



Linked Data Demos: The plot thickens

IGELU 2018 Developers Day | Prague Czech Republic

Laura Akerman

Discovery Systems and Metadata Librarian, Emory University

Josh Weisman

VP Development, Ex Libris



Linked Data Harvest

A brief exploration of things to do with BIBFRAME from Alma



 I am not a ~~farmer~~ developer



BIBFRAME - an RDF ontology and vocabulary

Bibliographic Framework Initiative

Initiated by the Library of Congress, BIBFRAME provides a foundation for the future of bibliographic description, both on the web, and in the broader networked world that is grounded in Linked Data techniques. A major focus of the initiative is to determine a transition path for the MARC 21 formats while preserving a robust data exchange that has supported resource sharing and cataloging cost savings in recent decades.



Model and Vocabulary (2.0)

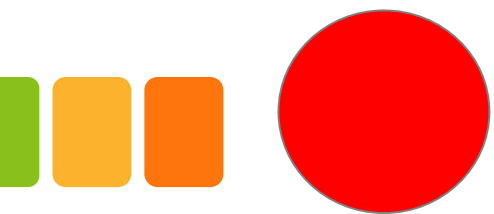
Model description, vocabulary presented in various views, guidelines, examples, analyses.

MARC 21 to BIBFRAME 2.0 Conversion Tools

What's New

BIBFRAME Update Forum at ALA Annual (June 24, 2018)

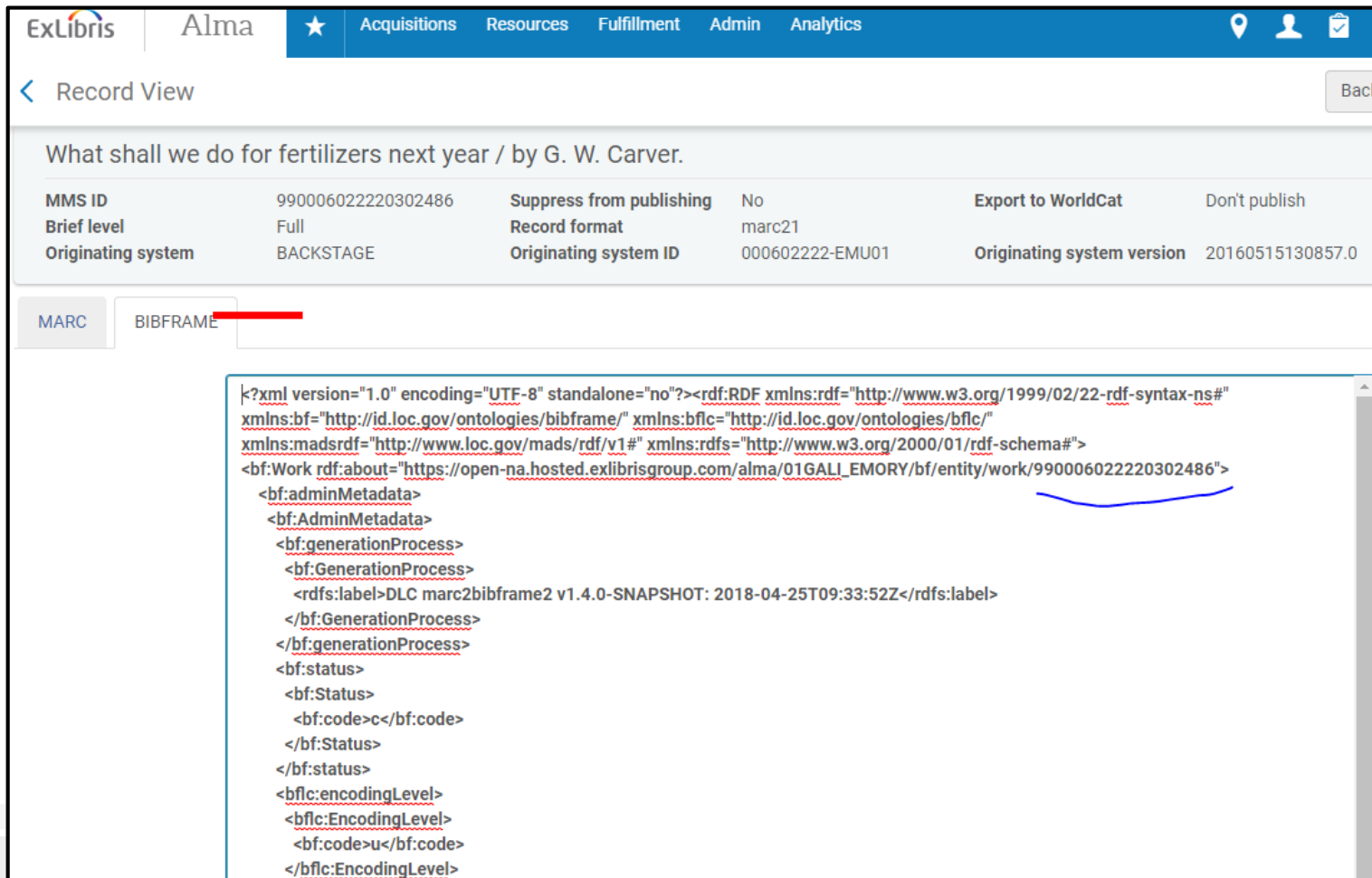
[View agenda and presentations »](#)



