



Generic Pseudonym Administration Service

Current Docker-Version of gPAS: 2023.1.0 (Mai 2023) Current Docker-Version of TTP-FHIR-Gateway: 2023.1.0 (Mai 2023), Details from [ReleaseNotes](#)

Aktualisierung der THS-Tools per Docker

Hintergrund

Am folgenden Beispiel wird die Aktualisierung der Docker-Container vom gPAS gezeigt. Grundsätzlich findet die Aktualisierung aller THS-Tools (E-PIX, gPAS, gICS, Dispatcher) auf dieselbe Weise statt.

Im Beispiel wird die bestehende und laufende Instanz vom gPAS als `<gpas-old>` bezeichnet. Die existierende Version (`<old-version>`) soll gesichert und ein Update auf eine neue Version vom gPAS (`<gpas-new>`, `<new-version>`) durchgeführt werden, ohne die bereits vorhandenen Daten in der MySQL-Datenbank zu verändern.

Ob die Instanzen vom gPAS laufen, kann mit folgenden Befehl geprüft werden:

```
docker ps -a
```

Handlungsanweisung

Neue Tool-Version von der THS-Webseite herunterladen

Die aktuelle Version von ths-greifswald.de/gpas herunterladen und entpacken, sowie auf das Host-System kopieren und sicherstellen, dass entsprechende Berechtigungen zum Ausführen der Dateien gesetzt sind.

```
CHMOD -R 755 /PFAD
```

Sichern der aktuellen Docker-Konfiguration

Um auf dem Host-System den derzeitigen Stand der gPAS-Konfiguration (Wildfly-Skripte, etc.) zu sichern, den entsprechenden Ordner per TAR-Archiv sichern:

```
tar czf backup-gpas-2022-03-31.tgz <gpas-old>/
```

Sichern der existierenden Datenbank

Um zusätzlich die Sicherung der existierenden Datenbank durchzuführen, wird ein MySQL-Dump über die Docker-Konsole angestoßen und die resultierende Export-Datei im Dateisystem vom Host abgelegt.

```
sudo docker exec gpas-<old-version>-mysql /usr/bin/mysqldump -u gpas_user -p gpas  
> backup-gpas-<old-version>-2022-03-31.sql
```

Der Name der bestehenden MySQL-Instanz muss entsprechend angepasst werden.

Aktualisieren der Datenbank

Für alle Versionen sind die Datenbank-Aktualisierungsskripte jeweils im Docker-Verzeichnis unter *<gpas-new>/update_scripts* zu finden. Die Update-Skripte müssen in den Docker-Container kopiert werden, wobei nur die Skripte erforderlich sind, welche die Version zwischen *<gpas-old>* zu *<gpas-new>* betreffen.

```
docker cp <gpas-new>/update_scripts/ gpas-<old-version>-mysql:/update-files/
```

Je nachdem von welcher Version aus gPAS aktualisiert werden soll, müssen die relevanten SQL-Skripte *chronologisch durchlaufen* werden.

Beispiel: Für ein Update von Version 1.10.0 auf 1.12.0 sind demzufolge die Skripte *update_database_gpas_1.10.x-1.11.0.sql* und *update_database_gpas_1.11.x-2.12.0.sql* auszuführen.

Dazu per MySQL Client mit der bestehenden Datenbank verbinden und die Update-Skripte nacheinander durchlaufen. Dies kann per *docker* realisiert werden (Nutzernamen und Passwörter ggf. anpassen).

Beispiel:

```
docker exec -it gpas-1.10.0-mysql /usr/bin/mysql -u gpas_user -p -e "USE gpas;  
$(cat gpas-new/standard/update_database_gpas_1.10.x-1.11.0.sql)"  
docker exec -it gpas-1.10.0-mysql /usr/bin/mysql -u gpas_user -p -e "USE gpas;  
$(cat gpas-new/standard/update_database_gpas_1.11.x-2.12.0.sql)"
```

Aktualisieren der Deployments und Wildfly-Konfiguration

Den Datenbank-Container nun herunterfahren

```
docker gpas-<old-version>-mysql down
```

Aktualisieren der Deployments

Die Deployments im `<gpas-old>` Verzeichnis auf dem Host-System löschen und die neuen Deployments hinein kopieren

```
rm -f <gpas-old>/deployments/*
cp -R <gpas-new>/deployments/ <gpas-old>/deployments/
```

Aktualisieren der Bezeichnung des MySQL Containers

```
sudo docker rename gpas-<old-version>-mysql gpas-<new-version>-mysql
```

Aktualisieren der JBOSS Konfigurationsskripte

Die alten Dateien können gesichert oder gelöscht und die neuen müssen eingespielt werden:

```
mv <gpas-old>/jboss <gpas-old>/jboss-<old-version>
cp -R <gpas-new>/jboss/ <gpas-old>/jboss
```

Mit Hilfe dieser Dateien wird JBOSS nach den Vorgaben aus den `*.env`-Dateien konfiguriert.

