**Proposal for a list of canonical task types**

*Jean Vanderdonckt, 26 September 2012*

**Aims and motivations**

Tasks involved in a task model are expressed in many different ways, sometimes referring to the same reality and are used for many different purposes (e.g., description, specification, generation, documentation). Providing a list of canonical task types consists of suggesting a series of task types that are expressive enough to cover a significant range of different tasks, while keeping consistency between their expression. The motivations for introducing such a list are the following :

* Provide the task analyst with some guidance on how to name, specify the type of a task in a task model
* Ensure some consistency between different task models, both in specifying and understanding tasks
* Link task types to adequate specifications of abstract user interfaces, and later on, to concrete user interfaces. For instance, some task types could be linked to potential interaction techniques.
* Encourage the reusability of existing definitions
* Facilitate the generation of user interfaces based on predefined task types
* Leave the task type as optional in order to preserve freedom and flexibility

The goal of this proposa lis not to come up with a complete list of canonical task types, but to identify a set of frequently modeled tasks.

**Current definition**

The MBUI - Task Models W3C Working Draft (dated 2 August 2012) contains the following definition :

***System Tasks***

Comparison — the system provides information that is useful for comparing pieces of information

GenerateAlerts — the system generates an alert

Feedback — the system provides feedback about some intensive computational activity

Grouping — the system provides pieces of information that are somehow grouped

Locate — the system provides information useful to locate something

Overview — the system provides an overview of some data

***Interaction Tasks***

Single Selection — the user interacts with the system to select one piece of information

Multiple Selection — the user interacts with the system to select multiple pieces of information

Edit — the user interacts with the system to change some piece of information

Control — the user interacts with the system to trigger some functionality

Zooming — the user interacts with the system to change the level of detail presented, e.g. zooming on a map or photograph

Filtering — the user interacts with the system to filter how some data is visualised, e.g. to show some aspects and to hide others

DetailOnDemand — the user interacts with the system to ask for some detail on some item

***User Tasks***

ProblemSolving — the user performs a cognitive activity to solve a problem

Comparing — the user performs a cognitive activity to compare pieces of information

Planning — the user performs some activity to plan what to do

***Abstract Tasks***

Search Information — the user performs an iterative search for a piece of information

Before proposing an extension of this classification, let us consider the following sources of information related to this question.

1. **Taxonomy of Foley**

Foley et al. (1984) describe a number of interaction tasks and techniques, and controlling tasks and techniques. They define an interaction task as a sentence in a formal language, e.g. the task "move entity" has three basic entities: a position, an entity and the imperative 'move'. The task of the system designer in selecting the most appropriate device consists in designing interaction techniques that minimize the work required of the human in perception, cognition as well as in motor activity. Each interaction task has a set of application requirements, defined by the context of the application in which the task is embedded. For instance, a particular positioning task may require dynamic, continuous feedback by means of a screen cursor. Six types of interaction tasks are defined, which are user-oriented, rather than system- or hardware-oriented :

1. *Select*= the user makes a selection from a set of alternatives;
2. *Position*= the user indicates a position on the interactive display;
3. *Orient*= the user orients an entity in 2-D or 3-D space;
4. *Path*= the user generates a path, which is a series of positions or orientations created over time;
5. *Quantify*= the user specifies a value (i.e., a number) to quantify a measure, such as the height of an entity, or the value, in Ohms, of a resistor;
6. *Text*= the user inputs a text string, used for example as an annotation on a drawing or as part of a page of text. Each of these types has its own requirements. Positioning, for instance, has dimensionality in 1-D, 2-D or 3-D.

