Search

- Community
  - Forum
  - Company Directory
  - Founder Directory
  - YC Network NavigatorNEW
  - Batch W2012
  - Launch YC
  - $YC Top Companies by Revenue
  - YC Top Companies by Valuation
  - YC Store
  - Alumni Groups
  - This week at YC
- Resources
  - User Manual
  - Deals
  - Professional Services Directory
  - Startup Library
  - Series A Manual
  - Bookface Companion
  - My Lists
- Contact YC
  - Book Office Hours
  - Financings & Transactions
  - People at YC
  - Report Bad Actors
  - Recommend Startups
  - Email Us

- Joel
  - My Profile
  - Forum Notifications
  - Forum Keyword Alerts
  - Log Out

Search

Home/All Posts/General

Bookmark

Notifications: Email

3

Open source first companies - which license do you use and why?

Hi all, I tried searching bookface but couldn't seem to find a clear answer. I'm already developing open source and releasing it under MIT, but am considering starting a new venture that will be open source first. Unfortunately the MIT license and Apache 2.0 license provide no copyright protection. As much as I hate to do it, I feel that for such an endeavor I would need to look at GPLv3, at least for the systems that are built (supporting libraries could broken into separate packages and made available under MIT) in order to ensure copyright and patent/IP protection.

I'm curious what other open-source-first companies have done? Thanks for your time and responses!

FlagEditDelete

3 days ago | 248 views

Joel Christner
Formerly of SlidePay (W12)
joel.christner@gmail.com

Comments

Add Comment

^

2

Paul ButlerDrifting in Space(W22)2 days ago▾

We stick with MIT, partly based on the fact that at past employers, code that wasn't one of a handful of licenses (MIT, Apache, and similar) were a lot harder to start using. MIT or Apache seem to be pretty standard for infrastructure stuff, even when the intent is commercial.

I think it depends what you are trying to get out of open-sourcing it, but if you're optimizing for open-source adoption, I think the more standard/permissive the license the better. My philosophy is that not being able to capture 100% of the value you create is a better problem to have than not being able to get anyone to use it.

ReplyFlag

^

2

Alexander KlizhentasTeleport(S15)2 days ago▾

I agree that Apache2/MIT licenses in the software make adoption in the enterprise much easier. There is one caveat: I would avoid pushing enterprise features that drive conversion into OSS, and then moving them into closed source after your product gains traction. This could turn your best OSS supporters into detractors and slow down growth.

The alternative is to have a well-defined OSS/Closed-source strategy right from the start and make it very transparent to the community.

For example, you can make baseline functionality available in OSS, but reserve advanced scalability, security and compliance features for enterprise closed-source commercial license.

This will also prevent some of the biggest customers from churning after first year into OSS, if they get comfortable with support and operation on their own.

ReplyFlag

^

1
Niranjan RavichandraCedana(S23)2 days ago▾

I’m personally a fan of AGPLv3, especially if your code can be deployed over a network. It’s what we use over at https://github.com/cedana/cedana-cli

ReplyFlag
^
0
Alexander KlizhentasTeleport(S15)2 days ago▾

These days there are many different licenses on the market, you can also craft your own.

Choosing the license depends on the business objective. For example, if you sell to enterprise, you could face resistance and aversion to GPLv3 because it’s always on corporate lawyer’s ban lists.

There are multiple interesting developments in the licensing world. For example, here is a commercial license from MariaDB:

https://mariadb.com/bsl-faq-mariadb/

You can also dual-license. For example, we decided to keep things simple for now and our open source code that only has OSS features is Apache2, while the enterprise code and cloud code is closed source commercial license.

I can introduce you to a great software license lawyer with a lot of experience in this area who can help you make a choice that works for you.

ReplyFlag
^
0
Félix MalfaitTwenty(S23)2 days ago▾

Is your app more of a standalone thing or something meant to be integrated within other apps? If meant to be integrated then MIT seems like the best option, but if it’s not something you’re looking to enable then a strong copyleft license does bring more protection.

In our case, we were MIT and switched to AGPLv3 this week just ahead of our launch: https://github.com/twentyhq/twenty (because we’re a CRM, so mostly standalone app)

Note AGPLv3 is stronger than GPLv3 because it applies to distribution of code over a network, so someone can’t offer a cloud offering of a competing modified version and not release the code. Depending on the nature of your product, it might be best to opt for this option too rather than GPLv3.

ReplyFlag
^
0
Joel ChristnerOPFormerly of SlidePay(W12)2 days ago▾EditDelete

Hi Felix, I’m envisioning a SaaS platform that could also be deployed privately. Thank you for your feedback!

ReplyFlag

🏠 Home 💼 Companies 📖 Manual 🔍 Search ☰ More