

# 筑梦小屋

梦是让人欣喜的回忆,梦是灵感的发源地. 暴走笑话 奶茶加盟 上海装修公司 模块电源 上海夜场招聘

博客园 闪存 首页 新随笔 联系 管理

随笔- 77 文章- 0 评论- 100

昵称： 枫枫  
园龄： 4年8个月  
粉丝： 99  
关注： 3  
+加关注

## 开源项目的各种协议简单介绍

很多企业在进行开发的时候，或多或少都会使用到开源项目。

这时候就要考虑到开源项目所执行的协议了。

1. 允许商用、闭源发布的协议：BSD、Apache、MIT 协议、LGPL（更宽松的GPL）

### 1.1 BSD 协议

BSD开源协议是一个给予使用者很大自由的协议。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。当你发布使用了BSD协议的代码，或者以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。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。

### 1.2 Apache协议

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：

1. 需要给代码的用户一份Apache Licence
2. 如果你修改了代码，需要在被修改的文件中说明。
3. 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。
4. 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。

英文原文：<http://www.apache.org/licenses/LICENSE-2.0.html>

### 1.3 MIT协议

MIT许可证之名源自麻省理工学院（Massachusetts Institute of Technology, MIT），又称「X条款」（X License）或「X11条款」（X11 License）

< 2012年2月 >						
日	一	二	三	四	五	六
29	30	31	1	<u>2</u>	3	4
5	<u>6</u>	7	<u>8</u>	9	10	11
12	13	14	15	16	17	18
19	20	21	22	<u>23</u>	24	25
26	27	28	29	1	2	3
4	5	6	7	8	9	10

## 搜索

找找看

谷歌搜索

## 常用链接

我的随笔  
我的评论  
我的参与  
最新评论  
我的标签  
更多链接

## 最新随笔

1. java.util.concurrent介绍
2. LogBack简易教程
3. 领域驱动设计基本理论知识总结(zhuan)
4. SyntaxHighlighter代码高亮
5. C/C++崩溃异常与Minidump
6. Java类加载全过程(zhuang)
7. SQL常用的语句和函数
8. 精妙Sql语句
9. win7重现xp传统工具栏
10. RedHat静态ip地址配置

## 随笔分类

Apache开源项目(4)  
C/C++(1)  
Html&CSS(1)  
J2EE(14)  
J2SE(20)  
JavaScript(2)  
Linux(3)  
MySQL(9)  
SSH(2)  
Swing(2)  
计算机网络(4)  
设计模式(7)  
生活小记(5)  
小窍门(6)

MIT内容与三条款BSD许可证（3-clause BSD license）内容颇为近似，但是赋予软件被授权人更大的权利与更少的限制。

被授权人有权利使用、复制、修改、合并、出版发行、散布、再授权及贩售软件及软件的副本。

被授权人可根据程式的需要修改授权条款为适当的内容。

在软件和软件的所有副本中都必须包含版权声明和许可声明。

此授权条款并非属copyleft的自由软件授权条款，允许在自由/开放源码软件或非自由软件（proprietary software）所使用。

此亦为MIT与BSD（The BSD license, 3-clause BSD license）本质上不同处。

MIT条款可与其他授权条款并存。另外，MIT条款也是自由软件基金会（FSF）所认可的自由软件授权条款，与GPL相容。

协议英文原文：<http://www.opensource.org/licenses/mit-license.php>

1.4 LGPL协议

LGPL最初是Library GPL的缩写，后来改称作Lesser GPL，即为更宽松的GPL。

当一个自由软件使用GPL声明时，该软件的使用者有权重新发布、修改该软件，并得到该软件的源代码；但只要使用者在其程序中使用了该自由软件，或者是使用修改后的软件，那么使用者的程序也必须公布其源代码，同时允许别人发布、修改。也就是说，使用GPL声明下的的自由软件开发出来的新软件也一定是自由软件。

**LGPL是GPL的变种，也是GNU为了得到更多的甚至是商用软件开发商的支持而提出的。**与GPL的最大不同是，可以私有使用**LGPL**授权的自由软件，开发出来的新软件可以是私有的而不需要是自由软件。所以任何公司在使用自由软件之前应该保证在LGPL或其它GPL变种的授权下。

2. 允许商用、半闭源的协议：MPL协议 - 使自己的代码和开源项目代码混合，只公开与开源项目代码交互部分的接口源代码，从而得到半闭源商用项目的发布。

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OSIA 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。

