

Dashboard Inhalte deployen

Für das Deployment des YunaML-Codes des zuvor erstellten Inhalts in die Datenbank wird das Tool **dsedep** verwendet.

dsedep ist ein Tool zur Erstellung, zur Verwaltung und zum Deployment von Dashboard Inhalten wie Widgets und Views. Während die CM-PortalDB die jeweilig aktuelle/deployte Version enthält, können über die YunaML-Dateien verschiedene Varianten Content verwaltet werden. Die Bearbeitung von Content sollte in den YunaML-Dateien erfolgen, so dass sich folgender Ablauf ergibt:

1. Initiale Extraktion der Datenbank-Struktur mit dsedep zum Erstellen der ersten Version der YunaML-Datei(en) 1.0
2. Bearbeiten der YunaML-Datei(en) und Erzeugen der nächsten Version 2.0
3. Deployen von Version 2.0 mit dsedep auf die Datenbank/das Portal
4. Bearbeiten der YunaML-Datei(en) und Erzeugen der nächsten Version 3.0
5. Deployen von Version 3.0 mit dsedep auf die Datenbank/das Portal
6. usw.



Source page access restriction: Click the link below to check if the page is accessible.

[/display/~domenik/dseDep-Schema+v0.3](#)

dsedep unterscheidet zwischen den Entitäten Views (Widgets), DataID und Filtern. Für jede Entität lassen sich eigene YunaML-Dateien erzeugen oder auch eine große YunaML-Dateien für alle drei Entitäten.



Ab Portalversion 0.53 werden im Zuge des Dashboard Inhalt-Deployments auch Translation-Keys und ihre Defaults in die Übersetzungstabellen der Datenbank persistiert.



Ab Portalversion 0.49 wird das Tool dseDep verwendet. Für alle vorherigen Versionen gilt die Nutzung des Tools spDep.

In diesem Zusammenhang ist für ein Dashboard Inhalt-Deployment in Portalversionen kleiner als 0.49 in allen Befehlen dseDep durch spDep zu ersetzen!

dsedep herunterladen

Die jeweils aktuelle Version von dsedep wird Ihnen ausgeliefert über die RPM-Pakete. So wird sichergestellt, dass Ihnen bei Anpassungen jeweils die korrekte Version von dsedep vorliegt.

Nutzung von dsedep

dsedep wird über die Eingabeaufforderung (Command Line Interface) aufgerufen und gesteuert. Ob Sie die Eingabeaufforderung Ihres Betriebssystems oder ein entsprechendes Tool eines Drittanbieters nutzen spielt dabei keine Rolle.



dseDep sollte sich im gleichen Ordner wie der zu deployende Dashboard Inhalt oder eine Ebene darüber befinden, da in dseDep der Ordner des zu deployenden Inhalt definiert wird.

Zunächst müssen Sie sich zum Speicherort von dsedep navigieren, um die Datei dort ausführen zu können. Nutzen Sie hierfür die Konsolenbefehle, die Ihnen zur Verfügung stehen:

Befehl	Bedeutung	Beispiel	Funktion
cd Ordnername	change directory (in einen untergeordneten Ordner)	cd git	Wechselt in den genannten Ordner, der innerhalb des zuvor aufgerufenen Ordners liegt. Im Beispiel wird der Unterordner "git" im Ordner "documents" geöffnet.

cd ..	change directory (in den übergeordneten Ordner)	cd ..	Wechselt in den übergeordneten Ordner zurück
dir	directory	dir	Listet die Inhalte des gewählten Ordners aus.

Nachdem Sie zum Speicherort der dsedep.jar navigiert sind, geben Sie zum Aufrufen von dsedep in die Kommandozeile den Befehl **java -jar dsedep.jar** ein. Daraufhin erscheint die folgende Ansicht, in der Sie alle zur Verfügung stehenden Funktionen von dsedep und deren Kommandos vorfinden:

```

C:\Users\domenik\Documents\git\de.eoda.dsp.content>java -jar spdep.jar
SP Deploy - (C) 2016 - eoda GmbH
Version: 0.2.2

usage: java -jar spdep.jar -i views.xml widgets.xml data.xml -s <server>
        -u <username> -p <password>
-d,--deploy          Deploy to database server
-D,--data <arg>      Deploy only this data definition.
-h,--help            Print this helpscreen
-i,--input <arg>     Input XML files
-o,--output <arg>    Output XML files
-p,--password <arg> Database Password (Optional, you will be
                    prompted.)
-R,--recursive       Searches input path instead of file
-ref,--reference <arg> Use user Reference tag
-s,--host <arg>      Database Server
-t,--schema <arg>   Database schema
-u,--username <arg> Database Username
-v,--verbose         Verbose output
-V,--view <arg>     Deploy only this view
-x,--extract         Extract database Server tables
-Z,--xls <arg>      Write the tables as xls tables as they are or
                    would be written.

C:\Users\domenik\Documents\git\de.eoda.dsp.content>

