A semantic ontology for intelligent library systems, built in OWL and Protégé

1. Business Context

Modern libraries manage vast collections of books, authors, editions, and metadata. Traditional relational databases support basic cataloguing but struggle to queries with natural language that require richer semantic representation that includes understanding of meaning (DeRidder, 2007), not just stored values.

Ontology-based modelling provides a structured, machine-interpretable representation of knowledge that supports inference, classification, and semantic search (Byrne and Goddard, 2010). Using Web Ontology Language (OWL) and the open-source ontology editor software Protégé, this project develops a library ontology capable of representing books, authors, copies, formats, subjects, editors, translators, and availability. The ontology enables automated reasoning to infer new knowledge, such as identifying available copies and their formats or classifying books by genre or audience, and thereby supporting more intelligent library services.

2. Justification of the Approach

An ontology-driven approach is appropriate for this domain because libraries naturally contain hierarchical categories –e.g., genres, subjects, formats-, well-defined relationships -e.g., “hasAuthor”, “hasCopy”- and constraints -e.g., a copy belongs to exactly one book-. OWL provides the expressive power to model these relationships formally, while Protégé offers a practical environment for implementation and reasoning.

Unlike relational schemas, ontologies support (Swe, 2011):

- Open-world reasoning, allowing incomplete but extensible knowledge.
- Class definitions and restrictions, enabling automated classification.

- Reusability and extensibility, allowing new genres, formats, or roles to be added without redesigning the schema.

This approach is justified because the goal is not merely to store data but to enable semantic search and inference (Byrne and Goddard, 2010). For example, a user may want to retrieve all "TranslatedBook", a class that does not need to be explicitly stated but is inferred from the registration of a book translator. OWL's reasoning capabilities make this possible (DeBellis, 2021).

3. Rationale for Ontology Design Choices

3.1 Class Structure

The ontology distinguishes between conceptual entities (classes) and concrete instances (individuals). Core classes include Book, Copy, Author, Publisher, Editor, Translator, Genre, Subject, Format, Audience, and Shelf (see Appendix Figure 1 and Figure 2).

Concepts such as Genre, Subject, Format, and Audience were modelled as classes, not individuals, because they represent categories with potential sub-hierarchies (DeBellis, 2021). This choice supports extensibility, allowing future expansion, like adding subgenres or new audience categories, without restructuring the ontology.

A limitation, however, is that modelling genres and subjects as classes means object property assertions cannot directly link individuals to these categories. Instead, individuals must be typed as instances of these classes. This is conceptually clean but may be less intuitive for users expecting a "hasGenre" assertion at the individual level (see Appendix Figure 6 and Figure 8).

3.2 Object Properties

Object properties capture semantic relationships, linking to concepts through identified domains and ranges (DeBellis, 2021) that can be represented below as (domain → range):

- **hasAuthor** (Book → Author)
- **hasPublisher** (Book → Publisher)
- **hasCopy** (Book → Copy)
- **copyOf** (Copy → Book)
- **hasFormat** (Copy → Format)
- **locatedOnShelf** (Copy → Shelf)
- **hasEditor** (Book → Editor)
- **hasTranslator** (Book → Translator)

Domains and ranges were carefully assigned to avoid unintended inferences. A strength of this approach is the clarity and consistency of the property schema (see Appendix Figure 6).

3.3 Data Properties

Literal attributes such as titles, names, publication years, and barcodes were modelled as data properties, not classes, to avoid over-modelling and keeps the ontology lightweight (DeBellis, 2021). Relevant domains were set. For example, “Book” is the domain of “title”. Also, the correct data type was assigned to each data property. For example, “title” has a range of “xsd:string” (see Appendix Figure 7). As setting correct data type ranges is fundamental for data quality (DeBellis, 2021), this potentially can be improved by introducing controlled vocabularies to avoid misspellings and typos in titles and names or regex patterns for barcodes.

