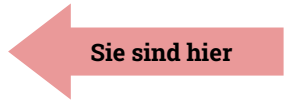


Potenziale von **Linked Open Data** entdecken

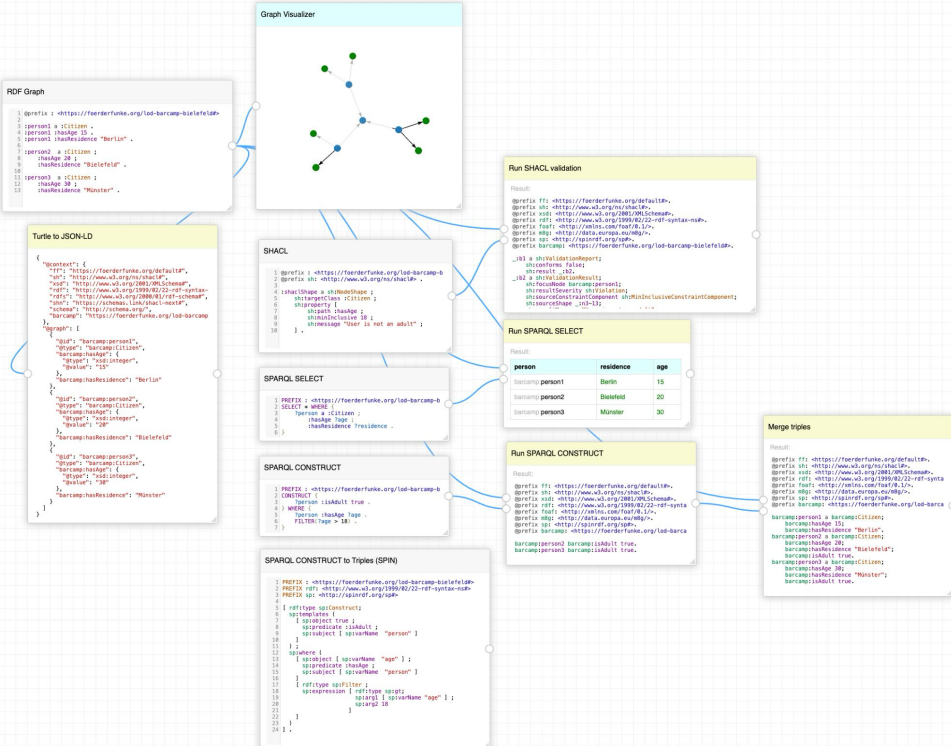
Hands-on-Sessions zur Entwicklung eines Knowledge Graphs
anhand eines konkreten Use Cases

21. März 2026, Benjamin Degenhart

Agenda

- 13:00-14:00 **Session 1: Präsentation & Demo** 
 - Intro: Linked Data
 - Blick ins Portal
 - Use Case 1: Besucherzahlen Bibliotheken (Einstiegsbeispiel)
 - Use Case 2: Familienfreundliches München (Deep Dive)
 - Diskussion
- 14:30-15:30 **Session 2: Hands-On**
 - Setup: Starter-Kit + Triplestore
 - Selbst ans Werk 🛠️🧐
 - Austausch

Intro: Linked Data



Vorteile von LOD

- Daten sind selbstbeschreibend
- Mit Stift & Papier visualisierbar → Exploration, Didaktik
- Graphentheorie unmittelbar anwendbar
- Treiber für Kollaboration: Nachnutzung auf tiefster Ebene
- Gartner: Schlüsseltechnologie für KI
- Semantische Technologien im Deutschland-Stacks (siehe [hier](#) und [hier](#))



"By 2030, universal semantic layers will be treated as critical infrastructure, alongside data platforms and cybersecurity."

— Source: Gartner Data & Analytics Summit 2026, Signature Series.



Quellen:

- <https://www.strategy.com/ja/software/blog/why-gartner-just-put-the-semantic-layer-on-the-same-level-as-cybersecurity>
- <https://juansequeda.substack.com/p/gartner-data-and-analytics-march>

Nachteile

- Tooling teilweise mühsam
- Fachanwendungen bieten oft keinen RDF-Export an
- "Extra-Invest" für Datenbereitsteller oft noch schwer verargumentierbar

Blick ins Portal

Use Case 1: **Besucherzahlen Bibliotheken**

Vergleich des relativen Wachstums der jährlichen
Besucherzahlen der bayerischen Bibliotheken.

