

Bridging the gap between business domains and Knowledge Graphs

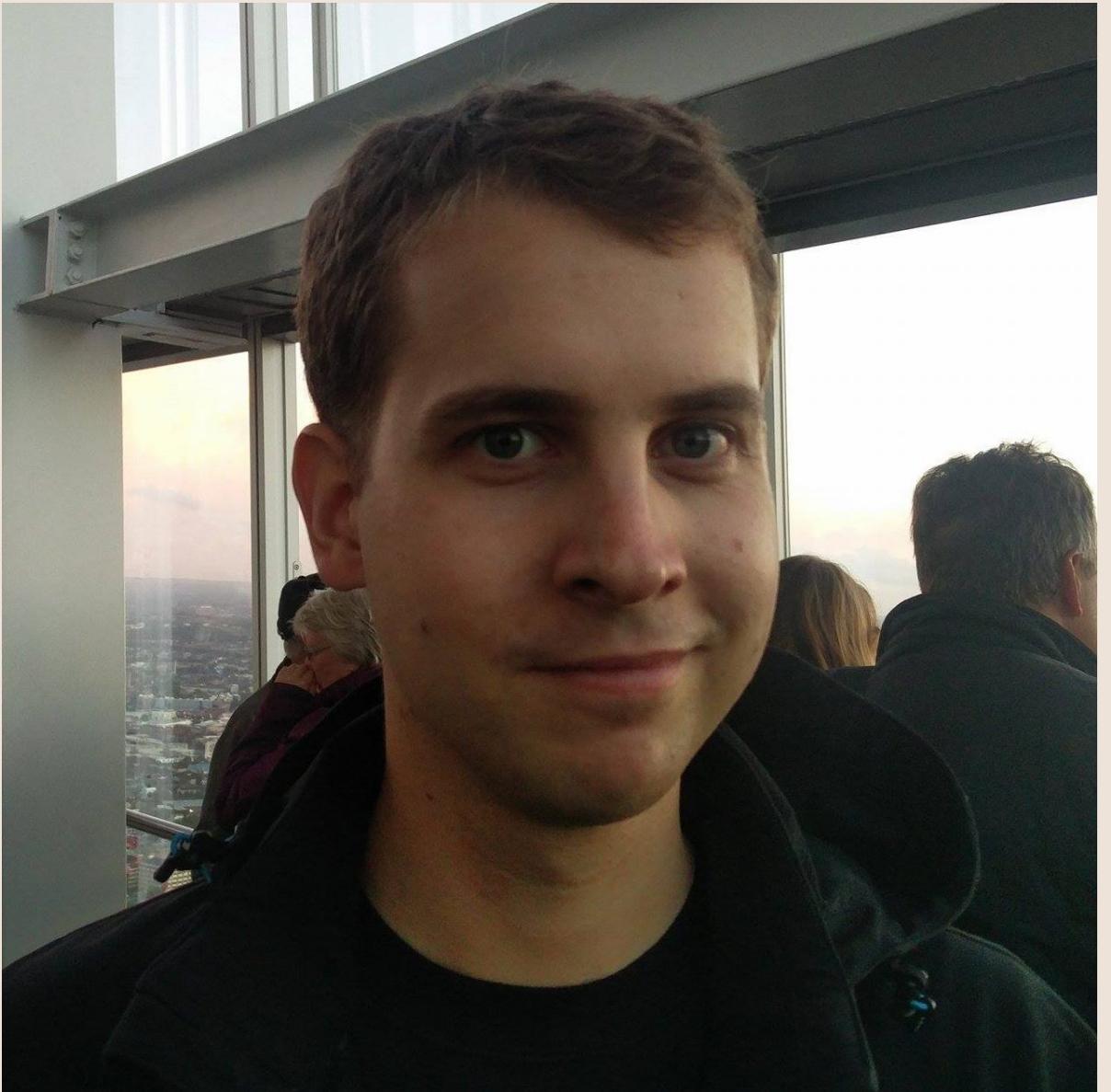
About the author

- Tomasz Pluskiewicz
- Zazuko GmbH
- Interests
 - Semantic Web
 - REST APIs
 - Hydra CG

