

# Angewandte Generative KI in den (digitalen) Geisteswissenschaften

Erzeugung und Modellierung von RDF-Daten

Workshop-Materialien: <http://chpollin.github.io/GM-DH>



Christopher Pollin

<https://chpollin.github.io>

Zentrum für Informationsmodellierung, Graz

<https://informationsmodellierung.uni-graz.at/de/institut>

Digital Humanities Craft OG

[www.dhcraft.org](http://www.dhcraft.org)

Workshop-Materialien: <http://chpollin.github.io/GM-DH>

**Donnerstag, 15.06., 10:00 - 11:00 Uhr:** Erzeugung und Modellierung von RDF-Daten

Kenntnisse in RDF, RDFS und Python werden empfohlen.

**Donnerstag, 22.06., 10:00 - 11:00 Uhr:** Digitale Edition und TEI: unstrukturierter Text nach TEI XML

Kenntnisse in XML und TEI werden empfohlen.

**Donnerstag, 29.06., 10:00 - 11:00 Uhr:** Datenvisualisierung und Data Stories: Analyse biographischer Daten von ins Exil geflüchteten Übersetzer\*innen mit Python und Jupyter Notebook

Kenntnisse in Python werden empfohlen.

<https://informationsmodellierung.uni-graz.at/de/neuigkeiten/detail/article/workshopreihe-gpt-und-prompt-engineering-fuer-die-digitalen-geisteswissenschaften/>

# Lernziele

- Transformation von CSV nach RDF mit GPT-4
- GPT-gestützte Erstellung eines RDF-Datenmodells
- “Multi Output Generation”:
  - RDFS
  - GPT-unterstützte Erstellung eines Transformationsszenarios von CSV nach RDF mit Python und rdflib.
- [GPT-unterstütztes Ontology Engineering. Am Beispiel von Transaktionen in historischen Rechnungsunterlagen.]

## Materialien

- Google Drive mit Slides, CSV und Jupyter Notebook.  
<https://drive.google.com/drive/folders/16-jSKdnmemqweolhIDpTgavfUoRr-mZU?usp=sharing>
- RDF Conversion Assistance. GPT-4, ChatGPT May 12 Version. 15.05.2023.  
<https://chat.openai.com/share/a3319750-3392-4e7a-812a-d6494bef03ac>.

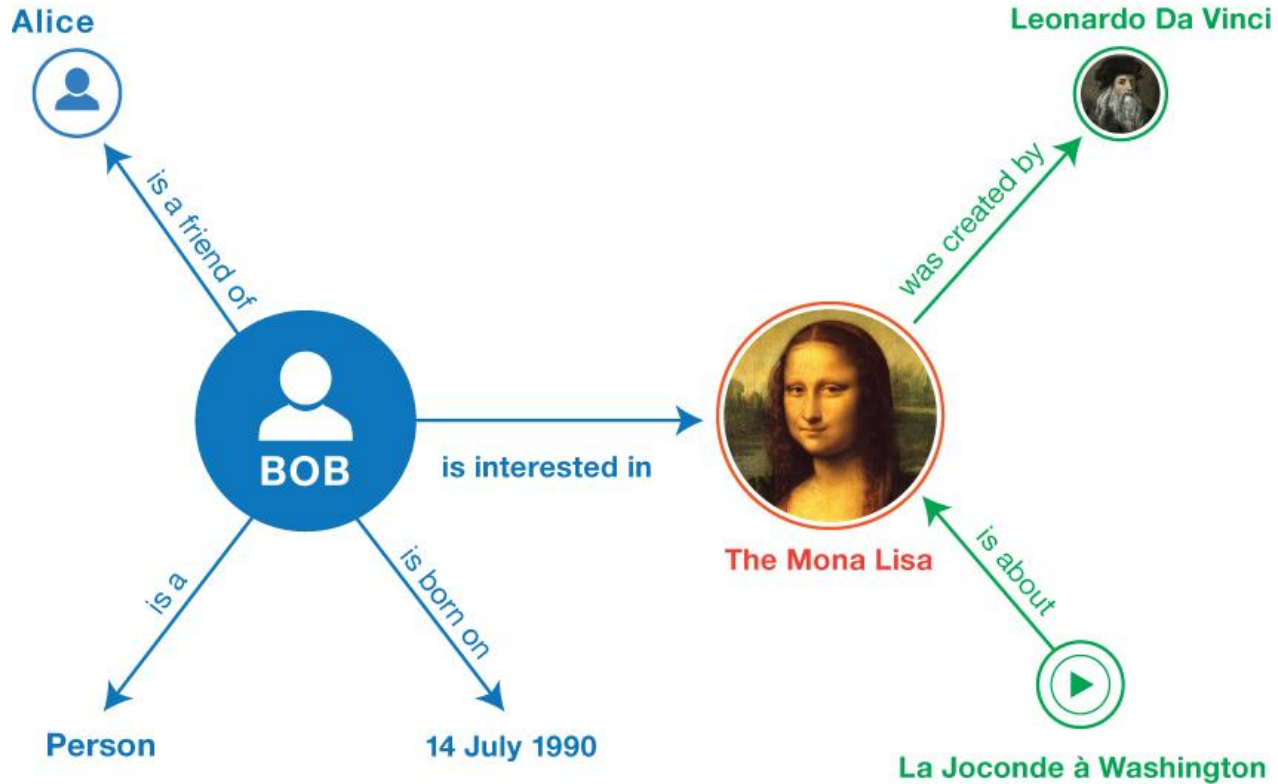
## YouTube und weitere Tutorials

- AI Explained, <https://www.youtube.com/@ai-explained->
- Matthew Berman, [https://www.youtube.com/@matthew\\_berman](https://www.youtube.com/@matthew_berman)
- Yannic Kilcher, <https://www.youtube.com/@YannicKilcher>
- ChatGPT Prompt Engineering for Developers, <https://learn.deeplearning.ai/chatgpt-prompt-eng>
- Hötter, Johannes and Warmuth, Christian. OpenHPI. ChatGPT: Was bedeutet generative KI für unsere Gesellschaft?  
<https://open.hpi.de/courses/kizukunft2023/items/3jMw1GWPssDCNQw63XXsn>

