Core Java

14. Packaging and Deployment

Packages

- A package is a set of classes to help with a common purpose
- A package can be used to group classes implementing the same purpose, albeit different functionality
- Package names are usually vendor names
- Java defines java.* and com.* as official, javax.* as extensions and org.* as community packages

Package Names

- Package prefixes are usually vendor specific
- Simple way out is to use website .suffixes in reverse: *com.sun*, *org.apache.jakarta*
- Package suffixes are usually product name followed by product module
- *org.apache.tomcat.context*, where Tomcat is the product and *context* is a module

JAR Files

- Putting a set of class files in a *JAR* (Java Archive)
- *JAR* files are usually ZIP files with a *.jar* extension
- The JAR files also have a *META-INF/* directory with a *MANIFEST.MF* file, describing what the package contains
- It is a clean way of distributing lots of class files

MANIFEST.MF

- Manifests contain information regarding the product in the *JAR* file, the vendor, the version
- Manifests sometimes contain digital signatures of files, which can be verified with the vendor's public-key/certificate, so you can verify it's origin.
- Manifests also contain sealing information for packages
- Manifests can also have *Main* class field

Package Sealing

- When a package is sealed, it means that all class files belonging to the package are in the *JAR*, and needn't be searched anywhere (in the *CLASSPATH*).
- Sometimes it may happen that a package can not be sealed, if it has large subpackages in it.
- Useful for tools/JVM in saving class-loading

Sample MANIFEST.MF

```
Manifest-Version: 1.0
Bundle-Vendor: Apache Software Foundation
Bundle-Name: Apache Derby 10.2
Bundle-Version: 10.2.1000006.452058
Sealed: true
Name: org/apache/derby/impl/tools/sysinfo/
Sealed: false
```

Sample MANIFEST.MF

```
Manifest-Version: 1.0
Main-Class: org.mysite.myprod.mymodule.Main
Created-By: Signtool (signtool 1.3)
Name: org/mysite/myprod/mymodule
Sealed: true
Name: org/mysite/myprod/mymodule/Main.class
Digest-Algorithms: MD5
MD5-Digest: 7m8uuUBeDYj7tRFr920COa==
```

Deployment

- When deploying applications, seperate reusable frameworks in seperate archives
- To make the framework vendor-neutral, seperate the implementation from the interface; Examples: *JDBC, Servlet API, JMS, DOM*
- System properties while running JVM
- Detect during runtime, use reflection to instantiate

Deployment

- Ship implementation and interface archives in final product
- Create a launcher to invoke JVM with the appropriate properties, if necessary
- Ensures better maintenance and reuse of code
- Document the classes clearly with *javadoc*
- Ship the usage documentation separately

Demonstration

- Compile and Execute a few programs

Questions?