

Getting Started With DSB

Version: 3.5

2015.11.26

(Binary Guy)

目录

一、	引言	2
二、	搭建开发环境.....	3
三、	编译 DSB 核心包.....	9
四、	编译 DSB 客户端包	10
五、	DSB 内核的启动	10

一、 引言

这篇文档将介绍如何基于 Eclipse 搭建 DSB 开发环境, 然后介绍如何编译 DSB 核心包和 DSB 客户包。

注意: DSB 核心包通常运行在服务器端软件中, 而 DSB Client 包是客户端远程调用服务端时需要引入的包。

这里假定你已经从 Git Server 上把代码 clone 到你的计算机上了, 其目录假定为:

C:\workspace\dev\dsb-opensource

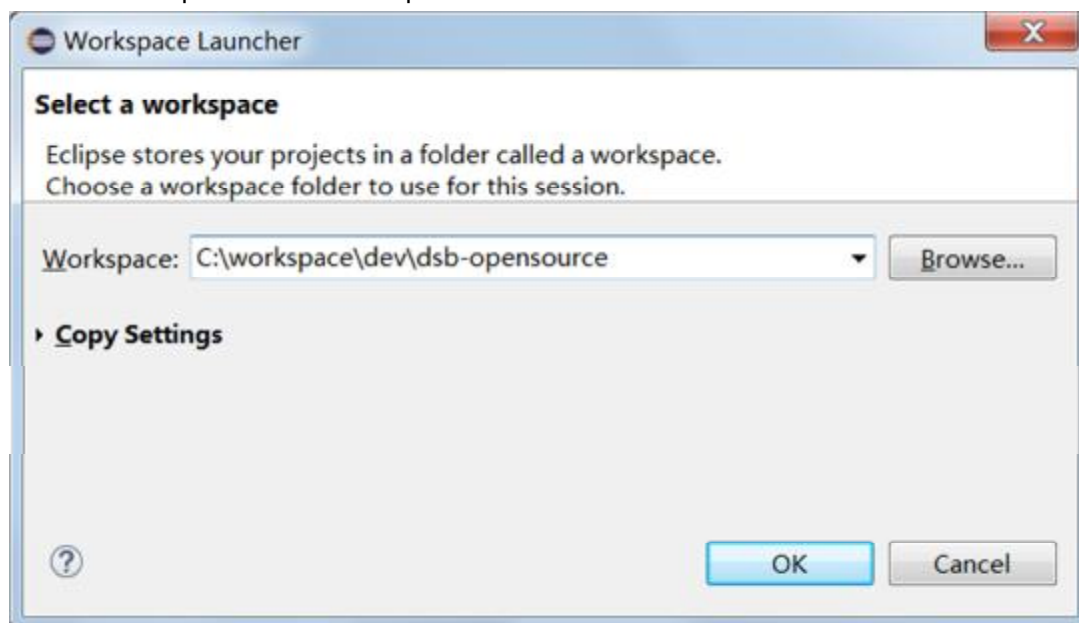
注意: OSChina 开源中国社区上 GIT Server 地址为:

