TIDELIFT LIFTER GUIDE

An open source business plan that lifts all boats

Tidelift gives maintainers and core teams—we call them *lifters*—a platform and a market for building a highly profitable business around their projects. We're here to help open source maintainers and core teams build reliable incomes around their open source work.

This is a background guide to Tidelift for open source maintainers.

February 2018

TIDELIFT

An open source business plan that lifts all boats

We're here to help open source maintainers and core teams build reliable income around their open source work.

This guide contains background and details about Tidelift that will interest maintainers. If you use open source projects but don't work on them, you might want to check out our website to see how a Tidelift subscription could help you.

Our intent is to grow Tidelift iteratively, releasing early and often, and adapting as we learn. We're always looking for feedback.

The problem we're trying to solve

There's a missed opportunity at the heart of open source today: some have called this the *open source sustainability* problem.

PROJECT MAINTAINERS would love to spend more time on their projects.

COMPANIES USING OPEN SOURCE would love to have stronger assurances around the security, licensing, and maintenance of their dependencies.

BUT THERE'S USUALLY NO GOOD WAY FOR COMPANIES TO PAY MAINTAINERS TO PROVIDE THE ASSURANCES THEY NEED.

Many maintainers who create successful projects get inspired by the impact their work is having on the world, and soon start to spend countless hours (perhaps at the expense of their "day job") on their projects.

Abandoned projects are common. Maintainers often burn out, frustrated by the demands on their unpaid time. Others move on to new roles that don't leave time to work on their project.

We'd like to see a different world: where creators can run with their successful projects—focusing their time and their careers on creating value for their users.

Why does open source exist, despite this problem?

Open source works where proprietary software did not. Because open source allows permissionless collaboration across the entire software industry, it's the most powerful way to build infrastructure software that everyone relies on… even though we haven't figured out how to keep open source humming as well as it could.

PROPRIETARY

good sustainability model, inadequate collaboration model

OPEN SOURCE

poor sustainability model, good collaboration model

OUR GOAL

good sustainability model, good collaboration model

Open source allows the software industry to bypass the bureaucracies, silos, and hierarchies found in every software-using and software-creating organization. It's only feasible to depend on a thousand npm packages because there's no need to negotiate with a vendor for each of those components. And it's possible to build the deep software stacks underlying modern apps because stack-builders can solve integration, performance, and design issues across the stack, without waiting for a vendor.

Open source also creates *community*, enabling us all to be part of something larger. Participants find lasting friendships, mentorship, and career growth—and project maintainers get the chance to make an impact by helping thousands of developers across the industry.

The challenge is to *preserve the advantages of open source* while layering in a financially rewarding model for maintainers. Switching back to proprietary software as the industry standard won't do that.

So let's find a different way.

> The challenge is to preserve the advantages of open source while while layering in a financially rewarding model for maintainers

How do people fund open source today?

