

- 1、Git概述
 - 1.1 版本控制
 - 1.2 版本控制工具
 - 1.2.1 集中式版本控制工具
 - 1.2.2 分布式版本控制工具
 - 1.3 关于Git
 - 1.3.1 Git发展历史
 - 1.3.2 git工作机制
 - 1.4 代码托管中心
- 2、Git安装
 - 2.1 下载地址
 - 2.2 安装步骤
- 3、Git常用命令
 - 3.1 设置用户签名
 - 3.2 初始化本地库
 - 3.3 查看本地库状态
 - 3.4 添加暂存区
 - 3.5 提交到本地库
 - 3.6 修改文件
 - 3.7 练习
 - 3.8 历史版本
 - 3.8.1 查看历史版本
 - 3.8.2 版本穿梭
 - 3.8.3 练习
- 4、Git的分支操作
 - 4.1 什么是分支
 - 4.2 分支的好处
 - 4.3 分支操作命令
 - 4.3.1 查看分支
 - 4.3.2 创建分支
 - 4.3.3 修改分支
 - 4.3.4 切换分支
 - 4.3.5 练习
 - 4.3.6 合并分支
 - 4.3.7 分支合并冲突
 - 4.3.8 解决合并冲突
 - 4.3.9 练习
- 5、Git团队协作机制
 - 5.1 团队内协作
 - 5.2 跨团队协作
- 6、Gitee操作
 - 6.1 简介
 - 6.2 码云帐号注册和登录
 - 6.3 创建远程仓库
 - 6.4 远程仓库操作
 - 6.4.1 创建远程仓库别名
 - 6.4.2 推送本地分支到远程仓库
 - 6.2.3 克隆远程仓库到本地
 - 6.2.4 邀请加入团队
 - 6.2.5 拉取远程库内容
 - 6.5 跨团队协作

- 7、IDEA集成Git
 - 7.1 IDEA中定位Git程序
 - 7.3 初始化本地库
 - 7.4 添加到暂存区
 - 7.5 提交到本地库
 - 7.6 切换版本
 - 7.7 创建分支
 - 7.8 切换分支
 - 7.9 合并分支
- 8、IDEA集成Gitee
 - 8.1 push 推送本地库到远程库
 - 8.2 pull 拉取远程库到本地库
 - 8.3 clone 克隆远程库到本地
- 9、工作中正确使用git
- 10、学习过程中使用git
 - 10.1 前提
 - 10.2 学校和家共同使用

1、Git概述

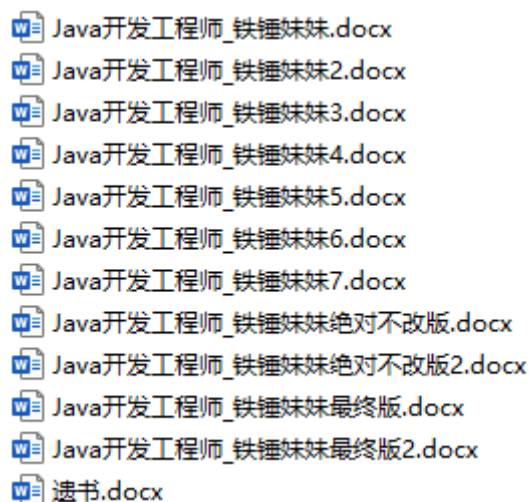
Git是一个开源的分布式版本控制系统，可以有效、高速地处理从很小到非常大的项目版本管理。于2005年以GPL发布。采用了分布式版本库的做法，不需要服务器端软件，就可以运作版本控制，使得源代码的发布和交流极其方便。

