

ASUR++: a Design Notation for Mobile Mixed Systems

Emmanuel Dubois⁽¹⁾, Philip Gray⁽²⁾, Laurence Nigay⁽³⁾

(1) IRIT – LIIHS, 118 route de Narbonne, 31062 Toulouse Cedex 4, France

(2) Department of Computing Science, University of Glasgow, UK

(3) CLIPS – IIHM, BP 53, 38 041 Grenoble Cedex 9, France

emmanuel.dubois@irit.fr, pdg@dcs.gla.ac.uk, laurence.nigay@imag.fr

Abstract. In this paper we present ASUR++, a notation for describing, and reasoning about the design of, mobile interactive computer systems that combine physical and digital objects and information: *mobile mixed systems*. ASUR++ helps a designer to specify the key characteristics of such systems and to focus on the relationship between physical objects and actions and digital information exchanges. Following a brief introduction to the notation, we illustrate its potential usefulness via examples based on the design of an augmented museum gallery. We conclude with a consideration of the integration of ASUR++ into the system development process and its augmentation via associated methods and tools.

1 Introduction

The design of interactive systems is perhaps more challenging now than it has ever been. The circumstances in which user-computer interaction take place are increasingly varied, as are the devices available for use in these settings. The mobility of users and the volatility of the interaction-significant context are increasingly common. The physicality of the interaction setting cannot be ignored, both because physical context plays a growing role in applications but also because interaction is mediated by devices deeply embedded in everyday surroundings. In this situation it is unclear how applicable are the well-developed and well-known interaction techniques and design principles that characterise interaction design for static, workstation-based applications.

The work reported here is directed at addressing these design challenges, i.e., how can we describe interactive systems in which mobility and physicality are key features, to support communication and reasoning about our design ideas? Designers of our physical environment can exploit maps and blueprints; designers of the computational environment have UML. Developers of the emerging mobile and pervasive computing environment need equivalent tools.

The structure of the remainder of the paper is as follows: We first set the context of our work by considering the nature of the domain, viz., mobile mixed systems, and current notational support for describing them. We then describe the ASUR++ notation and explain the uses of the notation for the design of mobile mixed systems during different design phases. Having presented the notation and its usefulness, we illustrate it by considering the design of an augmented museum gallery. We compare several design solutions, all of which are described using ASUR++. We then illustrate how ASUR++ may help the design team to interconnect the different aspects of the design of a mobile mixed system.

1.1 Augmented Reality and Augmented Virtuality

As defined in (Dubois et al 1999), a Mixed System is an interactive system combining physical and digital entities. There are two fundamental types of Mixed System:

- Systems that enhance interaction between the user and her/his real environment by providing additional capabilities and/or information. We call such systems *Augmented Reality* (AR) systems.
- Systems that make use of real objects to enhance the interaction between a user and a computer. We call such systems *Augmented Virtuality* (AV) systems.

On the one hand, the NaviCam system (Rekimoto & Katashi 1995), our MAGIC platform for archaeology (Renevier & Nigay 2001) and our Computer Assisted Surgery system CASPER (Dubois et al 2001) are examples of Augmented Reality systems: the three systems display situation-sensitive information by superimposing messages and pictures on a video see-through screen. On the other hand, the Tangible User Interface paradigm (Ishii & Ullmer 1997) belongs to Augmented Virtuality: physical objects such as bricks are used to interact with a computer. The design of such mixed systems, Augmented Reality as well as Augmented Virtuality, give rise to

particular challenges due to the new roles that physical objects can play in an interactive system. The design challenge lies in the fluid and harmonious fusion of the physical and digital worlds.

In addition, by taking advantage of new methods of communication, mobile technologies, and new interaction devices, interactive systems are no longer restricted to local use on a desktop; they are becoming increasingly mobile. Consequently, another challenge for mixed systems development and design is to integrate aspects of the user's interaction environment at run-time: these different aspects form the user's context and the resulting systems are also called "context-sensitive systems".

1.2 Key aspects of the design of a mobile mixed system

There is as yet no consensus on the information that needs to be considered during the design of mobile mixed systems systems. Nevertheless, four aspects have been the focus of recent research. Probably the most important is related to *the presentation, manipulation and exploitation of information in a mobile mixed system*, that is, a human interaction perspective. For example, Bellotti and Edwards (2001) present a framework helping the designer to take into account human aspects that cannot be sensed by any technologies but that are of importance for the user's interaction with a mobile mixed system. Winograd (2001) promotes "human-centered" architectures rather than technology- or communication-centered ones.

Another aspect of mobile mixed system design deals with *software engineering*. Given that inputs of such systems are no longer limited to those provided by the user, there is a crucial need for adapting and enriching the traditional architectural models. The context toolkit is an example of such work (Dey 2000).

A complementary dimension that has to be present in the mobile mixed system design area is *the design and development of technologies that sense*. A wide variety of sensors may be useful in capturing the user's environment. The design and deployment of such sensors are both significant here (Schmidt et al 1998).

Finally, to avoid the current exploratory approach that only leads to ad hoc solutions, *a more systematic development approach* is required. Developing a design method is required because existing HCI design methods are not adapted for mobile mixed systems since they do not

explicitly take into account the impact of physical entities and a user's interaction with them. For example, based on a model of sensed context, (Gray & Salber 2001) demonstrate how the method helps in expressing requirements for sensed context information and permits the exploration of the whole design space.

1.3 Existing design methods for mobile mixed systems

We are not aware of any existing design tools or system development methods, apart from ASUR++, that have been designed to capture and express the particular characteristics of mobile mixed systems. In the absence of such tools and methods, a designer wishing to characterise their design can choose to use:

- a task description notation such as ConcurTaskTrees (Paterno 1999),
- an application modelling notation such as UML, or
- informal descriptions captured in scenarios or prototypes.

Task-oriented description and analysis is centred on the *temporal* organisation of the interaction, rather than on the characteristics of the physical/digital information interface. That means the designer using a task notation can be faced with having to augment or annotate a description to identify the physical/digital boundary and may also have to commit too early to a consideration of the temporal properties of the elements involved in the interaction. Modelling languages like UML are well-suited to capture application functionality, but, apart from treating users as use case agents, no physical entities are represented or characterised. Several extensions to UML have been developed to overcome the limits of UML for characterising interaction (Van Harmelen 2001). For example, the User Interface diagrams of the UML_i notation (Pinheiro & Paton 2000) allow the modelling of abstract user interface components. However, this extension is concerned with the specification of user interfaces and, as we demonstrated in (Dubois et al 2002b), it doesn't fulfil the need for a way to express the overall system design independent of the user interfaces to particular digital components. Finally, scenario and prototype-based design approaches provide more expressive freedom than task and application modelling but don't offer the designer the ability to systematically compare, explore or generalise design solutions.

2 The ASUR++ Notation

ASUR++, as its name suggests, is an extension of an existing notation, ASUR. ASUR was designed for the description of the physical and digital entities that make up a mixed system, including user(s), physical and digital artefacts, and the physical and informational relationships among them (Dubois 2001). The purpose of ASUR is to help in reasoning about how to combine the physical and digital worlds, by identifying physical and digital objects involved in the system to be designed and the boundaries between the two worlds (Dubois et al 2002a). The features of ASUR, and its extensions to create ASUR++, are described below.

