

Multimodal Output Simulation Platform for Real-Time Military Systems

*Cyril Rousseau*¹ *Yacine Bellik*¹ *Frédéric Vernier*¹ *Didier Bazalgette*²

(1) LIMSI-CNRS
Université Paris XI
B.P. 133
91403, Orsay, France
surname.name@limsi.fr

(2) General Delegation for the Armament
DGA/DSP/STTC/DT/SH
8 Bd Victor
00303, Armées, France
didier.bazalgette@dga.defense.gouv.fr

Abstract

Interaction context is a major characteristic for future interactive systems. Interactions techniques must become dynamic and contextual in order to be adapted to more and more diversified environments, users and systems. In this paper, we study the influence of interaction context on system outputs. More precisely, we propose a model to express information according to running state of the interaction context and to manage resulting multimodal presentations. We also present a tool based on this model for specification and simulation of multimodal output systems. Finally, we describe an application of this platform through a task of marking out a target on the ground in a fighter plane cockpit.

1 Introduction

Interaction techniques and input-output devices become more and more diversified. Their evolution leads to a contextualization of the interaction. The interaction techniques are often developed for a certain type of application and for a particular interaction context. In this case, the richness of interaction between user and machine is very limited. That is why we need to adapt communication: in other words, we need to reorganize interaction according to the current state of the interaction context.

To solve this problem, software architectures of interactive systems need to be extended. A context module and new mechanisms in charge of interaction adaptation should be introduced. Without updating the architectural models, context elements might be less accessible to the developer and difficult to extend afterwards. Moreover, the influence of context on user interaction makes the design task less clear. Then it is not easy to specify a complete behavioural model of the system outputs. Indeed, interaction environments, capacities of systems, etc., are evolving every day and we have to regularly update these outputs. So the used approach has to guarantee the properties of reusability, extensibility and accessibility.

This paper presents a model for output multimodal systems design. It addresses the main problems of output multimodality and proposes new mechanisms allowing the expression of information according to the current state of the interaction context. A specification / simulation tool has been developed in order to apply and to validate this model. Its features are shortly described. An example illustrates the application of these concepts on a task of marking out a target on the ground in a fighter plane cockpit. An extension of the model stemming from this first command is also presented. The paper concludes with a discussion about the benefits and the lacks of the model.

2 Multimodal Presentation

Output multimodality consists in finding how to present information to the user (Bordegoni et al, 1997). This problem can be decomposed in three main steps:

- *What* is the information to present?
- *Which* modality(ies) should we use to present this information?
- *How* to present the information using this(ese) modality(ies)?

A fourth sub-problem *Why* (Karagiannidis et al, 1996) on the associated goals, can be added. This step is included in our design process (Rousseau et al, 2004) and does not influence in a direct way the model.

The figure 1 presents the process of an information expression. The following sections describe the different steps and the associated terminology.