英文原文：<http://www.mozilla.org/MPL/MPL-1.1.html>

3. 允许商用、不可闭源(需开源)的协议：GPL、AGPL 协议

3.1 GPL 协议 (用于软件产品)

在自由软件所使用的各种许可证之中，最为人们注意的也许是通用性公开许可证(General Public License，简称GPL)。

最新评论

- 1. Re:MySQL索引和查询优化  
--格先生
- 2. Re:MySQL索引和查询优化  
顶  
--格先生
- 3. Re:基于java的截图工具(含代码)  
你好，对这个很感兴趣！请问能把压缩包发给我吗？  
507824632@qq.com  
--java\_小刘
- 4. Re:JTable的入门详细介绍  
真心不错  
--小小菜鸟鸟
- 5. Re:基于java的截图工具(含代码)  
大家可以到这里下载源码：  
--Hongten

评论排行榜

- 1. 基于java的截图工具(含代码)(42)
- 2. 使用json-lib进行Java和JSON之间的转换(13)
- 3. Apache Mina 2.x 简易入门解析(9)
- 4. log4j详解及简易搭建(5)
- 5. java的3DES加密(4)

推荐排行榜

- 1. 使用json-lib进行Java和JSON之间的转换(9)
- 2. 基于java的截图工具(含代码)(7)
- 3. log4j详解及简易搭建(6)
- 4. JAVA之编码/解码 -- 各种环境下可能会发生的乱码问题及解决方案(6)
- 5. Apache Mina 2.x 简易入门解析(6)

GPL同其它的自由软件许可证一样，许可社会公众享有：运行、复制软件的自由，发行传播软件的自由，获得软件源码的自由，改进软件并将自己作出的改进版本向社会发行传播的自由。

GPL还规定：只要这种修改文本在整体上或者其某个部分来源于遵循GPL的程序，该修改文本的 整体就必须按照GPL流通，不仅该修改文本的源码必须向社会公开，而且对于这种修改文本的流通不准许附加修改者自己作出的限制。因此，一项遵循GPL流通 的程序不能同非自由的软件合并。GPL所表达的这种流通规则称为copyleft，表示与copyright(版权)的概念“相左”。

GPL协议最主要的几个原则：

1、确保软件自始至终都以开放源代码形式发布，保护开发成果不被窃取用作商业发售。任何一套软件，只要其中使用了受 GPL 协议保护的第三方软件的源程序，并向非开发人员发布时，软件本身也就自动成为受 GPL 保护并且约束的实体。也就是说，此时它必须开放源代码。

2、GPL 大致就是一个左侧版权（Copyleft，或译为“反版权”、“版权属左”、“版权所无”、“版责”等）的体现。你可以去掉所有原作的版权信息，只要你保持开源，并且随源代码、二进制版附上 GPL 的许可证就行，让后人可以很明确地得知此软件的授权信息。GPL 精髓就是，只要使软件在完整开源 的情况下，尽可能使使用者得到自由发挥的空间，使软件得到更快更好的发展。

3、无论软件以何种形式发布，都必须同时附上源代码。例如在 Web 上提供下载，就必须在二进制版本（如果有的话）下载的同一个页面，清楚地提供源代码下载的连接。如果以光盘形式发布，就必须同时附上源文件的光盘。

4、开发或维护遵循 GPL 协议开发的软件的公司或个人，可以对使用者收取一定的服务费用。但还是一句老话——必须无偿提供软件的完整源代码，不得将源代码与服务做捆绑或任何变相捆绑销售。

### 3.2 AGPL协议 （常用于函数库或者工具包）



这是一份 GNU 较宽松公共许可证非正式的中文翻译。它不是自由软件基金会所发布，并且不能适用于使用 GNU LGPL 的软体——只有 GNU LGPL 英文原文的版本才行。然而，我们希望这份翻译能帮助中文的使用者更了解 GNU LGPL。

GNU 较宽松公共许可证

1999.2，第 2.1 版

版权所有 (C) 1991, 1999 Free Software Foundation, Inc.  
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

允许每个人复制和发布本授权文件的完整副本，  
但不允许对它进行任何修改。

[这是第一次发表的较宽松公共许可证 (Lesser GPL) 版本。它同时也可视为 GNU 函数库公共许可证 (GNU Library Public License) 第 2 版的后继者，故称为 2.1 版]

