# Version 9.x

# Version 9.0 release notes

Rosetta 9.0 Release Notes.xlsx

## Announcements

- Refreshing DC configuration is necessary in order to support DC fields longer than 4K. Reinstalling Format Library will allow support for significant properties longer than 4K.
- Rosetta is now certified with RedHat 9

## Preservation

### Assign Multiple Classifications to Metadata Extractor

Users can associate a single metadata extractor/validator with several classifications (e.g. audio and video), eliminating duplicate plugin instances and simplifying maintenance.

#### What are the benefits of this feature?

Previously, the same tool had to be duplicated per classification (sometimes dozens of times). Now one plugin can serve multiple classifications, reducing operational overhead.

#### Prerequisites

- Access to Preservation > Format Library (Global or Local as applicable)
- Appropriate role to edit extractors/validators
- (Optional) Format Library Excel prepared for upload

#### Procedure

*Preservation > Format Library > Extractors (or Format Validators) > [row] > Update classification*

1. Open the extractor/validator list and choose **Update classification**.
2. In the dialog, check all relevant classifications (multi-select via checkboxes).
3. Save to apply. The tool becomes available to any format belonging to those classifications.

4. For each format, explicitly select the extractor/validator in the format edit page (availability doesn't imply auto-run).

Format-level selection logic:

- Associating an extractor with a classification exposes it as a selectable option to all formats in that classification.
- Admins still decide per format which extractor/validator applies

Excel Format Library import (bulk)

- Repeat the extractor name across multiple rows, each with a different classification
- Effect: On upload, the extractor/validator is registered under all listed classifications.

## Splitting Validation and Metadata Extraction - Updates

Control over splitting Metadata Extraction from Format Validation moved from the Validation Stack (task-chain checkbox) to each format's configuration. Admins decide for each format whether to:

- Run together using a combined plugin
- Split: run Metadata Extraction with plugin A and Format Validation with plugin B.

### What are the benefits of this feature?

The decision on whether different tools should be used for validation and extraction is now on the granular level of a file format. In addition, plugin management was improved to reduce maintenance of multiple services with single tools.

### Procedure

*Preservation > Format Library > Formats > Edit > [Format] > "Split metadata extraction from validation" (radio button) > pick Extractor and Validator*

1. Open the target **Format** and click **Edit**.
    - **No split:** Choose **Metadata Extractor; Format** Validator is disabled (combined flow).
    - **Yes, split:** Choose both **Metadata Extractor** and **Format Validator** (separate plugins).
2. In your **Validation Stack** (Task Chain), include tasks for **Technical Metadata Extraction** and **Format Validation**; execution will respect the per-format split setting. Order is configurable (e.g., **Fixity > Extraction > Validation**).

*Bulk configuration via Excel*

The **Format Library Excel** now includes a field to indicate whether a format is **split**. On upload, the format's radio button is automatically set accordingly.

---

**Note**

Customers can upload their **own** validators/extractors in the format library ZIP (JAR with XML descriptors). Once uploaded, they show under **Extractors > Format Validators** and become selectable per format.

---

**In addition**

A single JAR file can now expose multiple plugin types (e.g., both metadata extractor and format validator) by including multiple XML descriptor files inside the JAR.

**Under Configuration > Plugin Management > Add plugin instance**, you'll see multiple rows per JAR—one per XML descriptor—allowing installation of each type without duplicating the JAR.

## Fixity Job for AIP METS

The system supports **checksum generation and monitoring for METS files**, in addition to the existing fixity checks for content files stored in the permanent repository.

A new **METS Checksum** task has been introduced. This task operates at the **Intellectual Entity (IE)** level and ensures the integrity of the METS XML file.

This enhancement is additive and does not change existing file-level fixity behavior.

### What are the benefits of this feature?

While file-level fixity checks protect the integrity of preserved content, the **METS file is equally critical**: The feature extends integrity protection to the metadata layer, not just the binary files.

- Detects unintended changes or corruption in the METS XML that could compromise access, discovery, or preservation workflows.
- Aligns with digital preservation best practices by treating metadata as preservation-critical assets.

Checksum algorithms will be expanded at a later release to those supported for standard files.

