一步一步学习Git(1)——Git概述

现在的版本控制系统中，Git的人气越来越高，可能是因为Git是鼎鼎大名的大神Linus所写。最近在学习《版本控制之道——使用Git》，打算写一写读书笔记，也顺便让跟我一样刚接触的朋友一起来学习。

1. Git概述

版本控制系统(Version Control System, VCS)是用来帮助我们记录和追踪项目中各个文件的变化。传统的VCS有我们熟知的CVS，SVN。有传统当然有新颖的，Git就是这样一种新颖的版本控制系统，简单来说: Git是一种分布式版本控制系统(Distributed version control system, DVCS)。DVCS跟传统的VCS的区别就在于互相同步修改内容的方式不同。

下面我们将一步一步地逐渐认识Git。

2. 版本库

版本库(Repository)是版本控制系统用来存储所有历史数据的地方。比如说，存储文件的当前状态，历史修改时间，修改人员，修改原因等等。你可以把它想象为一个监控系统，只要你对里面的东西做过什么动作，它都记录得清清楚楚。

CVS和SVN这类版本控制系统属于集中是版本库模式，就是说，所有人都会把各自的修改提交到服务器上的一个公共版本库，而他们在本地也有一个工作目录树(待会再解释)，其内容就是版本库中最新的代码。如果要查询历史，就得询问服务器上的版本库，这往往就需要网络，而我们知道，不是任何时候都可以上网，这就是局限性。

如果你使用的是Git，那么这个问题就解决了，因为Git是分布式版本控制系统，所以你在本地会有自己的版本库，你的所有历史都存储在本地的版本库中。所以提交代码无须连接远程版本库，而是直接存储在本地版本库上面了。那么这里你可能有疑问：程序员如何同步？当然，还是要将修改上传到项目的主版本库，以后慢慢再学习。

3. 工作目录树

刚才提到过这个概念。什么是工作目录树(working tree)？工作目录树是版本库的一个“视图”，包括这个项目中所有的文件。这里要区分好版本库跟工作目录树。在Git中，版本库是存储在本地工作目录树的“.git”目录中(具体之后再讲)。而工作目录树是怎么来的？两种方法：一是初始化命令，二是克隆一个版本库，这些以后具体再说。简单说，就是你通过这两种方法中任何一种，生成了一个工作目录树，其中有一个目录叫“.git”，这个目录就是版本库。也不复杂吧：）

4. Git记录跟踪项目的方法

许多版本控制系统都是以文件为存储单位来存储，但是Git只是记录和跟踪组成该文件的各部分内容：若干字符和代码行。 Git同时为这些内容添加了一系列元数据，比如文件名，文件属性等等。 Git使用这种方法就降低了所需的磁盘空间了。这部分内容现在听起来会比较乱，我们以后再慢慢补充。

5. 标签

在你的项目中，一定有某一个点你需要作一个标记，比如说，你想发布一个版本。这时候也许你就要用到标签了，你可以为你的项目贴个标签，例如一个版本号，然后把你项目中所有代码都打包起来。

6. 分支与合并

假定你停留在一个问题上，想尝试编写代码解决这个问题，但是很有可能这个方法行不通，又怕到时候修改了代码恢复难。这个时候分支可以帮你解决问题。你可以在主干上创建一条新的分支，用这条分支来作为尝试，在这条分支上面修改代码，当然，这上面的修改并不影响主干上的代码。试验得出结果后，如果你不想要，也可以删除这条分支。

朋友，如果你是一个善良、正直、热爱学习并努力奋斗的人，请加我豆瓣或者QQ，我们一起进步:-)
QQ：397009575
Email: lin.jian1986@gmail.com
豆瓣： http://www.douban.com/people/3721525/

昵称：Linjian
园龄：5年8个月
粉丝：6
关注：0
+加关注

< 2010年7月 >

日	一	二	三	四	五	六
27	28	29	30	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31
1	2	3	4	5	6	7

搜索

找找看

随笔分类(45)

《代码大全》读书笔记(2)
Assembly(9)
C/C++(2)
CSAPP(2)
English Learning
Git(2)
Linux网络编程(2)
Python(3)
shell(2)
TCPL(4)
That's just life.(2)
经济学
数据结构(6)
职场心得(9)

随笔档案(47)

2011年1月 (4)
2010年11月 (6)
2010年10月 (8)
2010年9月 (15)
2010年8月 (4)
2010年7月 (6)
2010年6月 (4)

大部分分支还是需要合并到其他分支上面去的。合并，顾名思义，也就是把各个分支上面的修改都统一到一个分支上面来。这个是由Git来完成的，Git会比较各个分支之间的变化，确定变化后再进行合并。如果存在冲突，Git会告诉你发生了冲突，等着你自己来解决了。在这方面，Git提供了几种方法来解决冲突，这个以后我们再学习。

7. 锁机制

锁机制有两种，一种叫做严格锁。比如说，你现在修改这个文件，那么在你修改完并提交之前是没人能够修改这文件的，显然这不是高效率的方法，所以现在的版本控制系统都采用另外一种锁机制，叫乐观锁。当然跟严格锁相反，你可以和别人同时共同修改同一个文件，但是这里有一点需要注意的是提交的顺序所导致的问题：举个例子，两个人A和B同时修改了文件test.c，A修改后提交了，B修改后想提交，那么Git会通知B，在B修改的时候已经有人提交过新版本了，需要B确认是否存在冲突后进行提交。其实也不复杂，都是按照正常逻辑思维就可以明白的，呵呵。

PS：安装Git网上有不少方法，Git本来只可以在Linux上安装，因为它本来就起源于Linux嘛：）后来在windows下面也可以用，但是相对来说麻烦了一点，具体就不介绍了。我是在Ubuntu下面安装Git的，新立得软件包安装起来非常简单，下载安装也就一条命令：

sudo apt-get install git-core git-doc

注意看看需要哪些软件包作为前提，然后先安装这些软件就行了。

END

分类: Git

Linjian
关注 - 0
粉丝 - 6
+加关注

« 上一篇：<<A Byte Of Python>>第10章的例子——备份文件和目录
» 下一篇：Python小练习——创建简单地址簿

posted @ 2010-07-15 01:23 Linjian 阅读(418) 评论(1) 编辑 收藏

发表评论

#1楼 2013-08-13 09:57 | Kaiyu Lee

好文，看的明白，不自量力的去看英文教程，看一会儿就困了。。。

支持(0) 反对(0)

最新评论

1. Re:一步一步学习Git(2)——Git基本操作

大哥还有没？ 这么好懂，出个中文简单教程～

--Kaiyu Lee

2. Re:一步一步学习Git(1)——Git概述

好文，看的明白，不自量力的去看英文教程，看一会儿就困了。。。

--Kaiyu Lee

阅读排行榜

1. 写了个Linux下简单的FTP客户端程序(3228)
2. 一步一步学习Git(2)——Git基本操作(1596)
3. 关于接口设计的一点看法(1197)
4. 用shell脚本删除相同修改时间的文件(551)
5. 要想着以后总有人来维护你的代码(481)

评论排行榜

1. 一步一步学习Git(2)——Git基本操作(2)
2. 要想着以后总有人来维护你的代码(2)
3. 真正属于你的只有实力(2)
4. 一步一步学习Git(1)——Git概述(1)
5. 你不是一个人在战斗(1)

推荐排行榜

1. 要想着以后总有人来维护你的代码(2)
2. 循环队列数组实现(1)
3. 汇编语言复习摘要四——第一个汇编程序(1)