URL: https://stvp.stanford.edu/blog/videos/the-importance-of-establishing-critical-mass

Fleury talks about establishing and maintaining a critical mass within the market. Without it, he explains, you will find yourself in a very difficult situation and VCs will view you differently.

Transcript

Before you get money in open source, an important thing is to have already a critical mass established.. When we had that with number one development, number one user end, number one embedded -- that is the power of open source.. Once you're successful, people take your stuff in places and in ways that you haven't imagined and applied for example here at Stanford, I know that a team did research on highly available system that was the cover of Scientific American about a year ago.. You know, so there're always people taking your stuff and that gives you a lot of penetration.. It's almost as if you're looking at the submerged tip of the, excuse me, the tip of the iceberg is visible to you but the real adoption is under it.. So that as deep implications on the business model namely, Peter was referring to the cost of sales and the cost of going on monetizing the market.. In this case, people already used our software, an open source software in general.. You try to convince them to buy your value-added services, very different so is there critical mass or not? Because if there's no critical mass as an entrepreneur, you will find yourself in a very difficult situation as a VC you will look at it with different eyes.. If you don't have critical mass then, you'd still incur the cost of development because nobody's working for you for free.. Oh, and by the way even if you're at critical mass, most people won't work for you for free because we're engineers here..

We like working.. We're passionate about works.. Working for free is a different thing.. You know some sales guys think, oh developers need to be for free because they love it.. Oh, come on.. But it's really just like any normal startup.. You know, you have this high cost.. And on top of it, you don't have a license.. That's really the worst business model you can imagine.. Because young startups normally grow on license revenue, right? So you sell yourself then you get money upfront as opposed to give it away for free and people will pay you if they need your service..

That's a much riskier proposition.. So if you look at very young companies usually the revenue mix is about 80% license 20% maintenance and overtime that makes changes.. Very mature companies is the opposite.. 80% maintenance 20% license.. Why? Because the maintenance is what compounds over time, right? So that exponential growth what VCs like to call scalability.. It's actually a factor of the renewability of your contract.. Simple compound mathematics so the scalability of the model is actually in the maintenance.. But again, if you start without the license, you're on a tough situation.. For those of you that have a great software idea, you know, think about first of all is this fit for open source that disturbs up critical mass, infrastructure, software, and do you have a mass market that you'll derive maybe 1-2% monetization as opposed to licensing from everybody.. So again this is not the death of the traditional software model..

I think a lot of issues will still more than issues, a lot of markets will still have the license business model.. So frankly no critical masses poorly perform in stage of the business model.. Critical mass however, and you get the pretty lady.. Low cost of sales, marketing, low cost of distribution, low cost of the QA.. That's great, it's a highly performing stage...