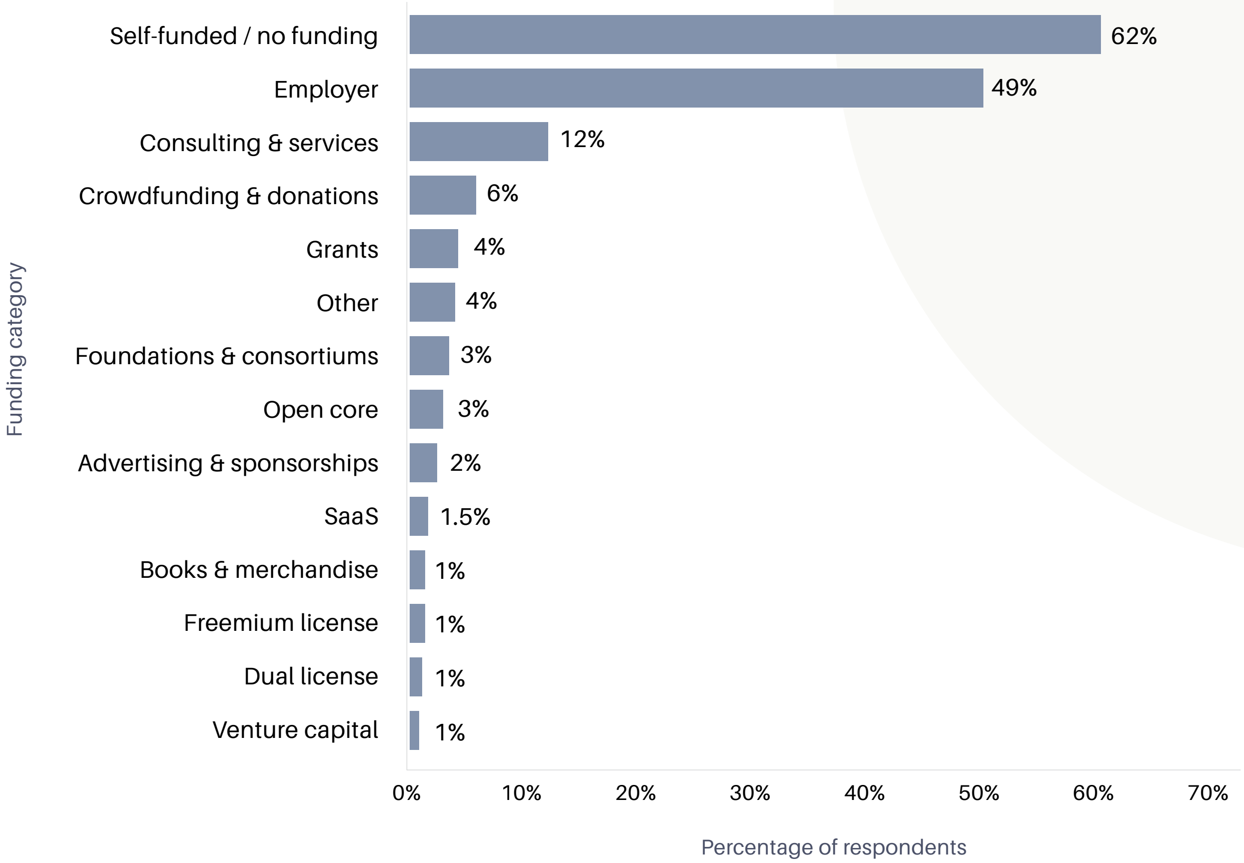
Ben Nickolls has a deep dive on this topic on the Tidelift blog, but in brief, people fund open source work today by cobbling things together. This is often unsatisfactory and inadequate.

In a survey we recently conducted at Tidelift, the most common scenarios for funding open source work today are by far:

- People work on open source as part of another job
- People work on open source for free while earning income elsewhere

Here's the breakdown:

HOW IS WORK ON OPEN SOURCE FUNDED TODAY?



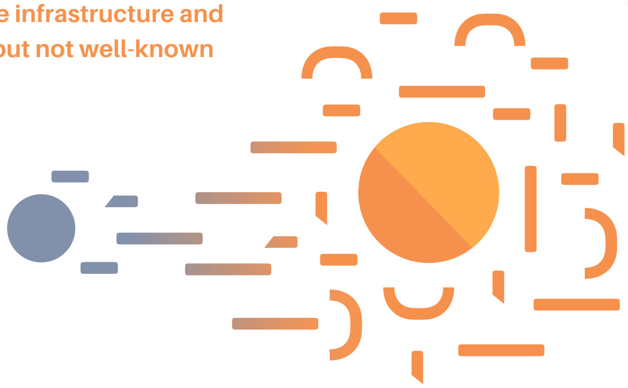
Each of these funding methods has limitations.

People allocate time from their day jobs, with or without official permission	But sooner or later, this time usually gets de-allocated in favor of other priorities
People use their spare time	But this often ends in burnout when they change jobs, or when their projects get enough users to become a burden
People earn money via consulting, merchandising, and conferences	But these are businesses in their own right, and crowd out time and resources that could be spent working on the project
Projects are funded through donations	But these mostly go to “famous” projects and often don’t add up to enough money
Projects are managed as commercial distributions with enterprise-oriented add-ons and assurances around an open core	This works well, but doesn’t scale down to small projects—and it requires venture-funded resource levels to get sales off the ground, pulling project founders away from working on their projects

When we look across the limited and fragmented landscape of open source funding options, we see two especially critical limitations:

There’s no model that reliably earns enough to support individuals or small teams (vs. venture-funded-startups and larger companies).

There’s no model for under-the-waterline infrastructure and libraries (projects that are widely-used, but not well-known famous frameworks).



The shape of an elegant solution

We've learned a lot about what works (and what doesn't) from existing open source companies over the last two decades. Here are some attributes that we feel are important for any solution to have covered:

- Make it easy for enterprise customers to buy

  Big companies have robust business requirements (and money to spend on them).

- Create a recurring and scalable revenue model for developers

  Developers want reliable income, rather than hourly or piecework compensation.

