A Design-Oriented Information-Flow Refinement of the ASUR Interaction Model

Emmanuel Dubois[1] and Philip Gray[2]

[1] IRIT-LIIHS, University of Toulouse, France
[2] GIST, Computing Science Department, University of Glasgow, UK
pdg@dcs.gla.ac.uk, Emmanuel.Dubois@irit.fr

Abstract. The last few years have seen an explosion of interaction possibilities opened up by ubiquitous computing, mobile devices, and tangible interaction. Our methods of modelling interaction, however, have not kept up. As is to be expected with such a rich situation, there are many ways in which interaction might be modelled, focussing, for example, on user tasks, physical location(s) and mobility, data flows or software elements. In this paper, we present a model and modelling technique intended to capture key aspects of user's interaction of interest to interactive system designers, at the stage of requirements capture and early design. In particular, we characterise the interaction as a physically mediated information exchange, emphasizing the physical entities involved and their relationships with the user and with one another. We apply the model to two examples in order to illustrate its expressive power.

Keywords: Mixed Interactive Systems, User's Interaction Modelling, Requirements Capture, Information flow characterisation, Design Analysis, Interaction Path.

1 Introduction

The last few years have seen an explosion of interaction possibilities opened up by ubiquitous computing, mobility, and tangible interaction. Techniques for modelling interaction in and with such systems, however, have not kept up. As is to be expected with such rich domains as these, there are many ways in which interaction might be modelled, focussing, for example, on user tasks, physical location(s) and mobility, data flows or software elements. The current situation with respect to such models presents designers with both feast and famine On the one hand, there is a large and bewildering variety of descriptive models available to us, originating from the world of conventional interactive systems: task models, models of interaction objects, software models and existing spatial models. On the other hand, we have very few descriptive models developed for capturing augmented reality, mobile, tangible and ubiquitous applications (hereafter, we refer to these in this paper as 'mixed interactive systems'[1]). These range from ASUR [5, 7], the basis of the work presented in this paper, that models interaction in its physical and digital aspects, through the Model of Mixed Interaction (MMI) [4] which focuses on interaction modality to MCPRD [8], a software architecture model for mixed reality.

As with software modelling, there is no single, monolithic model suitable for all software development purposes. Like UML, different models are useful for different purposes at different stages in the development process. However, unlike UML, designers of mixed interactive systems do not yet have a well-found set of models that are generally accepted, well-integrated with one another and that fit into a development process. Nevertheless, the first stages in this creating such a set are underway. Thus, ASUR, for example, now fits into a suite of models and into a development process. Figure 1 illustrates the approach.

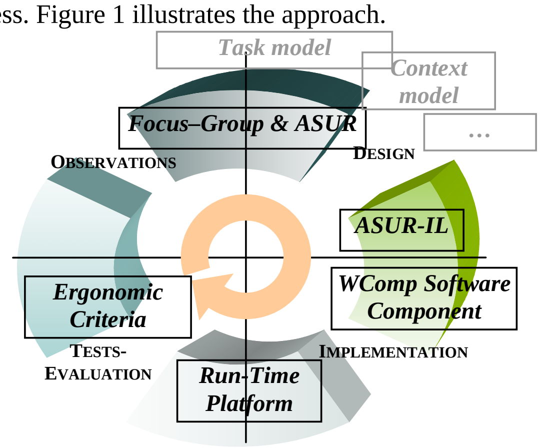

Fig. 1. An integration of models & design method for Mixed Interactive Systems (i.e. ubiquitous, mixed and mobile applications).

In this paper, we present an interaction model, based on ASUR, that is intended to better express the user's experience of the physical environment in order to communicate information to and from a computer system. The goal is to capture aspects of that experience that are:

- relevant to requirements capture and to the early stages of design,
- for the assessment, comparison and discovery of designs;
- without overly complicating the analyst's or designer's task.

Although our approach is presented as an extension of ASUR, its fundamental features are independent of ASUR and could be used on their own or incorporated into other similar modelling frameworks.

Following a short overview of ASUR, we present the new interaction model, illustrated with a small example of its application. We then introduce the notion of interaction groups and use it to analyse the design options available for another

[1] In this paper our use of the term 'mixed interactive systems' is merely intended to informally group systems that fall under the commonly used terms, 'augmented reality', 'mobile systems', 'tangible systems' and 'ubiquitous systems'. By so doing we do not imply any common definition.

example interaction technique. After briefly comparing our model to related approaches, we finish by drawing some general conclusions and considering future work.

2 ASUR Overview

ASUR is a notational-based model for describing user-system interaction in mixed interactive systems. ASUR is intended to help in reasoning about how to combine physical and digital “worlds” to achieve user-significant results. It is used in addition to a traditional user-system task description in order to identify objects involved in the interaction and at the boundaries between the two worlds. Adopting a user’s interaction point of view, the model is helpful in expressing the results of the requirements analysis and addressing the global design phase of a mixed interactive system. Indeed ASUR supports the description of the physical and digital entities that make up a mixed system, including adapters (Ain, Aout) bridging the gap between both digital and physical worlds, digital tools (Stool) or concepts (Sinfo, Sobj), user(s) (U) and real objects involved as tools (Rtool) or constituting the task focus (Robj). In addition, directed relationships (arrowed lines) express physical and/or digital information flows and associations among the components. To better specify these elements, viz., ASUR components and relationships, a number of characteristics have been identified, including such design-significant aspects as:

- For components: the location where the information carried by the component is perceivable or modifiable (top of table, half of the room, …), the sense or action required so that the user perceive or act on it (hearing, sight, touch, physical action, …) etc.
- For relationships: The dimensionality of communicated information (2D, 3D, stereoscopic, …), the type of language used (text, graphic, image, …), the point of view (ego/exo-centric, …), etc.

