# Linux搭建服务器

## Linux

### 详细过程

先查看是否安装过svn

[root@localhost ~]# rpm -qa |grep subversion

安装svn

yum -y install subversion

创建仓库 装好svn就可以新建仓库了。

创建库根路径 mkdir -p /root/svn

svnadmin create /root/svn/wenwenliang # wenwenliang是库名称

配置仓库

进入project目录，文件列表：

drwxr-xr-x. 2 root root 5 Jun 8 17:28 conf

drwxr-xr-x. 8 root root 18 Jun 23 17:31 db

drwxr-xr-x. 6 root root 15 Jun 8 17:33 db~

drwxr-xr-x. 6 root root 16 Jun 8 17:33 db~~

-rw-r--r--. 1 root root 2 Jun 8 17:28 format

drwxr-xr-x. 2 root root 13 Jun 8 23:30 hooks

drwxr-xr-x. 2 root root 4 Jun 8 23:00 locks

drwxr-xr-x. 2 root root 4 Jun 8 22:58 locks~

-rw-r--r--. 1 root root 229 Jun 8 17:28 README.txt

进入conf文件夹有三个文件 我们需要修改这三个文件。

authz//账号权限管理，

passwd //用户密码

svnserve.conf //配置上面两个文件的路径关联等

添加用户：修改passwd文件：

只需要在文件后面添加即可：

[users]

# harry = harryssecret

# sally = sallyssecret

#soft user

xxx=123 #用户名=密码

为用户添加权限操作：修改authz：

这里可以采用组管理，组管理要复杂些，也可以不分组。（原文件只需要添加内容）

[groups]

g\_admin=xxx,bbb #g\_admin 是组名 在groups下建立一个组

[wenwenliang:/] //这里需要自己加 设置svn的根目录 也可以直接 [/]

@g\_admin=rw //设置该目录下的组用户读写权限:r：读 w ：写

\*=r //所有用户可读

把修改svnserve.conf文件：关联上面两个文件等配置

只需要改五行，已标出

[general]

anon-access = none #不允许匿名访问 #没有登录不能操作

auth-access = write #可写 #登录后可以写操作

password-db =/root/svn\_conf/passwd #用户密码文件地址，其实不用绝对路径的，默认打开即可就是相对路径

authz-db = /root/svn\_conf/authz #用户名配置文件地址其实不用绝对路径的，默认打开即可就是相对路径

realm =project #根目录

到此，配置文件完成。

2.5 设置防火墙 关掉防火墙即可

一般位置为 /etc/sysconfig/,要想加入SVN端口（默认3690）,可用编辑器或vi命令加入以下行（commit上面）：

-A INPUT -i eth0 -p tcp -m tcp --dport 3690 -j ACCEPT

-A OUTPUT -o eth0 -p tcp -m tcp --sport 3690 -j ACCEPT

然后重启防火墙，service iptables restart  
如果启用失败，输入：

service iptables save

附: 关于防火墙的几个用法

临时关闭防火墙

systemctl stop firewalld

永久防火墙开机自启动

systemctl disable firewalld

临时打开防火墙

systemctl start firewalld

防火墙开机启动

systemctl enable firewalld

查看防火墙状态

systemctl status firewalld

启动SVN

svnserve -d -r /zfspool/svn/ 注意：这里不要指定到库名称

客户端访问仓库：

输入：svn://ip地址/project  即可。

2.7 利用systemctl设置开机自启动

CentOS 7的服务systemctl脚本存放在：/usr/lib/systemd/，有系统（system）和用户（user）之分，需要开机不登陆就能运行的程序，存在系统服务里，即：/usr/lib/systemd/system目录下  
每一个服务以.service结尾，一般会分为3部分：[Unit]、[Service]和[Install]

修改svnserve.service文件：

[Unit]

Description=Subversion protocol daemon

After=syslog.target network.target

[Service]

Type=forking

EnvironmentFile=/etc/sysconfig/svnserve<span> </span>//运行命令的参数

ExecStart=/usr/bin/svnserve --daemon --pid-file=/run/svnserve/svnserve.pid $OPTIONS<span> </span>//运行命令

[Install]

WantedBy=multi-user.target

其中参数所在的文件内容是：

# OPTIONS is used to pass command-line arguments to svnserve.

#

# Specify the repository location in -r parameter:

OPTIONS="-r /zfspool/svn/"