## Paper

- Baktash, Jawid Ahmad, and Mursal Dawodi. 2023. 'Gpt-4: A Review on Advancements and Opportunities in Natural Language Processing'. arXiv. <https://doi.org/10.48550/arXiv.2305.03195>.
- Møller, Anders Giovanni, Jacob Aarup Dalsgaard, Arianna Pera, and Luca Maria Aiello. 2023. 'Is a Prompt and a Few Samples All You Need? Using GPT-4 for Data Augmentation in Low-Resource Classification Tasks'. arXiv.  
<https://doi.org/10.48550/arXiv.2304.13861>.
- Mukherjee, Subhabrata, Arindam Mitra, Ganesh Jawahar, Sahaj Agarwal, Hamid Palangi, and Ahmed Awadallah. 2023. 'Orca: Progressive Learning from Complex Explanation Traces of GPT-4'. arXiv. <https://doi.org/10.48550/arXiv.2306.02707>.
- Roberts, Jonathan, Timo Lüddecke, Sowmen Das, Kai Han, and Samuel Albanie. 2023. 'GPT4GEO: How a Language Model Sees the World's Geography'. arXiv. <http://arxiv.org/abs/2306.00020>.
- Watkins, Ryan. 'Guidance for Researchers and Peer-Reviewers on the Ethical Use of Large Language Models (LLM) in Scientific Research Workflows'. OSF Preprints, 21 April 2023. <https://doi.org/10.31219/osf.io/6uh8p>.
- Yao, Shunyu, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. 'Tree of Thoughts: Deliberate Problem Solving with Large Language Models'. arXiv, 17 May 2023. <https://arxiv.org/abs/2305.10601>.

# Recap



# **GPT-unterstütztes Erstellen eines RDF-Datenmodells**

Am Beispiel eines Buches in der Bibliothek  
von Stefan Zweig

# Beispieldaten: ein Buch in der Bibliothek von Stefan Zweig

## Informationen zum Buch

- ID: SZDBIB.365
- Verfasser/in: Abraham, Pierre
- Titel: Proust : recherches sur la création intellectuelle
- Veröffentlichung: Paris : Rieder, 1930
- Reihe: Maîtres des littératures 7
- Sprache: Französisch
- Umfang/Einband: 91, LX Seiten : illustriert. 8°, Broschur

## Provenienzmerkmale

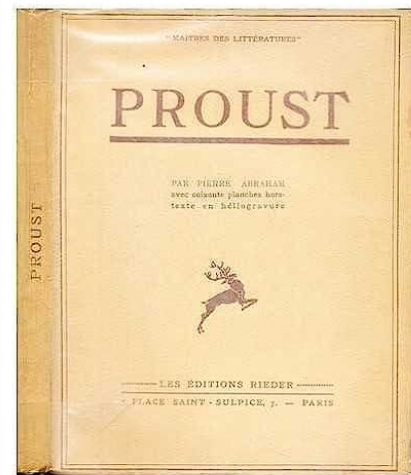
- Widmung: Pour Stefan Zweig, | en sympathie et en admiration | Pierre Abraham.
- Marginalie: Anmerkungen von Stefan Zweig in Bleistift
- Merkzeichen: Unterstreichungen in Bleistift
- Originalsignatur: B 3
- Heutiger Standort: Erben Stefan Zweigs, London A.I.01

## Abraham, Pierre

▼ Proust : recherches sur la création intellectuelle | 1930



Verfasser/in	Abraham, Pierre 🔍
Titel	<i>Proust : recherches sur la création intellectuelle</i>
Veröffentlichung	Paris : Rieder, 1930
Reihe	Maîtres des littératures 7
Sprache	Französisch
Umfang/Einband	91, LX Seiten : illustriert. 8°, Broschur
Widmung ⓘ	Pour Stefan Zweig,   en sympathie et en admiration   Pierre Abraham.
Marginalie ⓘ	Anmerkungen von Stefan Zweig in Bleistift
Merkzeichen ⓘ	Unterstreichungen in Bleistift
Originalsignatur ⓘ	B 3
Heutiger Standort	Erben Stefan Zweigs, London 🔍 A.I.01



Ein “FRBR Item” des  
Buches auf Amazon.

<https://www.amazon.com/Proust-recherches-cr%C3%A9ation-intellectuelle-Abraham/dp/B07173ZXMf>



# Iteration 1 - “I want to transform to RDF”



The following data is about a book in the library of Stefan Zweig:

...

ID: SZDBIB.365

Verfasser/in: Abraham, Pierre

Titel: Proust : recherches sur la création intellectuelle

Veröffentlichung: Paris : Rieder, 1930

Reihe: Maîtres des littératures 7

Sprache: Französisch

Umfang/Einband: 91, LX Seiten : illustriert. 8°, Broschur

Widmung: Pour Stefan Zweig, | en sympathie et en admiration | Pierre Abraham.

Marginalie: Anmerkungen von Stefan Zweig in Bleistift

Merkzeichen: Unterstreichungen in Bleistift

Originalsignatur: B 3

Heutiger Standort: Erben Stefan Zweigs, London A.I.01

...

I want to transform it to RDF. Ask me as many questions as you need to understand the domain.

**Anweisungen und  
Daten trennen.**

**Sich von GPT Fragen  
stellen lassen, um mehr  
Kontextinformation zu  
generieren.**

# Exkurs: Prompt Engineering 1

... Prompt Engineering ist Kontext, Kontext, Kontext!

## 1.2 Importance of prompt engineering in maximizing the effectiveness of ChatGPT

Prompt engineering is the art of crafting effective prompts that guide ChatGPT to generate desired responses. It plays a crucial role in maximizing the effectiveness of the language model by bridging the gap between user intent and model understanding. A well-engineered prompt can significantly improve the quality and relevance of ChatGPT's output, while a poorly crafted one may lead to unsatisfactory or erroneous responses.

For example, consider the following prompts and their corresponding ChatGPT responses:

**Prompt 1:** "Tell me about AI."

**Response 1:** "Artificial intelligence (AI) is a branch of computer science that deals with the creation and development of machines that can perform tasks requiring human-like intelligence. This includes learning, reasoning, problem-solving, and understanding natural language."

**Prompt 2:** "Provide a concise summary of artificial intelligence, its subfields, and current applications."

**Response 2:** "Artificial intelligence (AI) is the study of creating intelligent agents capable of performing tasks autonomously. Key subfields include machine learning, natural language processing, and robotics. Current applications range from virtual assistants and recommendation systems to autonomous vehicles and medical diagnosis."

The second prompt yields a more informative and focused response due to its clarity and specificity.