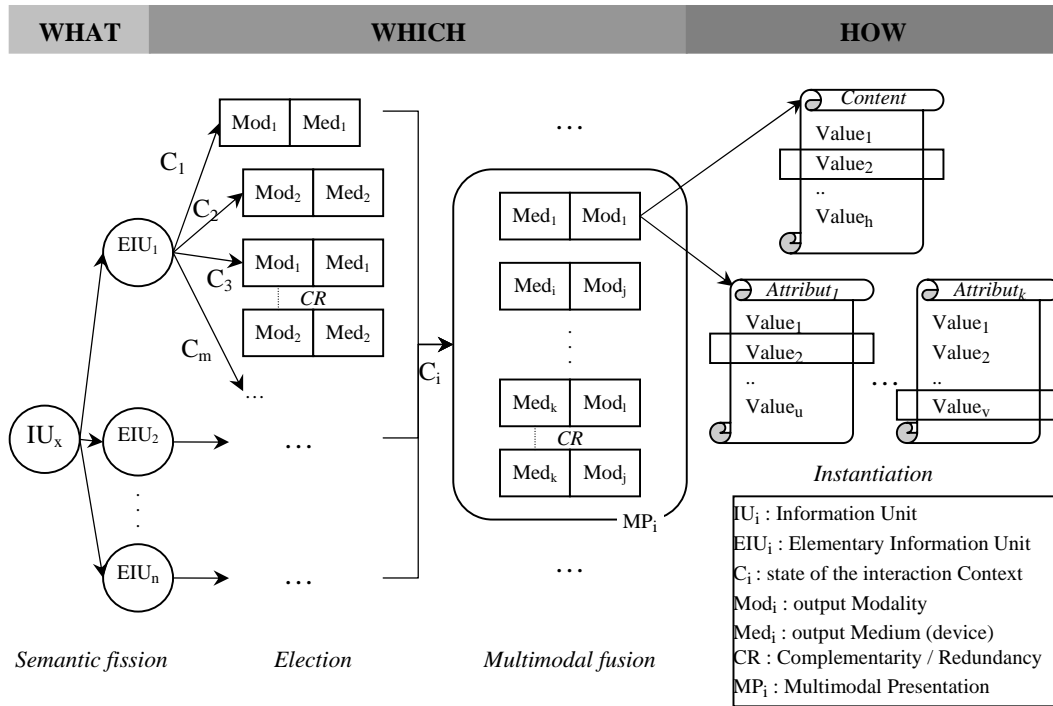


Figure 1: Creation of a multimodal presentation

2.1 What Information to Present?

This step handles the semantic information (Figure 1, IU_i). First, it is necessary to decompose the information in different semantical parts. This problem consists in selecting in this semantic information, elementaries informations¹ (Figure 1, EIU_i) which must be presented to the user. Let

¹ An Elementary Information Unit (EIU) is an atomic semantic information.

us take the example of our first case study : a fighter pilot marking out a target on the ground (Figure 2) which involves 3 output devices : HMTV(Helmet Mounted Visor), LRS (Large Reconfigurable Screen) and HAS (Helmet Audio System). The global semantic information is “Add mark at X (coordinates)”. This information can be decomposed into two elementary information units: the command (Add mark) and the mark coordinates (X).



Figure 2: The fighter aircraft simulator

Some authors use the word “fission” by the opposite to the word “fusion” to name the process of output modalities selection. We think this term is not relevant for this use. Actually there is a fission process, more precisely decomposition process but this fission takes place on a semantic level. So we prefer to talk about “semantic fission”. The decomposition process seems difficult to automate. At the present time, the designer must specify during the conception of the application any semantic information to present and the associated decomposition in elementary information units. An editor has been implemented in order to record these descriptions.

2.2 Which Modality(ies) to Use?

Once the information is decomposed, it is necessary to choose the presentation form (Figure 1, MP_i). This problem means selecting for each elementary information unit (Figure 1, EIU_i) a multimodal presentation composed of one or several interaction components (modes², modalities³, media⁴) (Bellik et al., 1995). These presentations must be adapted to the running state of the interaction context (Figure 1, C_i) (Dey et al., 2002). Elected presentations are then merged into only one multimodal presentation expressing the initial information (Figure 1, IU_i).

In the case of our example, in a nominal situation, the command feed-back is visually (mode) expressed as a text (modality) through the helmet mounted visor (medium). In the same way, the mark can be visually and auditively (mode) presented as a combination of a geometric shape (modality) displayed on the helmet and on the main cockpit screen (media) and a 3D earcon (modality) in the helmet audio system (medium). The state of the interaction context may change the multimodal presentation form. For example, if audio canal is already used, the mark will not be presented with a 3D earcon.

² An output mode corresponds to a user’s sensory system (visual, auditory, etc.).

³ An output modality is an information structure perceived by user (text, graphic, sound, etc.).

⁴ An output medium is an output device allowing the expression of an output modality (screen, speaker, etc.).

