**Git分支 - Git分支介绍**

**1、Git分支简介**

几乎所有的版本控制系统都以某种形式支持分支。 使用分支意味着，你可以把你的工作，从开发主线上分离开来，然后在不影响主线的同时继续工作。 在很多版本控制系统中，这是一个略微低效的过程，常常需要完全创建一个源代码目录的副本。对于大项目来说，这样的过程会耗费很多时间。

有人把 Git 的分支模型称为它的“**必杀技特性**”，也正因为这一特性，使得 Git 从众多版本控制系统中脱颖而出。

为何 Git 的分支模型如此出众呢？ Git 处理分支的方式可谓是难以置信的轻量，创建新分支这一操作几乎能在瞬间完成，并且在不同分支之间的切换操作也是一样便捷。 与许多其它版本控制系统不同，Git 鼓励在工作流程中频繁地使用分支与合并，哪怕一天之内进行许多次。

理解和精通这一特性，你便会意识到 Git 是如此的强大而又独特，并且从此真正改变你的开发方式。

**2、Git分支与SVN分支的区别**

**（1）SVN分支简单说明**

1. SVN中的分支就是版本库中的另外一个完整目录，且这个目录拥有完整的实际文件。（就相当于把版本库复制一份）
2. 如果团队工作成员想要开启新的分支，那么团队对中的每个人都会拥有和你一样的分支。（因为集中式版本管理系统）
3. 在SVN中合并分支是非常麻烦和繁琐的，经常会发生有些分支或文件被遗漏的情况。
4. 当你进行创建和切换分支比蜗牛还慢，简直让人无法忍受。

**（2）Git分支简单说明**

1. Git中的分支就是创建一个指针指向某一个提交，就是我可以在Git中的任意一个提交点（commit point）创建分支。
2. Git中可以任意在自己的本地版本库创建无限个分支。
3. 我可以开一个分支，做我喜欢的事。完全不需担心妨碍其他工作成员，只要不合并到主仓库中，没有一个工作成员会被影响。
4. 等到我不需要这个分支时，我只要把它从我的本地版本库删除即可，无痛无痒。
5. 在Git中很容易发现未被合并的分支，你能简单而快捷的合并这些文件。
6. 你可以从同一个工作目录下，快速的在几个分支间切换，无论创建、切换和删除分支，Git在1秒钟之内就能完成！无论你的版本库是1个文件还是1万个文件。

**（3）总结**

* 创建分支：Git分支是指针指向某次提交，而 SVN 分支是拷贝的目录。这个特性使 Git 的分支切换非常迅速，且创建成本非常低。
* 而且Git有本地分支，SVN无本地分支。

**3、工作中为什么要使用分支**

分支在实际中有什么用呢？假设你准备开发一个新功能，但是需要两周才能完成，第一周你写了50%的代码，如果立刻提交，由于代码还没写完，不完整的代码库会导致别人不能干活了。如果等代码全部写完再一次提交，又存在丢失每天进度的巨大风险。

现在有了分支，就不用怕了。你创建了一个属于你自己的分支，别人看不到，还继续在原来的分支上正常工作，而你在自己的分支上干活，想提交就提交，直到开发完毕后，再一次性合并到原来的分支上，这样既安全，又不影响别人工作。

在分支在开发过程中，如果某一个分支开发失败，不会对其他分支有任何影响。失败的分支删除重新开始即可。

**4、Git分支管理的好处**

1. 同时并行推进多个功能开发，提高开发效率。
2. 版本迭代更加清晰。
3. 利于代码review的实现，从而使整个团队开发更加规范，减少bug率。

|  |
| --- |
| 提示：分支对于团队来说作用太大了。  比如说一个项目现在是1.0版，那么开发团队可能要同时进行1.1版和2.0版的开发，这样代码就会出现较大分歧。这时候就需要用到分支了，不同的任务组在不同的分支上开发，互相之间不会影响。  再比如说，需要向项目中添加一个新功能，一般的团队都不会直接在主分支上修改，都会新建一个分支，在上面更改代码。这样做的好处就是保证主线代码的完整性和可用性，也就是说，主线上都是稳定的代码，可以直接拿来发布的。  参考：   * [https://www.liaoxuefeng.com/wiki/896043488029600/896954848507552](https://links.jianshu.com/go?to=https%3A%2F%2Fwww.liaoxuefeng.com%2Fwiki%2F896043488029600%2F896954848507552) |

作者：繁华似锦Fighting 链接：https://www.jianshu.com/p/368a6af30811 来源：简书 著作权归作者所有。商业转载请联系作者获得授权，非商业转载请注明出处。