

Open Source Explained



Prof. Dr. Dirk Riehle, M.B.A.

Univ. Erlangen / Bayave GmbH

2023-04-24 – TUM – Digitale Nachhaltigkeit

Licensed under [CC BY 4.0 International](https://creativecommons.org/licenses/by/4.0/)

Agenda



1. Open-source software
2. Using open-source software
3. The individual perspective
4. The community perspective
5. The commercial perspective
6. The country perspective
7. Digital sustainability

1. Open-Source Software



Software (Computer Programs)

Software

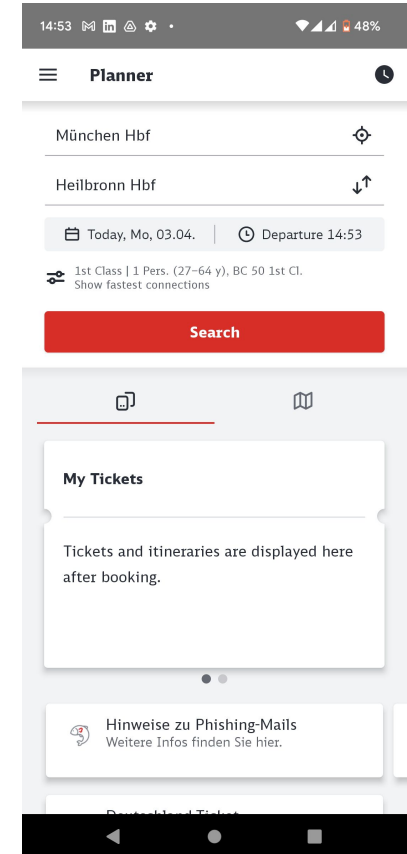
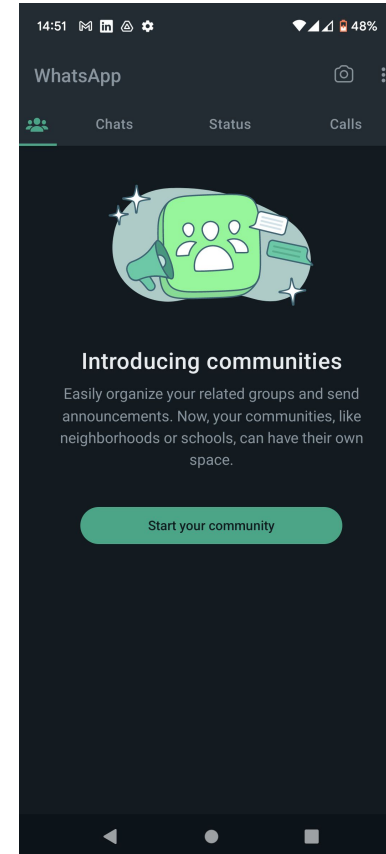
- Is often distributed in binary form
- With defined (limited) usage rights

Source code

- Is the blueprint for programs
- Is the source for binary code

Software vendors

- Sell usage rights to the software
- Usually withhold the source code



What's Wrong With Closed-Source Software?



The consequences of (strong) **vendor lock-in**

1. Costs / fee increases
2. Innovation blockage
3. Operational risk

And many more

Open-Source Software

Open-source software is

- Software given to you under an open source license

An open source license is

- A software license that grants you the right to
 - Use,
 - Modify, and
 - Distribute (modified or not) the software free-of-charge

This is the legal (license) definition [1]



[1] <https://opensource.org/osd>

A Minimal History of Free/Libre, and Open Source Software

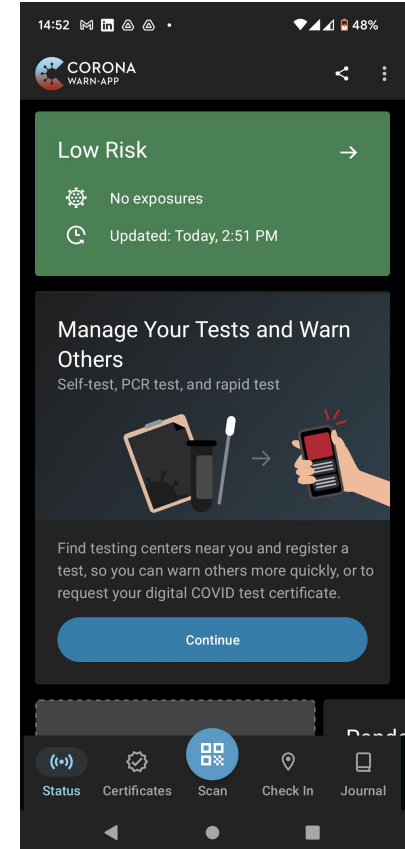


1. Originally, software was bundled with hardware, by-loaded
2. In the 70ties, by court ruling, software had to be unbundled
3. Frustration over closed-source software led to free software
4. Frustration over free software rhetoric led to open source software
5. Since then, open source has been commoditizing software