保存后并添加权限754。执行：

systemctl enable svnserve.service

然后，启用：

service svnserve restart

后续

找出该进程，直接kill掉。

ps -aux |grep svn

kill -9 进程id

删除版本库

rm -rf /var/svn/test

[groups]

admin = admin #admin为用户组,等号之后的admin为用户

test = fuhd,test

[test:/] #表示根目录（/var/svn/test），test: 对应前面配置的realm = test

@admin = rw ＃表示admin组对根目录有读写权限,r为读，w为写

[test:/test] #表示test目录（/var/svn/test/test）

@test = rw #表示test组对test目录有读写权限

启动SVN服务

# -d表示后台运行 -r 指定根目录是 /var/svn --listen-port：监听端口

svnserve -dr /var/svn --listen-port 3690

查看SVN服务

ps aux|grep svnserve #默认端口为：3690

注意打开防火墙和关闭selinux

vim /etc/sysconfig/selinux

SELINUX = disable #值修改为disable

文件上传到服务器后存放在服务器的哪个地方。

一般放在/home/svn/svntest/db/revs（根据我自己的目录结构）里面。

### 遇到的问题

出现权限问题

svnserve -d -r /zfspool/svn/ 启动服务的时候注意：这里不要指定到库名称

import的时候提示错误

svn: E000022: Error converting entry in directory '/var/www/test/images' to UTF-8

svn: E000022: Can't convert string from native encoding to 'UTF-8':

问题就是环境变量的编码设置的不好，解决办法

vim ~/.bashrc

加上

export LC\_ALL=en\_US.UTF-8

export.UTF-8

export LANGUAGE=en\_US.UTF-8

source ~/.bashrc

update的时候提示错误

post-commit hook failed (exit code 1) with output:

Conflict discovered in '/data/www/index.php',

Select: (p) postpone, (df) diff-ful, (e) edit.

(mc) mine-conflict, (tc) theirs-conflict,

(s) show all options: svn: E070014: Can't read stdin: End of file found

由于更新的问题，网站和svn里的文件不一致，从svn checkout一份文件覆盖到网站里面去

svn checkout --depth=empty file:///svn/www/ www

cd www

svn update index.php

其中，www为checkout后的文件夹名，自己可以随意指定，执行svn checkout命令后，会得到一个名为www的空目录，里面什么都没有，而实现这一效果的就是参数中指定的depth=empty。

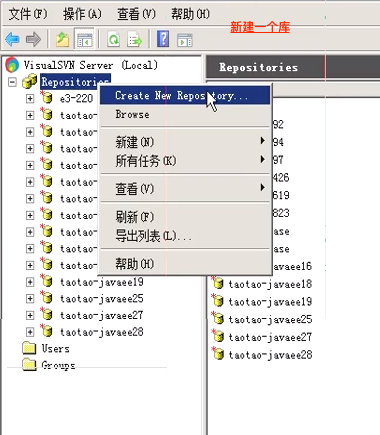
下一步就是进入刚刚checkout下来的www目录，执行update操作，想取哪个文件就checkout哪个文件就好了。最后把不一致的文件copy到网站去覆盖原有文件（最好把原有文件备份一份）

