

This QR code goes to my github which has this presentation, a related presentation I gave last year, and the normalization rules we are using at my institution.

You'll see images in this presentation primarily from our institutional repository, image credits in the speaker notes.

Image from: [http://library.nyam.org/colorourcollections/wp-content/uploads/sites/5/2018/02/WesleyanUniversity\\_ColorOurCollections\\_2018.pdf](http://library.nyam.org/colorourcollections/wp-content/uploads/sites/5/2018/02/WesleyanUniversity_ColorOurCollections_2018.pdf)

Olin Memorial Library, 1928, Vertical Files Photograph Collection, Special Collections & Archives, Wesleyan University

#### Presentation Abstract:

To ingest external data sources such as LibGuides, ArchivesSpace, and institutional repositories into Primo VE for discovery by your users, you will need to map and transform data as a part of that ingest process. To map and transform data, you need to use discovery normalization rules. In this session, learn to create and test discovery normalization rules for external data sources. Learn both Dublin Core and XML rule syntax, and why you might choose one over the other for your rules. See real-life examples of rules created for processing Islandora, LibGuides, ArchivesSpace, and Figshare data

sources. Prior to the conference, attendees will have the opportunity to submit their own examples of data transformations they would like to have addressed in the presentation.

**Ex Libris Product(s) related to this session:**

Primo VE

**Learning Outcome #1:**

Identify when to use Dublin Core and when to use XML normalization rules

**Learning Outcome #2**

Create and test both Dublin Core and XML normalization rules

**Learning Outcome #3**

Explain how discovery normalization rules relate to resource types

# This presentation is ...

About Discovery Normalization Rules  
for importing external data sources into  
Primo VE

Beginner to Intermediate level

Somewhat code-heavy

Records   Templates   **Rules**

Normalization (140)   Merge (20)   Brief level (6)

Normalization (Discovery) (9)

Private   5

Shared   4

ArchivesSpace Normalization (DC)  
DC

Islandora 2 DC XML to DC  
XML

Islandora MODS to DC  
XML

LibGuides Normalization (DC)  
DC

First, what this presentation is and is not  
It is...

About discovery normalization rules for external data sources

What does that mean?

They are rules for transforming the data from outside sources such as LibGuides, FigShare, Islandora, Digital Commons, ArchivesSpace  
Into Dublin Core data that Primo VE then uses for search, facets, and display. You write these rules to tell Primo how to treat the data from your external sources.

This is for beginner to intermediate users, so I'll go through setting up these normalization rules, how they fit into the import process overall, and then get into some specific rule examples and how to test your rules

It will be code-heavy when I get into the rule examples. If you aren't interested in writing rules yourself this presentation isn't right for you

# This presentation is **not** ...

Alma Normalization Rules for your MARC records

Setting up Discovery Import Profiles

The screenshot shows the Alma Rules interface. At the top, there are tabs for 'Records', 'Templates', and 'Rules'. The 'Rules' tab is selected. Below the tabs, there are two main sections: 'Normalization (140)' and 'Merge (20)'. Under 'Normalization (140)', there are four items listed: '755 field' (Drool), '755 Remove' (Drool), '902 Purchase - Alexander Street Press' (Drool), and '963 MARCIVCRDP' (Drool). A large red 'X' is drawn across the entire list of rules.

before we get started...

you might already be familiar with normalization rules for **MARC records in Alma**  
those are different from discovery normalization rules.

I'm not going to cover the entire process of setting up a Discovery Import Profile

I did cover that last year ...

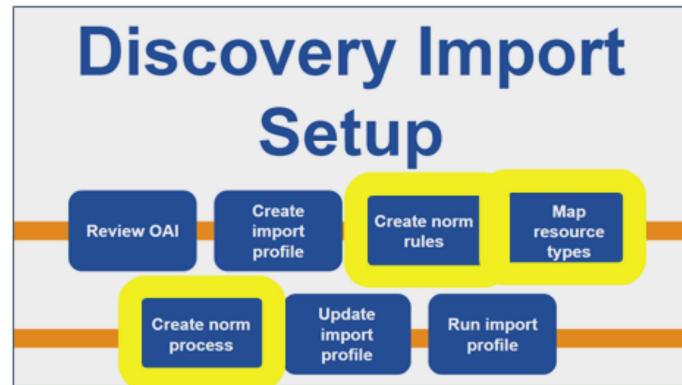
## Last year's topic ...

Stethers, Lori (2023) *Ingest Institutional Repository Content to Primo VE via OAI-PMH.*

<https://documents.el-una.org/id/eprint/2223/>  
<https://github.com/lstethers/primove-normrules/>



We're covering just the highlighted parts today



Last year's presentation that you can refer to to see the entire setup process for external discovery import is in the ELUNA document repository, also available in my github at the QR code I gave on the opening slide

Stethers, Lori (2023) [\*Ingest Institutional Repository Content to Primo VE via OAI-PMH.\*](#)  
and at <https://github.com/lstethers/primove-normrules/tree/main/presentation-files>

Today we'll be covering in more depth the 3 highlighted boxes, but not necessarily in that order

# Agenda



Here is what we'll be covering today

# Creating Discovery Normalization Rules



Image from: <https://digitalcollections.wesleyan.edu/object/upc-650>  
Scenic [College Row], 1959, University Photographer's Collection, Special  
Collections & Archives, Wesleyan University.

# Create New Normalization Rule (*animation*)

Alma > Config > Discovery >  
Loading External Data Sources >  
Normalization Rules for External Data Sources >  
New > Normalization (Discovery)

Access Level:  
Private or Shared  
  
Type:  
DC or XML

Animation of process:  
[https://drive.google.com/file/d/1Ve2UiNKunJsDCMIEBwK19Pak4k3Mm8Zq/view?usp=drive\\_link](https://drive.google.com/file/d/1Ve2UiNKunJsDCMIEBwK19Pak4k3Mm8Zq/view?usp=drive_link)

Here is how you get into the Normalization Rule editor to start creating norm rules.

Alma > Config > Discovery > Loading External Data Sources > Normalization Rules for External Data Sources > New > Normalization (Discovery)

While we're in here, I'll point out the Community section where you can see rules other institutions have shared.

There are a couple of choices you'll need to make while setting up the rule.  
I'll talk in a moment about whether to choose DC or XML rule type

Access Level: you can keep it private while you are writing and testing the rule, but you must choose Shared if you want to use this in a Normalization Process and Discovery Import Profile

A Private rule is only available to your specific user. Choosing Shared on the rule does not add it to the Community, it just makes it Shared your institution.

## Dublin Core or XML rules?

Dublin Core	XML
<ul style="list-style-type: none"><li>• default field mappings predefined</li><li>• limited evaluations and transformations</li><li>• simple and easy to understand</li></ul>	<ul style="list-style-type: none"><li>• no default field mappings defined</li><li>• with XPath expressions, many evaluations and transformations possible</li><li>• syntax can be challenging to understand</li><li>• helpful to learn some XPath</li></ul>

If you don't have DC data, you have to use XML norm rules.

If you do have DC data, you can choose whether to use DC or XML rules. There may be circumstances where you want to use XML norm rules on your DC data..

However you have to choose one or the other for the data source, you can't mix and match XML and DC rules for one data source.

