

Introduction

Terminology

- ▶ **Such as:** classifications, thesauri, ontologies...
- ▶ **Known as:** terminologies, knowledge organization systems, semantic artefacts, concept schemes, knowledge graphs...
- ▶ **Core idea:** things, not strings
- ▶ **Done with:** URLs, labels & links between

JSKOS In a nutshell

- ▶ Encoding SKOS and related artefacts (concepts, mappings, concordances...) in JSON
- ▶ Knowledge graph data format with focus on terminologies
- ▶ Established infrastructure for controlled vocabularies at VZG
- ▶ Mature specification at <https://gbv.github.io/jskos/>
JSKOS 0.7.0 just released yesterday

Background and motivation

- ▶ Project **coli-conc** (2013-): infrastructure for (mappings between) knowledge organization systems
- ▶ **Cocoda** web application for mapping:
 - ▶ Design based on service-oriented architecture (SOA)
 - ▶ Terminologies, mappings, accounts...
loosely coupled via APIs in the client \Rightarrow JSON
- ▶ Integration of federated data \Rightarrow RDF

Influences and references

- ▶ **SKOS** for core classes and properties
- ▶ **JSON-LD** and existing ontologies (DCAT, vCard, XKOS, SSSOM...) for URIs and mapping to RDF
- ▶ Skosmos and other **APIs**
- ▶ **Wikibase** Data Model
- ▶ **Standards:** [Web Annotation Data Model](#) for annotations, [GeoJSON](#) for geospatial data, [IIIF](#) for media, RFC 6570 for URI Templates, XSD for Data Types...

JSKOS Features by examples

Concepts

```
{  
  "type": [  
    "http://www.w3.org/2004/02/skos/core#Concept"  
  ],  
  "notation": [ "641" ],  
  "uri": "http://dewey.info/class/641/e23/",  
  "prefLabel": {  
    "en": "Food & drink",  
    "de": "Essen und Trinken"  
  }  
}
```

Class **641** from Dewey Decimal Classification (DDC)

Mappings

```
{
  "type": [ "http://www.w3.org/2004/02/skos/core#exactMatch" ],
  "from": {
    "memberSet": [ {
      "uri": "http://dewey.info/class/641/e22/",
      "prefLabel": { "en": "Food & drink" }
    } ]
  },
  "to": {
    "memberSet": [ {
      "uri": "http://www.wikidata.org/entity/Q118451828",
      "prefLabel": { "en": "food and beverage" }
    } ]
  },
  "justification":
    "https://w3id.org/semapv/vocab/ManualMappingCuration",
  "tool": [ { "url": "https://coli-conc.gbv.de/cocoda/" } ]
}
```


Concept Bundles

```
{  
  "uri": "http://dewey.info/class/641.50902/e23/",  
  "notation": [ "641.50902" ],  
  "memberList": [ {  
    "uri": "http://dewey.info/class/641/e23/",  
    "prefLabel": { "en": "Food & drink" }  
  }, {  
    "uri": "http://dewey.info/class/1--0902/e23/",  
    "prefLabel": {  
      "en": "6th to 15th century (500-1499)"  
    }  
  } ]  
}
```

Combined DDC class for “medieval cuisine”

Concept Occurrences

```
{  
  "database": {  
    "uri": "http://uri.gbv.de/database/opac-de-627"  
  },  
  "memberSet": [ {  
    "uri": "http://dewey.info/class/641.50902/e23/",  
    "prefLabel": { "en": "Fodd & drink" }  
  } ],  
  "count": 18,  
  "modified": "2025-11-17",  
  "url": "https://opac.k10plus.de/DB=2.299/CMD?ACT=SRCHA&I"  
}
```

18 records on medieval cuisine in K10plus catalogue

JSKOS Overview

JSKOS Object types

- ▶ Superclasses
 - ▶ **Item** (named entity) and **Resource** (anything)
 - ▶ ***Concept Bundles*** (combined concepts)
- ▶ ***Concepts*** and **Concept Schemes**
- ▶ ***Mappings*** and **Concordances**
- ▶ ***Occurrences*** of Concepts
- ▶ **Registries**, **Distributions** and **Services** to collect and access
- ▶ Subsets of JSON formats IIIF, GeoJSON, and Web Annotation

Common fields of JSKOS items

- ▶ prefLabel, altLabel...
- ▶ identifier
- ▶ previous, next
- ▶ startDate, endDate
- ▶ location, address, media...
- ▶ version, versionOf
- ▶ ...