## Windows

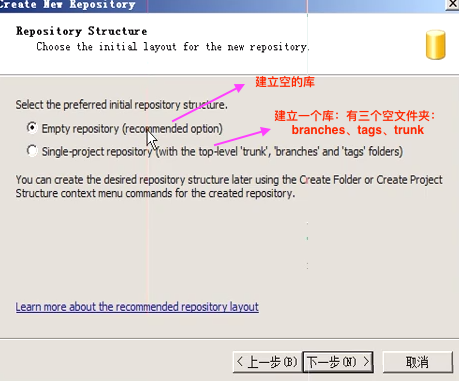
### 搭建服务器

使用工具VisualSVN Server

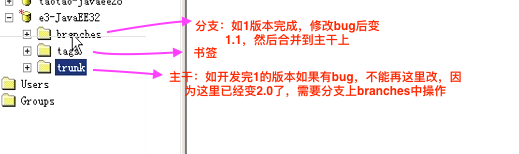
新建一个库



建一个文件名(这里不截图了)，然后下一选项注意问题：



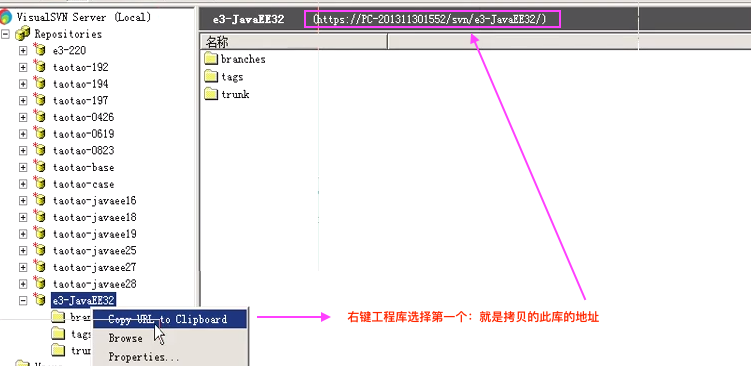
然后再一步(默认即可，老师也没管)，然后完成



# SVN导入导出

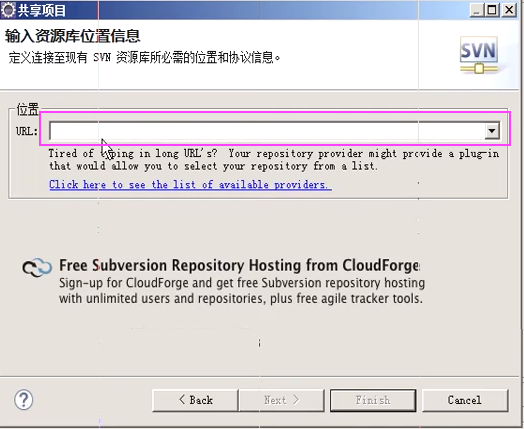
## 导出到服务器

库建好后怎么使用呢？

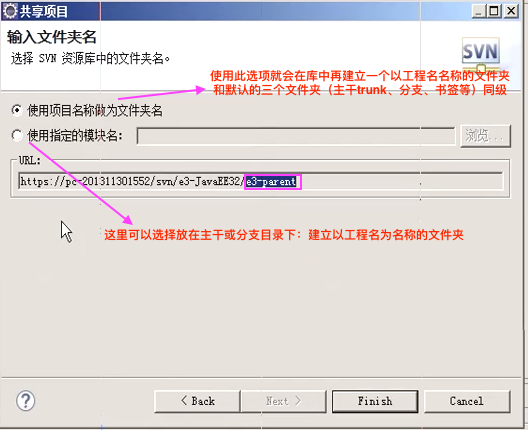


Eclipse中引用

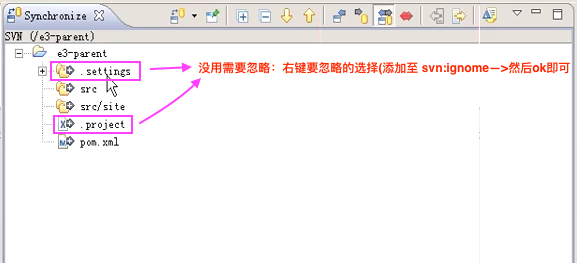
右键工程 Team ----选择Share Project—选择SVN---创建新的资源库位置这个位置放url