DC – default field mappings defined so you only have to write rules where you want to do something different than the defaults vs. XML you have to specify every field you want to bring into Primo

You'll see later on an example where I needed to use XML rules to accomplish what I wanted even though I had DC data.

Documentation:

[https://knowledge.exlibrisgroup.com/Primo/Product\\_Documentation/020Primo\\_VE/Primo\\_VE\\_\(English\)/100Loading\\_Records\\_from\\_External\\_Sources\\_into\\_Primo\\_V](https://knowledge.exlibrisgroup.com/Primo/Product_Documentation/020Primo_VE/Primo_VE_(English)/100Loading_Records_from_External_Sources_into_Primo_V)

[E/Configuring Normalization Rules for External Resources \(Primo VE\)](#)[#Normalization Rule Syntax for DC and XML Formats](#)

## The parts of a rule (DC and XML)

```
// comment
/* multi-line comment */

rule "rule name"
when
    condition1 AND
    (condition2 OR
     condition3)
then
    action1
    action2
end
```

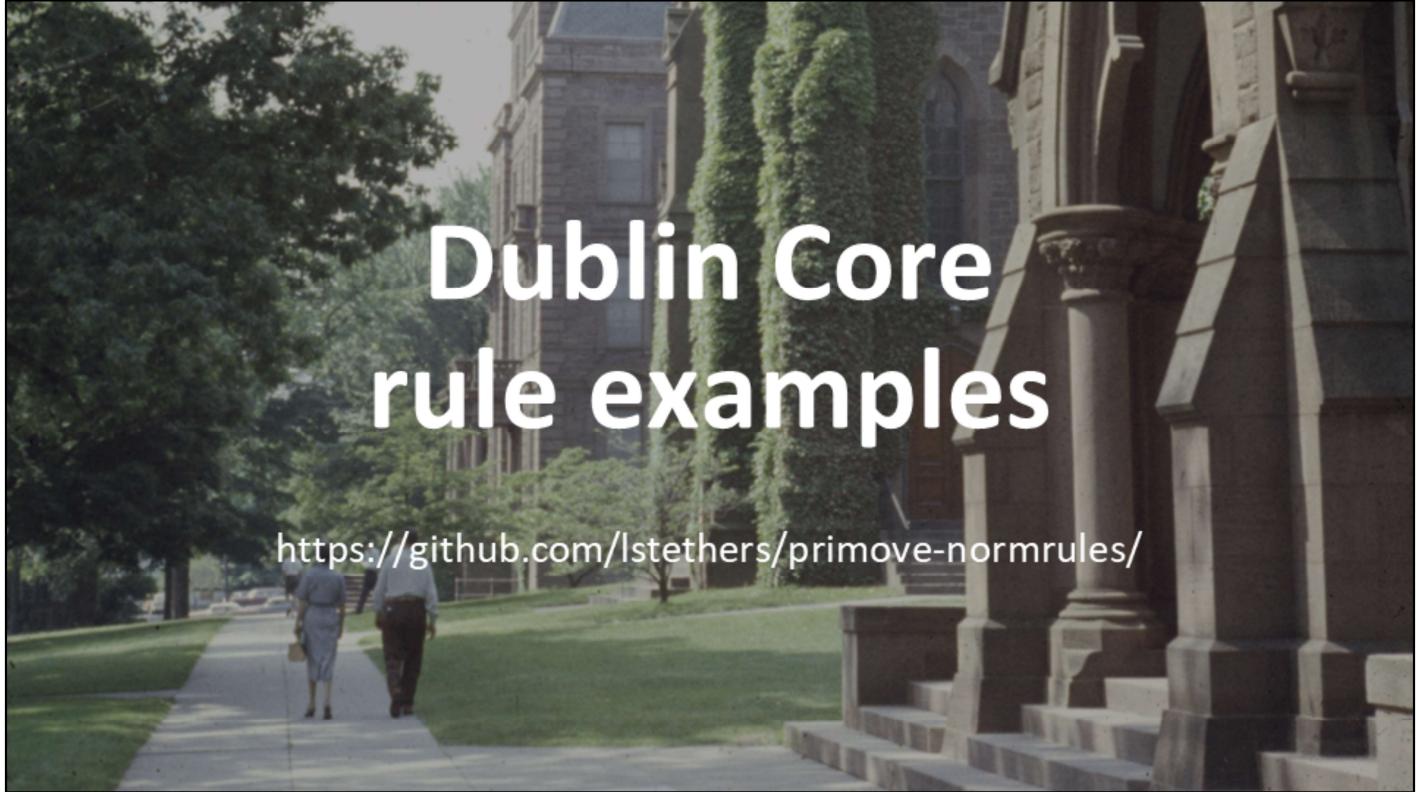


Documentation:

[https://knowledge.exlibrisgroup.com/Primo/Product Documentation/020Primo VE/Primo VE \(English\)/100Loading Records from External Sources into Primo VE/Configuring Normalization Rules for External Resources \(Primo VE\)#Normalization Rule Syntax for DC and XML Formats](https://knowledge.exlibrisgroup.com/Primo/Product_Documentation/020Primo_VE/Primo_VE_(English)/100Loading_Records_from_External_Sources_into_Primo_VE/Configuring_Normalization_Rules_for_External_Resources_(Primo_VE)#Normalization_Rule_Syntax_for_DC_and_XML_Formats)

Image from: <https://digitalcollections.wesleyan.edu/islandora/09001011-door-wall-garden-front>

Bacon, Henry. Door in wall with garden, front. 1886-1924. Henry Bacon Papers, Series VI: Personal, Subseries VI B: Sketches and watercolors of European scenes. Special Collections & Archives, Wesleyan University.



# Dublin Core rule examples

<https://github.com/lstethers/primove-normrules/>

I'm going to be showing code during this part. I'll be showing examples from portions of my normalization rule sets. To see the full rule set, see my github.

The full normalization rules that I'll be referencing are in github here:  
<https://github.com/lstethers/primove-normrules/>

Image from: <https://digitalcollections.wesleyan.edu/object/upc-650>  
Scenic [College Row], 1959, University Photographer's Collection, Special Collections & Archives, Wesleyan University.

Primo VE Display Field	DC Tags
Contributor	dcterms.contributor dc.contributor
Course Information	The course information (which includes Alma (not the Dublin Core source record)) <Course code>; <Course name>; <Course description> For more details, see <a href="#">Configuring Course Information</a>
Creation date	priority 1 - dc.date priority 2 - dcterms.date priority 3 - dcterms.created priority 4 - dcterms.issued priority 5 - dcterms.dateCopyrighted <div style="background-color: #ffffcc; padding: 5px; margin-top: 10px;"> <b>Note</b> Time-zoned dates (which         </div>

You might have some tags that don't get mapped by default so you should always double check.