None of the six interaction tasks, however, modifies the objects being displayed. If such a modification is needed, the user can achieve it by performing a selection (in particular, a command selection) to invoke such a modification. Foley et al. call such modifying tasks controlling tasks. They distinguish four such types of task:

1. *Stretching* involves grasping a particular feature and moving it to a new position (the result is a distortion);
2. *Sketching* involves manipulating a locating device to create an object by "freehand sketching";
3. *Manipulating* involves causing an object to move about in the viewable space under the control of an input device (manipulation includes dragging, twisting or scaling);
4. *Shaping* finally involves changing of general shape of lines or surfaces.
5. **Taxonomy of Lenorovitz**

Lenorovitz et al. (1984) built a taxonomy of input actions, i.e. of the types of information that can be presented in a user interface. The authors distinguish between four major sub-taxonomies of actions involved in User System Interaction (USI), i.e. taxonomies of terms used to characterize the activities and actions of users in a wide range of user-computer interactive applications. The four sub-taxonomies are (Figure 1):

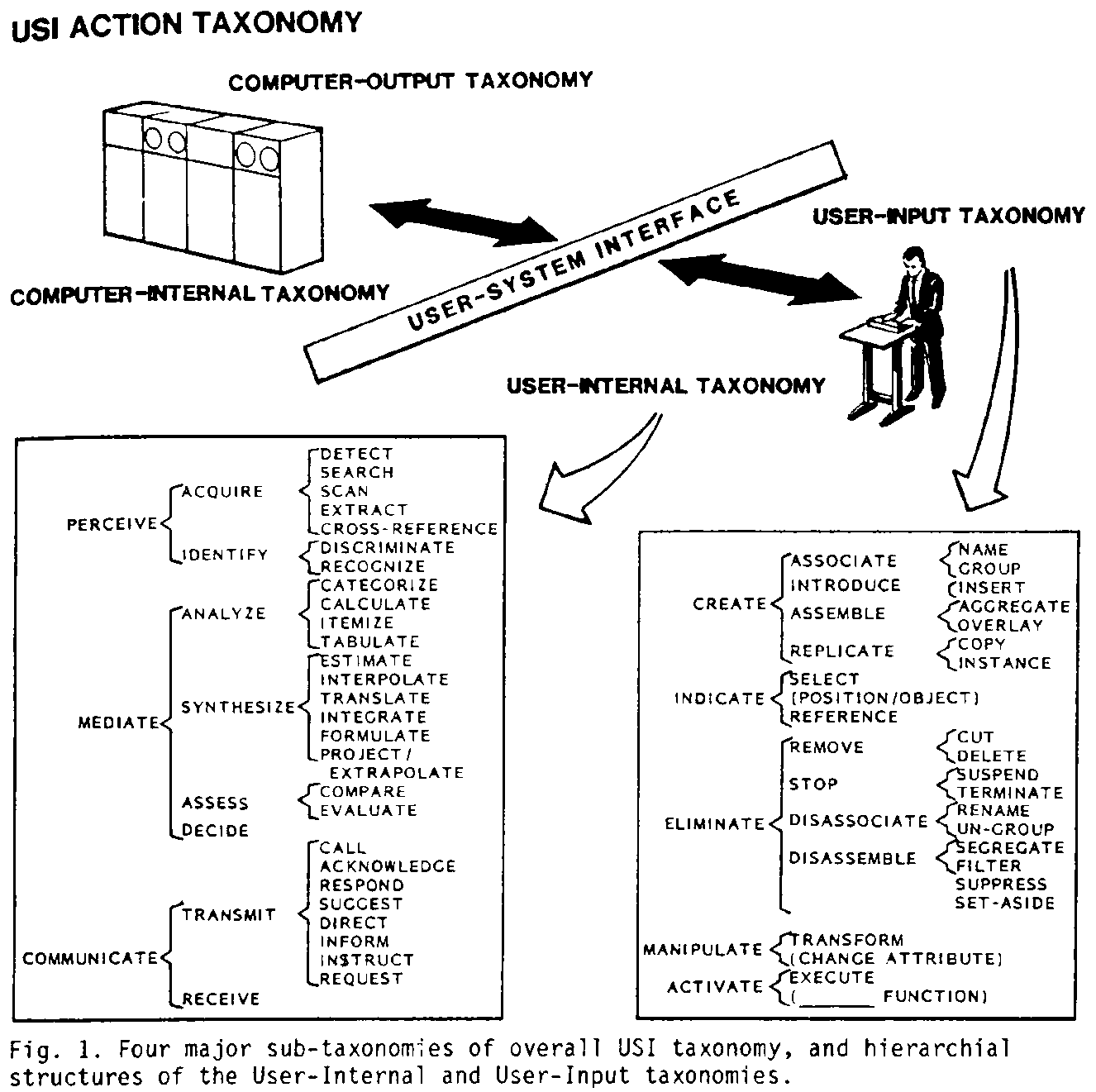
- A *computer-internal taxonomy*, which is concerned with actions which are central to the automated part of the USI, but are totally transparent to the user (e.g., internal computer hardware, operating systems software). This sub-taxonomy deals with issues totally in the domain of Computer Science.

