

开源协议漫谈

PuGong

Agenda

- ✦ 开源简介
- ✦ 常用开源协议说明
- ✦ 常见的开源软件及相关建议
- ✦ Q&A

开源简介

- ✦ 1980年代之前基本所有软件都是免费&开源
- ✦ 1980年代之后开始软件版权被广泛开始接受，商用软件开始流行
- ✦ 1983年 Free Software Movement启动；GNU Project成立，发布GNU Emacs, Debugger, C Compiler
- ✦ 1985年 GPLv1发布
- ✦ 1992年 Linux Kernel 0.12采用GPL协议
- ✦ 1993年 BSD协议发布，FreeBSD，NetBSD采用此协议
- ✦ 1998年 Netscape Navigator开源，发布Netscape Public License，后来更改为Mozzila 协议
- ✦ 1999年 Apache foundation成立，ASLv1发布，2000年更新为v1.1
- ✦ ...
- ✦ 2007年 Sun发布OpenJDK（GPL协议）
- ✦ 2005年 GIT
- ✦ 2006年 Hadoop
- ✦ 2008年 Android
- ✦ 2009 mongodb转为开源AGPLv3
- ✦ ...

常用开源协议

- ✦ MIT/BSD开源协议
- ✦ Apache开源协议
- ✦ GNU GPL开源协议组

常用术语说明

- 商用：是否可用做商业用途
- 分发：是否可以对软件进行二次分发
- 修改：是否可以对软件进行代码修改
- 专利许可：是否包含已被贡献者申请专利的部分的专利需求
- 私用：个人或者在组织内部使用
- 闭源：修改后的代码是否作为闭源使用
- 同协议：使用此软件的产品是否需要使用同协议
- 状态修改：状态的更改是否要做文档说明

MIT License & BSD License

- ✧ MIT和BSD都是比较宽泛的许可协议，作者只保留版权，允许使用方
 - ✧ 商用
 - ✧ 修改
 - ✧ 分发
 - ✧ 私用
- ✧ BSD相对MIT增加一个限制条款“No Endorsement”，不可以用开源代码的作者/机构名字和原来产品的名字做市场推广

Apache License version 2.0 (ASLv2)

- Apache协议在业界使用广泛，Apache基金会下面的开源软件都是ASLv2。协议允许使用方
 - 商用
 - 修改
 - 分发
 - 专利授权
 - 私用
 - 附加协议
- Apache协议限制了商标使用，即不可用开源软件的作者/机构的名字和产品名称做市场推广



GNU General Public License (GPL)

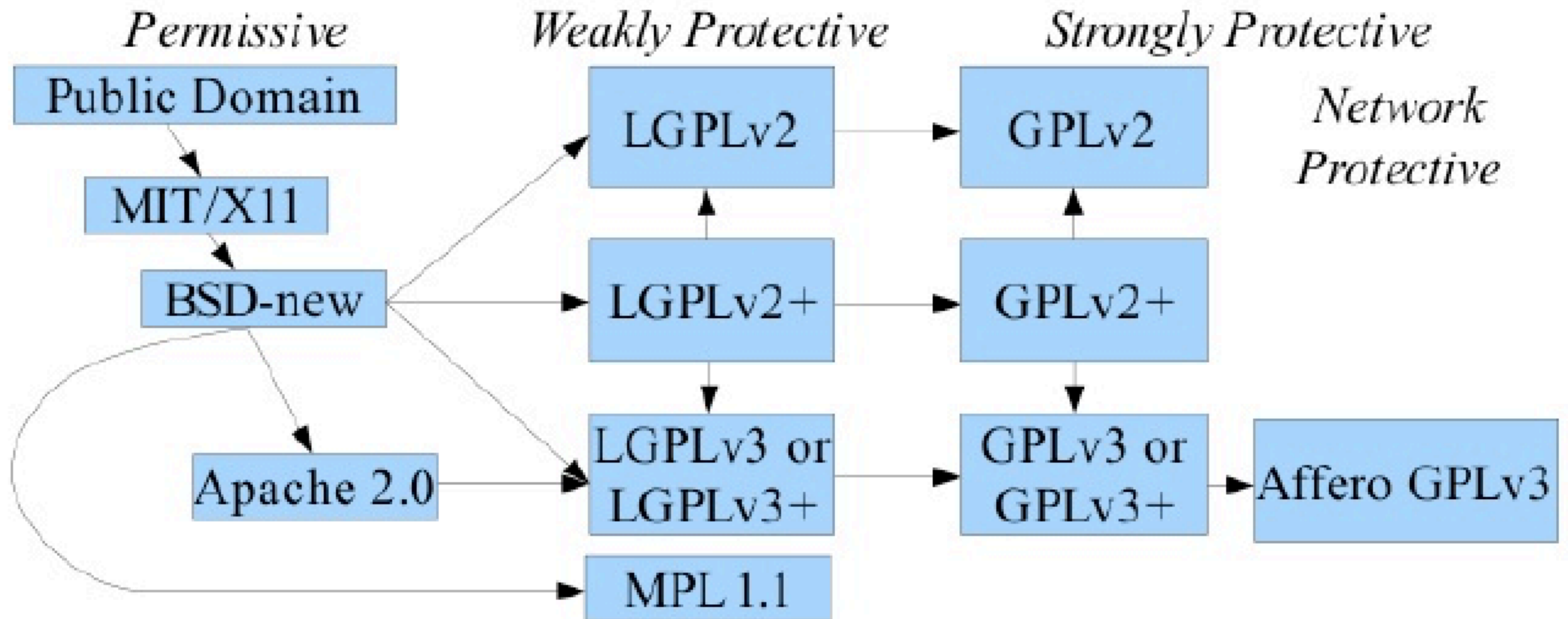
- 相对比较严格的开源协议，和Apache协议相比，GPL协议从v2开始包含著名的Liberty or Death条款 (Section 7)，即软件中使用了任何采用GPL协议的产品就必须也采用GPL协议，这就导致了GPL协议的传染性，因此对商业软件或者对代码有保密要求的软件不友好。
- 其他使用方面和Apache协议没有区别
- 由于GPLv2开始的传染性，扩展出LGPL (GNU Lesser General Public License): 允许商业软件以类库模式应用LGPL协议组件的时候不必开源