(Community) Open Source Projects

An open source (software) project is a

- Software component together with a
- Project community of people



History of Traditional Community Open Source Projects

Traditional community projects are

- Unincorporated community open source projects

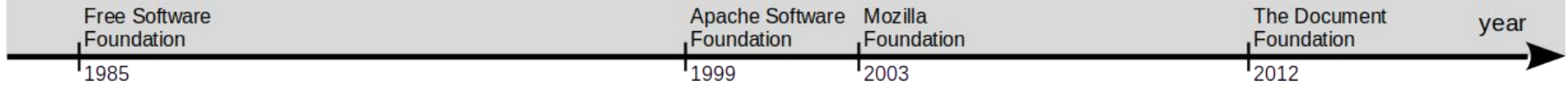


History of Open Source Software Foundations

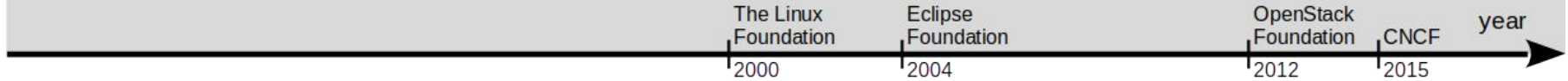
Open source foundations are

- Organizations set-up to manage community open source projects

Traditional open source foundations



Vendor-led open source foundations



User-led open source foundations



History of Commercial Open Source Software

Commercial open source software is

- Software developed by one or more vendors for commercial exploitation

Service and support firms

Cygnus
Solutions
1989

year

Commercial open source distributors

SUSE
1992

Red Hat
1994

Univention
2002

Canonical
2004

year

Single-vendor commercial open source firms

MySQL
1995

SugarCRM
2004

MongoDB
2008

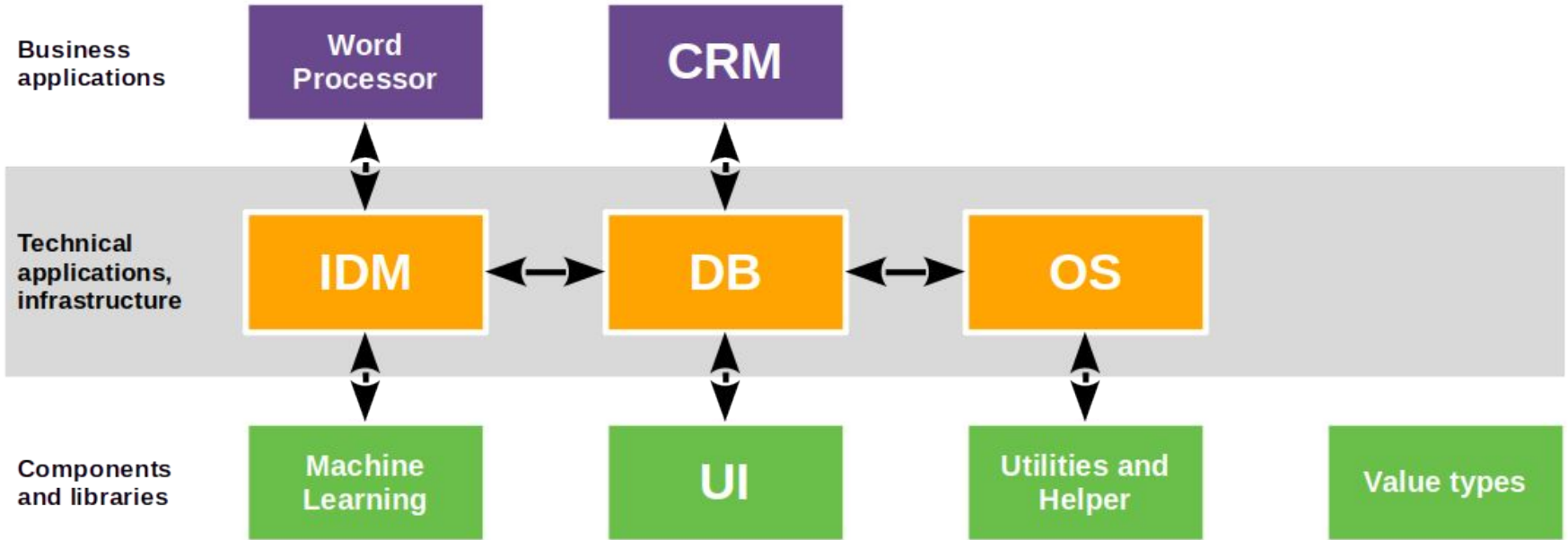
year

2. Using Open-Source Software



(Software Engineering)

All Software is Built From Software Components



U.S. Whitehouse Executive Order 2022-05-12 [1]



U.S. Whitehouse Executive Order on Improving the Nation's Cybersecurity [1]

- Sec. 4. Enhancing Software Supply Chain Security
 - **A purchaser must be provided a software bill of materials**

U.S. Cybersecurity & Infrastructure Security Agency

- Software bill of materials: <https://www.cisa.gov/sbom>

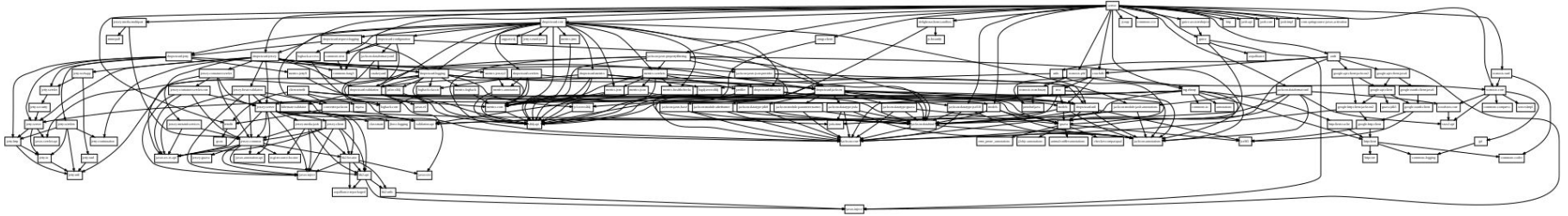
National Telecom and Infrastructure Administration

- <https://www.ntia.gov/SBOM>

[1] See <https://whitehouse.gov/briefing-room/presidential-actions/2021/05/12/executive-order-on-improving-the-nations-cybersecurity/>

Example Dependency Graph

Number of components in transitive closure of graph = 145 for JValue ODS v0.4.0



Software Composition Analysis (SCA)



Software composition analysis (SCA) is the

- **Detailed creation of a software bill of materials** often including
- **Extraction of license compliance artifacts** (copyright notices, license texts)

SCA can be performed to different degrees of detail and purpose

- License text and copyright notice extraction
- Source code snippet matching
- Due diligence

The JQuery License [1] (Based on MIT License Template)

