

svn和git的区别

一、svn的优势：

1. 优异的跨平台支持，对windows平台支持非常友好。
2. 简单易用，安装后稍微培训下就知道怎么操作。
3. 代码，需求，文档，涉及稿都可以用svn进行管理，适合不同部门的技术非技术的同事协作。

二、git的优势：

1. 去中心化：Git是没有中心服务器的，每个人机器上都是一个完整的库，我们平时开发代码时的中央服务器其实和我们自己机器上的库内容是完全一样的（格式有点不同，是bare的）。虽然平时大家都是将代码提交到中央服务器上再统一pull别人的代码，但实际情况你可以总是pull张三的库，然后push给李四等等操作。
2. 本地提交：本地提交好处主要有3点：一，断网提交。二，小步提交。可以对自己的阶段成果有跟踪，并且提高每次变更的安全性。三，本地库。这个和断网提交是同一个实现，但从需求角度出发则略有不同，主要是说即使只有自己一个人开发项目，也可以轻易的让自己的代码有版本跟踪，而不需要去费力建个什么svn server。四，本地回滚。这个其实是由于本地库的存在而产生的，但可以减少中央库上的冗余版本。
3. 分支策略：在Git实际开发中分支的分离和merge是属于日常操作，开启和合并分支成本相比SVN要小得多：SVN是复制一份代码到分支目录，Git则是在分支点做一下标记。随便一次冲突就会自动产生分支，所以大家每天都在与分支打交道。这便是弱化了分支的概念，由于分支成本很小，因此使得按功能分支的开发模式（每个分支一个功能，开发完了再merge到主干）变得非常简单，大家可以完全不需要再因为担心SCM成本太高而选用主干开发模式（所有功能都在主干上开发，到了发版本前再分离出分支）。

三、两者的工作流对比：

svn模式

1. 写代码。
2. 从服务器拉回服务器的当前版本库，并解决服务器版本库与本地代码的冲突。
3. 将本地代码提交到服务器。

git模式

1. 写代码。
2. 提交到本地版本库。
3. 从服务器拉回服务器的当前版本库，并解决服务器版本库与本地代码的冲突。
4. 将远程库与本地代码合并结果提交到本地版本库。
5. 将本地版本库推到服务器。

对比可以看出：分布式版本管理仅仅是增加了本地库这个概念，其余的概念与集中管理并无区别。——但是 svn 在与服务器同步之前无法提交代码，因而本地修改更容易出问题。

四、表格说明两者区别：

	集中式 (SVN)	分布式 (Git)
是否有中央服务器	有。开发人员需要从中央服务器获得最新版本的项目然后在本地开发，开发完推送给中央服务器。因此脱离服务器开发者是几乎无法工作的	没有中央服务器，开发人员本地都有 Local Repository
网络依赖	必须要联网才能工作，而且对网络的依赖性较强，如果推送的文件比较大而且网络状况欠佳，则提交文件的速度会受到很大的限制。	分布式在没有网络的情况下也可以执行commit、查看版本提交记录、以及分支操作，在有网络的情况下执行 push 到 Remote Repository。
文件存储格式	按照原始文件存储，体积较大	按照元数据方式存储，体积很小
是否有版本号	有	没有
分支操作的影响	创建新的分支则所有的人都会拥有和你一样的分支	分支操作不会影响其他开发人员
提交	提交的文件会直接记录到中央版本库	提交是本地操作，需要执行push操作才会到主要版本库

五、总结一下：

当研发成本比较低，协作开发人数不多，开发人员对于版本管理的水平参差不齐的时候，或者对于代码的安全性要求更高一点的时候，适合用svn。

而对于很多人参与开发，代码量比较大，或者高频次协作，跨公司，跨地域合作的情况下，更适合用git。