

GPL协议解读及合规建议

作者：贾媛媛 籍婷 丁畅 王聪睿子

引言

在第一篇文章《开源软件与法律探析导读》中，我们介绍了开源、开源协议（也称“开源许可证”）的相关概念、法律性质以及相关知识产权风险。

本文将针对目前发生法律纠纷较多的GPL协议进行详细解读。我们将梳理GPL协议的发展历程，介绍对企业影响最为显著的传染性条款，并分析和总结司法实践中的关注要点，如①GPL开源协议的性质、②开源项目贡献者之一是否有权提起诉讼、③传染性的判断标准、④使用GPL开源开发工具编写的软件是否会被传染等，从而向企业提供开源合规建议和指引，以期为企业的风险排查提供思路。

一、GPL开源协议简介

GPL开源协议是开源领域被广泛使用的协议，自1989年发布以来，共有3个版本（GPL v1/v2/v3） [1]，著名的Linux中就使用了GPL v2协议。GPL协议属于典型的著佐权协议，其具有强传染性，要求使用者公开相应源代码。因此，因违反GPL开源协议的规定而产生纠纷的情形较多，主要涉及GPL v2与GPL v3协议，故本文将重点介绍这两种开源协议。

1. GPL v2与GPL v3开源协议简介

GPL v2与GPL v3开源协议虽为两个独立版本，但作为GPL协议家族的成员，两者均继承了GPL类开源协议的基本宗旨与精神，都给予了用户使用开源软件的自由，同时要求用户保证前述自由的延续性（例如分发给下游用户时需要开放源码以及提供开源协议）。

GPL v2开源协议于1991年6月发布，被应用于Linux的Spacewalk，C/C++库的API Sanity Checker、华为OpenHarmony的一款打包工具等知名项目。

GPL v3开源协议于2007年6月29日发布，其应用于Linux的4Pane、微信小程序、SaaS运营平台的SAPI++、Ansible等知名项目。

为了更好地适应当时开源软件的开发和使用，GPL v3开源协议在GPL v2的基础上进行了澄清、细化和补充，主要涉及以下方面：

· 禁止 Tivoization

Tivoization指的是在设计硬件时，硬件厂商使用了开源软件，并通过硬件或者DRM限制（全称为“digital rights management”，是一系列访问控制技术，类似于数字“保护锁”，可以控制和限制数字媒体的使用过程），使得用户无法在原有硬件上运行修改过的开源软件[2]。也即，即便硬件厂商按照开源协议的要求公开了源代码，但用户仍然无法在硬件上正常运行修改后的源代码，实质上还是剥夺了用户对开源软件源代码进行修改的权利。

由于GPL v2未限制Tivoization，开源软件权利人无法通过GPLv2来规制硬件厂商的上述行为。因此，在更新的GPL v3开源协议中，第6条明确禁止Tivoization，要求硬件厂商不仅需要提供源码，还需要提供安装信息“*基于修改过的源码安装运行该产品中的受保护作品的修改版所需的任何方法、流程、认证码及其他信息*”，以保证用户可以在硬件上运行修改过的代码[3]。

· 要求开源软件贡献者进行专利许可

为避免开源软件贡献者通过开源协议授予用户著作权许可，但转而又针对用户提起专利侵权诉讼，GPL v3第11条明确规定“*每个贡献者就‘必要专利权利要求（essential patent claims）’授予用户非独占的、全球范围的、免费的‘专利许可’*”，其中“*‘必要专利权利要求’是指贡献者已经获取或将会获取的，可能会被他人遵守GPL条款的前提下，因制造、使用或销售本软件而侵犯的所有专利权利要求*”[4]，强制开源软件贡献者同时授予用户专利许可，以保障用户对开源软件的自由使用不会受到专利的限制，而GPL v2中没有类似规定。

