

开源软件是源代码可以任意获取的计算机软件，任何人都能查看、修改和分发他们认为合适的代码

<https://www.synopsys.com/glossary/what-are-open-source-licenses.html>

* + 宽松自由软件许可协议 Permissive free software licence

Apache License v2、

MIT

BSD

是一种对软件的使用、修改、传播等方式采用最低限制的自由软件许可协议条款类型。这种类型的软件许可协议将不保证原作品的派生作品会继续保持与原作品完全相同的相关限制条件，从而为原作品的自由使用、修改和传播等提供更大的空间

* + 著佐权许可证 copyleft license

在有限空间内的自由使用、修改和传播，且不得违背原作品的限制条款

*GPL 是典型的强著佐权（copyleft ）许可证，*

*LGPL、MPL 是弱著佐权（copyleft ）许可证。*

*SSPL 则是近年来 MongoDB 创建的一个新许可证，存在较大争议，开放源代码促进会 OSI 甚至认为 SSPL 就不是开源许可协议。*

==术语

|  |  |  |
| --- | --- | --- |
| 名词 | 中文 | 备注 |
| Commercial use | 商业应用 | 指是否可将开源用于商业目的 |
| Distribution | 发布发行 | 指是否允许发布发行依赖于此开源的项目 |
| Modification | 修改 | 指是否允许在开源项目上做修改 |
| Patent use | 申请专利 | 指是否允许用于专利申请 |
| Private use | 个人使用 | 指是否允许个人学习，修改和使用 |
| License and copyright notice | 注明协议和版权 | 指是否要求注明协议以及版权说明 |
| State changes | 注明变更 | 指对于开源代码的修改是否要明确说明 |
| Liability | 义务 | 指开源软件是否承担使用其后带来的义务 |
| Trademark use | 商标使用 | 指开源软件是否可用在商标或在商标中暗示使用 |
| Warranty | 报修维护 | 指开源软件是否提供后期维护 |
| Same License | 同协议下 | 指基于开源二次开发的产品若发布，是否要基于相同或相近的协议之下 |
| Disclose source | 公开源码 | 指基于开源开发的代码是否也需要开源 |

**1、GPL**

GPL支持强大的版权保护，可能是最具保护性的免费软件许可证。其背后的核心理念就是任何衍生作品，也必须在GPL下发布。具有以下特点：

* + 版权约束很强
  + 项目工作适合商业用途。
  + 被许可方可以修改项目。
  + 被许可方必须将源代码与衍生作品一起发布。
  + 衍生作品必须以相同的条款发布。

GPL是自由软件基金会（

[自由软件基金会](https://www.zhihu.com/search?q=%E8%87%AA%E7%94%B1%E8%BD%AF%E4%BB%B6%E5%9F%BA%E9%87%91%E4%BC%9A&search_source=Entity&hybrid_search_source=Entity&hybrid_search_extra=%7B%22sourceType%22%3A%22answer%22%2C%22sourceId%22%3A507675584%7D)（FSF）

）项目的指定许可证，包括Linux系统核心的各种GNU工具

大型项目，尤其是商业项目往往将GPL与一个或多个其他许可证结合使用

MySQL（GPL+商业许可证）：

**2、LGPL**

GPL V2）协议，可以翻译为更宽松的GPL协议。与GPL协议的区别为，后者如果只是对LGPL软件的程序库的程序进行调用而不是包含其源代码时，相关的源程序无需开源

* + 版权约束较弱（受限于动态关联的程序库）
  + 项目作品适合商业用途。
  + 被许可方可以修改项目。
  + 被许可方必须将源代码与衍生工作一起开源发布。
  + 如果你修改了项目，则必须以相同的条款发布修改后的作品。
  + 如果你使用项目作品，无需以相同的条款发布衍生作品

[http://OpenOffice.org](https://link.zhihu.com/?target=http%3A//OpenOffice.org) 3 （LGPL v3）

*来自 <*[*https://www.zhihu.com/question/19568896*](https://www.zhihu.com/question/19568896)*>*

**3、EPL**

[Eclipse Public License（EPL](https://www.zhihu.com/search?q=Eclipse%20Public%20License%EF%BC%88EPL&search_source=Entity&hybrid_search_source=Entity&hybrid_search_extra=%7B%22sourceType%22%3A%22answer%22%2C%22sourceId%22%3A507675584%7D)）

* + 较弱的版权约束（受限于软件“模块”）
  + 项目作品适合商业用途。
  + 被许可方可以修改项目。
  + 如果你修改了作品，则必须以相同的条款发布修改后的作品。
  + 如果你使用了作品，无需以相同的条款发布衍生作品。
  + 软件的商业分销商必须在因商业用途导致的诉讼/损害中保护或赔偿原始EPL贡献者。