[https://knowledge.exlibrisgroup.com/Primo/Product\\_Documentation/020Primo\\_VE/Primo\\_VE\\_\(English\)/120Other\\_Configurations/Mapping\\_to\\_the\\_Display%2C\\_Facets%2C\\_and\\_Search\\_Sections\\_in\\_the\\_Primo\\_VE\\_Record#Dublin\\_Core](https://knowledge.exlibrisgroup.com/Primo/Product_Documentation/020Primo_VE/Primo_VE_(English)/120Other_Configurations/Mapping_to_the_Display%2C_Facets%2C_and_Search_Sections_in_the_Primo_VE_Record#Dublin_Core)

## DC rule set example: LibGuides

```
rule "add OA indication"
when
    True
then
    set "Unrestricted online
access" in "dcterms"."accessRights"
end
rule "add research_guide resource type"
when
    True
then
    set "research_guide" in "dc"."type"
end
```

```
rule "add Research Guide to end of title"
when
    True
then
    set TEMP"1" to dc value "dc"."title"
    add suffix (TEMP"1"," Research
Guide")
    set "dc"."title" to TEMP"1"
end
```

With DC rules, you only need to specify  
the fields you want to transform.

Example of a DC rule set.

This is my full DC rule set for ingesting our LibGuides.  
I do very little to our LibGuides ingest, so there are just 3 rules in my set..  
The other fields are just mapped using the Ex Libris defaults, so I don't have to  
specify rules for them.

Now to get into the specifics of building rules...

## DC rule example: condition: “True” + action: “set” to constant value

```
// assign open access indicator on all records
```

```
rule "add OA indication"
```

```
when
```

```
    true
```

apply to every record

```
then
```

```
    set "Unrestricted online access" in  
        "dcterms"."accessRights"
```

set constant value in this  
field

```
end
```

On my LibGuides I add an OpenAccess indicator to all records.

breaking down what you see here:

- “when True” → so apply the rule to every record
- set a static value of “Unrestricted online access” in the dcterms.accessRights field

## DC rule example: condition: “exist” + action: “move”

// move value from dc.coverage to dc.subject

rule "move dc.coverage to dc.subject"

when

exist "dc"."coverage"

when this field exists

then

move "dc"."coverage" to "dc"."subject"

move it to dc.subject

end

This is something I'm doing on our ArchivesSpace records.

Our ArchivesSpace sends a value in dc.coverage, but we want it to end up in dc.subject for Primo.

Breaking down what you see here:

- when the dc.coverage field exists
- move its value to dc.subject

## DC rule example: condition: “not equals”

```
// change the publisher field to Wesleyan University for all existing  
publisher fields  
rule "override all dc.publisher values"
```

when

```
"dc"."publisher" not equals  
    "Wesleyan University"
```

when dc.publisher is not  
set to the value we want

then

```
set "Wesleyan University" in "dc":"publisher"
```

set it

end

This is something I'm doing on our ArchivesSpace records.

We want the publisher for all of our ArchivesSpace records in Primo to be “Wesleyan University”, but ArchivesSpace is sending something else for some records

breaking down what you see here:

- when dc.publisher is not equal to Wesleyan University
- set it to Wesleyan University

## DC rule example: actions: TEMP variable + a transformation

```
// append Research Guide to end of title  
// e.g. Political Science becomes Political  
Science Research Guide
```

```
set TEMP"1" to dc value "dc"."title"
```

put the dc.title value in  
TEMP1 variable

```
add suffix (TEMP"1"," Research Guide")
```

add suffix text to TEMP1

```
set "dc"."title" to TEMP"1"
```

set dc.title to the new  
value of TEMP1

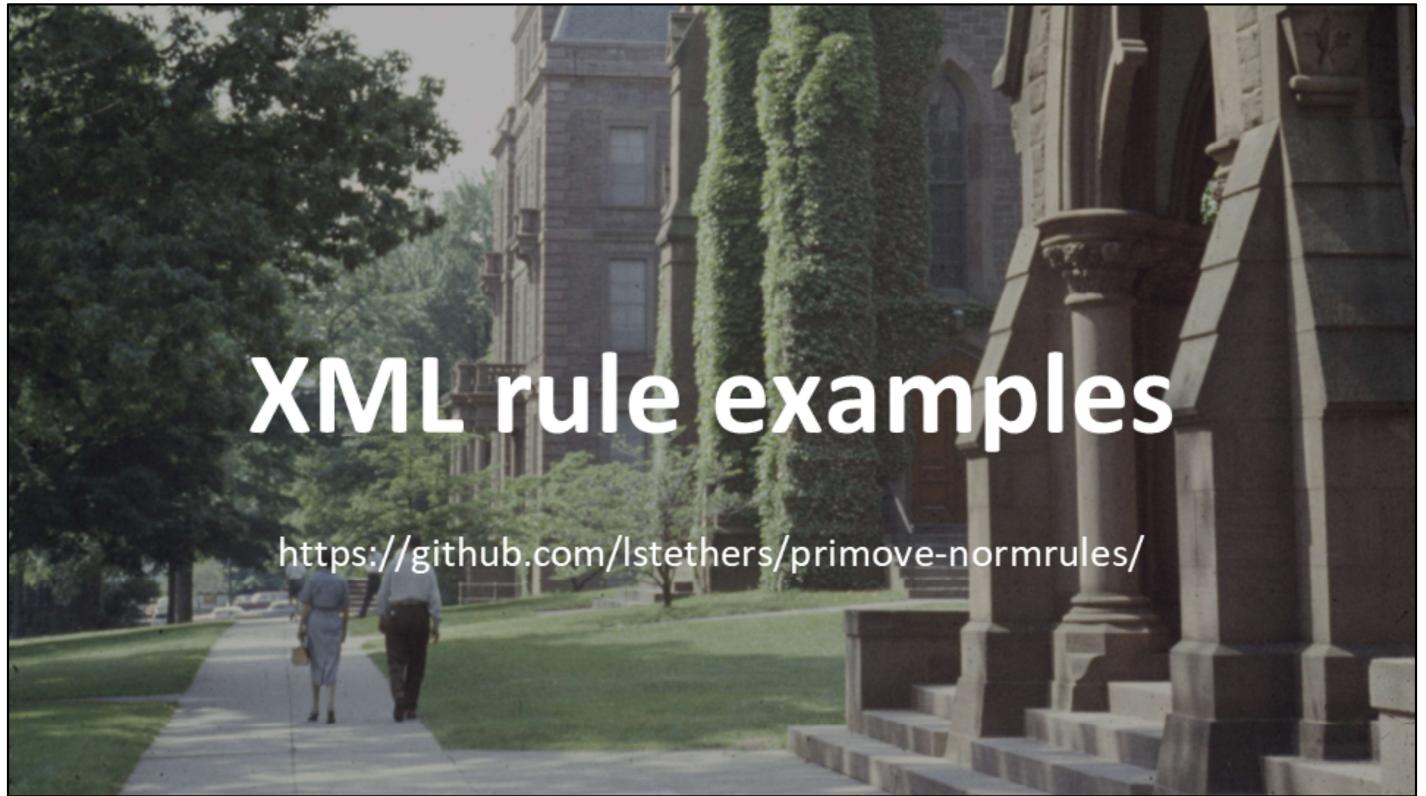
Here is another example from LibGuides.

I add “Research Guide” to the end of the title. For example, I want to change the title from “Political Science” to “Political Science Research Guide” to give it more context when it is discovered by a user in Primo.

To do this you need to use a TEMP variable.

