***IMAGINE***. The customisation will be made in the XML files included, although probably could be made using the own developer tool available in the members are of the ***IMAGINE*** Web site. |
| **Answer adapter Module** | Yes | It requires the developing of XSL scripts for each of the functionality supported by the ***IMAGINE*** installation. The scripts used for ***IMAGINE*** will be delivered as example. |

*Fig 2.2.4.1- Dummy application resources provided.*

## 2.3 How it works

The detailed ***IMAGINE*** architecture is the focus of the deliverable *D1.3 – Functional Specifications & Validation Plan*, although a summary is given in this section.

The ***IMAGINE*** voice enabled interface is a novel solution constructed upon the integration of key technologies for voice processing, natural language understanding, dialog handling, Web interfacing, etc.

The below diagram depicts the basic architecture of the ***IMAGINE*** solution as a three layered service. At the top level **Mobile device and voice integration,** are the modules in charge of the direct interaction with the user. The two speech tools will support the two involved languages in the system. The English Speech Tool is based on a Nuance solution and follows a grammar-based approach; the Spanish Speech Tool is provided by Telefónica I+D, member of the consortium, and corresponds with a stochastic paradigm (linguistic model), with concrete grammars included for concrete purposes.

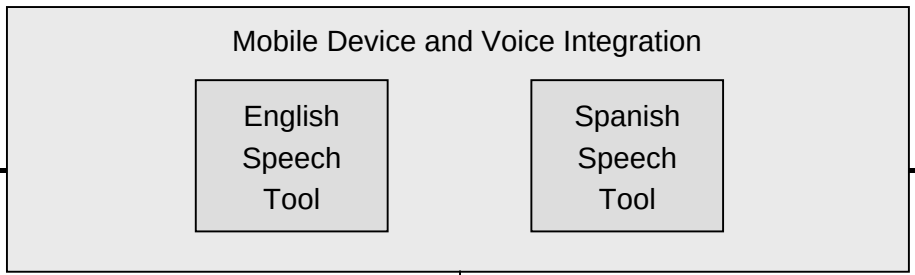


*Fig 2.3.1 - IMAGINE functional components.*

At the middle layer are those modules in charge of controlling the interaction with the user (dialog interaction, natural language processing and the generation of the output corresponding with the user request):

- **The Linguistic Module** is in charge of translating the user input transformed in written mode to a set of semantic facts, that represent the user utterance in an independent reference model, and from this model translate to the concrete application meaning.
- The **Dialog Handling Module** is the core of the ***IMAGINE*** system, every action happening in the system is orchestrated by the dialog. Its objective is to determine the functionality that the user is referring to, and then, to complete all the parameters related with the concrete request. Once the understanding is complete, the Executive Module is called for getting the concrete response.
- **The Executive Module** is the Back-End of ***IMAGINE***. It takes control of minor details in communication activities through the real media (by now http/https with HTML and Http with WML) It also recovers the results of the "out world", and translates them to terms of user understanding, i.e. sentences.

Finally, in the third level, the **Ontology e-Business Model** that represents the application to be interfaced in terms of a general formalism, not dependent on the current application interfaced but able to support new applications.

## 2.4 Developer subsystem

### 2.4.1 Methodology for integrating IMAGINE

IMAGINE is designed in such a way that it can be adapted to multiple applications that are already in existence, or that are in the development phase, if they are "reasonably manageable" with natural language either in written or speech model. If it is considered feasible to use IMAGINE with the application, then the rest of the phases are followed in order to get the adequate model of the application that IMAGINE can use to interface it.

The complete process is the following:

1. **Initial Analysis:** general analysis of the application in order to see if the NL integration with the application is feasible, before going on to a more detailed study. Applications with a mainly graphical interchange of information with users are not suitable to be interfaced via speech mode.
2. **User Needs Analysis:** determines the user requirements that will be the basis for the integration activities.

3. **Selection of Integration Solution:** analyses the application features and decides the way of integrating IMAGINE with the application, giving software and communication guidelines. Typical methods are HTTP, HTTPS, RMI, Web Services, etc
4. **Acquisition of the Application Knowledge:** analyses the system to extract the set of functionalities that the application offers, their dependencies, etc. Main work for the KNM (Knowledge Management Module) maintenance tool. On this basis, the AAM (Answer Adapter Module) can define the ad hoc scripts that parse the application's responses.
5. **Acquisition of Linguistic Information:** acquires the linguistic information necessary to meet with the NL coverage requirements of the application. The LPM (Linguistic Processing Module) and the ST (Speech Tool) are the main components requiring this information.
6. **Acquisition of Semantic Application Knowledge:** enables results of the LPM to be mapped onto the correct function calls. The LPM output is analysed by the SAM (Semantic Analysis Module), translating it into terms understood by the application.
7. **Integration with the Application:** the implementation of the specific modules that are necessary for the system to function (FEM). This is carried out via HTTP/HTTPS protocols, which are the most common nowadays. With little effort, it can be adapted to new paradigms such as Web Services, etc. if required.
8. **Test & Validation:** A test process to verify that the interface fulfils all specified requirements and to guarantee the final quality of the interface and user satisfaction.
9. **Maintenance:** provides the methods to follow when the interfaced application changes its functionality or environment.

### 2.4.2 Knowledge Manager (KNM)

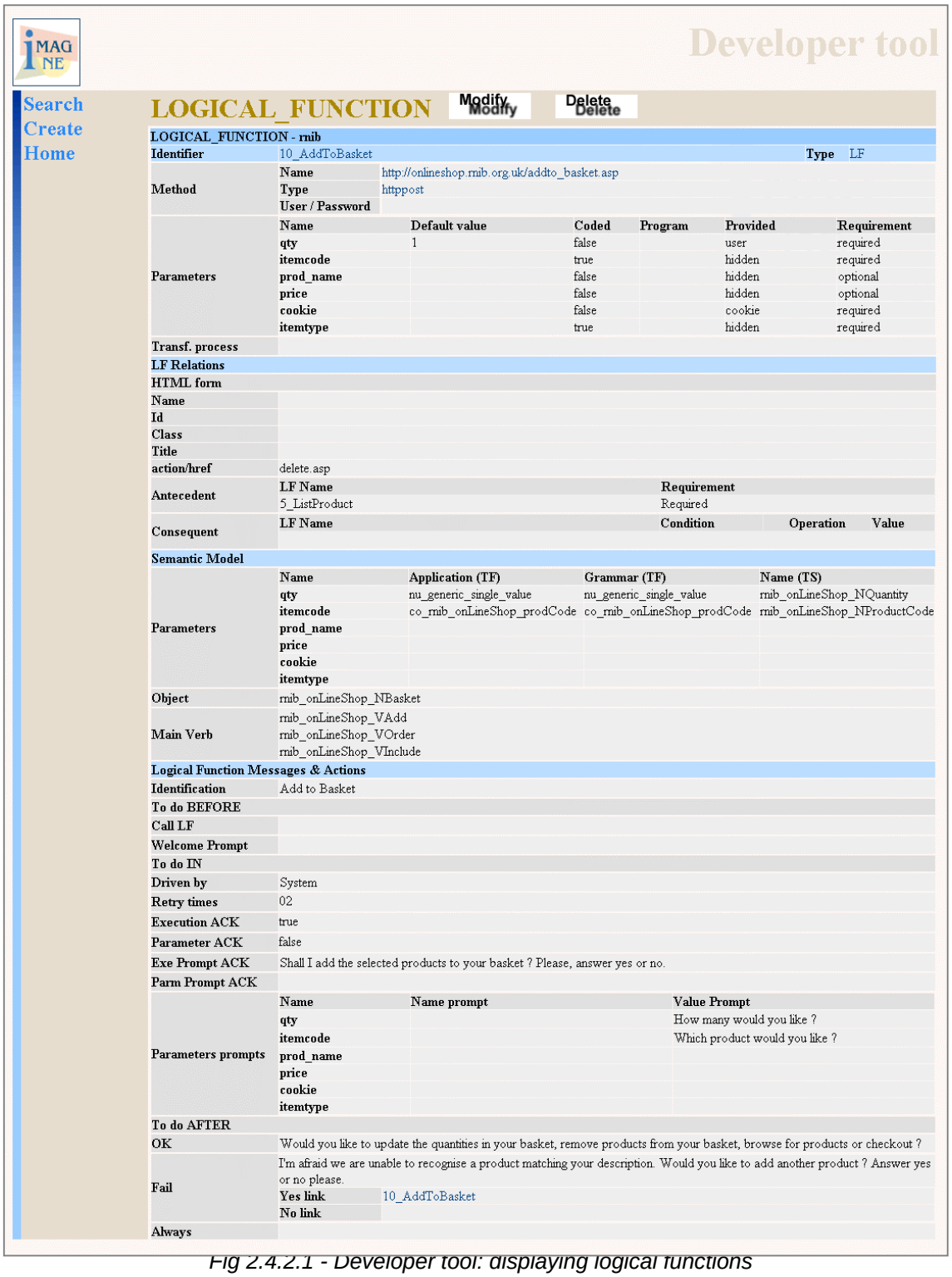
The core of ***IMAGINE*** is the dialogue. This component reads the model of the application to be interfaced and interprets the user utterances after being analysed by linguistic components. Those linguistic components translate the user sentence into semantic terms managed in the interfaced application. It is then when the dialogue manages the possible functionalities supported by the application, selecting among ambiguous functions and asking the user for non-provided parameters. An adequate maintenance of the KNM is therefore the key for the adaptation of ***IMAGINE*** to the application interfaced from a functional point of view. For this purpose, a developer tool has been provided, giving access to every detail related with this model.

A Logical Function represents a compact and concrete functionality. In the prototypes covered during the ***IMAGINE*** project time, example of Logical Function are: 10_AddToBasket, 5_ListProduct, 2_EventCategory, 5_directoryName, 6_RelatedLinks, etc All those LFs, can be directly found in the three different prototypes, and performed using the original interfaces provided by the applications.

The LFs contains at the same time various types of knowledge:

- the information about how to get the functionality in the application, i.e. in the case of Http/Https protocol, the concrete URI; the type of call (Get, Post,…),
- the parameters required for the method to call.
- the form that should be parsed to get details for next step in the LF execution process,
- the antecedents or consequents LFs that have relation with the current LF, or the relations with other steeps that all together compose the full functionality covered by the LF,
- the semantic model of the LF representing, this is based on the use of ***IMAGINE*** Type Formats and ***IMAGINE*** Type Synonyms,
- the dialog details like the prompts to ask the parameter values, the actions to do before or after the LF is executed, etc.

The main work with the KNM is done before ***IMAGINE*** starts to work with the interfaced application. Then, when the ***IMAGINE*** core starts, the full KNM is loaded in main memory, avoiding access to external systems, what imply a better performance rate. The work with the KNM, once the application has been modelled, is mainly to maintain the prompt texts, to tune the details of the parameter requirement, (values, requirement, etc), … There is another delicate task that is started with the first model of the application but requires a more refined work, it is the Semantic Model.



Developer tool

Search
Create
Home

## LOGICAL_FUNCTION

Modify Modify  Delete Delete

**LOGICAL_FUNCTION - rnib**

| | | | | | | |
|---|---|---|---|---|---|---|
| **Identifier** | 10_AddToBasket | | | | **Type** | LF |
| **Method** | **Name** | http://onlineshop.rnib.org.uk/addto_basket.asp | | | | |
| | **Type** | httppost | | | | |
| | **User / Password** | | | | | |
| **Parameters** | **Name** | **Default value** | **Coded** | **Program** | **Provided** | **Requirement** |
| | **qty** | 1 | false | | user | required |
| | **itemcode** | | true | | hidden | required |
| | **prod_name** | | false | | hidden | optional |
| | **price** | | false | | hidden | optional |
| | **cookie** | | false | | cookie | required |
| | **itemtype** | | true | | hidden | required |
| **Transf. process** | | | | | | |

**LF Relations**

| | | | | |
|---|---|---|---|---|
| **HTML form** | | | | |
| **Name** | | | | |
| **Id** | | | | |
| **Class** | | | | |
| **Title** | | | | |
| **action/href** | delete.asp | | | |
| **Antecedent** | **LF Name** | **Requirement** | | |
| | 5_ListProduct | Required | | |
| **Consequent** | **LF Name** | **Condition** | **Operation** | **Value** |

**Semantic Model**

| | | | | |
|---|---|---|---|---|
| **Parameters** | **Name** | **Application (TF)** | **Grammar (TF)** | **Name (TS)** |
| | **qty** | nu_generic_single_value | nu_generic_single_value | rnib_onLineShop_NQuantity |
| | **itemcode** | co_rnib_onLineShop_prodCode | co_rnib_onLineShop_prodCode | rnib_onLineShop_NProductCode |
| | **prod_name** | | | |
| | **price** | | | |
| | **cookie** | | | |
| | **itemtype** | | | |
| **Object** | rnib_onLineShop_NBasket | | | |
| **Main Verb** | rnib_onLineShop_VAdd<br>rnib_onLineShop_VOrder<br>rnib_onLineShop_VInclude | | | |

**Logical Function Messages & Actions**

| | | | |
|---|---|---|---|
| **Identification** | Add to Basket | | |
| **To do BEFORE** | | | |
| **Call LF** | | | |
| **Welcome Prompt** | | | |
| **To do IN** | | | |
| **Driven by** | System | | |
| **Retry times** | 02 | | |
| **Execution ACK** | true | | |
| **Parameter ACK** | false | | |
| **Exe Prompt ACK** | Shall I add the selected products to your basket ? Please, answer yes or no. | | |
| **Parm Prompt ACK** | | | |
| **Parameters prompts** | **Name** | **Name prompt** | **Value Prompt** |
| | **qty** | | How many would you like ? |
| | **itemcode** | | Which product would you like ? |
| | **prod_name** | | |
| | **price** | | |
| | **cookie** | | |
| | **itemtype** | | |
| **To do AFTER** | | | |
| **OK** | Would you like to update the quantities in your basket, remove products from your basket, browse for products or checkout ? | | |
| **Fail** | I'm afraid we are unable to recognise a product matching your description. Would you like to add another product ? Answer yes or no please. | | |
| | **Yes link** | 10_AddToBasket | |
| | **No link** | | |
| **Always** | | | |

*Fig 2.4.2.1 - Developer tool: displaying logical functions*

The Semantic Model represents a high level of representation of the application functionality. It supposes a medium step between the result of the Linguistic tools and how this Linguistic tools results could be understood, or lets say mapped, into the functionality of the application. The semantic model represents, for example, the mechanism to map the word basket, with the functionality, or functionalities that are related with the basket. In this case three LFs: 6_Basket, 7_RemoveFromBasket and 10_AddToBasket.

In order to see more details about Logical Functions and semantic model, it is recommended to read the documents D23 – Dialog and Interaction Model and D4.2 – Second prototype Integration Delivery Report.

### 2.4.3 The English Speech Tool (EST)

The English speech tool (EST) is responsible for connecting ***IMAGINE*** to a telephone line, interpreting voice input and synthesising speech output. Its role in the overall system is illustrated in the following diagram.

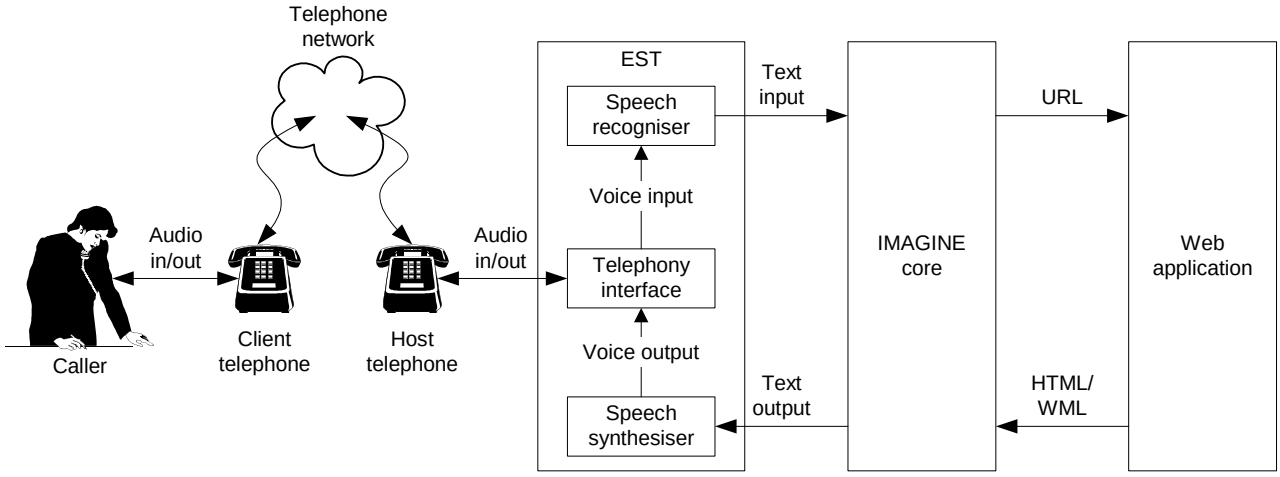



*Fig 2.4.3.1 – EST role in IMAGINE.*

**Telephony Interface**

The telephony interface connects the EST to an ordinary analogue telephone line via a Dialogic telephony card, i.e. the card acts as a telephone on the telephone network. When the module detects an incoming call, it answers the call by “picking up the telephone”, connects the audio input from the caller to the speech recogniser and the audio output to the caller to the speech synthesiser.