然后下一步：

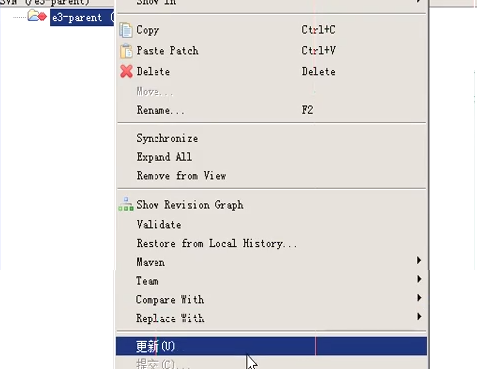


第一次让你输入SVN服务器用户名和密码---> 接着注释完成---> 提示要不要切换到同步视图-->切换后

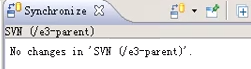


下一步右键工程名选择提交然后写入注释即可

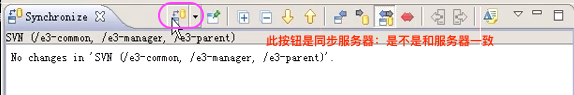
注意如果出现冲突的话：右键更新然后再进行提交



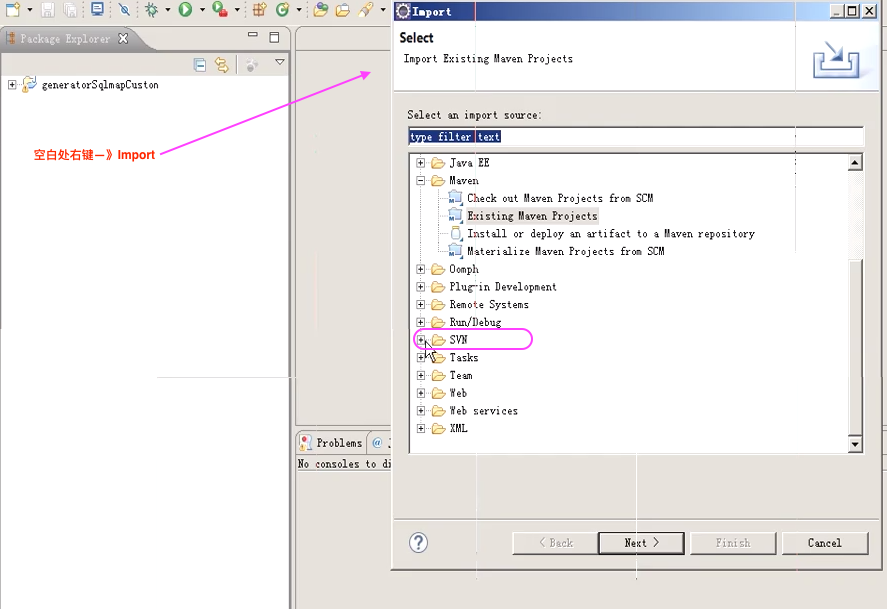
最后显示：就没问题了



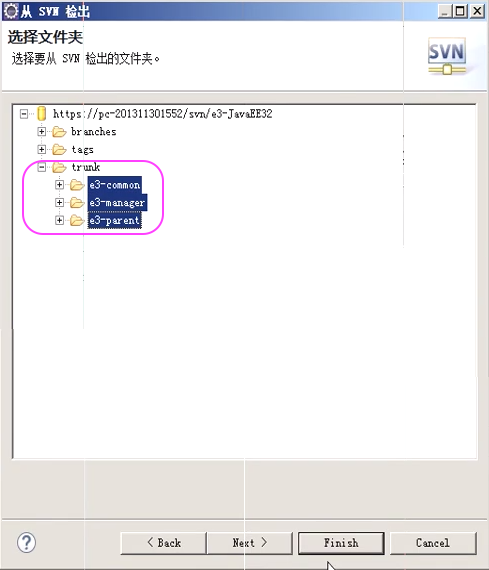
点击是否和服务器代码一致



## 从服务器导出

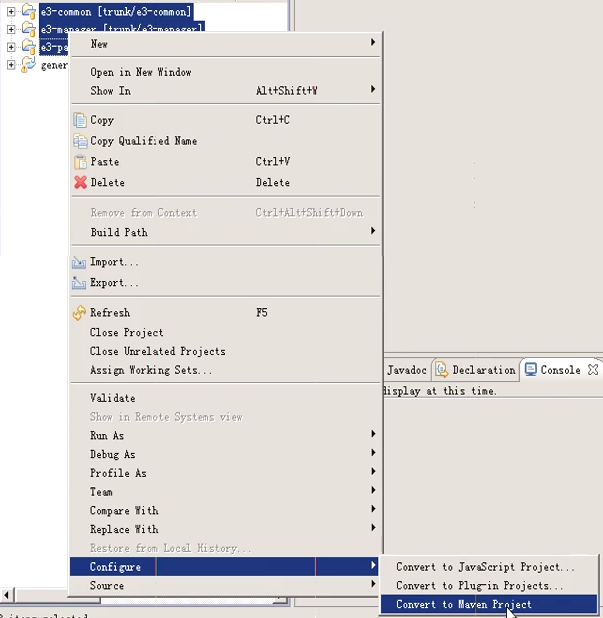


然后下一步—使用现有的资源库位置：选中需要导出的工程



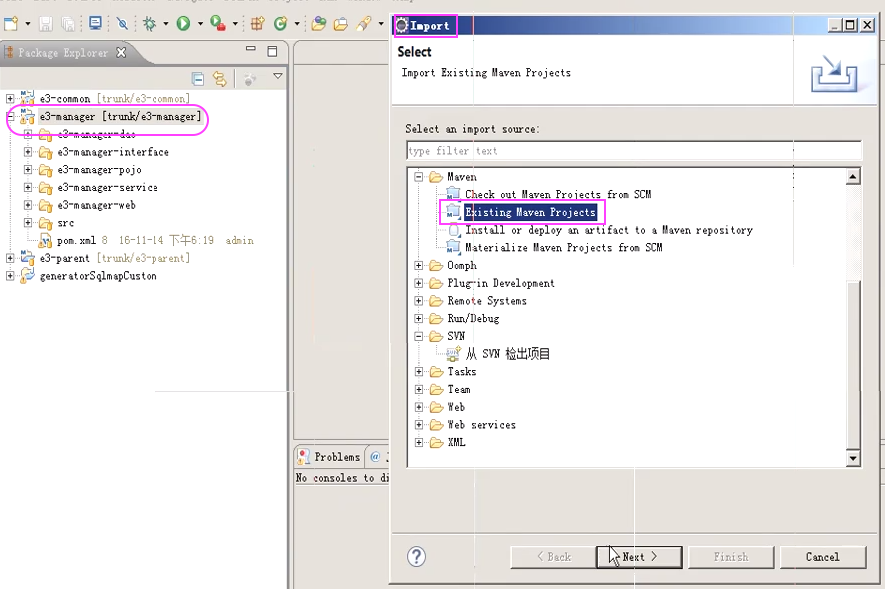


