

# 开源治理白皮书 (2018年)

中国信息通信研究院  
China Academy of Information and Communications Technology ,CAICT

2018年3月

---

---

## 版权声明

---

本白皮书版权属于中国信息通信研究院，并受法律保护。转载、摘编或利用其它方式使用本调查报告文字或者观点的，应注明来源。违反上述声明者，将追究其相关法律责任。

### 编写说明

**牵头单位：**中国信息通信研究院

**参与单位：**腾讯控股有限公司、中兴通讯股份有限公司、ZStack（上海云轴信息科技有限公司）、华为技术有限公司、阿里云计算有限公司、国际商业机器（中国）公司、红帽软件（北京）有限公司、EasyStack(北京易捷思达科技发展有限公司)、中国移动通信研究院、烽火通信科技股份有限公司、甲骨文(中国)软件系统有限公司、上海思华科技股份有限公司、中联润通信息技术有限公司、Mesosphere, Inc

**编写人：**栗蔚、郭雪、孔松、王柏钧、张亚军、孙辰星、李响、梅磊、蒋迪、刘艳华、李静远、田忠、赵文斌、刘宏亮、马轶慧、贾玄、涂文杰、齐鸣、张霖、冯文强、王建宏、曹治政

# 目录

一、开源概述 .....	5
1.1 开源定义和相关概念 .....	5
1.2 开源的市场应用 .....	10
1.3 开源的价值和意义 .....	12
二、全球开源发展现状 .....	15
2.1 开源发展历程 .....	15
2.2 企业参与开源的模式 .....	16
2.3 开源基金会 .....	18
三、中国开源发展现状 .....	23
3.1 开源发展历程 .....	23
3.2 企业参与开源的形式 .....	24
3.3 中国开源组织 .....	25
四、自主开源的关键环节 .....	27
4.1 事前准备和评估 .....	27
4.2 开启开源项目 .....	33
4.3 构建和维护开源社区 .....	36
4.4 开源社区治理 .....	37
4.5 开源项目评价 .....	38
五、开源商业模式 .....	41
六、开源技术的使用 .....	44
6.1 使用开源技术的风险 .....	45
6.2 如何选择开源技术 .....	45
6.3 如何使用开源技术 .....	47
七、开源面临的问题和挑战 .....	50
7.1 开源风险 .....	50
7.2 违约风险 .....	51
7.3 知识产权风险 .....	52
7.4 开源许可证兼容性风险 .....	57
7.5 安全风险 .....	57
八、总结 .....	59
附录：开源相关企业概述 .....	60
一、腾讯控股有限公司 .....	60
二、华为技术有限公司 .....	63
三、阿里云计算有限公司 .....	66
四、中兴通讯股份有限公司 .....	68
五、ZStack(上海云轴信息科技有限公司) .....	70
六、EasyStack(北京易捷思达科技发展有限公司) .....	71
七、红帽软件(北京)有限公司 .....	72
八、烽火通信科技股份有限公司 .....	73

---

## 前言

近几年开源技术快速发展，在云计算、移动互联网、大数据等领域逐渐形成技术主流。开源一方面可以突破技术壁垒，推动技术创新，另一方面，不可避免的带来知识产权、信息安全等一系列问题。我国企业在参与国际开源项目，跟随主流开源技术的同时，更多的影响和主导开源项目，逐步探索引领开源的可能。企业如何开源、项目开源流程、开源商业模式、如何使用开源等问题成为产业界关注的焦点。

《开源治理白皮书》首先介绍了开源的发展历程以及国内外主要的开源基金会及开源项目，针对如何开源、开源的商业模式、用户使用开源应注意的问题进行详细阐述，最后对企业开源面临的问题和挑战进行了概述。

---

## 一、 开源概述

### 1.1 开源定义和相关概念

**开源。**即开放一类技术或一种产品的源代码，源数据，源资产，可以是各行业的技术或产品，其范畴涵盖文化、产业、法律、技术等多个社会维度。如果开放的是软件代码，一般被称作开源软件。开源的实质是资产或资源（技术）共享，扩大社会价值，提升经济效率，减少交易壁垒和社会鸿沟。开源与开放标准、开放平台密切相关。

**开源软件。**一种版权持有人为任何人和任何目的提供学习、修改和分发权利，并公布源代码的计算机软件。通常具备以下 10 个特点<sup>1</sup>：

- 免费重新发行。当软件是来自不同来源的程序集成后的软件发行版本中的其中一个组件时，许可证不能限制任何团体销售和分发该软件，并且不能向这样的销售或分发收取许可费和其它费用。

- 源代码。程序包含源代码，并且必须允许以代码或已编译的形式发布。

- 衍生产品。许可证必须允许修改原产品和衍生产品，并且必须允许在与原始软件相同的许可情况下发布修改过的产品。

- 源代码完整性。许可证可以禁止他人以修改过的形式发布源代码，只在该许可证基于修改程度的目的时，才允许随源代码发布“补

---

<sup>1</sup> [<https://opensource.org/docs/definition.html>]

---

丁文件”。许可证必须明确允许发布根据修改过的源代码构建的软件。

许可证要求衍生产品必须附加不同于原始软件的名称或版本号。

-不得歧视任何人和团体。许可证不得歧视任何人和任何团体。

-不得歧视任何特定用途。许可证不得禁止任何人在特定领域内使用某一程序。

-许可证发布。附加在程序上的权利必须应用于那些重新发布程序的人，无需通过其它人额外加以许可使用。

-许可证不得专属于特定产品。附属于程序的权利不得仅限于作为特定软件发行版一部分的程序。

-许可证不得对其它软件加以限制。许可证不得对与已许可软件一起分配的其它软件附加任何限制。

-许可证必须技术中立。任何许可证都不可以基于单独的某项技术或界面风格。

**自由软件。**一种用户可以自由地运行、拷贝、分发、学习、修改并改进的软件。<sup>2</sup>自由软件需要具备以下四个特点：无论用户处于何种目的，必须可以按照用户意愿，自由地运行该软件；用户可以自由地学习并修改该软件，以此来帮助用户完成用户自己的计算，作为前提，用户必须可以访问到该软件的源代码；用户可以自由地分发该软件的拷贝，用户可以自由地分发该软件修改后的拷贝，用户可以把改

---

<sup>2</sup> <http://www.gnu.org/philosophy/free-sw.html>

---

进后的软件分享给整个社区而令他人收益，作为前提，用户必须可以访问到该软件的源代码。

**免费软件。**一种开发者拥有版权，保留控制发行、修改和销售权利的免费计算机软件，通常不发布源代码，以防用户修改源码。<sup>3</sup>

开源软件和自由软件。广义上认为，自由软件是开源软件的一个子集，自由软件的定义比开源软件更严格。

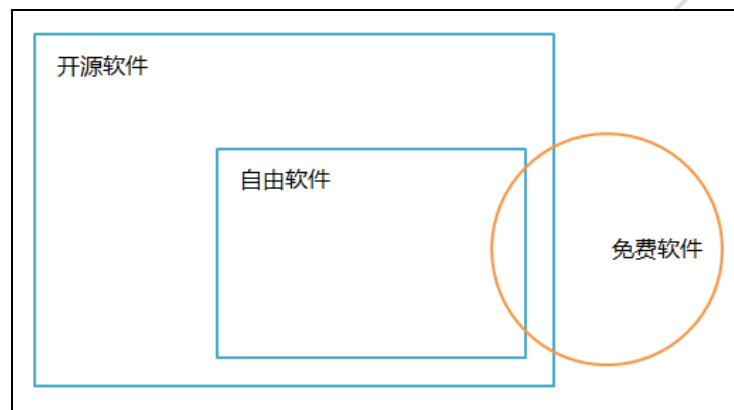


图1 开源软件、自有软件和免费软件的关系

开源软件和免费开源。开源软件是要求软件发行时附上源代码，并不一定免费；同样免费软件只是软件免费提供给用户使用，并不一定开源。

**开源许可证。**一种允许源代码、蓝图或设计在定义的条款和条件下被使用、修改和/或共享的计算机软件和其他产品的许可证。目前经过 OSI 认证的开源许可证共有 74 种，而最重要的仅有 6-10 种（最主要 2 种：GPL、Apache）。在开源商业化的浪潮下，适度宽松的 Apache 等许可证更受欢迎。

---

<sup>3</sup> <http://www.linfo.org/freeware.html>

---

这些许可证大致可以分为三类，一类是**传染型开源许可证** (Copyleft)，传染型开源许可证明确修改版本须以同一许可证发布，如果一个软件包含该协议下部分代码，完全发布时必须作为整体适用该协议，GNU General Public License Version 2 或 Version 3 (下称“GPL V2”或“GPL V3”)作为传染型开源许可证给予任何人自由复制、修改和发布 GPL 代码的权利，但是作为回报，所有以 GPL 协议发布的源代码的衍生，也必须按照 GPL 发布。**第二类是弱传染型开源许可证** (Weak-Copyleft)，如果一个软件包含该协议下部分代码，完全发布时某些部分必须适用该许可证，其它部分可在其它协议下发布，如 LGPL、MPL 等。**第三类是获准型许可证**，对已修改代码的许可方式没有任何要求，如 BSD 要求许可证附上许可证的原文以及所有开发者的版权资料，它允许原作品及修改版发行不公开源代码或以其它许可证发行。广泛使用的开源许可证包括 Apache-2.0, BSD-3-Clause, BSD-2-Clause, GPL, LGPL, MIT, MPL-2.0, CDDL-1.0, Eclipse 2.0.<sup>4</sup>

--GPL (GNU General Public License, GNU 通用公共许可证)：一种广泛使用的自由软件许可证，保证用户可以自由的运行、学习、分享和修改软件。许可证最初由自由软件基金会 (FSF) Richard Stallman 为 GNU 项目所撰写。GPL 是一个非盈利版权许可证，要求衍生作品只能在相同的许可条款下发布。GPL 的出发点是代码的开源使

---

<sup>4</sup> <https://opensource.org/licenses/category>



---

用和引用代码开源使用,不允许修改后和衍生的代码作为闭源的商业软件发布和销售。

--**LGPL (GNU Lesser General Public License, GNU 宽通用公共许可证)**: 一种由 FSF 颁布的自由软件许可证,允许开发者或公司在私有软件中使用,不要求使用 LGPL 许可代码的软件以 LGPL 方式发布。与 GPL 的强制性开源方式不同, LGPL 允许商业软件通过类库引用的方式使用 LGPL 类库而不需要开源商业软件的代码。

--**BSD (Berkeley Software Distribution)**: 允许使用者修改和重新发布代码,也允许使用或在 BSD 代码上开发商业软件并发布和销售。

--**MIT License**: 允许开发者任意处置该软件,包括使用,复制,修改,合并,发表,分发,再授权,或者销售。唯一的限制是,软件中必须包含许可提示。

--**Apache License**: 一种由 Apache 软件基金会发布的自由软件许可证,相对比较友好,被授权者可以发布商业化软件。

--**MPL (Mozilla Public License 1.1)**: MPL 协议允许免费重发布、免费修改,但要求修改后的代码版权归软件的发起者。

--**CDDL (Common Development and Distribution License)**: CDDL 开源许可证,是 MPL 的扩展协议,它允许公共版权使用,无专利费,并提供专利保护,可集成于商业软件中,允许自行发布许可。

--EPL (Eclipse Public License 1.0 ):EPL 允许 Recipients 任意使用、复制、分发、传播、展示、修改以及改后闭源的二次商业发布。

开源许可证的共同点是保留版权，差别是商业兼容性或共享权限不同。

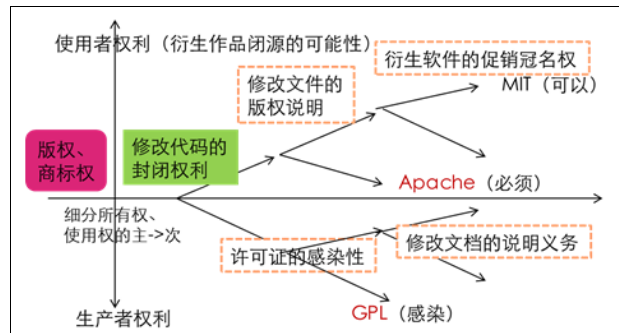


图 1 不同开源许可证的特点

## 1.2 开源的市场应用

开源软件市场应用广泛。据 Gartner 调查显示,99%的组织在其 IT 系统中使用了开源软件,同时开源软件在服务器操作系统、云计算领域、Web 领域都有比较广泛的应用。

开源软件市场规模稳居服务器操作系统首位。根据《Linux 内核开发报告 2017》显示,自从进入 Git 时代(2005 年 2.6.11 发布之后),共有 15637 名开发者为 Linux 内核的开发做出了贡献,这些开发者来自 1513 家公司。全球公有云上运行的负载有 90% 是 Linux 操作系统,在嵌入式市场的占有率是 62%,而在超算的市场占有率更是达到了 99%,还有,它运行在世界上超过 82% 的智能手机中,也是所有公有云厂商的主要支撑服务器 (90%)。

---

开源软件在云计算领域使用广泛。云计算领域开源目前主要以 IaaS 以及 PaaS 两个层面为主，IaaS 有 OpenStack、CloudStack、oVirt、ZStack 等，PaaS 层面有 OpenShift、Rancher、Cloud Foundry 以及调度平台 Kubernetes、Mesos 等。2017《OpenStack User Survey》显示：2017 年，OpenStack 全球部署将近 1000 次，比 2016 年增加 95%；亚洲超越北美成为 OpenStack 用户分布 Top1 区域。除 IT 以往，OpenStack 在其他行业也得到了广泛使用，Top4 用户行业为电信、研究、金融和政府。与 2016 年相比，IT 行业用户从 65% 下降到 55%。虽然 OpenStack 在 IT 业仍是最大行业应用领域，但电信、学术研究、零售和制造业都有所增长，尤其是金融和政府领域的增长最为显著。2013 年 Docker 发布之后，技术日渐崛起。截至 2014 年底，容器镜像下载量高达 1 亿；到 2017 年初，这一数量超过 80 亿。

开源软件在 Web 领域占据主要市场份额。根据 Netcraft《Web Server Survey》显示，调查发现，Nginx 市场份额增加到 21.4%，较上一年增长 3.84%，成为了面向 Web 的计算机市场上第三大服务器厂商。Apache 的市场份额下跌了 0.35 个百分点，它在面向 Web 的计算机市场的份额达到 42.8%。目前，至少有 7200 万个网站宣称在他们的服务器头部正在使用 Apache 2.2。Apache 已经成为互联网站点最为流行的 Web 服务平台。

---

开源软件在移动终端主导发展方向。Android 占据全球手机出货量主要市场份额。2015 年第三季度，市场份额高达 84.7%，主导移动应用软件发展。

开源软件在大数据方面的开源集中在可视化、基础分析工具、算法模型工具等方面。可视化一般以开源库为主，比如 d3.js、百度 ECharts 等；基础分析平台则有 Hadoop、Spark、Kylin 等；算法模型工具则有 TensorFlow、Weka、R 以及 Python 对应工具库等。

网络功能虚拟化（NFV）方面较为典型的开源项目有 OPNFV，它以 OpenStack 为支撑模板，现在也使用了 Kubernetes 容器化部分网络功能。

开源软件在人工智能领域的应用逐步推广起来。谷歌将 TensorFlow 开源，机器人操作系统 ROS 也大多采用开源技术。

## 1.3 开源的价值和意义

### 1.3.1 开源生态促进国家信息技术创新, 带动经济发展

开源有效促进技术创新。开源模式可以有效实现信息互通，获得关键技术最新源代码，利用全球技术资源快速推动技术发展迭代，打破技术壁垒，推动新技术普及。

开源可以实现软件自主可控。开源更加透明和公开，建立我国开源软件产业，可以有效保障信息安全，实现自主可控，同时可以保证

---

信息安全更易治理,产品和服务一般不存在恶意后门并可以不断改进或修补漏洞。

开源促进教育和科研事业发展。开源为高校师生提供了更多自主学习资源,学生能够直接、迅速的加入到开源项目中,技术水平不断提高,经验不断丰富。

开源促进行业信息化发展。开源模式可以有效降低应用成本和技术门槛,加快企业的信息化发展,从而带动促进我国经济的蓬勃发展。

### 1.3.2 软件厂商依托开源技术提升研发能力

降低研发成本。软件厂商借助开源技术降低研发成本,通过开源社区参与技术交流,熟悉开源技术使用方法,便于跟踪开源技术版本更新,根据业务需求进行适配。

知名开源项目均有业界高水平研发人员的参与,其源码在编码风格、算法思路等方面有许多供技术人员借鉴的地方。软件厂商研发人员在使用开源项目或基于开源项目进行二次开发的过程中,可以通过阅读源码等方式学习解决问题的创新方法。

软件厂商将项目开源后,项目的用户范围更广,应用场景更复杂,研发人员在开发时不能只考虑本公司的业务需求和人员使用情况,需更加注重代码兼容性、规范性等问题。

### 1.3.3 用户使用开源技术改变信息化路线

企业用户可在开源技术基础上进行定制化开发。终端用户信息系统需要实现的功能各有不同,相比闭源软件,开源软件更灵活,定制

---

化程度更高，终端用户企业可在开源代码的基础上做二次开发，实现特定场景和特定功能的需求，避免绑定风险。

使用开源技术，让企业专注于创新。随着更多的业务资源摆脱了开发软件的束缚，焦点转向创新。创造力在中小型企业中蓬勃发展，因为他们更能够创造竞争力的替代技术和专有软件。从而使自己获得比竞争对手更具有独特性和前瞻性的思维。