- A *computer-external taxonomy*, which deals with the various ways and means by which a computer can present or display information to the user. This taxonomy is closely tied to the fields of computer graphics and information display technology.

- A *user-internal taxonomy*, which deals with actions central to the user, but transparent to the computer system. The authors have devised a hierarchical system with three major classes of behaviours which users exhibit independently of the computer system: perceive, mediate and communicate. These classes are then further refined. This sub-taxonomy deals with issues of Psychology, and more precisely, Cognitive Science.

- A *User-Input taxonomy*, which directly incorporates many of the terms from Berliner et al. (1964), but with the addition of clear definitions for each term in the taxonomy. The approach is again focused on the conceptual level of interaction, rather than on a more 'physically-oriented' level that focuses on how this interaction should be conducted. The authors try to define a set of exhaustive, yet mutually exclusive goals that users might have in conveying information to a computer system. They term these goals CREATE, INDICATE, ELIMINATE, MANIPULATE and ACTIVATE. These terms were then refined to lower levels of detail, by first identifying sub-goals and then defining generic methods which might be utilized to achieve these goals. Again, the hierarchical terms are just the structural skeleton of the taxonomy.

Computer-output tasks, respectively user-internal tasks, could be interpreted as system tasks, respectively user tasks, while user-input tasks could be interepreted as interactive tasks.