此时发现不是maven工程：进行转换为maven工程



然后e3-managen的pom工程下还有子工程几个模块没有(这样便于开发)

右键e3-managen ---》import--》选择Maven

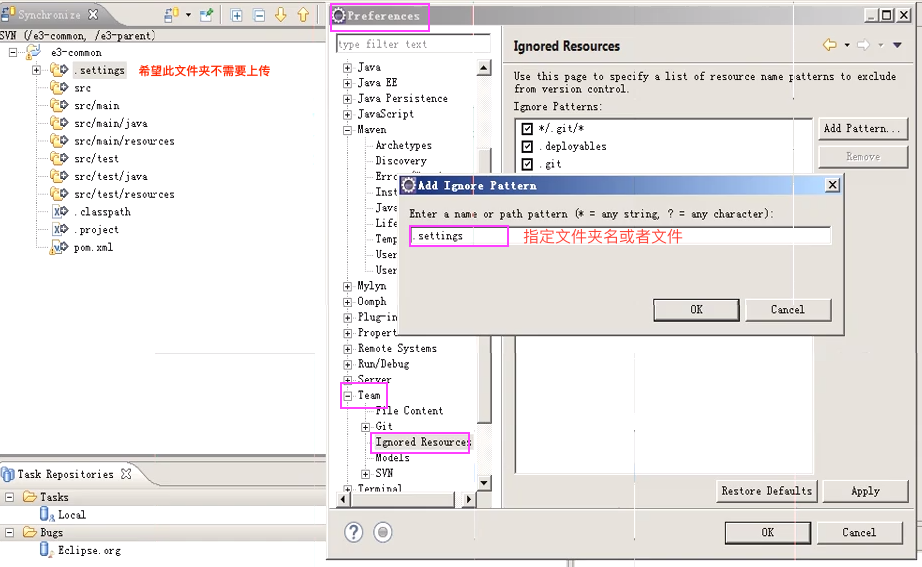


至此导出maven工程结束

# Eclipse

## 配置不需要上传文件

配置不需要上传的文件夹或文件



<https://www.cnblogs.com/hmwnote/p/8336325.html>

## [svn插件使用教程](https://www.cnblogs.com/hmwnote/p/8336325.html)

### **安装svn插件**

1、Help——>Install New Software

2、点击“Add”，在弹出的窗口里，输入任意name，及对应的SVN地址 http://subclipse.tigris.org/update\_1.8.x ，然后点“OK”

