

开源软件概述

周明辉

zhmh@pku.edu.cn

minghuizhou.github.io



北京大学
PEKING UNIVERSITY

纲要

- 开源软件的定义
- 开源软件的历史和现状
- 开源社区的结构和治理
- 开源软件生态的建设

什么是开源软件？

开源软件

- 开源软件是一种**源代码可以自由获取**的计算机软件。
- 发布开源软件需要附带**开源许可证**：
 - 开源许可证是对开源软件的知识产权进行规范和约束的法律合同：甲方是软件版权所有者，乙方是软件用户。
 - 软件的版权持有人在开源许可证的规定之下允许用户使用、修改以及分发该软件。许可证定义了开源软件用户的权利和义务。
 - 软件知识产权：版权，专利，商标。
- 开源许可证通常符合**开源的定义**的要求。

开源的定义

- 开源不仅意味着可以访问源代码，开源软件的分发条款必须满足下述条件：

- (1) 自由再发行；
- (2) 程序必须包含或方便取得源代码；
- (3) 许可证必须允许更改和派生程序；
- (4) 保护作者源代码的完整性；
- (5) 无个人或团体的歧视；
- (6) 无领域歧视；

.....

- (8) 许可证不能限制其他软件；
- (10) 许可证需要是技术中立的。



<https://opensource.org/osd>

The Open Source Definition

-- <https://opensource.org/osd>

Introduction

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

The license shall not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. The license shall not require a royalty or other fee for such sale.

2. Source Code

The program must include source code, and must allow distribution in source code as well as compiled form. Where some form of a product is not distributed with source code, there must be a well-publicized means of obtaining the source code for no more than a reasonable reproduction cost, preferably downloading via the Internet without charge. The source code must be the preferred form in which a programmer would modify the program. Deliberately obfuscated source code is not allowed. Intermediate forms such as the output of a preprocessor or translator are not allowed.

3. Derived Works

The license must allow modifications and derived works, and must allow them to be distributed under the same terms as the license of the original software.

4. Integrity of The Author's Source Code

The license may restrict source-code from being distributed in modified form only if the license allows the distribution of "patch files" with the source code for the purpose of modifying the program at build time. The license must explicitly permit distribution of software built from modified source code. The license may require derived works to carry a different name or version number from the original software.

5. No Discrimination Against Persons or Groups

The license must not discriminate against any person or group of persons.

6. No Discrimination Against Fields of Endeavor

The license must not restrict anyone from making use of the program in a specific field of endeavor. For example, it may not restrict the program from being used in a business, or from being used for genetic research.

7. Distribution of License

The rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties.

8. License Must Not Be Specific to a Product

The rights attached to the program must not depend on the program's being part of a particular software distribution. If the program is extracted from that distribution and used or distributed within the terms of the program's license, all parties to whom the program is redistributed should have the same rights as those that are granted in conjunction with the original software distribution.

9. License Must Not Restrict Other Software

The license must not place restrictions on other software that is distributed along with the licensed software. For example, the license must not insist that all other programs distributed on the same medium must be open-source software.

10. License Must Be Technology-Neutral

No provision of the license may be predicated on any individual technology or style of interface.

Last modified, 2007-03-22

开源许可证需要符合开源的定义的要求

- 开源社区存在大量不同类型的许可证，OSI 认证的开源许可证已有 126 个^[1]，在开源项目中的使用率 > 80%。
 - 宽松型，Apache/MulanPSL，关键特点：分发可以闭源
 - 传染型，GPL/MulanPubL，关键特点：分发必须开源

