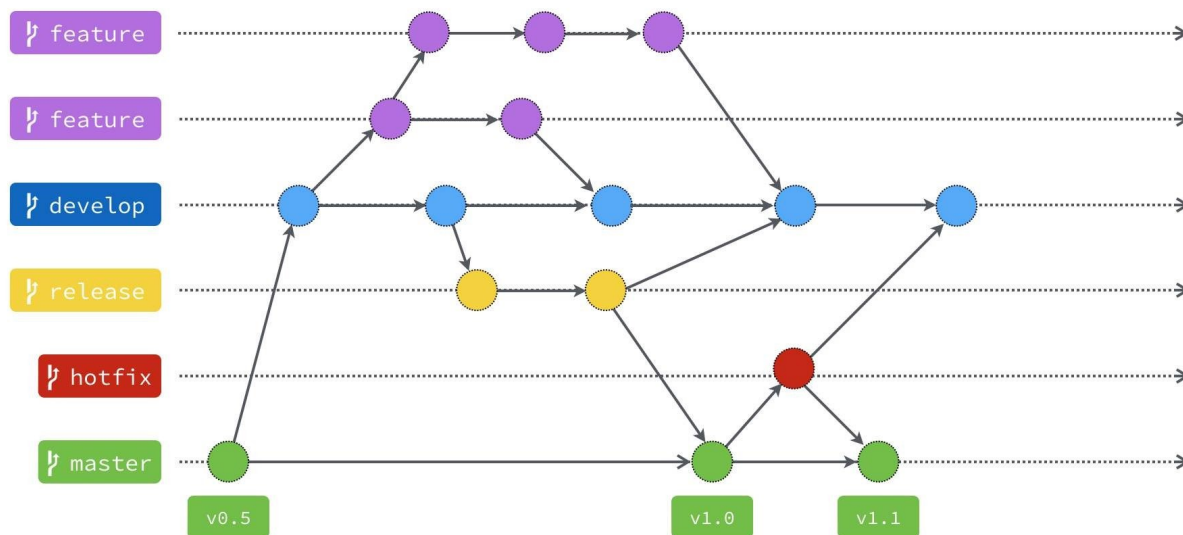


创建分支

Git最重要的一个知识就是分支的概念，在Git默认有个分支叫Master，也就是主分支。在这个分支上的代码一般都没有Bug，也就是线上运行的版本。



创建开发分支

```
git branch dev      # 创建分支
git checkout dev    # 切换分支

# 合并成一句话
git checkout -b dev
```

```
# liyang @ liyangdeMacBook-Pro in ~/Desktop/project on git:master o [21:20:00]
$ git checkout -b dev
Switched to a new branch 'dev'

# liyang @ liyangdeMacBook-Pro in ~/Desktop/project on git:dev o [21:20:06]
$
```

我们通过命令查看目前现在是在哪个分支上。

```
git branch
```

```
* dev
master
(END)
```

这个时候我再dev分支上修改1.txt内容

```
# liyang @ liyangdeMacBook-Pro in ~/Desktop/project on git:dev o [21:22:06]
$ vi 1.txt

# liyang @ liyangdeMacBook-Pro in ~/Desktop/project on git:dev x [21:22:23]
$ cat 1.txt
这是一个Git测试
这是修改内容
我又修改了内容
这是我再dev分支上加上的内容

# liyang @ liyangdeMacBook-Pro in ~/Desktop/project on git:dev x [21:22:24]
$
```

我们在dev分支上提交代码

```
# liyang @ liyangdeMacBook-Pro in ~/Desktop/project on git:dev x [21:24:58]
$ git add .

# liyang @ liyangdeMacBook-Pro in ~/Desktop/project on git:dev x [21:25:00]
$ git commit -m 'dev分支提交'
[dev 36791fe] dev分支提交
Committer: 李阳 <liyang@liyangdeMacBook-Pro.local>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

1 file changed, 1 insertion(+)

# liyang @ liyangdeMacBook-Pro in ~/Desktop/project on git:dev o [21:25:09]
$
```

我们在切换到master分支上看，在dev分支上加的内容可以在master分支上看到

```
# liyang @ liyangdeMacBook-Pro in ~/Desktop/project on git:dev o [21:25:09]
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.

# liyang @ liyangdeMacBook-Pro in ~/Desktop/project on git:master o [21:25:57]
$ cat 1.txt
这是一个Git测试
这是修改内容
我又修改了内容

# liyang @ liyangdeMacBook-Pro in ~/Desktop/project on git:master o [21:26:00]
$
```

我们在开发开发完成后，在把内容合并到master分支。只要在master分支执行如下命令

```
git merge dev
```

```
# liyang @ liyangdeMacBook-Pro in ~/Desktop/project on git:master o [21:27:45]
$ git merge dev
Updating 0ab242a..36791fe
Fast-forward
 1.txt | 1 +
 1 file changed, 1 insertion(+)

# liyang @ liyangdeMacBook-Pro in ~/Desktop/project on git:master o [21:27:51]
$ cat 1.txt
这是一个Git测试
这是修改内容
我又修改了内容
这是我再dev分支上加上的内容

# liyang @ liyangdeMacBook-Pro in ~/Desktop/project on git:master o [21:27:57]
$
```

可以看到在dev分支上修改的内容，已经合并到了master分支上，这个时候开发分支就可以删掉或者不删。

```
# liyang @ liyangdeMacBook-Pro in ~/Desktop/project on git:master o [21:35:16]  
$ git branch -d dev  
Deleted branch dev (was 36791fe).
```