# Exkurs: Prompt Engineering 2

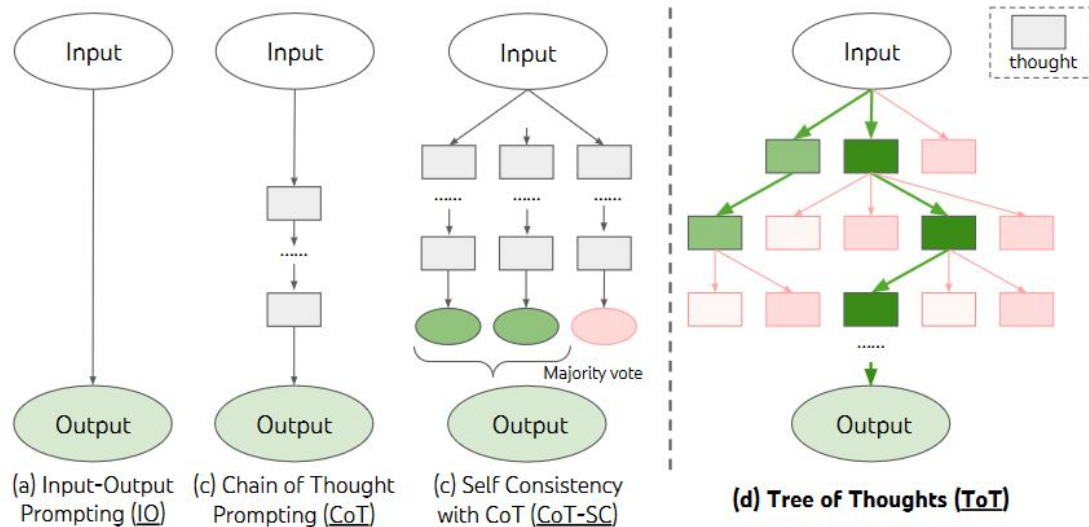


Figure 1: Schematic illustrating various approaches to problem solving with LLMs. Each rectangle box represents a *thought*, which is a coherent language sequence that serves as an intermediate step toward problem solving. See concrete examples of how thoughts are generated, evaluated, and searched in Figures 246.

Yao, Shunyu, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. 'Tree of Thoughts: Deliberate Problem Solving with Large Language Models'. arXiv, 17 May 2023.

<https://doi.org/10.48550/arXiv.2305.10601>.



To help you convert the information about the book in Stefan Zweig's library into RDF (Resource Description Framework) format, I would like to ask you the following questions:

1. What are the most important entities in this data that you would like to represent as RDF resources? For example: the book, author, publisher, etc.
2. What specific relationships or properties do you want to express between these entities? For example: authorship, publication date, dedication, etc.
3. Are there any existing RDF vocabularies or ontologies that you would like to use to represent the information? For example: Dublin Core, FOAF, Schema.org, etc.
4. Do you have any specific namespace URIs or prefixes you'd like to use for the RDF resources and properties?



1. all aspects are important. it should be very detailed.
2. a book is written by an author. a book has information about its publication. a book has specific information that marks it as a book of Zeig's library. an author has a first name and surname and identifiers. a book has information about its physical characteristics.
3. Dublin Core, FOAF, Schema.org and a domain specific namespace for Stefan Zweig Digital.

Now write the RDF source in Turtle serialisation.


**Man kann nun in der Konversation fortfahren, oder die Fragen und Antworten in einem neuen Prompt aufnehmen.**

**Strukturen von GPT zur Adressierung nutzen (1., 2., 3. etc.).**

**Output klar definieren.**



# GPT-4 Output



Based on your requirements, here is the RDF data in Turtle serialization:

```
turtle Copy code
```

```
@prefix dc: <http://purl.org/dc/elements/1.1/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix schema: <http://schema.org/> .
@prefix szd: <http://www.example.com/stefan_zweig_digital/> .

szd:SZDBIB.365
  a schema:Book ;
  dc:creator szd:AbrahamPierre ;
  dc:title "Proust : recherches sur la création intellectuelle" ;
  dc:date "1930" ;
  dc:publisher "Paris : Rieder" ;
  dc:format "91, LX Seiten : illustriert. 8°, Broschur" ;
  dc:language "Französisch" ;
  szd:dedication "Pour Stefan Zweig, | en sympathie et en admiration | Pie
  szd:marginalia "Anmerkungen von Stefan Zweig in Bleistift" ;
```

Informationen zum Buch

- ID: SZDBIB.365
- Verfasser/in: Abraham, Pierre
- Titel: Proust : recherches sur la création intellectuelle
- Veröffentlichung: Paris : Rieder, 1930
- Reihe: Maîtres des littératures 7
- Sprache: Französisch
- Umfang/Einband: 91, LX Seiten : illustriert. 8°, Broschur

Provenienzmerkmale

- Widmung: Pour Stefan Zweig, | en sympathie et en admiration | Pierre Abraham.
- Marginalie: Anmerkungen von Stefan Zweig in Bleistift
- Merkzeichen: Unterstreichungen in Bleistift
- Originalsignatur: B 3
- Heutiger Standort: Erben Stefan Zweigs, London A.I.01

http://www.example.com/stefan_zweig_digital/AbrahamPierre	
rdf:type	foaf:Person
foaf:givenName	Pierre
foaf:familyName	Abraham
foaf:isPrimaryTopicOf	http://www.example.com/stefan_zweig_digital/SZDBIB.365

http://www.example.com/stefan_zweig_digital/SZDBIB.365	
rdf:type	schema:Book
dc11:creator	http://www.example.com/stefan_zweig_digital/AbrahamPierre
dc11:title	Proust : recherches sur la création intellectuelle
dc11:date	1930
dc11:publisher	Paris : Rieder
dc11:format	91, LX Seiten : illustriert. 8°, Broschur
dc11:language	Französisch
http://www.example.com/stefan_zweig_digital/dedication	Pour Stefan Zweig,   en sympathie et en admiration   Pierre Abraham.
http://www.example.com/stefan_zweig_digital/marginalia	Anmerkungen von Stefan Zweig in Bleistift
http://www.example.com/stefan_zweig_digital/markings	Unterstreichungen in Bleistift
http://www.example.com/stefan_zweig_digital/originalSignature	B 3
http://www.example.com/stefan_zweig_digital/currentLocation	Erben Stefan Zweigs, London A.I.01
http://www.example.com/stefan_zweig_digital/series	Maîtres des littératures 7

```
1 @prefix dc: <http://purl.org/dc/elements/1.1/> .
2 @prefix foaf: <http://xmlns.com/foaf/0.1/> .
3 @prefix schema: <http://schema.org/> .
4 @prefix szd: <http://www.example.com/stefan_zweig_digital/> .
```

```
6 ▼ szd:SZDBIB.365
```