1.1 版本控制

- 何为版本控制

版本控制是一种在软件开发过程中用于管理对文件、目录或工程等内容的修改历史，方便查看更改历史记录，备份以便恢复以前的版本的软件工程技术。

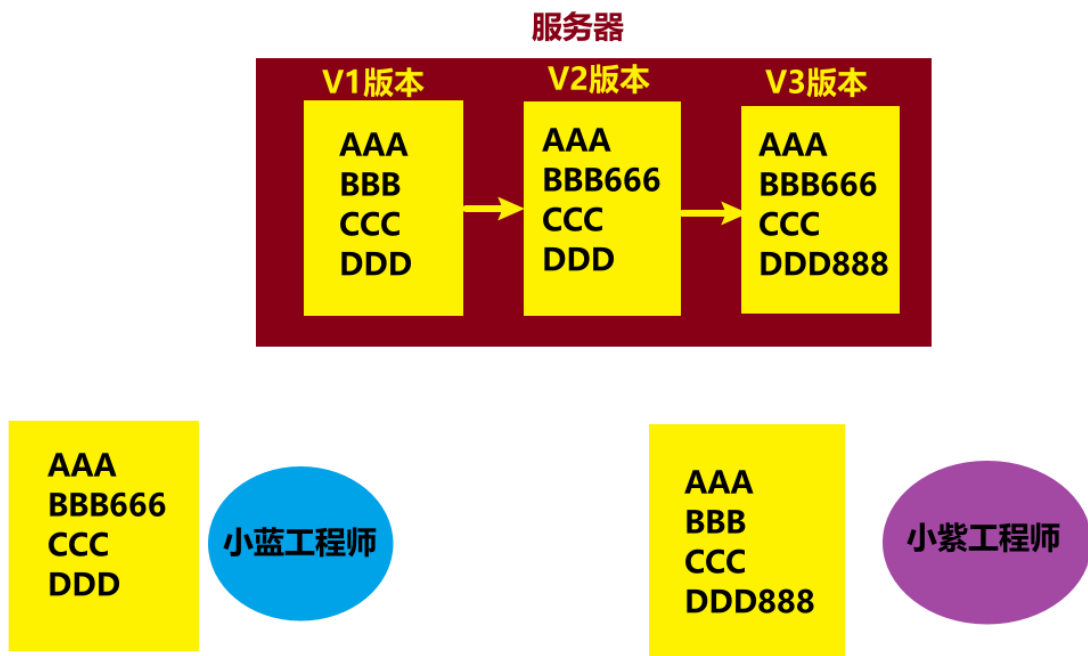
版本控制最主要的功能就是追踪文件的变更，每一次文件的改变，文件的版本号都将增加。除了记录版本变更外，版本控制的另一个重要功能是并行开发。软件开发往往是多人协同作业，版本控制可以有效地解决版本的同步以及不同开发者之间的开发通信问题，提高协同开发的效率。



Java开发工程师_铁锤妹妹.docx
Java开发工程师_铁锤妹妹2.docx
Java开发工程师_铁锤妹妹3.docx
Java开发工程师_铁锤妹妹4.docx
Java开发工程师_铁锤妹妹5.docx
Java开发工程师_铁锤妹妹6.docx
Java开发工程师_铁锤妹妹7.docx
Java开发工程师_铁锤妹妹绝对不改版.docx
Java开发工程师_铁锤妹妹绝对不改版2.docx
Java开发工程师_铁锤妹妹最终版.docx
Java开发工程师_铁锤妹妹最终版2.docx
遗书.docx

