

# **BUILD-IT: a brick-based tool for direct interaction**

Morten Fjeld<sup>(1)</sup>, Martin Bichsel<sup>(2)</sup>

<sup>(1)</sup>Institute for Hygiene and Applied Physiology (IHA)

<sup>(2)</sup>Institute for Design and Construction Methods (IKB)

Swiss Federal Institute of Technology (ETH Zurich)

Matthias Rauterberg

Center for Research on User-System Interaction (IPO)

Technical University Eindhoven (TUE)

## *Abstract*

BUILD-IT is a planning tool based on intuitive computer vision technology, supporting complex planning and configuration tasks. Based on real, tangible bricks as an interaction medium, it represents a new approach to Human Computer Interaction (HCI). It allows a group of people, seated around a table, to move virtual objects using a real brick as a interaction handler. Object manipulation and image display take place within the very same interaction space. Together with the image displayed on the table, a perspective view of the situation is projected on a vertical screen. The system offers all kinds of users access to state-of-the-art computing and visualisation, requiring little computer literacy. It offers a new way of interaction, facilitating team-based evaluation of alternative layouts.

## *Keywords*

Direct interaction, graspable interface, computer vision, augmented reality

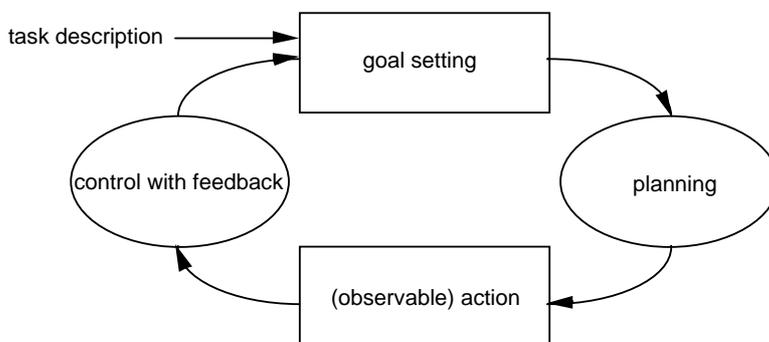
## Introduction

Supporting *natural behaviour* in Human-Computer-Interaction (HCI) is getting increasingly important. We suggest a new concept to *enhance* human expression and to *support* cognitive processes by making them visible.

To allow for a *natural* or *direct* way of task solving behaviour, we define a set of six *design principles*. These principles are then used as support to design an interaction tool called BUILD-IT. Based on tangible bricks as *interaction handlers*, this system enables users to interact with complex data in a direct way. We call our design concept the Natural User Interface (NUI).

## Outline of design principles

As pointed out in the introduction, there is a need for a concept bringing together cognitive (here: goal related) and motor activity. Based on task analysis, *action regulation theory* (Hacker, 1994) is one possible concept to answer this need. We choose action regulation theory as the *psychological basis* for this work. Within this tradition, high importance is given to the concept of *complete task*. A complete task starts with a goal setting part, followed by three subsequent steps (Figure 1).



**Figure 1** A complete activity cycle in action regulation theory.

In more detail, these four steps are:

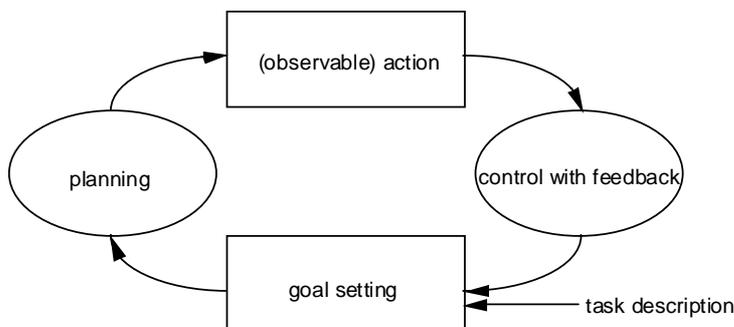
- individual setting of goals, given by the task description, and later on, given by controlled feedback,
- taking on planning functions, selecting tools and preparing actions necessary for goal attainment,

- physical (or even mental) performance functions with feedback on performance pertaining to possible corrections of actions, and
- controlled feedback on results and the possibility of checking the action results against goals.

