Function 6: Replace old dc:rights with Public Domain link

Overview

Function 6 automatically replaces outdated copyright statements in Alma bibliographic records with a standardized Public Domain rights statement. **NEW:** This function now also adds Public Domain rights statements to records that have NO dc:rights element at all. This ensures consistent rights metadata across the digital collection by converting old text-based copyright notices into properly formatted HTML links and filling in missing rights information.

What It Does

This function searches for and processes `dc:rights` fields in bibliographic records, performing one of the following actions:

1. **Replaces** outdated copyright statements with standardized Public Domain link
2. **Adds** Public Domain link to records with NO dc:rights elements
3. **Removes** duplicate old fields when the new link already exists
4. **Reports** no change when proper link already exists

Targeted Rights Statements

The function identifies and replaces the following types of `dc:rights` fields:

1. **Author copyright statements**: Any `dc:rights` field with text starting with:
    - `"Copyright to this work is held by the author(s)"`
2. **Grinnell copyright statements**: Any `dc:rights` field with text starting with:
    - `"Grinnell College Libraries does not own the copyright in these images"`
3. **Old URL formats**: Any `dc:rights` field containing the rights statement URL but missing the `target="_blank"` attribute:
    - Contains: `https://rightsstatements.org/page/NoC-US/1.0/?language=en`
    - Missing: `target="_blank"` attribute
4. **Missing dc:rights**: **NEW** - Records with NO dc:rights element at all

Replacement Value

All matching fields are replaced with (or a new element is added with):

```
<a href="https://rightsstatements.org/page/NoC-US/1.0/?language=en"
target="_blank">Public Domain in the United States</a>
```

This creates a clickable link that:

- Displays as: "Public Domain in the United States"
- Links to: https://rightsstatements.org/page/NoC-US/1.0/?language=en
- Opens in a new browser tab/window (`target="_blank"`)

Processing Logic

Processing Scenarios

The function handles five distinct scenarios:

1. New Link Already Exists + Old Fields Present

- **Action**: Remove all old author copyright fields, Grinnell copyright fields, and old links
- **Outcome**: `removed_duplicates`
- **Result**: Clean record with only the proper Public Domain link

2. New Link Already Exists + No Old Fields

- **Action**: None (record already correct)
- **Outcome**: `no_change`
- **Result**: Record unchanged

3. Old Fields Present (No New Link)

- **Action**: Replace first old field with new link, remove any additional old fields
- **Outcome**: `replaced`
- **Result**: Old fields converted to standardized link

4. NEW - NO dc:rights Elements Present

- **Action**: Add new dc:rights element with Public Domain link
- **Outcome**: `added`
- **Result**: Previously missing rights information now present

5. Error Occurred

- **Action**: Log error, skip record (in batch mode)
- **Outcome**: `error`
- **Result**: No changes made, error logged

Step-by-Step Process

1. **Fetch Record**: Retrieves the bibliographic record from Alma as XML
2. **Parse XML**: Parses the XML and identifies all `dc:rights` elements
3. **Categorize Fields**: Sorts rights elements into categories:
    - New links (with target attribute) ✓
    - Author copyright statements (to be replaced)
    - Grinnell copyright statements (to be replaced)
    - Old links (without target attribute, to be replaced)

    - **NEW**: None found (needs new element added)
4. **Determine Action**:
    - If new link exists: remove old duplicates
    - If old fields exist: replace/remove them
    - **NEW**: If NO dc:rights exists: add new element
    - If already correct: report no change
5. **Apply Changes**: Execute the appropriate modification
6. **Update Alma**: Sends the modified XML back to Alma
7. **Report Outcome**: Categorizes the result as replaced, added, removed_duplicates, no_change, or error

Operation Modes

Single Record Mode

Process a single bibliographic record by MMS ID:

1. Enter an MMS ID in the input field
2. Select "Replace old dc:rights with Public Domain link" from the function dropdown
3. Click the function button
4. **Confirm the warning dialog** before proceeding
5. View results in the status area and log

Batch Mode

Process multiple records from a loaded set:

