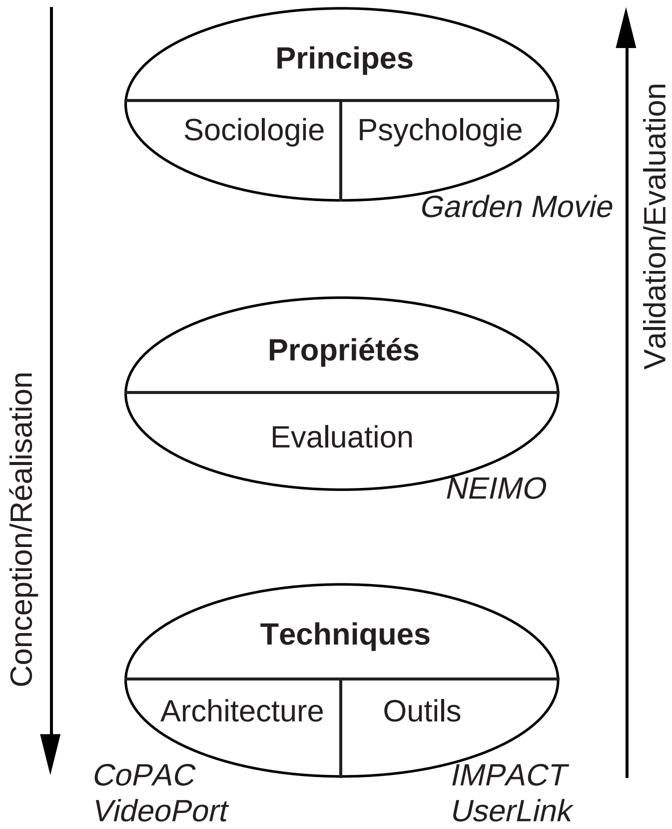
Conclusion

"Would you tell me, please,
which way I ought to go from here?"
"That depends a great deal
on where you want to get to," said the Cat

Lewis Carroll

Résumé de notre contribution

La figure ci-dessous reprend notre cadre structurant de la figure 1.8 (chapitre 1). Elle montre comment nous avons incarné chacun des trois niveaux et les contributions conceptuelles et techniques que nous y apportons.



Notre travail contribue à l'étude des systèmes multi-utilisateurs et de la communication homme-homme médiatisée selon quatre aspects complémentaires :

- face au nombre et à la diversité des disciplines dont l'étude de notre domaine doit tenir compte, nous avons proposé une structuration de l'analyse des apports des sciences non-informatiques à la conception et à la réalisation des systèmes multi-utilisateurs,

- à l'aide de ce cadre structurant, nous avons étudié l'extension aux systèmes multi-utilisateurs des acquis issus de l'étude de l'interaction homme-machine individuelle,

- pour chacun des aspects du domaine que nous avons étudiés, nous avons proposé des outils scientifiques comme les propriétés, les taxinomies, les espaces problèmes ou les modèles pour guider la réflexion sur les propositions existantes et développer de nouveaux concepts,

- enfin, des réalisations ont mis en œuvre et validé les concepts que nous avons développés.

Pour analyser l'intégration des apports des sciences sociales et humaines au processus de développement informatique, nous proposons une approche à trois niveaux : principes, propriétés et techniques (chapitre 1). Cette approche structurée nous permet de synthétiser les apports des sciences sociales et humaines sous forme de principes dont nous dérivons des propriétés. Nous montrons ensuite comment ces propriétés peuvent être vérifiées par l'utilisation de techniques appropriées d'évaluation et de réalisation.

Nous identifions plusieurs aspects significatifs de la conception et de la réalisation des systèmes multi-utilisateurs : les sciences sociales et humaines (chapitres 2 et 3), l'évaluation ergonomique (chapitre 5), la conception logicielle (chapitres 6 et 7) et les outils logiciels (chapitre 8). Pour chacun de ces aspects, nous évaluons et critiquons l'acquis provenant de l'étude des systèmes mono-utilisateurs dans la perspective des systèmes multi-utilisateurs. Nous proposons ensuite l'extension des concepts et des techniques présentés au cas des systèmes multi-utilisateurs et de la communication homme-homme médiatisée.

Nous proposons l'utilisation systématique des propriétés pour prendre en compte les enseignements des disciplines non-informatiques qui informent la conception et la réalisation logicielles. Les propriétés (chapitre 4) constituent un outil scientifique et permettent de jeter un pont entre les sciences "douces" comme les sciences humaines et sociales et la science "dure" qu'est l'informatique. Pour guider l'évaluation des travaux existants et pour élucider les composantes du domaine, nous proposons d'abord le trèfle fonctionnel, un espace de définition des systèmes multi-utilisateurs (chapitre 1), puis un espace problème pour les modèles d'architecture logicielle (chapitre 6), et la taxinomie IMPACT des mécanismes de connexion pour la communication homme-homme médiatisée (chapitre 8). Le modèle d'architecture logicielle CoPAC (chapitre 7) rassemble de façon équilibrée les requis fonctionnels des systèmes multi-utilisateurs que nous avons identifiés avec le modèle du trèfle.

Nous examinons la validité des concepts développés en les mettant en œuvre dans des réalisations. L'expérimentation Garden Movie, menée en collaboration avec des psychologues, étudie la communication humaine gestuelle et parlée par l'intermédiaire d'un lien audio/vidéo. Ses résultats font apparaître une difficulté de l'utilisation de la communication homme-homme médiatisée. Nous en déduisons une propriété d'après notre cadre structurant à trois niveaux. Nous avons étudié l'utilisation des propriétés pour l'évaluation ergonomique avec la plate-forme d'évaluation NEIMO en l'appliquant à l'évaluation de l'outil de communication Supratel. Nous avons examiné la validité du modèle d'architecture CoPAC en l'appliquant à trois réalisations : le jeu $(421)^n$, la plate-forme d'observation des utilisateurs NEIMO et le mediaspace VideoPort. Enfin nous avons appliqué l'espace de classification IMPACT pour évaluer les possibilités et les limitations de UserLink, notre bibliothèque de communication pour les médias continus.

Perspectives

Les perspectives de développement de notre travail s'organisent suivant quatre directions : l'exploration systématique des possibilités des mediaspaces pour la communication homme-homme médiatisée, la validation de notre approche structurée par de nouvelles collaborations avec des sciences non-informatiques, la validation extensive du modèle CoPAC et la prise en compte de la mobilité de l'utilisateur.

Par la construction d'espaces d'analyse, un de nos objectifs est l'exploration systématique des composantes d'une question. Pour la communication homme-homme médiatisée, nous avons fait un premier pas dans cette direction avec la classification IMPACT. D'autres espaces d'analyse nous permettront d'étudier de façon systématique les possibilités des mediaspaces. Le but de cette recherche est la construction d'un "espace des mediaspaces" qui rassemble les possibilités de ces outils de communication. Nous progressons dans cette direction avec l'expérimentation de caméras mobiles montées sur tourelle pivotante. L'utilisation d'un ensemble de caméras pivotantes afin de donner une vision globale et continue d'un espace distant est un de nos objectifs à court terme. Ce travail reprend en les étendant au cas de la communication homme-homme médiatisée les possibilités de vision à 360° d'outils comme Surround Video ou QuickTime VR.

Nous avons éprouvé la validité de notre vision structurée de l'apport des sciences non-informatiques (sous forme de principes, propriétés et techniques) avec notre collaboration pour l'expérimentation Garden Movie. Mais il n'est pas facile de concilier des méthodes d'analyse et des cultures scientifiques différentes. Nous voulons éprouver notre vision à trois niveaux en travaillant avec par exemple des spécialistes de l'éthique ou des sociologues. De notre point de vue, l'éthique informatique est amenée à prendre une importance croissante dans les années qui viennent. Collaborer tôt avec cette discipline permettra une compréhension mutuelle qui ne peut être que profitable. La participation de ces disciplines à l'évaluation doit aussi être étudiée.

Le modèle d'architecture logicielle CoPAC doit être validé de façon plus approfondie. Il nous faut réfléchir par exemple à l'application du modèle à des systèmes plus conséquents que ceux que nous avons étudiés. L'étude d'un système multi-utilisateur plus complet, intégrant de façon harmonieuse les trois facettes fonctionnelles, production, coordination et communication nous permettrait d'éprouver le modèle et de mettre en évidence ses limitations. Nous n'avons pas non plus abordé l'intégration du modèle avec des technologies comme les plates-formes à objets distribués.

L'interface homme-machine a jusqu'à présent principalement considéré un utilisateur assis devant un ordinateur de bureau. Mais les avancées technologiques permettent maintenant à l'utilisateur de profiter de son environnement : en utilisant des dispositifs mobiles, il peut se déplacer et utiliser l'informatique et les outils de communication dans de nouveaux lieux. Cette ubiquité augmente les possibilités de l'utilisateur mais lui impose peut-être de nouvelles contraintes. Nous pensons que cette étude peut donner lieu à l'élaboration de nouveaux principes et de nouvelles propriétés adaptés à ces outils novateurs. Ces outils posent aussi des questions originales à la communication homme-homme médiatisée.

Annexe A

The System Modelling Glossary

Salber D., Coutaz J. & Nigay L. (editors)
Laboratoire de Génie Informatique, University of Grenoble
Faconti G. & Paterno F.
CNUCE
Duke D. & Harrison M.
University of York

1st March 1994

Amodeus Project Document: System Modelling/WP 26

Une version hypertexte du System Modelling Glossary est consultable à l'adresse :

`http:://www-lgi.imag.fr/Les.Groupes/IHM/AMODEUSGlossary/GlossaryMain.html`

Abstract

The System Modelling Glossary gathers definitions for the vocabulary that is commonly used within the RP1 System Modelling Package. Its purpose is to serve as a support/companion for reading RP1 papers. It is also aimed at providing the Amodeus project with a reference vocabulary for discussing system modelling issues.

Instructions

Entries in the glossary may be words or expressions (e.g., “level of abstraction”). They are listed in alphabetical order. Some of the definitions may draw upon or replicate widely-accepted meanings. In this case, the source reference is denoted as [ref] and is listed in the “References” section at the beginning of the document.