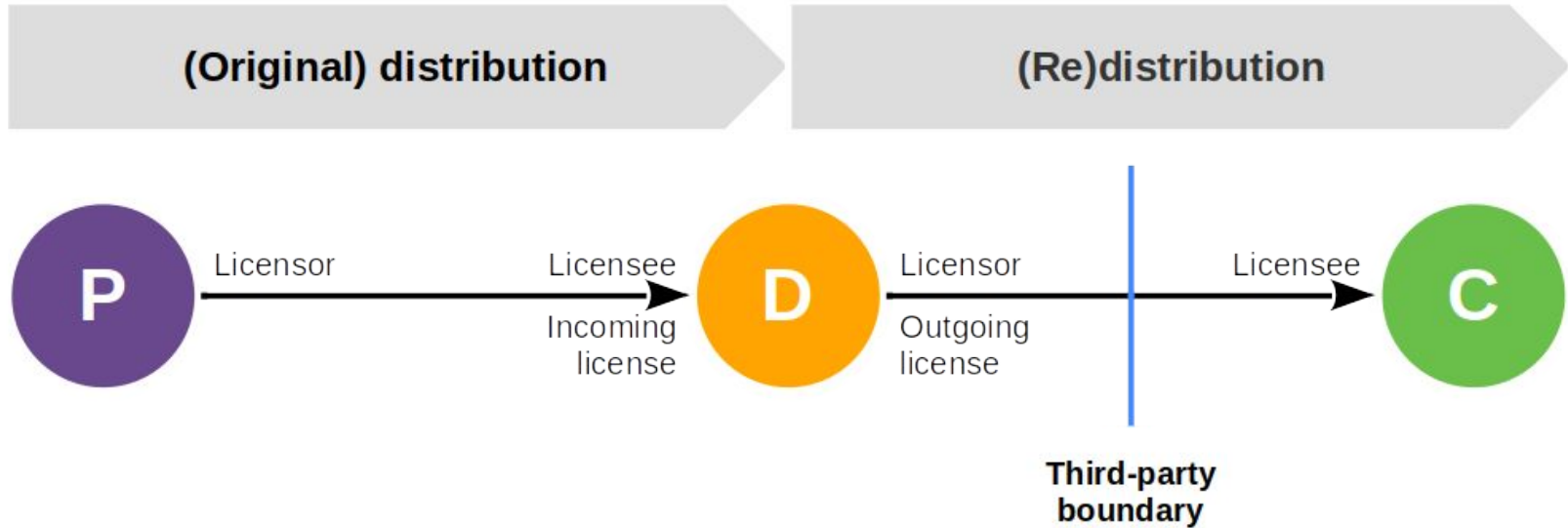
Copyright OpenJS Foundation and other contributors, <https://openjsf.org/>

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:

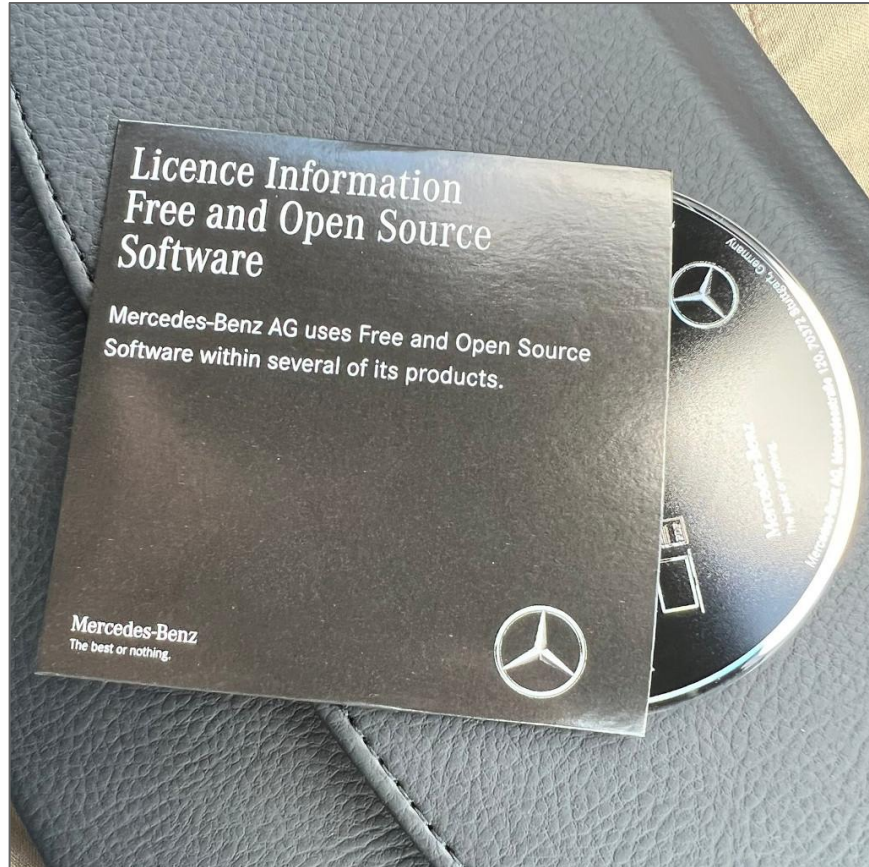
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

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.