26 Dateien der Deutschen Bibliotheksstatistik

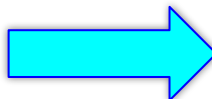
NR	Kurzname	Bibliotheksname	DBS-ID	Besuche
1	Abenberg GB	Kath. Öffentliche Bücherei	PA604	2.850
2	Abensberg StB	Stadtbücherei	QS450	11.123
3	Abensberg StB	Stadtbücherei Offenstetten	NR768	1.480
4	Achslach GB	Gemeindebibliothek	LR289	80
5	Adelschlag GB	Volksbücherei St. Michael-Pietenfeld	PG862	460
6	Adelschlag GB	Gemeindebücherei Ochsenfeld	PG904	780

Mapping von DBS-ID zu
Regionalschlüssel von lobid.org

dbSID	rs
DR159	053660004004
DS607	059150000000
DU562	031590034034

Bevölkerungszahlen, Bayerisches Landesamt für Statistik

AGS	Name	Stichtag	Insgesamt
09162	München, Landeshauptstadt	31.12.2024	1505005
09162	München, Landeshauptstadt	31.12.2023	1488719
09162	München, Landeshauptstadt	31.12.2022	1490516
09162	München, Landeshauptstadt	31.12.2021	1487708
09162	München, Landeshauptstadt	31.12.2020	1488202



```
lobid:AA241 a dev:Library;  
  schema:alternateName "Kulmbach StB";  
  schema:name "Bücherei am Stadtpark";  
  dev:dbsId "AA241";  
  dev:hasAnnualStatistic  
    dev:libstat-AA241-2005,  
    # ...  
    dev:libstat-AA241-2024.
```

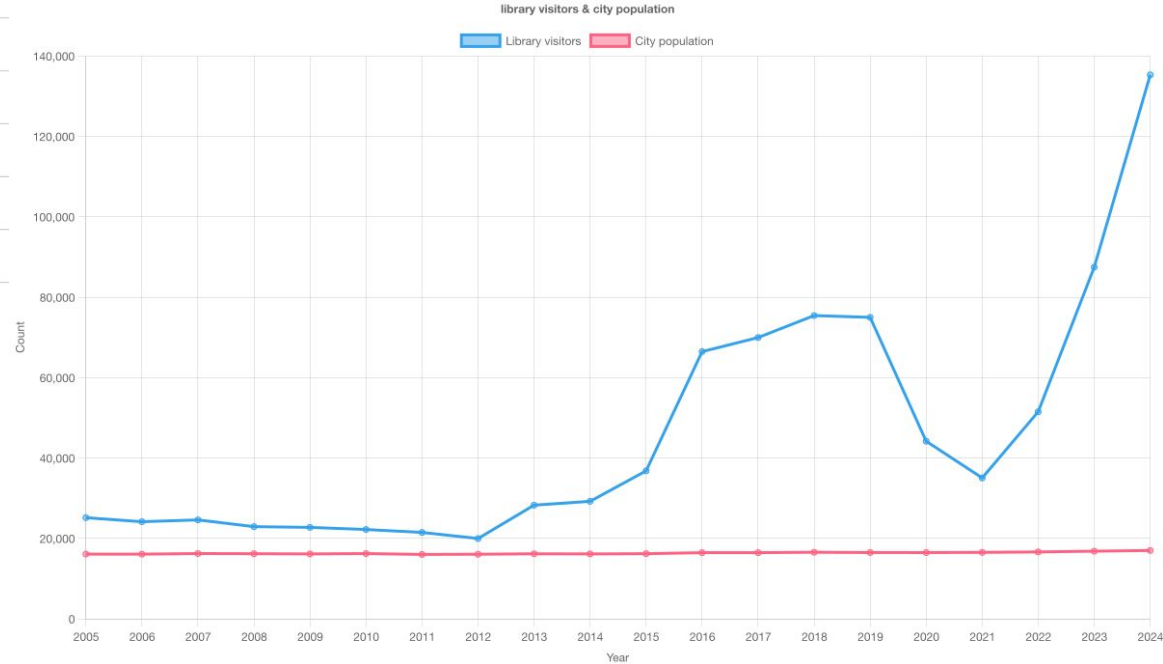
```
dev:libstat-AA241-2024 a dev:LibraryAnnualStatistic;  
  dev:hasVisitCount 44244;  
  dev:year 2024.
```

```
dev:dbsIdToAgs-9435 a dev:DbsIdToAgsMapping;  
  dev:hasAgs "09477128";  
  dev:hasDbsId "AA241".
```

```
dev:city-09477128 a dev:City;  
  schema:name "Kulmbach, GKSt";  
  dev:hasAgs "09477128";  
  dev:hasAnnualStatistic  
    dev:citystat-09477128-2005,  
    # ...  
    dev:citystat-09477128-2024.
```

```
dev:citystat-09477128-2024 a dev:CityAnnualStatistic;  
  dev:hasPopulation 26625;  
  dev:year 2024.
```

