

KService-Application



Inhaltsverzeichnis

1 Einführung	3
2 Installation	3
2.1 Deploy Konfiguration	3
2.2 Deploy	3
2.3 Volume vorbereiten	3
3 Konfiguration	5
3.1 Service Instanz definieren	5
3.2 Restlets definieren	6
3.2.1 Restlets allgemein	6
3.2.2 Report Restlet	7
3.2.3 Download Restlet	8



1 Einführung

Die KService-Application ermöglicht die Bereitstellung von RESTful Services zu K-Infinity. Es ist eine plattformunabhängige, leichtgewichtige Schnittstelle, die in der Regel XML als Ergebnis liefert.

2 Installation

2.1 Deploy Konfiguration

Zu Installation muss das das WAR-File (kservice-1.1.0.war) auf einem Tomcat-Server deployed werden. Benennen Sie das WAR-File so um, dass der Dateiname gleich lautet, wie das zukünftige Pfad auf dem Webserver.

Gewünschte Ziel-URL: `http://servername:8080/api`
Dateiname des WAR-Files: `api.war`

kem.properties

In der kem.properties (WEB-INF/kem.properties) müssen adresse der Bridges, der Volume-name und der Systemuser-Account konfiguriert werden.

Passen Sie dazu die Defaultwerte an.

kservice.properties

In der kservice.properties muss ein "servicelidentifier" gesetzt werden. So kann die Webaplikation die richtige Konfiguration identifizieren und laden. Diese bietet die Möglichkeit unterschiedliche Kservice-applikationen zu betreiben, die auf unterschiedliche Konfigurationen zugreifen.

Der DefaultWert ist "KService".

2.2 Deploy

Das angepasste WAR-File kann danach über den Tomcat-Manager deployed werden. In der Regel ist dieser unter folgender URL zu erreichen:

`http://servername:8080/manager/html`

Prüfen Sie die Installation durch folgenden Aufruf im Browser:

`http://servername:8080/config/reload`

Meldet der Server "config reloaded" konnte die Verbindung zu K-Infinity erfolgreich hergestellt werden.

2.3 Volume vorbereiten

Vorbereiten des Volumes

Die Konfiguration der einzelnen Restlets wird im K-Infinity-Volume vorgenommen. Dazu muss das entsprechende Schema importiert werden.

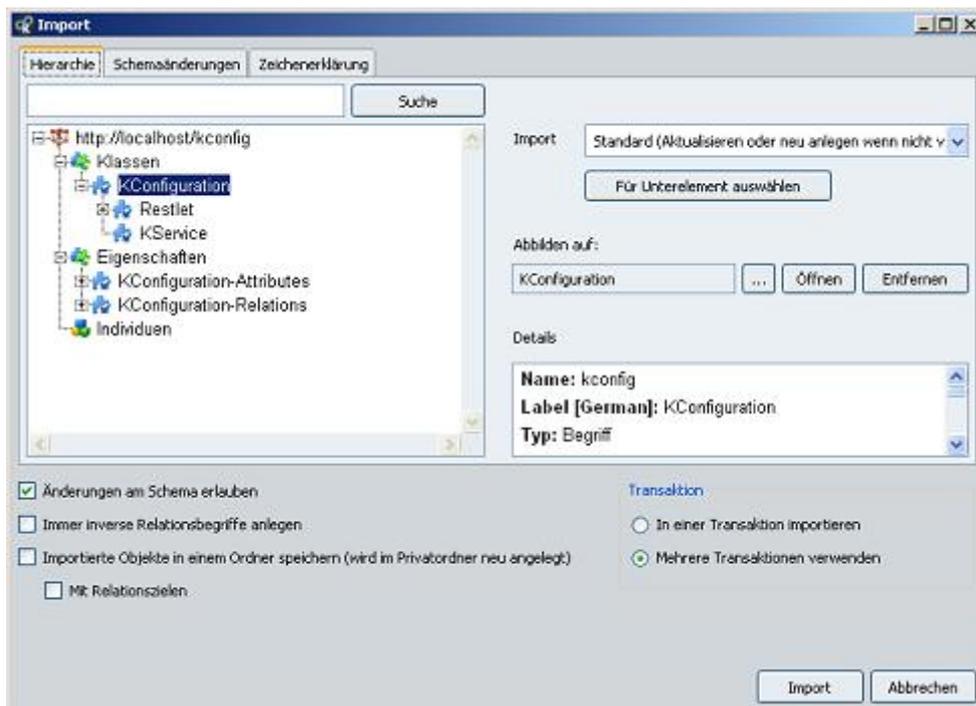


Öffnen Sie den KB und wählen die aus dem Menü die Option "Werkzeuge > RDF > RDF-Import".

