

# First International TEXT2SPARQL Challenge

Co-Located with Text2KG at  
ESWC25

**Dr. Sebastian Tramp**

CTO @ eccenca GmbH



# Agenda

- 10:40 - **Sebastian Tramp**: Welcome and Introduction to the Challenge
- 10:50 - Jan Wardenga and **Tobias Käfer**: Presentation of the [Institute of Applied Informatics and Formal Description Methods \(AIFB\) @ KIT](#)
- 11:05 - Mehrzad Shahinmoghdam, **Tommaso Soru** and Sanju Tiwari: Presentation of the [DBpedia Group](#)
- 11:20 - Daniel Henselmann, **Rene Dorsch** and Andreas Harth: Presentation of the [Fraunhofer Institute for Integrated Circuits \(IIS\)](#)
- 11:35 - Daniel Gerber, Lorenz Bühmann, **Lars-Peter Meyer**, Felix Brei and Claus Stadler: Presentation of the [ETi @ Institute for Applied Informatics \(InfAI\)](#)
- 11:50 - **Oleg Somov**, Daniil Berezin and Roman Avdeev: Presentation of the [Artificial Intelligence Research Institute \(AIRI\), Moscow](#) - remote
- 12:05 - Aleksandr Perevalov and **Andreas Both**: Presentation of the [WSE Research Group @ Leipzig University of Applied Sciences](#)
- 12:20 - **Edgard Marx** and Sebastian Tramp: Result Presentation / Feedback Session and Winner Ceremony
- 12:45 - Closing

# Overview

- 27 Participants
- 14 Affiliations from 6 Countries
- 12 Endpoints registered
- 7 Presentations
- 3 Withdrawals (after result publication)
- 2 Question Datasets
- 250 Questions