This selection process according to the interaction context is based on a behavioral model. The application behavioral model clarifies the interaction components adapted to the running state of the interaction context and allows the expression of given elementary information units. The formalization of this model can be made in different ways: tree / graph of decision, adaptation rules (Karagiannidis et al, 1996), ergonomic rules, list of invariants, etc. In our proposition the behavioral model has been formalized using rules. The rule premises consist in a partial description of an interaction context state. The rule conclusions define a contextual weight underlining the interest of the aimed interaction components. Other types of rules, such as CARE (Coutaz et al, 1996) rules, allow us to allocate a presentation composed of several modalities based on complementarity / redundancy criteria. We call the process of selecting presentations, the “election” process by analogy with a voting system. Our election process is based on a rules base (voters) which once applied add or remove points (votes) to certain modes, modalities or media (candidates), according to the current state of the interaction context (the overall political situation).

The fusion of the different multimodal presentations is a mechanism of coordination of the election results. So this mechanism verifies the coherence of the global presentation. The presentations should be temporarily and spatially coherent. In case of conflict, one or several presentations need to be cancelled and new elections launched.

2.3 How to Present the Information?

This step called “multimodal presentation instantiation” (André, 2000) consists in generating the multimodal presentation previously elected. This generation breaks up into two parts. The first one consists in choosing the concrete content to express through the modalities making up the allocated multimodal presentation. The second part makes choices on the presentation parameters (modality attributes, spatial and temporal parameters, etc.). Spatial and temporal parameters refer to “When” and “Where” points (Karagiannidis et al, 1996). In our example, the command feedback is presented through a text “Add mark command” (content) in police Arial 11 (text attribute) in the continuation of the last command feed backs list at the top to the left (spatial parameter) in the helmet mounted visor.

The presentation instantiation is at present time handled by a rendering engine specialized in avionic applications of our industrial partner Thales-Avionics. However, the implementation of a generic instantiation engine is in progress.

2.4 Life Cycle of a Multimodal Presentation

We just have seen how to present information and more precisely how to adapt multimodal presentations to the interaction context. However, the state of the interaction context may change, which raises the problem of the presentation validity in case of context evolution.

This problem affects only persistent presentations. Furthermore, all the evolutions of the interaction context do not require the verification of the presentation validity. The evolution must carry on at least one key element of the context. This key element is an instantiated criterion of the context occurring in a rule of the behavioral model. The list of these instantiated criteria characterizes the current state of the interaction context and thus specifies the validity of the elected presentation. The evolution of these criteria within the description of the context underlines an invalidation of the presentation.

Invalidation does not concern always the whole presentation but may affect only one or several elements of the presentation. Partial re-elections update the presentation regarding the new interaction context. This solution limits the number of new elections and thus improves the processing times. However the coherence of the updated presentation must be re-verified. This check may then require a total re-election of the presentation. To finish, the new presentation must be played in the last state known before the invalidation.

3 Multimodal Output Specification/Simulation Tool

The design of an output multimodal system is composed of two steps. A first step specifies the knowledge needed to apply the introduced concepts. A second step tests the validity of this specification through an application of the model. The following sub-sections describe the specification and the simulation of the system outputs.

3.1 Specification of the System Outputs

Outputs specification (Figure 3) can be divided into four tasks:

- Specification of the interaction components
- Modelling of the interaction context
- Specification of the information units
- Creation of a behavioral model

A tool called MOSTe (Multimodal Output Specification Tool) has been implemented in order to make easier the specification task. This tool is composed of one editor for each specification task. Figure 3 presents the editor of the behavioral model. The application behavioral model can be described by a set of rules. Each rule is specified graphically by describing its premises and its conclusions (Figure 3).