Open Source Use in Projects and Products



Open Source Legal Notices for an Infotainment Stack

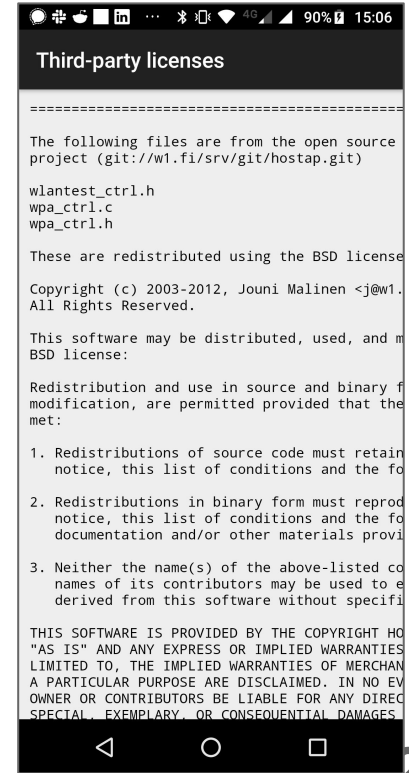
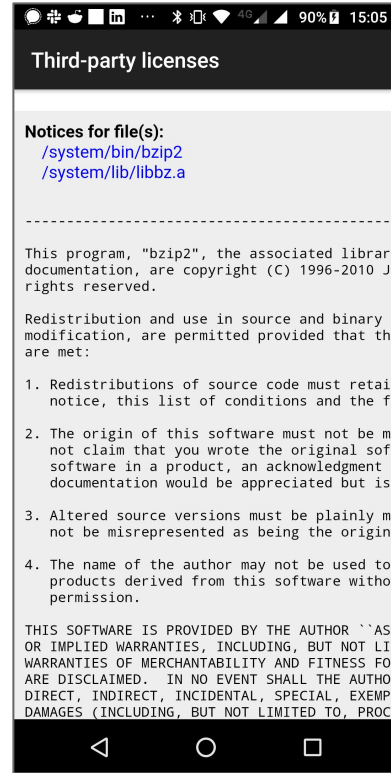
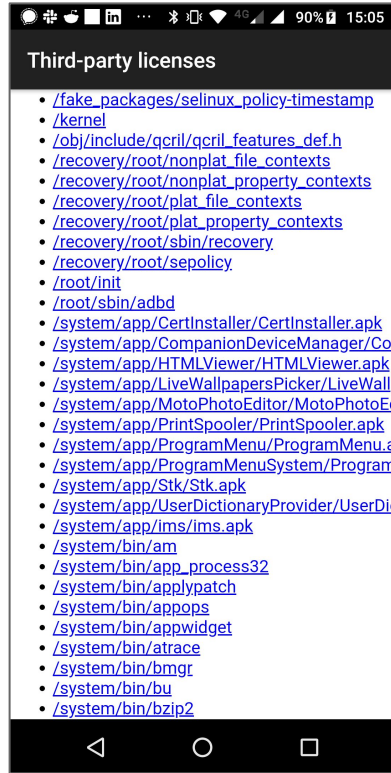
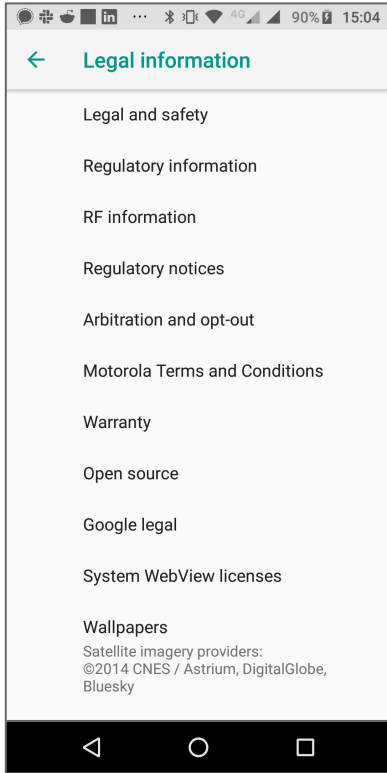


Index

Contents

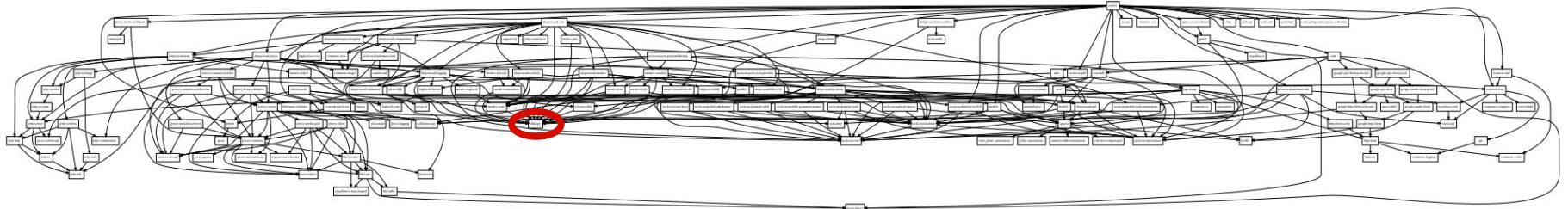
Overview	3
Note	3
Acme Labs BSD License	7
Apache License Version 2.0.....	7
Artistic License	10
BigDigits	13
Boost Software License	13
BSD License	20
BSD 2-Clause	25
BSD 3-Clause	44
BSD 4-Clause	113
BSD 4-Clause (Original).....	179
BSD TCPDUMP License.....	181
BSD Variants	182
Bzip2 License	202
curl License	203
dhcp License	203
Dropbear License	204
expat License	207
ezXML License	207
File	208
Fluendo License	208
FSF MIT License	210
FontConfig License	211
GDChart & gd-libgd	212
genx License	213
GNU GPL v 2.0	213
GNU LGPL v 2.0	1307
GNU LGPL v 2.1	1322
GTween License	1333
ICU License	1334
JasPer License, Version 2.0	1335
KSH License	1336
LibFFI License	1337
libJPEG License	1339
Liboil License	1344
libpcap License	1345
libpng License	1346
LibXSTL License	1353
Lua License	1354
Message-Digest Algorithm License	1354
MIT License	1355
MIT - Variants	1359
Mozilla Public License 2.0	1363
Nominum License	1368
mkqnx6fs license	1368
Oniguruma License	1369
OpenSSH License	1369
OpenSSL License	1383
Original BSD License.....	1393
PHP License Version 3.0.1.....	1393
Pixman License	1394
Radvd License	1396
RIPEMD-160 License.....	1396
SGI Free Software License B Version 2.0 ..	1397
Smic license	1398
Strace License.....	1398
SUN RPC License	1399
The Academic Free License, Version 2.1	1401
The FreeType Project License	1407
The ISC License.....	1424
The ISC - Angelos D. Keromytis License	1425
The ISC License - Variants	1425
Unicode License 2004	1437
Unique Licenses.....	1439
WebM Project License	1444
xinetd License	1445
zlib License.....	1446