To describe the merging of physical and digital entities and relationships, ASUR takes into account design-significant aspects highlighted in other approaches to characterising AR systems. These existing characteristics include:

- the type of data provided to the user (Asuma 1997; Feiner et al. 1993; Noma et al. 1996), which may be textual, 2D or 3D graphics, gesture, sound, speech or haptic, and
- the potential physical targets of enhancement, in order to combine physical and digital data (Mackay et al. 1998); the target may be users, physical objects or the environment.

To these characteristics, ASUR++ adds other factors related to the use of physical entities. ASUR++ thus combines and enriches aspects addressed in the different AR approaches. For example, user mobility in a mixed system requires us to consider the nature of the spatial relationships between users and the other entities involved in a mixed system. The new features added to ASUR to create ASUR++ enable a designer to express such spatial relationships between a user and an entity. For example, using ASUR++, one can express a condition that a user must be less than two meters from a specific physical object. This contributes to expression of the user's perspective, to be taken into account when designing and developing a mobile mixed system, as described in section 1.2. Although there is work in all the other areas mentioned (software engineering, sensors design, systematic design approach), there is little in way of exploration of the linking of these aspects. As a potential answer to this challenge, we will

illustrate how ASUR++ might be used to bridge between the different aspects of mobile mixed system design.

2.1 Components and Relationships

For a given task, ASUR++ describes an interactive system as a set of four kinds of entities, called components:

- Component S: computer system;
- Component U: user of the system;
- Component R: real object involved in the task as a tool (R_{tool}) or as an object intended to be modified as a consequence of the task (R_{object});
- Component A: input adapters (A_{in}) and output adapters (A_{out}) bridge the gap between the computer-provided entities (component S) and the physical world entities, composed of the user (component U) and of the real objects relevant to the task (components R_{object} and R_{tool}). Input adapters represent all kinds of sensors such as pressure, haptic, auditory, optical sensors, among others. This includes keyboards, cameras, microphones, and localizers. Output adapters may be any devices through which the user can perceive some information (e.g., screen, speakers, force-feedback devices, etc.).

We have identified three kinds of relationship between two ASUR components:

- Exchange of data: represented by an arrowed line ($A \rightarrow B$) from the component emitter (A) to the component receptor (B). This symbolises the transfer of information between two ASUR++ components. For example $A_{out} \rightarrow U$ may represent the user's perception of data displayed on a screen (i.e., an output adapter).
- Physical activity triggering an action: a double-line arrow ($A \Rightarrow B$) denotes the fact that when the component A meets a given spatial constraint with respect to component B (for example, A is no further than two meters from B), data will be exchanged along another specific relationship ($C \rightarrow D$). The spatial constraint and the relationship on which a transfer will be triggered are properties of this kind of relationship.

- Physical collocation: represented by a non-directed double line ($A=B$). This refers to a persistent physical proximity of two components. It might be used between any kind of components among those describing the user (U), the adapters (A_{in} and A_{out}) and the real entities (R_{tool} and R_{object}). In ASUR++ diagrams, this collocation is reinforced by a contour drawn around the components that are so collocated. This highlights groups of entities that move, or have to be multiply instantiated, if one of them is moved or multiply required. This contour is a single line contour if the set is mobile, and a double line if the set of components remains static during the interaction. This indication of grouping also makes it easier for the designer to deal with multiple instances of the collocation relationship, i.e., when more than one user or more than one instance of a physical object will be used. Multiplying the number of users or physical objects will lead to the multiplication of the components included in the contour.

Interaction with the system is thus represented by a set of relationships connected to the component U , representing the user. In the following section we present several important characteristics of the ASUR++ components and their relationships.

2.2 Characterisation of Components and Relationships

ASUR++ characteristics are those properties are needed in the design description for reasoning about and between designs. The characteristics described here are chosen as an initial set likely to be of value in thinking about mobile mixed reality systems. They include characteristics already identified in other AR design approaches, but also include additional aspects specific to the use of real objects in the interaction. Each relationship connected to the user defines a facet of the interaction, consisting of (i) an ASUR++ component from which information is provided for the user or to which the user provides information, and (ii) an ASUR++ relationship between the user and this component. ASUR++ characteristics exist for both components and different types of relationships as presented in the Table 1 below.

Table 1. Characteristics of ASUR++ components and relationships that make up the user’s interaction facets.

Characteristics of the components	Characteristics of the relationships →	Characteristics of the relationships ⇒
Perceptual/Action location: Physical area where the user has to focus in order to perceive information provided by the component or perform an action on it.	Representation frame of reference: Point of view from which information is perceived or expressed.	Representation frame of reference: Point of view from which information is perceived or expressed.
Perceptual/Action sense: Human sense required by the user to perceive information provided by the component (visual, audio, etc.) or to act on the component (speech or action).	Representation language: Bernsen's representation properties (Bernsen 1994) and dimensionality of the representation that carries relevant information.	Spatial condition of triggering: The condition under which an exchange of data is triggered.
Share: Number of users that can simultaneously access the component to perceive or provide information.	Concept: The application-significant concept about which data is carried by the relationship.	
	Concept relevance: The importance of this concept for the execution of the task.	

For example, the type of representation language of a data exchange relationship (see column 2 of table 1) might be textual, 2D graphical or 3D graphical; the choice of representation language type can be central in determining the nature and effectiveness of an interaction. Similarly, a spatial relationship, e.g., proximity, between a user and some physical object, may cause an information exchange to occur or to become available; this association between the physical relationship and the digital relationship, is captured via the “triggered relationship” property of the relevant physical relationship (see column 3 of table 1). A more detailed description of ASUR++, including full specifications of these component and relationship characteristics, is given in (Dubois et al 2002a).

2.3 ASUR++ Descriptions and Tasks

An ASUR++ description captures the characteristics of a mixed system relevant to a specific task or user activity. Relationships among the components represent informational and physical properties of both the static context and the dynamic behaviour of the system and related physical objects, during the activity in question. The validity of the ASUR++ description of a system is thus restricted to the duration of the user's activity. Consequently, to capture a set of tasks or activities, several ASUR++ diagrams may be necessary to describe the whole behaviour of the system. For example, in the field of Computer Assisted Medical Intervention (CAMI), three tasks are commonly required: acquiring data, planning the intervention (e.g. the intervention trajectory to reproduce during surgery) and guiding the clinician during the actual intervention (Troccaz et al 1996). Thus, to describe a complete CAMI system, three ASUR++ diagrams would be required, one for each of these main user activities.

2.4 ASUR++ and the Design of Mobile Mixed Reality Systems

ASUR++ provides a means of describing a number of aspects of interactive mixed reality systems: aspects that are potentially significant at different stages in the development process. We have only begun exploring the use of ASUR++ for the design process and thus have not yet integrated its use into any particular design methods, nor do we yet have a mature method for its use in handling the systematic description and analysis of mixed reality designs. What follows is intended to illustrate how ASUR++ could be incorporated in the development process.