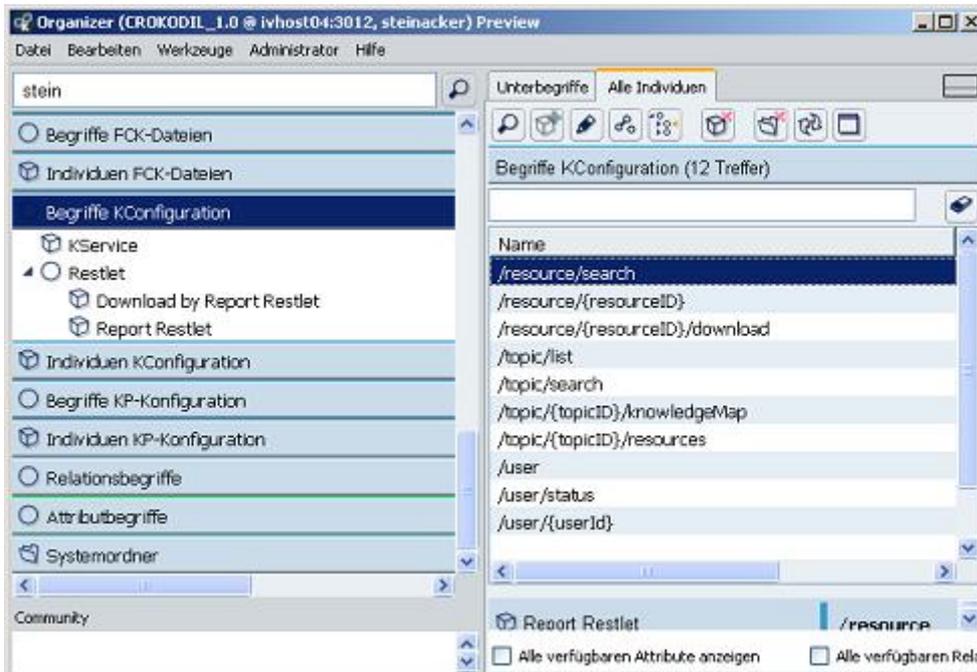
Als Quelle tragen Sie die URL `http://servername:8080/config/rdf` ein.



Der Import zeigt nun alle Änderungen am Schema. Um das neue Schema richtig zu platzieren, Bilden Sie den Begriff "Kconfigurator" auf Ihr Konfiguration-Teilnetz ab. Danach kann der Import durchgeführt werden.



Nach dem Import sind im Konfigurationsteilnetz die neuen Begriffe "KService" und "Restlet" zu sehen. Zur Konfiguration der Restlets werden Individuen dieser Begriffe erzeugt.



3 Konfiguration

HINWEIS: Bitte beachten Sie, dass nach jeder Änderung die Konfiguration erneut eingelesen werden muss.

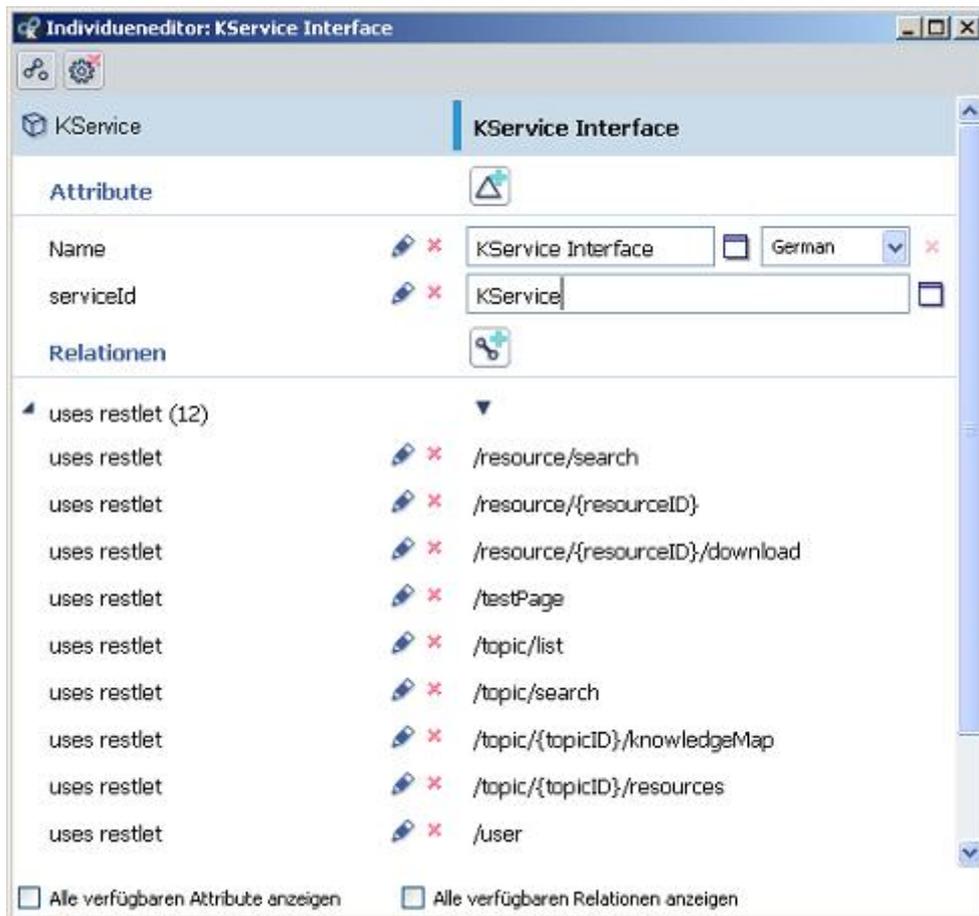
Dies veranlassen Sie durch den Aufruf der URL: <http://servername:8080/config/reload>