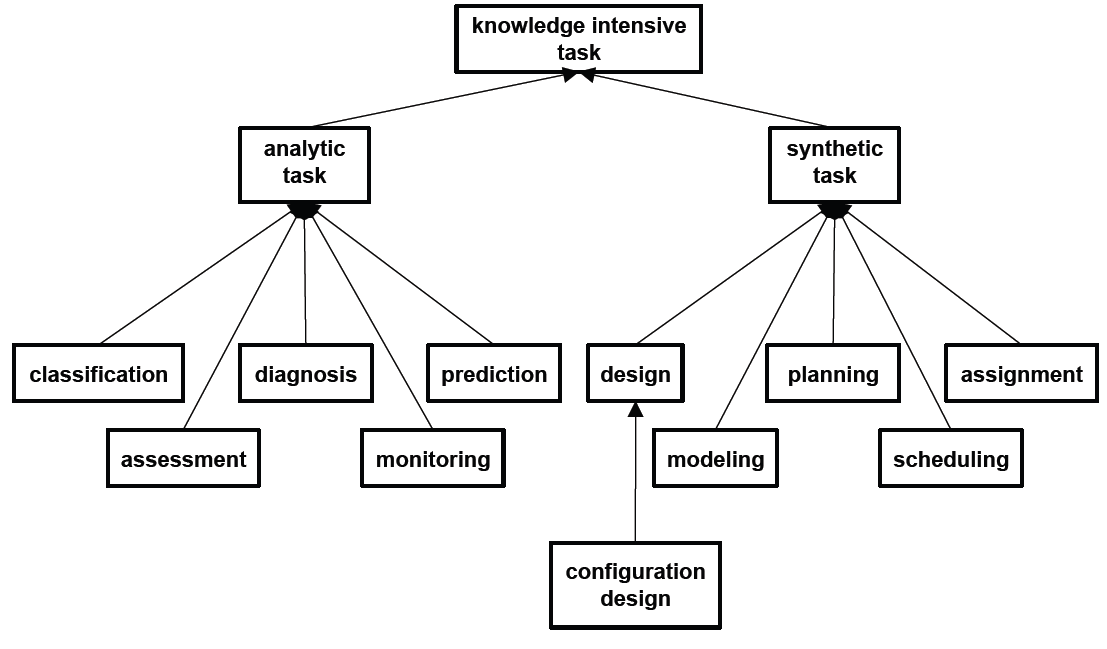


*Figure 1. Taxonomy of User Interface Actions (Lenorovitz et al. 1984)*

1. **The CommonKADS task taxonomy**

CommonKADS [] consists of a comprehensive methodology for developing Knowledge-Based Systems (KBSs) based on knowledge engineering, which consists of constructing different aspect models of human knowledge according to the knowledge-level principle: in knowledge modeling, first concentrate on the conceptual structure of knowledge, and leave the programming

details for later. CommonKADS defines a catalog of task types that are common to KBSs as follows (Figure 2) : a KBS task is either *analytic* if the system pre-exists (or not completely known) or *synthetic* if the system does not exist yet (but could appear later on).



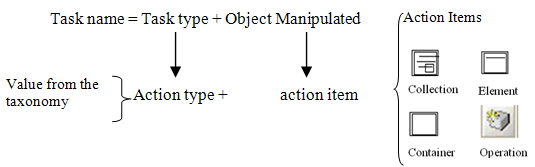
*Figure 2. Taxonomy of CommonKADS task types [].*

1. **List of canonical task types**

Table 1 reproduces a list of so-called task types resulting from a discourse analysis of task types based on a set of task models produced over time by gathering terms (synonyms) under a single category identified by a single task type. This list has been gathered in order to reduce the amount of task types and in order to encourage to name a task in a task model according to a simple sentence scheme that is made of : task name = task type + object manipulated (Figure 3).

|  |  |  |  |
| --- | --- | --- | --- |
| **Task type** | **Task refinements** | **Definition** | **Examples** |
| **Convey** | Communicate, Transmit, Call, Acknowledge, Respond, Answer, Suggest, Direct, Instruct, Request | The action to exchange information | Show details  Switch to summary |
| **Create** | Input, Encode, Enter, Associate, Name, Introduce, Insert, Assemble, Aggregate, Add | Specifies the creation of an item instance | New customer, blank slide |
| **Delete** | Eliminate, Remove, Cut, Ungroup, Disassociate | The action of deleting an item | Break connection, Delete file/slide |
| **Duplicate** | Copy, reproduce | Specifies the copy of an item | copy address, duplicate slide |
| **Filter** | Segregate, Set aside | The action of filtering an item | Filter email, segregate any modification on a data base when backing up |
| **Mediate** | Analyse, Synthesize, Compare, Evaluate, Decide | The action of intercede task items | Compare products characteristics on a online store |
| **Modify** | Change, Alter, Transform, Tuning, Rename, Segregate, Resize, Collapse, Expand | An action of modifying an item | Change shipping address, Tuning volume |
| **Move** | Relocate, Hide, Show, Position, Orient, Path, Travel | the action to change the location of an item | Put into address list, move up/ down? |
| **Navigation** | Go to | the action to find the way through containers | Navigation bar on a web browser |
| **Perceive** | Acquire, Detect, Search, Scan, Extract, Identify, Discriminate, Recognize, Locate, Examine, Monitor, Scan, Detect, | The action of identifying items and/or information from the items | Locate a destination in a map, observe the status bar while installing |
| **Reinitialize** | Wipe out, Clear, Erase | The action of cleaning an item | Clear form |
| **Select** | Pick, Choose | selection between items | group member picker, object selector |
| **Trigger** | Initiate/Start, Play, Search, Activate, Execute, Function, Record, Purchase | Specifies the beginning of an operation | Play audio/video file |
| **Stop** | End, Finish, Exit, Suspend, Complete, Terminate, Cancel | Specifies the end of an action | Stop searching/playing, cancel register |
| **Toggle** | Activate, Deactivate, Switch | The existence of two different states of an item | Bold on/off, encrypted mode, |