# Question Datasets

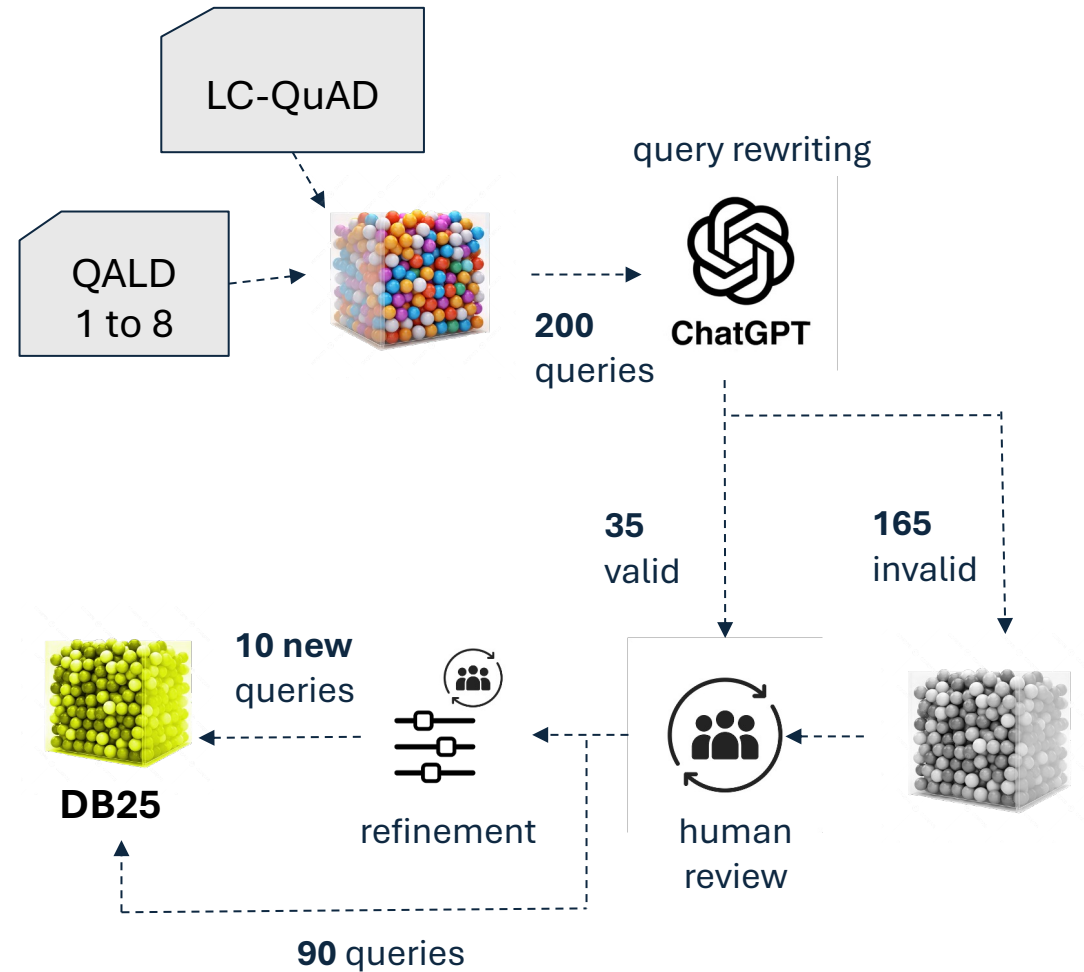
---

- DB25 – Dbpedia
  - 200 Questions
    - 100 in english
    - 100 in spanish
- CK25 – Corporate Knowledge
  - 50 Questions in english

```
- id: 15
question:
  en: What is the cheapest Encoder we can get from a french or german supplier?
features:
  - SELECT
  - ORDER
  - FILTER
  - LIMIT
classes:
  - :Product
  - :ProductCategory
  - :Supplier
  - :Price
properties:
  - :hasCategory
  - :price
  - :amount
  - :hasSupplier
  - :addressCountry
query:
  sparql: |
    PREFIX pv: <http://ld.company.org/prod-vocab/>
    SELECT DISTINCT ?result
    WHERE
    {
      ?result pv:hasCategory <http://ld.company.org/prod-instances/prod-cat-Encoder> .
      ?result pv:price ?priceR .
      ?priceR pv:amount ?price .
      ?result pv:hasSupplier ?supplier .
      ?supplier pv:addressCountry ?country .
      FILTER ( (?country = "France") || (?country = "Germany") )
    }
    ORDER BY ASC(?price)
    LIMIT 1
```