Die "Basics" mit Dublin Core

```
7   a schema:Book ;
8   dc:creator szd:AbrahamPierre ;
9   dc:title "Proust : recherches sur la création intellectuelle" ;
10  dc:date "1930" ;
11  dc:publisher "Paris : Rieder" ;
12  dc:format "91, LX Seiten : illustriert. 8°, Broschur" ;
13  dc:language "Französisch" ;
14  szd:dedication "Pour Stefan Zweig, | en sympathie et en admiration | Pierre Abraham." ;
15  szd:marginalia "Anmerkungen von Stefan Zweig in Bleistift" ;
16  szd:markings "Unterstreichungen in Bleistift" ;
17  szd:originalSignature "B 3" ;
18  szd:currentLocation "Erben Stefan Zweigs, London A.I.01" ;
19  szd:series "Maîtres des littératures 7" .
```

Domänenspezifisches mit dem szd-  
Namespace

```
21 ▼ szd:AbrahamPierre
```

```
22   a foaf:Person ;
23   foaf:givenName "Pierre" ;
24   foaf:familyName "Abraham" ;
25   foaf:isPrimaryTopicOf szd:SZDBIB.365 .
```

Personen Daten mit foaf beschrieben

## Iteration 2: “How can the RDF turtle be improved?”

http://www.example.com/stefan_zweig_digital/AbrahamPierre	
rdf:type	schema:Person
schema:givenName	Pierre
schema:familyName	Abraham
schema:creator	http://www.example.com/stefan_zweig_digital/SZDBIB.365

http://www.example.com/stefan_zweig_digital/Rieder	
rdf:type	schema:Organization
schema:name	Paris : Rieder

http://www.example.com/stefan_zweig_digital/Maitres_des_litteratures_7	
rdf:type	schema:BookSeries
schema:name	Maitres des littératures 7
schema:hasPart	http://www.example.com/stefan_zweig_digital/SZDBIB.365

http://www.example.com/stefan_zweig_digital/SZDBIB.365	
rdf:type	schema:Book
schema:author	http://www.example.com/stefan_zweig_digital/AbrahamPierre
schema:name	Proust : recherches sur la création intellectuelle
schema:datePublished	1930
schema:publisher	http://www.example.com/stefan_zweig_digital/Rieder
schema:numberOfPages	91, LX
schema:bookFormat	illustriert. 8°, Broschur
schema:inLanguage	Französisch
http://www.example.com/stefan_zweig_digital/dedication	Pour Stefan Zweig,   en sympathie et en admiration   Pierre Abraham.
http://www.example.com/stefan_zweig_digital/marginalia	Anmerkungen von Stefan Zweig in Bleistift
http://www.example.com/stefan_zweig_digital/markings	Unterstreichungen in Bleistift
http://www.example.com/stefan_zweig_digital/originalSignature	B 3
http://www.example.com/stefan_zweig_digital/currentLocation	Erben Stefan Zweigs, London A.I.01
dcterms:isPartOf	http://www.example.com/stefan_zweig_digital/Maitres_des_litteratures_7



szd:SZDBIB.365

```
a schema:Book ;
schema:author szd:AbrahamPierre ;
schema:name "Proust : recherches sur la création intellectuelle" ;
schema:datePublished "1930" ;
schema:publisher szd:Rieder ;
schema:numberOfPages "91, LX" ;
schema:bookFormat "illustriert. 8°, Broschur" ;
schema:inLanguage "Französisch" ;
szd:dedication "Pour Stefan Zweig, | en sympathie et en admiration | Pierre Abraham." ;
szd:marginalia "Anmerkungen von Stefan Zweig in Bleistift" ;
szd:markings "Unterstreichungen in Bleistift" ;
szd:originalSignature "B 3" ;
szd:currentLocation "Erben Stefan Zweigs, London A.I.01" ;
dcterms:isPartOf szd:Maîtres_des_littératures_7 .
```

Iteration 2

- *object property* für `schema:publisher` und URI für Publisher.
- Reihe als *object property* `dcTerms:isPartOf`.
- Einheitlicher Wechsel von “dc:” auf “schema:”.
- Scope von *schema.org* ist spezifischer als Dublin Core.
- Scope von *schema.org* ist immer, semantische Inhalte einer Webseite auszudrücken, wird aber in vielen LOD-Datensätzen verwendet.

szd:SZDBIB.365

```
a schema:Book ;
dc:creator szd:AbrahamPierre ;
dc:title "Proust : recherches sur la création intellectuelle" ;
dc:date "1930" ;
dc:publisher "Paris : Rieder" ;
dc:format "91, LX Seiten : illustriert. 8°, Broschur" ;
dc:language "Französisch" ;
szd:dedication "Pour Stefan Zweig, | en sympathie et en admiration | Pierre Abraham." ;
szd:marginalia "Anmerkungen von Stefan Zweig in Bleistift" ;
szd:markings "Unterstreichungen in Bleistift" ;
szd:originalSignature "B 3" ;
szd:currentLocation "Erben Stefan Zweigs, London A.I.01" ;
szd:series "Maîtres des littératures 7" .
```

