

这是针对英文原版页面的中文翻译。

使用GNU自由文档许可证的技巧

本文简要解释如何让文档使用GNU自由文档许可证。关于GNU软件许可证，比如GNU GPL等，请参考单独的页面。

如何使用FDL的可选功能？

我们有单独的页面来讨论如何使用FDL的可选功能。

许可证声明必须放在哪里？

将完整的许可声明放在您文档的“主要”文件里就足够了。举例来说，在*Emacs Lisp*参考手册中们有许多被`elisp.texi`所包括的文件。我们将完整的声明保存在`elisp.texi`中，而在其他文档们可以简单描述：

```
@c -*-texinfo*-  
@c 这是GNU Emacs Lisp参考手册。  
@c Copyright (C) 1990, 1991, 1992, 1993, 1994, 1999  
@c Free Software Foundation, Inc.  
@c 请参见elisp.texi文件来了解版权声明。
```

如何对待我正确处理文档所需要的脚本程序？

我们建议您也在GNU FDL下发布这些脚本程序。当然，如果这些脚本程序对于其他任务也通用，在GNU GPL下单独发布也是个好主意。

文档一定得要有恒定部分¹吗？

不。一个没有恒定部分的文档完全是可以接受的。

我是否应该确保文档带有一些恒定部分？

并没有特别要求。如果一个章节的内容应该是恒定的，那么就让它成为恒定部分。

我是否应该试著避免恒定部分？

并没有特别要求。如果一个章节的内容应该是恒定的，那么就让它成为恒定部分。

什么时候一个章节应该是恒定的呢？

首先您要知道的是，用来处理技术材料的章节不可以是恒定的。只有次要章节可以是恒定的，一个技术章节并不是次要章节。

如果某章节是您不被允许作出修改的文本，比如GNU GPL的一份副本，那么它必须是恒定的。您没有被允许修改它，那么您也不可以授权修改它。

（其中一个后果是：您不可以使用早先存在的涵盖技术材料的文本，如果您并没有被允许可以文本作出修改。）

当一个章节讨论自由软件的哲学时，使那个章节成为恒定部分是个好主意。举例来说，当我们GNU宣言（GNU Manifesto）放到使用手册中时，或是当我们包括一个解释“为什么自由文档是好的”的章节时，我们使该章节成为恒定部分。

译注

1. invariant section，恒定部分，再发布时不能删除或改变的部分。

Copyright © 2000, 2001, 2008, 2009, 2014, 2022 Free Software Foundation, Inc.

本页面使用 Creative Commons Attribution-NoDerivatives 4.0 International License 授权。

翻译: 刘昭宏 <chliu@gnu.org>

审校: 马雪萍

翻译团队: <CTT>, 2018, 2022。