Who Pays For Open Source?

Shaun Walker

Founder of Devessence Inc
Microsoft Most Valuable Professional (MVP)
Creator of DotNetNuke (DNN)
Creator of Oqtane
Chair of .NET Foundation Project Committee

Founder
devessence
+
oqtane

@sbwalker

My Open Source Journey

2001-2002 Created DotNetNuke as hobby project

2003 Released DotNetNuke as open source project

2004-2005 Donations, sponsorship, sponsored product enhancements

2006-2008 DNN Corp - donations, advertising, sponsorships, professional services

2009-2014 Venture capital, dual license, marketplace, open core, managed service

2015-2017 .NET Foundation volunteer

2018-2022 Created Oqtane open source project

2023+ Created Oqtane Labs Inc & Devessence Inc

Tried every possible open source business model, worked full-time on open source for 12+ years

= part-time / unpaid

= full time / commercial open source

Definition: Sustainability

"avoiding the depletion of **resources** in order to maintain an ecological **balance**"

Open Source Sustainability

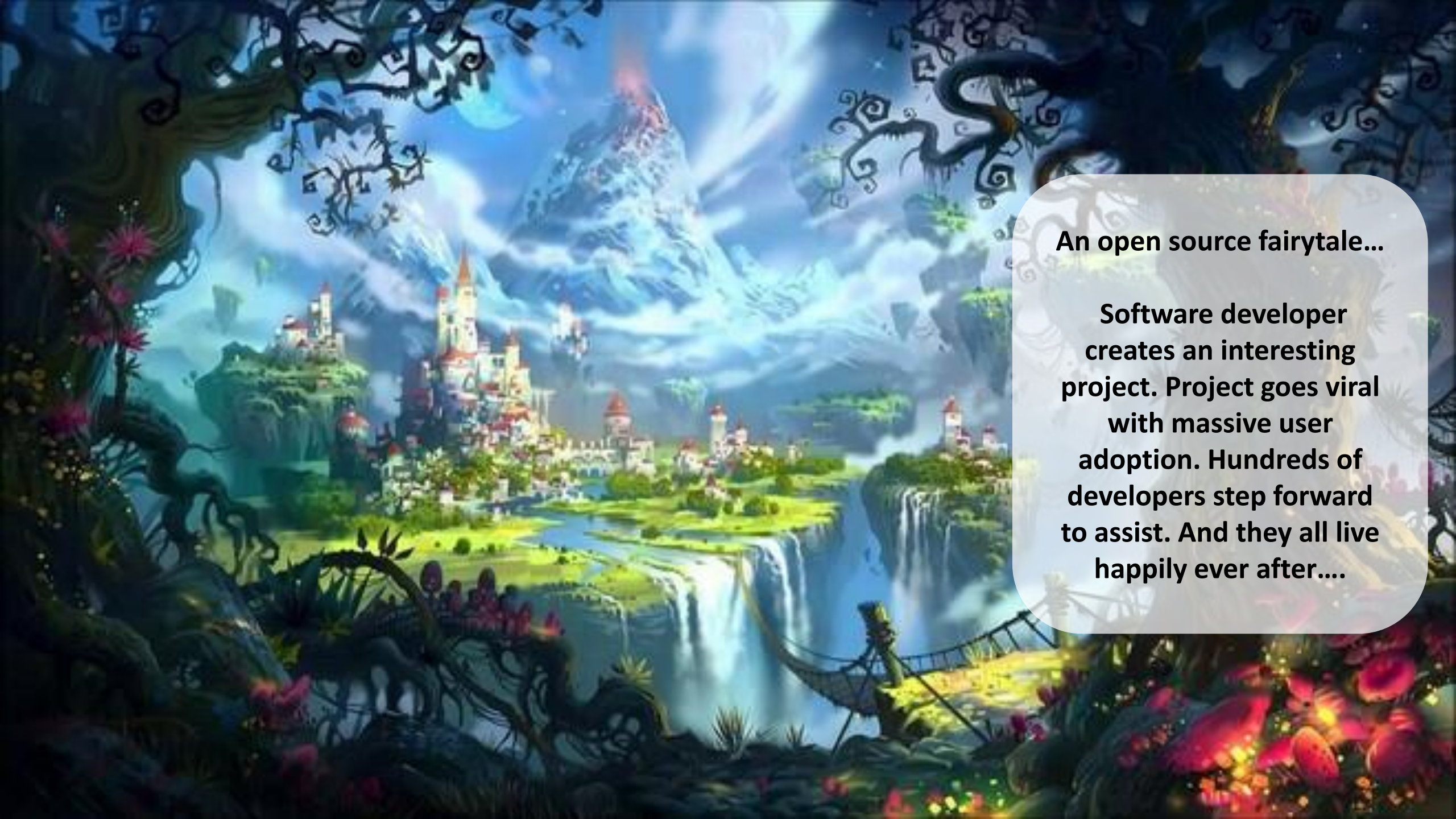
Resources = Contributors

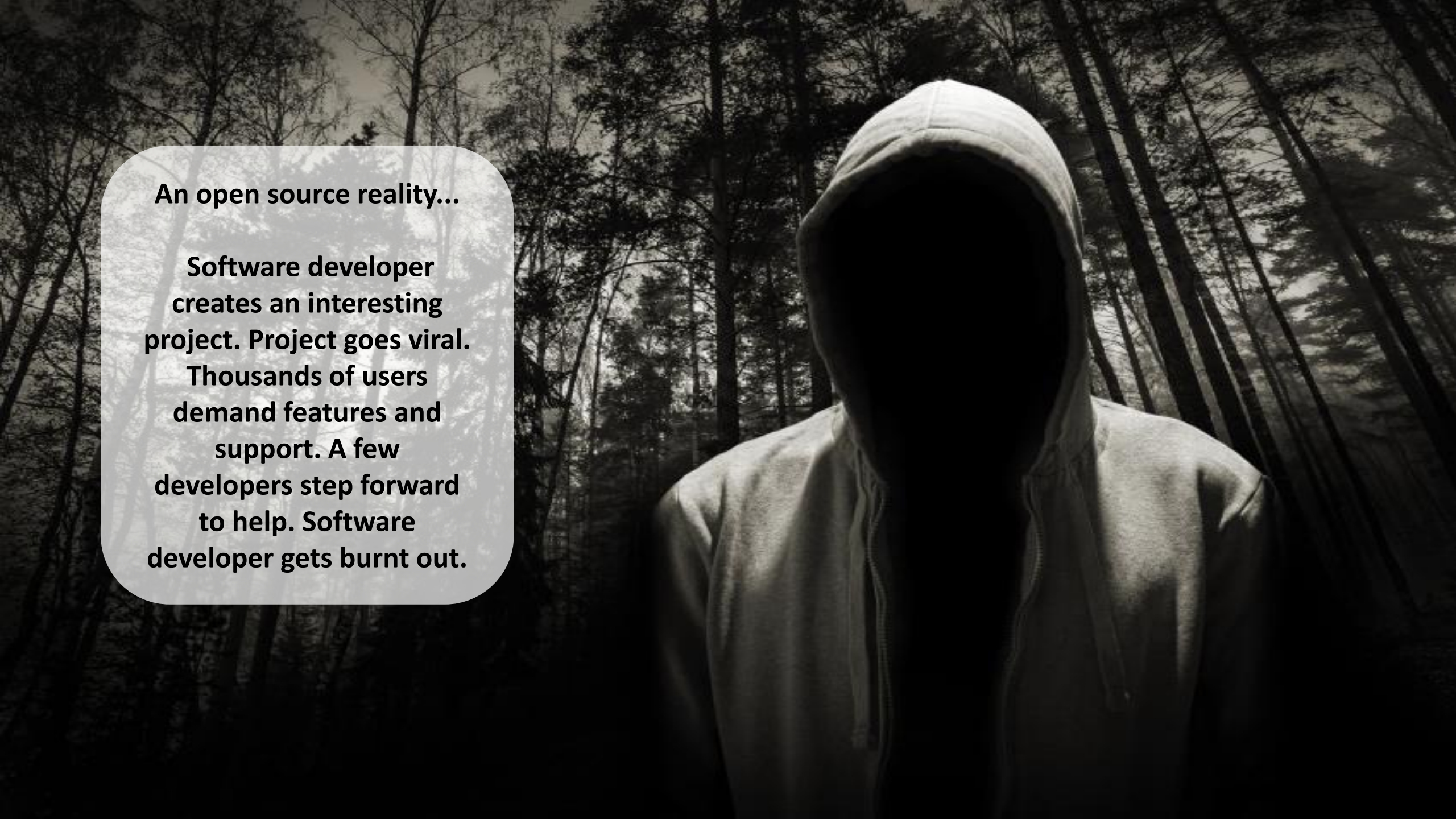
Contributors = Expertise/Time

Time = $$

$$ = Business Model


Delete
Open
Source


An open source fairytale…
Software developer creates an interesting project. Project goes viral with massive user adoption. Hundreds of developers step forward to assist. And they all live happily ever after….


An open source reality...
Software developer creates an interesting project. Project goes viral. Thousands of users demand features and support. A few developers step forward to help. Software developer gets burnt out.