Android Open Source Legal Notices



Where Log4J was Hiding

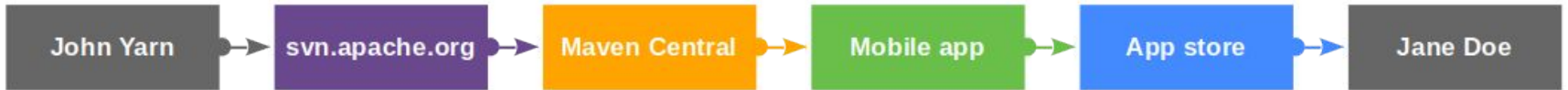
Log4J was hiding behind a configuration parameter for slf4j-api



The Software Supply Chain



Java, e.g. apache-commons



Javascript, e.g. VueJS



3. The Individual Perspective



(Labor Economics)

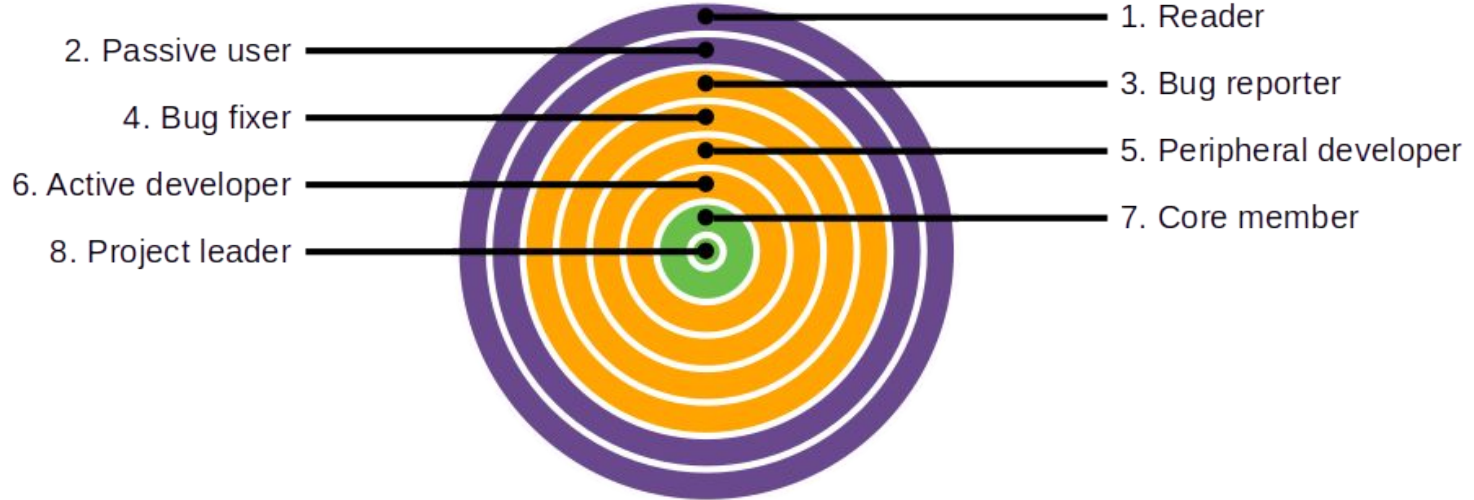
Reasons for Participating in Open Source Projects



Traditional reasons

- Fun
- Help others
- Do something useful
- Learn new skills

The Onion Model of Project Immigration



The Basic Career Ladder of Open Source



Economic Value of Committer Position

Type of work	Signal to employer	Value to employer
Using the software	Technical skills	Lower hiring risk
Contributing to project	Verified skills	Lower hiring risk
Leading the project	Peer validation	Lower hiring risk
Leading the project	Leadership	Marketing, recruiting
Leading the project	Position of power	Influence

Benefits to Employee / Open Source Programmer



1. Higher salary
2. Higher job security
3. Richer job experience

4. The Community Perspective



(Open Source Software Foundations)

Community vs. Vendor-Owned Open Source Software

Community open source software is open source software that

- Has an open governance process
- Is competitively non-differentiating
- Is communally owned (shared copyright)