#### 1.3.4 企业自主开源 引领技术发展路径

企业自主开源能够有效提升研发效率，提高代码质量。项目开源过程中可以吸引优秀的开发者和用户参与到代码贡献中，注入更多新鲜血液，让项目不断发展。与此同时，开源项目部署在不同的应用场景，可以暴露出项目更多的问题，节省测试成本。

企业自主开源能够引领技术发展，建立以开源企业为核心的生态圈。开源项目运营过程中，可以吸引潜在用户使用开源软件，让业内更多的企业、开发者了解开源项目所属企业的技术发展情况，通过开源技术建立提供商和用户的上下游生态圈，及时了解用户需求，抢占商业版图，带动企业良性发展。

## 二、全球开源发展现状

### 2.1 开源发展历程

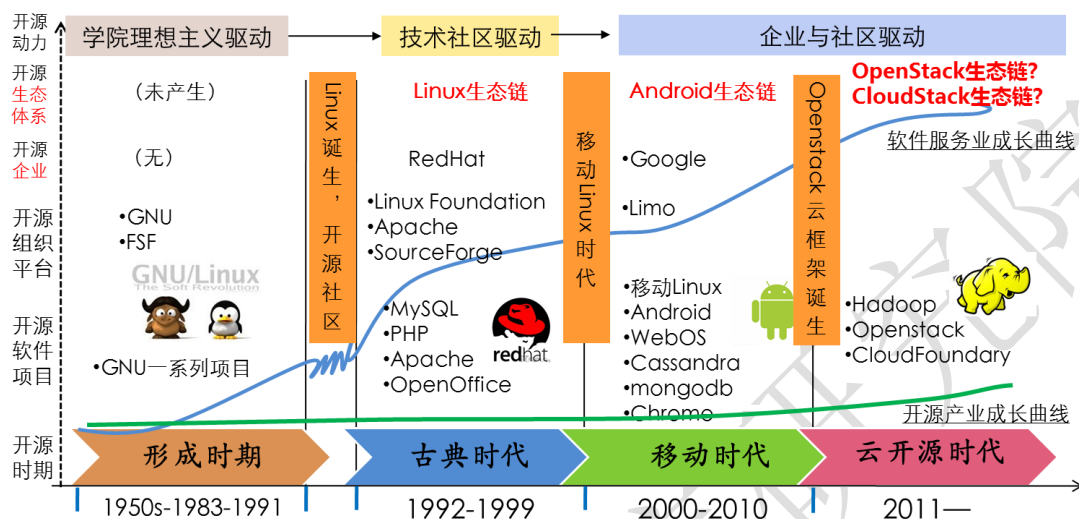


图2 开源发展历程

开源项目形成。1969年，AT&T 贝尔实验室研发出 Unix，在 AT&T 将其商业化之前，Unix 的代码被 Unix 社区共享，Unix 诞生为开源奠定了重要的基础；1984年，MIT 人工智能实验室资深工程师 Richard Stallman 发起 GNU 项目，目标是使用自有共享的代码来构建一个 Unix 克隆的操作系统，绝大多数 GNU 项目，均是 Stallman 在 1984 年开始构建的，今天仍处于自由和开源软件生态系统的中心位置；1985年，Stallman 创建自有软件基金会；1991年，加利福尼亚大学伯克利分校发布 Net/2 BSD 操作系统，自由的类 Unix 操作系统，发布后由于法律问题，没有得到大规模推广；1991年，Linux 内核出现。

Linux 始于 Linus Torvalds 开发的类 Unix 操作系统内核，采用 GPL 许可。2000年，为赞助 Linus Torvalds 的工作，成立 Linux 基金会。Linux 基金会的成立，为世界各地的开发者合作建立平台，有力推动了 Linux 发展。

---

开源社区形成，企业逐渐参与到开源社区做贡献。1995年，Apache 诞生。上世纪 90 年代，Web 软件还处于封闭专有的状态，1995 年一组由系统管理员组成的开发团队开始协作构建叫做 Apache HTTP 服务器的软件，它是基于国家超算应用中心的一款 Web 服务平台。1996 年，Apache 已经占据了 Web 服务器大部分的市场份额，1999 年成立 Apache 基金会。

企业和社区驱动开源。2000 以后开源体系开始向移动和云领域延伸，Google 等企业开始在移动互联网和云计算领域驱动开源，影响技术发展路线和市场格局。

开源代码托管。Linux 开发过程中协作最开始是通过邮件，后续迁移到源代码管理平台，Linux 使用的商业的 BitKeeper 的源代码管理平台 2005 年不再为 Linux 提供免费的使用权，Torvalds 写了 Git 的工具代替 BitKeeper。2008 年开始出现一些以 Web 形式托管的 Git 代码仓库如 Github，Git 将开源编码的开放度带到了更高的高度，让每个人都可以快速推出一个开源项目。

## 2.2 企业参与开源的模式

随着开源生态的不断建立，企业参与开源生态的热情度不断提升，参与开源的形式可以分为四类，一是企业可以将内部开源项目开源出来，作为开源发起者；二是企业可以贡献代码，反馈社区；三是企业



可以通过培训、组织活动等形式作为开源项目的推动者；四是企业使用开源技术，降低研发投入。

发起开源项目有两种模式，一种**公共开源模式**，可以是个人发起开源项目，形成开源社区，开放生产，比如 Linux，也可以是个人或商业公司捐献给开源基金会，开放生产，比如 OpenStack；另一种是**商业开源模式**，由企业发起开源，封闭生产，比如 Android。

表 1 两种开源模式对比

		公共开源模式 (基金会)			商业开源模式		封闭模式	比较
角度		Linux模式	Apache模式	Openstack模式	Android模式	Webkit模式	MS模式	异同
载体		Linux Fd	Apache Fd	Openstack Fd	AOSP	WOSP		
源代码开放程度		完全	完全	完全	延迟	差异	×	不同
开源模式	开源产权模式	✓	✓	✓	✓	✓	×	具备
	开源生产模式	✓	✓	✓	×	×	×	不同
	开源商业模式	×	×	✓	✓	×	×	不同
开源模式要素	所有权角度	GPL	Apache	Apache	Apache	LGPL/BSD	私有	商业性
	生产角度	社区	社区	社区	半封闭	半封闭	用户反馈	封闭
	交换角度	非盈利性	非盈利性	服务盈利	相关盈利	整机盈利	许可盈利	盈利性
运行特征	技术发起者	学术界	学术+商业	商业公司	商业公司		商业公司	商业性
	更新速度	很慢	慢	慢	反应快	快	快	加快
	生态控制力	弱	弱	弱	强	强	强	加强
	版本化	多	多	多	较多	几类	单一	集中
性质	开源	开源	开源	有限开源	并行	闭源		递减
例子	Apache	Mozilla类似	Cloudstack等	Chromium	Darwin	Win,Off,IE		—

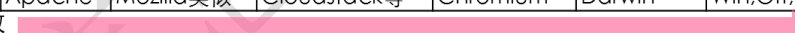
开放  封闭

表 2 常见开源项目的开源模式

	Linux	OpenStack	Android	Hadoop	MySQL
发起方	个人	用户企业	企业	开源基金会	企业
开源协议	GPL	Apache	Apache	Apache	GPL
开源形式	公共开源模式	公共开源模式	商业开源模式	公共开源模式	商业开源模式

贡献者。项目开源之后，其它企业可以反馈社区，贡献代码。典型的案例是 OpenStack，是一个非常庞大的项目，它由 NASA 的运维人员发起，现在全球也有数百企业参与开发并提交自己的代码，红帽

---

等企业积极参与代码贡献和社区贡献，带动开源项目良性发展，影响开源项目发展路径。

OpenStack 是为搭建公共及私有云建设与管理提供软件的开源项目，由 NASA 和 Rackspace 合作开发，经 Apache2.0 许可，以 Python 语言为基础，由多个组件组成的开源项目。NASA 由于战略调整，转向亚马逊云服务，由 OpenStack 的开发者转变为云平台的用户，于是将 OpenStack 开源，带动更多的技术人员参与 OpenStack。

推动者。由于开源项目本身的特点，很多企业在项目中乐意采用并传播开源。厂商在市场活动上也会对社区进行帮助，比如为其捐献、创办参与用户组、组织交流会等。

使用者。参与开源社区是为了软件的应用。例如 AT&T 加入 OpenStack 社区是为了使用 OpenStack 软件，AT&T 作为通信商业组织，希望使用 OpenStack 软件从传统的网络服务模式转移到基于云环境的 SDN/NFV 网络，AT&T 并没有直接通过 OpenStack 盈利也没有相应的业务，所以 AT&T 对 OpenStack 的控制管理需求较低，同时资源投入较少。

## 2.3 开源基金会

开源基金会作为管理和推广开源项目的非盈利机构，为开源项目提供法律、商业、技术等服务。目前国际上影响力比较大的基金会有 Apache 基金会、Linux 基金会等、OpenStack 基金会。开源基金会在微观上表现为开放性、创新性较强的集市生产方式，在宏观上表现为

公益性、合作性较强的多类产业合作组织的繁荣，其特点是以牺牲封闭方式下的组织效率换来以分布式、自组织为特征的开放性创新。

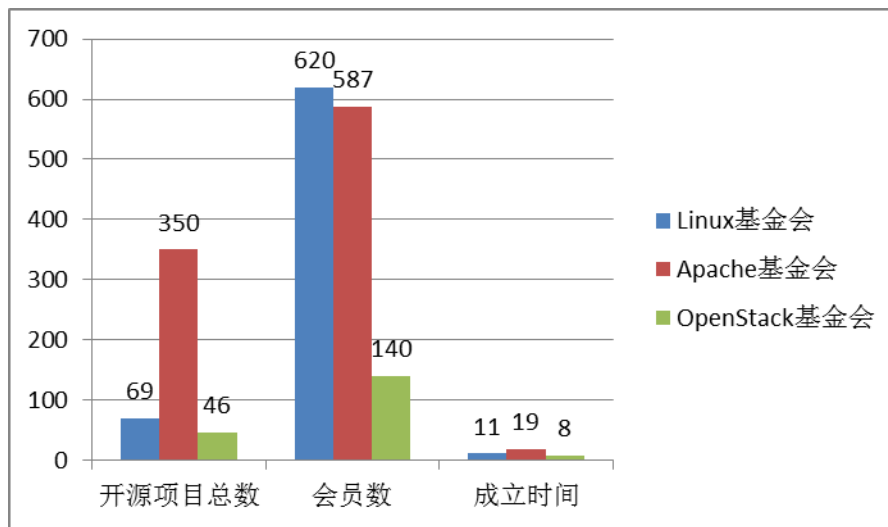


图3 Linux、Apache 和 OpenStack 基金会对比

### ➤ Apache 基金会

Apache 基金会成立于 1999 年，目前已成为世界上最为著名和成功的托管开源项目的基金会之一，目前管理超过 350 个开源项目。apache.org 站点每周访问量是 3 千 5 百万，每个月有接近 300 位新的代码贡献者。现共有 182 个顶级项目，59 个正在孵化中。Apache 旗下的所有项目都使用 Apache 许可证，也是目前最受欢迎的许可证之一。

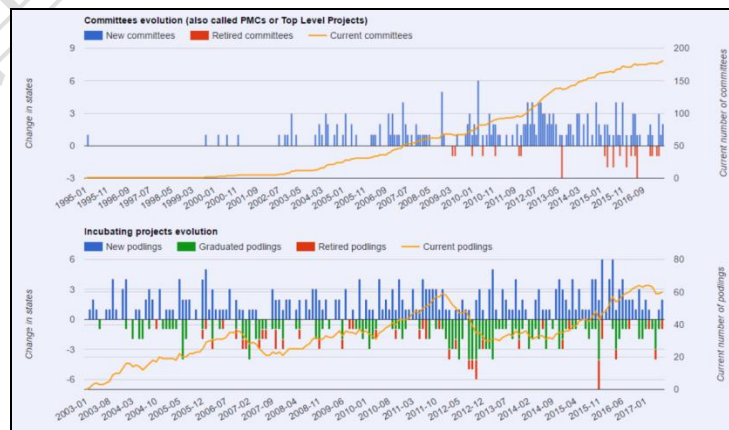


图4 Apache 基金会项目概览

精英制组织机构。Apache 基金会由个人组成，理事会管理与监督整个 Apache 软件基金会的商务与日常事务，并让它们能在符合章程的规定下正常地运作，为项目运行提供组织、法律、财务等支持，不会对项目发展施加影响。项目管理委员会，主要负责保证一个或者多个开源社区的活动都能运转良好。Apache 项目管理采用“精英”治理模式，项目开发人员分为用户、开发者、提交者、项目管理委员会等成员。

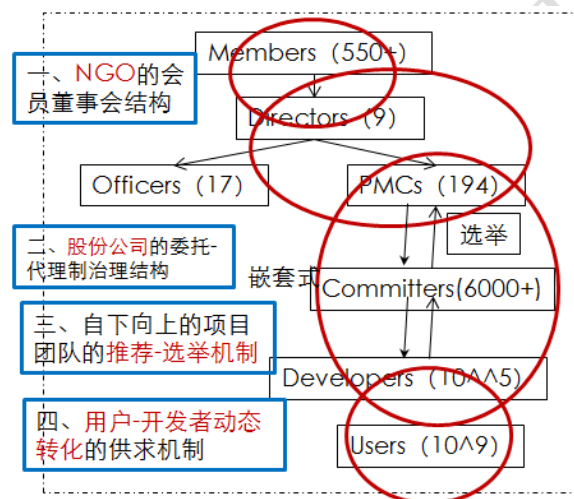


图 5 Apache 基金会组织架构

Apache 项目孵化帮助开源项目成长。无论是外部组织捐赠，还是 Apache 内部所创建的项目，都需要经过孵化才能够成为成熟项目，项目孵化过程将帮助新的项目解决所需的基础设施，指导社区精英模式的建立。

### ➤ Linux 基金会

Linux 基金会成立于 2007 年，由原开源开发实验室 (OSDL) 和自由标准协会 (FSG) 合并而成，旨在保护和激励自由的概念，通过 Linux 开发来加强合作。近几年，Linux 基金会逐渐成熟，开始孵化相关开

源项目，其中包括开放大型机项目、汽车应用，以及云计算相关的 OpenDaylight、OPNFV、Cloud Foundry 基金会、开放容器促进会、云原生计算基金会等。目前 Linux 基金会共有 69 个项目。



图 6 Linux 基金会项目概览

将会员划分等级进行管理。基金会会有三个不同等级的会员：银级、金级、白金级。各级别会员需承担的责任有所差别，白金级别会员同时拥有董事会席位，每年需缴纳会费 50 万美元。

➤ Cloud Foundry 基金会。

Cloud Foundry 是一款提供 PaaS 环境的开源项目。2014 年 12 月成立 Cloud Foundry 基金会，隶属于 Linux 基金会合作项目成员。

➤ 云原生计算基金会。

2015 年 8 月，在 Linux 基金会下成立 CNCF 基金会。云原生计算基金会组织由 Linux 基金会负责构建，其中包括老牌技术厂商比如谷歌、IBM、英特尔、Box、思科以及 VMware，除此之外 Docker、Cycle Computing、Mesosphere 以及 Weaveworks 等规模较小的厂商也投身于其中。

---

➤ **OpenStack 基金会。**

OpenStack 项目发起于 2010 年，由 NASA 和 Rackspace 联合，为数据中心基础设施管理创建一个可编程的基础设施即服务层，2012 年成立 OpenStack 基金会。基金会由多个社区组成，有运维、用户、开发者社区，有非常健壮的治理模式用于指导其管理和开发流程，并由董事会和技术委员会进行监督。目前共有 46 个项目，包括云？计算 6 个，存储备份复原 5 个，网络和内容传输 7 个，数据分析 3 个，安全认定 4 个，管理工具 6 个，施展工具 6 个，应用服务 4 个，监控测量 5 个。

➤ **自由软件基金会**

是一个致力于推广自由软件、促进计算机用户自由的美国民间非盈利性组织。它于 1985 年 10 月由理查德·斯托曼建立。其主要工作是执行 GNU 计划，开发更多的自由软件，完善自由软件理念。

➤ **Creative Commons**

简称 CC，中国大陆正式名称为知识共享，台湾正式名称为创用 CC，是一个非营利组织。此组织的主要宗旨是增加创意作品的流通可行性，作为其他人据以创作及共享的基础，并寻找适当的法律以确保上述理念。

