From MODS to OCLC through the WorldCat Metadata API

Shaun Akhtar, Dartmouth College Library

ALCTS Technical Services Workflow Efficiency Interest Group

ALA Midwinter 2016

@ShaunAkhtar #alamw16

Outline

Our scenario

WorldCat Metadata API overview

Why we wrote new code

Introducing `dcl_wc_metadata_api`

What we've learned

How to test it out

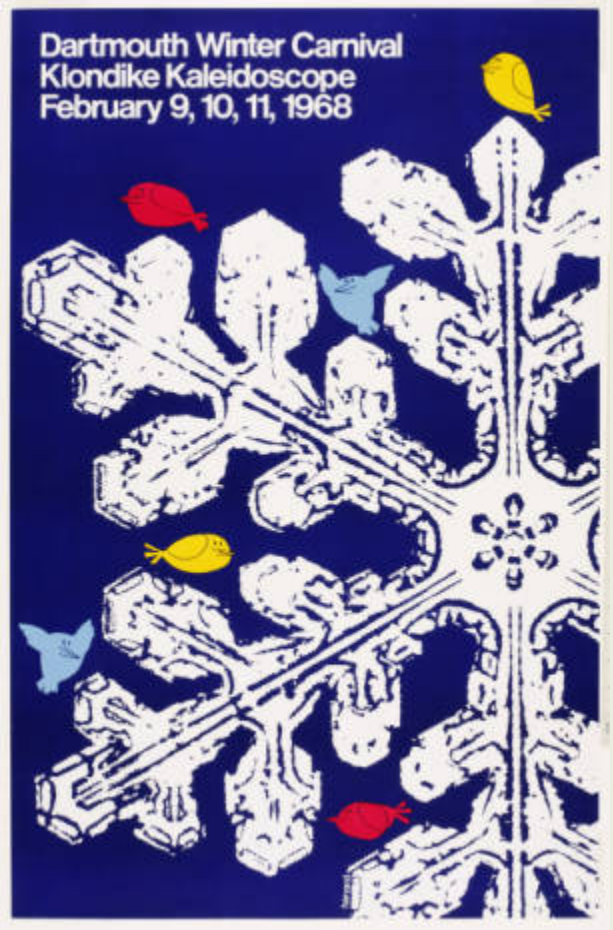
Starting Point

Metadata created in or transformed into MODS for local digital assets (unique digitized and born-digital material)

Includes new projects and retrospective conversion of metadata from previous projects

Many digital items and a lot of records, but limited item-level representation in WorldCat and few OCLC record numbers

Sample Collection: Dartmouth Winter Carnival Posters



Workflow Goals

Continue creating metadata in MODS

Ship records up to WorldCat and into local ILS

Merge OCLC numbers into MODS repository

Support iterative metadata enhancement

Minimize duplication of work

OCLC WorldCat Metadata API Overview

Debuted in June 2013

Enables direct access to WorldCat production data via HTTP

Expects and returns data serialized as MARCXML

Limited to institutions with WorldCat Discovery and OCLC Cataloging subscriptions

Largely handles actions one (record) at a time

Prior Work

Reese, Terry. "Opening the Door: A First Look at the OCLC WorldCat Metadata API." *Code4Lib Journal* 25 (2014). http://journal.code4lib.org/articles/9863