- 版本控制的作用

从个人开发转为团队协作



1.2 版本控制工具

1.2.1 集中式版本控制工具

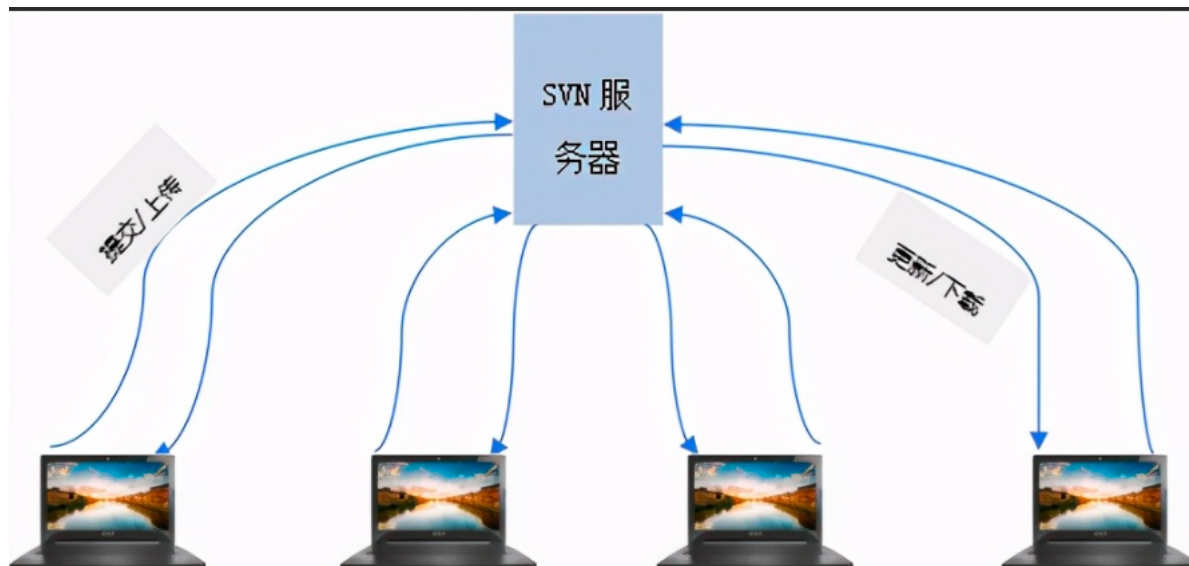
集中化的版本控制系统比如SVN，有一个单一的集中管理的服务器，保存所有文件的修订版本，而协同工作的人们都通过客户端连到这台服务器，取出最新的文件或者提交更新。多年以来，这已成为版本控制系统的标准做法。

优点

每个人都可以在一定程度上看到项目中的其他人正在做些什么。而管理员也可以轻松掌控每个开发者的权限，并且管理一个集中化的版本控制系统，要远比在各个客户端上维护本地数据库来得轻松容易。

