

Subworkflow – Add WRLC Retention

This subworkflow applies a WRLC retention commitment to a specified item by setting the retention-related fields in Alma for a given barcode and institution.

At a glance

- **Status:** Inactive ([parent workflow](#) is Active)
 - **Applies consortium-wide?:** Yes
 - **Runs on:** Alma Institution Zones via Network routing
 - **Trigger:** Executed by another workflow (subworkflow only; no independent trigger)
 - **Primary outcome:** Marks an item as committed to WRLC retention in Alma.
 - **Who receives results:** No direct output; results are returned to the calling workflow.
-

Why this exists

WRLC retention commitments are recorded at the item level in Alma using structured retention fields. When an item is designated for WRLC retention, those fields must be set consistently across institutions.

This subworkflow standardizes how retention commitments are applied, ensuring that parent workflows can add WRLC retention flags in a consistent and reusable way without duplicating logic.

What it does

- Receives input from a [parent workflow](#).
 - Preserves the incoming barcode and institution code for reporting.
 - Routes the API call to the correct Alma Institution Zone.
 - Retrieves the item by barcode.
 - Sets WRLC retention-related fields.
 - Writes the updated item back to Alma.
 - Returns the updated item data to the calling workflow.
-

Where it runs

- **Alma IZ(s):**
 - Dynamically determined via `_institutionCode` passed from the parent workflow.
- **Systems touched:**

- Alma APIs (item retrieval and update)
 - Alma Network routing node
 - **Reports / queries used:** None
-

How it works

Logic overview

1. Trigger (Subworkflow Execution)

- Starts via **“When Executed by Another Workflow.”**
- Requires input fields:
 - Barcode
 - Institution Code

If these fields are missing, the workflow may fail or route incorrectly.

2. Preserve Input

The **“Stash barcode”** code node:

- Copies:
 - Barcode
 - Institution Code
- Stores them internally as:
 - `_inputBarcode`
 - `_institutionCode`

This ensures the calling workflow can reference the original input values for error handling or reporting.

3. Route to Correct Institution

The **Alma Network** node:

- Uses `_institutionCode` to dynamically select the appropriate Institution Zone.
- Does not target all network members.

⚠ If `_institutionCode` is incorrect or missing, the API call may fail or affect the wrong IZ.

4. Retrieve Item

- Calls Alma API:
 - `GET /almaws/v1/items`

- Uses the provided barcode.
 - Always outputs data (even on error), allowing the parent workflow to handle failures.
-

5. Apply WRLC Retention Fields

The **“Add WRLC Retention Code”** node updates:

- `item_data.committed_to_retain`
 - Sets `value = true`
 - Sets `desc = "Yes"`
- `item_data.retention_reason`
 - Sets `value = WRLCRetentionDONOTDELETE`
 - Sets `desc = "WRLC Retention"`
- `item_data.retention_note`
 - Sets to empty string

⚠ Important behaviors:

- This workflow **does not validate** whether the item is already marked as retained.
- It forcibly sets the retention-related fields.
- It assumes the Alma item structure includes:
 - `item_data.committed_to_retain`
 - `item_data.retention_reason`

If Alma’s API structure changes, this node may fail.

6. Update Item

- Calls Alma API:
 - `PUT /almaws/v1/bibs/{mms_id}/holdings/{holding_id}/items/{item_pid}`
 - Writes the modified item back to Alma.
-

If results exist

- The item is marked as committed to WRLC retention.
- Retention reason is set to **WRLCRetentionDONOTDELETE (WRLC Retention)**.
- Updated item JSON is returned to the parent workflow.
- No independent notifications are sent.

If no results

- If the barcode is invalid or retrieval fails:
 - The workflow still outputs data (due to `continueRegularOutput`).
 - Error handling must be managed by the parent workflow.
-

Artifacts produced

- Updated Alma item record with WRLC retention applied.
 - No sets, reports, spreadsheets, or email notifications.
-

Revision #2

Created 19 February 2026 20:22:50 by Jackie Saavedra

Updated 19 February 2026 20:31:52 by Jackie Saavedra