· 从GPL v2到GPL v3，除了前述更新之外，还有允许破解、授权中止和恢复条款等更新。

2. GPL开源协议的传染性条款

一般情形下，GPL开源协议中的众多条款中，对企业影响最大的是传染性条款。传染性条款指的是，如果用户使用了适用GPL协议的开源软件，基于该开源软件创作的后续作品的

部分或全部都应当同样进行开源。举例而言：

- 若企业原封不动的使用适用GPL协议的开源软件，那么在后续分发该开源软件时，企业需要提供该开源软件对应的全部源代码。
- 若企业不仅使用开源软件，还在开源软件的基础上进行修改开发而形成了开源软件修改版，那么在分发该开源软件修改版时也需要公开整个开源软件修改版的源代码。
- 若企业将开源软件或其修改版作为新软件作品的一部分，也即新软件作品中包含该开源软件或其修改版的源代码，那么该新软件作品也很可能同样落入传染性条款，需要公开整个新软件作品的全部源代码。
- 若企业在设计面向用户的硬件产品时，使用了适用GPL v3的开源软件，那么该企业除了需要公开相应源代码，还可能需要同时提供安装信息，包括认证码、方法、流程及其他信息，以保证用户可以在硬件上正常运行修改后的开源软件。
- 例外情况：若与开源软件或其修改版相比，企业开发的新软件作品是分离且独立的，新软件作品与开源软件或其修改版仅仅处于同一个存储介质，新软件作品并非基于开源软件或其修改版，或也不属于和开源软件或其修改版一起构建出一个更大的软件，那么该新软件作品不会受到传染性条款的影响，无需公开全部源代码。

另外，企业在公开源代码时，既可以通过任何媒介（如Github、光盘、U盘）直接公开源代码；也可以在转发开源软件的目标代码或可执行文件时，提供一个可获取源码的书面要约，告知用户如何获取源码。

二、涉及GPL开源协议的法律问题

由于GPL协议提出时间较早、应用广泛且传染性较强，目前司法实践中出现的开源软件纠纷多涉及GPL协议。笔者梳理和总结了国内外相关的司法案例，主要涉及以下几类法律问题：

1. GPL协议的法律性质

确定GPL协议的法律性质是厘清各方当事人权利和义务的基础。如我们在第一篇文章《开源软件与法律探析导读》中所述，目前国内外的司法实践中倾向于将GPL协议认定为附解除条件的著作权许可合同。具体地，

· **罗盒vs.玩友一案**[5]中，广州知识产权法院认为：“第一，协议的内容具备合同特征，属于广义的合同范畴。第二，协议是非典型合同。第三，协议是格式合同。第四，对协议的承诺是通过行为作出。GPLv3协议具有合同性质，是授权方和用户订立的格式化著作权协议，属于我国合同法调整的范围。”并进一步指出“GPLv3协议属于附解除条件的著作权合同。许可条款是版权许可的条件，如果用户违背条款规定，那么许可的前提条件已不复存在，则GPLv3协议终止适用，用户获得的授权也将自动终止”。

· **Welte vs. D-Link一案**[6]中，德国法兰克福地区法院将GPL协议认定为附解除条件的合同，其中许可使用条件为解除条件，当被许可人未按许可使用条件使用时，合同解除、终止授权，被许可人继续使用则构成侵权。

· **Jacobsen vs. Katzer一案**[7]中，美国加州北区地区法院和美国联邦巡回上诉法院认为，由于权利人通过开源项目提高了其知名度，可以被认为是开源协议的对价，因此构成许可协议，超过协议部分的使用构成著作权侵权。

2. 开源项目贡献者之一是否有权提起诉讼

开源软件的开发过程中往往具有众多贡献者。例如，根据Github开源网站的规定，开源软件项目管理人上传开源代码并创建主分支，其他用户可以基于主分支创建副本（fork），创建自己的分支并自行修改维护。其他用户也可以发起请求（pull request），申请加入主分支，项目管理者同意后，其分支并入主分支，且该用户成为主分支的贡献者。

