THE TAXONOMY WORKBENCH

A multimedia database system for analysing representational modalities

1. Introduction

During a two-year period, the multimodal systems group at the Centre for Cognitive Science, Roskilde University, has been working on establishing and implementing parts of the research agenda for modality theory. The research agenda for modality theory is the following (Bernsen 1993a):

1. To establish sound conceptual and taxonomic foundations for describing and analysing any particular type of unimodal or multimodal output representation relevant to human-computer interaction (HCI);
2. to create a conceptual framework for describing and analysing interactive computer interfaces;
3. to develop a practical methodology for applying the results of steps (1) and (2) above to the problem of information mapping between work/task domains and human-computer interfaces in information systems design.

Modality theory thus aims to establish the theoretical and methodological basis for addressing the information mapping problem in its general form, i.e.:

> *Given any particular class of task domain information which needs to be exchanged between user and system during task performance, identify the set of input/output modalities which constitute an optimal solution to the representation and exchange of that information.*

An ultimate objective is to use results in building computerised tools for the support of interface design.

Work started by addressing the first part of the research agenda of modality theory, i.e. the development of a taxonomy of output modalities in the media of graphics, sound and touch. A (representational) *modality* is a way of representing information. It was realised from early on that work might benefit from the support of a software tool in which we could represent large numbers of samples of output representations for the purposes of analysing their properties and testing possible taxonomy schemes. This lead to the development of Version 1 of the taxonomy workbench (May and Bernsen 1993, May and Tobin 1993), which was demonstrated at INTERCHI '93 in Amsterdam. Version 1 is a database tool designed to assist research by (a) setting up a common multimedia/multimodal database of example output representations, (b) assisting the description and classification of these examples according to different assumptions about the modalities involved, and (c) enabling thought experiments such as, e.g., the testing of different hypotheses about features of the modalities and their interrelations.

When a robust, intuitively plausible and principled taxonomy of output modalities (Bernsen 1994a,b,c, cf. below) had been established, the workbench in its current configuration had done its job proving the usefulness of software support for modality theory development (Bernsen 1993b, May 1993a,b,c,d). This gave rise to the idea of re-designing the software tool with four objectives in mind: (1) to create a software demonstrator of the taxonomy of output modalities; (2) to use the demonstrator to further explore the functional properties of different output modalities in order to map out which information a particular modality is suited for representing; (3) to support exploration of the information mapping methodology (Bernsen and Bertels 1993, Verjans and Bernsen 1994, Verjans 1994, Bernsen and Verjans 1995); and (4) to move towards turning the taxonomy demonstrator into a support tool for multimodal interface design.

The identification of functional properties of modalities is important to the achievement of the third part of the research agenda of modality theory (i.e. information mapping). We view representational modalities as having two broad kinds of property: declarative properties and functional properties. *Declarative properties* are the properties assigned to a particular modality in order to define or describe what it is. Thus, for instance, linguistic modalities share the property of being syntactic-semantic systems of meaning. *Functional properties* characterise what a certain modality is good or bad, suitable or unsuitable at representing and sometimes under which conditions this is the case. Arbitrary acoustics, for instance, may serve useful alert functions in low-acoustic environments but not in high-acoustic ditto. Successful information mapping must be informed by knowledge of functional properties. Whereas many functional properties can be analytically derived from the declarative properties of modalities, to capture other sets of functional properties, an empirical approach is needed in which different modality samples are analysed to identify their functional characteristics, often in conjunction with scenarios of use.

Based on the considerations just outlined, the original taxonomy workbench has now been completely redesigned. Its main objective still is, of course, to provide a computer-aided platform for analysing different modalities drawn from its large database in order to identify functional properties of unimodal modalities. Thus, the comprehensiveness of the functional properties identified will largely determine the applicability to interface design of modality theory.

The most important differences between the two versions of the system are the following:

1) *change of scope:* from covering a variety of taxonomy ideas to concentrating on our current taxonomy of unimodal output modalities;
2) *change of focus:* from a declarative taxonomy to a combined declarative/functional taxonomy (cf. above);
3) *upgraded functionality:* Version 2 incorporates revised versions of the analysis and classification facilities of Version 1 as well as a much improved search facility;
4) *two systems instead of one:* the system now has two distinct parts, namely a taxonomy workbench and a taxonomy demonstrator. Initially, the workbench and the demonstrator were both implemented on the OMNIS 7 platform (Bernsen, Lu and May 1994). The fact that frequent modification of hypermedia documents is a rather laborious endeavour in OMNIS 7, later made us port the theory demonstrator into MOSAIC.

This paper presents Version 2 of the taxonomy workbench and its intended use in supporting empirical and analytical studies of multimodal representations, preceeded by an overview of the taxonomy of output modalities in Sect. 2. The classification tool is presented in Sect. 3 which covers the interface and its functionality in reasonable procedural detail. The search tool is similarly treated in Sect. 4. Sect. 5 concludes and discusses future work.