 [/tpluscode](https://github.com/tpluscode)

 [@tpluscode](https://twitter.com/tpluscode)

 t-code.pl



Agenda

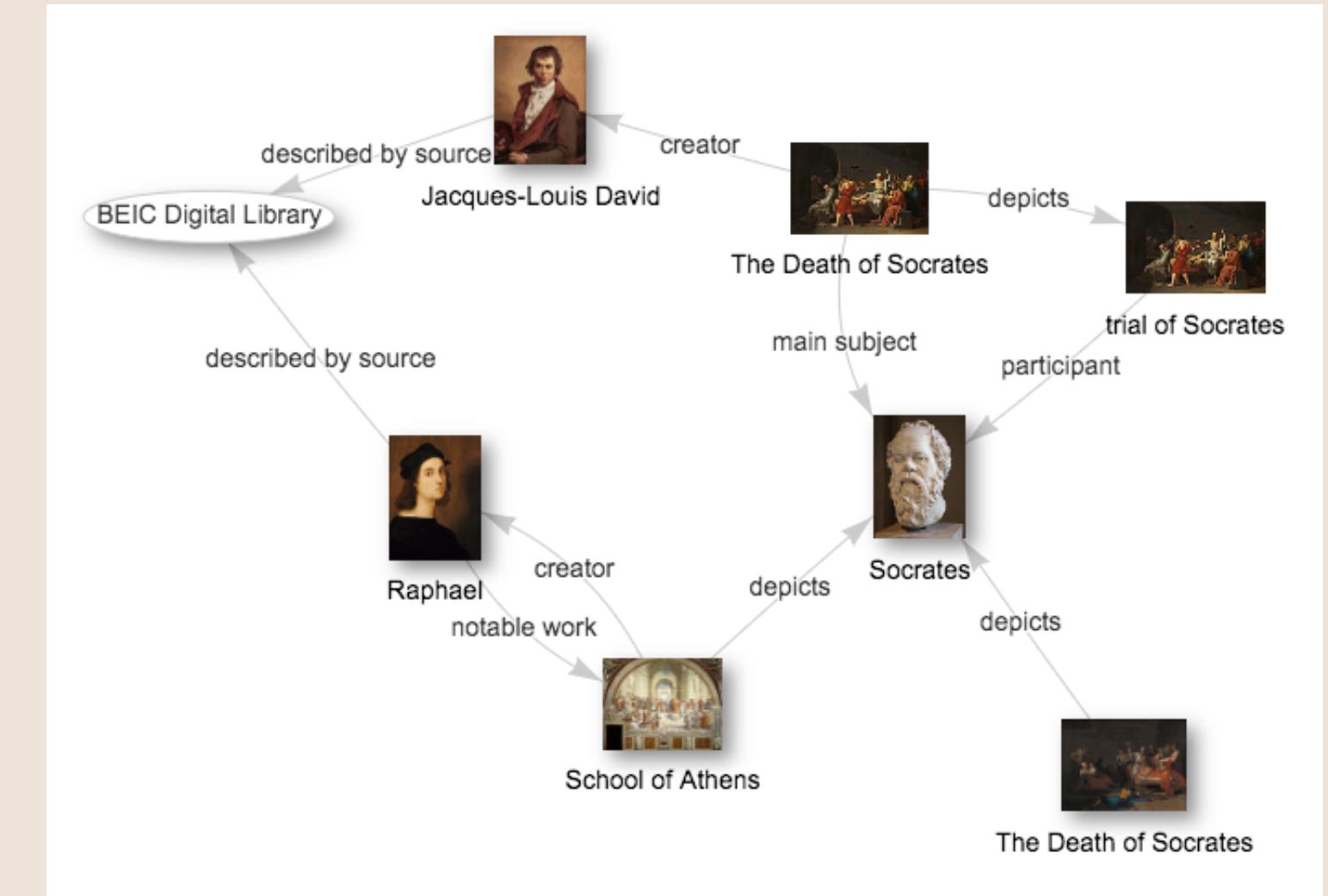
1. Intro (this)
2. Building an API backed by Knowledge Graph
 - Classic Example - Todo App
 - Step-by-step walkthrough
3. Showcase of basic UI (web app)
4. How to start from scratch
5. Closing remarks

Knowledge Graphs

What are they?