libName	cityName	avgLibVisitsPer1kGrowthPerYear
Stadtbücherei	Penzberg, St	336
Bibliotheks- und Informationszentrum	Haßfurt, St	310
Stadtbücherei	Eibelstadt, St	287
Gemeindebücherei	Mertingen	254
Gemeindebücherei	Buttenheim, M	248
Bücherei Bernbeuren	Bernbeuren	240
Gemeindebücherei	Falkenfels	218
Pfarr- und Gemeindebücherei	Perach	213
Stadtbibliothek	Volkach, St	191
Stadtbücherei	Bad Aibling, St	188



```

1 import { QueryEngine } from '@comunica/query-sparql'
2 import { fileURLToPath } from 'url'
3 import iconv from 'iconv-lite'
4 import path from 'path'
5 import fs from 'fs'
6
7 const ENDPOINT = 'https://open.bydata.de/api/sparql'
8 const LOBID_URL = 'https://lobid.org/organisations/search?q&dbid=0&format=csv&dbid=0'
9 const engine = new QueryEngine({
10   const OUTPUT_DIR = path.join(path.dirname(fileURLToPath(import.meta.url)), 'csvs')
11   fs.mkdirSync(OUTPUT_DIR, { recursive: true })
12
13 // 1. Library files
14
15 async function downloadLibraryFiles() {
16   let query = `
17     PREFIX dcot: <http://www.w3.org/ns/dcat#>
18     PREFIX dc: <http://purl.org/dc/terms/>
19     PREFIX filetype: <https://publications.europa.eu/...>
20     SELECT ?url WHERE {
21       ?ds a dcat:Dataset ;
22         dct:title ?title .
23       FILTER (SYSTRANS(?title, "Öffentliche BIBL...")
24       ?dsat dcat:distribution ?dist .
25       ?dist dcot:format filetype:CSV ;
26         dcat:downloadURL ?url .
27     }`
28   const bindingsStream = await engine.queryBindings(query, {
29     const bindings = await bindingsStream.toArray()
30     for (let binding of bindings) {
31       let url = binding.get('url').value
32       const res = await fetch(url)
33       const buffer = Buffer.from(await res.arrayBuffer())
34       // even so the header response says it's utf-8
35       let text = iconv.decode(buffer, 'windows-1252')
36       let lines = text.split(/\r\n/)
37       const year = lines[0].split(';').map(s => s.trim())
38       const last = lines.length - 1
39       const remove = new Set([0, 1, 3, last, last - 1])
40       // clean up csv
41       lines = lines.filter((_, i) => !remove.has(i))
42       const file = `${OUTPUT_DIR}/libraries-${year}.csv`
43       fs.writeFileSync(file, lines.join('\n'), { utf8: true })
44       console.log('Wrote', file)
45     }
46   }
47
48 await downloadLibraryFiles()
49
50 // 2. Population file "Bürgerinnen und Bürger", https://...
51
52 async function downloadPopulationFile() {
53   let query = `
54     PREFIX dcot: <http://www.w3.org/ns/dcat#>
55     PREFIX dc: <http://purl.org/dc/terms/>
56     SELECT ?url WHERE {
57       ?ds a dcat:Dataset ;
58         dct:title ?title .
59       FILTER (SYSTRANS(?title, "Bürgerinnen und Bürger")
60       ?dsat dcat:distribution ?dist .
61       ?dist dcot:format filetype:CSV ;
62         dcat:downloadURL ?url .
63     }`
64   const bindingsStream = await engine.queryBindings(query, {
65     const bindings = await bindingsStream.toArray()
66     let url = bindings[0].get('url').value
67     const res = await fetch(url)
68     const buffer = Buffer.from(await res.arrayBuffer())
69     let text = iconv.decode(buffer, 'windows-1252')
70     let lines = text.split(/\r\n/)
71     const last = lines.length - 1
72     const remove = new Set([0, 1, 2, 3, 4, last, last - 1, last - 2, last - 3])
73     lines = lines.filter((_, i) => !remove.has(i))
74     lines.unshift('AGS;Name;Stichtag;Insgesamt;männlich;weiblich')
75     const file = `${OUTPUT_DIR}/bevoelkerung.csv`
76     fs.writeFileSync(file, lines.join('\n'), { utf8: true })
77     console.log('Wrote', file)
78   }
79
80 await downloadPopulationFile()
81
82 // 3. Mapping of DBS-ID to rs (Regionalschlüssel, an extension of the Amtliche Gemein...
83
84 async function downloadLobidMapping() {
85   const res = await fetch(LOBID_URL)
86   const text = await res.text()
87   const file = `${OUTPUT_DIR}/lobid-mapping.csv`
88   fs.writeFileSync(file, text, { utf8: true })
89
90 await downloadLobidMapping()

