# Pattern Submission to CHI 2000 Workshop
# “Pattern Languages for Interaction Design: Building Momentum”

Martin Hitz, Inst. f. Informatik & Wirtschaftsinformatik, University of Vienna
Hitz@ACM.org

## Positions

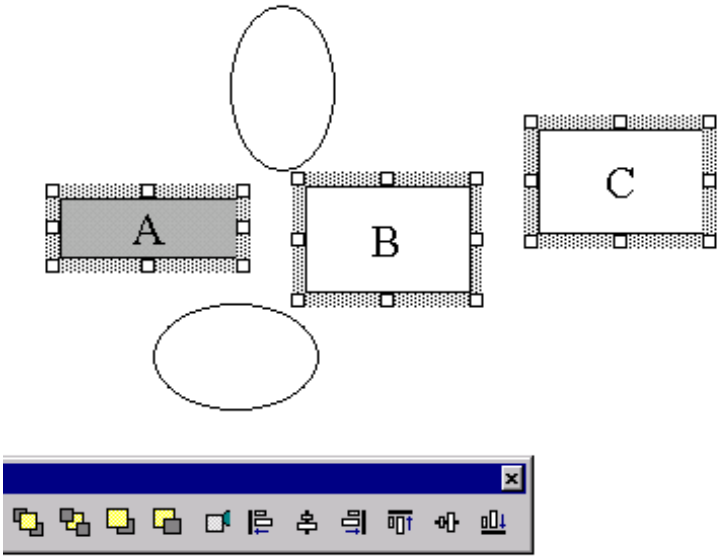
- The two organizing principles as discussed in last year’s Interact ’99 workshop [1] need a lot more elaboration in order to be effectively applied to a given pattern (or set of patterns). For example, it is unclear whether the pattern submitted below falls into the *Dialogue* or *Primitives* level with respect to the first organizing principle defined (BTW: from the fact that it doesn’t seem to fall into the *Component* category one is tempted to conclude that the scale of the first dimension is somewhat flawed...). Similarly, in order to assign a category from the second (process oriented) organizing dimension, the definitions of these levels need to be sounder.
- In addition to the above mentioned principles, I advocate the introduction some notion of *domain* to enhance the usability of a usability pattern language (note that this probably constitutes a usability meta pattern :-). Considering again the example pattern below, it would be very helpful if it could be looked up in some kind of domain index under “graphic editor” or “direct object manipulation”.
- As a software engineer, I would consider a pattern incomplete without some kind of semi-formal description (cf. the section “Schematic” below which caused at least 30 minutes of discussion at Interact ’99...) because it is exactly this level of accuracy in the pattern description where missing details and contradictions become obvious to the describing person.

## Pattern

Pattern Title

**Distinguished Reference Objects (DRO)**

Sensitizing Example




Problem Statement

In many graphic editors, alignment / distribution of a set of selected objects is guided by implicit rules not obvious to the user. Examples:

- The Microsoft Office drawing tool aligns selected objects with the *extreme object* among these. Thus, in the above example, “align bottom” aligns the three rectangles selected with the base line of rectangle B, while “align top” aligns the top lines of A and B with the top line of C.
- Adobe FrameMaker always aligns the selected objects with the *last object* selected. The result of “align top” in the example thus depends on which of the three rectangles was selected last. This approach is more flexible than Microsoft’s, because it allows for the alignment with the top line of A, or B, or C.

In both cases, the regime applied may be counter-intuitive to the novice user (or to users employing both tools). A similar critique applies to the task of distributing objects.

Thus, we need to

- distinguish reference objects (the “anchors” for alignment or distribution) within the set of selected objects (we then have three kinds of objects: not selected, selected, and reference)
- (optionally) introduce an additional selection mechanism to enable the user to mark objects of his/her choice as reference objects

Existing Examples

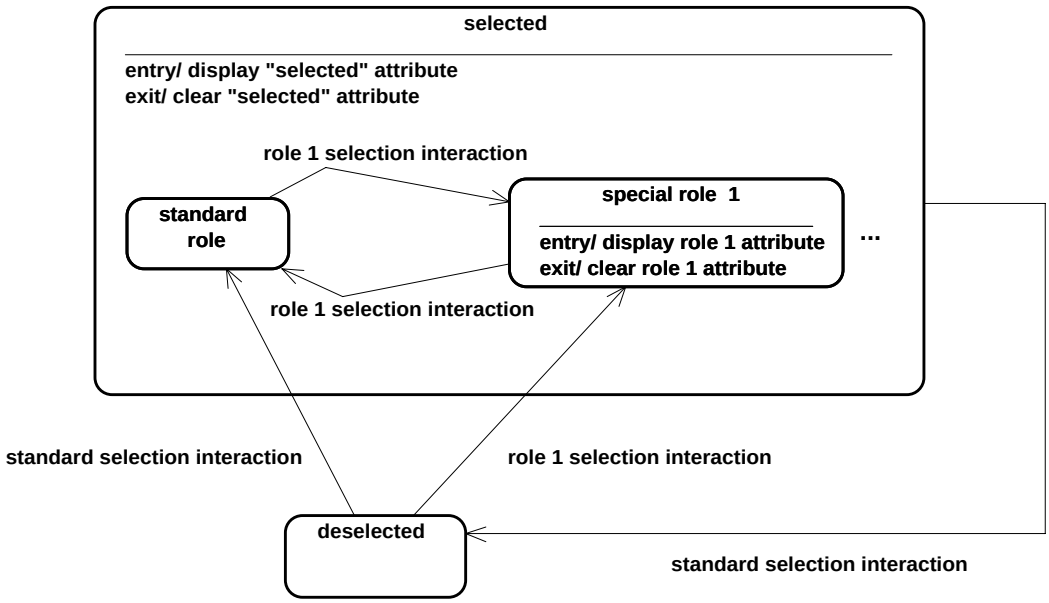
None to the author’s knowledge. However, **DRO** can be regarded as a generalization of the principle to distinguish selected from not selected objects which is applied in virtually all graphic editors. This distinction informs the user that a set of objects will be subjected to the following interaction - **DRO** additionally shows that some objects within the set play a specific role.

Formation of a General Solution

- Mandatory: Assign distinct highlighting attributes for each role in the subsequent operation.
  Example (cf. the picture above): none ⇒ will not take part in the operation, handles ⇒ will be aligned with the reference object, handles + grey background ⇒ reference object
- Optional: Define distinct selection mechanisms for each role in the subsequent operation to enable the user to assign such roles at will.
  Example: While shift key is pressed: left click ⇒ select / deselect object to be moved, right click ⇒ select + define / deselect reference object.

In the absence of the optional part, implicit role assignment rules may be used (e.g., the last object selected becomes the reference object). However, this is less flexible and less general (alignment needs one reference object, distributing needs two reference objects).

Schematic




**Example instantiations**

| | |
|---|---|
| "selected" attribute | handles |
| role 1 attribute | grey background |
| standard selection interaction | left mouse click |
| role 1 selection interaction | right mouse click |

Can be generalized to > 1 role

Reference to Constituent / Related Patterns

**DRO** may be combined with **Description @ Your Fingertips** [2] in order to clarify the semantics of role i.

Categories

Primitives (?)

## References


[1] J. Borchers et al.: Usability Pattern Language: Creating a Community. Internal draft, November 30, 1999, http://borchers.informatik.uni-ulm.de/I99/v2/I99PatternsWS.pdf

[2] R. Griffiths et al.: Example Pattern. Interact '99 Workshop 3: HCI Pattern Language, 1999, http://www.it.bton.ac.uk/staff/rng/UPLworkshop99/Poster.html