---

## 三、中国开源发展现状

### 3.1 开源发展历程

国内开源项目萌芽阶段。国内开源软件的发展始于 1997，当时中国第一个互联网（CERNET）刚刚建立，于此同时中国也出现了一些开源软件。最初我国开发的开源软件，主要解决的是 Linux 系统的汉化问题。但随着开源软件的发展，又出现了包括 SBM（Smart Boot Manager）、LVS（Linux Virtual Server）以及 Mini GUI 等在内的开源软件项目，使得开源软件在我国发展出现第一个高峰期。在此之后，国内开源软件的发展长期处于停滞状态，这和 2000 年左右 DotCOM 泡沫的破裂有一定的关系。DotCOM 泡沫的破裂，让许多梦想通过开源软件来创造商业奇迹的 Linux 发行版厂商很是受伤。

逐渐参与国际开源项目。2006 及之后的几年，学者们开始关注开源概念，开源的观念在中国普及开来。开源社区兴起，国内一些大型企业纷纷参与到开源项目中。2008 年金融危机后，传统企业为了降低 IT 的总拥有成本逐步使用 Linux 和开源软件。中国 Linux 套数比上年增长 16.6%（其中桌面 Linux 出货量 748 万套；服务器 Linux 出货量 7.7 万套）。由于开源在成本和安全等方面的优势，使客户在选型时将更多的眼光投向开源软件，以节省成本，这从另一个角度为开源企业带来了新的机遇。

行业化应用为开源发展带来新的契机。2009年，为带动2011年政府、铁路、民航、建筑、卫生等多个行业的信息化发展，中国政府出台了4万亿投资计划，开源软件的应用也在这轮投资中有着良好的上升。2012年到2015年，中国所有云部署项目中，完全不使用开源软件的云项目只占3%，开源软件在云计算领域得到日益广泛的应用。云计算的飞速发展，也为开源软件的发展提供了契机。

人工智能带动开源发展新机遇。2015年到2017年，出现机器学习平台开源大潮，谷歌、亚马逊、微软、IBM等国际互联网巨头为开源世界贡献了品类繁多的机器学习工具。国内BAT等行业巨头，也紧随其后，伴随着互联网浪潮在国内的发展，互联网企业承载着海量用户的业务压力，在这种压力之下，也涌现出一批顶级的开源项目发起者。

### 3.2 企业参与开源的形式

国内企业相比国际开源环境普遍对开源的接受度较低，以推动者和使用者为主，但近两年参与发起和共享的企业明显增多。

发起者。企业将内部项目开源出来，作为开源项目的发起者，国内华为、腾讯、阿里均积极主动发起开源项目，主要集中在云计算、大数据、存储、运维等领域，在国际上的影响力逐渐提升。

表3 国内开源项目

企业	开源项目	开源许可证	领域
华为	CarbonData	Apache 2.0	大数据
腾讯	蓝鲸	MIT	运维
阿里	Pouch	Apache 2.0	容器



---

ZStack	ZStack	Apache 2.0	云计算
--------	--------	------------	-----

贡献者。由于社区已经有很多优秀的开源项目，国内具备一定技术实力的厂商则作为开发者进入，自己在使用或者再发布的同时，也为社区作出贡献。OpenStack 是一个非常庞大的项目，现在全球也有数百企业参与开发并提交自己的代码，国内的众多公司，都加入该项目，提交代码，成为项目的核心开发公司，或成为项目模块的负责人，最终能够主导该项目的发展。OpenStack 最新版本——Pike 已经是 OpenStack 第 16 个版本，最新版本的全球代码贡献排名也新鲜出炉。在 Pike 代码贡献最终版本中，共有 45 家中国企业上榜；而上一个版本 Ocata 代码贡献的中国上榜企业是 37 家，环比增长高达 21%。

推动者。由于开源项目本身的特点，很多企业在项目中乐意采用并传播开源。厂商在市场活动上也会对社区进行帮助，比如为其捐献、创办参与用户组、组织交流会等。

使用者。国内终端用户企业积极参与开源社区，验证并反哺开源生态。中国移动、腾讯云在 2016 年和 2017 年先后获得 OpenStack Superuser 大奖。

### 3.3 中国开源组织

国内开源组织以开源联盟和开源协会居多，更多是推广开源项目的普及，组织开发者交流。国内目前云计算开源产业联盟（OSCAR）、开源社、中国开源软件推进联盟均在国内开源圈中有一定影响力。

---

云计算开源产业联盟（OSCAR）。成立于 2016 年 2 月，由中国信息通信研究院联合多家云计算开源技术公司发起成立，旨在促进我国云计算行业交流，加强企业和行业用户之间的沟通，推进云计算企业利用开源软件不断提升技术水平，目前国内 50 余家厂商均为 OSCAR 成员。主要工作包括以下几个方面：**一是**开源云计算开源领域研究，发布研究成果，发布了《中国云计算开源产业发展白皮书 第一部分：OpenStack 产业》等 5 个开源技术发展白皮书和《OpenStack 行业最佳实践》等多个规范指南，解决了各行业对开源技术不够了解，开源解决方案标准不统一，维护复杂等痛点；**二是**制定云计算开源相关标准，制定了《可信云开源解决方案》，持续开展开源解决方案标准制定及评测工作，累计 19 家 OpenStack 厂商及 5 家容器厂商通过评测，提升了产业的互操作性和产品质量；**三是**积极学习引进先进开源社区经验，探索了国内开源项目的运作机制，帮助企业用开源方式建立了多个开源项目，设计运营机制，社区管理规则，进行行业推广；**四是**积极推进开源技术在中国的人才储备，在北京、上海等地区的多个学校开展云计算开源技术公开课活动，同时联合社会培训机构，共同打造规范化的培训课程；**五是**推动云计算开源技术行业化应用，推动银行业、政务云、轨道交通、通信行业、保险行业等领域云计算开源技术的应用；**六是**积极开展国际合作，与 Linux 基金会、CNCF 基金会等国际开源组织积极开展合作，将评估标准与国际基金会的兼容

---

性认证标准进行对标，推动双方的评估方面的深度合作，将国际开源技术培训课程进行本地化推广。

开源社。开源社是由中国支持开源的企业，社区及个人所组织的一个开源联盟，旨在推广开源。各成员按照“共治、共识、贡献”的原则自发设立。联盟目前拥有五家核心成员和四十余家成员，旨在为中国开源软件和开源硬件的正确使用，授权许可，社区建设及治理，进行知识推广普及与提供相关服务。

中国开源软件促进联盟。是由致力于开源软件文化、技术、产业、教学、应用、支撑的企业、社区、客户、大专院校、科研院所、行业协会、支撑机构等组织自愿组成的、民主议事的民间行业联合体，非独立社团法人组织。在 2004 年 7 月 22 日于北京成立。

## 四、自主开源的关键环节

### 4.1 事前准备和评估

开源之前应做好充分的准备，一方面可以避免日后遇到不必要的麻烦，另一方面也有助于团队理清思路，为日后经营做好准备。

#### 4.1.1 业务评估

在开源前，业务方和项目负责人首先应对自身的项目和所处环境有充分的了解，并根据需要与相关人进行沟通，而后根据实际情况来制定具体的开源策略。在准备期间可参照如下条目：

第一，应认清开源动机

---

从开源能带来的价值出发，找到开源与自身业务价值和发展需要的契合点。只有找到了意义和价值，让开源行为拥有正当的动机，才能说服企业、团队甚至潜在的协作者予以支持和帮助。

对于大多数情况而言，开源的动机和目的可以归结为几点：

- ▶ 被动开源。出于行业惯例、规定或合规要求将项目开源。例如产品使用了带有传染性协议的开源代码，在分发时，依照协议和行业规定对项目进行合规化的开源处理。这样的开源动机，以合规为主要目的。
- ▶ 技术共享与同业交流，改善团队氛围和技术影响力。为了提升研发型团队的技术能力，与业界领先水平缩小技术上差距而采取的开放性尝试，或是宣讲团队所拥有的技术领先地位和能力将项目开源，接受同行评议与比对并根据同行反馈组织共建。这样的开源项目，一般以研发团队诉求和个人意愿为主要动机。由于现代的工程师文化本身就强调技术信仰与贡献精神，这一类开源项目重视分享，并在国内的现实当中较易落地。
- ▶ 开源作为产品特性以获取竞争优势。由于开源软件的开放透明等社会共识，使开源的解决方案更容易获取用户信任。开源降低了客户的获取和维护成本，能够争取更多的社区支持和口碑，健壮了信任度。此时产品选择以开源的形式发布，可以帮助在行业中获取竞争优势。在这种情况下，开源更注重商业模式和产业共建。

- 
- ▶ 体现社会责任和优化形象。企业在创造利润的同时，还要承担对消费者、环境乃至社会的责任，这要求企业必须超越把利润作为唯一目标的理念，强调对环境、对社会的贡献。通过技术手段，自身可以复用的项目通过开源的形式回馈行业和社会，不仅有助于企业实现社会责任，也有助于企业维护自身的公众形象。
  - ▶ 丰富行业应用，带动行业生态。在具体领域，通过开源的形式使行业的技术方案丰富，并借助开源的特点获得社区和开发者支持，有助于企业参与和经营行业生态，并最终影响到事实上的行业标准——开源的解决方案比起闭源软件更容易形成事实标准，从而比法定标准产生更大的能量。例如在微信开发者生态中，小程序的开源框架对微信构筑开发者生态十分有益，开源框架能够帮助企业开发者快速实现移动化服务能力。
  - ▶ 集中资源形成联盟，促进行业共建。随着社会分工不断细化，许多工作无法仅由一家企业单独完成，企业间需要密切配合并在产业链条上进行分工，同样对于一些重点项目，需要聚集多家企业的力量才能完成。此时，用开源治理的方式形成产业联盟，已经成为企业间相互协作的可选模式之一。
  - ▶ 推动产业革命，推进生产力发展与人类进步。由于开源技术的广泛应用，开源项目已经成为技术甚至科技发展的重要组成部分。促发展，谋进步，本是企业治理的终极理想。同样，开源

---

活动最根本的意义和价值，归根结底是为了通过推动科技革命，谋求生产力的发展，促进生产效率提升，最终推动人类进步。

对于业务来说，开源的动机可能包含以上一点或几点。在开源之前，应当对自身目标形成整体性的认识，避免盲目开源而导致无疾而终。

## 第二，应了解用户需求

按照开源使用的方式，开源软件的用户可分为三类。第一类，是开源软件的终端用户，这类用户更关注产品的功能以及使用体验和感受；第二类，是使用开源软件代码用于自身业务开发的开源使用者，用以实现自身业务或服务有关的应用，例如微信小程序开源框架的使用者，他们用微信小程序的开源框架实现对移动互联网用户的服务，此类用户更加注重开发者生态问题；第三类，是将开源软件集成在自身产品或解决方案中，向终端用户进行分发销售的软件厂商或集成服务商，他们生产的软件和集成方案需要再次向客户交付，因此对于开源软件，更加注重开源项目的发展情况、社区权益以及社区治理。根据不同的用户群体，会有不同的需求。分析用户在开源方面有哪方面的需求并为之准备解决方案，是开源项目在开源前应当酌情考虑的问题。

## 第三，分析行业特性和技术环境

---

置身于行业当中，行业的发展情况、行业技术能力、技术发展方向、解决方案饱和度等等因素，都与开源软件发布能达到的效果息息相关。例如，2015年以后在 Adobe Flash 平台上提供 ActionScript 开源服务框架，可能不会有太多收获，因为当时 Adobe Flash 的解决方案正在被市场和行业淘汰。对市场发展方向有充分理解，增强预估和把控能力，了解技术的发展变化趋势，对项目开源非常重要，有助于帮助开源的决策者和执行人制定合适的开源策略。

#### 第四，分析业务特性，与业务相关人员充分沟通

由于开源治理在国内尚属新兴方案，特别是开源软件的商业模式与传统软件销售模式有显著不同，在多数传统企业和软件开发商中，相关从业人员对开源治理的概念理解并不一定足够深刻，因此需要与相关人员，特别是软件销售、支持人员、合作伙伴、决策者等充分沟通，以便合作各方对开源方案有充分理解，保证开源本身与业务的发展方向一致，并与产品的商业化策略不相冲突。

#### 4.1.2 技术评估和自检自查

无论任何活动，都应满足尊重道德和遵守法律的双重前提，对于技术的运用也当如此。因此，应从技术应用上自检自查。第一，应保证所有代码的来源可查。在日常开发或准备开源时，应日常维护或特别整理一份代码来源的文本列表。这份文本列表应列明项目引用了哪些第三方的资源和源代码，以及这些资源和代码的项目名称、作者、出处、版本号和许可证。这样既标识了引用的第三方代码，也标示出

---

哪些代码是项目自有的，利于在需要时整理软件作品的著作权关系。

第二，根据引用的第三方资源列表，应对这些资源或源代码的开源许可证有足够理解，保证对其的使用符合其开源许可证——包括著作权、专利、商标等要求（具体风险详见第七章），若存在协议冲突，应对有冲突的资源或源代码进行剥离或替换处理。

此外，由于自由软件源代码的开源，可能涉及专利、商标、商业秘密、用户隐私等问题，亦应对软件代码进行规范化处理。应当移除与本软件无关的专利和商标使用，确保不会泄露商业秘密，并确保源代码中不包含用户隐私等实际数据或是仅包含符合法律和隐私规定的脱敏数据。

最后，由于开源代码也会影响单位、企业和个人的技术形象。首先，应从代码层面予以规范，包括代码风格、注释、无关依赖、配置等。第二、应准备尽量详尽的资料、说明和文档，确保开发者和最终用户可以轻易上手和快速使用，同时这些资料、说明和文档亦应符合道德与法律要求。

#### 4.1.3 合规审查

在规范的企事业单位中，为保障企业、用户和最终消费者的权益，会设立独立的开源审核机构或小组，在软件项目开源之前对项目进行合规性审查。审核小组一般由独立于业务的开源专业人士组成，他们对软件开源、技术发展趋势、著作权、法律、专利等须有深刻和专业的理解。



---

合规审查关注几个要点。第一，开源代码和文档中是否存在泄漏用户隐私和商业秘密的情况，如果有则应当移除。第二，确认已经解决已知的安全漏洞（包括自研代码和引用的第三方代码），如软件注入漏洞等。如仍然存在显著漏洞，应予以修补，避免开放被使用后影响公众安全。第三，确认软件引用的第三方资源和软件列表，使用专业工具进行扫描认定（例如 Blackduck Protex、FOSSID 等），并于业务方提供的引用列表进行比对确认，防止遗漏。第四，根据确认的第三方引用列表，逐个判断软件的使用在开源许可证等方面是否合规。第五，在专利、商标、著作权、用户隐私等方面，确认项目的开源与使用没有侵犯他人权益。最后经过总结和沟通给出审核意见，并对开源项目的业务团队予以一定程度的合规化培训。

经过业务评估、技术评估、合规审核之后就可以开始准备开源了。

## 4.2 开启开源项目

**确定开源项目。**确定开源项目对外名称、logo，一般选取易于辨识的名称。

**确定托管平台。**确定代码共享平台，GitHub 是目前使用率较高的代码托管平台。

**选择许可证。**启动开源项目前，必须确定使用的许可证、MIT、Apache 2.0、GPLv3 都是流行的开源许可证，根据开源项目特点及开源的目的，选择合适的许可证。比如 GPL 不适用于商业软件或者对代

码有保密要求的部门；**LGPL** 适合作为第三方类库被商业软件应用，但是不适合希望以 **LGPL** 协议代码为基础，通过修改和衍生的方式做二次开发的商业软件使用；**Apache** 对商业应用友好，使用者也可以在需要的时候修改代码来满足需要并作为开源或商业产品发布；**BSD**：允许在代码上开发商业软件发布和销售，很多企业选用开源产品首选 **BSD**，因为可以完全控制第三方代码，在必要的时候修改和二次开发。

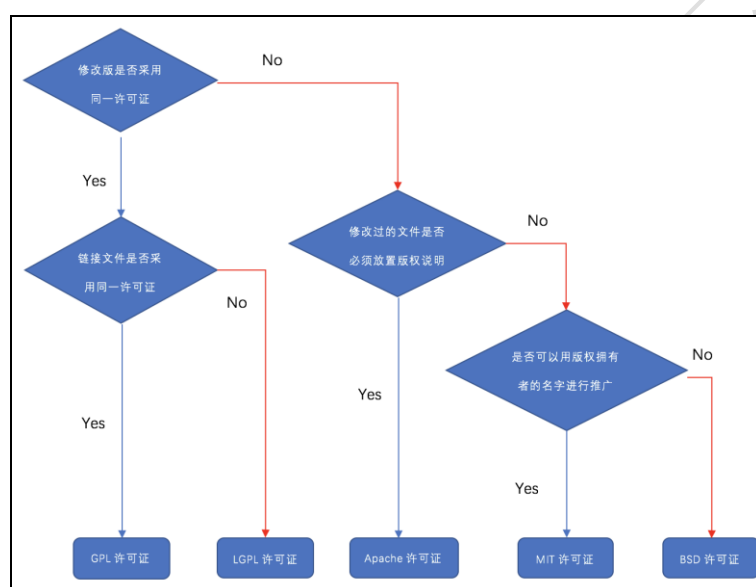


图 7 许可证选择示意图

编写项目说明。对所开源的项目进行具体描述，说明可以为用户做什么，包括项目介绍，应用场景，如何部署，如何开始，如何获取帮助，实现的目标等。

编写贡献说明。贡献说明里需要包括如何提交错误报告，如何建议新功能，如何配置环境和运行测试

建立行为准则。行为准则是一份确立项目参与者行为规范的文件，可以帮助你们促进健康、有建设性的社区行为，积极主动减少参与者

---

项目中变疲劳的可能性，并帮助你们在有人做出你们不同意的事情时采取相应行动。准则应该尽早建立，最好使用已有的准则比如贡献者盟约、Django 行为守则、Citizen 行为守则等。确定准则后，应确定如何执行行为准则，包括收集违规行为的渠道、调查违规行为的方式等。作为项目的维护者，应尽最大努力执行行为准则，对违反准则的人，采取恰当的回应方式和惩罚措施。

