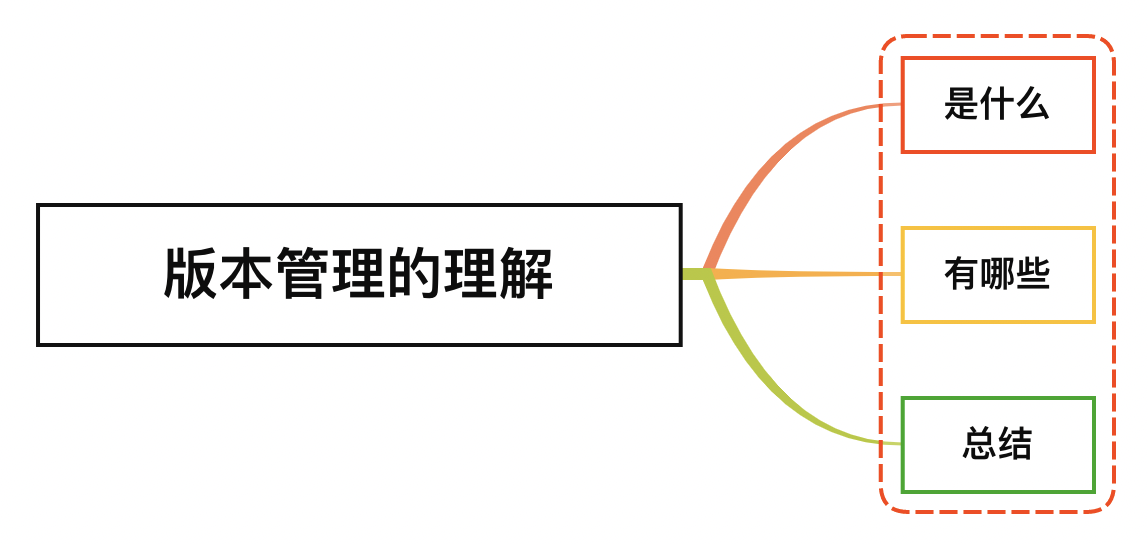
# 面试官：说说你对版本管理的理解？常用的版本管理工具有哪些？



## 一、是什么

版本控制（Version control），是维护工程蓝图的标准作法，能追踪工程蓝图从诞生一直到定案的过程。此外，版本控制也是一种软件工程技巧，借此能在软件开发的过程中，确保由不同人所编辑的同一程序文件都得到同步

