



Schnittstellenbeschreibung atlasFX REST



Version 3.1

Stand 11.06.2015

Herausgeber:

alta4 AG

Fleischstraße 57
54290 Trier
Germany

Fon: +49.651.96626.0
Fax: +49.651.96626.26

www.alta4.com
info@alta4.com

Inhaltsverzeichnis

1 Vorbemerkungen	3
2 Search Operation	4
2.1 URL	4
2.2 Parent Ressource	4
2.3 Description	4
2.4 New in 1.9.2	4
2.5 Ressource Hierachy	4
2.6 Parameters	5
2.7 JSON Response Syntax	5
2.8 Example	6
2.9 Demo	8
3 Icon Ressource	9
3.1 URL	9
3.2 Parent Ressource	9
3.3 Description	9
3.4 New in 1.9.2	9
3.5 Ressource Hierachy	9
3.6 Parameters	10
3.7 Response	10
3.8 Example	10

1 Vorbemerkungen

Die REST-Schnittstelle von atlasFX wird typischer Weise über http angesprochen. Sollte das atlasFX hinter einem Proxy mit SSL-Verschlüsselung geschützt sein, kann ebenfalls https verwendet werden. In dieser Dokumentation wird durchgängig http verwendet, obwohl die Aufrufe gegebenenfalls auch über https möglich sind.

Anmerkung: Eine aktuellere Version des Handbuchs finden Sie zukünftig [hier](#).

2 Search Operation

Anmerkung: Eine aktuellere Version des Handbuchs finden Sie zukünftig [hier](#).

2.1 URL

URL

`http://<atlasFX-url>/spring/rest/maps/<mapId>/search`

Anmerkung: Eine aktuellere Version des Handbuchs finden Sie zukünftig [hier](#).

2.2 Parent Ressource

Parent Ressource

Map

Anmerkung: Eine aktuellere Version des Handbuchs finden Sie zukünftig [hier](#).

2.3 Description

Description

Diese Operation stellt eine Suchfunktionalität zu Verfügung, die es einbindenden Webseiten erlaubt ein vorschlagslistenunterstützendes Suchfeld mit Icons und Verlinkung zu einer atlasFX Karte zu implementieren.

Anmerkung: Eine aktuellere Version des Handbuchs finden Sie zukünftig [hier](#).

2.4 New in 1.9.2

New in 1.9.2

Diese Schnittstelle steht erstmals seit der Version 1.9.2 zu Verfügung.

Anmerkung: Eine aktuellere Version des Handbuchs finden Sie zukünftig [hier](#).

2.5 Ressource Hierachy

Ressource Hierachy

- **MapList** `http://<atlasFX-url>/spring/rest/maps`
 - **Map** `/<mapId>`
 - **Search Operation** `/search`

Anmerkung: Eine aktuellere Version des Handbuchs finden Sie zukünftig [hier](#).

2.6 Parameters

Parameters

Parameter	Details
searchTerm	Hier soll die Benutzereingabe bzw. der Suchbegriff übergeben werden
urlTemplate	In diesem Parameter soll die Vorlage (Template) für die aufzurufende Url übergeben werden. Die Aufrufmöglichkeiten sind bei den jeweiligen Clients dokumentiert. Diese Vorlage kann die drei Template-Variablen \$mapId\$, \$layerId\$ und \$featureId\$ enthalten die für jedes Ergebnis individuell ersetzt werden. Dieser Mechanismus ermöglicht eine maximale Flexibilität bei der Konfiguration der Karte. Damit sind alle Funktionalitäten der verschiedenen Clients in Kombination mit sämtlichen Netzwerkkonfigurationsmöglichkeiten verwendbar.
useDefaultIcons	Dieser Parameter ist optional. Hier kann gesteuert werden, ob die atlasFX Default Icons verwendet werden sollen, falls kein Icon am Layer konfiguriert wurde. Gültige Werte sind hier true oder false.

Anmerkung: Eine aktuellere Version des Handbuchs finden Sie zukünftig [hier](#).

2.7 JSON Response Syntax

JSON Response Syntax

```
[
  {
    "preview" : <previewText>,
    "url" : <mapClientUrl>,
    "iconId" : <iconId>
  } //, ...
]
```

Anmerkung: Eine aktuellere Version des Handbuchs finden Sie zukünftig [hier](#).

2.8 Example

Example

Es folgt ein Beispiel, das online unter <http://roadshow.alta4cloud.com/atlasfx/js/index.html?mapId=530> nachvollzogen werden kann.