95% of users do not contribute to open source

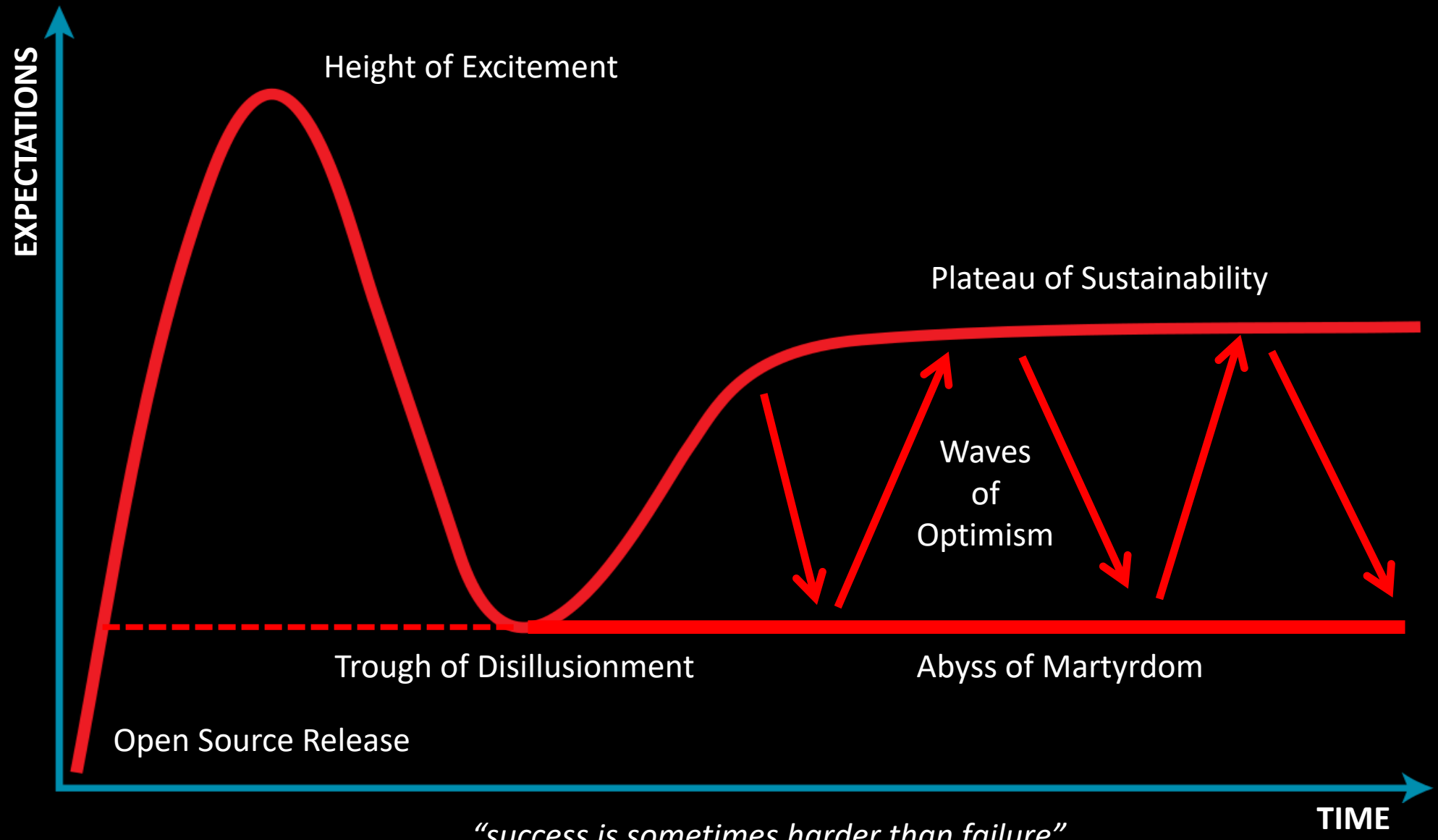
95% of contributions are done by 5% of contributors

Most open source projects are dependent upon 1-2 contributors

Losing a primary contributor often results in the abrupt end of an open source project

There is generally no succession/continuity plan or disaster recovery process in open source projects

Open Source Maintainer Hype Cycle



"success is sometimes harder than failure"

Ryan Chenkie
@ryanchenkie

Wanted!

10x rockstar developer 🔥

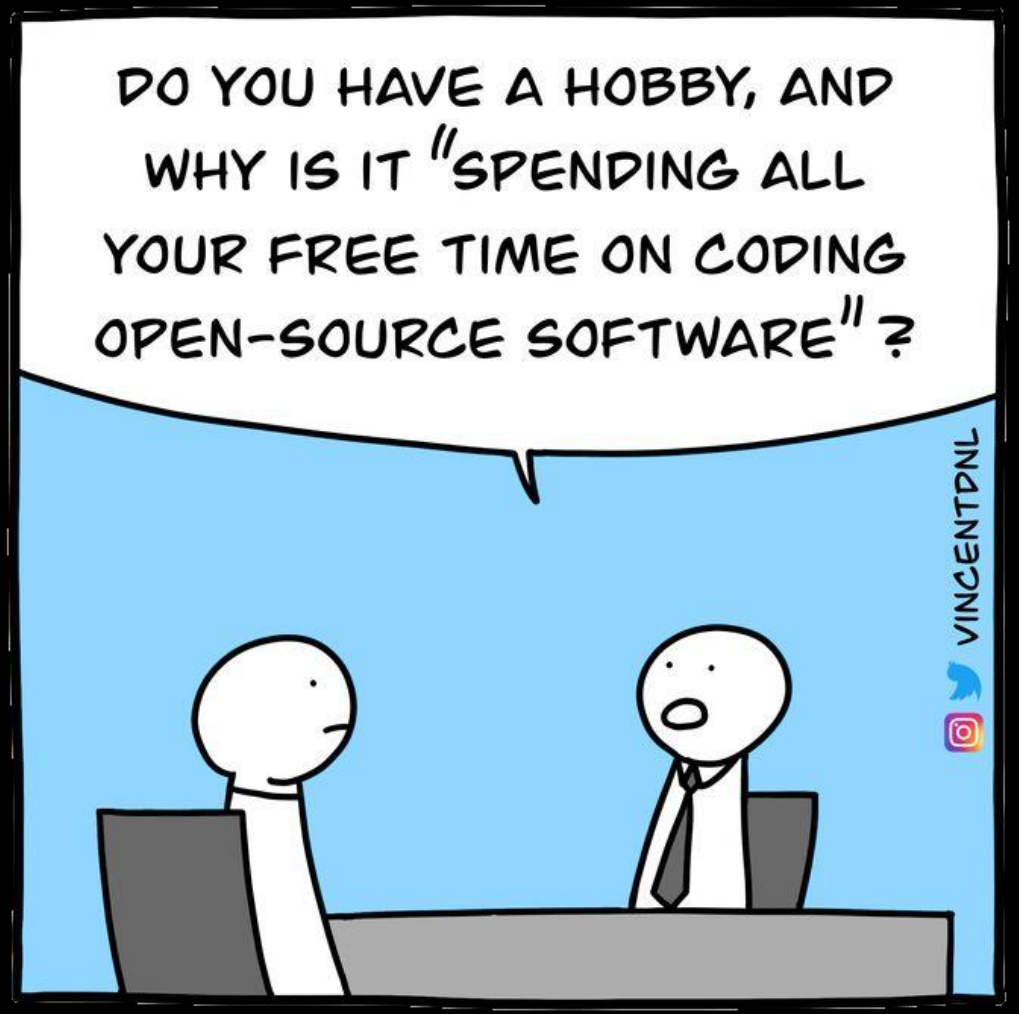
Responsibilities include:
* Merging PRs immediately
* Making new features on demand
* Fixing bugs right now