本版本由 Leo-Hong (leohca (at) yahoo.com) 翻译整理，Chao-Hong Liu 校正。

#### 导言

大多数软体许可证决意剥夺您共享和修改软体的自由。相反的，GNU 通用公共许可证力图保证您共享和修改自由软体的自由——保证自由软体对所有使用者都是自由的。

这个许可证，较宽松公共许可证，适用于一些由自由软件基金会与其他决定使用此许可证的软体作者，所特殊设计的软体套件——象是函数库。您也可以使用它，但我们建议您事先仔细考虑，基于以下的说明是否此许可证或原来的通用公共许可证在任何特殊情况下均为较好的方案。

当我们谈到自由软体时，我们所指的是自由，而不是价格。我们的 GNU 通用公共许可证是设计用以确保使您有发布自由软体备份的自由（如果您愿意，您可以对此项服务收取一定的费用）；确保您能收到程式原始码或者在您需要时能得到它；确保您能修改软体或将它的一部分用于新的自由软体；而且还确保您知道您可以做上述的这些事情。

为了保护您的权利，我们需要作出限制：禁止任何人否认您上述的权利，或者要求您放弃这些权利。如果您发布软件的副本，或者对之加以修改，这些规定就转化为您的责任。

例如，如果您发布此函数库的副本，不管是免费还是收取费用，您必须将您享有的一切权利给予接受者；您必须确保他们也能收到或得到原始程式码；如果您将此函数库与其他的程式码连结，您必须提供完整的目的对象文件和程序 (object file) 给接受者，则当他们修改此函数库并重新编译过后，可以重新与目的档连结。您并且要将这些条款给他们看，使他们知道他们有这样的权利。

我们采取两项措施来保护您的权利：（1）用版权来保护函数库。并且，（2）我们提供您这份许可证，赋予您复制，发布和（或）修改这些函数库的法律许可。

为了保护每个发布者，我们需要非常清楚地让每个人明白，自由函数库是没有担保责任的。如果由于某人修改了函数库，并继续加以传播，我们需要它的接受者明白：他们所得到的并不是原始的版本。故由其他人引入的任何问题，对原作者的声誉将不会有任何的影响。

最后，由于软体专利不断地威胁自由软体的存在，我们希望商业公司无法藉由自专利持有者取得一个受限的许可证，而有效地限制自由软体的使用者。因此，我们坚持一个函数库所能取得的任何专利，必须与本许可证所声明的“完全自由使用”一致。

<20040222>

大部分的 GNU 软体，包括一些函数库，是受到原来的 GNU 通用公共许可证的保护。本许可证，GNU 较宽松通用公共许可证，适用于特殊设计的函数库，且与原来的通用公共许可证有很大的不同。我们在特定的函数库中使用它，以准许非自由的程式可以与这些函数库连结。当一个程式与一个函数库连结，不论是静态连结或使用共享函数库，二者的结合可以合理地说是结合的作品，一个原来的函数库的衍生品。因此，原来的通用公共许可证只有在整个结合品满足其自由的标准时，才准许连结。较宽松通用公共许可证则以更宽松的标准允许其他程式码与本函数库连结。

我们称此许可证“较宽松”通用公共许可证，是因为它比起原来的通用公共许可证对使用者的自由做到较少的保护。在与非自由软体竞争时，它也提供其他自由软体的写作者较少的优势。这些不利之处正是我们使用原来的通用公共许可证于许多函数库的理由。然而，较宽松的许可证可在某些特殊场合下带来好处。例如，在少数情况下，可能会有特殊的需要而鼓励大家尽可能广泛地使用特定的函数库，因而使它成为实际上的标准。为了达到此目标，必须允许非自由的程式使用此函数库。一个较常发生的情况是一个自由的函数库与一个被广泛使用的非自由函数库做相同的工作，在此情况下，限制只有自由软体可以使用此自由函数库不会有好处，故我们如用了较宽松通用公共许可证。

在其他情况下，允许非自由程式使用特定的函数库，可以让更多的人使用自由软体的大部分。例如，允许非自由程式使用 GNU C 函数库可以让更多的人使用整个 GNU 作业系统，以及它的变形，GNU/Linux 作业系统。

