



企业内部开源&社交化编程 ——连接&聚合

林鼎军（邮箱：lindingjun@126.com）

程序员的工作环境变迁

哪个更好？这就是我今天和大家在接下来的时间里将要探讨的问题。

- 图书馆：借书
- 回家：网上查资料，发公司
- MSDN；
-

- 网络随使用

- 网络进行proxy优选，最优秀的开发网站供搜索、借鉴；

- 网络进行proxy优选，最优秀的开发网站供搜索、借鉴；
- 内部建立统一的知识社区，分享、汇集各种知识；
- 内部建立割裂的/统一的开发平台

两个小调查

1、在座各位中，有多少人前公司或现公司探索过引进开源的开发模式的？



2、在座各位，有多少人认识全以下这三位前辈的？



Topic

- ◆ 开源与商业闭源的战争
- ◆ 企业的内部开源探索
- ◆ 云时代的软件开发雏形
- ◆ 从开源看软件开发的本质
- ◆ 企业应如何连接&聚合
- ◆ 总结

馒头纷争：代码控制权争夺战

软件产业出现之前

出于教学与学术目的，交流各种源代码

- 大型机、中型机、小型机的时代，大多数人都买不起计算机；
- 有能力购买计算机的机构、大学，都聚集了大量的科研人员；
- 代码的交流，就像学术交流一样，毫不考虑挣钱的事情；

黑客伦理

- 对计算机的访问（以及任何能帮助你认识这个世界的事物）应该是不受限制的、完全的。任何人都有动手尝试的权利；
- 所有的信息都应该可以自由获取；
- 不迷信权威——促进分权；
- 评判黑客的标准应该是他们的技术，而不是那些没有实际用途的指标，比如学位、年龄、种族或职位；
- 你可以在计算机上创造艺术与美；
- 计算机可以让你的生活更美好，就像阿拉神灯，你可以让它听从你的召唤。



ENIAC 1946



DEC PDP-7 1965

商业“魔头”现身

出于商业目的，封闭代码，出售拷贝

- 微型计算机、个人电脑开始兴起，普通玩家之间，也会无偿的交换盗版拷贝；
- 1976年，比尔盖茨发表著名的《写给电脑爱好者的公开信》，倡导版权与利益。而且愤怒的将那些免费复制软件的家伙，称之为：窃贼！
 - 有谁会没有任何报酬的情况下来做这些专业的工作？什么样的爱好者可以为他的产品投入三个人年的开发时间，并且发现所有的错误、编写文档以及免费发布这个产品？
- 软件行业兴起！



IBM 5100 Portable
Computer 1975



Apple I 1976



An Open Letter to
Hobbyists

黑客的愤怒：GNU宣言及其计划

○ UNIX纷纷闭源，商业公司纷纷成立

- Novell owns AT&T's Unix: IBM-AIX、SGI-IRIX、SCO UNIX、SUN-Solaris
- Berkley (BSD) : SunOS、Ultrix、NetBSD、DEC-OSF、NeXTSTEP、Mac OS X
- 这些UNIX，收费昂贵、互不兼容、而且闭源，令人抓狂，令黑客们愤怒；

○ GNU/Linux的崛起

- Richard Stallman在1985年发表了GNU宣言，并于1989年起草了GPL，提出了Copyleft的概念。Emacs、GCC等工具纷纷问世；
 - GNU: GNU's Not Unix
- 1990年，Linus Torvalds在芬兰赫尔辛基大学读书期间，开始开发了Linux，1991年发布最初版本，并飞速发展至今，从此，GNU计划完成。



实践到理论，奠定成功的基础

- **Eric Steven Raymond: 《大教堂与集市》**
 - 1997年5月27日发表首次公开发表
 - 1999年由O'Reilly出版
 - 2014年，中文版被翻译引进中国（卫剑钊）
 - 在开源社区，被誉为“圣经”级的文献
- 《大教堂与集市》是开源运动的“毛泽东思想”，颠覆了传统的软件开发思路，影响了整个软件开发领域。作者Eric S. Raymond是开源运动的旗手、黑客文化第一理论家，他讲述了开源运动中惊心动魄的故事，提出了大量充满智慧的观念和经过检验的知识，给所有软件开发人员带来启迪。