Updates

As the vocabulary used within RP1 may evolve, some entries and definitions might become obsolete. Such entries are marked {Obsolete} and might be amended with a newer definition. Every member of the Amodeus project is welcome to suggest revisions to the RP1 glossary.

Flags

The three following entries are marked with an asterisk [*]: Affordance, Real-world validity and Task domain concept. The asterisk indicates that we strongly feel the need for other packages' input and comments on these definitions.

References

[CCS] A.J.R.G.Milner, *A Calculus of Communicating Systems*, Lecture Notes in Computer Science, Vol. 92, Springer Verlag, 1980.

[CGRM] Information technology - Computer graphics and image processing- *Computer Graphics Reference Model*, ISO 11072, International Organisation for Standardisation, 1992.

[Dictionary] *The American Heritage Dictionary of the English Language,* William Morris, editor, Houghton Mifflin Company, 1969.

[LOTOS] T. Bolognesi, H. Brinska,*Introduction to the ISO Language LOTOS*, Computer Networks and ISDN Systems 14, pp 25-59.

Abstraction [CGRM]
Processing element within a layer that transforms an information type. The transformed information is transferred to the next higher layer in a hierarchy or to another subsystem in the system when that layer is the highest one in that hierarchy.

Abstraction
(a) Process that transforms information into information whose semantic content and scope are richer/higher than the content and scope of the initial information.
(b) Result of the process of abstracting.

Abstraction function
Concept to denote the capacity of abstracting within a system.
See Rendering function.

Action [CCS or LOTOS]
Fact happening within a system identified by a name and by an information. The information can be either explicit (an instance of an information type is communicated) or implicitly associated to the event name (an information may be inferred from the event). Synonym for Event [CCS or LOTOS].

Action
(a) Transformation between two states of a system.
(b) Specification structure that defines a state transition by means of such a predicate.
(c) Indivisible operation performed by an agent whose effect can be perceived by another agent (in terms, for example, of events received through communication channels).
See Physical action, User physical action, System physical action, System conceptual action as subclasses of actions.

Action-tree [CCS or LOTOS]
Tree-like representation of a behaviour expression.

Affordance [*]
Property that the presentation of a system conveys information about the actions that can be performed by the user of that system. These properties relate to the "real world" understandings that the user has.

Agent
Object capable of initiating the performance of actions.

Alphabet
Union set of event names of a behaviour. Set of events in which some agent can engage.

Autonomous System
System capable of autonomous behaviour.

Backward recoverability
Property that the system provides the user with an undo facility to return to a previous state.
See Recovery, Recoverability.

Behaviour
(a) Set of all possible traces of a process. [CCS or LOTOS]
(b) An ordered set of states. See Trace.

Behaviour expression
Description of a behaviour within some formalism (e.g., action-tree, temporal logic, or process algebra).

Bias
Structure in a model that is not essential for characterising the behaviour or properties of a system.

Browsability
Property that the system provides the user with commands to make perceivable different portions of the system functional state. (By modifying the presentation state, the user may access different portions of the state of the functional core.)

Channel
(a) Common path shared by two or more components of a system.
(b) System structure/component ued for the communication of information.
See Physical channel, Digital channel, Human channel.

Command
(1) Set of physical actions transformed by the system into a system conceptual action.
(2) System conceptual action.
See Inspection command, Task domain command.

Communication [Dictionary]
Act of transmitting information.

Computation
Ordered set of operations that transforms the state of a system.

Component
Part of a whole. Depending on the context, can be instantiated as a software module, a subsystem, an agent, an interactor, an abstraction, a device, etc.

Conceptual action
Action performed by the system at the highest level of abstraction, i.e., a primitive function in the functional core.
See Level of abstraction.

Conceptual unit
Part of the state of the functional core that models a task domain concept at the highest level of abstraction.

Conformance
Property that the presentation of a system mirrors the underlying behaviour of the system.
See Layer conformance, Observability, Honesty.

Connectedness
Property of presentation that requires that notions of continuity in the state are faithfully rendered in the presentation.

Context
Set of state vectors used by a system in a computation. Examples of state vectors: state vector about the user, state vector about the presentation, state vector about the dialogue, state vector about the functional core, state vector about the environment.
See Dynamic context, Static context.

Continuity
The property that given a function f:A>B between ordered sets A and B, the neighbourhood around any point in set B corresponds to some neighbourhood around the inverse image of that point under f. This is a topological definition that generalises the

usual notion of continuity found for example in real analysis, and is useful in that it also applies in the case of discrete sets.

Deontic logic
A logic for reasoning about permissible or obligatory actions.

Detour
Interaction that is not directly involved in achieving a goal.

Device
(1) Physical artefact necessary to a system to acquire (input device) or deliver (output device) information. Examples include keyboard, loudspeaker, ears and mouth.
(2) Lowest level component of a system whose state changes result from physical actions.

Device assignment
Relation between a device over a state and a non empty subset of expressions of an interaction language. A device d is assigned in state s to a set E of expressions of a language l, if it does not exist any device equivalent to d over s and E. Assignment is permanent if the relation holds for any state. Assignment is total if the relation holds for E equals to the set of expressions that define l. For example, in Matis, the mouse is permanently assigned to the expression of window resizing in the direct manipulation interaction language.

Device equivalence
Relation between a non empty set of devices over a state and a non empty set of expressions in an interaction language. Devices in a set D are equivalent over a state s and a non empty set E of expressions in an interaction language L, if all of the expressions of E can be elaborated using either one of the devices in D. Equivalence is permanent if the relation holds for any state. Equivalence is total if the relation holds for E equals to the set of expressions that define L. For example, in Matis, keyboard and microphone are totally and permanently equivalent over natural language.

Device redundancy
Relation between a set of devices over a state and an expression of an interaction language. Devices of a set D are used redundantly in some state s for an expression e of a language l, if these devices are equivalent over s and e, and if they are used simultaneously to express e. For example, the user can spell a character using the microphone and type in the same character.

Device complementarity
Relation between a set of devices over a state and a non empty subset of expressions of an interaction language. Devices of a set D are complementary over a state s and a non empty set E of expressions of a language l, if E can be partitionned such that for each partition Ep of E, it exists a device d of D assigned over s and Ep. Complementarity is permanent if the relation holds for any state. Complementarity is total if the relation holds for E equals to the set of expressions of l. Language complementarity is best illustrated by spoken natural languages where concept names must be typed in. For example, in Munix, a multimodal user interface for Unix, commands that involve a file name such as remove, can be expressed using the microphone for the command name and options while file names must be elaborated with the keyboard.

Dialogue
(1) Same as interaction.
(2) Restriction on the behaviour of the interactive system to the set of actions performed with the participation of both human and computer processes.

Dialogue strategy
Set of behaviour expressions enforcing constraints over a dialogue. Examples of dialogue strategies include: reactive strategy (the system supplies the missing parameters of a command = default values), cooperative strategy (the system engages in a subdialogue to propose the set of possible values for a missing parameter and the user disposes), directive strategy (the system engages in a subdialogue to ask for the value of a missing parameter), negociated strategy (the system engages in a subdialogue to propose a value for the missing parameter and the user disposes), intentional strategy (the system engages in a subdialogue to check with the user that the value for the missing parameter is correct).

Dynamic context
Portion of the context that may vary over time, e.g., during a session. This variation can affect the values of the set of state variables, as well as the actual set of state variables (as in learning systems).

Dynamic system
See Autonomous system.

Equivalence
A relation that is reflexive, symmetric and transitive.
See Language equivalence, Device equivalence.

Event [CCS or LOTOS]
see Action [CCS or LOTOS]. The two terms have the same meaning in the CCS or LOTOS theory.

Event
(a) Data structure used to transfer information between components of a system.
(b) The observation that some action has occurred.

Event class
Name of an event.

Event instance
An instance of an event of a particular class.

Event occurrence
Same as Event instance.

Fan-in [CGRM]
Gathering of information units from multiple source processes to a single process.

Fan-out [CGRM]
Distribution of information units from one source process to multiple processes.

Feedback
See Response.

Fission
(a) Computation of a process abstracting/presenting an information type into a collection of different information types to be transferred to a set of processes.
(b) Decomposition of an information type at some level of abstraction into multiple information types of the same level of abstraction.
See Fan-out, Fusion.

Formal Method
Collection of mathematical structures, together with a precise syntax for defining instances of those structures and organising them into domain-specific abstractions. It should be associated with a method for eliciting these abstractions and transforming them.

Frame
A collection of variables that are referenced or modified by an action.

Functional core
Component of a system that implements task domain concepts.

Functional state
State vector of the functional core.

Fusion
(a) Computation of a process abstracting/concretizing a collection of information types received from distinct processes into a different information type to be transferred to another process.
(b) Composition of multiple information types at some level of abstraction into a single information type of the same level of abstraction.
See Fan-in, Fission.

Goal
Desired property of a system; usually expressed as a state or set of states that a user intends to achieve. Particularly for reactive or control systems, goals may also be expressed as predicates over the behaviour or trace of the system.

Guard
See Precondition.

Homomorphism
In general, structure preserving transformation where `structure' can be interpreted within many fields of mathematics. For example, a homomorphism between ordered sets <S,R> and <Q,T> is a function f from S to Q such that for all x,y in S, if (x,y) in R then (f(x), f(y)) is in T.

Honesty
Property that the presentation of the system renders its functional state appropriately (e.g., does not distort the functional state). Honesty is a necessary but not always sufficient condition for the user to be able to elaborate a correct mental representation of the functional state.
See Connectedness as a special case.

Human channel
Physical channel used by a human being to acquire (sensori channel) or deliver (motor channel) information.

Initiative
Denotes the agent that leads an interaction. In many cases the interactional initiative is clear. For example, in the case of a system like UNIX the initiative is with the user, whereas in many expert systems the system takes the initiative (the user simply has to respond to questions supplied by the expert system).

Input physical channel
A physical channel involved in the acquisition of information.

Interaction
Non-empty ordered set of events involving more than one agent.

Interaction dissonance
Occurs when an agent interacts incorrectly in the sense that an appropriate communication is directed to an inappropriate agent. This notion is intended to express the situations where two agents interact at cross purposes. A typical example of this occurs when a user interacts in a way that is appropriate through one window of a window based system with the wrong window selected.