木兰宽松许可证(宽松型许可证): <http://license.coscl.org.cn/MulanPSL2>

木兰公共许可证(传染型许可证): <http://license.coscl.org.cn/MulanPubL-2.0>

均采用中英文双语表述，以中文为准

【1】 https://spdx.org/licenses/?ivk_sa=1024320u, 2021/03/07

木兰许可证系列：中英文双语

大背景：

本土企业需求，中文开源社区的发展和成长需求。

具体需求：

中文解释权，少/无风险的开源项目和产品的发展需求。

第一个获得OSI认可的本土开源许可证：木兰宽松许可证MulanPSL2

5 Aug 2019

MulanPSL-
1.0 发布

14 Feb 2020

MulanPSL-2.0
OSI 认证

April 2020

Apache 基金会宣布
Apache2.0 跟 MulanPSL-2.0
兼容

23 March 2021

OSG-Japan 翻译并发布
mulanPSL-2.0 日文版

国内社区广泛支持



gitee



确实社区



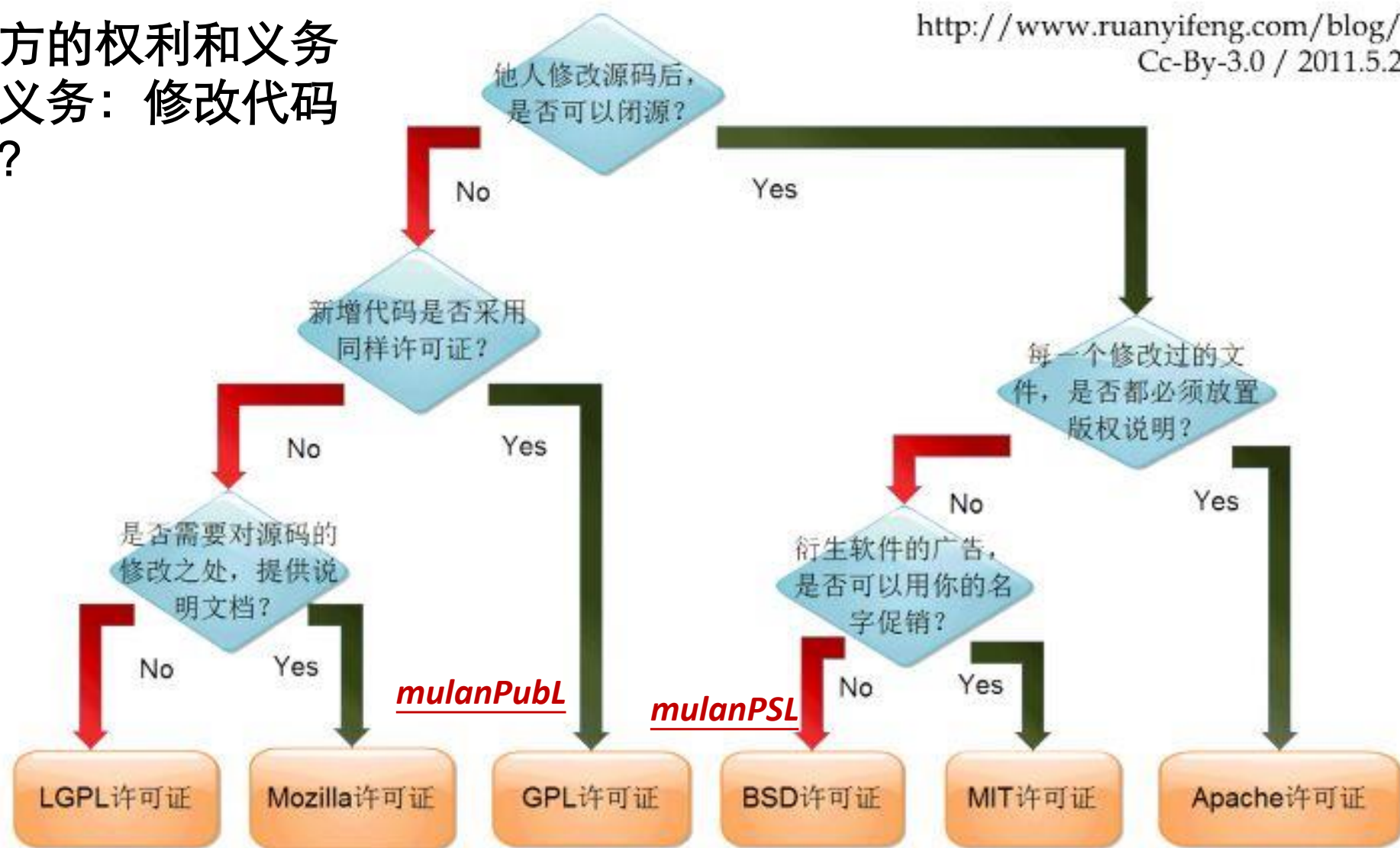
iHub



50k+ 项目采用了 MulanPSL-2.0，覆盖云计算、大数据、AI、OS 等，例如，openEuler，openGauss，香山(RISC-V处理器)。

不同开源许可证的使用

- 合同：定义乙方的权利和义务
- 主要区别在于义务：修改代码是否需要开源？



开源软件的历史和现状

开源软件的历史：创新的历史

大公司参与和主导开源社区，**混合模式**兴盛；开放众包、群智**创新**，开源延展到硬件、教育等领域

开源 (Open Source) 一词在1998年2月3日提出，在1998年开源峰会达成共识，开源**正式成为旗帜**，开源运动迅猛发展，影响力迅速扩大

Bill Gates推动**软件商业化**，软件单独售卖，源码不再免费可得；R.M. Stallman发起**自由软件free software**运动；各种开源社区兴起

软件商业化尚未出现或尚未成熟，软件跟硬件搭售，随源代码发布，学术共同体**自发开放和共享源代码**

多元生态发展阶段
(商业模式多元化)

原始萌芽阶段



1980

多家争鸣阶段



1998

共识达成阶段



2005

融合发展阶段

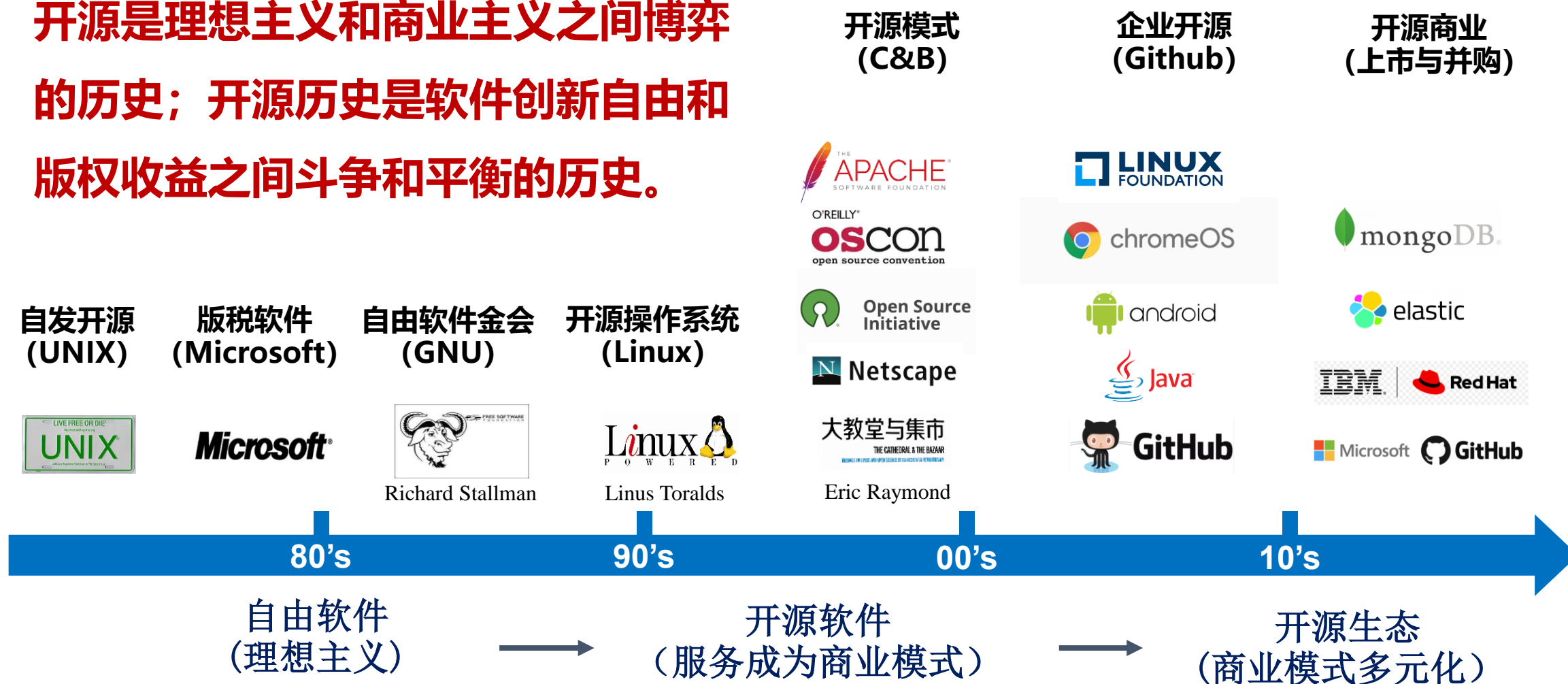


2021

软件从跟硬件搭售到单独售卖，是**专有**推动了创新；
从商业许可发展到开源许可，是**共享**推动了创新；
从开源发展到混合模式，是**共享/专有的融合**创新

开源软件的历史：自由和商业博弈的历史

开源是理想主义和商业主义之间博弈的历史；开源历史是软件创新自由和版权收益之间斗争和平衡的历史。



开源软件的历史：英雄们的历史



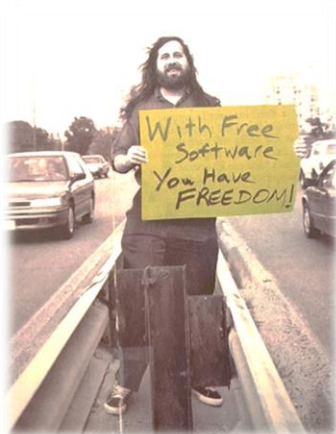
0. 1970年 启蒙年代

- Ken Thompson 和 Dennis Ritchie 缔造 Unix



1. 1984年 自由软件运动

- AT&T 将UNIX商业化
- Bill Gates的微软发布DOS和Windows
- Richard Stallman 发起GNU项目



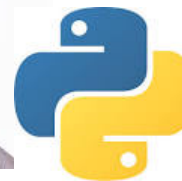
2. 1991年 Linux 内核系统

- Linus Torvalds 发起并发动社区
- 完善了GNU项目



3. 1998年 开源峰会

- Netscape 宣布开放Navigator浏览器的源代码
- “开源软件”替代“自由软件”，并广泛传播






4. 著名开源软件项目

- Larry Wall, perl, 1987
- Guido V. Rossum, Python, 1991
- R. Ihaka & R. Gentleman, R, 1993