The ASUR model in Figure 2 shows the interaction between a user and a 3D digital environment, using a “magic wand”. The user, User_0, handles and moves a physical wand (Rtool) that is tracked by a camera (Ain). The camera sends the position of the wand to a digital Activator (Stool) that may act on other digital entities. It also sends the position to a pointer (Sinfo) object. The pointer is in fact a representation of the end point of the physical wand (dashed arrow); this representation is useful for providing interaction feedback. If the functionality is activated, data such as the rotation angle of the wand is transferred to a 3D volume object (Sobj). Finally the 3D volume, the activator and the pointer are displayed on a screen (Aout) to be perceived by the user (U). A more detailed description of this example, including all the modelled characteristics, is presented in [7].

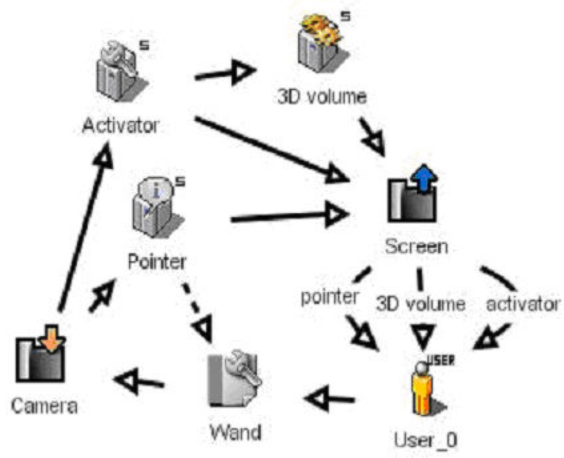

Fig. 2. An ASUR diagram example

An ASUR description of a mixed interactive system is thus useful in the early design phases to support the exploration and analysis of interaction designs. However it abstracts away features of software design and its implementation. Those two aspects are supported by a complementary model, ASUR-IL that stands for ASUR-Implementation Layer [6] and are out of scope of this paper.

Although ASUR captures the basic features of an interaction, it does not have the expressive power to say very much about the user's interactive activity or experience. It is this aspect that we have modified and which is presented in the remainder of this paper.

3 Modelling the Means of Interaction

One can use an application to communicate and/or receive information or to perform work (e.g., act on the world via the application). In this paper we use the term *interaction* to denote this kind of activity with an application and we characterise it as a sequence of information exchanges and/or actions between one or more users and one or more systems.

As described in section 2, any interaction originating from a user and that has at least one digital recipient (e.g., an application), or conversely, is mediated by an *adaptor*. Thus, to enter a name into an account record, one may need to use a keyboard. An interaction between two physical entities may be *mediated* by other physical entities. For example, one may use a stylus to interact with a PDA touch screen.

A sequence of such entities and their relationships used in an interaction forms an *interaction path*. The interaction exchange or action between elements in the path is conducted via one or more *interaction channels* along which information or action is communicated.

We begin this section with the description of a simple example. This is followed by a summary of the key elements in our ASUR extension which are then illustrated by applying them to the example. We then identify additional properties that can be expressed and explored on the basis of our model extension.

3.1 A Running Example: A Spongy Switch

Let us consider a very simple, if somewhat unusual, example: a “spongy switch”. An appropriately instrumented sponge might be used to communicate to some application one of two states: state 1 when the sponge is compressed and state 2 when the sponge is left uncompressed. At this stage in the analysis we do not yet specify how the compression is sensed, but merely that it can be. Figure 3 gives a simple ASUR diagram showing the entities and channels involved in this interaction path.

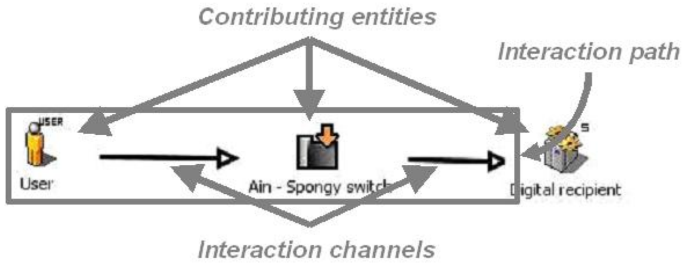

Fig. 3. ASUR diagram of a Spongy Switch.

There are two contributing entities: a user and an instrumented sponge. The arrowed lines indicate that the user must act on the sponge in some way in order to change its state of compression and the resulting change of compression can be transmitted to some digital recipient.

3.2 Interaction Entities

The ASUR model of figure 3 includes three entities in the interaction path: the user, the spongy switch and the digital recipient of sponge state changes. In order to further describe interaction paths, we distinguish two types of entities: adaptors and mediating entities.