2. Overview of the Taxonomy of Output Modalities

The current taxonomy of unimodal output modalities has been generated (Bernsen 1994b,c) from a set of basic properties (Fig.1). *Unimodal modalities* are representational modalities which, when combined together, constitute multimodal representations but which are not themselves multimodal. Unimodal modalities at the *super level* are defined by being either analogue or non-analogue, either arbitrary or non-arbitrary, and either linguistic or non-linguistic. At the *generic level*, unimodal modalities are characterised, in addition, by being either static or dynamic, as well as being physically realised in one of the three media of graphics, acoustics and haptics. Additional properties are needed to distinguish between modalities at the *atomic level.* For instance, the properties of text, discourse, label/keyword, notation, gesture, writing and speech are used to distinguish between different atomic linguistic modalities. In one part of the taxonomy, i.e. analogue graphics, a *sub-atomic level* has been added at which even more fine-grained distinctions are needed for the taxonomy to properly serve its purpose. In this way, unimodal modalities are individually defined through their basic properties.

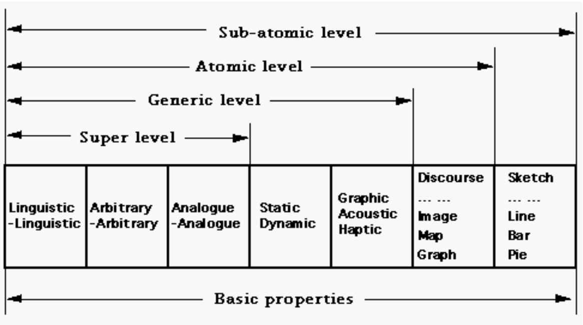


Figure 1. The underlying principles of the taxonomy of unimodal output modalities.

The hierarchical structure of the taxonomy is shown in Fig. 2. The taxonomy tree has 70 nodes, i.e. 4 at the super level, 20 at the generic level and 46 at the atomic level. In the

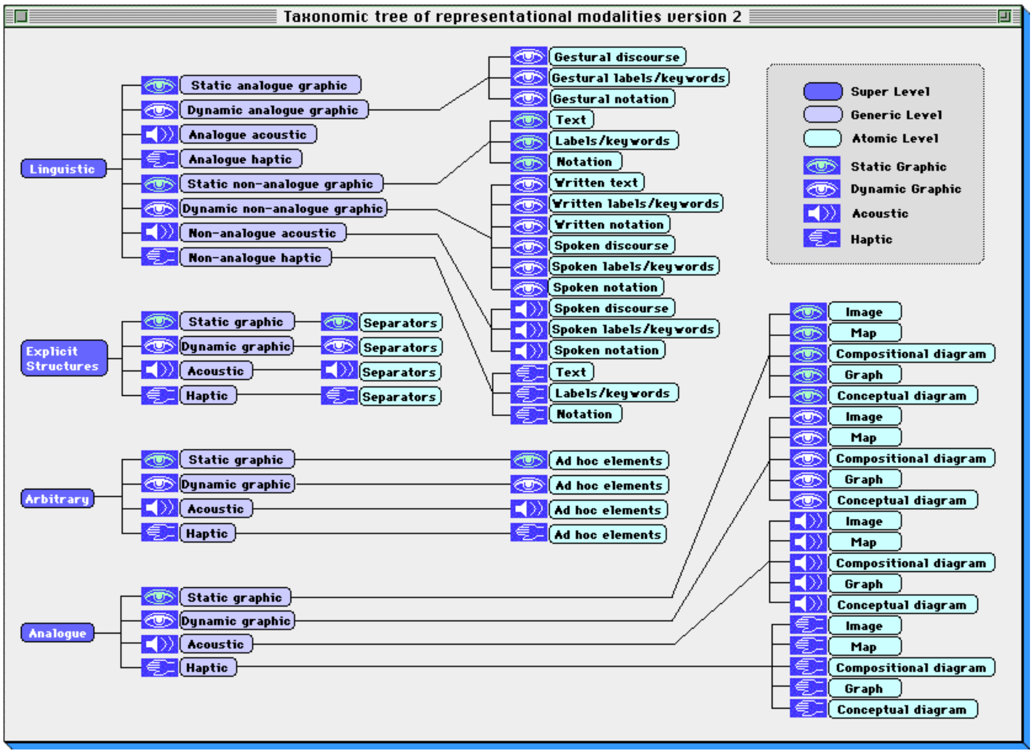


Figure 2. The taxonomy tree (OMNIS 7 implementation).

tree structure, colour, being one of the information channels of graphics, is used to carry differential information. The super, generic and atomic levels are differentiated by their background colours, i.e. blue, blue/grey and light green, respectively. Different media are marked by different analogue icons, i.e. graphics by an eye, acoustics by a loudspeaker and haptics by a hand. Static and dynamic graphics are differentiated through the foreground colours of their icons, i.e. green and white, respectively. Property inheritance links are shown as lines connecting different unimodal modalities from left to right. Via these links, properties are inherited from the super level down to the atomic and sub-atomic levels, the latter of which is not shown in Fig.2. The layout of the tree is mainly determined by spatial constraints. At the top right-hand corner, an explicit structure in a darker shade of grey contains the legend of the taxonomy tree.

