翻译来自[**https://www.mozilla.org/en-US/MPL/2.0/FAQ/**](https://www.mozilla.org/en-US/MPL/2.0/FAQ/)

**1. 什么是MPL-2.0?**

MPL-2.0是一个copyleft许可证。MPL-2.0的“文件级”copyleft旨在鼓励贡献者分享他们对代码所做的修改，同时仍然允许他们将代码与其他许可证（开放的或专有的）下的代码结合在一起，限制最小。

**2.为什么创建MPL?**

MPL填充了一个自由与开源软件许可证的有用的空间，位于Apache许可证（不需要修改共享）和GNU许可证（要求修改共享，范围比MPL更广）之间。

**3. 谁维护MPL?**

MPL是由Mozilla维护的，这是一个致力于在互联网上建立开放性、互操作性和个人授权的全球性非营利社区。

**4. 我想对我编写的软件使用MPL许可证。我需要做什么?**

要将MPL许可证应用于您所编写的软件，请将许可证附录A添加到项目的每个源代码文件中。这里有各种注释样式的示例标题。您还可以添加其他准确的版权所有权通知，例如版权所有者的名称，但这是不必要的。

**5. 我想使用在MPL下可用的软件。我必须做什么?**

没什么。像所有其他自由和开源软件一样，MPL下的软件可以供任何人（包括个人和公司）用于任何目的。MPL只在您想要在组织之外分发软件时要求履行相关义务。

**6. 我想在我的组织内分发在MPL下可用的软件，无论修改与否。我必须做什么?**

没什么。私有修改和分发（在公司或组织内部被视为“私有”）的权利是由自由和开源软件许可（包括MPL）保障的其中一项权利。

**7. 我想（在我的组织之外）分发由其他人在 MPL下构建的完整且未更改的可执行程序。我需要做什么?**

如果分发程序给你的人遵守了MPL，通常什么也不需要做。

检查看看分发程序给你的人是否遵守MPL，寻找告诉你可以在哪里找到源代码形式软件的通知（即检查符合3.2节(a)），然后检查源代码是否在那个地方，并包括一个通知，通知您源代码在MPL的条款下可用（即检查它是否符合3.1节）。

如果您只分发库，或者只分发您接收到的程序的某些部分，那么您需要根据第3.2(a)节的要求，采取额外步骤，确保告知程序的用户他们的权利。

如果您想在分发软件时添加您自己的条款，第3.2(b)条要求这些条款不得限制接收方在MPL下的权利，如果您对软件提供担保，第3.5条要求您明确说明它是你一个人提供的。

**8.我想（在我的组织之外）分发可执行程序或库，这些程序或库是我从别人在MPL许可的源代码编译而来的，未更改的，可能是独立的，也可以是较大作品的一部分。我需要做什么?**

你必须告知接收者在哪里可以获得你分发的可执行程序或库中的MPL许可的源代码(也就是说，你必须遵守章节3.2)。

您可以在您选择的许可下分发您创建的任何可执行文件，只要该许可不干扰在MPL条款下接收者对源代码的权利。

**9. 我想（在我的组织之外）分发我修改过的MPL许可源代码。我需要做什么?**

要查看完整的做法，请阅读许可证。然而，一般来说:。

1)，您必须通知接收者，源代码是根据MPL的条款(第3节)提供给他们的，包括你所创造的任何修改(定义见第1条)；

2）您必须按照本许可证第2节所述进行授权；

3）您必须尊重源代码中关于删除或更改通知的限制(第3.4节)。

**10.我想（在我的组织之外）分发一个基于我修改过的MPL许可源代码的可执行程序。我需要做什么?**

您必须提供Q9中所描述的MPL许可的源代码，并告知接收者他们如何获得这样的源代码（章节3.2）。

**11.MPL的“病毒式”程度如何?如果我在我的专有应用程序中使用MPL许可的代码，我必须放弃所有的源代码吗?**

不。本许可要求“修改”（定义见章节1.10）必须在MPL下获得许可，并向您分发源代码的任何人提供。然而，不包含MPL许可代码的新文件不是“修改”，因此不需要根据MPL的条款进行分发。

