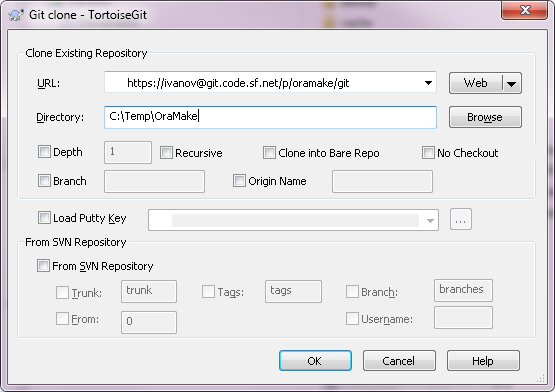
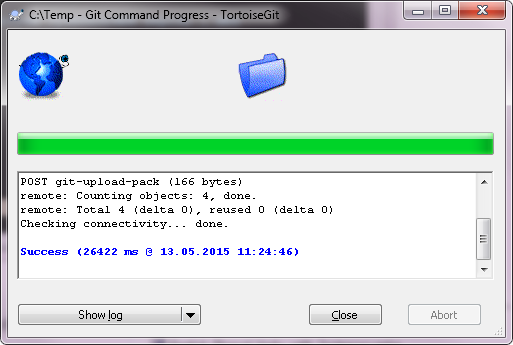
В каждом пункте есть вариант для командной строки git; далее следуют эквивалентные действия, которые можно проделать с помощью визуального клиента TortoiseGit.

1. Получаем git-репозитарий проекта ( ivanov - пользователь SF, под которым будет выполняться работа):

git clone https://ivanov@git.code.sf.net/p/oramake/git OraMake

****

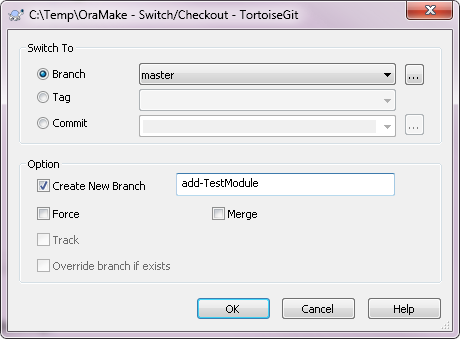
После выполнения клонирования должно появится окно вида:



2. Создаем ветку для добавления модуля TestModule и переходим на нее:

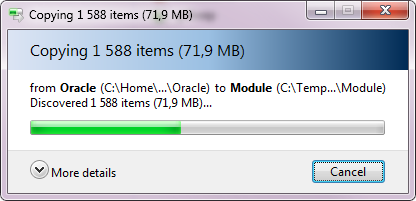
git checkout -b add-TestModule

P.S. По идее ветку для одного commit можно было бы не создавать, однако это иллюстрирует правильный подход к git, когда коммиты в основную ветку делаются в исключительных случаях.



3. Копируем файлы с текущим состоянием модуля (check-out Trunk модуля в Subversion) в подкаталог с именем модуля ( Module/TestModule).

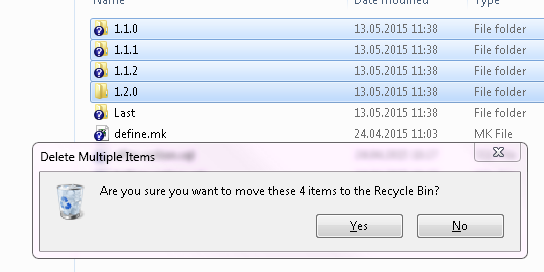
cp -R ../TestModule Module



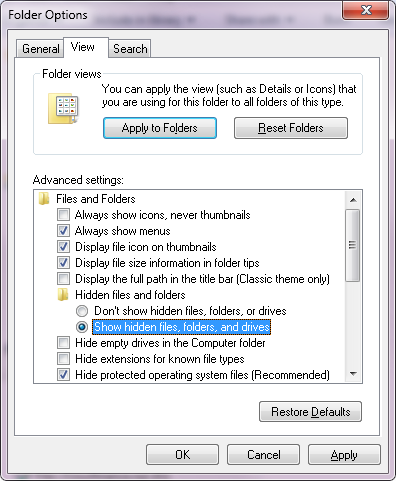
4. Удаляем файлы svn ( в директории .svn) и другие лишние файлы и директории.

rm -rf .svn App Doc DB/OmsModule

rm -rf DB/Install/Schema/\*.\*.\* DB/Install/Config/\*.\*.\* DB/Install/Grant/\*.\*.\*