### Procedure

1. A **METS Checksum** task is triggered for an Intellectual Entity.
2. For each version of the IE:
    - If no METS checksum exists:
        - The system generates an **MD5 checksum** for the METS file and stores it in the database.
    - If a checksum already exists:
        - A new checksum is generated and **compared to the previously stored value**.
3. If a mismatch is detected:
    - The discrepancy is recorded as an **error**.
    - Details are included in the **job report** for review and follow-up.

The task completes without modifying the METS file itself.

# Data Management

## Custom Thumbnail Support

Beyond the default “first file” thumbnail generation, admins can now create a dedicated **Representation** with **Usage Type = Thumbnail**, and upload the desired image(s). The system uses files in that representation as the official thumbnail for the IE. If unavailable or access-restricted, Rosetta **falls back** to the prior default.

### Procedure

**Search IE > Edit > Representations > Add Representation > Usage Type: Thumbnail > Upload file(s) > Done**

1. Open the IE and go to **Representations**.
2. **Add Representation** and set **Usage Type = Thumbnail** (previously only “View” was available).
3. Upload the image file(s) you want used as the thumbnail.
4. Save. The viewer now uses **this representation’s files** for the IE thumbnail.

If the thumbnail representation is missing, invalid, or access-rights prevent use, Rosetta **defaults** to generating from the **View** representation content (legacy behavior).

Preservation type (e.g., *Derivative* vs. preserved) **does not affect** thumbnail usage; usage type drives the behavior.

For bulk upload, the existing methods such as METS or CSV can be used for submitting a custom thumbnail per IE, according to the data model definitions above.

## Storage Integrity Job for S3 Protocol

The existing Storage Integrity job now supports S3 protocol in addition to NFS. This enhancement allows customers to verify file integrity across cloud-based or local S3 object storage, detecting missing and orphan files for S3 buckets using the same mechanism as NFS.

### What are the benefits of this feature?

- Enables integrity verification for cloud or local S3 storage environments.
- Reduces risk of undetected data loss or orphaned objects in S3 buckets.
- Provides consistent reporting across NFS and S3 using the same job framework.

### Prerequisites

- S3 storage configured in Rosetta as a permanent storage.
- AWS SDK (version noted in transcript: 1.x) installed and active in the environment.
- Appropriate permissions to run integrity jobs

### Procedure

**Preservation > Storage Integrity Job > Configure > Select Storage(s)**

1. Navigate to **Preservation > Storage Integrity Job.**
2. In the configuration screen, use the multi-select box to choose the storage(s):
    - NFS
    - S3 bucket(s)
3. Select the check type (Missing, Orphan) if applicable:
4. Start the job.
5. Review logs for results:
    - Missing files: expected but not found.
    - Orphan files: exist in storage but have no reference in Rosetta.
    - S3 paths include bucket and key; NFS paths show folder hierarchy.

## REST APIs - IE Update

Many REST APIs were introduced for updating IEs, allowing to displace SOAP-based applications

| **HTTP Method** | **Resource URL** | **Description** | **Parameters** |
|---|---|---|---|
| GET | rest/v0/ies/{ie_pid}/mets | Retrieves the METS of an IE | flags={flag_number} |
| POST | rest/v0/ies/{ie_pid} | Manage IE actions | op={action}&reason={reason} |
| GET | rest/v0/ies/{ie_pid}/md | Retrieves a list of IE MD's mids | type={type}&sub_type={sub_type}&limit=10&offset=0 |
| GET | rest/v0/ies/{ie_pid}/<br>representations/{rep_id}/md | Retrieves a list of Rep MD's mids | type={type}&sub_type={sub_type}&limit=10&offset=0 |
| GET | rest/v0/ies/{ie_pid}/representations/<br>{rep_id}/files/{file_pid}/md | Retrieves a list of File MD's mids | type={type}&sub_type={sub_type}&limit=10&offset=0 |
| **GET** | rest/v0/ies/{ie_pid}/md/{mid} | Retrieves the MD of an IE mid | |
| **GET** | rest/v0/ies/{ie_pid}/representations/{rep_id}/md/{mid} | Retrieves the MD of a Rep mid | |

