



University of Massachusetts Amherst

---

From the Selected Works of Jennifer Eustis

---

August 23, 2023

# Getting Started with LDLite for FOLIO Reporting

Jennifer Eustis, *University of Massachusetts Amherst*



This work is licensed under a [Creative Commons CC\\_BY-NC-SA International License](https://creativecommons.org/licenses/by-nc-sa/4.0/).



Available at: <https://works.bepress.com/jennifer-eustis/19/>

---

# Getting Started with LDLite for FOLIO Reporting

Jennifer Eustis, Moderated by Sharon Beltaine

---



# 1. Agenda

→ **LDLite**

What is LDLite and where can you find resources?

→ **Live Demo**

Just in case, screenshots have been included

→ **Questions**

# What is LDLite?

LDLite is a lightweight, open source reporting tool for Okapi-based services. It is part of the Library Data Platform project and provides basic LDP functions without requiring the server to be installed.

It is a way to provide access to FOLIO data for reporting and creating lists by pulling data from the FOLIO APIs into a postgres database.



# What you need to install LDLite?

This requires knowledge of Python and working with how Python creates a Postgres SQL database.

Here at the Five Colleges Consortium, one of our systems librarians installed and maintains LDLite on a server. Updates occur either nightly or on a weekly basis.



# What data can be found in LDLite?

If there is an API, then that data can find its way to LDLite.

You can download everything or data from one or a few selected FOLIO Apps.

There are no historical tables like in LDP1 or MetaDB.



# Pros and Cons to LD Lite?

## Pros

- Install & Maintain locally
- Download snippets of data
- Community of learners

## Cons

- No historical tables
- Locally installed and maintained
- No front end user interface



# Accessing The Data

You will need a database tool to set up a connection to the postgres database that LDLite creates. Here at the Five Colleges, we use [DBeaver](#). There is a free version and an enterprise version. You can see the difference between those features on DBeaver's website.





# Resources

Library Data Platform [Webpage](#)

Slack Workspace: [metadb-project.slack.com](https://metadb-project.slack.com)

GitHub: <https://github.com/library-data-platform/ldlite>

Queries from the Five Colleges:

<https://github.com/jenmawe/folio-ldlite>

<https://github.com/5-C-Folio/LDLite-Queries>



# Live Demo



## Fingers Crossed

The following slides have screenshots of the examples just in case LDLite isn't available. You'll be seeing the LDLite used by the Five Colleges Consortium.

```

10 --This query finds orphan instances or where there is an instance id and no holdings id.
2 --This query also picks up those instances marked for deletion that don't have all the fields filled out.
3 --Unlike the broader orphan instances, this filters for only those created after our migration date. Last ran 08-16-2023
4 SELECT
5     it.id AS instance_uuid,
6     it.hrid AS instance_hrid,
7     hrt.id AS holdings_uuid,
8     hrt.hrid AS holdings_hrid,
9     it.title,
10    sct.name AS statistical_code_name,
11    it.staff_suppress,
12    it.discovery_suppress,
13    it.metadata__created_date::date AS instance_record_created_date,
14    it.metadata__updated_date::date AS instance_record_updated_date,
15    ut.username AS last_user_to_edit_record,
16    split_part(ut.username, '@', 2) AS institution_email_part,
17    ut.personal__last_name,
18    ut.personal__first_name
19 FROM
20     inventory.instance__t it
21     left JOIN inventory.holdings_record__t hrt ON hrt.instance_id::uuid = it.id::uuid
22     LEFT JOIN inventory.instance__t__statistical_code_ids itsci ON itsci.id = it.id
23     LEFT JOIN inventory.statistical_code__t sct ON sct.id = itsci.statistical_code_ids
24     LEFT JOIN users.users__t ut ON ut.id = it.metadata__updated_by_user_id
25 WHERE
26     hrt.id IS NULL
27 AND
28     it.metadata__created_date > '2022-07-01'
29 ORDER BY
30     split_part(ut.username, '@', 2) ;

```

# Find Orphan Instances

What is an orphan instance? It is an instance connected to a marc srs record but not connected to a holdings or item record and that isn't suppressed from discovery and/or staff suppressed.

Issue: These appear in our discovery layer and are confusing to patrons.

This is a monthly query that we run to find these and do cleanup.



```

1 SELECT
2     DISTINCT it.id
3 FROM
4     folio_source_record.marctab m
5     LEFT JOIN inventory.instance__t it ON it.id::uuid = m.instance_id::uuid
6     LEFT JOIN inventory.holdings_record__t hrt ON hrt.instance_id::uuid = m.instance_id::uuid
7     LEFT JOIN inventory.location__t lt ON lt.id = hrt.permanent_location_id
8 WHERE
9     m.field = '583'
10 AND
11     m.CONTENT LIKE '%UMass copy: EAST commitmen%'
12 AND
13     it.discovery_suppress IS FALSE
14 AND
15     lt.name LIKE 'UM%';

```

Link to More LD Lite Queries:  
<https://github.com/5-C-Folio/LDlite-Queries>

```

SELECT
    DISTINCT sm.field,
    count(sm.field)
FROM
    folio_source_record.marctab sm
GROUP BY
    sm.field
ORDER BY
    sm.field;

```

# EAST & Other Queries

LDLite is a great tool to get lists of identifiers where you can then use that list to extract data from FOLIO. EAST is a good example to identify a set instance UUIDs and then use Data Export to extract the marc srs records.

Our systems librarian also has written scripts that provides a lightweight front end to getting lists or reports without having the person to use sql for example.





# Questions

If you think of something after WolfCon, join  
MetaDB Slack to continue the conversation.