When computer users pursue an activity, their goal may be more or less clear. Their actions may be classified according to goal-relatedness. Kirsh and Maglio (1994) considered motor activity as being either *epistemic* or *pragmatic*. Pragmatic actions have the primary function of bringing the user physically closer to a goal. In contrast, epistemic actions are chosen to unveil hidden information or to gain insight that otherwise would require much mental computation. Hence, physical actions facilitate mental activity, making it faster and more reliable. Cognitive complexity may also be reduced by epistemic actions.

Epistemic and pragmatic actions are, generally speaking, both present in task-solving behaviour. This applies to all levels of expertise. Independent of the level of expertise, pragmatic and epistemic actions are both necessary for successful task solving performance and should therefore be encouraged in the design of HCI tools.

Pragmatic actions seem to come close to Hacker's (1994) *goal-driven* actions. However, if no goal can be derived directly from a task description, the first part of solving the task is epistemic. In that case, a complete activity cycle starts with observable action, followed by goal setting and planning (Figure 2).



**Figure 2** A complete activity cycle in the case of epistemic action.

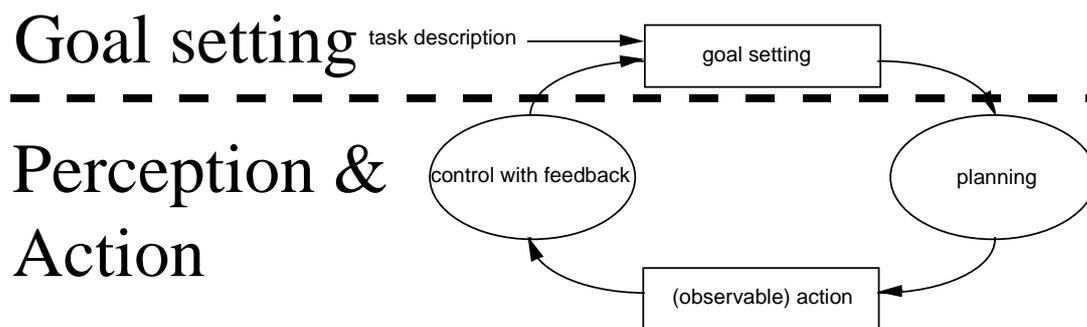
In the rest of this paper, we make the abstraction that pragmatic, as well as epistemic action *both* can be represented by Figure 1. This means that the top and bottom of the cycle in Figure 1 should no longer be taken literally. That figure is meant to transport *both* the idea of complete pragmatic *as well as* the idea of complete epistemic actions.

The first three design principles for graspable interfaces can be outlined:

- Assure that mistakes only imply low risk so that epistemic behaviour is being stimulated,
- allow users to choose between epistemic (exploratory) *and* pragmatic (goal-oriented) actions, and
- support a complete regulation of pragmatic *as well as* epistemic behaviour.

When manipulating objects in the *real world*, action space (hands and fingers) and perception space (the position of the object in the real world) *coincide* in time and space (Rauterberg, 1995). Hacker and Clauss (1976) proved that offering task-relevant information *in the same space* as where action takes place leads to increased performance.

With a screen-keyboard-mouse user interface, there is a *separation* between these two spaces, given by the separation of in- and output devices. An alternative approach to interface design (Rauterberg, 1995), is to let perception and action space coincide (Figure 3).



**Figure 3** User interface where perception and action space coincide.

Furthermore, to improve the feedback from interface to user, it is feasible to offer haptic (or: tactile) feedback. Akamatsu and MacKenzie (1996) showed how tactile feedback may improve task solving performance.

At this point, we merge the two preceding concepts of interface design. Interfaces offering i) a coincident perception and action space, and ii) haptic feedback, can be subsumed under Augmented Reality (AR). AR is based on real objects, augmented by computer-based, intelligent characteristics. AR recognises that people are used to the real world, which cannot be authentically reproduced by a computer. A first AR interface, Digital Desk, was suggested by Newman and Wellner (1992). Similar ideas were described by Tognazzini (1996). We will choose AR to be the *technological basis* for design of NUIs.

We find it important that real and virtual objects clearly indicate the kind of interaction they are meant to support. This idea stems from the concept of *affordances*, first suggested by Gibson (1986), later applied to design by Norman (1988). Applied to our system, this means that real interaction handlers and virtual, projected objects must be designed so that they clearly inform about the function they support, the structure they represent and the results they produce.