http://git.oschina.net/binary_guy/dsb

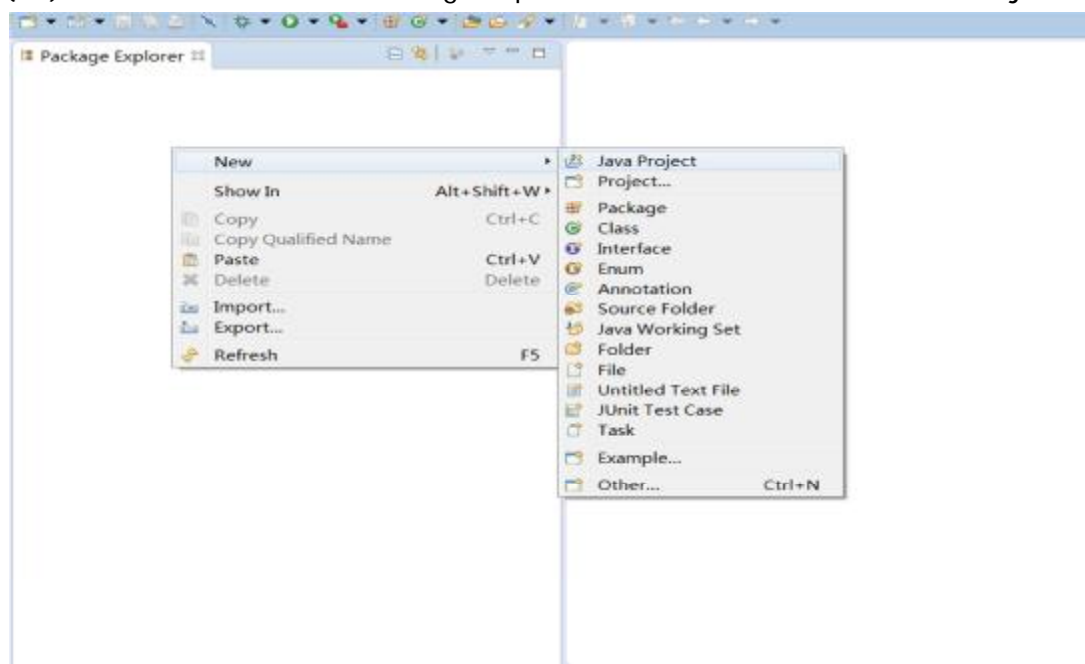
二、 搭建开发环境

(一) 打开 Eclipse, 把工作目录切换到目录:

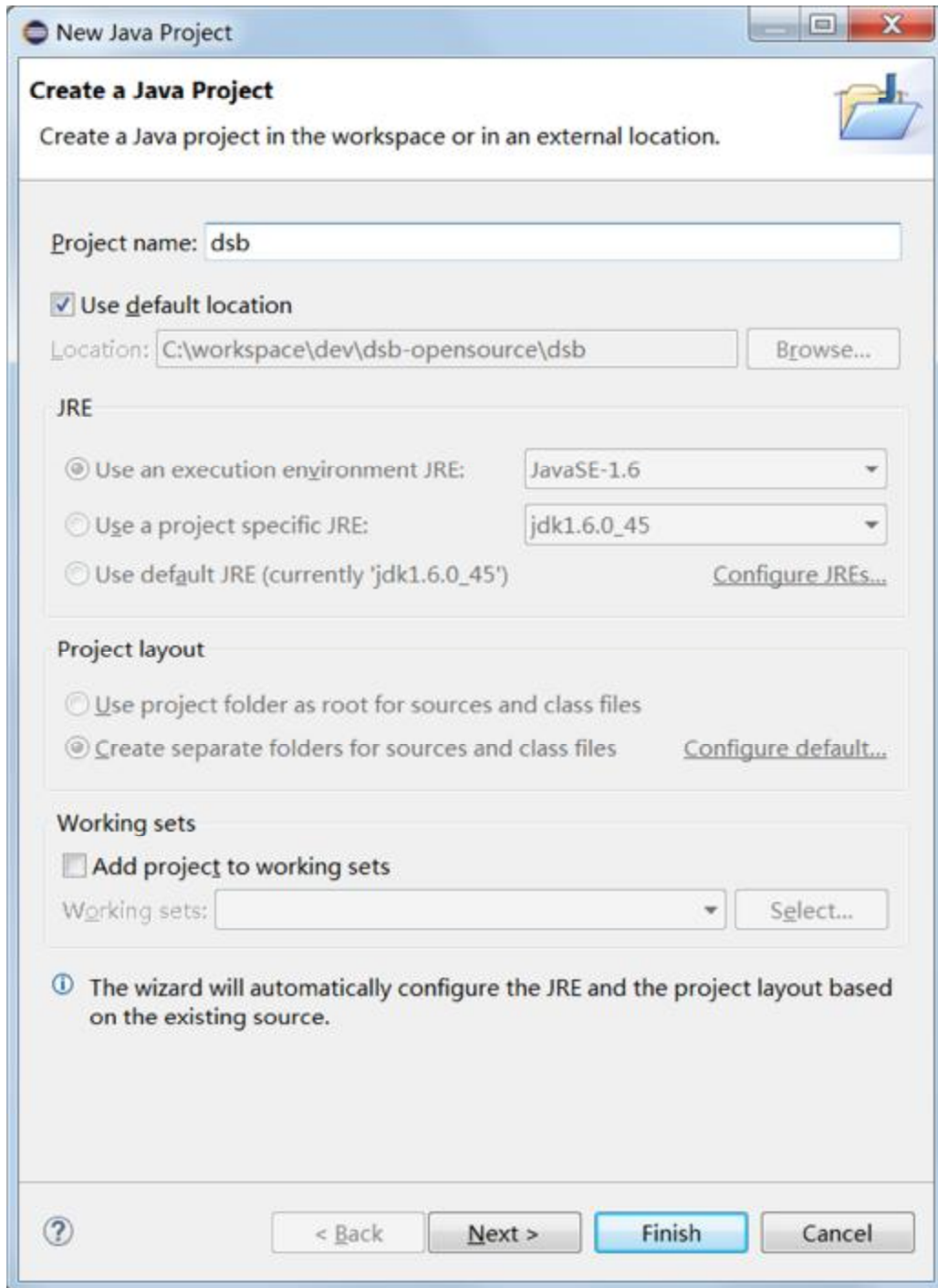
C:\workspace\dev\dsb-opensource



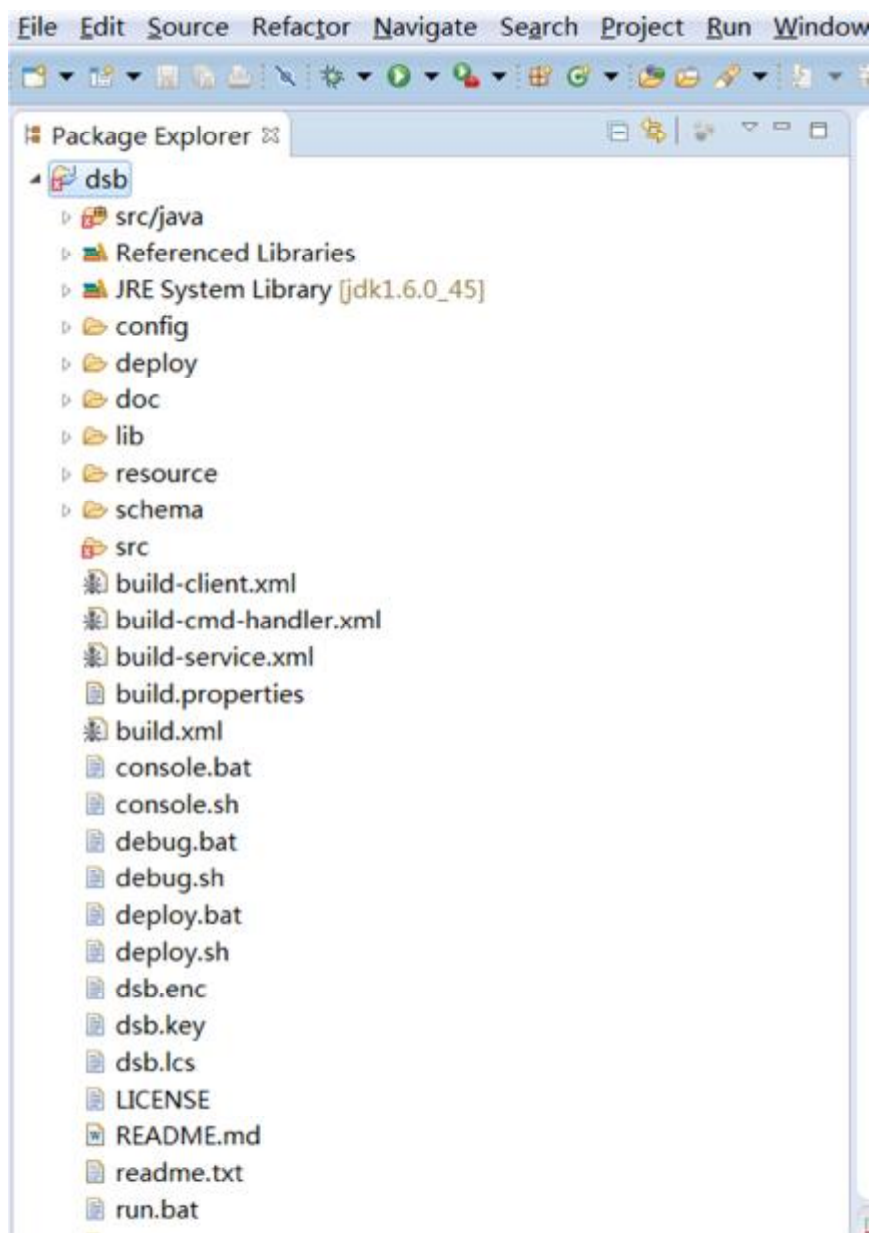
(二) 鼠标右击左边的【Package Explorer】, 选择【New】-> 【Java Project】



