ERCIM'94 workshop on Multimedia Multimodal User Interfaces, Nancy, Octobre 1994

Taxonomic Issues for Multimodal and Multimedia Interactive Systems

Joëlle Coutaz, Laurence Nigay, Daniel Salber

Laboratoire de Génie Informatique, IMAG
B.P. 53, 38041 Grenoble Cedex 9, France
Phone: +33 76 51 48 54, +33 76 51 44 40, Fax: +33 76 44 66 75
E-mail: Joelle.Coutaz@imag.fr, Laurence.Nigay@imag.fr, Daniel.Salber@imag.fr

Abstract. One trend in Human Computer Interaction is to extend the sensory-motor capabilities of computer systems to better match the natural communication means of humans. Parallel to the exploratory development of such systems, significant effort is being deployed in defining frameworks and taxonomies for reasoning about the design space of such systems. This article is a preliminary effort to review current frameworks and taxonomies, focussing on the ways they complement each other.

Keywords: Interaction framework, taxonomy, multimedia, multimodal.

1. Introduction

One of the current trends in Human Computer Interaction is to extend the sensory-motor capabilities of computer systems to augment the natural communication means of humans. Successful attempts in this direction are the concepts of virtual reality and artificial reality [Krueger 91]. A promising new paradigm is emerging with the notion of augmented reality [Mackay 93]. All of these "digitized realities" are illustrations of how multimedia and multimodal technology may be exploited in a useful and attractive way.

Up to now, the development of MultiModal and MultiMedia (M^4) interactive systems has been primarily exploratory. To complement this experimental approach, a significant effort is being deployed in defining frameworks and taxonomies for reasoning about the design space of such systems. This article is a preliminary effort to review current frameworks and taxonomies related to the M^4 technology. Frameworks and taxonomies are designed to satisfy a particular purpose. This article discusses frameworks and taxonomies that aim at complementary goals. We do not claim to be exhaustive nor to impose any terminology. Our goal is to provide a panoramic view of current efforts for M^4 frameworks and identify links between them.

The MSM (Multi-Sensory-Motor) framework will be used as the starting point of the discussion. We will then enrich this general structuring space with more focused taxonomies such as the design space of input devices and the taxonomy of pure output modalities.

2. The MSM framework

The underlying motivation for MSM is to provide system designers with a structured problem space. Although MSM can be used by user interface designers to conduct usability experiments, this framework is primarily system centered.

As shown in Figure 1, the MSM framework is comprised of 6 dimensions:

- Two dimensions deal with the notion of communication channel: the number and direction of the channels that a particular MSM system supports.

- The other four dimensions are used to characterize the degree of built-in cognitive sophistication of the system: levels of abstraction, context, fusion/fission, and granularity of concurrency.

These issues are presented succinctly in the following paragraphs. A more complete description can be found in [Coutaz 93].

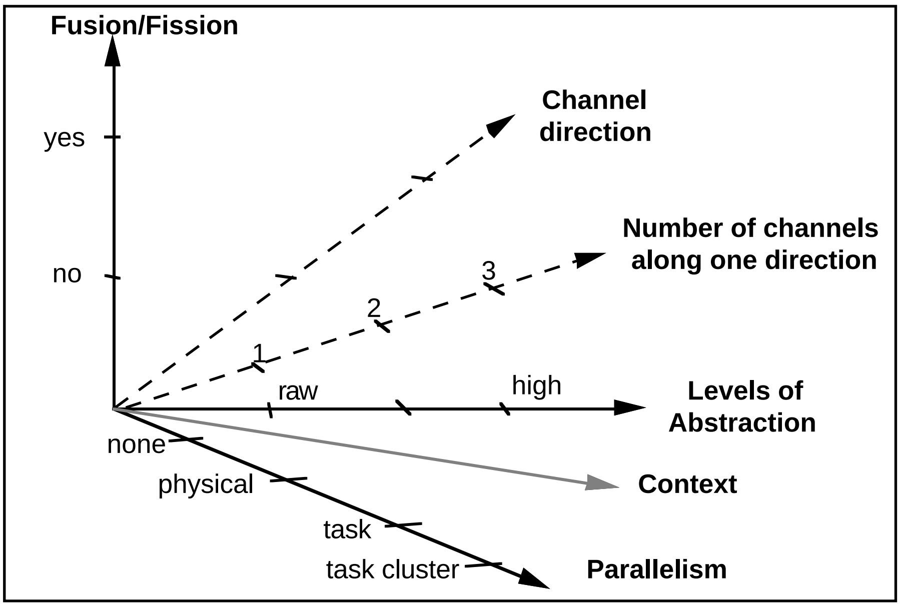

Figure 1: The MSM framework: A 6-D space to characterize multi-sensory-motor interactive systems.