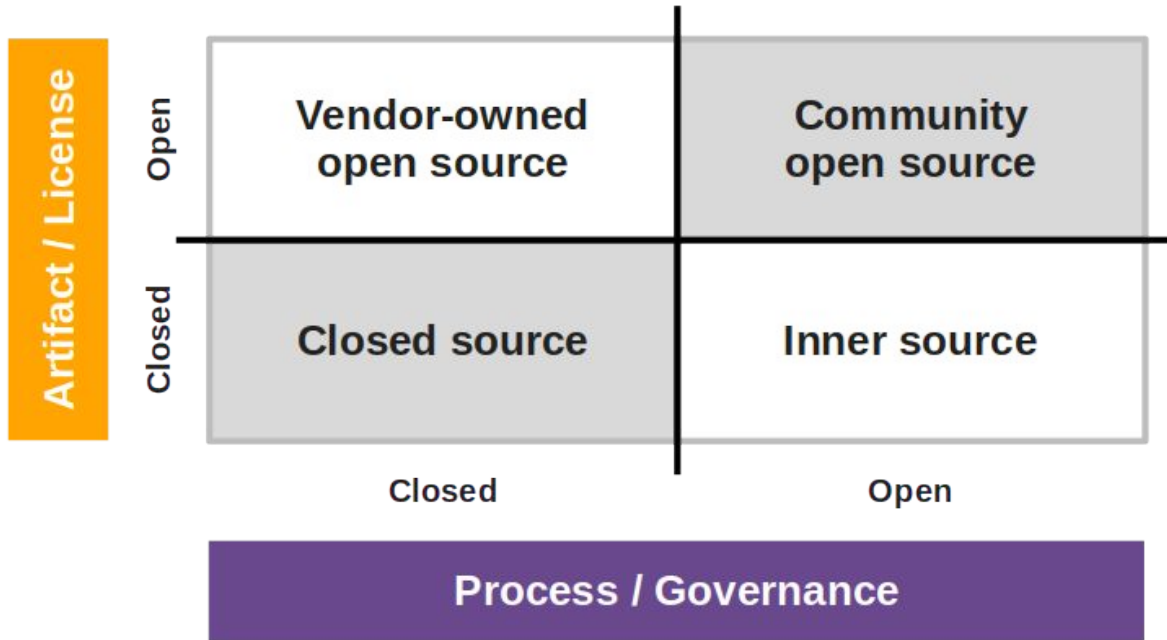


Single-vendor open source software is open source software that

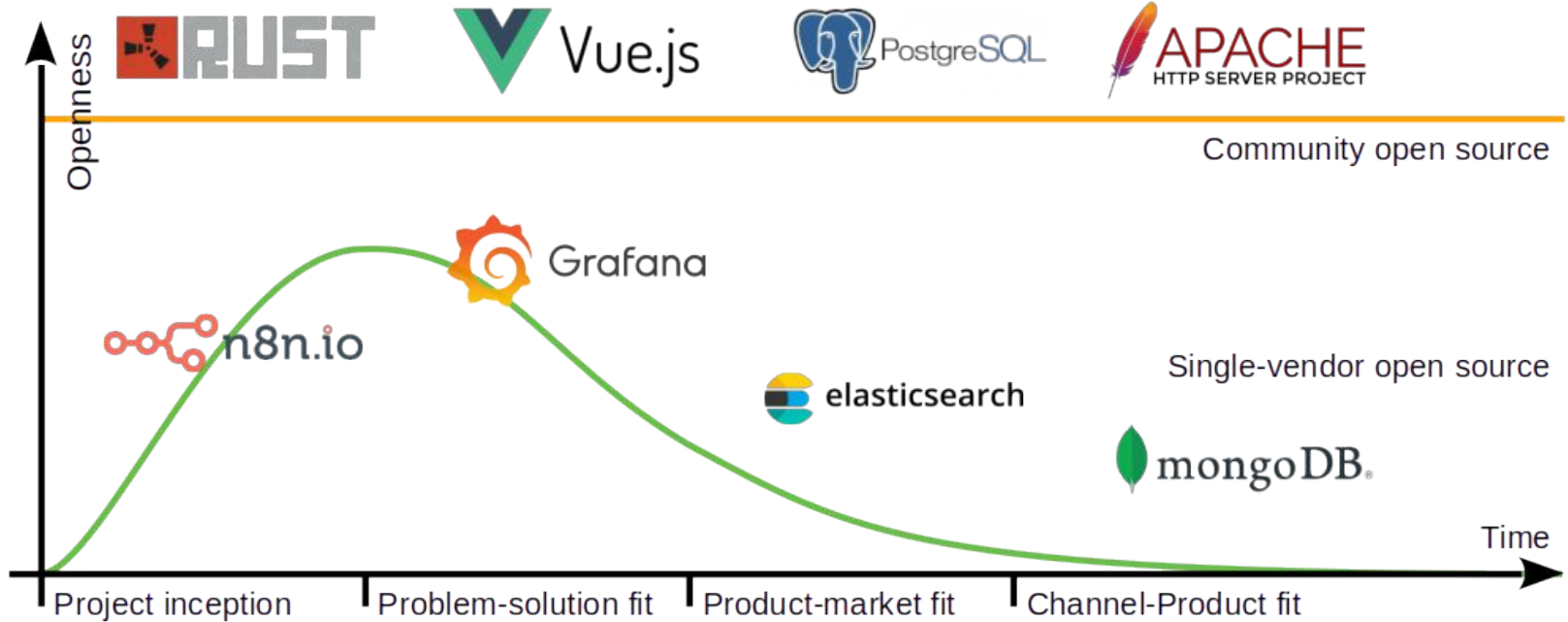
- Has a closed governance process
- Is exploited by a single vendor
- Is owned by that single vendor



Open Source License vs. Governance



Openness Over Time



Open Source Foundations

An open source foundation is

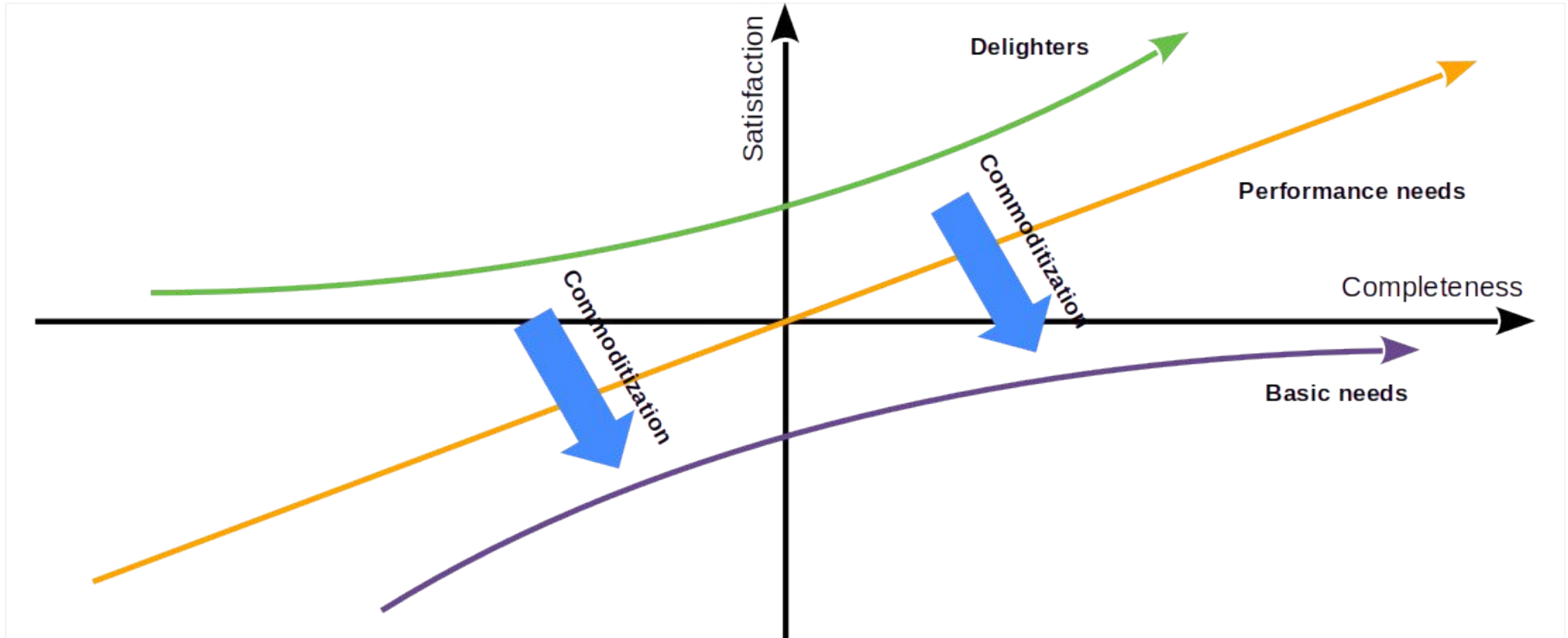
- A **non-profit** organization (foundation, consortium) [1]
- With the mission of **managing** / sponsoring / developing the development of
- **Competitively non-differentiating software**
- Made available to the public as **open source software**

