Operating Systems Interfaces

Leaps and bounds have been made over the short history of modern computing. However, one of the most interesting fields of progression has been in the area of operating systems interfaces. From the most basic punch card batch interfaces to the first announcement of the multi-touch based interface of the first Apple iPhone, user interfaces have been at the frontline of an intense technological arms race. Long ago, user interfaces were crude and there was no real time interaction between the user and the system at all. Most input was handled by punch cards and the wait to complete a single job could span hours. Monitors were eventually introduced and this was a large step in the direction of modern interfaces. After batch interfaces, the command-line interface was born. With command-line interfaces, feedback was nearly real time. Video display was introduced during the command-line era and thus made feedback and user input faster than ever before. Unix, one of the most notable interfaces, was a command-line interface. After numerous advances with command-line interfaces, graphical user interfaces (or GUI’s) exploded onto the scene. Pioneers in GUI development, Xerox PARC, set the standard for the modern GUI. They developed a computer that employed the use of a bit-mapped display and a mouse. This computer, called the Alto, had an interface that put to use icons, windows, scrollbars and other modern GUI concepts. After this, most GUI development involved mostly aesthetic improvements like color, pull-down menus, and pseudo-3D graphics. Companies like Apple and Microsoft pushed GUI advancement even further in the visual department. In 2007, Apple announced its iPhone that featured a multi-touch interface and virtual keyboard. Even though it wasn’t the first device to fully support a multi-touch interface, it popularized the idea of a touchscreen. So many amazing advancements have been made in the brief timeline of user interfaces. It’s definitely exciting and a little perplexing to think about what the next 60 years could hold.

What makes a quality, useful interface? In my opinion, besides the obvious things like efficiency and a quick response, it is most important that UI’s balance themselves between two different spectrums. The first spectrum is between clarity and concision. A UI should balance itself between these two concepts as effectively as possible. A quality UI should present everything to the user in a clear manner. Ambiguity leads to confusion on the user’s end. This allows the user to make good decisions. Organization and specific language are important to clarity. The trouble with clarity is not to overdo it. This is where concision is important. A concise UI doesn’t present too many things to the user at once so that things aren’t hard to find and the UI isn’t cluttered. If a UI can have the perfect amount of clarity and the right amount of concision, it enhances the overall user experience by making things easy to understand and easy to find. The second spectrum involves aesthetics. An interface needs to look good. A good-looking interface enhances the user experience by making the time a user spends on the system more pleasurable. However, once again, it is important not to overdo it. Many UI’s follow the Principle of Least Astonishment or POLA. POLA holds that users can only truly hold their attention to one item at a time. So, it is important not to confuse the user with too many things happening at once. Therefore, a good UI needs to be simple yet appealing to look at.

Many UI’s have distinct pros and cons involved with their use. For instance, command-line UI’s provide great control over the file system and operating system. However, they can be a nightmare for users as the commands they employ can be difficult to memorize. Command-line UI’s are often very resource conservative, but they aren’t very useful for multitasking. GUI’s tend to be very easy for users to learn, even with all of their features, because they very strong in the visual department. GUI’s also provide great control over file systems, although the command-line is still sometimes needed for more complex tasks. GUI’s also make multitasking a breeze with resizable, movable windows. GUI’s tend to consume more resources than other UI’s as well. Multi-touch UI’s offer a very natural way for the user to interact with the system. However, this causes them to be dirtied by fingerprints which can affect the user’s experience. Multi-touch UI’s are also great for fast operations. Pointing to an item, zooming, and drawing are all very fast on multi-touch UI’s. Multi-touch UI’s are more costly though because the technology is in its earlier stages.

From paper tape to high definition touchscreens, there have been a plethora of changes over the evolutionary life of the user interfaces. Each new UI offers its own set of pros and cons along with its own history and influence to the computing industry. User interfaces are one particular piece of an ever evolving universe of technology. They continue to surpass boundaries and amaze us all while providing the most enjoyable user experience possible. UI’s really are one of the most interesting parts of computing history. It’s hard to predict what the future will bring with UI’s in the coming years.

# Sources Used

Computer Hope. *Command line vs. GUI*. 2014. 1 September 2014. link: http://www.computerhope.com/issues/ch000619.htm

Fadeyev, Dmitry. *8 Characteristics of Successful User Interfaces - The Usability Post*. 15 April 2009. Article. 1 September 2014. link: http://usabilitypost.com/2009/04/15/8-characteristics-of-successful-user-interfaces/

Landley, Rob W. *The Art of Unix Usability*. Pearson Education Inc., 2004. Website. link: http://www.catb.org/~esr/writings/taouu/html/

*Principle of least astonishment - Wikipedia, the free encyclopedia*. 15 July 2014. Article. 1 September 2014. link: http://en.wikipedia.org/wiki/Principle\_of\_least\_astonishment

*User interface - Wikipedia, the free encyclopedia*. 18 August 2014. Article. 30 August 2014. link: http://en.wikipedia.org/wiki/User\_interface#Batch\_interface.2C\_1945.E2.80.931968