自由软件

开源软件的诞生与发展

开源软件的商业模式

商业模式	简介	特点	代表企业
Support 支持服务	<ul style="list-style-type: none">用户只需为技术支持及咨询服务买单	<ul style="list-style-type: none">人工外包作，利润率偏低工作可复制性低，scale较难客户转换率低，通常<1%	 
Hosting 托管	<ul style="list-style-type: none">供应商将其开源软件作为服务托管在云上，通过收取每月/每年的托管和服务费获利	<ul style="list-style-type: none">该模式成为了部分云厂商打包开源项目赚取利润的途径	 
Restrictive Licensing 限制性许可	<ul style="list-style-type: none">通过提供一个带有稍带限制的开源许可证来激励使用者进行付费	<ul style="list-style-type: none">许可证定义模糊，需要法院判决部分公司禁止使用该商业模式下的开源软件	 
Open-core 开放核心	<ul style="list-style-type: none">该模式下的大部分代码是开源的，而少数代码（针对企业用户）是专有的，需要收费专有部分可以打包成与开源基础部分连接的单独模块或服务，或者在分叉版本中分发	<ul style="list-style-type: none">该模式可以避免云厂商打包开源项目赚取利润难以拿捏开源范围的尺度很难将代码中的开源与专有特性完全分离	 
Hybrid Licensing 开放核心+混合许可	<ul style="list-style-type: none">最新的模式，在开放核心基础上进行了改进混合许可在同一个代码库中混合了开源代码和专有代码用户可以选择只使用开源代码，或者同时使用开源代码和专有软件代码	<ul style="list-style-type: none">代码在同一个代码库中，使管理和开发变得更容易允许用户方便升级到付费模式允许外部社区（比如GitHub）成员对专有软件功能模块进行改进	 

目前，开放核心+混合许可逐渐成为主流的商业模式，其原因在于：

- 开源软件商能够轻松管理代码库而不必拿捏开源的尺度
- 客户能够方便的从免费开源模式切换到付费模式（不需要额外部署，也不需要和销售沟通）
- 外部的开源社区也能对专有付费模块代码进行改进，降低了开发成本

来源：云启资本

今天的国际开源状况

全民开源

- GitHub
 - 2021 年新增代码仓库数量为 **6,100 + 万**，2020 年6,000+万，2019年4,400+万，2018年3,200+万。
 - 2021年开发者数量为**7,300+万**，2020年为5,600+万，2019年为4,000+万。
- Gitee
 - 整体托管代码仓库超过 **1500 万**
 - 整体用户数量超过 **600 万**

企业参与

Redhat调查了全世界1250位全球IT领导者，**90%正在使用企业开源软件**，其中IT基础架构现代化64%、应用开发54%、数字化转型53%

《**2021年企业开源现状**》，Redhat, 2021.3

Synopsys分析了17个行业1,500+个代码库，其中**98%**包含了开源代码。平均每个应用包含的开源组件数量**5年内增长259%**

《**2021年开源安全与风险分析报告**》，Synopsys, 2021.4

中国的开源现状

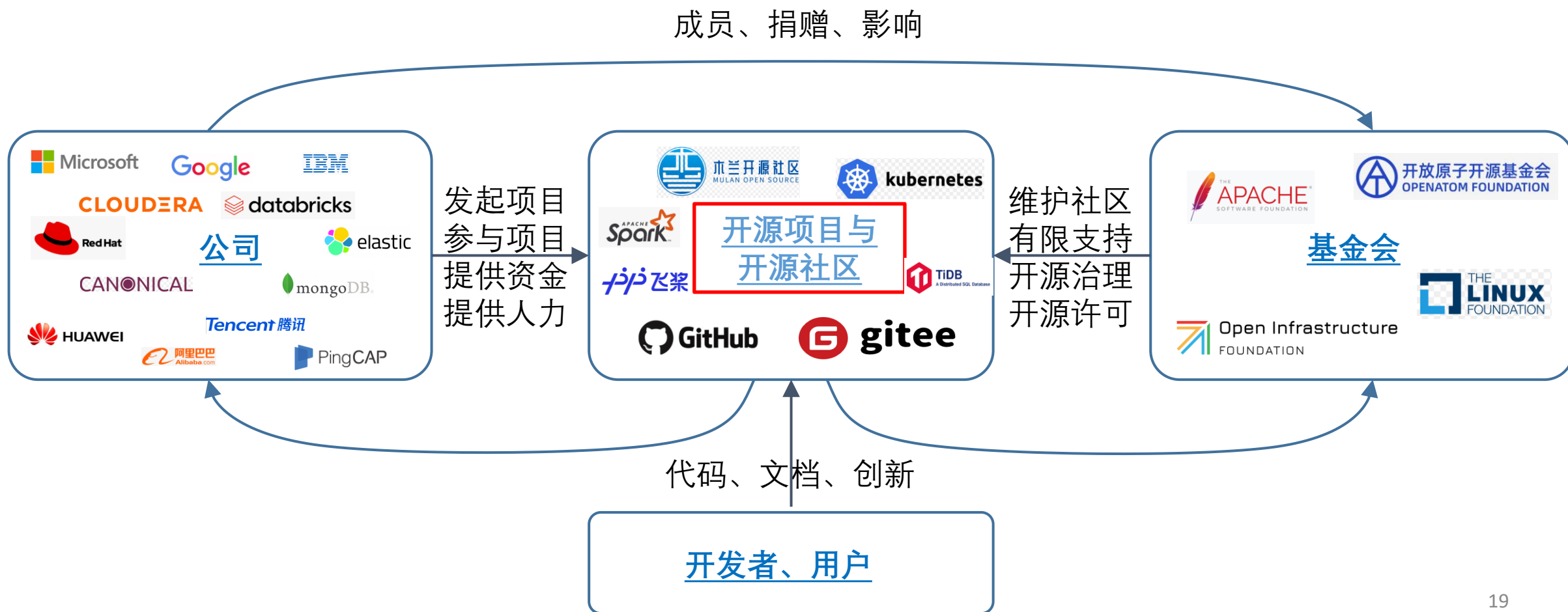
- 本土开源繁荣发展，渐入佳境
 - 优秀开源项目
 - 操作系统：openEuler, openHarmony
 - 数据库：TiDB, OceanBase, openGauss,
 - 人工智能：paddlepaddle, mindSpore,
 - 应用技术：Vue.js, echarts
 - 开源基础设施
 - 多个代码托管平台：Gitee, GitCode, GitLink
 - 第一个OSI 批准的中英文开源许可证：MulanPSL2.0 – 被50k+个项目采用
 - 一批开源软件独角兽或准独角兽创业企业，如PingCAP、Kyligence等，获得了风险投资的认可
- 蓬勃发展的开源社区和开源促进会
 - 第一个开源基金会：开放原子开源基金会
 - 学术共同体和民间：CCF开源发展委员会、木兰社区、开源社
 - 企业：开源雨林（华为）、腾源会（腾讯）、龙蜥社区（阿里）、蚂蚁开源
 - 军队：红山开源社区

中国的开源：机会和挑战并存

- 国家战略： 开源创新被列入十四五规划和2035远景目标
 - “支持数字技术开源社区等创新联合体发展，完善开源知识产权和法律体系，鼓励企业开放软件源代码、硬件设计和应用服务。”
- 本土的优势： 市场大（多样化需求）、人才多（人才储备强）、产业（供应链）体系完整。
- 存在的不足：
 - volunteering文化缺乏，用户创新缺乏，协作共赢欠缺，难以形成生态。
 - 以竞争、主导为出发点的“开源”仍然不少。
 - 缺乏商业模式。

