DCPAPERS

BIBFRAME, AI, and Open-Source Technologies: A Blue Core Project Report

Kalliopi Mathios[1,†], Jeremy Nelson[1,*,†]

[1]Stanford University Libraries, Stanford University, 557 Escondido Mall, Stanford, CA, 94305, USA

Abstract

The Blue Core project envisions a community-operated and owned BIBFRAME linked data store, where libraries collaborate in the creation and maintenance of linked data for production. As the project continues its work in prototype development, open-source tooling and AI technologies lay the groundwork for future iterations. With open data a pillar of the project, data contributed to Blue Core will be made available under a CC0 public domain dedication and available to all for reuse at no charge.

Keywords

Blue Core, Linked Data for Production, linked data, open data

1. Introduction

Blue Core was introduced at the 2023 BIBFRAME Workshop in Europe by Philip E. Schreur, Tom Cramer, Jason Kovari, and Simeon Warner, Principal Investigators of the Andrew W. Mellon funded Linked Data for Production project.[1] As early as 2014, Linked Data for Libraries, Linked Data for Libraries Labs, and Linked Data for Production (LD4P), advanced linked data in libraries by developing new tooling for BIBFRAME cataloging, while creating a movement built on multi-institution collaboration and openness. The LD4 Community, a direct result of LD4P, remains a vibrant hub of linked data practitioners.[2]

2. A Community Operated BIBFRAME Data Store

Following LD4P, a new collaboration between the Library of Congress, University of Pennsylvania, Stanford University, University of California Davis, and Cornell University, Blue Core envisions a linked data environment that incorporates lessons learned during LD4P and transitions MARC-based cataloging practices dependent on local tooling and record exchange to the shared creation and maintenance of BIBFRAME RDF. Current work is underway to create a community-operated BIBFRAME data store where ownership and creation of the metadata are shared among member institutions, eliminating the need for duplicative institutional copies, and bringing library linked open data to production at scale.

DCMI-2025 International Conference on Dublin Core and Metadata Applications

*Corresponding author.

†These authors contributed equally.

kmathios@stanford.edu (K. Mathios); jpnelson@stanford.edu (J. Nelson)

0009-0006-9294-742X (K. Mathios); 0000-0002-7543-3893 (J. Nelson)

https://doi.org/10.23106/dcmi.952583208

To accomplish these goals, participating libraries will leverage existing tools developed during LD4P. Unlike the Sinopia editor which allows for any kind of RDF data to be created, Blue Core will exclusively support BIBFRAME. The Blue Core BIBFRAME data store will integrate with both the Sinopia and Marva linked data editors, local Library Services Platforms (LSPs) such as Alma and FOLIO, and be made openly available through APIs that drive services. Available under a CC0 public domain dedication, all data contributed by member institutions will be open and free for all.

3. Blue Core Technical Architecture

The Blue Core technical infrastructure (see Figure 1) uses well-supported open-source projects that are also leveraged in library technology projects like FOLIO and Sinopia. The Blue Core API is built using FastAPI[3] with authentication and authorization provided by Keycloak.[4] The API manages all traffic between the external editors as well as other technologies like Jupyter Notebooks and command-line tools like curl to the Blue Core data store, implemented with Postgres.[5] The API also connects to the Blue Core Workflows that use Apache Airflow[6] for batch processing. The Blue Core data store and API also use Milvus[7] to store vectors of BIBFRAME Work and Instance RDF for use by AI Agents.

4. Next Steps

The Blue Core prototype technology stack is currently running on an AWS EC2 virtual machine using Docker containers. Our plan includes migrating to Kubernetes for the Minimum Viable Product and production stages of the project. The Blue Core API is built on the OpenAPI specification used by the Sinopia API project.[8] The Blue Core Workflows is a fork of the Sinopia-ILS Middleware project[9] with some modifications. Blue Core will use Sinopia's model of providing open read access for all RDF resources while requiring authentication for adding or editing resources. For reporting and management of the RDF, Blue Core may support a variant of the Sinopia Graph Explorer[10] that enables users to execute SPARQL queries against a subset of Blue Core data. In addition, Blue Core will leverage work being done by the BIBFRAME Interoperability Group to validate incoming BIBFRAME entities to promote normalization and reuse of its managed resources. Recognizing the potential of generative Artificial Intelligence, particularly Large Language Models (LLMs) like ChatGPT, Claude, and Google Gemini, Blue Core's technical infrastructure was designed from the beginning to allow integration of AI Agents with a popular open-source project PydanticAI.[11] The first AI Agent is a BIBFRAME Work and Instance duplication agent that will evaluate incoming RDF for close matches with existing BIBFRAME entities and provide options for automated detection while supporting "human-in-loop" in de-duplication workflows. A second AI Agent is a chatbot that allows user interaction through a narrative interface.

https://doi.org/10.23106/dcmi.952583208

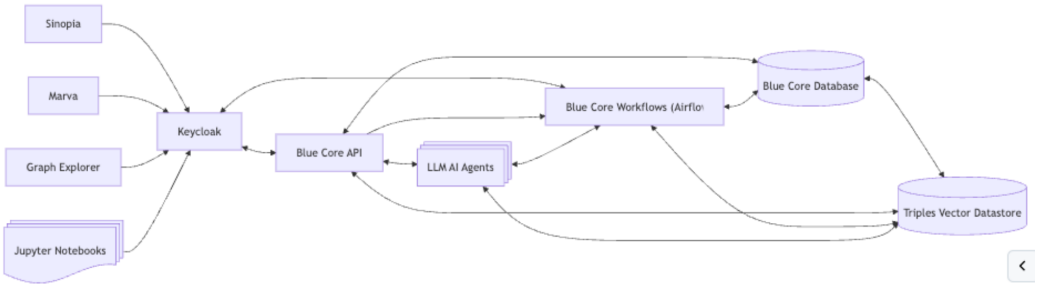

Figure 1: Blue Core technical architecture diagram. (https://github.com/orgs/blue-core-lod).

References

[1] P. Schreur, Linked Data for Production Phase 4: Truly Shared Data in an Institutionally Neutral Data Pool, Proceedings of BIBFRAME Workshop in Europe, Brussels Belgium, 2023. https://www.bfwe.eu/brussels_2023.
[2] LD4 Community Site. URL: https://ld4.io.
[3] FastAPI. URL: https://fastapi.tiangolo.com/.
[4] Keycloak. URL: https://www.keycloak.org/.
[5] Postgres. URL: https://www.postgresql.org/.
[6] Apache Airflow. URL: https://airflow.apache.org/.
[7] Milvus. URL: https://milvus.io/.
[8] Sinopia API 1.0.0. URL: https://ld4p.github.io/sinopia_api/.
[9] Sinopia ILS-Middleware. URL: https://github.com/ld4p/ils-middleware.
[10] Sinopia Graph Explorer. URL: https://github.com/ld4p/graph-explorer.
[11] PydanticAI. URL: https://ai.pydantic.dev/.

https://doi.org/10.23106/dcmi.952583208