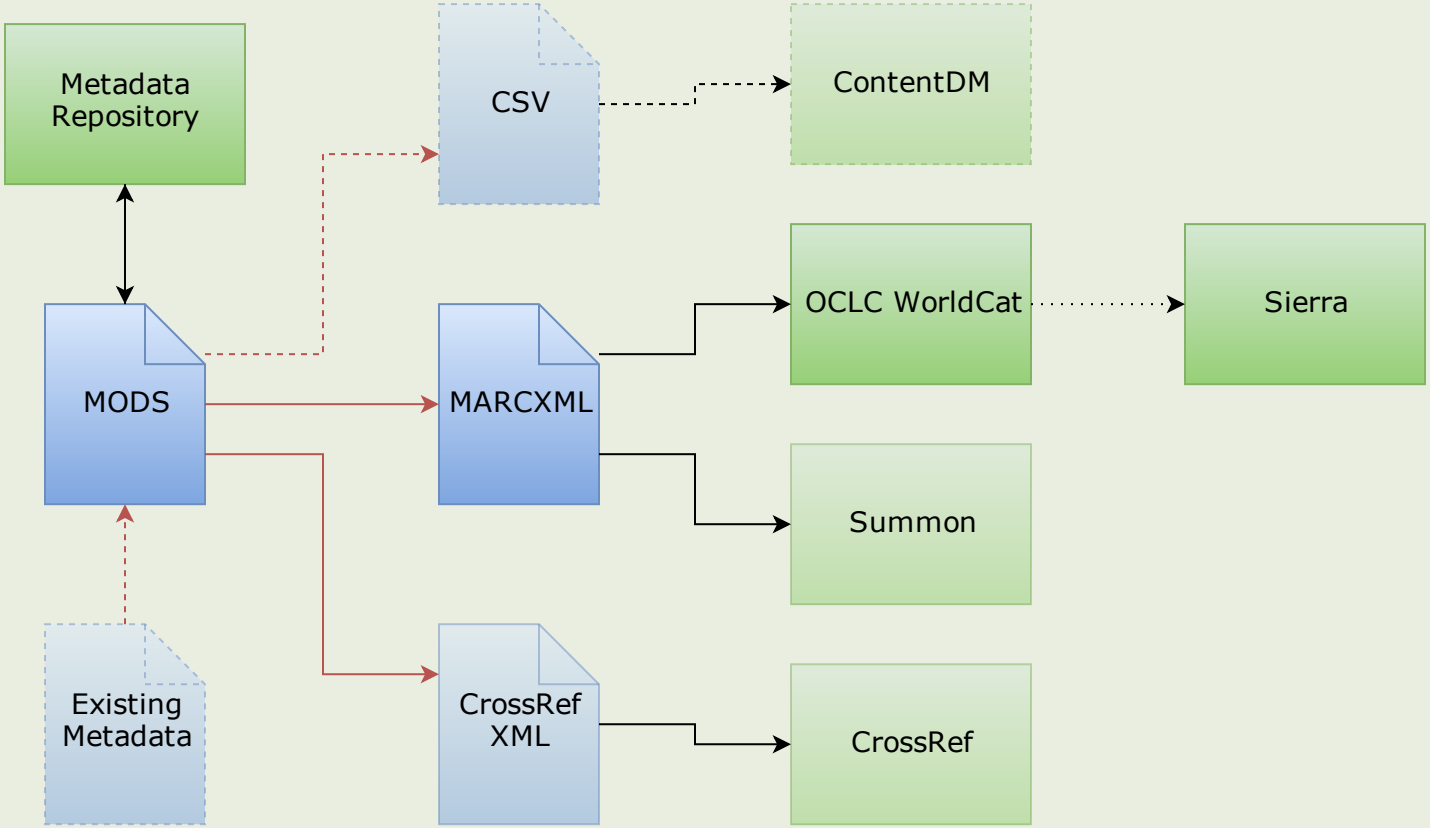
Johnston, Sarah. "Homegrown WorldCat Reclamation: Utilizing OCLC's WorldCat Metadata API to Reconcile Your Library's Holdings." *Code4Lib Journal* 27 (2015). http://journal.code4lib.org/articles/10328

OCLC WorldCat Metadata API Overview

An umbrella over actions on 4 types of WorldCat data:

- Bibliographic Resource (read, create, update)
- Holdings Resource (set, unset)
- Holding Codes (read)
- Local Bibliographic Data Resource (read, create, update, delete)

Ideal Workflow



Why a New Tool?

Digital workflows centered around XML, not binary MARC

Wanted to automate logging of data returned by OCLC

Why Ruby?

