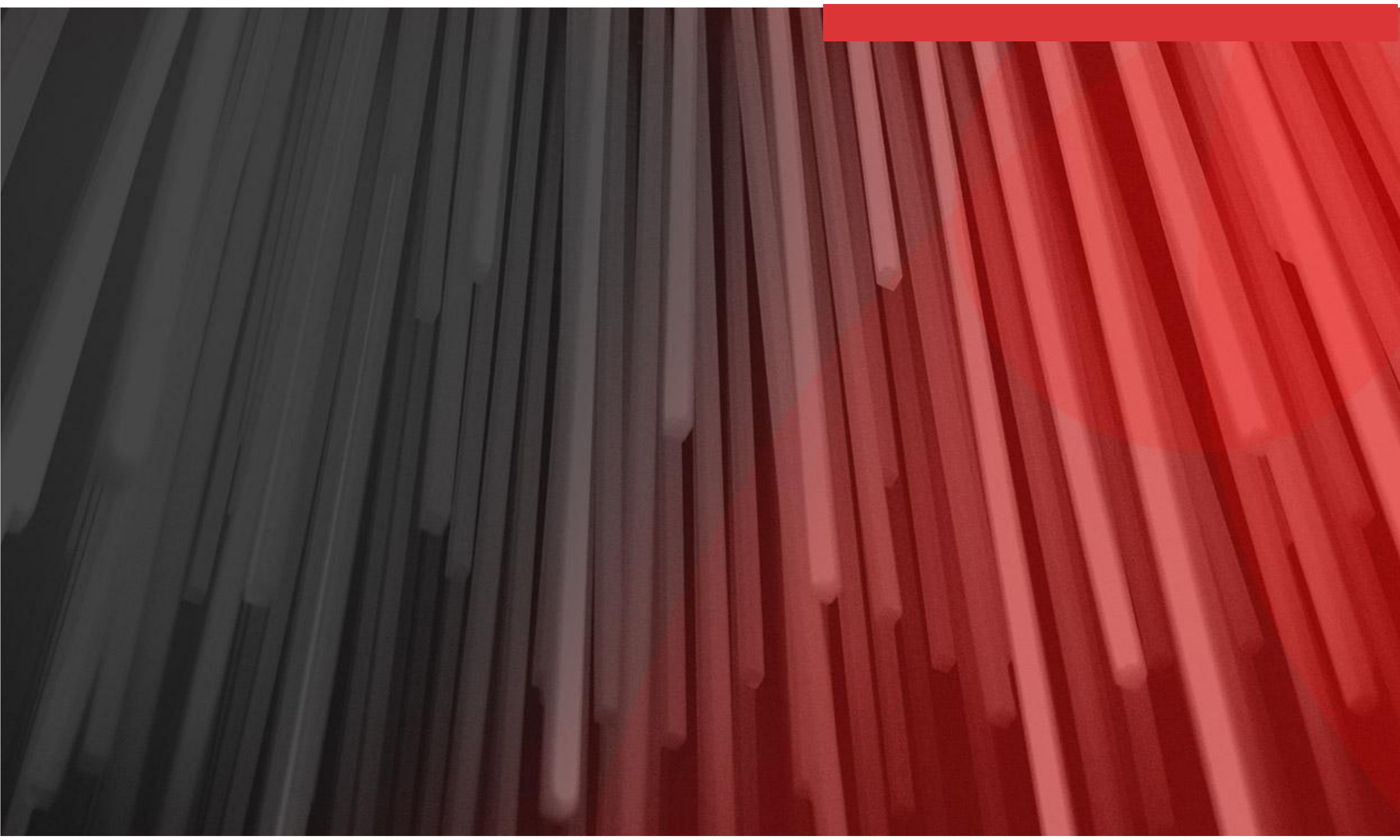
keep.
Preserving the future

OPERATIONS MANUAL



RODA ADMINISTRATIVE OPERATIONS FOR AIDA

OPERATIONS MANUAL

www.keep.pt

keep.
Preserving the future

ABOUT THE DOCUMENT

Identifier	MU221844		
Approved by	Luís Faria	Approved on	2022-01-30
Classification	Restricted		
Distribution	AIDA		

VERSIONS

#	Date	Author	Changes
1	2022-01-14	João Gomes	Initial document

SUMMARY

Individual archive institutions do not have sufficient knowledge, people and resources to tackle the long-term preservation of private law archives, which is why a number of them decided to join forces in the AIDA network. AIDA (Automatic Ingest Digital Archives[1]) is an agenda shared by nine institutions. This network of partners is aiming to take concrete steps, in the 2019-2023 policy period, to eliminate the most critical shortcomings for the sustainable long-term preservation of digital collections of private law archives.

In February 2021, resources were allocated to set up a pilot preservation system in the first phase of SCALA which will test the AIDA partners' shared preservation policy in practice.

RODA (Repository of Authentic Digital Records) is a long-term digital repository solution that delivers functionality for all the main functional units of the OAIS reference model. RODA is capable of ingesting, managing and providing access to various types of digital content produced by large corporations and public bodies.

This document describes the administrative operations developed for RODA in the context of the SCALA project for the AIDA network requirements.

[1] https://www.projectcest.be/wiki/Projecten:AIDA

TABLE OF CONTENTS

1 INTRODUCTION

AIDA (Automated Ingest Digital Archives) is a partnership of seven cultural heritage organizations and two advising partners, who have set up a roadmap for supporting and improving preservation of born-digital archives they take in. It concerns small to medium sized non-profit organizations, with limited resources and (in-house) IT support.

They all primarily manage private archives, involving increasingly also unordered born-digital archives, often stored on obsolete carriers, which cannot be processed immediately and urgently need to be stored in a secure environment. For these materials, the partnership developed a shared preservation strategy.

As part of their common roadmap, AIDA initiated the SCALA 1 - project which will develop a prototype for a shared preservation system that manages the ingest, storage and preservation procedures for these digital-born archives. AIDA obtained a grant from the Flemish Government to realize phase 1 of the project between May 2021 and April 2022.

2 ADMINISTRATIVE OPERATIONS

This section describes and documents the administrative operations developed for the digital preservation solution designed for the AIDA network requirements in the context of the SCALA project.