Qualified statements

country



Germany ...

start time

3 October 1990

▶ 1 reference



West Germany ...

end time

2 October 1990

start time

23 May 1949

Qualified statements

```
{ "prefLabel": { "en": "Göttingen" },
  "qualifiedRelations": {
    "http://www.wikidata.org/entity/P17": [ {
      "resource": {
        "uri": "http://www.wikidata.org/entity/Q183",
        "prefLabel": { "en": "Germany" }
      },
      "rank": "preferred",
      "startDate": "1990-10-03",
      "source": [ { ... } ]
    }, {
      "resource": {
        "uri": "http://www.wikidata.org/entity/Q713750",
        "prefLabel": { "en": "West Germany" }
      },
      "rank": "normal",
      "startDate": "1949-05-23", "endDate": "1990-10-02" } ] } }
```

Qualified statements

- ▶ Inspired by Wikibase data model
- ▶ Ranking (preferred/normal/deprecated)
- ▶ References
- ▶ Arbitrary fields, identified by URIs
 - ⇒ Full knowledge graph format
- ▶ Flexibility & extensibility at cost of full interoperability

Closed world statements

- ▶ *missing field*: unknown
- ▶ `[]`: no value exists
- ▶ `[..., null]`: more values exist

Applications of JSKOS

Use cases

- ▶ Cocoda, BARTOC, DANTE... at VZG
- ▶ NFDI4Objects & TS4NFDI
- ▶ Slowly being adopted outside of VZG
(2 or 3 implementations)

Applications & Libraries

- ▶ **jskos-server**: DBMS to store JSKOS data
- ▶ **jskos-vue**: widgets to nicely display JSKOS objects in browser (e.g. tree)
- ▶ **jskos-proxy**: serve and browse JSKOS data (terminologies) as Linked Open Data
- ▶ **jskos-tools**: helper functions for JSKOS data
- ▶ **jskos-cli**: tools to validate and convert JSKOS data
- ▶ **subjects-api**: query subject indexing of K10plus catalogue in JSKOS via JSKOS-API

Relevant, but not focus of this talk.

- ▶ GET/POST/PUT/PATCH data in JSKOS format
- ▶ Defined read & write endpoints for concepts, mappings, annotations...
- ▶ Optional authentication included
- ▶ Multiple implementations exist (servers and clients)!

Beyond JSKOS

Some limitations of JSKOS

- ▶ Labels MUST have language tag (use und for undefined)
- ▶ Notations don't have data types
- ▶ No support of SKOS Collections (yet?)
- ▶ Fields about entities and metadata are mixed (e.g. startDate vs. created)
- ▶ Wikibase data model only covered partially
- ▶ No standards body behind the specification

Alternatives

- ▶ Plain RDF or RDF*
- ▶ JSON-LD
- ▶ Property Graphs

Limitations of RDF

Most elements of JSKOS can directly be mapped from/to RDF!

- ▶ Agreement on global URIs required
- ▶ Triple store required to access data
- ▶ Reification (e.g. data about mappings)
- ▶ RDF* adds more complexity

Limitations of JSON-LD

Most of JSKOS is JSON-LD, there is a non-normative JSON-LD context!

- ▶ Triple store required to access data (!?)
- ▶ Multiple ways to express the same data
- ▶ Only JSON Schema can guarantee a strict JSON format

Property graphs

- ▶ Easier to work with than RDF/SPARQL (tools)
- ▶ Not for merging data
- ▶ No standard data format exists (see [Property Graph Exchange Format](#) for a proposal)

When to use JSKOS

- ▶ Access to terminology APIs from VZG, in particular mappings
- ▶ Processing of semantic artefacts in JSON
- ▶ RDF compatibility without RDF technology stack

Please ask & try out!