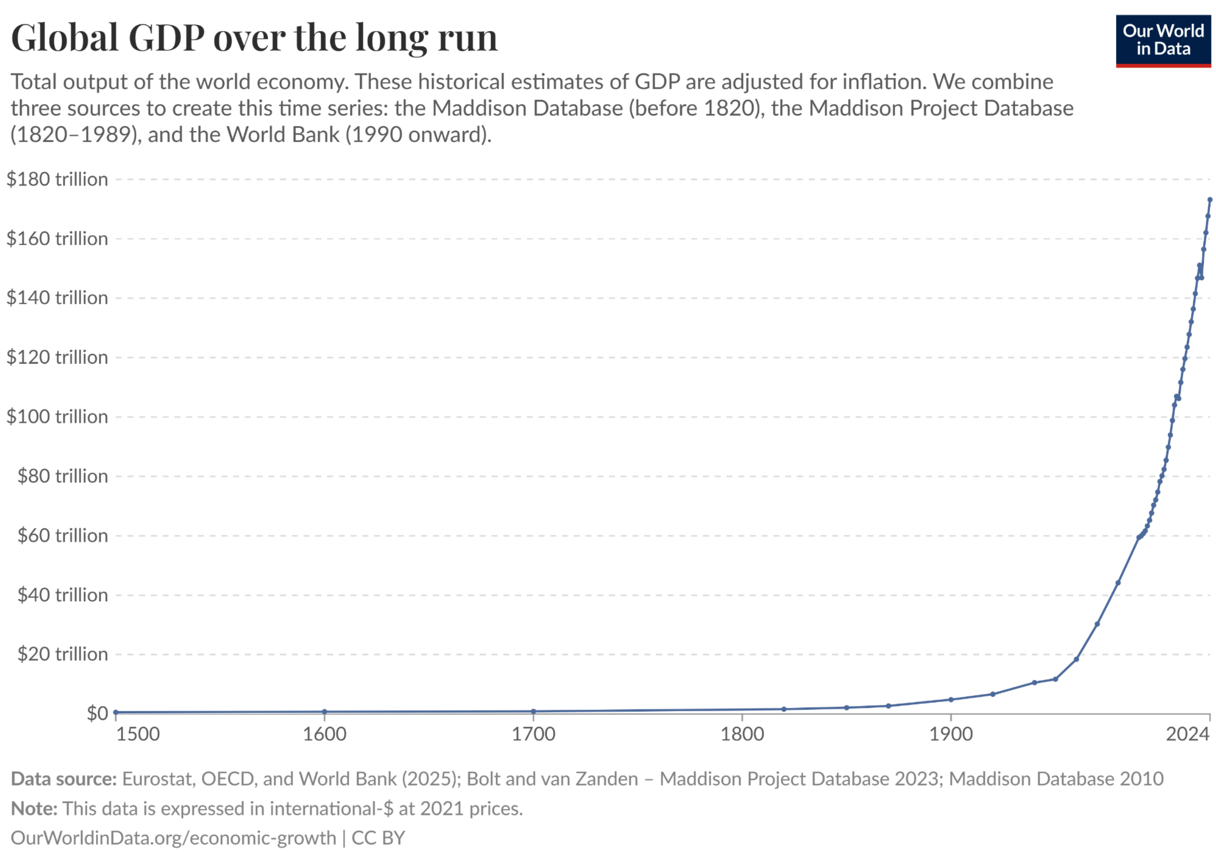
# Prosperous Software Movement

## Let's change the economics of software development with a third-wave open software movement

*Disclaimer: This is an active work-in-progress document, and most ideas are very much open for discussion with the broader open source community. We'd love to hear from you on how to make this work for your project.*

The past 2 open source movements have yielded transformative effects to the software industry. The **Free Software Movement** led to the birth of Linux, GNU, and countless fundamental technologies that lie at the heart of nearly all modern computing systems. The later **Open Source Movement** led to an explosion of open source software in nearly every industry. Today, 99% of companies use free and open source software, and 70%+ of lines of code shipped to production comes from open source libraries [1,2]. Free and open source software is the bedrock of the entire economy, which has been growing exponentially with technology.