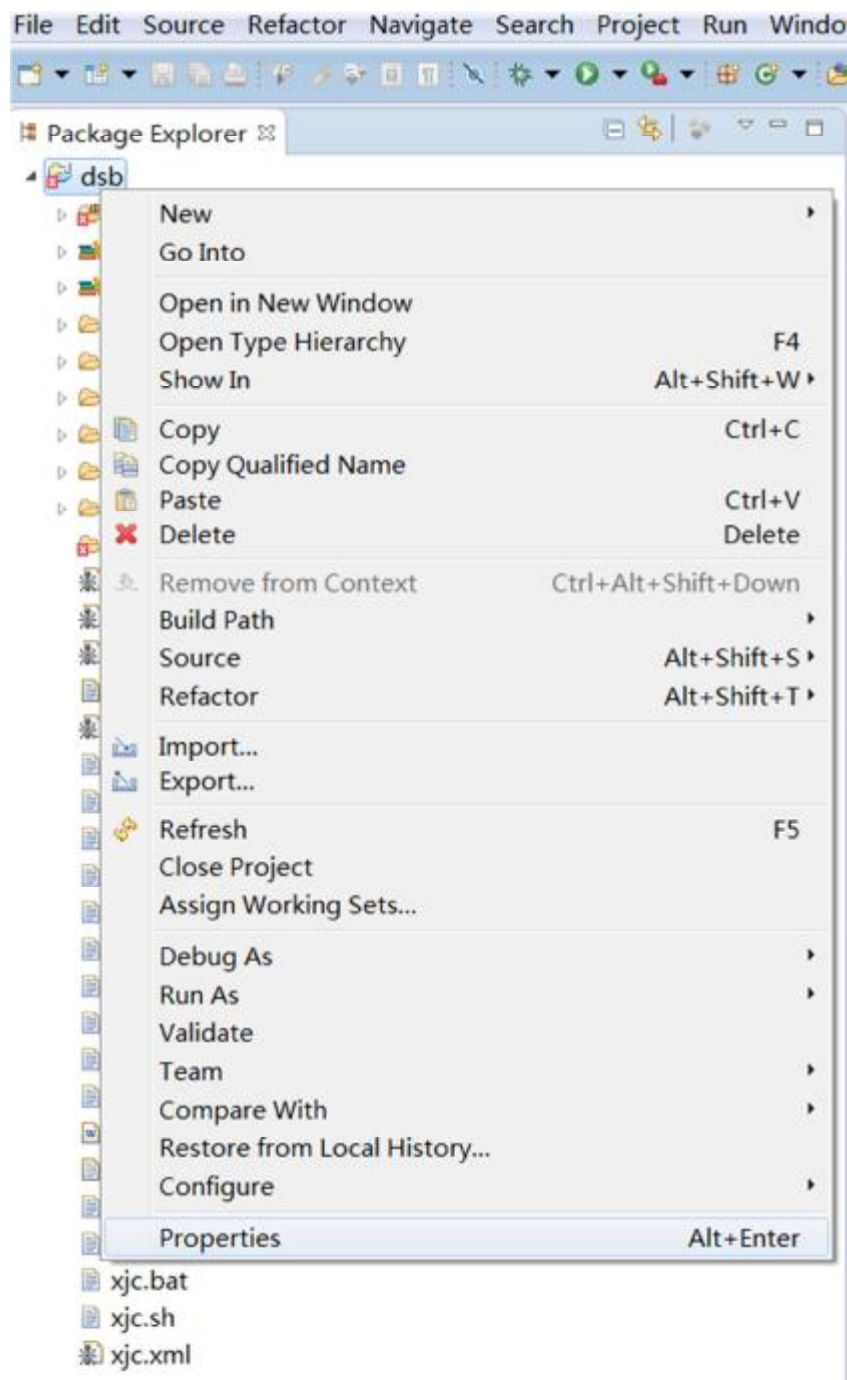
(三) 在弹出的【New Java Project】窗口中输入【dsb】，然后点击【Finish】