3.1 Service Instanz definieren

Zunächst muss eine neue Service-Instanz angelegt werden. Sie haben die Möglichkeit mehrere Service-Instanzen zu konfigurieren, die von unterschiedlichen KService-Applikationen angesprochen werden. Erzeugen Sie dazu ein neues Individuum vom Begriff "KService".

An dieser Instanz muss ein frei wählbarer Name und die serviceld vergeben werden. Die serviceld muss dem serviceldentifier in der kservice.properties entsprechen. Haben Sie diesen nicht geändert, ist der Defaultwert "KService".

Diese Instanz dient als Einstiegspunkt für die Webapplikation. Alle Restlets, die unter dieser Service-Instanz verfügbar sein sollen, müssen auch über die Relation "uses restlet" zugeordnet werden!



3.2 Restlets definieren

3.2.1 Restlets allgemein

Jede Methode, die Bestandteil der Service-API ist, wird durch ein eigenes Restlet-Individuum repräsentiert. Das Restlet wird durch einen frei definierbares Path-Pattern adressiert. Hier sollten die allgemeinen Regeln für RESTful Services eingehalten werden.

Variablen

Im Path-Pattern können Variablen definiert werden. Diese werden in geschweiften Klammern geschrieben. Diese Variablen sind required und somit fester Bestandteil des Path-Patterns. Optionale Variablen werden im QueryString übergeben.

Beispiel:

Pattern: `http://servername:8080/api/user/{userID}/address`

Aufruf: `http://servername:8080/api/user/testUserID/address?show=private`

userID ist eine required Variable. Für die Verarbeitung des Aufrufs, wird die Variable userID mit dem Wert "testUserID" befüllt. Die definierte Logik kann so den passenden Datensatz selektieren.

show ist eine optionale Variable. Sie kann, muss beim Aufruf aber nicht angegeben werden.



In diesem Beispiel würde das Restlet die Variable abfragen. Wenn sie gesetzt ist, wird nur die Private Adresse geliefert. Fehlt die Angabe, werden alle Adressen geliefert.

Operation

Zu den Regeln der RESTful Services gehört die Unterscheidung der Operation durch die Art der Anfrage. Ein Http-GET-Request steht dabei für einen lesenden Zugriff. POST für eine Modifikation, PUT für neu Erzeugen und DELETE für löschen.

Die Kservice-Applikation unterscheidet zwischen diesen Request-Typen und erlaubt bei GET-Request keine Modifikation der Daten! Wählen Sie deshalb bitte immer den passenden Request-Typ.

3.2.2 Report Restlet

Das Report-Restlet führt einen KScript-Report aus und schickt den generierten Output als Response zurück. Alle Variablen aus Path-Pattern und QueryString stehen im KScript als Variablen zur Verfügung...