Interaction language
Language used by the user or the system to exchange information. A language defines the set of all possible well-formed expressions, i.e., the conventional assembly of symbols, that convey meaning. The generation of a symbol or a set of symbols, results from a physical action. It is modelled as an event produced via some physical channel carrying a message which is this symbol or set of symbols.

Interaction object
An agent that participates in an abstraction and/or presentation transformation. It may interact directly with a human agent.
See also: Interactor.

Interaction trajectory
See Interaction.

Interactional invariance
Measure of the vulnerability of an interaction to the actions of other agents that are not taking part directly in the interaction. The issue is the way that interactive system goals are influenced by autonomous behaviour as would be a concern if we were talking of autonomous systems. An interactive system should be capable of supporting certain objectives regardless of the other activities that are going on autonomously within the system.

Interactionally-rich
(a) Qualifies interactive systems that support multiple input interaction languages and/or multiple output interaction languages and these languages can be conveyed via multiple input devices and/or multiple output devices.
(b) is used to refer to interactive systems which engage user or users in a "natural" interactive environment. Users can employ the multiple interaction languages or modalities to have a more natural interaction with the objects of the work domain.
In this sense, therefore, a hammer which uses few modalities and interaction languages is interactionally rich because of the texture and directness of the interaction.

Interactive system
System for which a subsystem is explicitly defined to be a human.

Interactor
See Interaction object.

Language assignment
Relation between an interaction language over a state and a non empty subset of conceptual units of a system. An interaction language l is assigned in state s to a set of conceptual units C, if it does not exist any interaction language equivalent to l over s and c. Assignment is permanent if the relation holds for any state. Assignment is total if the relation holds for C equals to the set of conceptual units of the system.

Language complementarity
Relation between a set of interaction languages over a state and a non empty subset of conceptual units. Interaction languages of a set L are complementary over a state s and a non empty set C of conceptual units of the system, if C can be partitionned such that for each partition Cp of C, it exists a language l of L assigned over s and Cp. Complementarity is permanent if the relation holds for any state. Complementarity is total if the relation holds for C equals to the set of conceptual units of the system. Language complementarity is best illustrated by coreferential expressions. For example, in Matis, natural language and direct manipulation are complementary over the conceptual unit “city” and any state where the specification of a city name is possible : “flights from this city” and selection of a city name through direct manipulation.

Language equivalence
Relation between a set of interaction languages over a state and a non empty subset of conceptual units of a system. Interaction languages of a set L are equivalent over a state s and a non empty set C of conceptual units of the system, if all of the conceptual units in C can be represented using either one of the language in L. Equivalence is permanent if the relation holds for any state. Equivalence is total if the relation holds for C equals to the set of conceptual units of the system. For example, in Matis, direct manipulation language and natural language are permanently equivalent for specifying requests.

Language redundancy
Relation between a set of interaction languages over a state and a conceptual unit of a system. Interaction languages of a set are used redundantly in some state s for a conceptual unit c, if these languages are equivalent over s and c, and if they are used simultaneously to represent c. For example, a wall is represented redundantly by the system via a red line (graphics interaction language) and the message “mind the red wall!” (natural language).

Layer conformance
(a) Relation between the abstraction/presentation transformations and the information types bound to two adjacent layers with respect to some invariants.
(b) Generalisation of conformance between any pair of layers.

Layer
Hierarchical partition of a subsystem into elements for which specific processing and/or transferring entities as well as information types are precisely defined.

Level of abstraction
(a) Layer within a system whose information types are characterized by a given semantic content and scope. The lowest level of abstraction corresponds to the poorest information type with regard to scope and content. The highest level of abstraction corresponds to the richest information type with regard to scope and content. These levels as well as any level in-between depend on the perspective or the objective of the modeller and/or the modelling technique.
(b) different perspectives in the design process of a system.

Media
Same as physical device type or set of physical devices types used for communication.

Message
Any value transmitted on a channel.

Metaphor
Representation of one system model in terms of another that has some real world significance or analogy, compare affordance. The classic example is the metaphor by which the file and directory structure of an operating system is understood in terms of a desktop with windows (folders) and icons.
See Homomorphism.

Migratability
Ability of the interactive system to dynamically transfer performance between agents.

Modal
Having possible modes or worlds; modal logic is a system for reasoning about possible and necessary truth.

Modality
Method characterized by the type of usage of interaction languages and physical devices for communicating information at the user interface.

Mode
Same as Context.

Model
Any representation of a real or imagined system.

Multidevice
Property of the system that provides the user with more than one physical device (simultaneously or not) to communicate information to the system and/or uses more than one physical device (simultaneously or not) to communicate information to the user.
See Multimedia.

Multilanguage
Property of the system that provides the user with more than one interaction language (simultaneously or not) to communicate information to the system and/or uses more than one interaction language (simultaneously or not) to communicate information to the user.

Multimedia
(a) Use of more than one medium for communication.
(b) Same as multidevice.

Multimediality
Property of a system that supports multimedia interaction.

Multimodal
Which provides the user with more than one modality (simultaneously or not) to communicate information to the system and/or uses more than one modality (simultaneously or not) to communicate information to the user.

Multimodality
Property of a system that supports multimodal interaction.

Objective
In general we shall be interested in the extent to which a system supports a set of objectives. An objective is considered to be a pre- and post-condition on state which may be linked with the agent who owns the objective.

Observability
Property that the presentation of a system contains sufficient information to allow the user to determine the functional state of the system.
See Conformance, Browsability, Honesty.

Ordered Set
A set S and relation R over S such that R is reflexive, transitive, and maybe antisymmetric .

Output physical channel
Physical channel that delivers information.

Parallelism
Simultaneity in the activity of multiple agents. Two events performed by different agents engaging in parallel activities may occur in overlapping time intervals.

Passive command
Command whose effect does not modify the state of the functional core of a system.

Performance
Interaction trajectory that achieves an objective.

Physical action
Action performed either by the user or by the system on a physical device.

Physical channel
Channel involving a physical device.
See Input physical channel, Output physical channel, Channel, Human channel.

Poset
Ordered set <S,R> where the relation R is a antisymmetric.

Postcondition
Predicate that describes the state of a system after the performance of some action. Actions are often specified by pre- andpost-condition pairs.

Potential
Scope that exists for exploiting the functionality of the interactive system at any stage in an interaction irrespective of any interactional initiative. "Event" potential is the user's scope for engaging in alternative actions yet continuing to achieve the same objective (for example filling a form by entering the fields in any possible permutation). "Objective" potential is the user's scope for achieving alternative objectives at a particular stage in the interaction. Hence low objective potential occurs when a user is locked into a single objective at a stage in the interaction.

Precondition
Predicate on the state of a system that determines those states from which an action can be carried out.

Predicate
A boolean-valued function of the state, behaviour, or trace of a system. Apredicate may represent a property.

Prefix
(of an ordered set <S,R>) is a subset T of S such that for every x in S and y in T, if (x,y) is in R then x is in T. Note that the empty set <0,R> and the full set <S,R> are always prefixes of<S,R>.

Prefix Closure
(of an ordered set <S,R>) is the set of all ordered sets <T,R> such that <T,R> is a prefix of <S,R>.

Presentation [CGRM]
Processing element within a layer that transforms an information type. The transformed information is transferred to the next lower layer in a hierarchy or to another sub-system in the system, when that layer is the lowest one in that hierarchy.

Presentation
(a) That part of the system or component of a system whose state can be perceived and potentially perceived by the user through browsing commands.
(b) Perceivable state of the system or component of a system or potentially perceivable state of the system or component of a system through inspection commands.

Presentation state
State vector of the presentation component of the system.

Process
Same as Agent.

Property
An observable of a system that can be described by a predicate and measurable (if used to assess the usability of a system).

Proportionality
Property that holds when measurable properties of a presentation are proportional (in the mathematical sense) to some value or function in the functional state.

Reachability
Property that requires that some state or set of states can be reached from a given state through the actions defined on a system.

Real world validity [*]
that a metaphor corresponds to the users intuitions about what the system will do, for example the mapping of left in (visual) space to an appropriate notion in the underlying model.
See Metaphor, Affordance.

Recovery
Performance of actions that take a system from some `unsafe' or undesired state to one satisfying some safety property.
See Recoverability.

Recoverability
Property that the system provides the user with commands to undo the effect of some action.
See Recovery, Undoability.

Refinement
(1) A model C of a system that includes the observables of a model A and whose behaviour is consistent with that of A - the actual consistency requirements vary between refinement notions. C is called the concrete system and A is the abstract system. Usually C will introduce new observables.
(2) The process of developing a refinement (1) for some system.

Rendering, a
Some perceivable presentation.

Render, to
To make perceivable.

Rendering function
Concept to denote the concretization activity within a system.
See Abstraction function.

Rendering relation
A relation between the functional state and presentation of a system or component.

Repair
See Recoverability.

Response
An event generated or caused by some agent.

Sequential process
Process whose behaviour is a totally ordered set of events, i.e., a process that receives at most one event at a time (e.g., if two events arrive at the same time, one is processed, one is lost or delayed).

Semantic delegation
Transfer of task domain concepts from the functional core into the user interface.
See Semantic repair.

Semantic repair
Semantic delegation used to augment the functional core with concepts that were forgotten in a previous implementation of the functional core and that cannot be incorporated in the functional core for technical reasons (e.g., source code is not available).

Session, user
Step between the initial state and the terminal state used by the system for the user.

Software architecture, a
A particular set of software components satisfying a set of relationships and described at some level of refinement (does not necessarily derive from an architecture model).

Software Architecture Model
Guide (model) for developing a software architecture.

Software Architecture Model, Conceptual
Guide (model) for developing a software architecture whose components are described at a conceptual level.

Software Component
Software unit that encapsulates a set of logically connected operations and data.

State
Assignment of values to names representing the observables of a system.

State vector, of a component, of a system
Names that define the state of a component, of a system.

Static context
Portion of the context that does not vary over session but may be modified between sessions.

Step
Any pair of states in a behaviour. For action A, an A-step is a pair of states that are related by A.

Stimulus
Event received by some component of a system from another.