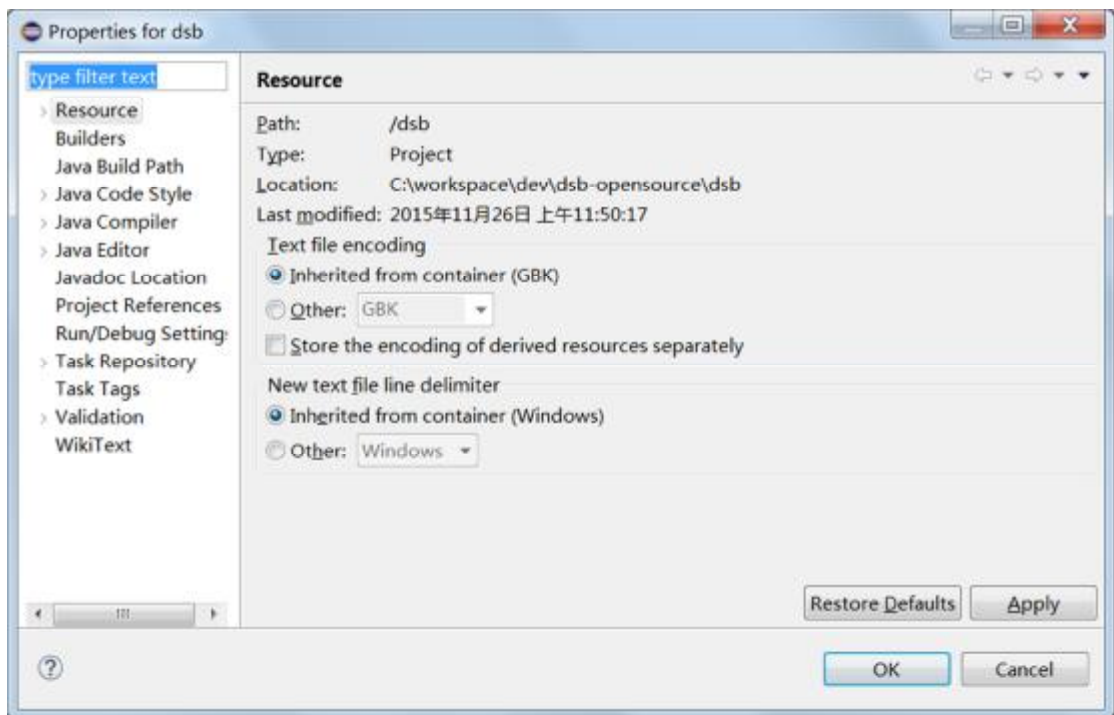
(四) 操作完成之后，你会看到如下示意图



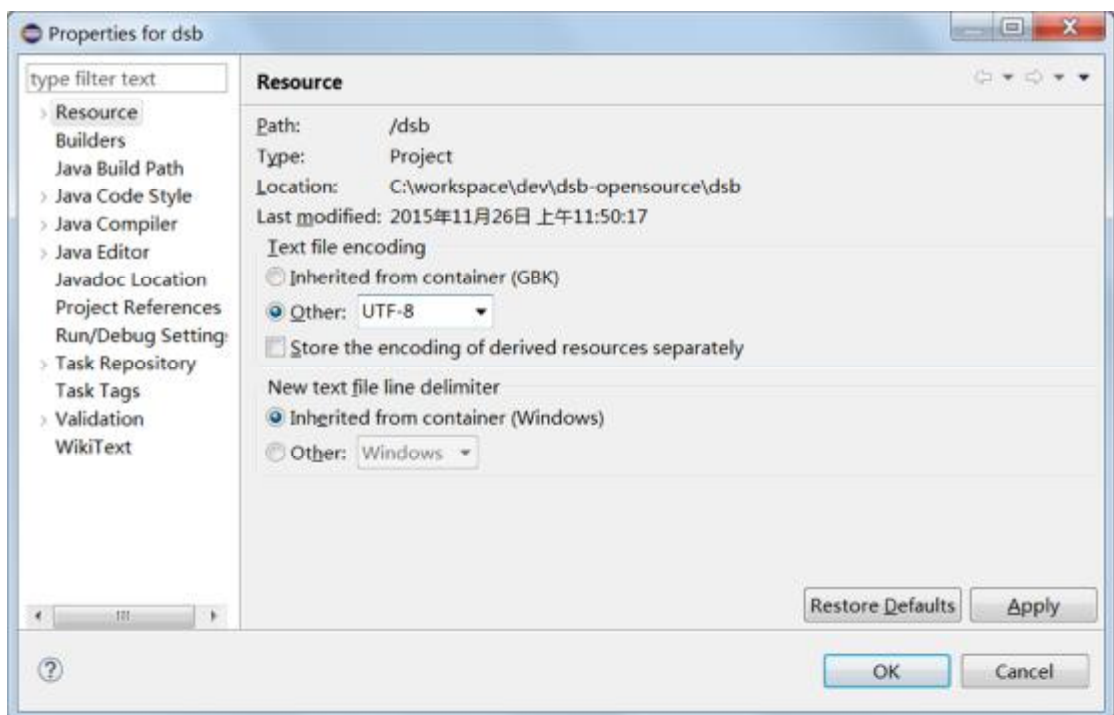
(五) 项目会红色错误提示，在项目名称上右键点击，弹出的窗口中，点击【Property】



