ABSTRACT
HabilisDraw is a tool-based drawing environment that contains analogs of physical tools, such as pens, rulers, pushpins, and so forth. The environment is designed to exploit users' intuitions about physical interactions between tools and objects. We are currently porting HabilisDraw to the DiamondTouch in order to explore issues in two-handed tool use.

Keywords
Tool use, drawing, interface design

1. INTRODUCTION
The idea of a workshop or toolbench has been adopted as the basis for many graphical user interfaces. There are important ways, however, that go beyond the virtual/physical distinction, in which interfaces have not adopted the conventions of physical tools and tool-based environments. Our view is that if designers expect users' experience with everyday physical tools to carry over into using software tools, it will be worthwhile to take a close look at the relationship between physical and software tools.

In the HabilisDraw system [2, 4, 5], users produce drawings through persistent tools that encapsulate behavior and information, that can be used in conjunction with one another, and that embody rich cues about their appropriate usage. HabilisDraw tools are designed to leverage the same kinds of activities that people carry out when using physical tools. These software tools have some similarities to past work “local tools” in the KidPad drawing environment [1] and other efforts, but our research has focused on the conceptual foundations of tools and their use rather than design issues.

Our current development effort concentrates on the possibilities of two-handed tool use, based on the DiamondTouch input device from MERL. The demonstration of the system involves users creating drawings using HabilisDraw, wearing gloves in which the contact of the thumb and forefinger of each hand are separately sensed. A short video showing interaction with the system will be continuously shown as well.1

2. HABILISDRAW
HabilisDraw is being developed to take advantage of the inherent ability of humans to use tools through direct manual manipulation. While the design requires a robust simulation of physics and object constraints, the extra work in simulation results in a powerful user interface with a relatively small action set. The physical (i.e. real-world) components of the interface are simple and intuitive, while the simulated components are generalized and behave consistently with the user's expectations. The version of HabilisDraw developed for desktop platforms includes rulers, compasses, pens, ink wells, pushpins, and lenses, some of which are shown in Figure 1.

Our current work extends HabilisDraw to take advantage of two-handed interaction. The hardware components of the system are the DiamondTouch multi-user collaborative input device, an overhead-mounted projector, and two sensor gloves per user. The projector is mounted above the DiamondTouch with a mirror to project the display onto the reflective white surface of the device. Each user puts on one or two gloves, each with a sensor for the thumb and forefinger. While an ideal interface would have sensors for all fingers and possibly even the palm and sides of the hand, hardware limitations restrict us to four inputs per user or two inputs per hand.

The software side of HabilisDraw represents a metaphorical desktop much like the now-standard operating system desktop user interface. However, in this case, the desktop is designed for drawing, rather than file organization and application handling. Tools are persistent, general objects that the user can opportunistically assign a specific purpose and mode of operation.

We are extending the HabilisDraw toolset beyond the tools already mentioned in two ways. Low-level objects are to include paper, string, wire, pins/dowels, and tape. High-level objects, or tool objects, will include pens/pencils, erasers, a paper cutting arm, and rulers. Low-level objects in HabilisDraw are arguably more powerful than pre-classified high-level objects. Using low-level objects, the user can construct compound objects with specialized, repeatable applications. The low-level objects currently under development are as follows:

Paper: Paper is obviously fundamental to the design of a drawing program simulating real-world interactions. Paper in HabilisDraw serves both the standard purpose of providing a marking surface and, as the intuitive counterpart of this purpose, providing a mask

Copyright held by author/owner.

1www.csc.ncsu.edu/faculty/stamant/papers/sg02.avi shows a sample video.
for stenciling. By cutting custom forms of paper with the cutting arm and using tape to combine shapes, one can create stencils and masks for drawing repeated shapes with constrained boundaries.

**String:** Generally, string can be used as a length constraint. Using a measure of string attached to two different objects, the distance between the objects can be limited to the length of the string. A string could be pinned closer than its length at both ends and used to draw an ellipse by nesting a pen in the crook of the string. Its uses are not limited to constraint problems; however, ink could be applied to the string to draw curves and paths.

**Wire:** Wire acts like string in that it can be molded freeform into curves and shapes, except that when molded, wire maintains its shape, providing a rigid length constraint. If two objects must be no more than a certain distance apart, string will constrain them effectively. If they must be exactly a certain distance apart, wire will serve the purpose better. Wire can also be inked to mark a curve or path like string.

**Pins and dowels:** Pins and dowels are constraints that can lock the position of an object while allowing free rotation. A pin will act as just a single point by which a string, wire, or sheet of paper can be fixed but allowed to rotate about the pin. Two pins will prohibit rotation of an object. A dowel acts like a pin by constraining movement, but occupies a two-dimensional space.

**Tape:** Applying tape between any two objects will constrain them to the same relative position and rotation, much like regular tape. Combining pieces of paper with tape allows one to construct concave shapes for stenciling or other masking tasks. Many different behaviors can be modeled simply by joining objects together, such as using a pinned string taped to a stencil to rotate the stencil uniformly around an axis.

### 3. DISCUSSION

While many physical tools can in principle be used with a single hand, the non-dominant hand contributes significantly in practice, essentially by establishing and maintaining context for the dominant hand. Given appropriate input devices in software environments, users tend to exhibit the same kind of asymmetrical two-handed behavior (e.g. [3]). Part of the promise of our work is the identification of conceptual connections between two-handed interface design and models of physical tool use.

In a paper under preparation, we describe fourteen guidelines for building tool-based interactive software, generalizing from observations of physical tool use. Some of these guidelines (omitted for space reasons) deal with two-handed tool use; they are based on the observation that when the non-dominant hand contributes to effective tool use, it is generally in order to stabilize the environment (e.g., to brace the body or to hold materials in place) or to constrain objects to appropriate configurations (e.g., to position an object for viewing or physical access.) In HabilisDraw, for example, the ruler can be moved into place with one hand while the other simultaneously pulls objects to it for alignment, a natural and efficient division of labor.

We believe that for applications in which there are strong physical metaphors, founding two-handed interfaces on observations of the way people use physical tools will lead to more effective interaction.

### 4. ACKNOWLEDGMENTS

This effort was supported by the National Science Foundation under award 0083281 and by a gift of the DiamondTouch system from the Mitsubishi Electric Research Laboratories.

### 5. REFERENCES