2.1 INSTITUTION REGISTRATION

To add an institution, the following items of information are required:

- Organization’s name;
- OR identifier;
- At least one email to receive notifications;
- The path of the SFTP share drop folder engine, provided by the server manager.

For each new institution it is necessary to create a group, a user for the ingestion and a new “fonds” intellectual entity in RODA.

Step 1: Create the group:

1. On RODA go to Administration menu and click on “Users and Groups”
2. On “Users and Groups” page, find the kebab menu next to the search button and then click "Add group"
3. Add a group with the organization name and add the permissions that are in RODA user types

Step 2: Create the ingest user:

1. On RODA go to Administration menu and click on “Users and Groups”;

2. On “Users and Groups” page, find the kebab menu next to the search button and then click "Add user";
3. Add a user with the following parameters:
   a. User name : ingest-<OR_ID>
   b. Full name: INGEST_<Organization_name>
   c. Email: ingest_<Organization_name>@scala.meemoo.be
   d. Groups: ingest

Step 3: Create a new “fonds” intellectual entity

1. On RODA go to Catalogue page;
2. Find the kebab menu next to the search button and then click “Create intellectual entity”;
3. Add a new intellectual entity with the following parameters:
   a. Type: Dublin Core
   b. Metadata, Title: <Organization_name>
   c. Metadata, Type: Fonds

Step 4: After creating the “fonds” intellectual entity, it is necessary to associate the group and the ingestion user to that fonds:

1. On RODA go to Catalogue page and find the new “fonds” intellectual entity;
2. Select the entity, find the kebab menu next to the search button and then click “Permissions”;
3. Click on “ADD PERMISSION” action;
4. Search for the group created for this institution and click on “SELECT” button;
5. Do the same step above for the ingest user;
6. For assigned groups permission select only the “READ” option;
7. For assigned users permission select only the “Create” option for ingest user;
8. Click on the “APPLY TO HIERARCHY” action.

It is highly recommended that the following actions be done by a specialist in implementations with containers in a linux environment

Step 5: For the configuration of the dropfolder mechanism, a file on the server must be modified and the service must be restarted for the actions to take effect.

1. Find the docker-compose.yml file on:
   /roda/data/git/roda-aida/01-code/deploys/production
2. Increment the env `RODA_DROP_FOLDERS_QTY` and add the following environment vars to roda service:

```
RODA_MONITOR_DROP_USER_N=<INGEST_USER_NAME>
RODA_MONITOR_DROP_PASSWORD_N=<INGEST_USER_PASS>
RODA_DROP_FOLDER_N=/roda/data/sftp/<OR_ID>/incoming/
RODA_DROP_INGEST_PLUGIN_N=org.roda.core.plugins.external.aida.AIDAIngestPlugin
'RODA_DROP_INGEST_PLUGIN_PARAMETERS_N={
        "parameter.parent_id":"<FOUND_ID>",
 "parameter.sip_to_aip_class":"org.roda.core.plugins.plugins.ingest.EARKSIP2ToAIPPlugin",
        "parameter.or_identifier":"<OR_ID>",
        "parameter.email_notification":"<email>",
        "parameter.can_be_auto_submitted":"false",
        "parameter.do_auto_accept":"false",
        "parameter.do_aip_prune": "false"
 }'
```

Where N is the number following the last configured institution

3. Restart the service so the new configuration take effect

   `docker-compose up -d`

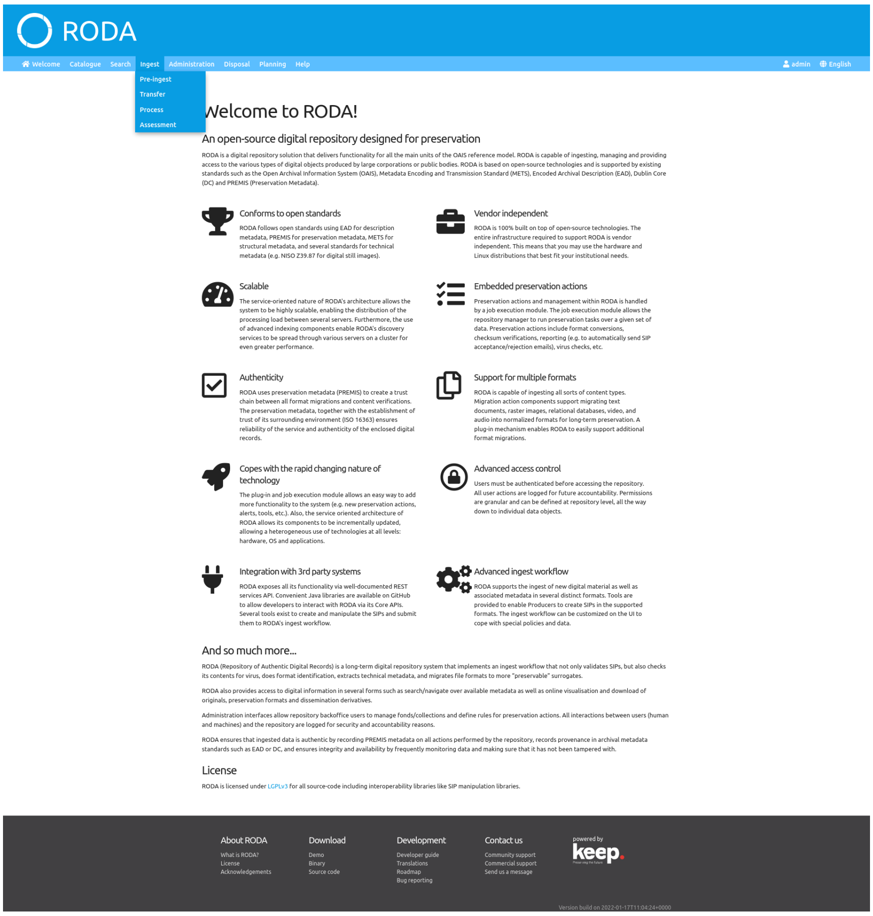
2.2 MANUAL INGEST

Content is usually ingested via an automatic drop folder mechanism, where new files are detected in the shared folder and ingested, but in cases where the default ingest parameters are not adequate or in case an ingest must be redone, a manual ingest procedure can be started.