3.2.1 Adaptors

By definition, an adaptor must perform a transform of the information on the incoming channel to that on the outgoing channel, one channel belonging to the physical environment and the other to the digital world.

This transformation can be simply an analogue to digital transformation but, in more abstract formulations of the adaptor, the transformation may perform other operations as well. In fact, the analogue to digital conversion is part of the definition of an adaptor. However, it’s often useful to bundle this function with both sensing on the one hand and useful low-level transformations on the other hand. The level of abstraction is not fixed by our modelling technique but by the use to which the description is put.

For example, the adaptor used to localise the wand in figure 2 may either be in charge of grabbing a picture of the scene and detecting the presence of the wand (basic transformation from video capture to Boolean value) or grabbing a picture, detecting the presence of the wand and providing a digital recipient with a 4x4 matrix indicating the position and orientation of the wand. In this last situation, a converter is considered as part of the adaptor.

Presently, accelerometers, compasses, magnetometers, etc are supplied as special purpose devices and thus may be usefully modelled separately. However, we can anticipate that in the future these sensors will be fully integrated into mobile and pervasive devices such that they can be abstracted away as part of an interaction path. This is for example the case with Pan-Tilt-Zoom cameras that integrate automatic tracking of moving entities [0].

3.2.2 Mediating Entities

In figure 2, the physical wand manipulated by the user has no integrated mechanism supporting the encoding of its physical position into digital data. Therefore, it is not an adaptor. In this, as in most, cases, the physical entity constitutes an intermediate stage in the communication.

By definition, entities required to support intermediate stages in the communication and which are not themselves adaptors are mediating entities. We distinguish two different types of mediating entities: interaction carriers and contextual entities.

3.2.2.1 Interaction Carriers. Interaction carriers are mediating entities that are necessary for information communication. Carriers can

- provide a means of changing the user experience without changing the interaction functionality (e.g., the use of a stylus rather than a finger when interacting with a touch sensitive display),
- support “action at a distance” (e.g., a light pen) or
- act as a storage or feedback mechanism (e.g., handwriting on a piece of paper left as a trace by a digital pen).

The concept of interaction carriers can be further refined by identifying “active” and “passive” carriers:

- Active carriers are transmitters of non-persistent information along the interaction path. For example, a stylus transmits to a precise position on a touch screen a force generated by a user; the wand in figure 2 represents a position, etc.
- Passive carriers can carry, and store, part of the information communicated along the interaction path. For example, a tangible object left in a particular position on a table can serve as a physical storage device and the information might be picked up later via a camera.

3.2.2.2 Contextual Entities. In addition to interaction carriers there may be other physical entities involved in an interaction, such as the table on which the sponge may be placed. We call these contextual entities.

3.3 Characterising Interaction Paths

So far our description of the spongy switch in figure 3 doesn't tell us very much of interest. For example, there is nothing yet to distinguish a user's actual physical manipulation of this device from the manipulation of, say, a light switch. Furthermore, we cannot tell what sort of information the switch can communicate nor how the state of the sponge is sensed. We propose to anchor the required expressiveness in the description of the interaction paths.

Information paths are characterised by five basic properties. Two of the properties apply to the interaction channel itself, two others apply to the participating entities, one to the originating element and the other to the receiving element, and a final property applies to the entire path.

3.3.1 Channel Properties

3.3.1.1 Medium. When the interaction channel is physical (e.g. between a physical participating entity and an Ain, Aout or another physical participating entity), the medium is "the physical means by which the information is transmitted", that is, a set of physical characteristics or properties, used to communicate information.

When the interaction channel is digital (e.g., from an Ain to a digital entity or between two digital entities) we may want to capture information about the nature of the connection, e.g., bandwidth, uptime, whether it is wireless (e.g., rf, infrared, 802.11), etc.

In figure 2, the medium of the interaction channel between the physical wand and the camera in charge of its localisation is visual; the tracking of the camera is a visual based detection.

3.3.1.2 Representation. This is a description of the coding scheme, or language, used to encode the information in the medium.

Note that there may be multiple levels of representation of the information. For example, a command to switch a light off or on might be represented as a sentence in a natural language, which is itself represented in auditory form for transmission to the input adaptor (modifying the medium; i.e., causing vibration in the intervening air). It is this auditory form which is used directly to modify the medium; the other representations (i.e., the natural language sentence and the operational command) may be formulated mentally by the user and subsequently may be extracted via an interpretation process by the input adaptor or other system components downstream from the adaptor.

In figure 2, the representation of the interaction channel between the physical wand and the camera in charge of its localisation is the position of the extremity of the wand in the physical space.

3.3.2 Properties of Participating Entities

3.3.2.1 Method of modification. This refers to the method of manipulating or otherwise affecting the medium. In the case of user-generated input, the user must act upon the medium to produce the state of the medium, or changes in its state, that are information encodings (i.e., that structure the medium according to the coding scheme). Similarly, an output adaptor must modify the medium of its channel. A speaker, for example, would use vibration of the speaker cone to set up vibrations in the air forming the medium of its channel to the user.