Junit 、jetty、编程语言Clojure

**4.Mozilla Public License (MPL)**

MPL旨在成为严格许可证（如GPL）和宽松许可证（如MIT许可证）之间的折中方案

“发证单位”是源文件。许可方不得限制MPL涵盖的任何文件的用户权限和访问权限。但是同一个项目也可以包含专有的[非MPL许可文件](https://www.zhihu.com/search?q=%E9%9D%9EMPL%E8%AE%B8%E5%8F%AF%E6%96%87%E4%BB%B6&search_source=Entity&hybrid_search_source=Entity&hybrid_search_extra=%7B%22sourceType%22%3A%22answer%22%2C%22sourceId%22%3A507675584%7D)。如果授予对MPL许可文件的访问权限，则可以在任何许可下发布生成的项目。

MPL具有以下特点：

* + 版权约束较弱（受限于单个文件）
  + 项目作品适合商业用途。
  + 被许可方可以修改项目。
  + 被许可方必须提供引用说明。
  + 被许可方可以根据不同条款重新发布衍生作品
  + 被许可方不得重新许可MPL许可的资源
  + 被许可方必须将其衍生作品与MPL许可的源代码一起分发。

办公套件LibreOffice

**5.Apache License 2.0 (ASL)**

ASL是Apache 软件基金会项目使用的唯一许可证。广泛认为ASL对商业友好，已在该组织之外得到大量应用。在ASL下发布企业级项目并不稀奇。

区别主要在于前者额外提供了一份简易的[专利许可授权](https://www.zhihu.com/search?q=%E4%B8%93%E5%88%A9%E8%AE%B8%E5%8F%AF%E6%8E%88%E6%9D%83&search_source=Entity&hybrid_search_source=Entity&hybrid_search_extra=%7B%22sourceType%22%3A%22answer%22%2C%22sourceId%22%3A75772529%7D)，明确禁止商标使用权以及要求明确指明所有修改过的文件（state changes）。应用案例有：Apache 家族、SVN

Apache许可证具有以下特点：

* + 非版权
  + 项目作品适合商业用途。
  + 被许可方可以修改项目。
  + 被许可方必须提供引用说明。
  + 被许可方可以根据不同条款重新分配衍生作品。
  + 被许可方不必将其衍生作品和源代码一起分发

Apache Spark（ASL v2.0）：

Spring Framework（ASL v2.0）：

**6.MIT License**

MIT许可证

这是一个非常受欢迎的许可证，甚至可能是最受欢迎的。它对重复使用的限制极少，可以轻松地与其他许可证相关联，包括GPL和专有许可证。

具有以下特点：

* + 非版权
  + 项目作品适合商业用途。
  + 被许可方可以修改项目。
  + 被许可方必须提供引用说明。
  + 被许可方可以根据不同条款重新发布衍生作品
  + 被许可方不必将其衍生作品和源代码一起发布。

Node.js：

**7.BSD许可证**

BSD许可证有三种版本：初版的4句版许可证，“修订版”3句版许可证和“简化版”2句版许可证。这三种版本都在使用理念上高度接近MIT许可证。事实上，2句版BSD许可证和MIT许可证之间的实际差异很小。

3句版和4句版BSD许可证增加了对名称重用和广告的更多要求。如果你想保护自己的产品或品牌名称，可以考虑使用这两版许可证。

BSD许可证具有以下特点：

非版权

* + 项目作品适合商业用途。
  + 被许可方可以修改工作。
  + 被许可方必须提供引用说明。
  + 被许可方可以根据不同条款重新发布衍生作品。
  + 被许可方不必将其衍生作品和源代码一起发布。
  + 被许可方不得使用原作者名称或商标来为衍生作品背书（3句版和4句版BSD）
  + 被许可方必须在提及此项目功能或用途的所有广告材料中致谢项目原作者（4句版BSD）

Django（3句版BSD）：

*来自 <*[*https://www.zhihu.com/question/19568896*](https://www.zhihu.com/question/19568896)*>*

Ruby（2句版BSD和自定义许可证）：

*来自 <*[*https://www.zhihu.com/question/19568896*](https://www.zhihu.com/question/19568896)*>*

Redis（3句版BSD）：

*来自 <*[*https://www.zhihu.com/question/19568896*](https://www.zhihu.com/question/19568896)*>*

* + **我能不使用任何开源协议吗？**

如果项目没有明确注明所适用的开源许可证，则应用项目作者司法管辖区的“默认”版权。换句话说，永远不要将“不用许可证”当成一种隐式授权，让他人随心所欲地使用你的项目。事实恰恰相反：即使没有明确的许可证，你，项目的作者，事实上并未放弃法律授予的任何权利。