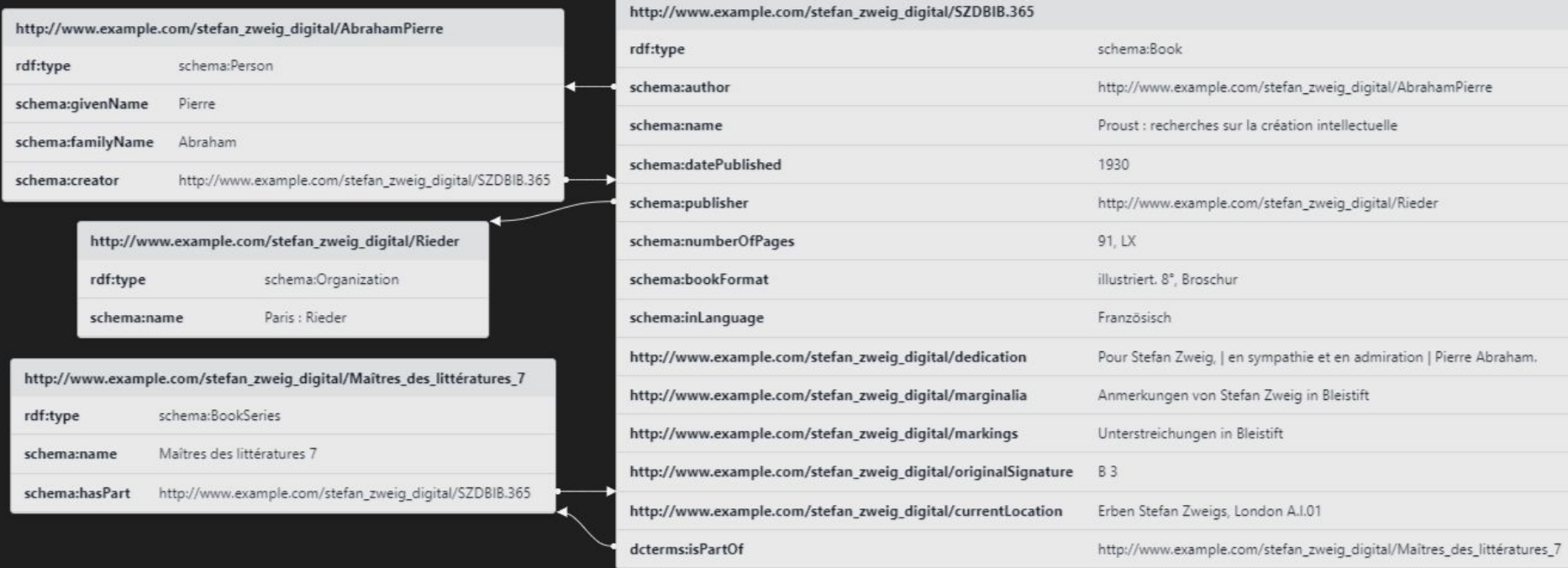
Iteration 1

```

23  szd:AbrahamPierre
24      a schema:Person ;
25      schema:givenName "Pierre" ;
26      schema:familyName "Abraham" ;
27      schema:creator szd:SZDBIB.365 .
28
29  szd:Rieder
30      a schema:Organization ;
31      schema:name "Paris : Rieder" .
32
33  szd:Maîtres_des_littératures_7
34      a schema:BookSeries ;
35      schema:name "Maîtres des littératures 7" ;
36      schema:hasPart szd:SZDBIB.365 .

```

- Named Entities bekommen eigene URI.
- schema:creator  
szd:SZDBIB-365 und  
dc:creator  
szd:AbrahamPierre ist redundant und könnte auch spezifischer sein mit schema:author.



- `dcterms:isPartOf` und `schema:hasPart`
  - Domain und Range in beide Richtungen erzeugen Redundanz.
  - Einheitlichere Vorgehensweise: `dcterms` oder `schema` (laut GPT-4 sind sie nicht äquivalent, <https://chat.openai.com/share/5d25fc44-94dd-4819-9bea-a0e20fe0b8de>)

## Iteration 3: “List all possible improvements you know of for the following RDF in Turtle serialisation”. GPT4 Antwort:

Here is a list of possible improvements for the provided RDF in Turtle serialization:

1. Use more specific properties for the book format and language:
  - Replace ``schema:bookFormat`` with ``schema:bookFormatType`` and use a more specific term, such as `"Paperback"` or `"Hardcover"` instead of the full description.
  - Replace ``schema:inLanguage`` with a language tag, such as ``schema:inLanguage "fr"`` for French.
2. Use URIs for languages, book formats, and other controlled vocabulary terms:
  - Use the Library of Congress language URI for French:  
``<http://id.loc.gov/vocabulary/iso639-1/fr>``
  - Use a controlled vocabulary for book formats, such as the RDA Content Type:  
``<http://rdaregistry.info/termList/RDAContentType/1020>`` for `"text"`
3. Use more specific properties for the book's dedication, marginalia, markings, original signature, and current location:
  - Create a separate resource for the book's physical copy and use more specific properties from a relevant ontology, such as the FRBRoo ontology, to describe the copy's unique features.

## Iteration 3: “List all possible improvements you know of for the following RDF in Turtle serialisation”. Fortsetzung der GPT-4-Antwort:

4. Add missing information, such as the **book's ISBN**, the **author's birth and death dates**, and the **publisher's location**.
5. Use more descriptive URIs for resources and properties:
  - Use a **base URI** for the szd namespace that better reflects the Stefan Zweig Digital domain, such as ``http://stefanzweig.digital/ontology/``
  - **Create URIs for the author, publisher, and book series that are more human-readable** and include their names or abbreviations, such as ``szd:Pierre_Abraham``, ``szd:Rieder_Paris``, and ``szd:Maîtres_des_littératures``.
6. Add more metadata about the RDF graph itself, such as its creation date, creator, and licensing information, by using the ``dcterm`` and ``void`` (**Vocabulary of Interlinked Datasets**) namespaces.
7. Use ``rdfs:label`` or ``skos:prefLabel`` to provide human-readable labels for **resources and properties in the szd namespace**.
8. Use the ``rdfs:comment`` property to provide **explanatory notes or descriptions for resources and properties in the szd namespace**.



write a prompt including the following text:

Here is a list of possible improvements for the provided RDF in Turtle serialization:

Use more specific properties for the book format and language:

Replace `schema:bookFormat` with `schema:bookFormatType` and use a more specific term, such as "Paperback" or "Hardcover" instead of the full description.

Wir wollen sicherstellen, dass all diese Dinge berücksichtigt werden. Deswegen lassen wir uns von GPT-4 den Prompt schreiben und fügen die RDF-Daten zu den "possible improvements" aus Iteration 3 hinzu.