**Speech Recogniser**

The speech recogniser is responsible for converting spoken words in an audio signal into text. The module is built on a commercial speech recogniser. Recognition grammars have been developed to customise the recogniser such that it understands typical utterances that are relevant to the RNIB Online Shop application, e.g. “I want to place an order”. The module applies basic interpretation to the transcription to filter out superfluous words (e.g. “I want to um place an er order” is interpreted and converted into “I want to place an order” for output), simplify complex input (e.g. “Freddy f r e d d y” is interpreted as “Freddy”) and adds markup to complex types such as postcode and date (e.g. “s o fourteen seven n p” is interpreted as “<postcode>so14 7np</postcode>”). This approach to recognition improves the interpretation accuracy of the ***IMAGINE*** core as the linguistic modules within the core was primarily designed for written text which does not include speech acts and complex types in transcription form.
The implemented speech recogniser can interpret the following types of information as well as utterances that are relevant to a range of requests.

- Date, day, month, year
- Letter and spelling
- Credit card details including credit card number, expiry date, card holder name
- Telephone number, number, VAT registration number
- Title (e.g. Mr.), email address, password, gender, post code, house number, number
- Popular (top 95%) first names and surnames in the UK
- Product name, product code, product category and section names for RNIB products
- Request for login, product search, shopping, registration, changing stored details
- Request for help, changing speech output preference, hang up, transfer call

**Recognition grammar**

Utterances that are understood by the speech recogniser are defined in a recognition grammar. Experiments were conducted to assess several approaches to grammar writing. The first used hand crafted rules to define exact linguistic structures that can be transcribed by the recogniser. This approach

generated a large grammar that became increasingly difficult to maintain as the interaction between the different rules became intractable as the grammar grows. The second used a statistical language model that was automatically derived from example text. The method is easy to maintain as a novice to add new utterances to the grammar by providing example utterances. However, recognition accuracy and speed performance were poor as the method requires a vast amount of training data to obtain an accurate statistical model. The third adopted a hybrid approach where the recognition grammar was partitioned according to function (e.g. add product to basket, registration, etc.) and each function was described by a statistical language model. A proprietary grammar editing tool was developed to enable a novice user to modify the grammar. The result is a maintainable grammar with improved accuracy. However, the interaction between sub-grammars also became intractable as the grammar grows. The final approach used an proprietary grammar specification method that maps utterance fragments to complete sentences. The result is a highly flexible recognition grammar which has the best accuracy and speed performance. However, all maintenance has to be carried out by an expert grammar writer. This is the approach that is used in the final system.

**Speech Synthesiser**

The speech synthesiser is responsible for converting written text into spoken form. The module is build on a commercial speech synthesiser which is capable of generating speech in different voices, speed and volume. The synthesiser takes the text output from ***IMAGINE*** core, applies basic formatting (e.g. such that currency value is read out correctly) and generates the spoken words as an audio signal which is sent to the caller via the telephony interface.

**Internal Functions**

The EST has a selection of functions that are triggered by speech input and handled within the EST. First, the module analyses the transcription generated by the speech recogniser to determine whether the transcription is sufficiently accurate for output. If the confidence value is low, the EST will automatically prompt the caller to repeat or rephrase the last utterance for improved transcription accuracy.
Speech synthesiser preferences are also handled internally within the EST. The caller can ask the system to speaker louder, quieter, faster and slower by voice. The EST analyses the transcription to determine whether the request is relevant to speech synthesis control, if so, the request action is carried out internally within the EST.

The prompts associated with EST internal functions are defined in a proprietary prompt definition language which facilitates the use of randomly selected alternative prompts to add variety to the EST response.

## 2.5 Other results

### 2.5.1 Software AG

The ***IMAGINE*** project has been until now the latest EU funded project about Natural Languages interfaces in which SAGE has been involved. The current results gathers the experience of many years in the Natural Language interfaces arena and seems from SAGE point of view a very good result very near of a commercial product.

The collaboration among partners has been especially good with some of them, what helps a lot in the current results.

The use of XML, XSL, Java, RMI, as widely accepted standards have helped very much in the coordination of a very interconnected system, where the fail of only one component, gives the fail of the full system. From technical point of view, there are no doubts until now, that platforms, languages and paradigms selected have been the correct, and no problems with them has been stated.

It is considered that an extension of three months, based on the own non expended ***IMAGINE*** resources, probably would provided the indispensable feedback from end users to make the final tuning of the system. At the moment, this feedback will not be possible by the short remaining time between the end of prototype and the end of the project.

### 2.5.2 IT Innovation

The IT Innovation Centre is an autonomous industry-facing part of the Department of Electronics and Computer Science at the University of Southampton. Some of the core competencies include the ability to

architect, engineer and integrate innovative IT systems. We deliver strategies, road maps, proofs-of-concept, demonstrators and novel operational systems. IT Innovation has participated in 46 ESPRIT and IST projects over the past decade, managing 14 of them. The IT Innovation Centre employs world-class experts in their fields and most qualified to PhD level. Our facilities are dedicated solely to our own projects, including our computer systems, which are based on a secure LAN.

***IMAGINE*** represents our track record in the area of delivery e-commerce applications within a multimodal and multi-lingual infrastructure. We provide an interface for existing e-business applications so that can be accessed via wireless communication devices via voice. The key aim is: Interfacing Mobile Applications and Voice Natural Language Interactivity. IT Innovation provides the core functionality and implementation expertise for the development of the English Speech Tool and the integration for the English e-business application.

In IT Innovation our core competency in this area has been further enhanced by working with the other technology providers and software developers in the consortium and the alliance formed with Nuance communications. It also allowed us to further continue our collaborative work with IBM UK and the results of this work have already been exploited in terms of consultancy 'for the financial sector' on the basis of the knowledge and understanding acquired through our work in ***IMAGINE***.

We intend to follow up the work done in a variety of potential proposals under the current IST Framework VI work programme in the areas of cognitive systems, mobile and collaborative working and GRID based technologies.

### 2.5.3 IAI

IAI's major task was to design and develop the linguistic processing modules. The linguistic processing had to be developed for two languages and two different applications.

From a technological point of view, the most challenging aspect of the ***IMAGINE*** project was to cope with the output of a speech recognition system by having this output undergo a deep linguistic analysis the result of which being meant to be the input for an application specific interpretation by a knowledge based module. It was actually this hypothesis that drove the ***IMAGINE*** project as a whole, namely that it is feasible, beneficiary and innovative to exactly try this, the deep analysis of utterances and thereby achieve better results as available commercial software that is currently sold on the market place. Commercial systems in this area like those by Nuance or other vendors that are already installed in commercial applications like the United Airline's flight ticket booking system generally work with much simpler techniques like a combination of application specific knowledge and keyword spotting techniques that try to guess from the keywords that are spotted and the knowledge which step in the application is now required at this stage. They do not try to model a kind of (basic) understanding as it was done in ***IMAGINE***.

In betting on full analysis, IAI tried out two strategies. One was to use a full fledged NLP platform that provides a full typed feature unification system and to integrate a concept of chunk parsing into this. The other was to use a robust language checker based on a flat syntax analysis and enhance it in a way that it allows for the (robust) calculation of predicate argument structures that can be processed further by the application specific modules without giving away its robustness.

From a non-technical point of view the most interesting experience was to work together with potential users of this (emerging) technology. The work gave very useful insights into what the specific user requirements actually are and the conditions for the feasibility of establishing a service based on the technology developed in the ***IMAGINE*** project.

### 2.5.4 Telefónica I+D

***IMAGINE*** project has caused an important impact at TID, although we had a very limited participation in the project, then it has been a very rewarding experience working together with Technology Providers and specially with potential real Users. Thanks to this project, we have broadened our views, usually limited to Telefonica's internal market and known the problems of another users.
Following similar working lines to that used in IMAGINE project but more focused on our proprietary platform, that is able to manage scores of simultaneous calls and has its own Dialogue Manager, we will apply information management techniques to ongoing commercial services belonging to different companies at Telefónica Group. With the experience acquired in the project, we will also be able to update real services and creating new ones for our customers in Spain and Latin America.

We already have products based on Speech Technology working in the market and consider that Speech Technology will play a very important role in the near future with the developing of third generation mobile communications, where an important number of voice-based services could be launched.

### 2.5.5 Royal National Institute of the Blind

As lead partners in coordinating the provision of the user requirements for the development of the project's prototypes, RNIB provided information for their own system, and collated information from the other user partner and research from the development partners into the technical requirements for the systems, e.g. security issues.

The system that RNIB provided is a web based e-commerce application. This would be the base application to which a prototype IMAGINE interface could be applied. As owners of the RNIB Online Shop, a detailed technical description could be provided to the IMAGINE developers as part of the user requirements task. Along side the technical documentation, a full description of what was required from a voice interface was detailed as a set of use cases. In order to produce as comprehensive a corpus as possible for the system, telephone calls between RNIB Customer Services staff and RNIB customers were recorded and transcribed so that types of grammatical structures and vocabulary used by people ordering RNIB products were available. In addition to this, comprehensive lists of RNIB product names, product name synonyms, product category names, RNIB business customer names and others were provided.

For the purposes of validating the prototypes produced, RNIB led this stage by producing validation plans, which would be used by both user partners in the project. The evaluations themselves as an iterative process provided insights for both users and developers into the usability of these types of systems. They also provided insight into the viability of producing voice systems that interact directly with established web applications and the pros and cons of this approach.

By working with the developers to produce the interface to the requirements given in the first stages, and by carrying out usability evaluations on the system produced, far more knowledge of how voice applications work and develop has grown. The knowledge gained from this work is extremely useful to the RNIB in its research of voice systems. Research into improving technological access to information and services for visually impaired people is one of the main objectives of the iSys department at the RNIB, and the ever increasing interest in voice systems in this area means that any knowledge gained from this project will be invaluable to the RNIB's future efforts.

### 2.5.6 Iniciativas Digital Media

IDM is the digital company of a mass media communication group. As a communication company our goal is to give information to all the people. For this reason we think that the information that we produce has to be available when the people want. Is for this reason that we have to deliver the information in a wide range of channels (www, wap, sms, mms, telephone, tv, radio, etc…) IMAGINE is our first experience involved in voice natural language technology and we consider it very interesting to research and provide information by telephone in the future. Not everybody is connected by PC's to the Web but almost all the population have access and know how to use a telephone.

Although, Viapolis it's not running nowadays in a real context, IMAGINE technology allow us develop new products running in Web using voice natural language interactivity. This technology allows us to make new business in the future (classifieds ads, access to news information, newspaper history, …) As a users, to be involved in the IMAGINE project and to work with developers, help us to understand the wide range of possibilities that with voice natural language technologies can achieve.

# 3. Results and Achievements

## 3.1 WP1. Preparation

### 3.1.1 T11. Def of User & System requirements

This task culminated in the production of the D11 User & System Requirements document which became the reference for producing the Spanish and English prototypes for the systems provided.

Both user partners provided an application to be interfaced by an Imagine prototype – the RNIB providing the RNIB Online Shop, a web based ecommerce application, and Viapolis providing Viapolis.com, a WAP information service.

In order to complete the task, RNIB, as experts in the field, provided a list of general user requirements to be considered when developing any system, but tailored to highlight their application to voice based systems. The aim was to highlight the issues surrounding user acceptance and usability of systems. The topic of system acceptability was covered, identifying it as a measure of how well a system satisfies the needs and requirements of its users and other potential stakeholders i.e. service providers and intermediaries. Acceptability was shown to be mainly determined by a system's usability, among other things, being the degree to which a user can make use of the functionality of a system.

The topic of usability was discussed, pointing out the measures that can be used to determine how usable a system is – learnability, efficiency, memorability, errors and satisfaction, along with other contributory factors. A look at system acceptability from the user perspective was then provided, the analysis considering the three groups – end users, service providers and intermediaries – highlighting general characteristics and typical tasks for each, based on Imagine. The conclusions for the general user requirements included making sure that a user is satisfied with their use of the system, and the benefits the user accrues are greater than the costs they endure; from the service provider's perspective, success was deemed to be a measure of both satisfied customers and increased numbers of transactions; for the intermediary, the primary measure of success was deemed to be derived from the satisfaction of the users.

As user partners in the project, RNIB and Viapolis provided details required by the developers about their base applications. From RNIB, an architectural description and a functional description of the Online Shop were provided. The architectural description presented the technical set up of the system with regard to platforms, network structure, any constraints and future plans in this area. The functional description presented a detailed description of the functionality of the application and a proposed set of functionalities to be included as part of the voice interface to be developed. Example dialogues illustrating this functionality were also supplied, developed from analysis of Customer Response telephone calls between RNIB Customer Services representatives and customers. Use case scenarios of the proposed system were also developed from this analysis.

From Viapolis, architectural and functional descriptions of their system were supplied including the platforms and specifications used for Viapolis.com. The functional description described the differences between the web based application and the WAP version, which would be utilised in the scope of this project. The current WAP functionalities were described and an overview of the functionalities to be emulated by the voice interface was provided. Viapolis carried out a user survey for the voice system, and the results of the survey and the use case scenarios to come out of the analysis were supplied. Brief example dialogues were also presented as a result of this research.

Although dialogues and discussions surrounding the type of natural language expected to be used within the proposed systems were researched and presented as part of this task, further work was required to produce more comprehensive corpora. Therefore, the task of providing further corpus data was extended into task T21 of the project, a more extensive collaboration between the user partners and IAI resulting in larger quantities of data being used.

Research by the developers into the system requirements was also detailed during this task, covering voice processing requirements, natural language processing requirements, interactivity requirements and accessibility and security requirements.

For voice processing, requirements included large-vocabulary speech recognition and synthesis, the ability to recognise conversational speech and speaker independent recognition, among others. Details specific

to the English Language SpeechTool and the Spanish Language SpeechTool were presented separately. For the English tool, technology supplied by Nuance Communications was identified as meeting the core functional requirements. A discussion of the form and content of the English corpus was presented and an issue surrounding the use of proper names for the RNIB application was highlighted. For the Spanish tool, constraints and requirements determined by the specific architecture of the tool were highlighted, including the ability to adjust articulation speed and volume and the need for correct labelling of different variable fields in the corpus construction.

The natural language processing requirements involved mainly a discussion into the Linguistic Processing Module (LPM) that maps word strings on a semantic representation. For linguistic processing, a preprocessing module was deemed to be required along with a module that covers syntactic and semantic analysis of natural language strings.

With regard to interactivity requirements, a discussion into how Imagine must support different interaction mechanisms with users was presented. The discussion pointed to the fact that the most important part of interaction would be adequate dialogue management, and this would be covered by the Dialog Manager (DM). It also included methods for dealing with ill-formed or incomplete input and confirmation and verification strategies in recognition of the fact that the DM would have to deal with determining what the user wants to know or do, even though the information elicited from the user may not be sufficient to enable the system to interface the target application. In order to support the user requirements for both user partners, ideas were presented surrounding how he dialogue control would cope with ill-formed or incomplete input. It was also shown that the DM would specify a set of functions as part of the environment available in all dialogue states e.g. contextual help, session start and transfer of calls to a human operator.

For accessibility and security requirements, a discussion of issues surrounding communication with the target application and output processing was presented. Web application security was pointed out to be the responsibility of the application provider, not the Internet carriers. Also discussed was the way in which speaker verification technology can aid in security over the telephone, pointing out that it can enhance customer convenience by eliminating the need for PINs/passwords, or can augment the PIN/password process if the PIN/password process is required.

For the Viapolis application, it was concluded that Imagine must have an interface that makes HTTP requests to the Viapolis WAP server, parses output (WML) and converts it to text. For the RNIB application, Imagine must have an interface that makes HTTP requests to the RNIB web server, parses the output (HTML) and converts it to text.

As a conclusion to the security requirements, a discussion of IP security was supplied. This presented opinion that for quality of service to be maintained, the system should protect itself from Denial of Service (DoS) attacks. It was pointed out that IP security's key objective is to ensure that traffic between the client and the server is safe from data modification, interception and access by unauthenticated third parties. With IP Security implemented, interactivity via an internal firewall and a DMZ would provide the infrastructure for an overall secure architectural framework.

| **Results in deliverables** |
|---|
| D11 – User & System requirements |

### *3.1.2* T12. Lingware preparation

The task of this work package was to do the specification work for the subsequent implementation of the ***IMAGINE*** system. For the linguistic processing module this meant that the structure of the lexicon, the structure of the grammar, the strategy for the grammatical analysis and the strategy for parsing had to be defined. Last but not least, two corpora had to be built. This was done by IAI in this work package. All the work had to be done for two linguistic modules, one for the English application, the other for the Spanish application.

The Spanish lingware module is based on a previous system that was developed in the **MELISSA** project. The platform for which the Spanish module was designed for is called ALEP. Its main characteristics are that it is unification based, that it provides full flegded typed feature structures. ALEP is very flexible and could be enhanced by and integrated with an implementation of a chunk parsing strategy that was

developed in another context. The preparation of the implementation and the specification of the system on the basis of these prerequisites was done successfully in the first 6 months of the project.

The specification work comprised the design of the whole processing chain from text handling to deep (partial) parsing of the specific text sort to be expected as utterances by the users of the Spanish application. The design work had to bear in mind two major constraints: One was to specify the whole sequence of processing under the specific assumptions of the chunk parsing. Another one was to design the system under the assumption that the input to the parser was text that resulted from a speech recognition module. Spoken language is different to written language in many respects. Among the differences are that there are different syntactic structures to be expected, slightly defective input, sometimes different vocabulary.

The English system was based on a grammar checking formalism that allows for identifying items that occur in form of a complex feature structure and process these structures. The system (which includes a formalism and an implementation) is called KURD which denotes the actions on feature structures that can be performed by this system (kill, unify, replace, delete). KURD is a formalism for shallow post-morphological processing. It expects input from the morphological analyser MPRO. This input is modified according to a number of rules. KURD-rules are specified in a rule-file and sequentially applied to objects read from the input stream. (KURD is the basis for a number of successfully marketed products in the area of language checking).