尽管较宽松通用公共许可证对使用者的自由是较少的保护的，它却能确保与此函数库连结的程式的使用者拥有自由，而且具有使用修改过的函数库版本来执行该程式的必要方法。

以下是复制、发布、以及修改的精确条款与条件。请注意“基于函数库的作品”以及“使用函数库的作品”之间的差异：前者包含来自函数库修改过的原始码；而后者则必须与函数库结合才能执行。

有关复制，发布和修改的条款和条件

0. 本许可证适用于任何软体函数库，或其他包含了由版权所有者的加入的注意事项的程式，或其他有公信力的团体宣称其程式可以在较宽松通用公共许可证（也称之为“本许可证”）的条款下发布。每一位许可证接受者以“您”来称呼。

一个“函数库”意指一些软体函数的集合，以及或准备好的资料以方便与应用程式（其使用了其中某些函数与资料）连结形成可执行的程式。

以下，“函数库”一词指的是任何在本条款下发布的这一类软体函数库或作品，一个“基于本函数库的作品”意指函数库或任何在版权法下的衍生作品：也就是说，一个包含了本函数库或其一部分的作品，可以是原封不动的，或经过修改的，和/或直接翻译成其他语言的。（在下文中，翻译是不受限地包含在“修改”的条款中。）

作品的“原始码”意指对作品进行修改最优先择取的形式。对函数库而言，完整的原始码意指所有模组的所有原始程式，加上有关的介面的定义，加上控制函数库的安装和编译的 script。

本许可证条款不适用于复制，发布和修改以外的活动。这些活动超出这些条款的范围。使用本函数库来执行本程式的动作不受条款的限制，而程式的输出只有在其内容所构成的作品是基于本函数库时（与在什么样的工具中使用本函数库来输出无关），这一条款才适用。以上是否为真则取决于本函数库具体用来做什么。

1. 只要您在每一程式副本上明显和恰当地宣告版权声明和不承担担保的声明，并保持此许可证的声明和没有担保的声明完整无损，并和程式一起给其他每位程式接受者一份许可证的副本，您就可以用任何媒体复制和发布您收到的函数库的完整原始码。

您可以为转让副本的实际行动收取一定费用。您也可以选择提供担保以换取一定的费用。

2. 只要您同时满足下面的所有条件，您就可以按前面第一款的要求修改函数库的一个或几个副本或它的任何部分，以此形成基于此函数库的作品，并且复制和发布这一经过修改的程式或作品：

被修改的作品本身必须是一个软体函数库。

您必须在修改过的档案中附有明确的说明：您修改了此一档案及任何修改的日期。

您必须让整个作品允许第三方在此许可证条款下可以免费使用。

如果修改过的函数库其某个设备使用到了「使用本函数库的应用程式」所提供的函数或资料表格，却不是当此设备被呼叫时以参数列传入时，则您必须确实做到，当應用程式不提供这样的函数或表格时，则此设备依旧能工作，且其执行的任何目的仍然有意义。

(例如，一个函数库的函数用来计算平方根，其目的是有完整的定义且与应用程式是无关的。因此，2d 小节要求任何本函数库会使用的，由應用程式所提供的函数或表格必须是选择性的：如果應用程式不提供的话，则计算平方根的函数必须依旧能计算平方根)

这些要求适用于整个修改过的作品。如果能够确定作品的一部分并非本函数库的衍生产品，且可以合理地单独考虑并将它与原作品分开的话，则当您将它作为独立的 作品发布时，它不受此许可证和其条款的约束。但是当您将这部分与基于本函数库的作品一同发布时，则整个套件将受到本许可证条款约束，其对于其他许可证持有 人的使用范围扩大到整个产品，也就是套件的每个部分，不管它是谁写的。

因此，本条款的意图不在于索取权利，或剥夺完全由您完成的作品权利，而是履行权利来控制基于本函数库的集体作品或衍生作品的发布。 此外，将与本函数库无关的作品和本函数库 (或基于本函数库的作品) 一起放在贮存媒体或发布媒体的同一卷上，并不导致将其他作品置于此许可证的约束范围之内。

3. 对于一个函数库的副本，您可以选择性地使用原来的 GNU 通用公共许可证上的条款来取代本许可证上的条款。如果您要这么做，您必须修改所有的参考到本许可证的注意事项，使它们指向原来的 GNU 通用公共许可证，第二版，以取代本许可证 (如果有比第二版的原来的 GNU 通用公共许可证更新的版本出现的话，则如果您愿意的话可以特别指明使用新版)。请不要对这些注意事项做出其他的改变。