Software engineering organises design and implementation into six phases: requirements definition, specification, implementation, testing, installation and maintenance (ANSI/IEEE 1989). ASUR++ is a design notation and can therefore be used during the requirements definition phase and the specification phase.

For the requirements definition, an ASUR++ description may help to describe the services that the system must provide and the links between the physical and digital worlds. At this stage, the usefulness of an ASUR++ diagram will be similar to that of a UML use case diagram (Stevens & Pooley 2000), except that it also integrates physical entities involved in the system.

Traditionally, during external specification designers deal with the concrete design of the user interfaces, i.e. the windows, buttons, modalities, etc. Due to the merging of physical and digital entities, mobile mixed system design involves design questions that are not part of this traditional approach. A new aspect of external specification must be taken into account: the abstract design of the interaction, which deals with identification and choice of the physical and digital components involved in the system, especially the adaptors (ASUR++ components A_{out} and A_{in}) and their combination. While existing methods and approaches remain valid in the field of mobile mixed systems design to support the concrete user interface design, ASUR++ complements these approaches and addresses another part of the external specification: the abstract interaction design. Indeed, ASUR++ is intended to provide a resource for analysts. It can be used to systematise thinking about design problems for mobile mixed systems. We will demonstrate this point in the following section. Several design solutions can be described using the same modelling approach, enabling easy comparisons. Nevertheless we do not claim that use of ASUR++ alone is sufficient for identifying an optimal or complete design solution. As holds true for any modelling notation, ASUR++ is a tool for the mind and a vehicle for communicating design alternatives.

3 Describing and Analysing Design Alternatives using ASUR++

In this section we examine several different views of a design, each capturing features that can be significant during different steps of the design of a mobile mixed system design.

3.1 An Augmented Museum Scenario

As a vehicle for presenting ASUR++, we use an example taken from the City Project, a project developed within the Equator Interdisciplinary Research Consortium¹. Based on the work of Charles Rennie Mackintosh, a Glaswegian architect of the early 1900s, the City Project has been exploring the augmentation of the permanent Charles Rennie Mackintosh Interpretation Centre. Situated in the Lighthouse, this gallery is an architecture and design centre in Glasgow,

containing exhibits related to Mackintosh's life and work (Brown et al 2002). The aim of this part of the project is to study the impact of combining multiple media to support visitors' activities, especially collaborative activities involving users in the real museum interacting with users exploring a digital version of the same museum ("co-visiting"). For the visitor to the real museum, the system being created is aimed at providing visitors with digital information tailored to visitor's current context. This information tailoring mainly relies on tracking visitor's motions in the museum and identifying the location of the exhibits. Visitor activities are thus embedded with computational capabilities. To do so, the Lighthouse has been equipped with a ultra-sound-based localisation system that provides the location of the visitors.

There are several services that will be provided by the system in the Lighthouse. In this paper, we consider only a single service offered to visitors: following a visit path in the museum. In this scenario, the proposed system will provide AR support to guide visitors through a pre-defined path of exhibits. A path is composed of a set of exhibits, in a given order, that the visitor may observe. A set of paths is saved in a database. Each exhibit on the predefined path has some associated textual comments. In addition we assume that the visitor who wishes to follow a predefined path is already connected to the system and that he has already chosen a path. The main issues of this scenario are twofold: the visitor is mobile and has to be localised in the museum and the system has to know where the user is with respect to the exhibits, in order to provide the right information. Under these conditions, a visitor receives information related to:

- The path to follow: this consists of a set of textual directions and distances separating the current position of the visitor from the next exhibit of the followed path;
- The exhibits: once a visitor reaches the next exhibit along the path he/she is following, the system provides data about the exhibit not perceivable in the museum (e.g. background information about the exhibit and related items not located in the museum) according to preferences defined by the visitor (e.g., social/historical context of the exhibit, curator's commentary , etc.).

¹ www.equator.ac.uk

The role of the computer system is to provide this information to the user based on his/her location relative to the set of exhibits.

It is important to note that this example does not represent the design of an existing system, nor is it a history of an actual design development. Rather, we have chosen this scenario because it represents a realistic design problem (i.e., the design brief is a real one). However, the goals of our example scenario don't correspond to the goals of the City Project and the design alternatives that we present below are our own and don't represent any that have been developed during the City Project. Furthermore, the "augmented museum gallery" scenario has been tackled by a number of other projects as well (Abowd et al 1997; Cheverst et al. 2000). We believe that using a reasonably well-understood application makes it easier to communicate the features of ASUR++ even if the problem and our proposed solutions are neither novel nor likely to be optimal.

3.2 Abstract Description of the Scenario using ASUR++

In terms of ASUR++, the visitor is the component U , an exhibit is a component R_{object} observed by the visitor ($R_{\text{object}} \rightarrow U$) and component S includes the database that contains user paths and information related to the exhibits.

The system provides information related to the path and to the exhibit: $S(\text{path}) \rightarrow U$, $S(\text{exhibit}) \rightarrow U$. This information will be displayed according to the user's position with regard to the position of the exhibit ($R_{\text{object}} \Rightarrow U$ triggers $(R_{\text{object}} \Rightarrow U) \rightarrow U$ and $R_{\text{object}} \rightarrow U$); that is, the user will receive information relevant to the exhibit that he/she is near. The spatial relationship between the user and the exhibit must thus be a source of information to the system.

Figure 1 shows the resulting high-level ASUR++ description.

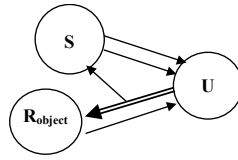


Fig. 1. Abstract description of scenario in ASUR++.

We can elaborate the abstract ASUR++ description of figure 1 by refining the relationships among the components. In the next section we begin by focussing on the relationship between system and user, examining the ASUR++ representation of two design alternatives: (1) Using one adapter to convey both kinds of information to the user, both path and exhibit, and (2) using an adapter for each kind of information.

3.3 Reasoning About Output Design Solutions

Output design refers more generally to the design of the part of a Mobile Mixed System that will provide information to the user. In the first abstract description, given in figure 1, this is identified by the set of arrows that transfer information to the user. There are two kinds of output information: that provided by the computer system, related to both the path and the exhibit, and that provided by the user's perception of the physical exhibit. The latter is fixed since it is related to the user's natural perception. Reasoning about the output design will thus focus on the output provided by the computer system, taking into account the physical realisation of the information. Three aspects will be illustrated in this section to show how ASUR++ facilitates design reasoning in term of adapter type or perceptual or cognitive issues.

3.3.1 Using Two Output Adapters

Adapter elicitation level.

To follow the chosen path, the visitor must be able to perceive the guidance information provided by the system. An output adapter (A_{out1}) is thus required. One relationship from this component is connected to the visitor (component U), denoting the transfer of information related to the path to follow: $A_{out1}(\text{path}) \rightarrow U$. Furthermore, an ASUR++ relationship from the component S to the component A_{out1} is required because information provided by the A_{out1} component is generated

by the database (component S): $S \rightarrow A_{out1}$. Similar reasoning applies to the transfer of information related to the exhibits, leading to the identification of a second output adapter A_{out2} and the relationships $S \rightarrow A_{out2}$ and $A_{out2}(\text{path}) \rightarrow U$. These two output adapters might be placed in the gallery infrastructure or they might be portable and carried about by the user. In what follows, we examine the latter case².