  Developers would like to see their income grow over time—and they'd like the opportunity to "win big" by getting paid based on the impact their software has, not just the hours they put into it.

- Limit free riding

  The overall attitude: the code is free, but extra professional services and assurances cost money—especially when they're enterprise-specific.

- Give professional development teams one source for open source

  Navigating a purchasing process for each of thousands of dependencies isn't practical. Successfully making an enterprise sale costs a software company $5,000 at the low end and many times that at the high end, making it an impossibility (or an improbability at best) to separately sell each of thousands of dependencies. The purchaser tends to spend even more than the seller. Bundling projects together keeps the "permissionless" feel of open source—and allows developers to add new dependencies easily if they're included in a subscription they've already purchased.

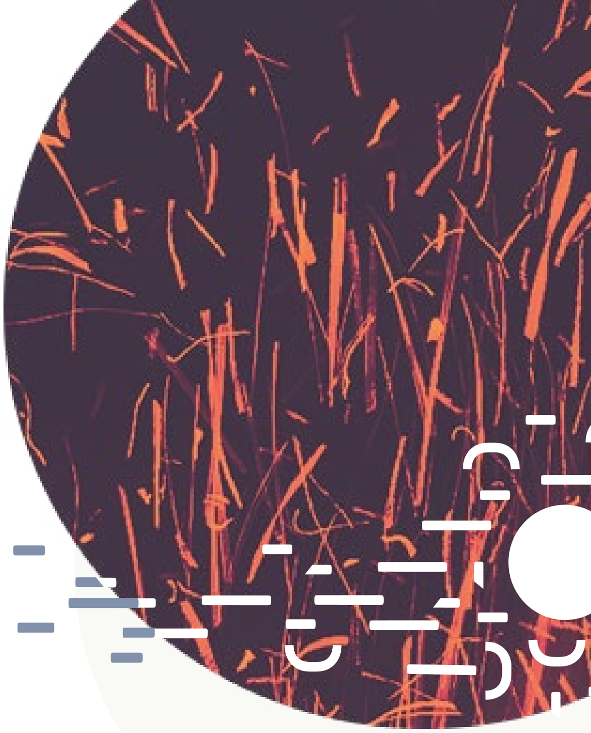
- Give open source maintainers effective sales, marketing, and administration tools

  By building the business part once, then plugging many projects into it, we allow maintainers to keep their focus on the code rather than creating business operations from scratch.

- Scale to cover open source in its entirety

  Users would like stronger assurances around every package they're using.

  Developers would like their package to be to be on the supported list—rather than excluded from a short list of blessed packages chosen by a vendor.



So how do we do this, in detail?

Here's what we're doing at Tidelift:

We create and sell subscriptions that cover open source dependencies

To get professional-grade assurances and long-term maintenance on their open source stack, companies can buy Tidelift subscriptions that cover all of the packages they use.

In our recent survey, **83% of organizations** said they will pay for well-managed, supported, licensed, and secure versions of the open source software they already rely on. Over 50% of organizations are developing custom approval processes and tools to vet and approve new open source dependencies. But these home-grown solutions tend to be low quality—they don't have a lot of depth, and create a hassle for their development teams.

Lifters agree to help subscribers

For each package, we'll define (and continually evolve) a set of responsibilities and tasks to ensure the package meets subscriber expectations around security, maintenance, and licensing. Lifting a package means agreeing to take ownership of these responsibilities.

From a subscriber perspective, lifted packages are pre-vetted and maintained to a known standard.

Lifting a package requires a bit more work than most maintainers would typically do on their own time—but we believe it will be very manageable, leaving most of a maintainer's time free to focus on developing their project. Some lifting work can also be done by non-maintainers, as a way to help out a project on a volunteer basis.

> The average case is that packages have a single maintainer, who also acts as the single lifter, and who also gets paid. But we want to support more complex cases.

We track subscriber dependencies to split up revenue

As a condition of their subscription, subscribers agree to track their dependencies with Tidelift's tools (and we're offering a dependency analysis service on Tidelift.com to give people a good reason to get started).

Using this information, we flood subscription payments down through the dependency graph—including dependencies of dependencies, transitively—so each subscriber's fees go to support the (lifted) packages they use. Projects with lots of users will therefore make more money than those with few users.

Tidelift will split up by package, but not among individuals

We'll compute a revenue share for each package, based on that package's usage and a measure of its size or complexity. This measure will be loosely defined—related to but not literally the actual amount of code in the package. The goal is to avoid financial incentives to split or merge packages.