Stuttering step
Step that leaves some subset of the observables unchanged.

Subsystem
Component of a system that behaves like a system.

System
A whole or collection of components capable of performing information processing and/or information transfer, and whose behaviour can be observed and possibly measured.

System physical action
Action performed by the system whose effect is made perceivable through output physical channels and possibly captured by human input channels.

System state
Same as State.
See Functional state, Presentation state.

Task
(a) A goal, together with some procedure or ordered set of actions that will achieve the goal.
(b) An ordered set of actions that will achieve a given goal when performed from a state satisfying some precondition.

Task domain concept [*]
Concept identified by task analysis as relevant to the user to accomplish the tasks in that domain.

Task domain command
Command whose effect modifies the functional core component of the system.

Template
Subset of the observables of a system that are relevant to the performance of some task.

Trace
An ordered set of events.
See Trajectory.

Transition
A pair of states, usually defining the effect of an action on a system.

Undoability
Same as Backward recoverability.

User interface system
Component of a system that implements the abstraction and/or rendering process of information between the functional core and the physical devices of the system. It includes the presentation component.

User interface
Common shortcut for User interface system.

User interface state
State vector of the user interface subsystem.

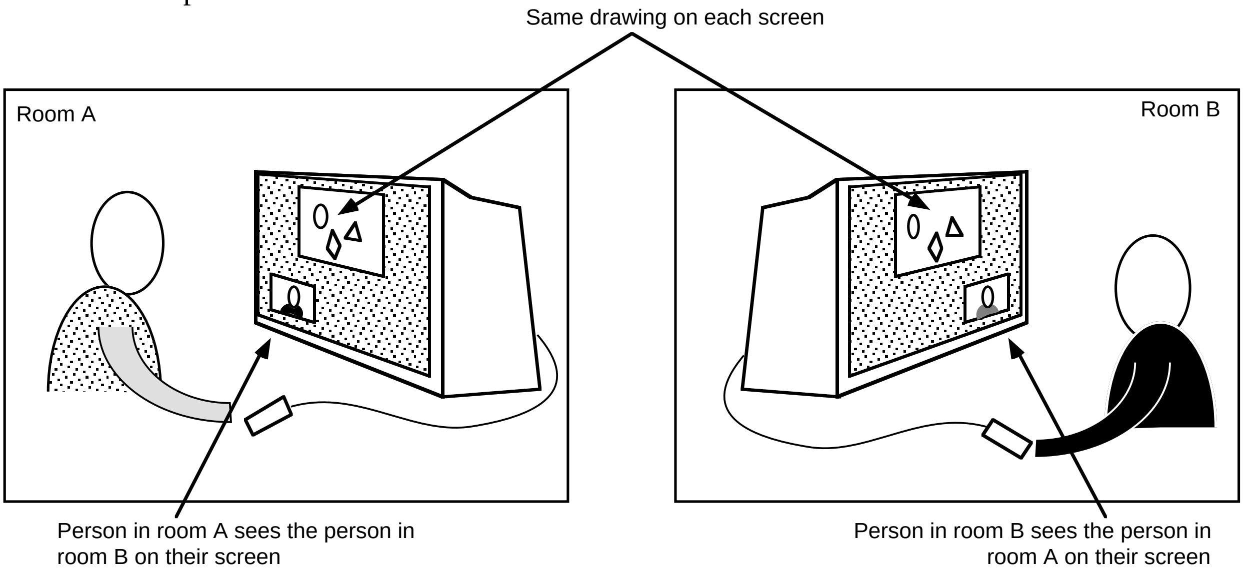
User physical action
Physical action initiated by the user.

Annexe B

Instructions de l'expérimentation "Garden Movie"

Ces instructions étaient lues aux sujets pour leur présenter l'expérimentation.

Modern computer systems are being developed so that people in different offices can see each other on their computer screens.
This can help them when they are discussing a piece of work that they can both see on their computers:



This experiment is looking at the usefulness of different camera positions.
You are playing the role of one of the people in the figure. You will see a 'window' from a drawing program, and your task is to move two of the objects in the window to new positions.
Your 'colleague' in another room will tell you which two objects to move, and where to move them to. You will be able to hear him and see a video of him on your screen.
You will now be shown an example…

Here is a 'drawing' that you and your colleague can both see. Both of you can see exactly the same picture.

Here is the 'video camera' that will show your colleague. Since this is just an experiment, we are using a recording instead of a 'live' picture. At the start of each clip, you will be able to see your colleague.

He will ask you to move two of the objects in the drawing. Listen and watch carefully, because the clip will vanish when he has finished talking.

As soon as he has finished talking, please move the objects he has indicated to their new positions.

You can move objects by moving the mouse pointer over them, and holding the button down while you drag them around. When you release the button, you 'let go' of the objects and they stay where you have left them.

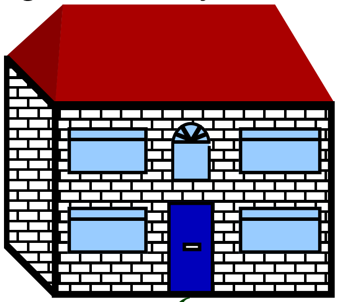
You can move all of the objects as many times as you want, but you cannot move them out of the drawing window.

Try it now…

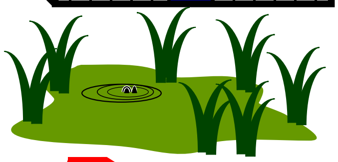
If you forget what he wanted you to do, or you were not clear about his instructions, you can press the 'play again' button with the mouse pointer. He will repeat his instructions, and the objects will be returned to their original places.
As soon as you have finished moving the objects, click the 'OK' button.
Please work as quickly as you can without making mistakes. Do you have any questions?
The rest of the experiment is divided into several sections.
In each section you will be given a different view of your 'colleague', or a different 'drawing' to work on.
There will be several clips for you to work on in each section.

In this section, you will see your colleague from the front. The picture you will see of him has been taken by a video camera placed just above his computer screen.
The drawing that you are working on is a plan of a garden. The objects it includes are:

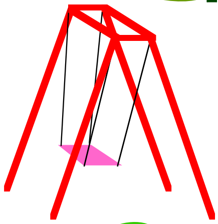
a house



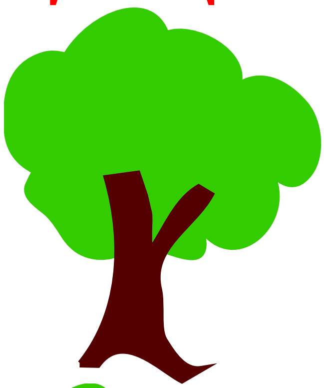
a pond



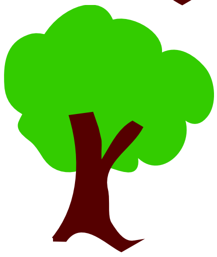
a swing



a big tree



a small tree

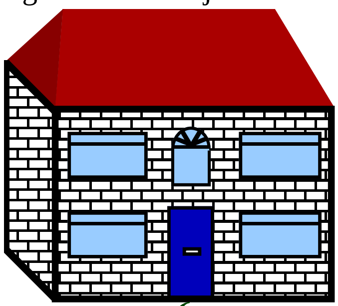


You can move these objects to the left or to the right, and you can move them behind each other, or in front of each other.

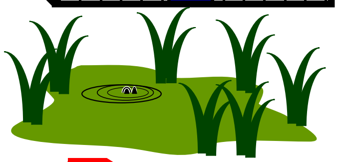
In this section, you will see your colleague from the front. The picture you will see of him has been taken by a video camera placed just above his computer screen, and has been reversed to form a mirror image.

The drawing that you are working on is a plan of a garden. The objects it includes are:

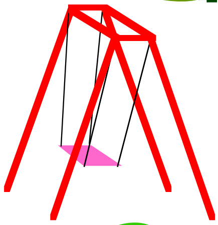
a house



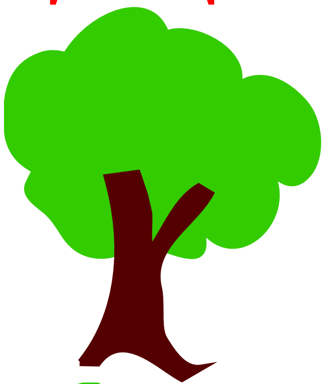
a pond



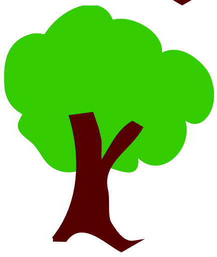
a swing



a big tree



a small tree

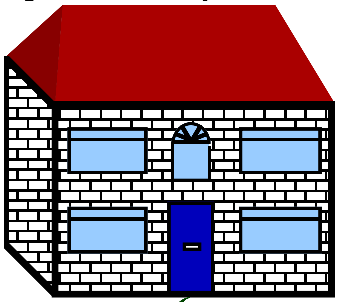


You can move these objects to the left or to the right, and you can move them behind each other, or in front of each other.

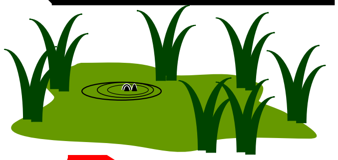
In this section, you will see your colleague from the side. The picture you will see of him has been taken by a video camera placed slightly behind him.

The drawing that you are working on is a plan of a garden. The objects it includes are:

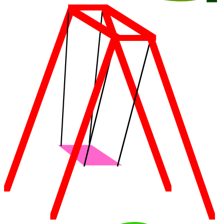
a house



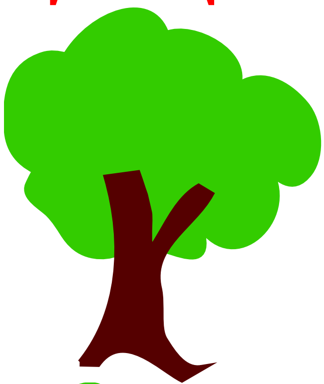
a pond



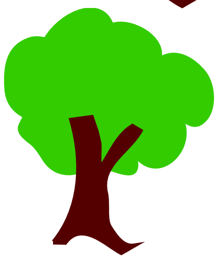
a swing



a big tree



a small tree

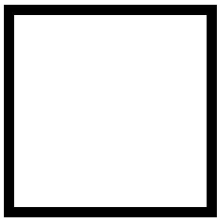


You can move these objects to the left or to the right, and you can move them behind each other, or in front of each other.