Mediating entities may also play a role here. In figure 2, when considering the interaction channel from the wand to the camera, the method of modification used by the wand onto the channel is the movement of the wand: movements of the wand affect the (visual) medium of the channel by changing the wand position (the representation).

In some cases the source of the information may perform no active modification of the medium. Consequently, the information is extracted from the channel via the active sensing process of an appropriately “stateful” sensor. For example, a camera (plus image processing) may be able to determine that some object in its field of view has not moved. This is perhaps the limiting or degenerate case of “affecting” the medium; i.e., the medium is “affected” by not being changed.

In the case of digital to digital channels, the method of modification is typically of no interest for purposes of interaction design. However, other related properties of the channel may be significant, e.g., push vs. pull; continuous vs. intermittent, average and peak load.

3.3.2.2 Sensing Mechanism. This depicts the device(s) and process(es) by which the state or changes in state of the medium are captured by the information recipient. In addition to a camera, as in the magic wand example, other typical sensing mechanisms include, among many others: pressure sensors, touch screens, microphones, cameras including integrated image processing, such as motion detection, accelerometers, graphical and tactile displays, speakers and earphones

If the communication has a user as the ultimate recipient, then sensing mechanisms include all the normal human perceptual channels.

3.3.3 Properties of the Overall Interaction Path

The properties presented so far express how an interaction path might communicate information or initiate action. The *intended user model* refers to what the user should know about the interaction in order to carry it out successfully. It may refer to one atomic interaction path (e.g., a channel plus its source and destination) or it may refer to more complex paths. We distinguish two parts of the intended user model: its core (or content) and its context.

3.3.3.1 Intended User Model (Core)

This first dimension of the Intended User Model is the specification of the information that is *intended* to be communicated. This applies both to exchanges from and to the user. It is intended by the designer, ideally internalised and/or understood by the user and often indicated/represented in the system [9].

In the magic wand example, the core of the IUM of the interaction path between the user and the camera (through the physical wand) is the activation of the command that will affect the 3D volume. The user has to know that manipulating the wand is required to activate the command.

3.3.3.2 Intended User Model (Context)

The contextual IUM refers to all other pieces of user knowledge necessary to carry out the interaction successfully. This might include being aware of associations ("clicking on *this* button will cause *that* object to disappear") or understanding the mechanism by which the interaction is realised ("my face is being captured by that camera").

In the magic wand example, the contextual IUM must include the boundaries of the physical space in which the wand is localised and outside of which the wand is no longer visible and can no longer be used to activate a command.

3.4 Path Properties of the Spongy Switch

3.4.1 Applying the model to the Spongy Switch.

Each of the relationships, between user and sponge and between sponge and sensor, can be characterised using our path properties. The model illustrated in figure 3 has an interaction path consisting of one originating entity (the user), one channel and one recipient entity (the instrumented sponge). We have, however, also shown, an additional channel, linking the sponge to some receiving digital recipient (e.g., a concept or component in the application). To keep this example simple, our description of the path properties will apply to the user-sponge path only.

We shall use a simple table to present the path properties:

Table 1. Path properties of the Spongy Switch.

Medium	The state of compression of the sponge
Representation	A set of discriminable compression values. For our example, we will choose two, compressed and uncompressed.
Modification Method	The sponge's compression state is modified by the user squeezing or releasing the sponge.
Sensing	Sensing is via a pressure sensor embedded in the sponge.
IUM	At this stage, without yet having contextualised the interaction technique, the intended user model can only be described as communicating one of two discrete states, otherwise uninterpreted. In the context of a digital whiteboard, squeezing the spongy switch might correspond to selecting the eraser.
Other properties	

3.4.2 Refining the Spongy Switch Model.

3.4.2.1 Decomposition. The spongy switch description so far does not separate out the sponge from its pressure sensor; they are treated as a single integrated entity. It's often useful to treat a complex interaction device or mediating entity in this way, abstracting over its internal composition. However, it can also be useful at times to refine the description, revealing details of its internal structure as illustrated in Figure 4. In this case we have two channels, one from the user to the sponge and one from the sponge to the adaptor (pressure sensor). We leave channel C unspecified here for purposes of simplicity.

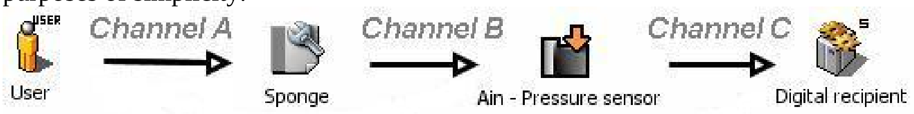

Fig. 4. A Refined Diagram showing Spongy Switch Internal Structure.

Channel A has sponge compression as its medium and channel B has sponge internal pressure as its medium. Notice that now there is a transformation from the channel A representation to that of channel B (i.e., from states of compression to pressure states). Indeed, it may well be that the user is capable of placing the sponge in a number of different degrees of compression (i.e., the channel A representation has more than 2 states) but that the sensor can only recognize two different levels of pressure. Additionally, the subsequent channel, from sensor to digital recipient, may itself have a different resolution, with the sensor reducing the number of discriminable states communicated on that channel.

Table 2. Path properties of the refined description of the Spongy Switch.

