（一）CCNET配置

首先规划一下打包服务器上的目录结构：

1）C:\Source\存放每次自动打包时从SVN上获取到的源代码。

其中的子目录按照版本进行组织，如图8-3所示：

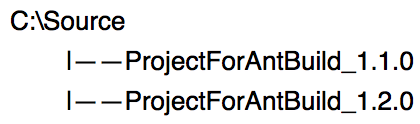


图8-3 Source目录下的子目录组织结构

ProjectForAntBuild\_1.1.0表示是1.1.0版本的源代码，以此类推。

2）C:\ProjectForAntBuild 这个目录要事先准备好，每次打包生成的apk都会放在这里。

其中的子目录按照版本进行组织，如图8-4所示：

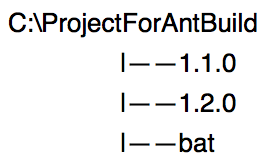


图8-4 ProjectForAntBuild目录下的子目录组织结构

点击1.1.0子目录，看到里面的组织结构，如图8-5所示：

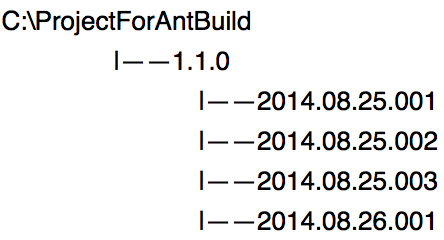


图8-5目录1.1.0下的子目录组织结构

再点击其中的一个子目录，就能看到每次打包的apk文件了，如图8-6所示：

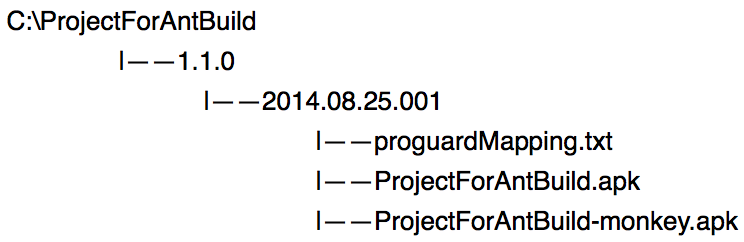


图8-6目录1.1.0下的某个子目录的组织结构

3）C:\ProjectForAntBuild\bat，这个目录下存放的是CCNET每次打包时执行的bat脚本文件。前面介绍过，CCNET通过执行bat脚本来间接执行Ant脚本。

因为每个版本自动打包时获取的源代码都是按照版本号来存放的，所以不同版本都有一个自己的bat文件，如图8-7所示：

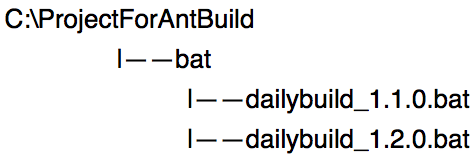


图8-7目录bat下的可执行文件列表

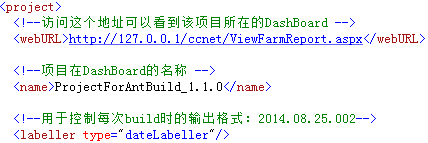
4）C:\BuildTask\Android\buildlogs，里面存放每次build的日志。其中的子目录按照版本进行组织。