For reasons I don’t know, you can’t do transformation actions on the dc fields themselves. you have to transfer their values to a temporary variable, operate on that, then transfer the resulting value back to a dc field.

1. create a TEMP1 variable and put the value of dc.title in it
2. add the text “ Research Guide” to the end of the title in the TEMP1 variable – using add suffix, which is a transformation available to us
3. set dc.title to the value of the TEMP1 variable



# XML rule examples

<https://github.com/lstethers/primove-normrules/>

A note on using XPath expressions: you can only use XPATH version 1.0 functions  
(<https://github.com/primousers/primo-ve-norm/wiki/3.-Known-Issue-with-Discovery-Import-Norm-Rules>)

Image from: <https://digitalcollections.wesleyan.edu/object/upc-650>  
Scenic [College Row], 1959, University Photographer's Collection, Special  
Collections & Archives, Wesleyan University.

## XML rule set example: Islandora 2

```
rule "Contributor"
when
    exist "/*[local-name()='contributor']/text()"
then
    copy "/*[local-name()='contributor']/text()" to
"dc"."contributor"
end
rule "Coverage"
when
    exist "/*[local-name()='coverage']/text()"
then
    copy "/*[local-name()='coverage']/text()" to
"dc"."coverage"
end
rule "Date"
when
    exist "/*[local-name()='date']/text()"
then
    copy "/*[local-name()='date']/text()" to "dc"."date"
end
```

```
rule "Format"
when
    exist "/*[local-name()='format']/text()"
then
    copy "/*[local-name()='format']/text()" to
"dc"."format"
end
rule "Main identifier (from record header)"
when
    exist "/*[local-name()='record']/*[local-
name()='header']/*[local-name()='identifier']"
```

*... and on and on and on*

With XML rules, all fields must be explicitly specified to appear in the Primo VE record.

Example of an XML rule set. This is just a small portion of this rule set.

With XML rules you are transforming the XML to DC fields for Primo.

You have to explicitly say which field you are moving each into because there are no default mappings for XML

\*Even\* if you are starting with DC data, if you are using XML rules, you must map each field.

Now to get into the specifics of some rules...

## XML rule example: condition: exist + action: copy

```
// move Contributor value to dc.contributor  
  
rule "Contributor"  
  
when  
    exist "//*[local-name()='contributor']/text()"  
  
then  
    copy "//*[local-name()='contributor']/text()"  
        to "dc"."contributor"  
  
end
```

wherever contributor  
field exists

copy it to dc.contributor

Here is a very basic XML example. Note that you have to use Xpath to get to the data here.

Our Islandora records have a dc.contributor field that we want to map to dc.contributor in Primo

breaking down what you see here:

- contributor field exists
- copy its value to dc.contributor

If there are multiple Contributor fields in the XML record, you only need to specify the rule once, it will act on every matching field

## XML rule example: XPath to use data from record header

```
rule "Main identifier (from record header)"

when
  exist "//*[local-name()='record']/*[local-name()='header']/*[local-
  name()='identifier']"

then
  copy "//*[local-name()='record']/*[local-name()='header']/*[local-
  name()='identifier']/." to "dc"."identifier"

end
```

When identifier field  
exists in header

copy it to dc.identifier

This is from our Islandora records.

I'm using a more specific XPATH to target just the identifier element that's in the record header,  
rather than any identifier field in the metadata portion of the record

## XML rule example: operators and parentheses

**when**

exist "//\*[local-name()='rights']/text()"

**AND**

wherever rights field exists

(exist "//\*[local-name()='rights'][(contains(text(),'No Copyright - United States'))]"

and contains either of these values

OR  
exist "//\*[local-name()='rights'][(contains(text(),'No Known Copyright'))]"")

set constant value in dcterms.accessrights

**then**

set "Unrestricted online access" in "dcterms"."accessRights"

I'm going to start showing you only the when... then portions from here on out to be able to show more on a page

For our Islandora records, we want to add an OpenAccess indicator only under certain circumstances, based upon the value in the "rights" field.

breaking down what you see here:

- when the 'rights' field exists
- and
- it contains either 'No Copyright - United States' or 'No Known Copyright' <- note that you can use parentheses to group the logic

in this case, setting the value in the dc field is the same as it was in DC rules

## XML rule example: XPath functions

when

exist "//\*[local-name()='subject'][**not[contains(.,'LH1.W45 A6')]**]"

whenever subject does  
not contain this text

then

copy "//\*[local-name()='subject'][**not[contains(.,'LH1.W45 A6')]**]" to  
"dc"."subject"

copy it to dc.subject

end

The records for our school newspaper have a subject identifier that has the newspaper's shelving call number in it.

We don't want that to appear in the digital records as a subject, so here we're copying all subjects EXCEPT those that contain the call number

this is an example of using some XPATH conditions in your rules – not and contains here are XPATH functions

## XML rule example: Xpath element occurrences

when

exist "//\*[local-name()='title'][1]/text()"

then

copy "//\*[local-name()='title'][1]/text()" to "dc.title"

end

copy the **first** occurrence  
of the title to dc.title

We want only the first occurrence of the title field to be placed in dc.title (the rest will go into dcterms.alternative, see next slide)

Here we are using Xpath notation to indicate we want only the first occurrence of the title element – the [1] after the title

## XML rule example: Xpath element occurrences + Xpath function

when

```
exist "//*[local-name()='title'][position()>1]/text()"
```

then

```
copy "//*[local-name()='title'][position()>1]/text()" to "dcterms"."alternative"
```

end

copy every occurrence after  
the first occurrence of title  
to dcterms.alternative

Now we'll assign the other occurrences of the title element.

Here we are using Xpath “position” function to get the element position, then we evaluate it to determine if it is greater than 1.

Title position > 1 — so any occurrence of the title after the first one will be copied to dcterms.alternative

## XML rule example: TEMP variable + a regular expression transformation

```
// remove parentheses and number from Creator  
// e.g. Jane Doe (123456) becomes Jane Doe
```

put the creator value  
in TEMP1 variable

```
set TEMP"1" to xpath "//*[local-name()='creator']/text()"
```

use 'remove  
substring' action with  
regex to remove  
matching text

```
remove substring using regex (TEMP"1","\\s*(\\d*)$")  
set TEMP"1" in "dc"."creator"
```

set dc.creator to the  
new value of TEMP1

This is an example for our Figshare repository.

The creator values come from Figshare with a number in parentheses after the person's name.

I want to remove the number and parentheses.

I'm using a normalization rule transformation ("remove substring using regex") with a regular expression to remove the space, parentheses, and number at the end of the string.

NOTE:

<https://github.com/primousers/primo-ve-norm/wiki/3.-Known-Issue-with-Discovery-Import-Norm-Rules> -- only certain regular expression escape sequences are supported

# Norm Rules and Resource Types



Image from: <https://digitalcollections.wesleyan.edu/object/upc-650>  
Scenic [College Row], 1959, University Photographer's Collection, Special  
Collections & Archives, Wesleyan University.