	Channel A	**Channel B**
Medium	The state of compression of the sponge	Internal pressure of the sponge.
Representation	A set of discriminable compression values. For our example, we will choose two, compressed and uncompressed.	A set of discriminable pressure values.
Modification Method	The sponge's compression state is modified by the user squeezing or releasing the sponge.	None
Sensing	None	Sensing is via a pressure sensor.
IUM	There is one intended user model, which is the same as the unrefined path (see section x).	
Other properties		

3.4.2.2 Feedback. So far, we have only shown an input path. Clearly, feedback paths are necessary. Figure 5 illustrates a possible design, identifying three paths, one at the physical level, one that indicates the interpretation of the sponge manipulation and a final one that presents the results of application significant operations. We revisit this topic in section 4.2.1. on a more concrete example to analyse these feedback paths.

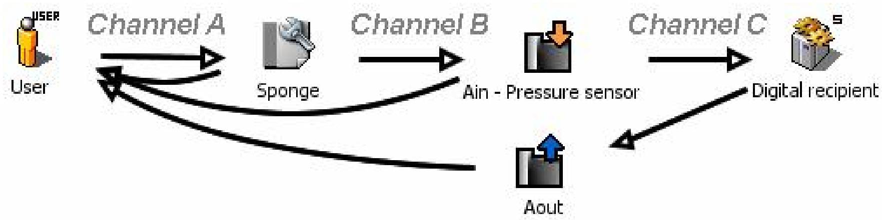

Fig. 5. A second Refined Diagram showing Spongy Switch Feedback.

3.4.3 Exploring a design space

Our spongy switch description, although still very simple, already enables us to begin exploring an interaction design space. We can find alternative entities that will function similarly within an interaction path unchanged with respect to its information communication properties. For example, if the medium of channel B becomes visual, while its representation and the associated method of manipulation remain unchanged, the sensing mechanism might be changed to image capture. In order to leave channel C unchanged, the Ain also has to be able to derive a level of compression value. To satisfy these new design options, the pressure sensor might be replaced by a camera, positioned to capture the shape of the sponge and that encapsulates a "sponge shape to compression level mapping system". Since channels A and C remain unchanged, this replacement can safely be made without either changing the user experience or the interaction functionality, as modelled[2]. Such a replacement, leaving the user experience (including the distinctive aspects of squeezing a sponge) the same and the system functionality unchanged could be useful in order to reduce implementation costs (no need to construct a special sponge) or to increase mobility (no need for wire dangling from the sponge).

3.5 Refining the properties of the interaction path

So far we have not addressed the question of the level of abstraction of the descriptions in a characterisation of an interaction path. The level in the spongy switch example is perhaps sufficient to communicate a reasonably concrete design solution. However, there are likely to be features of the interaction which need further refinement, either to make the specification sufficiently precise to be implemented or to identify key features affecting its usability. For example, one will need to know how many discriminable compression states are necessary for the application and are achievable with a particular sponge/sensor combination. Additionally, the weight of a participating entity may be significant; the lightness of a sponge might make it a good candidate for elderly users who have weak muscles.

We believe that this refinement will normally occur as part of an iterative process. In the early stages of a design, we may simply identify the need to output an image of

[2] Of course, there may be other properties of the interaction which would impinge on the two end channels. For example, the video might be slower, use more bandwidth and might be a disturbing presence in the user's environment.

a digital object. However, this is not sufficient for an implementation, for which additional details of the form of rendering will be needed.

4 Studying Interaction Groups

So far we have introduced the principles and characteristics of the interaction path concept and illustrated them via a simple example. In this section we use a different example to introduce the notion of *interaction groups*, built on top of interaction paths, and we explain how that notion can be used to capture and analyse design alternatives.

4.1 The PDA balloon case study

4.1.1 Overview

Our second example is based on an interaction technique developed to demonstrate the use of sensors as captors [11]. This interaction technique involves a user; an adaptor for input that is able to capture, analyse and identify squeezing actions of a user; a PDA that the user holds in the hand; and the PDA's display, used to present representations of digital entities.

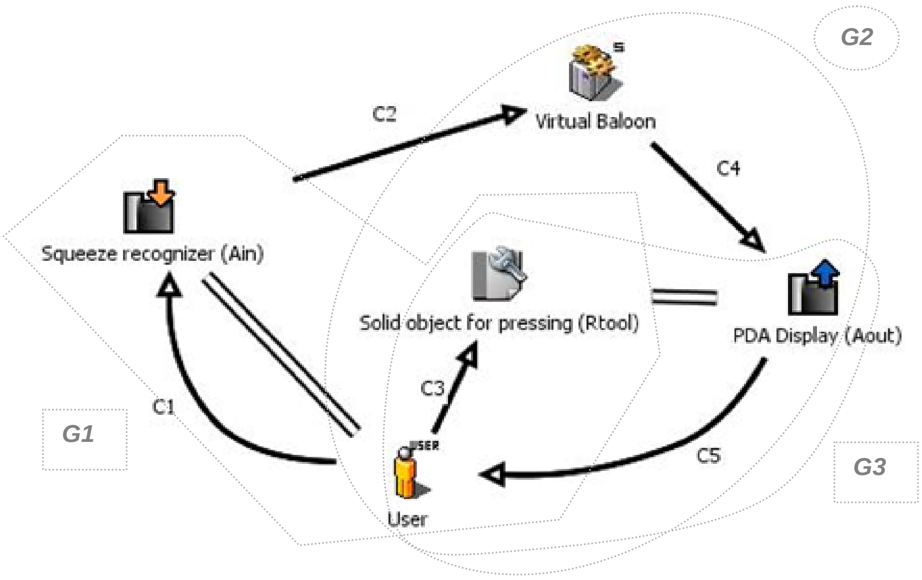

Fig. 6. Basic ASUR diagram for the PDA balloon and interaction groups (cf. section 4.2).