- 贡献者盟约是一个被 40000 个开源项目所使用的行为守则，采用 Creative Commons Attribution 4.0 International Public License。盟约期望能够让不同种族、地区、性别的人参与到开源中，营造一个友好、协作、互相理解的环境。你可以在项目中创建 CODE\_OF\_CONDUCT.md 文件并通过这个文件表达你对开源项目参与者或贡献者的尊敬和感激。使用贡献者盟约，需将盟约的文档添加到项目中，同时必须添加一个联系方式，以便人们揭露违反盟约的行为或事件。即使将盟约添加到项目中，仍会出现骚扰、歧视等问题，作为一个项目维护者，你必须致力于执行守则，使用盟约之前，花点时间讨论并决定如何处理可能会出现的各种问题。将强制执行策略和过程记录下来，并将其添加到 README 中或其他适当的位置，一定要考虑你的项目团队是否有意愿和成熟度来贯彻执行这些程序。

- 
- Django 行为守则同样采用 Creative Commons Attribution license, 是 Django 团队和社区为解决成员多样性导致的沟通交流问题所设定的。守则包括六大要求, 分别是: 友好和耐心, 欢迎各种背景和身份的人加入, 能够周全的考虑同事、用户, 尊敬他人, 举止、言语恰当, 面对分歧相互理解。
  - Citizen 行为守则旨在为 COMMUNITY\_NAME 社区所有成员提供一个友好、安全和包容的环境, 不论性别、性取向、能力、种族、社会经济地位和宗教(或缺乏)。守则规定了社区允许的行为和不允许的行为, 如果社区成员违反了该守则, 社区组织者可采取任何他们认为适当的行动, 包括暂时禁止或永久驱逐社区而不必提前警告(有偿活动时不退还相关费用)。

### 4.3 构建和维护开源社区

培育开源用户。线上推广, 在相关开源社区推广; 线下活动, 针对用户举办研讨活动。

完善贡献机制和行为准则。在社区运行过程中, 修订和完善行为准则。

记录工作进程。让更多的人参与到每一个过程, 保持项目透明。

---

积极回应反馈。对于参与者提出的项目请求，应尽量回答，对于无法及时回复的情况，也要尽早确认，提高用户参与度。

建立公共沟通渠道。通过建立公共的沟通渠道保证社区公开透明。

#### 4.4 开源社区治理

内部治理：当社区规模不断壮大时，为提高社区运行的效率和效能，社区将开展内部治理。根据模块化、角色分工、决策者、培训、流程与文档、制度化、领导力综合维度的高低，治理结构分为两种：独裁的治理结构和民主的治理结构。前者强调自上而下的管理，后者强调自下而上。

- “贤明君主”模式：项目开发人员一般都在百人以上，任何自由程序员都可以提交自己的修改工作，但只有项目主管才能合并到正式的发布版。

- “民主”模式：核心开发小组一般都在百人以上，分成若干小组，每个小组都有 1-2 个领导者，主导开发者定期召开开发者大会讨论。

- “精英”模式：项目开发人员权限分明，开发人员通过对项目作出贡献获得认可来影响项目。

外部治理：当外部参与主体增多时，社区进入外部治理阶段。第一类外部参与主体是基金会。基金会的参与可能会在一定程度上改变开源项目的发展方向和进展。目前介入开源的基金会主要有两种：第一

---

种积极支持社区及其成员发展，且给与其充分的自由决策权，包括对软件的版权许可。第二种，如 GNOME 或 Apache 基金会，直接支持开源软件项目，公司或社区一旦使用其核心代码作为商业软件，需要注明该基金会的版权，基金会对于软件的更新、发布、商标等有直接责任（de Laat, 2007）。第二类外部参与主体是公司。公司参与开源，多从战略角度出发，利用商业模式不同程度地介入开源软件项目（Cohen & Levinthal, 1990），或者以与开源社区合作开发的形式间接影响项目发展（West & Gallagher, 2006）。

## 4.5 开源项目评价

### 4.5.1 开源项目活跃度评价

贡献者数量。包括外部和内部的比例，开源项目起初大部分来自内部开发人员的贡献，随着代码的使用和演化，越来越多的外部人员参与贡献，可持续性的开源项目应该具有多样化的社区。

提交问题数量。问题的数量以及如何解决这些问题可以反映开源项目的开源使用水平以及维护人员对用户需求的响应程度。

每个贡献者提交的数量。包括外部和内部，这有助于评价开源项目是否持续的吸引新的贡献者。

外部采用者的数量。每个开源项目都应该有一个跟踪采用该软件组织的方法，如果外部使用者的数量随着时间推移而增长，说明开源

---

项目往健康的方向发展，如果采用者数量逐渐减少，说明这个开源项目有过时的倾向。

流行程度。项目网站的访问者，GitHub 的关注者，社交网络关注者，新闻提及，活动频率。

影响程度。贡献者的多样性，下载数，外部贡献公司数量，采用数量，衍生的商业产品的数量和质量。

#### 4.5.2 开源项目成熟度评估

国际上比较流行的开源项目评估模型 OSMM Capgemini, OSMM Navica, QSOS, OpenBRR, QualiPso Opensrouce Maturity Mode。

OSMM: Creative Commons 许可下发布，能够在企业使用或开发 Free/Libre Open Source Software (FLOSS) 中建立信任。该模型划分了不同的等级，每个等级包括不同的信任元素 (trustworthy elements, TWE)，TWE 有两个来源：从 FLOSS 开发者、用户等处调查收集得到，CMMI 过程领域。OSMM 包括三个成熟度等级，基础等级、中间等级和高级等级。

QSOS (Qualification and Selection of Open Source software) : GFDL 许可下发布，该模型定义了一个四步迭代过程：1、定义和组织将要评估的内容 (常见的开源标准、风险以及技术领域特有的功能)；2、根据上面定义的标准评估竞争软件并对标准评分；3、将标准组织

---

到评估轴中，定义权值等滤波；4、通过在步骤 3 中设计的过滤系统来评分所有竞争软件，选择适当的 OSS<sup>5</sup>。

**OpenBRR:** 定义了一个开放、标准的评估过程，目前处于 RFC 阶段，该模型试图整合公司约束（特别是测试和可靠性），并重点关注共享和成本节约问题<sup>6</sup>。

参考上述成熟度评估模型，开源项目成熟度评价可以考虑以下几个方面：

--可用性。软件功能和操作是否符合用户预期，是否能解决用户实际问题。

--项目质量。项目代码是否遵循行业开发标准，项目是否有自动化测试工具，项目是否遵循严格的发布流程。

--性能。该软件项目性能是否能够满足正常的批量或并发使用需求。

--文档。该项目是否有规范的使用文档和开发文档。

--社区建设。该项目的社区人数是否达到一定数量，社区是否活跃。

--服务与支持。该项目是否有专职人员做社区服务，用户遇到问题是否能有途径获得社区帮助。

--许可证清晰。该项目是否有清晰的许可证说明。

---

<sup>5</sup> <http://www.qsos.org/>

<sup>6</sup> <http://www.openbrr.org/>



---

--有落地用户验证。该项目是否已经有用户在实际生产环境中使用。

## 五. 开源商业模式

随着开源文化普及，开源软件得到认可。但用户要使用开源软件，需要有公司对开源软件进行技术支持，而产生了商业的需求。另外规模企业过去采购软件时，时常遇到软件升级费用非常高，以及无法替换供应商等问题，开源软件其开放性可解决企业与供应商的绑定，增加企业在软件销售或者升级的议价权。增加了对开源软件的商业需求。

双许可证。所谓的双许可证是指其软件是基于开源许可证的，但是还有不同的许可条款。用户可以无偿使用无须付费的、开源的版本，这仍然属于商业版本的一部分，若用户有进一步的需求，诸如商业的技术支持和服务等则需要另行付费了。作为全球领先的数据库软件，MySQL 产品采取了开源许可与私有许可的双重许可模式。MySQL 公司对产品代码拥有完整的著作权 (copyright)。在开源许可之下，软件的源代码完全公开，任何人都可以下载 MySQL 软件来使用、修改和传播。如果某商业客户希望在其商业软件中集成 MySQL 并保持原有软件的私有性，那么必须选择私有许可，即向 MySQL 公司支付一定的许可费。采用混合许可的优点在于通过许可协议差异化来最大化产品网络外部性带来的收益。

---

提供增值功能。开源软件的一个典型商业模式，就是通过提供增值功能获得用户的付费。用户在使用基础功能的过程中，不需要付费，但用户的某些特殊需求，可以通过增值功能来满足，用户需要这些增值功能的情况下，可以付费购买。

销售企业就绪产品，提供专业服务。开源开发模式是更快速促进创新的最佳模式。企业需要产品，而不是项目，许多开源公司不了解项目与产品之间的这种差异，提供的解决方案缺乏稳定性、兼容性以及其他各种能力或非功能需求，而这些恰恰是企业客户运行其关键任务应用所必须依赖的东西。红帽是市场上最大的开源软件解决方案提供商之一，实行产品订阅的收费模式，从上百万项目中嗅出企业可能需要的项目与功能，然后利用自身的专业能力打造出标准化商用产品，搭配企业级服务。

以 SaaS 形式出售。有了云计算之后，以软件即服务的方式赚取利润，不失为一种可行的办法，尤以应用软件、CMS、HR、ERP、CRM 等系统，让在线用户按需付费、即用即付的订阅方式来完成整个过程，而软件本身是开源的。

广告。在开源软件中安装广告的播放插件，一旦安装后服务器就可以发布广告。这种模式适用于互联网厂商，适用于任何许可证。

依商业许可重新发行。一些宽松的许可证，如 Apache、BSD 等，是允许以商业且闭源的方式二次发行的。这其中最为著名的例子就是苹果公司的 MacOSX 操作系统，其内核是使用的 BSD Unix，但是其二

---

次发行也是顺理成章。这样的方式，也是我们本土常见的方式，比如 OpenStack 采用是非常宽松的 Apache 协议，再次商业发行，包括自己修改的、新增的代码是可以不开源的。

建立生态。通过将私有项目以开源方式运作来建立生态，号召更多的开发人员来参与并基于开源系统开发，随着开源系统在开发人员中变得流行，甚至成为行业标准，便可以利用生态产生收入。例如谷歌在 2008 年开源 Android，现在，Android 成为使用最广泛的手机操作系统，有超过 40 万的开发人员和十亿多用户。Android 设备上投放广告带来的收入和应用程序开发商使用 Android 系统支付的费用，一年能为谷歌带来数十亿美元。又如，谷歌在 2015 年 11 月时将 TensorFlow 开源，随着 TensorFlow 被全世界各地的开发人员采用并流行起来，谷歌便可以推出一个在谷歌云平台上运行的 TensorFlow 版本，个人和企业用户在谷歌的数据中心运行 AI 软件时需要付费。

总结各类开源的商业模式，可以发现开源目前有四种长期的盈利或补偿方式：(1) 销售企业就绪产品+配套专业服务；(2) 细分权利按服务项目收费；(3) 绑定收费；(4) 间接获益。其中最重要的是前三种。这三种盈利方式，也是开源企业实现长期、规模盈利必须满足两个必要条件。一是产品具有原始创新(高增值)+商业模式创新（收费模式/服务模式），二是企业产品链足够长，软件作为集成产品的一部分(整合创新，价格转移)。

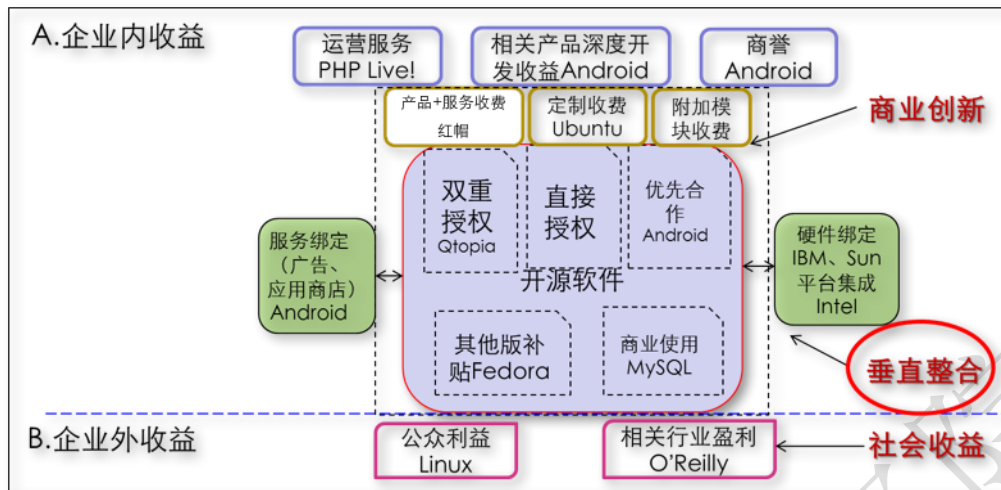


图8 开源商业模式的创新及条件

## 六、开源技术的使用

开源技术使用者可以分为三类。第一类是使用开源技术的用户，是开源软件的终端用户；第二类是进行二次开发并使用的开源技术的用户，是使用开源软件代码用于自身业务开发的开源使用者；第三类是进行二次开发并进行售卖的软件厂商，是将开源软件集成在自身产品或解决方案中，向终端用户进行分发销售的软件厂商或集成服务商。

根据开源用户的三种类型，开源技术的使用方式同样可分为三种。第一种是直接使用原生开源技术，第二种是对开源技术进行二次开发再进行使用，第三种是购买基于开源技术的商业软件。

---

## 6.1 使用开源技术的风险

不管是直接使用开源技术，还是购买开源技术的商业软件都需要考虑开源技术带来的风险，不同的是规避开源技术风险的责任主体不同。

企业在使用或引入开源软件或代码时，需要考虑开源风险、违约风险、知识产权风险、开源许可证兼容性风险，以及数据安全及隐私风险，具体可参见第七章开源面临的问题和挑战。

## 6.2 如何选择开源技术

由于开源软件可以帮助一个公司或者研发机构快速的开发出适合市场需要和科研需求的平台和产品，因此在面临产品研发和项目方案制定时，选用开源软件就是一个非常快速有效的手段。而是否以及如何引入开源软件，就要从以下几个方面来加以考虑：

- a) 需求满足度：对于终端用户来说，在选择开源技术的时候需要考虑是否满足自身业务场景需求，对于软件厂商在选择开源技术的时候需要考虑是否满足用户的需求，如果不完全满足用户需求，则可能需要进行二次开发。一般来说，选用的开源软件应满足绝大多数业务需求才能选用。因为一般来说开源软件的方案和代码大多数都不是选用者自己贡献的，因此在后续软件开发和维护中存在因不熟悉开源软件现有方案和代码带来的

---

额外问题及其相关工作量。如果开源软件对产品需求的满足度不够高，则上述问题给选用者带来的额外工作量可能会超过彻底重新自行研发的工作量，这样就得不偿失了。

- b) 技术先进性：即这个开源软件的技术在同类闭源或开源软件中所处的水平是否先进。这点的主要考虑因素是因为开源软件众多，而开发者也有自己的技术偏好，但这种偏好不一定符合未来技术发展趋势，因而需要进一步衡量。
- c) 开源许可证：一般来说应尽可能选用具有比较宽松的开源许可证的开源软件。对于选用如 GPL 系列的相对比较严的许可证的开源软件，应注意在产品中的使用方式。需要防止 GPL 系列许可证软件带来的开源传染性问题的。
- d) 软件成熟度。应尽量选择相对比较成熟的主流开源软件，降低后续技术跟随方向偏离业界主流的风险。一般建议选用业界 Top3 的开源软件。
- e) 运维能力。在选择开源软件前需要查看是否具有相对比较完善的开源方案日志、是否具有命令行、管理控制台等维护工具能够查看系统运行情况，是否有故障检测和恢复能力。
- f) 商业支持。开源的源代码每年都急剧上升和翻倍增长，对于使用免费软件的企业用户带来了巨大的挑战，涉及技术路线选择、稳定性、安全性和可持续性开发等，需要考虑市场上是否有可以提供企业就绪型产品与配套服务的开源厂商。