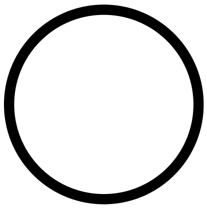
In this section, you will see your colleague from the front. The picture you will see of him has been taken by a video camera placed just above his computer screen.

The drawing that you are working on is a geometrical design. The objects it includes are:

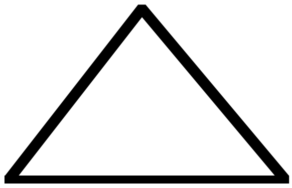
a square



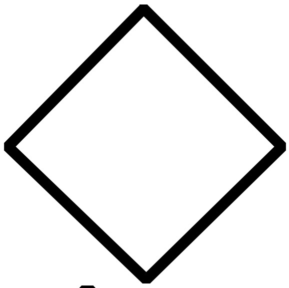
a circle



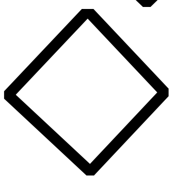
a triangle



a big diamond



a small diamond

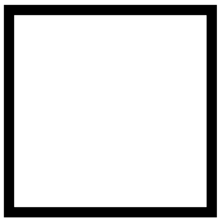


You can move these objects to the left or to the right, and you can move them above each other, or below each other.

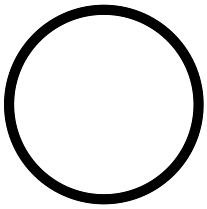
In this section, you will see your colleague from the front. The picture you will see of him has been taken by a video camera placed just above his computer screen, and has been reversed to form a mirror image.

The drawing that you are working on is a geometrical design. The objects it includes are:

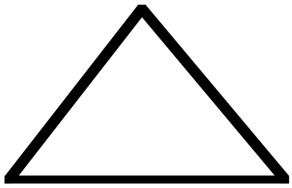
a square



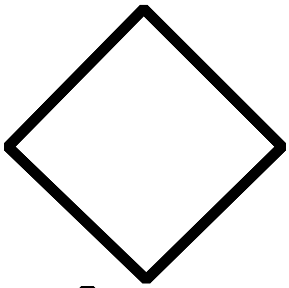
a circle



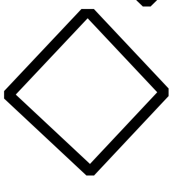
a triangle



a big diamond



a small diamond

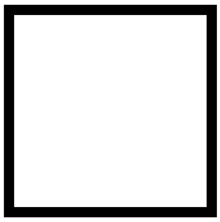


You can move these objects to the left or to the right, and you can move them above each other, or below each other.

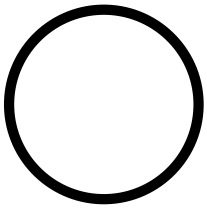
In this section, you will see your colleague from the side. The picture you will see of him has been taken by a video camera placed slightly behind him.

The drawing that you are working on is a geometrical design. The objects it includes are:

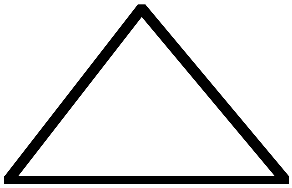
a square



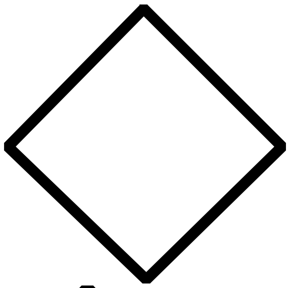
a circle



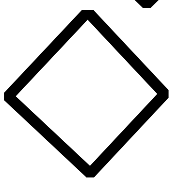
a triangle



a big diamond



a small diamond



You can move these objects to the left or to the right, and you can move them above each other, or below each other.

Références bibliographiques

[Abowd 1994]
G. D. Abowd. *Defining reference models and software architectural styles for cooperative systems,* Workshop on Software architectures for cooperative systems at CSCW'94, ACM Conference on Computer-Supported Cooperative Work, Chapel Hill, North Carolina, USA, 1994.

[Andler 1992]
D. Andler. *Introduction,* in *Introduction aux sciences cognitives.* D. Andler, (ed.) Gallimard, Paris, France, 1992.

[Apple 1988]
The Knowledge Navigator, vidéo. Apple Computer Inc., Cupertino, California, USA, 1988.

[Apple 1991]
Apple. *The Apple Event Manager,* Apple Computer Inc., Cupertino, California, USA, 1991.

[Apple 1993]
Apple. *OpenDoc Human Interface Guidelines (preliminary),* Apple Computer Inc., 1993.

[Apple 1994]
Finder 7.5. Logiciel pour Macintosh. Apple Computer Inc., Cupertino, California, USA, 1994.

[Baecker 1992]
R. M. Baecker, D. Nastos, L. R. Posner et K. L. Mawby. *The user-centred iterative design of collaborative writing software,* Workshop on Real Time Group Drawing and Writing Tools (CSCW'92, ACM Conference on Computer-Supported Cooperative Work), Toronto, Canada, 1992.

[Balbo 1994]
S. Balbo. *Evaluation ergonomique des interfaces utilisateur: un pas vers l'automatisation.* Thèse de doctorat, Université Joseph Fourier, Grenoble I, 1994.

[Ballé 1992]
C. Ballé. *Sociologie des organisations,* Presses Universitaires de France, Paris, France, 1992.

[Barlow 1994]
J. P. Barlow. *The Economy of Ideas,* in *Wired*, 2(03), mars 1994.

[Barnard 1985]
P. J. Barnard. *Cognitive Resources and the Learning of Computer Dialogs*, in *Interfacing Thought*, Cognitive aspects of Human Computer Interaction. J. M. Carroll, (ed.) MIT Press, 1985. pp. 112-158.

[Barnard 1992]
P. J. Barnard et J. May. *Real time blending of data streams: a key problem for the cognitive modelling of user behaviour with multimodal systems,* Amodeus Project, Working Paper, UM/WP 26, 1992.

[Barnard 1994]
P. J. Barnard, J. May et D. Salber. *Deixis and Points of View in Media Spaces*, Amodeus Project, Working Paper, UM/WP 19, 1994.

[Bass 1992]
L. Bass, R. Little, R. Pellegrino, S. Reed, R. Seacord, S. Sheppard et M. R. Szczur. *The UIMS Tool Developers' Workshop: A Metamodel for the Runtime Architecture of an Interactive System,* in *SIGCHI Bulletin*, 24(1), janvier 1992. pp. 32-37.

[Bass 1994]
L. Bass et G. Abowd. *Software Architecture: A Tutorial Introduction*, Software Engineering Institute, Carnegie-Mellon University, Tutorial, 1994.

[Bellotti 1994]
V. Bellotti et A. MacLean. *Integrating and Communicating Design Perspectives with QOC Design Rationale*, Amodeus Project, Working Paper, ID/WP29, 1994.

[Bentley 1992]
R. Bentley, J. A. Hughes, D. Randall, T. Rodden, P. Sawyer, D. Shapiro et I. Sommerville. *Ethnographically-informed systems design for air traffic control*, CSCW'92, ACM Conference on Computer-Supported Cooperative Work, Toronto, Canada, 1992. pp. 123-129.

[Bentley 1994]
R. Bentley, T. Rodden, P. Sawyer et I. Sommerville. *Architectural Support for Cooperative Multiuser Interfaces*, in *IEEE Computer*, 27(5), mai 1994. pp. 37-46.

[Berne 1995]
Y.-H. Berne et B. Hocq. *SIDECOM (Suivi d'individus destiné à l'extension de la communication médiatisée)*, Equipe IHM, Laboratoire de Génie Informatique, Institut IMAG, Rapport de stage, 1995.

[Bernoux 1985]
P. Bernoux. *La sociologie des organisations*, Editions du Seuil, Paris, France, 1985.

[Bernsen 1994]
N. O. Bernsen et J. Ramsay. *An executive summary of the DSD framework illustrated by two worked exemplars*, Amodeus Project, 1994.

[Bier 1991]
E. A. Bier et S. Freeman. *MMM: A User Interface Architecture for Shared Editors on a Single Screen*, UIST'91, ACM Symposium on User Interface Software and Technology, Hilton Head, South Carolina, USA, 1991. pp. 79-86.

[Bly 1993]
S. A. Bly, S. R. Harrison et S. Irwin. *Media Spaces: Bringing People Together in a Video, Audio, and Computing Environment*, in *Communications of the ACM*, 36(1), janvier 1993.

[Boersma 1994]
P. Boersma. *Experimental Research into Usability and Organisational Impact of Workflow Software*. Master's Thesis, Université de Twente, Pays-Bas, 1994.

[Bourguet 1992]
M.-L. Bourguet. *Conception et réalisation d'une interface de dialogue personne-machine multimodale*. Thèse de doctorat, Institut National Polytechnique de Grenoble, 1992.

[Brothers 1990]
L. Brothers, V. Sembugamoorthy et M. Muller. *ICICLE: Groupware for Code Inspection*, CSCW'90, ACM Conference on Computer Supported Cooperative Work, Los Angeles, California, USA, 1990. pp. 169-181.

[Bruckman 1994]
A. Bruckman. *Approaches to Managing Deviant Behaviour in Virtual Communities (Panel)*, CHI 95, ACM Conference on Human Factors in Computing Systems, Boston, Massachussets, USA, 1994. pp. 183-184.

[Bulick 1989]
S. Bulick, M. Abel, D. Corey, J. Schmidt et S. Coffin. *The US WEST Advanced Technologies Prototype Multi-media Communications System*, GLOBECOM'89, IEEE Global Telecommunications Conference, Dallas, Texas, USA, 1989.

[Buxton 1990]
W. Buxton et T. Moran. *EuroPARC's Integrated Interactive Intermedia Facility (IIIF): Early Experiences*, IFIP Conference on Multi-User Interfaces and Applications, Heraklion, Crète, 1990.

