

# Search/Retrieve via URL (SRU)

SRU steht für Search/Retrieve via URL. Über SRU-Schnittstellen können gezielt bestimmte Indizes im jeweiligen System nach bestimmten Begriffen durchsucht werden.

Einen guten Einstieg in SRU bietet [Wikipedia](#). Details zum SRU-Standard liefert die offizielle [Dokumentation](#) der Library of Congress.

## Explain-Operation

Mit der Explain-Operation werden Informationen zu Abfragemöglichkeiten angezeigt:

```
<Basis-URL>?operation=explain
```

Beispiel:

[https://uzb.swisscovery.slsp.ch/view/sru/41SLSP\\_UZB?version=1.2&operation=explain](https://uzb.swisscovery.slsp.ch/view/sru/41SLSP_UZB?version=1.2&operation=explain)

Unter dem XML-Tag `<indexInfo>` werden die Namen der durchsuchbaren Indices gelistet. Pro Index wird ausgewiesen, welche sogenannten *Relation Operators* erlaubt sind. Beispiel:

```
<index>
  <ns:title>dc:title</ns:title>
  <map>
    <name set="alma">dc_title</name>
  </map>
  <configInfo>
    <supports type="relation">all</supports>
    <supports type="relation">=</supports>
    <supports type="relation">==</supports>
    <supports type="emptyTerm"/>
  </configInfo>
</index>
```

Unter dem Tag <schemaInfo> sind die verfügbaren Metadatenformate aufgelistet. Aus dem untenstehenden Beispiel geht hervor, dass die betreffende SRU-Schnittstelle z.B. mit recordSchema=dc abgefragt werden kann, um Metadaten im Dublin Core Format zurückzubekommen.

```
<schemaInfo>
  <schema identifier="http://www.loc.gov/standards/marcxml/schema/MARC21slim.xsd" name="marcxml"
sort="true"/>
  <schema identifier="info:srw/schema/1/dc-v1.1" name="dc" sort="true"/>
  <schema identifier="info:srw/schema/1/mods-v3.5" name="mods" sort="true"/>
  <schema identifier="info:srw/schema/1/dcx-v1.0" name="dcx" sort="true"/>
  <schema identifier="info:srw/schema/8/unimarcxml-v0.1" name="unimarcxml" sort="true"/>
  <schema identifier="http://www.nl.go.kr/kormarc/" name="kormarcxml" sort="true"/>
  <schema identifier="http://www.nlc.cn/" name="cnmarcxml" sort="true"/>
  <schema identifier="http://www.loc.gov/standards/iso20775/" name="isohold" sort="true"/>
</schemaInfo>
```

## searchRetrieve-Operation

Für die eigentliche Suchabfrage wird die searchRetrieve-Operation genutzt:

```
<Basis-URL>?operation=searchRetrieve
```

## Contextual Query Language (CQL)

Suchanfragen werden in der [Contextual Query Language](#) (CQL) formuliert. Mithilfe von Booleschen Operatoren (and, or, not) können komplexe Abfragen erstellt werden.

Spezielle Zeichen wie Leerschläge und Anführungszeichen müssen URL-encoded werden. Beispiele:

```
%20 Leerschlag
%22 Anführungszeichen
%3D Gleichheitszeichen
%2F Forward slash
%3A Doppelpunkt
%2E Punkt
%2D minus
```

Für das URL-Encoding kann z.B. auch dieses Online-Tool genutzt werden: <https://urlencoder.org>.

# Durch Resultatseiten blättern

Bei einer Query wird jeweils nur eine Teilmenge der Treffer direkt zurückgeliefert. Dies ist serverseitig definiert - oft sind es 10 Records, sozusagen die erste Resultatseite. Die absolute Anzahl Treffer kann aus dem Tag `<numberOfRecords>` herausgelesen werden. Beispiel:

```
<numberOfRecords>629</numberOfRecords>
```

Im Element `<nextRecordPosition>` kann überprüft werden, welches Record als nächstes kommt. Mit dieser Zahl kann die nächste Resultatseite aufgerufen werden, indem der `startRecord`-Parameter an die vorangegangene Suche angehängt wird. Beispiel:

[https://uzb.swisscovery.slsp.ch/view/sru/41SLSP\\_UZB?version=1.2&operation=searchRetrieve&recordSchema=dc&query=alma.title=python&startRecord=11](https://uzb.swisscovery.slsp.ch/view/sru/41SLSP_UZB?version=1.2&operation=searchRetrieve&recordSchema=dc&query=alma.title=python&startRecord=11)

---

Version #5

Erstellt: 5 August 2022 09:43:11 von Team Bibliotheks-informatik

Zuletzt aktualisiert: 20 September 2022 15:01:14 von Team Bibliotheks-informatik