缺点

缺点是中央服务器的单点故障。如果服务器宕机一小时，那么在这一小时内，谁都无法提交更新，也就无法协同工作。



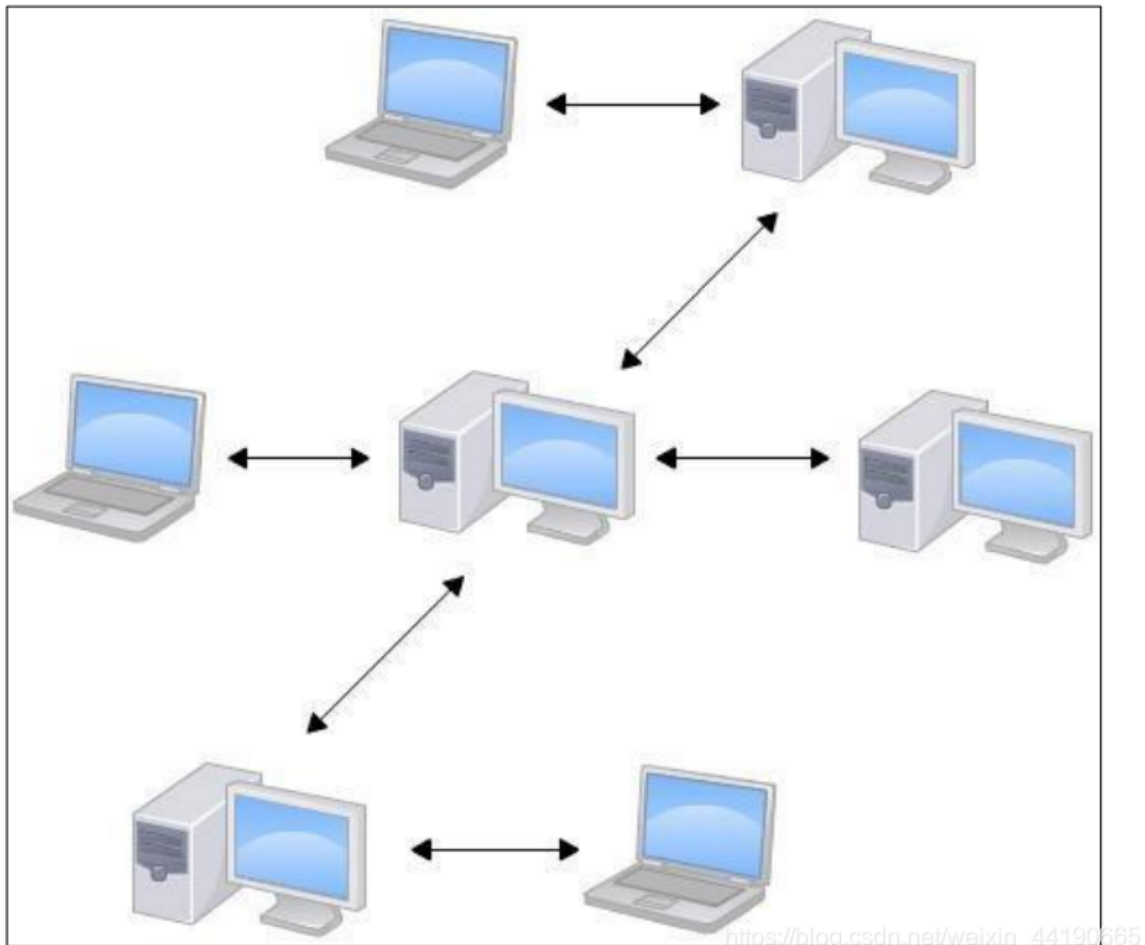
1.2.2 分布式版本控制工具

分布式版本控制工具（如Git）是一种允许开发者在本地存储和管理代码版本的系统，它不需要依赖中央服务器。这种工具可以在本地存储完整的版本历史，包括所有修改、合并等详细信息。开发者可以在本地进行提交和版本控制，并且可以随时与远程仓库进行同步。

相比于传统的集中式版本控制系统（如SVN），分布式版本控制系统具有更高的灵活性和安全性。由于每个开发者都可以在本地存储完整的代码库副本，因此可以更好地**支持远程协作和离线开发**。