*World GDP since 1500 (Source)*

The predominant software licenses of today do not reflect the financial realities of the modern economy. Not only do open source developers rarely benefit from this incredible growth, many struggle to find support for their project. This common scenario applies even with many projects that form the fundamental infrastructure that made this economic growth possible [3,4,5,6,7,8]. Past open software movements focused on principles of freedom (e.g. GPL, AGPL) or open collaboration (e.g. Apache, MIT).

We propose that we need a new phase of open source, tentatively called the ***Prosperous Software Movement***, with the goal of enabling all the value creators in the economy to financially benefit from the prosperity they helped create. To meet this challenge, we believe there needs to be a new open source software license that rewards open source developers, while holding true to the principles that make the open source community great. We tentatively call this license the ***Prosperous Software License (PSL)***.

At the heart of this proposal is the intention of **revenue-sharing**. The mechanism is simple: if you use PSL-licensed software in your stack and you make over $X in annual revenue, you must distribute Y% of your revenue to **any open source projects**.

We expect that not all companies will be willing to share in revenue or license their software in this way. However, we revenue-share for many reasons beyond gratitude and altruism. We believe that fairly compensating value creators and incentivizing more innovation at every layer of the technology stack will lead to an ecosystem of new digital products that collectively out-innovate the alternative group of uncooperative companies.

If you share this intention, please reach out to us and engage in a dialogue to help us figure out how we can make this work for your organization.

# Why is this important now?

Proponents of open source software have made many sound arguments why we should fund open source software. It makes our software supply chain more secure, when developers have the resources to maintain and fix bugs [9]. Open source software lowers the barrier to entry in a global digital market, facilitating better economic inclusion, especially in developing countries [10]. Lastly and most obviously, it is the right thing to do. You inevitably use open source software every day, either directly or indirectly in nearly every piece of software.

However as PSL is an economic proposal, we'll focus on a forward-looking economic argument:

## Funding open source will accelerate the economy

Open source software is digital public infrastructure, accessible to anyone on Earth with a computer. When we invest in better infrastructure, we bring step-changes to economic growth. Relying on donation platforms to fund public infrastructure is insufficient to bring the type of society–wide coordinated investment we need to make to build the future of technology. Nobody likes to pay tolls or taxes, but we cannot deny that the economy would look a lot smaller without the interstate highway system.

### The growing AI economy is the time to embody these changes

We are living in a unique moment in time, where the tech industry is actively re-imagining every layer of the stack, including:

- How data is stored and processed with new types of databases and storage systems optimized for AI
- How applications are architected with new AI application frameworks
- How users interact with machines via voice-control, brain-computer interfaces, and intelligence directly embedded into everyday devices
- How devices coordinate in new network topologies to give a cohesive user experience

We are living through one of the most inventive moments in human history. As we displace the prior generation of software, we are building brand new open source technology to realize this future. Now is the opportunity to embed new principles into our collective development stack, which is already dominated by open source software.

## Design Goals

To ground the design of this new license, we suggest the following goals:

### Let donors choose which open source projects get funded 💵🔀

Value is in the eye of the beholder. There are no universal metrics that can determine how valuable an open source project is. The best way to scale judgement is to let each donor allocate revenue-sharing percentages to the projects they find most worthy.

### No self-dealing ❌🔄

Revenue-sharing should benefit individuals and organizations outside of the donor's control. If a donor could simply benefit themselves from direct or indirect transactions, it would run counter to the intention of revenue-sharing with others. We can draw inspiration from existing legal language around self-dealing.

### Collective bargaining 👪

Individuals and companies can and often do pledge to donate to open source [9] without the need for a new license. However, PSL could form the basis of a new form of collective bargaining through copyright law. As more projects choose to license their software with PSL, it grows the movement's collective bargaining power in positive-sum ways. Companies that build within the revenue-sharing ecosystem benefit from the accelerationist effects of a growing corpus of open source technology. Companies that build software outside of this ecosystem must do so at the cost of reduced velocity.

### Wide compatibility with various software licenses and business models 🏞️

PSL should be compatible with a wide variety of software licenses, from commercial proprietary software to free GPL software. Similarly, PSL should be compatible with a wide variety of business models, use cases, and deployment models. The intention is not to coerce other software developers to adopt the same license. Rather, PSL software simply adds a revenue-sharing requirement to its users. Being able to easily dual-license with PSL would greatly benefit adoption.