Compensation:
* Stars
* +1's
* Threats, general and specific
* Public shaming

Apply at OSS Inc today

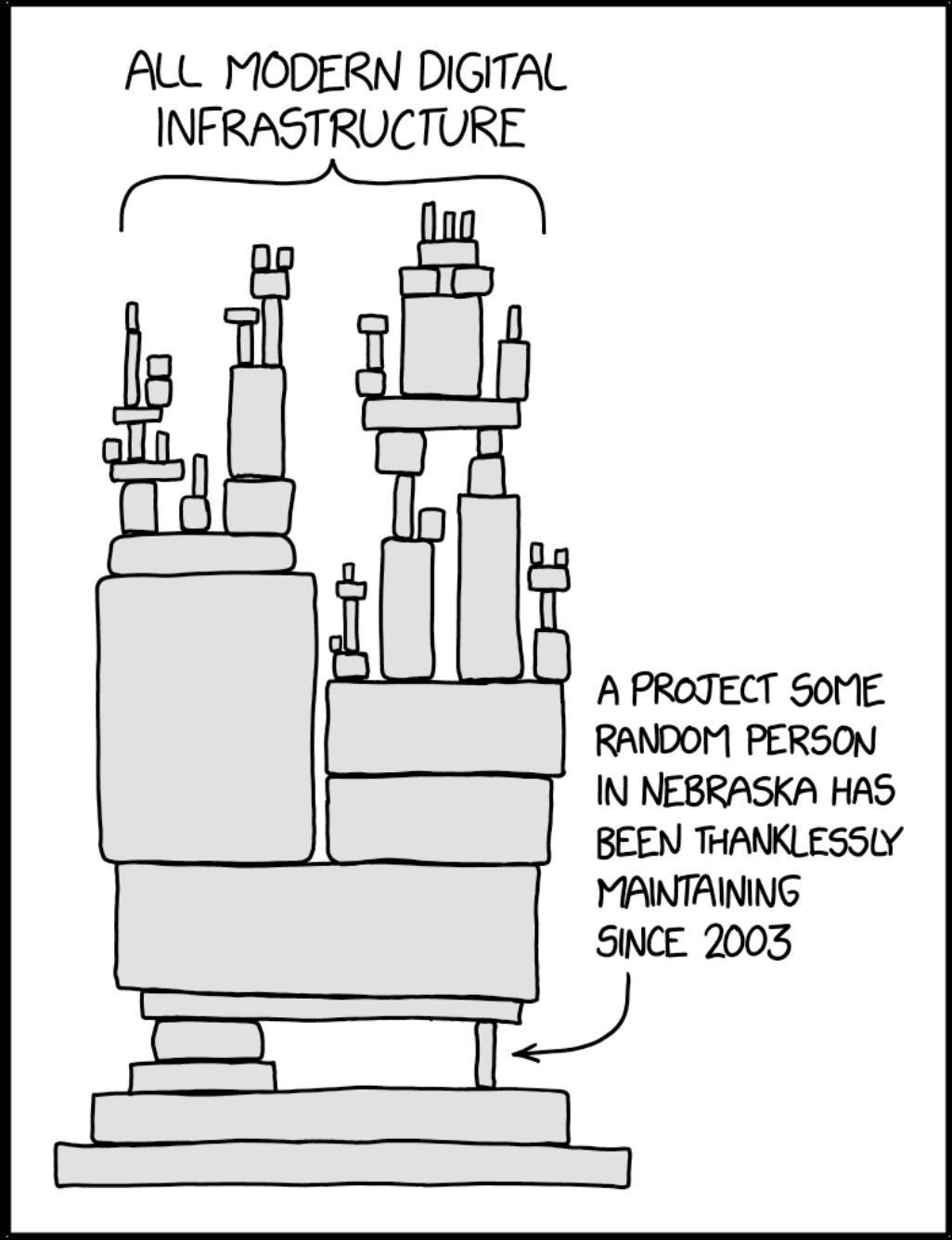
10:24 AM · Nov 28, 2018 · Twitter Web Client