# DC Fields and resource types

1. discovery:resourceType
2. dcterms:type + Dublin Core Type to Discovery Type Mapping
3. dc:type + Dublin Core Type to Discovery Type Mapping

Dublin Core type to Discovery Type mapping

	DC source	Discovery type
1	academic addresses (documents)	archival_material_manuscript
2	accessions registers	archival_material_manuscript
3	administrative records	archival_material_manuscript
4	annual reports	reports
5	architectural reprographics (copies)	archival_material_manuscript
6	archival_material_manuscripts	archival_material_manuscript
7	articles	articles
8	articles of incorporation	archival_material_manuscript
9	autobiographies (literary works)	books
10	awards	archival_material_manuscript
11	bibliographies	reference_entries
12	biographies (literary works)	books
13	black-and-white slides	images
14	books	books
15	brochures	archival_material_manuscript

For external data sources, Primo VE chooses resource type in this order.  
So you can set any of these 3 values in your norm rules:

1. discovery:resourceType field
2. dcterms:type field – then looks it up in **Dublin Core Type to Discovery Type Mapping**
3. dc:type field then looks it up in **Dublin Core Type to Discovery Type Mapping**

If you use dcterms:type or dc:type make sure you also map that value in the mapping table

If no matching resource type found in the mapping table, uses the value set as default in the mapping table (this is how you end up with a bunch of stuff set as “Other”)

*Alma Config > Discovery > Loading External Data Sources > Dublin Core Type to Discovery Type Mapping*

Documentation:

[https://knowledge.exlibrisgroup.com/Primo/Product Documentation/020Primo VE/Primo VE \(English\)/100Loading Records from External Sources into Primo VE/Configuring Normalization Rules for External Resources \(Primo VE\)#Mapping Resource Types from External Data Sources](https://knowledge.exlibrisgroup.com/Primo/Product Documentation/020Primo VE/Primo VE (English)/100Loading Records from External Sources into Primo VE/Configuring Normalization Rules for External Resources (Primo VE)#Mapping Resource Types from External Data Sources)

[https://knowledge.exlibrisgroup.com/Primo/Product Documentation/020Primo VE/Primo VE \(English\)/100Loading Records from External Sources into Primo VE/Using the Mapping Table to Map Resource Types](https://knowledge.exlibrisgroup.com/Primo/Product Documentation/020Primo VE/Primo VE (English)/100Loading Records from External Sources into Primo VE/Using the Mapping Table to Map Resource Types)

## Set constant resource type

rule “set resource type of Archives”

```
when  
  true
```

```
then  
  set “archival_material_manuscripts”  
    in “dc”:“type”
```

```
end
```



Archivesspace records - I want to setting a static resource type value for all of the records.

Here you'd also need to have archival\_material\_manuscripts in the [Dublin Core Type to Discovery Type Mapping table](#)

Image from: <https://digitalcollections.wesleyan.edu/object/coeducation-13>  
Unidentified group photo of Wesleyan women. Pach Bros (January 01, 1890).  
Coeducation Collection, Box 2, Folder 3, Special Collections & Archives, Wesleyan University.

## Set variable resource type

rule “set resource type”

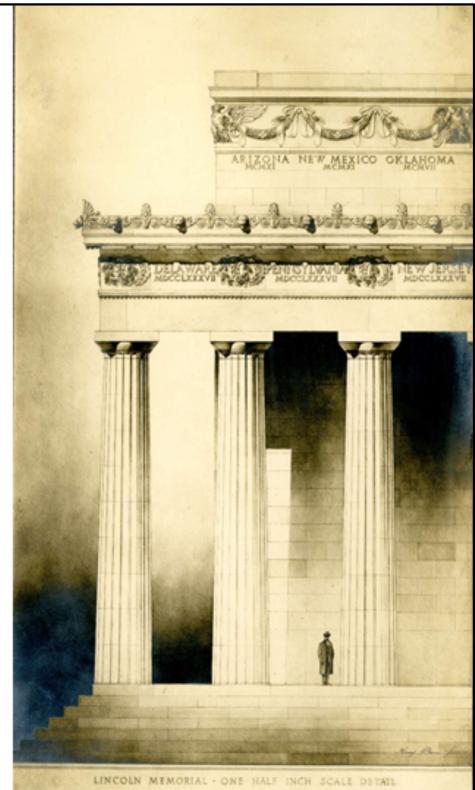
when

```
exist "//*[local-name()='type'][1]/text()"
```

then

```
copy "//*[local-name()='type'][1]/text()"  
to "dc"."type"
```

end



Here I want a variable resource type, based on the value in the type field.

I’m taking the value from “type” and putting into dc.type

then you need to have those values mapped in the Dublin Core Type to Discovery Type Mapping table

Image from:

<https://digitalcollections.wesleyan.edu/islandora/080070007-photographs-other-plans-delaware-corner>

Bacon, Henry. Photographs of other plans, Delaware corner. 1886 to 1924. Henry Bacon Papers, Series V, Lincoln Memorial, Subseries V D, Construction, Box 8, Special Collections & Archives, Wesleyan University.

# Using Discovery Normalization Rules



Image from: <https://digitalcollections.wesleyan.edu/object/upc-650>  
Scenic [College Row], 1959, University Photographer's Collection, Special  
Collections & Archives, Wesleyan University.

# Utilize the Normalization Rules in Discovery Import Profiles

The screenshot shows the 'Import Profile Details' screen for '01CTW\_WU\_LIBGUIDES'. The 'Normalization' tab is selected. A callout box labeled 'A Discovery Import Profile' points to this screen. Below it, another callout box labeled 'Uses a Normalization Process' points to the 'Process Details - Task Parameters' screen for 'DISCOVERY\_BIB\_MMS'. This screen shows the 'Task Parameters' tab selected. A callout box labeled 'Which contains Normalization Rule Set(s)' points to the right pane where normalization rules are listed. The right pane includes a search bar, a table with columns for 'Normalization (141)', 'Merge (20)', and 'Brief level (6)', and a detailed view of the 'LibGuides Normalization (DC)' rule set.

Normalization (141)	Merge (20)	Brief level (6)
Normalization (Discovery) (11)		
Search in list		
Islandora 2 DC XML to DC	XML	
Islandora MODS to DC	XML	
LibGuides Normalization (DC)	DC	

```
rule "add OA indication"
when
  True
then
  set "Unrestricted online access" in "dc"
end

// set all records to resource type of rese
rule "add research_guide resource type"
when
  True
then
  set "research_guide" in "dc" "type"
```

It's a few steps to get your normalization rules connected to your discovery import profile.

In the Discovery Import Profile the Normalization Tab uses a Normalization Process. The Normalization Process has tasks which are made up of your normalization rule set(s).

## Create Normalization Rule Process with Wizard (*animation*)

*Alma >Config > Discovery > Loading  
External Data Sources > Normalization  
Process Task > Add Process*

Animation of process:

[https://drive.google.com/file/d/1Ve-MaUbMDD6imSZIJtf2zMafA3TYUnrG/view?usp=drive\\_link](https://drive.google.com/file/d/1Ve-MaUbMDD6imSZIJtf2zMafA3TYUnrG/view?usp=drive_link)

A normalization rule process is a process that runs one or more normalization rule sets.

You need to create this process to be able to add the norm rules to your discovery import profile – you can't just add the rules directly.

Here is how to create a Normalization Rule Process.