[Buxton 1993]
W. A. S. Buxton. *Telepresence: Integrated Shared Task and Person Spaces,* in *Readings in Groupware and Computer-Supported Work.* R. M. Baecker, (ed.) Morgan Kaufman, San Mateo, California, USA, 1993. pp. 816-822.

[Buxton 1994]
W. Buxton. Integrating the Periphery and Context: A New Taxonomy of Telematics, ERGO-IA'94, Biarritz, France, 1994.

[Card 1983]
S. K. Card, T. P. Moran et A. Newell. *The Psychology of Human-Computer Interaction*, Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1983.

[Charon 1993]
J.-P. Charon, L. Tézier, M. Carli et S. Liska. *Le projet Supratel : étude de l'utilisabilité d'un terminal de communication multiservices*, DESS Génie Informatique, Laboratoire de Génie Informatique, Université Joseph Fourier Grenoble 1, Rapport de stage, 1993.

[Clark 1991]
H. H. Clark et S. E. Brennan. *Grounding in Communication*, in *Perspectives on Socially Shared Cognition.* L. B. Resnick, J. M. Levine et S. D. Teasley, (eds.). American Psychological Association, Washington, DC, USA, 1991. pp. 127-149.

[Coiffier 1990]
E. Coiffier, Y. Crozet, D. Dehoux-Grafmeyer, F. Faure et J.-F. Renaud. *Sociologie basique*, Nathan, Paris, France, 1990.

[Coutaz 1987]
J. Coutaz. *PAC, an Implementation Model for Dialog Design*, Interact'87, IFIP Conference on Human-Computer Interaction, Stuttgart, 1987. pp. 431-436.

[Coutaz 1990]
J. Coutaz. *Interface homme-ordinateur : conception et réalisation*, Dunod, Paris, France, 1990.

[Coutaz 1992]
J. Coutaz, G. Abowd et L. Nigay. *Refining Software Engineering Quality Factors with HCI Factors*, HCI'92, UK, 1992.

[Coutaz 1993]
J. Coutaz, L. Nigay et D. Salber. *The MSM Framework: A Design Space for Multi-Sensori-Motor Systems*, in *Human-Computer Interaction, Third International Conference, EWHCI'93, Moscow, Russia, August 3-7, 1993, Selected Papers*. Lecture Notes in Computer Science n° 753, L. J. Bass, J. Gornostaev et C. Unger, (eds.). Springer-Verlag, Berlin, 1993. pp. 231-241.

[Coutaz 1995]
J. Coutaz, L. Nigay, D. Salber, A. E. Blandford, J. May et R. M. Y. Young. *Four Easy Pieces for Assessing the Usability of Multimodal Interaction*, INTERACT'95, IFIP Fifth International Conference on Human Computer Interaction, 25-29 juin 1995, Lillehammer, Norvège (à paraître), 1995.

[Croisy 1994]
P. Croisy. *Modèle multi-agent et conception d'applications coopératives interactives*, IHM'94, Sixièmes journées sur l'Ingénierie des Interfaces Homme-Machine, Lille, France, 1994. pp. 139-144.

[CU-SeeMe 1995]
CU-SeeMe . Logiciel pour Apple Macintosh. Cornell University, 1995.

[Cypher 1991]
A. Cypher. *EAGER: Programming Repetitive Tasks by Example*, CHI'91, ACM Conference on Human Factors in Computing Systems, New Orleans, Louisiana, USA, 1991. pp. 33-39.

[Dahlbäck 1988]
N. Dahlbäck et A. Jönsson. *Talking to a computer is not like talking to your best friend*, SCAI-88, Scandinavian Conference on Artificial Intelligence, 1988. pp. 53-68.

[Dahlbäck 1989]
N. Dahlbäck et A. Jönsson. *Empirical studies of discourse representations for natural language interfaces*, Fourth conference of the european chapter of the ACL, 1989. pp. 291-298.

[Daly-Jones 1994]
O. Daly-Jones. *Characterising the Social Salience of Electronically Mediated Communication*, CHI'94, ACM Conference on Human Factors in Computing Systems, Boston, Massachussets, USA, 1994. pp. 93-94.

[Decouchant 1994]
D. Decouchant. *Rétroaction de groupe et édition coopérative de documents structurés*, IHM'94, Sixièmes Journées sur l'Ingénierie des Interfaces Homme-Machine, Lille, France, 1994. pp. 145-150.

[Dewan 1992]
P. Dewan. *Principles of Designing Multi-User User Interfaces Development Environments*, IFIP TC2/WG2.7 Working Conference on Engineering for Human-Computer Interaction, Ellivuori, Finland, 1992. pp. 35-50.

[Diaper 1989]
D. Diaper. *The Wizard's Apprentice: A Program to Help Analyse Natural Language Dialogues*, 5th Conference of the British Computer Society HCI SIG, 1989.

[Dix 1993]
A. Dix, J. Finlay, G. Abowd et R. Beale. *Human-Computer Interaction*, Prentice Hall, New York, 1993.

[Dorner 1995]
Eudora 1.5.1. Logiciel pour Apple Macintosh. Steve Dorner & Qualcomm, 1995.

[Dourish 1992]
P. Dourish et S. A. Bly. *Portholes: Supporting Awareness in a Distributed Work Group*, CHI'92, ACM Conference on Human Factors in Computing Systems, Monterey, California, USA, 1992. pp. 541-547.

[Dourish 1995]
P. Dourish. *Developing a Reflective Model of Collaborative Systems*, in *ACM Transactions on Computer-Human Interaction*, 2(1), March 1995. pp. 40-63.

[Dowell 1989]
J. Dowell et J. Long. *Towards a conception for an engineering discipline of human factors*, in *Ergonomics*, 32(11), pp. 1513-1535.

[Duke 1995]
D. J. Duke, (editor), A. Aboulafia, A. E. Blandford, S. J. Buckingham-Shum, J. Darzentas, J. May, L. Nigay, J. Ramsay, D. Salber et S. Verjans. *Integration Techniques for Multi-Disciplinary HCI Modelling: A Survey*, Projet européen ESPRIT BRA 7040 AMODEUS 2, RP3 Working Paper (en cours de soumission), ID/WP, 1995.

[Egido 1988]
C. Egido. *Videoconferencing as a Technology to Support Group Work: A Review of its Failures*, CSCW'88, ACM Conference on Computer-Supported Cooperative Work, 1988. pp. 13-24.

[Ellis 1991]
C. A. Ellis, S. J. Gibbs et G. L. Rein. *Groupware: some issues and experiences,* in *Communications of the ACM*, 34(1), janvier 1991. pp. 38-58.

[Ellis 1994] (chapitre 1)
C. Ellis et J. Wainer. *A Conceptual Model of Groupware*, CSCW 94, ACM Conference on Computer Supported Cooperative Work, Chapel Hill, North Carolina, USA, 1994. pp. 79-88.

[Ellis 1994] (chapitre 6)
C. Ellis. *Keepers, Synchronizers, Communicators and Agents*, Workshop on Software architectures for cooperative systems at CSCW'94, ACM Conference on Computer-Supported Cooperative Work, Chapel Hill, North Carolina, USA, 1994.

[Fish 1992]
R. S. Fish, R. E. Kraut, R. W. Root et R. E. Rice. *Evaluating Video as a Technology for Informal Communications*, CHI'92, ACM Conference on Human Factors in Computing Systems, Monterey, California, USA, 1992. pp. 32-48.

[Flores 1988]
F. Flores, M. Graves, B. Hartfield et T. Winograd. *Computer Systems and the Design of Organizational Interaction*, in *ACM Transactions on Office Information Systems*, 6(2), avril 1988. pp. 153-172.

[France 1995]
E. France. *UK Data Protection Registrar*, *Keynote speech*, Ethicomp'95, An International Conference on the Ethical Issues of Using Information Technology, Leicester, UK, 1995.

[Freeman 1993]
S. M. G. Freeman. *An Architecture for Distributed User Interfaces.* Ph.D Thesis, Darwin College, University of Cambridge, 1993.

[Garfinkel 1989]
D. Garfinkel. *The Shared X Multiuser Interface User's Guide*, Hewlett-Packard, Research Report, STL-TM-89-07, 1989.

[Garlan 1993]
D. Garlan et M. Shaw. *An introduction to software architecture,* in *Advances in Software Engineering and Knowledge Engineering*. World Scientific Publishing Company, 1993.

[Gaver 1992] (chapitres 2 et 8)
W. W. Gaver, T. Moran, A. MacLean, L. Lövstrand, P. Dourish, K. Carter et W. Buxton. *Realizing a Video Environment: EuroPARC's RAVE System,* CHI'92, ACM Conference on Human Factors in Computing Systems, Monterey, California, USA, 1992. pp. 27-36.

[Gaver 1992] (chapitre 4)
W. W. Gaver. *The Affordances of Media Spaces for Collaboration*, Proceedings of ACM CSCW'92 Conference on Computer-Supported Cooperative Work, Toronto, Canada, 1992. pp. 17-24.

[Gaver 1993]
W. W. Gaver, A. Sellen, C. Heath et P. Luff. *One is not Enough: Multiple Views in a Media Space*, InterCHI'93, ACM/IFIP Conference on Human Factors in Computing Systems, Amsterdam, Pays-Bas, 1993. pp. 335-341.

[Gaver 1995]
W. W. Gaver, G. Smets et K. Overbeeke. *A Virtual Window on Media Space*, CHI'95, ACM Conference on Human Factors in Computing Systems, Denver, Colorado, USA, 1995. pp. 257-264.

[Gibson 1979]
J. J. Gibson. *The ecological approach to visual perception*, Houghton Mifflin, New York, New York, USA, 1979.

[Godwin 1994]
M. Godwin. *How to make virtual communities work*, in *Wired*, 2(6), juin 1994. pp. 92.

[Gould 1987]
J. D. Gould, S. J. Boies, S. Levy, J. T. Richards et J. Schoonard. *The 1984 Olympic message system: a test of behavioral principles of system design*, in *Communications of the ACM*, 30(9),

[Grudin 1989]
J. Grudin. *Why groupware applications fail: problems in design and evaluation*, in *Office: Technology and People*. Elsevier, 1989. pp. 245.