3.4 Restrictions and Class Definitions

Restrictions were added to enable reasoning:

- Copy $\sqsubseteq$ copyOf exactly 1 Book

- Copy ⊑ locatedOnShelf exactly 1 Shelf
- AvailableCopy ≡ Copy and (hasStatus value Available)
- BorrowableBook ≡ Book and (hasCopy some AvailableCopy)
- DigitalCopy ≡ Copy and (hasFormat some DigitalFormat)
- ChildrenBook ≡ Book and (hasAudience some Child)

These definitions allow Protégé's reasoner to classify individuals automatically (DeBellis, 2021). A strength here is the clear separation between necessary and sufficient conditions (see Appendix Figure 9).

4. Analysis of Outputs and Evidence of Testing

4.1 Creation of Individuals

To test the ontology, three complete book entries were added as individuals, along with their authors, publishers, translators if translated, and one copy for each. The books are:

- Orientalism by Edward Said (1978)
- The Lodging House by Khairy Shalaby (2002)
- Difference and Repetition by Gilles Deleuze (1968)

Each book was linked to its author, publisher, genre, subject, audience, and at least one copy. Translators were added where appropriate.

4.2 Reasoner Results

Running the Pellet reasoner produced several successful inferences:

- **Orientalism_1978** was classified as **AdultBook** and **NonFictionBook**.

- **TheLodgingHouse_2002** was inferred to be a **AdultBook**, **FictionBook** and **TranslatedBook**.

- **DifferenceAndRepetition** was classified under **AdultBook, FictionBook** and **TranslatedBook.**

- Copies were correctly classified as **AvailableCopy** and **PhysicalCopy**.

These results confirm that the ontology behaves as intended. A strength of the design is that reasoning produces meaningful, domain-relevant classifications without manual tagging (see Appendix Figure 8). However, not all inferences resulted as expected. For example, the three books should have been inferred as “BorrowableBooks”, which was not the case. This likely points to the need for more sophisticated axioms for relevant classes and object properties.

A further limitation observed during testing is that the ontology currently lacks temporal modelling, like borrowing history, and cannot represent multiple editions of the same book. These would be valuable extensions.

5. Summary of Strengths and Limitations

The methods used in this ontology, with its clear class hierarchies, object properties, restrictions and reasoning, are directly applicable to real-world library systems, serving semantic search engines, as well as borrowing systems, cataloguing workflows and recommendation engines.

However, foreseeable limitations include:

- More axioms are needed to allow wider inferences, like BorrowableBooks.
- Modelling genres and subjects as classes prevents direct object-property assertions at the individual level, which may feel unintuitive.

- No temporal modelling to represent borrowing events, due dates and circulation history.

- No representation of multiple editions, series, or co-authorship, which are common in bibliographic data.
- Possibility of integration with external linked-data standards (BIBFRAME, Dublin Core, schema.org) needs to be further studied and planned to allow potential interoperability (Jett *et al.*, 2016).
- Current ontology does not include multilingual metadata or cross-edition relationships.

6. Conclusion

This project demonstrates that ontology-based modeling provides a strong solution to modern library knowledge management challenges by enabling semantic search, automated classification, and logical consistency. The design, focused on explicit class hierarchies and property restrictions, was justified by the need for accurate and scalable reasoning about bibliographic entities and their relationships. Empirical testing confirmed the effectiveness of the ontology, though some limitations were noted, such as restricted temporal modeling and the absence of multi-editions metadata. Importantly, this approach is well suited to real-world applications, mainly library search engines, but also cataloguing, recommendations, and borrowing workflows. Future improvements should address integration with external standards, modeling editions and co-authorship, and temporal events.

7. Reference List

Byrne, G. and Goddard, L. (2010) ‘The Strongest Link: Libraries and Linked Data’, *D-Lib Magazine*, November. Available at: https://www.dlib.org/dlib/november10/byrne/11byrne.html (Accessed: 17 January 2026).