根据著作权法的相关规定，一般认为不同贡献者对各自贡献的部分享有著作权。因而，在此类案件中，被控侵权人往往会质疑开源贡献者作为原告的适格性，主张开源软件是合作作品，具有众多贡献者，原告并非唯一权利人，只有在获取所有贡献者的许可后才能提起著作权侵权之诉。

但实践中，开源软件的众多贡献者往往来自不同的地区或国家，若要获取所有贡献者的许可后才能提起著作权侵权之诉，将会花费大量的时间和成本，无法有效保障开源软件权利人的合法权益，不利于开源软件的发展。

这种情况下，我国及其他国家的法院倾向于认为：起决定性作用的项目管理者可以作为著作权人起诉，而无需经过其他贡献者的授权；若贡献者未能证明其具体贡献以及相应的著作权，则不能作为著作权人起诉。具体地，

· 我国法院观点：起决定性作用的项目管理者可以作为著作权人起诉，而无需经过其他贡献者的授权——**罗盒vs.风灵**[8]

权利人罗盒公司为开源代码罗盒（Virtual App）的项目管理者，该项目中，共有30多个贡献者对涉案源代码做出了修改。被控侵权人风灵公司等抗辩涉案软件是合作作品，没有证据证明其他贡献者将著作权转让给原告，因此罗盒公司不是适格原告，无权就开源代码提起诉讼。

最高人民法院认为：“*原告的股东作为项目管理者上传了大量Virtual App初始版本源代码，对涉案软件源代码的形成起到了决定性作用，其他贡献者对Virtual App的著作权是否产生实质影响尚不明确；且即使开源代码的其他贡献者对其贡献部分享有独立的著作权，基于GPL3.0协议，项目管理者与贡献者之间也存在着相互许可的关系。其他用户提交代码并申请并入主分支这一行为表示其默示同意项目管理者作为普通被许可人，可以提起侵权之诉。综上，罗盒公司作为提交了涉案软件绝大部分源代码的项目管理者，其提起本案诉讼无需经过其他贡献者的授权。*”

· **德国法院观点：若贡献者未能证明其具体贡献以及相应的著作权，则不能作为著作权人起诉——McHardy vs.Geniatech案[9]**

Linux是知名的开源操作系统，Patrick McHardy负责维护Linux的子系统Netfilter，该子系统为Linux内核提供与网络相关的操作。Geniatech制造的卫星电视接收器采用了Linux操作系统。Patrick McHardy认为Geniatech侵犯其著作权，并针对Geniatech使用的任何版本的Linux系统寻求禁令。

德国科隆法院高等法院认为：第一，Patrick McHardy只是一个编辑作者（例如做补丁）而非Linux和Netfilter的共同作者；第二，Netfilter早在1999年就存在，比Patrick McHardy首次为Netfilter做出贡献早3年，也比Patrick McHardy加入核心团队早5年；原告无法充分证明其对Linux操作系统的贡献以及该贡献可以构成著作权法上的作品，因此Patrick McHardy无权对整个Linux操作系统主张禁令。另外，在该案中，被告还提交了Patrick McHardy涉嫌滥用诉权的证据。

3. GPL传染性判断依据

如前所述，传染性条款对企业的影响较大。如果落入传染性条款，那么企业需要公开源代码，否则将要面临违约或者侵权风险。因此，传染性的判断对于企业而言尤为重要。近年来，在最高院和各地法院发布的多个案例中，在判断传染性时考虑了以下因素：

· 开源代码与其他代码所处的**层级框架**是否相同、各个层级框架是否适用相同的开源协

议；

- 开源代码与其他代码的**展示方式、所用技术、功能分工**是否存在明显不同（如前端代码和后端代码）；
- 开源代码与其他代码的**隔离技术手段**、所处的文件夹是否相同、是否使用同一**共享地址空间**并连接在一起；
- 开源代码与其他代码之间**通信机制**（exec、pipes、rpc、sockets和命令行参数等）、**通信语义**等。