Alma >Config > Discovery > Loading External Data Sources > Normalization Process Task > Add Process

Wizard to complete the rule process setup:

### 1. Process details

Business entity: Discovery bib records

Type: Choose Discovery DC normalization or Discovery generic XML normalization based upon whether you have created XML or DC rules

### 2. Process details - General Information

Name, description, and active/inactive

### 3. Process Details - Add Task

and empty screen! choose the Add task button on the right

You'll get only one option, which depends upon whether you've selected to create an XML or DC rule process

Whichever you are offered check the box and click Add.

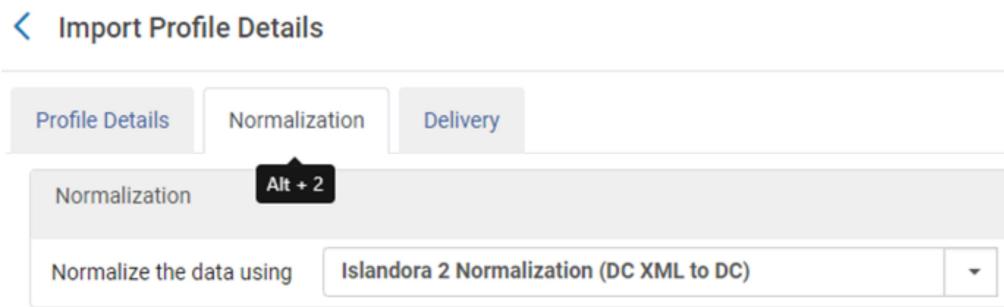
You'll need one task per norm rule, so if you've put them all into one rule set just add one task. If you've created multiple rule sets, add a task for each rule set.

#### 4. Process Details - Task Parameters

For each task that you added, use the drop-down to select a rule set for it.

## Add Norm Rule Process to your Discovery Import Profile

*Alma > Config > Discovery > Loading External Data Sources > Discovery Import Profiles > (choose your profile) > Normalization tab > choose your normalization process*



Here is how to add the normalization process to your Discovery Import Profile.  
Alma > Config > Discovery > Loading External Data Sources > Discovery Import Profiles > (choose your profile) > Normalization tab >  
Choose your process

# Testing Discovery Normalization Rules



Image from: <https://digitalcollections.wesleyan.edu/object/upc-650>  
Scenic [College Row], 1959, University Photographer's Collection, Special  
Collections & Archives, Wesleyan University.

## Some Methods of Testing Norm Rules

1. From the norm rules testing tool
2. Harvest subsets
3. View records in Primo



There are multiple ways to test

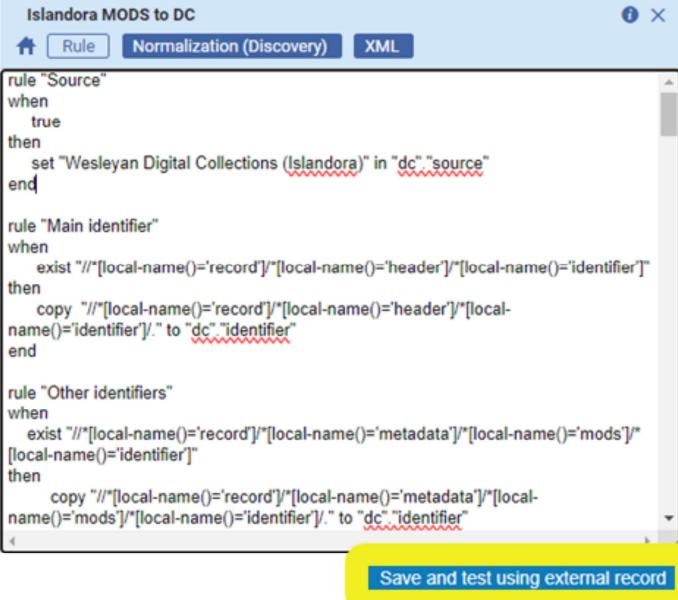
You can do all of this on a sandbox (doesn't have to be premium) to test harvesting using the normalization rules/processes you've defined before doing it on production.

You can also do all of this on production, but if you have access to a sandbox, always better to do it there!

Image from: <https://digitalcollections.wesleyan.edu/object/wes-covid19%3A57>

Eaton, Liza. Untitled. April 01, 2020. Wesleyan University collection on COVID-19, Special Collections & Archives, Wesleyan University.

## Testing: Norm Rules Testing Tool (*animation*)



The screenshot shows a software interface titled "Islandora MODS to DC". The top navigation bar includes "Rule", "Normalization (Discovery)", and "XML". The main content area displays three XSLT-like rules:

```
rule "Source"
when
  true
then
  set "Wesleyan Digital Collections (Islandora)" in "dc" "source"
end

rule "Main identifier"
when
  exist "//*[local-name()='record']/*[local-name()='header']/*[local-name()='identifier']"
then
  copy "//*[local-name()='record']/*[local-name()='header']/*[local-name()='identifier']/. to "dc" "identifier"
end

rule "Other identifiers"
when
  exist "//*[local-name()='record']/*[local-name()='metadata']/*[local-name()='mods']/*[local-name()='identifier']"
then
  copy "//*[local-name()='record']/*[local-name()='metadata']/*[local-name()='mods']/*[local-name()='identifier']/. to "dc" "identifier"
<
```

A yellow button at the bottom right of the code area contains the text "Save and test using external record".

Animation of process:

[https://drive.google.com/file/d/1VdWY4btEADgt-nf7AdgJqVDfvCdRMFrX/view?usp=drive\\_link](https://drive.google.com/file/d/1VdWY4btEADgt-nf7AdgJqVDfvCdRMFrX/view?usp=drive_link)

Using the Normalization Rule testing tool in Alma. This doesn't actually do any harvesting of records, just shows you how they would be altered

I need to have a text file that has the XML of the record I want to use in my test. I only need the parts between the <record> tags and I've just grabbed this from my OAI feed.

Animation...

To get to the testing tool, from the Discovery Config menu you can either choose Test Normalization Rule or you can go into the rule editor and choose Save and Test using external record.

In the testing page you'll select whether you are processing it as a DC or Generic

XML record, browse to the file you want to use – remember to hit upload! Then the record will appear in the Source Record box on the left.

Then select whether you want to run a norm rule or a norm rule process on it. If I'm testing a complex rule that I think is going to require a lot of testing and tweaking I'll isolate that to its own rule set and test just that one rule separately until I get it working how I want, then push into the rule set and/or norm rule process and test my rules together.

Click Run test and the output of the rules will appear in the Normalized Record box.

Issues that come up:

- you may see “no records were found” in the normalized record box. That means it doesn't like the format of your source record. It only needs the <record></record> portions, not all the OAI headers and stuff

See example record files from Wesleyan sources in the github:

<https://github.com/lstethers/primove-normrules/tree/main/presentation-files>

## Testing: Harvest subsets

If you don't want to test your rules on the whole set of data, you can limit by date or set name.