To perform the manual ingestion in RODA through the interface, the first step is to click on the navigation bar on the ingestion menu, with the following options:

- Pré-ingest
- Transfer
- Process
- Assessment

To make an ingest you need to select or upload one or more SIP's in the system, so you need to click on the Transfer button in the menu described in above and shown in the Figure 1.



Welcome to RODA!

An open-source digital repository designed for preservation

RODA is a digital repository solution that delivers functionality for all the main units of the OAIS reference model. RODA is capable of ingesting, managing and providing access to the various types of digital objects produced by large corporations or public bodies. RODA is based on open-source technologies and is supported by existing standards such as the Open Archival Information System (OAIS), Metadata Encoding and Transmission Standard (METS), Encoded Archival Description (EAD), Dublin Core (DC) and PREMIS (Preservation Metadata).

Conforms to open standards

RODA follows open standards using EAD for description metadata, PREMIS for preservation metadata, METS for structural metadata, and several standards for technical metadata (e.g. NISO Z39.87 for digital still images).

Vendor independent

RODA is 100% built on top of open-source technologies. The entire infrastructure required to support RODA is vendor independent. This means that you may use the hardware and Linux distributions that best fit your institutional needs.

Scalable

The service-oriented nature of RODA's architecture allows the system to be highly scalable, enabling the distribution of the processing load between several servers. Furthermore, the use of advanced indexing components enable RODA's discovery services to be spread through various servers on a cluster for even greater performance.

Embedded preservation actions

Preservation actions and management within RODA is handled by a job execution module. The job execution module allows the repository manager to run preservation tasks over a given set of data. Preservation actions include format conversions, checksum verifications, reporting (e.g. to automatically send SIP acceptance/rejection emails), virus checks, etc.

Authenticity

RODA uses preservation metadata (PREMIS) to create a trust chain between all format migrations and content verifications. The preservation metadata, together with the establishment of trust of its surrounding environment (ISO 16363) ensures reliability of the service and authenticity of the enclosed digital records.

Support for multiple formats

RODA is capable of ingesting all sorts of content types. Migration action components support migrating text documents, raster images, relational databases, video, and audio into normalized formats for long-term preservation. A plug-in mechanism enables RODA to easily support additional format migrations.

Copes with the rapid changing nature of technology

The plug-in and job execution module allows an easy way to add more functionality to the system (e.g. new preservation actions, alerts, tools, etc.). Also, the service oriented architecture of RODA allows its components to be incrementally updated, allowing a heterogeneous use of technologies at all levels: hardware, OS and applications.

Advanced access control

Users must be authenticated before accessing the repository. All user actions are logged for future accountability. Permissions are granular and can be defined at repository level, all the way down to individual data objects.

Integration with 3rd party systems

RODA exposes all its functionality via well-documented REST services API. Convenient Java libraries are available on GitHub to allow developers to interact with RODA via its Core APIs. Several tools exist to create and manipulate the SIPs and submit them to RODA's ingest workflow.

Advanced ingest workflow

RODA supports the ingest of new digital material as well as associated metadata in several distinct formats. Tools are provided to enable Producers to create SIPs in the supported formats. The ingest workflow can be customized on the UI to cope with special policies and data.

And so much more...

RODA (Repository of Authentic Digital Records) is a long-term digital repository system that implements an ingest workflow that not only validates SIPs, but also checks its contents for virus, does format identification, extracts technical metadata, and migrates file formats to more “preservable” surrogates.

RODA also provides access to digital information in several forms such as search/navigate over available metadata as well as online visualisation and download of originals, preservation formats and dissemination derivatives.

Administration interfaces allow repository backoffice users to manage fonds/collections and define rules for preservation actions. All interactions between users (human and machines) and the repository are logged for security and accountability reasons.

RODA ensures that ingested data is authentic by recording PREMIS metadata on all actions performed by the repository, records provenance in archival metadata standards such as EAD or DC, and ensures integrity and availability by frequently monitoring data and making sure that it has not been tampered with.

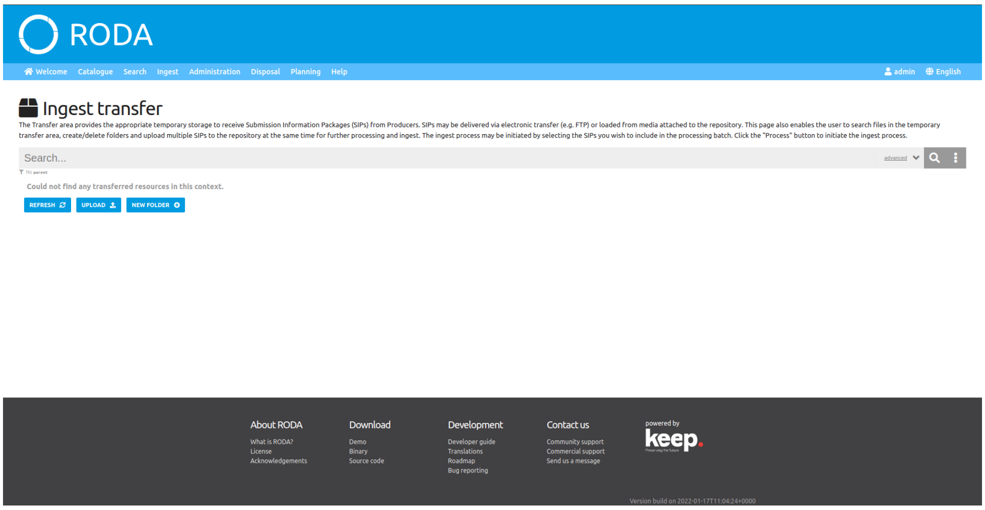
License

RODA is licensed under LGPLv3 for all source-code including interoperability libraries like SIP manipulation libraries.

Version build on 2022-01-17T11:04:24+0000

Figure 1. - Ingest Menu.

This button sends you to the transfer page, as can be seen in Figure 2.