There were two prerequisites for using the KURD system the availability of which for ***IMAGINE*** was investigated during the preparation phase. The first prerequisite was that the output of a tagger is available (which is the case). The MPRO system provides an English tagger for unlimited text. The second condition was a bit more problematic, namely that the KURD formalism and the implementation is enhanced by an operation that was not available at that time. This operation is the building of a tree over a sequence of items to be defined in a condition part of a KURD rule. Both prerequisites were provided during the preparation phase.

However, the tree building operation has a number of limitations that have to be borne in mind in the design work of the preparation phase. These limitations are deliberate to keep the system robust and efficient. One of these limitations is that no recursion is available. This requires a very careful design of the linguistic module.

Apart from designing the processing chain from text handling to semantics it was a task to determine an interface structure for the output of the linguistic processing that is common to both the Spanish as well as the English module and that can be handled by the system that introduces application specific knowledge, the SAM module. This required careful specification for the representation of all kinds of linguistic phenomena, such as predicate – argument –structure, semantic class systems for modalities, time, aspect, quantification and determination.

Another major task for both was to specify the user specific resources. An elaborate type system for both user specific resources, catalogue items etc. had to be defined.

| **Results in deliverables** |
|---|
| D12 – Lingware specifications |

### 3.1.3 T13. Func Specs & Architectural Design

The goal of this task has been to obtain the ***IMAGINE*** architecture based in two premises:

- The MELISSA architecture.
- The results of task 1.1: user requirements.

From the former meetings, it was stated that the **MELISSA** architecture needs a more robust SAM and a more elaborated Dialog (called controller at **MELISSA**), and an updated FEM able to access trough Http protocol. Although the rest of the components are near of the current ***IMAGINE*** architecture, most of them, has been redesigned and developed again. The following figure shows the complete ***IMAGINE*** architecture.

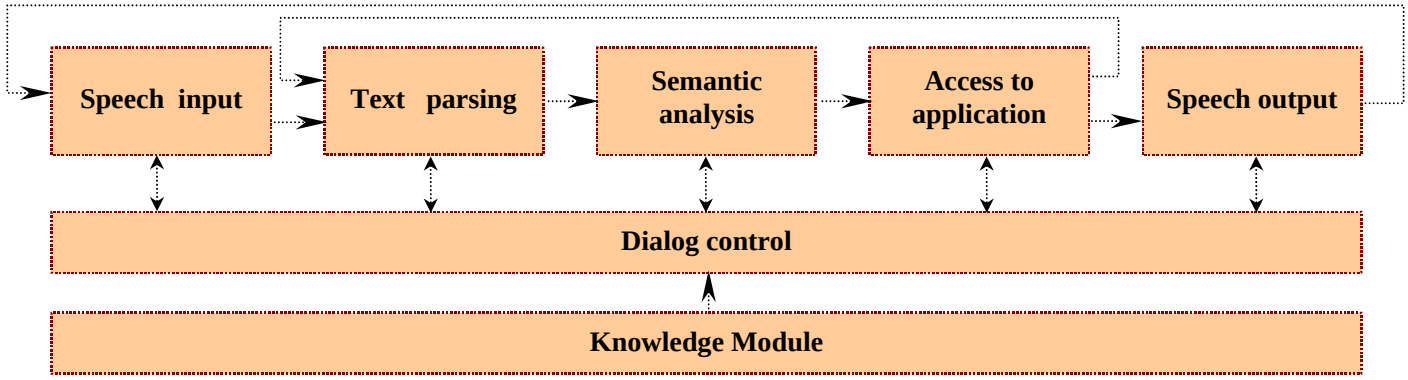



*Fig 3.1.3.1 - IMAGINE architecture summary.*

The analysis and design has been followed an UML approach with the corresponding diagrams to each phase, and the extensive use of the UML resources to clarify the complex relations between the components, mainly between the speech tools and the ***IMAGINE*** core.

***IMAGINE*** has been designed as a central resource, the dialog, what orchestrates all the interactions with the user and the software components. The dialog uses all the components as resources: Speech tools, Linguistic Components, etc. The objective of the dialog is to give answers to the user questions using the resources prepared for concrete actions:

- **English and Spanish Speech input/output:** Tools solving the connection with users. They include two functionalities: speech input and output and phone control. Those functionalities are focused in various components but seem from ***IMAGINE*** as a single functionality. They represent the user from DHM point of view.
- **Dialogue Control:** Is the main component in the ***IMAGINE*** core. It knows every dialog issues, and plays the role of controller for the full system. It delivers the first user prompt for welcome the user in the application and control every steps of the user inside ***IMAGINE***. It collaborates with the user in order to fill in every forms required to reach the application interfaced, and uses the ***IMAGINE*** resources to parse the user input and to get the application results.
- **Text parsing:** Includes every components that have to resolve the adaptation of a user input to an ***IMAGINE*** internal format.
  - **Text handling:** Is the component of linguistic modules that have to transform data in user language to internal application codes.
  - **Linguistic Processing:** Is the parsing of user input utterances. Translate user transcribed speech in to text, to a set of semantic facts gathering the meaning of user sentence.
- **Semantic Analysis:** Select among every logical function in the application. Translate LPM semantic facts output into logical functions.
- **Access to application:** This component includes every activities involved in the connection with the application interfaced.
  - **Front End Module:** Solve the connections with the applications to reach the functionality selected by the user. This connection can be solved via HTTP, RMI, CORBA, ...
  - **Answer Adapter Module:** The answer delivered by the application interfaced has to be translated to text in order to finally be translated to speech.
- **Knowledge Management Module:** This is the ***IMAGINE*** repository, it contains every details of the application interfaced and the rest of persistent information in ***IMAGINE***.

As concrete results of this task, the deliverable *D13 - Functional Specifications and Validation Plan*, has been elaborated. It has been the medium used to store and interchange all the agreements, achievements and subsequent updates being generated 15 releases.

| Results in deliverables |
|---|
| D13 – Functional Specifications & Validation Plan |

## 3.2 WP2. Natural language interface

### *3.2.1* T21. Linguistic processor

The linguistic processing which foresees a deep analysis (in a robust manner) resulting in a predicate-argument-structure had to be implemented for the text sorts to be expected, for the two languages, English and Spanish, and for the two different applications which are expressions of requests by users to make the application do things and respond to questions.

For the Spanish language the application was an information access point for retrieving information on restaurants, trains, theatres, cinemas etc. and the possibility to book tables in these restaurants and order tickets for the trains, cinemas and theatres and buy them over the internet.

For the English application an e-commerce cite provided by RNIB for blind people, but also sighted people has been chosen which requires to do actions like ordering, choosing, browsing for information, buying items from a large catalogue by using NL. Other actions that go along with the previous ones are to log on to the site by identifying oneself, also log off when the whole process is finished. It includes handling the shopping cart, putting things into the cart, and removing items from the cart.

The task in this work package was to provide the processing modules (which cannot completely be separated from the task below about creating resources).

These modules mainly consist of:

1. For Spanish: an analysis module that has different submodules, a text handling module, a syntactic module, and a semantic module
2. The same for English, though the sequence of processing is different.

Very importantly, there is at the end of the linguistic processing a conversion module that converts the output of the linguistic analysis into an XML-based format that is required by the application specific interpreters. This is however a module that simply converts one representation format into another.

Finally, a small application that was to prove feasibility of concept for German was provided in form of a German grammar and lexicon that is able to produce IMAGINE output that can be processed by an application specific module (SAM) that allows the user to access that SAGE Web site that provides information about the RTD activities of the company.

Separate modules that allow for access also were produced for Spanish and English.

| Results in deliverables |
|---|
| D21 – First version of NL components |
| D22 – Second version of NL components |

### *3.2.2* T22. Mobile device & Voice Integration

This task consists on the integration of a voice speech system with IMAGINE. Two different approcahes have been followed for the integration of English and Spanish voice systems. They are following described:

#### 3.2.2.1 English

**Overview**

Voice input and output to the IMAGINE system in English is handled by the English speech tool (EST). Mobile or fixed telephone access to the system is achieved by connecting the system to a public telephone

network via a telephony interface card which enables a computer to answer telephone calls, play audio output and record audio input. In other words, a caller dials a telephone number that calls the telephony interface using his/her telephone or mobile phone. Once the interface detects the call, it will "pick up the phone" (establish connection) and direct the audio input and output to the speech recogniser and speech synthesiser, respectively. This makes it possible for the speech recogniser to record the audio input, analyse the signal and generate a transcript of the spoken input. The speech synthesiser can also convert textual input into spoken words and play the audio output to the caller.

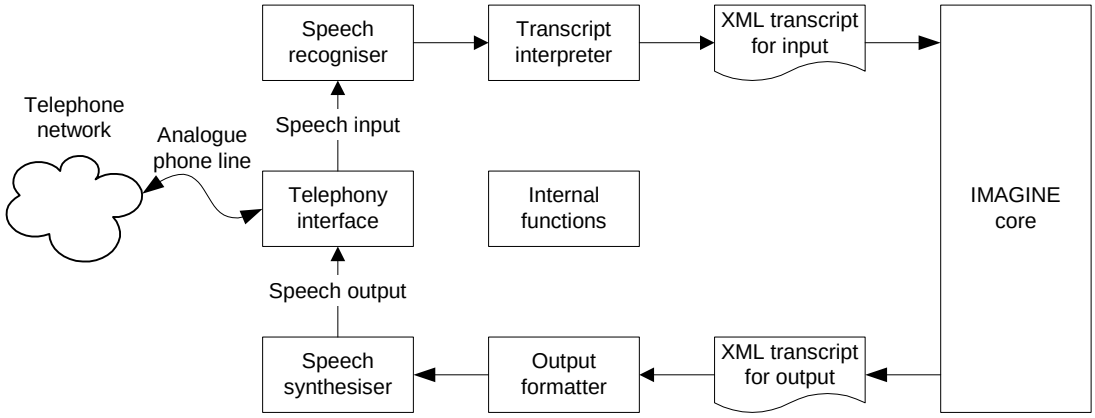



*Fig 3.2.2.1.1 – English Speech Tool deployment.*

Development effort was largely devoted to the internal functions and recognition grammar for the speech recogniser. The internal functions enable the speech recogniser to prompt the user to repeat an utterance if recognition accuracy is poor. It also handles all the caller adjustable speech generation parameters such that a caller can ask the system to speak louder, quieter, faster, slower or transfer the call to a live agent by saying the request.

The recognition grammar defines the scope of recognition, i.e. what sentences are understood by the system. The English speech tool is not an open transcription device that can transcribe an utterance. It can only transcribe the sentences that are defined in the recognition grammar. This approach to recognition improve transcription accuracy and speed performance by limiting the search space. Furthermore, it improves the adaptability of EST as new utterances can be added to the recogniser by a developer with minimal effort.

The transcript interpreter is a novel design that was developed in this project. Existing speech recognition modules aim to produce a perfect transcript for the speech input. This implies the linguistics modules in the ***IMAGINE*** core has to interpret spoken text. Given the linguistic modules were originally designed for written text, the transcript interpreter was added to simplify the transcript such that the output is clean and grammatical like written text, thus, increasing the interpretation accuracy and coverage of the ***IMAGINE*** core. Furthermore, this approach improves the adaptability of the overall ***IMAGINE*** system as adding new phrases and constructs to the recognition grammar is significantly simpler than modifying the linguistic modules within the system.

**Recognition Grammar**

Several recognition grammars were developed based on different technologies. The main difficulty was finding the balance between flexibility, adaptability, accuracy and speed performance. The following sections presents a summary of each approach and discusses their merits and pitfalls.

**Context free Grammar – G1**

The first grammar implementation was based on context free grammar rules. The grammar defines all the possible utterances that can be understood by the system using regular expressions. The speed performance of this approach is excellent. Furthermore, all the possible transcription output is known at compile time, thus, simplifying the integration of EST with other linguistic components. The main drawback of this approach is that the grammar is completely rigid and cannot accept any utterance that is not defined in the grammar, i.e. even minor variations are rejected. The grammar rules are human readable and editable, however, modifying a large scale context free grammar requires expertise. Development and

management effort for such grammars increases exponentially as with the grammar size. To conclude, this approach to recognition grammar specification is ideal for small systems but impractical for large grammars.

**Statistical Language Model – G2**

The second implementation was based on a statistical model of the domain text. The grammar is defined as N-grams. The aim is to obtain an accurate estimate of the probably of a word occurrence given limited context. For instance, a bi-gram model predicts the probability of a word occurring given the previous word, e.g. the word “is” is more likely to follow “this” than “are”. The probabilities are estimated by analysing a large corpus of relevant text, i.e. the training set. The application of such grammar requires the recogniser to find a sequence of words that maximises the likelihood of occurrence. The search procedure is complex and computationally intensive, thus, the speed performance of this approach is poor. However, the recognition grammar is highly flexible and robust. It can interpret utterances that are wildly different those in the training set. Practical experiments were conducted to evaluate this method. Test results show the method is feasible and practical for a domain where the vocabulary is small and the training set is large. In the context of the RNIB application, the small training corpus was not directly relevant to the target domain. It contained a large vocabulary, thus, recognition accuracy was poor. To conclude, this approach to recognition grammar specification is flexible and convenient but relies on the availability of a large training corpus. Integration with other linguistic components is difficult as the method can generate an infinite set of outputs.

**Hybrid Grammar – G3**

The logical progression from previous work is the development of a hybrid method that combines context free grammar with statistical language models. The method used a hierarchical statistical language model to combine sentence structures that are based on either a statistical model or context free grammar. In other words, parts of utterances are implemented separately with context free grammars (e.g. post code) and statistical language models (e.g. product name). The complete utterances are then modelled as a statistical language model that refers to these parts. The aim is to increase accuracy and speed performance by reducing the search space with specialised models for different parts. A utility was developed to enable a novice to modify the grammar. To conclude, this approach to recognition grammar specification is flexible, convenient and facilitates non-expert editing. The accuracy and speed performance of the approach is good, however, integration with other linguistic components is difficult as the method can generate an infinite set of outputs.

**Interpretation Grammar – G4**

The final method which is used in the deployed system is a novel approach that was developed in this project. The recognition grammar maps all possible input to a finite set of output to simplify integration with other linguistic components. This is achieved by focusing on keywords and key phrases that implies the full sentence, e.g. “place order” implies “I want to place an order”. The grammar is implemented with context free rules. The grammar consists of a set of filler phrases for modelling superfluous input (e.g. “um”, “er”) and a set of key phrases for identifying the complete input (e.g. “place order”, “order”, “buy something” implies “I want to place an order”). The grammar has excellent accuracy and speed performance but it has to be crafted by an expert grammar developer. Integration with other linguistic components is simple as the grammar generates a finite set of transcripts as output.

**A comparison of Grammar technologies**

The following table presents a summary of the grammar technologies. Accuracy refers to the recognition accuracy of the grammar. Adaptability is measured by the level of expertise required to change the grammar. Speed is the runtime performance of the recognition grammar. Flexibility is measured by the grammar’s tolerance to new input that does not exist in the training data. Finally, integration refers to the complexity involved in integrating the grammar output with other linguistic components. The table shows the interpretation grammar is the best grammar for use in most practical applications.

| | **Context free grammar** | **Statistical model** | **Hybrid grammar** | **Interpretation grammar** |
|---|---|---|---|---|
| *Accuracy* | Excellent | Poor | Good | Excellent |
| *Adaptability* | Poor | Excellent | Excellent | Good |
| *Speed* | Excellent | Poor | Good | Excellent |
| *Flexibility* | Poor | Excellent | Good | Good |
| *Integration* | Good | Poor | Poor | Excellent |

*Fig 3.2.2.1.2 – Comparative grammar table.*

| **Results in deliverables** |
|---|
| D21 – First version of NL components |
| D22 – Second version of NL components |

### *3.2.3* T23. Semantic Analyser & e-dialog management

This task includes two of the main tasks of ***IMAGINE***: the semantic analysis and the dialog or interaction management. Both components, although independent modules, work closely to understand what the user is asking to the system, in terms of the interfaced application semantics. Both modules use as reference the KNM, repository where the application knowledge is stored.

The focus of the Dialog is to interact with the user to complete all the required details for executing one functionality. In order to do that, the dialog needs to manage the user-stated meanings in an adequate format. This translation is made by two components: the Linguistic Processing Module (LPM) and the Semantic Analysis Module (SAM).

The LPM gets the user input as a row text stream and translate it to a well known set of semantic facts related among then. This output follows a very restricted format described in an XML DTD. The translation of such general-purpose semantic facts, to concrete elements in the interfaced application is made by the SAM.

For example, the sentence: "*I would like to search for products*" is translated by the LPM to:

```
<LpmOutput>
 <Interpretation>
  <pred>
   <atom>
    <type>
     <imaginetype>
      <name>generic_VSearch</name>
      <text>Search for products</text>
      <code><value></value>
      <description></description>
      </code>
      <lemma><value>search</value></lemma>
      <accuracy><empty/></accuracy>
     </imaginetype>
    </type>
   </atom>
   <var>var1</var>
  </pred>
  <index>
   <indextype>event</indextype>
   <var>var1</var>
  </index>
  <tense>
   <tensevalue>pres</tensevalue>
   <var>var1</var>
  </tense>
  <polarity>
   <polvalue> 1 </polvalue>
   <var>var1</var>
  </polarity>
  <mforce>
   <modal>nil</modal>
   <var>var1</var>
  </mforce>
  <pred>
   <atom>
    <type>
     <imaginetype>
      <name>rnib_onLineShop_NProduct</name>
      <text>for products</text>
      <code><value></value>
      <description>product</description>
      </code>
      <lemma><value>product</value></lemma>
      <accuracy><empty/></accuracy>
     </imaginetype>
    </type>
   </atom>
   <var>var2</var>
  </pred>
  <rel>
   <reltype>arg2</reltype>
   <var1>var1</var1>
   <var2>var2</var2>
  </rel>
  <index>
   <indextype>nevent</indextype>
   <var>var2</var>
  </index>
  <qforce>
   <quant></quant>
   <agr>plu</agr>
   <def></def>
   <var>var2</var>
  </qforce>
 </Interpretation>
</LpmOutput>
```

*Fig 3.2.3.1 – LPM output example 1.*

The LPM representation retrieves five possible alternative understandings of the sentence, as the item <name>rnib_onLineShop_NProduct</name>, when translated into the application space can refer to: 5_ListProduct, 4_ListSubcat, 3_ListCat, 2_ListAz or 1_Search.