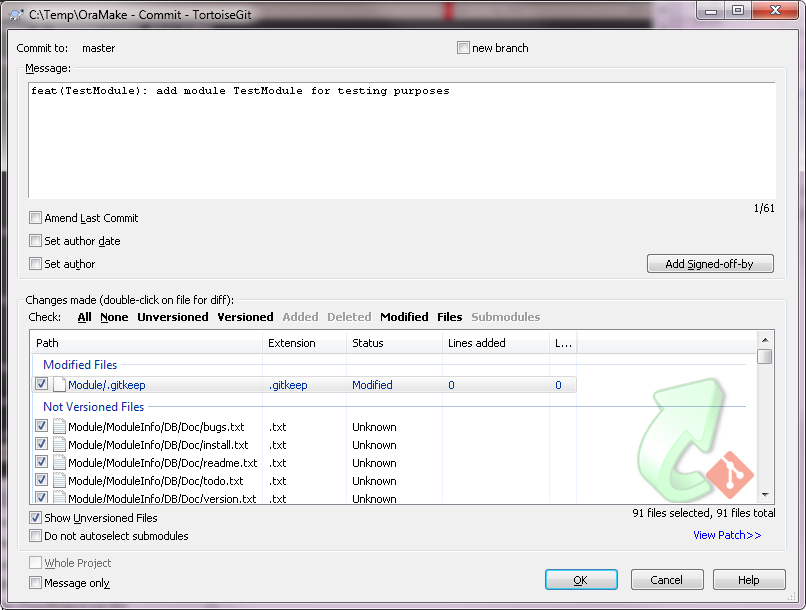
Нужно, чтобы была включена опция каталога “Show Hidden Files”.Tools -> Folder Options -> View



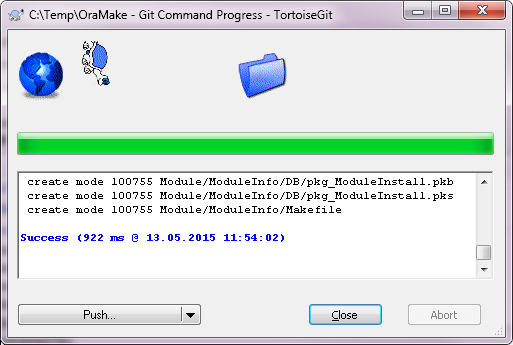
5. Коммитим файлы модуля

git add .

git commit -m 'feat(TestModule): add module TestModule for testing purposes'



В диалоговом окне нужно отметить файлы “All”.



6. Переключаемся на ветку master и подливаем в нее изменения из add-TestModule ( т.к. ветка add-TestModule не публиковалась, подливаем с помощью rebase):

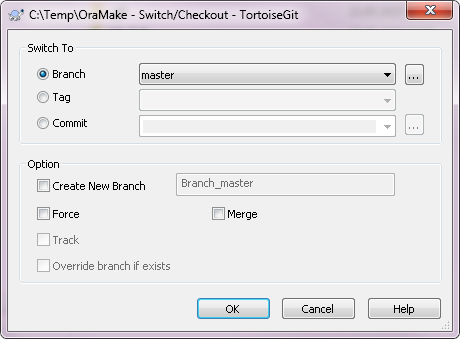
git checkout master

git rebase add-TestModule

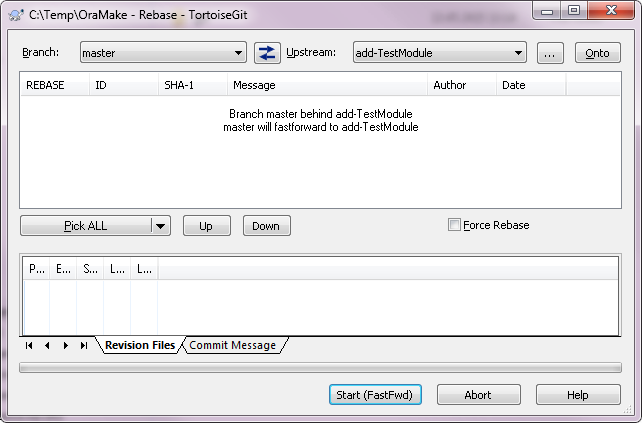
Если все ок, можно удалить локальную ветку:

git branch -d add-TestModule

Выбираем в контекстном меню Switch/Checkout…

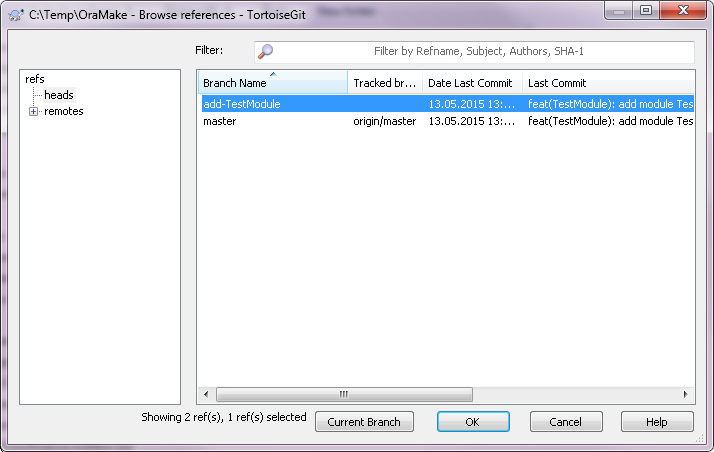


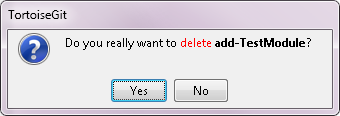
Выбираем в контекстном меню Rebase… Выбираем add-TestModule в качестве Upstream.



В контекстном меню выбрать Browse references…

И удалить ветку add-TestModule с помощью пункта контекстного меню “Delete branch”.





7. Заливаем изменения на SF.

git push