Now, the final three design principles for can be established:

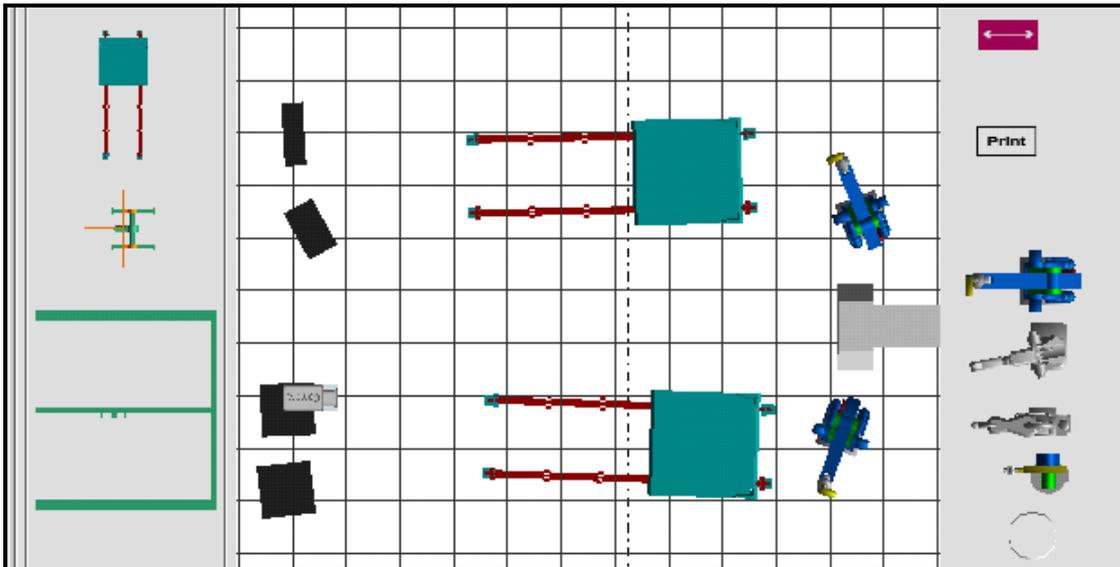
- Support users to take on planning functions in a direct and intuitive way,
- clearly indicate which objects and tools are useful for task solving accomplishment, and
- clearly show the results of user actions.

### *Design and implementation of BUILD-IT*



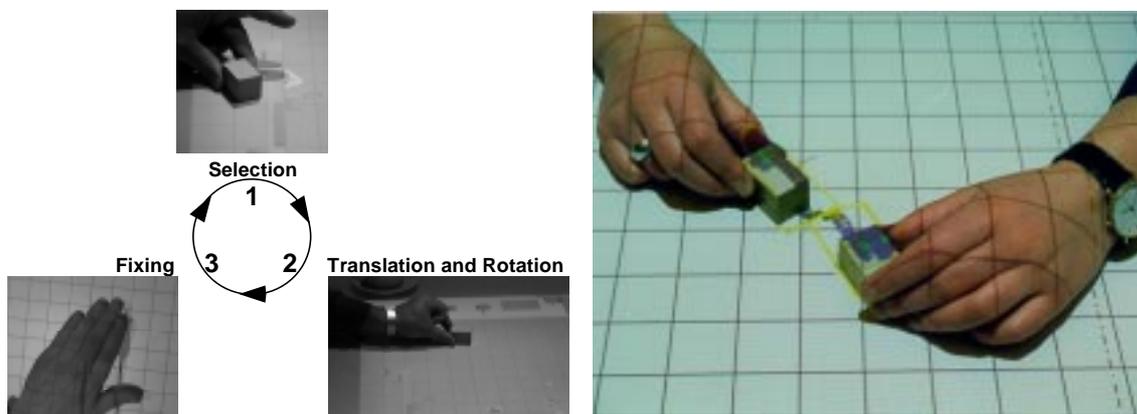
**Figure 4 BUILD-IT, a brick-based NUI-instantiation supporting multi-expert, task solving activity.**

Guided by the outlined principles, we designed a *brick-based* NUI instantiation (Figure 4). *Brick-based* means that graspable bricks are used as interaction handlers, or mediators, between users and virtual objects. As task context, we chose that of planning activities for factory design. A prototype system, called BUILD-IT, was realised (Fjeld, Bichsel and Rauterberg, 1998). This is an application that supports engineers in designing assembly lines and building factories. The system enables users, grouped around a table, to interact in a space of *virtual* and *real* world objects. A vertical screen gives a *side view* of the plant. In the table working area there are *menu areas*, used to select new objects, and a *plan view* where such objects can manipulated (Figure 5).