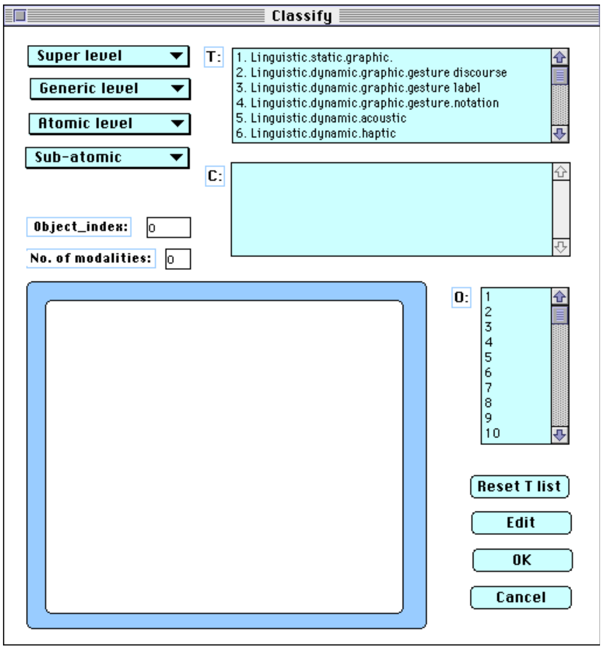
3. Classifying Objects

For the workbench to serve its purpose, all the samples in the database, whether unimodal or multimodal, should be classified in accordance with the taxonomy. A classification tool has been designed for this purpose.

3.1 The Interface

The classification tool interface is shown in Fig. 3. In the top left-hand corner of the window, four pulldown menus are grouped together. They represent classification information at the *Super level, Generic level, Atomic level* and *Sub-Atomic level,* respectively. Items in the pulldown menus are properties which are introduced at that particular level (see Fig. 4). The pulldown menus are used to narrow down choice options.

in the T list. There are three lists: T, C, and O. All 70 unimodal modalities are listed in the *T list.* In the *O list,* all object indices in the database are listed. The *C list* shows all the unimodal modalities which jointly constitute a particular sample. Four push buttons, labeled *Reset T list, Edit, OK,* and *Cancel* are grouped together in the bottom right-hand corner of the window. To the left of the push buttons is an *Object display* window. Above the object display area are two message windows. The *Object_index* message window shows the object index of the object displayed in the object display window. The *No. of modalities* message window shows the



1. Linguistic.static.graphic.
2. Linguistic.dynamic.graphic.gesture discourse
3. Linguistic.dynamic.graphic.gesture label
4. Linguistic.dynamic.graphic.gesture.notation
5. Linguistic.dynamic.acoustic
6. Linguistic.dynamic.haptic

Object_index: 0

No. of modalities: 0

1
2
3
4
5
6
7
8
9
10

Figure 3. The classification tool interface.

number of unimodal modalities which constitute the classified object.

3.2 Functionality

3.2.1 Classifying objects

To classify an object, the following steps are normally followed:

(1) *load the object into the classification window*

When double-clicking on the object-index in the *O list,* the sample will appear in the *Object display window* and the object index number is shown in the *Object_index* message window. If the sample has not been classified, the *C list* will be empty and the *No. of modalities* message window shows 0.

(2) *enter classification mode*

Press the *Edit* push button to enter classification mode.