下文将结合最高院和各地法院的几个典型案例，进行简要分析，以期为企业提供传染性的判断指引。

- **罗盒vs.玩友一案**[10]中，罗盒是Virtual App的权利人，并在GitHub上公开了Virtual App的源代码，适用GPLv3协议。玩友提供的被控侵权软件“微信视频美颜版等”，部分代码与Virtual App相似，却未开源代码。**权利人罗盒认为玩友因使用了Virtual App，其被诉侵权软件整体被传染，但未向用户提供源代码，构成侵权。**

广州知识产权法院认为：*“对于在逻辑上与开源代码有关联性且整体发布的衍生作品，只要其中有一部分适用了GPLv3协议发布，那么整个衍生作品都必须适用GPLv3协议而公开”*，并基于玩友未能举证其使用的开源代码是独立的或在各个独立的层级框架中适用不同的开源授权许可协议，最终认定被诉侵权软件整体应遵循GPLv3开源协议，其未向用户提供源代码的行为违反了GPLv3的规定。GPLv3协议自动终止，玩友构成侵权。

- **不乱买vs.闪亮时尚一案**[11]中，原告不乱买起诉被告闪亮时尚的网站未经许可使用了原告网站后端源代码等构成侵权。对此，**闪亮时尚抗辩**不乱买公司的后端代码受前端代码开源协议的传染，应当遵循GPL协议向所有第三方无偿开源。

最高院认为：针对“后端代码是否受前端代码开源协议的传染”，“.....前端代码与后端代码在展示方式、所用技术、功能分工等上均存在明显不同，不能因前端代码与后端代码之间存在交互配合就认定二者属于一体”，并基于GPLv3第5条的规定进一步认定“GPL协议的‘传染性’.....不包括与其联合的其他独立程序.....后端代码是独立于前端代码的其他程序，并不受GPL协议的约束，无需强制开源”。

- **数字天堂vs.柚子科技一案**[12]中，原告数字天堂起诉被告柚子科技的API Cloud软件侵犯了原告HBuilder开发工具软件中的三个插件。对此，**被告抗辩**原告的HBuilder软件中使用了

受GPL协议约束的软件，HBuilder软件整体被传染，需要公开源代码并向用户授予著作权许可，因此，任何第三方有权在GPL协议授权下使用其代码并构建衍生软件作品，被告不侵权。

北京知识产权法院认为：“对于原告涉案三个插件而言，在其所处文件夹中并无GPL开源协议文件，而HBuilder软件的根目录下亦不存在GPL开源协议文件的情况下，尽管HBuilder软件其他文件夹中包含GPL开源协议文件，但该协议对于涉案三个插件并无拘束力，据此，涉案三个插件并不属于该协议中所指应被开源的衍生产品或修订版本，二被告认为原告软件为开源软件的相关抗辩理由不能成立”。北京市高级人民法院亦认可。

由上述案例可知，GPL的传染性除了可以作为权利人主张构成侵权或违约的理由，也可能成为被诉侵权人主张不构成侵权的抗辩理由。例如，在罗盒vs.玩友一案中，权利人主张被诉侵权软件使用了开源代码，但是未履行开源义务，因而构成侵权；在不乱买vs.闪亮时尚和数字天堂vs.柚子科技两案中，被控侵权人则抗辩权利软件使用了开源协议或包含开源代码，受GPL协议约束，权利软件被传染，权利人具有开源义务，因此被控侵权人使用开源代码不侵犯权利人的著作权。

另外，在网经科技vs.浙江亿邦、苏州启奥^[13]一案中，最高院提及了关于传染性的判断因素“关于涉案软件是否受GPLv2协议约束，该问题涉及底层系统软件是否受GPLv2协议约束、上层功能软件是否构成GPLv2协议项下‘独立且分离的程序’、二者间采用的隔离技术手段、通信方式、通信内容等如何界定以及软件领域对GPLv2协议传导性的通常理解与行业管理等因素”。然而，在该案中，最高院并未对GPLv2的传染性进行认定，而是认为开源系统的衍生软件是否违反GPL协议需另案判断，故本文仅简要介绍并不做深入探讨。