一旦在一个副本上做了这样的改变，则该副本就无法撤回这样的改变，故原来的 GNU 通用公共许可证将适用于所有后续的副本以及由此副本衍生出来的作品。

此一选择性适用于当您想要将一部分的函数库原始码复制到一个非函数库的程式使用时。

4. 您可以以目标码或可执行形式复制或发布本函数库 (或符合第 2 款，基于本函数库的作品)，只要您遵守前面的第 1、2 款，并同时提供完整的相关机器可读的原始码，而这些原始码必须在前面的第 1 与第 2 款条件下，在一般习惯上用来做软体交换的媒体上发布。

如果所发布的目标码是由指定的地点提供拷贝索取，那么由同一地点所提供等价的原始码拷贝索取可以算作原始码的发布，即使第三方不强求与目标码一起复制原始码。

5. 一个程式若包含不经任何部分修改的函数库，但却是设计经由编译或连结的方式与本函数库一同工作者，称之为 **"使用函数库的作品"**。这样的作品，严格地说，并非本函数库的衍生作品，因而不在本许可证的范围之内。

然而，将 **"使用函数库的作品"** 与本函数库连结而产生可执行程式，则是本函数库的衍生品 (因为它包函了本函数库的一部分)，而不是 **"使用函数库的作品"**，因此其可执行程式包含在本许可证的范围内。第 6 款说明了发布此可执行程式的条款。

当 **"使用函数库的作品"** 使用了函数库部分的标头档内容时，则此作品即使其原始码不属于本函数库的衍生品，但其目标码仍然是。这一点是否为真特别在是否本作品可以在不需要本函数库即可连结，或者是否该作品本身也是一个函数库时特别明显。

如果这样的目标档只使用数字参数、资料结构层级与附属品、以及小巨集和小内嵌式 (小于或等于十行)，则此目标档的使用是不受限的，不论它是否是合法的衍生作品。(但可执行程式若包函此目标档以及一部分的函数库，仍然将在第 6 款的规定下)

否则的话，如果本作品是本函数库的衍生品，您必须第 6 款的规定下发布该作品的目标码。任何包含该作品的可执行程式也在第 6 款的范围内，不论它们是否直接与本函数库连结。

6. 做为上述条款的例外情况，您也可以将 **"使用函数库的作品"** 与本函数库结合或连结，以产生包含部分本函数库的作品，并在允许使用者自身使用时可以修改该作品，以及在对修改进行反组译除错的情况下，您可以依照您的选择发布该作品。

您必须在每个作品的副本突出出如下的注意事项：本函数库在作品中被使用，以及本函数库以及它的使用是在本许可证的规定下。您必须提供本许可证的副本。如果 该作品在执行时显示版权声明，您必须在其中包含本函数库的版权声明，以及指引使用者取得本许可证的副本。同时，您必须做到以下其中一件事：

必须将完整的机器可读的函数库原始码包含在该作品中，包括任何该作品使用到的改变 (这些改变必须在前述第 1 与第 2 款的要求下发布)；而且，如果该作品是一个与函数库连结的「完整的、机器可读的 **"使用函数库的作品"**」，则要有目标码和/或原始码，如此使用者可以修改本函数库且可以重新连结，以产生包函修改过的函数库的修改过的可执行程式。(理所当然的若使用者修改了函数库的档案定义内容时，则该作品不必然可以重新编译以使用修改过的定



义。)

在与函数库连结时使用适当的分享函数库连结机制。一个适当的机制是： (1) 在执行时使用已存在于使用者的电脑中的函数库副本，而不是将函数库的函数复制到可执行程式里，以及 (2) 如果使用者安装了一份修改过的函数库，只要修改过的版本在介面上与该作品在编译连结时所用的版本是相容的，则该执行程式可以与修改过的函数库运作良好。

在该作品内提供书面报价，有效期不少于三年，以提供同样的使用者上述第 6a 款中的内容，费用不得超过该程式发布的实际成本。 如果所发布的作品是由指定的地点提供拷贝索取，则由同一地点提供上述内容的等价拷贝索取。