Communication channel
A communication channel can be viewed as the temporal, virtual, or physical link that makes the exchange of information possible between communicating entities (e.g., a human being and a computer system). Instead of considering the linkage dimension of communication, MSM stresses the importance of the sources and recipients involved in a communication act. Thus, a communication channel covers a set of sensory (or effector) means through which particular types of information can be received (or transmitted) and processed.

A sensor is a physical device that allows a communicating entity to acquire information from the environment (e.g., another communicating entity). An effector plays the symmetrical role for transmitting information to the environment. Interestingly, sensors and effectors are not insulated randomly. Multiple sensors (effectors) may be grouped together to form a cluster associated to a processing facility. This grouping of physical devices under the hat of a processing unit corresponds to a communication channel. This view of a communication channel matches nicely the ICS psychological model [Barnard 87].

For example, the retinas capture space-time patterns of photons which are processed by the visual subsystem into a mental form usable by the representational or effector subsystems. The retinas (which are two input physical devices) and the visual subsystem (which is the corresponding processing facility) define a *human communication channel*. As an example from the computer side, the X window server handles both mouse and keyboard input devices. It transforms interrupt signals into a higher level representation, an "X event", that may be of interest to client processes. The keyboard, the mouse, and the X server define a *digital communication channel.*

Information types conveyed by human and digital communication channels define an abstraction from the physical representations used by I/O devices. This abstraction is the boundary with higher internal representations. It conveys phenomena, not meaning. Meaning is covered by the internal processes, responsible for executing the interpretation and rendering functions. In his framework, Frohlich adopts a similar perspective although he would stick to a more signal level representation [Frohlich 91]: "A channel is an interface across which there is a transformation of energy ... Over the human interface channel, there is a transformation of electrochemical energy within nerve cells into noise and movement energy, and of audio, visual and haptic energy into electromechanical energy." Over the computer interface, similar transformations apply to electrical energy. In section 3, we will present another definition of the concept of channel.

Interpretation and Rendering
Information acquired by input digital channels is transformed through multiple process activities. This sequence of input transformations forms the interpretation function. In the other direction, internal information (e.g., system state) is transformed to be made perceivable to the user. This sequence of output transformations defines the rendering function. The interpretation and the rendering functions can be both characterized with four intertwined ingredients: 1) level of abstraction, 2) context, 3) fusion/fission, and 4) parallelism.

1) Level of abstraction. The notion of level of abstraction expresses the degree of transformation that the interpretation and rendering functions perform on information. It also covers the variety of representations that the system supports, ranging from raw data to symbolic forms. The span of representations should be considered on a per-digital channel basis. Thus, for a given digital input channel, the interpretation function can be characterized by its power of "abstracting" raw data into higher representational expressions. The rendering function is characterized by the level of abstraction it starts from to produce perceivable raw information through output digital channels.

Computer vision, speech recognition as well as speech synthesis systems operate along these principles. For example, speech input may be recorded as a signal, or described as a sequence of phonemes, or interpreted as a meaningful parsed sentence. Each representation corresponds to a particular level of abstraction resulting from an interpretation function. For output, the process is similar: data may be produced from symbolic abstract representations or from a lower level of abstraction without any computational knowledge about meaning. For example, a vocal message may be synthesized from an abstract representation of meaning, from a pre-stored text (i.e., text-to-speech) or may simply be replayed from a previous recording.

2) Context. The capacity of a system to abstract along a channel may vary dynamically with respect to "contextual variables". Contextual variables are like cognitive filters. They form a set of internal state parameters used by the representational processes to control the interpretation/rendering function. For example, in vi, when in command mode, typed text is transformed into a high level abstraction whereas the same text entered in input mode is recorded as is without any transformation. Contextual variables constrain the configuration of digital processes used at some point in time to process

information. We observe an analogy with the cognitive resources configuration claimed in ICS [Barnard 87].