OAI Details

OAI Base URL *	https://digitalcollections.wesleyan.edu/oai/request				
Authentication	<input type="checkbox"/> Connect and Edit				
Repository Name	Islandora 8	Earliest Date Stamp	11/07/2022 12:15		
Granularity	YYYY-MM-DDThh:mm:ssZ	Admin EMail/s	@wesleyan.edu		
Metadata Prefix	oai_dc				
Set	All Sets				
Identifier Prefix					
Harvest Start Date	01/31/2024 10:43	X	Harvest End Date	02/09/2024 00:00	X

In the Discovery Import Profile, you can limit what records you harvest by sets (if your data source has sets) or by harvest start and end dates, perhaps just harvesting records modified in a particular day or week.

You can do this on the sandbox if you don't want to harvest the entire data set from your source, but just want to test a small number of records.

You could also do this to test a subset of records on production before harvesting everything.

Remember to reset your harvest start date or set name when you want to harvest everything!

## Testing: Repeating Harvests

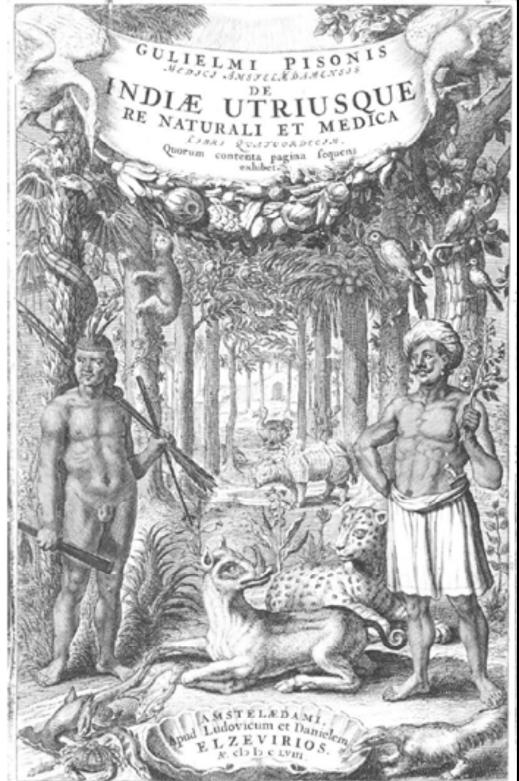
When repeating test harvests, you will need to:

Reset the harvest start date on your import profile

and

Either:

- Use the “Reload and Delete” option on the import profile
- Use Admin > Run a Job > Delete External Data Sources to delete the records, then run the import profile



If you run your discovery import OAI profile either in full or to harvest a subset, the next time you run a harvest it will only harvest records added or modified since the prior harvest date/time.

Therefore you need to reset your harvest date back to a prior date if you want to reharvest the same records or the entire set.

You should either use the import profile's Reload and Delete or the Delete External Data Sources job to delete the already harvested records from Primo.

Documentation for Delete External Data Sources job:

[https://knowledge.exlibrisgroup.com/Primo/Product\\_Documentation/020Primo\\_VE/Primo\\_VE\\_\(English\)/100Loading\\_Records\\_from\\_External\\_Sources\\_into\\_Primo\\_VE/Deleting\\_External\\_Data\\_Sources\\_for\\_Primo\\_VE](https://knowledge.exlibrisgroup.com/Primo/Product_Documentation/020Primo_VE/Primo_VE_(English)/100Loading_Records_from_External_Sources_into_Primo_VE/Deleting_External_Data_Sources_for_Primo_VE)

Image from: [http://library.nyam.org/colorourcollections/wp-content/uploads/sites/5/2018/02/WesleyanUniversity\\_ColorOurCollections\\_2018.pdf](http://library.nyam.org/colorourcollections/wp-content/uploads/sites/5/2018/02/WesleyanUniversity_ColorOurCollections_2018.pdf)  
Piso, Willem, 1611-1678. ... De Indiæ utriusque re naturali et medica libri quatuordecim ... Amsterdam: Elzevir, 1658. Special Collections & Archives, Wesleyan University.

# Testing: Primo records overall

Search for your identifier prefix and look for...

Facets:

- Resource types, esp “Other”
- Unexpected facet values

Records:

- Total # of records
- Records without titles
- Open access indicators

The screenshot shows the OneSearch interface with the search query "oai:digitalcollections.wesleyan.edu\*". The results page displays 28,232 results on page 1. On the left, there are facets for Refine My Results, Sort by (Relevance), Repeat My Search In (WorldCat, Google Scholar), Availability (Available Online, Open Access), Format (Theses and Dissertations, Images, Newspapers, Archival Material and Manuscripts, Articles), Subject, Publication/Creation Date, Author/Creator, and Language. The main area shows five search results, each with a thumbnail, title, author, and a "See online access" link. The results are numbered 1 through 5.

Rank	Title	Author	Date	Action
1	Viral-mediated Overexpression of Neuroligin2 in the Leads to Enhanced Synaptic Inhibition	Weiss, Eric Smith; Naegele, Janice R.	2016-04-15	See online access
2	Through Everchanging Tracks of Neverchanging Sp Joyce's Ulysses as a Spatial Practice	Trehaft-Ali, May Tina; Nascimento, Claudia Tatiane	2017-04-15	See online access
3	Sociolinguistic Identity and Change in Salvadoran S	Fleming, Andrew Peter; Neary, Louise	2020-04-15	See online access
4	More Coffins than Cradles: Low Fertility in Bologna	Teske, Erin; Nerenberg, Ellen	2007-05-01	See online access
5	From The Steps Of A Burning House: Blackness, Don			

If you wait for the records to be indexed and then search you can search for individual records, but I find it most useful to retrieve everything harvested from that data source and I do that by searching for the identifier prefix. you can see that in the example shown here.

Here I've searched for my identifier prefix to get all records from this data source.

I'm looking at the facets to check for:

“other” resource type – are there additional resource types that should be mapped?

weird subjects, authors, pub dates

I'm looking at search results to check for:

records without titles,

appearance of open access indicators,

correct resource types

## Testing: Primo full record details

- Data present, accurate, and formatted correctly
- Unwanted fields
- Extraneous data
- Online access link

Use the Staff View link to see the harvested and normalized DC record

DISSERTATION  
Viral-mediated Overexpression of Neuroligin2 in the Adult Hippocampus Leads to Enhanced Synaptic Inhibition  
Weiss, Eric Smith; Naegele, Janice R.  
2016-04-15  
[See online access >](#)

---

Online access  
[Online access](#)

---

Report a problem

---

Online access >  
[Staff View >](#)

---

Tools

E-MAIL    EXPORT TO EXCEL    PRINT    PERMALINK    CITATION    ENDNOTE RIS    ENDNOT

EXPORT BIBTEX    ADD TO COURSE RESERVE

---

Details

Title	Viral-mediated Overexpression of Neuroligin2 in the Adult Hippocampus Leads to Enhanced Synaptic Inhibition
Contributor	Weiss, Eric Smith > Naegele, Janice R. >
Publication/Creation Date	2016-04-15
Format	47 pages electronic

Are the data in the record details present? Are they formatted correctly? Are any of your expected fields missing?

Does the online access link work? (that's defined in the import profile, but you might be feeding it from a norm rule)

Are any unwanted fields or data showing up?

