

第七章 创新社团

现在我们已经清楚，用户经常能够创新，并且经常无偿公开他们的创新。但是，用户之间的非正式合作情况是怎样的？在创新中，有组织的合作情况又是怎样的？答案是，在用户创新中，这两种情况都很常见。用户与用户之间的非正式合作很平常，如帮助其他用户创新；用户与创新社团有组织的创新也很平常。创新团队通常拥有能促进创新速度和效率的工具和基础设施，运用这些工具和设施，用户能够得以开发、监测和扩散他们的创新。

本章将首先阐述用户创新是非常广泛的，所以可以将其有效地组织在一起。随后，我将以软件开发领域非常成功的一种创新社团形式——自由和开源软件项目为例，探讨这种社团所能提供的功能。最后我要指出的是，创新团队并不仅仅局限于软件之类的信息产品的开发中，为了证实这一点，我将以风筝冲浪运动技术设备开发中的用户创新团队加以说明。

用户创新广泛分布

当用户的需求各不相同，而创新者的信息具有粘滞性时，产品开发活动往往广泛分布于各种用户，而不是仅仅由少量的高效的用户创新者进行。同时，不同的用户可能会有不同的创新。正如第五章所展示的，个体用户和企业用户往往倾向于能满足自己特定需求的创新，属于自己的“低成本创新利基”。例如，一个专攻高台跳跃的山地车手，如果同时又是整形外科医生的话，他就可能综合这两类知识进行创新：他可能发明一个座位悬椅以减少跳跃着陆时的震动；而另一个具有不同背景的同样的山地车手——假设具有航空工程方面的知识——就可能综合航空工程方面的知识提出不同的创新。佛莱明（Fleming, 2001）研究了通过已有元素的新组合而进行的创新，发现创新者会利用其在不同社团中的成员资格而将原先无关的元素加以组合。鲍德温和克拉克（Baldwin & Clark, 2003）以及汉克（Henkel, 2004a）则从理论上探讨了这种创新。

这之中的潜在逻辑回应了埃里克·雷蒙德在软件调试的“林纳斯定律”中所提出的观点。在软件业，一些细微的代码错误和“电脑臭虫”的发现修补成本非常高（Brooks, 1979）。但是，雷蒙德认为，同样的任务如果能向大的软件用户社团开放，就可以极大地降低成本，并且更快、更有效，因为每一个用户都可能拥有识别和修补其中一些“臭虫”的信息。因此，雷蒙德认为“假设有一个足够大的程序测试员和合作开发库，那么几乎每一个问题都可以快速地识别，因为对其中的一些人而言，问题的修正方法是轻而易举的。或者，不太正式地说‘有足够多的眼睛，就可以让所有问题浮现’”，“更多的用户可以发现更多的问题，是因为增加用户就增加了发现问题重点的途径……对于“电脑臭虫”特征，每一个用户都有不同的感性认识和分析方法，有不同的看问题角度。所以增加更多的程序测试员，……就增加了某些人的问题分析方法恰好与某些问题相匹配的程度，这样，对**那个人**而言，问题就很简单”，雷蒙德如此解释（Raymond, 1999, pp.41-44）。

这种情况类似于分散的用户创新，每一个用户都有创新的不同需求和创新相关的其他资源，这对**那个用户**而言就可以开发一个特定的低成本创新（问题简

单)。**若干**用户的资源集合就可能经常意味着,有些资源正是解决许多创新开发问题恰巧所需要的(注意,这并不意味着**所有**创新问题都能被用户低成本地完成,甚至并不意味着这些创新问题用户都能完成。事实上,只有当制造商的产品开发规模经济效应大于个体用户整体所拥有的更多创新资源带来的补偿时,用户才会觉得创新比较便宜)。

一些数据支持这些解释。在第二章中,我们看到用户倾向于开发非常不同的创新。为了检验具有商业意义的创新是由少数的还是由大部分用户开发的,我搜寻了一些记录了后来被商业化的重要创新的功能资源的研究。如表7.1所示,在这些研究中,大部分归功于用户的重要创新是由不同的用户完成的。换言之,用户创新是广泛分布的,由具有不同需求、不同粘滞信息的用户完成。