大教堂与集市——五部曲

- 《黑客圈简史》——简要介绍了黑客社区的缘起与发展状况；
- 《大教堂与集市》——探讨Linux的成功之道，并以自己发起的一个项目做实验，总结相当多的宝贵经验
 - 如果你有正确的态度，有趣的事情自然会找到你；
 - 把你的用户当成开发合作者对待，如果想让代码质量快速提升并有效排错，这是最省心的途径；
 - 如果有足够多的β测试者和合作开发者，几乎所有的问题都会很快显现，然后自然有人会把它解决；
 - 仅次于拥有好主意的是，识别来自用户的好主意，有时候后者会更好；
 - 如果开发者协调者有至少一个像Internet这样好的沟通媒介，并且知道如何不靠强制来领导，那么多人合作必然强于单兵作战；
- 《开垦心智层》——开源社区的风俗习惯背后的法理分析与文化分析，尤其是对礼物文化的分析，非常精彩
 - 命令体系、交换经济、礼物文化
 - 充裕性会使命令关系难以维持，会使文化关系变成无意义的游戏。在礼物文化中，社会地位并不取决于你控制了什么，而是你给予了什么
- 《魔法锅》——开源项目的经济基础与博弈分析，并进而指出在何种条件下，闭源应该走向开源；
- 《黑客的反击》——介绍了受到《大教堂与集市》的激励，1998年，网景公司最终决定开源的故事，以及Open Source组织的成立经过



集市逆转案例：Linux不可思议的发展历程

○ Linux 1991年发布第一个开源版本

- 通过**互联网**聚集了大量的志愿者，没有严格的质量标准，没有强有力的机构协调管理；
- 最简策略：每周发布，然后接受反馈
- 到1993年底，Linux在稳定性与可靠性上，已经与很多商业UNIX不相上下，并能支持比商业UNIX多得多的软件
- 大多数小型UNIX供应商倒闭