| | | | |
|---|---|---|---|
| **GET** | rest/v0/ies/{ie_pid}/representations/{rep_id}/files/{file_pid}/md/{mid} | Retrieves the MD of a File mid | |
| **PUT** | rest/v0/ies/{ie_pid}/md/{mid} | Updates the MD of an IE mid | commit={true/false}&reason={reason} |
| **PUT** | rest/v0/ies/{ie_pid}/representations/{rep_id}/md/{mid} | Updates the MD of a Rep mid | commit={true/false}&reason={reason} |
| **PUT** | rest/v0/ies/{ie_pid}/representations/{rep_id}/files/{file_pid}/md/{mid} | Updates the MD of a File mid | commit={true/false}&reason={reason} |
| **DELETE** | rest/v0/ies/{ie_pid}/md/{mid} | Deletes the MD of an IE mid | commit={true/false}&reason={reason} |
| **DELETE** | rest/v0/ies/{ie_pid}/representations/{rep_id}/md/{mid} | Deletes the MD of a Rep mid | commit={true/false}&reason={reason} |
| **DELETE** | rest/v0/ies/{ie_pid}/representations/{rep_id}/files/{file_pid}/md/{mid} | Deletes the MD of a File mid | commit={true/false}&reason={reason} |
| **POST** | rest/v0/ies/{ie_pid}/md | Adds a MD to an IE | commit={true/false}&reason={reason} |
| **POST** | rest/v0/ies/{ie_pid}/representations/{rep_id}/md | Adds a MD to a Rep | commit={true/false}&reason={reason} |
| **POST** | rest/v0/ies/{ie_pid}/representations/{rep_id}/files/{file_pid}/md | Adds a MD to a File | commit={true/false}&reason={reason} |

### Handle Property Values Larger than 4k Characters

Support searching for property values longer than 4k characters.

---

**Note**

Refreshing DC configuration is necessary in order to support DC fields longer than 4K when the column is selected in the search results. Reinstalling Format Library will allow support for significant properties longer than 4K.

---

## Delivery

## Closed Captions in the Universal Viewer

Rosetta supports **storing and delivering closed captions (subtitles) for video content**.
This enhancement allows institutions to preserve caption files alongside video assets and deliver them to end users as part of the improved accessibility of the viewing experience.

Closed captions are managed as a dedicated representation associated with a video and are delivered seamlessly through the Universal Viewer.

*What is included*

- Support for WebVTT (.vtt) caption files
- Delivery through the Universal Viewer
- Support via ingest, UI, and APIs


### What are the benefits of this feature?

Closed captions are essential for:

- **Accessibility compliance** and inclusive access
- Supporting **multilingual audiences**

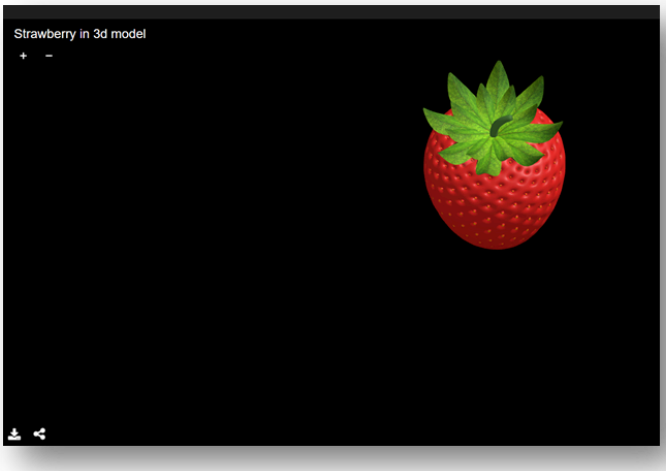
### Procedure

1. Closed captions are stored in Rosetta as a **separate representation linked to a video**.
2. One or more caption files (for example, in different languages) can be associated with a video.
3. When the video is delivered through the Universal Viewer, available closed captions are automatically offered to the end user within the viewer's native controls.
4. Captions can be added through ingest, manual workflows, or APIs, using the same processes already in place for other representations.

## 3D in Universal Viewer

The Universal Viewer now supports 3D formats .glb and .gltf.

Note that for .gltf, only self-contained files are supported, not textual files which are pointers to additional files.




## Infrastructure

### 3rd Party Upgrades

Rosetta is aligned with the latest 3rd party components in order to provide the most secure environment as possible. 9.0 upgrades include:

- Java 17.0.17
- Tomcat 9.0.111
- httpd 2.4.65
- Universal Viewer 4.2.1
- openssl 3.0.18