但请记住，许可证既支配权利也支配义务。你有没有想过为什么这么多许可证文本中都有一份粗体大写的关于产品保证的免责声明，或者更常见的是没有保证？这是为了保护作品的所有者免受隐性担保或用户假设。你最不想看到的就是因为发布了开源项目而被起诉吧！

* + **我能使用自定义开源协议吗？**

你能，但**最好别这么干**。

作为一种合同，许可证不能（在大多数司法管辖区）凌驾于地方法律之上。因此，在全球各地难以强制执行许可权利。一旦牵扯到官司，在法官面前为“标准”开源许可证辩护会更容易（难度要低一些）。事实上，已经出现了类似的案件。显然，使用自定义许可证这官司很难打。

此外，自定义许可证可能会与其他许可证发生冲突，从而导致你的项目在法律上不利。

* + **我能使用多个开源协议吗？**

是的，可以，使用多个开源许可证并不罕见。特别是当你想根据开源项目开展商业业务时，最好是多考虑几个开源协议

作者：景略集智

链接：https://www.zhihu.com/question/19568896/answer/507675584

来源：知乎

著作权归作者所有。商业转载请联系作者获得授权，非商业转载请注明出处。

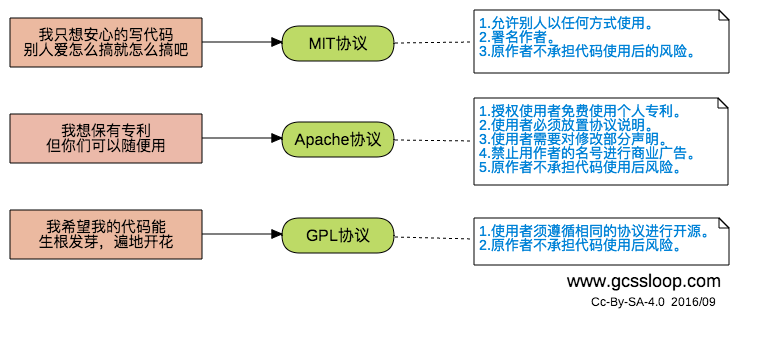
**为什么要添加开源协议?**

首先是对作者的保护，防止知识成果被恶意利用。

* + 开源协议中一般都包含免责声明(禁止代码的作者承担代码使用后的风险及产生的后果)，比如你开源了一个破解[智能锁](https://www.zhihu.com/search?q=%E6%99%BA%E8%83%BD%E9%94%81&search_source=Entity&hybrid_search_source=Entity&hybrid_search_extra=%7B%22sourceType%22%3A%22answer%22%2C%22sourceId%22%3A122973704%7D)的代码，如果有人利用这个去盗窃导致他人损失，你是无需承担责任的。

其次是对使用者的保护，方便使用者。

* + 使用者一看就知道自己允许进行哪些操作，不允许进行哪些操作。
  + 未添加协议的代码默认是作者保留所有权利的(*对此不同国家的法律可能稍微存在区别*)，这就像一颗[定时炸弹](https://www.zhihu.com/search?q=%E5%AE%9A%E6%97%B6%E7%82%B8%E5%BC%B9&search_source=Entity&hybrid_search_source=Entity&hybrid_search_extra=%7B%22sourceType%22%3A%22answer%22%2C%22sourceId%22%3A122973704%7D)，如果你在项目中使用了这一份没有协议的代码，原作者只要能证明你未经许可使用了他的代码，是能够起诉你的。



开源组织

1）FSF（Free Software Foundation，自由软件基金会)

“自由” 和 “开源” 的理解，FSF 建立了开源领域的第一个 “copyleft” 属性的许可证 - GPL (GNU Public License)

2）OSI（Open Source Initiative，开放源代码促进会）

OSI 组织批准过的许可大概有 80 种，包括 Apache License v2、GPL v2、MIT/BSD 等

是在原教旨主义开源 (最早的开源运动发起和推动者们) 与软件工业/商业之间激烈矛盾中，寻求更平衡的体系和治理机制

FSF 与 OSI 是推广和维护开源秩序的非盈利组织，维护着 “开源” 的定义以及主要的开源软件协议递交、讨论与审核。只要条款被审核通过是符合开放源代码定义的，就可以称之为开放源码授权条款，采用开放源码条款散布授权的软件即是开放源码软件，若一份商业产品中包含有开放源码软件，其包装上可以标上开放源码促进会的证明标章，

Open source license compliance 许可证合规

<https://www.synopsys.com/blogs/software-security/auditing-dependencies-software-due-diligence.html>

<https://www.synopsys.com/blogs/software-security/top-open-source-licenses.html>

SCA与软件供应链安全

SCA 软件组成分析

SBOM 软件物料清单