○ Linux vs. Windows的市场表现

	Linux	Windows
桌面与笔记本	1.74%	>88%
服务器	38.6%	32.6%
超级计算机	96.4%	<0.4%



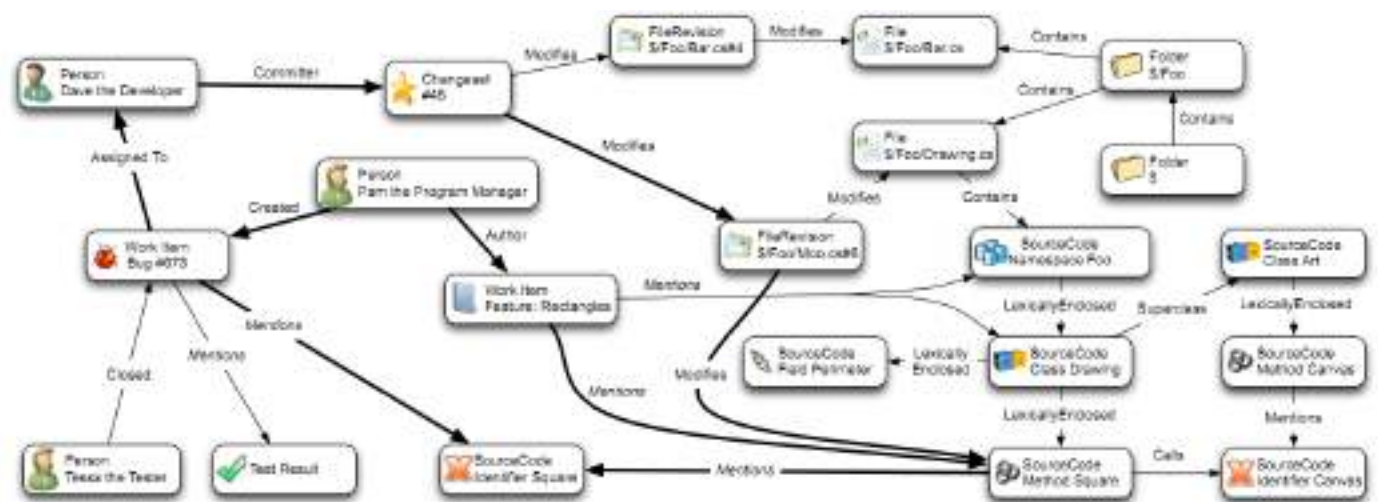
微软的探索

- 最早明确研究将社交网络应用于提升编程效率的是微软，2005年微软研究院1名教授带着1名博士生开始了Codebook项目的研究；
 - 成果一 体现为两篇论文：《Codebook: Social Networking over Code》（2009）、《Codebook: Discovering and Exploiting Relationships in Software Repositories》（ICSE 2010）
 - 成果二 开发了两个平台：微软内部社交编程平台Codebook及外部社交编程平台CodePlex（2005）

1. Introduction

Software development is highly social. A software engineer's daily work requires coordination with many other people and tracking many shared artifacts. Today, this coordination and tracking is largely accomplished through frequent communication and monitoring of artifacts: attending meetings; chatting in the hallway; reading code revision messages; examining work item statuses; exchanging emails; and so on. These coordination activities both take time away from technical work and are subject to the usual communication breakdowns, like untimely, inaccurate, misdirected or missing information.

Recently, social networking web services, like LinkedIn, MySpace, and Facebook, have provided a new way for people to coordinate and keep track of each other's activities. A Facebook user, for example, declares other users as "friends" and (after their approval) sees a "newsfeed" of their Facebook activities, like posting messages and photos.



More...



- IBM内部的开源社区称为Community Source;
- 有32000名注册用户, 11000名项目参与者, 1700多个项目;
- 采用SVN或CVS配置管理工具;
- 适用于平台、组件、内部工具、可共享才有开源的意义;
- 第一代基于Gforge开发, 第二代Community Source, 第三代JazzHub+Bluemix, 内置RTC项目管理工具, 和RAM资产管理系统。未来支持Git;
- 定位: 可重用资产的协同开发平台;
- 闭源仍然占IBM项目的绝大多数, 90%以上;
- 理念: “大家都用的东西应该交给社区, 而不是公司。”



- 2001年即开始在内部引进开源实践, 统称为Inner Source;
 - 最多人员参与的项目是平台、组件类项目
 - 最多下载的项目是框架、脚本和库之类的项目
- 基于开源软件Sourceforge建立称为iSourced的Portal。诺西的NSN成立后运行两个实例
 - 到07年有400多个项目
 - 数千名参与者
 - 95%项目少于25名开发者
 - 采用敏捷开发方法
 - 采用SVN或CVS配置管理工具
 - 采用CruiseControl或Bamboo持续集成工具
- 目标: 促进创新, 避免重复造轮子, 提升个体开发人员的软件共享意识。




- 称为Development Community;
- 有27000多名注册用户, 1000多个项目;
- 采用SVN配置管理工具
- 适用于DoD内部可复用软件和服务的协同开发和部署;
- 基于CollabNet公司的工具开发;
- 定位: 是DoD内部加速开发和部署可复用软件和服务的协作环境;
- 分别支持public和private项目;
- 支持“社交化编程”, 最早将互联网的协作与透明化融入软件的组织;
- 理念: “用互联网变革DoD的软件开发和部署”。
- 是DoD始于1992年SRI计划的最新支撑平台。

Google Code & Developers



- Google Code是Google公司利用自身服务器资源提供的开发人员主页。其最初目的是为Google自身的开源软件提供开发平台，以及协助开发者扩展Google产品的功能。随着Google开源之夏等项目的推进，Google Code正在成为一个开放的项目托管平台，类似Sourceforge提供版本控制、问题跟踪、Wiki、下载托管等工具。截至2009年3月，已有超过50,000个项目以Google Code为家。

Google code Search
search code.google.com

 **Google Developers**

Looking for Google APIs and Tools?

Google Developers is now the place to find all Google developer documentation, resources, events, and products.

developers.google.com

Project Hosting
Project Hosting on Google Code provides a free collaborative development environment for open source projects.