有了上述的规划，下面介绍ccnet.config配置文件就简单多了：

ccnet.config由多个project标签组成，每个project代表一个项目，对于我们的App来说，不同的版本就是不同的项目。

Project标签下有几个重要的属性，和一些重要的组件。

1）我们先介绍属性，如下所示，对属性的说明混排在注释中：



你可以访问webURL中的地址，如图8-7所示，我们称之为Dashboard：

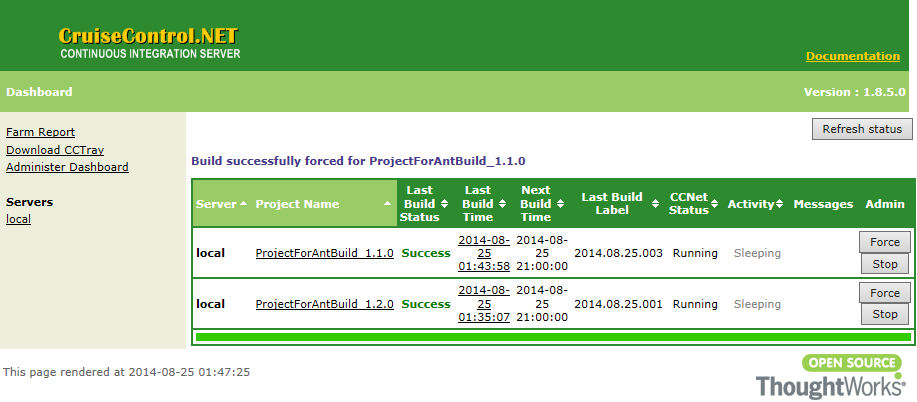
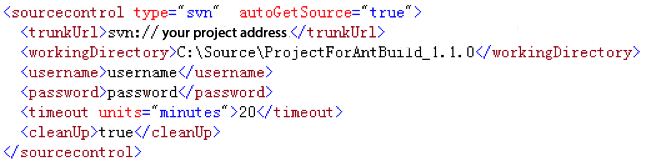


图8-7 CCNET首页

右下方显示的上当前的project列表，点击Force按钮可以立刻就开始进行build。

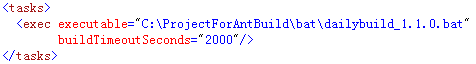
注：服务器端一定要确保CruiseControl.NET这个控制台程序是开启的，才能保证用户端能正常运行上述网站并且build成功。切记切记，我当初曾因为这个问题查了半天而不得要领。

2）sourcecontrol组件，用于自动从SVN服务器上获取最新代码，放到workingDirectory这个工作目录下。

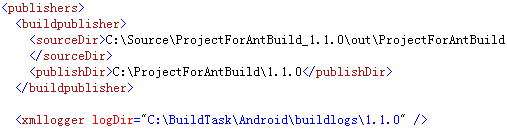


此外，CCNET还支持GIT或者VSTF，请大家查看CCNET使用手册自行研究。

3）tasks组件，也就是每天自动build或者手动点击Force按钮所执行的程序，由于CCNET1.8.5版本目前还不支持直接执行Ant脚本，所以我们编写了一个bat文件，CCNET执行bat，然后由bat执行Ant脚本。

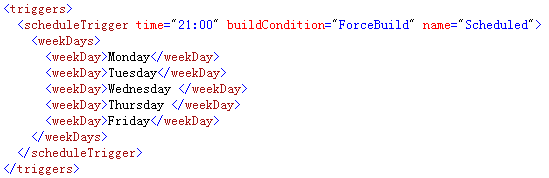


4）publishers组件，在build完成后，把sourceDir目录中的apk包复制到publishDir目录中。



此外，还通过xmllogger标签，指定了每次build的日志输出目录。

5）triggers组件，设置触发build的条件。以下是工作日每天晚上9点来执行build代码：



还可以设置以下触发器：

* 每次checkIn自动触发build。
* 每隔多少秒执行一次build。

请自行去查CCNET的使用手册。

6）email组件，发送每次build的结果，无论成功与否都会发。

这里的配置分为3部分：

1. 谁来发

正规起见，作为企业级开发项目，我们还是要专门申请个公司邮箱，比如AndroidBuild@abc.com，由它来作为邮件发起者。



2. 发给谁

这里就涉及到Group的定义了。我们一般会定义2种Group：

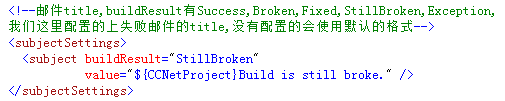
* buildmaster，总是会收到所有build结果的邮件。
* developers，只会收到某些情况下的build结果的邮件，比如build失败。



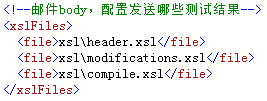
3. 邮件格式[[1]](#footnote-1)