3) Fusion and Fission. Fusion refers to the combination of several chunks of information to form new chunks. Fission refers to the decomposition phenomenon. Considering fusion for the interpretation function, information chunks may (or may not) originate from distinct digital input channels or from distinct contexts. For example, the sequence of events “mouse-down, mouse-up” that occurs in the palette of a graphics editor are two information chunks that originate from the same input channel and from the same context (i.e., the palette agent). They are combined within the context of the palette agent to form a higher information chunk (i.e., the selection of a geometric class). The drawing area constitutes another context maintained by a dedicated agent. Events that occur in the drawing area are interpreted as the effective parameters of the geometric function. They are combined to the selected geometric class by a “cement agent” to complete the function call in the task-domain.

Thus, in the graphics editor example, fusion occurs between information chunks originating from the same digital channel but, as the interpretation proceeds at higher levels of abstraction, it also involves different contexts maintained by distinct agents. The “put that there” paradigm, however, exemplified by Cubricon [Neal 88] offers an example of fusion between chunks originating from distinct input digital channels. In this example, fusion is required to solve the coreferences expressed through distinct channels.

As for fission, it may be the case that information coming from a single input channel or from a single context need to be decomposed in order to be understood at a higher level of abstraction. For example, the utterance “show me the red circle in a new window” is received through a single digital channel but references two domains of discourse: that of the graphics task (i.e., “the red circle”) and that of the user interface (i.e., “a new window”). In order to satisfy the request, the system has to decompose the sentence into two high level functions: “create a window” and “draw a red circle” in the newly created window.

4) Parallelism. Representation and usage of time is a complex issue. In this discussion, we are concerned with the role of time within the interpretation and rendering functions. How does time relate to levels of abstraction and contexts? How does it interfere with fusion and fission? Parallelism in the user interface may appear at multiple grains: at the physical level, at the task and task cluster levels.

Parallelism at the physical level allows the user to trigger multiple input devices simultaneously. If these devices are organized along distinct channels, then the user solicits multiple input digital channels in parallel. Similarly, physical parallelism for output may take the form of simultaneous outputs through distinct digital channels or may occur through a single channel. The fission example “watch this wall” associated with “the blinking red line”, requires parallelism at the physical level using multiple digital output channels.

From the system’s perspective, a task (i.e, an elementary task) cannot be decomposed further but in terms of physical actions. For input, an elementary task is usually called a command, that is, the smallest fusion/fission of physical user’s actions that changes the system state. True parallelism at the command level allows the user to issue multiple commands simultaneously. It necessarily relies on the availability of parallelism at the physical level. Pseudo-parallelism at the command level, allows the user to build several commands in an interleaved way as in multithread dialogues. Then, parallelism at the physical level is not required.

The MSM framework identifies features useful for making a clear distinction between “multimedia-lity” and multimodality from a system’s perspective.

Multimedia and multimodal interactive systems
Both multimedia and multimodal systems are characterized by communicating information either through multiple input digital channels or through multiple output digital channels, or both. The multiplicity of communication channels along one direction (whether it be input or output) provides the basis for multimedia-lity and multimodality.

The distinction between multimedia-lity and multimodality lies in the degree of built-in cognitive sophistication of the system along the axis "level of abstraction". Multimodality is characterized by the capacity of the system to interpret raw inputs up to high levels of abstraction (e.g., that of the functional core portion of the system) or to render information starting from high level representations. Although multimedia-lity includes interpretation and rendering, it is not capable of handling the highest task-domain level representations.

An MSM system may be both multimedia and multimodal. For example, an hypermedia system would illustrate task-domain concepts using images and sound replayed from a CD-ROM, and it would be controlled by the user in a multimodal way using both speech and mouse to navigate through the hyperspace. Note that current multimedia systems are all able to handle the highest task-domain level representations but they do so for commands only and through a unique channel. Thus any multimedia system is at least monomodal in order to recognize input commands.

In summary, the MSM framework identifies features that are of interest to the software designers of M^4 interactive systems. In particular, it explicitly points out the engineering difficulties for devising elegant and reliable solutions to the combination of parallelism, fusion and fission within contexts at multiple levels of abstraction. For example, at a workshop on software architectures for multimodal interactive systems [IHMM 92], we have identified a number of technical issues related to fusion (e.g., eager/lazy fusion, distributed/centralized fusion, depth first/breadth first strategy, and fusion criteria such as temporal proximity and logical complementarity).