Explore project labels:

Python	JavaScript	Django	Web	Google
Java	AJAX	Rails	Plugin	Android
CPlusPlus	MySQL	DotNet	Game	AppEngine
PHP	Flash	jQuery	Database	GWT

Project Hosting on Google Code will close on January 25th, 2016. [More info...](#)

©Google - [Google Developers](#) - [Terms of Service](#) - [Privacy Policy](#)

2015年1月底关闭

More.....



阿里的taocode



小米MiCode



豆瓣CODE

CODE 的第一行代码开始于
2012 年 2 月 14 日，在豆瓣
内部使用已有 2 年，

社区新锐：Github



- Github成立于2008年，有超过800万注册用户，托管超过1000万个项目；
- GitHub 已经发展到 200 多个雇员，分布在全美 30 个州和海外 14 个国家。70% 的雇员是分布在世界各地的，他们都在家里或者咖啡馆里办公。
- 在工作沟通中，所有员工都是使用一种叫做 **campfire** 的网络聊天室进行协同工作，这样的 campfire 总共有 80 个，每个员工都可以加入多个 campfire 去进行不同的项目。campfire是GitHub员工的虚拟办公室。
- **GitHub最初的理念是**：Social Coding，后改为：Build software better, together.
- **2013年，美国教研机构在美国国家科学基金的支持下进行了关于GitHub社交化编程的研究**，其论文有《Social Coding in GitHub: Transparency and Collaboration in an Open Software Repository》、《Using developer activity data to enhance awareness during collocative software development》等，这些研究从各个方面论述了社交化编程给软件开发带来的变化。
- **GitHub在国内模仿者众多**：Gitcafe（被Coding.net收购）、Git@OSChina、CSDN CODE、Coding.net



For everything you build

Host and manage your code on GitHub. You can keep your work private or share it with the world.



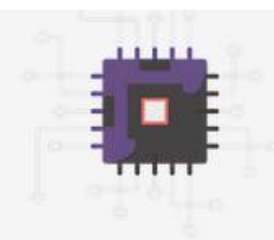
A better way to work

From hobbyists to professionals, GitHub helps developers simplify the way they build software.



Millions of projects

GitHub is home to millions of open source projects. Try one out or get inspired to create your own.



One platform, from start to finish

With hundreds of integrations, GitHub is flexible enough to be at the center of your development process.

在GitHub，程序员们怎么玩的？



○ 开源项目

- 几乎全世界最著名的项目，都在GitHub上设立镜像，或者直接在GitHub上开发，例如去年Facebook在GitHub上创建了新的开源组织TODO (talk openly, develop openly)，Dropbox、GitHub、Google、Khan Academy、Stripe、Square、Twitter和Walmart Labs等纷纷加入；

○ 玩游戏

- 2048，程序员们都玩疯了

○ 写作

- 数学家、法律学家、专利律师甚至音乐家，开始在GitHub上协同写作

○ 写博客

- GitHub Pages的出现，为程序员们提供了一个非常Geek的博客托管平台

○ 人才库

- 当GitHub汇聚了越来越多的程序员，而这些程序员在GitHub日夜不停地开发着各种不同的开源项目，一个全球最大的编程人才库，就此形成了；

○ 全球软件创新中心、软件基因库

- 当所有双创软件都在GitHub上设立镜像，或者直接在GitHub上开发，加上已经入住的1000多万项目，一个全球软件创新中心、软件基因库形成；

GitHub能这样的玩是因为：社交化编程



- **社交化编程**是一种新型软件开发方法，它将社交理念、社交网络与软件开发过程紧密结合，使软件开发人员聚焦于项目本身，紧紧围绕项目的基本元素，如概念、需求、算法、架构等，进行沟通交流，并同步或异步协作，共同构建成功的项目以及活跃的虚拟社区；