The three classic open source foundations



[1] I'm using the terms foundation and consortium synonymously

Not-Competitively Differentiating



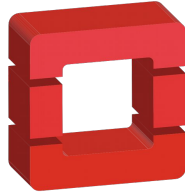
Innovation vs. Commoditization



Benefits of an Open Source Foundation

A (well-designed) open source foundation provides

- A fair and equal playing field for its members
- With clear governance and intellectual property rules



Components For Use in Products

Needed by **product vendors**



Also **developed** by these vendors

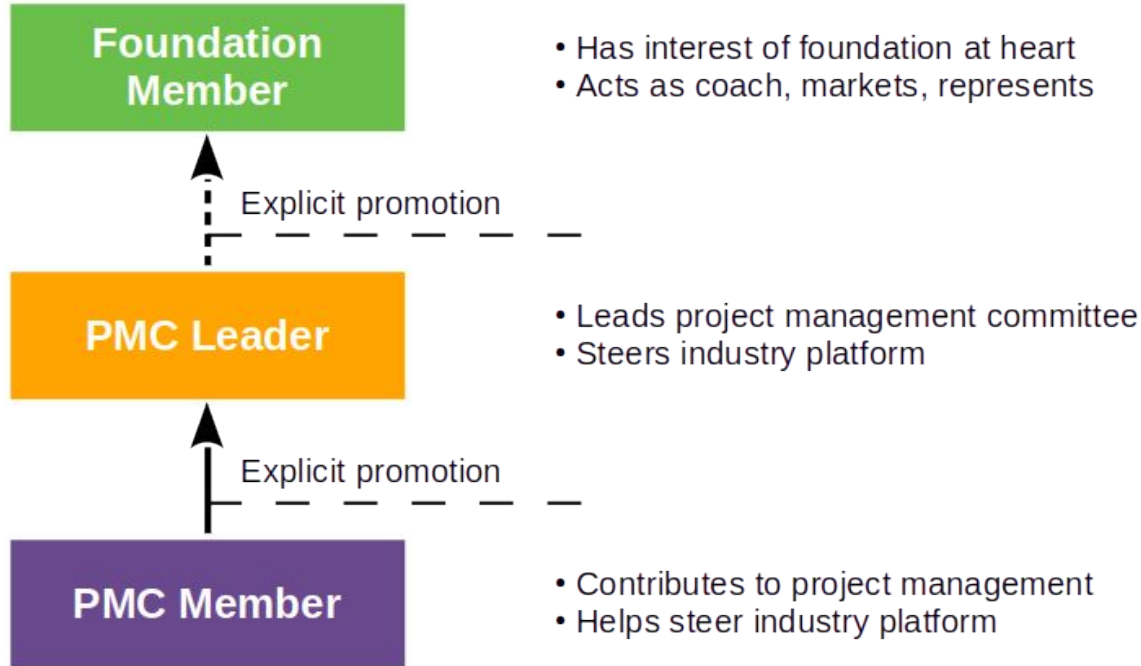
Applications For Use in Business Operations

Needed by **everyone** (in the business domain)



Sometimes **developed**, more often **sponsored** by using businesses

The Extended Career Ladder of Open Source



5. The Commercial Perspective



(Commercial Open Source Firms)

How to Make Money with Open Source? 1 / 2



You can't make money with something that is free.

There is no open source business model.

Open source is a business strategy with many benefits.

You can make money with a closed complement.

How to Make Money with Open Source? 2 / 2

	Open-source software	Closed complement
Hardware	Google Tensorflow	Google's Tensor Processing Units
Software	Apache Lucene	Elastic's Enterprise Search
Operations	PostgreSQL	Credativ PostgreSQL operations
Consulting	Wordpress	Web Design Berlin
Support	gcc	Clever Solutions

Open Source, Vendor Lock-in, and Return on Investment

Complement	Lock-in	Lock-in Base
Hardware	High	(Exclusively owned) intellectual property
Software	High	(Exclusively owned) intellectual property
Operations	Medium	Mixed (IP, Capital, Position)
Consulting	Low	Knowledge
Support	Low	Knowledge

Types of “Open Source” Businesses



1. Consulting and service firms
2. Open-source distributor firms
3. Single-vendor open source firms

Three Generations of Single-Vendor Open Source Firms



The first generation (199x-2002)

- MySQL, Sleepycat Software (BerkeleyDB), TrollTech (Qt), ...
- Focus: Trailblazing / the pioneers

The second generation (2002-2008)

- SugarCRM, MuleSoft, Jaspersoft, ...
- Focus: **On-premise business applications**

The third generation (since 2008)

- MongoDB, Confluent, Redis Labs, ...
- Focus: **Cloud-based infrastructure components**

Benefits of Open Source Strategy by Business Function

Business function	Benefits that accrue
Marketing	Generate leads faster, better, cheaper
Sales	Sell more effectively and efficiently
Business development	Identify partner opportunities better
Product management	Identify new market needs faster, better, cheaper
Software development	Build a superior product faster, cheaper
Product support	Support product at lower cost

Intellectual Property Strategy



Intellectual property rights imperative (of single-vendor open source)

- “Always act in such a way that you, and only you, possess the right to provide the open source project under a license of your choice.” [1]