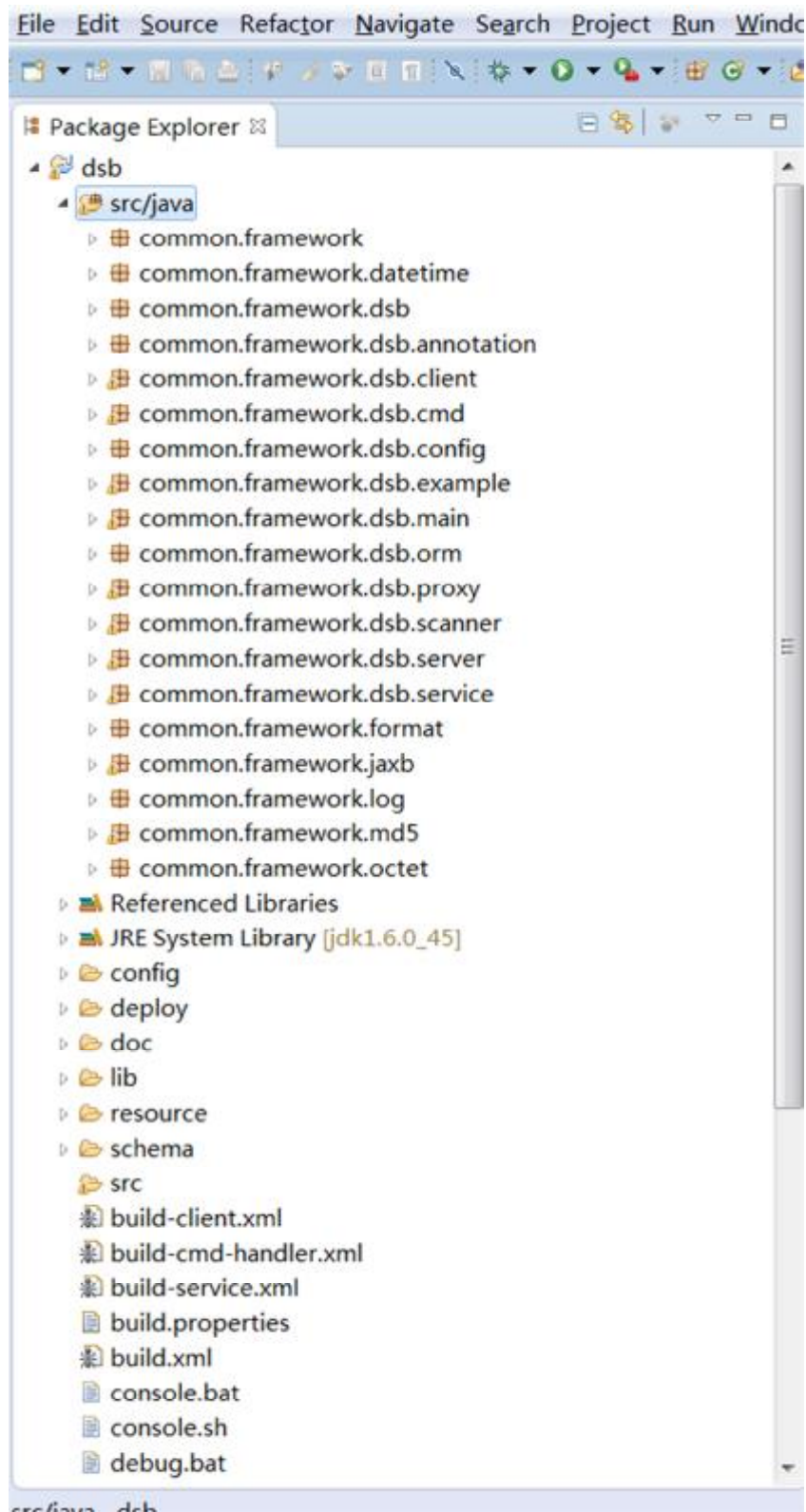
(六) 此时弹出项目的属性窗口如下



(七) 在【Text File Encoding】中选择【UTF-8】，然后点击【OK】



(八) 接下来项目应该不会有错误提示了，如下图：

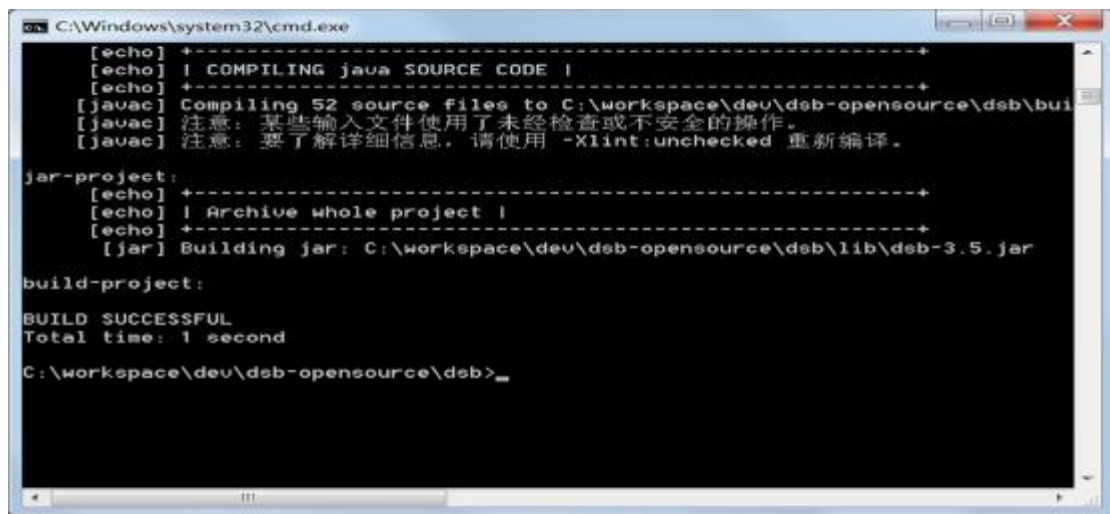


三、 编译 DSB 核心包

关于 JDK 以及 Apache Ant 的配置，网络上和很多 Java 书籍中都有介绍。这里假定 JDK 和 Apache Ant 都已经配置完毕，并且可以通过命令行执行命令 `java`, `javac`, `ant` 命令。

打开命令行窗口，进入目录：

```
C:\workspace\dev\dsb-opensource\dsb
```



```
C:\Windows\system32\cmd.exe
[echo] +-----+
[echo] | COMPILING java SOURCE CODE |