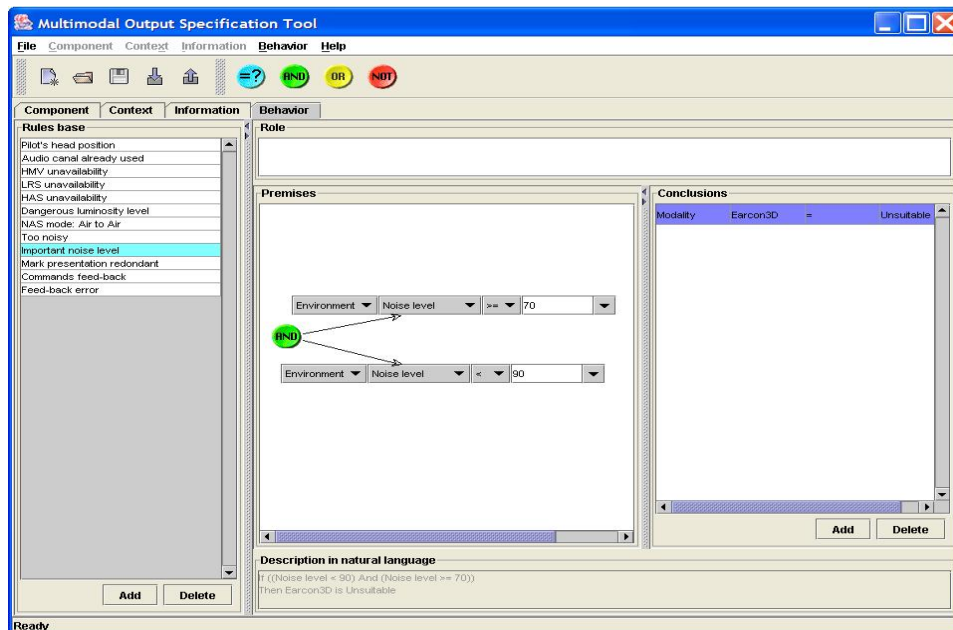


Figure 3: The behavioral model editor.

At the end of the fourth task, the resulting specification is exported for a next use (Figure 4). This exportation creates a set of four files describing each step of the specification in MOXML (Multimodal Output eXtended Markup Language). MOXML is a data representation language based on XML (eXtended Markup Language) with a set of tags describing all the needed elements in an output multimodal system. These files are afterward run by the simulation kernel introduced into the Figure 5 (Rousseau et al, 2004).

```

<rule name="Canal audio occupied" number="2">
  <premises>
    <premise number="1">
      <elt_left model="system" criterion="audio canal occupied" />
      <operator name="=" />
      <elt_right type="boolean" value="true" />
    </premise>
  </premises>
  <conclusions>
    <conclusion number="1">
      <target level="modality" name="3D Earcon" />
      <effect value="forbidden" />
    </conclusion>
  </conclusions>
</rule>

```

Figure 4: A rule example described in MOXML.

3.2 Simulation of the System Outputs

An implementation of the architecture model (Figure 5) simulates the outputs specifications exported by MOSTe. This system called MOST (Multimodal Output Simulation Tool) contains in particular a generic election module which is responsible of the choice of the pertinent modalities (depending on the current interaction context) and a management module for current multimodal presentations after they have been elected. The links with the other application modules (dialog controller, context module and system media) are managed by RMI (Remote Method Invocation) connections which allows to support distributed architecture. The specification files are parsed during the MOST initialization process and any modification on the specification only needs a re-initialization.

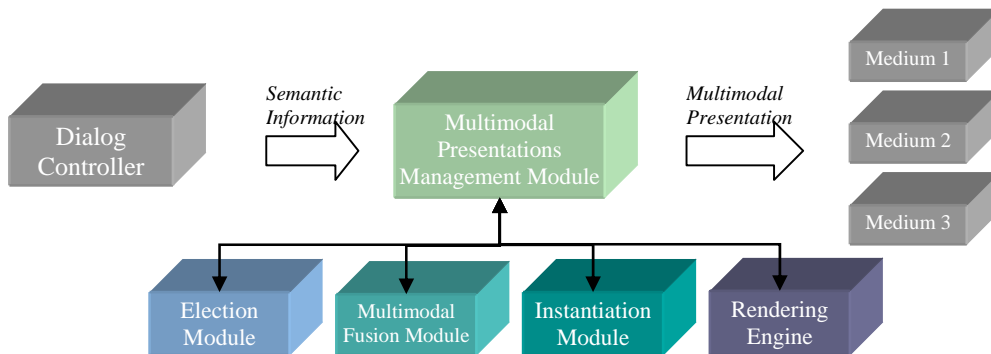


Figure 5: The architecture model.