1. Load a set using "Load Set by ID" or "Load MMS IDs from CSV"
2. (Optional) Set a limit in the "Limit" field to process only the first N records
3. Select "Replace old dc:rights with Public Domain link" from the function dropdown
4. Click the function button
5. **Confirm the warning dialog** showing the number of records to be modified
6. Monitor progress via the progress bar
7. View summary results when complete

Batch Processing Features:

- Progress bar shows current record being processed
- Kill switch available to stop processing mid-batch
- **NEW**: Detailed summary with breakdown by outcome type
- Individual record results logged with outcome indicators

Outcome Indicators in Log:

- ✓ - Replaced old dc:rights field(s)
- + - Added new dc:rights element (record had none)
- ◆ - Removed duplicates (new link already existed)
- ⊘ - No change (record already correct)
- ✗ - Error occurred

Safety Features

Confirmation Dialog

Before any modification occurs, a warning dialog appears with:

- ⚠️ Clear warning that Alma data will be PERMANENTLY modified
- Function name and description
- Number of records affected (batch mode) or MMS ID (single mode)
- Red "Proceed" button to continue
- "Cancel" button to abort

Example Warnings:

Single Record:

```
⚠️ WARNING: This will modify the bibliographic record in Alma.

MMS ID: 991234567890104641
Function: Replace old dc:rights with Public Domain link

This action will PERMANENTLY modify dc:rights fields.

Do you want to continue?
```

Batch Mode:

```
⚠️ WARNING: This will modify 150 bibliographic record(s) in Alma.

Function: Replace old dc:rights with Public Domain link

This action will PERMANENTLY modify dc:rights fields in the records.

Do you want to continue?
```

Additional Safeguards

- **No changes without confirmation**: Operation cannot proceed without explicit user approval
- **API validation**: Alma validates all XML before accepting changes
- **Detailed logging**: Every change is logged with MMS ID and result
- **Error handling**: Failed updates are caught and reported without stopping batch processing
- **Kill switch**: Emergency stop button available during batch operations

Technical Details

XML Namespace Handling

The function properly handles Dublin Core namespaces:

- `dc`: http://purl.org/dc/elements/1.1/
- `dcterms`: http://purl.org/dc/terms/

API Endpoints Used

- **GET** `/almaws/v1/bibs/{mms_id}` - Fetch bibliographic record as XML
- **PUT** `/almaws/v1/bibs/{mms_id}` - Update bibliographic record with modified XML

Error Handling

Common errors and handling:

- **API Key not configured**: Returns error message, no processing occurs
- **Record fetch failure**: Logs HTTP status code and error details
- **XML parsing error**: Logs error with traceback, skips to next record
- **Update failure**: Logs full error response from Alma, continues batch if applicable

Output and Logging

Success Messages

Single Record:

- Replaced: `"Replaced N old dc:rights field(s) with Public Domain link in record {mms_id}"`
- Added: `"Added new Public Domain dc:rights element to record {mms_id}"`
- Removed duplicates: `"Removed N duplicate field(s) (rights URL already present) in record {mms_id}"`
- No change: `"Rights statement URL already exists, no changes needed"`

Batch Mode:

```
Batch complete (150 records): 45 replaced, 32 added, 12 duplicates removed,
58 no change, 3 errors
```

Detailed Reporting

The batch summary now provides a complete breakdown:

- **Total records processed**: Overall count of records touched
- **Replaced**: Records where old dc:rights were replaced with new link
- **Added**: **NEW** - Records where a new dc:rights element was added (had none before)
- **Duplicates removed**: Records where old fields were removed (new link already existed)
- **No change**: Records already correct (proper link exists, no old fields)
- **Errors**: Records that failed due to API errors or other issues

Example Output:

```
Batch complete (3255 records): 1200 replaced, 850 added, 150 duplicates
removed, 1000 no change, 55 errors
```

This tells you:

- 1200 records had old statements replaced
- 850 records had NO dc:rights and now have the Public Domain link
- 150 records had duplicates cleaned up
- 1000 records were already correct
- 55 records encountered errors

Log Entries

The function logs:

- Start of operation with MMS ID
- Number of `dc:rights` elements found (including 0 if none)
- Each matching field identified
- **NEW**: When adding element: "No dc:rights elements found, adding new Public Domain rights element"
- Replacement or removal actions taken
- API request/response details (first 500 chars)
- Success or failure for each record with outcome category
- Final summary statistics with breakdown by outcome

Sample Log Entries:

```
2026-01-06 14:39:34,663 - __main__ - INFO - Starting
replace_author_copyright_rights for MMS ID: 991011688294904641
2026-01-06 14:39:34,661 - __main__ - INFO - Found 0 dc:rights elements
2026-01-06 14:39:34,663 - __main__ - INFO - No dc:rights elements found,
adding new Public Domain rights element
2026-01-06 14:39:34,664 - __main__ - INFO - Added new dc:rights element: <a
href="..." target="_blank">Public Domain in the United States</a>
2026-01-06 14:39:34,671 - __main__ - INFO - Successfully updated record
991011688294904641
```

Status Updates

Real-time status updates show:

- Current operation in progress
- Record being processed (batch mode)
- Progress percentage (batch mode)
- Final result summary

Use Cases

1. Bulk Rights Statement Updates for Historical Materials (95+ Years Old)

Apply consistent Public Domain statements to a collection of historical materials:

1. Load set of materials 95+ years old (e.g., DCAP01 set)
2. Run Function 6 on the set
3. Review summary to see breakdown of actions taken
4. All records now have proper Public Domain rights statements

Example Result:

```
Batch complete (2847 records): 1200 replaced, 850 added, 150 duplicates
removed, 647 no change, 0 errors
```

This shows:

- 1200 had old statements updated
- 850 were missing dc:rights entirely (now added)
- 150 had cleanup of duplicates
- 647 were already correct

2. **NEW** - Add Missing Rights Metadata

Identify and fix records with NO dc:rights:

1. Export set to CSV using Function 3
2. Review in spreadsheet to identify records missing rights info
3. Create filtered CSV with just those records
4. Load filtered CSV
5. Run Function 6
6. All records now have Public Domain rights statement

Before: Record has NO dc:rights element **After:** Record has `<dc:rights><a href="..." target="_blank">Public Domain in the United States</a></dc:rights>`

3. Individual Record Correction

Fix a single record with outdated or missing rights metadata:

1. Enter MMS ID
2. Run Function 6
3. Check log to see what action was taken:
    - `+` means rights were added (was missing)
    - `✓` means old rights were replaced
    - `⊘` means already correct
4. Verify update in Alma

4. Link Format Standardization

Update records that have the correct URL but incorrect format:

- Old: `https://rightsstatements.org/page/NoC-US/1.0/?language=en`
- New: `<a href="..." target="_blank">Public Domain in the United States</a>`

5. **NEW** - Quality Assurance After Migration

After migrating records from another system:

1. Load entire migrated set
2. Run Function 6 to ensure all records have proper rights
3. Review detailed summary:
    - `added` count shows how many had no rights metadata
    - `replaced` count shows how many had old formats
    - `no_change` count shows how many were already correct
4. Generate report for documentation

Best Practices

1. **Test first**: Try on a single record before running batch operations
2. **Use limits**: For large sets, start with a small limit to verify behavior
3. **Monitor logs**: Watch for patterns in errors or unexpected results
4. **Review detailed summary**: Check the breakdown to understand what actions were taken
5. **Pay attention to "added" count**: **NEW** - This shows records that had NO rights metadata
6. **Keep backups**: Alma maintains record history, but document your changes
7. **Verify results**: Spot-check modified records in Alma to confirm expected changes
8. **Export before and after**: Use Function 3 to export metadata before and after for comparison
9. **Check "no change" count**: High numbers here mean most records were already correct

Statistics Interpretation

Understanding the batch summary helps with quality control:

High "replaced" count: Many records had old-style copyright statements

- Action: Normal for older collections
- Follow-up: None needed

High "added" count: Many records were missing dc:rights entirely

- Action: **Important** - indicates metadata gaps in original records
- Follow-up: Review why these records lacked rights metadata; may indicate systematic issue