**Figure 5** In the centre, a *plan view* with objects (robots, tables etc.). On the sides, *menu areas* with objects and functions (virtual camera, print etc.).

The working principle of BUILD-IT is shown in Figure 6. Users select an object by putting the brick at the object positions. Objects can be translated, rotated and de-selected by simple brick manipulation. Using a material brick, everyday motor patterns like grasping, moving, rotating are activated. When the brick is covered, the virtual object stays put.



**Figure 6** The basic steps for brick-based user manipulations (left), and two-handed interaction (right).

To allow for two handed operations, the system supports multi-brick interaction (Figure 6). A second effect of multi-brick interaction, is that several users can take part in a simultaneous design process.

Graphical display is based on the class library MET++ (Ackermann, 1996). The system can read and render arbitrary virtual 3D objects (Figure

5). These objects are transferred from a Computer Aided Design (CAD) system to BUILD-IT (Fjeld, Jourdan, Bichsel and Rauterberg, 1998) using Virtual Reality Modelling Language (VRML).

### *Discussion and future work*

The system dynamically supports user needs for natural behaviour and control. It gives an immediate feedback to support planning and goal setting, assuring a complete regulation of the working cycle. Since the cost of making mistakes is low, the system enhances *epistemic* and *pragmatic* action.

Most novel research questions triggered by this project come from developing and working with the system. However, some of these *questions* can be pursued *apart* from the system. Hence, experiments can be divided into two different kinds; *mock-up* (design of hardware) and *real* (design of software and interaction between hard- and software ).

Future work will explore design of interaction handler(s) and how to combine software and hardware in an intelligible way. Functionality for one- and/or two-handed interaction is another relevant topic. Finally, offering spatial navigation (virtual camera handling, scrolling and zooming), object scaling and grouping, are highly relevant questions.

### *References*

- Ackermann, P. (1996) *Developing Object-Oriented Multimedia Software Based on the MET++ Application Framework*. dpunkt Verlag für digitale Technologie.
- Akamatsu, M. & MacKenzie, I. S. (1996), 'Movement characteristics using a mouse with tactile and force feedback'. *International Journal of Human-Computer Studies*, Vol. 45, pp. 483-493.
- Fjeld, M., Bichsel, M. & Rauterberg, M. (1998), 'BUILD-IT: An Intuitive Design Tool Based on Direct Object Manipulation', in L. Wachsmut & M. Frölich (eds.), *Gesture and Sign Language in Human-Computer Interaction. Lecture Notes in Artificial Intelligence*, Vol. 1371. Springer-Verlag, pp. 297-308.
- Fjeld, M., Jourdan, F., Bichsel, M. & Rauterberg, M. (1998), 'BUILD-IT: an intuitive simulation tool for multi-expert layout processes', in M. Engeli & V. Hrdliczka (eds.), *Fortschritte in der Simulationstechnik*. vdf Hochschulverlag: Zurich, pp 411-418.

- Gibson, J. J. (1986) *The ecological approach to visual perception*. L. Erlbaum: London.
- Hacker, W. & Clauss, A. (1976), 'Kognitive Operationen, inneres Modell und Leistung bei einer Montagetätigkeit', in W. Hacker (ed.) *Psychische Regulation von Arbeitstätigkeiten*. Deutscher Verlag der Wissenschaften: Berlin, pp. 88-102.
- Hacker, W. (1994), 'Action regulation theory and occupational psychology. Review of German empirical research since 1987'. *The German Journal of Psychology*, Vol. 18(2), pp. 91-120.
- Kirsh, D. & Maglio, P. (1994), 'On Distinguishing Epistemic from Pragmatic Action'. *Cognitive Science*, Vol. 18, pp. 513-549.
- Newman, W. & Wellner, P. (1992), 'A Desk Supporting Computer-base Interaction with Paper Documents', in *Proceedings of the CHI'92*, pp. 587-592.
- Norman, D. A. (1988) *The psychology of everyday things*. BasicBooks-HarperCollins, pp. 87-104.
- Rauterberg, M. (1995) *Ueber die Quantifizierung software-ergonomischer Richtlinien, PhD Thesis*. University of Zurich: Zurich, p. 206.
- Tognazzini, B. (1996) *Tog on Software Design*. Addison-Wesley: Reading MA.