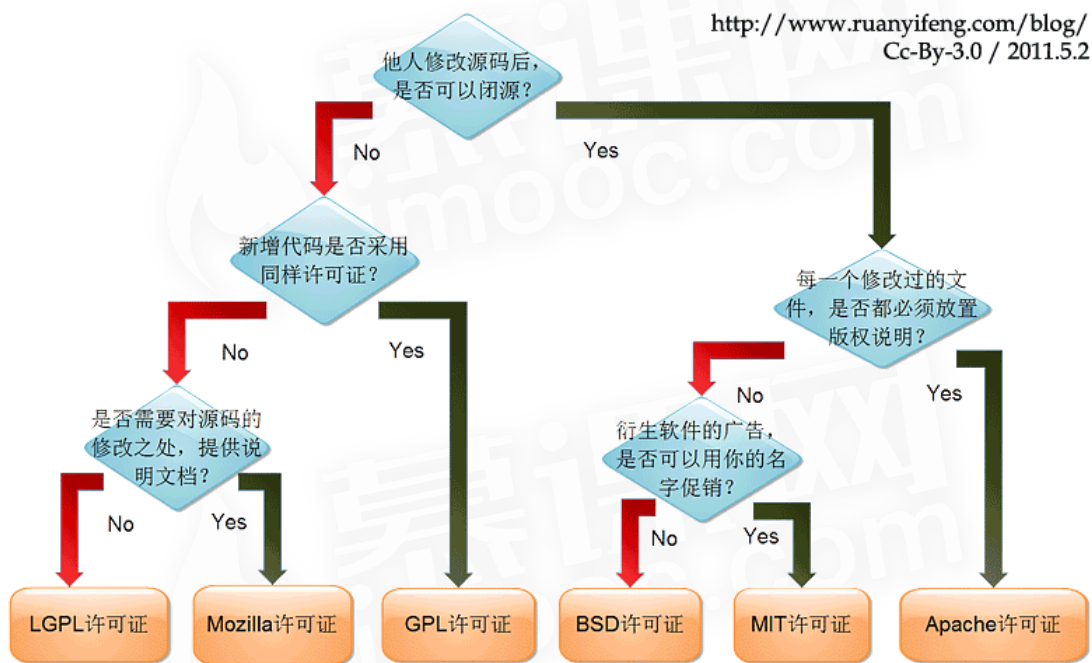


## 开源许可协议

开源许可证 GPL、BSD、MIT、Mozilla、Apache 和 LGPL 的区别。



以下是上述协议的简单介绍：

### 1 BSD 开源协议

**BSD 开源协议**是一个给予使用者很大自由的协议。基本上使用者可以“为所欲为”，可以自由的使用，修改源代码，也可以将修改后的代码作为开源或者专有软件再发布。

但“为所欲为”的前提当你发布使用了 BSD 协议的代码，或则以 BSD 协议代码为基础做二次开发自己的产品时，需要满足三个条件：

如果再发布的产品中包含源代码，则在源代码中必须带有原来代码中的 BSD 协议。 如果再发布的只是二进制类库/软件，则需要在类库/软件的文档和版权声

明中包含原来代码中的 BSD 协议。 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。

BSD 代码鼓励代码共享，但需要尊重代码作者的著作权。BSD 由于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发布和销售，因此是对商业集成很友好的协议。而很多的公司企业在选用开源产品的时候都首选 BSD 协议，因为可以完全控制这些第三方的代码，在必要的时候可以修改或者二次开发。

## 2 Apache Licence 2.0

Apache Licence 是著名的非盈利开源组织 Apache 采用的协议。该协议和 BSD 类似，同样鼓励代码共享和尊重原作者的著作权，同样允许代码修改，再发布（作为开源或商业软件）。需要满足的条件也和 BSD 类似：

**需要给代码的用户一份 Apache Licence 如果你修改了代码，需要再被修改的文件中说明。 在延伸的代码中（修改和有源代码衍生的代码中）需要带有原来代码中的协议，商标，专利声明和其他原来作者规定需要包含的说明。 如果再发布的产品中包含一个 Notice 文件，则在 Notice 文件中需要带有 Apache Licence。你可以在 Notice 中增加自己的许可，但不可以表现为对 Apache Licence 构成更改。**

Apache Licence 也是对商业应用友好的许可。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/销售。

