Consider Cognitive Load to Improve Interface Design

Robert Bobkoskie
rbobkoskie3@gatech.edu

1 USER TASKS

An interesting exercise in Human Computer Interaction (HCI) would be to take some time during ones day and reflect on the tasks accomplished over that time. It's not quite as easy as one would expect, as tasks can be defined and interpreted in many ways depending on context and human perception. One must also consider the human factor when thinking about the users direct manipulation of the interface. For example, I'm Visually Impaired (VI), thus, many visually dominant interfaces are literally invisible to me. Additional stimuli, such as haptic or auditory, when added to the interface greatly narrows a VI users gulfs of execution and evaluation. Supplementing the interface with feedback from these channels can achieve metaphorical invisible interface for the VP user.

User Tasks	User Goals	User Interface	Object
Get weather forecast for day	Dress appropriately	Amazon Alexa	Weather forecast
Enter workplace	Successful security scan, elevator access	Reader, Buttons	Scanner, Elevator
Login to computer	Successful login	Keyboard, GUI	Computer
Address email	Prioritize tasks, clear inbox	Mouse, Keyboard, GUI	MS Outlook
Prepare tea	A hot cup of tea	Levers	Faucet, Teapot
Host conference call	Open meeting room	Phone buttons, Mouse, Keyboard, GUI	Cisco WebEx
Write code	Bug free code that satisfies user requirements	Integrated Development Environment (IDE)	Program

Table 1: User Activities Over One Hour

Table 1 describes my tasks and goals, as well as interfaces used to perform the tasks over a 1-hour period of my day. The last column in table 1 defines the object, which is the entity being manipulating or transforming from input into output) from my actions.

One may consider macro-tasks, then subdivided tasks down to a more discreet level. Alternatively, micro-tasks can be consolidated into a single task. For example, start morning routine was sub tasked to [Enter workplace, Login to computer, Address email, Prepare tea]. Conversely, the task Enter Workplace was aggregated from subtasks [Enter Building, Take elevator to office floor].

1.1 Interface Interaction: A Discussion About the Level of Directness and Invisibility

For this topic, I'll choose four of the sets in table 1 [task, goal, interface, object], discuss the level of directness and invisibility of the interaction. The level of directness and invisibility of the interface will be evaluated. The list is presented in descending order of directness, i.e., the first item has the shortest directive distance from the interface to the object and highest invisibility.

- **Prepare tea**: To prepare tea, I fill an electric tea pot and boil water. The interfaces are physical levers on the faucet and teapot. The level of interaction with the task is very high with a short directive distance to the object. The interface is haptic and invisible. I included this task to represent a baseline for a short directive distance and a high level of invisibility.

- **Get weather forecast for day**: The weather condition and forecast for the day determine what I wear. To obtain this information, before even looking out the window, I ask "Alexa, what's the weather?". The response is quick and accurate. The level of directness, or my level of interaction with the task is very high, such that I don't think about the interface. In fact, asking Alexa for the weather would be the same as if I asked a person for the forecast. This audible interface is a strong representation of and behaves muck like the physical object (weather person), thus, the interface is invisible. This high level of direct manipulation, invisible interface, is achieved mostly through great design. As I recall, for my first attempt, I queried "Alexa, What's the weather?", and received the expected output of the current and forecast for the day. The assumption in this analysis is that, to get the output, one needs to know the keyword "Alexa" to access the interface.

- **Login to computer**: Logging into my computer has a greater directive distance relative to the task of obtaining the weather. I need to use a keyboard and mouse to login. I need to consider where I position the mouse and what to type. The interface is visible. My contemplation about this interface has been consistent over time, as has the directive distance.
- **Write code**: This combination of interfaces present the longest directive distance, thus the most visible interfaces. I use a combination of command line and Notepad++ for my IDE. This approach had a large gulf of execution and evaluation upon initial use. Over time, I've learned hotkeys and the bash shell to shorten the directive distance and gulfs of execution and evaluation. However, there are times when I forget commands and need to relearn. There are also new builds that introduce commands that also impose a learning curve.

2 TASK INVISIBILITY THROUGH LEARNING

It's possible for interfaces to become invisible to the user as the user advance along the learning curve for the interface. This is in contrast to an interface having an immediate and profound level of invisibility through great design principles.

One example for an interface that has grown more invisible through learning is setting a programmable thermostat. Older thermostats were analog and were controlled by perhaps two or three physical levers: one to select heat/cool, one for the temperature and perhaps on/off. This interface was very direct and invisible. The high level of directness and invisibility was achieved through basic design and a small feature-set. With the goal of energy efficiency, thermostats were developed to be programmable, thus allowing a user to define temperature settings for any given time and day. For example, a user could set the temperature for the hours when they were not home, vs. when they occupied their residence (see figure 1). Adding new features made the task of setting the temperature more complex, thus increased the directive distance between the user and the task.

Over time, I've learned to use some, but not all of the features. I sometimes go to the subject-matter expert, my wife, or look at the instructions listed on inside cover of the thermostat. Unfortunately, this programmable thermostat, though newer that the analog style is now outdated, making it more challenging to locate user manuals online. This is a general issue with relying on learning, rather than good design to decrease the directive distance.