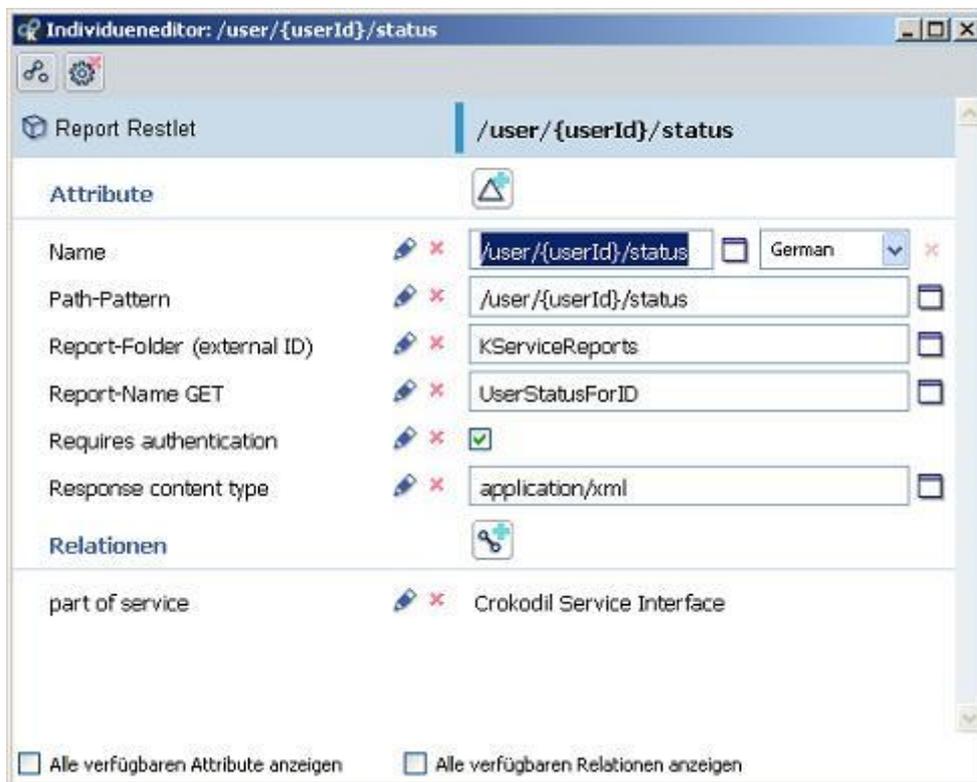
Beim ReportRestlet müssen folgende Konfigurationen vorgenommen werden

Variable	Required	Beschreibung
Path -Pattern	Ja	Pfad zum Restlet ab der Basis-URL, beginnend mit /
ReportFolder	Ja	External ID des ReportFolders, in dem sich der Report befindet
Report-Name GET	Default: GET-Request disabled	Name des Reports der bei einem GET-Request ausgeführt werden soll.
Report-Name POST	Default: POST-Request disabled	Name des Reports der bei einem POST-Request ausgeführt werden soll.
Report-Name PUT	Default: PUT-Request disabled	Name des Reports der bei einem PUT-Request ausgeführt werden soll.
Report-Name DELETE	Default: DELETE-Request disabled	Name des Reports der bei einem DELETE-Request ausgeführt werden soll.
Response content type	Default: application/xml	Content-Type des Response-Headers (z.B. text/html)
Requires authentication	Default : true	Gibt an ob sich der User authentifizieren muss



Execute as system user	Default: false	Ermöglicht die Ausführung als Systemuser. Dies wird z.B. benötigt wenn der User selbst kein Recht hat ein Topic zu modifizieren, die spezielle Funktion jedoch aufrufen können soll.
Part of Service	Ja	Relation zur Service-Instanz, unter der das Restlet verfügbar sein soll

Konfigurationsbeispiel



3.2.3 Download Restlet

Das Download-Restlet ermöglicht den Download eines Blobs aus K-Infinity. Die DMID des Blob-Attribut wird dabei durch einen Report ermittelt. Der Report selbst schreibt nur die DMID des Attributes in den Output. Das Download-Restlet liefert dann die zugehörige Datei an den aufrufenden Client aus.

Hier ist Zu beachten, dass Download-Restlets nur den Request-Typ GET unterstützen!

Variable	Required	Beschreibung
----------	----------	--------------



Path -Pattern	Ja	Pfad zum Restlet ab der Basis-URL, beginnend mit /
ReportFolder	Ja	External ID des ReportFolders, in dem sich der Report befindet
Report-Name	Ja	Name des Reports im Folder. Nur GET-Request möglich
Requires authenticon	Default : true	Gibt an ob sich der User authentifizieren muss
Execute as system user	Default: false	Ermöglicht den Download als Systemuser.
Part of Service	Ja	Relation zur Service-Instanz, unter der das Restlet verfügbar sein soll