MOST presents any information recorded in the specification according to the specified interaction context and spies any evolutions of the context requiring invalidations and afterward new elections. More details about the architecture model can be found in (Rousseau & al., 2004).

4 Applications

An application was carried out on the INTUITION (multimodal interaction integrating innovative technologies) project. This project is partly funded by French DGA (General Delegation for the Armament) and includes three laboratories (LIMSI-CNRS, CLIPS-IMAG and LIHS-IRIT) and an industrial partner (Thales-Avionics). The project objective is to develop an adaptation platform for the new Human-Computer Interaction technologies. The first application of this platform was about interaction in a fighter plane cockpit. This application is applied on the Thales-Avionics's simulator. This simulator is mainly composed of a graphical environment (based on the flight simulator X-PLANE), scenarios (managed by GESTAC, a Thales-Avionics application) and a multimodal interaction kernel which is composed by 3 systems: ICARE, Petshop and MOST. ICARE (Bouchet & Nigay, 2004) catches and analyses pilot's input interactions. Input events are sent to a dialog controller specified with Petshop tool (Bastide et al, 1998) through Petri networks. Finally the information presentation is managed by MOST.

The main objective of this application was to verify real time constraint and to check communication between the different partner's modules. The extensibility of the approach was also studied. A first prototype of the application has been implemented to validate the different components of the platform. This prototype is about a task of marking out a target on the ground.

4.1 Marking Out a Target on the Ground

Outputs specification of this first task is presented below. Figure 6 presents the interaction components diagram (HMV: Helmet Mounted Visor, LRS: Large Reconfigurable Screen and HAS: Helmet Audio System). Interaction context is composed of three models and eight criteria presented in the Table 1.

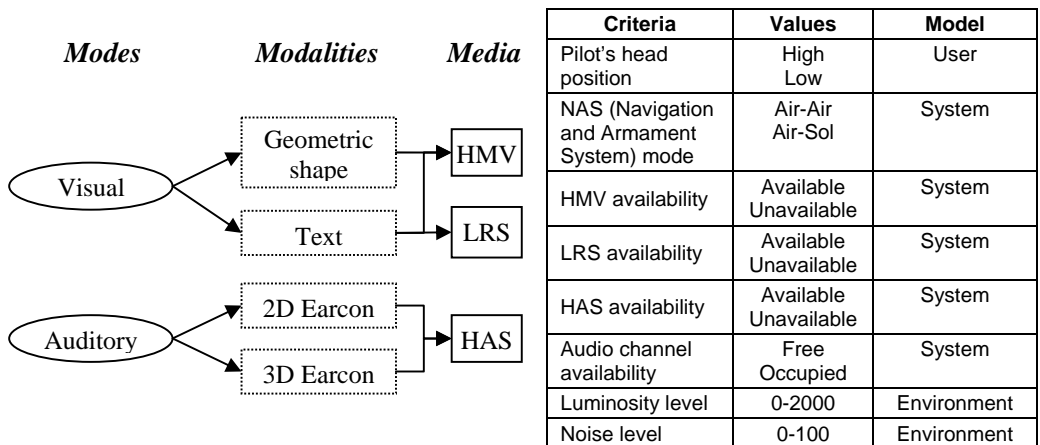


Figure 6: Interaction components diagram.

