

# S.mon Build/Deployment

## 1 Einführung

In diesem Dokument wird beschrieben, wie man die Quellen kompiliert und welche andere Schritte notwendig sind, bevor man mit der Installation anfangen kann.

## 2 Anforderungen

Um das Build-Prozeß durchführen zu können, werden folgende Tools benötigt:

- Java Development Kit 1.3 (Siehe <http://java.sun.com/>)
- Ant 1.5.1 (Siehe <http://jakarta.apache.org/ant/>)

Selbstverständlich braucht man auch den Zugang zu dem S.mon Projektverzeichnis, um auf die Quellen zugreifen zu können.

## 3 Build-Prozeß

### 3.1 Umgebung

Bevor das Kompilieren von Java-Klassen erfolgen kann, müssen zwei Umgebungsvariablen gesetzt werden:

- JAVA\_HOME (zeigt auf das Verzeichnis, wo JDK installiert ist)
- ANT\_HOME (zeigt auf das Verzeichnis, wo Ant installiert ist)

Windows	Unix (bash)
set JAVA_HOME=c:\jdk1.3 set ANT_HOME=c:\ant1.5.1	export JAVA_HOME /usr/local/jdk1.3 export ANT_HOME /usr/local/ant1.5.1

### 3.2 S.mon Projektverzeichnis

Wir treffen hier eine Annahme, wo sich das Projektverzeichnis auf der jeweiligen Plattform befindet:

- Windows (c:\smon)
- Unix (/usr/smon)

### 3.3 Kompilieren der Java-Quellen

#### 3.3.1 Windows

Um die Java-Klassen zu kompilieren bitte den folgenden Befehl ausführen:

```
%ANT_HOME%\bin\ant -buildfile c:\smon\config\ant_build.xml
```

oder

```
%ANT_HOME%\bin\ant -buildfile c:\smon\config\ant_build.xml build
```

#### 3.3.2 Unix

Das Kompilieren wird durch den folgenden Befehl gestartet:

```
${ANT_HOME}/bin/ant -buildfile /usr/smon/config/ant_build.xml
```

oder

```
${ANT_HOME}/bin/ant -buildfile /usr/smon/config/ant_build.xml build
```

### 3.4 Ergebnis

Als Ergebnis des Build-Prozesses wird ein Unterverzeichnis namens „build“ in dem S.mon Root-Verzeichnis (Projekt-Directory) angelegt. In diesem Verzeichnis sind die kompilierten Java-Klassen zu finden.

### 3.5 Clean-Build

Das oben beschriebene Verfahren kompiliert nur die modifizierten oder die noch nicht kompilierten Java-Sources.

Will man ein komplettes Build-Prozeß durchführen, so ist zuerst das Aufräumen des Verzeichnisses „build“ notwendig.

Diese Aufgabe wird unter **Windows** durch folgenden Aufruf erledigt:

```
%ANT_HOME%\bin\ant -buildfile c:\smon\config\ant_build.xml cleanBuild
```

oder

```
%ANT_HOME%\bin\ant -buildfile c:\smon\config\ant_build.xml clean
```

Unter **Unix** ist folgender Aufruf notwendig:

```
${ANT_HOME}/bin/ant -buildfile /usr/smon/config/ant_build.xml cleanBuild
```

oder

```
${ANT_HOME}/bin/ant -buildfile /usr/smon/config/ant_build.xml clean
```

Bei der Übergabe des Parameters „cleanBuild“ an das Ant-Programm wird nur das Build-Verzeichnis aufgeräumt. In dem zweiten Fall werden auch die Distribution- und Deployment-Verzeichnisse gelöscht.

Nach diesem Schritt geht man weiter so vor, wie es in dem Kapitel 3.3 beschrieben ist.

## 4 Distribution

Nicht alle kompilierten Klassen sollen verteilt werden. Manche Klassen besitzen die Information über den privaten Schlüssel, welcher für die Erzeugung von License-Files benötigt wird. Diese Klassen sollen natürlich geheim bleiben und dürfen beim Kunden nicht installiert werden.

Um das Reverse-Engineering zu verhindern, werden die kompilierten Klassen durch den Einsatz eines Bytecode-Obfuscator geschützt. Diese Aufgabe wird durch das Tool **CodeShield** erledigt.

Wie erzeugt man die Distributionsdateien?

Unter **Windows**:

```
%ANT_HOME%\bin\ant -buildfile c:\smon\config\ant_build.xml dist
```

Unter **Unix**:

```
${ANT_HOME}/bin/ant -buildfile /usr/smon/config/ant_build.xml dist
```

Als Ergebnis wird in beiden Fällen ein Unterverzeichnis „dist“ in dem Root-Directory des Projektes angelegt, wo die Distributionsklassen zu finden sind.

## 5 Deployment

Für den Tomcat-Server wird eine Deployment-Datei benötigt. Diese Datei enthält alle Informationen (XML-Deskriptor, Klassen, Web-Seiten, ...), die für den Betrieb des S.mon-Servlets notwendig sind.

Das Deployment-Task hat die Aufgabe, diese Datei zu erzeugen. Nach einem Erfolg, wird die Datei „smon.war“ in dem Unterverzeichnis „deploy“ angelegt.

Dazu muss man unter **Windows** folgenden Befehl absetzen:

```
%ANT_HOME%\bin\ant -buildfile c:\smon\config\ant_build.xml deploy
```

und unter **Unix**:

```
${ANT_HOME}/bin/ant -buildfile /usr/smon/config/ant_build.xml deploy
```

## 6 Hinweise

Man muss die oben beschriebenen Schritte nicht einzeln durchführen. In der Konfigurationsdatei für Ant sind die Abhängigkeiten zwischen den Tasks berücksichtigt worden. Man kann gleich an das Ant-Programm den Parameter „deploy“ übergeben, um alle notwendigen Schritte zu erledigen.

Ein Kunde wird selbstverständlich nicht nur die Datei „smon.war“ brauchen.

Es werden noch folgende Dateien benötigt:

- SQL-Script, um eine Datenbank anzulegen
- XSL-Dateien für das Layout
- Ein License-File