即使你创建了一个更大的作品（如1.7节中定义的），可以将非MPL许可文件与MPL许可文件一起使用、编译或分发，例如，允许使用MPL代码静态链接到更大的专有软件作为其中的一部分来分发。这在更强的copyleft许可条款下通常是不可能的。

**12.与LGPL和GPL的copyleft相比，MPL的copyleft的范围如何?**

一般来说，MPL、LGPL和GPL的范围可以概括为以下三个:

MPL: copyleft适用于任何包含MPL代码的文件；

LGPL: copyleft适用于任何基于LGPL代码的库；

GPL: copyleft适用于所有基于GPL代码的软件；

但是，我们建议阅读许可证以更好地理解它们的范围，特别是理解LGPL和CPL如何定义“基于”。

**13.我可以在同一个可执行程序中结合MPL许可代码和BSD许可代码吗?Apache呢？**

是的，两个都可以。Mozilla目前就使用BSD许可的代码，例如在Firefox中用来解码WebM视频的libvpx是在BSD许可下使用的。

**14.我可以在同一个可执行程序中组合MPL许可代码和(L) gpl许可代码吗?**

是的，通过在第3.3节的条款下创建一个“更大的作品”，必须满足三个要求:

1. 软件不能“与二级许可不兼容”。“软件可能与二次许可不兼容”的方式有两种:原始作者通过在附录B中添加文件头来标记它，或者原始作者在MPL-1.1下分发了软件，且没有使用(L) GPL双重或三重许可代码。

2. 较大的作品必须是“覆盖软件与一个或多个二级许可证作品的组合”。所以你不能只是说“我真的更喜欢(L) CPL，您必须有与另一个现有GPL工作组合的需求。(这与传统的双许可证不同，后者不需要您合并，而是允许您简单地说“我决定只使用GPL”。)

3您必须在(L) GPL下“额外分发”。换句话说，您必须让您的接收者在MPL和(L) GPL下都可以使用MPL许可的源代码。你的接收者的下游可以接受(L) GPL only或MPL only。这与传统的双许可不同，传统的双许可允许您始终选择其中一个许可证。

没有一个与GPL兼容的许可证能够完美地保留原始作者重用下游衍生产品的能力，但最后的这两个限制有助于在最广泛的情况下增加这种重用发生的可能性。我们已经编写了一个文档，解释了如何在实际操作中做到这一点，例如，如何处理许可头文件等等（https://www.mozilla.org/en-US/MPL/2.0/combining-mpl-and-gpl/）。

**15.第3.3节与其他许可证混合的能力是否会导致更多的许可证扩散?**

根据Mozilla使用MPL 1.1的经验，以及我们一些顾问的经验，在实践中许可证不兼容通常是通过使用自定义的附加兼容许可或双(三)许可证来解决的，然而这种方式都会进一步使许可证交互和扩散分析复杂化。我们认为，第3.3节将这些解决方案替换为针对最常见情况的单一标准化解决方案，应该减少而不是增加许可证扩散的实际风险。

**16. "minified" JavaScript是源代码吗?**

不。"minified" JavaScript，虽然不是软件工程意义上的“可执行”，但对人类来说很难阅读、编辑和修改，因此，它不是“修改的首选形式，它不是许可证定义的源代码”，因此"minified" JavaScript是可执行格式，当你分发MPL许可的"minified" Javascript时，应满足可执行格式分发许可证中规定的义务。这意味着，在分发"minified" JavaScript时，您不需要保留MPL模板文件（以“This Source Code Form…”开头），但是您需要遵守3.2(a)节，通知接收者如何获得源代码的副本，具体如何做将取决于他们如何获得该副本，一种方法是在使用JavaScript的页面或JavaScript文件本身中包含带有源代码链接的注释。

注意，将"minified" JavaScript作为可执行文件，通过允许MPL代码与非MPL JavaScript源代码组合成单个文件，而不要求非MPL代码在MPL的条款下分发，增加了分发者的灵活性。

**17. “分发”是什么意思?**

MPL使用“分发”是指将软件的副本交付给另一个人或实体，我们使用“分发”并不是指“在不交付软件副本的情况下通过网络使功能可用”。例如，在一个基于web的应用程序中，在服务器上运行的代码不是“分发”给用户的，但发送给客户端的代码(例如HTML、CSS、JavaScript)却被视为“分发”。

