Matching Human Abilities and Limitations:
The Basis for Better Future User Interfaces

David Kieras

University of Michigan

What would make good future user interfaces? Ones that work well with human abilities and limitations.

This approach is not the same as reality-based interfaces - these may or may not help because first, non-computer tasks do not necessarily relate to computer tasks, and second, not all current real-world tasks or situations are well-suited for humans, making them poor models for interfaces.

To elaborate, the reality we evolved for was hunter-gatherer culture. This implies certain powerful abilities, but they may not be manifest in an arbitrarily selected current real task. Manually controlling a glass and steel box moving in a crowd of similar boxes at 70 mph without getting killed is a remarkable skill that most of us can perform pretty well, But this does not mean that it is "natural" and thus a basis for a new interface technology - for example, it is very hard to learn and requires (or should require) great concentration.

Another problem with reality-based interfaces is that the reality of tasks is constantly changing with technology changes. For example, "cut and paste" used to refer to an actual physical activity in document preparation, but not any more. Changing a cursor position on a vertical display by changing the relative position of a small box on a nearby horizontal surface was not a real world activity before, but it is now. Basing future design on the changing nature of "reality" is not obviously a good idea.

A better way to see what could be done better in the future is to examine what made GUIs better at least for time than command language interfaces (CLIs), and then extrapolate some of the lessons learned to the future. The qualities of GUIs have not been fully appreciated; in fact, it would be better to say that GUIs have been misappreciated; it is time for a re-interpretation of the GUI concept.

As a first step, a brief survey of some relevant aspects of human perception, cognition and action will set the stage. One perceptual-motor ability is that visual search is extremely efficient; people are able to use peripherally-visible properties of objects to guide visual search. In addition, humans have good memory for stable visual locations.

Another human ability is the ease of aimed movements to visually identified objects. This has recently been shown to be immune to Hick's law effects, at least as long as the stimuli and

responses are spatially compatible. But even keystrokes are heavily influenced by spatial compatibility factors; classic research shows that in a simple choice reaction task using a row of response buttons, the human first identifies what response to make, then determines where to make it; then the keystroke is made.

Another set of perceptual abilities is the omnidirectionality, parallelizability, and spatiality of the auditory channel. Localized sound can be used to designate spatial locations, but the auditory system does not require a particular orientation to perceive a sound. Sound can be responded to simultaneously with other channels. However, perception of external speech can be interfered with by simultaneous or subvocalized speech and vice-versa.

A familar aspect of human memory often cited in the interface area is that people can recognize information better than they can recall it.

People have several forms of working memory; the best explored is verbal working memory, which is usually represented in speech-based forms. A little-appreciated problem with speech-based interfaces is that speech input or speech production interferes with the use of verbal working memory.

A frequently overlooked, but well-documented form of working memory is task working memory: people have an extremely large and reliable working memory for information in tasks in which they are highly skilled. This means that systems and interfaces in complex task domain have rather different memory support requirements depending on the skill of the user.

A final area concerns human ability to acquire procedural knowledge. Humans have a powerful ability to easily learn procedures as long as they are simple and consistent, and these aspects of learnability can be quantified in terms of procedural knowledge representations. These results lead to a direct characterization of interface consistency: Interfaces that have consistent procedures can be learned much more easily because the learner can simply re-use the shared procedural knowledge. This is the classic common elements theory of transfer of learning in modern theoretical garb.

However, it is known that transfer from previous skills can be weak unless the common elements are quite similar. A common result is that transfer of training fails to appear in many educational contexts - e.g. learning Latin will not help you learn arithmetic. This argues that expecting skills to be transportable from one domain to another is dubious, especially if the relationship is not obvious to the learner.

In spite of this limitation, people can do a good job of learning how to use an unfamiliar system if their knowledge of the system (or "mental model") supports inferring the needed procedures - which is possible only if the interface has a non-arbitrary relationship to the system structure or knowledge domain. For example, we can usually figure out how to operate an unfamiliar digital clock because such devices have similar mechanisms and shared economic constraints on the technology cause them to have controls and displays that interface to these mechanisms in

certain typical ways. At the other extreme, we can usually figure out how to navigate through menus or web pages based on meaning of the words present in the interface and our knowledge of the domain.

The good and bad points of current GUI methodology in these terms can now be briefly presented. First, the good points are:

1. The visual presentation and point-and-click actions on visible objects takes advantage of the great ease and efficiency of human perceptual-motor abilities.

2. Heavy reliance on superior recognition memory.

3. Conventions leading to stable spatial positions of commonly accessed objects, eliminating the need for visual search.

4. Conventions leading to a standard set of menu items for common actions across applications, based on common functions and features, greatly reducing the amount of procedural knowledge to be learned. In contrast, CLIs in practice are inconsistent across applications.

5. Pressure to use a small, simple set of procedures since the interface operations are so limited at the perceptual-motor level, consisting only of various combination of mouse points and button/keystroke operations. Although some GUIs have managed to be otherwise, in general this pressure has resulted in simple and consistent procedures for picking commands or options once the correct display is present. In contrast, CLIs typically display little or no such consistency; each command and its options are typically a special case.

6. Using words and graphical presentations often directly related to the task domain.

The bad aspects of current GUIs in these terms are:

1. An inadequate use of visual guidance in icon search. It is difficult to find objects whose locations are not already known. Most serious is that overall color coding is rendered almost useless by the tendency to use detailed multicolored icons.

2. Serious under-utilization of the auditory channel. Except for an occasional "beep", the auditory channel of almost all computer systems is not used to facilitate task performance.

3. Under-use of speech I/O. Despite repeated attempts, perhaps this is due to incompatibility with verbal working memory or other speech activity.

4.. Extreme featuritis of many current GUI applications. This means that the basic procedural simplicity of the GUI is obscured by arbitrary and difficult-to-learn menu

transversals to access features often arranged without regard to their frequency or utility, or even the domain semantics, that as a result can not support interference of the complex procedures. Another aspect is that instead of recognizing on-screen items as relevant, the user must recall a sequence of menu picks, thus relying on a less powerful form of memory. The overall benefit compared to CLIs is remarkably reduced.

As a side note, traditional CLIs were generally far less usable than they could have been. A CLI could also have relatively simple and consistent procedures. If other aspects of the interface were carefully designed, the ease of learning and use of a CLI would be competitive with that of a GUI.

Thus future interfaces can be considerably improved by taking more advantage of what humans are good at, and relying less on what people have trouble with. In particular, future interfaces should

1. Exploit human visual search and spatial abilities better by re-thinking how the interface objects are presented.

2. Use the auditory and speech channels as parallel and possibility spatial I/O channels, while ensuring that interference with verbal memory is minimized.

3. Use existing analysis and modeling techniques to ensure procedural simplicity and consistency.

4. Ensure that the semantics of the domain can support inference of procedures, through the development of techniques for mapping domain semantics to/from an interface (e.g. a combination of cognitive walkthrough and latent semantic analysis).

5. Support the differing memory requirements of tasks by designing interfaces to either provide intensive memory support for the task state, or not interfere with expert use where it is not needed.