Ingest transfer

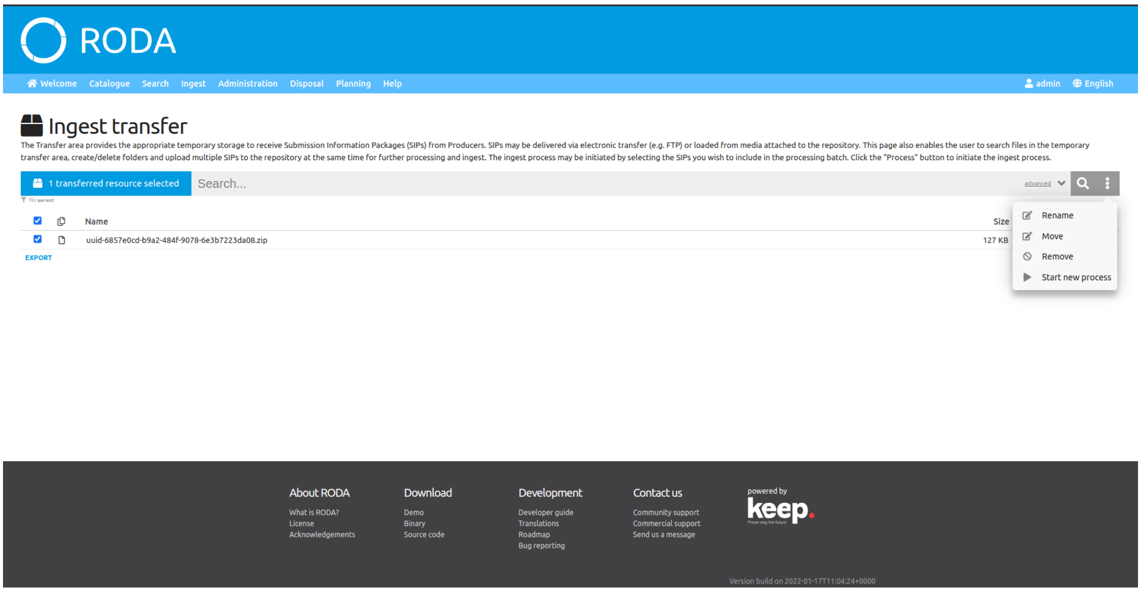
The Transfer area provides the appropriate temporary storage to receive Submission Information Packages (SIPs) from Producers. SIPs may be delivered via electronic transfer (e.g. FTP) or loaded from media attached to the repository. This page also enables the user to search files in the temporary transfer area, create/delete folders and upload multiple SIPs to the repository at the same time for further processing and ingest. The ingest process may be initiated by selecting the SIPs you wish to include in the processing batch. Click the "Process" button to initiate the ingest process.

Could not find any transferred resources in this context.

Version build on 2022-01-17T11:04:24+0000

Figure 2. - Ingest transfer view (1).

In this page click on the upload button, to upload one or more SIP's, when the SIP's upload is done the page will be the same as shown in Figure 3.



Ingest transfer

The Transfer area provides the appropriate temporary storage to receive Submission Information Packages (SIPs) from Producers. SIPs may be delivered via electronic transfer (e.g. FTP) or loaded from media attached to the repository. This page also enables the user to search files in the temporary transfer area, create/delete folders and upload multiple SIPs to the repository at the same time for further processing and ingest. The ingest process may be initiated by selecting the SIPs you wish to include in the processing batch. Click the "Process" button to initiate the ingest process.

☑	Name	Size
☑	uuid-6857e0cd-b9a2-484f-9078-6e3b7223da08.zip	127 KB

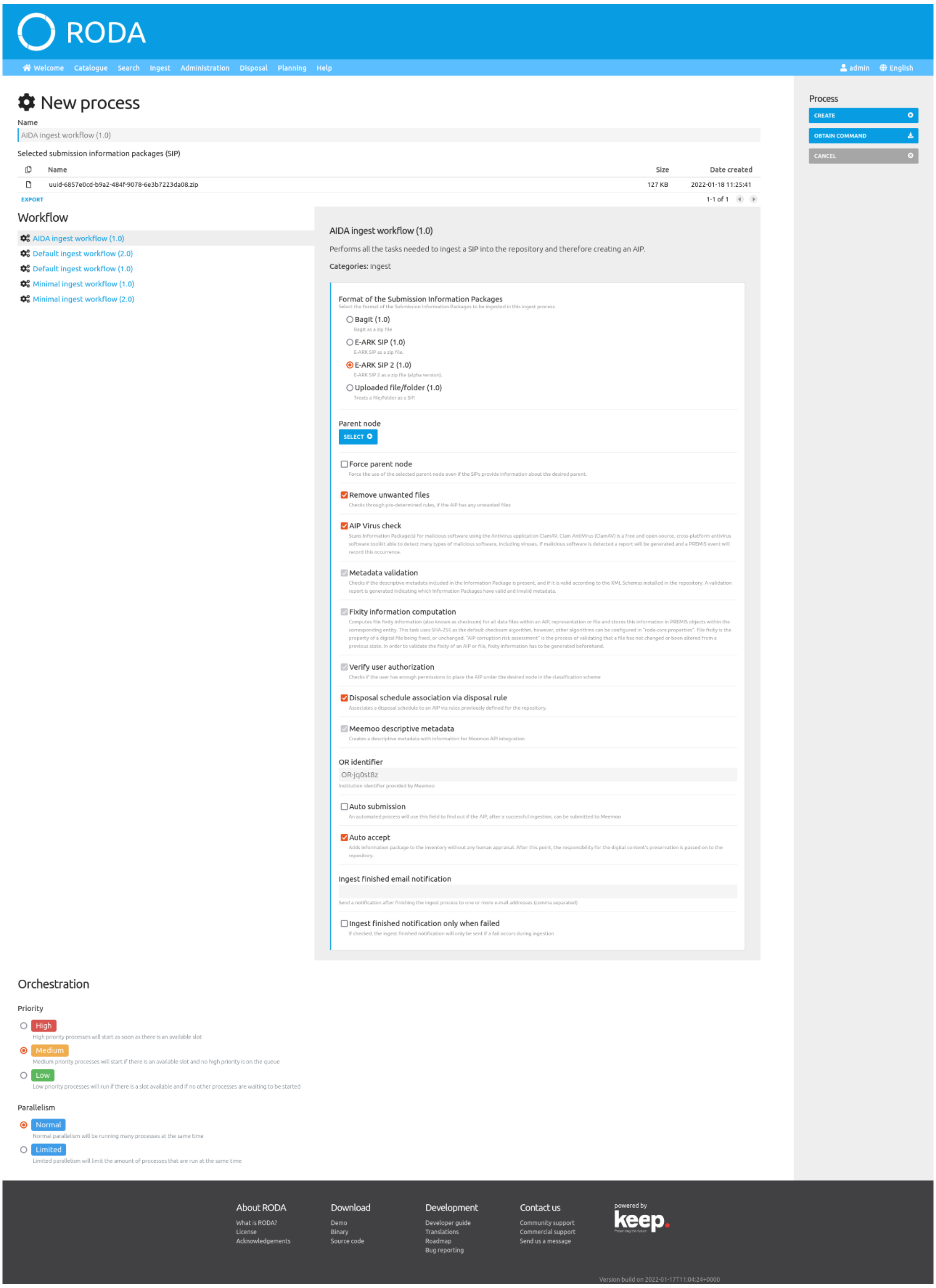
Version build on 2022-01-17T11:04:24+0000