DeBellis, M. (2021) *Practical Guide to Building OWL Ontologies Using Protégé 5.5 and Plugins*. Available at: https://drive.google.com/file/d/1A3Y8T6nIfXQ_UQOpCAr_HFSCwpTqELeP/view (Accessed: 17 January 2026).

DeRidder, J.L. (2007) ‘The Immediate Prospects for the Application of Ontologies in Digital Libraries’, *Knowledge Organization*, 34(4), pp. 227–246. Available at: https://doi.org/10.5771/0943-7444-2007-4-227.

Jett, J. et al. (2016) ‘Enhancing Scholarly Use of Digital Libraries: A Comparative Survey and Review of Bibliographic Metadata Ontologies’, in *Proceedings of the 16th ACM/IEEE-CS on Joint Conference on Digital Libraries. 16th ACM/IEEE-CS on Joint Conference on Digital Libraries*, New York, NY, USA: Association for Computing Machinery, pp. 35–44. Available at: https://dl.acm.org/doi/pdf/10.1145/2910896.2910903 (Accessed: 17 January 2026).

Swe, T.M. (2011) ‘Intelligence Information Retrieval within Digital Library Using Domain Ontology’, Computer Science & Information Technology (CS & IT), 02(2011), pp. 363–373. Available at: https://doi.org/10.5121/csit.2011.1232.

8. Appendix

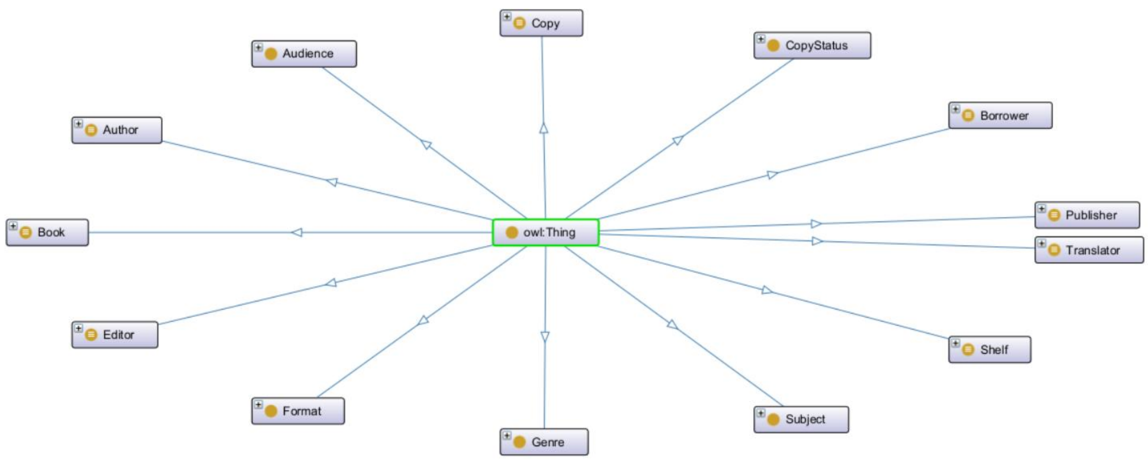

Figure 1 Visualisation of main classes using OntoGraph | Source: Author's own image