[echo] +-----+
[javac] Compiling 52 source files to C:\workspace\dev\dsb-opensource\dsb\build
[javac] 注意：某些输入文件使用了未经检查或不安全的操作。
[javac] 注意：要了解详细信息，请使用 -Xlint:unchecked 重新编译。

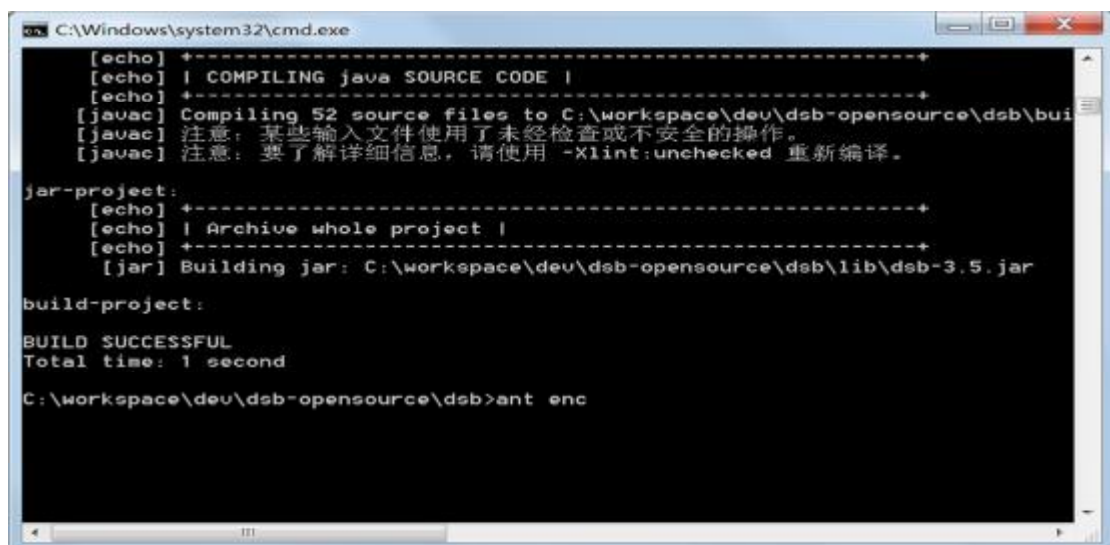
jar-project:
[echo] +-----+
[echo] | Archive whole project |
[echo] +-----+
[jar] Building jar: C:\workspace\dev\dsb-opensource\dsb\lib\dsb-3.5.jar

build-project:
BUILD SUCCESSFUL
Total time: 1 second

C:\workspace\dev\dsb-opensource\dsb>
```

