## git、gerrit注意事项

### 1、gerrit权限列表

1）项目资源库创建

2）项目读权限和项目代码推送权限（写权限）

3）代码审核权限（具体打分为-2、-1、0、+1、+2，2分代表方能入库）

4）代码入库（已审核的代码合并进入项目资源库）

5）项目分支创建

6）项目标签创建

备注：某项目成员可同时具备不同权限

### 2、各成员权限划分

#### 1、注册成员

凡是注册成功的gerrit用户，都可根据项目列表进行项目克隆，同时具备

2）项目读权限和项目代码推送权限（写权限）

#### 2、项目普通成员

作为项目小组普通开发人员，具备以下基本权限

2）项目读权限和项目代码推送权限（写权限）

3）代码审核权限-1~+1

#### 3、小组负责人

2）项目读权限和项目代码推送权限（写权限）

3）代码审核权限-2~+2

6）项目标签创建

#### 4、项目负责人

2）项目读权限和项目代码推送权限（写权限）

3）代码审核权限-2~+2

4）代码入库（已审核的代码合并进入项目资源库）

6）项目标签创建

#### 5、其他成员

除项目各个开发组成员外，有临时其他同事需查看项目库资源库，如非特殊情况，应只具备注册成员同等权限。

#### 6、gerrit管理员

管理员具有全部权限

1）项目资源库创建

2）项目读权限和项目代码推送权限（写权限）

3）代码审核权限（具体打分为-2、-1、0、+1、+2，2分代表方能入库）

4）代码入库（已审核的代码合并进入项目资源库）

5）项目分支创建

6）项目标签创建

### 3、分支、标签管理

#### 1、项目分支创建策略

1）项目应该只有两个常设分支master和devel（可以根据项目需要进行自主命名）,日常开发只应该在devel分支上,master作为项目创建后默认分支，更多的作用应该是用于发布时使用，而devel分支主要用于开发使用。

2）新分支创建权限不开放，应该由项目负责人或者配置管理员统一管理，如果有具体的项目客观需要，可以与项目负责人进行沟通决定。

3）除了以上两个常设分支外，我们还可以适当分出不同用途的分支，使用完后也该删除，保持两个常驻分支。

#### 2、项目标签创建策略

项目标签的具体权限可以申请下放，配置管理员应该只分配具有注释功能的分支权限，希望标签创建应该具备以下几点

1）应该清晰、明确，一目了然。

2）具体查看（show）时应该有详细的目的描述。

3）不应该过多的创建而使得追溯进行变得复杂。

### 4、提交规范

为使项目提交具有清晰明确的可追溯性，希望广大开发人员提交遵守以下原则

1）提交编辑之前，请确保工作库为最新状态，避免不必要的麻烦。

2）为避免不同环境查看生成中文乱码，请一致使用英文提交日志。

3）尽量清晰明确的描述所提交代码解决的问题。

4)尽量确保一条日志对应解决一个问题。

5）在本地工作库提交的代码，应及时的推送至服务端，每次推送应该为一条提交记录，多条记录推送前应该合并为一条记录，也可以在推送完一条提交记录后进行不提交，以避免不必要的麻烦。

6）在gerrit已经评审过的代码及时的进行合并，并且确保依赖的代码已合并入库，这也是减少冲突的方式之一。

7）项目组(参与GIT管理的各项目)成员 张庆保,郭建省,李广钊,刘铁铮,李达,吴嘉祥,拥有代码的+2审核权,其他人员,均为+1权限,且在提交代码时,不允许自己给自己审核的情况,必须两人(或以上)审核通过后,才能将代码提交到代码库中.  
   只有在代码有小问题(比如书写格式不符合规范之类的),但是代码又必须提交GIT库时(不提交会影响其他人第二天的开发工作),才允许使用+2审核权限,其他情况下禁止使用+2审核权.