```

02-libraries

csvs

- bevoelkerung.csv
- libraries-1999.csv
- libraries-2000.csv
- libraries-2001.csv
- libraries-2002.csv
- libraries-2003.csv
- libraries-2004.csv
- libraries-2005.csv
- libraries-2006.csv
- libraries-2007.csv
- libraries-2008.csv
- libraries-2009.csv
- libraries-2010.csv
- libraries-2011.csv
- libraries-2012.csv
- libraries-2013.csv
- libraries-2014.csv
- libraries-2015.csv
- libraries-2016.csv
- libraries-2017.csv
- libraries-2018.csv
- libraries-2019.csv
- libraries-2020.csv
- libraries-2021.csv
- libraries-2022.csv
- libraries-2023.csv
- libraries-2024.csv
- lobid-mapping.csv

- build-graphs.js
- download-csvs.js

```

1 import { addTtl, getRdf, nestStore, storeToTurtle } from '@foerderung/see-ops-utils'
2 import { fileURLToPath } from 'url'
3 import Papa from 'papaparse'
4 import path from 'path'
5 import fs from 'fs'
6
7 const THIS_DIR = path.dirname(fileURLToPath(import.meta.url))
8 const CSV_DIR = path.join(THIS_DIR, 'csvs')
9 const OUTPUT_TTL = path.join(THIS_DIR, 'triples.ttl')
10 const prefixes = {
11   'http://www.w3.org/2001/XMLSchema#': 'xsd:',
12   'http://www.w3.org/2001/XMLSchema#': 'xsd:',
13   'http://lobid.org/organisations/DE-': 'lobid:',
14   'http://schema.org/': 'schema:',
15   'http://www.w3.org/1999/02/22-rdf-syntax-ns#': 'rdf:'
16 }
17 const a = 'http://www.w3.org/1999/02/22-rdf-syntax-ns#type'
18 const dev = (localName) => `https://open.bydata.de/api/hub/dev/${localName}`
19
20 let store = nestStore()
21
22 function parseLibraryFile(filePath, filename) {
23   const year = Number(filename.split('-')[1])
24   if (year < 2005) return // we start with 2005 because in prior years "Besuche" was sparsely filled
25   const csv = Papa.parse(fs.readFileSync(filePath, 'utf8'), { header: true, skipEmptyLines: true,
26
27   for (let row of csv.data) {
28     if (row.Besuche !== "0") continue
29     const sub = `https://lobid.org/organisations/DE-${year}`
30     addRdf(store, sub, a, dev('library'))
31     addRdf(store, sub, 'http://schema.org/alternateName', row.name)
32     addRdf(store, sub, 'http://schema.org/name', row.lobidName)
33     addRdf(store, sub, 'http://schema.org/name', row.lobidName)
34     addRdf(store, sub, 'http://schema.org/name', row.lobidName)
35     const stat = dev(`${year}-ID-${year}`)
36     addRdf(store, sub, 'http://schema.org/statistic', stat)
37     addRdf(store, sub, 'http://schema.org/statistic', stat)
38     addRdf(store, sub, 'http://schema.org/statistic', stat)
39     addRdf(store, sub, 'http://schema.org/statistic', stat)
40     addRdf(store, sub, 'http://schema.org/statistic', stat)
41     addRdf(store, sub, 'http://schema.org/statistic', stat)
42     addRdf(store, sub, 'http://schema.org/statistic', stat)
43     addRdf(store, sub, 'http://schema.org/statistic', stat)
44     addRdf(store, sub, 'http://schema.org/statistic', stat)
45     addRdf(store, sub, 'http://schema.org/statistic', stat)
46     addRdf(store, sub, 'http://schema.org/statistic', stat)
47     addRdf(store, sub, 'http://schema.org/statistic', stat)
48     addRdf(store, sub, 'http://schema.org/statistic', stat)
49     addRdf(store, sub, 'http://schema.org/statistic', stat)
50     addRdf(store, sub, 'http://schema.org/statistic', stat)
51     addRdf(store, sub, 'http://schema.org/statistic', stat)
52     addRdf(store, sub, 'http://schema.org/statistic', stat)
53     addRdf(store, sub, 'http://schema.org/statistic', stat)
54     addRdf(store, sub, 'http://schema.org/statistic', stat)
55     addRdf(store, sub, 'http://schema.org/statistic', stat)
56     addRdf(store, sub, 'http://schema.org/statistic', stat)
57     addRdf(store, sub, 'http://schema.org/statistic', stat)
58     addRdf(store, sub, 'http://schema.org/statistic', stat)
59     addRdf(store, sub, 'http://schema.org/statistic', stat)
60     addRdf(store, sub, 'http://schema.org/statistic', stat)
61     addRdf(store, sub, 'http://schema.org/statistic', stat)
62     addRdf(store, sub, 'http://schema.org/statistic', stat)
63     addRdf(store, sub, 'http://schema.org/statistic', stat)
64     addRdf(store, sub, 'http://schema.org/statistic', stat)
65     addRdf(store, sub, 'http://schema.org/statistic', stat)