表7.1

用户创新是广泛分布的,很少有用户完成超过一项的重大创新(NA:数据无法获取)

在这些重大创新的开发中用户的数量

	1	2	3	6	NA	样本量(n)
科学仪器	28	0	1	0	1	32
a	20	1	0	1	0	28
科学仪器	19	1	0	0	8	29
b	7	0	0	0	0	7
科学仪器						
b						
工艺设备						
c						
运动器械						
d						

- a. 来源: von Hippel, 1988, 附录: 门电路(GC)、透射电子显微镜(TEM)、核磁共振(NMR)创新。
- b. 来源: Riggs & von Hippel, Esca & AES。
- c. 来源: von Hippel, 1988, 附录: 半导体和拉挤工艺设备创新。
- d. 来源: Shah, 2000, 附录A: 溜冰板、滑雪板、帆板创新。

创新社团

用户创新者可能普遍愿意公开他们的信息,但是,正如我们所了解的,他们是广泛分布的,每个人只有一项或少量的创新。如果他们的信息可以采用一定的方法让他人方便地获取,那么这些用户整体“无偿公开信息的普遍性”的价值就可以提高,这就是“创新社团”的重要功能之一。

我将“创新社团”定义为一个由相互联系的个体和公司组成的有目的的联结,这些个体和公司通过面对面沟通、电子或其他沟通方式等信息转换机制而发生相

互联系。他们可以存在于一个成员群的界限范围内，但并不尽然；他们经常为参与者整合社团的特质——这里“社团”就被定义为是一个有目的的“个体联结的网络，能够提供交往、支持、信息、归属感、社会认同感（Wellman et al. 2002, p.4）”¹，但也不尽然。

创新社团的成员和贡献者可以是用户和/或制造商。当至少有一些创造发明，并且愿意公开，同时有人对公开的信息感兴趣的时候，创新社团就可能得以发展。在前面的章节中，我们发现这种条件在用户创新中普遍存在：用户在许多领域创新，并且经常无偿公开；而这种公开的信息通常被制造商利用以创造具有商业价值的产品——这也是许多用户对这种信息感兴趣的标志。

创新社团通常是限定在特定领域中，通常作为与特定创新小类目相关的信息的集合点和储藏室。它们可能仅仅由信息库和通讯录（以传统的和电子出版物的形式）组成。例如，userinnovation.mit.edu就是这样一个专业网站，研究者可以在这个网站上发布关于用户创新的发现和思想的文字。贡献者和非贡献者都可以无偿地接触和浏览这个网站，非常方便地获取这些信息。

创新社团也可以为参与者提供其他的重要功能。聊天室和公开发布的电子邮件列表使得贡献者可以交流思想、提供互助；还可以提供给社团成员有助于用户开发、评价和整合他们工作的工具——这些工具经常是社团成员自己开发的。

所有上述的社团功能和其他更多的功能都可以在自由开源软件项目的开发社团中找到。这种特别的创新社团也为学术界提供了一些普遍现象，也引起了公众注意，所以我将对它们加以详细描述。首先我将探讨自由开源软件本身的历史和性质，然后概括开发和维护这种自由开源软件开发项目（基于社团的开发）的关键特征。

开源软件

在计算机程序编制的早期，商业“套装”软件很少——如果你需要能满足特殊需求的特殊程序，你一般是自己编写程序或者雇佣他人替你编写程序。20世纪六七十年代的大部分软件都是由高校和公司实验室中的科学家和工程师编写的，在他们的研究文化中，可以无偿提供和交换自己编写的软件，可以建立在他人软件的基础上或对他人软件进行修改，并且可以免费共享修改后的程序。这种共享的行为是“黑客文化”的一个核心特征（在开源程序员社团中，“黑客”是一个褒义词，它代表有才气、乐于奉献的程序员²）。

1969年，美国国防部下属国防先进技术研究计划署（DARPA）成立了ARPANET（先进技术研究计划网），这是第一个贯穿全美的高速计算机网络。这个网络最后连接了数百个大学、国防项目承包商、研究实验室，最后通过互联网获得成功。ARPANET允许黑客大范围地、方便地、低成本地交换软件编码和其他信息，也允许他们传播黑客的行为规范。