BIBFRAME is on the move...

- July 24, 2018 - at ALA Annual Meeting, Beacher Wiggins, Director for Acquisitions and Bibliographic access said "With this update, the library is announcing that it is committed to implementing BIBFRAME as a viable replacement for MARC."
- Quoted as saying elsewhere -- "within 2-5 years."

Gathering BIBFRAME from Alma web client



The screenshot displays the Alma web client interface. At the top, there is a navigation bar with the ExLibris logo, the Alma title, and a menu with options: Acquisitions, Resources, Fulfillment, Admin, and Analytics. Below this, a 'Record View' header is visible. The main content area shows the title 'What shall we do for fertilizers next year / by G. W. Carver.' and a table of metadata. The table includes fields such as MMS ID, Brief level, Originating system, Suppress from publishing, Record format, Originating system ID, Export to WorldCat, and Don't publish. Below the table, there are tabs for MARC and BIBFRAME. The BIBFRAME tab is selected, and a red line is drawn under the 'BIBFRAME' tab label. The BIBFRAME record is displayed in a text area, showing XML code. A blue line is drawn under the URL 'https://open-na.hosted.exlibrisgroup.com/alma/01GALI_EMORY/bf/entity/work/99000602220302486' in the XML code.

Record View

What shall we do for fertilizers next year / by G. W. Carver.

MMS ID	99000602220302486	Suppress from publishing	No	Export to WorldCat	Don't publish
Brief level	Full	Record format	marc21		
Originating system	BACKSTAGE	Originating system ID	000602222-EMU01	Originating system version	20160515130857.0

MARC BIBFRAME

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?><rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:bf="http://id.loc.gov/ontologies/bibframe/" xmlns:bfcl="http://id.loc.gov/ontologies/bfcl/"
xmlns:madsrdf="http://www.loc.gov/mads/rdf/v1#" xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
<bf:Work rdf:about="https://open-na.hosted.exlibrisgroup.com/alma/01GALI_EMORY/bf/entity/work/99000602220302486">
  <bf:adminMetadata>
    <bf:AdminMetadata>
      <bf:generationProcess>
        <bf:GenerationProcess>
          <rdfs:label>DLC marc2bibframe2 v1.4.0-SNAPSHOT: 2018-04-25T09:33:52Z</rdfs:label>
        </bf:GenerationProcess>
      </bf:generationProcess>
    </bf:status>
    <bf:Status>
      <bf:code>c</bf:code>
    </bf:Status>
  </bf:status>
  <bfcl:encodingLevel>
    <bfcl:EncodingLevel>
      <bf:code>u</bf:code>
    </bfcl:EncodingLevel>
```


Using Alma API to get BIBFRAME - so easy

- Activate the linked data profile
- Example request

https://open-na.hosted.exlibrisgroup.com/alma/o1GALI_EMORY/bf/entity/instance/99000602220302486

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?><rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <bf:Work rdf:about="https://open-na.hosted.exlibrisgroup.com/alma/o1GALI_EMORY/bf/entity/work/instance/99000602220302486">
    <bf:adminMetadata>
      <bf:AdminMetadata>
        <bf:generationProcess>
          <bf:GenerationProcess>
            <rdfs:label>DLC marc2bibframe2 v1.4.0-SNAPSHOT: 2018-04-29T01:03:54Z</rdfs:label>
          </bf:GenerationProcess>
        </bf:generationProcess>
        <bf:status>
          <bf:Status>
            <bf:code>c</bf:code>
          </bf:Status>
        </bf:status>
      </bf:AdminMetadata>
    </bf:adminMetadata>
  </bf:Work>
</rdf:RDF>
```