The ASUR++ description shown in figure 2, represents this state of affairs, two output adapters carried by the user, by two physical collocation relationships: $A_{out1}=U$ and $A_{out2}=U$. It is reinforced by the contour drawn around the three components.

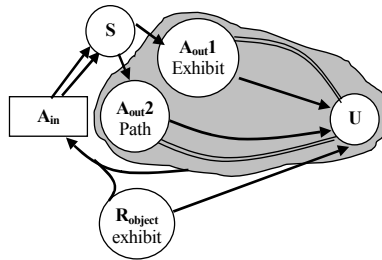


Fig. 2. Partial ASUR++ description of the scenario using two output adapters.

Now, as stated in our scenario and captured in the ASUR++ abstract description, the system has to be aware of the locations of the visitor and the exhibit in order to provide the right information. Consequently, an input adapter (A_{in}) is required to get these positions and transfer it to the computer system: $U \rightarrow A_{in}$, $A_{in} \rightarrow S$ for the visitor's location and $R_{object} \rightarrow A_{in}$, $A_{in} \rightarrow S$ for the location of the exhibit. But, the component U is part of a set of components that are spatially collocated. Consequently, the relationship $U \rightarrow A_{in}$ can be connected to the contour of the set rather than to the user, leaving the designer free to decide which component of this set to localise. However, since the input adapter is not further explored at this point, the ASUR++ diagrammatic representation represents this adapter as a square rather than as a circle.

² The former case can also be captured using ASUR++, but space limitations prevent us from considering it in this paper. Moreover, if output adapters are placed in the gallery infrastructure, there would (probably) be a single screen for each exhibit, a less flexible solution if there are many concurrent visitors with very different interests and information needs.

Reasoning at a perceptual level

This analysis relies on the ASUR++ components' perceptual characteristics, including the perceptual environment (i.e. where the perception takes place), the sense(s) used, and the ability of the adapter to share the information it provides among one or several users (see Table 1, section 2.2).

The perceptual senses we may envision for this design are the visual and auditory senses for both adapters; in addition, the haptic sense might be used to convey the path to follow. Given that two adapters are used, different combinations of these senses might be used to realise the system. With respect to location of perception, design issues will be highly dependent on choice of modality. For example, if we consider the situation in which both information, related to the path and the exhibit, are visually conveyed via a palmtop device, s/he will have to look alternatively at the device carrying information about the path (A_{out2}), at the one carrying information about the exhibit (A_{out1}) and at the physical exhibit (R_{object}). This is an example of perceptual incompatibility, an AR ergonomic property introduced in (Dubois et al 2002a), which may annoy the visitor. Projecting information into the same visual field as the actual exhibit may lead to a partial occlusion of the real exhibits, and is probably not a good solution in a museum context, but would be extremely interesting in an Computer Assisted Surgery system for example, where a surgeon may wish to have a permanent view of the patient and to perceive collocated guidance information for the surgical tools s/he is manipulating (Dubois et al 2001).

Other aspects may also have to be considered, regarding the context of use of the system being designed. The gallery is likely to have a number of visitors and visitors often operate in groups. The choice of audio for a personal output adapter might be more intrusive than a visual adapter, disrupting social interaction among other visitors. However, audio might also promote "co-visiting".

Finally, taking into account the number of users that should be able to access to the information leads us to consider three cases: restricting to one user only, allowing a group of users to access to the data or to broadcast the information to every visitor present in the same place. Again, the context of use of the system will greatly impact on the choice of one of these

possibilities. For example, if we choose to limit access to the data to one person only, the consequence is that a group guided by a leader is not possible because the members of the group can't read the data related to the exhibit.

Note that the use of ASUR++ does not offer a way of resolving these design choices (that remains a question of usability evaluation and/or the use of appropriate guidelines), but it does provide a means of expressing the aspect of the system to which the choices apply, viz., the physical realisation of the output adapters between system and user.

Reasoning at a cognitive level

This analysis is based on ASUR++'s characterisation of the language and the frame of reference of the representation conveyed by a relationship (see table 1, section 2.2).

The languages that may be used to express information about the exhibit include text, graphics (2D or 3D) and speech. In addition, path information may be conveyed by sounds (non-speech audio) or tactile stimulation. The resulting possible combinations are of course highly dependent on the choices made in the previous phase, concerning the human senses the adapter will exploit.

The usability of particular representational combinations also has to be considered. This can be assessed, of course, via interaction design patterns, analytic evaluation in terms of ergonomics principles and/or psychological theories modelling the cognitive processes of the users, or empirical user studies. For example, if we consider that the path is provided using a textual language rather than with graphics, the user has to interpret the presented textual information in terms of her/his physical 3D environment. This interpretation introduces a cognitive discontinuity (an AR ergonomic property introduced in (Dubois et al 2002a)) which, in this case, may complicate the task for the user.

The frame of reference of the representation of the information related to the exhibit must be presented from a user's point of view so that s/he can access it. The path may be expressed in different frames of reference: a user-centred point of view (e.g., "turn right at the urn") or in a global reference scheme (e.g., a map). The impact of choosing either one or the other is not immediately apparent and again may need observational studies, design patterns or analytic studies, in collaboration with usability professionals or psychologists.

3.3.2 Using Only One Output Adapter

Adapter elicitation level

The only differences between this ASUR++ description, shown in figure 3, and the one presented in Figure 2, is (i) the use of a single output adapter and (ii) the existence of two ASUR++ relationships between the user and this adapter. These two relationships indicate that the adapter provides to the user information related both to the path and to the exhibits. Using only one output adapter instead of two will have an impact on the design possibilities identified in the following phases of the ASUR++ based reasoning process.

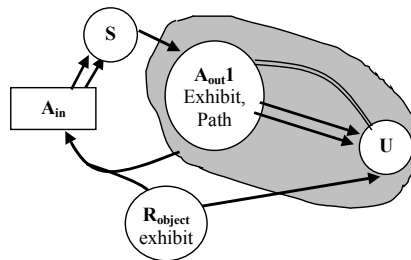


Fig. 3. Partial ASUR++ description of the scenario using one output adapter.

Reasoning at a perceptual level

The limitation to one adapter restricts the possible design solutions and forces trade-offs. First of all, haptic feedback is no longer a viable alternative since it is unlikely to be suitable for information related to the exhibits. Significant compromises will also have to be made if either audio or visual techniques are used on their own.

Reasoning at a cognitive level

Consider the case of a visual output adapter. The likely possible languages are either text or graphics. If text is used, there remains the problem of a cognitive discontinuity when conveying path information textually, but presenting all the information (path and exhibit) via the same representation might be considered to offer a form of coherence in the output interaction. An observational study might be conducted to assess this hypothesis.