Criteria	Values	Model
Pilot's head position	High Low	User
NAS (Navigation and Armament System) mode	Air-Air Air-Sol	System
HMV availability	Available Unavailable	System
LRS availability	Available Unavailable	System
HAS availability	Available Unavailable	System
Audio channel availability	Free Occupied	System
Luminosity level	0-2000	Environment
Noise level	0-100	Environment

Table 1: Interaction context.

Four information units (add valid mark, add invalid mark, refresh mark and remove mark) are managed by system. Finally, the application behavior is specified through a base of twelve election rules described in the Table 2.

Table 2: Election rules of the behavioral model.

Id	Name	Description in natural language
1	Pilot's head position	If the pilot's head position is low Then do not use HMV medium
2	Audio channel used	If the audio channel is already in use Then do not use Earcon3D modality
3	HMV unavailability	If the HMV is not available Then do not use HMV medium
4	LRS unavailability	If the LRS is not available Then do not use LRS medium
5	HAS unavailability	If the HAS is not available Then do not use HAS medium
6	Dangerous luminosity	If the luminosity level is superior to 1500 lux Then do not use HMV medium
7	Air to Air NAS mode	If the NAS mode is Air-Air Then do not present the information
8	Too noisy	If the noise level is superior to 90 dB Then do not use Auditory mode
9	Important noise	If the noise level is between 70 and 90 dB Then Earcon3D is unsuitable
10	Mark presentation	If the current EIU is a 3D point Then use Redundancy property
11	Commands feed-back	If the current EIU is a command feed back Then use text and try to express it with the HMV
12	Error presentation	If the current EIU is an error feed back Then use Earcon 2D modality

In a nominal situation, only rules 10 and 11 are applied. In the case, pilot's head position is low, a new rule (1) changes the form of the last presentation (stops the use of the helmet mounted visor). In this case, the system adapts itself to the interaction context by choosing the large reconfigurable screen as a first display area. Now, let's suppose that a mark is already presented. If the pilot lowers his head to look at something in the cockpit, we have a context evolution which needs an invalidation of the mark presentation and requires new election.

4.2 Switching On / Off the Radar

Following this prototype, an integration phase began. This phase consists in increasing the model of the multimodal interaction module by managing more and more tasks. The impact of this integration on our first outputs specification is interesting. Let us take the case of a task consisting in switching on / off the radar. This task is made with a button from the cockpit tactile display. To manage this new task, it is necessary to add at first the new medium "Tactile Display" in the interaction components diagram. The interaction context model does not need to evolve whereas the list of the information units must be updated with the addition of two new information units: "radar on" and "radar off". These UIs do not require a treatment and are considered like refresh information. Thus behavioral model does not need to be updated. The management of this new task only needs re initialization of the simulation tool to load the updated specification.

5 MOST versus Expert System

There are fundamental differences between MOST and an expert system (Jackson, 1999). In an expert system the facts base is fed by the rules conclusions, whereas in MOST the election rules do not influence (in a direct way) the facts base. This facts base (models criteria used to represent the context) is fed by physical sensors (accelerometer, position tracker, etc.).

Moreover, the structure of an election rule does not correspond to the structure of an expert system rule. In the case of an expert system, premises and conclusions are facts. For this reason, a rule

conclusion can enrich the base of facts. In the case of the MOST rules, premises and conclusions do not belong to the same category. Indeed, the rule premises are based on knowledge about the interaction context and the conclusions directly relate to the interaction components. Therefore a rule conclusion cannot influence in a direct way the premises of another rule. There is no direct dependence between premises and conclusions of election rules.

This has a considerable consequence on the election process. Unlike a true inference engine, our engine only needs a single application of the rules base to produce a multimodal presentation. This precisely enables us to achieve real-time processing. Table 3 illustrates the processing times for information presentation in the case of the “Ground marking out in a fighter plane” prototype. All these differences mean that MOST is not an expert system.

Table 3: Processing times for information presentation in the “Ground marking out in a fighter plane” application.