# DB25 DBpedia Dataset

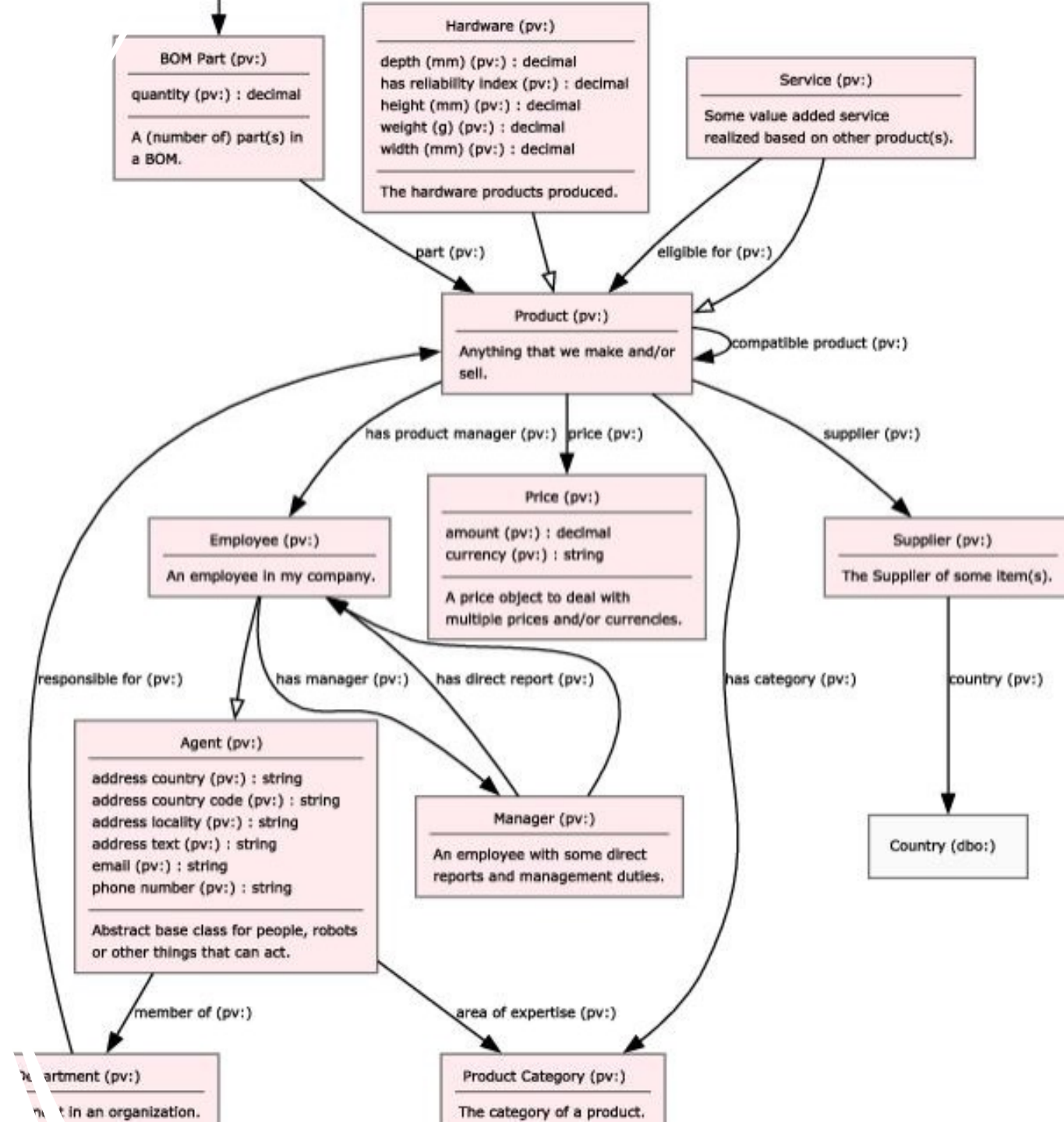
- We randomly generated 200 questions and queries using QALD-1 to 8 + LC-QuAD (en + es)
- Out of the 200, 35 returned proper results on the first try.
- The other 65 we had to fix from the remaining 165
  - We selected 100 and reviewed and corrected all of them.
  - We refined 10 questions with GROUP BY and ORDER BY to make the final benchmark balanced in terms of query features.





# CK25 Corporate Dataset

- Showcase Ontology from the Scratch
- Resource Generation with eccenca Corporate Memory
- Manual Creation of 50 Questions after Interviews with typical Stakeholders



# Procedure

- **Challengers had to register upfront**
- We requested to provide an HTTP(S) endpoint with a certain (easy to implement) interface
- We developed a command line client to ask questions on these endpoints
- In the 5 days evaluation phase we sent 250 requests to each endpoint
  - We recorded not only the result content but also the complete response object

The screenshot shows the GitHub repository page for 'AKSW / text2sparql.aksw.org'. The 'Issues' tab is selected, showing a list of 6 issues. The issues are related to endpoint registration and are all marked as 'Closed'. The issues are:

- Fetch queries from the endpoints (#66) - seebi opened 28 days ago
- analyse result files (#68) - seebi opened 28 days ago
- label:registration (search filter)

The list of issues is filtered by 'label:registration' and shows 9 issues. The issues are:

