同生活中的许多伟大事件一样，Git 诞生于一个极富纷争大举创新的年代。1991年，Linus创建了开源的Linux，并且有着为数众多的参与者。虽然有世界各地的志愿者为Linux编写代码，但是绝大多数的 Linux 内核维护工作都花在了提交补丁和保存归档的繁琐事务上（1991－2002年间）。在这期间，所有的源代码都是由Linus手工合并。因为Linus坚定地反对CVS和SVN，这些集中式的版本控制系统（集中式和分布式我们会在接下来的内容讲解）不但速度慢，而且必须联网才能使用。虽然有一些商用的版本控制系统，比CVS、SVN好用，但那是付费的，和Linux的开源精神不符。

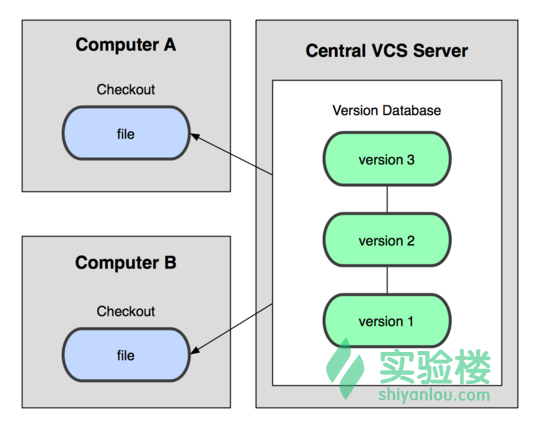
不过，到了2002 年，Linux系统已经发展了十年了，代码库之大让Linus很难继续通过手工方式管理了，社区的弟兄们也对这种方式表达了强烈不满，于是整个项目组启用了一个商业版本的分布式版本控制系统 BitKeeper 来管理和维护代码。BitKeeper的东家BitMover公司出于人道主义精神，授权Linux社区免费使用这个版本控制系统。安定团结的大好局面在2005年被打破，开发BitKeeper 的商业公司同 Linux 内核开源社区的合作关系结束，原因是Linux社区牛人聚集，开发Samba的Andrew试图破解BitKeeper的协议，这么干的其实也不只他一个，但是被BitMover公司发现了，于是BitMover公司收回了Linux社区的免费使用权。这就迫使Linux开源社区（ 特别是Linux的缔造者 Linus Torvalds ）不得不吸取教训，只有开发一套属于自己的版本控制系统才不至于重蹈覆辙。

他们对新的系统制订了若干目标：速度 、 简单的设计 、 对非线性开发模式的强力支持（允许上千个并行开发的分支）、完全分布式、有能力高效管理类似 Linux 内核一样的超大规模项目（速度和数据量）。自诞生于 2005 年以来，Git 日臻成熟完善，迅速成为最流行的分布式版本控制系统，在高度易用的同时，仍然保留着初期设定的目标。它的速度飞快，极其适合管理大项目，它还有着令人难以置信的非线性分支管理系统，可以应付各种复杂的项目开发需求。2008年，GitHub网站上线了，它为开源项目免费提供Git存储，无数开源项目开始迁移至GitHub，包括jQuery，PHP，Ruby等等。

历史就是这么偶然，如果不是当年BitMover公司威胁Linux社区，可能现在我们就没有免费而超级好用的Git了。

Linus一直痛恨的CVS及SVN都是集中式的版本控制系统，而Git是分布式版本控制系统，集中式和分布式版本控制系统有什么区别呢？

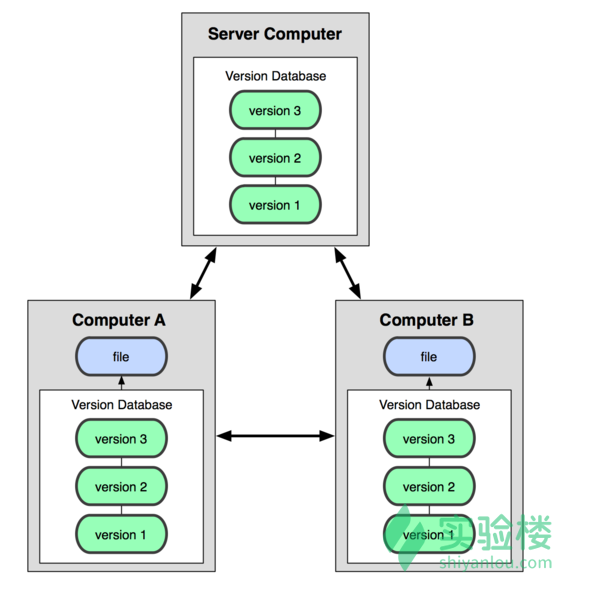
先说集中式版本控制系统，版本库是集中存放在中央服务器的，而大家工作的时候，用的都是自己的电脑，所以要先从中央服务器取得最新的版本，然后开始工作，工作完成，再把自己的修订推送给中央服务器。这类系统，都有一个单一的集中管理的服务器，保存所有文件的修订版本，而协同工作的人们都通过客户端连到这台服务器，取出最新的文件或者提交更新。



那分布式版本控制系统与集中式版本控制系统有何不同呢？首先，分布式版本控制系统根本没有“中央服务器”，每个人的电脑上都是一个完整的版本库，这样，你工作的时候，就不需要联网了，因为版本库就在你自己的电脑上。既然每个人电脑上都有一个完整的版本库，那多个人如何协作呢？比方说你在自己电脑上改了文件A，你的同事也在他的电脑上改了文件A，这时，你们俩之间只需把各自的修改推送给对方，就可以互相看到对方的修改了。

和集中式版本控制系统相比，分布式版本控制系统的安全性要高很多，因为每个人电脑里都有完整的版本库，某一个人的电脑坏掉了不要紧，随便从其他人那里复制一个就可以了。而集中式版本控制系统的中央服务器要是出了问题，所有人都没法干活了。

在实际使用分布式版本控制系统的时候，其实很少在两人之间的电脑上推送版本库的修改，因为可能你们俩不在一个局域网内，两台电脑互相访问不了，也可能今天你的同事病了，他的电脑压根没有开机。因此，分布式版本控制系统通常也有一台充当“中央服务器”的电脑，但这个服务器的作用仅仅是用来方便“交换”大家的修改，没有它大家也一样干活，只是交换修改不方便而已。



许多这类系统都可以指定和若干不同的远端代码仓库进行交互。籍此，你就可以在同一个项目中，分别和不同工作小组的人相互协作。你可以根据需要设定不同的协作流程，比如层次模型式的工作流，而这在以前的集中式系统中是无法实现的。