866 Retweets **57** Quote Tweets **4,454** Likes



How Users Think Open Source Projects Are Maintained

How Open Source Projects Are Really Maintained




For Open Source Maintainers
Open Source is not "free as in speech" or "free as in beer"…
Open Source is "free as in toilet"
(credit: Geoffroy Couprie)

Immutable Laws For Maintainers

Open Source is based upon freedom

Freedom reduces leverage and control over usage

Open source project contributors made a decision to give their IP away for free

Users are not obligated to contribute anything back

Contributors are not entitled to receive anything from users

Open Source Myths

1. **Open source projects have unlimited resources**

   Theoretically true… however the reality is that the majority of people simply want to use open source software… very few people actually contribute.

2. **Increasing the number of contributors will help prevent maintainer burnout**

   Opposite is actually true… more contributors = more community support = more burnout

3. **Open source products follow different rules than commercial software products**

   Open source is still governed by standard product/market fit criteria. They require a dedicated product owner to manage the product journey.

4. **There is a business model which satisfies the needs of all open source projects**

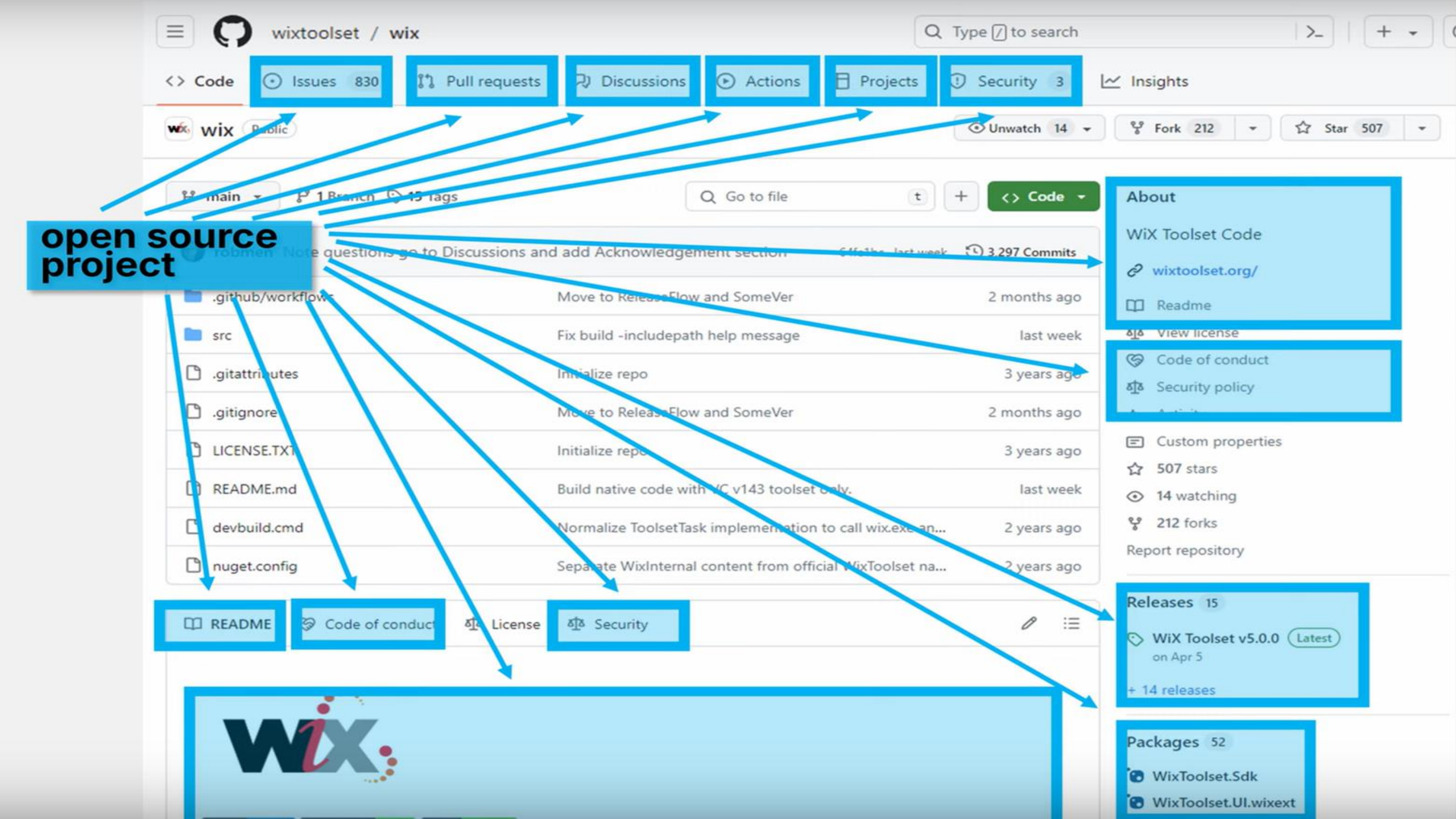
   The sustainability needs and constraints of each open source project are different.