66     addRdf(store, sub, 'http://schema.org/statistic', stat)
67     addRdf(store, sub, 'http://schema.org/statistic', stat)
68     addRdf(store, sub, 'http://schema.org/statistic', stat)
69     addRdf(store, sub, 'http://schema.org/statistic', stat)
70     addRdf(store, sub, 'http://schema.org/statistic', stat)
71     addRdf(store, sub, 'http://schema.org/statistic', stat)
72     addRdf(store, sub, 'http://schema.org/statistic', stat)
73     addRdf(store, sub, 'http://schema.org/statistic', stat)
74     addRdf(store, sub, 'http://schema.org/statistic', stat)
75     addRdf(store, sub, 'http://schema.org/statistic', stat)
76     addRdf(store, sub, 'http://schema.org/statistic', stat)
77     addRdf(store, sub, 'http://schema.org/statistic', stat)
78     addRdf(store, sub, 'http://schema.org/statistic', stat)
79     addRdf(store, sub, 'http://schema.org/statistic', stat)
80     addRdf(store, sub, 'http://schema.org/statistic', stat)
81     addRdf(store, sub, 'http://schema.org/statistic', stat)
82     addRdf(store, sub, 'http://schema.org/statistic', stat)
83     addRdf(store, sub, 'http://schema.org/statistic', stat)
84     addRdf(store, sub, 'http://schema.org/statistic', stat)
85     addRdf(store, sub, 'http://schema.org/statistic', stat)
86     addRdf(store, sub, 'http://schema.org/statistic', stat)
87     addRdf(store, sub, 'http://schema.org/statistic', stat)
88     addRdf(store, sub, 'http://schema.org/statistic', stat)
89     addRdf(store, sub, 'http://schema.org/statistic', stat)
90     addRdf(store, sub, 'http://schema.org/statistic', stat)
91     addRdf(store, sub, 'http://schema.org/statistic', stat)
92     addRdf(store, sub, 'http://schema.org/statistic', stat)
93     addRdf(store, sub, 'http://schema.org/statistic', stat)
94     addRdf(store, sub, 'http://schema.org/statistic', stat)
95     addRdf(store, sub, 'http://schema.org/statistic', stat)
96     addRdf(store, sub, 'http://schema.org/statistic', stat)
97     addRdf(store, sub, 'http://schema.org/statistic', stat)
98     addRdf(store, sub, 'http://schema.org/statistic', stat)
99     addRdf(store, sub, 'http://schema.org/statistic', stat)
100    addRdf(store, sub, 'http://schema.org/statistic', stat)

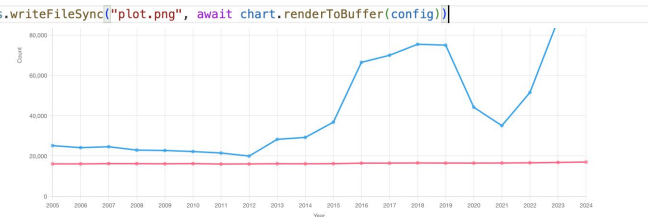
```

```

1 import { ChartJSNodeCanvas } from 'chartjs-node-canvas'
2 import { QueryEngine } from '@comunica/query-sparql-file'
3 import fs from 'fs'
4
5 const query = `
6   PREFIX dev: <https://open.bydata.de/api/hub/dev#>
7   PREFIX lobid: <https://lobid.org/organisations/DE->
8   SELECT ?year ?visitors ?population WHERE {
9     VALUES (?lib ?city) { (lobid:H0104 dev:city-09190141) }
10    ?lib dev:hasAnnualStatistic ?ls .
11    ?city dev:hasAnnualStatistic ?cs .
12    ?ls a dev:LibraryAnnualStatistic ;
13      dev:year ?year ;
14      dev:hasVisitCount ?visitors .
15    ?cs a dev:CityAnnualStatistic ;
16      dev:year ?year ;
17      dev:hasPopulation ?population .
18  } ORDER BY ?year`
19
20 const engine = new QueryEngine()
21 const bindingsStream = await engine.queryBindings(query, {
22   sources: [{ type: "file", value: "triples.ttl" }]
23 })
24
25 const data = { years: [], visitors: [], population: [] }
26 for (let row of await bindingsStream.toArray()) {
27   data.years.push(Number(row.get("year").value))
28   data.visitors.push(Number(row.get("visitors").value))
29   data.population.push(Number(row.get("population").value))
30 }
31
32 for (let i = 0; i < data.visitors.length - 1; i++) {
33   let thisYear = data.visitors[i]
34   let nextYear = data.visitors[i + 1]
35 }
36
37 const chart = new ChartJSNodeCanvas({ width: 1200, height: 700, backgroundColor: "white" })
38 const config = {
39   type: "line",
40   data: {
41     labels: data.years,
42     datasets: [
43       { label: "Library visitors", data: data.visitors },
44       { label: "City population", data: data.population }],
45     options: {
46       plugins: { title: { display: true, text: "Library visitors & city population" } },
47       scales: {
48         x: { title: { display: true, text: "Year" } },
49         y: { type: "linear", min: 0, title: { display: true, text: "Count" } }
50       }
51     }
52   }
53 }
54
55 fs.writeFileSync("plot.png", await chart.renderToBuffer(config))