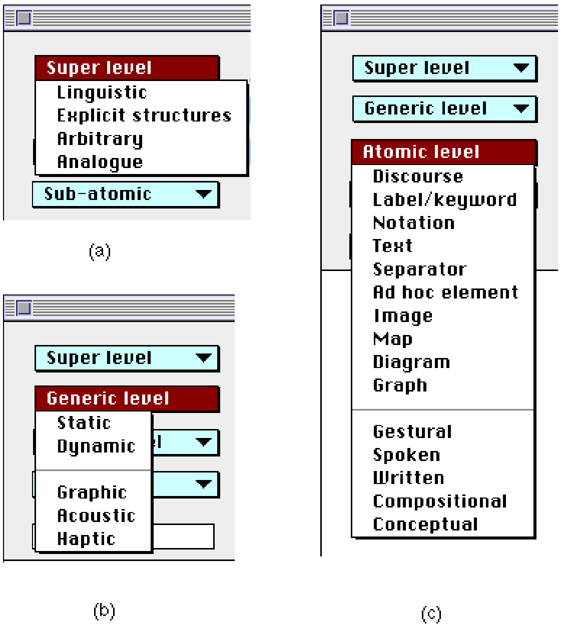
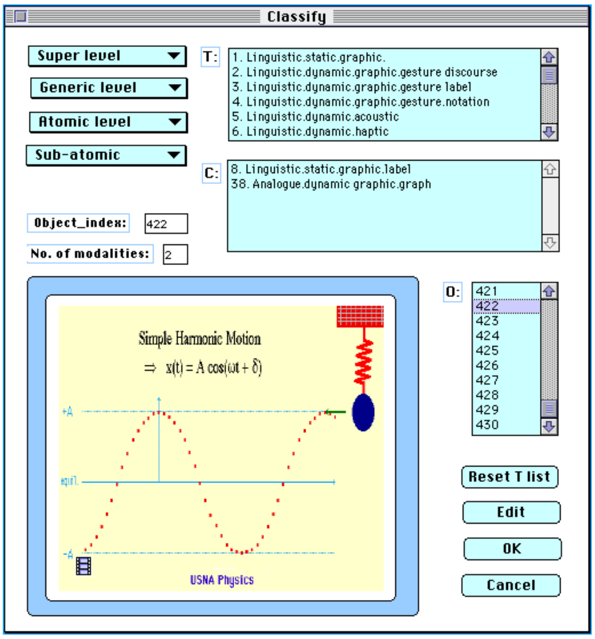


Figure 4. The classification pulldown menus. (a) Super level, (b) Generic level, (c) Atomic level.

(3) *use pulldown menus to narrow down options available in the T list* (optional, see also step (4))

The pulldown menus are designed for naive users who have difficulty in identifying a valid unimodal modality and locating it in the T list, as well as for objects which are difficult to classify. The pulldown menus provide a way of focusing on one property at a time so that eventually, step by step, the appropriate modality will be identified. The menus operate directly on the T list. Initially, the entire set of unimodal modalities is present in the T list. When an item is selected from a menu, the T list is updated such that all and only the unimodal modalities which contain the chosen property are presented.



T:
1. Linguistic.static.graphic.
2. Linguistic.dynamic.graphic.gesture discourse
3. Linguistic.dynamic.graphic.gesture label
4. Linguistic.dynamic.graphic.gesture.notation
5. Linguistic.dynamic.acoustic
6. Linguistic.dynamic.haptic

C:
8. Linguistic.static.graphic.label
38. Analogue.dynamic graphic.graph

Object_index: 422

No. of modalities: 2

O:
421
422
423
424
425
426
427
428
429
430

Figure 5. Object classification exemplified.

Consider, for example, the object numbered 422 in the database (Fig. 5). This object is a video clip imported from a piece of electronic educational material. As it is dynamic, one may inspect how it works by pressing the film-clip icon in the lower left-hand corner. The object is multimodal. To classify it, we focus on one component of the representation at a time. Let us focus on the mathematical formula immediately below the title. An experienced user will easily classify the formula as linguistic.static.graphic.notation and effortlessly locate it in the T list. Doing so, however, may be hard for novice users. Instead, the novice user may first choose *Notation* from the atomic level. As a result, the *T list* will list all and only the notational unimodal modalities as shown in Fig. 6(a). The user may go on to choose *Static* from the generic level menu. In this case, the *T list* will contain all and only the unimodal modalities which are both notational and static as shown in Fig. 6(b). Now it has become much easier to classify this aspect of the multimodal object as linguistic.static.graphic.notation.

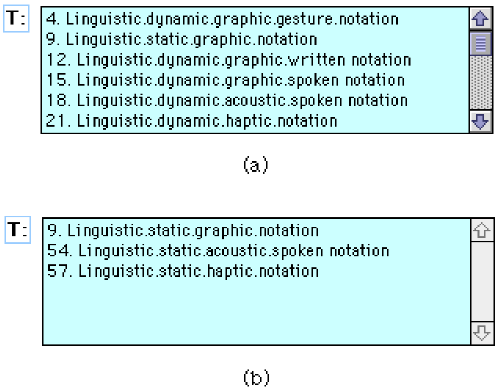
(4) *choose the right unimodal modality from the T list*
Users may either navigate directly in the T list or use pulldown menus to assist in finding the right unimodal modality (cf. Fig. 6). Once the proper unimodal modality has been identified, double-clicking on it in the *T list* will copy it into the *C list.*

(5) *modify classification* (optional)
If an incorrect unimodal modality has been copied into the C list, users can easily undo the result by double-clicking on it.

(6) *save classification*
To save the classification listed in the C list, press the OK push button. In addition to the save action, clicking on the OK button causes two more system actions, i.e., exit of editing mode and reset of the list T to its initial state.

(7) *reset list T* (optional)

When classifying multimodal representations, the pulldown menus can be used iteratively to focus on different unimodal aspects of the same object. Before shifting focus to a novel aspect, the *T list* should be reset to its initial state by clicking on the *Reset T list* push button.