[Grudin 1994]
J. Grudin. *CSCW: History and Focus*, in *IEEE Computer*, 27(5), mai 1994. pp. 19-26.

[Hammontree 1992]
M. L. Hammontree, J. J. Hendrickson et B. W. Hensley. *Integrated data capture and analysis tools for research and testing on graphical user interfaces*, CHI'92, ACM Conference on Human Factors in Computing Systems, Monterey, California, USA, 1992. pp. 431-432.

[Harper 1992]
R. Harper. *Looking at Ourselves: An Examination of the Social Organization of Two Research Laboratories*, CSCW'92, ACM Conference on Computer-Supported Cooperative Work, Toronto, Canada, 1992. pp. 330-337.

[Harrison 1994]
B. Harrison, M. Mantei, G. Beirne et T. Narine. *Communicating About Communicating: Cross-Disciplinary Design of a Media Space Interface*, CHI'94, ACM Conference on Human Factors in Computing Systems, Boston, Massachussets, USA, 1994. pp. 124-130.

[Heath 1991]
C. Heath et P. Luff. *Collaborative Activity and Technological Design: Task Coordination in the London Underground Control Rooms*, ECSCW'91, European Conference on Computer-Supported Cooperative Work, Amsterdam, Pays-Bas, 1991.

[Heath 1992]
C. Heath et P. Luff. *Media Space and Communicative Asymmetries: Preliminary Observations of Video-Mediated Interaction*, in *Human-Computer Interaction*, 7(3), 1992. pp. 315-246.

[Hill 1992]
R. D. Hill. *The Abstraction-Link-View Paradigm: Using Constraints to Connect User Interfaces to Applications*, CHI'92, ACM Conference on Human Factors in Computing Systems, Monterey, California, USA, 1992. pp. 335-342.

[Hiltz 1978]
S. R. Hiltz et M. Turoff. *The Network Nation*, MIT Press, Cambridge, Massachussets, USA, 1978.

[Hix 1993]
D. Hix et H. R. Hartson. *Developing User Interfaces, Ensuring Usability Through Product & Process,* John Wiley & Sons, New York, New York, 1993.

[Hollan 1992]
J. Hollan et S. Stornetta. *Beyond Being There*, CHI'92, ACM Conference on Human Factors in Computing Systems, Monterey, California, USA, 1992. pp. 119-125.

[Insignia 1994]
SoftWindows 1.0. Logiciel pour Macintosh. Insignia Solutions, 1994.

[Ishii 1992]
H. Ishii et M. Kobayashi. *ClearBoard: A Seamless Medium for Shared Drawing and Conversation with Eye Contact,* CHI'92, ACM Conference on Human Factors in Computing Systems, Monterey, California, USA, 1992. pp. 525-532.

[Jambon 1994]
F. Jambon et L. Karsenty. *Formalisation des interfaces et travail coopératif : quelles conséquences ?*, IHM 94, Sixièmes Journées sur l'Ingénierie des Interfaces Homme-Machine, Lille, France, 1994. pp. 163-168.

[Johnson 1994]
D. G. Johnson. *Computer Ethics,* Prentice Hall, 1994.

[Karsenty 1994]
A. Karsenty. *GroupDesign : un collecticiel synchrone pour l'édition partagée de documents*. Thèse de doctorat, Université d'Orsay Paris-Sud, 1994.

[Kazman 1994]
R. Kazman, L. Bass, G. Abowd et M. Webb. *SAAM: A Method for Analyzing the Properties of Software Architectures*, ICSE-16, Sorrento, Italy, 1994.

[Koegel Buford 1994]
J. F. Koegel Buford. *Middleware System Services Architecture*, in *Multimedia Systems*. J. F. Koegel Buford, (ed.) Addison-Wesley, New York, New York, USA, 1994. pp. 221-244.

[Kosbie 1994]
D. S. Kosbie. *Hierarchical Events in Graphical User Interfaces*, CHI'94, ACM Conference on Human Factors in Computing Systems, Boston, Massachussets, USA, 1994. pp. 131-132.

[Krakowiak 1990]
S. Krakowiak, M. Meysembourg, H. Nguyen Van, M. Riveill et C. Roisin. *Design and Implementation of an object-oriented strongly typed language for distributed applications*, in *Journal of Object-Oriented Programming*, septembre 1990.

[Lewis 1991]
C. Lewis, P. Polson, C. Wharton et J. Rieman. *Testing a Walkthrough Methodology for Theory-Based Design of Walk-Up-and-Use Interfaces*, CHI'91, ACM Conference on Human Factors in Computing Systems, New Orleans, Louisiana, USA, 1991. pp. 235-242.

[Lewis 1994]
P. H. Lewis. *An Ad (Gasp!) in Cyberspace*, in *The New York Times*, 1994(19/4), avril 1994. pp. D1:2.

[Lischetti 1994]
N. Lischetti. *SupraAnalyse : un outil d'aide à l'analyse de l'utilisabilité. Application à un terminal de télécommunication multiservices*. Mémoire CNAM, Laboratoire de Génie Informatique, Université Joseph Fourier Grenoble 1, 1994.

[Long 1989]
J. Long et J. Dowell. *Conceptions of the Discipline of HCI: Craft, Applied Science, and Engineering*, Fifth Conference of the British Computer Society HCI SIG, 1989.

[Lövstrand 1991]
L. Lövstrand. *Being Selectively Aware with the Khronika System*, ECSCW'91, European Conference on Computer Supported Cooperative Work, Amsterdam, Pays-Bas, 1991.

[Mackay 1991]
W. E. Mackay. *Triggers and Barriers to Customizing Software*, Proceedings of ACM CHI'91 Conference on Human Factors in Computing Systems, New Orleans, Louisiana, USA, 1991. pp. 153-160.

[Mackay 1995]
W. E. Mackay. *Ethics, Lies and Videotape…*, CHI'95, ACM Conference on Human Factors in Computing Systems, Denver, Colorado, USA, 1995. pp. 138-145.

[Mackinlay 1991]
J. D. Mackinlay, G. G. Robertson et S. K. Card. *The Perspective Wall: Detail and Context Smoothly Integrated*, CHI'91, ACM Conference on Human Factors in Computing Systems, New Orleans, Louisiana, USA, 1991. pp. 173-179.

[MacLean 1991]
A. MacLean, R. M. Young, V. Bellotti et T. Moran. *Questions, Options and Criteria: Elements of Design Space Analysis,* in *Human-Computer Interaction*, 6(3 & 4), 1991. pp. 201-250.

[MacLeod 1993]
M. MacLeod. *DRUM, Diagnostic Recorder for Usability Measurement,* NPL, DITC HCI Group, Teddington, UK, 1993.

[MacroMedia 1993]
Director 3.1.1. Logiciel pour Apple Macintosh. MacroMedia Inc., 1993.

[Malone 1986]
T. W. Malone, K. R. Grant et F. A. Turbak. *The Information Lens: An Intelligent System for Information Sharing in Organizations*, CHI'86, ACM Conference on Human Factors in Computing Systems, 1986. pp. 1-8.

[Maner 1995]
W. Maner. *Unique Ethical Problems in Information Technology*, Ethicomp'95, An International Conference on the Ethical Issues of Using Information Technology, Leicester, UK, 1995.

[Mantei 1989]
M. Mantei. *Observation of Executives Using a Computer Supported Meeting Environment,* in *Readings in Groupware and Computer-Supported Cooperative Work.* R. M. Baecker, (ed.) Morgan Kaufman, San Mateo, California, US, 1989. pp. 695-708.

[Mantei 1991]
M. Mantei, R. M. Baecker, A. Sellen, W. Buxton, T. Milligan et B. Wellman. *Experiences in the use of a Media Space*, CHI'91, ACM Conference on Human Factors in Computing Systems, New Orleans, Louisiana, USA, 1991. pp. 203-208.

[Mason 1986]
R. O. Mason. *Four Ethical Issues of the Information Age,* in *MIS Quarterly*, 10(1), janvier 1986. pp. 486-498.

[Maulsby 1993]
D. Maulsby, S. Greenberg et R. Mander. *Prototyping an Intelligent Agent through Wizard of Oz*, InterCHI'93, ACM/IFIP Conference on Human Factors in Computing Systems, Amsterdam, Pays-Bas, 1993. pp. 277-284.

[McCall 1977]
J. McCall. *Factors in Software Quality*, General Electric Eds., 1977.

[McCarthy 1994]
J. C. McCarthy et A. F. Monk. *Channels, conversation, cooperation and relevance: all you wanted to know about communication but were afraid to ask*, in *Collaborative Computing*, 1(1), mars 1994. pp. 35-60.

[McGurk 1976]
H. McGurk et J. MacDonald. *Hearing lips and seeing voices*, in *Nature*, numéro(264), 1976. pp. 746-748.

[McQuail 1987]
D. McQuail. *Mass Communication Theory, An Introduction*, SAGE Publications, London, UK, 1987.

[Metral 1994]
M. Metral. *MAXIMS: A Learning Interface Agent For Eudora (A User's Guide to the System)*, MIT Media Lab, Technical report, 1994.

[Meyer 1990]
B. Meyer. *Conception et Programmation par Objets*, InterEditions, Paris, France, 1990.

[Mignot 1993]
C. Mignot, C. Valot et N. Carbonell. *An Experimental Study of Future 'Natural' Multimodal Human-Computer Interaction*, InterCHI'93, ACM/IFIP Conference on Human Factors in Computing Systems, Amsterdam, Pays-Bas, 1993. pp. 67-68.

[Minneman 1991]
S. L. Minneman et S. A. Bly. *Managing a trois: A Study of a Multi-User Drawing Tool in Distributed Design Work*, CHI'91, ACM Conference on Human Factors in Computing Systems, New Orleans, Louisiana, USA, 1991. pp. 217-224.

[Moor 1985]
J. H. Moor. *What is Computer Ethics ?*, in *Metaphilosophy*, 16(4), octobre 1985. pp. 226-275.

[Myst 1993]
Myst 1.0. Logiciel pour Macintosh (CD-ROM). 1993.

[Neumann 1995]
P. G. Neumann. *Computer-related Risks*, Addison-Wesley, 1995.