We'll then determine who can make decisions on behalf of a package (in most cases, this is the package's single maintainer). That person can tell us where to send revenue assigned to the package. Tidelift will not decide whether or how to subdivide revenue within a package—projects that feel multiple entities should be paid for one package will decide on their own principles for that.

Tidelift provides business services: sales, marketing, operations, legal, and finance

Our role is to create everything that's shared amongst projects (rather than per-project). We want to be a "business-in-a-box" toolkit for open source projects, allowing maintainers to focus on what they do best—code.

We provide a service to open source projects:

YOU SIGN UP TO HELP SUBSCRIBERS.

WE HELP YOU LEARN TO DO IT AND GIVE YOU THE TOOLS YOU NEED.

YOU GET PAID.

We're paying for assurances and scalable work, not for hours

This isn't a piecework gig or bug bounty—so we won't be paying by the hour or by the task. Instead, if a project is lifted, there's no cap (or floor) on how much it earns. It works like a license fee, not like an hourly wage.

What does lifting a package involve?

We are continuing to refine and evolve what lifters do and what benefits subscribers receive. But broadly, we see three key categories of subscriber benefits:

- Security
- Licensing
- Maintenance

Over time, we'll build tools and automation to help lifters do their work quickly and efficiently.

We may need to segment some "levels" of lifting a package, which could look something like this:

- Some tasks are doable nights and weekends, while others might require more full-time focus
- Some tasks are doable by anyone, while others might require a maintainer or core team member (with the ability to publish new versions of a package)

To start, we're focusing on lower-commitment assurances. In the future, we might add higher levels of service (that also have the potential to earn more income).

Enhanced data for automated dependency analysis

We have a dependency analysis and monitoring tool on Tidelift.com, which is bundled with subscriptions.

This tool uses data from Libraries.io, but will be enhanced further with the participation of lifters in the future.

Lifter work in this area boils down to reviewing, correcting, and extending the Libraries.io-scraped data for each release to ensure we have accurate data on:

- Licenses (in SPDX format)
- Known CVEs
- API or ABI breaks (especially for packages that don't use semver)

We'd also like lifters to write some helpful, human-readable release notes for each release. By putting them in a consistent format, we'll be able to automate rolling them up into a custom newsletter for our subscribers.

Assurance of ongoing maintenance and quality bar

Promising to stick around and maintain (or gracefully hand off) a package is a step up from off-the-shelf open source, and maintainers could do this. Package maintainership has high churn today—and packages are orphaned constantly.

As another layer of maintenance assurance, we're developing a metric for package quality.

Lifted packages assure customers that the quality score will be kept high.

Assurance of security response

When packages have security issues, there's an industry-accepted responsible disclosure process for handling them. At a high level, it looks something like this:

- Have a way to confidentially contact the project
- Create a CVE number for the vulnerability
- Coordinate with affected vendors to create a fix before announcing the problem

Maintainers lifting their packages agree to develop and release a patch for any security issues, working with a coordinator at Tidelift to navigate the CVE and embargo process. We'll walk you through it—but we need a domain expert who knows the code for a package and can publish a new release.

Assurance of accurate license information

This sounds simple, but quite a few packages don't accomplish it. It's simply an assurance that the license information on the package is accurate (and in machine-readable format).

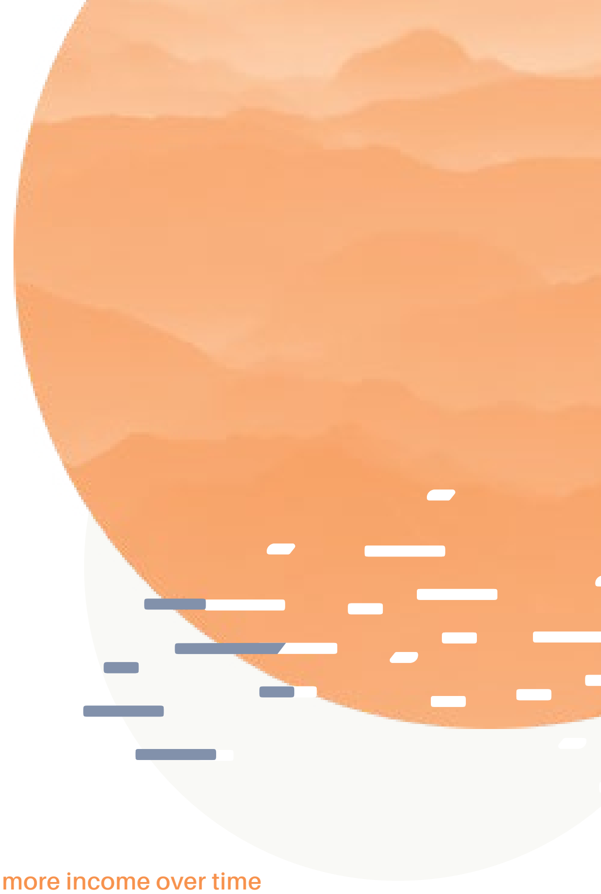
This enables subscribers to set license policies and easily detect any violations of those policies.