开源社区的结构与治理

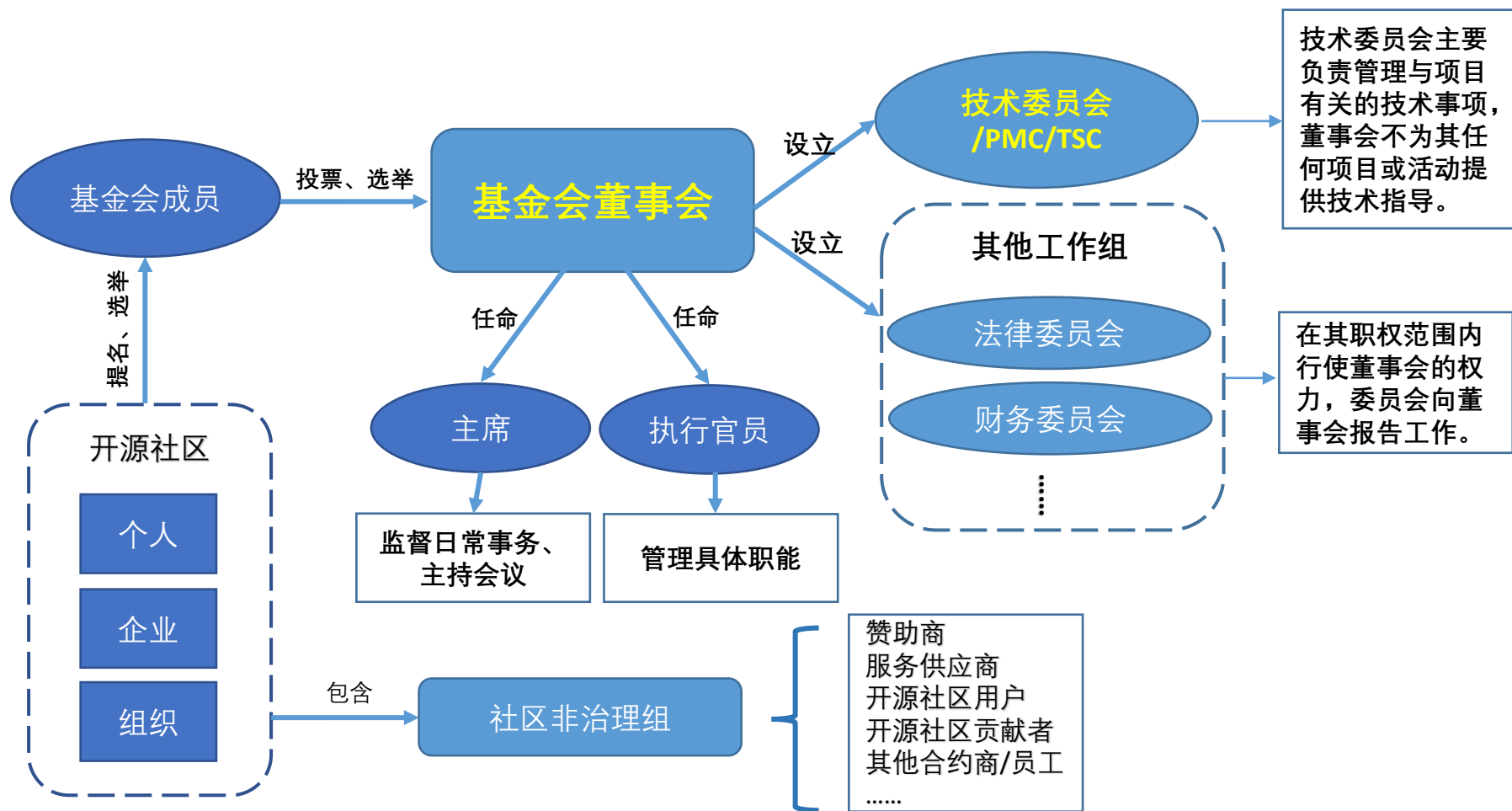
开源生态的基本结构与要素



开源基金会的角色

- 任何人都可以在Internet上发布项目的源代码，但是要构建一个可持续的项目社区，仅靠代码是不够的，而基金会就扮演了“社区看门人”的角色。
- 开源基金会作为管理和推广开源项目的非盈利机构，为开源项目提供基础设施、活动、培训以及法律、商业、技术等服务。

开源基金会的组织架构



Apache基金会(Apache Software Foundation, ASF)



成立于1999年

涉及大数据、云、搜索和CMS、DevOps和构建管理、物联网和边缘计算、机器人和深度学习、服务器、Web框架等领域。



一个项目要进入Apache基金会，需要按照基金会的要求进行“孵化”。Apache孵化器就是为那些想要进入Apache基金会的项目提供服务的，孵化一般需要一年半的时间，满足一系列质量要求之后方可毕业，通过孵化毕业的项目要么成为顶级的ASF项目，要么成为其他顶级项目的子项目。



中国计算机学会开源发展委员会：CCF ODC



平台研发

以开源模式打造开放共建的CCF开源服务平台，包括开源项目开发孵化平台、开源人才培养服务平台，为产教研深度融合提供全方位支持

社区建设

探索建立CCF开源项目孵化机制，为产学研领域成果孵化提供全方位服务
推进开源人才培养，形成产学研联动培养模式

人才培养

联接汇聚科教资源、产业资源和社会资源等，形成产、教、研联动的人才培养模式，推进开源课程资源建设与共享，培养优秀创新人才

木兰开源社区

木兰开源社区：对国家科研成果开源托管、产业界中小微开源项目开展孵化运营

● 汇聚了专项中22个项目的170余项开源成果

<div>首页 开源软件 资讯 活动 开源许可证 软件工程云服务 社区内容推荐 木兰确实托管平台 大家都在搜...</div>		
<div>开源社区信息监控与推荐系统 (Primrose) - 东南大学 开源生态系统监控与推荐平台 云计算 MulanPSL-2.0 python</div>	<div>智能云开发辅助机器人 (Cosine Robot) - 东南大学、南京大学、山东... 贯穿软件开发全生命周期的辅助机器人 MulanPSL-2.0 云计算 python</div>	<div>AppPop: 软件流行度预测工具 - 西北工业大学 针对影响软件流行度的多种因素建模, 预测软件的流行度 数据处理 MulanPSL-2.0</div>
<div>X-ADMM: 软件情景自适应工具 - 西北工业大学 根据终端设备的情景信息, 对于软件模型通过剪裁、分割等方法进行动态调整, 从而实现软件与设备的自适应... 数据处理 MulanPSL-2.0</div>	<div>CompEvo: 构件级在线更新系统 - 南京大学 面向Java web应用的可独立为软件提供服务的构件级级在线更新系统 云原生 MulanPSL-2.0</div>	<div>Hanabi: 软件缺陷自动修复工具软件 - 北京大学 给定一个有缺陷的程序及揭示缺陷的测试, 本项目全自动化地尝试修复缺陷 数据处理 GPLv3</div>
<div>TreeGen: 面向功能增强的程序代码自动生成工具 - 北京大学 给定一串自然语言描述, 本项目通过人工智能相关技术自动化生成程序代码。 数据处理 MulanPSL-2.0</div>	<div>Relax: 资源竞争检测和消解工具 - 国防科技大学 以配置为出发点, 通过监控系统资源状态并调整与资源相关的配置项, 从而消解资源冲突, 提高软件执行效率 数据处理 MulanPSL-2.0</div>	<div>MBCGO: 基于模型转换的可成长自主无人系统 & 配置项成长寻优 - 北京航空航天大学 基于模型基的可成长自主无人系统, 自动调整配置项参数, 适应任务、资源、环境的要求 数据处理 MulanPSL-2.0</div>
<div>ciru: 容器Checkpoint/Restore机制 - 中国科学院软件研究所</div>	<div>Ripple: 大规模容器集群配置系统 - 中国科学院软件研究所</div>	<div>Javelus: Java软件在线更新系统 - 南京大学 基于OpenJDK实现的工业级Java软件在线更新系统</div>