Erläuterung zum Testsystem

<atlasFX-url>	<code>http://roadshow.alta4cloud.com/atlasfx</code>	Dies ist die Url unter der der Context der Webapplication verfügbar ist.
<mapId>	530	In diesem Beispiel verwenden wir die Karte 530. Diese Id identifiziert eine atlasFX-Karte, die einfach im atlasFX CMS abgelesen werden kann.

Zunächst müssen die Parameter bestückt werden. Der erste Parameter `searchTerm` ist `tivial`. Hier wird der eingegebene Suchbegriff übergeben. Bei Verwendung von `Http Get` ist zu beachten, dass die Parameter für die Verwendung in der Url encodiert werden müssen (Url Encoding). Das atlasFX Backend verwendet UTF-8 als Uri Encoding. Daher müssen Umlaute zum Beispiel 'ö' als `%C3%B6` kodiert werden.

Als zweiten Parameter muss ein Template für die zu generierenden Urls übergeben werden. Mit dieser Schnittstelle können Sie beliebige Urls generieren lassen, die mit Werten des Suchergebnisses parametrisiert werden. In diesem Beispiel soll einfach der atlasFX JS-Client mit dem gefundenen Objekt aufgerufen werden. Die erste Herausforderung besteht darin, dass das atlasFX-Backend nicht weiß, unter welcher Url es vom Internet aus erreichbar ist. Daher geben wir die `<atlasFX-url>` im Template Parameter mit. In diesem Beispiel wäre das `http://roadshow.alta4cloud.com/atlasfx`. Da die atlasFX-url alleine nicht ausreicht um den JavaScript-Client aufzurufen wird an den Pfad noch `/js/index.html` angehängen. Das Url Template sieht nun wie folgt aus: `http://roadshow.alta4cloud.com/atlasfx/js/index.html`.

Weiterhin ist es erforderlich dem JS-Client eine `MapId` zu übergeben. Hier kommt wieder unsere Beispiel-`MapId 530` zum Einsatz. Es ist wichtig die gleiche Karte aufzurufen, auf der auch die Suche selbst durchgeführt worden ist, da die Suchergebnisse eindeutig einer Karte zuzuordnen sind und in einer anderen Karte nicht aufgerufen werden können. Die `MapId` wird dem Client als Url Parameter `mapId` übergeben. Für weitere Informationen zum Aufruf des Clients schauen Sie bitte in der Dokumentation des Clients nach. Sie können übrigens auf den Flex-Client nach gleichem Schema aufrufen. Hierzu ist eine etwas andere Url zu verwenden. Mit der `MapId` sieht unser Url Template nun so aus: <http://roadshow.alta4cloud.com/atlasfx/js/index.html?mapId=530>. Um Fehler zu vermeiden, können Sie auch die Template-Variable `$mapId$` verwenden. Diese Variable wird durch die Id der Karte ersetzt auf der Sie die Suche ausführen. Unter Verwendung der Template-Variablen sieht das Url Template nun so aus: `http://roadshow.alta4cloud.com/atlasfx/js/index.html?mapId=$mapId$`. Nach der Prozessierung der Anfrage an die REST-Schnittstelle ist die Template-Variable durch den aktuellen Wert ersetzt worden. Für eine Anfrage unter Verwendung der Karte 530 würde die zurückgegebene Url nun so `http://roadshow.alta4cloud.com/atlasfx/js/index.html?mapId=530` zurückgegeben werden. Meistens ist es Sinn und Zweck dieser



Suchanfrage das gefundene Objekt, im GIS-Kontext Feature genannt in einer Karte darzustellen. Daher muss das darzustellende Feature an den Karten-Client übergeben werden. Der atlasFX-JS-Client stellt dazu die Url Parameter `mapLayerId` und `featureId` bereit. In atlasFX wird ein Feature durch eine Id, die innerhalb eines Layers eindeutig ist, und die entsprechende Layer Id eindeutig beschrieben. Diese beiden Ids stehen ebenfalls als Template-Variable zu Verfügung. Ergänzen wir nun abschließend unser Url Template zu `http://roadshow.alta4cloud.com/atlasfx/js/index.html?mapId=$mapId$%26featureId=$featureId$%26mapLayerId=$layerId$`. Bitte beachten Sie hier, dass wir das `&`-Zeichen innerhalb des Url Templates mit `%26` escaped wurde, damit dieses nicht beim Aufruf der Suchschnittstelle ausgewertet wird.

Den optionalen Parameter `useDefaultIcons` lassen wir im ersten Beispiel weg.

Beispiel Parameter

Nehmen wir an, der Benutzer tippt 'Martin-Grundschule' in das Suchfeld ein.

