

A New Spin to Software Platform Design

2008

2 min read • 488 words

I wrote this article two years ago, before I found OS X.

The irony is that OS X already embodied most of the principles I was advocating for here. Sometimes the future we envision already exists—we just haven't discovered it yet.

As I've said before, I find many reasons to believe that modern commercialized software platforms are severely lacking in many, many areas. This should not come as a surprise to anyone. Perhaps basic utility-inclusion is not the only solution though. Perhaps the basic priority structure and ethics that development and marketing teams utilize should be forced into question.

Essentially, most major computer software corporations, are, all in all, trying to make money. No matter how hard you try to find a way around this, or justify why these companies try to do the things that they do, the only answer is money. These companies are simply trying to make a quick buck. This concept has worked incredibly well for years, but we seem to have a bit of a problem with foresight. After a while, people get rather bored with the same old concepts being presented to them in new and exciting ways. This is why Microsoft needs to release a new operating system every once in a while. Microsoft's current problem is that the masses are beginning to realize their other options.

So here's my proposal for the long-term design and marketing strategy 2.0:

An operating system should first be a place of power, consistency, stability, scalability, and flexibility. Included would be a robust and fully scriptable toolset which can be manipulated and presented both graphically and statistically.

Second, the user interface should be very well thought out and planned, with ample room for improvement down the road. Its purpose should first be a place of usability, workflow, and creativity. Task-related workflow and presentation customization, accessible to all types of users, is crucial to the success of the UI. Second, the User Interface should be a mode of personal expression and aesthetic preference. This should never take precedence over the overall stability, usability, or general usefulness of a desktop system, for any given reason.

Lastly, the user application platform system needs to be designed. A centralized repository of applications is an incredibly efficient method for application distribution

Written in 2008, this predicted the App Store revolution that would transform software distribution. Apple launched the iOS App Store later that same year, validating this vision of centralized, curated software repositories.

. This repository would be a dynamic, centralized database of application software and packages that are intended for different groups of people. Most major Linux distributions use this heavily, as well as Apple for it's iPod and iPhone applications, and it has been proven to work well.

Any of these rules should have the ability to be broken easily by advanced power users for technical reasons/needs. This should be in no way advertised or demonstrated.

Anyone up for the challenge?