* Documented on Developer Network:

<https://developers.exlibrisgroup.com/alma/apis/bibs/GET/gwPcGlyo21om4RTvtjbPlc5mU+8DB2nncwZbtvBJcCRbmJlr9MLGJF78JRyq7vUf/af2fb69d-64f4-42bc-bb05-d8aoae56936e>



Publishing BIBFRAME from Alma

Publishing Profile Details

CancelSave

Profile Name	Akerman_bf	Profile Id	13766547600002486
--------------	------------	------------	-------------------

Profile Details

Data Enrichment

Profile Details

Content Type

Bibliographic

Profile Name *

Akerman_bf

Profile Description

Publish specified set as BIBFRAME

Publishing Parameters

Status

☒ Active ☐ Inactive

Scheduling

Not scheduled

Publishing Mode

☒ Incremental ☐ Full ☐ Date Range ☐ Republish Set

Email Notifications

Content

Set name *

cognition

Publish on:

Bibliographic level

Output format

BIBFRAME Bibliographic



...publishing details

Output format **BIBFRAME Bibliographic**

Publishing Protocol ▼

FTP ☒

FTP configuration * **turing** ▼

Sub-directory **laura_bf**

Disable file compression ☐

Compressed file extension **tar.gz**

Physical format **XML** ▼

Number of records in file **One File** ▼

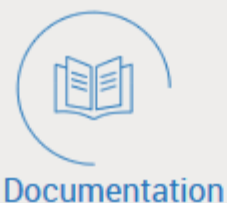
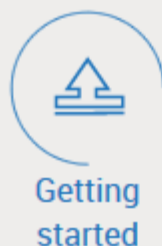
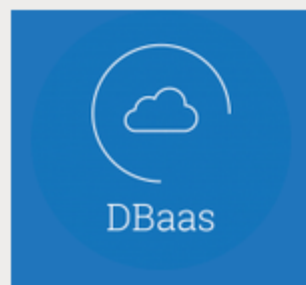
Use default file name ☐ No ☒ Yes

File name prefix * **BF**

Triplestore... so many choices... ?



<https://ontotext.com/products/graphdb/>



Try it for free

By hitting this button we'll redirect you to fill a tiny-mini form. Thank you in advance!

[or Compare editions](#)

GraphDB Database-as-a-Service (in the Cloud)

GraphDB DBaaS is the perfect solution for scenarios with small or medium database size and query load, where investing in software licenses and provisioning and maintaining an on-premise 24/7 server does not cost optimal.





This fully managed version of GraphDB is available as part of Ontotext Cognitive Platform along with a set of services for low-cost, on-demand text analytics and metadata management on the cloud.

It provides an enterprise-grade RDF Database-as-a-Service (DBaaS), where the user does not need to deal with data base administration tasks such as installation and upgrades, provisioning and deployment, backups and restores, as well as ensuring database availability.