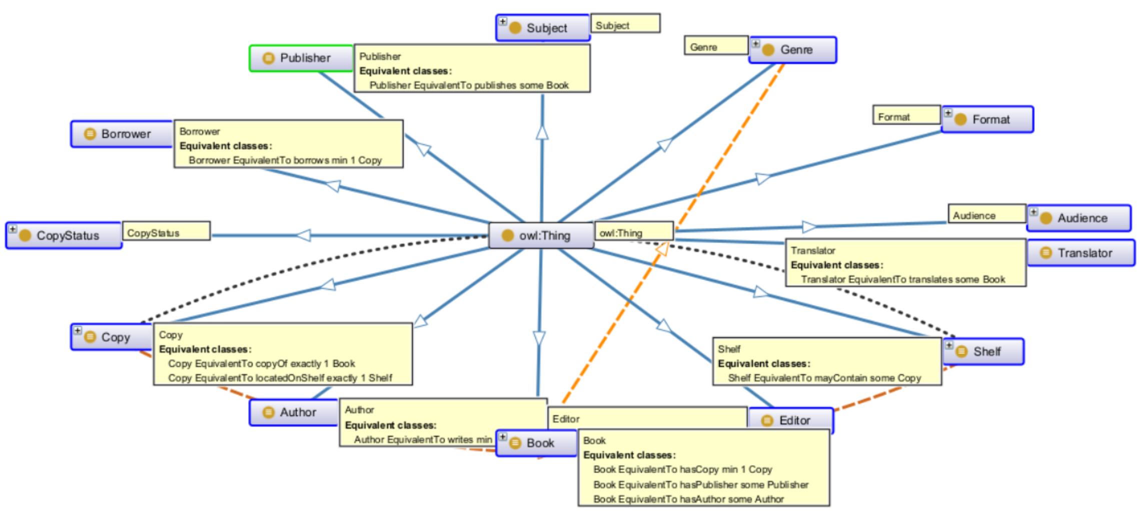

Figure 2 Main classes connected and annotated | Source: Author's own image

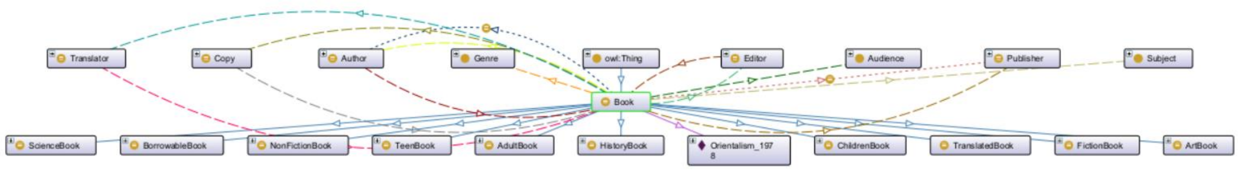

Figure 3 Focus on "Book" class, its subclasses, related classes and one individual | Source: Author's own image

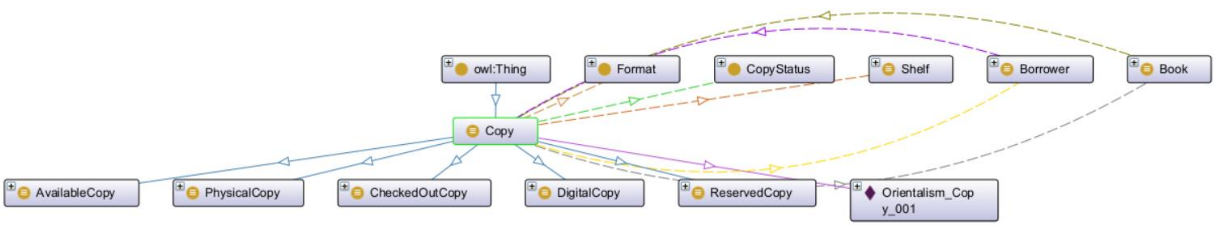

Figure 4 Focus on "Copy" class, its subclasses, related classes and one individual | Source: Author's own image