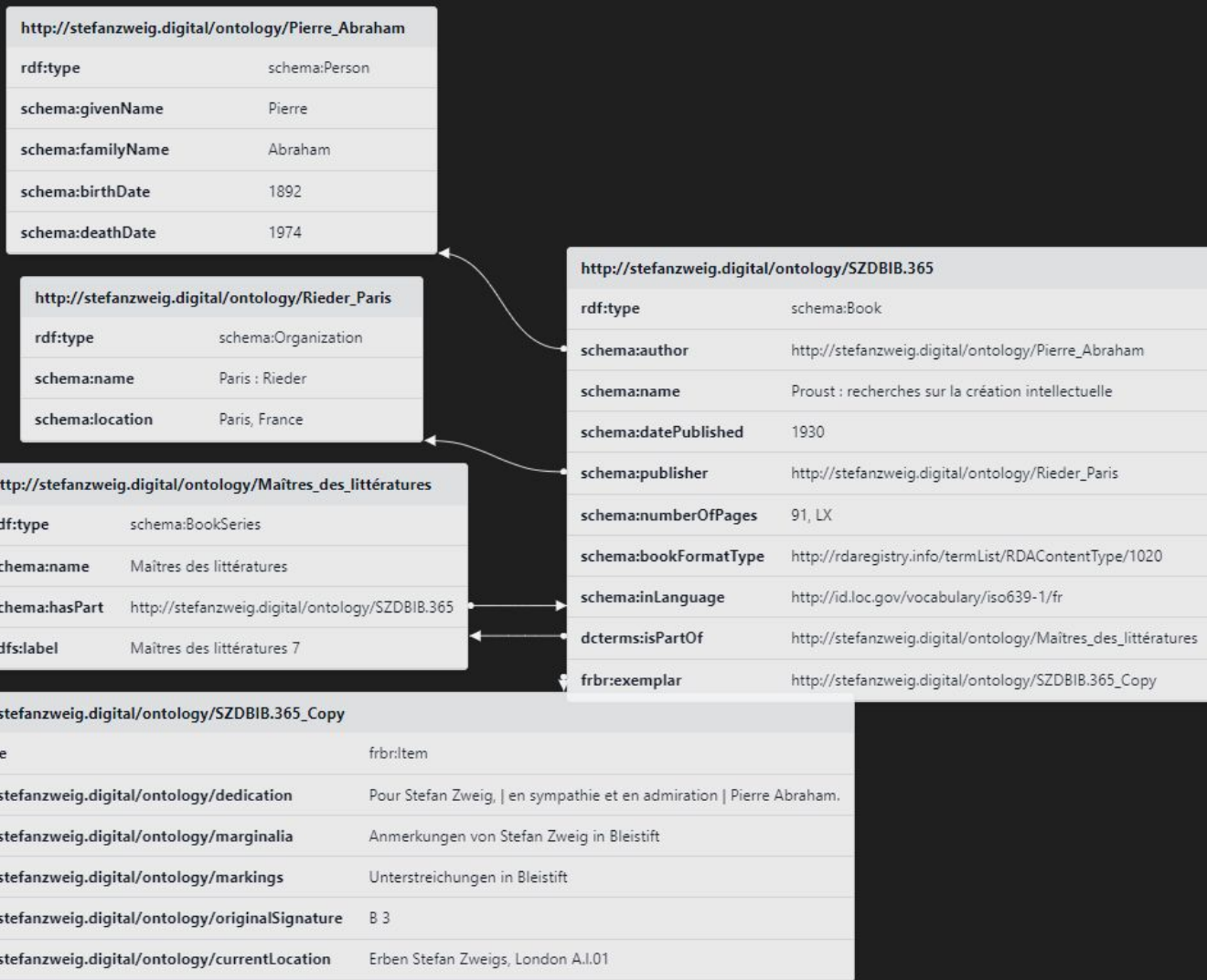
# Iteration 3

Hat selbstständig  
(korrekte!) Geburts- und  
Sterbedaten hinzugefügt.

“Paris : Rieder” wurde  
selbstständig “Paris,  
Frankreich” zugeordnet.

Trennung von frbr:Item  
(das physische Buch mit  
den Provenienz-  
merkmalen) und  
schema:Book.

Sprache mit URI auf  
kontrolliertes Vokabular  
normalisiert.



# Iteration 4

Your RDF in Turtle serialization is already quite good, but there are still several improvements you could make to enhance the data quality, readability, and interoperability.

1. **\*\*Add ISBN Information\*\***: If you have the book's ISBN, it's a good practice to include it as a unique identifier.
2. **\*\*Use Standard Properties for Book Annotation\*\***: Instead of using custom properties like ``szd:marginalia``, ``szd:markings``, and ``szd:dedication``, you could adopt properties from established ontologies. For instance, the Dublin Core terms (`dcterms:`) vocabulary provides ``dcterms:description`` and ``dcterms:abstract`` that could serve this purpose.
3. **\*\*Add Licensing Information\*\***: If the data are under a specific license, use ``dcterms:license`` to link to the URL of the license.
4. **\*\*Metadata About the Graph\*\***: Use ``dcterms:created`` and ``dcterms:creator`` to add information about the creation of the RDF data itself.



# Iteration 4

5. **\*\*Improve Property Use for ``frbr:Item``\*\***: It might be better to use more standardized properties for ``szd:dedication``, ``szd:marginalia``, ``szd:markings``, ``szd:originalSignature``, and ``szd:currentLocation``. **Consider adopting a more specific ontology for bibliographic items, such as Bibframe (<https://www.loc.gov/bibframe/>), which includes properties like ``bf:annotation``, ``bf:note``, and ``bf:heldBy``.**

→ kann man sich anschauen

6. **\*\*Provide human-readable labels and comments\*\***: Use ``rdfs:label`` or ``skos:prefLabel`` to provide human-readable labels for all resources, and ``rdfs:comment`` for descriptive comments.

→ ist good practice

7. **\*\*Specify Language for Literal Values\*\***: For literal values, especially for ``szd:dedication``, ``szd:marginalia``, and ``szd:markings``, you should specify the language tag, e.g., ``"Anmerkungen von Stefan Zweig in Bleistift"@de``.

→ ist good practice

8. **\*\*Add More Information About The Author and Publisher\*\***: Include more information about the author, such as nationality or biography, and about the publisher, such as founding date or type (e.g., commercial, university press).

→ möglich, aber etwas anderes Thema

# **“Multi Output Generation”**

## RDFS erzeugen

# Erzeugung des RDFS auf Basis der erarbeiteten Prompts



Creating an RDF Schema (RDFS) involves defining classes and properties that will be used in the dataset. In this case, it seems you have used properties from several existing vocabularies (Schema.org, FRBR, Dublin Core, etc.), and a few custom properties under the ``szd:`` namespace.