GNU General Public License (GPL)

- ✦ 由于GPLv2对违规处理比较严格，在一些组织的推动下推出了GPLv3，主要更改体现在：
 - ✦ v2违规后立刻取消许可证，自由在版权人同意后才能恢复
 - ✦ v3对企业相对友好，违规方可以申请临时许可证，也可以在停止侵权后的60天内版权人没有提出异议的话自动恢复
- ✦ 由于SaaS的日益广泛，在GPLv3的基础上扩展了AGPLv3 (GNU Affero General Public License)，增加了“Network use is distribution”，即通过网络提供服务也是一种分发方式，需要遵守AGPLv3协议开源

开源协议图示



常用开源软件

- ✦ 操作系统&虚拟化工具
- ✦ Web容器
- ✦ 消息队列
- ✦ 数据库
- ✦ 开发工具及框架
- ✦ 其他

常用开源软件

✧ 操作系统

- ✧ Linux Kernel: GPLv2
- ✧ FreeBSD: BSD
- ✧ X-Windows: MIT
- ✧ Android: ASLv2

✧ 虚拟化工具

- ✧ Kubernetes: ASLv2
- ✧ Openstack: ASLv2
- ✧ Docker: ASLv2
- ✧ Vagrant: MIT
- ✧ Virtualbox: GPLv2; 扩展包: Personal Use and Evaluation License (PUFL),不可商用

常用开源软件

✧ Web容器

- ✧ Nginx: BSD
- ✧ tomcat: ASLv2
- ✧ Wildfly(formly jboss): LGPL
- ✧ Jetty: ASLv2
- ✧ apache httpd: ASLv2

- ✧ NodeJS: MIT

✧ 消息队列

- ✧ Kafka: ASLv2
- ✧ RocketMQ: ASLv2
- ✧ RabbitMQ: Mozilla
- ✧ zeroMQ:LGPL

常用开源软件

✧ 数据库

- ✧ REDIS: BSD协议
- ✧ PostgreSQL: postgresql协议, 和 MIT/BSD类似
- ✧ MySQL: GPLv2
- ✧ MariaDB: GPLv2 (驱动: LGPLv2.1 or later)

- ✧ Hadoop: ASLv2
- ✧ Spark: ASLv2
- ✧ HBase: ASLv2
- ✧ Elastic Search: ASLv2
- ✧ MongoDB: AGPLv3和商业协议; 驱动 ASLv2
- ✧ neo4j: AGPLv3和商业协议