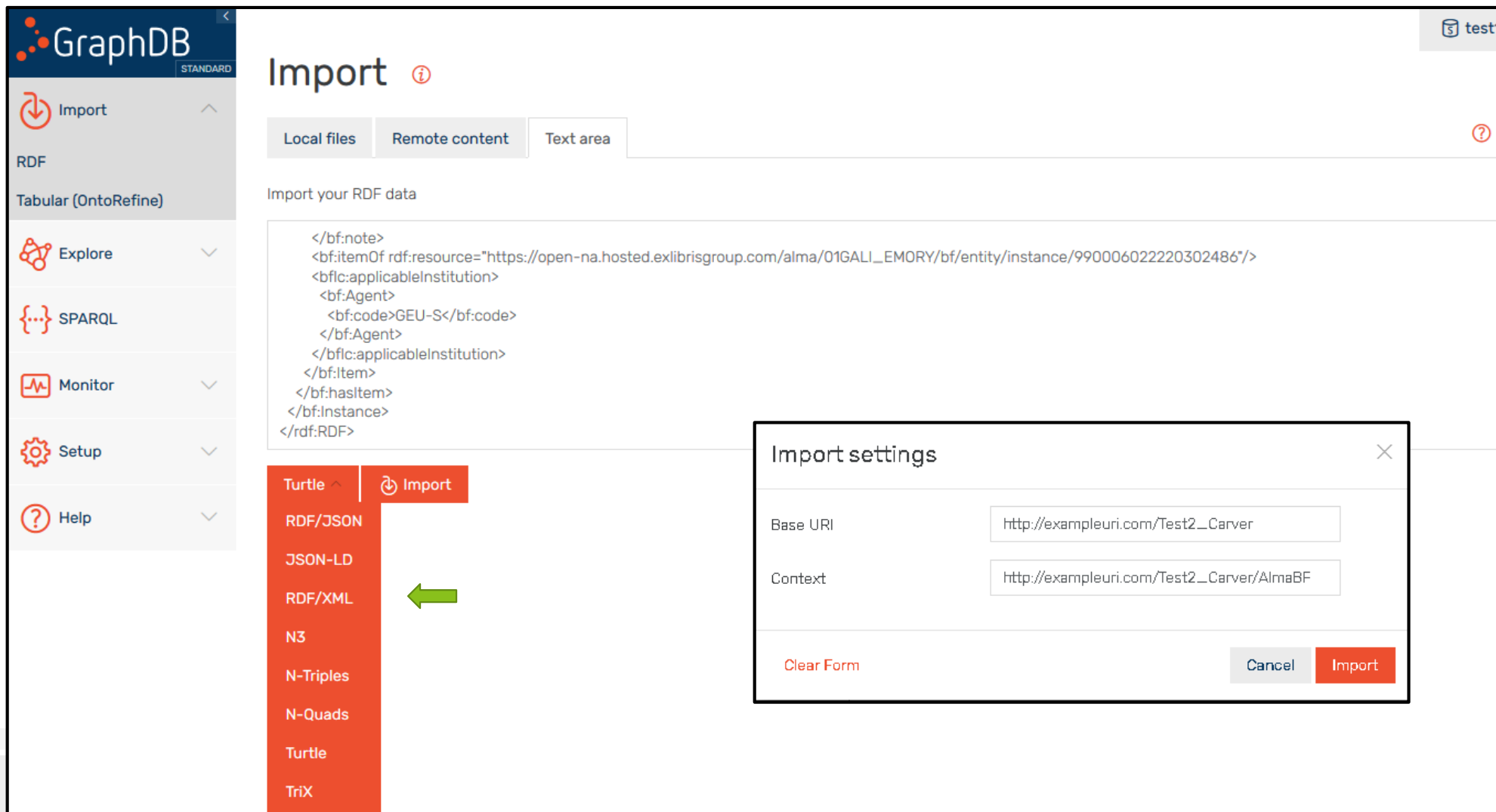
Databases

Databases

Create Database

BL1	Test2_Carver Type: Basic - Layer 1	 1%		<div>LaunchClone</div>
BL1	Akerm_Bibframe_experir Type: Basic - Layer 1	 3%		<div>Resume</div>

Test import copy/pasted from Alma cataloging



The screenshot displays the GraphDB web interface. On the left is a sidebar with navigation options: Import, RDF, Tabular (OntoRefine), Explore, SPARQL, Monitor, Setup, and Help. The main area is titled 'Import' and contains tabs for 'Local files', 'Remote content', and 'Text area'. The 'Text area' tab is active, showing a text input field with RDF data. Below the text area is a list of import formats: Turtle, RDF/JSON, JSON-LD, RDF/XML, N3, N-Triples, N-Quads, Turtle, and TriX. A green arrow points to the 'RDF/XML' option. An 'Import' button is located to the right of the format list. An 'Import settings' dialog box is open in the foreground, containing fields for 'Base URI' and 'Context', both with example values. The dialog also includes 'Clear Form', 'Cancel', and 'Import' buttons.