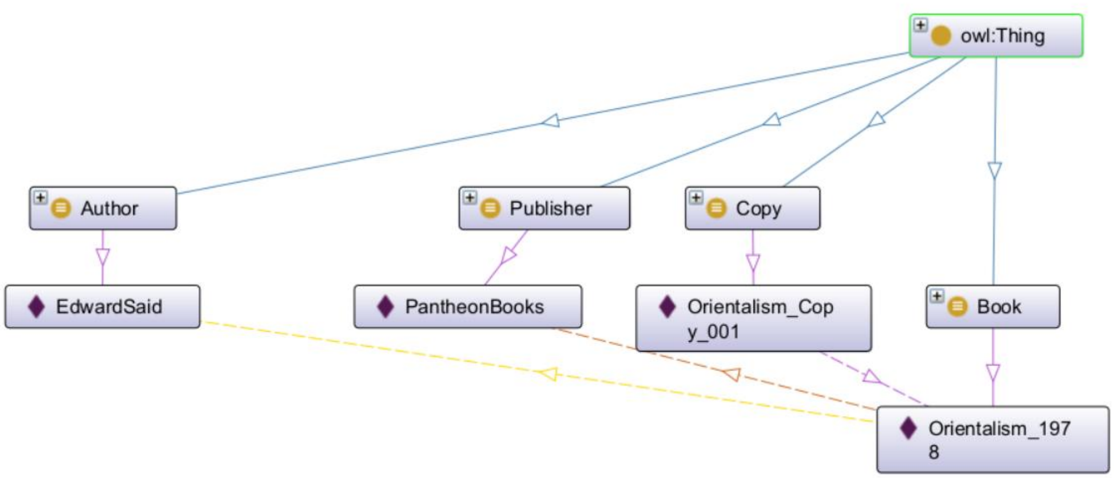

Figure 5 Example for one individual/Book (Ortientalism_1978) | Source: Author's own image

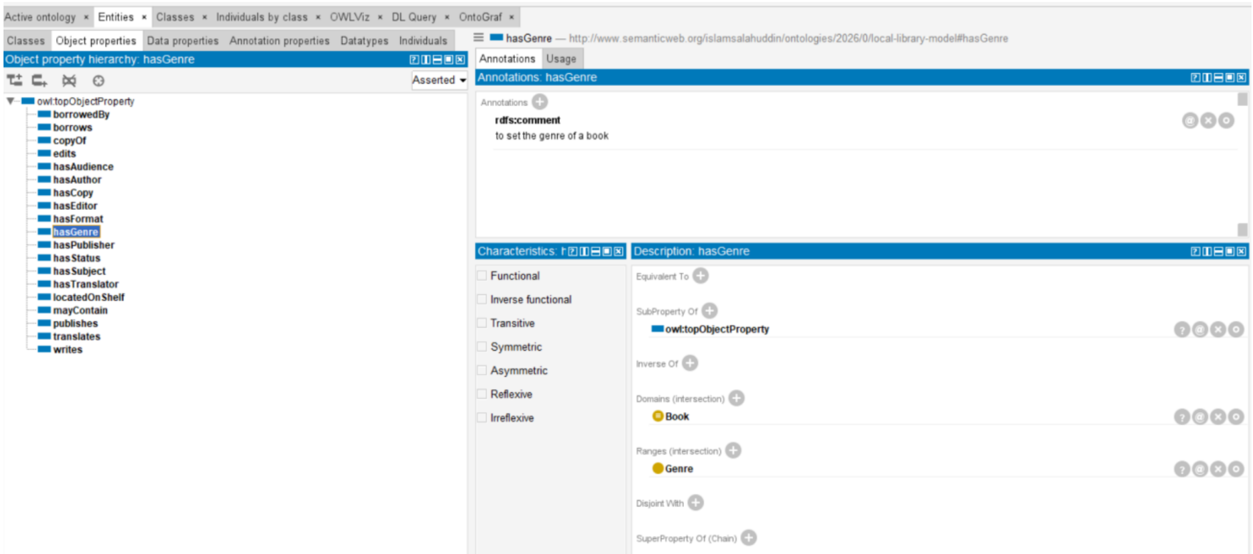

Figure 6 Full list of object properties and example set domain and range | Source: Author's own image