Use contributor assignment to maintain ownership

- Almost all single-vendor open source firms require copyright transfer for any contributions to maintain full IP ownership [2]

[1] Riehle, D. (2009). [The Intellectual Property Rights Imperative.](#)

[2] All you really need is a relicensing right though

Licensing Strategy for On-Premise Applications [1]

For the commercial product

- A commercial license for on-premise use

For the open source software

- An aggressive copyleft license e.g. AGPLv3

Application



IP Strategy for Cloud Components [1]

For the commercial product

- A subscription-based license for use of the cloud service

For the open source software

- A weak copyleft license e.g. LGPL or APGLv3 + MIT

Component



[1] Mostly used for cloud infrastructure components of the current third-generation firms

Enter the Hyperscalers

Hyperscalers can meet and beat the vendor's value proposition

- Cost efficiency
- Quality of service



Source-Available Licenses



A license that is like an open source license except that

- The licensee loses the usage rights if they compete with the licensor

Example source-available licenses:

- Business Source License (BSL) 1.1 by MariaDB Corporation Ab (2017) [1]
- Server-side Public License (SSPL) by MongoDB Inc. (2018)
- Redis Source Available License (RSAL) by Redis Ltd. (2019)

Also see the Polyform project for a license creation toolbox [4]

[1] See <https://mariadb.com/bsl11>

[2] See <https://www.mongodb.com/licensing/server-side-public-license>

[3] See <https://redis.com/redis-source-available-license>

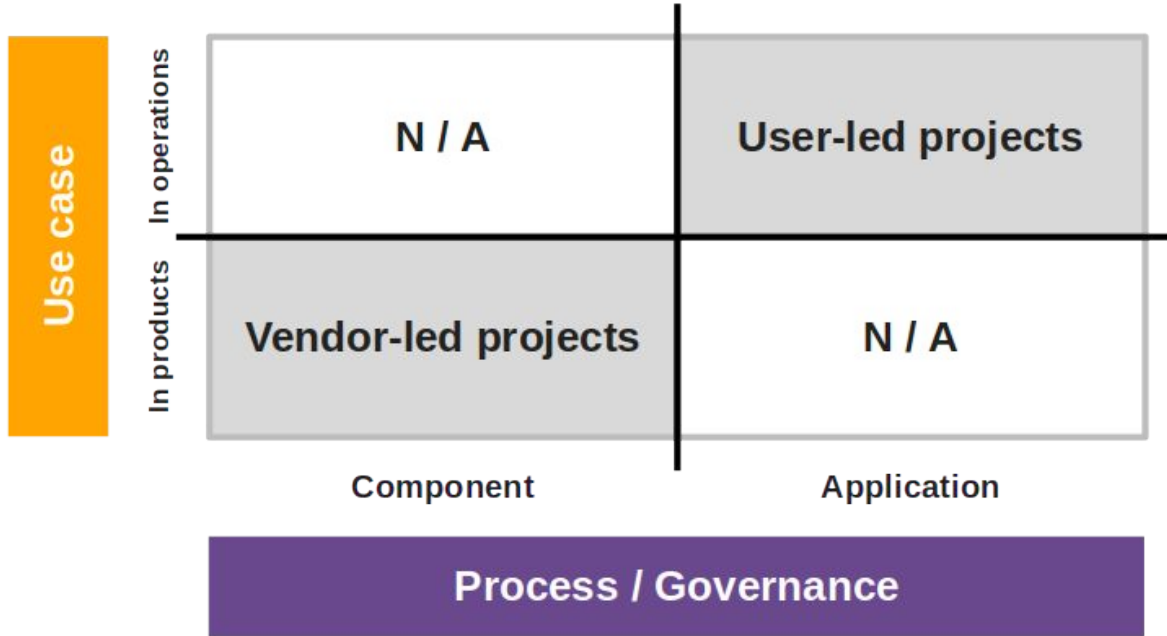
[4] See <https://polyformproject.org/>

6. The Country Perspective



(Country-Level Competitiveness)

Vendor-led vs. User-led Open Source Projects



Example 1: Academy Software Foundation

The Academy Software Foundation is

- A **Linux Foundation collaboration project**
- **Adopting**, managing, and sponsoring open source
- For the production of movies

Premier members are both movie studios and vendors

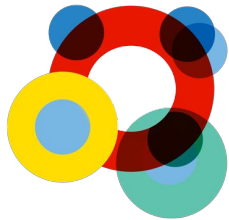


Example 2: Open Logistics Foundation

The Open Logistics Foundation is

- A user-led open source foundation that is
- Managing and developing open source
- For the operations of logistics companies

Members are primarily German logistics firms (for now)



open logistics
foundation



...

Country-Level Competitiveness



Benefits of homogeneity and co-location

- More effective understanding and collaboration
- Short communication paths, information flow
- Faster diffusion of best practices by person

Joining forces in a user-led open source consortium

- Lifts the boats of everyone in the core cluster
- Relative to those who are not in the cluster

A country should help its industries get organized

- To strengthen the countries competitiveness

7. Digital Sustainability



Digital Sustainability and Sovereignty



Benefits of reducing vendor lock-in (recap)

- No / less price pressure
- No innovation blockage
- Lower operational risks

These benefits repeat each other on different levels

- The individual
- Single companies
- Whole consortia
- The country

Summary



1. Open-source software
2. Using open-source software
3. The individual perspective
4. The community perspective
5. The commercial perspective
6. The country perspective
7. Digital sustainability

Thank You! Any Questions?



Prof. Dr. Dirk Riehle, M.B.A.

dirk.riehle@fau.de, <https://oss.cs.fau.de>

dirk.riehle@bayave.com, <https://bayave.com>

dirk@riehle.org, <https://dirkriehle.com>

Twitter: [@dirkriehle](https://twitter.com/dirkriehle), [@profriehle](https://twitter.com/profriehle)

Legal Notices



License

- Licensed under the [CC BY 4.0 International](https://creativecommons.org/licenses/by/4.0/) license

Copyright

- © Copyright 2023 Dirk Riehle, some rights reserved