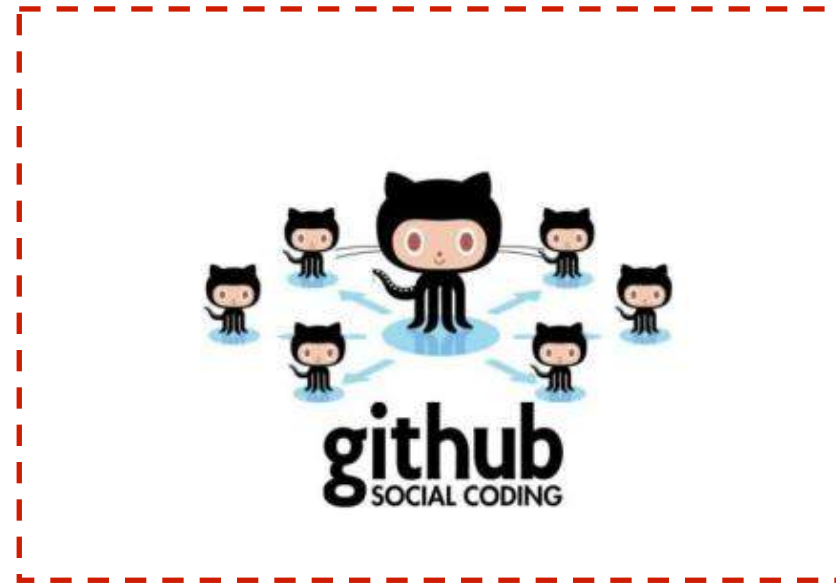
软件开发过程/活动



社交理念及社交网络



社交化编程



软件开发实际上是高质量的社交活动

- 社交——是社会上人与人的交际往来，是人们运用一定的方式（工具）传递信息、交流思想，以达到某种目的的社会活动。
- 软件开发其实就是以开发成功一款有用的软件的社交活动。它的社交包括现实的社交，如面对面的沟通，还包括通过多种社交网络进行的虚拟社交。一般来说，后者更重要，因为可追溯、可度量、可积累成经验库等；
- 因此，社交网络适用于软件开发活动。

email	1971	Blogger、LiveJournal	1999	Twitter、Spotify	2006
www	1991	Wikipedia	2000	Tumblr	2007
个人站点 (Justin's Links from the Underground)	1994	Meetup.com (线下兴趣交友)	2001	Groupon	2008
Classmates.com	1995	Friendster (个人主页交友)	2002	Foursquare	2009
Ask.com (搜索引擎)	1996	MySpace、WordPress	2003	Google Buzz	2010
AIM、Weblog	1997	Facebook、Flickr	2004	Google+	2011
Open Diary (公开/私密, 评论)	1998	YouTube	2005	Pinterest	2012

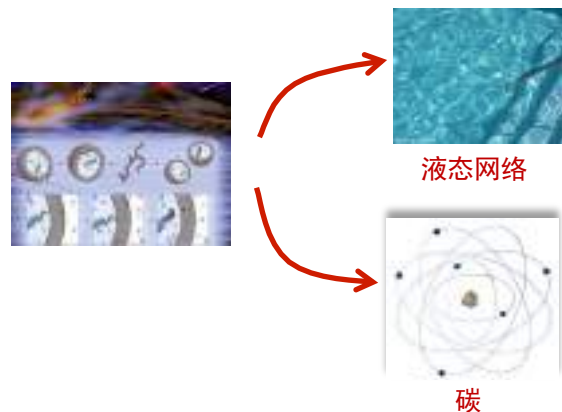
隐含的一个结论：软件开发人员既是知识工作者，又是社会人。

软件开发应该处于一锅“原始汤”中

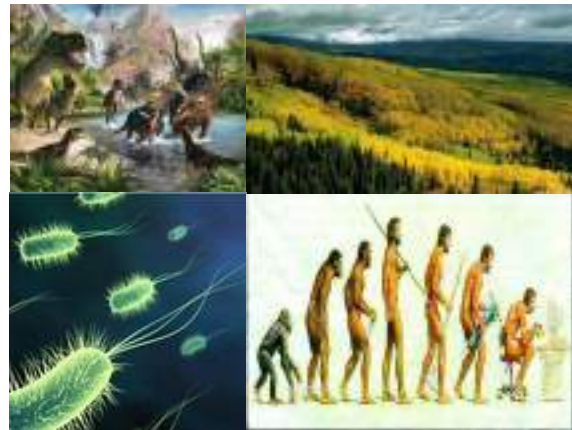
原始汤 (primordial soup)