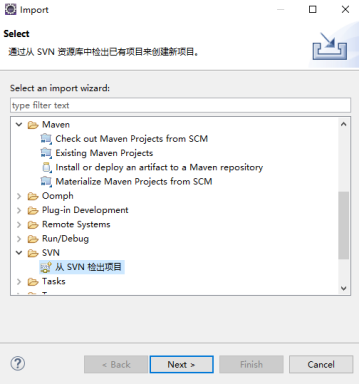
3、将出现的2个选项都选中，然后一路点Next，就可以了

4、安装成功标志：

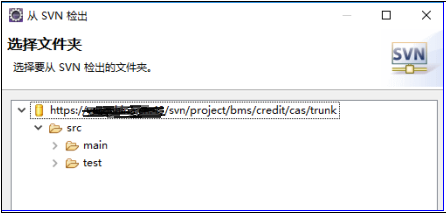


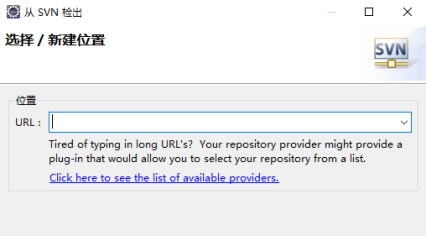
### 从svn检出项目

1、开始检出程序：打开eclipse->文件->import->SVN->选择"从svn检出项目”单击"下一步"



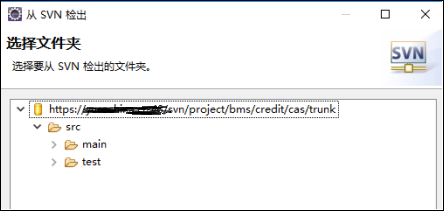
2、以前检出过项目，配置了svn地址，直接选择需要检出项目的svn地址即可，然后跳到第4步。如果是第一次检出项目，从没有配置过svn地址，选择“创建新的资源库位置”，单击“下一步”



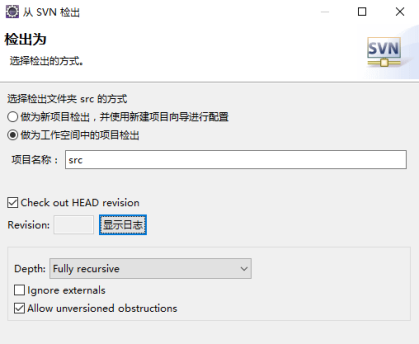


3、输入svn地址，如果地址错误这里会有提示，输入成功，单击“完成”

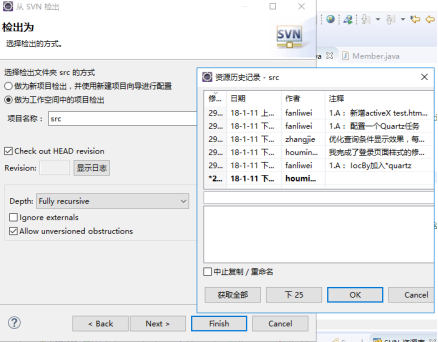
4、进入svn库项目，可以看到树形的项目列表，选择需要检出的项目，单击选中即可，单击“下一步”

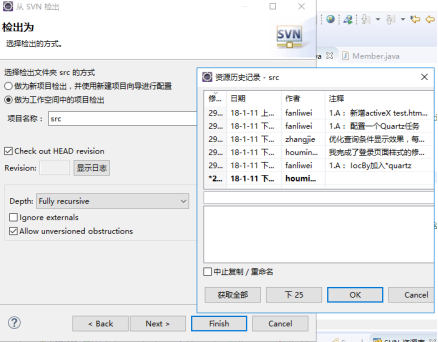


5、一般为了方便识别，不需要改检出的项目名称，也不需要进行其他的配置，直接选择“作为工作空间中的项目检出”即可，如果要检出最新版本，什么都不需要更改，直接单击“完成”即可。



6、想选择检出的版本，单击“显示日志”，在弹出框中选择想要检出的svn版本号，单击“确定”，然后检出就可以了





## svn更新项目

### 为什么要先更新再提交？

更新：把他人上传的代码更新到本地环境，你在这个环境下把自己的代码合并到这个版本中。

从svn检出的项目每个文件后面都带有版本号、提交时间、提交人等标识，并且这些标识不尽相同。如：

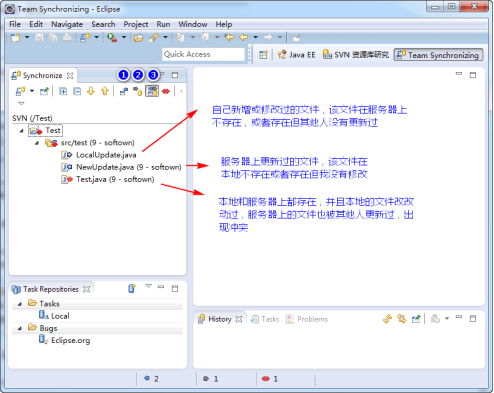
https://images2017.cnblogs.com/blog/1323776/201801/1323776-20180123152602725-1205015314.png