Here is a basic RDFS file defining those custom ``szd:`` properties in English and German:

turtle

Copy code

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix szd: <http://stefanzweig.digital/ontology/> .

szd:dedication
  a rdfs:Property ;
  rdfs:label "dedication"@en ;
  rdfs:label "Widmung"@de ;
  rdfs:comment "A personal message written in the book by the author or a
  rdfs:comment "Eine persönliche Nachricht, die vom Autor oder einem frühe
```

```

1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
2 @prefix szd: <http://stefanzweig.digital/ontology/> .
3
4 szd:dedication
5   a rdfs:Property ;
6   rdfs:label "dedication"@en ;
7   rdfs:label "Widmung"@de ;
8   rdfs:comment "A personal message written in the book by the author or a previous owner."@en ;
9   rdfs:comment "Eine persönliche Nachricht, die vom Autor oder einem früheren Besitzer im Buch geschrieben wurde."@de .
10
11 szd:marginalia
12   a rdfs:Property ;
13   rdfs:label "marginalia"@en ;
14   rdfs:label "Marginalien"@de ;
15   rdfs:comment "Notes written in the margins of the book."@en ;
16   rdfs:comment "Notizen, die in den Rändern des Buches geschrieben wurden."@de .
17
18 szd:markings
19   a rdfs:Property ;
20   rdfs:label "markings"@en ;
21   rdfs:label "Markierungen"@de ;
22   rdfs:comment "Marks or highlights made in the book."@en ;
23   rdfs:comment "Markierungen oder Hervorhebungen, die im Buch gemacht wurden."@de .
24
25 szd:originalSignature
26   a rdfs:Property ;
27   rdfs:label "original signature"@en ;
28   rdfs:label "Originalsignatur"@de ;
29   rdfs:comment "The original signature or code marked in the book."@en ;
30   rdfs:comment "Die ursprüngliche Signatur oder der Code, der im Buch markiert wurde."@de .
31
32 szd:currentLocation
33   a rdfs:Property ;
34   rdfs:label "current location"@en ;
35   rdfs:label "aktueller Standort"@de ;
36   rdfs:comment "The current physical location of the book."@en ;
37   rdfs:comment "Der aktuelle physische Standort des Buches."@de .

```

## Multiple Output Approach

```
1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
2 @prefix szd: <http://stefanzweig.digital/ontology/> .
3 @prefix cidoc: <http://www.cidoc-crm.org/cidoc-crm/> .
4
5 szd:dedication
6   a rdfs:Property ;
7   rdfs:label "dedication"@en ;
8   rdfs:label "Widmung"@de ;
9   rdfs:comment "A personal message written in the book by the author or a previous owner."@en ;
10  rdfs:comment "Eine persönliche Nachricht, die vom Autor oder einem früheren Besitzer im Buch geschrieben wurde."@de ;
11  rdfs:domain cidoc:E84_Information_Carrier ;
12  rdfs:range rdfs:Literal .
13
14 szd:marginalia
15   a rdfs:Property ;
16   rdfs:label "marginalia"@en ;
17   rdfs:label "Marginalien"@de ;
18   rdfs:comment "Notes written in the margins of the book."@en ;
19   rdfs:comment "Notizen, die in den Rändern des Buches geschrieben wurden."@de ;
20   rdfs:domain cidoc:E84_Information_Carrier ;
21   rdfs:range rdfs:Literal .
22
23 szd:markings
24   a rdfs:Property ;
25   rdfs:label "markings"@en ;
26   rdfs:label "Markierungen"@de ;
27   rdfs:comment "Marks or highlights made in the book."@en ;
28   rdfs:comment "Markierungen oder Hervorhebungen, die im Buch gemacht wurden."@de ;
29   rdfs:domain cidoc:E84_Information_Carrier ;
30   rdfs:range rdfs:Literal .
31
```

GPT-unterstützte Erstellung eines  
Transformationsszenarios von CSV nach  
RDF mit Python und rdflib.



Write a Python script that transforms data from a csv file into an rdf file using rdflib.

CSV:

'''

ID, Verfasser/in, Titel, Veröffentlichung, Reihe, Sprache, Umfang/Einband, Widmung,  
Marginalie, Merkzeichen, Originalsignatur, Heutiger Standort:

SZDBIB.365, "Abraham, Pierre", "Proust : recherches sur la création intellectuelle", "Paris :  
Rieder, 1930", "Maîtres des littératures 7", Französisch, "91, LX Seiten : illustriert. 8°,  
Broschur", "Pour Stefan Zweig, | en sympathie et en admiration | Pierre Abraham.",  
"Anmerkungen von Stefan Zweig in Bleistift", "Unterstreichungen in Bleistift", "B 3" "Erben  
Stefan Zweigs, London A.I.01"

'''

The output should correspond to this rdf file:

'''

@prefix dc: <http://purl.org/dc/elements/1.1/> .

@prefix dcterms: <http://purl.org/dc/terms/> .

@prefix foaf: <http://xmlns.com/foaf/0.1/> .

@prefix schema: <http://schema.org/> .

@prefix szd: <http://stefanzweig.digital/ontology/> .

@prefix frbr: <http://purl.org/vocab/frbr/core#> .

@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .





This is a very straightforward task with Python's CSV and RDFLib libraries. Before we write the code, we need to know a few more things:

1. The exact structure of your CSV file.
2. The mapping of CSV columns to RDF triples (subject, predicate, object).

Based on the CSV and RDF data you have provided, I will assume the following mappings:

- "ID" -> Book ID (subject)
- "Verfasser/in" -> Author's name, which we can split into "givenName" and "familyName"
- "Titel" -> Book title (``schema:name``)
- "Veröffentlichung" -> Publication data, which we can split into "datePublished" and "publisher"
- "Reihe" -> Part of series (``dcterms:isPartOf``)
- "Sprache" -> Language of the book (``schema:inLanguage``)
- "Umfang/Einband" -> Number of pages (``schema:numberOfPages``)
- "Widmung" -> Dedication (``szd:dedication``)
- "Marginalie" -> Marginalia (``szd:marginalia``)
- "Merkzeichen" -> Markings (``szd:markings``)
- "Originalsignatur" -> Original signature (``szd:originalSignature``)
- "Heutiger Standort" -> Current location (``szd:currentLocation``)





\*szd:SZDBIB.365 = <https://gams.uni-graz.at/o:szd.bibliothek#SZDBIB.365>

\*szd:originalSignature = <https://www.stefanzweig.digital/o:szd.ontology#originalSignature>



The rest looks good. continue.



Thanks for your clarification. Now I will write a Python script to transform the CSV data into RDF using RDFLib. Please make sure to install RDFLib in your Python environment using pip:



Copy code

