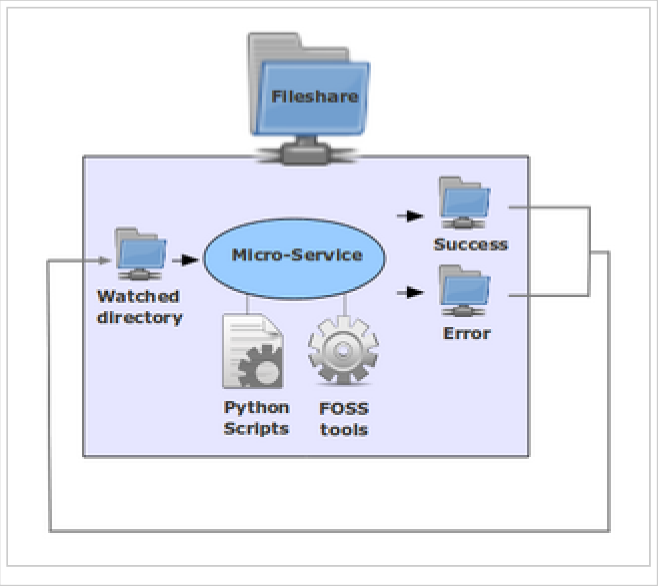
Micro-services

From Archivematica

Main Page > Documentation > Technical Architecture > Micro-services



Archivematica implements a micro-service (http://www.cdlib.org/services/uc3/curation/) approach to digital preservation. The Archivematica micro-services are granular system tasks which operate on a conceptual entity that is equivalent to an OAIS information package: Submission Information Package (SIP), Archival Information Package (AIP), Dissemination Information Package (DIP). The physical structure of an information package will include files, checksums, logs, submission documentation, XML metadata, etc..

These information packages are processed using a series of micro-services. Micro-services are provided by a combination of Archivematica Python scripts and one or more of the free, open-source software tools bundled in the Archivematica system. Each micro-service results in a success or error state and the information package is processed accordingly by the next micro-service. There are a variety of mechanisms used to connect the various micro-services together into complex, custom workflows. Resulting in a complete ingest to access system.

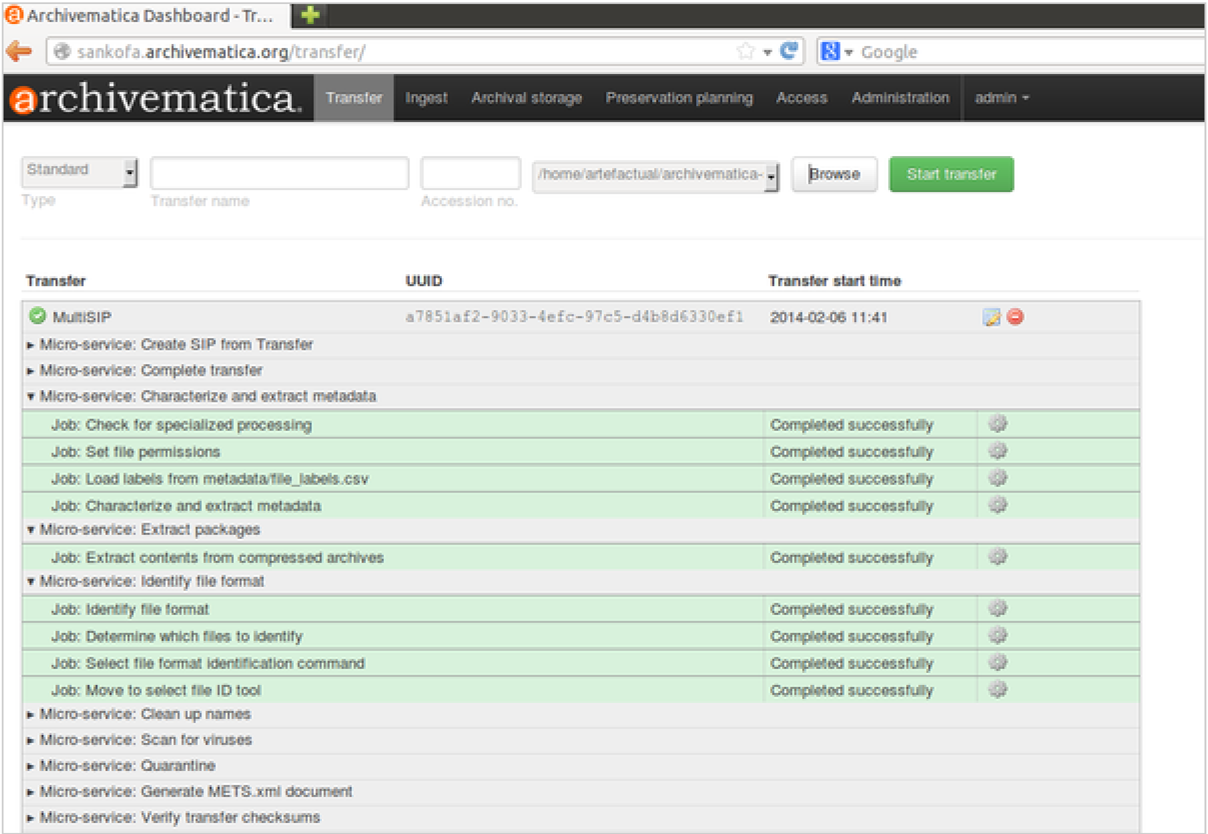
Archivematica implements a default ingest to access workflow that is compliant with the ISO-OAIS functional model. Micro-services can be distributed to processing clusters for highly scalable configurations.