API wrapper already available (thanks to Terry Reese)

Preparing for a local Hydra/Fedora implementation

Development Goals

Use the API wrapper to batch up actions requested on multiple records

Programmatically retrieve records and/or error messages

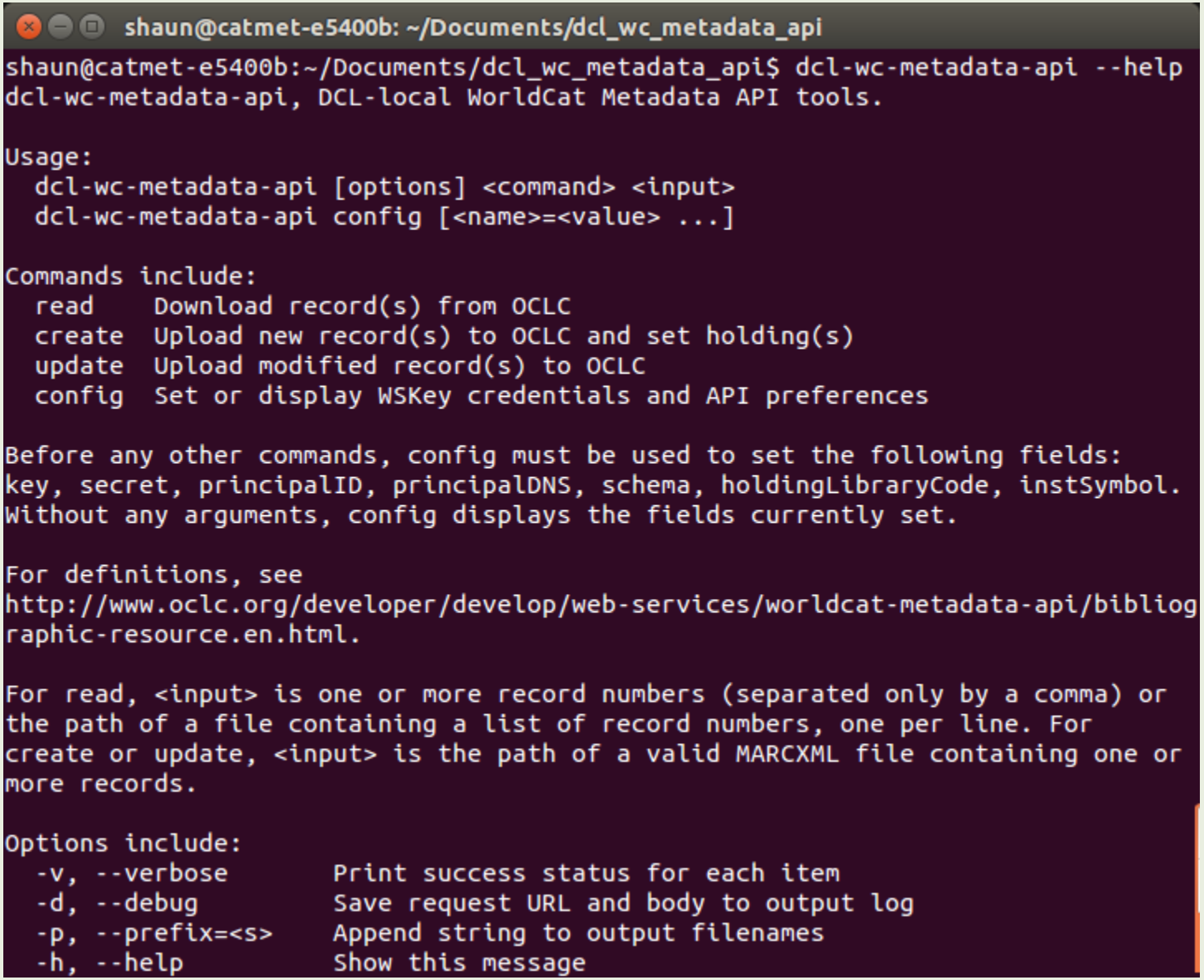
Have data ready for the next step in our workflow

Components

Terry Reese's `wc_metadata_api` wrapper (including OCLC's authentication scripts)