*Table 1. List of Canonical User Interface Action task types.*



*Figure 3. Guideline for task type naming*

A same task type is not restricted to a particular task category and could be interpreted in some way depending on the interaction. For instance, table 2, respectively table 3, illustrates different interpretations for a same task type (reinitialize, respectively mediate) depending on the task category. For instance, Reinitialize could be trigerred by the user, the system or both in a mixed-initiative way.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Task Type** | **Task**  **Item** | **Task category** | | |
| **Interactive** | **System** | **User** |
| **Reinitialize** | Collection | All the customer registration elements (name, address) are set to their default values | A system response to restore a task item to its default value. | Make the decision of reinitializing a task item |
| Container | A button that clear a form or restore to the default values |
| Element | Erasing a text field |
| Operation | Pressing a button to restore a variable to its default value |

*Table 2. Reinitialize task type examples*

Table 3 Mediate User Interface actions examples

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Task Type** | **Task**  **Item** | **Task category** | | |
| **Interactive** | **System** | **User** |
| **Mediate** | Collection | Compare products by price | Google search evaluating the best ranked pages to present the results of a query. | Analyse the data details (author, name, publisher, …) of a book |
| Container | Compare side by side documents in word | Decide the layout of a slide when creating a new one | Compare a list of books |
| Element | Evaluate a video watched on YouTube | Evaluate the security risk of a password | Determine the date of a trip |
| Operation | Decide which operation to apply to a combination of CTRL keys. | Propose different arrangement of the results of a query. | Decide which operation will be used with a special key on a joystick |

*Table 3. Mediate task type examples*

New proposal to be discussed

A task is said to be

* *automatic* (A) iff its fulfillment only required computer resources ;
* *manual* (M) iff its fulfillment only requires human resources ;
* *interactive* (I) iff its fulfillment requires both human and computer resources ;
* *mechanical* (ME) iff its fulfillment involves machine resources that are not connected to computer resources (e.g., a machine, a robot, an autonomous agent).

In principle, a task could be initiated from a user, a system, an autonomous agent, or any combination.

*Communicate*: the task to convey information from the user to the system, vice versa or both ways. Related terms : Convey, Transmit, Call, Acknowledge, Respond, Answer, Suggest, Direct, Instruct, Request.

*Create*: the task to specify a new information item. Related terms : Input, Encode, Enter, Associate, Name, Introduce, Insert, Assemble, Aggregate, Add

*Delete* : the task to delete an information item. Related terms : Eliminate, Remove, Cut, Ungroup, Disassociate.

*Duplicate* : the task to reproduce an information item. Related terms : copy, reproduce

*Manipulate* : the task to manipulate one or many information items. Sub-types : transform, change, edit.

*Mediate* : the task to inter-relate various information items. Related terms : Analyse, Synthesize, Compare, Evaluate, Decide.

*Select*: the task to select one or multiple information items from a collection. Sub-types : indicate, position, orient, spacialize.

**To be expanded**

**References**

1. Foley, J. D. & Van Dam, A., Fundamentals of Interactive Computer Graphics, Reading: Addison-Wesley, 1984
2. González-Calleros, J.M., Guerrero-García, J., Vanderdonckt, J., Muñoz-Arteaga, J., Towards Canonical Task Types for User Interface Design, Proc. of Joint 4th Latin American Conference on Human-Computer Interaction-7th Latin American Web Congress LA-Web/CLIHC'2009 (Merida, November 9-11, 2009), E. Chavez, E. Furtado, A. Moran (Eds.), IEEE Computer Society Press, Los Alamitos, 2009, pp. 63-70.
3. Guus Schreiber, Hans Akkermans, Anjo Anjewierden, Robert de Hoog, Nigel Shadbolt, Walter Van de Velde and Bob Wielinga. Knowledge Engineering and Management: The CommonKADS Methodology, MIT Press, New York, 2000.