More importantly, using the same adapter for both information streams may interact badly with the physical properties of the adapter. For example, it may be difficult to present all the relevant information concurrently via a palm-sized display. Once again, the context of use of one adapter is proven to be important to take into account when envisioning a design solution.

3.4 Reasoning About Input Design Solutions

Thus far we have explored the design space of system's output to the user. We now focus on the design of input, that is, the ways the computer system will get information from the physical world and from the user. Note that the input aspect of the user's interaction with the system is only a subset of the whole input design.

As shown in figure 1, the system needs to be aware of the spatial relationship of the visitor to the exhibits. The most direct design solution is to utilise two input adapters, one dedicated to localising the exhibit, while the second is dedicated to localising the visitor. We describe this solution in the next section. For the purposes of our example, we assume the “single adapter” output design.

3.4.1 Using Two Input Adapters

In order to be aware of the visitor's location in the museum an input adapter (A_{in1}) is required to retrieve the position of the visitor in the museum ($U \rightarrow A_{in1}$) (more exactly the set of collocated ASUR++ components that includes the user), and to transfer the position to the computer system ($A_{in1} \rightarrow S$). In addition, a second input adapter (A_{in2}) is required to locate the exhibit ($R_{object} \rightarrow A_{in2}$) and transfer the location to the computer system ($A_{in2} \rightarrow S$). The ASUR++ description of the overall system using one output adapter and two input adapters is presented in the figure 4.

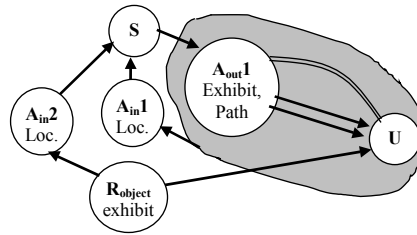


Fig. 4. ASUR++ description of the scenario using one output adapter and two input adapters.

Considering the concept of localisation, the system has to deal with two inputs: the one related to the user's location and the one related to the exhibit's location. Matching these two sources of information may be a problem for the computer system and is similar to a discontinuity problem on the user's side. Solutions to address this problem can be driven by the solutions envisioned when a discontinuity problem is identified on the user's side. It would thus be better to:

- Use only one reference scheme in which to encode the location information provided by the adapters (similar to a cognitive discontinuity problem)
- Track only one entity (similar to a perceptual discontinuity problem)

Addressing the first kind of problem is relatively easy. One global reference scheme may be used. The second problem is harder to address. On the user's side, this led us to group the two output adapters into only one. We explore this solution in the next section.

3.4.2 *Using One Input Adapter*

In this present case, grouping the two adapters may be achieved by using one of the following mechanisms:

- Avoiding the need of the relationship between the exhibit and the input adapter, or of the relationship between the user and the input adapter.
- Grouping the exhibit with the user or the exhibit with the input adapter.

Avoiding the need of a relationship

Let us first consider the localisation of the exhibit. A solution could be to use a static model of the positions of the exhibits. In fact this is achievable by adding a field in the exhibit database holding the location of the exhibit in the museum. Consequently, having the position of the visitor in the

museum is sufficient to find in the database the exhibit which has the nearest coordinates and thus to display the right information. To represent the existence of a virtual model of the physical exhibit, we refine the ASUR++ component S (computer System) by adding a decoration to the S node: V- R_{object} (virtual model of the real entity associated with the component R_{object}). The new ASUR++ diagram is presented on the left-hand side of figure 5.

Avoiding the need of the visitors' localisation could be achieved in two ways: either the display of information is time dependant or the user is static and the exhibit moves in front of him. In fact, in the first case, time dependent display of information is similar to providing the computer system with a virtual model of the visitor's motion based on the time. But, the visitor might be rapidly lost if he spends more time than planned in front of an exhibit. This solution is thus quite risky. The second solution seems to be more reliable. Its technical realisation is another question. However, in this futuristic situation the user and the devices s/he is carrying would be static and the exhibits would automatically pass in front of the visitor. The ASUR++ diagram representation of this design variant is presented on the right-hand side of figure 5.

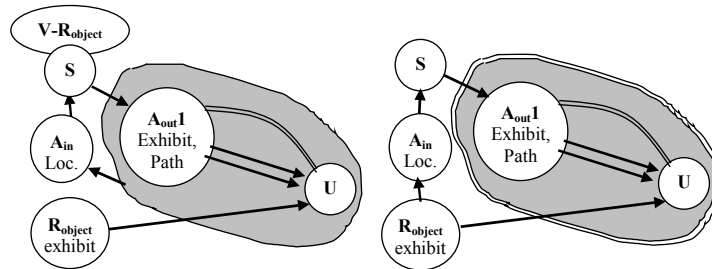


Fig. 5. Complete ASUR++ description of the scenario using one output adapter and one input adapter when avoiding the localisation need of the exhibit (left) or of the visitor (right).

Grouping mechanism

The role of the grouping mechanism is to physically link a component with an input adapter, so that when this adapter sends information to the system about another component, the system also knows where the information comes from.

One way of implementing this mechanism consists of installing near an exhibit the input adapter responsible for the visitor's localisation. This is represented by a physical collocation

relationship between the exhibit and the input adapter ($R_{object}=A_{in}$). The set of components that make up the exhibit and the input adapter remain static. The system can determine the visitor's position by associating the visitor's identity with the exhibit to which the input adapter is collocated. Thus the system can display the right information to the user. To be localised, the visitor, or more exactly the set of components that includes the visitor, has to come near the exhibit. In terms of ASUR++, when the visitor comes near the exhibit, it triggers the transfer of information from the visitor to the input adapter. The following relations emerge: $U \Rightarrow A_{in}$ and $U \rightarrow A_{in}$. The ASUR++ diagram of this system is shown in the left part of figure 6.

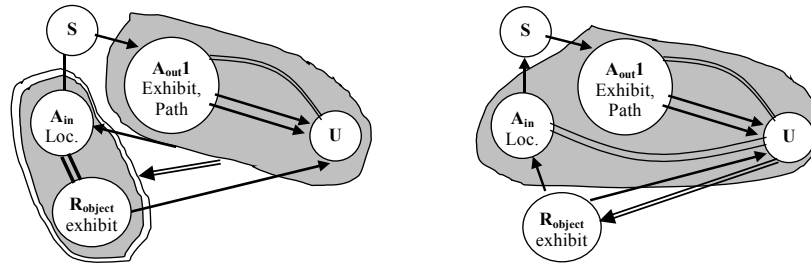


Fig. 6. Complete ASUR++ description of the scenario using one output adapter plus one input adapter grouped with the exhibit (left) or with the visitor (right).

Examples of devices that might play the role of the component A_{in} as described here are motion detectors or an rfid tag and sensor. In the later case, the relationship denoting the transfer of information between the user and the input adapter requires the addition of the rfid emitter to the set of components carried by the visitor. The relationship between the "visitor set" and the "adapter set" is $U \rightarrow A_{in}$. The designer has to think about which component of the "visitor set" to embed: the visitor or the output adapter.