● 重点孵化12项科教界和产业界开源项目

项目名称	贡献单位	项目介绍	许可证	时间	类别
Kube-OVN	灵雀云	Kube-OVN是一款由灵雀云自主研发的开源企业级云原生Kubernetes容器网络编排系统。	Apache 2.0	2020.12.30	企业成果
PiFlow	中科院计算机网络信息中心	PiFlow是一个基于分布式计算框架技术开发的大数据流水线处理与调度系统。该系统将大数据采集、清洗、存储与分析进行抽象和组件化开发, 以所见即所得、拖拽配置的简洁方式实现大数据处理流程化配置、运行与智能监控。	Apache 2.0	2021.6.9	科技成果
PostMan	华中科技大学	PostMan是一个网络功能中间件, 通过高效的按需组包与批处理, 能够快速地缓解突发流量所引起的后端服务性能骤降问题。	MulanPSL-1.0	2021.6.9	科技成果
skyline	浪潮	Skyline是对标OpenStack社区Horizon项目, 在易用性、页面性能等方面进行深度优化, 提供简单、易用、高效的OpenStack控制台。	MulanPSL-2.0	2021.6.9	企业成果
OceanBase-Client	蚂蚁集团	OceanBase Client (简称 OBClient) 是一个基于 MariaDB 开发的客户端工具。您可以使用 OBClient 访问 OceanBase 数据库的集群。	GPL	2021.6.9	企业成果
SRS	杨成立	SRS是一个简单高效的实时视频服务器, 支持RTMP/WebRTC/HLS/HTTP-FLV/SRT/GB28181, 应用于直播和WebRTC等互联网视频场景。	MIT	2021.6.9	个人
zCore	清华大学	zCore操作系统是基于Rust语言编写的新一代操作系统。	MIT	2021.7.13	科技成果
DADI	阿里云	DADI 是 Data Accelerator for Disaggregated Infrastructure 的缩写, 旨在为计算存储分离架构提供各种可能的数据访问加速技术。	Apache2.0	2021.7.13	企业成果
LinkWeChat	江冬勤	LinkWeChat 是一款基于人工智能的企业微信 SCRM 系统, 为企业构建私域流量营销系统的综合解决方案, 助力企业提高社交客户运营效率。	GPL	2021.7.13	个人
Furion	百小僧	Furion 是基于 .NET5/6 平台开发的 C# 底层开发框架, 支持 Web、控制台、IoT等领域开发, 框架设计理念是让 .NET 开发更简单, 更通用, 更流行。	MulanPSL-2.0	2021.8.10	个人
建木	九州云	建木自动化平台以触发器、流程编排、任务分发等功能为平台核心, 可以用在各类使用场景下, 包括但不限于, CI/CD、DevOps、自动化运维、多业务系统集成等使用场景。	MulanPSL-2.0	2021.9.9	企业成果
Open-Digger	华东师范大学	OpenDigger 是由 X-lab 发起的一个开源数据分析报告开源项目, 这个项目旨在凝聚全球开发者的智慧共同对开源相关数据进行分析统计, 以使开发者可以更好的理解和参与开源。	Apache2.0	2021.11.9	科技成果



涉及音视频、云原生、网络、大数据、操作系统等领域

开源生态的建设

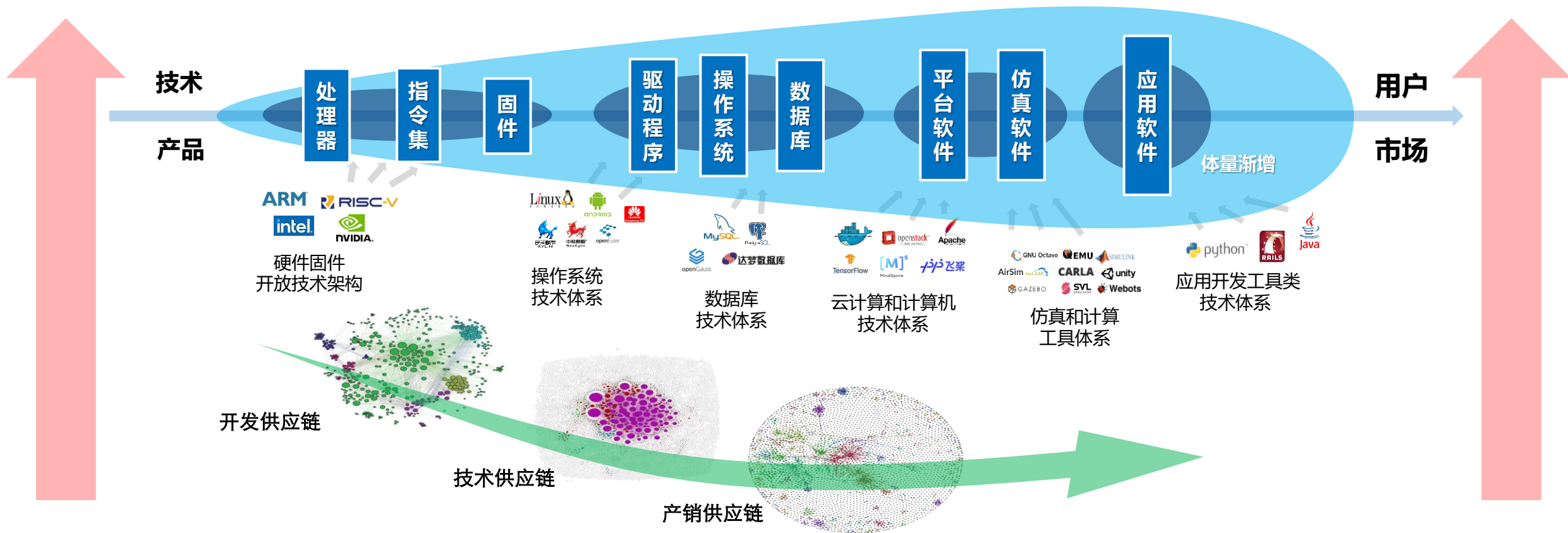
- 产业、科研、教育

开源项目运作的核心框架

- **代码托管**：选择许可证，选择托管平台，（选择基金会），开放源代码
 - 软件的生态定位
- **社区使能**：定义系统化贡献指南，经营多样化社区
 - 定义指南：CONTRIBUTING.MD，为新手任务打标签
 - 运营：生态的形成需要有三类角色：软件/产品/服务的生产者、提供者和消费者
- **版本发布和迭代**：新特征开发和发布、老版本持续维护的周期
- **项目可持续**：有用户，有社区，有市场