共享的黑客文化在一群20世纪六七十年代以麻省理工学院（MIT）人工智能实验室为基地的程序员——软件黑客中非常常见（Levy, 1984），80年代这个群体受到了严重打击，MIT将他们的黑客成员所创作的程序特许给一个商业公司，这个公司基于商业惯例，马上限制了软件的源代码³的使用，这就限制了非公司的个体——包括那些在这个软件开发中起到协助作用的MIT黑客——继续使用它作为未来学习和开发的平台。

杰出的MIT人工智能实验室程序员理查德·斯托曼，因为无法接触到共同开发的源代码而苦恼不已；同时，他也为软件业出现的一个普遍趋势——开发专有

软件和不允许其他人研究和修改的软件——而感到不愉快。斯托曼认为这些现象在道义上是不合理的，侵犯了软件使用者自由学习和发明的权利。1985年，他创建了自由软件基金会（FSF），着手研究和传播能保护自由接触软件黑客开发的所有软件的合法机制。斯托曼这种开创性的思想利用了现有的版权法中的有关规定，软件作者如果愿意让他们的软件成为“自由”软件，可以利用他们自身的版权来签署许可，确保给未来的使用者许多权利。他们可以很方便地通过签署一个标准的软件许可来让与这些权利。斯托曼开发了用来实施这种开创性思想的许可证——通用公共许可证（GPL，有时被称为“版责（copyleft）¹”，与版权（copyright）相对）。这些自由软件的基本权利包括可以无偿使用、研究和修改源代码的权利，以及无偿向其他用户公开修改或未经修改的版本的权力。其他人也开发了让与类似权利的许可证，现在在自由软件领域有许多这样的证书。自由开源软件许可并不如早先所定义的无偿公开那样让与用户完整的权利，用户使用具有类似GPL这样的许可证的软件时禁止了一些特定的行为，例如，用户不能在它们用于出售的专有软件中包含GPL软件的内容⁴。事实上，开源软件源代码的贡献者们非常关心这种禁令的执行以确保所有用户仍然可以接触到他们的源码（O'Mahony, 2003）。

自由软件的观点并没有很快成为主流，产业界对此尤其抱怀疑态度。1998年，布鲁斯·潘伦斯和埃里克·雷蒙德认为在斯托曼的“自由”软件这个术语中存在着一个重大问题，这可能在商人们耳边敲响了不祥的警钟。因此，他们和其他杰出的黑客们一起展开了开源软件运动（Perens, 1999）。开源软件运用了自由软件运动所倡导的许可证形式，但其主要理念不同于自由软件运动，更强调它的许可实践的实际意义，而不是确保用户有运用自由软件和开源软件的自由的道德意义。在实业界和学术界，“开源”这个术语现在普遍用来表示自由和开源软件，这也是我在本书中所采用的术语。

开源软件已经成为一种重要的文化和经济现象，开源软件项目的数量在急剧增加。2004年，Sourceforge.net⁵——一个开源软件项目的主要基础设施供应者和知识库——拥有83000个项目以及超过870000的注册用户；许多由商业公司开发的软件在开源许可证的条件下发布。

开源软件开发项目

软件能被冠以“开源”无关于它是如何被开发的以及是由谁开发的，这个术语指的是表明应用这个软件的许可类型。然而，开源软件可以让所有人免费获取的事实，已经促进了一些典型开源软件项目开发的实践，它与商业软件开发模式有极大的不同——看起来非常类似于前面提及的“黑客文化”的行为。

商业软件供应商通常希望出售他们开发的程序，所以他们严格限制公司的员工和合约商利用他们软件产品的源码。这种限制的结果是，只有少数内部人拥有修正和改善这个专有程序的信息（Meyer & Lopez, 1995; Young, Smith & Grimm, 1996; Conner & Prahalad, 1996）。相反，所有人都可以免费接触由作者发布的开源软件的源代码。在早期的黑客时代，这种学习、使用和修改软件的自由是通过信息共享和程序的合作开发来实现的——通常是刻录了程序的计算机磁带和磁盘的实物共享和交换；在如今的互联网时代，计算机硬件、软件和网络技术的快速发展使得共享开发以更大规模更快速度创建和维持，同时，由于更易懂的项目设计和通过网络获得的预包装的基础支持，新项目的实施变得日益简单。