Another way of applying this mechanism is to group the input adapter with the visitor. The information provided to the system by the adapter refers here to the localisation of an exhibit, given that the relationship between the adapter and the visitor is known and fixed. In this case, the input adapter has to be physically collocated with the user ($A_{in}=U$) and is added into the mobile "visitor's set". When the visitor approaches the exhibit, it triggers the exchange of information between the exhibit and the input adapter responsible for the localisation of the exhibit: $U \Rightarrow R_{object}$ and its

associated triggered relationship $R_{\text{object}} \rightarrow A_{\text{in}}$. The right side of figure 6 illustrates this alternative. A candidate A_{in} component for this version would be an rfid on the exhibit or, more elaborately, a camera with an image processing module added in the computer system to automatically recognise the exhibit in front of the camera.

3.5 Scalability of Design Solutions

Consider the design solutions shown in figure 6. Both of them satisfy the system's functional requirements. So far we have only considered the system with respect to a single user. However, the context of use of this mobile mixed system is a museum, which may involve multiple visitors and, of course, a number of exhibits. Thus, reasoning at a larger scale means in this case considering the existence of several users and exhibits at the same time. Describing the large-scale system with ASUR++ will result in multiple U components (visitors) and multiple R_{object} components (exhibits). Given that these components are organised as collocated sets, an ASUR++ description will be based on the use of several of these sets, generating as many as necessary to characterise the system at the new scale.

When the input adapter is collocated with the exhibit (left part of figure 6), multiple exhibits will result in multiple input adapters, one for each exhibit. Multiple visitors will require multiple output adapters. The left side of figure 7 shows this ASUR++ description. On the other hand, when the input adapter is collocated with the user, multiple exhibits have no influence on the devices to connect to the system, but multiple users result in the need for multiple adapters for input and output as illustrated on the right side of figure 7.

On the basis of these large scale descriptions, it is possible to assess alternative solutions by considering aspects such as implementation complexity or cost. Indeed, the description reveals the number of required adapters for input and output and also indicates whether exhibits must be modified or users equipped with devices to wear or carry. For example, if the number of exhibits is very high in comparison to the number of simultaneous visitors, then the right-hand description of figure 7 may be better. This is also the case if the exhibits of the museum are subject to be frequently removal or change.

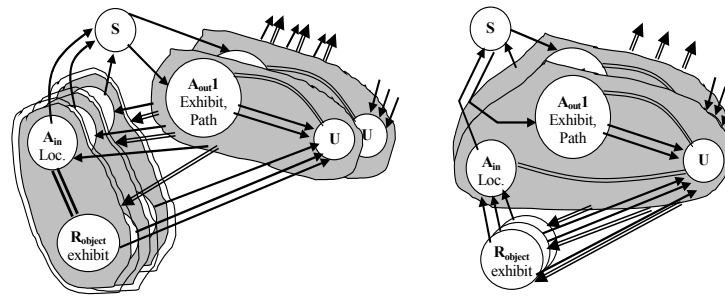


Fig. 7. ASUR++ description of the large scale version scenario using one output adapter and one input adapter when grouping the input adapter with the exhibit (left) or with the visitor (right).³

4 ASUR++: Utility and Usability

So far, we have presented a notation for the design of mobile mixed systems. We have presented and analysed several design solutions for an augmented museum gallery, expressed using this notation. For this small scenario we came up with a set of nine different models. A designer would next have to develop a suitable solution for his/her application, based on these models and on the use of complementary existing methods and approaches. To do so, other aspects may have to be taken into account, including:

- Evaluation of ergonomic properties, to elicit the best design solution of the ones identified, from a user's perspective;
- Software engineering, to support the software realisation and reusability of components;
- Sensor design and development, to identify the best suitable devices and the way they should be combined;
- Use of a methodical approach, to be sure of having explored the whole design space.

All these aspects may be related to ASUR++, as illustrated in the following section that briefly reports other areas we explored, or plan to, with ASUR++.

³ For sake of clarity, the ASUR++ relationship between the second user and the input adapter or exhibits are only partially represented.

4.1 ASUR++: a bridge between different design perspectives

4.1.1 ASUR++ and ergonomic properties

ASUR++ diagrams clearly highlight the different elements of the user's interaction with the whole mobile mixed system: manipulation of physical objects, perception of information on a device, input of data on another, etc. These aspects of the user's interaction appear as the set of ASUR++ relationships connected to the user on the one hand and to a real object or adapter on the other. Usability evaluation of a mobile mixed system will be based on these aspects, in particular on the relationships, and the components that contribute to the relationships, as well as their respective ASUR++ characteristics (see table 1, section 2.2).

Let us take the example of perceptual compatibility in the case of the modelling of the scenario illustrated in figure 4. Compatibility at the perceptual level, as defined in (Dubois et al 2002a), denotes how easy or difficult it is for the user to perceive all the concepts provided by the system at a given time. Based on an ASUR++ model, this ergonomic property relies on the analysis of the perceptual environments of every component involved in the interaction, where a perceptual environment is defined by the sense used for perceiving and the location where the user has to focus to get the information.

In figure 4 the components involved in the interaction are the adapters for output, carrying information about the path and the exhibit, and the physical exhibit. The human sense required for perceiving the exhibit is vision and the location is the site of the exhibit. If the adapter presents the information about path and exhibit via text or graphics, the human sense required will also be vision. In addition, if the adapter is a PDA, the location where the visitor has to focus to get this information will be his/her hand. Consequently, this will be a case of perceptual incompatibility since the visitor has to focus simultaneously on two separate locations. An alternative could be a speech-based output adapter, so that the visitor could observe the exhibit and listen to the explanations simultaneously.

Other ergonomic properties and their expression in ASUR++ diagrams and characteristics are introduced and illustrated in (Dubois et al 2002a).

4.1.2 ASUR++ and software engineering

Associating an ASUR++ description with the underlying software is intended to enable a designer to assess the feasibility of a user interface design given software development constraints and to evaluate the impact on the software design of a user interface solution. To study this linking, we have explored linking ASUR++ with UMLi, an extension of UML, the Unified Modelling Language. UMLi augments traditional UML diagrams with a *User Interface diagram*. This diagram provides additional support for interaction classes by visually representing containment and four main interaction elements (inputter, displayer, action invoker and editors). UMLi also adds specific object flows and states in order to tightly couple this structural model with a behavioural model represented by a UML activity diagram.

ASUR++ diagrams describe and model the physical environment and a user's interaction, while UMLi diagrams contribute an abstract user interface, with the underlying functional core modelled in UML. Links clearly exist between UML and UMLi. We have taken a first step towards a model-based design environment for mobile mixed systems by adding links between ASUR++ and UMLi (Dubois et al 2002b).

In addition to this connection between ASUR++ and UMLi, in (Dubois 2001) we have also established links between ASUR++ diagrams and a software architecture model. This is also crucial for mobile mixed systems, in order to promote the development and reusability of software components for mobile mixed systems. Such components might include specialised visualisation, localisation, or interaction components.