## 3 GPL

我们很熟悉的 Linux 就是采用了 **GPL**。GPL 协议和 BSD, Apache Licence 等鼓励代码重用的许可很不一样。GPL 的出发点是代码的开源/免费使用和引用/修改/衍生代码的开源/免费使用，**但不允许修改后和衍生的代码做为闭源的商业软件发布和销售**。这也就是为什么我们能免费的各种 linux，包括商业公司的 linux 和 linux 上各种各样的由个人，组织，以及商业软件公司开发的免费软件了。

GPL 协议的主要内容是只要在一个软件中使用(“使用”指类库引用,修改后的代码或者衍生代码)GPL 协议的产品,则该软件产品必须也采用 GPL 协议,既必须也是开源和免费。这就是所谓的“传染性”。GPL 协议的产品作为一个单独的产品使用没有任何问题,还可以享受免费的优势。

由于 GPL 严格要求使用了 GPL 类库的软件产品必须使用 GPL 协议,对于使用 GPL 协议的开源代码,商业软件或者对代码有保密要求的部门就不适合集成/采用作为类库和二次开发的基础。

其它细节如再发布的时候需要伴随 GPL 协议等和 BSD/Apache 等类似。

## 4 LGPL

LGPL 是 GPL 的一个为主要为类库使用设计的开源协议,和 GPL 要求任何使用/修改/衍生之 GPL 类库的软件必须采用 GPL 协议不同。**LGPL 允许商业软件通过类库引用(link)方式使用 LGPL 类库而不需要开源商业软件的代码,这使得采用 LGPL 协议的开源代码可以被商业软件作为类库引用并发布和销售。**

但是如果修改 LGPL 协议的代码或者衍生,则所有修改的代码,涉及修改部分的额外代码和衍生的代码都必须采用 LGPL 协议。因此 LGPL 协议的开源代码很适合作为第三方类库被商业软件引用,但不适合希望以 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采用。

**GPL/LGPL 都保障原作者的知识产权,避免有人利用开源代码复制并开发类似的产品**

## 5 MIT

**MIT 是和 BSD 一样宽范的许可协议,作者只想保留版权,而无任何其他了限制.也就是说,你必须在你的发行版里包含原许可协议的声明,无论你是以二进制发布的还是以源代码发布的.**

## 6 MPL

MPL 是 The Mozilla Public License 的简写，是 1998 年初 Netscape 的 Mozilla 小组为其开源软件项目设计的软件许可证。MPL 许可证出现的最重要原因就是，Netscape 公司认为 GPL 许可证没有很好地平衡开发者对源代码的需求和他们利用源代码获得的利益。同著名的 GPL 许可证和 BSD 许可证相比，MPL 在许多权利与义务的约定方面与它们相同（因为都是符合 OSI 认定的开源软件许可证）。但是，相比而言 MPL 还有以下几个显著的不同之处：

- ◆ MPL 虽然要求对于经 MPL 许可证发布的源代码的修改也要以 MPL 许可证的方式再许可出来，以保证其他人可以在 MPL 的条款下共享源代码。但是，在 MPL 许可证中对“发布”的定义是“以源代码方式发布的文件”，这就意味着 **MPL 允许一个企业在自己已有的源代码库上加一个接口，除了接口程序的源代码以 MPL 许可证的形式对外许可外，源代码库中的源代码就可以不用 MPL 许可证的方式强制对外许可。**这些，就为借鉴别人的源代码用做自己商业软件开发的行为留了一个豁口。
- ◆ MPL 许可证第三条第 7 款中允许被许可人将经过 MPL 许可证获得的源代码同自己其他类型的代码混合得到自己的软件程序。
- ◆ 对软件专利的态度，MPL 许可证不像 GPL 许可证那样明确表示反对软件专利，但是却明确要求源代码的提供者不能提供已经受专利保护的源代码（除非他本人是专利权人，并书面向公众免费许可这些源代码），也不能在将这些源代码以开放源代码许可证形式许可后再去申请与这些源代码有关的专利。
- ◆ 对源代码的定义 而在 MPL（1.1 版本）许可证中，对源代码的定义是：“源代码指的是对作品进行修改最优先择取的形式，它包括：所有模块的所有源程序，加上有关的接口的定义，加上控制可执行作品的安装和编译的‘原本’（原文为‘Script’），或者不是与初始源代码显著不同的源代码就是被源代码贡献者选择的从公共领域可以得到的程序代码。”
- ◆ MPL 许可证第 3 条有专门的一款是关于对源代码修改进行描述的规定，就是要求所有再发布者都得有一个专门的文件就对源代码程序修改的时间和修改的方式有描述。