Based on this interaction technique, an application has been designed to enable a user to interact with a virtual balloon. Figure 6 presents the basic ASUR model of this application. The virtual balloon is presented via an image on a PDA display. The user can change the balloon size by “inflating” it; this is carried out by squeezing on the PDA case (denoted as “solid object for pressing”). Each squeeze will increase the size of the balloon by one level.

The most interesting feature of this application is the indirect relationship between the user’s squeeze and the system’s sensing of that event. There is no sensor on the PDA; rather an accelerometer is placed on the user’s forearm[3]. This accelerometer detects muscle tremor. The squeeze action increases this tremor and the Ain uses an algorithm for recognising the distinctive tremor pattern associated with a squeeze that is sufficiently strong.

The next section details the characteristics of the interaction channels that will be referenced when illustrating the subsequent interaction group analysis.

4.1.2 Channel Descriptions

Channel C1 represents the link between the user and the “Squeeze recognizer” (Ain). In the particular use of this interaction technique, inflating a digital balloon, there is no intended user model since the sensor is intended to be completely invisible for the user. The digital channel C2 is required to transmit the captured information to the digital resource that manages the digital balloon. Channel represents an interaction the purpose of which is simply to motivate the user to generate muscle tremor. However, in order to produce an effective design the method of generating this tremor must be appropriately linked semantically to the notion of balloon inflation; we shall return to this issue in section 4.2 below.

C4 is another digital channel required to present properties of the digital balloon through an adaptor for output. In our case, the property of interest is the size of the digital balloon. Finally, channel C5 transfers data that will be perceived by the user; the information carried by the channel must represent the size of the balloon. Table 3 summarises all the characteristics of these interaction channels.

4.2 Interaction Groups

We use the term ‘interaction group’ to refer to a set of entities and channels that together have properties that are relevant to a particular design issue. As will become evident, there are typically many such interaction groups that can be identified for a particular interaction design. Some of these groups will be universal (applicable to any design) while others will depend on the task and context or on the requirements of an analysis performed by a specialist (ergonomist, ethnographer, device designer, software engineer). For example, entities or channels that represent or transfer information about a single common concept or that share the same type of constraints

[3] In the original technique developed by Strachan et al. [11], the tremor sensor is actually mounted onto the PDA-case. In order to better illustrate the grouping mechanism presented here, we use a slightly different design technique, in which the tremor sensor is indeed fixed on the user’s arm.

may form an interaction group. The set of all interaction groups for a given design forms a potentially complex graph of associations, with different views for different purposes. Via the PDA balloon example, we present several different groupings that exemplify the sorts of groups that are likely to be of interest for many interaction designs.

Table 3. Characteristics of the interaction channels of the PDA balloon.

Channels	**C1**	**C2**	**C3**	**C4**	**C5**
Medium	Muscle tension	Digital	Pressure on the Rtool	Digital	Light
Represen-tation	Recognizable tremor pattern	ONE discrete command	2 values: squeezing / grasping	Set of balloon sizes	Image
Modification Method	Tremor	Not relevant	Hand squeezing	Not relevant	Light modulation
Sensing	Accelero-meter	Specific API	Null	Specific API	Visual
IUM	None: the user is not supposed to be aware of it	N/A	Single hard squeeze inflates by one level.	Not applicable	Size of a balloon
Other interesting properties	Granularity of the squeezing detection	Wired or Wireless connection		Property of interest: balloon size	Attributes of the chosen representation

4.2.1 Grouping for feedback

Grouping for feedback aims at identifying entities and channels involved in an interaction flow linking the response of the system to actions of the user. To promote this group and ensure that the feedback will be effectively perceived, it is important to consider the definition of the characteristics of these channels as a whole: for example if using audio as modification method of C5, it is probably not adapted to adopt the same modification method for C1 (loop) since both channels will be used almost simultaneously each time. In addition, it is also important to clearly differentiate the characteristics of these channels from those of other channels involved in the interaction but not in the feedback group.

In the case of the PDA balloon, one feedback group includes all the channels: C1 and C3 in parallel, C2, C4 and C5. As a consequence, C2 and C4 must persist throughout the interaction and must not be interrupted due to, say, a poor WiFi connection. It is also important that the different values of the representation carried by C4 are correspondingly represented via C5.

Channels C3 and C5 alone also form a separate articulatory feedback group because acting on the Rtool through C3 automatically triggers effects perceivable via C5. However, feedback is only one of the features of this grouping; we will discuss it again in section 4.2.4.