¹ 也译作版权属左、反版权，译者注。

今天，开源软件开发项目往往由寻求能满足自身需要的个体或小群体首先发起。雷蒙德（Raymond, 1999, p.32）认为“每一个好的软件工作都始于搔到了开发者的痒处”，“太多的软件开发者将他们的时光消磨于那些他们即不需要也不喜欢的项目中，但是在这个世界（开源世界）中不……”。一个项目的创始人往往成为项目的“所有者”和“养护工”，负担起项目管理的责任⁶。在早期，这些个体和群体往往开发一个能反映其期望功能的粗糙的初始程序版本，所有人都可以通过从项目的网站上下载而免费获得这种初始版本的源代码。项目创始人同时提供相应的基础设施，以便有兴趣使用和进一步开发程序的人可以用来寻求帮助、提供信息或提供新的开源代码供他人探讨和检验。对于那些能成功引起他人兴趣的项目，其他人会下载、使用、探讨这个程序——其中一些人会继续创作新的或修正的程序，之中大部分然后会在项目网站上发布他们所做的一切，以便感兴趣的人使用和鉴定。被项目创始人认为具有卓越质量和普遍意义的新程序和改良程序会被添加到程序正版中。在许多项目中，对程序正版的添加只限于少数值得信赖的开发者，这些少数人随后成为了拥有程序贡献者所没有的访问权的编程看门人（von Krogh & Spaeth, 2002）。

开源软件项目参与者能获得的关键工具和设施包括向所有人开放的特定目标的电子邮件列表。这之中，有程序使用者报告在使用软件过程遇到的软件不足之处（“电脑臭虫”）的邮件列表；也有程序开发者可以分享项目的下一步骤、需要添加的好品质等想法的邮件列表等。所有这些邮件列表向所有人开放，并且公开保存，这样任何人都可以再次访问，了解关于某一主题的去和现在的建议和意见。同时，开源软件项目的程序贡献者往往共同拥有一些必要的工具，如专业的软件语言；它们往往不是某个项目专用的，而是在网站上可以获得。所有贡献者共享的基本工具箱可以极大地方便相互交流。而且，开源软件项目拥有版本控制的软件，这样贡献者可以将提交的新程序插入现有的项目程序中，检测新的程序是否会引出现有程序的故障；如果是，这个工具可以帮助轻松回复到原有状态。这使得人们有更多的实践机会进行“试试看”的检测，使得新贡献程序无意中破坏程序的风险减少。开源软件项目的工具箱随着实践而发展，并且不断地被用户创新者所改善，现在个体项目能运用如Sourceforge.net等网站提供的标准设施开始工作。

下面两个简短的案例可以帮助我们进一步了解开源软件项目的特色。

阿帕奇网页服务器软件

阿帕奇网页服务器软件可以用于那些拥有网页并为网络浏览者提供所要求信息的网页服务器计算机。这类计算机是基于互联网的全球网基础设施的关键元素。

最后演化成阿帕奇软件的网页服务器软件是伊利诺大学（University of Illinois）大学生罗布·迈科库（Rob McCool）为伊利诺大学国家超级运算应用中心（NCSA）工作时所开发的。迈科库开发以及定期修正的源码被发布在网上，这样其他网站上的用户可以下载、使用、修改以及进一步开发。1994年中当迈科库离开NCSA时，一小群已经在自己的网站中采用了他的网页服务器软件的网主决定承担起继续开发的任务。核心的八个用户收集了所有的文档和修正缺陷的方案，并发布了一个补丁。随着时间过去，这个“补丁”网页服务器软件演化成了阿帕奇。大量的用户反馈和修正导致了阿帕奇1.0的产生，阿帕奇1.0发布于1995年12月1日。

