Enterprise Web

Application War File

Static HTML Pages

Style Sheets JavaScript Files

**images**

**WEB-INF**

favicon.png

< any other required graphics >

struts-config.xml (Struts 1 only)

<application-name>.properties (“modern” applications) web.xml

struts.xml (Struts 2 only) logging.properties (“modern” applications)

< other Struts 2 configurations >

**WEB-INF/classes**

<application-name>.jar

< required Struts libraries >

**WEB-INF/lib**

*Note that Struts prefers to have its libraries bundled into the application’s war file. The required libraries are linked to a “master” source in this directory. During the creation of the war file, the binaries are properly brought into the package.*

**Repository**

**/<webapps-repository-path>/<application-name>**

*An application must contain an emissaries directory.*

*Within that directory, one should store a standard Java properties file named using the <customer- code>.properties convention. That file contains settings such as the customer’s data file locations, necessary usernames and passwords, and other configuration information specific to the customer.*

*The repository can also contain any other on-disk resources needed by the application. For example, image files or uploaded user documents are frequently stored within the repository.*

/emissaries

< any other required directories >

Configuration Files

**Tomcat lib**

**/<tomcat-application-path>/lib**

*While generally frowned upon, we store “shared” libraries in Tomcat’s lib directory. These libraries are always stored as symbolic links so that they are easy to distinguish from “standard” Tomcat libraries.*

*Things such as the PostgreSQL driver, some Apache commons libraries, and PDF generation libraries are all examples of libraries that we place here for use during runtime as opposed to bundling them with the application in the war file.*