This post was published to 周国通 at 1:13:37 PM 5/23/2018

Jenkins结合.net平台工具之Opencover

首先先介绍一下Opencover是什么,Opencover是.net平台下用于生成单元测试覆盖率报告的一款软件,是.net平台下为数不多的一款免费覆盖率报告工具,类似的工具还有Jetbrains的dotcover以衣VisualStudio自带的单元测试覆盖率工具,dotcover是付费软件,visualStudio自带的单元测试覆盖率报告工具仅限于在VisualStudio中使用,无法与Jenkins结合实现自动化,并且它只集成在企业版里面,社区版里是没有这个功能的.

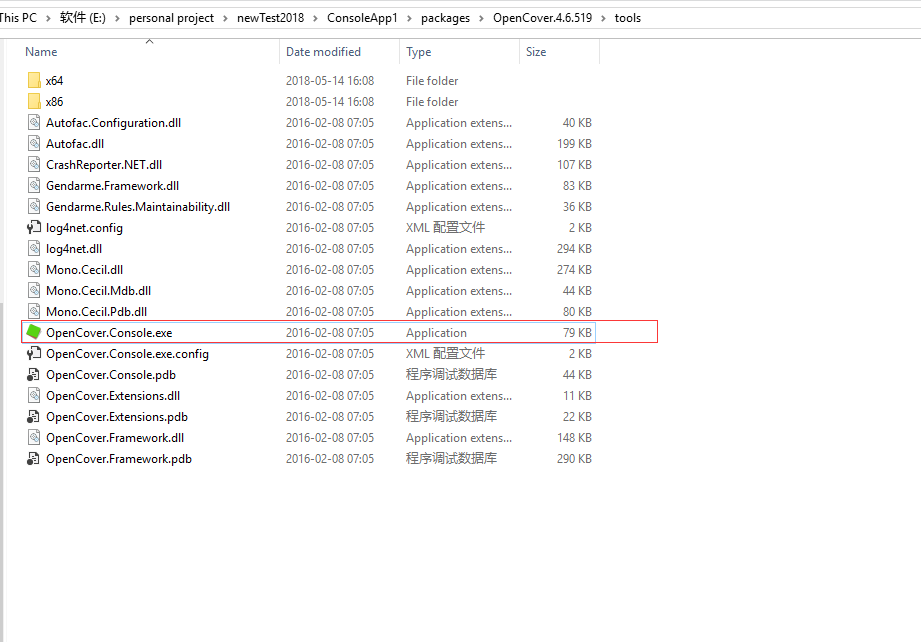
以下是Opencover的github仓库地址

<https://github.com/OpenCover/opencover>

下面讲解一下如何使用Opencover生成单元测试覆盖率报告.

这里我们仍然基于上一节的项目,这里需要引入另一个Nuget包,opencover

添加引用完成以后项目的packages目录下面有这样一个目录packages\OpenCover.xxx \tools(xxx为版本号)下面有一个OpenCover.Console.exe文件,它是我们与Jekins结合生成单元测试代码覆盖率报告要使用到的.



单元测试代码覆盖率报工具,顾名思义它是对单元测试进行分析生成报告的工具,因此它是依赖于单元测试的,我们使用的是Nunit,上一节我们通过命令生成了单元测试,这里我们把上一节的命令写成一个bat批处理文件,以供opencover.console.exe调用

代码如下

"E:\personal project\newTest2018\ConsoleApp1\packages\NUnit.ConsoleRunner.3.8.0\tools\nunit3-console.exe" "E:\personal project\newTest2018\ConsoleApp1\NunitTest\bin\Debug\NunitTest.dll" --work:"E:\personal project\newTest2018\ConsoleApp1\NunitTest\bin\Debug"

这里需要注意的是请不要复制上面的路径,而是根据自己的实际情况(Nunit控制台工具所在位置,要进行的项目生成的dll所在位置以及工作目录)

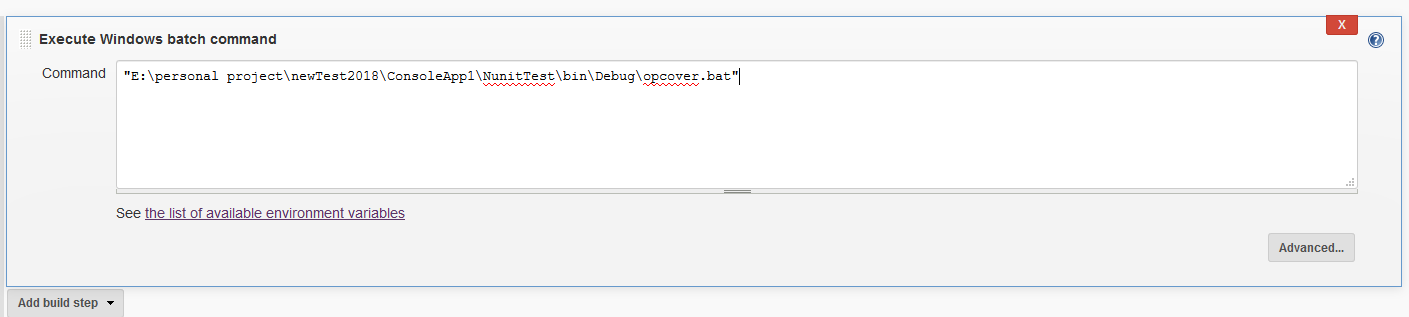
我们把这个bat命名为runnuit.bat(注意名称可以随便起,不必和示例一样)