开源软件项目的核心挑战：生态的形成和持续

- 开源成功的核心表征是生态，生态有两个维度：
 - 水平横向：软硬件全栈，各类供应链关系网络
 - 垂直纵向：项目社区的汇聚、协作、可持续



本土开源操作系统

- openEuler: 旨在通过开源社区构建一个支持多处理架构和多芯片的生态体系。

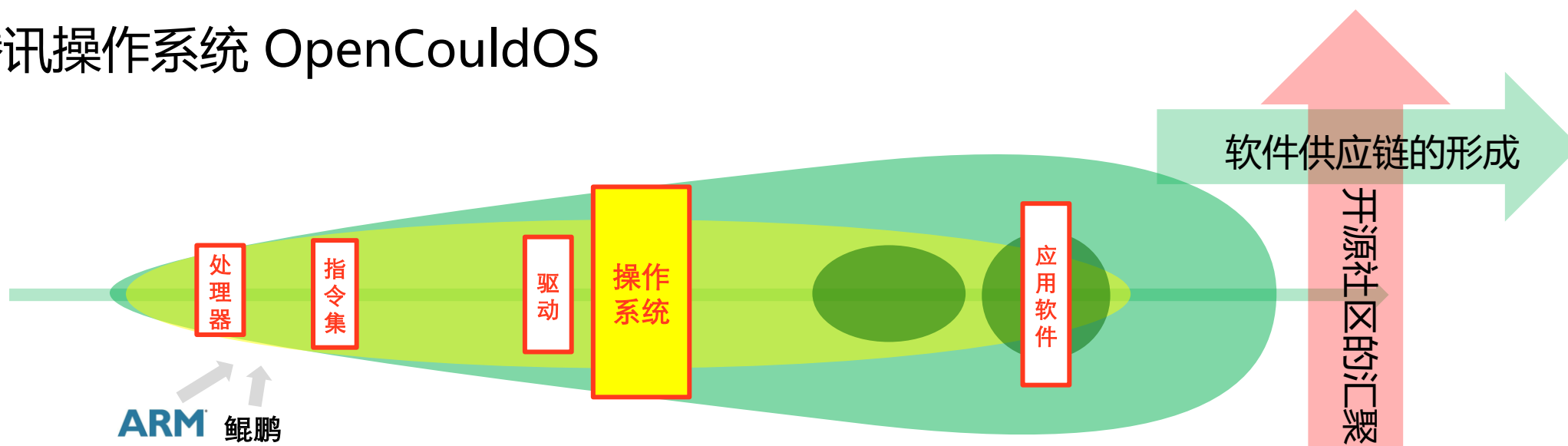
- 软硬件供应链: 鲲鹏, Linux kernel, openEuler, 基于openE的OS发行版  openEuler

- 开源社区: 华为, Euler OSV, 个体用户,

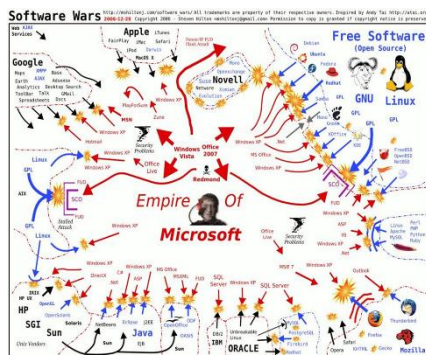


- 龙蜥操作系统 AnolisOS

- 腾讯操作系统 OpenCouldOS



理解开源生态：数据驱动的精细化度量



开源生态大数据



海量成功案例和最佳实践



开源生态构建的
机制机理、方法
技术和支撑工具

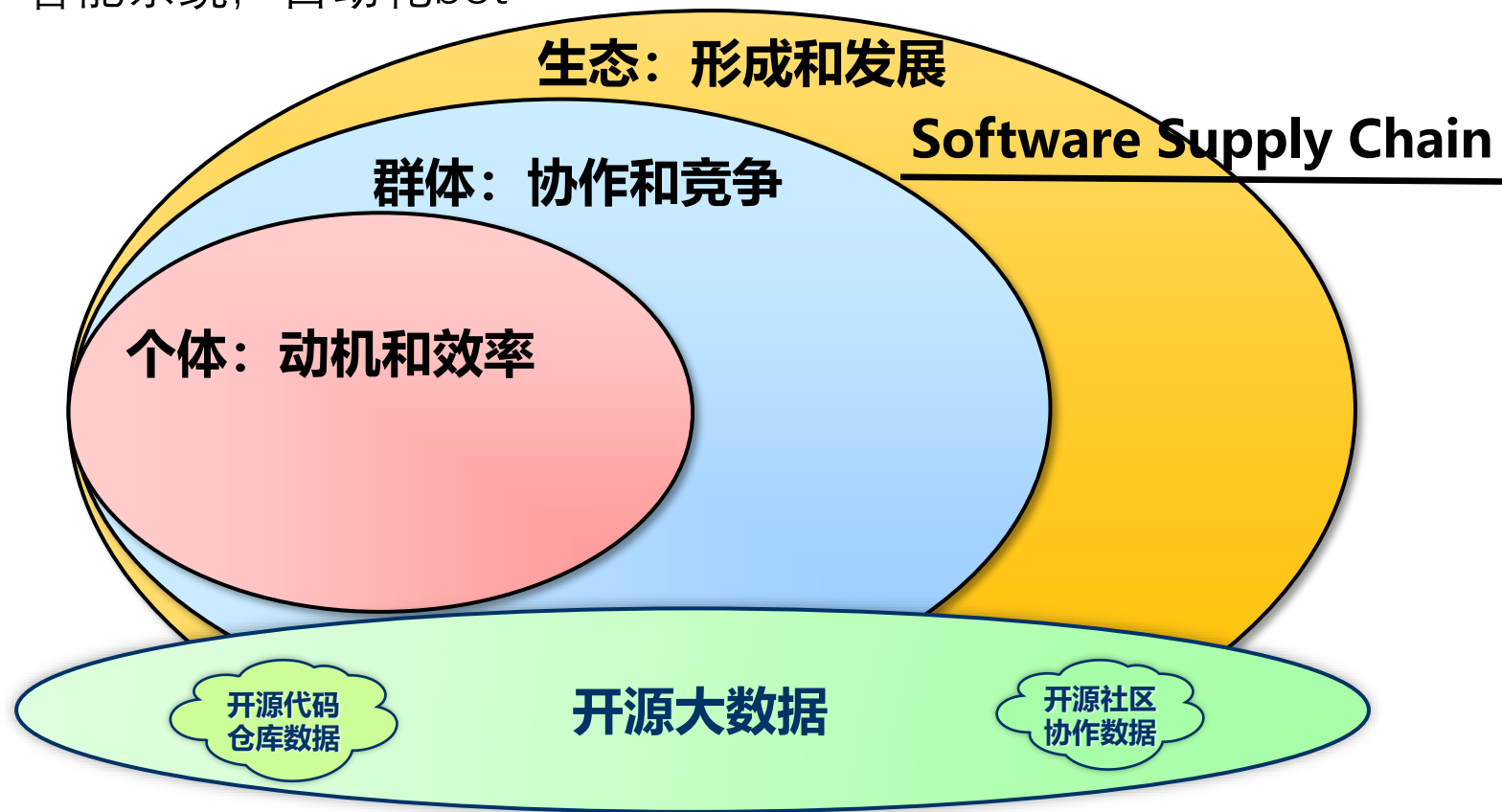
生态运作模式

开源数字社会学/
开源动力学

北京大学开源分析实验室：
<https://osslab-pku.github.io>

开源动力学

- ❑ 基础：开源活动大数据
- ❑ 方法：数据挖掘、智能分析、学科交叉（管理学、认知学等）
- ❑ 对象：开源软件供应链 ~ 个体/群体/生态
 - 发现型研究：度量复杂系统，发现规律
 - 发明型研究：智能系统，自动化bot