T:
4. Linguistic.dynamic.graphic.gesture.notation
9. Linguistic.static.graphic.notation
12. Linguistic.dynamic.graphic.written notation
15. Linguistic.dynamic.graphic.spoken notation
18. Linguistic.dynamic.acoustic.spoken notation
21. Linguistic.dynamic.haptic.notation

(a)

T:
9. Linguistic.static.graphic.notation
54. Linguistic.static.acoustic.spoken notation
57. Linguistic.static.haptic.notation

(b)

Figure 6. Response of List T to pulldown menu operations. (a) List of notational modalities. (b) List of static and notational modalities.

(8) *abort classification and exit edit mode* (optional)

At any stage of the classification process, the user may abort and exit editing mode by clicking on the *Cancel* push button.

3.2.2 Viewing classification

When double-clicking on the object index in the *O list,* the object will appear in the *Object display window* and the object index number is shown in the *Object_index* message window. If the object has already been classified, the *No. of modality* message window will show the number of its constituent unimodal modalities and the *C list* will present the constituent unimodal modalities.

In the example of Fig. 5, the object index number is 422. The object has been classified as a multimodal representation consisting of 5 unimodal atoms. The atoms are:

linguistic.static.graphic.label
linguistic static.graphic.notation
explicit.structure.static.graphic.separator

analogue.dynamic.graphic.image
analogue.dynamic.graphic.graph

4. Search in the Workbench

Once objects have been classified, search on any number of properties will greatly assist the study of individual unimodal or multimodal representations. The search tool interface was designed to the following requirements:

(1) be as error proof as possible. For example, the keying in of search specifications was discarded because it is considered an error prone input technique which, in addition, puts an extra cognitive load on users. The keying in of search specifications requires users to be familiar with the notation in advance;

(2) be easy to use. Users should not be required to know file structures and connections nor the search syntax;

(3) be flexible. Users may initiate search at any stage in the search specification process.

4.1 What to Search for

(1) *Search by classification* (including atomic level distinctions).

This is to enable users to retrieve relevant objects by selecting any number of combinations of properties in the taxonomy. For instance, a user could search for objects which are *Linguistic* only, or objects which are both *Linguistic* and *Dynamic,* etc.

(2) *Search by negation*.

This is search for objects which lack a particular property.

(3*) Exact match or inclusive search*.

Any classified object or record in the database will be characterised by a set of unimodal modalities. Let X denote the sub-set of unimodal modalities which characterises an object in the database and let Y denote the sub-set which has been specified in a particular search (both X and Y can be either unimodal or multimodal). In *exact match search,* only objects where X=Y will be returned. *Inclusive search* will return all objects where X=Y or X⊃Y (i.e., Y is a sub-set of X).

4.2 The Search Interface

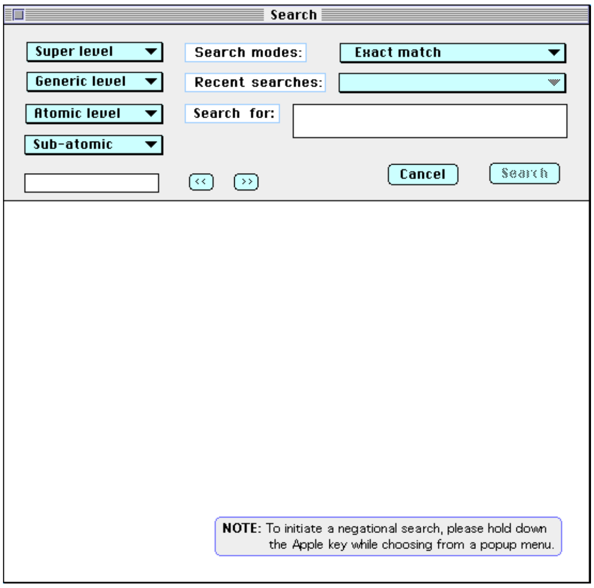
Fig. 7 shows the search window which has two parts. The upper part with grey background is a control area. The lower part is a display area where the search result is represented as an iconic list. In the control area, there is a group of pulldown menus, two pulldown lists, two small message windows and four push buttons.

4.2.1 The pulldown menus

Each pulldown menu lists the properties introduced at that particular level of the taxonomy (cf. Fig. 4). Users specify their search from these menus. Search may be initiated from any of the menus using any number of valid combinations of properties. At the generic and atomic levels, a horizontal line separates the items in two parts, an upper and a lower part. Items from these two parts may be combined in specifying search whereas no items may be combined with other items from the same part of the same menu nor may the items at the super level be combined with each other.

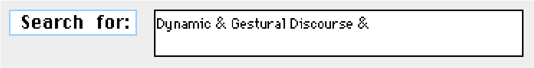
4.2.2 The search specification message window

The small *Search for* message window (Fig. 8) keeps track of the selections a user has made from the pulldown menus.



NOTE: To initiate a negational search, please hold down the Apple key while choosing from a popup menu.