[Nielsen 1994]
J. Nielsen. *Usability Engineering*, Academic Press, Boston, Massachussets, USA, 1994.

[Nigay 1994] (chapitre 1)
L. Nigay, J. Coutaz et D. Salber. *Initial draft preliminary on-the-fly PAC analysis for the ECOM Interface*, Projet européen ESPRIT BRA 7040 AMODEUS 2, Integration Report, SM/IR 4, 1994.

[Nigay 1994] (chapitres 3, 4, 5, 6 et 7)
L. Nigay. *Conception et réalisation des systèmes interactifs: Application aux Interfaces Multimodales*. Thèse de doctorat, Université Joseph Fourier Grenoble I, 1994.

[Norman 1988]
D. Norman. *The Psychology of Everyday Things*, Basic Books Publishing, 1988.

[Okada 1994]
K.-i. Okada, F. Maeda, Y. Ichikawaa et Y. Matsushita. *Multiparty Videoconferencing at Virtual Social Distance: MAJIC Design*, CSCW'94, ACM Conference on Computer-Supported Cooperative Work, Chapel Hill, North Carolina, USA, 1994. pp. 385-393.

[Orwell 1950]
G. Orwell. *1984*, Gallimard, Paris, France, 1950.

[Ott 1993]
M. Ott, J. P. Lewis et I. Cox. *Teleconferencing Eye Contact Using a Virtual Camera*, InterCHI'93, ACM/IFIP Conference on Human Factors in Computing Systems, Amsterdam, Pays-Bas, 1993. pp. 109-110.

[Ouadou 1994]
K. E. Ouadou. *AMF : un modèle d'architecture multi-Agents Multi-Facettes pour interfaces homme-machine et les outils associés.* Thèse de Doctorat, Ecole Centrale de Lyon, 1994.

[Pagani 1993]
D. Pagani et W. E. Mackay. *Bringing Media Spaces into the Real World*, ECSCW'93, European Conference on Computer-Supported Cooperative Work, Milan, Italie, 1993.

[Patterson 1990]
J. F. Patterson, R. D. Hill, S. L. Rohall et W. S. Meeks. *Rendezvous: An Architecture for Synchronous Multi-User Applications*, CSCW'90, ACM Conference on Computer-Supported Cooperative Work, Los Angeles, California, USA, 1990. pp. 317-328.

[Patterson 1994]
J. F. Patterson. *A Taxonomy of Architectures for Synchronous Groupware Applications*, Workshop on Software architectures for cooperative systems at CSCW'94, ACM Conference on Computer-Supported Cooperative Work, Chapel Hill, North Carolina, USA, 1994.

[Platt 1995]
R. G. Platt et B. Morrison. *Ethical and Social Implications of the Internet*, Ethicomp'95, An International Conference on the Ethical Issues of Using Information Technology, Leicester, UK, 1995.

[Polity 1990]
Y. Polity, J.-M. Francony, R. Palermiti, P. Falzon et S. Kazma. *Recueil de dialogues homme-machine en langue naturelle écrite*, Criss, Les cahiers du Criss n°17, 1990.

[Poltrock 1995]
S. Poltrock et J. Grudin. *Groupware and Workflow: A Survey of Systems and Behavioral Issues*, Tutorial at CHI'95, ACM Conference on Human Factors in Computing Systems, Denver, Colorado, USA, 1995.

[Reinhardt 1993]
A. Reinhardt. *Video Conquers the Desktop,* in *BYTE,* 18(9), septembre 1993. pp. 64-80.

[Richards 1984]
M. Richards et K. Underwood. *How Should People and Computers Speak to Each Other*, Interact'84, IFIP Conference on Human-Computer Interaction, 1984. pp. 268-273.

[Roseman 1992]
M. Roseman et S. Greenberg. *GROUPKIT: A Groupware Toolkit for Building Real-Time Conferencing Applications*, CSCW'92, ACM Conference on Computer-Supported Cooperative Work, Toronto, Canada, 1992. pp. 43-50.

[Rouncefield 1994]
M. Rouncefield, J. A. Hughes, T. Rodden et S. Viller. *Working with "Constant Interruption": CSCW and the Small Office*, CSCW 94, ACM Conference on Computer-Supported Cooperative Work, Chapel Hill, North Carolina, USA, 1994. pp. 275-286.

[Salber 1994] (chapitre 1)
D. Salber et J. Coutaz. *Taxinomie des mécanismes de connexion pour la communication homme-machine-homme*, IHM'94, Sixièmes Journées sur l'Ingénierie des Interfaces Homme-Machine, Lille, France, 1994. pp. 183-189.

[Salber 1994] (chapitre 2)
D. Salber et J. Coutaz. *Fenêtres sur groupes: des MediaSpaces pour collaborer et communiquer*, 3èmes Journées Internationales sur L'Interface des Mondes Réels et Virtuels '94, Montpellier, France, 1994. pp. 309-318.

[Salber 1994] (chapitre 3)
D. Salber, J. Coutaz, L. Nigay, (editors), G. Faconti, F. Paterno', D. Duke et M. Harrison. *The System Modelling Glossary*, Amodeus Project, System Modelling Working Paper, SM/WP 26, 1994.

[Salber 1995]
D. Salber, J. Coutaz, D. Decouchant et M. Riveill. *De l'observabilité et de l'honnêteté : le cas du contrôle d'accès dans la Communication Homme-Homme Médiatisée*, soumis à IHM'95, Conférence sur l'Ingénierie des Interfaces Homme-Machine, Toulouse, France, 1995.

[Savetz 1994]
K. Savetz. *Internet Fax FAQ*, disponible sur Internet, FAQ, 1994.

[Scapin 1990]
D. L. Scapin. *Des critères ergonomiques pour l'évaluation et la conception d'interfaces utilisateur*, XXVIè Congrès de la SELF, Montréal, Canada, 1990.

[Schneiderman 1995]
B. Schneiderman. *The Info Superhighway: For The People*, in *Communications of the ACM*, 38(1), janvier 1995. pp. 162.

[Sellen 1992]
A. J. Sellen. *Speech Patterns in Video-Mediated Conversations*, CHI'92, ACM Conference on Human Factors in Computing Systems, Monterey, California, USA, 1992. pp. 49-59.

[Senach 1990]
B. Senach. *Evaluation ergonomique des interfaces homme-machine : une revue de la littérature*, INRIA, Programme 8, Communication Homme-Machine, n° 1180, 1990.

[Shackel 1991]
B. Shackel et S. Richardson. *Human Factors for Informatics Usability*, Cambridge University Press, 1991.

[Shaw 1995]
M. Shaw et D. Garlan. *Software Architecture. Perspectives on an Emerging Discipline*, Prentice Hall, 1995.

[Smith 1986]
S. L. Smith et J. N. Mosier. *A design evaluation checklist for user-system interface software*, The MITRE Corporation, Bedford, Massachussets, USA, #MTR-9480 EDS_TR_84-358, 1986.

[Smolensky 1992]
P. Smolensky. *IA connexionniste, IA symbolique et cerveau*, in *Introduction aux sciences cognitives*. D. Andler, (ed.) Gallimard, Paris, 1992. pp. 77-106.

[Stefik 1988]
M. Stefik, G. Foster, D. G. Bobrow, K. Kahn, S. Lanning et L. Suchman. *Beyond the Chalkboard: Computer Support for Collaboration and Problem Solving in Meetings*, in *Computer-Supported Cooperative Work: A Book of Readings*. I. Greif, (ed.) Morgan Kaufman, San Mateo, California, 1988. pp. 335-366.

[Stults 1986]
R. Stults. *MediaSpace*, Xerox PARC, Rapport technique, 1986.

[Suchman 1987]
L. Suchman. *Plans and Situated Actions,* Cambridge University Press, Cambridge, 1987.

[Tang 1991]
J. C. Tang et S. L. Minneman. *VideoWhiteboard: Video Shadows to Support Remote Collaboration*, CHI'91, ACM Conference on Human Factors in Computing Systems, New Orleans, Louisiana, USA, 1991. pp. 315-322.

[Tang 1994]
J. C. Tang et M. Rua. *Montage: Providing Teleproximity for Distributed Groups*, CHI'94, ACM Conference on Human Factors in Computing Systems, Boston, Massachussets, USA, 1994. pp. 37-43.

[Thompson 1967]
J. D. Thompson. *Organizations in Action,* McGraw-Hill, New York, New York, USA, 1967.

[Tognazzini 1991]
B. Tognazzini. *On High-Altitude Computing,* in *Apple Directions*, 1991.

[Tokuda 1994]
H. Tokuda. *Operating System Support for Continuous Media Applications*, in *Multimedia Systems*. J. F. Koegel Buford, (ed.) Addison-Wesley, New York, New York, USA, 1994. pp. 201-220.

[Valentin 1993]
A. Valentin, G. Vallery et R. Lucongsang. *L'évaluation ergonomique des logiciels, une démarche itérative de conception,* ANACT, 1993.

[Watabe 1990]
K. Watabe, S. Sakata, K. Maeno, H. Fukuoka et T. Ohmori. *Distributed Multiparty Desktop Conferencing System: MERMAID*, CSCW'90, ACM Conference on Computer Supported Cooperative Work, Los Angeles, California, USA, 1990. pp. 27-38.

[Weber 1947]
M. Weber. *The Theory of Social and Economic Organization,* Oxford University Press, New York, New York, USA, 1947.

[Wylie 1970]
L. Wylie. *Chanzeaux, village d'Anjou,* Gallimard, Paris, France, 1970.

[Young 1994]
R. M. Young et G. D. Abowd. *Multi-perspective Modelling of Interface Design Issues: Undo in a Collaborative Editor*, in *People and Computers IX: Proceedings of HCI'94*. G. Cockton, S. W. Draper et G. R. S. Weir, (eds.). Cambridge University Press, Cambridge, UK, 1994. pp. 249-260.

[Zimmerman 1993]
PGP (Pretty Good Privacy) 2.3ui. Logiciel pour Macintosh. Phil Zimmerman, 1993.

Table des figures

Table des matières

Première Partie
Les apports de sciences non-informatiques

Troisième Partie
Mise en œuvre : science et technologie informatiques