1871年，达尔文在一封信里这样写道：“生命最早很可能在一个热的小的池子里面。”这就是后来著名的“原始汤”；

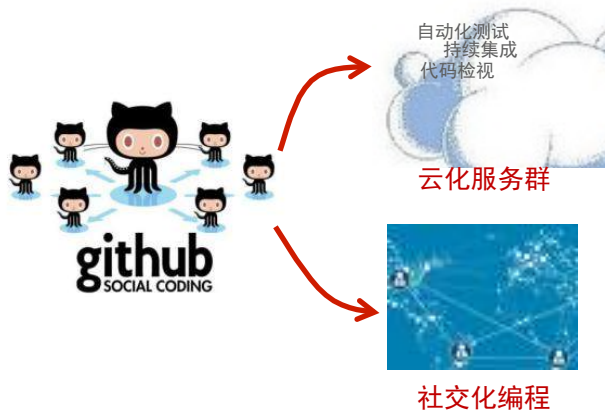
1977年，美国科学家在加拉巴戈斯群岛找到了它。



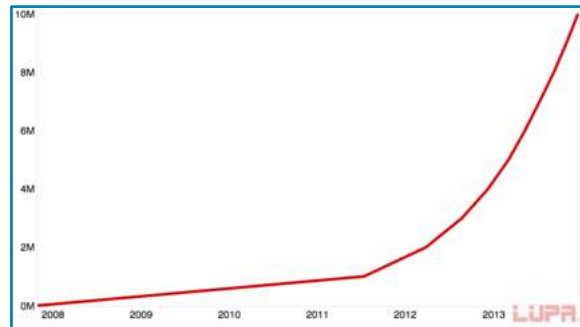
进化



Github——全球最大的社交化编程社区



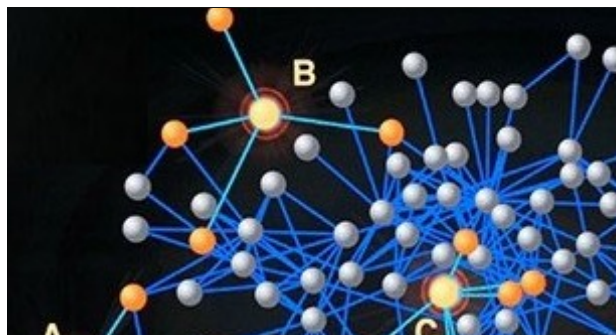
进化



移动 云计 书籍 音乐 图片
互联 算 创作 创作 协作 ...

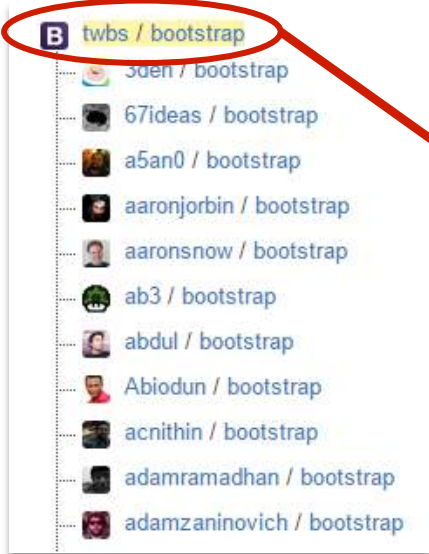
软件开发本质是——连接&聚合

- 人与人关系路径满足六度分隔理论，越短越好。



- 人与人、人与项目、项目与项目应该构建起全方位连接；并通过最佳的社交连接方式，产生最大的价值；

- 构筑起全部研发人员的“兴趣图谱”。



Popular repositories

riotjs-todomvc	TodoMVC for Riot JS	36 ★
dot_hash	Access hash properties with the dot notation	8 ★
todomvc	Helping you select an MV* framework - Todo ...	5 ★
vimfiles	my vim files	4 ★
ruby-sanitizer	The simplest string cleaner ever made	2 ★

Repositories contributed to

yahoo/atomizer	A tool for creating Atomic CSS, a collection of..
yahoo/fluxible-immutable-utils	A mixin that provides a convenient interface f...
yahoo/jsx-test	An easy way to test your React Components (...)
salesforce-ux/design-system	Salesforce Lightning Design System
synack/docker-rsync	Docker + rsync

Marcelo Eden
3den

Yahoo!
 Sunnyvale
 eden@3den.org
 http://www.linkedin.com/in/marcelo...

