florentcrivello.com /index.php/2020/01/06/how-apple-sells-software-at-hardware-prices/

How Apple Sells Software At Hardware Prices

3-4 minutes

I ran this poll the other day on Twitter:

The answers highlight a point I often make: that when people buy an iPhone for $1,100, they're really paying $600 for a phone, and $500 for iOS, making it the most expensive consumer software in the world.

This also means Apple's business is more software than hardware. Of course, if you ask them, the company will reject the dichotomy. "We want to make great products," they'll say in a calm British voice, "which entails integrating software and hardware."

But good hardware isn't a moat, and it's been a long time since the hardware of Android phones has become better than that of iPhones. Fanboys will retort that that's because Apple wants to make sure everything is absolutely perfect before shipping, and that they sell a ton of phones, a scale which comes with significant supply chain challenges.

But Samsung sells almost twice as many phones as Apple, and they were still years ahead shipping "phablets," edge to edge displays, and multiple cameras (which pictures are still immensely better than the last iPhones').

I'm sure most iPhone customers would love to get these hardware innovations, and they'd probably switch to Android, if it wasn't for, well, Android.

The tech industry has cooled down a lot on hardware businesses over the last few years, which take a lot more time and money to get off the ground, don't scale as well, and, worst of all, have no moat. If you design a good hardware product, you'll just get cloned into oblivion by Shenzhen, which will move faster and sell their product for cheaper. Adding insult to the injury, the theft will often come from your own Chinese suppliers — and you'll have no recourse against them.

But Apple seems to have found a way to get software-like margins in a hardware business: by packaging both together. Nobody would ever buy an *OS* for $500 — but psychologically, buying a *phone* for $1,100 feels different.

But of course, throwing code into something doesn't magically give it 70% margins. No software can give a moat it doesn't have to a hardware business. In the case of iOS, the moats are the App Store's network effects (developers build for the platform with users, who adopt the platform with developers, who…), and the costs of switching to Android.

(There's a case to be made that Apple actually also has a moat on hardware, thanks to its brand. That's how they make so much money on the Apple Watch and the AirPods, both of which are now bigger businesses than the iPod ever was — despite their having no meaningful software ecosystem.)

In short, the way to give software-like margins to a hardware business — or, conversely, to sell software at hardware-like prices — is to:

1. Build software with a moat (I didn't say this would be easy)

2. Bundle it with hardware

3. Leverage your hardware's new moat to price it higher

4. Profit