Figure 3. - Ingest Transfer view (2).

To start the ingest process in this page you need to select all the SIP’s that you want to ingest and click on the three dots menu on the right side. This click open a menu with the following options:

- Rename
- Move
- Remove
- Start new process

To continue the ingest process after selecting all the SIP’s for the process, go to **”Start new process”** in the menu described above. After clicking **“Start new process”**, the RODA will show a new page, as shown in Figure 4.



New process

Name

AIDA ingest workflow (1.0)

Selected submission information packages (SIP)

Name	Size	Date created
uuid-6857e0cd-b9a2-484f-9078-6e3b7223da08.zip	127 KB	2022-01-18 11:25:41

1-1 of 1

Workflow

- AIDA ingest workflow (1.0)
- Default ingest workflow (2.0)
- Default ingest workflow (1.0)
- Minimal ingest workflow (1.0)
- Minimal ingest workflow (2.0)

AIDA ingest workflow (1.0)

Performs all the tasks needed to ingest a SIP into the repository and therefore creating an AIP.

Categories: ingest

Format of the Submission Information Packages
Select the format of the Submission Information Packages to be ingested in this ingest process.

○ BagIt (1.0)
BagIt as a zip file

○ E-ARK SIP (1.0)
E-ARK SIP as a zip file.

● E-ARK SIP 2 (1.0)
E-ARK SIP 2 as a zip file (alpha version).

○ Uploaded file/folder (1.0)
Treats a file/folder as a SIP.

Parent node

□ Force parent node
Force the use of the selected parent node even if the SIPs provide information about the desired parent.

☑ Remove unwanted files
Checks through pre-determined rules, if the AIP has any unwanted files

☑ AIP Virus check
Scans Information Package(s) for malicious software using the Antivirus application ClamAV. Clam AntiVirus (ClamAV) is a free and open-source, cross-platform antivirus software toolkit able to detect many types of malicious software, including viruses. If malicious software is detected a report will be generated and a PREMIS event will record this occurrence.

☑ Metadata validation
Checks if the descriptive metadata included in the Information Package is present, and if it is valid according to the XML Schemas installed in the repository. A validation report is generated indicating which Information Packages have valid and invalid metadata.

☑ Fixity information computation
Computes file fixity information (also known as checksum) for all data files within an AIP, representation or file and stores this information in PREMIS objects within the corresponding entity. This task uses SHA-256 as the default checksum algorithm, however, other algorithms can be configured in "roda-core.properties". File fixity is the property of a digital file being fixed, or unchanged. "AIP corruption risk assessment" is the process of validating that a file has not changed or been altered from a previous state. In order to validate the fixity of an AIP or file, fixity information has to be generated beforehand.

☑ Verify user authorization
Checks if the user has enough permissions to place the AIP under the desired node in the classification scheme

☑ Disposal schedule association via disposal rule
Associates a disposal schedule to an AIP via rules previously defined for the repository.

☑ Meemoo descriptive metadata
Creates a descriptive metadata with information for Meemoo API integration

OR identifier
OR-jq0st8z
Institution identifier provided by Meemoo

□ Auto submission
An automated process will use this field to find out if the AIP, after a successful ingestion, can be submitted to Meemoo

☑ Auto accept
Adds information package to the inventory without any human appraisal. After this point, the responsibility for the digital content's preservation is passed on to the repository.

Ingest finished email notification
Send a notification after finishing the ingest process to one or more e-mail addresses (comma separated)

□ Ingest finished notification only when failed
If checked, the ingest finished notification will only be sent if a fail occurs during ingestion

Orchestration

Priority

○ High
High priority processes will start as soon as there is an available slot

● Medium
Medium priority processes will start if there is an available slot and no high priority is on the queue

○ Low
Low priority processes will run if there is a slot available and if no other processes are waiting to be started

Parallelism

● Normal
Normal parallelism will be running many processes at the same time

○ Limited
Limited parallelism will limit the amount of processes that are run at the same time