5. **Sustainability is more difficult in the Microsoft ecosystem**

   Every open source ecosystem suffers from sustainability challenges – the "grass is not greener" elsewhere. In fact, monetization is even more difficult in more idealistic ecosystems

6. **Changing a product license is a business model**

   A product license is a small part of an overall business strategy which must include both product and operational aspects to become a viable business


open source project
wixtoolset / wix
Type / to search
Code
Issues 830
Pull requests
Discussions
Actions
Projects
Security 3
Insights
wix
Public
Unwatch 14
Fork 212
Star 507
main
1 Branch
15 tags
Go to file
t
Code
3,297 Commits
.github/workflows
Move to ReleaseFlow and SomeVer
2 months ago
src
Fix build -includepath help message
last week
.gitattributes
Initialize repo
3 years ago
.gitignore
Move to ReleaseFlow and SomeVer
2 months ago
LICENSE.TXT
Initialize repo
3 years ago
README.md
Build native code with VC v143 toolset only.
last week
devbuild.cmd
Normalize ToolsetTask implementation to call wix.exe an...
2 years ago
nuget.config
Separate WixInternal content from official WixToolset na...
2 years ago
README
Code of conduct
License
Security
About
WiX Toolset Code
wixtoolset.org/
Readme
View license
Code of conduct
Security policy
Custom properties
507 stars
14 watching
212 forks
Report repository
Releases 15
WiX Toolset v5.0.0
Latest
on Apr 5
+ 14 releases
Packages 52
WixToolset.Sdk
WixToolset.UI.wixext

Definition of Insanity

"Doing the same thing over and over and expecting different results"

Open Source

"It's sort of like consulting… where clients ask you to create brilliant solutions to really complicated problems… but in this case you don't get paid"

- anonymous

MONETIZING OPEN SOURCE

Maintainers would like to be compensated for the work they perform on open source

WHAT DO PEOPLE PAY MONEY FOR?

1. **Products or services they need which are not available for free.**
2. **If there is a free option, they will pay for additional benefits such as support, features, etc…**
3. **Taxes/Licenses (regulatory compliance)**
4. **Investments (ie. they want equity and/or a return on investment)**
5. **Charitable donations (but often only if it is tax deductible)**

GitHub Sponsors

"Invest in the open source projects you depend on. Contributors are working behind the scenes to make open source better for everyone—give them the help and recognition they deserve."

Donations are generally NOT a sustainable revenue model…

Open Source + Open For Business

Commercial Open Source (COSS)

Approaches:

1. **Proactive** - *Developer creates a commercial open source business and leverages an open source product as a channel for marketing and distribution. (Rare)*

2. **Reactive** - *Developer creates an open source project without a business plan and later decides they want to monetize it. (Common)*

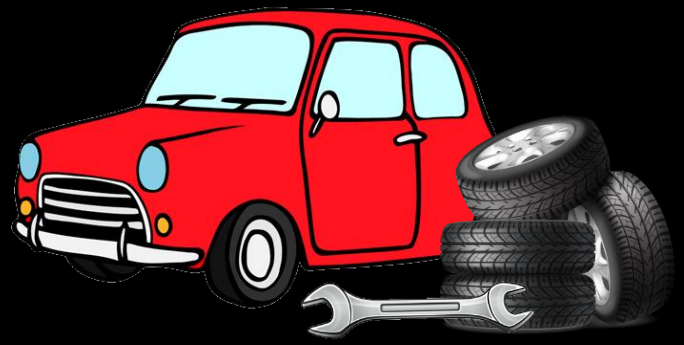
<u>Would you start a business without a business plan?</u>

Open source projects that have no long term plan risk becoming a victim of their own success.

Commercial Open Source Models

Open Source

Extensions/Tools



Marketplace

Professional Services / Sponsored Enhancements

Dual License

Open Core

Managed Service

It is important to note that most Commercial Open Source companies will adopt multiple approached as part of their business strategy.

Commercial Open Source Life Cycle

Life Cycle Stage	Hobby	Community	Professional	Enterprise
Downloads*	10,000+	100,000+	1,000,000+	10,000,000+
Actual Users	100+	1,000+	10,000+	100,000+
Contributors	1+	10+	100+	1000+
FTEs	0	0-1	**1-10+**	**10-100+**

Crossing this threshold is difficult

* vanity metric

$ Donations, Advertising, Grants, Sponsorships
(cash flow - passive revenue, not scalable)

$$ Professional Services, Training
(cash flow - labor intensive, not scalable)

$$$ Dual License, Open Core, Marketplace, Managed Service
(cash flow + equity – often requires startup capital)

$$$$ Unicorn/IPO
(equity – venture capital growth focus)

Success Probability Factors

	0%	+20%
Project Type	Library	Platform
Project Age (yrs)	1-3	3-10
Market Size	Niche	Large
Target Buyer	SMB	Enterprise
Competition	Abundant	Scarce**
	0%	100%

** Having at least 1 enterprise competitor helps validate the market opportunity

Open Source Software Licensing

Open Source Initiative (OSI) - a 501(c)3 California public benefit corporation founded in 1998.

"As steward of the Open Source Definition, we set the foundation for the Open Source Software ecosystem."