```
pip install rdflib
```

And here is the script:

python

Copy code

```
import csv
from rdflib import Graph, URIRef, Namespace, NamespaceManager
from rdflib.namespace import RDF, RDFS, XSD, NamespaceManager
```

Share



Regenerate response



Continue generating

**Einige Fehlerkorrekturen ... nur das Kopieren der Fehlermeldung zurück in GPT**

```
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix frbr: <http://purl.org/vocab/frbr/core#> .
@prefix schema1: <http://schema.org/> .
@prefix szd: <https://www.stefanzweig.digital/o:szd.ontology#> .
```

```
<https://gams.uni-graz.at/o:szd.bibliothek#SZDBIB.365> a schema1:Book ;
  dcterms:isPartOf <https://gams.uni-graz.at/o:szd.bibliothek#series.1> ;
  frbr:exemplar <https://gams.uni-graz.at/o:szd.bibliothek#SZDBIB.365_Copy> ;
  schema1:author <https%3A//gams.uni-graz.at/o%3Aszd.bibliothek%23SZDPER.1> ;
  schema1:bookFormatType <http://rdaregistry.info/termList/RDAContentType/> ;
  schema1:datePublished "Unknown" ;
  schema1:inLanguage <http://id.loc.gov/vocabulary/iso639-1/fr> ;
  schema1:name " Pierre\" ;
  schema1:numberOfPages " \"Maîtres des littératures 7\" ;
  schema1:publisher <https://gams.uni-graz.at/o:szd.bibliothek#pub.1> .
```

```
<https%3A//gams.uni-graz.at/o%3Aszd.bibliothek%23SZDPER.1> a schema1:Person ;
  schema1:name "[ ' \"Abraham' ]" .
```

```
<https://gams.uni-graz.at/o:szd.bibliothek#SZDBIB.365_Copy> a frbr:Item ;
  szd:currentLocation " \"Pour Stefan Zweig\" ;
  szd:dedication " Französisch\" ;
  szd:marginalia " \"91\" ;
  szd:markings " LX Seiten : illustriert. 8°\" ;
  szd:originalSignature " Broschur\" .
```

```
<https://gams.uni-graz.at/o:szd.bibliothek#pub.1> a schema1:Organization ;
  schema1:location "Paris, France\" ;
  schema1:name "\"Proust : recherches sur la création intellectuelle\"\" .
```

# Zusammenfassung

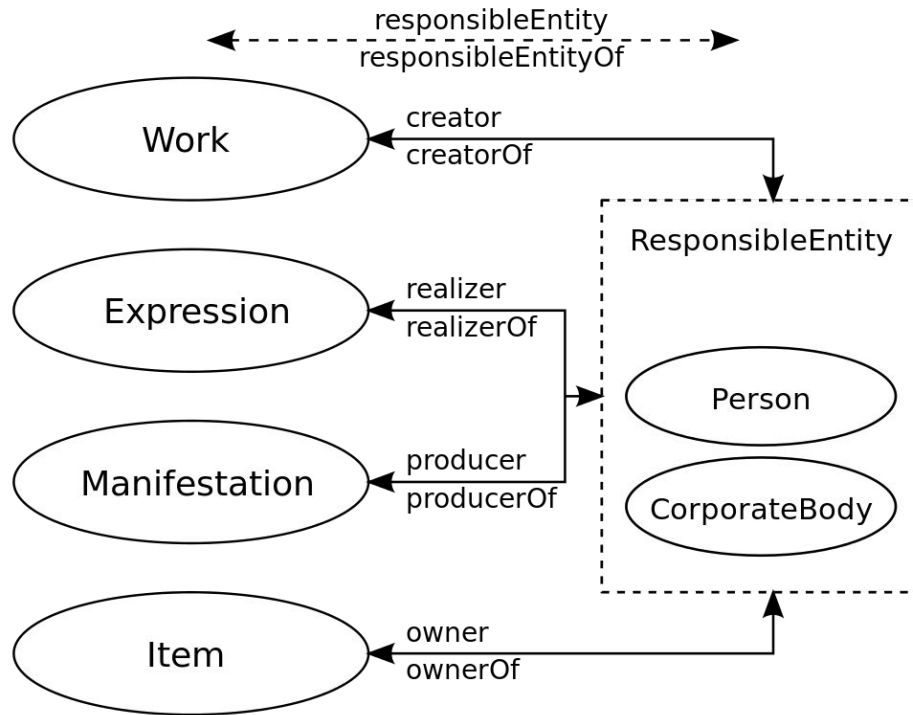
- Welche Classes und Properties verwende ich?
  - Unterstützung bei der Suche nach Vokabularien und Ontologien, die zur Domäne passen.
  - Weiteres Experimentieren erforderlich: Wie komplex kann man das machen?  
Funktioniert es für alle oder nur für bestimmte Domänen?  
Wie helfen GPT-4 Live Web Access + Tools und Literatursuche?
- Datentransformation von strukturierten (und unstrukturierten) Daten funktioniert gut mit GPT-4.
- GPT-4 “versteht” Text und Domäne: es macht (selbstständig) komplexere Modellierungsentscheidungen: frbr:Work vs. frbr:Item
- Aber bleibt es ein Expert\*innen-Tool?
- “Multi Output Generation”
  - Modell
  - RDFS
  - Python Script

# **GPT-unterstütztes Ontology Engineering**

## Am Beispiel von Transaktionen in historischen Rechnungsunterlagen.

<https://chat.openai.com/share/410016b5-a8a0-4d4c-9632-7395169388d0>

Anhang



- **Werk:** "Proust : recherches sur la création intellectuelle" von Autor X.
- **Ausdrücke:** Originaltext in Französisch, Englische Übersetzung, überarbeitete Ausgabe.
- **Manifestationen:** Erste Druckauflage 2023, PDF-Version auf der Website der Bibliothek.
- **Exemplare:** Jedes physische Buch in der Bibliothek, jede heruntergeladene Kopie der PDF-Version.

FRBR Beispiel Proust. GPT-4, ChatGPT May 24  
Version. 14.0.2023.

<https://chat.openai.com/share/9ae6568f-2db9-407a-8eab-47cef0d89830>