Version build on 2022-01-17T11:04:24+0000

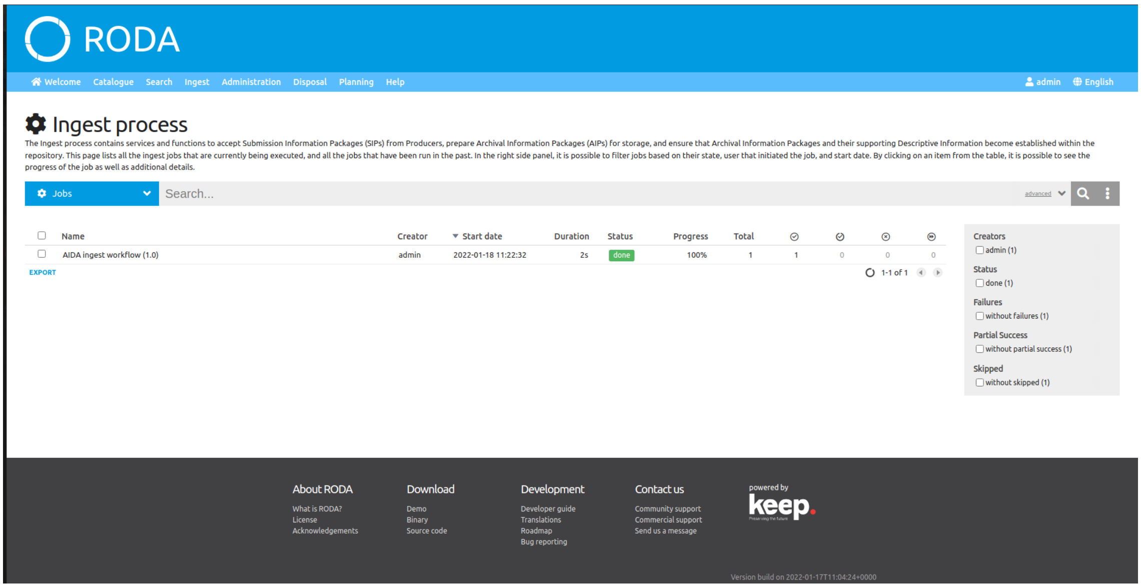
Figure 4. - New process view.

In this page will appear five options on the left side, being them:

- AIDA ingest workflow (1.0)
- Default ingest workflow (2.0)
- Default ingest workflow (1.0)
- Minimal ingest workflow (1.0)
- Minimal ingest workflow (2.0)

By default the **"AIDA ingest workflow (1.0)"** is selected, this is the option with all necessary steps to ingest the AIP with AIDA specific requirements. On the right side of the screen you find a panel with options selected by default and fields to fill in like the **"Ingest finished email notification"** this is the email address that RODA will send the notification of the ingest process. After concluding the configuration of ingest workflow, click on the **"create"** button on the right side of the panel and the process starts.

This click sends you to the page with all job's. In this page you can check the state of your job and see the job report by clicking on the row of your job.



Ingest process

The Ingest process contains services and functions to accept Submission Information Packages (SIPs) from Producers, prepare Archival Information Packages (AIPs) for storage, and ensure that Archival Information Packages and their supporting Descriptive Information become established within the repository. This page lists all the ingest jobs that are currently being executed, and all the jobs that have been run in the past. In the right side panel, it is possible to filter jobs based on their state, user that initiated the job, and start date. By clicking on an item from the table, it is possible to see the progress of the job as well as additional details.

□	Name	Creator	Start date	Duration	Status	Progress	Total	✓	✓	⊗	⊙
□	AIDA ingest workflow (1.0)	admin	2022-01-18 11:22:32	2s	done	100%	1	1	0	0	0

Creators
□ admin (1)
Status
□ done (1)
Failures
□ without failures (1)
Partial Success
□ without partial success (1)
Skipped
□ without skipped (1)

Version build on 2022-01-17T11:04:24+0000

Figure 5. - Ingest process view.

If the ingest process finished with success you can see the AIP or AIP's created in the catalogue page by clicking the **"Catalogue"** button in the navigation bar.

2.3 MEEMO SYNC

Operations relative to submitting or updating records into meemoo storage on E-ARK AIP 2.0.4 format, or to restore all records from AIPs archived in meemoo.

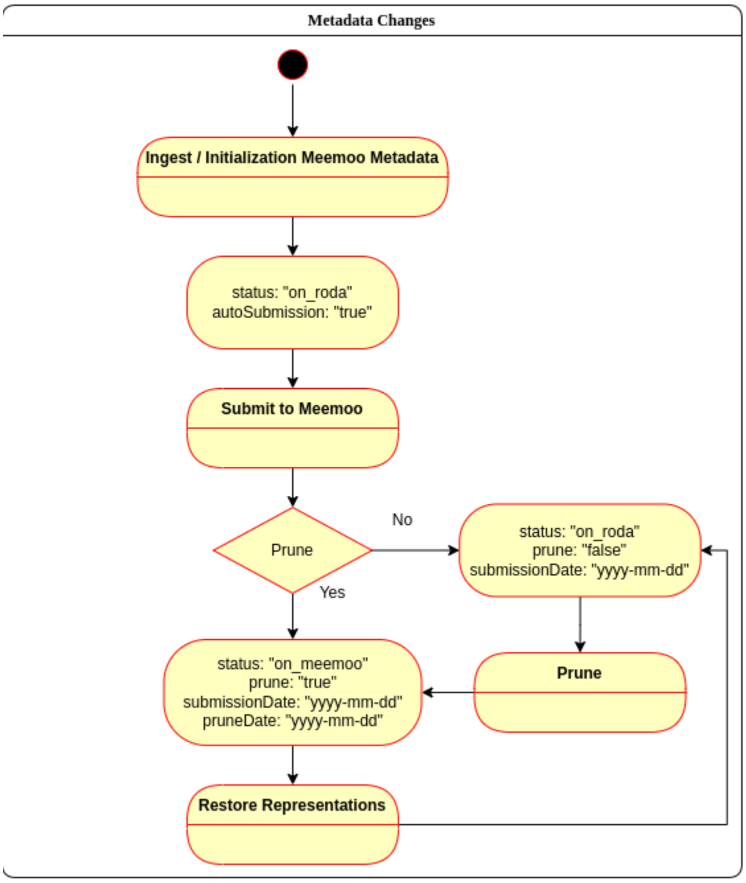


Figure 6 - Metadata Changes.

Figure 6 depicts all metadata changes during the tasks after the AIP ingest process. During the ingest process the meemoo descriptive metadata file is created. This file has the Synchronization AIP status with the value **"on_roda"**, during the submission to meemoo, if prune representations are activated, the meemoo metadata changes the Synchronization AIP status to **"on_meemoo"**, and add the submission date and prune date to the metadata file.

If prune representations are deactivated, the file only adds the submission date to the metadata file. When the status of the AIP is **"on_meemoo"** the Restore operation, as it can be seen above, changes the status again to **"on_roda"** and adds a restore date to the metadata file, and the prune flag is set to false, because the representations of the AIP are again available in RODA.