透过文档控制，能记录任何工程项目内各个模块的改动历程，并为每次改动编上序号

一种简单的版本控制形式如下：赋给图的初版一个版本等级“A”。当做了第一次改变后，版本等级改为“B”，以此类推

版本控制能提供项目的设计者，将设计恢复到之前任一状态的选择权

简言之，你的修改只要提到到版本控制系统，基本都可以找回，版本控制系统就像一台时光机器，可以让你回到任何一个时间点

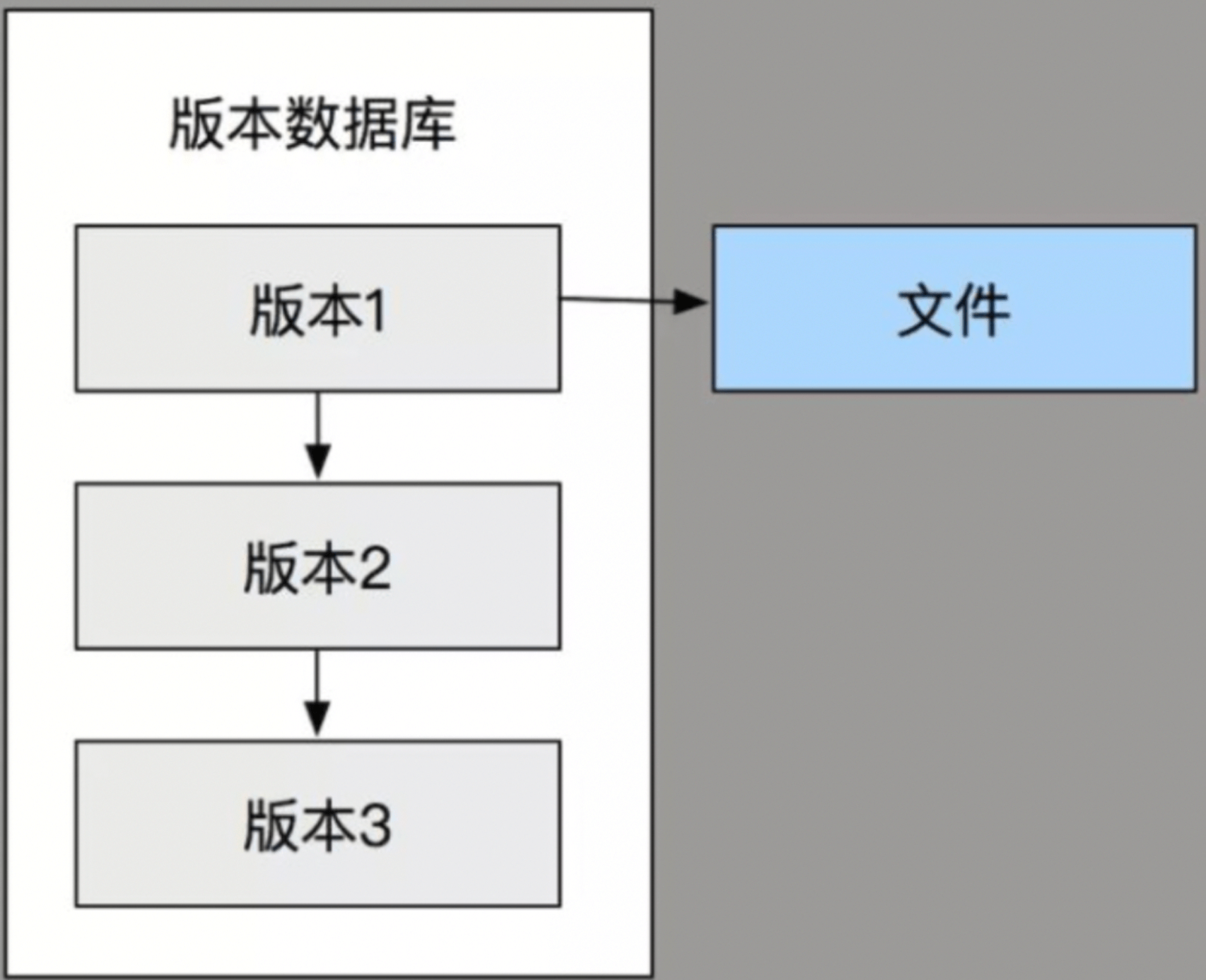
## 二、有哪些

版本控制系统在当今的软件开发中，被认为是理所当然的配备工具之一，根据类别可以分成：

* 本地版本控制系统
* 集中式版本控制系统
* 分布式版本控制系统

### 本地版本控制系统

结构如下图所示：



优点：

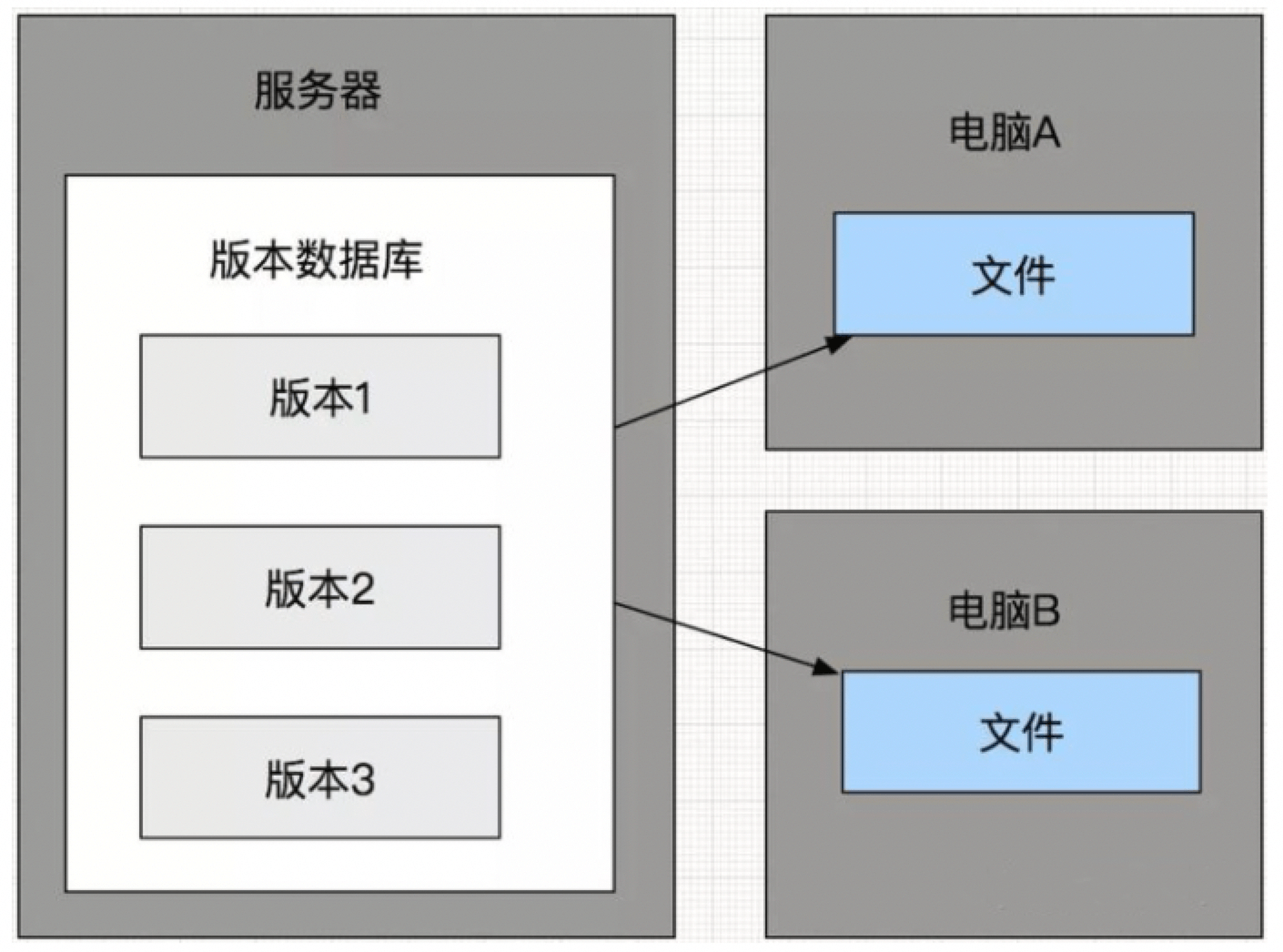
* 简单，很多系统中都有内置
* 适合管理文本，如系统配置

缺点：

* 其不支持远程操作，因此并不适合多人版本开发

### 集中式版本控制系统

结构如下图所示：



优点：

* 适合多人团队协作开发