- Add https://text2sparql.cc-eti.org/text2sparqli from ETI, InfAI @LorenzBuehmann (registration) - AKSW/text2sparql.aksw.org#62 - by LorenzBuehmann was closed on Apr 28 - ✓ 1/1
- Add https://dbpedia-nspm3-primary.hf.space/codegen and https://dbpedia-nspm3-primary.hf.space/starcoder from DBpedia @mommi84 (registration) - AKSW/text2sparql.aksw.org#57 - by mommi84 was closed on Apr 22 - ✓ 1/1
- Add https://sacred-tightly-platypus.ngrok-free.app/ from LABIC @BruceNeves (registration) - AKSW/text2sparql.aksw.org#54 - by BruceNeves was closed on Apr 22 - ✓ 1/1
- Add https://graf.ti.rw.fau.de//text2sparql and https://graf.ti.rw.fau.de/q/text2sparql from Fraunhofer IIS @Quarkse (registration) - AKSW/text2sparql.aksw.org#53 - by Quarkse was closed on Apr 22
- Add https://text2sparql-avdeev-roman.amvera.io/ from MIPT @RomanAvdeev (registration) - AKSW/text2sparql.aksw.org#50 - by RomanAvdeev was closed on Apr 22
- Add http://demo1.franz.com:7000/t2s/api from Franz, Inc. @franzinc (registration) - AKSW/text2sparql.aksw.org#45 - by franzinc was closed on Apr 21
- Add http://167.172.162.197:8000 from AIFB @Branchenprimus (registration) - AKSW/text2sparql.aksw.org#44 - by Branchenprimus was closed on Apr 21
- Add https://desktop-47kug2k.tail6a5b76.ts.net/ from LACODAM @BaptisteAmice (registration) - AKSW/text2sparql.aksw.org#43 - by BaptisteAmice was closed on Apr 21
- Add https://wse-research.org/kgqagent/api from WSE Research group @Perevalov (registration) - AKSW/text2sparql.aksw.org#35 - by Perevalov was closed on Apr 15

# Procedure

- Challengers had to register upfront
- **We requested to provide an HTTP(S) endpoint with a certain (easy to implement) interface**
- We developed a command line client to ask questions on these endpoints
- In the 5 days evaluation phase we sent 250 requests to each endpoint
  - We recorded not only the result content but also the complete response object

Challenge

Search

GitHub  
☆ 4 🌟 10

The deployed service needs to provide a simple API which [specification](#). Basically you have to support two GET parameters, `dataset` and `question`. In addition to that, here is an example implementation using FastAPI:

Example

```
"""text2sparql-api"""

import fastapi

app = fastapi.FastAPI(
    title="TEXT2SPARQL API Example",
)

KNOWN_DATASETS = [
    "https://text2sparql.aksw.org/2025/dbpedia/",
    "https://text2sparql.aksw.org/2025/corporate/"
]

@app.get("/")
async def get_answer(question: str, dataset: str):
    if dataset not in KNOWN_DATASETS:
        raise fastapi.HTTPException(404, "Unknown dataset ...")
    return {
        "dataset": dataset,
        "question": question,
        "query": "... SPARQL here ..."
    }
```

Your registration is done, if we merge your data into our repository.

In case you want to **self-evaluate your endpoint** with the same client we are using for the evaluation, follow this recipe:

Self-Evaluation using the TEXT2SPARQL command line client

```
# Install the client (use your preferred way)
$ pipx install text2sparql-client

# prepare a questions file like this
$ cat questions.yaml
---
dataset:
  id: https://text2sparql.aksw.org/2025/corporate/
questions:
```

Table of contents

[Description](#)

[Knowledge Graphs for Evaluation](#)

[DBpedia \(Large Knowledge Graph\)](#)

[Corporate Knowledge \(Small Knowledge Graph\)](#)

[Benchmark Dataset](#)

[Training Set](#)

[Test Set](#)

[Evaluation](#)

[Process](#)

[Metrics](#)