四年中，经过许多用户的不断修正和改善，阿帕奇已经成为互联网上最受欢迎的网页服务器软件，并因其杰出获得了许多产业奖项。尽管受到了来自微软、网景等商业软件开发商的强大竞争，它现在仍被全球数百万网站中的60%以上所使用。用户和其他人对阿帕奇的修正和更新仍在继续，而其新版本的发布则由核心的22位志愿者加以协调。

Fetchmail——互联网电子邮件应用程序

Fetchmail是用来从中央服务器“收取(fetch)”邮件到本地计算机的一种互联网电子邮件应用程序。这个开源项目的开发、维护和改善是由埃里克·雷蒙德来领导的(1999)。

1993年雷蒙德开始思考邮件收发中存在的问题，因为他不满意现有的问题解决方法。他回忆道(Raymond, 1999, p.31)，“我所要的，是能够将我的邮件传到我家的系统snark上，这样有邮件时我就能注意到，然后能用我本地的工具处理它”。雷蒙德决定尝试开发一个更好的解决方案。他首先在开源的数据库中寻找已有的、编程良好的应用程序以作为其开发的基础。他知道他可以在他人可能的相关工作的基础上进行开发，在开源软件(那时普遍称为自由软件)世界中，这种方法是被理解和重视的。雷蒙德研究了许多备选的开源程序，选定了应用很少的“popclient”。雷蒙德提出了对这个程序的许多改善，并且将建议寄给了当时的popclient维护员，但popclient维护员已经对继续开发这个程序失去了兴趣，所以他的回复是把popclient传给雷蒙德，这样雷蒙德可以进一步开发popclient。

雷蒙德接受了成为popclient维护员的职责，在随后的几个月中，他接受了其他用户的意见和建议，极大地改进了这个程序。他经常通过专门设立的电子公告板上的信息与popclient用户沟通，精心地发展活跃popclient用户程序员列表。许多用户回复以他们发现的问题的信息以及可能的解决办法、提供他们为自己使用而作的改进。这些意见和建议的质量通常是非常高的，因为“这些贡献不是来自于随机样本，而是来自于那些对使用这个软件、了解其工作原理、试图解决所遇到的问题并且提出看来合理的解决办法感兴趣的用户。任何一个有过这种经历的用户很可能可以提供一些有用的贡献(同上, p.42)。”

最后，雷蒙德实现了一个创新的设计，他知道这个软件能良好运行，因为他和他的合作开发者每天使用、检测和完善它。Popclient(现在更名为fetchmail)成为了数百万用户使用的标准软件。雷蒙德仍然领导一群志愿者维护这个软件并根据新的用户需求和条件完善这个软件。

创新社团物质产品的开发

用户创新社团不仅仅局限于类似软件这样的信息产品，他们在物质产品的开发中也很活跃，并且方法很相似。正如创新社团为信息产品贡献的例子，社团贡献给物质产品的包括从简单的信息交换场地到提供工具和基础设施装备良好的场所。法兰克和沙对运动产品的研究描述了相对简单的社团结构。他们研究的滑雪社团由一些来自世界各地的半专业运动员组成，这些运动员每年大约有十次机会在欧洲、北美和日本的比赛中相见。根据法兰克和沙的研究，社团成员相互之间非常了解，并且花了相当多的时间在一起。他们互相帮助，开发和改良他们的运动设备，但是这个社团没有配备用于创新的专门工具。

更多复杂的物质产品创新社团在工具和基础设施方面类似于开源软件开发社团。我们以最近成立的致力于开发和传播新型风筝冲浪设备信息的社团为例，风筝冲浪是一种水上运动项目，使用者站在一块有点像冲浪板的特制滑板上，由一只巨型的可操纵方向的风筝牵引。有关的设备和技术已经发展成熟，有经验的风筝冲浪运动员在迎风和逆风情况下都可以操纵风筝，每次冲浪者和滑板能腾空数十秒。