The SAM, after checking the possibilities with the KNM, considers that the user sentence can be understood in various ways. Those ways are the various functionalities that the user could be referring to: 2_ListAz or 1_Search.

The dialog, after some analysis to select the most adequate interpretation, according to previous interactions with the user, asks the user to select among the possible interpretations: "*What do you mean? Please select one of the following options by saying its number or say stop if you don't want any of them: 1 To search products according to a search criteria or 2 List of products starting with a concrete character.*

In the following example: "*I would like to see a list of products*" we show how in this case the Dialog module filters the possible alternatives retrieved by the Semantic Analysis, on the basis of the "User State Diagram" (please refer to section 3.3.1.3 for further detail)

```
<LpmOutput>
 <Interpretation>
  <pred>
   <atom>
    <type>
     <imaginetype>
      <name>generic_VList</name>
      <text>I would like to list
products</text>

<code><value></value><description></descripti
on></code>
      <lemma><value>list</value></lemma>
      <accuracy><empty/></accuracy>
     </imaginetype>
    </type>
   </atom>
   <var>var1</var>
  </pred>
  <index>
   <indextype>event</indextype>
   <var>var1</var>
  </index>
  <tense>
   <tensevalue>pret</tensevalue>
   <var>var1</var>
  </tense>
  <polarity>
   <polvalue> 1 </polvalue>
   <var>var1</var>
  </polarity>
  <mforce>
   <modal>want</modal>
   <var>var1</var>
  </mforce>
  <pred>
   <atom>
    <type>
     <imaginetype>
      <name>generic_human</name>
      <text>I</text>

<code><value></value><description></descripti
on></code>
      <lemma><value>I</value></lemma>
      <accuracy><empty/></accuracy>
     </imaginetype>
    </type>
   </atom>
   <var>var2</var>
  </pred>
  <rel>
   <reltype>arg1</reltype>
   <var1>var1</var1>
   <var2>var2</var2>
  </rel>
  <index>
   <indextype>nevent</indextype>
   <var>var2</var>
  </index>
  <qforce>
   <quant></quant>
   <agr>sg</agr>
   <def></def>
   <var>var2</var>
  </qforce>
  <pred>
   <atom>
    <type>
     <imaginetype>
      <name>rnib_onLineShop_NProduct</name>
      <text>products</text>
      <code><value></value>

<description>product</description></code>
      <lemma><value>product</value></lemma>
      <accuracy><empty/></accuracy>
     </imaginetype>
    </type>
   </atom>
   <var>var3</var>
  </pred>
  <rel>
   <reltype>arg2</reltype>
   <var1>var1</var1>
   <var2>var3</var2>
  </rel>
  <index>
   <indextype>nevent</indextype>
   <var>var3</var>
  </index>
  <qforce>
   <quant></quant>
   <agr>plu</agr>
   <def></def>
   <var>var3</var>
  </qforce>
 </Interpretation>
</LpmOutput>
```

*Fig 3.2.3.2 – LPM output example 2.*

As it can appreciated in the figure, the LPM representation retrieves as in the previous case, the item <name>rnib_onLineShop_NProduct</name>. The Semantic Analysis module expands it to the five possible alternatives, with no possibility to filter them. The SAM output is: 5_ListProduct, 4_ListSubcat, 3_ListCat, 2_ListAz or 1_Search.

The SAM, after checking the possibilities with the KNM, considers that the user sentence can be understood in various ways. Those ways are the various functionalities that the user could be referring to: 2_ListAz or 1_Search.

The dialog, after some analysis to select the most adequate interpretation, according to previous interactions with the user and the “User State diagram”, discards the option 4_ListSubcat, and then asks the user to select among the possible interpretations: “*What do you mean? Please select one of the following options by saying its number or say stop if you don't want any of them: 1 To search products according to a search criteria or 2 List of products starting with a concrete character or 3 List products by category or 4 Description of products*”

### 3.2.3.1 Dialog Handling Module

The key on which the operation of the Dialog Handling Module (DHM) is based, is the model of target applications in terms of logical functions, dependencies between them, codification of proper names in types; the so called ontology e-business model.

An interaction with the user starts when the user connects with an enabled phone number that is the access point to one of the applications interfaced. In this moment a new session for this user is created. The system receives the user with a stored message that is represented in the Welcome function. The user decides to get some information from the system and a sentence is received by the dialogue.

By each sentence received by the dialogue, the dialog has to decide what is the intention of the user with this utterance. To do this, the DHM uses the Linguistic Module (LM). The LM includes the modules able to cope with syntactic and semnatic analysis: LPM and SAM. The DHM knows what type of input could be received at every moment maintaining the interaction context. In order to be able to know this, the dialogue stores in session each one of the associated steps to the course of the interaction. Next figure shows the DHM general process.

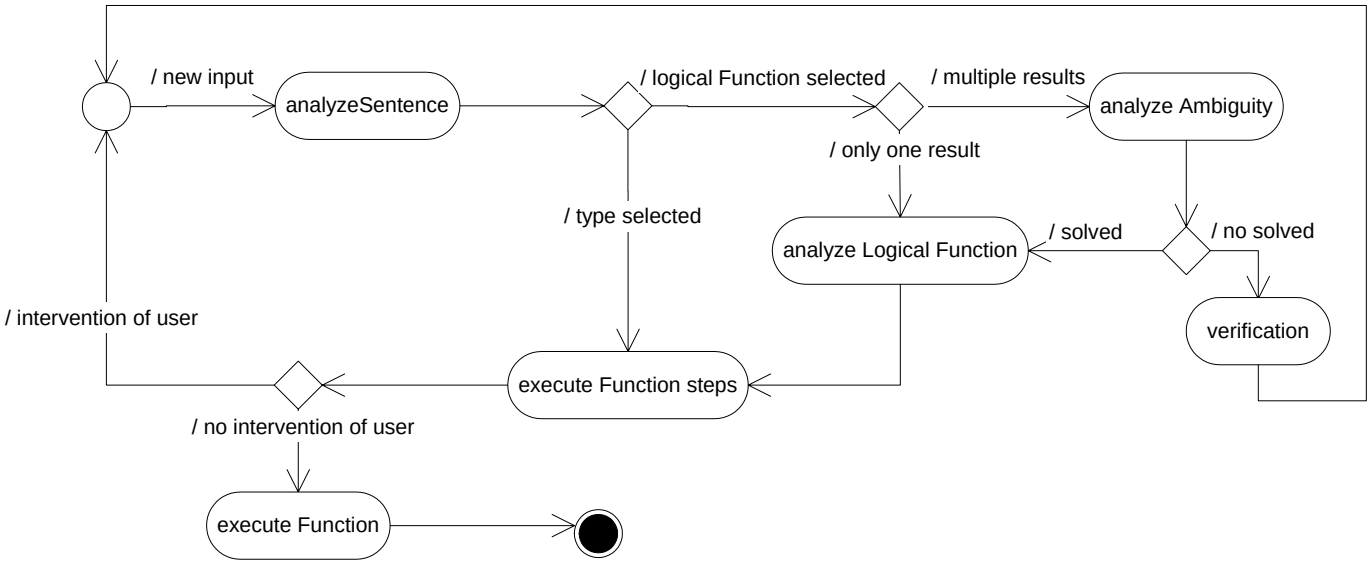



*Fig 3.2.3.3 – Dialog activity diagram.*

In order to carry out this task, it is essential the collaboration of all the components that comprise IMAGINE, co-ordinated by the Dialogue Control, and the modelisation of target applications in terms of logical functions, dependencies between them, codification of types (type formats and synonyms) referred in the logical functions. This modelisation is the base for the Dialog Handling Module.

The strategy implemented in the dialogue management allows that the initiative is taken by user or by system depending on the functionality. Dialogue management has to determine what questions the system should ask, in what order and when, but the user could decide which information wishes to facilitate to the system in what order and when.

Some problems derived from ill-formed sentences or incomplete or inaccurate user inputs, could affect the result of some of the components of IMAGINE, as for instance the speech recognition and language understanding components. For this reason the Dialog Manager needs to implement some mechanisms in order to help solving what the user wants to know. To deal with input that the system recognises as ill-formed or incomplete, dialogue manager uses confirmation strategies to verify that the input recognised by the system is indeed the user intention. In some cases the dialogue manager has no choice but has to suggest the user to express the request in another way.

### 3.2.3.2 Semantic Analysis Module

The Semantic analysis is a language independent component that bases its work in the idea that every interaction with the interfaced application is structured from semantic point of view according to the next structure or parts of it:

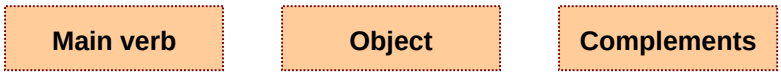



*Fig 3.2.3.4 – Semantic structure of user utterances.*

The previous three concepts are constructed in terms of IMAGINE types. The semantic analysis, according to the execution context provided by the dialog, analyses the text parsing output looking for complete functionalities or concrete IMAGINE types.

Going into a more technical detail, the Semantic Analysis takes as input the LPM output which is a representation of the user utterance in terms of semantics facts. The input can be a list of possible interpretations coming from the LPM, each one having potentially more interpretations once they are translated into the application context. The Semantic Analysis processes this structure and based on the knowledge about the application, retrieves a similar structure, now in terms of logical functions. In the process, the Semantic Analysis can choose only one alternative from among the initial possible ones, or can retrieve more than one, depending on the application knowledge and the functionality this provides.

Application functionality

The semantic that IMAGINE will understand is restricted to the semantics that the application is already providing like *list of products*, *registration*, *log in*, *list of restaurants*, etc. There is no limit in the number of functionalities to map and in the number of constituents for each functionality. The semantic analysis does not reason nor deduce user intentions that are not directly expressed in terms of the application functionality.

The number and nature of the functionality to achieve in the interfaced application have direct relation with the level of ambiguity that the semantic analysis has to cope with. The greater number of functions the greater potential ambiguity. Also, the most semantic similarity among functions the most potential ambiguity. It is very difficult to measure the level of ambiguity that a concrete IMAGINE installation could have but is directly related with the two previous factors. Is the Dialog the module in charge to manage this ambiguity in the most adequate way.

IMAGINE types

As shown before, the functionalities are composed by three main constituents, based on the use of IMAGINE types. There is a taxonomy of IMAGINE types that includes dozens of types, being the minimum semantic components managed by the semantic analysis. These taxonomies include general use concepts as *date*, *number*, etc, or more application dependent as *product code*, *place categories*, etc.

The independent understanding of these constituents is very important when IMAGINE, by means of the dialog, takes the initiative of the dialog with the user, asking him/her about concrete details for completing the information required in a specific application functionality.

| **Results in deliverables** |
|---|
| D21 – First version of NL components |
| D22 – Second version of NL components |
| D23 – Dialog and interaction model |

### 3.2.4 T24. Linguistic resources

Linguistic resources for the two language modules Spanish and English include a number of grammars and lexicons for the processing of the specific utterances to be expected by applications of the two languages. The two modules are available in different formats due to the fact that they were produced on two different platforms.

In more detail, the linguistic resources comprise:

1. For Spanish:
   a) A text handling module that contains the rules for processing text handling phenomena such as figures, names, dates, etc. It is an extremely important module in view of the application.

b) A lexicon that contains general language terms that are need for expressing the actions a user would like to express.

c) A grammar to handle the natural language expressions for doing the actions that a user would like to do.

d) Very importantly, a stock of application specific resources that cover the items in the databases of the application such as hundreds of names of towns, theatres, cinemas, restaurants, etc. These resources had to be created from scratch in the ***IMAGINE*** project to be able to cover the requests for actions by the user.

e) This resource is available as in the format of a typed feature structure and contains all the relevant information.

2. For English:

a) A morphological processing module that allows handling any free English input and returns a tagged output.

b) A lexicon of general language words that are necessary to express actions to be done in the application.

c) Grammars handling the structures for expressing the actions to be performed for the English application. It consists of a sequence of modules that perform different analyses such as the introduction of valency the building of a tree structure in the way it has been described above, namely as a non-recursive representation that contains a predicate argument structure in its top node.

d) As well as for the Spanish module there is a huge lexicon (actually a number of different lexicons) that contains all the application specific resources such as the complete catalogue of items that can be bought at RNIB's site. It contains all the classifications of the goods, the categories they belong to etc.

The linguistic resources go far beyond what is required for handling the two applications. They are generalised resources.

| **Results in deliverables** |
|---|
| D21 – First version of NL components |
| D22 – Second version of NL components |

## 3.3 WP3. e-commerce applications integration

### 3.3.1 T31. Ontology e-Business Model formalisation.

As advanced in previous sections, various components use intensively the information included in the Knowledge Management Module, the so-called KNM. This KNM has a structure for represent the information of the interfaced applications. The task 31 is devoted to such KNM definition and to model the various application interfaced during the IMAGINE project.

The KNM is the addition of three main concepts:

- Logical Functions.
- IMAGINE types.
- State diagrams.

#### 3.3.1.1 Logical functions

Followed by the experience in MELISSA, it has been defined the concept of Logical Function. A Logical Function represents the minimum and compact functionality provided by one interfaced application. This is similar to the concept of API (Application Program Interface), but in the case of IMAGINE, this API definition has to be general enough to support any kind of interfaced API. This means that the IMAGINE Logical Function must be flexible enough to represent the functionality of a HTTP URL, a Web service, etc. The ordered set of Logical Functions representing the application functionality is called the application model.

A Logical Function can be composed by different unitary actions. For example, the Logical Function "To pay", can be composed by different steps: first to obtain some known data of the user from the system, next to ask the user about the pay media (CC or account) and finally, make the effective pay depending on the type of user payment option selected. In order to represent this, it has been defined three types of Logical Functions: LF, SubLF and Fictitious.

From the model point of view there are two types of Logical Functions: Environment Logical Functions and Application Especific Logical Functions. The first set includes those Logical Functions that could occur at any interfaced application. It includes the functionalities like Help, Quit, etc. The Application Especific Logical Functions are those representing the functionality of the interfaced application and have sense only in the context of the application. For example Payment, ShowThe BasketContent, etc.

#### 3.3.1.2 IMAGINE types

**IMAGINE Type Formats**

The Logical Functions represent the functionality of the interfaced application as it is explained in the previous section. The structure of the API represented in the Logical Function, use to be a method or process name completed with a set of parameters. The parameters are values that belong to a type, typically integers, strings, booleam, etc. The general purpose of IMAGINE requires to have a general representation of the application types. This representation must work in the same way for every new application that were interfaced by IMAGINE.

There is another very relevant concept around parameters and types. Typically, the values in the API parameters use to be codified information representing real life measurements or concepts. For example, there are many ways to express a date depending on the language, the user, etc, but finally, it must be provided to the API according to some format. Other example of this is the use of codes, for example for products in a shop. The API use to require a codified value for each possible product.

In IMAGINE, the values to fill such API parameters are obtained from the user utterances, provided of course in Natural Language. Besides, the users interfacing the applications are typically not experts in the application interfaced, and even being experts, it is unreal to expect the users to provide the API information in the API format: dates, codes, booleam values, etc. Therefore is the IMAGINE responsibility, to translate the user input utterances to the adequate set of codes and formats that the interfaced application requires.

IMAGINE defines a taxonomy of types that includes every types that can be managed in IMAGINE. This taxonomy includes two kind of types: general use types and application dependent types. The general use types are those typically used in every application: dates, numbers, etc. These general use types can be completed with specific application types, like for example dates with special format used in a concrete application. But the main source of application dependent types are obtained from the coded values like products, place names, activities names, etc that has a description known by the user, but that the corresponding code is managed in the application API. All those types are gathered in a big taxonomy of IMAGINE types called IMAGINE type formats, see the section IMAGINE types.

**IMAGINE Type Synonyms**

It has been taken advantage of the Text handling Module (THM) chunking process for the Semantic Analysis Module (SAM). The main task of the SAM is to select the adequate Logical Function according to the meaning of the user utterance. This process is a kind of bridge between the user sentence and the Logical Function contents.

The Logical Function represents the API to call in terms of IMAGINE type formats and other details. Besides this application side in the Logical Function structure, the Logical Function has been enriched with semantic information that can help the SAM in the process of selecting the most adequate LF. This extra information adds semantic features to the Logical Function.

The semantic features included in the Logical Function are mainly two:

- Main verbs used in the corpus related with this Logical Function.
- Main object used in the corpus related with this Logical Function.

The main verb and the object are the most relevant linguistic features in the corpus gathered for this kind of application. Both semantic features plus the complements, are the majority of linguistic phenomena expected in the user utterances according to the language expected. Such complements are represented in the Logical Function by the parameters of the API. Therefore, the own Logical Function structure represents a semantic model for the API

#### 3.3.1.3 State Diagrams

Once the minor details are modelled and every functionality is represented in the corresponding Logical Function, it is the time to represent the information about the relation among the Logical Functions. It has been defined three types of such relations:

- IMAGINE state diagram.
- User State Diagram.
- Application State Diagram.