If the submission date of the AIP is after the last update date and the AIP has representations, we can prune the representations from RODA, which changes the metadata of the AIP to **"on_meemoo"**.

Preservation action should only be executed when the AIP is **"on_roda"**, as this is when the information is available. A preservation action might change the AIP, which should then be re-submitted to RODA.

2.3.1 SUBMIT TO MEEMOO

The first check when submitting the AIP to Meemoo is to check if the AIP has representations. If the AIP has representations the next step is check if it is possible to submit the AIP to meemoo, in this check it is verified if the AIP has already been submitted and if the AIP is still being processed by meemoo.

After checking that the AIP can be submitted, it is checked if it is possible to create the sidecar, for it to be possible it is necessary if the AIP metadata has the mandatory fields, being them:

- ead/archdesc/did/repository/corpname
- ead/archdesc/did/unittitle
- ead/archdesc/did/unitid@label='localId'
- ead/archdesc/did/origination@label='creator'
- ead/archdesc/did/origination@label='producer'

```
<?xml version="1.0" encoding="UTF-8"?>
<VIAA xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:dcterms="http://purl.org/dc/terms/"> <!--meemoo sidecar-->
 <CP>AMSAB-IG</CP> <!--Source: descriptive metadata ead/archdesc/did/repository/corpname-->
 <CP_id>OR-jq0st8z</CP_id> <!--OR-ID (source?)-->
 <dc_title>aanwinst van ABVV</dc_title><!--Source:descriptive metadata
ead/archdesc/did/unittitle-->
 <dc_description>collectie van een digitale dragers</dc_description> <!--Source: descriptive
metadata ead/archdesc/did/scopecontent-->
 <dc_identifier_localid>collectie12345</dc_identifier_localid> <!--Source: descriptive metadata
ead/archdesc/did/unitid@label='localId'.
   Local ID. See https://github.com/Automatic-Ingest-Digital-Archives/SCALA/issues/57-->
 <dc_identifier_localids type="list">
   <ScalaID>a84be406-38a5-4002-a20a-188abd83ff83</ScalaID> <!--Source: AIP. AIP ID. See
https://github.com/Automatic-Ingest-Digital-Archives/SCALA/issues/54-->
 </dc_identifier_localids>
 <dc_creators type="list">
   <Archiefvormer>ABVV</Archiefvormer> <!--Source: descriptive metadata
ead/archdesc/did/origination@label='creator'/name Client Request-->
 </dc_creators>
 <dc_publishers type="list">
   <publisher>AIDA</publisher> <!--Source: descriptive metadata
ead/archdesc/did/origination@label='producer'/name Client Request. This field is hardcoded to
AIDA in RODA-In.-->
 </dc_publishers>
 <dc_rights_comment>niet-raadpleegbaar</dc_rights_comment> <!--Source: descriptive metadata
ead/archdesc/did/accessrestrict→
 <md5>582925fef639c663e0abf9c47cad0727</md5> <!--Source: AIP→
</VIAA>
```

The code above shows the meemoo sidecar and the rules to create the sidecar, as well as the mandatory fields described above.

After these verifications the “Create METS” plugin is run, this plugin will create or update the METS file in AIP making it according to the E-ARK AIP 2.0.4 specification. If the creation was successful the submission will continue, if the creation fails the submission also fails.

Like was explained above, initially the state is **"on_roda"** when AIP is submitted with the pruning option enabled the state is changed to **"on_meemoo"** because the AIP representations are removed from RODA. If the prune option is disabled, RODA keeps the AIP representations and for this reason the state remains **"on_roda"**. This plugin creates after submission a preservation event of the type **"Transfer"** at the repository level .

2.3.2 PRUNE

The Pruning Representations only can execute if the aip is already on meemoo and the last update date is lower than the submission date of the AIP. If passes this two conditions this plugin will check if the AIP has representations, if it has representations it removes these representations from the AIP and changes the meemoo metadata with the new state of the AIP which is pruned .

```xml
<?xml version="1.0" encoding="UTF-8"?>
<metadata>
   <aipVersion>1</aipVersion>
   <identifier>OR-jq0st8z</identifier>
   <syncAIPStatus>on_roda</syncAIPStatus>
   <submissionDate></submissionDate>
   <restoreDate></restoreDate>
   <prune>true</prune>
   <archiveStatus></archiveStatus>
   <autoSubmission>false</autoSubmission>
   <notificationEmail></notificationEmail>
</metadata>
```