Figure 7. The search interface.



Search for: Dynamic & Gestural Discourse &

Figure 8. The search specification message window.

4.2.3 The cancel and search buttons

On opening the search window, the *Search* button is inactive. This button becomes active as soon as a menu item has been chosen. When the *Search* button is clicked, the specified search is initiated and the search specification is simultaneously stored in the *Recent searches* pulldown list. The *Cancel* button is used to clear the two message windows and the display area.

4.2.4 Search modes

There are two options in the *Search modes* list (Fig. 9), i.e. *Exact match* and *Inclusive.* If *Exact match* is set (by default), only unimodal modality objects are being searched. If the user wants to consider multimodal objects which have some particular unimodal modality as constituent, then *Inclusive* should be set prior to initiating search.

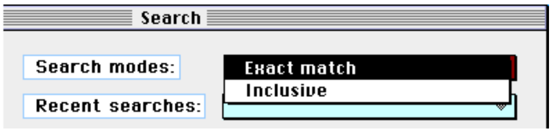


Figure 9. The Search modes pulldown list.

4.2.5 The iconic list

The search result is, firstly, presented in the small message window immediately beneath the pulldown menus. This window presents the number of matched objects. If at least one item was found, the items are, secondly, presented in the display area in the form of small icons which are scaled-down versions of the matched objects (Fig.12). If more than 20 items were found, only the first 20 items are displayed. The advantage of using icons is rather obvious, as they are much more informative than, e.g., a list of object-index numbers.

4.2.6 Recent searches

The *Recent searches* pulldown list (Fig. 10) is a temporary buffer where the 10 most recent search specifications are stored in the order of entry. The most recent specification ranks top in the list. Users can directly initiate a previous search from this list by selecting it. However, if the current search mode is different from the one in which the previous search was initiated, a different result will be produced. The *Recent searches* pulldown list is cleared when the search application is closed.

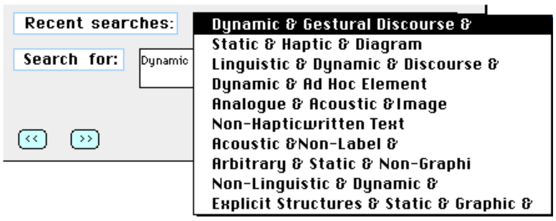

Recent searches:
Search for:
Dynamic
<<
>>
Dynamic & Gestural Discourse &
Static & Haptic & Diagram
Linguistic & Dynamic & Discourse &
Dynamic & Ad Hoc Element
Analogue & Acoustic & Image
Non-Hapticwritten Text
Acoustic &Non-Label &
Arbitrary & Static & Non-Graphi
Non-Linguistic & Dynamic &
Explicit Structures & Static & Graphic &

Figure 10. The Recent searches pulldown list.

4.3 How to Search

4.3.1 Setting search mode

By default, *Exact match* is set on opening the search application. Users can always switch to the desired search mode by directly selecting it from the *Search modes* pulldown list. If only unimodal representation samples are of interest, *Exact match* should be selected. Otherwise, *Inclusive* is selected.

4.3.2 Initiating search specification

Search specifications can only be made through the *levels* pulldown menus. To search for objects having a certain property, users only need to choose that property from the menus. There is no need to follow any specific order nor to specify the number of properties in advance. Users may start specification from the *Super level, Generic level* or *Atomic level* menu and start search at any time. Whatever is selected from these menus, is presented in the *Search for* message window. For example, if a user wants to specify *Linguistic* and *Discourse*, s/he could either first select *Linguistic* from the *Super level* menu and then *Discourse* from the *Atomic level* menu or vice versa.

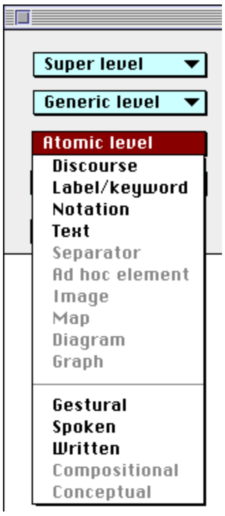


Figure 11. Items incompatible with *Linguistic* are disabled.

The *levels* pulldown menus have a feature which is used to block illegitimate combinations of properties. Initially, all the menu items are active. As soon as an item has been chosen, those items which are incompatible with it are disabled. This may be illustrated by the following example. Having chosen *Linguistic* from the *Super level* menu, all items in the *Atomic level* menu which are incompatible with *Linguistic* are disabled (cf. the grey items in Fig. 11). This mechanism leaves users with only valid combinations to select from.