---

g) 社区活跃度：即开源社区是否仍然活跃。从技术上说即是这个开源软件未来是否可以持续更新，如果已经很多年没有更改，则有相当可能这个社区已经无人维护，不能根据未来的技术发展添加更多功能，保持技术活力。

h) 软件生态：应优选生态系统比较完备的开源软件。优选有实力比较雄厚的基金会支持的，有业界主流厂商积极参与贡献和选用的开源软件。

### 6.3 如何使用开源技术

由于开源技术在使用过程中存在的风险问题，所以应分步骤，有节奏的使用开源技术。通常先熟悉开源软件进行测试，规划应急预案，为满足业务场景需求进行二次开发。

测试。通读开源项目的设计文档，了解设计原理；核对每个配置项的作用和影响，识别关键配置项；进行多种场景的性能测试；进行压力测试，连续运行程序，分析 CPU、内存、磁盘 IO 等指标；故障测试，模拟断电、拔网线、重启 100 次以上等故障。

灰度发布。先在非核心业务上使用。

应急预案。对于重要的业务、使用开源技术时，最好有另外一个成熟的方案做备份。

二次开发。二次开发过程需要先确定开源软件跟随策略，再设计编码，测试发布。

- 
- 确定开源软件跟随策略。开源软件跟随策略一般分为两种。一种是“私有分支”策略，一种是“社区先行”（Upstream First）策略。

**私有分支策略**即在公司内建立基于开源社区某个版本的私有的产品版本分支，并在其上增加定制功能，经测试后发布给用户使用。后续私有分支的版本随开源社区版本发展而发展。私有分支版本前进一个版本时，选择社区的一个新版本（不一定跟随每个社区版本），然后合入以前开发的定制功能。这种策略为国内外很多厂家采用。其主要优点是可以比较迅速的反应用户的需求，及时的提供可用的解决方案和软件给用户使用。同时，作为差异化竞争力的私有定制功能可以不开源，增强产品功能方面的竞争力。缺点是由于存在私有定制功能，导致跟随社区版本发展时每次都需要进行功能整合并进行大量测试。这会消耗很多的工作量。而且存在未来社区中也出现了类似功能，导致私有功能与社区实现不兼容的问题。

**社区先行策略**指商用产品所需的所有功能均首先在开源社区中进行开发，并贡献到开源社区。商用版本直接继承开源社区版本的功能，并根据产品需要进行模块和功能裁剪。同时进行商用标准的测试工作。在测试中发现的故障，经修改后也及时贡献到社区。这种策略的主要应用厂商是



---

红帽 (Red Hat)。此策略的优点是，可以快速的跟随开源社区版本。防止每次版本跟随时需要额外合入大量自行开发的私有功能，消耗大量工作量。缺点是因为社区版本节奏相对比较慢，从而导致商用版本在响应用户需求方面有时会滞后。另一方面讲，这也保证了企业用户可以使用到更成熟稳定的开源技术产品。

- 方案设计。基于开源软件的二次开发的方案设计，除按照普通功能开发流程以外，还应特别关注以下方面：
  - a. 需要看社区中是否已经在进行类似功能的开发，如果已经正在进行开发，则可以等社区开发完成。
  - b. 需要考虑与社区现有接口的兼容性，防止未来存在不兼容的风险。
  - c. 使用开源软件中的接口时要注意接口版本问题。防止出现依赖即将被废弃接口的问题。
  - d. 需要考虑开源许可证的要求。对于限制比较严格的许可证，需要考虑新开发功能的存在形式。是直接修改开源代码，或者采用库的形式，还是采用独立服务的形式等。
- 编码。基于开源软件的二次开发的编码，除按照普通编码的要求以外，还应特别关注以下方面：
  - a. 编码需要注意遵循开源许可证的要求，特别需要注意不要清除原版权信息，并添加开发者的版权信息。

---

b. 需要遵循开源社区的编码规范和要求，便于后续在需要时将代码贡献到开源社区。

- 测试。基于开源软件的二次开发的测试，除按照普通测试的要求以外，还应注意尽量引入开源社区原 CI 等测试系统和工具链。
- 发布。可采用灰度发布的方式。
- 运维。采用 DevOps 运维方式。需要有完善的版本升级方案。支持跨社区大版本的，（尽量）不中断业务的平滑升级方式。

## 七、开源面临的问题和挑战

个人或企业在使用或引入开源软件或代码时，在法律或业务执行层面，可能面临不同的风险，如：1. 开源风险，2. 违约风险，3. 知识产权风险，4. 开源许可证兼容性风险，以及 5. 数据安全及隐私风险。以下将就各类风险进行简要介绍：

### 7.1 开源风险

个人或企业在使用或引入开源软件时，可能面临到的开源风险（即公开源代码），主要在于个人或企业对于其持有或拥有的私有软件或代码，因为使用或引入了适用传染型或弱传染型开源许可证的开源软件或代码，而依该传染型或弱传染型开源许可证所规范的义务或要求，将导致其私有软件必须对外公开源代码。

---

传染型开源许可证。一般常见的传染型开源许可证，如：GNU General Public License Version 2 或 Version 3（下称“GPL V2”或“GPL V3”），当个人或企业在其私有软件，使用或引入适用此类传染型开源许可证的开源软件，又进行二次分发时（如：企业将适用 GPL V2 的开源软件与其私有软件进行结合，并将该结合产生的衍生作品，分发至企业外部或客户），如此将触发此类传染型开源许可证的开源义务，将可能导致个人或企业需将其私有软件对外公开源代码。

弱传染型开源许可证。一般常见的弱传染型开源许可证，如：GNU Lesser General Public License Version 2.1（下称“LGPL V2.1”），当个人或企业在其私有软件，使用或引入适用此类弱传染型开源许可证的开源软件，并对该开源软件进行特定行为（如：修改），又进行二次分发时（如：企业将适用 LGPL V2.1 的开源软件进行修改，与其私有软件进行结合，并将该结合产生的衍生作品，分发至企业外部或客户），如此将触发此类弱传染型开源许可证的开源义务，将可能导致个人或企业需将其私有软件对外公开源代码。

## 7.2 违约风险

目前各国法律或法院，对于开源软件使用者，在违反开源许可证的义务或要求的情况下，是否构成合同违约，仍未有一致规定或见解。然而，在2017年，美国加州北部地区联邦法院的 *Artifex Software*,

---

Inc. v. Hancor, Inc.<sup>7</sup>一案可以作为日后参考，该案承审法官明确了，该案被告(即开源软件使用者)，在违反开源许可证(GPL V3)的义务或要求的情况下，除可能构成知识产权侵权外，亦可能构成合同违约，而原告(即开源软件作者或权利人)在此情况下，除可寻求知识产权侵权救济外，也可以从合同法上寻求救济。

### 7.3 知识产权风险

开源软件提倡公开、自由与创新等开源精神，为推动软件产业的发展起到了积极作用。但是，个人或企业在使用或引入开源软件的过程中，将不可避免地面临知识产权上的风险。如个人或企业在使用或引入开源软件，因为不了解知识产权风险而引起相关法律或商业争议，将可能给个人或企业在经济或声誉等方面带来巨大的损失。

除法律法规的保护外，开源软件的作者或权利人主要是通过开源许可证对其知识产权进行许可与约束。开源许可证从法律上来说，可视为一种合同和许可，是开源软件使用者得以依开源许可证来合理使用开源软件的合法凭证，并且在该使用者有相应行为(如：使用开源软件)时，就可视为该使用者与开源软件的作者或权利人意思表示一致(即：达成合意)，而愿意接受开源许可证的约束。若开源软件使用者未依照相应的开源许可证，来使用开源软件，将可能侵犯开源软件的作者或权利人的知识产权。

---

<sup>7</sup> *Artifex Software, Inc. v. Hancor, Inc.*, No. 16-cv-06982-JSC, 2017 U.S. Dist. LEXIS 62815, Doc. 32 (N.D. Cal. Apr. 25, 2017).

---

此外，多数开源许可证中皆含有免责条款，声明适用该开源许可证的开源软件系依“现状”（AS IS）提供，其作者（或贡献者）不提供任何担保责任（无论明示或默示），且若因使用或引入该开源软件而引起任何责任，其作者（或贡献者）将无须负责。因此，在使用或引入开源软件的过程中，若因该开源软件曾使用或引入第三方私有软件或代码，则后续个人或企业在使用或引入该开源软件而遭第三方权利人诉讼时，该个人或企业将可能需负担相关侵权责任，而无法向该开源软件的作者（或贡献者）求偿。

一般来说，个人或企业在使用或引入开源软件的知识产权风险，通常表现为以下几个方面：1. 著作权，2. 专利权，3. 商标权，以及4. 商业秘密。

#### a) 著作权侵权

除在商业或学术领域外，开源软件通常是由个人或一群彼此之间没有正式联系的开发者所共同完成的，每个开源软件的开发者都是该开源软件的贡献者。在这种特殊的开发模式下，个人或开发者可能由于缺少法律意识，很容易出现开源软件的权利归属混乱的情形。一般而言，除了开源软件原始作者外，其他任何参与开源软件的贡献者，在法律上可能都不是开源软件的著作权所有人。

在使用或引入开源软件时，可能还会面临著作权瑕疵和著作权陷阱的问题。由于参与开源软件开发的贡献者可能人数众多，且任何开源软件使用者只要遵守开源软件许可证的义务与要求，都可以自由的

---

使用、修改或分发开源软件，因此很可能会导致侵权代码的流入开源软件中，从而使开源软件或其衍生作品存有著作权侵权的风险。

另外，在使用或引入开源软件时，若未遵守相应的开源许可证时(如：未依开源许可证提供源代码，未附上开源软件作者或权利人的著作权声明或开源许可证原文，或未注明修改信息)都可能因此侵犯开源软件作者或权利人的著作权。

### **b) 专利权**

由于著作权主要保护的是作品本身的表现形式(如：代码的呈现方式)，而专利权则要保护的是思想与观点(如：软件或代码上的发明构思或技术方法)，两者保护的对象、条件与要求等皆有不同，因此个人或企业在使用或引入开源软件时，如未遵守相应的开源许可证，则可能同时侵害开源软件作者或权利人的著作权与专利权。

因为开源软件的特殊开发模式，在开源软件的开发、改进或分发过程中，可能融入了很多贡献者的贡献，若未进行事前规划或排查，也极有可能侵害第三方的专利权。

专利许可。开源软件所倡导的自由共享精神，与专利权所要保护的独占性与排他性，在本质上存有差异。因此，部分开源许可证(如：GPL V3)则含有明确的专利许可条款，许可开源软件的相关专利权给开源软件使用者，使该开源软件使用者得以依据该开源许可证使用该开源软件。另外一些宽松开源许可证(Permissive License，如 BSD、MIT)则没有明确的提到专利许可条款。因此，在使用或引入该等适用

---

未存有专利许可条款的开源许可证的开源软件时，是否会构成专利侵权，目前仍未有定论。在存有明确专利许可条款的开源许可证(如：GPL V3)中，开源软件的作者、贡献者或权利人需要将其在该开源软件中的相关专利权，向该开源软件使用者进行许可，因此在使用或引入这类开源软件时，若确实遵守该开源许可证的义务与要求，则侵害开源软件的作者、贡献者或权利人在该开源软件中的相关专利的侵权风险较小。然而，在使用或引入没有明确专利许可条款的开源许可证(如 BSD、MIT)的开源软件时，则可能蕴含着一定的专利侵权风险。

专利报复条款。在部分开源许可证中其实都有明确或隐性的指出专利相关的权益。因此为了防止有人恶意提起法律诉讼，部分开源许可证包含“专利报复”条款，如果开源软件的使用者对任何第三方提出专利侵权的主张或诉讼，主张这个开源软件侵害其所拥有的专利权，此时该开源软件的作者或权利人对于这位使用者的相关专利授权将会反制性地被终止。前述被终止的专利许可范围随报复条款的规定而有所不同，有些报复条款可能仅终止专利许可，但是有些条款也可能终止整份许可证所许可的权利，在后者的状况，也就是说著作权方面的许可都会一并连带被终止，从而该名提出专利侵权主张的使用者，将自此无法再使用、复制、修改与分发该开源软件。

### c) 商标权

开源软件的商标分两种类型：开源社区的商标；开源软件商标。Apache, Linux 本身就是一个商标，开源社区组织作为一个自发建立

---

的非官方组织，为了在开放源代码软件领域实现统一标准的目的，将自己定位为一个行业协会性质，在一般情况下，使用开源社区的商标是需要经过正式许可并付费。未经开源社区正式许可使用开源社区的商标，将可能会构成商标侵权。

许多开源软件都申请注册了相应的商标。开源软件的权利人进行开源并不代表其授予商标的使用许可，一般开源软件的权利人都会保留商标的许可。因此，如果未经正式许可使用了开源软件的商标，可能会构成商标侵权。因此，为避免商标侵权风险，最好在开源软件发布或分发之前，认真检查开源项目名称是否与已经存在的开源项目的商标冲突。

#### **d) 商业秘密**

不同开源许可证下的开源软件，其对使用者的限制与义务亦有所不同。以 GPL 类的传染型开源许可证为例，开发者在自己的私有软件或代码加入了 GPL 开源软件或代码时，将受 GPL 类开源许可证的“传染”，而需要依据 GPL 类开源许可证的义务和要求，将其私有软件或代码进行开源，如果开发者自己的私有软件或代码原本是一项商业秘密/技术秘密，但是因为使用了 GPL 类的开源软件或代码而导致需要将其私有软件或代码进行开源，将使其本身或其所属企业的商业秘密/技术秘密被迫公开。

开源软件涉及著作权，专利权，商标权等综合知识产权问题，个人与企业在进行开源时应选择合适的开源许可证，充分了解开源许可



---

证内容，严格遵守开源许可证的义务与要求、谨慎使用开源软件的商标和标识、对开源软件进行扫描检查、控管第三方具有风险的代码引入、确定第三方代码来源和开源许可证。

若企业建立完善的开源软件管理体制和流程，规范开源软件与代码的使用，将能有效控制开源软件带来的知识产权风险。

## 7.4 开源许可证兼容性风险

由于各类开源许可证的义务与要求存有差异，因此在使用或引入开源软件时，需要注意各开源软件所适用的开源许可证的兼容性风险。以传染型开源许可证中常见的 GPL 开源许可证为例，在 GNU 的网站上<sup>8</sup>，即详细列出何种开源许可证是否与 GPL 开源许可证兼容。若在使用或引入开源软件时，未注意兼容性风险，则可能导致违反相应的开源许可证义务或要求。

## 7.5 安全风险

由于前述开源软件的特性(如：由多个贡献者共同完成、开源许可证存有免责条款等)，个人或企业在使用或引入开源软件时，也必须注意数据安全及隐私风险，否则若使用或引入的开源软件存有恶意代码、病毒或造成隐私泄露，将对个人或企业带来不小危害。

开源软件存在的安全问题较为严重，安全漏洞是主要的问题，同时后门等问题同样存在。开源软件的安全缺陷密度较高，根据 NVD 数

---

<sup>8</sup> <https://www.gnu.org/licenses/license-list.en.html#GPLIncompatibleLicenses>

据统计，截至 2017 年 2 月，全球开源软件相关的已知安全漏洞已超过 28000 个。目前很多企业在软件开发过程中，对开源软件的使用流程不规范，不能完全列出开源软件使用列表。早在 2006 年，美国国土安全部就开展“开源软件代码测试计划”，对大量开源软件进行安全隐患的筛选和加固，截至 2017 年 2 月，累计检测各种开源软件 7000 多个，发现大量安全缺陷。系统信息泄露、密码管理、资源注入、跨站请求伪造、跨站脚本、HTTP 消息头注入、SQL 注入、越界访问、命令注入、内存泄漏是开源软件主要的安全风险。

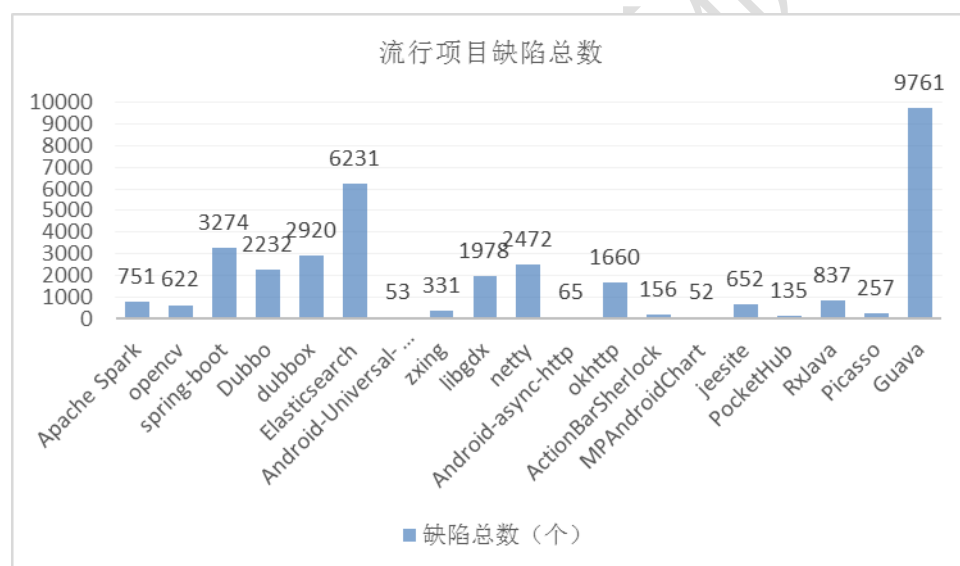


图 9 开源软件项目缺陷总数

由于开源软件使用与取得的便利性，可能导致忽略开源软件带来的风险，因此个人或企业在使用或引入开源软件时，除从技术或商业层面进行考量外，也需注意前述开源软件可能带来的相关风险，或设立合适的管控机制，以降低相关风险，避免造成重大损失。