# Procedure

- Challengers had to register upfront
- We requested to provide an HTTP(S) endpoint with a certain (easy to implement) interface
- **We developed a command line client to ask questions on these endpoints**
- In the 5 days evaluation phase we sent 250 requests to each endpoint
  - We recorded not only the result content but also the complete response object

```
seebi-testing ➤ text2sparql
Usage: text2sparql [OPTIONS] COMMAND [ARGS]...

TEXT2SPARQL Client

This command line tool can be used to retrieve answers from a TEXT2SPARQL
conform server.

For information on the TEXT2SPARQL challenge, have a look at:
https://text2sparql.aksw.org/

Options:
  --version      Show the version and exit.
  -d, --debug    Enable output of debug information.
  -h, --help     Show this message and exit.

Commands:
  ask           Query a TEXT2SPARQL endpoint
  evaluate      Evaluate the results from a TEXT2SPARQL endpoint.
  serve        Provide a TEXT2SPARQL testing endpoint

seebi-testing ➤ text2sparql ask --help
Usage: text2sparql ask [OPTIONS] QUESTIONS_FILE URL

Query a TEXT2SPARQL endpoint

Use a questions YAML file and send each question to a TEXT2SPARQL conform
endpoint. This command will create a sqlite database (--answers-db) saving
the responses.

Options:
  --answers-db FILE  Where to save the endpoint responses. [default:
                     responses.db]
  --timeout INTEGER  Timeout in seconds. [default: 600]
  -o, --output FILE  Save JSON output to this file. [default: -]
  --cache / --no-cache  If possible, return a cached response from the answers
                        database. [default: cache]
  -h, --help         Show this message and exit.

seebi-testing ➤
```

# Procedure

---

- Challengers had to register upfront
- We requested to provide an HTTP(S) endpoint with a certain (easy to implement) interface
- We developed a command line client to ask questions on these endpoints
- **In the 5 days evaluation phase we sent 250 requests to each endpoint**
  - We recorded not only the result content but also the complete response object

```
seebi-testing ➤ tree
├── AIFB
│   ├── ck25_answers.json
│   ├── ck25_responses.db
│   ├── db25_answers.json
│   └── db25_responses.db
├── DBPEDIA-CG
│   ├── ck25_answers.json
│   ├── ck25_responses.db
│   ├── db25_answers.json
│   └── db25_responses.db
├── DBPEDIA-CL
│   ├── ck25_answers.json
│   ├── ck25_responses.db
│   ├── db25_answers.json
│   └── db25_responses.db
├── DBPEDIA-SC
│   ├── ck25_answers.json
│   ├── ck25_responses.db
│   ├── db25_answers.json
│   └── db25_responses.db
├── FRANZ
│   ├── ck25_answers.json
│   ├── ck25_responses.db
│   ├── db25_answers.json
│   └── db25_responses.db
├── IIS-L
│   ├── ck25_answers.json
│   ├── ck25_responses.db
│   ├── db25_answers.json
│   └── db25_responses.db
├── IIS-Q
│   ├── ck25_answers.json
│   ├── ck25_responses.db
│   ├── db25_answers.json
│   └── db25_responses.db
├── INFAI
│   ├── ck25_answers.json
│   ├── ck25_responses.db
│   ├── db25_answers.json
│   └── db25_responses.db
└── LABIC
    ├── ck25_answers.json
    └── ck25_responses.db
```



# Agenda

- 10:40 - **Sebastian Tramp**: Welcome and Introduction to the Challenge
- 10:50 - Jan Wardenga and **Tobias Käfer**: Presentation of the [Institute of Applied Informatics and Formal Description Methods \(AIFB\) @ KIT](#)
- 11:05 - Mehrzad Shahinmoghadam, **Tommaso Soru** and Sanju Tiwari: Presentation of the [DBpedia Group](#)
- 11:20 - Daniel Henselmann, **Rene Dorsch** and Andreas Harth: Presentation of the [Fraunhofer Institute for Integrated Circuits \(IIS\)](#)
- 11:35 - Daniel Gerber, Lorenz Bühmann, **Lars-Peter Meyer**, Felix Brei and Claus Stadler: Presentation of the [ETi @ Institute for Applied Informatics \(InfAI\)](#)
- 11:50 - **Oleg Somov**, Daniil Berezin and Roman Avdeev: Presentation of the [Artificial Intelligence Research Institute \(AIRI\), Moscow](#) - remote
- 12:05 - Aleksandr Perevalov and **Andreas Both**: Presentation of the [WSE Research Group @ Leipzig University of Applied Sciences](#)
- 12:20 - **Edgard Marx** and Sebastian Tramp: Result Presentation / Feedback Session and Winner Ceremony
- 12:45 - Closing