4.2.2 Grouping based on coherence among properties

Some groups join together elements with related properties in order to generate a coherent effect, such as visual continuity. For example, a grouping might associate a set of channels and assert that they must all use the same medium (e.g. Visual) or indeed must use different media (e.g., visual and sound) in order to provide perceptual continuity.

A first example of this sort of grouping for coherence is the group called G1 on the PDA balloon diagram (see figure 6). This group, identified at the design level, consists of the User, the PDA case (Rtool), the accelerometer (Ain), and channels C1 and C3. It is based on coherence between the modification method of channels C1 and C3. Therefore, a change to one of these modification methods must ensure that this coherence property is maintained: that is, the modification method of C1 (i.e., tremor) must be an indicator, effect or co-occurrence, of the modification method of C3 (squeezing with hand). In other words, by grouping the two channels, we are saying that they work together as a single mechanism and it clearly expresses that these elements used simultaneously enable the inflation activity.

An alternate implementation solution might consist of changing the adaptor to a camera, thus changing C1 so that it captures visual properties of the user's modification method in C3. Clearly, for this to be acceptable, the modification method properties of C1 (e.g., visual deformation of the muscles in the forearm or characteristic distortion of the Rtool) must correspond to the squeeze manipulation of C3.

A second example is the group G2 that consists of the digital balloon, the PDA display and case, the user and channels C3, C4 and C5. The group identified is based on coherence among the Intended User Model of the involved channels. G2 captures the notion that the user interface elements (i.e., PDA display and PDA case) together form a representation of the presented concept (the virtual balloon). Ensuring a coherent IUM among channels of this group may also be reinforced by applying constraints or associations to other properties of the channels, such as:

- the PDA display must show a visual representation of the virtual balloon, i.e., something that looks like a balloon (constraint) and its displayed size must correspond to the virtual balloon size property (association).
- the method of manipulation of C3 should correspond to the method of manipulation of a real balloon, to reinforce the association of the action with the intended type of virtual object expressed via the "other properties" part of C3 (association)

More generally groups of interaction paths based on coherence among properties could refer to any single common property or set of properties shared over several interaction paths or channels (e.g., all of the paths that use video sensing mechanism, all those that use grasping, all those that participate in the same IUM, etc.). The potential force of analysis based on these groups is that it allows the specification and refinement of the different forms and levels of articulatory, perceptual and cognitive continuity that may be considered when evaluating an interactive system.

4.2.4 Action and effect association

This expresses a semantic association that links user interface elements to certain application concepts. The goal is generally to help the user to cognitively unify elements of the groups. Such grouping can lead to requirements on several properties of the elements in the group.

G3 is such a group in the PDA balloon example. The group consists of User, PDA case, PDA display plus C3 and C5. This group is not only a feedback group. Indeed, the purpose is to unify the actions on the PDA case with the resulting effects presented in the PDA display to help the user associate the squeeze on the case as the *cause* of the inflation. There are three aspects of this grouping that serve to reinforce the cognitive association of the action and the effect:

- the physical closeness of PDA case and PDA display (represented by the physical proximity relationship on the ASUR diagram,
- the feedback loop of C3 followed by C5 and
- the fact that the PDA case and the PDA display are both in the user's visual field at the same time (this property is not directly expressible in the diagram; however this could be added to "other properties" of C3 and C5).

4.2.5 Other groupings

While we have examined several interaction groups arising from an initial analysis of our simple example, the value of the interaction grouping concept is potentially much greater. Part of our future work is to further explore the sorts of purposes to which interaction groupings can be put. Among potential groups of interest are sets of inputs that must be combined to perform some task, e.g., a speech input with a gesture input ("put that there"). This would correspond to a form of grouping for multimodal coordination

The correspondences expressed in G1 motivate a sub-grouping of the Rtool and the Ain entities to create an "abstract instrument" with a single perceived input channel, C1, and a single output channel, C2. This concept of "abstract instrument" need further investigations but constitutes another form of grouping and establishes a clear parallel with the notion of instrument in the instrumental interaction [2].

A grouping for distribution / communications might be used to assert that a set of services/concepts must reside on the same machine or indeed be distributed or use a common form of communication.

In the case of collaborative systems, groupings of paths may show information flows among or between users. Additionally, agronomists may want to group physical devices and their locations.

5 Relationship to other models

Card et al's input modelling language is perhaps the closest to our model in its attention to the physical and concrete aspects of an interaction [3]. However, we are interested in embedding this aspect into a larger descriptive framework that includes both the physical context, feedback loops and its role in information exchanges with an application.

Our information-exchange model could be deemed a variant of instrumental interaction [2]. We have pointed out, for example, how our interaction group mechanism can be used to specify abstract interaction instruments. However, our model highlights and refines the informational and physical aspects of the interaction. Consequently, our model can be considered complementary, and a possible addition to or refinement of, the instrumental interaction model.

Coutrix & Nigay [4] offer a recent approach that, like our model, combines both the physical and digital dimensions of the interaction. Their interest is primarily in the transformations of information through mediating software components that together express interaction modalities. Our approach, however, includes a richer description of the interaction from the point of view of a user's manipulative and perceptual actions and their relationship to a user's intentions. Again we believe that these are complementary descriptions that could benefit from being used together.