You can use the Staff View link to see the DC that Primo has for this record after normalization - really helpful for troubleshooting

## Testing: View records in Primo immediately (*animation*)

Get newly loaded records' MMS IDs from the Harvest log

Discovery Import Profile > Ellipses menu on your profile > History > Ellipses menu next to the job you just ran > Report > Report table > Ellipses next to New Records Added > Preview records

Use the MMS ID to build a URL to the Primo record

1 - 20 of 28,939		
MMS ID	Originating System ID	
1 9933223773503768	oai:digitalcollections.wesleyan.edu:node-39508	
2 9933223773403768	oai:digitalcollections.wesleyan.edu:node-39509	
3 9933223774303768	oai:digitalcollections.wesleyan.edu:node-39500	
4 9933223582603768	oai:digitalcollections.wesleyan.edu:node-26871	
5 9933223582703768	oai:digitalcollections.wesleyan.edu:node-26870	

Animation of process:

[https://drive.google.com/file/d/1Vd2n3y8ihZln6DQU4OAfc\\_qk0tkJimmU/view?usp=drive\\_link](https://drive.google.com/file/d/1Vd2n3y8ihZln6DQU4OAfc_qk0tkJimmU/view?usp=drive_link)

However if you want to look at your records right away, you can go to the

Discovery Import Profile > Ellipses menu on your profile > History > Ellipses menu next to the job you just ran > Report > Report table > Ellipses next to New Records Added > Preview records – it will show the MMS ID and originating system identifier for all the added records.

Then you can use the MMS ID to build a URL to get directly to the full record in Primo:

[https://onesearch.wesleyan.edu/discovery/fulldisplay?docid=alma9933223773503768&vid=01CTW\\_WU:CTWWU](https://onesearch.wesleyan.edu/discovery/fulldisplay?docid=alma9933223773503768&vid=01CTW_WU:CTWWU)

# Additional Resources



Image from: <https://digitalcollections.wesleyan.edu/object/upc-650>  
Scenic [College Row], 1959, University Photographer's Collection, Special  
Collections & Archives, Wesleyan University.

# Ex Libris Resources

- [Loading Records from External Sources into Primo VE](#)
- [Configuring Normalization Rules for External Resources \(Primo VE\)](#)
- [Mapping to the Display, Facets, and Search Sections in the Primo VE Record \(Dublin Core\)](#)



[https://knowledge.exlibrisgroup.com/Primo/Product\\_Documentation/020Primo\\_VE/Primo\\_VE\\_\(English\)/100Loading\\_Records\\_from\\_External\\_Sources\\_into\\_Primo\\_VE](https://knowledge.exlibrisgroup.com/Primo/Product_Documentation/020Primo_VE/Primo_VE_(English)/100Loading_Records_from_External_Sources_into_Primo_VE)

[https://knowledge.exlibrisgroup.com/Primo/Product\\_Documentation/020Primo\\_VE/Primo\\_VE\\_\(English\)/100Loading\\_Records\\_from\\_External\\_Sources\\_into\\_Primo\\_VE/Configuring\\_Normalization\\_Rules\\_for\\_External\\_Resources\\_\(Primo\\_VE\)](https://knowledge.exlibrisgroup.com/Primo/Product_Documentation/020Primo_VE/Primo_VE_(English)/100Loading_Records_from_External_Sources_into_Primo_VE/Configuring_Normalization_Rules_for_External_Resources_(Primo_VE))

[https://knowledge.exlibrisgroup.com/Primo/Product\\_Documentation/020Primo\\_VE/Primo\\_VE\\_\(English\)/120Other\\_Configurations/Mapping\\_to\\_the\\_Display%2C\\_Facets%2C\\_and\\_Search\\_Sections\\_in\\_the\\_Primo\\_VE\\_Record#Dublin\\_Core](https://knowledge.exlibrisgroup.com/Primo/Product_Documentation/020Primo_VE/Primo_VE_(English)/120Other_Configurations/Mapping_to_the_Display%2C_Facets%2C_and_Search_Sections_in_the_Primo_VE_Record#Dublin_Core)

Image from: <https://digitalcollections.wesleyan.edu/islandora/09001001-large-house-garden-front>

**Bacon, Henry. Large house with garden, front.** 1886-1924. Henry Bacon Papers, Series VI: Personal, Subseries VI B: Sketches and watercolors of European scenes, Special Collections & Archives, Wesleyan University.

## Community Resources

- [W3 schools XPath tutorial](#)
- [XPath 1.0](#)
- [Primo VE Norm Rule wiki](#) and associated [Rule Repository](#) on github
- [Community Knowledge Entry](#) with norm rules for transforming MODS
- [Mike Wan's github of norm rules](#)
- "Community" rules in the Norm Rule Editor



[https://www.w3schools.com/xml/xpath\\_intro.asp](https://www.w3schools.com/xml/xpath_intro.asp)  
<https://www.w3.org/TR/1999/REC-xpath-19991116/>

ELUNA Primo Working Group's Discovery Import Normalization Rules Project Team made the wiki and rule repository:

<https://github.com/primousers/primo-ve-norm/wiki/>

<https://github.com/primousers/primo-ve-norm>

[https://knowledge.exlibrisgroup.com/Alma/Community\\_Knowledge/Generic\\_XML\\_Normalization\\_Rules\\_and\\_XPath\\_Tips](https://knowledge.exlibrisgroup.com/Alma/Community_Knowledge/Generic_XML_Normalization_Rules_and_XPath_Tips)  
<https://github.com/mwan-work/PrimoVE-External-Data-Normalization/tree/main>

Image from: <https://digitalcollections.wesleyan.edu/islandora/08007002-photographs->

other-plans-east-elevation

Bacon, Henry. **Photographs of other  
plans, East elevation.** 1886-1924. Henry

Bacon Papers, Series V, Lincoln Memorial, Subseries V D, Construction, Box 8,  
Special Collections & Archives, Wesleyan  
University.



Image from: [http://library.nyam.org/colorourcollections/wp-content/uploads/sites/5/2018/02/WesleyanUniversity\\_ColorOurCollections\\_2018.pdf](http://library.nyam.org/colorourcollections/wp-content/uploads/sites/5/2018/02/WesleyanUniversity_ColorOurCollections_2018.pdf)

Olin Memorial Library, 1928, Vertical Files Photograph Collection, Special Collections & Archives, Wesleyan University



# Bonus Content

valid conditions for both DC and XML:

- **true** – Unconditionally applies the rule to all records.
- **exist** – Applies the rule when a specific field exists.
- **not exist** – Applies the rule when a specific field does not exist.
- **equals** – Applies the rule when the entire contents of the specified field match.
- **not equals** – Applies the rule when the entire contents of the specified field does not match.

DC Actions:

- **copy** – Copies the value from one field to another field.
- **move** – Moves the value from one field to another field.
- **remove** – Removes the value from a field.
- **set** – Sets the value of a field directly.

## XML Actions:

- **copy** – Copies the value from one field to another field.
- **copy all** - import more than one instance of a field (such as creator or ISSN) based on a specific separator. To utilize this action, you must create a normalization rule for each instance of the field that you want to support.
- **set** – Sets the value of a field directly.

## Transformations (XML and DC):

- **remove substring using regex**
- **concatenate with delimiter**
- **add prefix**
- **add suffix**
- **replace string by string**
- **return list using regex**
- **remove string**
- **remove leading and trailing spaces**
- **replace spaces**
- **lower case**