The Antediluvian KVM

Along with the arrival of the first computer came the pursuit of the best way for humans to interface with the machine's logic. A monitor provided a view of what the device was doing, and a keyboard gave us a voice that could reach deep into its circuitry. It wasn't long before Doug Englebart's ingenious mouse gave us a more convenient way to fiddle with what appeared on the screen, and so we arrived at the completed KVM interface—keyboard, video, mouse. And then we got stuck.

Since 1964, we have hung suspended from cables with only an occasional lurching attempt to find an escape. Most innovation has involved trying to cut the cables without changing what is essentially a layer of separation between us and our computers.

The most interesting innovation has struggled along in three areas: voice recognition, touch interfaces, and BCI (brain-computer interfaces).

Voice recognition is getting better as telephonic systems ask us questions in a friendly, usually female, voice. But the limited success of the Dragon NaturallySpeaking speech recognition program (born in 1982) probably has less to do with how well the latest version works—it's really very good—and more to do with people's comfort level with talking to their computers. It looks cool if you're talking to the CPU that is flying your spaceship, but less cool if you're in your cubicle reciting correction cues such as "scratch that" or "spell that" to a monitor.

BCI research is just emerging from the animal research stage. Although there have been amazing human experiments where brain and muscle have been computerized to control a prosthetic device, and a blind man has been given limited monochromatic vision with a computerized implant, most of the benefits from the research are anything but imminent.

That leaves the touch interface. The touch screen has been successful at Home Depot and in some voting booths but not in home computing. The tablet PC has had limited success, but there is more hope with handheld devices. Consider the howling success of Apple's iPod. If you spend more than a few minutes thinking about Apple's music player, the focus shifts away from its storage capacity and conventional menus and synchronizing to the click-wheel navigation (interface). Using a single digit, you turn it on and off, open and close menus, scroll, go back and forth, adjust the volume, and pause/play with just two gestures—push to click or hold down and sweep in a circle to scroll and control volume. It's intuitive, elegant, and, above all, it creates a sense of deft control over the powerful little system.

The next step up for the touch interface might be Apple's recently released iPhone. If you haven't seen the ads, visit www.apple.com. With only one button on its face, a home button, you tap on-screen buttons, scroll with a finger swipe, pinch your index finger and thumb together to zoom in, open them to zoom out, double tap to enlarge, and turn the device sideways for a landscape view of what is normally in portrait format. It's a phone, but it's also a computer running OSX, and your way into its system and processes is your hand—one hand.

If it's successful, this little computer might be a bridge to a future interface. The challenge will be to add commands by expanding the list of intuitive gestures. If that happens, it just might become the hand that can pull us out of our KVM entrapment. ∎