```

Befehlskürzel	Befehlsname	Bedeutung / Funktion
-d	deploy	Deploy to database server
-clean	clean	Cleans database Server tables (deletes the content from tables, not the tables themself)
-D	data <arg>	Deploy only this data definition
-h	help	Print this helpscreen
-i	input <arg>	Input XML files / Folder of content files to be deployed
-lang	language <arg>	Generate translation file (.json) in the current working directory containing translation key-value pairs for the specified language
-o	output <arg>	Output XML files
-p	password <arg>	Database password (optional, you will be prompted)
-R	recursive	Searches input path instead of file
-ref	reference <arg>	Use user reference tag
-s	host <arg>	Database server (IP adress)
-t	schema <database> <sp-schema> <portal-schema>	Database schema. Up to three strings: Database, 'sp'-Schema, 'portal'-Schema. Must be specified in this order. Only database is required.
-u	username <arg>	Database username
-v	verbose	Verbose output
-V	view <arg>	Deploy only this view
-x	extract	Extract database server tables

-Z	xls <arg>	Write the tables as xls as they are or would be written
----	-----------	---

Zur Nutzung der dsedep-Befehle ist folgendes Schema anzuwenden:

`java -jar dsedep.jar -Befehlskürzel1 Info zu Befehlskürzel1 -Befehlskürzel2 Info zu Befehlskürzel2 -Befehlskürzel3 Info zu Befehlskürzel3`

Die Farben sind willkürlich gewählt und sollen nur zur verdeutlichung der Befehle und den zugehörigen Informationen dienen)

Beispiele

Deployment mit Referenz-Tag zur Nutzung von Dashboard-Versionen:

`java -jar dsedep.jar -i Content/ -R -u Username -p Password -s 123.456.789.0 -t Schema -d -ref NameOfRefTag`

Beispiel: `java -jar dsedep.jar -i Content/ -R -u ServerAdmin -p TollesPasswort12345 -s 192.168.0.1 -t CM-PortalDB sp -d -ref domenik`

Beispiel zum Abruf der Datenbank als Excel-File

`java -jar Tools/dsedep.jar -i Content/ -R -u Username -p Password -s 123.456.789.0 -t Schema -Z`

Beispiel für ein Deployment:

`java -jar dsedep.jar -d -i views.xml widgets.xml data.xml -t CM-PortalDB sp -s 123.456.789.0 -u Username -p 123456`

Deployment der gesamten XML-Definition in die DB:

`java -jar dsedep.jar -i Dateiname.xml -u Username -p Password -s 123.456.789.0 -t Schema -d`

(Rekursives) Deployment aller XML-Definitionen (innerhalb des Verzeichnisses und aller Unterordner)

`java -jar dsedep.jar -i Dateiname.xml -R -u Username -p Password -s 123.456.789.0 -t Schema -d`

Deployment einer einzelnen View in die DB:

`java -jar dsedep.jar -i *.xml -u Username -p Password -s 123.456.789.0 -t Schema -d -V viewname (aus view.xml)`

Deployment einer einzelnen Datendefinition (DataID) in die DB:

`java -jar dsedep.jar -i *.xml -u Username -p Password -s 123.456.789.0 -t Schema -d -D friendlyname (aus data.xml)`

Beispiel für das Löschen der zu einem Referenz-Tag gehörenden YUNAML-Inhalte:

`java -jar dsedep.jar -s 123.456.789.0 -t Schema -u Username -p Password --clean -ref demo`



Achtung bei Verwendung des Befehls --clean

Mit dem Befehl "clean" können Inhalte unter einem Referenz-Tag gelöscht werden. Wird kein Referenz-Tag angegeben, werden ALLE YUNAML-Inhalte gelöscht, unabhängig davon, ob sie über einen Referenz-Tag verfügen oder nicht! Dieser Vorgang kann nicht rückgängig gemacht werden und muss nicht erneut bestätigt werden.


Generierung einer Übersetzungsdatei (.json) aus den im Dashboard Inhalt und in der Datenbank definierten Translation-Keys

`java -jar dsedep.jar -i Dateiname.xml -u Username -p Password -s 123.456.789.0 -t Schema sp portal -lang deutsch -d`

Erzeugt die Datei 'deutsch.json' im aktuellen Arbeitsverzeichnis. Diese bildet die Sicht des Nutzers im Portal ab und enthält dementsprechend alle Übersetzungen, die dem Nutzer bei der ausgewählten Sprache angezeigt werden.

 **Änderung des Verhaltens für Übersetzungen**

- Translation-Keys aus dem Dashboard Inhalt, die nicht über einen Standardwert verfügen, werden mit dem Translation-Key als Wert versehen.
- Da die Translation-Keys mit Standardwerten in die Datenbank geschrieben werden, wird bei jedem Deployment Inhalt-Standard-Übersetzung in der Datenbank überschrieben (⚠ Übersetzungen die nicht auf den im Dashboard definierten Standardwerten basieren, sind hiervon nicht betroffen). **Diese fungiert für alle Sprachen unter jedem Referenz-Tag als Rückfallwert.**
- Bei der Verwendung von dsedep wird der Dashboard-Developer informiert , welche Translation-Keys noch nicht in der Datenbank vorhanden sind.

 Beim Aufruf der dsedep.jar ist der Pfad für die YunaML-Input-Files zu beachten. Beim Deployen ist die Abfrage yes/no mit 'yes' zu beantworten.