A command-line interface

A "manager" to queue the input requests, run through a batch of operations, aggregate results, and save the output


shaun@catmet-e5400b: ~/Documents/dcl_wc_metadata_api
shaun@catmet-e5400b:~/Documents/dcl_wc_metadata_api$ dcl-wc-metadata-api --help
dcl-wc-metadata-api, DCL-local WorldCat Metadata API tools.
Usage:
dcl-wc-metadata-api [options] <command> <input>
dcl-wc-metadata-api config [<name>=<value> ...]
Commands include:
read Download record(s) from OCLC
create Upload new record(s) to OCLC and set holding(s)
update Upload modified record(s) to OCLC
config Set or display WSKey credentials and API preferences
Before any other commands, config must be used to set the following fields:
key, secret, principalID, principalDNS, schema, holdingLibraryCode, instSymbol.
Without any arguments, config displays the fields currently set.
For definitions, see
http://www.oclc.org/developer/develop/web-services/worldcat-metadata-api/bibliog
raphic-resource.en.html.
For read, <input> is one or more record numbers (separated only by a comma) or
the path of a file containing a list of record numbers, one per line. For
create or update, <input> is the path of a valid MARCXML file containing one or
more records.
Options include:
-v, --verbose Print success status for each item
-d, --debug Save request URL and body to output log
-p, --prefix=<s> Append string to output filenames
-h, --help Show this message

Record Creation

```
# dwcposters-c078-marc.xml

<?xml version="1.0" encoding="UTF-8"?>
<marc:record xmlns:marc="http://www.loc.gov/MARC21/slim">
   <marc:leader>     nkm  22     Ki 4500</marc:leader>
   <marc:controlfield tag="007">cr||n|</marc:controlfield>
   <marc:controlfield tag="008">150120s1938    xx |||| ||||| o|||ineng d
</marc:controlfield>
   <marc:datafield tag="035" ind1=" " ind2=" ">
      <marc:subfield code="a">(DRB)dwcposters-c078</marc:subfield>
   </marc:datafield>
...
```

Record Creation

```
$ dcl-wc-metadata-api create ~/Desktop/dcl-ruby/input/dwcposters-c078-marc.xml

OCLC WorldCat Metadata API: Create operation
Created 1 record, 0 failed
Records written to wc-create-20150803155439.xml
Log written to wc-create-20150803155439-log.txt
```

Record Creation

```
# wc-create-20150803155439.xml

<?xml version="1.0"?>
<collection xmlns="http://www.loc.gov/MARC21/slim">
  <record>
        <leader>00000nkm a2200000Ki 4500</leader>
        <controlfield tag="001">ocn915392573</controlfield>
        <controlfield tag="003">OCoLC</controlfield>
        <controlfield tag="005">20150803155434.9</controlfield>
...
```

Record Creation

```
# wc-create-20150803155439-log.txt

RESULT(S)

(DRB)dwcposters-c078: created
915392573: holding set
```

Record Creation

```
$ shaun@catmet-e5400b:~/Documents/dcl_wc_metadata_api$ dcl-wc-metadata-api -v -p
"dwcposters" create ~/Desktop/dcl-ruby/input/marc-batch-2015110513520969.xml

(DRB)dwcposters-c012: created
(DRB)dwcposters-c013: created
(DRB)dwcposters-c014: created
(DRB)dwcposters-c019: failed
(DRB)dwcposters-c020: created
(DRB)dwcposters-c025: created
(DRB)dwcposters-c029: created
(DRB)dwcposters-c030: created
(DRB)dwcposters-c079: created
(DRB)dwcposters-c080: created
[...]
```

Log Files

```
RESULT(S)

(DRB)dwcposters-c019: failed
[...]
  <oclc:error>
    <oclc:code type="application">CAT-VALIDATION</oclc:code>
    <oclc:message>Record is invalid</oclc:message>
    <oclc:detail type="application/xml">
      <validationErrors xmlns="">
        <validationError type="variable field">
          <field occurrence="3" name="500"/>
          <message>Invalid character in position 1 in 1st $a in 3rd 500
          - data must be ALA characters.</message>
        </validationError>
      </validationErrors>
    </oclc:detail>
```

Discoveries

Creating a new master record with your holdings set ...

Requires 2 API operations, not 1

Setting holdings on a batch of records can be done ...

But only 50 at a time

Updating an existing record via the API ...