GraphDB STANDARD

Import

RDF

Tabular (OntoRefine)

Explore

SPARQL

Monitor

Setup

Help

Import

Local files Remote content Text area

Import your RDF data

```
</bf:note>
<bf:itemOf rdf:resource="https://open-na.hosted.exlibrisgroup.com/alma/01GALI_EMORY/bf/entity/instance/990006022220302486"/>
<bfic:applicableInstitution>
  <bf:Agent>
    <bf:code>GEU-S</bf:code>
  </bf:Agent>
</bfic:applicableInstitution>
</bf:item>
</bf:hasItem>
</bf:Instance>
</rdf:RDF>
```

Turtle ^ Import

RDF/JSON

JSON-LD

RDF/XML

N3

N-Triples

N-Quads

Turtle

TriX

Import settings





Base URI http://exampleuri.com/Test2_Carver

Context http://exampleuri.com/Test2_Carver/AlmaBF

Clear Form Cancel Import

Source: http://exampleuri.com/Test2_Carver/AlmaBF

subject	predicate	object	context	all
Explicit only 				
Show Blank Nodes Download as  Visual graph				

	subject 	predicate 	object 	context 
1	https://open-na.hosted.exlibrisgroup.com/alma/01GALI_EMORY/bf/entity/work/990006022220302486	bf:adminMetadata	_:node0	http://exampleuri.com/Test2_Carver/AlmaBF
2	https://open-na.hosted.exlibrisgroup.com/alma/01GALI_EMORY/bf/entity/work/990006022220302486	bf:content	http://id.loc.gov/vocabulary/contentTypes/txt	http://exampleuri.com/Test2_Carver/AlmaBF
3	https://open-na.hosted.exlibrisgroup.com/alma/01GALI_EMORY/bf/entity/work/990006022220302486	bf:contribution	_:node22	http://exampleuri.com/Test2_Carver/AlmaBF
4	https://open-na.hosted.exlibrisgroup.com/alma/01GALI_EMORY/bf/entity/work/990006022220302486	bf:contribution	_:node24	http://exampleuri.com/Test2_Carver/AlmaBF
5	https://open-na.hosted.exlibrisgroup.com/alma/01GALI_EMORY/bf/entity/work/990006022220302486	bf:contribution	_:node28	http://exampleuri.com/Test2_Carver/AlmaBF



Add more triples about G. W. Carver

- Bibframe -> Bibframe ontology
- Bibframe -> id.loc.gov authority data for Carver
- id.loc.gov authority -> VIAF
- no VIAF link to dbpedia or wikidata ☹ -skip it!
- Wikidata – Carver
- Added (manually) an **owl:sameAs** between the authority ID and Wikidata ID for Carver.

`<http://id.loc.gov/authorities/names/n50034776>`

`<http://www.w3.org/2002/07/owl#sameAs>`

`<http://www.wikidata.org/entity/Q296898>`

- No way to get Primo ID from Alma BIBFRAME directly...



Queries

- SPARQL endpoint allows simple and complex queries
- A simple query is "give me all your triples with the LC URI for Carver as object".

- PREFIX owl: <<http://www.w3.org/2002/07/owl#>>

```
SELECT DISTINCT ?s ?p
```

```
WHERE {
```

```
    ?s ?p <http://id.loc.gov/authorities/names/n50034776>.
```

```
} LIMIT 1000
```