4.3.3 Specifying negational search (optional)

In addition to search by specified properties, users may search for objects which lack certain properties. To specify a negational search, hold down the *Apple* key while choosing an item from a pulldown menu. Negated properties can be combined with other negated properties as well as with positive properties. Consider the following example. Having selected *Linguistic* and *Static* from the *Super level* and *Generic level* menus, respectively, the user holds down the *Apple* key while selecting *Text* from the *Atomic level* menu. In effect, *Linguistic & Static & Non-Text* has been specified. The implication of the search specification is the following. In *Exact match* mode, all those unimodal objects which are linguistic, static and non-textual will be matched, i.e. those unimodal objects which belong to one of the following categories: the static discourse, label/keyword, and notation modalities. In *Inclusive* mode, all those objects (whether unimodal or multimodal) having a constituent modality that is linguistic, static and non-textual will be matched.

4.3.4 Modifying a search specification (optional)

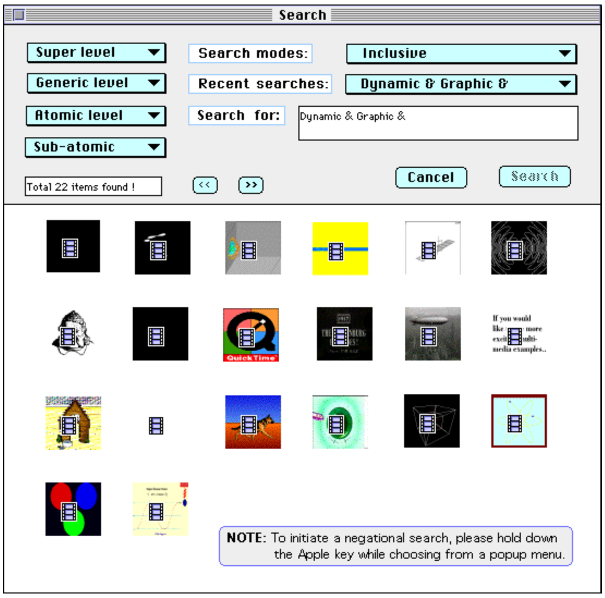
A search specification is normally modifiable provided that the substituting items are not disabled in the pulldown menus. In this case, modification is simply done by re-selecting the intended items from the menus. Re-selection will cancel the previous selection. If the substituting items are disabled, however, the search specification must be cancelled by pressing the *Cancel* push button. Then steps 4.3.2 and/or 4.3.3 above should be repeated.

4.3.5 Initiating search

Once a search has been specified, it can be carried out immediately by clicking the *Search* button. When this button is clicked, four things happen: (a) search begins according to the search specification shown in the *Search for* message window; (b) the search specification is copied to the *Recent searches* message window; (c) the pulldown menus are restored to their initial states; and (d) the result of the search is displayed.

4.3.6 Browsing through the search result list (optional)

Immediately after a search has been executed, the number of items found and the first 20 items (if more than 20 items were found) are iconically displayed (see Fig. 12). If more than 20 items were found, users may use the *Increment* and *Decrement* buttons to browse through the iconic list in steps of 20 items. The *Increment* and *Decrement* buttons are only enabled when an appropriate action can be done through them. Both buttons are disabled before initiating search. The *Increment* button is enabled when search has resulted in more than 20 matched objects**.** Having browsed through to the end of the result list, the *Increment* button will be disabled. The *Decrement* button is disabled when the first item in the result list is being displayed in the display area and otherwise enabled.



Search modes: Inclusive

Recent searches: Dynamic & Graphic &

Search for: Dynamic & Graphic &

Total 22 items found !

NOTE: To initiate a negational search, please hold down the Apple key while choosing from a popup menu.

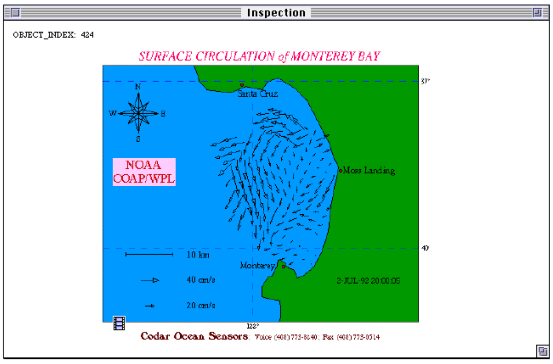
Figure12. An example search result.

4.3.7 Inspecting objects

An object iconically listed in the display area may be inspected in detail by transferring it to the *Inspection* window (see Fig. 13). This is done by mouse-clicking on the item. An object in the *Inspection* window is re-sized to fill the screen so that minute details are clearly visible. Acoustic and haptic objects as well as multimodal objects including acoustics and haptics will similarly be presented on clicking their icons.

4.3.8 Initiating search from the Recent search pulldown list

Any one among the last 10 search specifications can be re-initiated directly from the *Recent search* pulldown list by selecting it. One should be aware, however, that the search mode notified in the *Search modes* pulldown list takes effect.



OBJECT_INDEX: 424

Figure13. A search result displayed in the Inspection window.