The system orientation of MSM needs however to be complemented with a more user-centered perspective. Also, the notions of channel, level of abstraction, and context must be refined to get a better understanding of the nature of M^4 interaction. In the following section, we present frameworks and issues that aim at responding to this need. A more detailed presentation of these issues can be found in [Nigay 94].

3. Frameworks as MSM Refinements

The MSM framework defines a channel as a set of input and output devices under the control of a processing unit. Although this notion of channel is acceptable for an overall analysis of an interactive system, it is not satisfactory when one needs to characterize the system at a finer grain of interaction. The "design space of input devices" such as that of Mackinlay et al. [Mackinlay 90], and the "taxonomy of pure modalities" developed by Bernsen [Bersen 93], are frameworks that valuably refine the "channel" and "direction" dimensions of MSM.

3.1. The design space of input devices

One of the seminal works in this area is the taxonomy presented by Foley et al. which relates input devices to graphics subtasks [Foley 84]. Graphics subtasks include selection, location, and orientation. Each of these tasks can be accomplished using distinct classes of input devices such as direct locators and button pushes. For example, a location task may be accomplished using direct or indirect locators, direction keys, or direct picks. In turn, a device class covers instances of physical input devices. For example, a direct locator includes a touch panel whereas the mouse, the tablet, and the joystick are indirect locators. The interesting property of Foley's taxonomy is to make explicit the relationship between tasks and devices. But, as observed by Mackinlay et al. in [Mackinlay 90], single devices appear many times in the taxonomy. As a result, it is difficult to understand the similarities among devices.

Buxton's taxonomy does not relate tasks to devices but provides a clearer picture about input devices properties [Buxton 83]. His classification is comprised of 3 axis:

- the property sensed by the device (position, pressure, motion),
- for each property, the number of dimensions sensed by the device (for example, a mouse returns two values for the position property),
- the sensing type which distinguishes between devices that work by touch (such as the touch screen) and devices that require mechanical intermediary (such as the mouse). This distinction was already identified by Foley with the notions of direct and indirect locators. It was not envisioned however as an orthogonal dimension applicable to other classes of devices.

Although Buxton's taxonomy is an interesting step forward, it does not support the microphone nor discrete input devices. Mackinlay et al. aim at more generality with the combination of individual devices into complex input controls [Mackinlay 90]. The theory focuses on the semantic information that must be communicated by input devices from a user to the application. This semantic-driven approach aids in the precision of the low levels of abstraction introduced by MSM.

In Mackinlay et al. model, an input device is defined by a 6-tuple <M, In, Out, S, R, W>. M is a manipulation operator which covers the physical properties sensed by the device. These properties can be characterized in terms of position (absolute or relative) and force (absolute or relative). Position and force are geometrically characterized as either linear (as for the mouse position) or rotary (as for a knob). In is the input set over which a manipulation operator senses a value (e.g., a continuous rotation of a knob between 0^o and 90^o). Out specifies the domain of values into which the input domain is mapped by the device (e.g., loudness in decibels from the angle of the knob). The mapping between In and Out is defined by R, the resolution function of the device. Note that R formally models the processing facility embedded in the MSM channels.

In addition to the definition of the notion of primitive input device, Mackinlay et al. introduce a notation to describe connections and hierarchical relationships between devices. This mechanism provides a powerful bridge that gradually leaves the physical world to enter the virtual and logical digital domain. By extension, one can join the theory of interactors as developped in the ESPRIT BR Amodeus project [Duke 92], which in turn leads gracefully to conceptual architectural models such as PAC-Amodeus [Nigay 93]. Although attractive, the semantic analysis and taxonomy presented in [Mackinlay 90] are concerned with input devices only. The taxonomy of pure modalities developed by Bernsen deals with output representations.