### Stay “Open Source” 📖💻

There is an enduring power and emotional attachment that people around the world feel to the “open source” movement. We believe PSL is a way to grow the movement from within. Compared to the other source-available licenses, PSL should not restrict distribution or discriminate against specific fields of endeavor [10]. Anyone in the world, from students to professional software developers should be able to freely use the software, fork, and contribute to the software.

### Enforceable 🤝

While PSL does not coerce any software developers to use the license, if companies do use PSL-licensed software, we must have the legal framework to enforce compliance in domestic and international courts. The legal theory should be sound, yet compatible with the ethos of the open source community.

## How will PSL work?

### Revenue-sharing to the open source community

TODO: diagram that visualizes the process.

1. An open source developer chooses to license their software with PSL (e.g. Library-Foo)
    - We should make it easy to dual-license with other popular open source licenses (e.g. MIT, Apache, GPL)
    - The developer chooses the revenue-sharing percentage. For example: PSL-1% or PSL-5%.
2. An engineer at a tech company, Startup-Bar, uses Library-Foo to build a commercial service.
3. While Startup-Bar is young and unprofitable, there are no obligations. They use PSL-licensed software freely as they would any other open source software, allowing them to focus on growth and product development.
4. When Startup-Bar hits certain revenue thresholds, the revenue-sharing clause kicks in, requiring a certain percentage of revenue to go to open source projects.

5. Startup-Bar has a deadline (e.g. 1 calendar year) to ***donate to any open source*** project in their stack, not limited to PSL-software.
    - It is up to the donor to choose how to split donations to different projects, giving them the freedom to reward projects they value most.
6. If anyone forks Library-Foo, they must inherit the PSL provisions in the fork.
7. If a company uses 2 pieces of PSL-software, we aggregate their revenue-sharing percentages. TBD if it should be MIN(), MAX(), AVG(), MEDIAN().
    - For example, MAX(PSL-5, PSL1) = 5% revenue-sharing

## Why use PSL?

### Independent open source developers

These developers will be the primary source of collective bargaining power. As more developers choose to license their software PSL, it increases the surface area of software that would be subject to revenue-sharing. Independent developers are the primary audience for this proposal, who hopefully most likely identify with the mission and purpose. These can be individual maintainers or potentially early open core companies.

### Open source projects under corporate hosts

Companies of all sizes, from big tech companies to open core startups release open source software. They often do it for strategic reasons, whether it helps top-of-funnel sales or erodes a competitor's edge. These developers have the least reason to license their software as PSL, self-imposing a revenue-sharing requirement on themselves. In an ideal world, the PSL movement grows enough cachet to incentivize even these open source projects to join the movement, especially if their products cater towards developer users.

### Non-commercial consumers of PSL software

Non-commercial users of PSL software, from hobbyists to academics would be unaffected by the revenue-sharing requirements. They freely use PSL software just like any other open source software.

### Commercial consumers of PSL software

Corporations are highly rational organizations. As long as the revenue-sharing costs are less than the switching costs to an alternative, they will continue to use it. This calculus will depend on what software component we are considering. For example if there are many non-PSL open source alternatives (as in SQL databases), then there will be an incentive to avoid the PSL software. If there are no open source alternatives, but many commercial alternatives, it will depend on the market pricing power of this class of software. On the other end of the spectrum, if there are no alternatives, then we would need to estimate the cost of the enterprise to build their own version from scratch. As the corpus of PSL-licensed software grows, it gives the PSL community the ability to charge higher revenue-sharing percentages.

# Strategy 1: Go-to-market via mass adoption

Key to this movement is coalition-building. The collective bargaining effects work best with the largest possible surface area of projects joining the PSL community.

## Establish membership benefits

The movement starts with developers choosing to re-license their software. It is an opt-in action. We should create a membership model that confers benefits to joining, from soft benefits like marketing/branding, to hard benefits like patent-pooling.

## Start with a negligibly small revenue-sharing requirement

We should start with a negligibly small revenue-sharing component (e.g. 1bps => PSL-0.01%). This means a $1M ARR company would pay $100/year and a $1B ARR company would pay $100K/year. For most companies, this amount is trivially small enough to not be worth the time to swap out one library for another.

