

科研实践报告

摘要

参加 PandaX 第四期的计划，了解到当前的工作进度是要通过改变以前几期项目的数据结构来建立当前的数据结构，这是因为这个项目的数据采集的方式和以前的项目有些不同，我们需要用到大量的计算机并让他们同时运作。同时学习 Git, GitHub, GitLab 等工具的使用方法，从而参加到项目的代码的编写当中。

关键字：Git; GitHub; GitLab; PandaX; MongoDB; 数据结构

背景:

由于 PandaX 前几期的项目使用的服务器只有一台，所以情况比较简单，数据结构也比较简单。第四期的计划中由于数据量巨大，所以我们需要用到 MongoDB 以及多台计算机来作为服务器，由于计算机数量的改变所以整个数据结构的改变。

我们需要将采集到的信号通过光电倍增管，每一个计算机连接多个光电倍增管，所有的计算机将处理过后的信号储存到 MongoDB 的数据库里，最后利用一台主计算机将数据库里的所有数据进行读取并处理。

由于是多台计算机将数据同时写入一个数据库的，为了保证数据的有序性，我们必须要将每一个事件的时间同时写入数据库，这样在读取数据库中的数据的时候我们才能得到有效的数据。通常地，我们在这里使用的是所谓的时间戳来记录时间。

我们当前的主要的任务就是改变整个数据结构，由于数据结构的改变，我们相应的数据的读取、写入、处理的方法也要进行相应的改变，所以具体涉

及到的很多的代码的重构。

前期的原来的数据结构是一层层分级的，在一个头文件里有很多行构成，每个行中又储存相应的事件的信息，而每一个事件又可以分成很多子项目，就这样一层一层地分下去。

第四期的计划中，我们需要自己重新构建数据结构，要将每一个事件不再打包，因为我们要将每一个事件打上一个相应的时间戳，所以我们不能再将其像原来一样进行包装。

Git 是一个分布式版本控制系统¹，是用来管理文件改动的一个系统，通过 Git 我们就不用再怕误删的发生，同时我们可以随时更换文件的版本，我们也可以利用 Git 来进行项目的合作。

分布式版本控制系统的优势就是每一个计算机都可以作为文件版本管理的服务器，这样就不用连接特定的服务器才能进行管理了。

通常我们使用 Git 来进行代码的版本的管理，Git 的具体功能有很多：创建版本库，我们可以随时切换到任意的版本，通过这种方式，我们可以管理文件的修改，撤销修改，删除文件等等；创建远程仓库，创建远程仓库的话我们需要一个服务器来储存，而且通过远程仓库，我们可以任意拷贝 Git 仓库；分支管理，通过创建分支我们可以让我们对代码进行试验性的改动，而不影响主代码，同时通过创建分支我们可以进行项目的合作，不同的人创建不同的分支，每个分支实现不同的功能，功能测试通过之后我们就可以将分支合并到主分支上从而让项目真正实现一个功能。

GitHub 是一个用来存放 Git 仓库的服务器²，同时它是一个开源的免费的远程服务器，同时它也是一个开源的协作社区，所以 GitHub 上面的任意的项

目就都可以让广大的人民群众参与进来，从而在一定的程度上减少了项目的 Bug 修复的难度，可以让广大的人民群众参与到 Bug 的修复和功能的完善上来。

GitLab 和 GitHub 一样，也是存放 Git 仓库的服务器³，但是相对于 GitHub 来说，GitLab 的开放性就没有这么好了，因为 GitLab 提供私有的 Git 仓库。不过这样的设置相对来说作为一个团队的开发项目比较合适，每一个团队可以建立一个 Git 仓库来进行项目的管理，而且每个人对每个仓库的权限可以设置成不一样的，所以整个仓库的安全性相对来说更高了。

工作内容:

参加 PandaX 项目的第四期，了解第四期的工作的内容，以及当前的工作进度，并了解接下来的工作内容。

参与 PandaX 计划中的 chain-bs3 这个项目。

参加 PandaX 第三期的会议，了解 PandaX 第三期执行的内容。

学习使用 Git 这个版本分布式管理器。

学习使用 GitHub, GitLab 来进行项目的合作。

学习使用 MongoDB 在 C++语言环境下的实现方法。

困难:

由于 PandaX 的第四期的项目是大概一年以前就启动了，而当时项目的负责人现在已经离开了 PandaX 第四期这个团队，而且项目并没有完成，所以当时的代码就留存了下来，所以如果我们要继续开展 PandaX 第四期的工作的话，需要先将原来的

代码都看过一边。

当时写的代码的注释较少，而且自身的 C++ 语言的熟练程度不够，所以阅读起来有一定的难度，要理解原来的程序需要一定的时间。

后续工作计划:

利用 MongoDB 来对原来的 PandaX 的数据结构进行重组。

继续使用 Git, GitHub, GitLab, 并熟悉它们的操作，从而进行项目的有效的管理。

参考文献：

1. Wikipedia contributors. "Git." *Wikipedia, The Free Encyclopedia*. Wikipedia, The Free Encyclopedia, 22 Apr. 2017. Web. 23 Apr. 2017.
2. Wikipedia contributors. "GitHub." *Wikipedia, The Free Encyclopedia*. Wikipedia, The Free Encyclopedia, 20 Apr. 2017. Web. 23 Apr. 2017.
3. Wikipedia contributors. "GitLab." *Wikipedia, The Free Encyclopedia*. Wikipedia, The Free Encyclopedia, 10 Apr. 2017. Web. 23 Apr. 2017.