	Election	Presentation size (modalities number)	Processing times (ms)
Add valid mark	X	3	20
Add invalid mark	X	3	15
Refresh mark	-	0	2
Remove mark	X	1	10

6 Conclusion

In this paper, a model and a tool for the design and specification of dynamic and contextual multimodal output systems have been presented. A first application allowed us to validate our model, to check real-time constraint and to underline missing features in our tools.

Our approach is based on a behavioral model formalized by a base of election rules. This formalism has the advantage to propose a simple reasoning (If ... Then instructions) limiting the learning cost and making easier the implementation. However it is rather easy to get lost during the rules base conception. That is why, we think of adding another formalism such as decision tree. This new formalism will give a global view of the model conception additional to the local view of the election rules.

The simulation engine is actually incomplete. We are working on the multimodal fusion as well as an instantiation engine. In both cases, we need to better specify these steps before the implementation. These developments should be made on a second application of the INTUITION project. This second application concerns an air traffic control system and will allow us to check our approach genericity and design/specification rapidity.

On the evaluation side, a first evaluation of the fighter plane cockpit application will be made by Thales-Avionics with experienced pilots during the summer of this year (2005). A second evaluation on the software tool itself will also be done during this year. Evaluation of the avionic application will validate or invalidate the associated specification. A validated application must be then re-implemented for a real use according to the specification. However it raises the problem of the evaluation pertinence. Indeed, it seems difficult to say that the evaluation validity will be necessarily confirmed in the real system. The evaluation of the fighter application should help us on this point.

7 References

- André, E. (2000). The generation of multimedia presentations. In R. Dale, H. Moisl and H. Somers (Ed.), *Techniques and Applications for the Processing of Language as Text* (pp. 305-327). Marcel Dekker Inc.
- Bastide, R., Palanque, P., Le Duc, H., & Munoz, J. (1998). Integrating Rendering Specifications into a Formalism for the Design of Interactive Systems. In Proceeding of the 5th Eurographics workshop on Design: *Specification and Verification of Interactive systems* (DSV-IS'98).
- Bellik, Y., Ferrari, S., Néel, F., Teil D., Pierre, E. & Tachoures V. (1995). Interaction Multimodale: Concepts et Architecture. *L'Interface des Mondes Réels et Virtuels*.
- Bordegoni, M., Faconti, G., Maybury, M.T., Rist, T., Ruggieri, S., Trahanias, P. & Wilson, M. (1997). A Standard Reference Model for Intelligent Multimedia Presentation Systems. *Computer Standards & Interfaces*, 18(6-7):477-496.
- Bouchet, J. & Nigay, L. (2004). ICARE: A Component Based Approach for the Design and Development of Multimodal Interfaces. Extended Abstracts of CHI'04.
- Coutaz, J., Nigay, L., & Salber, D. (1995). Multimodality from the user and system perspectives. In Proceeding of ERCIM (European Research Consortium for Informatics and Mathematics) workshop on *User interface for all*.
- Dey, A. K., Salber, D. & Abowd, G.D. (2002). A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. In Moran, T.P. and Dourish, P. (Ed.), *Context-Aware computing: a special triple issue of Human-Computer Interaction*.
- Jackson, P. (1999). *Introduction to Expert Systems*. Harlow: Addison-Wesley Longman Limited.
- Karagiannidis, C., Koumpis, A., & Stephanidis, C. (1996). Deciding 'What', 'When', 'Why', and 'How' to adapt in intelligent multimedia presentation systems. In Proceeding of ECAI (European Conference on Artificial Intelligence) workshop on *Towards a standard reference model for intelligent multimedia presentation systems*.
- Rousseau, C., Bellik, Y., Vernier, F. & Bazalgette, D. (2004). Architecture framework for output multimodal systems design. In Proceeding of OZCHI'04.

8 Acknowledgements

The work presented in the paper is partly funded by French DGA (General Delegation for the Armament) under contract #00.70.624.00.470.75.96.