这里涉及到3部分内容：

* 邮件标题



* 邮件主体



* 邮件附件，有同学会在这里把build时的日志文件发出来，也有同学会把两次build之间的代码签入历史放在一个mod文件中，然后作为附件发出来。[[2]](#footnote-2)

（二）搭建IIS站点

在IIS站点创建一个虚拟Web站点，指向打包生成的apk所在的文件夹，对于我们的项目而言，是C:\ProjectForAntBuild。如图8-9所示：

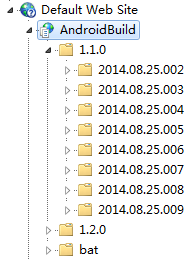


图8-9 IIS站点下可浏览到的子目录

这样我们就可以输入以下地址访问每日自定打包的目录了（假定服务器的地址是192.168.1.14）：

http://192.168.1.14/androidbuild/

在手机上一路点下去，就能看到提供apk下载的页面，如图8-10所示：

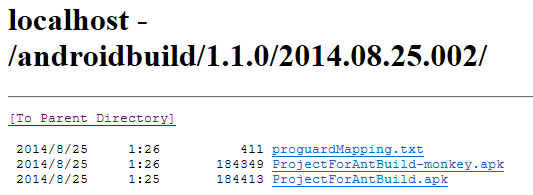


图8-10 使用IIS搭建的apk下载站点

注：一定要开启IIS的目录浏览功能，才能在网页上浏览Web站点的目录。如图8-11所示，双击“目录浏览”这个icon：

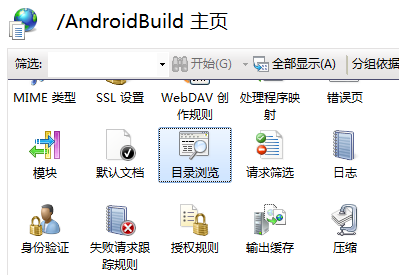


图8-11 在IIS配置页面进入“目录浏览”配置页面

进入“目录浏览”的配置页面，点击右侧的“启动”即可，如图8-12所示：

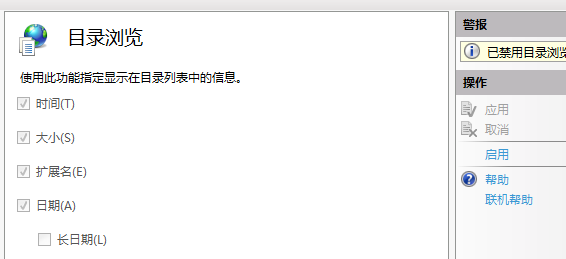


图8-12 在IIS中启动“目录浏览”功能

此外，还要支持支持apk下载，在IIS管理器中点击MIME类型，在里面添加.apk这个后缀，如图8-13所示：

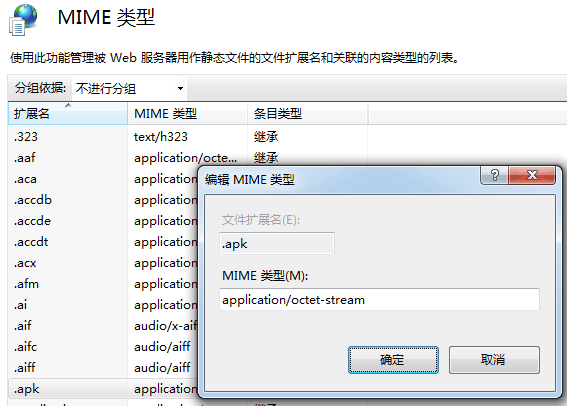


图8-13 在IIS中开启apk下载功能

1. 对于不同角色接收不同build结果的邮件的策略，请参考博客园CoderZh的这篇文章：《CCNET+MSBuild+SVN实时构建的优化总结》，地址如下：

   http://www.cnblogs.com/coderzh/archive/2009/04/05/1429858.html [↑](#footnote-ref-1)
2. 关于CCNET的更多配置信息，请参见博客园海华的这篇文章：http://www.cnblogs.com/harveyChoi/archive/2013/07/22/hello\_cc\_net.html [↑](#footnote-ref-2)