为风筝冲浪运动设计风筝是一个非常复杂的任务，涉及低速空气动力这个至今人们并不熟悉的领域。早期风筝冲浪用的风筝都是由用户爱好者开发和制造的，他们相互帮助发明了风筝冲浪技术和风筝冲浪设备。大约在2001年，长期对风筝冲浪和冲浪用风筝开发感兴趣的MIT博士生索尔·格里菲思，认为良好的在线社团互动有助于风筝冲浪的发展，因此，他建立了一个全球风筝冲浪运动用户创新者社团的网站（www.zeroprestige.com）。格里菲思首先在网站上公布了一些他设计的风筝的款式，提供了一些关于风筝制造和使用的有心得和工具；他邀请其他人免费下载这些信息，如果愿意也可以贡献他们自己的信息。不久，其他创新者开始发布他们自己的风筝设计、为新手提供的建议，以及一些精良的设计工具，如空气动力建模软件和快速原型设计软件等；其中至少有一位是受聘于航空公司的空气动力学家。

注意，物质产品在设计阶段是信息产品。早先，设计的信息反映在一大张被称为技术图纸的纸上，可以复印和共享；同单位的设计者可以了解和评价技术图纸上的信息，机械师则用这些信息来制造信息所代表的物质产品。今天，新产品的的设计一般被编辑成计算机辅助设计（CAD）文档，这些文档可以被设计者设计成二维的或三维的透视图；其中所包含的设计可以用不同的工程技术工具进行自动评价，用以确定如设计的产品能否承受它们可能受到的压力之类的问题；CAD文档可以随后被下载到计算机控制的生产设备，这些设备就可以制造出设计所包含的零部件。

风筝冲浪社团共享设计信息的例子可以说明信息产品和物质产品之间的密切联系。最初，社团的用户通过在互联网上传递简单的草图来交换设计思想。后来社团成员发现制帆工厂用来从大匹面料上裁剪船帆的计算机控制裁剪机可以用来裁剪冲浪用风筝的面料；他们也了解到制帆工厂对他们的业务感兴趣。因此，创新社团的成员开始以CAD文档传递适合于制帆工厂裁剪设备的设计。如果用户满意某个设计，他可以将CAD文档传递给当地的制帆工厂以便裁剪，然后用户将他们缝合在一起或者委托缝纫厂缝合。这种方式下，从信息产品变成物质产品只需要不到一个星期的时间，一个完工的风筝总成本不过区区数百美元，远远低于市场上风筝的价格。

用户对用户的帮助

显然，用户创新社团可以通过提供工具的形式为个体创新者提供有效的支持，这些创新社团中的用户也倾向于以协作的方式行动。即，用户不仅仅传播和评价已经完成的创新，他们也是其他重要服务的志愿者，如在开发和推广创新中互相帮助。

弗兰克和沙（Franke & Shah, 2003）研究了四个运动社团中用户互相帮助创新的频率，发现这种帮助非常普遍（表7.2）。他们也发现，那些提供帮助的人更可能本身也是创新者（表7.3）。被帮助的人的满意度非常高，其中的79%认同“如

果我遇到相似的问题，我还会询问同一批人”。杰普森（Jeppesen, 2005）也有类似的发现：在计算机游戏领域有大量的自愿的用户对用户的帮助。

表7.2
为创新者提供帮助的人数

提供帮助的人数	案例数	比例（%）
0	0	0
1	3	6
2	14	26
3-5	25	47
6-10	8	15
>10	3	6
合计	53	100

资料来源：Frank & Shah, 2003, 表4。

7.3
创新者帮助他人创新的倾向 ($p < 0.0001$)

	创新者	非创新者	合计
提供过帮助	28	13	41
	32	115	147
没有提供过帮助	60	128	
合计			

资料来源：Frank & Shah, 2003, 表7。

这种帮助现象是创新社团对社团成员的价值的重要体现。为什么人们愿意提供帮助是一个值得分析的课题。答案现在并不完善，但随着调查研究的进一步进行，神秘将会减少。现在已有的一个答案是这对提供帮助者有私人利益，对无偿公布创新者也是如此（Lakhani & von Hippel, 2003）。换言之，无偿帮助的出现可能可以用以前讨论过的创新动机的私人-集体模型来解释。

