Docker - automatisierte Scan-Verarbeitung

1. Allgemeines

Die bei uns eingesetzten Server Synology NAS lassen eine gewisse Verarbeitung auf Command line Ebene zu. Das hat aber Grenzen, besonders, wenn für die Ausführung irgendwelches Debian-Packages oder sonstigen Tools geladen werden müssen. Das KOENNTE man zwar mittels IPKG-Pakete irgendwie hinkriegen, aber das führt dazu, dass bei jedem Update von Synology gezittert werden muss, ob es noch läuft.

DOCKER bietet eine kleine virtuelle Umgebung an, inder wir eine Software laufen lassen können.

Die OCR-Erkennung der gescannten PDF ist so ein Beispiel. Sie benötigt tesseract, pdftk, zbarimg etc., und zwar möglichst in der allerneusten Version, damit es am besten funktioniert.

In Docker arbeitet man mit Images, die entweder allgemeingültig sind oder von uns angelegt wurden (User planbee). Damit wir Systemumgebung und Programm auseinanderhalten, wurden zwei Images erstellt:

- `planbee/syno-debian-ocr`: Es enthält eine Debian-Version mit allen notwendigen Packages installiert.
- `planbee/syno-ocr`: Es basiert auf syno-debian-ocr und ergänzt dieses mit den notwendigen OCR-Scripts ocr_loop.sh und ocr.sh

syno-debian-ocr

Im Januar 2018 wurde die Umgebung auf das neuste, unstable Debian-SID aktualisiert, um von der tollen tesseract-Version 4.0 zu profitieren. Die Umgebung wurde so aufgebaut:

```
# 1. Hole das Image debian-sid
docker pull debian:sid-slim
# 2. Starte den Container mit einem Prompt
docker run -ti --entrypoint=bash --name debian debian:sid-slim
# 4. Installiere tesseract-ocr und alle anderen benötigten Tools:
apt-get update
apt-get -y install tesseract-ocr tesseract-ocr-deu poppler-utils zbar-tools
pdftk nano
exit
# 5. Speichere den aktuellen Container als neues Repository-Image
docker login
docker commit debian planbee/syno-debian-ocr
docker push planbee/syno-debian-ocr:latest
```

3. syno-ocr

Im Syno-ocr sind die Scripts für die Verarbeitung enthalten. Die Scripts prüfen alle /src* - Verzeichnisse und verarbeiten die Dateien von /srcXXX nach /dstXXX

```
# um ein neues Image anhand des aktuellen Verzeichnisses anzulegen:
docker build -t planbee/syno-ocr .

# um einen Container zu erstellen auf dem Image
docker create \
        --volume /volume1/transfer/Scan/Belege/raw:/src1 --volume
/volume1/transfer/Scan/Belege:/dst1 \
        --volume /volume1/transfer/Scan/Schulungsnotizen/raw:/src2 --volume
/volume1/transfer/Scan/Schulungsnotizen:/dst2 \
        --volume /volume1/transfer/Scan/Ausbildungen/raw:/src3 --volume
/volume1/transfer/Scan/Ausbildungen:/dst3 \
        --name ocrpdf planbee/syno-ocr

# oder einen Test-Container
docker run -ti --entrypoint=/bin/bash \
        --volume /volume1/transfer/Scan/Belege/raw:/src1 --volume
/volume1/transfer/Scan/Belege:/dst1 \
        --name ocrtest planbee/syno-ocr


# um diesen Container zu starten
docker start ocrtest

# um diesen Container zu stoppen
docker stop ocrtest

# um diesen Container zu löschen
docker rm ocrtest

# um das ganze Image vom Rechner zu killen
docker rmi planbee/syno-ocr
```

Aufbau des Debian-SID Basis-Images

```
# 1. Hole das Image debian-sid
docker pull debian:sid-slim
# 2. Erstelle einen Container dafür
docker create --name debian debian:sid-slim
# 3. Starte den Container mit einem Prompt
docker run -ti --entrypoint=bash debian:sid-slim
# 4. Installiere tesseract-ocr und alle anderen benötigten Tools:
```

```
apt-get update
apt-get -y install tesseract-ocr tesseract-ocr-deu poppler-utils zbar-tools
pdftk
exit
# 5. Speichere den aktuellen Container als neues Repository-Image
docker login
docker commit debian planbee/syno-debian-test
docker push planbee/syno-debian-test:latest
```