```

... dafür



Use Case 2: Familienfreundliches München

Spielplätze mit öffentlichen Toiletten und Cafés in der Nähe.

Quellen:

- open.bydata: Öffentliche Spielplätze (Baureferat der LHM)
- open.bydata: WC-Standorte (GeodatenService München)
- OpenStreetMap: Cafés

Vorteile von mehr LOD im Portal

- Die Zeit von **“idea to insight”** kann drastisch verkürzt werden
 - senkt die initialen Aufwände für Datenaufbereitung und -verknüpfung
- Interoperabilität/“Verschneidbarkeit” aller Rohdaten erhöht sich
- Beim “letzten Stück”, dem Query, können LLMs gut helfen (MCP-Server)
- Hat das Potential die ganze Herangehensweise rund um *Experimente, Anwendungsfälle und investigative Fragestellungen* mit Hilfe von offenen Daten zu transformieren → schneller, explorativer, iterativer

Anschlussmöglichkeiten

- Katalysator für fachlichen Diskurs/Harmonisierung?
- Semantische Operationen sind deklarativ
 - Programmiersprache leicht austauschbar
 - “Hochkonzentrierte Wissensartefakte” beste Grundlage für KI
 - Verschiebt ausführbare Logik hin zur Fachexpertise
- Leichtgewichtige Web-Tools im Portal denkbar
 - semOps-Flows direkt im Portal anlegen, ausführen und teilen
 - Marktplatz der Arbeitsabläufe, “Ende-zu-Ende Showroom”

Diskussion

benjamin.degenhart@foerderfunke.org

Agenda

- 13:00-14:00 **Session 1: Präsentation & Demo**
 - Intro: Linked Data
 - Blick ins Portal
 - Use Case 1: Besucherzahlen Bibliotheken (Einstiegsbeispiel)
 - Use Case 2: Familienfreundliches München (Deep Dive)
 - Diskussion
- 14:30-15:30 **Session 2: Hands-On**
 - Setup: Starter-Kit + Triplestore
 - Selbst ans Werk 🛠️🧐
 - Austausch

Sie sind hier