3.2. The taxonomy of pure output modalities

Bernsen defines a pure external modality as an uncombined representational information. It is external as opposed to the mental representations elaborated by humans. For example, spoken and written language, real and arbitrary sound, diagram and non-diagram pictures, graphs and real touch are pure external output modalities. An output pure modality is defined by a specific medium of expression and a profile [Bernsen 93]. If we refer to MSM, a medium of expression is equivalent to an output device. A profile is "constituted by its characteristics [i.e., that of the modality] as selected from the following list of binary opposites: analogue/non-analogue, arbitrary/non-arbitrary, static/dynamic, linguistic/non-linguistic".

The analogue/non analogue dimension expresses the similarity/difference between the modality used and what is represented. "The less recognizable similarity there is between what is represented and its representation, the more we may have to rely on additional knowledge of the representational conventions used in order to decode particular representations" [Bernsen 93]. For example, real world sound representations and diagrammatic pictures are analogue representations. On the other hand, spoken and written languages, graphs and arbitrary sound representations are not.

The arbitrary/non arbitrary dimension expresses the difference between "...representations which, in order to perform their representational function, rely on an already existing system of meaning and representations which do not" [Bernsen 93]. This definition sounds very close to that of "analogueness". In general, analogue representations are not arbitrary whereas non-analogue are arbitrary. There are exceptions to this common sense mapping. In particular, spoken, written, and touch languages are both non-analogue and non-arbitrary. Similarly, graphs which organize data according to conventional mapping principles are non-analogue and non-arbitrary.

Static and dynamic representational modalities have distinct implications on usability and robustness. In [Sellen 92], Sellen et al. provide a useful classification of feedback as transient/non-transient, avoidable/non-avoidable, and sustained/non-sustained. Typically, in normal conditions, sonic feedback is transient but non-avoidable. It is sustained by the system when repeated until the user takes the appropriate action.

In summary and in a more formal way, an output pure modality is characterized by the 5-tuple <M, An, Ar, T, L>. M denotes the output device chosen for rendering the representational information, An and Ar cover the properties of analogy and arbitrariness, while T and L respectively correspond to temporal and linguistic properties. The 5-tuple defines a framework for reasoning about output modalities in a general way. In some cases, the framework may not be precise enough to take sound design decisions. For example, the designer may need to reason about very specific attributes of the modality such as colors, textures, and shapes. The notion of channel, assimilated by Bernsen as modality attributes, seems to satisfy this need.

Although the frameworks and taxonomies presented above are both driven by the artefact, they address distinct levels of abstraction. For input, taxonomies deal primarily with low level physical properties. On the other hand, Bernsen's taxonomy for output is concerned with the mapping problem between the external representations produced by the system and the internal representations elaborated by the user. As a result, one observes a discrepancy between the scopes of applicability of the two taxonomies. This gap could be fulfilled in two ways. On one hand, the notion of channel, viewed as a modality attribute, could be refined and structured to identify physical properties of output devices. In the other direction, the composition mechanism offered by Mackinlay is a way to bridge the gap between the physical world and higher logical input representations. The UOM framework based on the availability, the capacity of choosing and combining interaction languages and physical devices presented in [Nigay 94] is a promising way to capture new interaction techniques at different levels of abstraction, both from the user and the system perspectives.

MSM, UOM, the design space of input devices, and the taxonomy of pure output modalities, are useful tools for thought. They are interesting for classifying a particular system in the problem space, they are appropriate for comparing the usability of distinct user interfaces, they help structuring the reasoning process, but they are not prescriptive enough to drive design decisions. Non specialists need heuristic support.

3.2. Heuristics for M4 design

The effort developed in the INTUITIVE ESPRIT project is an example of investigation about how to allocate media for well-structured tasks. The method starts with the definition of a TKS-based task model which specifies information requirements about the domain. Task specification is followed by a resource analysis which results in a resource model. For every domain data, the resource model defines the role of the information (i.e., spatial, descriptive, temporal, operational) and the corresponding representational available modality or media (e.g., still image, text). The task model is then complemented with dialogue acts based on the Rhetorical Structure Theory, to specify the desired communicative effects for each task step. Media selection based on heuristics rules can now be performed. These heuristics are divided in three categories:

- Advices on choice of media based on the role of domain data (for example, “spatial object relationship - prefer visual media”),
- General heuristics such as “present same material on two channels if available” or “use text and still images for key messages”),
- Validating heuristics to check that the combination of media ensures attention or does not overload the human processing. For example, “do not present different subject matters on separate channels” or “do not present a large amount of information on non persistent media”.