* 代码集中化管理

缺点：

* 单点故障
* 必须联网，无法单机工作

代表工具有SVN、CVS：

### SVN

TortoiseSVN是一款非常易于使用的跨平台的 版本控制/版本控制/源代码控制软件

### CVS

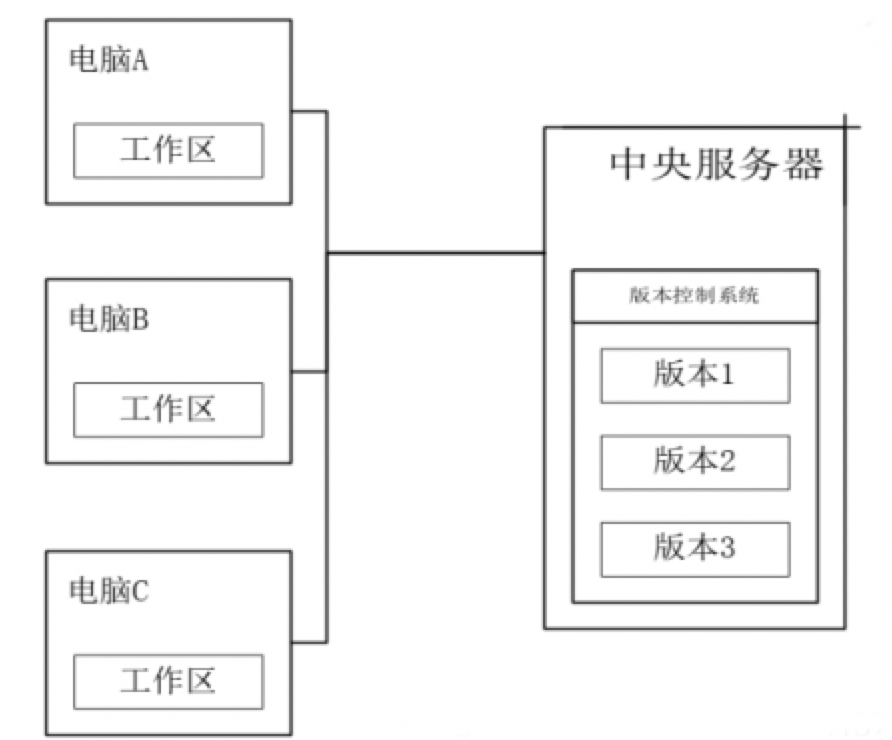
CVS是版本控制系统，是源配置管理（SCM）的重要组成部分。使用它，您可以记录源文件和文档的历史记录

老牌的版本控制系统，它是基于客户端/服务器的行为使得其可容纳多用户，构成网络也很方便

这一特性使得CVS成为位于不同地点的人同时处理数据文件（特别是程序的源代码）时的首选

#### 分布式版本控制系统

结构如下图：



优点：

* 适合多人团队协作开发
* 代码集中化管理
* 可以离线工作
* 每个计算机都是一个完整仓库

分布式版本管理系统每个计算机都有一个完整的仓库，可本地提交，可以做到离线工作，则不用像集中管理那样因为断网情况而无法工作

代表工具为Git、HG：

### Git

Git是目前世界上最先进的分布式版本控制系统，旨在快速高效地处理从小型到大型项目的所有事务

特性：易于学习，占用内存小，具有闪电般快速的性能

使用Git和Gitlab搭建版本控制环境是现在互联网公司最流行的版本控制方式

### HG

Mercurial是一个免费的分布式源代码管理工具。它可以有效地处理任何规模的项目，并提供简单直观的界面

Mercurial是一种轻量级分布式版本控制系统，采用 Python语言实现，易于学习和使用，扩展性强

## 三、总结

版本控制系统的优点如下：

* 记录文件所有历史变化，这是版本控制系统的基本能力
* 随时恢复到任意时间点，历史记录功能使我们不怕改错代码了
* 支持多功能并行开发，通常版本控制系统都支持分支，保证了并行开发的可行
* 多人协作并行开发，对于多人协作项目，支持多人协作开发的版本管理将事半功倍

## 参考文献

* https://pm.readthedocs.io/vcs/understanding.html
* https://zh.wikipedia.org/wiki/%E7%89%88%E6%9C%AC%E6%8E%A7%E5%88%B6