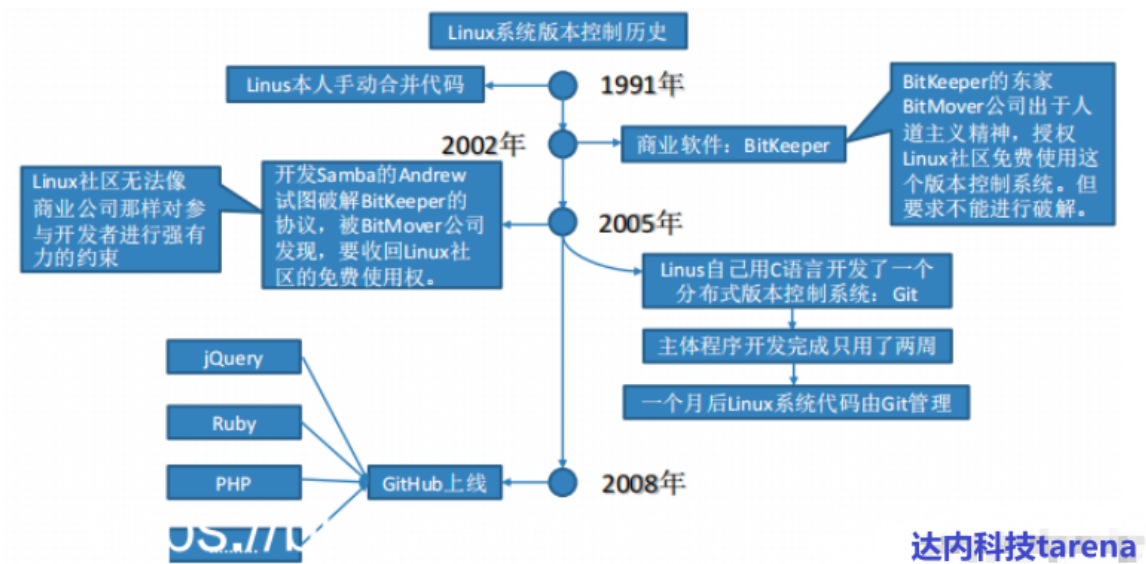
优点

1. 服务器断网的情况下也可以进行开发（因为版本控制是在本地进行的）
2. 每个客户端保存的也都是整个完整的项目（包含历史记录，更加安全）



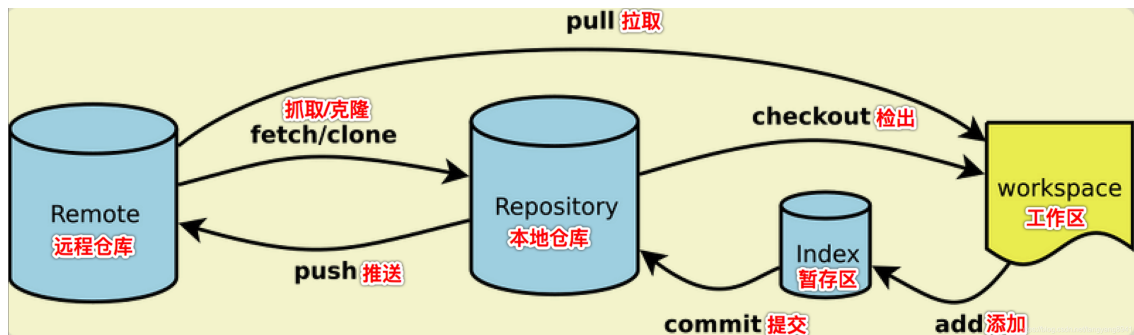
1.3 关于Git

1.3.1 Git发展历史

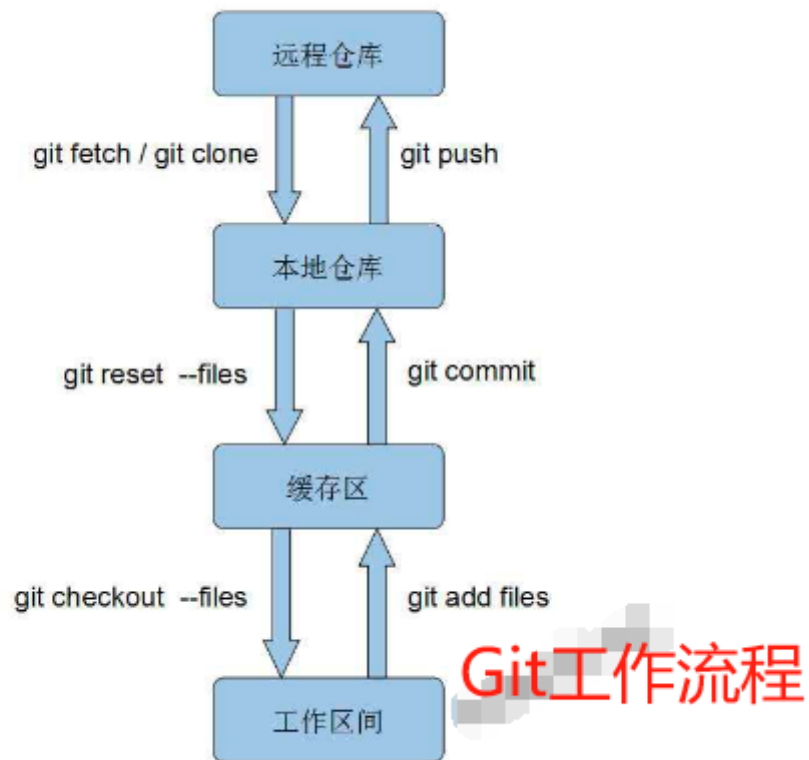


1.3.2 git工作机制

- 图解1



- 图解2



以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/916041052013010125>