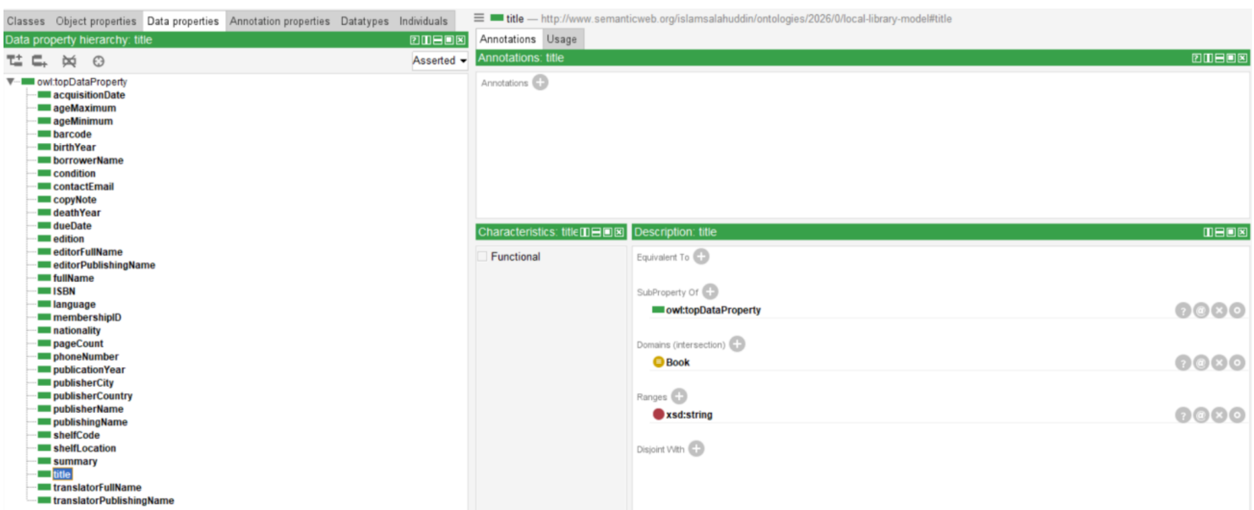

Figure 7 Full list of data properties and example set domain and range | Source: Author's own image

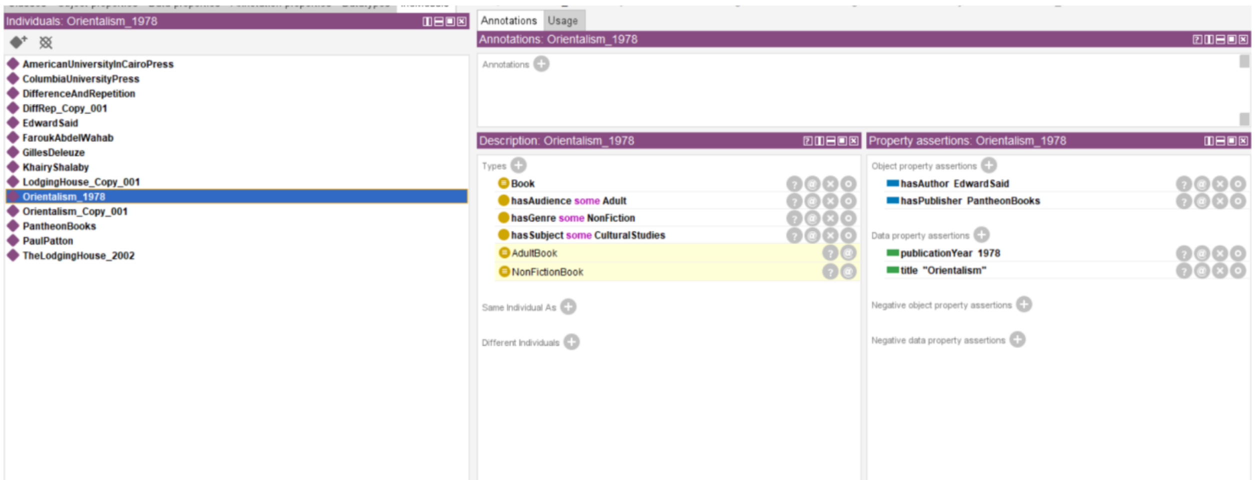

Figure 8 Full list of individuals added for the three books with the defined and inferred classifications and relations | Source: Author's own image