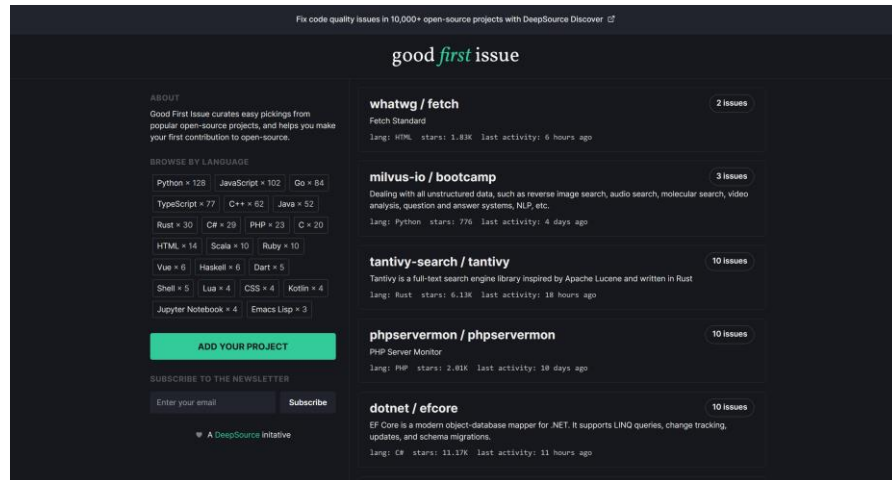
个体层次: motivation/productivity/learning

❑ 如何吸引贡献者?

- Extract features of issues to Tag Good first issues that suit newcomers.

❑ 如何保留有价值的贡献者?

- People behave differently when joining projects, that can be measured and used to predict their future behavior.



<https://github.com/osslab-pku/gfi-bot>

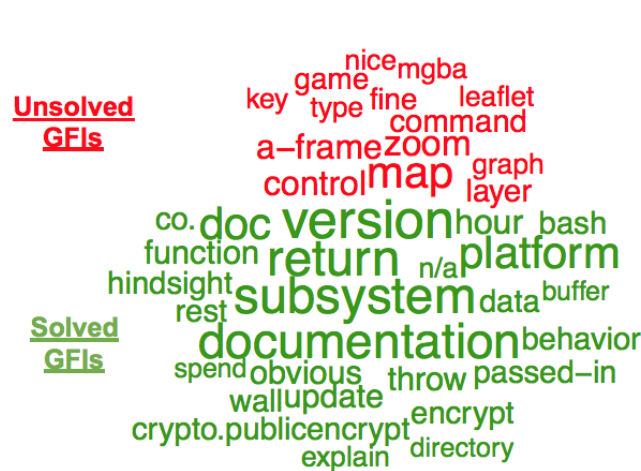
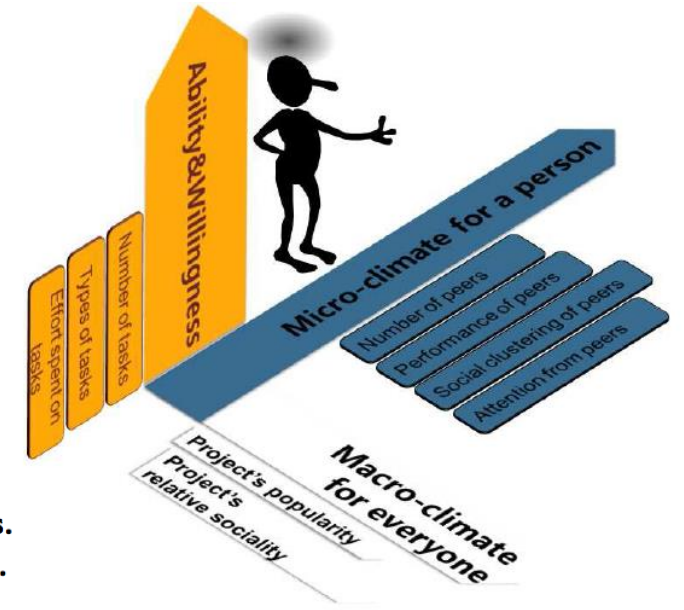


Figure 10: Comparison of solved GFIs and unsolved GFIs.
The top 50 keywords with the highest frequency are shown.



Xiao et al. Recommending good first issues in GitHub OSS projects. ICSE'2022

Tan, Zhou and Sun. A First Look at Good First Issues on GitHub. FSE'2020

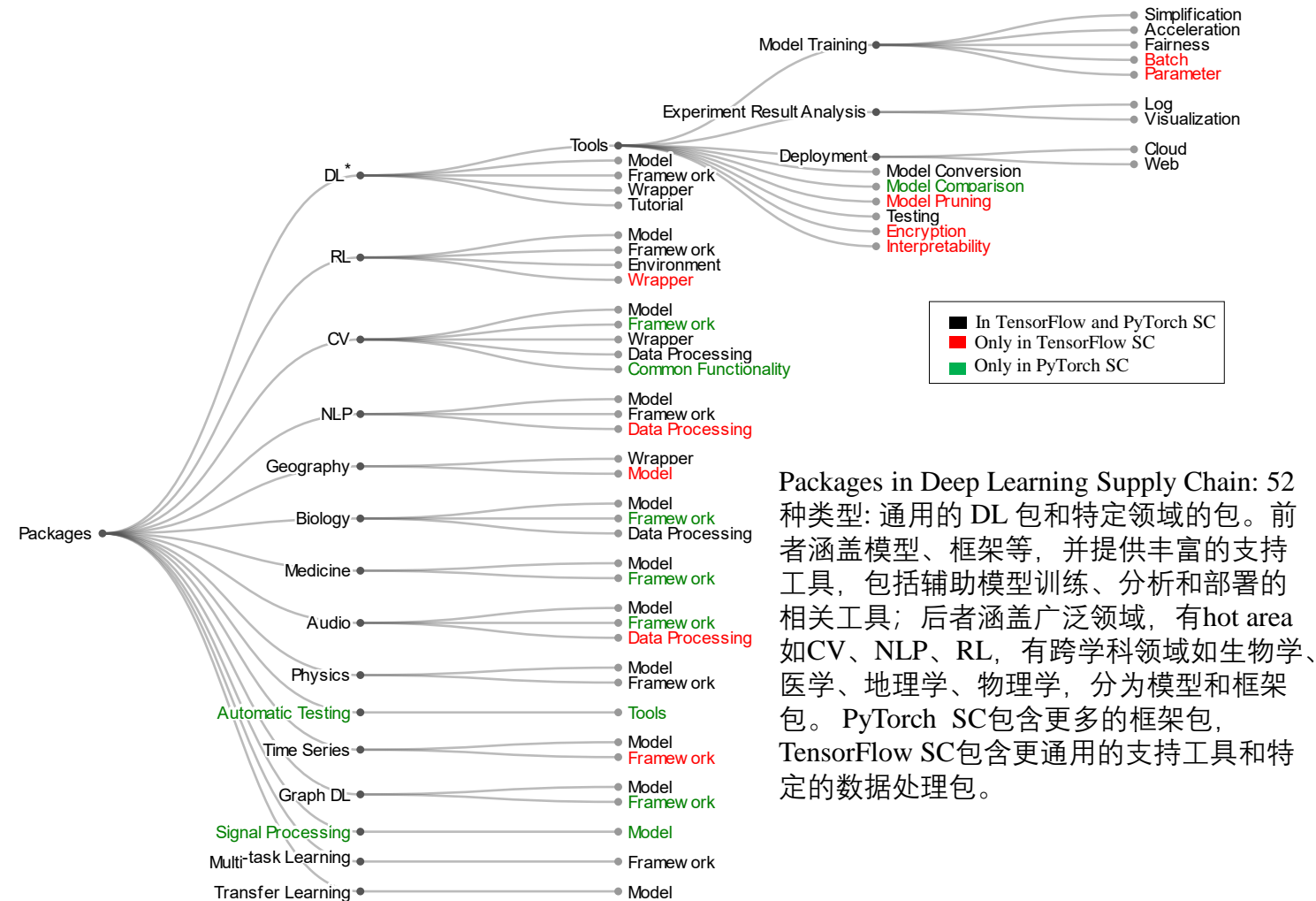
Zhou and Mockus. Who Will Stay in the FLOSS Community? Modeling Participant's Initial Behavior. IEEE Transaction of Software engineering. Jan 2015

