Brian Cullinan

Assignment 1

Critical Interface Review

Good interfaces:

A microwave:

People have generally taken the easy of using a microwave for granted. But there are many features that it has that allow it to be a great interface as a whole. For example, there is an array of numeric buttons that one can press; there is a start and stop button, and usually some kind of status display. A person can figure out how a particular microwave works almost immediately. When a button is pressed a person expects the microwave to either start immediately or be able to control some arbitrary time that the microwave runs. The screen on the microwave enables to user to read some kind of prompt and enter input that affects the microwaves state. Every microwave has the same kind of labeling and interface; numbers, start, and stop. It is simple yet elegant.

The Wii-mote:

The Wii-mote is used as a pointing devices that itself then interfaces with the Wii console. A Wii-mote is an intuitive interface because it is used just like you would use your hand to point at some object. It immediately becomes an extension of your natural instinct to touch or point in the direction of something you are trying to use. The buttons on the Wii-mote have various uses, but as a whole the Wii-mote can simply be held towards the screen and you immediately get the expected reaction from the console. It can control the position of a pointer finger on the screen for example, which makes sense because we have already established that the Wii-mote is just like pointing at something. Contrary to a mouse which is a relatively bad interface since it can be rotated and used in the opposite direction of the screen and there is no bounds of where the mouse can be pointing to move the pointer on the screen.

A steering wheel:

This interface has both good and bad design aspects. A good aspect is the shape. The wheel shape allows you to place your hands on it at any angle and still be able to control its motion and the motion of the car. This is beneficial because being able to move our arms in the car to different positions prevents the likelihood of arm cramps from being in the same position too long. I steering wheel can also be controlled by a knee, freeing up both hands for other activities, however that method of control is generally not supported by the DMV. This interface, like the Wii-mote, expands on our natural instincts, when we turn the steering wheel in a direction the car also turns in that directions. In order for us to turn the car, we must turn the steering wheel. So the vehicle goes in the same direction as the wheel turns. If we want to make doughnuts with the car, we turn the wheel in a circle. The more the wheel turns, the more the car turns, and the quicker the wheel turns, the quicker the car turns; this comes very natural for us. Contrary to steering wheels, on most video game consoles, turning is controlled by moving a joystick to the right or left. This does not make sense, because moving a stick right or left is linear, yet the camera view is making a circular motion, the correct motion for pushing a joystick right or left is a strafing motion, which is sometimes an option.

Bad Interfaces:

A refrigerator:

The goal of a refrigerator is to keep things cold. This aspect it does rather well. It is even divided into 2 separate sections one section for keeping things below freezing point, and 1 section for keeping things just above freezing point. On many standard refrigerators there isn’t anything that tells the user just how cold things are. However a refrigerator has quite a few short comings. Besides being able to separate items into freezing and not freezing, there is very little organization. Items can be placed anywhere in the refrigerator. If you are lucky there is a holder for eggs, and maybe even a hold for canned beverages. A refrigerator usually consists of a few shelves and a few draws. The problem with this interface is that there is no way of telling exactly what items it contains without jumbling or iterating through all of the items. So finding a single item can take up to O(n2) time, since usually you have to move half the items around to search for an item in the back. To make it worse, in order to find items the refrigerator door must be opened, the user generally has to bend down or is some uncomfortable position to search the lower shelves, and cold air escapes the entire time. So the refrigerator works against itself while someone is accessing it. Unfortunately it has become a standard for refrigerators to function in this manner. A few minor changes could have been made in its original design that would allow for being a better interface. For example a clear material could be used to provide a window to the contents inside, that way find items would let cold air out. Advancements in internal lighting could allow a user to select which item they are looking for from a list and it would light up where the item is sitting, this would allow nearly O(1) constant time searching for items.

A box of Kleenex:

A box of Kleenex is a poor interface to the contained Kleenex tissues. One reason is because it is buggy. There is no static amount of force that has to be applied to pulling each Kleenex out of the box. When usage is initialized it takes more force to extract a Kleenex because the box is full and space is limited. When the contents are nearly empty, it takes much less force to extract a tissue. This brings me to my next point, just like Microsoft windows, it is very buggy. If the wrong amount of force is applied to extracting a tissue, things can go wrong. The tissue will tear which most of the time ruins the usefulness of the tissue. If not enough force is applied, the concurrent tissue will not be pulled through the dispensing hole in the box. This causes even more problems. When a tissue is not protruding through the hole in the job of the box the user is forced to reach into the small hole in the top, and hopefully without tearing the box, extract the next tissue. Finally, the access of multiple tissues at a time is made difficult by this poor interface. Hypothetically if someone’s nose exploded and multiple tissues were needed to remedy the situation, a Kleenex would have to be extracted 1 at a time. This of course takes more time then we would like. So the interface is slow, or the buffer size is too small for our uses as you might say. Multiple Kleenex extraction can only be accomplished by reaching your hand into the box and sorting through layers of Kleenex to pull out all at once, again risking the integrity of the box. There is no uniform action for extracting tissues.

The Windows All Programs list:

This is a bad interface because it makes finding programs difficult. On Windows, there are very limited ways of launching a program, and this is the worst way. On a Mac, the dock can be used to access programs; it can now even list the programs of that type that are open. Windows does not have this feature. Here are some reasons why the All Programs list is a bad interface. At first glance one might think “All Programs”, that must be all the programs installed on my computer. But this is not the case. All Programs contains many different types of items. Including links to webpages, and single files that are not programs. This occurs because programs that are installed are in control of putting items on the All Programs list; sort is not toggle-able. The list is not sorted by default, so every time a program is installed a sort by name must be run from the list. It is annoying to find programs that are installed if the list is not sorted. The list can become large, and there are not display options for it. It will almost always cover up whatever you are currently working on. For example, I cannot watch a movie while I am using the All Programs list.