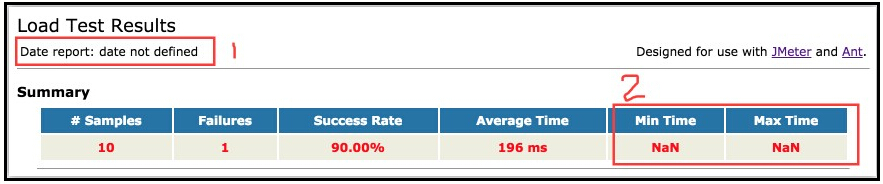
**一、本文目的：**  
　　之前写了两篇文章搭建持续集成[**接口测试**](javascript:;)平台(Jenkins+Ant+Jmeter)和ANT批量执行Jmeter**[脚本](javascript:;" \t "_self)**，功能实现上都没有什么问题，但是最后生成的报告有一点小问题，虽然不影响使用，但总是感觉不够完美，具体问题如下：  
　　1、Date report这里的时间没有正确显示出来  
　　2、Summary里的字段Min Time和Max Time显示的是NaN，没有显示正确的时间。



　　本文主要解决上述两个问题，具体报告如何生成可参考上面提到的两篇文章。  
**二、Jmeter+Ant报告生成原理：**  
　　在解决问题之前，让我们先弄清楚Jmeter+Ant是生成报告的原理，知道原理后我们就可以很从容的解决问题了。另外，如果后续我们想定制报告也就很容易了。  
　　1、在Jmeter的extras目录下，官方已经为我们提供了一个现成的实例，我们只需要在该目录下执行ant命令就可以生成一个数据文件Test.jtl和一个报告Test.html（PS：如果没有安装ant需要先去安装，不会用ant的自行百度）。  
　　2、知道了上一步后，接下来我们研究的重点就是extras目录下的build.xml文件了，具体研究细节就不在此细说了，因为这就部分内容属于ant的范围，有需要的可以自行补充下ant的知识。  
**三、解决问题：**  
　　1、找到extras目录下build.xml文件内容的如下部分：它的作用就是把生成的.jtl数据文件转换为.html格式的报告  
　　具体属性的含义：  
　　in：生成的.jtl文件的路径  
　　out：生成的.html文件的路径  
　　style：从.jtl转换成.html所使用的样式，这些文件也在extras目录下，这个是最重要的部分，如果我们想定制报告可以自行定义这个文件或者从网上找一些合适的文件。  
　　param：参数，这几个参数可以在style中使用，下面会讲到。  
　　path：添加jar包的路径，在下面的target中有引用到（下面第9行）。  
1     <path id="xslt.classpath">//jar的路径  
2         <fileset dir="${lib.dir}" includes="xalan\*.jar"/>  
3         <fileset dir="${lib.dir}" includes="serializer\*.jar"/>  
4     </path>  
5 　　 ／／把生成的.jtl数据文件转换成.html格式的报告  
6     <target name="xslt-report" depends="\_message\_xalan">  
7         <tstamp><format property="report.datestamp" pattern="yyyy/MM/dd HH:mm"/></tstamp>  
8         <xslt  
9             classpathref="xslt.classpath"  
10             force="true"  
11             in="${testpath}/${test}.jtl"  
12             out="${testpath}/${test}.html"  
13             style="${basedir}/jmeter-results-detail-report${style\_version}.xsl">  
14             <param name="showData" expression="${show-data}"/>  
15             <param name="titleReport" expression="${report.title}"/>  
16             <param name="dateReport" expression="${report.datestamp}"/>  
17         </xslt>  
18     </target>  
　　2、通过上面的分析，style 文件是整成报告转换的关键，那我们就打开jmeter-results-detail-report\_21.xsl这个文件看看，找到如下几行，这里也定义了三个参数跟build.xml文件中一样，由此可猜测，我们只要在build.xml文件中定义dateReport这个参数和值，在style中就可以按如下方式使用，经过测试证明猜测是正确的，那么第一个问题就解决了：  
　　1 <xsl:param    name="showData" select="'n'"/>  
　　2 <xsl:param    name="titleReport" select="'Load Test Results'"/>  
　　3 <xsl:param    name="dateReport" select="'date not defined'"/>  
　　3、对于第二个问题，其实把第1步中的代码复制到我们自己的build.xml文件中后，可以一并解决这两个问题。解决第二个总是的关键代码是第9行和第10行：  
　　classpathref="xslt.classpath"  
　　force="true"  
　　而第9行又引用了第1~4行  
　　4、到此，两个问题都解决了，最终修改后的build.xml文件如下，标黄是解决总是添加的代码：  
<?xml version="1.0" encoding="UTF-8"?>  
<project name="ant-jmeter-test" default="run" basedir=".">  
<!-- 需要改成自己本地的 Jmeter 目录-->  
<property name="jmeter.home" value="/Users/Tools/Jmeter" />  
<property name="report.title" value="接口测试"/>  
<!-- jmeter生成jtl格式的结果报告的路径-->  
<property name="jmeter.result.jtl.dir" value="/Users/Desktop/jmx/report" />  
<!-- jmeter生成html格式的结果报告的路径-->  
<property name="jmeter.result.html.dir" value="/Users/Desktop/jmx/report" />  
<!-- 生成的报告的前缀-->  
<property name="ReportName" value="TestReport" />  
<property name="jmeter.result.jtlName" value="${jmeter.result.jtl.dir}/${ReportName}.jtl" />  
<property name="jmeter.result.htmlName" value="${jmeter.result.html.dir}/${ReportName}.html" />  
<target name="run">  
<antcall target="test" />  
<antcall target="report" />  
</target>  
<target name="test">  
<taskdef name="jmeter" classname="org.programmerplanet.ant.taskdefs.jmeter.JMeterTask" />  
<jmeter jmeterhome="${jmeter.home}" resultlog="${jmeter.result.jtlName}">  
<!-- 声明要运行的脚本"\*.jmx"指包含此目录下的所有jmeter脚本-->  
<testplans dir="/Users/Desktop/jmx" includes="\*.jmx" />  
<property name="jmeter.save.saveservice.output\_format" value="xml"/>  
</jmeter>  
</target>  
<path id="xslt.classpath">  
<fileset dir="${jmeter.home}/lib" includes="xalan\*.jar"/>  
<fileset dir="${jmeter.home}/lib" includes="serializer\*.jar"/>  
</path>  
<target name="report">  
<tstamp> <format property="report.datestamp" pattern="yyyy/MM/dd HH:mm" /></tstamp>  
<xslt  
classpathref="xslt.classpath"  
force="true"  
in="${jmeter.result.jtlName}"  
out="${jmeter.result.htmlName}"  
style="${jmeter.home}/extras/jmeter-results-detail-report\_21.xsl">  
<param name="dateReport" expression="${report.datestamp}"/>  
</xslt>  
<!-- 因为上面生成报告的时候，不会将相关的图片也一起拷贝至目标目录，所以，需要手动拷贝 -->  
<copy todir="${jmeter.result.html.dir}">  
<fileset dir="${jmeter.home}/extras">  
<include name="collapse.png" />  
<include name="expand.png" />  
</fileset>  
</copy>  
</target>  
</project>  
　　5、修改后的报告：