Before the process of Prune, the tag **<prune>** on meemoo.xml file has the value false, the AIP synchronization status is **"on_roda"** and doesn't exist as a prune date tag as can be seen in code above. After that the value of the tag **<prune>** will be true, the synchronization AIP status changes to "on_meemoo" and is added to the meemoo.xml file the prune date of the AIP as can be seen in code below.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<metadata>
   <aipVersion>1</aipVersion>
   <identifier>OR-jq0st8z</identifier>
   <syncAIPStatus>on_meemoo</syncAIPStatus>
   <submissionDate></submissionDate>
   <restoreDate></restoreDate>
   <prune>true</prune>
   <pruneDate>yyyy-mm-dd</pruneDate>
   <archiveStatus></archiveStatus>
   <autoSubmission>false</autoSubmission>
   <notificationEmail></notificationEmail>
</metadata>
```

This plugin creates a preservation event of the type **"Destruction"** at the repository level.

2.3.3 RESTORE FROM MEEMOO

The Restore Pruned Representations plugin checks if the AIP is saved on Meemoo, if the AIP has been found it looks for the last version of it and starts the process of restoring the representations of the AIP. This process replaces the old AIP with the AIP with representations.

The initial state as explained above is **“on_meemoo”**, after the process of restoring the state is changed to **“on_roda”** and is added a restore date to the metadata file and the prune tag changes to false. This change occurs because the RODA has the AIP representations stored again in the system, and for this reason the state is **“on_roda”** .

After this process of restore is complete the plugin executes three additional plugins being them:

- Fixity Check plugin
- File Format Identification Plugin
- Virus check Plugin

This plugin like the submission plugin creates a preservention event of the type **“Transfer”** at the repository level.

3 PROJECT-SPECIFIC CONFIGURATIONS

3.1 INGEST WORKFLOW

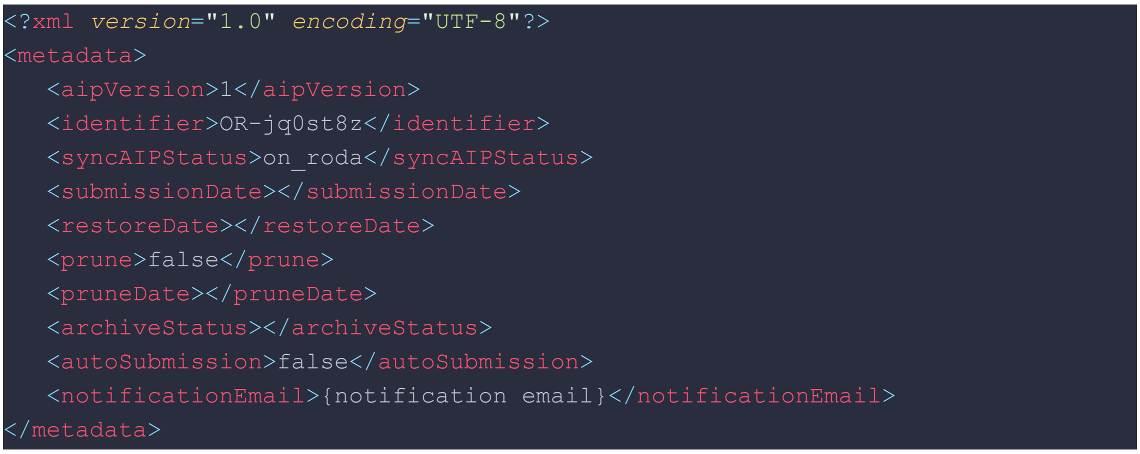
3.1.1 REMOVE UNWANTED FILES

This plugin checks through predetermined rules, if the AIP has any unwanted files. The rules configuration is under config named ingest.ignore, and contains the following default ones:

- `!NAVQUIC.0`
- `Desktop`
- `Desktop.ini`
- `DesktopPrintersDB`
- `.DS_Store`
- `.DS_store`
- `.FBCIndex`
- `FileId.dat`
- `FINDER.DAT`
- `OpenFolderListDF_`
- `Picasa.ini`
- `.picasa.ini`
- `Thumbs.db`
- `._.DS_store`
- `.FBCLockFolder`
- `Resource.frk`
- `TheFindByContentFolder`
- `TheVolumeSettingsFolder`
- `.Trash`
- `.trash`
- `.Trashes`
- `NAV Quickscan*`
- `._*`

3.1.2 MEEMOO DESCRIPTIVE METADATA

This plugin creates a descriptive metadata with information for **Meemoo** API integration. After the execution of this plugin the AIP has a new descriptive metadata file named meemoo.xml.



```xml
<?xml version="1.0" encoding="UTF-8"?>
<metadata>
   <aipVersion>1</aipVersion>
   <identifier>OR-jq0st8z</identifier>
   <syncAIPStatus>on_roda</syncAIPStatus>
   <submissionDate></submissionDate>
   <restoreDate></restoreDate>
   <prune>false</prune>
   <pruneDate></pruneDate>
   <archiveStatus></archiveStatus>
   <autoSubmission>false</autoSubmission>
   <notificationEmail>{notification email}</notificationEmail>
</metadata>
```

This plugin will create a descriptive metadata file with the information represented above, such as the auto submission flag, if the AIP is pruned or not, the initial synchronization AIP status **"on_roda"** and the notification email.

3.2 ACCESS FEATURES

The advanced search and facets[2] have been customized for this project, adding search for the following additional attributes:

- Meemoo AIP version
- Meemoo Identifier
- Meemoo Archive Status
- Producer

In terms of facets, the attributes added are the following:

- Meemoo prune
- Meemoo archive status

3.3 PRESERVATION ACTIONS

The following additional plugins with preservation actions were added to the digital preservation repository solution for AIDA.

3.3.1 CREATE E-ARK AIP 2.0 MANIFEST FILES (METS.XML)

Plugin that generates E-ARK AIP 2.0 manifest files ("METS.xml") from existing AIP information in the storage layer. This plugin only works with filesystem as the storage service.

3.3.2 IMAGE CONVERSION (IMAGEMAGICK)

ImageMagick is a tool that can read and write images in a variety of formats (over 200) including PNG, JPEG, JPEG-2000, GIF, TIFF, DPX, EXR, WebP, Postscript, PDF, and SVG. ImageMagick can also be used to resize, flip, mirror, rotate, distort, shear and transform images, adjust image colours, apply various special effects, or draw text, lines, polygons, ellipses and Bézier curves (e.g. set Command arguments to " -resample 90" to resize the image to 90 dpi). The results of conversion will be placed on a new representation under the same Archival Information Package (AIP) where the files were originally found. A PREMIS event is also recorded after the task is run. For a full list of supported formats, please visit http://www.imagemagick.org/script/formats.php

[2] A facet is a filter list with a counter that appears on a sidebar on search to aid filtering through the results.

3.3.3 OFFICE DOCUMENTS CONVERSION (UNOCONV)

Converts office files using the “unoconv” (Universal Office Converter), which uses LibreOffice[3] to convert Office files. The results of conversion will be placed on a new representation under the same Archival Information Package (AIP) where the files were originally found. A PREMIS event is also recorded after the task is run. “unoconv” is a tool that converts between any document format that OpenOffice understands. It uses OpenOffice's UNO bindings for non-interactive conversion of documents. Supported document formats include Open Document Format (odt), MS Word (doc), MS Office Open/MS OOXML (ooxml), Portable Document Format (pdf), HTML (html), XHTML (xhtml), RTF (rtf), Docbook (docbook), and more. The outcome of this task is the creation of a new OpenOffice (and thus unoconv) that supports various import and export formats. Not all formats that can be imported can be exported and vice versa. For a full list of supported formats, please visit - http://dag.wiee.rs/home-made/unoconv/

[3] https://www.libreoffice.org