SPARQL result

Filter query results

⚠ Showing results from 1 to 57 of 57. Query took 0.3s, yesterday at 20:4

	s	p
1	http://id.loc.gov/authorities/names/n50034776	owl:sameAs
2	wd:Q16124998	owl:sameAs
3	wd:Q296898	owl:sameAs
4	_:node9	bf:agent
5	_:node10	bf:identifies
6	_:node81	cs:subjectOfChange
7	_:node82	cs:subjectOfChange
8	wdata:Q296898	schema:about
9	https://zh.wikipedia.org/wiki/%E5%96%AC%E6%B2%BB%C2%B7%E8%8F%AF%E7%9B%9B%E9%A0%93%C2%B7%E5%8D%A1%E5%BC%97	schema:about
10	https://pl.wikipedia.org/wiki/George_Washington_Carver	schema:about
11	https://eu.wikipedia.org/wiki/George_Washington_Carver	schema:about
12	https://fr.wikipedia.org/wiki/George_Washington_Carver	schema:about

Visual graph



What could we do with these triples?

```
16 <script type=text/javascript>
17   (function () {
18     window.addEventListener("load",
19       function sparqlQuery() {
20         var username = "<key>";
21         var password = "<password>";
22         var url = "https://rdf.ontotext.com/4134567067/Akerm_Bibframe_experiment/repositories/test1";
23         var method = "POST";
24         var str = [<"query goes here">].join("\n");
25         var dataString = "query=" + encodeURIComponent(str);
26         var xhr = new XMLHttpRequest();
27         xhr.open(method, url, true);
28         xhr.withCredentials = true;
29         xhr.setRequestHeader("Authorization", "Basic " + btoa(username + ":" + password));
30         xhr.setRequestHeader("Accept", "application/sparql-results+xml");
31         xhr.setRequestHeader("Content-Type", "application/x-www-form-urlencoded");
32         xhr.setRequestHeader("Content-Length", base64DecToArr(btoa(str)).buffer.byteLength);
33         xhr.send(dataString);
34         window.removeEventListener("load", sparqlQuery);
35       });
36   })();
37
38 </script>
```



Thanks for your attention! Now, what do YOU want to do?

BIBFRAME: <https://www.loc.gov/bibframe/>

GraphDB: <https://rdf.ontotext.com>

RDFlib (Javascript library):
<https://github.com/linkeddata/rdfliib.js>

IGELU/ELUNA Linked Open Data Working Group
<https://igelu.org/special-interests/lod>



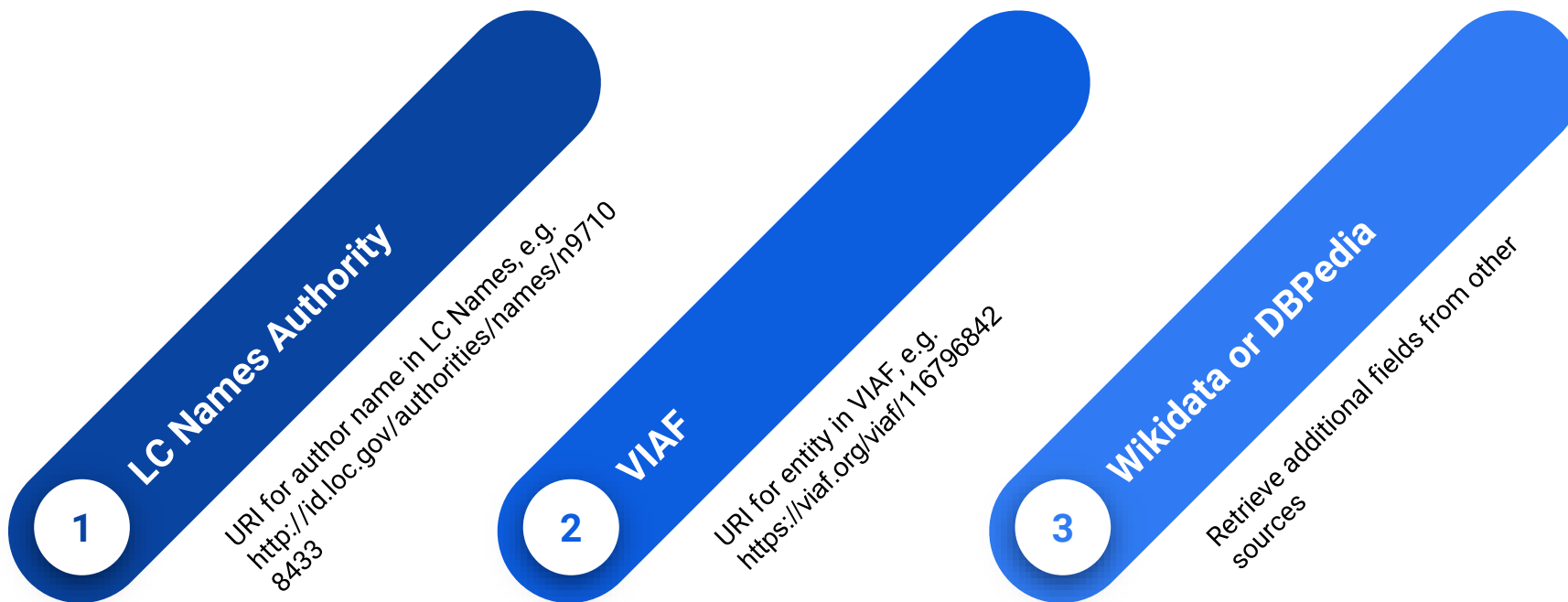
Laura Akerman
Discovery Systems
and Metadata
Librarian
Emory University
liblna@emory.edu



