

GwResourceDeployer - Eseguibile

Versione 1.5.0

ReleaseNote

- **issue #2** il deployer Keycloak ora dispiega tutti i client il cui file .json è presente nella folder **gw_package/keycloak** (in precedenza veniva gestito un solo client tramite la property *keycloak.clientRepresentationFile*, ora rimossa). Vedi la sezione [Folder keycloak](#) della guida alla struttura risorse per i dettagli.

Si tratta di un eseguibile a riga di comando, che prende in ingresso dei parametri

Questa guida è da intendersi come passo successivo a alla procedura di setting-up delle risorse da dispiegare, trattata [qui](#).

Procedura

- procurarsi la folder *gw_package*, contenente le risorse da dispiegare, e posizionarla al path desiderato. Esempio:

```
C:\gw_package
```

- scaricare [gwresourcesdeployer-1.5.0-bin.zip](#) e scompattarlo al path desiderato. Ad esempio in:

```
C:\gwResourcesDeployer-1.5.0-bin
```

- aprire una console con (in Windows: run ⇒ cmd)
- posizionarsi al path dell'eseguibile. Esempio:

```
cd C:\gwResourcesDeployer-1.5.0-bin
```

- eseguire il **gwdeploy.bat** con, **in ordine**, i parametri descritti. Template:

```
gwdeploy [gw_package folder] [spring active profiles]
```

. Ogni parametro va separato dal precedente con uno spazio bianco. Dettaglio dei parametri, in ordine:

- **gw_package folder**: path alla *folder gw_package*. Esempio:

```
C:\gw_package
```

Se il path contiene *spazi bianchi*, racchiudere tutto fra doppi apici. Esempio:

```
"C:\path to\gw_package"
```

- **spring active profiles**: gli *Spring active profiles* relativi database. Valori ammessi [**'oracle'**,**'postgres'**,**'sqlserver'**]. Esempio:

```
postgres
```

Esempi codice a riga di comando

Ecco alcuni esempi di sequenze di comandi da eseguire da riga di comando, in base alle situazioni

Esempio 1, assunto che:

- *C:\gwResourcesDeployer* sia il path dell'eseguibile
- *C:\gw_package* sia il path delle risorse
- database *postgres*

```
cd C:\gwResourcesDeployer-1.5.0-bin
gwdeploy C:\gw_package postgres
```

Esempio 2, assunto che:

- *C:\gwResourcesDeployer* sia il path dell'eseguibile
- *C:\path to\gw_package* sia il path delle risorse
- database *oracle*

```
cd C:\gwResourcesDeployer-1.5.0-bin
gwdeploy "C:\path to\gw_package" oracle
```

Note

Sistemi con JVM precedente alla Java 9

Per questi sistemi bisogna utilizzare il **gwdeploy_prejava9.bat** specifico disponibile dentro l'archivio dell'eseguibile.

Note

Naturalmente bisogna modificare coerentemente il comando da lanciare.

Esempio JVM precedente alla Java 9, assunto che:

- *C:\gwResourcesDeployer* sia il path dell'eseguibile
- *C:\path to\gw_package* sia il path delle risorse
- database *oracle*

```
cd C:\gwResourcesDeployer-1.5.0-bin
```

```
gwdeploy_prejava9 "C:\path to\gw_package" oracle
```

Mapguide

Ci sono meccanismi noti, che **consigliano fortemente di far girare l'eseguibile dalla stessa macchina di installazione dell'applicativo:**

- l'eseguibile fallisce se eseguito da macchine che non hanno l'opportuna configurazione di sistema (.dll mapguide in classPath e variabile di sistema impostata). Tipicamente, nelle macchine predisposte dai sistemisti per il rilascio, dove è previsto MapGuide, tali configurazioni iniziali sono sempre già state fatte al momento del deploy delle risorse. Si potrebbe però far girare l'eseguibile dalla propria macchina la folder *gw_package* disponibile come percorso di rete. In questo caso, in mancanza della configurazione, può essere generato l'errore in fase di dispiegamento:

```
-----
-----
24-11-2021 18:09:35,483 [main] GEOWEB INFO

GwDeployer - deployMapguideResources()
Adjusting path with file name..
adjusted path: I:\projects\CDE35\gw_package\mapguide\webconfig.ini
Configured path refers to an existing file
Initializing MapGuide with the file:
I:\projects\CDE35\gw_package\mapguide\webconfig.ini
-----
-----

2021-11-24 18:09:35.612 ERROR 9556 --- [           main]
o.s.boot.SpringApplication : Application run failed

java.lang.UnsatisfiedLinkError: no MapGuideJavaApi in java.library.path
    at java.lang.ClassLoader.loadLibrary(Unknown Source)
~[na:1.8.0_281]
    at java.lang.Runtime.loadLibrary0(Unknown Source)
~[na:1.8.0_281]
    at java.lang.System.loadLibrary(Unknown Source) ~[na:1.8.0_281]
    at
org.osgeo.mapguide.MapGuideJavaApiJNI.<clinit>(MapGuideJavaApiJNI.java:
2957) ~[mapguide-3.1.1.9378.jar:na]
    at
org.osgeo.mapguide.MapGuideJavaApi.MgInitializeWebTier(MapGuideJavaApi.
java:14) ~[mapguide-3.1.1.9378.jar:na]
    at
com.geowebframework.gwResourcesDeployer.service.MapguideDeployerService
.deploy(MapguideDeployerService.java:260)
~[gwResourcesDeployer-1.2.0.jar:1.2.0]
    at
com.geowebframework.gwResourcesDeployer.GwDeployer.deployMapguideResour
ces(GwDeployer.java:277) [gwResourcesDeployer-1.2.0.jar:1.2.0]
```

```
        at
com.geowebframework.gwResourcesDeployer.GwDeployer.run(GwDeployer.java:
138) [gwResourcesDeployer-1.2.0.jar:1.2.0]
        at
org.springframework.boot.SpringApplication.callRunner(SpringApplication
.java:819) [spring-boot-2.4.5.jar:2.4.5]
        at
org.springframework.boot.SpringApplication.callRunners(SpringApplicatio
n.java:803) [spring-boot-2.4.5.jar:2.4.5]
        at
org.springframework.boot.SpringApplication.run(SpringApplication.java:3
46) [spring-boot-2.4.5.jar:2.4.5]
        at
org.springframework.boot.SpringApplication.run(SpringApplication.java:1
340) [spring-boot-2.4.5.jar:2.4.5]
        at
org.springframework.boot.SpringApplication.run(SpringApplication.java:1
329) [spring-boot-2.4.5.jar:2.4.5]
        at
com.geowebframework.gwResourcesDeployer.GwDeployer.main(GwDeployer.java
:94) [gwResourcesDeployer-1.2.0.jar:1.2.0]
```

From:
<https://wiki.geowebframework.com/> - **GeowebFramework**

Permanent link:
https://wiki.geowebframework.com/doku.php?id=gwusermanual:gw_resources_deployer_executable_1_5_0

Last update: **2026/09/09 15:14**