---

## 八、总结

开源可以有效推动技术创新，降低企业研发门槛，消除国际技术壁垒，带动国家信息科技发展。与此同时，开源带来的知识产权、法律、安全等风险问题同样应该引起关注。近几年我国企业参与开源形式的热情逐渐提升，开始探索自主开源的可能性，国家应为开源发展提供良好的政策环境，企业应抓住开源带来的机遇，利用开源推动企业发展。

我国应培育开源发展的政策环境。完善开源的法律保护机制，加强开源软件的社会认知度，推动开源社区发展，支持开源社区进行培训和研讨活动，加强开源人才培养，支持国内开发者参与国际主流开源项目，同时推动国内自由开源项目在国际的影响力。

开源相关企业寻求在开源生态的立足点。积极参与国际开源项目，了解技术发展趋势，提升社区地位和话语权。要提高开源风险意识，遵守开源许可证，匹配企业发展策略，依托开源寻求友好的商业模式。探索自主开源发展模式，引领的开源生态圈，提升行业影响力。

---

## 附录：开源相关企业概述

### 一、腾讯控股有限公司

#### 1、开源概览

腾讯是目前中国领先的互联网增值服务提供商。2010年后，腾讯开始全面拥抱“开放”战略，技术研发也在这种环境下，开始向共享、复用和开源迈进。

从2011年开始，腾讯在内部提倡以公共组件的形式共享和复用代码；2012年，腾讯尝试发布了第一个对外开源项目；2013年，旨在鼓励和推进腾讯内、外部开源落地执行的腾讯代码文化项目启动；2014年，制定和发布了腾讯的开源策略和具体流程，并发布第一批的6个官方开源项目；2015年，在腾讯技术管理委员会的支持下成立了腾讯开源联盟（TOSA），整体指导和管理腾讯开源工作；2016年，开始以GitHub为主要阵地进行开源项目经营；截止到2018年1月，正式发布的腾讯对外开源项目共计49个。截止到2018年一月，腾讯的开源项目累计在GitHub获得了11万Star，在国内和国际社区收获了解与认可。

#### 2、开源组织经验

经腾讯技术管理委员会授权，由腾讯研发管理职能部门牵头腾讯法务、合规、专利、安全等相关部门，整合为开源管理小组在开源流程、安全、风险、建议方面对腾讯业务提供专业服务。小组会帮助开源项目整理第三方组件的引用，通过第三方扫描工具（如blackduck, fossid等）识别和确认第三方代码有效性，同时会进行人工筛查，确保第三方使用腾讯代码时安全可控。在协议选取上，管理小组会对项目的开源许可证进行合规化的建议，并向业务推荐开放性的许可证。

在文化方面，由研发管理部CODE平台中心负责组织建设腾讯代码文化和内部开源，鼓励腾讯项目首先在腾讯内部流动，并对有价值的项目引导和帮助对外开源。同时，在腾讯技术管理委员会的授权下牵头设立了“腾讯开源联盟”，由不同业务的技术专家、负责人、技术领袖组成开源联盟组委会和专家团，在开源文化、开源经验、开源活动等方面对开源项目施以指导和帮助。

#### 3、自主开源项目

##### ➤ 微信开源

**微信开源的情况和项目简介。**微信是腾讯公司于2011年推出的即时通讯服务，截止到2017年第三季度，微信与Wechat的合并月活跃账号数已达9.8亿。自创立之初，微信一直在丰富软件功能以便为用户的日常生活带来便利。于此同时，微信也一直注重通过技术开源来回馈社会。截止到2018年1月1日，微信共开放了22个开源项目，在GitHub共获得约7万Star。

微信开源项目特点，一是覆盖广。微信开源覆盖了微信后台、微信终端、微信前端等方向，包含了网络基础组件、通信、存储、数据库、数据交换、生物认证、热补丁、网页开发、UI库等多个领域的基础和关键组件。这些组件均在行业中有广泛影响。二是开放透明。开源的项目在微信内部广泛使用，多数项目的开源版本，与微信内部使用的版本一致。三是持续性强。微信开源自2013年第一个项目起，一直坚持摸索技术开放。由于技术栈规划合理，所开源项目均在业务中长期使用维护。

**微信开源的组织形式和方法论。**一是行成开源指导小组。由微信基础产品部、技术架构部的技术专家和团队负责人组成微信开源指导小组，规划、引导和协助微信的项目开源。二是与开源流程团队充分协作。微信开源项目与腾讯开源工作小组充分合作，在开源合规、开源发布、开源宣传方面，引入腾讯集团资源，增强项目的发布、宣传能力。三是内部和外部交流。在腾讯公司内部，微信开源的兴趣社团举办了多次内部交流活动，通过分享交流会的形式，对内进行开源经验、治理方面广泛诚恳的交流。在对外传播方面，微信开源项目曾通过微信公开课、Qcon 等会议进行宣讲和释疑，取得了良好的反响。四是通过公司激励和奖励。例如：2014 腾讯代码文化行业贡献奖 (libco) 2016 年腾讯卓越研发奖 (PaxosStore) 2016 微信事业群 WeWIN 奖 (微信开源) 2017 腾讯开源特别贡献奖 (Tinker) 等。

**微信开源的行业意义和价值。**经过持续经营，微信开源已经逐渐形成如下几个优秀系列：微信后台系列的 libco、paxosstore、phxpaxos、phxrpc、phxsql、phxqueue 等项目服务了微信的后台海量架构系统，给行业提供了全面优秀的解决方案。微信终端系列包含 tinker 动态更新解决方案、mars 终端跨平台通讯解决方案、wcdb 终端跨平台本地数据库解决方案以及 soter 生物认证解决方案等，均在终端研发的各自领域中处于领先地位，充分展示了腾讯团队在移动开发方面的实力，并帮助很多开发者获取相应的技术能力。微信前端系列由 weui、vConsole、weflow 等知名项目为代表，为前端开发者提供了方便的开发工具和框架，帮助构筑了微信开发者生态，特别给微信开放平台的开发者用户提供了标准化的工具框架。

微信作为中国移动互联网第一大应用，通过自身能力的开源开放，为行业做出了表率。

#### ➤ 腾讯云 TStack

**TStack 项目简介。**腾讯云 TStack 是腾讯云基于自身强大技术能力和海量运营经验推出的专有云平台，提供集 IaaS、PaaS 和 SaaS 为一体的综合云服务解决方案，同时具备混合云管理能力，助力政府、企业构建稳定安全的云环境和繁荣的云生态。TStack 基于 OpenStack 开发并做了大量的优化和改进，整合了腾讯内部大量的 PaaS 能力和 SaaS 能力，如大数据、数据库、LBS、AI 等能力，形成了基础云，监控云和自助云三大产品。实现了异构云环境管理，混合云管理等关键技术，在腾讯内部服务 IT 业务，承载了 QQ、微信、游戏等业务的开发测试工作。提供集计算、存储、网络、安全、监控、运营等能力为一体的专有云解决方案，让用户聚焦于挖掘自身业务的价值。

**TStack 开源方法论。**一是**参加社区活动并分享经验。**在全球云计算开源大会上，TStack 分享了综合云的优秀经验，并通过了 OpenStack 互操作能力测试；参加了 OpenStack Days China 和 OpenStack Sydney Submit 等主题会议，发起了多个技术专题分享和主题演讲，向全世界 OpenStack 用户分享了 TStack 的实践经验，获得了 OpenStack Superuser 大奖。二是**积极贡献开源代码。**通过提交 bug 或者 blueprint 的形式反馈到 OpenStack 社区，并积极在 OpenStack 开源社区贡献代码，促进了开源 OpenStack 的成熟和稳定。

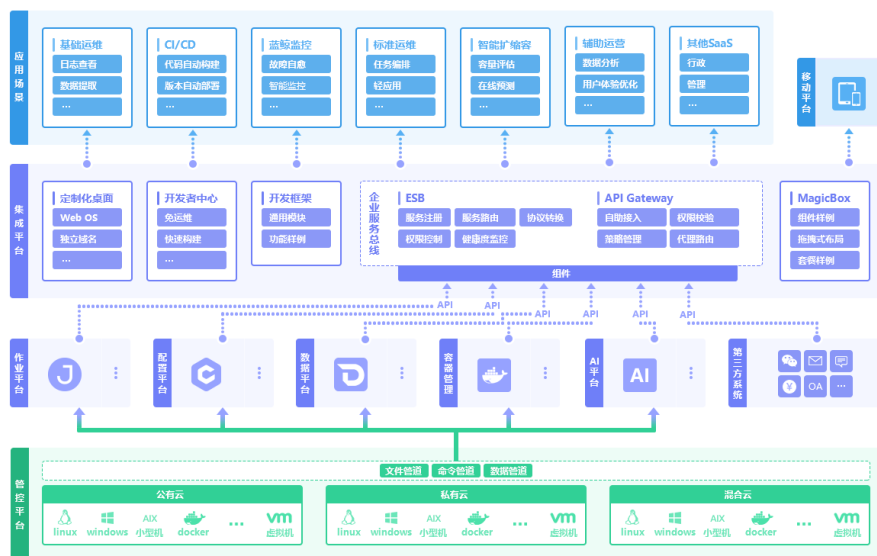
**TStack 开源的行业意义。**TStack 作为腾讯的专有云平台，在服务于腾讯内部的同时，腾讯云已将 TStack 平台以及运营服务经验推广到中国政企市场，并与全国 15 多个省、50 多个城市签署合作协议。

乘着“源于开源，回馈开源”的宗旨，腾讯云 TStack 平台在使用开源技术的时候，不断的把互联网海量实践经验不断分享到 OpenStack 开源社区，促进开源事业的繁荣和发展。

### ➤ 蓝鲸智云

**简介。**蓝鲸智云，简称蓝鲸，是腾讯游戏运营部沉淀多年的技术运营支撑体系，承担着腾讯数百款业务线上运营的使命。它是一套基于 PaaS 的技术解决方案，提供了完善的前后台开发框架、调度引擎、公共组件等模块，帮助运维快速实现“基础服务无人值守”及“增值服务”，并进一步通过 DevOps 的落地实现企业更全面和可持续的效率提升。

蓝鲸智云体系由八大平台组成，即管控平台、配置平台、作业平台、数据平台、容器管理平台、挖掘平台、集成平台和移动平台，为各种云（公有云、私有云、混合云）的用户提供不同场景、不同需求的一站式技术运营解决方案。



(蓝鲸智云体系架构图)

**开放&首个云计算运维平台参考框架标准。**多年以来，运维一直都是滞后于企业发展速度的一个领域，严重影响着企业的转型和扩张。建设一整套体系化的通用运维平台，一直是运维从业人员孜孜不倦追求的梦想，腾讯作为全球领先的互联网企业，依托复杂的场景和快速迭代的业务需求，天然的衍生出了这一套能够适配各类行业的一体化运维平台——蓝鲸，并主动免费对外开放了这一成果，回馈行业。一是免费对外开放蓝鲸社区版，助力企业快速搭建企业内的运维自动化体系，通过提升自身能力，低成本地构建运维工具，自建运营系统。二是建立公共社区，提供了从社区版搭建部署、功能指导、开发培训以及更新升级的一站式服务，通过详尽的文档体系、QQ群、BBS、线上培训、线下沙龙以及一对一的客服服务，为各行业的运维从业人员在企业内部落地蓝鲸提供了多种官方帮助渠道。三是与中国信息通信研究院、云计算开源产业联盟携手对外发布了首个运维行业标准：基于蓝鲸智云开源项目-《云计算运维平台参考框架标准》，为行业建设通用的云计算运维平台提供了历经实践验证的参考依据，解决了业界运维人员长期盲目建设的历史问题。

### ➤ TARS

**TARS 项目简介。**TARS(<https://github.com/Tencent/Tars>)是腾讯自研的高性能 RPC 与微服务治理和开发运营一体化解决方案。它基于 TARS 协议,在腾讯内部颇受欢迎,高效的支撑了各大核心业务。

TARS 同时支持多种计算机语言,在敏捷开发层面能支持快速构建及自动生产代码,具备服务发现、智能调度、容灾容错、柔性熔断等高可用特性,并提供完备的立体化监控措施及一整套完善的可视化管理平台助力高效运营。通过 TARS 可快速采用微服务的方式构建稳定可靠的分布式应用,实现完整有效的服务治理。对比其他已经开源的微服务开发框架,TARS 在多语言支持、协议扩展、开发运营、立体监控等方面具有更全面的功效和更强的竞争力。在外部与金融、教育、医疗、政企等行业的多家知名企业和创业企业进行深度合作,共同为互联网+生态添砖加瓦。

**TARS 开源方法论。**TARS 团队在不断进行技术革新的同时,将自身在微服务治理的经验、海量服务运营的经验在腾讯内外部进行分享,促进内部团队和外部合作伙伴共同进步、共同发展。首先,在团队内部建立开源氛围,通过设置个人及团队的开源贡献激励奖项,鼓励更多的人参与到开源活动里来。在 2017 年分别表彰了对 TARS 开源有着重大贡献或在开源社区积极活跃的 4 个团队及 2 个人,团队开源氛围浓厚。其次,在外部技术交流方面,TARS 对外输出多篇技术文章,覆盖多个知名媒体,《你所不知的 TARS,却帮阅文集团完成技术架构变革》、《腾讯与阅文技术合作 微服务框架 TARS 再添 PHP》、《大公司开源并非易事,如何用产品思维去做?》等文章的阅读量均在腾讯开源官方公众号的前列。同时积极参加各项内外部线下交流活动。最后,TARS 通过开源与外部企业建立合作,覆盖创业公司、知名企业,传统互联网业务到新兴的互联网+业务,在金融、教育、医疗、政企等方面均有应用,其中也诞生了不少优秀的合作伙伴。

**TARS 开源的行业意义。**TARS 作为腾讯的多语言高性能 RPC 框架及微服务一体化治理方案,能帮助开发者减轻负担,提升效率,将精力专注于业务逻辑。基于 TARS 部署运行的服务节点规模已经达到上万个。TARS 开源使其服务于腾讯内部的同时,在外部也能与数十家企事业单位深度合作,共同打造互联网+生态。

## 二、华为技术有限公司

### 1、开源概览

华为是全球领先的信息与通信技术(ICT)解决方案供应商,专注于 ICT 领域,坚持稳健经营、持续创新、开放合作,在电信运营商、企业、终端和云计算等领域构筑了端到端的解决方案优势,为运营商客户、企业客户和消费者提供有竞争力的 ICT 解决方案、产品和服务,并致力于使能未来信息社会、构建更美好的全联接世界。

开源作为软件行业的一种新趋势,华为多年来都在积极的参与开源,拥抱开源。2008 年,华为开始积极地参与社区,并逐渐为许多开源项目或组织的重要贡献者。同年,华为成立了开源软件能力中心,负责公司整体开源能力建设,开源软件的应用于开发正式融入华为的研发流程。2013 年,华为成立开源战略委员会,将开源上升至企业战略层面,积极参与到开源生态构建的大潮中。

### 2、参与的开源组织

开源组织	参与层级
Linux 基金会	白金会员
Apache 基金会	黄金会员
OpenStack 基金会	白金会员
OPNFV 基金会	白金会员
CNCF 基金会	白金会员
Cloud Foundry 基金会	黄金会员
Linaro	Core Member
ONF	合作伙伴
OCI	创始会员
CII	会员
Open Daylight 基金会	白银会员

### 3、开源贡献

华为在云计算领域参与多个重要项目，具体如下

类型	开源项目
平台	Kubernetes, OCI, Docker, Hadoop, Spark, Cloud Foundry, PostgreSQL
基础设施	OpenStack, Linux kernel, Linaro, OCP, CoreOS, Open Daylight, CII, ONOS, ONAP, MESOS, KVM

下面以华为在 OpenStack 社区中发挥的作用为例，来说明华为对开源社区的贡献。

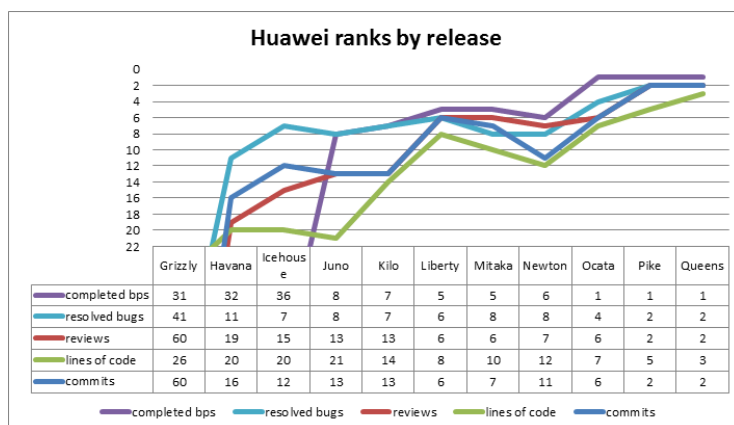
#### ➤ OpenStack 贡献

华为自 2012 年加入 OpenStack，一直是 OpenStack 的忠实拥护者，整个生态系统始于 OpenStack Grizzly 版本，该版本于 2013 年 6 月发布。华为云操作系统（FusionSphere 5.0 及更高版本）基于 OpenStack 架构，在 2015 年 5 月通过了 OpenStack Powered 认证并获得 OpenStack Powered 商标使用权。2016 年 10 月，华为私有云和华为搭建的德电公有云均通过了 Interop Challenge 的负载测试，标志着二者都具备了 OpenStack 跨云的互操作性，华为是唯一一家通过该测试的中国厂商。