(a) Analog Thermostat

(b) Programmable Thermostat

Figure 1: Analog Microwave Interface

To shorten the directive distance and narrow the gulfs of execution and evaluation for this interface, some better design principle could be implemented. I will try not to discuss or incorporate principles that are already on the market for programmable interfaces. My approach for a better design to achieve a level of invisibility greater than or equal to my current level would involve two approaches;

- **Improve physical design**: Keep It Simple Stupid (KISS). Minimize interface clutter by eliminating and aggregating button functions. For example, eliminate the heat/cool setting, just use the temperature setting to determine heating/cooling. To set temperature, use a rotary knob with a push to set feature.
- **Incorporate an AI home assistant device**: Implement an audio channel to program the thermostat. This can be accomplished by one of the many AI home assistants available on the market (see the discussion on Alexa in section: Interface Interaction: A Discussion About the Level of Directness and Invisibility). A user can ask "Alexa" to set temperature, thus eliminating the users need to think about what buttons to push. This can lead to a significant and immediate shortening of the directive distance. Over time, Alexa can learn the users behavior and preferences and make suggestions, thus increasing the users interaction with the task and reducing the distance between the user and the interface.

- **Login to computer**: Logging into my computer has a greater directive distance relative to the task of obtaining the weather. I need to use a keyboard and mouse to login. I need to consider where I position the mouse and what to type. The interface is visible. My contemplation about this interface has been consistent over time, as has the directive distance.
- **Write code**: This combination of interfaces present the longest directive distance, thus the most visible interfaces. I use a combination of command line and Notepad++ for my IDE. This approach had a large gulf of execution and evaluation upon initial use. Over time, I've learned hotkeys and the bash shell to shorten the directive distance and gulfs of execution and evaluation. However, there are times when I forget commands and need to relearn. There are also new builds that introduce commands that also impose a learning curve.

3 TASK FEEDBACK AND HUMAN PERCEPTION

Recently, I've had the fortune (not necessarily good) to prepare nightly meals. As I fumble through the process, I find human perception to be vital in procedural learning. Sight, sound and touch provide critical stimuli. For example, a person can see that a hamburger is raw or burnt based on the color of the meat. Haptic information is received by touching a hot frying pan, or the door of a conventional oven. The sizzle or boiling of liquid provide relevant auditory feedback to aide even the novice chef.

Some tools of the trade rely on a single sense to provide feedback, and could benefit from additional modalities. For each perception in the list below, I'll discuss how that sense could be incorporated into some aspect of meal preparation to provide additional feedback.

- **Visual**: This is a challenging sense to identify tasks that could rely more heavily on its implementation. This is the sense that humans are most dependent upon to receive information about the world. Ironically, as a visually impaired person, I don't rely as heavily on this sense and welcome feedback from alternate sources. From my perspective, rather than textual feedback, I'd prefer to "see" more visual feedback in terms of dark/light color, or flashes of light. For example, if I were to leave the oven door open, a flashing light could alert me to that condition. Instead of controls that just click upon engagement, adding visual feedback like an indicator that changes color would provide additional information that the control is engaged/disengaged.

- **Auditory**: Revisiting the open door. Consider working with another person in the kitchen. This person is graciously doing the dishes and accidently leaves the dish washer door fully open. i being, of limited vision, rushes across the room to check on a rapidly progressing grease fire ... when **WHAM** my shin intersects perpendicularly with the open dish washer door (providing an enormous amount of nociception). A life saver would be to have that open door condition emit some auditory information, thus alerting me of this condition. item **Haptic**: Smooth surfaces on many kitchen appliance provide very little haptic information. Many times, these surfaces hide controls and indicators under their surface. For example, touchscreens are as haptic-less as they are ubiquitous. Incorporating some vibration or raised surface would provide a user with haptic information to sense controls and indicators when visual feedback is limited.

Some additional feedback that could be implemented in the task domain of meal preparation would be through the use of **thermoception**[1] feedback. This sense provides the ability to sense heat and cold. Consider the example of detecting heat from a conventional oven door. Unless the door is extremely well insulated, heat can be detected. Realistically, this information is limited in what a human can deduce from the feedback. Basic information that the oven is turned on/off can be obtained. Now consider a microwave oven. This device provides no haptic information through the door. An interesting concept would be to incorporate heat into the microwave door. The heat could vary depending on the temperature of the item being cooked. A quick touch on the door would provide the user with information on the state of the item being heated.

4 REDUCE COGNITIVE LOAD FOR BETTER INTERFACE DESIGN

Interfaces between a user and their task(s) have evolved, sometimes becoming invisible to the user. This can be accomplished by reducing the cognitive load that a human requires to perform a task through the interface. Good design with a

[1] Thermoception

minimal amount of learning by the user is optimal to achieve this objective. Let's consider two methods for designing an interface to reduce cognitive load:

- **Emphasizing essential content while minimizing clutter**: The evolution of multi-media devices has spawned an evolution in remote controls to access these devices. Take a television and its corresponding Remote Control (RC). The complexity of the remote control can confound even the brightest geek squad associate. Eliminating or consolidating these buttons would lessen the gulfs of execution and evaluation for the task of media control. Many features this control provides are unused, thus unnecessary and only increase cognitive load on the user. The designer should question and research user behavior with respect to these RCs and prune out the underutilized operations. For example, my Verizon remote control has two buttons dedicated to turning on the TV. They both seem to turn on the TV, so one could be eliminated. Less is more.

Figure 2: Remote Controls

- **Using multiple modalities**: I find the interface that Microsoft provides for logging into a PC very challenging to operate because it relies exclusively on visual perception. Even worse is that high contrast settings that I require to view such output is not available until after login. Moving the mouse icon into the login fields is challenging and requires my full attention. Aside from a touchscreen, which already exists, a better interface would incorporate auditory and perhaps haptic information relayed to the user through the mouse. Some verbal or sound could be provided to inform the user on the position of the mouse on the screen. In a similar fashion, haptic information through the mouse could offer this feedback.