4. 使用GPL开源开发工具编写的软件是否需要受GPL协议约束

在上述案例中，主要是针对在开源软件的源代码基础上进行再创作而得到的衍生作品的传染性判断，例如开源软件本身或其修改版本的源代码构成衍生作品的一部分。除此之外，在实践中还存在使用受GPL协议约束的开源开发工具编写软件的情况。其中，开源开发工具本身的作用仅是开发工具，使用者所使用的也仅仅是其开发功能，而不会直接基于开源开发工具的源代码来获得衍生作品，换句话说，用其编写出的衍生软件作品并不包含开源开发工具的源代码。这种情况下，根据目前的国内司法案例，倾向于认为“使用开源开发工具编写的软件本身不必然地受开源协议的影响”。具体地，

· 天津网城vs.浙江阿凡提^[14]一案中，天津网城拥有权利的ShopNC软件本身系使用开源软件PHP编写的，PHP使用Creative Commons Attribution 3.0协议（以下简称“CC协议”）。该

案的争议焦点在于：1) 根据开源软件PHP编写的软件是否属于开源软件；2) PHP软件手册自带的CC协议是否使得ShopNC软件也应受CC协议传导而开源。

宁波市中级人民法院认为：“本案两被上诉人主张著作权保护的ShopNC电商系统计算机软件由PHP语言编写，但计算机语言本身具有工具属性，以特定计算机语言编写的ShopNC软件作品通过作者创造性的智力劳动所表现的作品独创性与计算机语言之间并未体现以后者为基础的派生关系，故也并非属于PHP语言演绎作品。即使以演绎作品的角度审查，CC协议的上述开源传导性，指向的是原作品的演绎作品，涉案ShopNC软件也并不受软件手册CC协议开源义务的约束。”

三、GPL协议的开源合规建议

结合上文对GPL开源协议的分析，我们提供以下开源合规建议供企业参考，以期为企业的风险排查提供思路。

1. 要求和审核权利人的权利基础

如收到权利人发送的警告函或维权通知，企业应首先要求权利人提供权属证明材料，包括但不限于，权利人的版权证明材料、权利人对代码贡献部分的证明材料等。对于很多外国软件公司，通常会授权给其中国子公司、分销商、知识产权代理机构或律所来处理软件维权事宜，此时企业也应要求其提供权利人的授权材料等，以验证授权链条是否完整。

2. 确认是否适用GPL协议以及具体类型

企业应审查和确认其使用的开源代码所对应的开源协议类型，如GPL协议中的GPL v2、GPL v3协议，亦或是其他类型的协议，如SSPL、AGPL、BSD、MIT等（对于其他类型的协议，我们将在后续文章中详细说明）。此外，企业还应厘清其使用的开源协议和开源代码的对应关系，例如A模块使用GPL v2协议，B模块使用BSD协议等。

值得注意的是，有些开源软件可能包含双重许可或者多重许可。例如，可以同时适用多种开源协议以及商业许可协议，企业可以结合自身业务特点以及发展情况择一适用。例如，一些国际知名的软件公司开发的多个开源软件可以同时适用LGPL v3、GPL v2或GPL v3开源许可证，使用者可以选择其一。

3. 自查是否存在违反GPL许可证要求的事项

如企业在开源代码基础上进行开发，应结合自身具体的应用场景，判断是否触发传染性条款，是否具有开源义务以及应当开放源代码的范围。例如，通常从以下方面进行自查：

- 1) 针对使用场景，审查对于开源代码的使用是否仅限于企业内部，还是会向外部分发；
- 2) 针对分发方式，审查是通过软件副本的方式进行分发，还是通过远程网络交互等方式进行调用（如SaaS）；
- 3) 针对开源软件与企业开发的软件其他部分之间的关系来判断传染范围，例如，可以结合开源代码与企业开发的软件其它部分所处的层级框架、显示方式、功能分工、所处的位置或文件夹、隔离技术手段、通信机制、通信语义等因素进行判断。