What is a Knowledge Graph?

- [KG] represents a network of real-world entities—i.e. objects, events, situations, or concepts—and illustrates the relationship between them^{IBM1}
- A digital structure that represents knowledge as concepts and the relationships between them (facts)^{WIKI1}
- A knowledge model: a collection of interlinked descriptions of concepts, entities, relationships and events^{ONTOTEXT1}



^{IBM1} [What is a knowledge graph?](#)

^{WIKI1} [Knowledge graph](#)

^{ONTOTEXT1} [What is a Knowledge Graph?](#)

Knowledge Graphs

Where to they come from?

Knowledge Graphs

Where do they come from?

- Knowledge Graph – A Powerful Data Science Technique to Mine Information from Text

"mine information from Wikipedia pages"

"make this text data readable for machines"

Knowledge Graphs

Where to they come from?

- Ontotext: How-to: Building Knowledge Graphs in 10 Steps

"Apply ETL tools to convert your data to RDF"

Knowledge Graphs

Where do they come from?

- How to Create a Knowledge Graph from Data?

"Large organizations generate lot of internal data"

"creating a knowledge graph by integrating the data available from structured sources"

Knowledge Graphs

Where to they come from?

- neo4j Tutorial: Build a Knowledge Graph using NLP and Ontologies

"entities extracted using APOC NLP procedures and Ontologies extracted using neosemantics"

*Data integration consumes
35%-65% of IT budget*

— Data-Centric Manifesto^{CENTRIC}

^{CENTRIC} <http://datacentricmanifesto.org/principles/>

Can we do different?

(better?)

Knowledge Graph

Not as an afterthought

Turtles all the way down

1. Almost everything is part of the knowledge graph
2. As little code as possible
3. Adopt standards

Team-friendly

1. Adopt and learn from best industry practices
2. Publish data as APIs
3. Compelling, end-to-end toolset

Project setup

hypermedia.app

Tools for building hypermedia-driven application using open web standards

- [!\[\]\(5338ce82ce7b6718f3db173c378c93da_img.jpg\) hypermedia-app](#)
 - [Creta](#)
Hydra APIs in node (express)
 - [Shaperone](#)
SHACL Form builder
 - [Roadshow](#)
SHACL UI builder
 - [Shell](#)
Framework for a complete web app
- Based on standards
 - [SPARQL](#)
 - [RDFS/SHACL](#)
 - [Web Access Control](#)
 - [Hydra](#)
 - [RFC 7807 - Problem Details for HTTP APIs](#)
 - [RFC 6570 - IRI Template](#)
 - [Web Components](#)
- All Open Source

Project setup

Runtime setup

<https://a.maze.link/kgc-tutorial-start>

- Node monorepo (yarn)
- Local docker setup (via Lando)
- Including local graph database (AllegroGraph)
 - requirements are SPARQL & reasoning

Showtime





imgflip.com

Recap

- Everything is a resource
- Graph Per Resource
- Explicit authorizations (API-only)
- Standards
- Inferencing
- Low-code but not no-code

Bonus

Starting from scratch is easy too!

 [hypermedia-app/creta-template-app](#)

Install pipx and then:

```
pipx install "copier==6.0.0b0"
copier -a .copier-creta.yaml \
    gh:hypermedia-app/creta-project-template.git \
    project-name
```

Update any time:

```
copier -a .copier-creta.yaml -s apps/api update
```

Roadmap

Creta:

- Knossos: Performance optimizations
- Knossos: SHACL AF 1.1
- Talos: watch mode
- Talos: migrations
- Talos: trig
- Talos: dynamic representations
- Talos: resource verification

Shaperone:

- More live examples
- Design types (look & feel)
- Best practices

Roadshow:

- Stabilize
- Best practices
- Documentation !

Call to action