The IMAGINE State Diagram shows the dependencies of the Logical Functions among then according to already executed order. It is required, to execute the LF 1, that the LF 101 has been executed previously. This kind of dependencies happen when there are various steeps for the execution of a Logical Function or when it is necessary to access the application interfaced to get some specific information for the adequate execution of the main Logical Function. This kind of information is typically the value for internal vars not provided by the user: session codes, server controls, etc. But required for the effective execution of the Logical Function.

The User State Diagram reflects the expected behaviour of the user in the interface with the system. For example, it is expected that the first activity of the user in the Online Shop is to ask for the existence of products, and maybe them, add then to the basket, and finally to pay the full purchase. This information is very relevant for the English Speech Tool, based on a grammar model, because the test of the most reasonable grammar according to the application context, can derive in a more efficient system.

The Application State Diagram represents the preconditions that have to be carried out for the execution of a concrete LF. For example, in the RNIB Online Shop it is not possible to add products to the basket if the user has not logged in the system.

| **Results in deliverables** |
|---|
| D23 – Dialog and interaction model |

### 3.3.2 T32. Front-end integration

In the process of use IMAGINE to access an application, once the user intention is completely understood, and all the details required are filled, the pending task is to connect with the interfaced application and get the application answer. This process is located in the main component called Executive Module (EM). The EM uses two modules:

- The Front End Module. This component cope with the details of the connection with the interfaced application, accessing “physically” to the point where the selected functionality is.
- The Answer Adapter Module. Once the application answers, this component takes control of such answer in order to translate it to Natural Language terms.

#### 3.3.2.1 Front End Module

The Front End Module (FEM) is the point from where IMAGINE connects with the interfaced application. It takes the control of communications and effectively connects with the application. The application will not notice the difference between this call and other calls coming from another potential clients using this API.

The FEM is designed to interface various types of API like Http URI's, what is the method currently supported, but it can easily include another protocol types like Web Services, RMI, etc. The modularity in the design allows the inclusion of specific software for the new protocol with no changes in any other parts of the system.

In the current version, the FEM is prepared to work with Http, Https protocols, managing most typical Http functions as GET or POST, cookies control, links redirection, etc. The answer delivered by the application interfaced is unwrapped of protocol details and deliver back to be managed by the component that the EM decides, usually the AAM.

#### 3.3.2.2 Answer Adapater Module

The function of the Answer Adapter Module (AAM) is to identify and extract information to be reused by ***IMAGINE***. Information is typically taken from a HTML or WML page generated or called by the system vendor. The layout structure of these pages is important, as it is this which determines where information within the page resides.

Ideally, IMAGINE specific tags would be embedded within the system vendor pages to make information extraction non-structure dependent as specified in the second AAM enhancement proposal but due to time limit constraints this was not carried out.

Interfacing with AAM is done by two methods:

- **getUserPrompt:** A HTML/WML page is passed to the AAM along with its corresponding XSLT script. This is converted to an XML standard to reduce the possibility of parsing errors such as non-standard or non-closed tags. The XSLT scripts extract information based on whether the page contains an error or not. The AAM determines this by the commented field that the system vendor must include within the header of such pages which are to be processed by IMAGINE, i.e.: `<!--<imagine.aam.response state="normal"/>-->` or `<!--<imagine.aam.response state="error"/>-->`

  In either case, information is extracted by the XSLT scripts based on its XPath within the page. This assumes that information will always reside in a specific place. Changes to this will inherently mean that the incorrect text or no text at all will be extracted. The output from the AAM is given in a presentable form that can be interpreted without knowing the rest of the information in the original page. This is enclosed within the tag pair: `<imagine.aam.response state="X"> </imagine.aam.response>`, where X may represent either "normal" or "error".

- **getHiddenVars:** This method returns the "hidden" information within the page which is passed. These are typically variable names and values attached to parameters within forms or attributes as part of URI's. Information that is extracted is structured based up on searches made within the page, groups of variables and variable types, so that the answer returned is in a useable form.

  A specified form (whether by '`Action`' name or '`HREF`' link) returns value variable pairs within <set> </set> tagged pairs. Variable names are placed within `<var> </var>` tags and their corresponding values are places within `<value> </value>`, directly proceeding the `<var>` tagged pair.

  `Select/Option` HTML tags within a form are handled differently as the description and whether an option is also represented is also given, such that, e.g.:

```
<var>shipping_title</var>
<value selected="true" description="Mr.">Mr.</value>
<var>shipping_title</var>
<value option="true" description="Mrs.">Mrs.</value>
```

  displaying whether an option is selected or not respectively.

  Search parameters may also be passed to return specific information within a form if it exists, however if it does not, then an empty `<AAMOutput/>` is returned.

It is foreseeable that a system vendor will want to develop their Web site. This will mean that their corresponding AAM XSLT scripts will require continuous updates to guarantee the XPath for specific text locations always point to the right place.

#### 3.3.2.3 Question answering

As part of the studies done during the project time, it has been considered interesting from the very beginning, to study what is the level of information extraction that IMAGINE can do from an interfaced application.

Typically, the interfaced application is prepared to provide information according the paradigm selected for the interface, 90% graphical mode. This graphical mode can rely in HTML pages, XML + XSL, text screens, WML, etc. The objective of IMAGINE is to translate such information to a text stream that can be performed to the user via the Text to Speech ability of the speech tool.

The question has been what is the IMAGINE capability to analyse the information provided by the application and use it to give more detailed or even elaborated answers to the end users. This capability has been focused in the so-called question answering.

Due to the flexibility of the natural language, the end user will have the fill of freedom to talk with the system as if it were a human. In this relaxation of the communicative process, the limits that the system had, had not be very clear for general users, and users could try to ask more information that the concrete data given by the interfaced application.

The Question answering study try to analyse the mechanism required in IMGAINE and in the interfaced application to provide adequate answers to such questions.

| **Results in deliverables** |
|---|
| D31 – First version of e-commerce components |
| D32 – Second version of e-commerce components |
| D34 – Question Answering (not committed in the contract) |

### 3.3.3 T33. Development & Integration Tools

Future service providers will need to be able to model and maintain the IMAGINE interface to their base applications. A set of tools has been made available to allow certain updates to be carried out by these groups of users. By components, the possibility of maintenance with specific tools is the next:

- **KNM:** Tool allowing the maintenance of all files included there.
- **EST:** includes a grammar workbench tool for updating the grammars.
- **AAM:** requires to design ad-hoc XSL scripts for each of the pages delivered by the application (in HTML, WML protocols).
- **SST:** requires developers direct update.
- **LPM:** requires developer direct update.

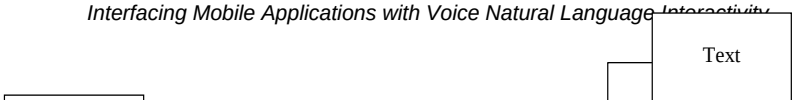
Rest of components does not require adaptations.

#### 3.3.3.1 Knowledge Manager (KNM)

The Knowledge Manager (KNM) requires various skills for its maintenance. The KNM includes technical information, like whether a cookie is used in a particular HTTP method call, but also includes high level information such as the text to be prompted to the end users. Due to these mixed contents, different administrator profiles are required.

In general, it is recommended that a technical administrator, in close collaboration with an end user representative, begins the development of the model, defining the Logical Functions and gathering all the technical details around it. Once this first step has ended, the administrator concerned with end user





requirements (could be the same as the previous administrator), will define and refine the prompts, the dialogue details and preferably the semantic model.

*Fig 3.3.3.1.1 – Full set of knowledge in the KNM.*

The understanding process of the user's utterance depends on many conditions, and the adequate content of the prompts can be more relevant than other technical details. Once the user provides an input to the system, the understanding of this input is determined by the content of the semantic model. For the two tasks above, a technical collaboration by IMAGINE developers or technical staff is not required. The tools maintaining the KNM can be managed by the application providers.

The main information stored in the KNM is:

- **Application:** The KNM can support various models. Each model belongs to an application. The applications can be maintained with their own tools.
- **IMAGINE Type Format & IMAGINE Type Synonyms:** The IMAGINE Types are a set of values that cover all the necessary semantics referring to the interfaced application, in order to tag the LFs and the user's input.
- **Prompt, Text and Language**: All text in the system is stored in a common structure. This allows IMAGINE to be multilingual, changing the prompt language or the trace messages from one language to another by just updating a variable in the imagine.ini file.
- **Forms:** Depending on the way that the functionality is deployed in the interfaced application, one common LF could be required to call a set of different methods in the interfaced application. This process is used to get values required for the last step in the process of executing a LF. Typically, those values are obtained from the so-called Hidden Vars in HTML forms, this being the reason for the file.
- **Parameters:** Parameters are in fact included as part of the LF they refer to. The complexity of LFs and parameters tends to suggest that they should be separated out into different repositories. However, the close relationship between LFs and their parameters makes it preferable to use the same tool for their maintenance. See LF maintenance screen in sections above.

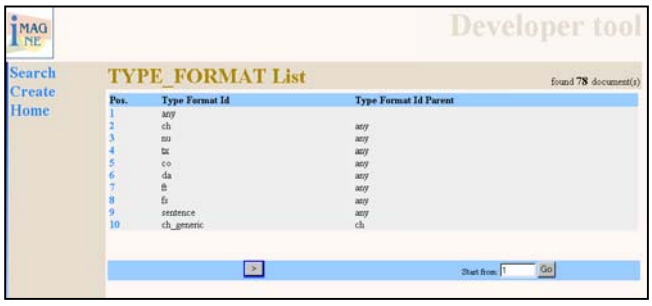



*Fig 3.3.3.1.2 – Two screens of KNM tool.*

### 3.3.3.2 English Speech Tool (EST)

The EST is both a speech recogniser and text-to-speech converter. The speech recognition engine within EST requires a *recognition package* to define the utterances that are recognisable by EST. The EST Grammar WorkBench (EST/GW) is a tool for defining and modifying recognition packages for EST.

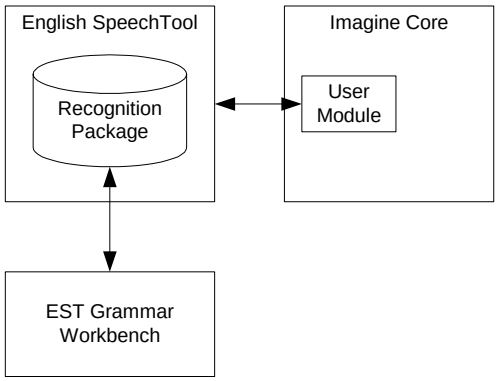



*Fig 3.3.3.2.1 – Grammar workbench position.*

The EST/GW interface consists of two main components. The grammar window on the left of the divider presents the grammars that exist in a recognition package. The specification window on the right of the divider presents the details about a selected grammar.

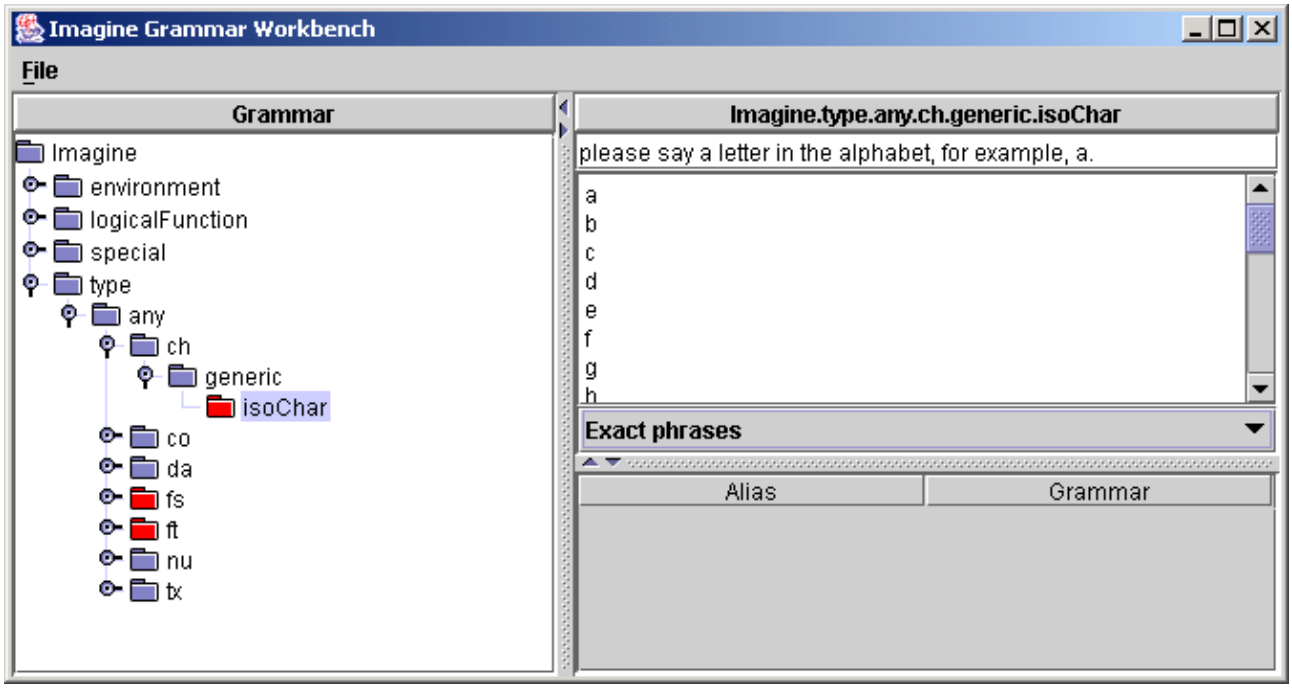



*Fig 3.3.3.2.2 – Grammar Workbench interface.*

EST/GW generates a *.GC* file to store all the details about a recognition package. This is a recognition package definition file which is only used by EST/GW. The compiler within EST/GW generates the recognition package that is used by EST.

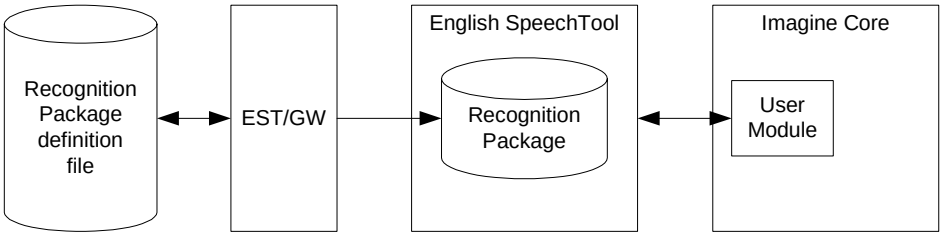



*Fig 3.3.3.2.3 – Recognition package usage.*

| **Results in deliverables** |
|---|
| Not defined. Software available. |

### 3.3.4 T34. E-commerce secure transactions

Security plays an important role in e-Commerce systems if they are to be successful and accepted by their users in keeping information and transactions confidential.

This deliverable for this task is concerned with the technologies involved in communicating data securely, paying particular attention to e-Commerce aspect of ***IMAGINE***, allowing for an automated, yet non-restrictive location access and keeping its usage simple for its users. For this reason, natural language interfaces are recognised as potentially the user-friendliest of all interfaces – interacting with the system by means of voice as they would with a normal sales person or operator. This coupled with voice-based security measures can give the ability for user access by phone whether it's fixed or mobile as well as across the Internet.

Nuance Communications provides a useful set of development tools for the production of such a system. The Nuance Verifier allows users to be authenticated by their spoken voice. This does alleviate a lot of the hassle of remembering awkward passwords which could possibly be stolen or guessed as the user themselves can only gain access to the a system which supports the Verifier.

Sun Microsystems Java 2 Micro Edition is a small footprint platform specifically targeted towards mobile phones and portable devices to allow for the creation of applications for these devices. Although J2ME is not yet fully mature, it shows potential in supporting HTTPS and SSL for the ability of sending data securely across a wireless medium. This means that future development of applications will allow the user to access the system on the move providing full security as well as the initial access to the system

***IMAGINE*** is required to give the confidence to the user that the transactions they are participating in and the information which is being exchanged is done under secure conditions.

A demonstrator using Nuance components has been built outside of the core IMAGINE System as an experimental test to gain an overview of the competence and applicability of Voice verification and authentication.

***IMAGINE*** could in the future be further developed as part of a Web services infrastructure – taking this into account a short overview of security for Web services is provided for future versions and development of ***IMAGINE***. An assessment of how the audio output would be streamed to state-of-the-art mobile communication clients within a Web services infrastructure has begun within the duration of the project. The security requirements for this are also evaluated as part of this.

| **Results in deliverables** |
|---|
| D33 – Security for Transactions |

## 3.4 WP4. Pilots validation

### 3.4.1 T41. Integration & support

The main result of the WP2 and WP3 is the software supporting the IMAGINE functionality. The WP4 activities are focused on the tasks related with the integration and the validation of the prototypes.

The theme of the task 4.1 is the integration activities. Integration activity is the integration process of the modules delivered by each of the developer partners. It has been considered of a very high value, to have a common set of tests for the full consortium, especially for the partners responsible of developments and this is included in this document also.

The high degree of integration requires that every module work fine, because the output of one component is the input for the next, therefore co-ordinated tests must be performed. It has been provided two resources for the adequate and integrated tests:

- Common log file, that can be accessed by every components in the system.
- Common test sets, that must be performed by every developer partner.

Previously to each prototype delivery, a new version of the corresponding deliverable D41 or D42 has been released. They contained always the result of the tests with the prototype in use. Each of the mentioned reports has included the next sections as interesting information around the prototype:

- **Summary of integration**. Summary and explanation of each prototype delivered, with the timing and components delivery dates.
- **Installation**. The installation steps and the known issues are the focus of this section.
- **Functionality supported**. This chapter clarified the state of the functionality covered by the prototype.
- **IMAGINE user guide**. Explanation in the most simple way, without technical details, of the IMAGINE coverage from different points of view: linguistic structures, semantic understanding, etc.
- **Test cases**. Every prototype has been tested with a common set of sentences (with minor updates along the different versions). This grants that we check always the full functionality, noticing if previous working cases stop working.