Requires an exact match on the 005 and 040$d

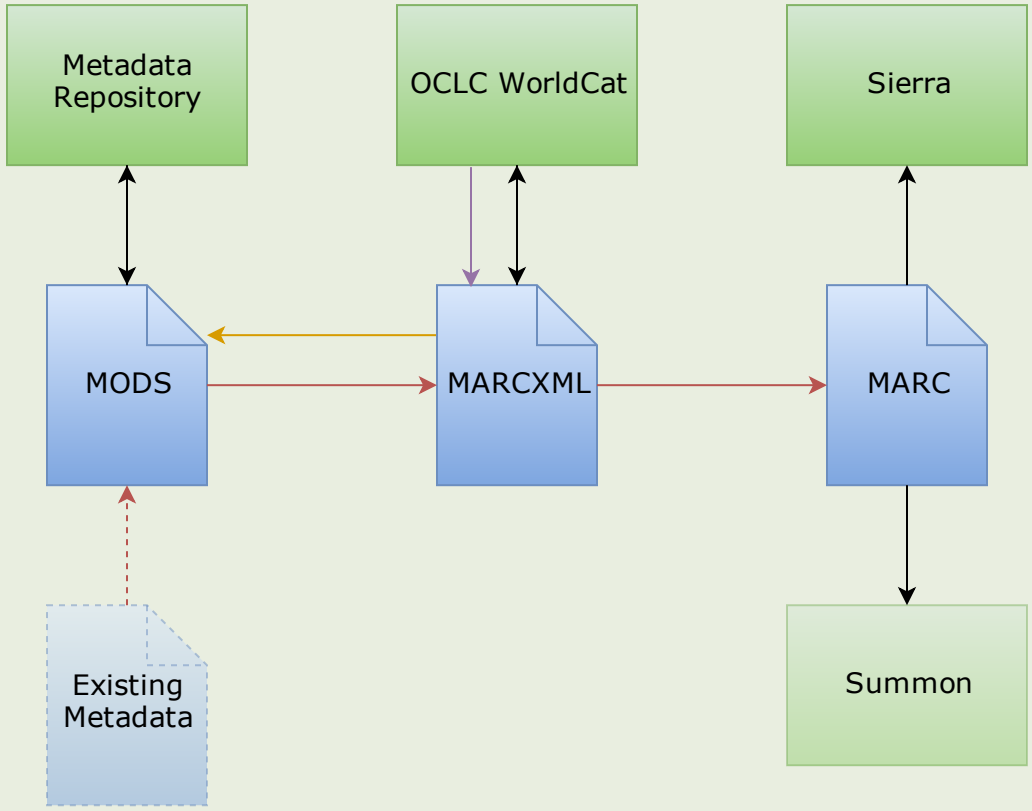
Workflow Additions

Necessary to pull records from WorldCat and check for changes before submitting any updates

Some WorldCat administrative metadata being stored locally in MODS repository to support resubmission

```
<recordContentSource authority="oclcorg">DRB</recordContentSource>
<recordInfoNote type="modifying agency">OCLCF</recordInfoNote>
<recordInfoNote type="modifying agency">DRB</recordInfoNote>
<recordCreationDate encoding="w3cdtf">2015-01-20</recordCreationDate>
<recordChangeDate encoding="w3cdtf">2015-11-05</recordChangeDate>
<recordChangeDate encoding="iso8601">20151015171111.1</recordChangeDate>
<recordOrigin>Record created through conversion from CONTENTdm XML.</recordOrigin>
<recordIdentifier source="DRB">dwcposters-c008</recordIdentifier>
<recordIdentifier source="OCoLC">ocn905625873</recordIdentifier>
```

Current Workflow



Installation

Code and documentation on GitHub

Registration with OCLC required to receive an API key

One configuration command needed to add your key, institutional symbol, etc.

Support

Written in Ruby 2.0.0 for use on 2.0+

Still under active development

Works on Linux ... Mac testers welcomed ... hoping to get it working on Windows soon

Next Steps

Continue developing code and documentation

Apply to new digital records

Explore applications for non-digital unique materials
(see how it fits alongside Connexion use)

Acknowledgments

Cecilia Tittemore

Bill Ghezzi and Barb Bushor

Carla Galarza

Terry Reese

Karen Coombs and the OCLC Developer Network

Links

Slides

https://akhtars.github.io/alamw16-presentation/

OCLC WorldCat Metadata API documentation

http://www.oclc.org/developer/develop/web-services/worldcat-metadata-api.en.html

Code on GitHub

https://github.com/akhtars/dcl_wc_metadata_api

Contact information

shaun.y.akhtar@dartmouth.edu

@ShaunAkhtar