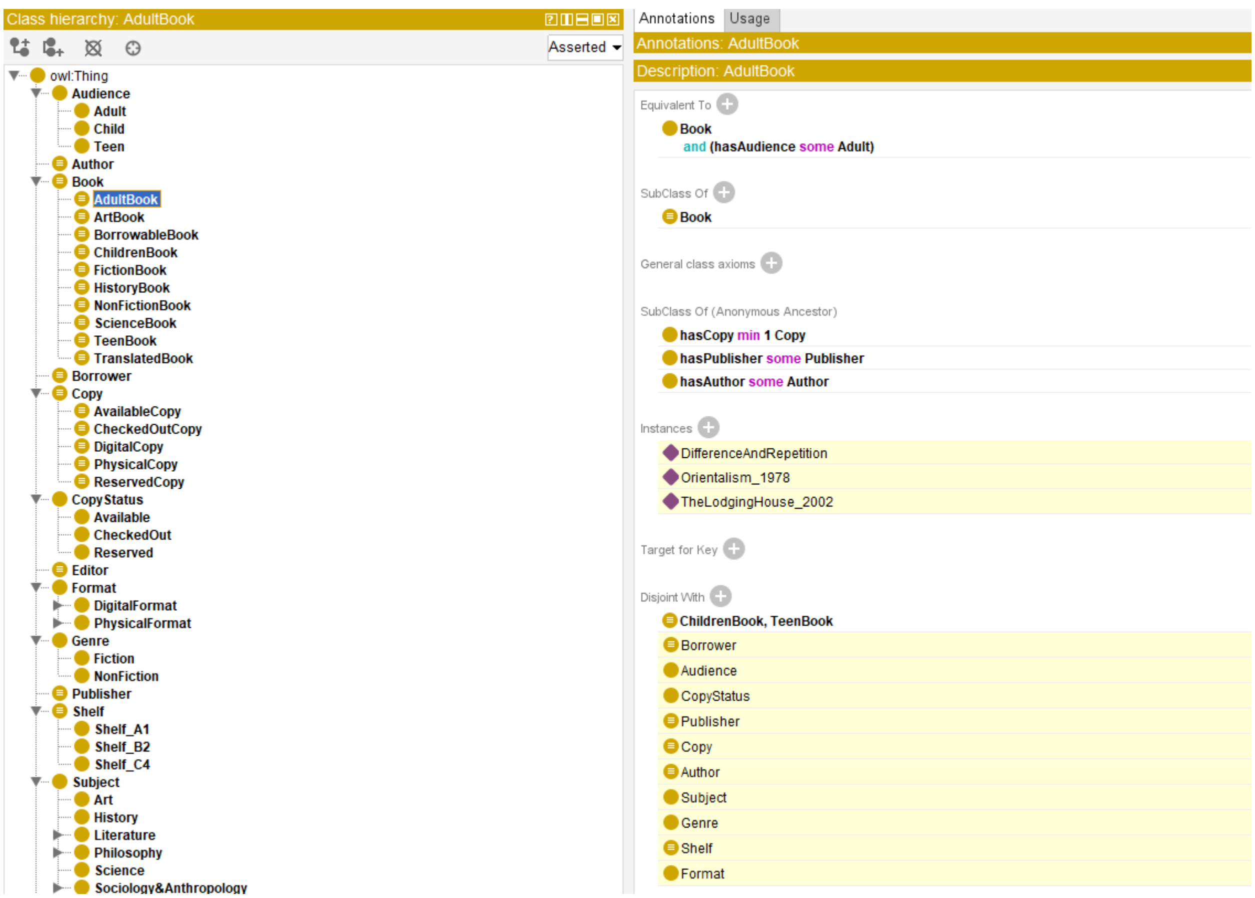

Figure 9 Example class restrictions and inferred instances | Source: Author's own image