Parameter	Beispielwert
<code>searchTerm</code>	Martin-Grundschule
<code>urlTemplate</code>	<code>http://roadshow.alta4cloud.com/atlasfx/js/index.html?mapId=26featureId=\$featureId\$%26mapLayerId=\$layerId\$</code>

Daraus ergibt sich folgender Aufruf der atlasFX-REST-Schnittstelle

[http://roadshow.alta4cloud.com/atlasfx/spring/rest/maps/530/search?searchTerm=Martin-Grundschule&urlTemplate=http://roadshow.alta4cloud.com/atlasfx/js/index.html?mapId=\\$mapId\\$%26featureId=\\$featureId\\$%26mapLayerId=\\$layerId\\$](http://roadshow.alta4cloud.com/atlasfx/spring/rest/maps/530/search?searchTerm=Martin-Grundschule&urlTemplate=http://roadshow.alta4cloud.com/atlasfx/js/index.html?mapId=$mapId$%26featureId=$featureId$%26mapLayerId=$layerId$)

Die Antwort des Suchservice

```
[
  {
    "iconId": 4,
    "preview": "Trier,GS Trier Martin",
    "url": "http://roadshow.alta4cloud.com/atlasfx/js/index.html?
mapId=530&featureId=657&mapLayerId=2889"
  }
]
```

Hier ist zu erkennen, dass lediglich ein Objekt zurückgeliefert wurde. Das Ergebnisobjekt weist die drei Properties `iconId`, `preview` und `url` auf. Wie Sie das Icon anhand seiner Id abfragen können wird im Abschnitt [Icon](#)



[Ressource](#) erläutert. Das Property preview enthält eine Zeichenkette die das Ergebnis beschreibt. Sie kann zum Beispiel in der Vorschlagsliste dargestellt werden. Das Property `url` enthält die Url, die sich durch Variablenersetzung im Url Template entstanden ist. Diese kann zum Beispiel aufgerufen werden um den Karten-Client zu starten.

Anmerkung: Eine aktuellere Version des Handbuchs finden Sie zukünftig [hier](#).

2.9 Demo

Es ist eine [Demoanwendung](#) verfügbar.

Anmerkung: Eine aktuellere Version des Handbuchs finden Sie zukünftig [hier](#).

3 Icon Ressource

Anmerkung: Eine aktuellere Version des Handbuchs finden Sie zukünftig [hier](#).

3.1 URL

URL

`http://<atlasFX-url>/spring/rest/icons/<iconId>`

Anmerkung: Eine aktuellere Version des Handbuchs finden Sie zukünftig [hier](#).

3.2 Parent Ressource

Parent Ressource

IconList

Anmerkung: Eine aktuellere Version des Handbuchs finden Sie zukünftig [hier](#).

3.3 Description

Description

Diese Ressource enthält die Bilddaten eines Icons.

Anmerkung: Eine aktuellere Version des Handbuchs finden Sie zukünftig [hier](#).

3.4 New in 1.9.2

New in 1.9.2

Diese Ressource steht erstmals seit der Version 1.9.2 zu Verfügung.

Anmerkung: Eine aktuellere Version des Handbuchs finden Sie zukünftig [hier](#).

3.5 Ressource Hierachy

Ressource Hierachy

- **IconList** `http://<atlasFX-url>/spring/rest/icons`
 - **Icon** `/<iconId>`

Anmerkung: Eine aktuellere Version des Handbuchs finden Sie zukünftig [hier](#).

3.6 Parameters

Parameters

Diese Ressource unterstützt keine Url Parameter.

Anmerkung: Eine aktuellere Version des Handbuchs finden Sie zukünftig [hier](#).

3.7 Response

Response

Die Bilddatei wird mit binär zurückgegeben. Die Responseheader werden entsprechend gesetzt, sodass die Url direkt im Browser verwendet werden kann

Anmerkung: Eine aktuellere Version des Handbuchs finden Sie zukünftig [hier](#).

3.8 Example

Example

Es folgt ein Beispiel, das online unter <http://roadshow.alta4cloud.com/atlasfx/js/index.html?mapId=530> nachvollzogen werden kann.

Erläuterung zum Testsystem

<atlasFX-url>	http://roadshow.alta4cloud.com/atlasfx	Dies ist die Url unter der der Context der a Webapplication verfügbar ist.
<iconId>	4	In diesem Beispiel möchten wir das Icon 4 abrufen.

Das Icon kann einfach unter <http://roadshow.alta4cloud.com/atlasfx/spring/rest/icons/4> abgerufen werden.

Anmerkung: Eine aktuellere Version des Handbuchs finden Sie zukünftig [hier](#).