Author Cards made easy with LD



“Classic” Linked Data Scenario





UW Madison Tech Blog

Tech Blog



<http://bit.ly/2I3xQC8>

Using the Linked Data Integration API to Enhance Discovery

Steve Meyer on September 22nd, 2016

CODE & APPS

ALMA

At the University of Wisconsin-Madison we have been taking advantage of the Alma Linked Data integration API to enhance an experimental version of our local discovery system with info cards about some of the identities found in bibliographic records. See, for example, [Gertrude Stein on Picasso](#). The catalog will attempt to retrieve some brief biographical information about Pablo Picasso and Gertrude Stein for this record:

Pablo Picasso [Collapse](#)

Brief Biography

Description from Getty Linked Open Data: Long-lived and very influential Spanish artist, active in France. He dominated 20th-century European art. With Georges Braque, he is credited with inventing Cubism. (Sources: Grove Dictionary of Art online (1999-2002); LCNAF Library of Congress Name Authority File [n.d.])

<http://vocab.getty.edu/ulan/500009666>







UW Madison BibCard






- Encapsulates following of the links
- Hides issues such as:
 - Lack of outbound links from LC records
 - Resolving LC ID to VIAF








UW Madison BibCard


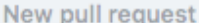
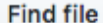

 **UW-Madison-Library** / **bibcard**


 Watch 7  Star 20  Fork 1





 Code  Issues 0  Pull requests 0  Projects 0  Insights

Library Linked Data for building knowledge cards.

 63 commits  1 branch  0 releases  3 contributors  MIT

Branch: master    

 **Steve Meyer** Update dependencies for rdf 3.0 Latest commit 8aaa011 on Jan 9

 bin	Empty gem	2 years ago
 lib	Updating wikidata alma maters queries	11 months ago
 spec	Updating wikidata alma maters queries	11 months ago
 .gitignore	Ignore a built gem	2 years ago

<https://github.com/UW-Madison-Library/bibcard>



So how hard would it be....

... to implement the “classic” LD scenario?


Toolbox:

- Alma JSON-LD API
- Madison BibCard
- Ruby script



Step 1: Alma JSON-LD API

- Activate the Linked Data integration profile
- Access the record URI ([sample](#)):
`https://open-na.hosted.exlibrisgroup.com/alma/{INST_CODE}/bibs/{MS_ID}.jsonld`



Step 2: Get BibCard

- To install the gem, follow the instructions at:
<https://github.com/UW-Madison-Library/bibcard>



Step 3- Short Ruby Script

- Retrieve JSON LD

```
RestClient.get(JSONLD_URL + ARGV[0])
```

- Create BibCard object

```
person = BibCard.person(lcnaf_uri)
```

- Populate ViewModel
- Write file

Simple as that....

Author



J. K. Rowling

Born: 31-Jul-1965

British novelist, author of the
Harry Potter series

[More info](#)



Resources & What's next

- Gist: <http://bit.ly/2HDpVw5>
- More use cases!
- More usage!
- Feedback!



Questions?