# Ant 入门学习教程及实例

1. Ant应用场景，优点
2. Ant在PC上的部署
3. Ant详细解析----属性的介绍及使用
4. Android打包实例

**属性的介绍及使用**

project

target 继承关系

antcall

date type

arg参数的形式 p85

available

time

exec

condition

1. 一个简单的Ant例子

Apache Ant，是一个将软件编译、测试、部署等步骤联系在一起加以自动化的一个工具，大多用于Java环境中的软件开发。

本人使用Ant的场景是拿它在后台执行编译、打包Android工程的操作，在Ant工作期间，仍可正常使用Ecelipse，避免痛苦的等待。

在使用之前应先[下载Ant](http://ant.apache.org/bindownload.cgi)并部署执行环境，细节上来说就是个添加系统环境变量的过程，具体可参考：[ant安装、环境变量配置及验证](http://www.cnblogs.com/yuzhongwusan/archive/2013/03/26/2982411.html)。

下面直接给出一个Ant的例子：

<?xml version="1.0" encoding="UTF-8"?>

<project name="Hello\_World" default="Echo" basedir=".">

<target name="Echo" description="Just print 'Hello world'''.">

<echo message="Hello world! I'm ant."></echo>

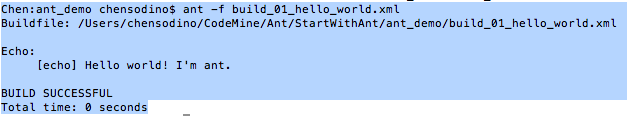
</target>

</project>

Ant并没有定义它自己的语法，它的执行文件用XML编写。每个Ant文件由单个的project元素组成，该元素可以包含一个或多个target元素。一个目标(target)是生成过程中已定义的一个步骤，可以定义任意数量的操作，比如编译源代码，对Apk进行签名等。这些操作本身由其他专用任务(Task)标签执行。通常为了编写的灵活性，把数量较多的操作根据其业务逻辑划分到单独的目标(target)元素中，这样可以有选择性的执行整体生成过程中的单独部分。例如，可以调用某些目标元素，直接对安装包进行重新签名，而不需要从头执行编译操作。

以上代码存储在文件build\_01\_hello\_world.xml中。执行该文件则需命令：

ant –f build\_01\_hello\_world.xml，运行效果如下：



执行命令后，第一行显示编译的文件绝对路径;空格一行后的“Echo:”则表示当前执行的target名为“Echo”;“[echo]”表示当前输出是由Ant的echo任务(Task)发起的。