And more...

The above is a baseline set of assurances that we've confirmed that subscribers value. There's much more we will add over time (like giving lifters the option to do less to fit around their day job, or to do more to earn more income).

Pilot the workflow and then document

We are currently running pilot projects with early lifters and customers, to refine and explore how projects can best help their commercial users. When we're ready to scale this up, we'll create a more detailed lifter guide that documents the maintenance, security, and licensing practices subscribers are looking for, and outlines how to bring a package up to speed.



What does a business around a project look like?

Businesses large and small

Right now, it's a struggle to create a small business around an open source project. There are a couple of success stories, but common paths include consulting (which pulls maintainers away from working on the project) or venture funding (which only works for the largest projects).

Tidelift is creating a smoother gradient, allowing people to get started with a few dollars on evenings and weekends, gradually ramping up to enough earning potential to pay one or a few people for full-time work. Ultimately, we hope this model becomes robust enough to support teams seeking substantial financing.

In the world of SaaS products, there's already a gradient like this—lots of people build small businesses around proprietary software and web apps. But it isn't feasible today for typical open source infrastructure projects, because the known business models don't support the work that maintainers want to get done.

Simple to get started

There's no need to jump through a bunch of legal hoops to start earning money on Tidelift. In the simple case, maintainers will sign up to provide assurances around their packages, give us their payment information, and start getting paid.

Potential to build more income over time

Many successful businesses build multiple "revenue streams," rather than relying on a single source. Businesses around open source will be the same.

Participants in the Tidelift ecosystem may increase their income in many ways:

- ➜ Increase the popularity of their package—more users means more subscribers means more money
- ➜ Promote the Tidelift subscription on their project website or in other ways, to turn more users into paid subscribers
- ➜ Maintain more packages
- ➜ Offer consulting or other services (charging enough to make it worthwhile)
- ➜ Offer books, videos, courses, or trainings
- ➜ Offer add-ons to the open core
- ➜ Use dual licensing to charge an additional license fee

In the world we want to see, for many maintainers, a share of Tidelift subscription revenue on its own will be enough to support their work. Others may decide to use the subscription share as a base, while adding other income sources. Where possible, we'd love to offer tools and guidance through Tidelift to help maintainers grow their income.

What can projects do to help make this a reality?

If the idea of a monthly payment from your commercial users sounds exciting, here's how to get your project there sooner rather than later.

Tell us you're interested

Get in touch on Tidelift.com—let us know what open source projects you're working on and what you're thinking about when it comes to Tidelift.

Reach out to your users

Each Tidelift subscriber *using your package* means more money for the project. We don't recommend doing anything crass like filling your home page with ads. However, when the time comes, we'd strongly encourage mentioning—in a tasteful way—that commercial users who want professional assurances can buy a subscription for it.

If you point potential subscribers to Tidelift, we'll have sales and marketing ready to guide them from being potential subscribers to actual customers.

We'd also encourage you to talk to commercial users about their needs and help us incorporate what you learn into the Tidelift platform. As you find interested users, we'd love to meet them as well. Each project will have slightly different users with their own needs and perspectives.

Contribute to Libraries.io and bibliothecary

To pay packages on Tidelift, we'll use Libraries.io as a data source. It supports a lot of ecosystems, but not all—there are package managers that Libraries.io doesn't understand, or that aren't fully supported. We can use help from experts in each ecosystem to ensure we accurately scan that ecosystem.

Talk to us

We're a friendly bunch of long-time contributors and supporters of open source. There's a lot we haven't figured out yet—maybe you can help us! What would make Tidelift work for you? What questions do you have? Let's try to solve it together.

Please get in touch if you have thoughts. info@tidelift.com is the right place to email us, or you can use the contact form on Tidelift.com.

LET'S DO THIS

In the coming years, we're going to put 100% of our efforts into making open source work better —for everyone — software development teams AND open source maintainers. We hope you'll join us.