In the following figure, it is shown the temporal delivery of the prototypes.

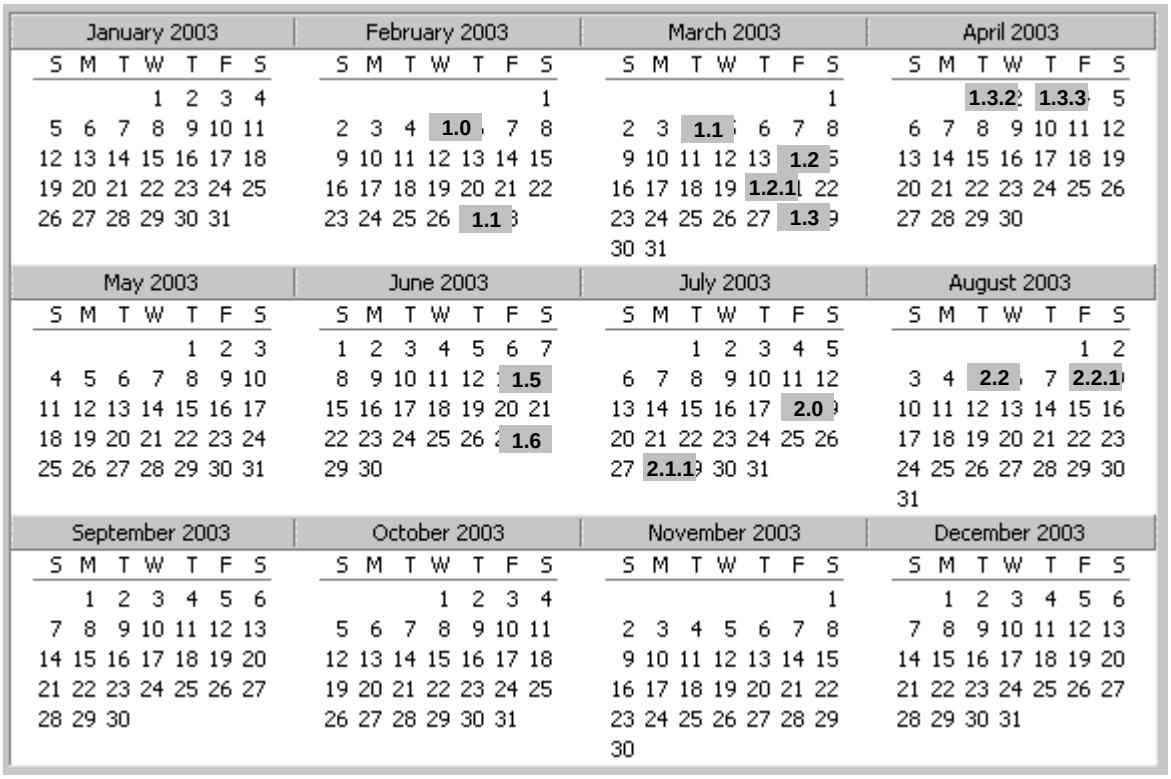


*Fig 3.4.1.1 – Prototypes delivery*

| Results in deliverables |
|---|
| D41 – First prototype delivery report |
| D42 – Second prototype delivery report |

### 3.4.2 T42. Evaluation and assessment

T42 consisted of completing validation tasks and usability evaluations on the English and Spanish prototypes, and also the production of training materials. To begin with, a validation plan was drawn up that would be referenced by both user partners in order to carry out evaluations of the prototypes produced for each site. A brief synopsis follows:

**Receive Prototype 1**

- Conduct heuristic evaluation and validation testing with users
- Identify problems that must be dealt with
- Outline what the system must be like before we launch it for 'real word' testing

**Receive Prototype 2**

- Pilot the system with two or three people using a checklist to ensure it meets our success criteria as stated
- Until the system meets stated requirements it will not be launched on real RNIB neither IDM customers
- If it does not meet requirements, we will say what needs to be changed
- If such changes are made the system will be piloted again and so on iteratively until it is ready for launch
- Conduct validation testing with prototype 2

The plan, as it is set out above, would allow for as many problems as possible with the first few prototypes released to be ironed out in preparation for less-constrained user testing and real-world testing towards the end of the project.

During the planning stage, it was concluded that if maintenance tools were going to be provided by the developers so that service providers would be able to update the IMAGINE system themselves, then validation testing on these tools should also be included in the plan.

The original validation timeline was drawn up thus:

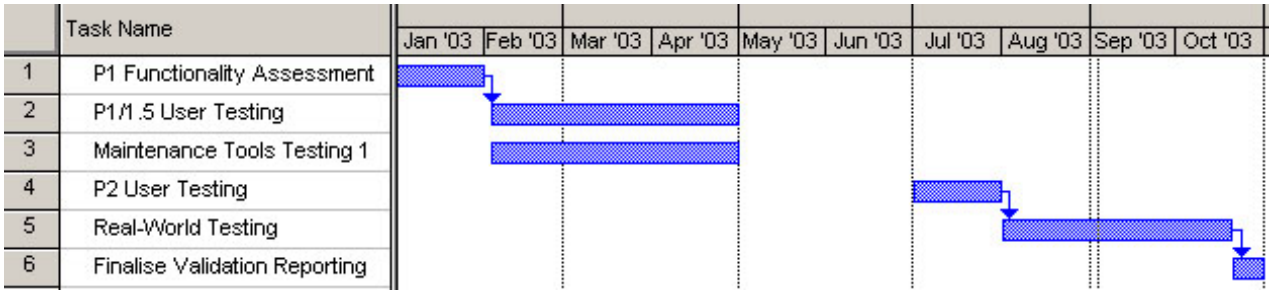



*Figure 3.4.2.1 Original Validation Timeline*

The delays further on in the project meant that when the plan was drawn up, it had been decided that the first prototype would not be ready for user testing until the beginning of January. An application for an

extension to the project until the end of October had also been agreed by the Consortium at this stage, hence the end date shown.

In order to validate the maintenance tools to be provided, documentation supporting the tools would be included as part of the training materials document, D45. As a voice interface accessible via the telephone, anyone wishing to access a particular service interfaced by IMAGINE, e.g. to order RNIB products, should be able to do so by just ringing the number and being guided through the process by the system itself, designed to be as intuitive as possible for novice users. Therefore, no end user training materials are required for such a system and the D45 provides a synopsis of why this is the case, but points out that "online" help is provided in order to guide the user further if the need arises.

Therefore, the training materials document has provided a platform for the maintenance and update of IMAGINE systems. In order to produce the documentation surrounding updates and configuration, etc. a set of maintenance scenarios were supplied by the user partners to the developers. RNIB provided a set of scenarios detailing future changes that will be made to the Online Shop, IDM providing scenarios that would apply to Viapolis.com.

Using the scenarios as a reference, guides to updating the IMAGINE modules were documented and are included in D45. They include a guide to using the KNM Maintenance Tool, a guide to updating the Answer Adaptor Module, a reference to configuring the English Speech Tool and a guide to updating prompts for the English Speech Tool.

Certain aspects of the system do not allow for the maintenance process to be fully controlled by the service provider. Corpus updates must be accompanied by developer assistance. A workbench for the user to adapt the English Speech Tool grammar was planned to be available, but due to changes in the way the EST now "talks" to the IMAGINE core, the workbench is not applicable at present. It is possible that an adapted workbench could be available for future developments of the system. The Spanish Speech Tool has been developed by and will be maintained by TID. Therefore, any required changes to the SST will need to be reported to and carried out by the developers.

Along with the training materials documentation, a description of the process for developing an IMAGINE interface for other systems has been provided. The main structure of this process is outlined below.

1. Initial Analysis
2. User Needs Analysis
3. Selection of Integration Solution
4. Acquisition of the Application Knowledge
5. Acquisition of Linguistic Information
6. Acquisition of Semantic Application Knowledge
7. Integration with the Application
8. Test & Validation
9. Maintenance

Throughout the validation phase, updates were made to the plan given the delays in releasing the first prototypes, followed by problems in producing prototypes that could successfully be evaluated according to the plan. For both the English and Spanish systems, the validation plan was revised and the final version is shown below.

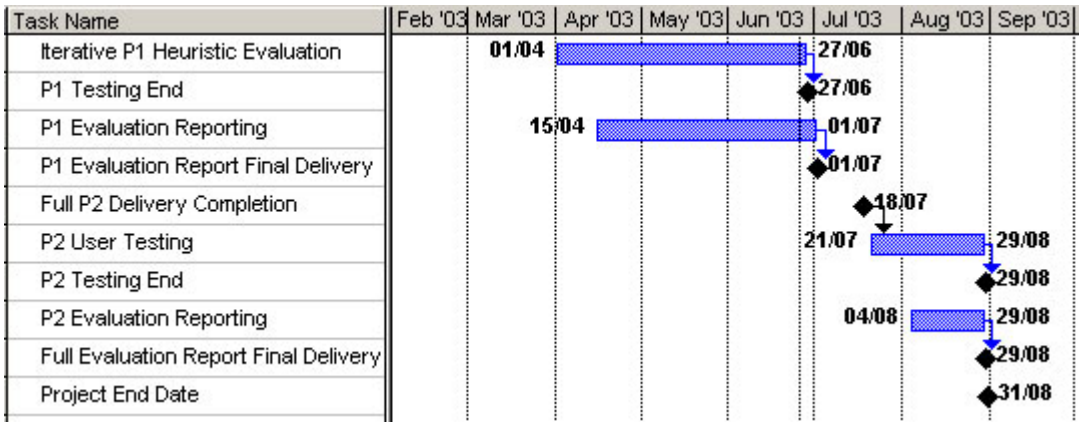



*Figure 3.4.2.2 Final Validation Timeline*

However, further delays were experienced with both systems, and a synopsis of the English system evaluations follows.

From P1.2 onwards until the release of the first prototype 2, each version of prototype 1 was functionality tested by evaluators at RNIB. Each release saw an increasing amount of functionality and a better system to interact with, however the functionality assessments carried out in the hope of progressing to the next stage of the validation plan were unsuccessful. The assessments showed that any users exposed to the system would more than likely fail most tasks attempted.

During this phase of prototype 1 testing, it was decided that real-world testing of the planned prototype 2 would not be carried out due to the decreasing time available until the end of the project. It was also decided that as the system at this stage was not usable enough for heuristic evaluations to take place, user focus group testing could certainly not be considered until the release of prototype 2.

The release of prototype 2 was again met with further functionality assessments and the results concluded that the system was still unable to meet a user's needs enough to warrant the start of the heuristic evaluation phase. Further updates made on the basis of these results improved the system to the extent that its assessment allowed for the next phase to begin. It was here though that user focus group testing was eliminated from the validation plan due to the shortened time now available up to the end of the project.

As a result of the final functionality assessment carried out and the heuristic evaluations that followed, the system in its present state has been deemed unusable. A few technical problems are apparent and it is recommended that these types of problems be addressed and solved on any further development of this system. It is stated in the D43 document that these problems may have contributed to some of the usability problems identified by the heuristic evaluations, however, many more usability problems would have to be addressed to allow this system to be a workable and usable system for RNIB customers.

The heuristic evaluations were carried out by five computer system/usability experts, including one who is visually impaired. The results were extremely consistent and most of the problems were identified by more than one of the evaluators. A usability expert applied severity ratings to the results and concluded that the system could certainly not be deemed usable in its present state due to the amount of major and catastrophic usability problems identified.

The conclusions to the evaluation phase recommend that on any further development of the system, attention be paid to the technical problems and all of the major and catastrophic usability problems identified. It is also recommended that further research and development is made into certain areas that need to be addressed when applying such a system as an additional interface, which sits on top of and interacts with a web based application.

A synopsis of the Spanish evaluation follows.

The report D44 presents the results of the validation activities and usability studies carried out by IDM on the prototype IMAGINE interface to the Viapolis application. The report explains the results of all the procedures and usability studies carried out to the end of the project.

The results gathered during the demonstration phase have been collected along a period of around two months, and the general results are provided in this document. It is important to make some remarks: due to the delays of the prototypes it was impossible to have a prototype useful to test in a real context. The consortium decided then to test the prototypes using internal personnel at IDM. Following the usability studies some errors were solved by the developers, but not enough to test in a real context with real users. Some errors exist when using the speech tool that make it difficult to test the prototype in a real user context.

IDM tested the prototype using internal users at IDM. The users are people of both sexes and people who have Spanish/Castilian as a mother language.

In general, with improvements to the speech tool in place, it could be enough to test IMAGINE using real users. The tests carried out with the IDM users show us that the system works quite well and improvements would be seen by making some minors changes to the system.

| **Results in deliverables** |
|---|
| D43 – Validation report & Usability studies site 1 |
| D44 – Validation report & Usability studies site 2 |
| D45 – Training materials |

### 3.4.3 T43. NL usability studies

Usability is the measure and thus a major criterion for the success of any software project, thus also for IMAGINE. Usability is a scientific topic that involves methods and concepts from many disciplines among others from cognitive science, software ergonomics. There is a rich literature on software usability, ISO norms, standards, though less so for the specialised software IMAGINE has produced, speech enabled interfaces. A usability study, thus, can use a lot of available resources, concepts, methodologies, a wealth of papers and investigations, standard questionnaires, studies etc.. It was the task of WP4 to be done by IAI to provide a description that gives a guideline to the choice of criteria, methods for assessing and evaluating the usability of IMAGINE.

The major point of the usability study concerning IMAGINE was that it has to compare the usability of the system with classical GUI access and evaluate whether the speech enabled interface that is provided by IMAGINE has any (positive) impact on the usability.

The document gave some definition of what usability is, what the criteria are for assessing usability. The basis of this discussion was a paper that had been produced for the MELISSA project. In MELISSA there was an extensive usability study that tried to evaluate natural language interfaces in comparison to other user interfaces. This study was reviewed and built the foil for the discussion about how to adapt general findings about software usability to the IMAGINE scenario.

Another question was how to make the usability study, which methods to apply, which of them are best suited for the IMAGINE scenario.

A last section investigated the specificity of a usability study concerning speech products by reviewing the literature on the usability of speech systems. These more general considerations about usability were meant to be followed by the production of the usability study of the two users applications.

| **Results in deliverables** |
|---|
| D46 – Usability study (deliverable not committed in the contract) |

## 3.5 WP5. Exploitation & Dissemination

### 3.5.1 T51. Initial market studies

This task aims to position IMAGINE in the market from the very beginning stages of the project. As result of this study, the main output has been the *D51 – Initial market studies*. The structure of such study has been divided in two major sections:

a) The analysis of the market structure and context, identifying and analysing the key technologies, relationships that condition the market structure and actors.

b) The definition of IMAGINE position based on a crossed analysis of the above pivotal elements.

The determination of the IMAGINE context is an activity of high importance for the further derivation of its positioning at the market. The context has been derived from the confluence and analysis of three major aspects:

- ***The technological context:*** A study of the core technologies underlying in the Voice enabled solution, has identified as major players in this field. IMAGINE is positioned somewhere at the confluence of three key technologies:
  - *Mobile speech technology,* supports interaction with an application via a standard cell phone, and automates the call centre traditionally used to support this mode of interaction. It refers specifically to telephony/Interactive voice response platforms supporting automated voice interaction. These systems may include the following core technologies: automatic speech recognition (ASR), text to Speech Synthesis (TSS) and Automatic Speaker Identification.
  - *Natural Language technology*. This technology allows users to interact with a system, using speech or written language, in a natural, conversational manner.
  - *Mutli-modal interface design*. These interfaces support interaction with a system through a range of devices and channels; the user interface is multi-faceted.