Smith [10] applies a flownet model to the description and analysis of design-significant features of a system involving haptic interaction. This model, like ours, is designed to enable low-cost exploration of concrete interaction design issues such as the continuity of physical actions and the coherence and adequacy of feedback. Smith's approach, unlike ours, can deal with the dynamics of interaction and, indeed, it would be interesting to add flownet semantics to our model to augment this aspect. However, Smith's model does not include an intended user model nor our feature of (potentially extensible) interaction groups. Additionally, Smith's model stops at the point of a user's generation of input and/or consumption of output and thus does not capture the role of mediating objects. As we have suggested above, differences between sufficiently similar models, such as those just noted, offer opportunities for cross-fertilisation.

6 Conclusions

The model we have presented in this paper takes seriously the fact that interaction is both a concrete phenomenon, embedded in a physical context, and also a complex combination of information exchanges that support activity in a mixed physical/digital world. It picks out aspects that are potentially design significant and organises them in a way that is intended to facilitate design reasoning (e.g., making comparisons between choices of device, identifying new solutions, finding problems).

We have developed our model in order to refine and enrich an existing model, ASUR. We have found this approach to be fruitful and the new elements introduced here seem to fit comfortably with the original model. However, it remains to be determined if this association relies on some fundamental connection between the original ASUR notion (i.e., component-based composition, interaction-centred viewpoint) and these new concepts related to physically realised interaction channels. In other words, it may be possible to take ideas from our approach and use them to augment other models, such as those referred to in section 5.

The ASUR interaction model is intended to be very high level. It is not intended to capture the way in which the information communication is structured or realised via actual interactors or dialogue sequences in particular languages. It is designed to focus on key aspects of the interaction from the point of view of features that need to be identified early in the design process. Further work is needed to link descriptions of interaction using our model into a development process leading to effective implementations.

Acknowledgments. The authors wish to acknowledge the award by the University Paul Sabatier of Toulouse of a visiting professorship to Philip Gray during the period March- September 2006. This award helped make possible the collaboration resulting in this paper.

References

1. Auto Tracking VB-C50i Network Pan/Tilt/Zoom Camera by Canon, http://www.nuspectra.com/detail.aspx?ID=1078
2. Beaudouin-Lafon, M.: Instrumental Interaction: an Interaction Model for Designing Post-WIMP User Interfaces. In: Proc. CHI 2000. ACM Press (2000) 446-453
3. Card, S. K., Mackinlay, J. D., and Robertson, G. G.: A morphological analysis of the design space of input devices : *ACM Trans. Inf. Syst.* , Vol 9 (2) (Apr. 1991) 99-122
4. Coutrix C., Nigay L.: Mixed Reality: A Model of Mixed Interaction. Proceedings of AVI 2006, (Venezia, Italy, 23-26 May 2006), ACM Press (2006) 43-50
5. Dubois, E., Gray, P.D., Nigay, L.: ASUR++: a Design Notation for Mobile Mixed Systems. In: Paterno, F. (ed) Interacting With Computers, Vol 15 (2003) 497-520.
6. Dubois E., Gauffre G., Bach C., Salembier P.: Participatory Design Meets Mixed Reality Design Models - Implementation based on a Formal Instrumentation of an Informal Design Approach. In: Calvary, G, Pribeanu, C. Santucci, G., and Vanderdonckt, J. (eds.): Computer-Aided Design of User Interfaces V. Information Systems Series. Springer-Verlag (2006) 71-84
7. Dupuy-Chessa, S., Dubois , E.: Requirements and Impacts of Model Driven Engineering on Mixed Systems Design. In: Gérard, S., Favre, J-M., Muller, P-A., and Blanc, X. (eds.): Proceedings of IDM'05, Paris, France (2005) 43-54
8. Ishii H., Ullmer B.,: Emerging Frameworks for Tangible User Interfaces. IBM Systems Journal 39 (2000) 915-931
9. Norman, D.: Cognitive Engineering. In: User Centered System Design: New Perspectives on Human-Computer Interaction. Lawrence Erlbaum Associates (1986) 31-61
10. Smith, S. P.: Exploring the Specification of Haptic Interaction. In: Doherty, G. and Blandford, A. (eds): Proc. DSV-IS 2006, Dublin (2006) 146-159
11. Strachan, S. & Murray-Smith, R.: Muscle Tremor as an Input Mechanism. In: Proc. UIST’04, Vol. 2 (2004)

Questions

Prasun Dewan:

Question: Are you expecting to build tools for verifying that the modelled properties are actually implemented?

Answer: This is not currently planned: this is meant to be a lightweight means of reasoning. Verification isn't on our current to-do list.

Question: What is the practical use of it?

Answer: As a means of communicating the characteristics of the application.

Laurence Nigay:

Question: In your example, the key problem is that the user cannot observe the input mechanism. Did you think about honesty or observability?

Answer: I don't think the lack of awareness of the channel is a problem.

Panos Markopoulos:

Question: It would be interesting to see if the reasoning power that this approach delivers actually helps designers?

Answer: That's proposed in the validation stream.

.