5. Discussion and Ongoing Work

Arguably, the taxonomy of output modalities (Bernsen 1994b,c) offers the first taxonomy of all possible output representations across the media of graphics, acoustics and haptics. To the extent that the taxonomy is scientifically stable, we have a much needed conceptual tool for classifying, indexing, storing, searching and retrieving representations for use at the human-computer interface. The workbench described in this paper is a practical implementation of this tool. This conceptual tool would, moreover, be indispensable when it comes to systematically articulating which types of information particular representational modalities are good or bad at representing and hence in the formulation of standards and guidelines concerning multimedia and multimodal systems.

For the time being, however, the taxonomy of output modalities is in need of further analysis and testing to ensure its completeness, orthogonality and robustness. A crucial way of testing the taxonomy is to apply it to large numbers of real-life representations in order to assess its classificatory power as well as the intersubjective robustness of the classifications made. The taxonomy workbench has been designed for this purpose. We will thus be using the workbench to obtain more knowledge about issues such as the following:

- are there unimodal modalities not covered by the taxonomy?
- what are the declarative and functional properties of particular unimodal modalities?
- how well do particular unimodal modalities combine to form multimodal representations?
- which combinations of unimodal modalities are such that the combination enhances the expressive power of each?
- how do experimental subjects classify unimodal and multimodal representations in situations where they are (i) uninformed about the taxonomy, (ii) informed about the taxonomy?

Work on these fronts is ongoing and results are being captured in the theory demonstrator part of our work on computerised tools to support the design of multimedia and multimodal systems.

References

Bernsen, N.O. (1993a): A research agenda for modality theory. In Cox, R., Petre, M., Brna, P. and Lee, J. (Eds.): *Proceedings of the Workshop on Graphical Representations, Reasoning and Communication.* World Conference on Artificial Intelligence in Education, Edinburgh, August, 43-46.

Bernsen, N.O. (1993b): Matching information and interface modalities. An example study. *Esprit Basic Research project GRACE Deliverable* 2.1.1.

Bernsen, N.O. (1994a): Modality theory in support of multimodal interface design. In *Proceedings of the AAAI Spring Symposium on Intelligent Multi-Media Multi-Modal Systems,* Stanford, March, 37-44.

Bernsen, N.O., (1994b): Foundations of multimodal representations. A taxonomy of representational modalities. *Interacting with Computers* Vol. 6 No. 4.

Bernsen, N.O. (1994c): A revised generation of the taxonomy of output modalities. *Esprit Basic Research project AMODEUS-2 Working Paper* RP5-TM-WP11, 1994.

Bernsen, N.O. and Bertels, A. (1993): A methodology for mapping information from task domains to interactive modalities. *Esprit Basic Research project GRACE Deliverable* 10.1.3.

Bernsen, N.O., Lu, S. and May, M. (1994): Towards a design support tool for multimodal interface design. The taxonomy workbench and theory demonstrator. *Esprit Basic Research project AMODEUS-2 Working Paper* RP5-TM-WP5.

Bernsen, N.O. and Verjans, S. (1995): Designing interfaces by information mapping. A case study of the CERD design. *Esprit Basic Research project AMODEUS-2 Working Paper* (forthcoming).

May, M. (1993a): Representations and homomorphisms. A taxonomy of representations for HCI, part 1. *Esprit Basic Research project GRACE Deliverable* 2.1.5.

May, M. (1993b): Levels of representations and mappings. A taxonomy of representations for HCI, part 2. *Esprit Basic Research project GRACE Deliverable* 2.1.6.

May, M. (1993c): From semantic types to multimodal presentations: Case studies. A taxonomy of representations for HCI, part 3. *Esprit Basic Research project GRACE Deliverable* 2.1.7.

May, M. (1993d): Taxonomy and levels of mappings in the construction of multimodal interfaces. *Workshop Notes from AI-ED '93:* World Conference on Artificial Intelligence in Education, Edinburgh, August.*Workshop on Graphical Representations, Reasoning and Communication*.

May, M. (1994): Analogue atomic modalities. *Esprit Basic Research project AMODEUS-2 Working Paper* RP5-TM-WP7.

May, M. and Bernsen, N.O. (1993): The taxonomy workbench. *Esprit Basic Research project GRACE Deliverable* 2.1.8.

May, M. and Tobin, R. (1993): The taxonomy workbench. Purpose, functionality and layout. *Esprit Basic Research project GRACE Deliverable* 2.1.9.

Verjans, S. (1994): EuroCODE: Preliminary information mapping analysis of an AMODEUS common exemplar. *Esprit Basic Research project AMODEUS-2 Internal Report* RP5-TM-IR1.

Verjans, S. and Bernsen, N.O. (1994): PaTerm: A case study in information mapping. *Esprit Basic Research project AMODEUS-2 Working Paper* RP5-TM-WP6.

Acknowledgements. The work reported in this paper was done on grants from the Esprit Basic Research project AMODEUS-2 and the Danish Research Council for the Natural Sciences whose support is gratefully acknowledged.