  The investigation has concluded with a map showing the distribution of companies and products defining the technological base, and providing an understanding of the positioning of IMAGINE within the related technology markets, as indicated in the following figure:

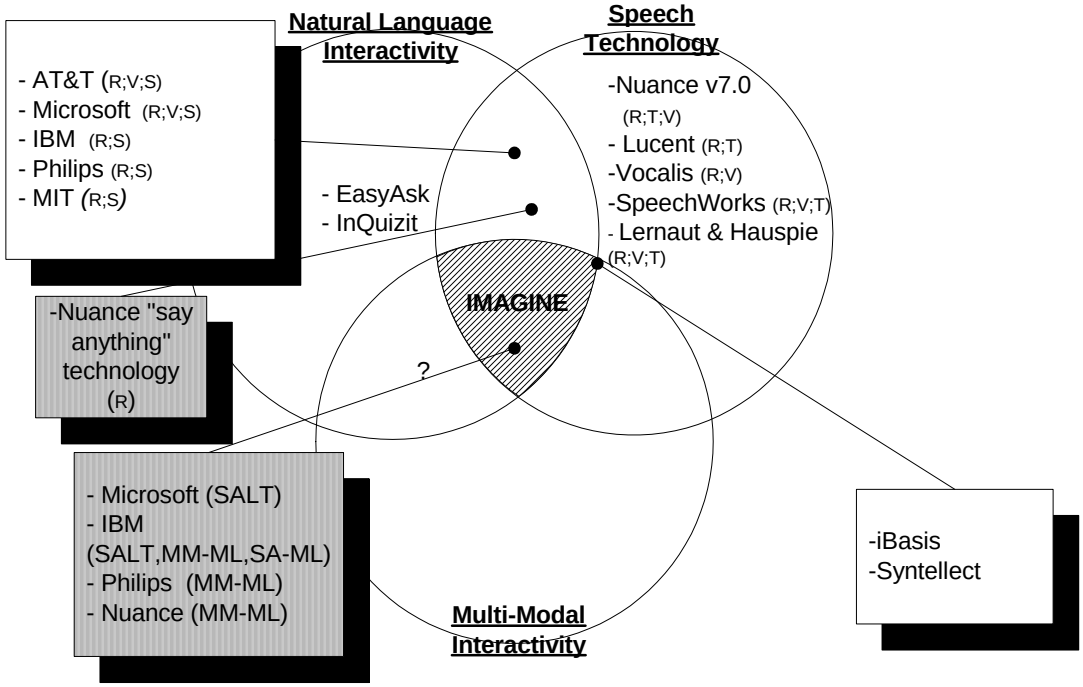



*Fig 3.5.1.1 – IMAGINE position at the market.*

- ***The market structure:*** A full and complete description of the market context requires careful consideration of a range of interconnected factors, in addition to detailing the enabling technologies. Specifically, the market context for IMAGINE is defined by the following structural elements:
  - The technology basis is the *wise integration of key and innovative core technologies*: Speech Processing, Natural Language Understanding, Mobile devices accessing to the Web.
  - The *underlying enable technology* is the wireless Web, which currently is improving its capabilities.
  - The *key players* are diverse ranging from core technology providers, integrators, service providers, e-Business services providers, mobile technology providers and operators.
  - The producers are launching *exploitation initiatives* of different nature: solid software houses investing in developing and consolidation of these novel technologies; new software houses have arose with people specialised in the technologies; Telecommunication houses are expanding their business through the establishment of agreements with technology providers and speech enable developers. Cooperation initiatives play an important role, as a way to exploit a product that requires the convergence of so different technologies.
  - The *role of standardisation* is a factor to consider as a way to evolve with the technology, and thus be open and interoperable with other complementary products and applications.
- ***The Users:*** It is commonly known that the success of every product at the market is directly determined by the users. This is more important in the case of IMAGINE solution, corresponding it with an interface to provide added value to the users by means of an easy-to use interface. This is why Users are pivotal for the definition of the product: users are demanding voice enabled services with a set of requirements that IMAGINE must consider as the reference point to design the solution to be finally presented at the market.

  In cooperation with T11 task (User Requirements), the following types of users have been identified:
  - *End Users,* those users making use of the voice interface. They correspond with the general public; a relevant characterisation of these users is that they are known or unknown, untutored; sometimes users are hesitant, impatient, lacking clarity, etc. The usability factor will be a key for the acceptance of the system by these users.
  - *Service Providers,* as the organisation providing e-Business services to be integrated with the IMAGINE voice interface to provide added value to their customers, the End-Users. Service Providers are demanding voice and multimodal interfaces to be plugged-in with their applications whatever the architecture of these applications is.
  - *Intermediaries:* these are the organisations in charge of integrating the application services with the IMAGINE system. Intermediaries demand a platform to easily deploy the voice interface to the e-Business applications.

### *IMAGINE positioning at the market:*

In a second phase, the Market Analysis has determined the key factors of IMAGINE solution, evaluating these factors, and comparing them with the direct competition of IMAGINE.

Since IMAGINE combines speech technology, multi-modal access and natural language interactivity, the system will compete against products that integrate the base technologies.

The *competition* has been identified as the set of existing systems combine telephony control, speech recognition and synthesis to support interactive voice-based interfaces. Several commercial systems also combine speech technology with natural language interactivity, thereby offering speech interfaces with conversational capabilities. The key features of IMAGINE are specifically compared with those offered by the following commercial systems.

- *IBM* produce an integrated solution *DirectTalk,* that combines their popular ViaVoice natural language recognition system with telephony functionality, dialogue management and text-to-speech capabilities.
- *Philip's integrated IVR system SpeechMania* features both Natural language understanding and dialogue management components.
- *Nuance* offers integration with other enabling technologies as a core feature of its commercially available recognition engine. Specifically, the recognition system provides native telephony support, fully integrated voice verification/validation, tight integration with its own text-to-speech system, and dialog management.

- The *SpeechWorks system*, called *SpeechSite*, is specifically designed to support automated call-attendant.

The analysis has identified the following key characteristics that provide a differentiation to IMAGINE solution with respect to the products already existing in the market.

- Comprehensive support for multi-modal interactivity with support for a wide range of devices.
- Sophisticated dialog management supporting a mixed-initiative conversational style.
- High-level support for Web and WAP interfacing, with a very low impact on existing infrastructure and minimal additional resource requirements.

Finally the study has concluded with a first definition to *IMAGINE products and services*:

- A *voice-enabled natural interface* to e-Business applications, tested at two specific e-business organisations at sites in Spain and the UK (thus using Spanish and English respectively).
- A *set of tools* to help 3rd party developers to integrate the voice Natural Language interface to their applications.

A set of *additional services* marketed in an on-demand and ad-hoc basis, intended to support e-Business organisations with a final solution completely integrated in their applications, or to provide support to integrators in the process of integrating the voice enabled interface with the e-Business applications.

| **Results in deliverables** |
|---|
| D51 – Initial market studies |

### 3.5.2 T52. Exploitation Plans

Having as final objective the definition of the IMAGINE Exploitation plan, this task has started with an initial definition of the business model to further refine during the project time. Various draft has been produced during the project time gathering all the aspects related with the Exploitation Plan, as baseline for the work under this direction.

The latest tests done with the IMAGINE prototype has clarified to the consortium the state of the IMAGINE prototype. The situation shows that most of the problems stated can be considered a tuning problem or the limit of the state-of-the-art in some technologies. In the tests also a few issues points out to the own IMAGINE paradigm, what are more worrisome. With this context, the general approach to exploitation is that the IMAGINE concept appears to have some opportunities, and that no too much effort is required to get a pre-commercial system.

The objectives of the exploitation plan are focused in finding out the references, the markets, the plans, etc that IMAGINE consortium should follow to reach the reasonable space it can find in the current a next future market.

In this sense, the plan to exploit IMAGINE is not a decisive go to market quickly but to do a third loop in the creation of the product, and then face a more commercial phase. The end of the IMAGINE project time does not allow to finish the pending activities. The real risk with this in the edge system, recommend us to find alternative funding resources for a further, and expected last, product definition and packaging.

#### 3.5.2.1 IMAGINE Product and Services

According with the project objectives we distinguish two types of products that are addressed to different targets:

- A Voice-enabled Natural Language Interface (VNLI) to e-Business applications, tested at two specific e-business organisations at sites in Spain and the UK (thus using Spanish and English respectively).
- A set of Development Tools (DT) to help 3rd party developers to integrate the VNLI to their applications.

The runtime and development system will be marketed separately, as the target audiences could be different. There will thus be two product definitions that are given below.

### 3.5.2.2 SWOT analysis

As result of the SWOT analysis done in the document *D54 – Exploitation plan*, there is a set of features that make IMAGINE different of other alternatives in the NLP scenario. Considered as a general approach to the interfacing of systems especially in Internet, it could be the alternative to products based in VoiceXML, giving a more simple and flexible approach. In the other side, ad-hoc developments for concrete systems can work better than the adaptation of IMAGINE, but are by sure more expensive and the inversion is not reusable in another systems. So IMAGINE seems to be in the middle path between two bounds:

- The general approach of VoiceXML, being a very well known and supported standard, but with the difficulty of generating scripts by non-expert staff and the limitations plus complexity of such scripts.
- The better performance of ad-hoc tailored systems, getting the most that technology can provide. With a complete private solution, with the disadvantages that such approach could have.

Considering all previous factors, IMAGINE should invest the next efforts in getting advantage of the user validation tests in order to fix or improve the current behaviour. It should try to find the market niche at the middle point between those two kind of applications, strengthening the simplicity for adaptation to new applications. This simple adaptation will also be the alternative for ad-hoc systems. It should not be forgot to provide the adequate resources that make the current state-of-the-art technologies, mainly in the speech recognition field, to work in the optimum way with IMAGINE.

### 3.5.2.3 Exploitation phases

IMAGINE has defined a new concept not covered by other solutions in the market. At the end of the project time, it has not reached the state of a commercial tool, although the way to reach this state should not be very large either.

The conditions stated within the consortium to go into a market strategy require having at least one real reference for IMAGINE. It was expected that RNIB was such reference, but according to the current evaluation tests, and the lack of resources to finish the IMAGINE tuning and latest updates, this reference will not be possible by the moment. In the case of Viapolis is more difficult, because the own company disappeared months ago, and such prototype becomes a test rather than a real application.

The consortium considers that the risk to invest their own resources is high, and although the concept is promising, it will try to find alternative funding resources to complete the tasks and to get some real reference. The expected process is shown in the following figure:

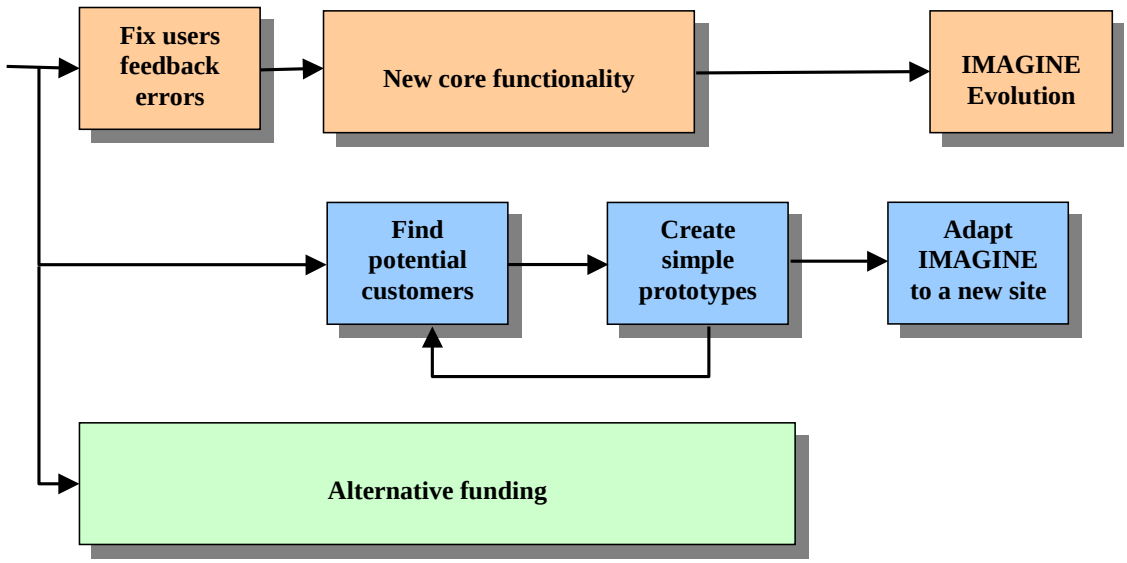



*Fig. 3.5.2.1.1 – Exploitation plan phases*

| Results in deliverables |
|---|
| D54 – Exploitation plan |
| D55 – Technology and implementation plan |

### 3.5.3 T53. Dissemination

Dissemination and Communication about the **IMAGINE** Project has been ongoing throughout the project and has primarily been based on each individual consortium members internal strategic objectives, expertise and opportunities. The *D53 – Dissemination and use plan* deliverable has explained the goals in dissemination for the project. Most of them are achieved during the project time:

- Brochure with the IMAGINE objectives and the partners profiles and interest in the project.
- A project Web site was available within the first quarter of the project and this provides access to the basic background, objectives and aims of the project. A text version of the site has been included also for an easy access to blind or partially sighted users. The site is accessible in English, Spanish and German.
- Conferences and workshops have been attended where specific presentations and demonstrations of the **IMAGINE** System have been made. By partners, the congress collaborations are:
  - IAI – COLING paper.
  - RNIB – AAATE paper.
  - SAGE – SEPLN projects & demonstrations paper.
- The so-called dummy application is being packaged during the last days of the project. This package will allow third party people, to install IMAGINE in their own installations and apply it to get information from very simple static Web sites, similar to the IMAGINE Web site.
- Dissemination material has been gathered in the Web site for a common dissemination image of the project.

We have also engaged in more targeted 'knowledge transfer activity' in appropriate market sectors for e-commerce and B2C areas. This has been based on attempting to realise the potential of integrated Exploitation and Dissemination activity. The result of some of this activity have led to proposals for further collaborative work with other interested technology providers and end-users in a variety of sectors.

A formal communication committee was organised at the start of the project and the dissemination of the project results will be maintained for a considerable period after the end date of the project.

| **Results in deliverables** |
|---|
| D52 – Project presentation |
| D53 – Dissemination and use plan |
| D56 – Final public report |

## 3.6 WP6. Management

### 3.6.1 T61. Administration

This task includes the overall administration of the project: contracting, legal agreements (including IPR sharing and protection), cost statements production and management, etc. The following activities have been performed during the project period.

- Preparation of the consortium agreement.
- Preparation of templates for various reporting documents: quarterly reports, progress reports, meeting agendas, meeting minutes, cost statements, …
- Gather of the different partner contributions to the periodical reports (quarterly reports, progress reports, cost statements, …) and delivery to the commission.
- Preparation and delivery of the project documentation (mainly deliverables) during the project life cycle.

- Intermediary between the partners and the commission for the questions arisen during the project: extension, change of a partner (Viapolis -> IDM), costs support, etc
- General administration activities.

| Results in deliverables |
|---|
| No deliverables for this task. |

### 3.6.2 T62. Technical mgmt, QA, CMB, HD & rep

The technical management of the project requires a set of activities that guarantee the adequate development of the project. It has to assure that the activities are correctly coordinated and that have the resources and guidelines adequate to success. In order to help in this issue, various activities has been performed during the project time:

- XX consortium meetings (see below section). Each of those meetings has required a preparation phase and a post meeting work to generate the adequate documentation: minutes, to do action list, etc.
- Concrete documents: annual report, document for the extension of the project, new contract due partner change (Viapolis -> IDM), …
- Definition of a set of measures to the adequate partners interrelation in the various activities to be performed during the project: technical, management, etc. Those measures are summarised in the document *D61 – Quality assurance guidelines*.
- Generation of this report *D62 – Final report*.
- Maintenance of an internal Web site for partner communication.

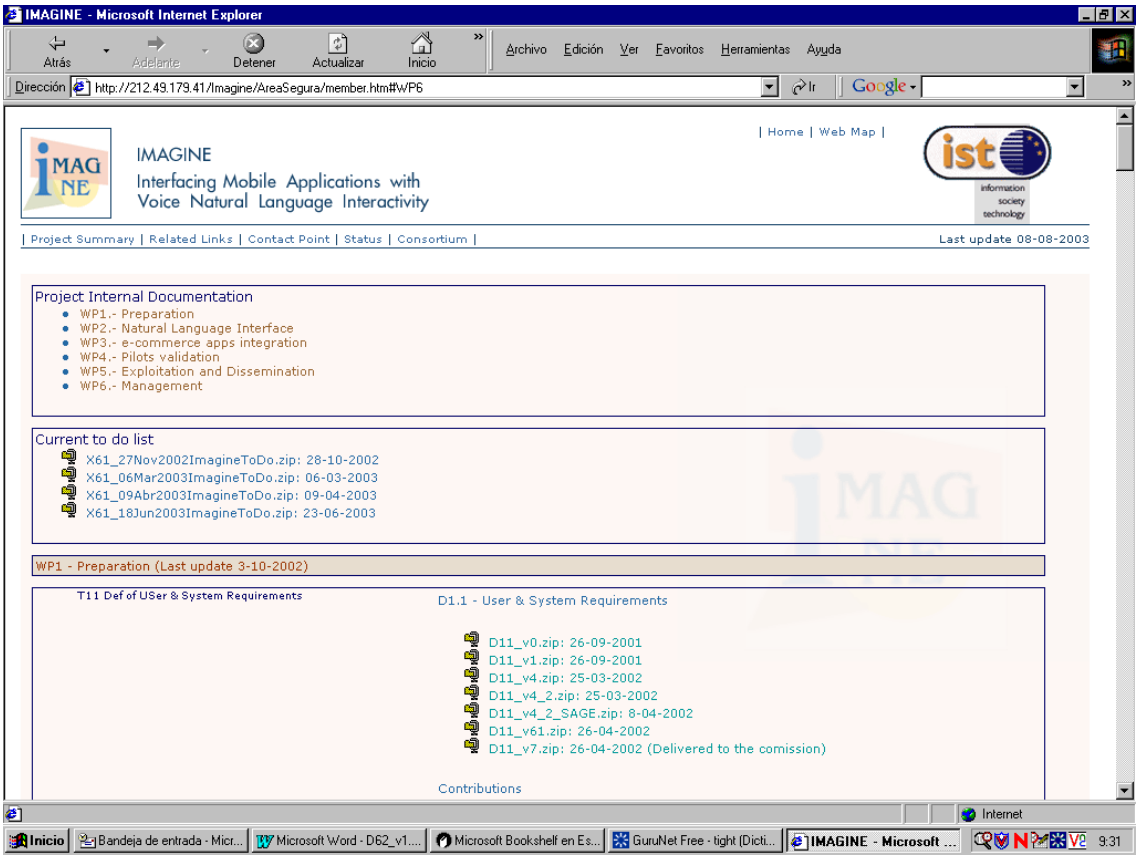



*Fig 3.6.2.1 – Members IMAGINE Web site.*

| **Results in deliverables** |
| --- |
| D61 – Quality assurance guidelines |
| D62 – Final report |

# 4. Management Information

## 4.1 Statement of progress

The fit of the project with the plan has been a general delay of near two months in various activities: user requirements, lingware preparation, prototype deliveries, etc. This delay has affected the validation activities. This has been the reason to ask for the extension of three months not accepted by the commission. In the opposite sense, another activities as the task 23 has been started and finished before the expected plan.

At the end of the project, all the expected activities to be done have been finished, although some of then with such delay of near two months.

