**Git的工作流程**

**Git**是分布式的版本控制系统，服务器就是个体的电脑。我们可以通过Git在自己的电脑上访问本地文件和资源，也可以可以与远程的Git仓库相联系，我对其内容的简要理解就是：

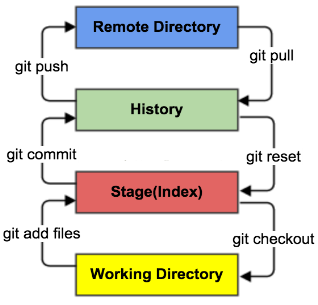
Git本地有三个工作区域

1. 工作目录（Working Directory）
2. 暂存区（Stage）
3. Git仓库（Repository）

**远程有一个工作区域**

1. 远程仓库（Remote Directory）

**Git**的流程大致如下图所示：



**Repository**

将上图转换成语言描述的话：

一开始我们在工作目录中修改文件，然后通过git add files暂存文件，即将文件的快照放入暂存区域，最后如果需要提交的时候，通过git commit –m””提交到本地的Git仓库

最后我们可以将本地仓库的文件通过git push上传到Github上面的远程仓库。

**Git分支**

**Git**分支意味着可以把工作从主线开发上分离出来，以免影响开发主线。可以自己创建新的分支，也可以自由地在分支中切换，以及合并本地分支。

**如何恰当地合并分支？**

**首先**我们得用git branch命令新建一个分支，并且用git checkout命令进行切换到该分支上，就可以在分支进行操作了。

如果想回到之前的分支，只需要执行git checkout master就可以了。

若做出了修改之后想要合并之前的状态，只需要用git merge+分支名称合并回master分支。如果需要删除不需要的分支，使用git banch –d就可以了。

Build-essential

libgtk2.0-dev

libavcodec-dev

libavformat-dev

libjpeg-dev

libtiff5-dev

cmake

libswscale-dev

libjasper-dev

libopenexr-dev

python-dev

python-numpy

python-tk

libtbb-dev

libeigen3-dev

yasm

libfaac-dev

libopencore-amrnb-dev

libopencore-amrwb-dev

libtheora-dev

libvorbis-dev

libxvidcore-dev

libx264-dev

libqt4-dev

libqt4-opengl-dev

sphinx-common

texlive-latex-extra

libv4l-dev

libdc1394-22-dev

libavcodec-dev

libavformat-dev

libswscale-dev

default-jdk

ant

libvtk5-qt4-dev

qt5-default