2017 年华为成为 OpenStack 基金会的白金成员，体现了华为在 OpenStack 领域的影响力持续提升。截止到 2018 年 3 月，华为在 OpenStack 社区中有 7 个项目 PTL (Project Team lead)、2 个 TC (Technical Committee) 成员、26 个核心开发者 (Core Member)。由华为创建的社区正式项目包括 Tricircle、Service Chain、Kuryr、Dragonflow、aodh、Karbor、Cyborg 等。

华为对 OpenStack Juno 及其后续版本的代码贡献持续提升，在 2018 年 2 月 OpenStack 社区发布的 Queens 版本中，华为在核心项目中贡献：Completed Blueprints 全球第 1，Resolved Bugs 全球第 2，Reviews 全球第 2，Lines of Code 全球第 3，Commits 全球第 2，尤其是社区最重要的 Completed Blueprint 指标，华为贡献居全球第 1，再创新高；综合贡献在国内持续保持第 1 位。（数据来源：<http://stackalytics.com/> 2018/03/10）





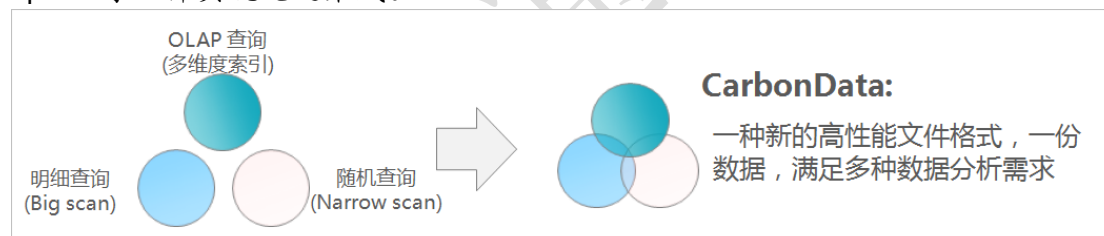
#### 4、自主开源项目

在参与已有开源项目的同时，华为也主动将内部项目对外开源，通过开源与更多开发者、合作伙伴推动技术与产业的发展。

##### ➤ CabornData (Apache 2.0)

CarbonData 项目，该项目于 2016 年 6 月进入 Apache 孵化器，2017 年 4 月毕业，正式成为 Apache 顶级项目 (TLP)。项目地址：  
<https://github.com/apache/incubator-carbondata>

CarbonData 是一种大数据高效存储格式解决方案。针对当前大数据领域分析场景需求各异而导致的存储冗余问题，CarbonData 提供了一种新的融合数据存储方案，以一份数据同时支持多种大数据应用场景，并与 Apache 社区的 Hadoop、Spark 等组件实现无缝集成。



基于核心技术的掌握，华为 FusionInsight 大数据平台成为了众多客户的优选合作伙伴。截止目前，华为 FusionInsight 大数据平台已经服务于全球 40 多个国家和地区的 700 多家客户，广泛应用于金融、政府、公共安全、运营商、教育、能源、交通、医疗等多个行业。

##### ➤ 微服务框架 ServiceComb (Apache 2.0)

ServiceComb 于 2017 年 6 月正式开源，并在 11 月份以全票通过的方式进入 Apache 软件基金会孵化器。项目地址：  
<https://servicecomb.incubator.apache.org/>

ServiceComb 是一个微服务框架，它能够帮助企业快速构建云原生应用，通过一系列解决方案帮助用户快速开发微服务应用的同时实现对这些微服务应用的高效运维管理。

ServiceComb 本身有强大的技术背景，它源于华为云 CSE (Cloud Service Engine) 平台。CSE 在华为公司内部商用了 3 年多时间，沉淀了电信行业高性能、低延时能力，也历经华为 VMALL 商城的历练，积累了华为公司内部微服务化架构演进的经验。同时，有越来越多的 ISV、传统企业与 CloudNative 企业使用 CSE。

---

相较于其他微服务框架，ServiceComb 目标是成为一个更加全面、横跨多种功能与语言的框架，让用户可以有更多选择的空间，ServiceComb 提供了侵入式的底座框架，也结合多语言以及历史遗留系统改造的需要提供非侵入的 Service Mesh 的实现，确保用户可以根据自己的实际情况对框架进行定制。

➤ LiteOS (BSD-3-Clause)

LiteOS 代码于 2017 年 1 月正式开源，项目地址：

<https://github.com/LITEOS>

LiteOS 提供了一个轻量级完整的软件栈，完美匹配物联网资源受限设备节点开发的需要；其低功耗、高实时处理等竞争力特性，能很好满足典型的物联网应用场景，可广泛应用于智能家居、穿戴式、车联网、制造业等领域。

目前 LiteOS 在智能仪表、智能停车、智慧路灯等方面已有实际应用。

### 三、阿里云计算有限公司

#### 1、开源历程

阿里云 ([www.aliyun.com](http://www.aliyun.com)) 创立于 2009 年，是全球领先的云计算及人工智能科技公司，为 200 多个国家和地区的企业、开发者和政府机构提供服务。阿里云特别重视对云计算、大数据技术基础的研发，飞天云操作系统核心技术完全自主可控，总体技术达到国际领先水平，是我国信息基础核心技术领域的重大科技成果，于 2017 年获得电子学会科技进步特等奖。基于飞天操作系统的云服务首批通过中央网信办云服务网络安全审查增强级，首个通过公安部云等保四级测评，已经成为亚太地区合规认证最完备的云计算平台。

#### 2、开源共享

阿里巴巴一直积极与国际开源组织合作。阿里巴巴是 Linux 的活跃开发者，此前已经为 Linux 内核提交了 290 多个 Patch，在国内互联网公司当中贡献度领先。同时，阿里也是开源项目 Xen 的顾问委员会成员，Xen 是 Linux 基金会合作项目。

阿里巴巴目前是 Linux 基金会的金牌会员，CNCF 的金牌成员，除了 Linux 和 Cloud Native，阿里在 MySQL、JVM、Web 服务器等知名项目中均有杰出贡献，阿里是全球对 MySQL 数据库贡献最多的公司之一，共提交了近 300 个 Bug 汇报和功能补丁，阿里云工程师彭立勋也成为数据库领域知名开源社区 MariaDB 基金会首位中国成员，同时阿里云也成为 MariaDB 的白金会员，并将所有自身产品的补丁贡献给 MariaDB。阿里巴巴也是 Apache 基金会、FSF 基金会、Open Computer 项目和开源组织 WebScaleSQL 的成员之一，并赞助 NTF 基金会、ISC 社区。

在开源社区融合方面，继 AWS 和 Azure 之后，阿里云也实现了对 Hadoop 对象存储的支持，成为第三家获得 Hadoop 官方支持对象存储能力的公共云公司。此外，阿里云还携手容器技术领域 6 家顶级公司成立 Containerd. IO 社区。

#### 3、自主开源项目

在开源社区贡献度方面，阿里巴巴一直都在积极地参与和贡献开源社区。仅 2016 年，阿里巴巴就开源了 Ant Design、Hilo、Weex、Freeline、Macaca、AliSQL、EGG 等软件，其中，RocketMQ、Weex 进入 Apache 基金会开始孵化。由开源中国公布的“2016 年度最受欢迎中国开源软件评选 TOP20”榜单中，阿里巴巴占 4 席，分别是：数据库连接池 Druid、JSON 解析库 fastjson、分布式服务框架 Dubbo、

---

分布式消息中间件 Apache RocketMQ。其中 Weex、Ant-design、Dubbo、Fastjson 在 GitHub 上 Star 已经破万，Alibaba 在 GitHub 上 Star 数超过 170,000，组织排名前十。

### ➤ Pouch

Pouch 是阿里巴巴集团发起的容器开源项目，旨在推进容器技术的发展进程。Pouch 的目标是加快云生态的发展，共建容器技术标准（OCI），从而使容器技术可以在云时代成为应用开发的基石性技术。“快速实现存量应用容器化”是 Pouch 的最大特性。Pouch 开源项目地址为 <https://github.com/alibaba/pouch>。

Pouch 基于 Linux 内核容器特性，并在此基础上做了大量的优化和改进，整合了阿里巴巴内部超大规模容器场景下积累的丰富经验，形成了拥有自身差异化特色的容器引擎解决方案。Pouch 容器技术在阿里巴巴已经承载大量的业务负载，在 2017 年的双 11 实现在线容器数量 100 万+，已应用到阿里巴巴 100% 的在线电商应用中，覆盖阿里巴巴大部分业务部门，服务的场景包含搜索、支付、广告、地图、物流、阿里云专有云等业务。Pouch 保障容器环境与原有运维体系完全兼容，保障用户聚焦挖掘自身价值，并屏蔽底层基础设置的运维管理，为电商企业向容器迁移节省了巨大成本。

**Pouch 开源方法论。**一是积极与行业分享容器经验。Pouch 自宣布开源开始，即以技术普惠作为最重要的目标之一。Pouch 在生产环境中积累了以下 4 个重要特性：富容器、隔离性强、多容器运行时支持、P2P 镜像快速分发等。Pouch 突破性地提出“富容器”的概念，帮助企业存量应用迅速容器化。富容器另辟蹊径，在容器的启动过程中将运维需求的应用软件部署在容器中，保障容器环境与原有运维体系完全兼容，可以在百万量级的超大规模容器应用中实现敏捷、低成本部署。隔离性方面，Pouch 是全球首个支持内核 quota 以及 lxcfs 实现文件系统限额与资源视角隔离的开源容器项目。资源视角隔离方面，Pouch 在全球做到了从 0 到 1 的突破，dirquota 增强了容器的隔离性能，极大扩展了容器技术的适用场景。Pouch 容器引擎是全球第一个同时支持多个容器运行时类型的技术。Pouch 支持管理容器以及 runV 容器，创新性实现了虚拟机与容器的混部能力，提高数据中心的资源利用率，为云场景提供了“多租户”的技术支撑。Pouch 通过使用 P2P 技术同时结合智能压缩、智能流控等多种创新技术，解决大规模文件下载以及跨网络隔离等场景下各种文件分发难题，大幅提高数据预热、大规模容器镜像分发等业务能力。Pouch 可使得容器镜像分发速度比 native 方式提高数十倍，registry 网络出口流量降低 2 个数量级。Pouch 多次在国内外知名技术会议上发起技术专题分享和主题演讲，向全球的容器生态参与者分享阿里巴巴的实践经验。二是积极建立社区。Pouch 团队非常注重开源社区的建设，并且迅速建立起了拥有接近 50 位开源者，近百家行业用户的社区。同时，Pouch 团队在代码开发以及产品打磨的过程中，解决了诸多上游社区容器标注 OCI、runC 和 containerd 等组件的问题，并通过开源的方式将代码反馈到上游项目中，为行业标准作出应有的贡献。

**Pouch 开源的行业意义。**Pouch 作为国产容器第一选择，在服务于阿里巴巴内部众多业务时，即体现了 Pouch 的价值普适性，抱着技术普惠的态度，Pouch 开源给行业的数字化转型带来直接价值。Pouch 凭借差异化的优势，帮助企业迅速实现大存量业务容器化，做好数字化转型的第一步。在高效、安全、快速容器化方面，Pouch 不仅是国内，同时也是全球的容器方案首选。

## ➤ AliSQL

AliSQL 起源于阿里巴巴集团，是基于 MySQL 的一个分支，在社区的基础上，做了大量的创新型的功能和性能改进，支撑了大量的阿里巴巴集团业务和公有云客户，被验证是一个性能高效，运行稳定的版本。AliSQL 于 2016 年 10 月宣布开源，回馈社区，到目前为止共进行了 9 次发布，保持着持续的活跃，并获得了大量用户的关注和使用。

AliSQL 作为 MySQL 的分支，在社区版本基础上新增许多功能：Sequence 引擎，为数据库用户快捷方便的获取单调递增的唯一值提供了引擎支持，并简化了语法支持；TokuDB 引擎集成，AliSQL 集成了 TokuDB 引擎的支持，提升大数据量写入和压缩功能；并发拥塞控制，使用 Thread running, SQL filter, 串行化记录更新等大量并发拥塞控制机制，保证数据库大并发情况下的性能持续稳定；动态加字段，在 InnoDB 引擎新增 comfort 记录格式，实现毫秒级加字段的功能，避免过度 copy 数据导致的变更风险；大量性能监控信息等，新增 Table, Index 使用统计，内存使用统计等各类业务和性能统计信息，提升 AliSQL 的诊断能力。

通过对 MySQL 的改进，第一，新增 Sequence, TokuDB 等引擎的支持，使 AliSQL 适应多场景需求：Sequence 引擎解决客户 ID 生成的难题，TokuDB 引擎在数据压缩方面多达 5 倍的压缩能力，解决物联网等用户的大数据存储问题。第二，大幅提升性能，提高业务处理吞吐能力：通用场景下，AliSQL 相比官方 MySQL 能提升多达 1 倍的性能提升，并在减库存等特定场景下，AliSQL 相比官方 MySQL 有多达上百倍的性能提升。第三，新增更多监控信息，方便用户性能诊断：比如 Table/Index 的使用统计、线程 Memory 的使用统计、CPU time 的统计等。

目前，完全自主设计的 Sequence 引擎，多源复制，线程 Memory 监控，已经被官方和 MariaDB 社区接纳。其中，创新设计的 Sequence 引擎，解决了 MySQL 社区和官方长期以来没有 ID 生成器的问题，并兼任了 Oracle, PG, SQL Server 等常用数据库的语法；多源复制扩充了数据在节点间的多向流动，解决公有云等场景下的弹性扩缩容难题；线程 Memory 监控，提供了运维更量化的指标来诊断内存使用。通过对 AliSQL 进行的创新性能优化，使 AliSQL 在保持稳定的前提下，大幅提升了性能，多合一倍的通用性能提升，大幅度的降低了用户的使用成本，也促进了 MySQL 社区的技术进步。

## 四、中兴通讯股份有限公司

### 1、开源项目总览

中兴通讯积极的参与主流的国际开源社区的工作，包括积极参与社区代码贡献，积极参与国际社区交流，以及将开源社区通过产品的整合实现商用落地。这些开源社区涵盖了操作系统、虚拟化、云计算、存储、SDN、NFV、大数据、人工智能等诸多重要且热门的技术领域。

目前中兴通讯主要参与的社区如下：

开源组织	参与层级
Linux 基金会	银牌会员
LF Networking Fund (LFN)	白金会员
Cloud Native Computing Foundation	金牌会员
OpenStack Foundation	金牌会员
DPDK	金牌会员

- **OpenStack**。中兴通讯目前是 OpenStack 黄金会员。在近三个版本中贡献 (Commit) 排名均在社区前 15 名内，且贡献的新功能数 (Completed BP) 保持在前 5。目前拥有 10 名 Core Contributor 和 1 名 PTL。是国内乃至全球业界领先的 OpenStack 开源贡献厂商。中兴的云管理平台 TECS 是一个基于 OpenStack 的产品，目前已经跟随到社区的 Pike 版本，且通过了社区兼容性认证，获得了 OpenStack Powered 授权。TECS 已经在包括 VEON 等全球领先运营商在内的多国 NFVI 项目中落地，在国内的三大电信运营商的数据中心以及各地政企项目中也均有落地。
- **OPNFV**。中兴通讯是 OPNFV 的白金会员。在近三个版本内贡献排名保持前五，而且连续两个版本保持前二。目前拥有董事会和 TSC 席位，并拥有 4 名 PTL。中兴通讯 TECS 在 2018 年首批通过了 OPNFV 的认证，获得了认证标志授权。
- **ONAP**。中兴通讯是 ONAP 的白金会员，也是创始成员之一。ONAP 是 Linux 基金会旗下两大开源 NFV MANO 工作组 Open-O 和 ECOMP 于 2017 年合并形成的一个组织。中兴原先是 Open-O 的主要贡献者，因此在 ONAP 中也占据了主要代码贡献的地位。中兴目前在 ONAP 中牵头两个项目，分别是 ZENAP MSB 微服务框架和 Holmes 告警分析。
- **Ceph**。中兴通讯是 Ceph 目前的主要贡献者之一。连续三个版本贡献排名第二，仅次于拥有 Ceph 创始人和核心团队的 Redhat。中兴拥有 TSC 成员一人和 Core Contributor 一人。
- **OpenDaylight**。中兴通讯是 OpenDaylight 的白金会员。目前中兴在 OpenDaylight 最新版本的核心模块贡献上升到第二，并且在 OpenDaylight 中有三个立项：OF Config, BIER 和 P4。
- **CNCF (Kubernetes)**。中兴通讯是 CNCF 的黄金会员，主要关注和贡献 Kubernetes。中兴目前在 Kubernetes 的总贡献上排名第五，是目前的主要贡献者之一。
- **Docker (Moby)**。中兴通讯非常重视 Docker 开源社区，并积极投入。目前在社区中的历史总贡献上排名第七。是社区的主要贡献者之一。