也就是说，一个项目是由若干个具有不同版本号的文件组成的，虽然我早上从SVN上检出的项目中的每个文件都是最新版本的，如index.jsp版本号是2971，但是工作一天之后，当我想要提交代码的时候，我不能保证服务器中项目的每个文件还是和早上刚刚从SVN检出时一样，可能这期间有同事已经提交过代码，所以其修改过的某个文件的版本号就发生了改变，如index.jsp的版本号变2972，而我本地的这个文件的版本号并没有改变，还是2971，如果我同样修改了index.jsp，不更新直接提交，那么我提交的index.jsp版本号也会从2971变为2972，这样将导致我的文件会直接将同事提交的这个文件覆盖，同事的劳动成果将白费。所以，提交代码之前应该先“更新”一下，更新的并不是所有文件，而是项目中的个别文件，这样做的目的是为了能使自己的版本与别人更新的版本保持一致。

### 更新的步骤(解决冲突等操作)

1.右键点击指定的项目，在弹出的关联菜单中点击Team->与资源库同步。

此时，我们就进入了团队同步视图。如下图所示：

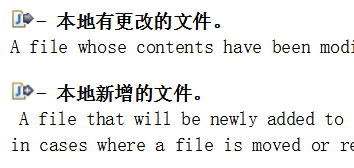


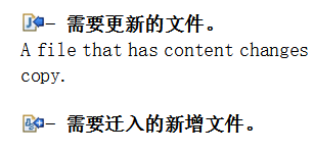
LocalUpdate.java是我们在本地新建的一个文件，服务器上之前不存在该文件；

NewUpdate.java是其他人新建并提交的文件；

Test.java是本地代码在未更新到最新版本的情况下，对代码进行了更改，出现了冲突。

不同标志的含义：





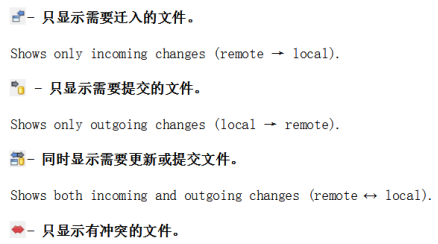
点击图中编号①的图标，可以只显示服务器上更新过的文件；

点击图标②，可以只显示我们本地改动过的文件；

图标③就是Eclipse默认的当前视图，同时显示本地和服务器更新过的文件，

可以从箭头的方向判断，箭头向左就是需要从svn更新到本地的，向右就是需要从本地提交到SVN的。

不同标志的含义：



2.更新

在上图中，选中服务器上更新过的一个或多个文件/文件夹，然后点击鼠标右键，在弹出的关联菜单中，点击【更新】，即可将SVN服务器最新版本的该文件更新到本地项目中(选中项目根目录点击【更新】，就表示将服务器上所有改动过的文件全部更新到本地)。

3.解决冲突

冲突情况1：服务器代码和自己代码改动的地方相同

这种情况比较复杂，一般只能将自己的代码保存一份到本地，然后直接将服务器代码更新下来，然后在新的代码上修改自己的逻辑。

冲突情况2：服务器代码和自己代码改动的地方不同

这种情况比较常见，就是你跟别的开发者都在更改同一个文件，但是由于开发任务是分开的，所以更改的地方是不同的，解决办法如下：

①双击冲突文件进入比较视图

②找到服务器上新增的但本地没有的代码块，选中此代码块，点击右上角的按钮将服务器代码复制到本地（其他地方也同此操作），如果点击按钮没用，则关闭比较视图重新打开

③将所有服务器上新增的代码复制到本地文件后，右键冲突文件选择标记为合并即可，此时该文件就会消失在Synchronize视图下，并且该文件的代码已经成功将本地代码和远程代码合并到一起

## 提交项目

提交：将合并好的，并能够使项目顺利运行的版本提交到svn，他人再根据你提交的版本开发，保证代码不会冲突。

## 还原文件到历史版本

1、右键点击指定的文件，在弹出的关联菜单中点击Team->显示资源历史记录

2、此时会显示当前文件的所有历史版本列表，右键点击指定的版本，在弹出的关联菜单中点击"获取内容"，即可将本地文件替换为该版本的文件，最后直接点击"提交"即可。