常用开源软件

- ✧ 开发工具及框架

- ✧ ATOM: MIT

- ✧ VSCode: MIT

- ✧ Eclipse: EPL

- ✧ JUnit: EPL

- ✧ JMeter: Apache v2

- ✧ Jenkins: MIT

- ✧ Spring家族: ASLv2

- ✧ Dubbo: ASLv2

- ✧ Ruby on Rails: MIT

- ✧ 大多数JS库 (VueJS, AngularJS, Backbone.js等) : MIT

- ✧ Django: BSD

- ✧ Ruby: BSD

常用开源软件

- ✧ 其他

- ✧ Firefox: Mozilla

- ✧ 7-zip: LGPLv2.1

- ✧ Bitcoin core: MIT

- ✧ PuTTY: MIT

- ✧ Gitlab: MIT

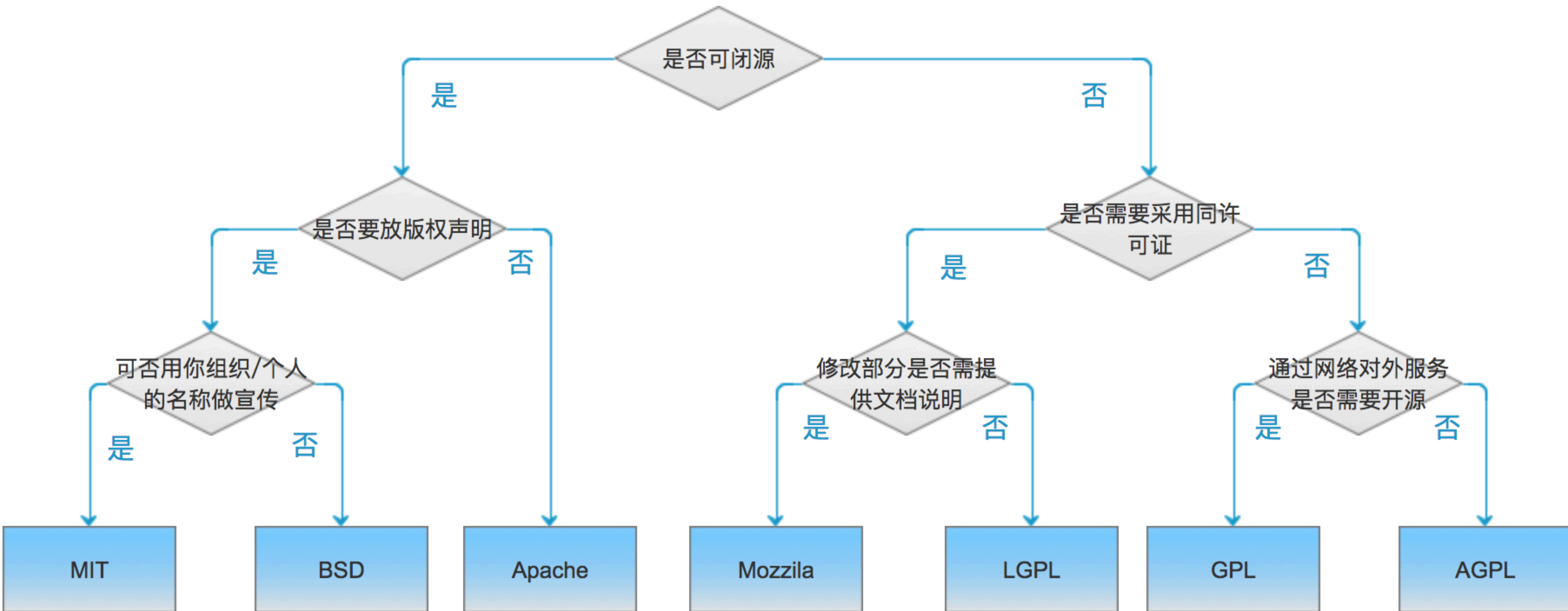
开源软件采用建议

- ✦ 优先采用MIT, BSD及ASLv2协议的软件
- ✦ 对于GPL及类似带传染性的协议的软件建议慎用
 - ✦ 内部使用没什么影响
 - ✦ 如果要用做商用, 对外提供服务的话尽量避免使用此类协议的开源软件, 如果没有无传染性协议的替代品则尽量将使用开源的组件尽量小, 同时升级为独立产品

自研软件开源建议

- ✦ 成立开源责任小组（虚拟）
- ✦ 确定开源规范：
 - ✦ 采用的开源协议：二次开发采用原协议；自主开发 Apache协议；
 - ✦ 可开源软件的条件
- ✦ 确认可开源的产品
- ✦ 确定开源产品的负责人
- ✦ 确定版本发布流程：版本规划，脱敏等

自研软件开源建议 - 开源协议选择



Q&A

Thanks