Aktualisieren der Umgebungsvariablen für die JBOSS Konfigurationsskripte

In der aktuellen Version liegen die zugehörigen `*.env`-Dateien im Unterordner `./envs`. In älteren Versionen lagen diese direkt im Wurzelverzeichnis des Dockerpaketes. Falls eine solche Version aktualisiert werden soll, müssen zuvor die `*.env`-Dateien in den Unterordner `./envs` verschoben werden:

```
mkdir <gpas-old>/envs
mv <gpas-old>/*.env <gpas-old>/envs/
```

Dieser Ordner sollte auch gesichert werden:

```
cp -R <gpas-old>/envs <gpas-old>/envs-<old-version>
```

Nun muss die Liste der neuen und umbenannten `ENV`-Variablen in der `README_gICS.md` im Wurzelverzeichnis des Dockerpaketes studiert werden, um gegebenenfalls die `*.env`-Dateien entsprechend anzupassen.

Achtung: unter Umständen wurden auch die Namen der `*.env`-Dateien geändert. Dies muss auf auf jeden Fall angepasst werden.

Alternativ kann man mit etwas höherem Aufwand die Anpassungen der alten `*.env`-Dateien manuell in die neuen übertragen, um größtmögliche Ähnlichkeit zwischen den angepassten und ausgelieferten `*.env`-Dateien zu bewahren. Dazu müssen die alten angepassten Dateien gesichert und die neuen eingespielt werden:

```
mv <gpas-old>/envs <gpas-old>/envs-<old-version>
cp -R <gpas-new>/envs/ <gpas-old>/envs
```

Anschließend müssen alle manuellen Anpassungen der alten in die neuen `*.env`-Dateien übertragen werden. Dabei ist hohe Aufmerksamkeit erforderlich. Wir empfehlen die Nutzung eines grafischen Diff-Werkzeugs (z.B. [devart Code Compare Free](#)). Für zukünftige Updates ist es hilfreich, die bestehenden Skeletons, Beispiele und Kommentare nicht zu ändern, sondern die eigenen Zeilen zu ergänzen und durch einen leicht wiederauffindbaren Kommentar zu markieren.

Aktualisieren der Docker-Compose-Konfiguration

Die alte angepasste Datei muss gesichert und die neue eingespielt werden:

```
mv <gpas-old>/docker-compose.yml <gpas-old>/docker-compose-<old-version>.yml
cp <gpas-new>/docker-compose.yml <gpas-old>/docker-compose.yml
```

Wahrscheinlich müssen auch hier die Anpassungen der alten `docker-compose.yml` (insbesondere für Ports und Volumes) in die neue übertragen werden (am besten wieder mit Hilfe eines grafischen Diff-Werkzeugs).

Aktualisieren der Dokumentationsdateien

Schließlich empfiehlt es sich, auch die aktualisierten Dokumentationsdateien zu übertragen:

```
rm -f <gpas-old>/*.md
cp <gpas-new>/*.md <gpas-old>/
rm -f <gpas-old>/docs/*
cp -R <gpas-new>/docs/ <gpas-old>/docs/
```

Anpassen des Eigentümer-Benutzers

```
chown 999 <gpas-new>/sqls
chown 1000 <gpas-new>/deployments
chown 1000 <gpas-new>/logs
chown 1000 <gpas-new>/jboss
```

Starten des aktualisierten Containers

Den aktualisierten Container mittels folgendem Befehl starten (-d um Container im Hintergrund zu starten):

```
docker-compose up -d
```

Den Erfolg der Aktualisierung prüfen durch Aufruf des Web-Frontends unter <http://IPADDRESS:8080/gpas-web>.

Im Fehlerfall: Wiederherstellung der Datenbank

Im Fehlerfall, kann die bisherige Datenbank wiederhergestellt werden (sofern die Anleitung befolgt wurde). Nutzernamen und Passwort ggf. anpassen.

```
docker exec -it gpas-<new-version>-mysql /usr/bin/mysql -u gpas_user -p -e "USE gpas; $(cat backup-gpas-2022-03-31.sql)"
```

Additional Information

The gPAS was developed by the University Medicine Greifswald and published in 2013 as part of the [MOSAIC-Project](#) (funded by the DFG HO 1937/2-1).

Selected functionalities of gPAS were developed as part of the following research projects:

- MIRACUM (funded by the German Federal Ministry of Education and Research 01ZZ1801M)

Credits

Concept and implementation: L. Geidel

Web-Client: A. Blumentritt, M. Bialke, F.M. Moser

Docker: R. Schuldt

TTP-FHIR Gateway für gPAS: M. Bialke, P. Penndorf, L. Geidel, S. Lang, F.M. Moser

License

License: AGPLv3, <https://www.gnu.org/licenses/agpl-3.0.en.html>

Copyright: 2013 - 2023 University Medicine Greifswald

Contact: <https://www.ths-greifswald.de/kontakt/>

Publications

- <https://dx.doi.org/10.3414/ME14-01-0133>
- <https://dx.doi.org/10.1186/s12967-015-0545-6>

Supported languages

German, English