4.1.3 ASUR++ and the design of sensors

Sensors are represented in ASUR++ by adapters (components A_{in} and A_{out}). Indeed, these components support the communication between the physical and digital worlds. Results in the field of sensor design may thus be linked to ASUR++ diagrams. Choosing the best suitable adapter may then rely on its ASUR++ characteristics but also on complementary studies concerning the integration of a sensor into the user's environment or the suitability, in terms of usability, technical feasibility or cost, of combinations of adapters.

4.1.4 ASUR++ support of a systematic approach

Though no detailed method of use as yet exists for ASUR++, we have shown via the examples in this paper that it can help to characterise different design solutions for a given mobile mixed system scenario. Developing a systematic method based on ASUR++ will contribute to this last identified crucial aspect of mobile mixed system design.

4.2 Usability of the notation ASUR++

The use of ASUR++ may thus appear as a first design rationale approach for Mobile Mixed Systems. Indeed, design rationale is still lightly defined, but, as stated in (Lee & Lai 1996), the major interests of a design rationale are the ability to

- Justify a designed artefact ;
- Explore the design space ;
- Understand the underlying human-computer interactions.

Justifying an artefact designed with ASUR++ may rely on the ergonomic or software analysis as briefly reported in section 4.1. In addition, since mobile mixed systems combine physical and digital artefacts, the user's interaction is far more complex than in a traditional interactive system. We have illustrated in section 2.1 that ASUR++ diagrams highlight the different user's interaction facets, thus helping the designer to understand the complex interaction induced by the mobile mixed system being designed. Finally, as mentioned in section 4.1, we have not yet any ASUR++ method of use which would ensure the systematic exploration of the design space. Consequently we have not proven that ASUR++ was capable of supporting the exploration of the design space. But at least, we have illustrated in section 3 that ASUR++ has the capability of expressing different design solutions.

(McLean et al 1996) also highlight that a design rationale has to be an explicit representation of the design process and decisions, so that every members of the design team are able to read and understand it and that it serves as a vehicle for communication. The question of the usability of a ASUR++ is thus to consider. Even if ASUR++ is in its early stage of development, we report in the following paragraphs some works we did to study and strengthen ASUR++ usability.

4.2.1 Using ASUR++ in an early design process

During the summer of 2002, we supervised an Information Technology MSc project to design a mobile information system for newcomers to the University of Glasgow. The student, who had never encountered ASUR++ before, was given a tutorial on its use and invited to employ ASUR++ in the design process. The choice of using ASUR++, and the particular purpose for which ASUR++ might be used, was left to the student's discretion. The initial design was developed without ASUR++ and involved the use of GPS to determine the location of the user. As it turned out, GPS didn't provide sufficiently accurate location information and the design had to be re-thought. It was at this point that the student, on her own initiative, chose to use ASUR++ to develop a new solution and to compare it with the GPS solution, ⁴ and without further help successfully modelled the two solutions.

This first informal study is promising in terms of the usability and usefulness of the notation, since an inexperienced user was able to use it to express a genuine design issue and its resolution. Further studies will have to be performed to investigate the relative advantages compared to other notational solutions, the expressive range and scalability of ASUR++ diagrams and the relationship of the use of ASUR++ to the discovery of design problems and good solutions.

4.2.2 An ASUR++ editor

Finally, since the use of the ASUR++ notation relies mainly on graphical representations, i.e., diagrams, we are currently developing a graphical editor. As a first step, this editor provides the designer with a graphical environment to draw ASUR++ diagrams, specify the components and relationships, and save and reload existing ASUR++ diagrams. Figure 8 presents a snapshot of an early version of the editor.

⁴ The second solution involved delivering to the user photos of key landmarks along the path to the next building, using user input to determine their location near each photo.

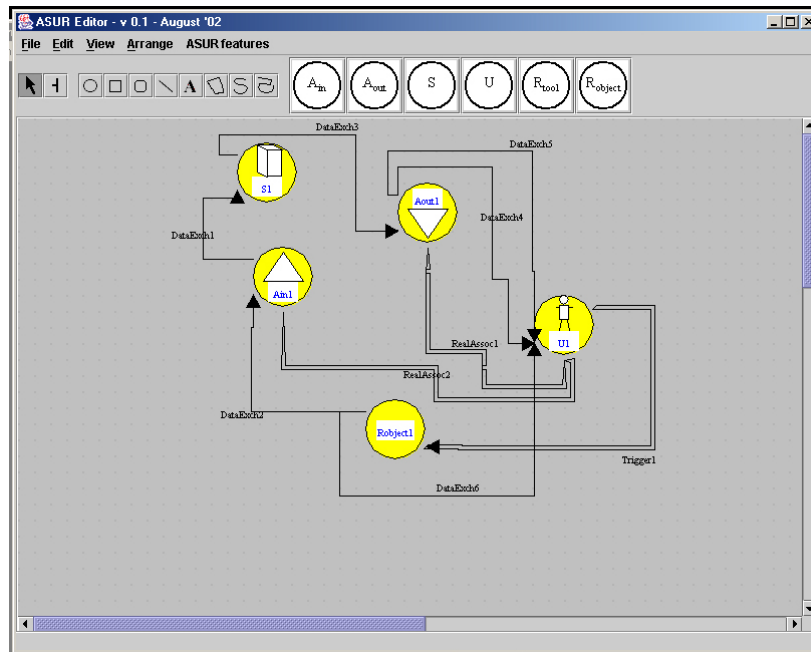


Fig. 8. A snapshot of an early version of the ASUR++ Editor

Currently we are working on adding two additional linked editable views, for physical objects only (an “architectural view”) and for computational objects only (an “application model” view). It is also intended to provide the designer with predefined interaction patterns expressed in ASUR++. Future work will include the ability to:

- export a design in UML and
- verify ergonomic properties expressed in terms of ASUR++ characteristics.

5 Conclusions and Future work

ASUR++ offers a way of describing and reasoning about the intersection between the physical and the digital in mobile mixed systems. ASUR++ is intended to provide a resource for analysts. It can be used to systematise thinking about design problems for mobile mixed systems, as demonstrated in the paper. Several design solutions have been described using the same modelling approach, enabling easy comparisons. The notation, with its underlying semantics,

encourages the analyst to think about design issues in a particular way. In particular, ASUR++ prompts the analyst:

- to study the spatial and other physical relationships amongst the entities involved in the system: physical objects, adapters and users,
- to study the scalability of the design solutions.

As we have pointed out in a previous study (Bellotti et al., 1996), "Like a screwdriver, a modelling approach concentrates force (of reasoning) in the appropriate area; it does not mean that there is no role for the artisan and no element of skill and judgement involved." Thus, work remains to be done both to develop the expressiveness of ASUR++ and its method of use as well as its integration into the larger activity of mobile mixed system development. As we have shown in Bellotti et al (1996), the use of multiple modelling techniques extends the range of perspectives on the design problem. Diverse notations can work in concert and in a complementary fashion to identify and propose corrections to design flaws. Connecting ASUR++ with other existing methods will allow the designer to validate solutions with respect to different criteria. Further work needs to be done in this direction, but we have already proven that ASUR is a good candidate to support and interconnect different methods. That is why we believe that ASUR++ constitutes a promising starting point for developing a design method that bridges between the required physical description of the sensed environment, the software engineering of the underlying digital application and user-centered studies.