## 2、开源历程

中兴通讯从 2014 年开始参与开源社区贡献工作，主要是在 OpenStack 社区，从修改 Bug 做起，逐渐认识和了解开源社区的特点和工作机制。2015 年中，中兴扩大了参与开源社区的范围，增加参与了 OPNFV 和 OpenDaylight 社区。并且首次在 OPNFV 社区中占据了 TSC 席位，并成功在 OpenDaylight 社区建立了新的开源子项目。从 2016 年开始，中兴全面的投入到开源社区工作上来，投入了大量的人力和物力，有 100+ 的开发人员参与了社区贡献工作。中兴成立了多个面向不同社区的由专职人员和兼职人员共同组成的开源团队。开源工作不再是像前两年一样的游击作战，而是成建制的部队作战。2017 年，中兴在开源社区工作中全面开花，参与开源社区达到十几个，并在各开源社区中占据了多个 TSC、PTL 和 Core 的席位。中兴将产品的发展与开源社区的演进紧密结合，在功能贡献上形成双向流动，积极推动并实现了开源软件在国内外的商用落地。

## 3、开源项目管理

中兴通讯在内部成立了开源专家委员会，统筹各开源社区的工作。下设各开源社区的虚拟团队。团队中包含专职和兼职社区工作人员。同时在产品研发中，中兴全面引入开源社区的机制和工具链，做到内部研发和外部开源工作方式一致化。从而利于公司内部产品研发的开发者优先在开源社区中工作。中兴通讯还积

---

极的鼓励公司内开发者在开源社区活动中进行技术交流,积极的推动公司员工在各开源峰会上进行演讲。并通过各种手段予以激励。

#### 4、开源贡献

Daisy 是来源于中兴通讯的面自动化安装工具项目。Daisy 安装器内部采用 OpenStack 的 Kolla 项目作为部署引擎,支持构建和部署基于社区最佳实践的容器化的 OpenStack、OpenDaylight 以及基于 DPDK 的相关媒体面加速组件。从 E 版本 Daisy 第一次发布开始,用户可以通过 Daisy 提供的脚本方便地构建新版本 OpenStack 容器镜像并对正在运行的 OpenStack 进行升级或者回滚。除此之外,Daisy 还通过组播等手段,强化控制平面的镜像分发速率,优化大规模部署效率。

除了可以通过 GUI 部署 OpenStack, Daisy 更推荐通过脚本和配置文件的方式来部署 OPNFV 的各种标准场景,结合 Jenkins 搭建 DevOps 部署任务可以极大地实现场景配置的代码化和场景部署的自动化。目前支持的场景有纯 OpenStack 和 OpenStack+ODL,后续版本计划支持 OpenStack+OVS\_DPDK (或 FD.io)、Kubernetes 以及 ARM 平台等更多场景。

Daisy 目前在 OpenStack 中建立了 daisycloud-core 项目,托管了核心代码,并且在 OPNFV 中以 daisycloud-core 作为上游项目建立了面向 NFV 场景的 Daisy4NFV 安装器项目。目前 Daisy4NFV 目前已经成为 OPNFV 官方的安装器之一。

### 五、Zstack (上海云轴信息科技有限公司)

#### 1、开源概述

ZStack 是上海云轴信息科技有限公司于 2015 年 4 月发起并维护的新一代开源云计算 IaaS (基础架构即服务) 平台。它主要面向的是未来的智能数据中心,通过提供完善的 API 来管理包括计算、存储和网络在内的数据中心的各种资源。项目主页为 <https://github.com/zstackio/>。

#### 2、开源历程

ZStack 于 2015 年 4 月首发 0.6 版本后,迅速获得技术圈关注。国内 CSDN、InfoQ、51CTO、IT168 及国外 The New Stack 等都对 ZStack 的架构和技术进行了报道。阿里云(2015 年 5 月 13 日)、百度开放云(2015 年 9 月 5 日)、金山云(2015 年 10 月 8 日)和网易研究院(2016 年 1 月 26 日)分别邀请 ZStack 创始团队进行技术交流,为他们的工程师团队讲解 ZStack 架构和技术。2016 年 3 月,ZStack 发布 1.0 版本。代表该开源云平台的稳定性已经可以用在生产环境。2017 年 1 月,ZStack 获得阿里云领投的 A 轮投资,成为阿里云投资的唯一 IaaS 开源项目。2017 年 6 月,ZStack 发布 2.0 版本(即 ZStack 混合云产品),操作界面全新改版。目前,ZStack 继续致力于产品化,在降低云计算使用门槛道路上继续开拓发展,最新版为 2.2.3。

#### 3、开源项目管理

ZStack 通过 github 完成开源项目的管理。开发者可以通过 github fork ZStack 的代码仓库,在自己的分支上修改代码后,提交到 ZStack 的代码仓库,由对应的维护者审核代码,没有问题后,merge 到 ZStack 的源码。目前,ZStack 的代码仓库分为几个部分:ZStack 核心开源引擎、Stack 组件和工具、Stack 自动化测试框架。ZStack 的开源平台为 github,通过 github,用户可以和 ZStack 开发者交流,在 issue 页面提出问题。在 github 的 wiki 页面,用户和开发者也可以通过公开的文档,了解 ZStack 的开发、测试等细节内容。

---

同时，针对国内的用户情况，ZStack 建立了 ZStack 中国开源社区的 QQ 群，参与到开源社区的人数接近 2500 人。

#### 4、社区治理

目前，ZStack 的社区是一个开放社区，所有开发者都可以提交代码到 ZStack 代码仓库，所有用户也可以提交问题到社区。ZStack 的 QQ 群社区是最为活跃的交流平台，ZStack 有专职人员在社区内回答用户的提问，帮助用户解决使用 ZStack 过程中遇到的问题。目前，有包括高校教师、互联网公司、存储厂商、在校学生等大量开发者向 ZStack 社区贡献了他们自己的代码，帮助 ZStack 在国内提高影响力，获得更多的关注。

在社区的治理方面，ZStack 要求每个社区成员在交流过程中本着“平等开放、合作共享”的交流原则，不允许有任何攻击性语言，不允许有任何广告行为，不允许有任何技术讨论之外的发言。一旦违反了社区“平等开放、合作共享”的交流原则，管理员会对社区成员进行警告，多次违反交流原则的社区成员，社区管理委员会会对其采取封号、踢出 ZStack 中国社区。

#### 5、自主开源项目

目前，ZStack 的代码仓库分为几个部分：

➤ ZStack 开源云平台核心引擎。该项目是 ZStack 开源云平台的核心引擎，包含了 ZStack 包括云主机管理、物理机管理、存储管理、网络管理等核心功能组件。用户通过编译该云平台核心引擎，可以自己打造一个包含基本功能的 IaaS 平台。该项目使用的开源许可证为《Apache License 2.0》，该项目的开源社区建立在 github 网站上，用户可以通过 github 提交代码、提交问题以及通过 wiki 和开发团队交流。另外，该项目提供一个 ZStack 中国社区 QQ 交流群，有专职社区工作人员维护社区秩序，帮助社区成员和解答用户问题。

目前，已经有大量用户基于 ZStack 实现了他们自己的云平台，包括滴滴、饿了么、同盾科技等，也有一些国外公司也在使用 ZStack 替代一些传统的商业云平台。ZStack 目前是 IaaS 产业用户常见的一个选择。

➤ ZStack 组件和工具。该项目包含了 ZStack 的自动化安装、部署、升级的工具以及 ZStack 命令行工具及代理服务。该项目使用的开源许可证为《Apache License 2.0》该项目的开源社区建立在 github 网站上，用户可以通过 github 提交代码、提交问题以及通过 wiki 和开发团队交流。

➤ ZStack 自动化测试框架。该项目实现了 ZStack 的自动化测试框架，保证了 ZStack 项目每次发布的质量。该项目使用的开源许可证为《Apache License 2.0》，的开源社区建立在 github 网站上，用户可以通过 github 提交代码、提交问题以及通过 wiki 和开发团队交流。

#### 6、参与的开源项目

ZStack 项目的发起要感谢（不局限于）以下开源项目：Linux 内核、Ansible、Libvirt。

### 六. EasyStack (北京易捷思达科技发展有限公司)

中国开源云计算的领导者 EasyStack (北京易捷思达科技发展有限公司) 成立于 2014 年 2 月，由中国最早从事 OpenStack 研发的 IBM 中国研发中心



的核心团队创建，基于 OpenStack、Ceph、Kubernetes、Docker 等一系列开源软件为企业级客户提供开放、稳定可靠、高性能的云计算产品与服务。

EasyStack 坚持开源云计算理念，持续参与社区核心功能开发。在最新发布的 OpenStack Ocata 版本中，EasyStack 核心代码贡献排名全球第八。2016 年，EasyStack 成为 OpenStack 基金会黄金会员，并拥有独立董事 1 名和核心项目组 PTL 2 名。2016 年 8 月，EasyStack 加入 Linux 基金会，并成为 CNCF 和 OCI 会员，在最新的 Kubernetes 代码贡献排名中全球第九，在最新的 Ceph 分布式存储代码贡献排名中全球第八。

EasyStack 自成立以来已为国家电网、人民银行、邮储银行、建设银行、中国银联、中国移动、中国电信、海通证券、国泰君安、兴业数金、江苏农信、TCL、海尔、顺丰、百世汇通、上海汽车、宇通客车、沃尔玛、可口可乐、清华大学、天河云等超过 200 家企业级客户提供云产品和服务，涵盖金融、电信、政府、能源、交通、制造、教育、医疗等各大行业，在 2016 年度中国 OpenStack 十大用户中独占五席，走在中国开源云计算实践的最前沿。2016 年，EasyStack 荣膺 Gartner Cool Vendor in China，成为上榜的唯一一家开源云计算企业。2017 年入选 Gartner OpenStack Competitive Landscape 全球八大厂商；2017 年入选 OpenStack 基金会全球用户调查十大 OpenStack 厂商。

## 七、红帽软件（北京）有限公司

### 1、开源概述

红帽是世界领先的开源解决方案供应商，也是世界标准普尔 500 指数公司之一。以社区推动的方法提供可靠和高性能的云计算、Linux、中间件、存储和虚拟化技术。红帽还提供屡获殊荣的技术支持，培训和咨询服务。作为全球网络中的企业、合作伙伴和开源社区的组织核心，红帽促进创建相关的技术创新，为客户解放资源以获得增长并为 IT 的未来做好准备。自从 1993 年红帽公司成立并推出首个 Linux 版本，红帽一直在拓展其技术系列。



2014 至今，红帽收购了多家公司和多项技术，增强了公司的整体产品系列：  
Inktank—提供基于 Ceph 项目的云存储技术。

eNovance—不仅是 OpenStack 开发社区的重要贡献者，而且提供云 /OpenStack 咨询/部署服务。

CentOS—（红帽企业 Linux 克隆版本的经销商）整合到红帽中，为 CentOS 机构提供资金和资源，帮助这些机构生存下来，并开发基于红帽企业 Linux 的新解决方案。

FeedHenry—针对企业并围绕开源组件而设计的下一代移动应用平台。



---

**Ansible**—致力于提供强大的 IT 自动化解决方案,旨在帮助企业向无故障 IT 演进。客户可以在私有云和公有云中部署和管理应用,通过 DevOps 举措加快服务交付,理顺 OpenStack 安装和升级流程,并且通过简化编排和配置而加快容器的采用。

**3scale**—应用编程接口 (API) 管理技术的领先提供商。通过将 3scale 加入现有的技术组合中,红帽借助简化的云集成和基于微服务的架构而增强了 API 经济的实现能力。

**Codenvy**—云原生开发工具提供商,可帮助开发人员更轻松创建基于容器的现代云原生应用。红帽计划将 Codenvy 融入托管开发环境 OpenShift.io 中,以便在 OpenShift 上构建混合云服务。

**CoreOS**—容器原生解决方案创新者与领导厂商,CoreOS 的功能结合红帽已经就绪的 Kubernetes 和基于容器的组合,包括红帽 OpenShift,红帽将持续促进行业领先混合云平台的采用与开发。

2012 年,红帽成为首家营收超过 10 亿美元的开源厂商,2016 年,红帽成为全球第一家收入超 20 亿美元的开源解决方案供应商。预计未来四到五年,红帽将迈向 50 亿美元俱乐部。

截至红帽 2018 财年第三季度(2017 年 9 月 1 日到 2017 年 11 月 30 日),红帽已经实现了连续 63 个季度持续增长。得益于企业用户对混合云计算技术(包括实现和管理混合云、私有云环境所需的核心技术、容器平台和解决方案)的强劲需求,红帽的订阅收入和总收入再次实现超过 20% 的年增长。同时,单一订单超过百万美金以上的交易量成为常态,同比增长 30%。

## 2、参与和整合的开源项目

**参与。**社区有大量开源项目,红帽在许多项目的开发中扮演领导角色,例如 Linux 内核、KVM 虚拟化、OpenStack、Kubernetes 等。

**整合。**红帽发起并将开源项目整合为两个主要“整合”项目: Fedora (红帽企业 Linux 的基础) 和 JBoss 开发者社区 (红帽 JBoss Middleware 系列产品的的基础)。这些提供了免费的工作环境,作为软件开发实验室,并且为以后可能整合到红帽商用产品中的特定提供基础。红帽也是围绕云、存储和虚拟化的全球其他整合项目的领先发起者。公司还开展了许多更小规模的整合项目。开源的开发是具有高度专业化的活动。红帽企业 Linux 由几百个项目组成,而红帽是其中许多项目的主要开发贡献者,包括 Linux 内核、KVM、SELinux、GCC、WildFly、Gluster、Ceph、GFS 等项目。这种专业和非正式开发的组合提供了一个高度动态的环境。这种方式效率非常高,因此,目前大多数软件开发项目都采用开源模式。

## 八、烽火通信科技股份有限公司

### 1、开源项目总览

烽火是国际知名的信息通信网络产品与解决方案提供商,近年来积极探索并加速 ICT 转型,立足于光通信,业务深入拓展至信息技术与通信技术融合而生的广泛领域,如云计算、大数据、智慧城市、SDN/NFV 等,客户遍布国内、国际和信息化三大市场,为客户提供领先的端到端信息通信网络产品和解决方案。

---

烽火积极地参与国际社区 OpenStack 基金会、OpenNFV、CNCF、Ceph 等，国内社区 OSCAR、Ceph 中国社区等，不仅支持社区的各种技术、市场等交流活动，还积极地支持社区代码贡献。以 OpenStack 为例，国内公司几乎已经成为国内云计算私有云市场标准，因此国内公司参与 OpenStack 基金会积极性高涨。

## 2、开源项目历程

烽火从 2012 年开始进入开源云计算领域，先从内部国家重点实验室开始测试，于 2013 年底选定 OpenStack 作为烽火云平台的核心支撑技术，2015 年底发布 FitCloud 云网一体化战略，烽火成功地把开源软件应用于商业领域。2017 年 5 月基金会全票通过，烽火成功晋升为 OpenStack 基金会黄金会员。

## 3、开源贡献

在开源项目贡献方面，烽火重点推进 OpenStack、openNFV、Ceph 等。

以 OpenStack 版本 Pike 为例。Pike 已经是 OpenStack 第 16 个版本，最新版本的全球代码贡献排名也新鲜出炉。截至目前为止，烽火在社区已经培养出了 4 名项目 Core reviewer，Commit 数量达到 1048 个，位列国内公司第二名，完成 BP 个数为 11 个，位列国内公司第三名，解决 BUG 个数为 165 个，位列国内公司第三名，贡献代码行数 51531 行，位列国内公司第三名。在 2018 年 2 月份发布的最新 Queens 版本中，烽火贡献依然活跃，也取得不俗的成绩，烽火通信在 commits 贡献、Completed Blueprints 贡献、Reviews 贡献和 Resolved Bugs 贡献四个维度都位列国内前四。

烽火技术专家担任 Mogan 项目 PTL (Project Team Leader，社区项目团队负责人)。Mogan 项目是 Openstack 社区负责裸机资源管理的项目，我们熟知的 Ironic 项目就是 Mogan 项目的一个 Driver，Mogan 还可以接入其他的裸设备。

在 OPNFV 社区开源项目中，烽火技术专家担任 Compaas4nfv 项目 committer，成为国内少数几人之一。在 Ceph 社区开源项目中，烽火提交了诸多 patch，做出了一定的贡献。

通过 OpenStack 二次深度开发，把开源软件产品化商业化，成功发布烽火 FitOS 云操作系统。通过诸多商业项目，促使开源软件与行业应用结合，与大数据、容器、软件定义网络、软件定义存储等技术融合，基于以上烽火在开源技术领域和商业领域的积累，烽火在很多行业取得了诸多标杆性项目，比如政务行业有湖北楚天云、云南党建云等，在教育行业有湖北省教育云，在交通行业有中航信云，在制造领域有光电产业云等。