## Build a coalition of OSS projects

With a small revenue-sharing component, we try to amass the largest coalition of open source projects to sign up as possible through grassroots advocacy.

## Onboard companies to revenue share small amounts

Through auditable platforms (e.g. like Open Collective), we begin collecting annual revenue on behalf of open source projects.

## Ratchet up

As the movement grows, we should make it easy for individual OSS projects to ratchet up their revenue-sharing percentage (e.g. from PSL-0.01% to PSL-0.1% to PSL-1%)

# Strategy 2: Go-to-market in a single niche vertical

While mass adoption is the dream scenario, we should also pursue an alternate adoption strategy where we have more control over the outcomes. We'll use the blockchain data industry as an example, but we could do this in any vertical

## Small-group coordination

Among the handful of open source / public goods oriented data projects, we agree on a revenue-sharing percentage that best represents our intention to collaborate on shared value. For low-levels of collaboration and data-sharing maybe we use PSL-5%. For deeply interconnected projects, maybe we use PSL-20% or PSL-40%. For example, we could setup a working group amongst [Open Source Observer](), growthepie, L2Beat, DefiLlama, blockchain-etl, cryo, and Goldsky.

### Release open source code and/or open source data

In an ideal world, we release both open source code and open source data to the community. This is more accessible, more friendly, more community-oriented. Ideally, this drives more awareness, usage, and contributions to the open source data movement.

### Grow the network of open source contributors

Because there is clear advanced commitment to revenue-share, new teams pop up to build innovative datasets and data products for the blockchain ecosystem.

### Outgrow the uncooperative data industry

This PSL-organized data community eventually out-innovates the alternative universe of proprietary uncooperative data companies. We grow the market share of PSL-oriented companies to eventually dominate the blockchain data market.

### Evangelize the case study

Once we've shown that there is a better way to organize competitive force, we try to bring this model to other domains/verticals/use cases.

## Open Questions

This post is just the beginning of a long journey towards building a PSL movement. There remain many open questions we will need to decide as a community, before launching. In this section, I'll raise some questions and offer some suggestions, but expect these answers to change as the community engages in deeper discussion:

**How much should we charge in the revenue-sharing requirement?**
As a starting point, I'd suggest similar parameters as the Epic Games ecosystem:

- Free for any use case under $1M in revenue
- Once you reach >$1M in revenue, pay 5% towards your open source projects

**Why revenue? Why not profits?**
Tech companies are famously unprofitable. Many companies that are market-dominant or traded in public markets choose to stay unprofitable to fuel growth and expansion. Revenue is one of the best indicators of traction and business maturity.

**Why not include a fraction of the revenue-sharing for the "triggering software"?**

**What about free-riders on the open source side?**

AFL

**Why cash donations?**
Many tech companies will donate resources, like CI compute hours or developer time. Regardless of intention, this often gives the corporation power and influence over the open source project. Cash is the ultimate unburdened resource, giving the open source maintainer the freedom to allocate resources towards their most pressing needs. We would never expect companies to accept payment for their proprietary products in the form of volunteer time.

**What revenues are subject to the revenue-sharing requirement?**
Many big tech companies have many independent products with independent P&L. We should limit the revenue-sharing component to the products and services that leverage the PSL software, rather than the revenue of the whole corporation.

**Who decides the revenue-sharing percentage requirements in the license?**
As mentioned above, the bargaining power of any particular piece of software will depend on the market they operate in. As such, we suggest giving the PSL developer the freedom to choose the revenue-sharing percentage. We hope to provide high-level industry guidance, support, and defaults so that developers are not overwhelmed by the choice.

**What about using the PSL license for data and content?**
In the AI economy, thinking about how we license data and information is as important as how we license software. Consider the case where an AI lab trains a coding LLM model on PSL software. In this case, we suggest that the revenue-sharing requirements should also apply to the LLM service. Similarly, we can imagine licensing books, articles, and data as PSL.