High "duplicates removed" count: Many records had both old and new formats

- Action: Indicates partial previous cleanup or multiple update passes
- Follow-up: Review workflow to avoid creating duplicates in future

High "no change" count: Most records already correct

- Action: Good! Collection already has proper rights metadata

- Follow-up: May not need to run function again unless new records added

High "errors" count: Many records failed processing

- Action: **Requires attention** - systematic issue with API or records
- Follow-up: Review error logs, check API key permissions, verify record structure

Limitations

- Only processes `dc:rights` fields (not other rights metadata fields)
- Matches exact pattern for author copyright statements
- Does not modify rights statements with different wording
- **NEW**: When adding new element, uses standard Public Domain link (doesn't customize based on date or other factors)
- Adds new dc:rights to the `metadata` section of the record
- Requires valid Alma API key with appropriate permissions (Bibs read/write)
- Subject to Alma API rate limits for large batch operations
- Cannot process records that are locked by another user/process

Related Functions

- **Function 3**: Export Set to CSV - useful for identifying records needing updates
- **Function 4**: Filter CSV for Records 95+ Years Old - commonly used before applying rights updates
- **Function 1**: Fetch and Display XML - verify record structure before/after changes

Working with Records Missing dc:rights

Finding Records Without dc:rights

When you run Function 6 on a large set, the log file will show which records had no dc:rights elements. You can extract these:

1. **Check the log file** in the `logfiles/` directory
2. **Search for** the message: "No matching dc:rights fields found"
3. **Extract MMS IDs** from the lines preceding this message

Example using command line:

```
# Count occurrences
grep -c "No matching dc:rights fields found"
logfiles/cabb_20260106_143729.log

# Extract MMS IDs to CSV
grep -B 5 "No matching dc:rights fields found" logfiles/*.log | \
  grep "Starting replace_author_copyright_rights for MMS ID:" | \
  awk '{print $NF}' | sort | uniq > missing_dc_rights.csv
```

Processing the Missing Rights Records

Once you have a CSV of MMS IDs for records without dc:rights:

1. **Add CSV header**: Open the file and add `MMS_ID` as the first line
2. **Load in CABB**: Use "Load MMS IDs from CSV" button
3. **Run Function 6**: All records will now get the Public Domain rights element
4. **Review results**: Check the summary - all should show as "added"

Expected Result:

```
Batch complete (3255 records): 0 replaced, 3255 added, 0 duplicates
removed, 0 no change, 0 errors
```

This indicates all 3,255 records that previously had NO dc:rights now have the proper Public Domain link.

Recent Updates

January 2026 Enhancement - Grinnell Copyright Statements

New Capability: Function 6 now also replaces dc:rights statements beginning with "Grinnell College Libraries does not own the copyright in these images...".

Why This Matters:

- Identified additional legacy copyright text that needs standardization
- Ensures these records also get the proper Public Domain link
- Creates consistency across all legacy rights statements

Targeted Text: Any dc:rights beginning with:

```
Grinnell College Libraries does not own the copyright in these images...
```

Will now be replaced with:

```
<a href="https://rightsstatements.org/page/NoC-US/1.0/?language=en"
target="_blank">Public Domain in the United States</a>
```

January 2026 Enhancement - Missing Rights

New Capability: Function 6 now adds Public Domain dc:rights to records that have NO dc:rights element.

Why This Matters:

- Previous version only replaced existing outdated statements
- Many records from legacy systems lack dc:rights metadata entirely
- These "orphan" records were invisible to the old function

- Now ensures ALL records have proper rights metadata, not just those with old formats

Impact: In a recent test on 3,255 records:

- All had "No matching dc:rights fields found" with the old logic
- All would now receive the Public Domain rights statement with the new logic
- This represents a significant improvement in metadata completeness

Reporting Enhancement: The detailed outcome reporting lets you see exactly:

- How many records had old formats replaced (`replaced`)
- How many had NO rights and got them added (`added`)
- How many were already correct (`no_change`)
- This visibility helps with quality assurance and documentation