The Open Source Definition

Open source doesn't just mean access to the source code. The distribution terms of open-source software must comply with the following criteria:

1. **Free Redistribution**
2. **Source Code**
3. **Derived Works**
4. **Integrity of The Author's Source Code**
5. **No Discrimination Against Persons or Groups**
6. **No Discrimination Against Fields of Endeavor**
7. **Distribution of License**
8. **License Must Not Be Specific to a Product**
9. **License Must Not Restrict Other Software**
10. **License Must Be Technology-Neutral**

Categories:
Permissive – almost no restrictions on usage
Non-Permissive (Copyleft) – some usage restrictions

Open Source Licenses

MIT License

Copyright <YEAR> <COPYRIGHT HOLDER>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Who has the right to define or change the terms of usage

The freedoms for usage offered by this license. Note that other open source licenses such as the GPL may contain more restrictions, which reduces user's rights but may also limit the project's adoption.

Attribution requirements (ie. don't try to claim you created this software)

The business assurances offered are non-existent – it is truly "**use at your own risk**". Organizations needs to assess how critical an open source component is to their overall business. Commercial licenses offer warranties, indemnification, and service level agreements (SLA) to mitigate risk.

Business Source License (BSL)

A "source available" license that allows anyone to view or use the licensed code for internal or testing purposes, but places limitations on commercial use by restricting the ability for other companies to use the IP to create competitive service offerings.

BSL and variations are used by very large COSS vendors such as HashiCorp, MongoDB, Redis Labs, Cockroach Labs, Confluent, Sentry, etc…

NOT Open Source!

Immutable Laws For Consumers

You are taking a dependency on software which you have no control over

Free usage means that you are not entitled to any support, maintenance, or enhancements

You do not have a formal business relationship with the maintainer of the software

Maintainers have the right to change the license at any time


Businesses often need to manage risks related to using open source software
accept
reduce
strategy
RISK
MANAGEMENT
control
transfer
mitigate
avoid

Public Vulnerabilities

Vulnerabilities are disclosed publicly, but it's essential to apply patches quickly to prevent exploitation. Stay vigilant with updates!

Lack of Security Guarantees

Open-source software relies on community support, but not all projects provide formal security guarantees. Be proactive in security management.

Intellectual Property Risks

Navigating complex licenses can lead to legal challenges, especially with "copyleft" clauses that may require sharing modifications to open-source code when redistributed.

Unmaintained Projects

Some projects become abandoned over time, leading to unpatched vulnerabilities. Relying on such software introduces security weaknesses.

Malicious Packages

Attackers can inject harmful code into seemingly legitimate packages. Always vet and monitor your dependencies with trusted tools.

Operational Overhead

Managing open-source components requires careful tracking of versions, licenses, and security updates, adding to operational workload.

Poor Development Practices

Some open-source projects may lack proper documentation, testing, or version control, leading to unreliable software with potential vulnerabilities.

Dependency Risks

Untracked or outdated dependencies can introduce hidden security threats, making regular audits and automated tools crucial.

License Compliance Issues

Non-compliance with open-source licenses can result in legal issues. Understanding and adhering to licensing terms is essential.

Risks With Using Open Source Software

Open source software offers great benefits but also has some risks.

Software Supply Chain Management

Should include all dependencies such as components (e.g. infrastructure, hardware, operating systems, cloud services, etc…), the people who wrote them, and the sources they come from, such as registries, GitHub repositories, etc…

A commercial open source product can provide benefits that mitigate risks

Commercial Open Source Benefits

Commercial license	Companies prefer to have a formal agreements with other companies which protects their interests and mitigate risk. They want a “throat to choke”.
A **tested and certified build**	Companies prefer to use “official” product releases – especially those which have gone through more thorough certification. This also addresses any perceived software supply chain risks.
Management of product **dependencies** and **licenses**	Companies are afraid of dependencies and license compliance. A commercial product should provide documentation of all dependencies and third party licenses.
Prioritized support with Service Level Agreement (SLA)	Companies like having an insurance policy for unexpected complications which may jeopardize business continuity
Private **knowledge base** and communication channel	Companies want prioritized access so their issues can be escalated and resolved immediately
Full product **documentation**	Documentation is a valuable asset which companies are willing to pay for as it helps them offset internal training and support costs