我们把opencover要执行的命令也放在bat文件里,这里命名opcover.bat

"C:\Users\tylerzhou\.nuget\packages\opencover\4.6.519\tools\OpenCover.Console.exe" -target:"E:\personal project\newTest2018\ConsoleApp1\NunitTest\bin\Debug\runnuit.bat" -targetargs:"/nologo /noshadow" -targetdir:"E:\personal project\newTest2018\ConsoleApp1\NunitTest\bin\Debug" -register:"user" -filter:"+[NunitTest\*]\*" -output:"E:\personal project\newTest2018\ConsoleApp1\NunitTest\bin\Debug\results.xml"

这里对以上代码的关键部分进行说明:首先第一个空格前是Opencover.Console.exe文件所在的位置,用于执行它. Target是它要运行的单元测试命令目标,是一个批处理文件,我们使用刚才创建的runnuit.bat文件(这里我们都使用的是绝路径,以防对Jenkins不熟悉造成各种不必要麻烦).targetdir则为工作目录, output用于指定生成的覆盖率报告文件所在的位置. –filter用于过滤要进行单元测试覆盖的结果.如果不指定,它会对加载的Nunit库里的代码也进行覆盖率测试,这不是我们想要的.我们目前只测试了NunitTest.dll这个文件,大家可以把它替换为自己的文件名.

下面我们新建一个名为OpencoverTest的freestyle项目,在Build栏里添加一个windows batch command步骤,然后在命令里面指定本地opcover.bat文件所在路径.

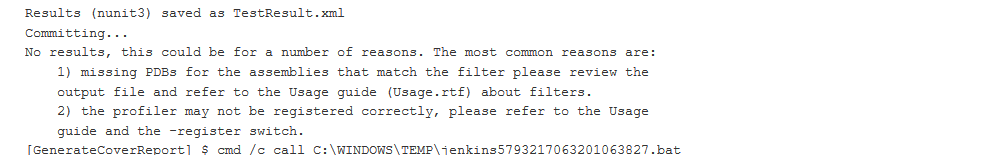


需要注意的时你可能看到我们实际上创建了两个bat,一个是runnuit.bat另一个是opcover.bat这里我们并没有在步骤里面添加runnuit.bat这是因为opcover.bat的target参数里面引用了它,当opcover.bat执行的时候会先执行runnuit.bat.

添加命令以后,我们点击保存,然后执行Build Now.

如果配置成功,则将会生成一条成功构建记录.下面介绍一个坑:

如果你仔细查看Console Out,会发现有如下类似语句



其中Results(nunit3)saved as TEstResult.xml是runnuitt.bat生成的日志文件,下面是opcover生成的,可以看到是No Resultsxxxxx

如果你在本地执行opcover.bat则报告和在Jenkins上生成的是不一样,本地是能成功执行的.这个问题经历了一翻折腾才解决.原因在于opcover在生成代码覆盖率报告的时候,要访问本地dll和pdb文件,

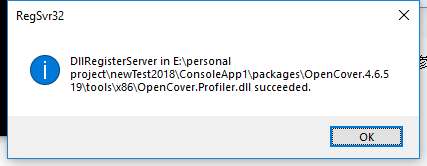
在本地执行的时候注册-register user是ok的,到了jenkins环境中却权限不足,猜测可能是Jenkins执行环境用户和本地默认执行可执行文件时的用户是不同的.用于对windows权限不是太清楚,这里仅是猜测.

下面说一下解决办法:

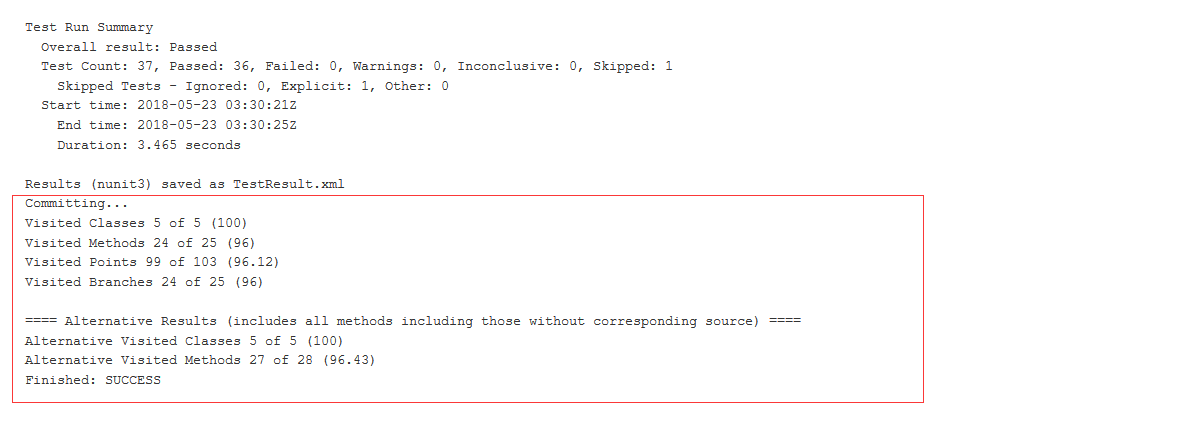
以管理员身份打开cmd,然后执行regsvr32 "E:\personal project\newTest2018\ConsoleApp1\packages\OpenCover.4.6.519\tools\x86\OpenCover.Profiler.dll"

需要注意的是OpenCover.Profiler.dll"在所在项目的packages目录下的opcoverxxx\tools目录下,大家的项目绝对路径可能和我的不一样,不要复制.

执行以上操作成功后会有弹窗提示成功.



然后打开Jenkins再次执行Build now,查看console out,将会看到类似如下日志



此时已成功生成报告.我们打开本地生成的results.xml就可以看到opencover生成的报告了