确定使用者已经收到该作品的一份复制，或是您已经寄给该使用者一份复制品。

对于一个可执行程式，其所需的 **“使用函数库的作品”** 的形式必须包括任何要从中再产生可执行程式时所需的资料与工具程式。然而，有一个特殊例外，其所发布的内容不需要包括任何一般与「可执行本程式的作业系统」的主要部分 (如编译器、核心等) 一起发布的部分 (不论是原始码或可执行码)，除非这些组成部分和可执行作品结合在一起。

有一个可能情况是，这些要求与其他通常不与作业系统在一起的私有函数库的版权限制相抵触，这样的抵触表示您不能将它们与本函数库一起用于您发布的可执行程式中。

7. 您可以将使用本函数库的函数库设备，以及其他不在本许可证范围内的函数库，对等地放入一个单独的函数库中，并在基于本函数库的作品以及其他函数库在其他状态下同意可以个别发布，以及您做到以下两点的情况下，您可以发布此结合的函数库：

将基于本函数库的作品单独不与其他函数库设备结合地，与此结合的函数库一同发布。该作品必须在上述条款的规定下发布。

在此结合的函数库中明显地指出其中一部分的作品是基于本函数库，并且说明那里可以找到同样不具结合形式的作品。

8. 除非您明确按许可证提出的要求去做，否则您不能复制、修改、转发许可证、与本函数库连结、和发布本函数库。任何试图用其他方式复制、修改、转发许可证、与 本函数库连结、和发布本函数库是无效的，而且将自动结束许可证赋予您的权利。然而，对那些从您那里按许可证条款得到副本和权利的人们，只要他们继续全面履行条款，许可证赋予他们的权利仍然有效。

9. 您没有在许可证上签字，因而您没有必要一定接受此一许可证。然而，没有任何其他东西赋予您修改和发布本函数库及其衍生作品的权利。如果您不接受许可证，这些行为是法律禁止的。因此，如果您修改或发布函数库 (或任何基于函数库的作品)，您就表明您接受这一许可证以及它的所有有关复制、发布和修改本函数库或基于它的作品的条款和条件。

10. 每当您重新发布函数库 (或任何基于函数库的作品) 时，接受者自动从原始许可证颁发者那里接到受这些条款和条件支配的复制、发布、连结或修改本函数库的许可。您不可以强迫接受者履行除了这里赋予他们的权利之外的其他限制。您也没有强求第三方履行许可证条款的义务。

11. 如果由于法院判决或违反专利的指控或任何其他原因 (不限于专利问题) 的结果，使得强加于您的条件 (不管是法院判决，协议书或其他) 和许可证的条件有冲突时，他们也不能令您背离许可证的条款。在您不能同时满足本许可证规定的义务及其他相关的义务来发布函数库时，则结果您只能根本不发 布函数库。例如，如果某一专利许可证不允许所有直接或间接从您那里接受副本的人们，在不付专利费的情况下重新发布函数库，唯一能同时满足两方面要求的办法 是停止发布函数库。

如果本条款的任何部分在特定的环境下无效或无法实施，就使用条款的其余部分，并将这部分条款作为整体用于其他环境。 本条款的目的不在于引诱您侵犯专利或其他财产权的要求，或争论这种要求的有效性。本条款的主要目的在于保护自由软体发布系统的完整性。它是通过公共许可证 的应用来实现的。许多人已依赖同是出自此系统的应用程式，经由此系统发布大量自由软体而做出慷慨的奉献。作者/捐献者有权决定他/她是否通过任何其他系统 发布软体，许可证持有人不能强加这种选择。

本节的目的旨在明确说明许可证其余部分可能产生的结果。

12. 如果由于专利或者由于有版权的介面问题使函数库在某些国家的发布和使用受到限制，则在许可证约束下的原始版权拥有者可以增加发布地区的限制条款，将这些国家明确排除在外，并在这些国家以外的地区发布函数库。在这种情况下，许可证套件含的限制条款和许可证正文一样有效。 13. 自由软体基金会可能随时出版较宽松通用公共许可证的修改版或新版。新版和当前的版本在原则上保持一致，但在提到新问题时或有关事项时，在细节上可能出现差别。

每一版本都有不同的版本号。如果函数库指定可适用的许可证版本号以及 **“任何更新的版本”**，您有权选择遵循指定的版本或自由软体基金会以后出版的新版本。如果函数库未指定许可证版本，您可选择自由软体基金会已经出版的任何版本。 14. 如果您愿意将函数库的一部分结合到其他自由程式中，而它们的发布条件不同，请写信给作者，要求准予使用。如果是自由软体基金会加以版权保护的软体，写信给 自由软体基金会，我们有时会作为例外的情况处理。我们的决定受两个主要目标的指导，这两个主要目标是：我们的自由软体的衍生作品继续保持自由状态，以及从 整体