**What about a membership model with additional benefits?**
e.g. patent-pooling

**What if companies ban using PSL?**

**What qualifies as an open source recipient?**
To make it easy, clear, and unambiguous, we suggest donors can fund any software project that is licensed with an [OSI-approved license](), which reviews licenses to meet the [Open Source Definition](). We hope they will agree that PSL itself is also compliant with this definition.

**What constitutes commercial usage of PSL software?**
We'll leave it to the lawyers to help draft the best language here, but we hope there are industry best practices already in place from years of experience with GPL, AGPL, LGPL.

**When an open source project receives funds from a PSL donor, are they also required to revenue-share?**

That would be ideal, creating a networked flow of funds deeper into the technology stack. This enables developers at every layer of the stack to benefit from the movement's cash infusion, not just front-line software tools and libraries.

**What are the adoption goals?**
It is a stretch goal to make PSL expectations universally accepted in all software. A more achievable goal in the short-term is to grow within a particular industry first (e.g. AI, crypto, or cloud devops, etc). Still TBD.

**What is the best legal theory for enforcement?**

**What if a corporation donates to a non-profit that it has outsized influence over?**
A corporation could try to circumvent the intention of PSL, by donating to a non-profit that it has outsized influence or control over, which would give them the ability to leverage the funds to develop the OSS that directly benefits them. This would violate the spirit of "no self-dealing", but we leave it as a legal open question whether we can avoid this.

**Should we "profit-left"?**
Previously a stated goal was to not coerce any developer to choose the PSL license. There is an argument to be made that the movement would spread more quickly if we had a "profit-left" clause. I imagine something similar to copyleft, but spreads the revenue-sharing requirement to any software that uses PSL software. A topic for discussion and consideration, as I believe the movement could succeed with or without it.

**How do companies report donations?**

**How do these percentages add up if there are multiple PSL-dependencies?**
Max(), MIN(), MEDIAN()?

**How to handle change of license?**
Adoption. Or later changing the percentage?

**What about anyone can sue provision?**
Dd

**Are non-profit or governments exempt?**

# Related work

## Related licenses

- Copyleft licenses (e.g. GPL)
- Copyfair
- Source-available commercial licenses
    - BSL
    - Functional Source License
        - https://fair.io/faq/
    - Unreal
        - Free for any use case under $1M in revenue
        - Free for all educators / schools
        - Free if you publish your game in the Epic Games store (you pay the store transaction fees)
        - Otherwise >$1M in revenue, pay 5% royalty to Epic
    - SSPL

## Related revenue-sharing initiatives

- Optimism retro funding - 15% of sequencer profit or 2.5% of revenue
- OSS Pledge - $2K per year per developer in company
- Automattic pledge - Contribute 5% of time/resources

# Conclusion

TODO: make this outro, a banger

Long-term, we want to show that the economic value of software built using this license directly or indirectly can out-innovate, out-compete, and out-grow the alternative universe of uncooperative software.

This won't be for everyone. It is up to us to show that new software built in this model will out-innovate the alternative, like the free software movement did with Linux. It might take time, but like the free software movement showed us,

If you're an open source project that is interested, let's figure out how to make this work for you
If you're a lawyer with experience writing software licenses, we need help
If you're an open source advocate, we will need help evangelizing this when we have it ready.

# Appendix

## Consequences

- Imagine a world where anyone building awesome OSS (e.g. Docker / sqlmesh) doesn't have to waste time searching for a enterprise offering to make money, and focus on building FOSS.
- Imagine a world where we have 10x more open source software, across every sector of the economy
- Imagine a world with significantly more economic inclusion
- Imagine a world with significantly more transparency
- Imagine a world with significantly higher GDP

## Potential Partners

- Crypto
  - Ethereum Foundation
  - Eigenlayer
  - Optimism Foundation
  - Protocol Labs
- OSS non-profits
  - NumFocus
  - Linux Foundation
- Companies
  - Wordpress / Automattic
  - Sentry
  - FOSS Funders
- Donation platforms
  - GitHub sponsors
  - Open Collective
- Source code management systems
  - GitHub
  - GitLab

## Problems with current system

- Open source projects need to find some other way to monetize (e.g. auxiliary services)

## References


- https://www.theguardian.com/sustainable-business/2016/apr/05/india-csr-law-requires-companies-profits-to-charity-is-it-working