Zhou and Mockus. What make long term contributors: Willingness and opportunity in OSS community. ICSE'2012

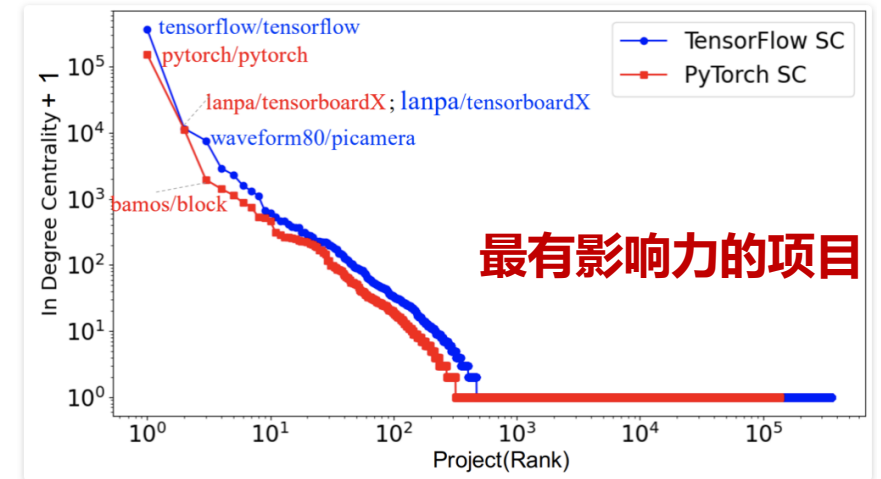
开源软件供应链：风险控制和智能分析

□ 供应链的网络结构、关键节点和风险传播

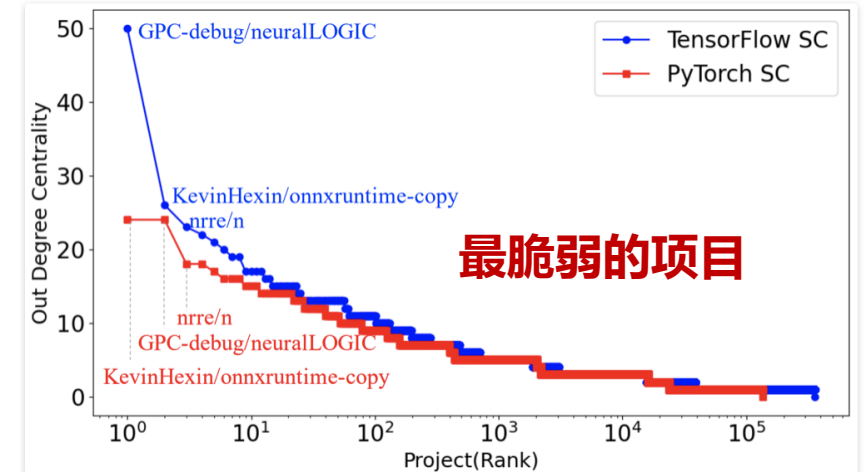
➤ 机器学习AI供应链：以TensorFlow/PyTorch为原点建立project/package依赖链条



Packages in Deep Learning Supply Chain: 52 种类型: 通用的 DL 包和特定领域的包。前者涵盖模型、框架等, 并提供丰富的支持工具, 包括辅助模型训练、分析和部署的相关工具; 后者涵盖广泛领域, 有 hot area 如 CV、NLP、RL, 有跨学科领域如生物学、医学、地理学、物理学, 分为模型和框架包。PyTorch SC 包含更多的框架包, TensorFlow SC 包含更通用的支持工具和特定的数据处理包。



In-degree Distribution of the Projects.



Out-degree Distribution of the Projects.

开源课程的建设和普及

□ 开源开发有其不同于传统CS教育体系的文化、技术和实践

□ 将开源软件融入到现有课程建设

- 操作系统、数据库、编译原理、……

□ “开源软件开发”课程--北大CS本科生课程

- 课程目的：帮助了解开源软件开发，学习如何参与开源项目和社区，训练开源思维。
- 课程内容：以学生参与开源项目的实践为整体线索，讲授开源软件开发涉及的理论知识、方法技术和工程实践。

用开源模式开发开源课程

[Discussions](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

main 1 branch 0 tags

Go to file Add file Code

minghuizhou Update contributing.md 0a80a75 6 days ago 5 commits

LICENSE	Initial commit	7 days ago
README.md	Add files via upload	6 days ago
contributing.md	Update contributing.md	6 days ago
contributors	Create contributors	6 days ago
outline.md	Add files via upload	6 days ago

README.md

本课程为大学计算机系本科生和研究生设计，课程目的是使得计算机领域的同学们对于开源软件的开发方法、开发过程和支持技术有深入的了解，对于如何将抽象方法和理论知识结合到工程中有更好的锻炼，对于参与开源软件社区有基本的掌握。



• 开源库地址：

- <https://github.com/osslab-pku/OSSDevelopment/>
- <https://gitee.com/osslab/ossdevelopment>

开源是现在和未来，机会和挑战并存

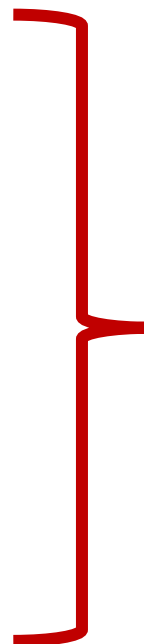
- 人人有责，从我做起



~7300万开发者
~2亿代码仓库



~ 600万开发者
~1500万代码仓库



开源产业

开源研究

开源教育

建议阅读的文献

- *The Cathedral & the Bazaar* (大教堂与集市). Raymond, E.S. (1999). O'Reilly Retrieved from <http://www.catb.org/~esr/writings/cathedral-bazaar/>
- 人月神话. 弗雷德里克·布鲁克斯. 出版社: 清华大学出版社. 译者: 汪颖. 出版年: 2002-11. ISBN: 9787302059325
- 开源的成功之路 (The Success of Open Source), 史蒂文 (美国), 外语教学与研究出版社, 2007-6, ISBN: 9787560066363.
- A. Mockus, R. T. Fielding, and J. Herbsleb, “Two case studies of open source software development: Apache and Mozilla,” ACM Trans. Softw. Eng. Methodol. vol. 11, no. 3, pp. 1–38, Jul. 2002.
- Minghui Zhou and Audris Mockus. Who Will Stay in the FLOSS Community? Modelling Participant's Initial Behaviour. IEEE Transactions on Software Engineering. vol.41, no.1, pp.82-99, Jan. 1 2015.
- 周明辉,张伟,尹刚. 开源软件的量化分析.中国计算机学会通讯.第12卷,第2期. 2016年2月.