

# Abgabe für VSP im WS'15

## 1 EINFÜHRUNG

---

Im Rahmen des Praktikums für verteilte Systeme im Wintersemester 2015 sollen Sie Server- und Client-Komponenten in Java implementieren. Als Abgabe sollen die Komponenten auf einem Docker-Host ausgeführt werden.

Die Motivation für diese Form der Abgabe ist die Minimierung des Administrationsaufwandes für die Studierenden. Auch Kollisionen zwischen den Teams sollen so gering wie möglich sein. Als Alternative zu einer vollständigen (virtuellen Linux) Maschine erlaubt ein Docker Container eine maximale Automatisierung von Erstellung, Installation und Ausführung eines Netzwerk-Servers.

## 2 VORBEREITUNG

---

Bevor der erste Server laufen kann muss jeder Teilnehmer die folgenden Schritte ausführen. Dies muss in der Regel nur einmal erledigt werden.

1. Legen Sie in der Informatik OwnCloud eine Ordner nach dem Muster  
`VSP_<username>`  
an. Im Folgenden wird dieser Ordner als `$OC_USERDIR` bezeichnet
2. Geben Sie den Ordner für den User „VS Docker Robot“ frei. Wichtig: den User, **nicht** die Gruppe „vs\_docker“

Wenn Sie im Team zusammen arbeiten reicht es eigentlich wenn ein Teammitglied diese Freigabe einrichtet. Wenn jeder aber mal schnell etwas ausprobieren möchte, macht es sich das alle Mitglieder sich ein solches Verzeichnis einrichten.

## 3 AUSFÜHRBARE JARS

---

Basis für den Container ist eine Jar-Datei, die wie ein beliebiges Programm ausgeführt werden kann.

Zum Erstellen (natürlich den Namen der Klasse entsprechend anpassen und den Usernamen einsetzen):

```
C:\mywork> jar cvfe vsp_<username>.jar MyMainClass *.class
```

Mehr Details unter: <http://www.skylit.com/javamethods/faqs/createjar.html>

Die resultierende Jar-Datei in einen Ordner namens `$OC_USERDIR/docker_0/java` legen und unbedingt folgende Namenskonventionen einhalten: `vsp_<username>.jar`.

### 3.1 ABHÄNGIGKEITEN

Wenn die Java-Anwendung weitere Bibliotheken benötigt, müssen diese ebenso in den `$OC_USERDIR/docker_0/java` Ordner platziert werden. Diese können beliebig heißen, vorausgesetzt die Namen enden auf „.jar“. Nur dann werden sie in den Classpath der JVM eingetragen.

### 3.2 IDENTIFIKATION DES JARS

TODO: Erzeugen und Merken des SHA2 Hashs unter Windows

## 4 AUSGABEN

---

Die Ausgaben des JVM Prozesses können über eine (derzeit rudimentäre) Weboberfläche erhalten werden.

Einfach auf <http://vs-docker.informatik.haw-hamburg.de/> aufrufen. Hier sind weitere Details zu sehen.

## 5 DOCKER CONTAINER

---

Diesen Schritt macht ein Script im Moment automatisch. Es ist hier vor allem zum Verständnis des Gesamtsystems. Wenn Sie einen Container selbst erstellen wollen, dann müssen Sie sich auch darum kümmern, dass er läuft.

In das Verzeichnis, das den docker ordner enthält, das Dockerfile legen

```
docker build
```

aufrufen.

Die resultierende Docker-Datei im Verzeichniss „`$OC_USERDIR/docker_0/container`“ ablegen.

Images zwischen Hosts migrieren: <http://www.jamescoyle.net/how-to/1512-export-and-import-a-docker-image-between-nodes>

Wenn ein Ordner „`$OC_USERDIR/docker_0/java`“ existiert, so wird dieser vorrangig verwendet. Siehe auch nächsten Abschnitt.

### 5.1 MEHRERE CONTAINER

Wenn neben dem „docker\_0“ auch weitere Verzeichnisse „docker\_1“, „docker\_2“ existieren, dann werden entsprechend viele Container erzeugt und gestartet.

### 5.2 PORTS

Für die Kommunikation mit den Containern wird der folgende Port auf der Maschine `vs-docker.informatik.haw-hamburg.de` gebunden:

10 000 + Helios Nr. + Container Nummer

gebunden auf

4567

Die Helios Nummer ist der numerische Teil des Usernamens. Wenn hier Konflikte auftauchen, bitte Herrn Behnke ansprechen. Die Container Nummer ist der Suffix

Es ist möglich, dass im Netzwerk der HAW diese Ports gefiltert werden. Daher kann die REST Schnittstelle auch über

<https://vs-docker/ports/<nummer des Ports>/<endpunkt>>

erreicht werden. Dies geschieht über einen Proxy.

## 6 VERZEICHNISDIENST

---

Unter <http://vs-docker.informatik.haw-hamburg.de:8053/service> kann ein REST-Service für die Konfiguration erreicht werden. Von den Docker-Containern aus ist dieser unter einer separat definierten URL zu erreichen. Diese wird in der Umgebungsvariable DIRECTORY\_SERVICE\_URL im Docker Container zur Verfügung gestellt.

Dieser Verzeichnisdienst liefert unter dem Usernamen des Nutzers der die Container in die OwnCloud gestellt hat einen Eintrag für die URLs unter denen die einzelnen Dienst-Endpunkte erreicht werden können. Hiermit kann der Client Dienst konfiguriert werden.

## 7 CLIENT ZUGRIFF VON AUßEN

---

Der Client Service ist unter dem Port 9<Nummer des Usernamens> auf vs-docker.informatik.haw-hamburg.de zu erreichen.

## 8 NACHBEMERKUNGEN

---

Die derzeitige Abgabe über die OwnCloud und das zyklische Aktivieren der Container ist nicht die optimale Lösung. Daher wird sicher dieser Schritte wahrscheinlich während des Semester noch ändern.