上促进软体的共享和重复利用。

没有担保

15. 由于函数库准予免费使用，在适用法准许的范围内，对函数库没有担保。除非另有书面说明，版权所有者和/或其他提供函数库的人们 "一样" 不提供任何类型的担保，不论是明确的，还是隐含的，包括但不限于可销售和适合特定用途的隐含保证。全部的风险，如函数库的质量和性能问题都由您来承担。如果函数库出现缺陷，您应当承担所有必要的服务、修复和改正的费用。

16. 除非适用法或书面协议的要求，在任何情况下，任何版权所有或任何按许可证条款修改和发布函数库的人们都不对您的损失负有任何责任。包括由于使用或不能使用函数库引起的任何一般的、特殊的、偶然发生的或重大的损失（包括但不限于数据的损失，或者数据变得不精确，或者您或第三方的持续的损失，或者函数库不能和其他软体协调运行等）。即使版权所有和其他人提到这种损失的可能性也不例外。

如何将这些条款用到您新的函数库

如果您开发了新函数库，而且您需要它得到公众最大限度的利用，要做到这一点的最好办法是将它变为自由软体，使得每个人都能在遵守本条款（或者是在原来的通用公共许可证的条款）的基础上对它进行修改和重新发布。

为了做到这一点，请将函数库附上下列声明。最安全的方式是将它放在每个原始码档案的开头，以便最有效地传递拒绝担保的信息。每个文件至少应有 "版权所有" 行以及在什么地方能看到声明全文的说明。

用一行空间描述函数库的名称和它的用途简单说明

版权所有 (C) 19XX 作者姓名

这一函数库是自由软体，您可以遵照自由软体基金会出版的 GNU 较宽松通用公共许可证条款来修改和重新发布这一程式，或者用许可证的第二版，或者（根据您的选择）用任何更新的版本。

发布这一函数库的目的是希望它有用，但没有任何担保。甚至没有适合特定目的而隐含的担保。更详细的情况请参阅 GNU 较宽松通用公共许可证。

您应该已经和函数库一起收到一份 GNU 较宽松通用公共许可证的副本。如果还没有，写信给：

Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA.

此外还应加上如何藉由电子邮件或一般信件与您保持联系的信息。

如果需要，您应该取得您的上司（如果您是程式员）或您的学校签署放弃函数库版权的声明。下面只是一个例子，您应该改变相应的名称：

Yoyodyne 公司以此方式放弃 James Random Hacker 所写的 'Frob' 函数库（用以扭转 knobs 的函数库）的全部版权利益。

Ty coon 签名, 1990.4.1

Ty coon 副总裁



分类: [J2EE](#)

标签: [协议](#)、[开源协议](#)

好文要顶

关注我

收藏该文



枫枫

[关注 - 3](#)

[粉丝 - 99](#)

[+加关注](#)

0

0

(请您对文章做出评价)

« 上一篇: [openfire3.7.1源码编译过程](#)

» 下一篇: [Java内存分配、管理\(转\)](#)

posted @ 2012-02-02 14:49 枫枫 阅读(2036) 评论(0) 编辑 收藏

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问网站首页](#)。

【推荐】50万行VC++源码: 大型组态工控、电力仿真CAD与GIS源码库

【推荐】融云即时通讯云 – 专注为 App 开发者提供IM云服务

【推荐】UCloud开年大礼，充5000返1000；买云主机送CDN，详情点击



最新IT新闻:

- 大数据怎么创造价值？硅谷最牛商业分析团队告诉你
- 投资三大法则：简单为王、趋势为王、增速为王
- 一篇文章看懂苹果和FBI四大回合的交锋
- 谷歌发布全景视频：这就是最强数据中心
- 内向的人很难成为群体程序员吗？

» 更多新闻...



最新知识库文章:

- 为什么未来是全栈工程师的世界？
- 程序bug导致了天大的损失，要枪毙程序猿吗？
- 如何运维千台以上游戏云服务器
- 架构漫谈（一）：什么是架构？
- 架构的本质

» 更多知识库文章...

Copyright ©2016 枫枫