Arens et al. have developed an automated multimedia presentation planner. As in INTUITIVE, the allocation process is based on the description of a discourse structure [Arens 93]. The discourse structure is a tree-like structure of discourse segments which define the basic organization of the information to be presented. It is neutral with regard to the output modalities but expresses communicative goals of the system. For example, two discourse segments related by the “equative” relation express the goal of rendering the same information in different forms.

4. Conclusion

In this article, we have provided an overview of a number of frameworks and taxonomies that usefully enrich and extend our system-centered experience of M^4 systems. In the next future, we will elaborate on our current definition of the UOM framework.

Acknowledgements

This work has been supported by project ESPRIT BR 7040 AMODEUS2 and by PRC Communication Homme-Machine.

References

[Arens 93]
Y. Arens, E. Hovy, S. Van Mulken, “Structure and Rules in Automated Multimedia Presentation Planning”, IJCAI’93, pp. 1253-1259.

[Barnard 87]
P. Barnard, “Cognitive Resources and the Learning of Computer Dialogs", in Interfacing Thought, Cognitive aspects of Human Computer Interaction, J.M. Carroll Ed., MIT Press Publ., pp. 112-158.

[Bersen 93]
N. O. Bersen, “Taxonomy of HCI Systems: State of the Art”, ESPRIT BR GRACE, deliverable 2.1, 1993.

[Buxton 83]
W.Buxton, “There’s more to Interaction than meets the eye: Some Issues in Manual Input”, in D. A. Norman & Draper Ed., User Centered System Design, Lawrence Erlbaum, 1983, pp. 319-337.

[Coutaz 93]
J. Coutaz, L. Nigay, d. Salber, “The MSM framework: A Design Space for Multi-Sensori-Motor Systems; EWHCI’93, East/West Human Computer Interaction, Moscow, August, 1993.

[Duke 92]
D. Duke, M. Harrison, “Abstract Models for Interaction Objects”, ESPRIT BR 7040 Amodeus Project document, System Modelling/WP1, Nov. 1992.

[Foley 84]
J.D. Foley, V.L. Wallace, P.Chan, “The Human Factors of computer Graphics interactiontechniques”, IEEE computer Graphics and Applications, 4(11), 1984, pp. 13-48

[Frohlich 91]
D. Frohlich, “The Design Space of Interfaces”, Multimedia Systems, Interaction and Appplications, 1rst Eurographics workshop, Stockholm, Springer-Verlag, 1991, pp. 53-69.

[IHMM 92]
F. Azémard, T. Baudel, Y. Bellik, M.L. Bourguet, N. Carbonell, J. Coutaz, J.C. Martin, L. Nigay, K. Ouadou, P. Palanque, D.Teil, N. Vigouroux, “Interfaces Multimodales et Architecture Logicielle”, Atelier IHM’92, ENST, Décembre, 1992.

[Krueger 91]
Krueger M.W., Artificial Reality II, Addison Wesley, 1991.

[Mackinlay 90]
J.Mackinlay, S. Card, G. Robertson, “A Semantic Analysis of the Design Space of Input Devices”, Human Computer Interaction, Lawrence Erlbaum, Vol. 5, Numbers 2 & 3, 1990, pp. 145-190.

[Neal 88]
J. Neal, C. Thielman, K. Bettinger, J. Byoun, "Multi-modal References in Human-Computer Dialogue", Proceedings of AAAI-88, 1988, pp. 819-823.

[Nigay 93]
L. Nigay, J. Coutaz, “A design space for multimodal interfaces: concurrent processing and data fusion”, INTERCHI’93 Proceedings, Amsterdam, May, 1993, pp. 172-178.

[Nigay 94]
Conception et modélisation logicielles des systèmes interactifs : application aux interfaces multimodales, Thèse de doctorat de l’Université Joseph Fourier, Grenoble, 1994, 315 pages.

[Sellen 92]
A. Sellen, G.P. Kurtenbach, W. Buxton, “The Prevention of Mode Errors Through Sensory Feedback”, Human Computer Interaction, Lawrence Erlbaum,Vol. 7, No 2, 1992, pp. 141-164.

[Sutcliffe 93]
A. Sutcliffe, “Designing Multimedia Interfaces”, EWHCI’93, Moscou, Volume II, pp. 123-133.