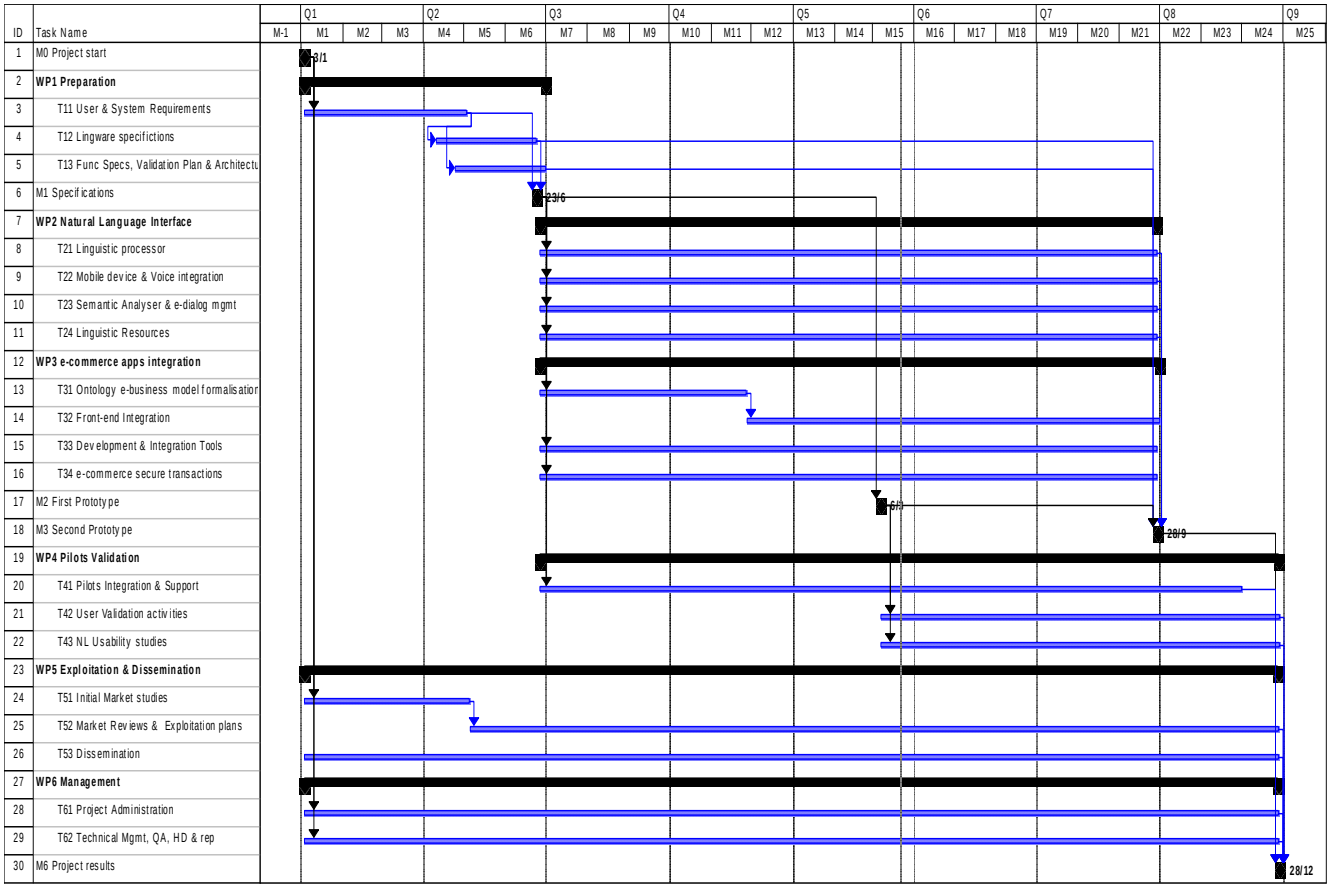



*Fig 4.1.1 – Project work plan.*

## 4.2 Resources usage

Adding all the resources stated in the previous quarterly and progress reports, the complete usage of resources is the following.

| Cumulative Project effort in person-months | | | | |
|---|---|---|---|---|
| Originally | | | Actually | |
| Plan. | Cumul. | Period | Spent | Cumul. |
| 18,8 | 18,8 | Sep 01 – Nov 01 | 17,22 | 17,22 |
| 15,6 | 34,5 | Dec 01 – Feb 02 | 23,97 | 41,19 |
| 44,1 | 78,6 | Mar 02 – May 02 | 29,45 | 70,64 |
| 33,1 | 111,7 | Jun 02 – Aug 02 | 38,66 | 109,30 |
| 36,4 | 148,0 | Sep 02 – Nov 02 | 39,13 | 148,43 |
| 44,6 | 192,6 | Dec 02 – Feb 03 | 33,16 | 181,59 |
| 44,6 | 237,2 | Mar 03 – May 03 | 41,89 | 223,48 |
| 19,4 | 256,6 | Jun 03 – Aug 03 | 45,06 | 268,5 |

*Fig 4.2.1 – Resources usage by period.*

## 4.3 Deliverables Summary

Following the complete list of deliverables with the original completion date and the real delivery date.

| Deliverable | Original completion date | Delivered |
|---|---|---|
| D11 – User & System requirements | MO4 = Dec 2001 | 26-Apr-2002 |
| D12 – Lingware specifications | MO6 = Feb 2002 | 25-Apr-2002 |
| D13 – System design | MO6 = Feb 2002 | 26-Apr-2002 |
| D21 – First version of NL components | M15 = Nov 2002 | 24-Feb-2003 |
| D22 – Second version of NL components | M12 = Jul 2003 | 31-Jun-2003 |
| D23 – Dialog & interaction model | M12 = Aug 2002 | 14-Mar-2003 |
| D31 – First version of e-commerce components | M12 = Nov 2002 | 28-Feb-2003 |
| D32 – Second version of e-commerce components | M21 = Jun 2003 | 31-Jun-2003 |
| D33 – Security for transactions | M12 = Aug 2002 | 13-Sep-2002 |
| D34 – Question answering | - | 31-Aug-2003 |
| D41 – First prototype integration delivery report | M16 = Dec 2002 | 04-Apr-2003 |
| D42 – Second prototype integration delivery report | M22 = Jun 2002 | 31-Aug-2003 |
| D43 – Validation report & usability studies site 1 | M24 = Aug 2003 | 31-Aug-2003 |
| D44 – Validation report & usability studies site 2 | M16 = Aug 2003 | 23-Oct-2003 |
| D45 – Training materials | M15 = Nov 2002 | 31-Aug-2003 |
| D46 – Usability studies | - | 23-Aug-2003 |
| D51 – Initial market studies | MO6 = Feb 2002 | 02-Apr-2002 |
| D52 – Project presentation | M03 = Nov 2001 | 15-Jan-2002 |
| D53 – Dissemination & use plan | MO6 = Feb 2001 | 05-Mar-2002 |
| D54 – Exploitation plan | M24 = Aug 2003 | 20 Oct-2003 |
| D55 – Technology implementation plan | M24 = Aug 2003 | 24 Oct-2003 |
| D56 – Final public report | M24 = Aug 2003 | 8 Oct-2003 |
| D61 – Quality assurance guidelines | MO4 = Dec 2001 | 25-Feb-2002 |
| D62 – Final report | M24 = Aug 2003 | 8-Oct-2003 |

*Fig 4.3.1 – The full list of IMAGINE deliverables.*

## 4.4 Problems encountered

Although the main objectives of the project has been achieved and that problems are back of us, those main difficulties were the following:

- The lack of collaboration of Viapolis from the beginning stages of the project has delivered in a less evolution of the Spanish prototype in relation with the English system.
- The integration of the Spanish speech tool with the IMAGINE core has been a very complex, time consuming and hard to coordinate task. It has spent more resources than expected.

- The interchange format between LPM and SAM has been a huge effort task of agreement that has needed months to be solved. This has been a reason of the delay in other related tasks.
- Requirement for Voice Card Hardwire to support barge-in functionality. Problems with the hardware selected for the English speech tool.
- The delays earlier on in the project, and delays in the delivery schedule of prototypes, has created a situation where time for validation has been seriously cut down.
- The delay in the following of the plan in different tasks by different partners, in a so integrated project, has been the reason of prototypes delays.
- TID has had resource assigned problems, considering that the resources allocated for their task are insufficient.

## 4.5 Project Meetings

Following, we give the list of the general project meetings with the main topics description.

| *Date* | *Issue* | *Place* | *Description* |
|---|---|---|---|
| 6-7/09/2001 | Kick-off meeting | SAGE (Madrid) | General Consortium meeting: Its main objectives were to organise the cooperation between the consortium members, to coordinate the early management aspects of the work, and to launch the activities of the initial tasks of the project: T11, T13, T51 and T53. |
| 15/10/2001 | User Requirements | Viapolis (Madrid) | Bilateral meeting among Viapolis and SAGE to clarify aspects regarding User Requirements and the Pilot site (integration of the NLI with Viapolis application). |
| 30/10/2001 | Technical Meeting | TID (Madrid) | Bilateral meeting among TID and SAGE to identify the key aspects for integration of Voice technologies with IMAGINE platform. |
| | User Requirements Planning | RNIB (Peterborough) | Meeting held at RNIB Peterborough with Ilesh Dattani and Paul Wilken of IT Innovation attending.<br>The meeting brought other iSys staff up to speed with the project and a plan was set for the next stage of work. |
| | User Requirements | RNIB (Peterborough) | Meeting between Rachel Pick and two representatives from RNIB Customers Services.<br>The meeting gave an opportunity for iSys to find out whether the Customer Response telephone line could be analysed as a means of producing the user requirements for the Natural Language System. Operatives on the Customer Response line deal with customers ordering products and therefore could potentially provide insight into how a Natural Language version of the RNIB Online Shop could work.<br>As permission was given for a listen-in session to be held, this was arranged.<br>Permission was also given for a number of recording sessions of phone calls to Customer Response to be recorded for analysis. |
| | User Requirements | RNIB (Peterborough) | Meeting held at RNIB Peterborough with Ilesh Dattani and Paul Wilken of IT Innovation attending.<br>This meeting was held prior to the general meeting in Saarbrucken. The meeting covered progress so far and the way forward. |

| | | | |
|---|---|---|---|
| 11-12/12/2001 | Consortium Meeting | IAI (Saarbrucken) | Consortium meeting: Its main objectives are:<br>• To assess the progress and achievements in the first quarter of the project, mainly in what regards User Requirements and Market Analysis.<br>• To reach an agreement in the selection of the two pilot sites.<br>• To discuss details in what regards system architecture and functional specifications.<br>• To organise the tasks to be carried out in the following period. |
| 19/12/2001 | User Requirements | Viapolis (Madrid) | Meeting among Viapolis, IAI and SAGE, to close the details for the final selection of Viapolis application. In the meeting, the input expected from Viapolis in what regards the information to provide about the application and other details required for the completion of D11, were clarified. |
| 06/01/2002<br>22/01/2002 | Deployment of the application | RNIB (Peterborough) | Bilateral meeting RNIB – IT Innovation. This meeting was also attended by four members of the iSys team. The meeting was primarily to discuss the functionality requirements and their feasibility. The machine specifications for the application were also discussed, along with a primary look at how the machine will fit into the Online Shop architecture. |
| 10/01/2002 | Speech integration | IT Innovation | Bilateral meeting between IT Innovation and IBM. |
| 14/03/2002 | iSys project team (RNIB), IT Innovation | Teleconference | A teleconference between the iSys project team and Ilesh Dattani of IT Innovation was held to discuss issues surrounding the use of voice-authentication software. This software has been suggested by IT Innovation as an alternative to an explicit log in procedure to eliminate the need for the customer to remember a username and password. However, a number of points were raised about how this could limit the user-base of the system and affect the way in which we would like the system to work. As the discussion was primarily to decide how to approach these issues within the User Requirements document, it was decided that all options surrounding this procedure (including the use of voice-authentication) should be included. The actual way in which the log in procedure will be implemented will be decided at a later date during more appropriate technical discussions. |
| 17-19/04/2002 | Technical Meeting | SAGE (Madrid) | After the design period, this meeting is focused in confirm the general architectural designs and every minor details to communicate every modules with the central **IMAGINE** component: the dialog handling module. The technical sessions were hold on April 17-18-19th, 2002. |
| 19/04/2002 | Project Committee Meeting | SAGE (Madrid) | It is objective of the meeting also, the review of the current state, the development state of every tasks and the planning for the next period. The Project Committee session was hold on April 19th, 2002. |
| 13/05/2002 | Technical meeting: Voice integration | TID (Madrid) | Clarify binary connection among Spanish Speech Tool and **IMAGINE** |
| 21/06/2002 | LPM output and SAM review. | IAI (Barcelona) | Agreement on the SAM work and clarification of the requirements in the LPM output for SAM input. |
| 04/07/2002 | Corpus and speech tool linguistic model. | Viapolis (Madrid) | TID, Viapolis, SAGE. Viapolis is required to provide adequate input data to TID, in order to generate the language model for the Viapolis application. |
| 24/07/2002 | Validation | RNIB (Peterborough) | A meeting was held on 24th July between Rachel Pick and Helen Graupp of the RNIB project team. This was the initial meeting to discuss the validation planning for T42. The report of this meeting will be sent to the consortium in the near future. |

| | | | |
|---|---|---|---|
| 18/12/2002 | Validation | RNIB (Peterborough) | A meeting was held between Rachel Pick and Helen Graupp of iSys to revise the validation planning in light of the expected delay in the 1st prototype delivery. |
| 06/02/2003 | Planning meeting | SAGE (Madrid) | Pending issues in the Viapolis collaboration in the IMAGINE project. |
| 24/02/003 | Installation of P1 | RNIB (Peterborough) | IMAGINE Installation and Network integration meeting. |
| 27/02/2003 | Substitution of Viapolis by Iniciativas Digital Media | SAGE (Madrid) | SAGE, ITI, TID, IDM held a meeting for clarifying the role of IDM in the IMAGINE project, what are the responsibilities. |
| 05/03/2003 | Consortium meeting | RNIB (Peterborough) | Full consortium meeting. State of the project and review preparation. The technical details are the most relevant topics in the meeting. Exploitation and management issues where addressed also. |
| 6-8/04/2003 | Consortium meeting & informal review | IAI (Saarbrucken) | Full consortium meeting. State of the project and informal review. It has been a three days meeting:<br>• Preparing the review for the next day.<br>• Informal review and analysis of the project officer recommendations.<br>• Consortium meeting for further activities. |

## 4.6 Updated Project Snapshot

***This page is intended for internal use by the Commission staff and will be kept confidential***

| PROJECT No: IST-2000-29490 | ACRONYM: IMAGINE |
|---|---|

**TARGET MARKETS:**

The target markets for the commercialisation of the IMAGINE product are wide, due to the fact that a natural language Interface can be plugged-in with general purpose software but with specific one as well. IMAGINE has made a special effort to support the requirements of e-commerce applications: Internet, http, access via mobile phone, etc. The main mediums foreseen to market IMAGINE will be the software houses that would integrate the interface into their products; the VARs which could act as an intermediate to the big organisations and the big organisations themselves. The tool is general enough to support access to any application although the current IMAGINE version is optimised to work over Internet.

**TECHNOLOGIES:**

The IMAGINE project uses de facto standards in IT arena in order to be exploitable with no extra requirements in a wide range of installations. Based 90% in JAVA and RMI, it can be deployed in any JAVA platform. The RMI use for interconnection, allows new tools (Speech tools for example) to connect through a standardised way to the IMAGINE core. The ALEP platform which resulted from a series of previous EC/EU projects in the field of Language Engineering is being used and improved with the IMAGINE experience for support the Spanish language.

**EXPLOITABLE RESULTS:**

IMAGINE, inherited from previous project MELISSA, produces software, methods and standards in the Natural Language field:

- The software machinery that implements the end-user interface, a distributed run-time system that allows to access the application through natural language via speech mode but also in written mode.
- A development subsystem consisting of an integrated suite of tools that provide the methods and tools to obtain the application knowledge required by the run-time system.
- Definition of methods for the integration of a natural language interface into software applications, that could be used as a proposal for adoption as a public standard, to the appropriate bodies.

**ANTICIPATED OR DEMONSTRATED BENEFITS:**

The main benefit is the *extended use of software applications* by different people profiles: impaired people, occasional users, managers and secretaries, and in general people with necessities of information. All these users will improve its satisfaction having a better access to information by means of a friendly and intuitive interface, without need of knowing the application functioning.

The *company* will also obtain measurable benefits: an increase of the personnel productivity, with an associated decrease of the software support staff work, as well as savings in personnel training. On the other hand, IMAGINE will allow the enterprise a gradual IT planning by enlarging the life cycle of existing applications with its powerful interface; quality benefits derived from the standards compliance and the involved technology, and all relying on the company image.

**APPLICATION AND ASSESSMENT OF RESULTS:**

The IMAGINE system has been developed and tested with an existing application which is used at the RNIB organisation and another in the La Vanguardia newspaper. The RNIB application is an Online Shop currently in production. IMAGINE operates with the site as another user, the own Online Shop does not distinguish when is a human user (through an HTML navigator) or when is IMAGINE. In the case of La Vanguardia, IMAGINE interfaces an application not in use, is part of the old Viapolis content (former partner in the project). The experiences with the system are good, there are two reports devoted to the experiences with these tests.

**INNOVATIVE ASPECTS**:

The IMAGINE project has innovative aspects in the fields of Natural Language Understanding and commands generation. To cope with its main objective which is to interface the application functions from NL input, the run-time subsystem includes a well-defined knowledge representation of the application and associated semantics and a powerful treatment of the linguistic, semantics and contextual aspects. It also deals with the generation commands area from an analysis of the applications and its specific knowledge.

***This page is intended for internal use by the Commission staff and will be kept confidential***