**18.MPL应该用于非软件作品吗?**

MPL是基于软件而编写的，通常只适用于软件。然而，为了一致性和简单性，将MPL用于非软件作品(例如文档、图像和声音文件)可能是合适的，这些作品主要是为使用MPL许可的软件而编写的。

**19.我可以删除或更改软件显示的通知语句吗?**

第3.4节禁止源码中对许可通知的大量更改，我们不鼓励更改在软件在执行时的许可和通知声明。然而，这种更改是允许的，因此源代码可以在用户界面非常不同的软件之间重用。

**20.我下载了一个MPL软件。就第1.1节而言，我是该软件的“所有者”吗?**

不。就第1.1节而言，仅仅下载该软件并不使您成为所有者。

**21.MPL 2.0是否允许我通过改变MPL来制作自己的许可证?**

是的。但是，和MPL-1.1一样，我们强烈建议你不要这么做，因为这几乎肯定会使你的软件变得不那么受欢迎和广泛使用。软件开发人员和公司已经意识到并理解像MPL这样的流行许可证。如果你创建了自己的许可证，他们将不得不对你的变化进行法律评估，可能会得出这样的结论:不值得这么做，或者，您可能不小心使您的软件许可证与自由软件定义或开放源码定义不兼容，或者与MPL 2.0兼容的其他常用自由软件许可证不兼容。

如果您喜欢MPL 2.0，就按原样使用它吧——它是一个清晰的、现代化的、国际化的、通用的许可证。在其许可条款中，没有任何特定于Mozilla项目或特定于某个国家的条款。

有关创建自己的许可证导致的问题的更多信息，请参阅维基百科关于许可证扩散的文章（https://en.wikipedia.org/wiki/License\_proliferation）。

**22.MPL 2.0是否要求在每个文件中都包含MPL 2.0许可通知头?**

许可证通知必须以某种方式“附加”到每个文件中（章节1.4）。在不可能或不切实际的情况下，可以通过将通知放在接收者“可能会寻找这样的通知”的地方来满足这个要求，例如在与文件相同的目录中的许可文件。许可通知尽可能短（例如附录a，3行），以便尽可能多地放入不同类型的文件中。

虽然许可证允许将许可证通知放在文件本身之外的其他地方，但单独的文件很容易会在没有软件授权的情况下分发，因此，将许可通知放在文件中是确保接收方始终得到许可通知的最可靠的方法。为了方便您，Mozilla已经生成了样板文件用几种常见格式格式化的头文件“评论”语法，合适用于复制和粘贴。

**23.在章节1.2的贡献者版本定义中 “使用”是什么意思?**

第1.2节中的“使用”是指在创建贡献或修改过程中采取的行动。

**24.“贡献者”指创建“覆盖软件”的人，“覆盖软件”是包括MPL许可源代码的“可执行格式”。这是否意味着编译未修改的代码来创建可执行文件会使某些人成为贡献者?**

不。我们希望贡献者是那些对程序进行了添加或更改，向已有代码添加了新的表达（版权保护作品）的人。纯粹的编译并不是一种授权行为，它不会创建这样的添加，因此不会使您成为一个贡献者。

**25.如果有人不在每个文件使用样板文件，而只是分发一个完整的MPL-2.0副本和他们的代码，会发生什么?**

该代码是在普通的MPL-2.0下许可的，它不被认为与二级许可不兼容。使代码与二次许可兼容是默认的，“不兼容”要求许可方做出积极的选择，即附录B。

唯一的例外是，如果代码过去是直接MPL-1.1，升级为MPL-2.0，在这种情况下，它将是不兼容的二级许可(第1.5 b节)。

**26.我想通过快递或其他物理机制分发源代码，就履行第3.2(a)条规定的义务而言，这是否“合理”?**

由于成本低、方便，分销几乎完全通过互联网进行。虽然MPL在必要时允许选择，但一般而言，没有特定的必要目的而增加成本和复杂性的任何机制(如快递)都是不合理的，因此不符合第3.2(a)条的要求。