Finally, another research avenue involves identifying recurrent ASUR++ diagrams that can be generalised and applied across different application domains. Such diagrams might describe reusable interaction design patterns for mobile mixed systems. Furthermore, such interaction design patterns expressed using ASUR++ may then be translated in terms of software architectural patterns, such as the ones we presented in (Nigay & Coutaz 1997), providing assistance with realising the implementation of the patterns.

6 References

- Abowd, G., D., Atkeson, C., G., Hong, J., Long, S., Kooper, R., Pinkerton, M. 1997. Cyberguide: a mobile context-aware tour guide. *Wireless Networks* Vol. 3, 5, Special issue: mobile computing and networking. Kluwer Academic Publishers, Hingham, MA, USA (1997) 421-433.
- ANSI/IEEE Standard 729-1983. 1989. *Software Engineering Standards*. IEEE, New York.
- Azuma, R., T. 1997. A survey of Augmented Reality. In *Presence: Teleoperators and Virtual Environments* Vol. 6, 4, (1997) 355-385.
- Bellotti, V., Keith Edwards. 2001. Intelligibility and Accountability: Human Considerations in Context Aware Systems, in *Human Computer Interaction Journal*, 16:2-4, Special Issue on Context Aware Computing.
- Bernsen, O. 1994. Foundations of multimodal representations. A taxonomy of representational modalities, in *Journal Interacting with Computers*, Vol. 6, 4, (1994), 347-371.
- Barry Brown, Ian MacColl, Matthew Chalmers, Areti Galani, Cliff Randell, Anthony Steed. 2002. Lessons from the lighthouse: Collaboration in a shared mixed reality system. In press. To appear in: *Proceedings of CHI '2003*
- Cheverst, K., Davies, N., Mitchell, K., Friday, A., Efstratiou, C. 2000. Developing a context-aware electronic tourist guide: some issues and experiences. *Conf. Proc. of CHI'2000*. Netherlands. ACM Press, NY, USA, (2000) 17-24.
- Dey, A., K. 2000. *Providing Architectural Support for Building Context-Aware Applications*, PhD thesis, College of Computing, Georgia Institute of Technology, (2000).
- Dubois, E., Nigay, L., Troccaz, J., Chavanon, O., Carrat, L.: Classification Space for Augmented Surgery, an Augmented Reality Case Study. In *Conf. Proc. of Interact'99* (1999) 353-359.
- Dubois, E.: *Chirurgie Augmentée : un Cas de Réalité Augmentée ; Conception et Réalisation Centrées sur l'Utilisateur*. PhD Thesis University of Grenoble I, France (2001) 275 pages.

- Dubois, E., Nigay, L., Troccaz, J., Carrat, L., Chavanon, O.: A methodological tool for computer-assisted surgery interface design: its application to computer-assisted pericardial puncture. In Westwood, J. D. (ed.): Conference Proceedings of MMVR'2001. IOS Press (2001) 136 - 139.
- Dubois, E., Nigay, L., Troccaz, J.: Assessing Continuity and Compatibility in Augmented Reality Systems. To appear in Int. Journal on Universal Access in the Information Society, Special Issue on Continuous Interaction in Future Computing Systems. Springer-Verlag, Berlin (2002).
- Dubois, E., Pinheiro da Silva, P., Gray, P., D., "Notational Support for the Design of Augmented Reality Systems", to appear in the proceedings of the international conference DSV-IS'02, Germany, (2002), 8 pages.
- Feiner, S., MacIntyre, B., Seligmann, D. 1993. Knowledge-Based Augmented Reality. Communication of the ACM, Vol. 7 (1993) 53-61.
- Gray, P., Salber, D., "Modelling and Using Sensed Context Information in the Design of Interactive Applications", Conference Proceedings of EHCT'01, Springer Verlag, Little, M. R. & Nigay, L. (eds), Canada, (2001), p. 317-336.
- Ishii, H., Ullmer, B. 1997. Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms. In Conference Proceedings of CHI'97. ACM Press (1997) 234-241.
- Lee, J., Lai, K.-Y., What's in Design Rationale?, in Design Rationale: Concepts, Techniques and Use, (chapter 2), Lawrence Erlbaum Associates, Moran, T., P. & Carroll, J., M. (Eds), (1996).
- Mackay, W.E., Fayard, A.-L., Frobert, L., Médini, L. 1998. Reinventing the Familiar: an Augmented Reality Design Space for Air Traffic Control, In Proceedings of CHI'98, LA, (1998) 558-565
- MacLean, A., Young, R., M., Bellotti, V., M., E., Moran, T., P., Questions, Options and Criteria: Elements of Design Space Analysis, in Design Rationale: Concepts, Techniques and Use, (chapter 3), Lawrence Erlbaum Associates, Moran, T., P. & Carroll, J., M. (Eds), (1996).
- Nigay, L., Coutaz, 1997, J.: Software architecture modelling: Bridging Two Worlds using Ergonomics and Software Properties. In Palanque, P., Paterno, F. (eds.): Formal Methods in Human-Computer Interaction, Springer-Verlag, Berlin Heidelberg New York (1997) 49-73.

- Noma, H., Miyasato, T., Kishino, F. 1996. A palmtop display for dextrous manipulation with haptic sensation. In Conference Proceedings of CHI'96. ACM Press (1996) 126-133.
- Paterno, Fabio. Model-Based Design and Evaluation of Interactive Applications. Springer (1999). ISBN 1-85233-155-0
- Pinheiro da Silva, P., Paton, N.W., UML*i*: The Unified Modeling Language for Interactive Applications, in Proc. of the 3rd Conf. UML'00, UK, p. 117-132, Springer, (2000).
- Rekimoto, J., Katashi N. 1995. The World through the Computer: Computer Augmented Interaction with Real World Environments. In Proc. of UIST'95. ACM Press (1995) 29-36.
- Renevier, P., Nigay L. Mobile Collaborative Augmented Reality, the Augmente Stroll. In Little, R. Nigay, L. (eds): Proceedings of EHCI'2001, Revisited papers, LNCS 2254. Springer-Verlag, Berlin Heidelberg New York (2001) 315-334.
- Schmidt, A., Beigl, M., Gellersen, H., W., There is more to context than Location; Environment sensing technologies for adaptive mobile user interfaces, Workshop on Interactive Applications of Mobile Computing IMC'98 (1998).
- Stevens, P., Pooley, R., Using UML: Software Engineering with Objects and Components. Addison-Wesley, (2000).
- Troccaz, J., Lavallée, S., Cinquin, P., Computer augmented surgery, Human Movement Science 15, (1996) 445-475.
- Van Harmelen, Mark. Object Modeling and User Interface Design: Designing Interactive Systems. Addison Wesley (2001). ISBN 0-201-65789-9.
- Winograd, T. 2001. Architectures for Context, in Human-Computer Interaction Journal, 16:2-3, Special Issue on Context Aware Computing, (2001).