此外，GPL协议还规定了权利恢复期，若被控侵权人确实存在违约，如在一定期限内及时纠正，则授权恢复。

4. 与权利人进行商务谈判

企业进行核查后，可根据自身需求，选择跟权利人进行商务谈判，积极沟通，以便尽快解决纠纷，降低对企业商誉的影响。例如，经内部自查后，发现短时间内难以改正所有违反GPL的事项，尤其如果按照传染性条款的规定，企业需要公开后期开发的源代码，可能会导致企业失去竞争优势，也可能导致企业提供的服务或产品受到第三方的恶意篡改和破坏时，企业则可以考虑与权利人进行谈判，以通过购买其商业许可的方式解决潜在纠纷。

5. 内部构建开源合规制度

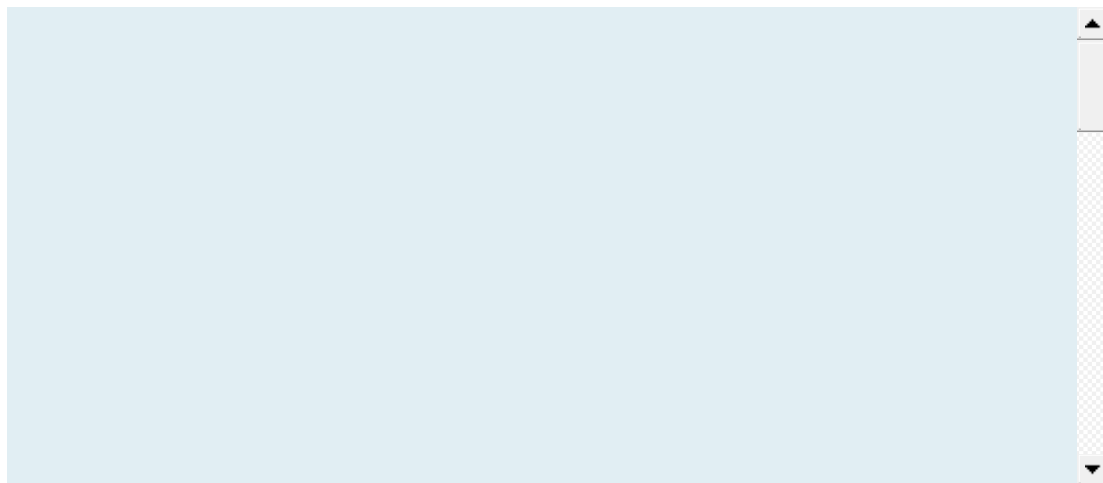
企业应在内部设立开源合规体系，制定开源合规制度，加强开源合规意识，对内部人员进行培训，明确开源软件的使用和限制，制定开源合规和风险应对策略。同时，对于企业较为重要的软件，在开发过程中，应尽量选择使用那些本身对于商业用途更为友好的开源协议，以降低开源合规风险。

结语

随着我国计算机行业的飞速发展，开源技术的重要价值日渐凸显，开源合规也越来越受关注。在众多的开源协议中，GPL开源协议是开源领域被广泛使用的协议之一，也是目前司法案例中涉及最多的开源协议。在本文中，我们针对GPL协议进行了详细解读，梳理了GPL协议的发展历程；介绍了对企业影响最为显著的传染性条款和开源义务；并结合典型案例，分析和总结了司法实践关注的要点，以期帮助企业综合全面的了解使用开源软件的注意要点及相关法律风险，从而合理制定合规策略以及进行合规核查和整改。

后续我们将陆续对其他开源协议进行详细解读和合规分析，例如AGPL、SSPL、LGPL、LLaMA LICENSE AGREEMENT、BigScience RAIL License V1.0等。

*本文首发于知产力



2023-12-27