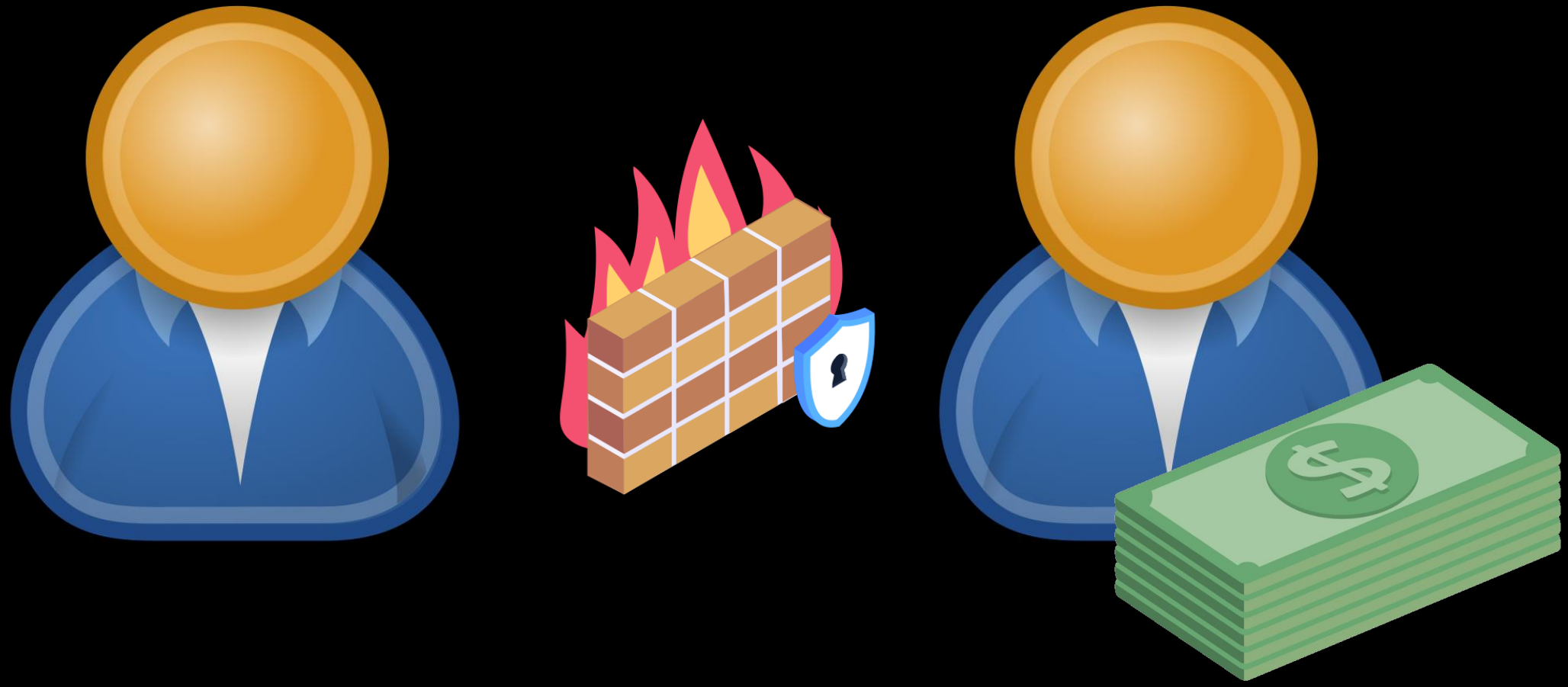
Commercial Open Source Benefits

Notifications of **security updates**	Companies want to ensure their infrastructure is secure and they want to be proactively notified so they can take action
Product updates with notifications of new releases	Companies want to ensure their infrastructure is reliable and they want to be proactively notified of new releases
Product copyright **indemnification**	Companies want to limit their risk and liability related to copyright infringement issues in the product
Commercial **extensions**	Companies may need additional functionality which is highly specialized, valuable, or not applicable to all users.
Integration **consulting services, training, certifications**	Companies may need integration or development assistance with implementation of the product. They may also need training on how to use the product effectively, which may be a requirement for support service eligibility or escalation.

Community Segmentation

Users

Customers



Business Operations

Business Operational Requirements

- **Commercial Entity** – register a company and set up a bank account
- **Insurance** – business insurance and product indemnification insurance
- **Trademark** – IP protection to ensure you can protect and leverage your brand

- **Pricing** – market analysis of how much to charge for your commercial product (don't sell yourself short).
- **License Agreement** – create commercial license agreement (you may need legal assistance)
- **License Enforcement** – product licensing mechanism to restrict product usage

- **Product Usage Tracking** – capture and understand your product installation and usage metrics

- **Marketing** – website to describe your commercial product benefits and generate awareness through all channels – community, search, social, etc…
- **Lead Generation** – capture contact information (ie. email address) from your users. The easiest way is to offer a newsletter and use email automation software.
- **Sales Pipeline** – qualify your leads and nurture them through the sales pipeline using a CRM, email marketing, lead scoring, direct sales engagement, etc…
- **Purchase** – mechanism to allow customers to pay for your product and get provisioned for services

- **Support** – online support ticket system and staffing plan for satisfying your SLA promises

Market Size & Conversion

DotNetNuke

Downloads
8,000,000

Installations
800,000

Licenses
4,000

Conversion
0.05%

Demand Generation is a critical aspect to any successful business.

Open source has the potential to generate tremendous demand.

It is important to recognize that traditional “vanity” metrics have very low conversion ratios (ie. Nuget)

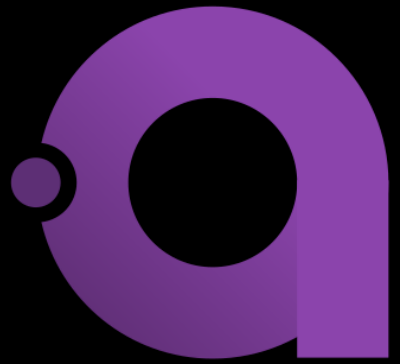
Metrics related to actual usage are far better measures of engagement


DISTRIBUTION
PRODUCTS
BUSINESS MODEL
SERVICES
REVENUE


The challenge is achieving the right balance


To preserve and grow your open source community

ImageSharp



Avalonia

Popular Commercial Open Source (COSS) .NET Projects

MediatR

AutoMapper


The path to open source
sustainability is challenging…

Who Pays For Open Source?

Thank You!

@sbwalker