编译 DSB 框架核心包，只需要输入命令即可：

```
ant enc
```



```
C:\Windows\system32\cmd.exe
[echo] +-----+
[echo] | COMPILING java SOURCE CODE |
[echo] +-----+
[javac] Compiling 52 source files to C:\workspace\dev\dsb-opensource\dsb\build
[javac] 注意：某些输入文件使用了未经检查或不安全的操作。
[javac] 注意：要了解详细信息，请使用 -Xlint:unchecked 重新编译。

jar-project:
[echo] +-----+
[echo] | Archive whole project |
[echo] +-----+
[jar] Building jar: C:\workspace\dev\dsb-opensource\dsb\lib\dsb-3.5.jar

build-project:
BUILD SUCCESSFUL
Total time: 1 second

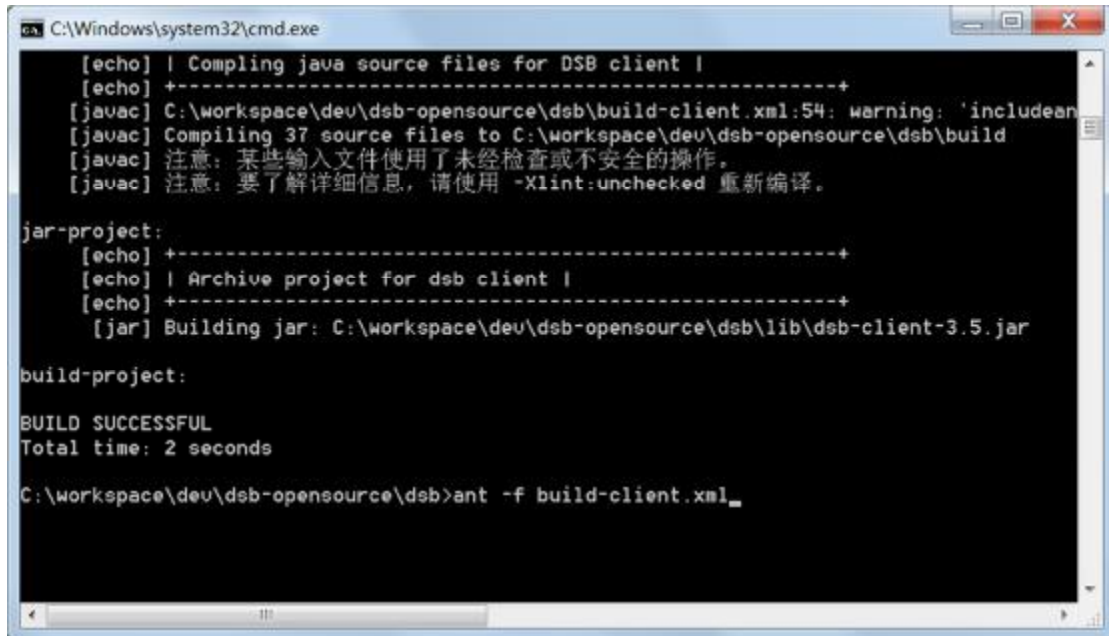
C:\workspace\dev\dsb-opensource\dsb>ant enc
```

编译完成之后，会在 `dsb/lib` 目录下生成 `dsb-3.5.jar`

四、 编译 DSB 客户端包

编译 DSB Client 包，只需要输入命令即可：

```
ant -f build-client.xml
```



```
C:\Windows\system32\cmd.exe

[echo] | Compiling java source files for DSB client |
[echo] +-----+
[javac] C:\workspace\dev\dsb-opensource\dsb\build-client.xml:54: warning: 'includean
[javac] Compiling 37 source files to C:\workspace\dev\dsb-opensource\dsb\build
[javac] 注意：某些输入文件使用了未经检查或不安全的操作。
[javac] 注意：要了解详细信息，请使用 -Xlint:unchecked 重新编译。

jar-project:
[echo] +-----+
[echo] | Archive project for dsb client |
[echo] +-----+
[jar] Building jar: C:\workspace\dev\dsb-opensource\dsb\lib\dsb-client-3.5.jar

build-project:

BUILD SUCCESSFUL
Total time: 2 seconds

C:\workspace\dev\dsb-opensource\dsb>ant -f build-client.xml_
```

编译完成之后，会在 dsb/lib 目录下生成 dsb-client-3.5.jar

五、 DSB 内核的启动

Windows 环境：

通过命令行进入 DSB 目录，输入命令：run.bat

如果希望 JVM 以 Debug 模式启动，输入命令：debug.bat

Linux 环境：

通过 Shell 命令行进入 DSB 目录，输入命令：run.sh

如果希望 JVM 以 Debug 模式启动，输入命令：debug.sh