Joined on 28 Mar 2008

23 Following
 88 Starred
 13 Following

一些需要做的基础研究

○ 多维度研究整个软件业的趋势

- 研究清楚传统开源社区的运营规则，比如Apache、Eclipse等；
- 从软件行业的整体发展历程上去研究及掌握当前的软件发展阶段及趋势；
- 从工具或基础设施的角度去研究清楚Apache工具套、SourceForge工具套、Github工具套等；

○ 定义及应用社交化编程的多种社交形态

- 私密社交化编程——用于核心机密项目、算法、新技术等开发；
- 熟人社教化编程——用于“项目团队+上下游项目团队”组成的项目群，使用社交化编程基础设施（如Github企业版）+ 传统开源社区的运营规则 + 传统闭源项目的计划、度量、质量控制，融合而成的企业核心开发平台，它应该是企业内部未来开发生态的内核；
- 陌生人社教化编程——用于所有通过组件、民间创新项目、外部开源优秀项目跟踪镜像及内部研究、应用或参与形成的社区，甚至任何需要协作的项目，比如文档开发、文章写作、设计等，从而建立起全司研发人员的“兴趣图谱”，更好地利用开发人员的碎片时间；

社交化编程 vs. 传统开发

	社交化编程	传统开发
规划	由社区在民主的基础上进行决策和管理，精英管理模式。	由MKT进行规划，或由技术委员会进行规划。
开发	小型开发团队，敏捷、迭代式开发，频繁发布，快速发布	基于承诺，按项目方式进行开发，范围、时间、预算等。
人力	没有固定的人力，计划不确定，滚动计划。	专门预算人力，专款专用，有严格计划。
工具	云化+社交化的项目托管平台，由Wiki、Maillist、缺陷跟踪、讨论、评论、拉分支等基本功能	封闭式的单兵作战重型工具、重型文档
维护	不承担维护责任，由客户自行维护，可能提供基本的咨询或培训。	专业的维护团队，公司定义的维护职责。
质量	同行评审，众测、专家检视等，基于人的能力多一点	严格的质量管控，基于过程的能力多一点

不断满足程序员最根本的诉求



随时,随地
编码或定位问题



梳理项目形态发生的变化

云上的“社交化编程平台”

企业内部

开源界



总结

- 内部开源——给软件人员精心炮制的“原始汤”；
- 社交化编程——通过连接&聚合，重组人与人、项目与项目、人与项目三者之间的关系，催生各种化学反应，既促进了项目的快速进化，又提升了个人能力，更打破时间和空间的约束，协作无处不在。
- 最理想的企业内部开源&社交化编程平台：一个平台三大服务四大核心理念。

一旦开喝，啥也不重要了.....

- 年会是一种很奇怪的活动，事前各种准备，担心这个不好，那个不好，然而一旦开喝，这一切都不重要了.....
- 内部开源&社交化编程也是如此，当平台上线的那一刻起，大家一窝蜂就开喝了，之前的一切担忧，一切精心准备都会显得多余.....



但那一刻，我却无比地清醒……

- 开源圣经《大教堂与集市》P27写道：“……**在你第一次把问题解决的时候，你往往并不了解这个问题，第二次你才可能知道怎么把事情做好。所以，如果你想做对事情，至少要再做一次。**”
- 所以，大家可以将我的PPT倒叙再来一遍，或者，将这些拆开重组后再来一遍，相信会取得更好的效果。开源圣经《大教堂与集市》P27接着写道：“……**当你对一个程序不再感兴趣时，你最后的责任就是把它交给一个可以胜任的接棒者。**”