Contents

- 1 Current Micro-services in Archivematica
  - 1.1 Transfer
  - 1.2 Ingest
- 2 Historical Micro-services in Archivematica Releases

Current Micro-services in Archivematica

A micro-service may consist of a number of discrete tasks, or jobs. In the Archivematica dashboard, micro-services are always shown, while jobs may be viewed by expanding the micro-service (i.e. by clicking on the grey background behind the micro-service name).



Type: Standard
Transfer name:
Accession no.:
/home/artefactual/archivematica-

Transfer	UUID	Transfer start time	
MultiSIP	a7851af2-9033-4efc-97c5-d4b8d6330ef1	2014-02-06 11:41	
▸ Micro-service: Create SIP from Transfer			
▸ Micro-service: Complete transfer			
▾ Micro-service: Characterize and extract metadata			
Job: Check for specialized processing		Completed successfully	
Job: Set file permissions		Completed successfully	
Job: Load labels from metadata/file_labels.csv		Completed successfully	
Job: Characterize and extract metadata		Completed successfully	
▾ Micro-service: Extract packages			
Job: Extract contents from compressed archives		Completed successfully	
▾ Micro-service: Identify file format			
Job: Identify file format		Completed successfully	
Job: Determine which files to identify		Completed successfully	
Job: Select file format identification command		Completed successfully	
Job: Move to select file ID tool		Completed successfully	
▸ Micro-service: Clean up names			
▸ Micro-service: Scan for viruses			
▸ Micro-service: Quarantine			
▸ Micro-service: Generate METS.xml document			
▸ Micro-service: Verify transfer checksums			

Archivematica dashboard showing a micro-service and its jobs

The table below shows micro-services and jobs in the current release of Archivematica (**Updated July 2015 for release 1.4.1**).

Transfer

Micro-service	Description
Approve Transfer [Collapse]	This is the approval step that moves the transfer into the Archivematica processing pipeline.
Verify transfer compliance [Collapse] Set file permissions Move to processing directory Set transfer type: (Standard, Zipped bag, Unzipped bag, DSpace, Maildir) Remove hidden files and directories Remove unneeded files	Moves the transfer to a processing directory based on selected transfer type (standard, zipped bag, unzipped bag, DSpace export or maildir). Verifies that the transfer conforms to the folder structure required for processing in Archivematica and restructures if required. The structure is as follows: */logs/*, */metadata/*, */metadata/submissionDocumentation/*, */objects/*.

Attempt restructure for compliance Verify transfer compliance Verify mets_structmap.xml compliance	
Rename with transfer UUID [Collapse] Rename with transfer UUID	Directly associates the transfer with its metadata by appending the transfer UUID to the transfer directory name.
Include default Transfer processingMCP.xml [Collapse] Include default Transfer processingMCP.xml	Adds a file named processingMCP.xml to the root of the transfer. This is a configurable xml file to pre-configure processing decisions. It can configure workflow options such as creating transfer backups, quarantining the transfer and selecting a SIP creation option.
Assign file UUIDs and checksums [Collapse] Set file permissions Assign file UUIDs to objects Assign checksums and file sizes to objects	Assigns a unique universal identifier and sha-256 checksum to each file in the */objects/* directory and sets file permission to allow for continued processing.
Verify transfer checksums [Collapse] Verify metadata directory checksums	Checks any checksum files that were placed in the */metadata/* folder of the transfer prior to moving the transfer into Archivematica.
Generate METS.xml document [Collapse] Generate METS.xml document	Generates a basic METS file with a fileSec and structMap to record the presence of all objects in the */objects/* directory and their locations in any subdirectories. Designed to capture the original order of the transfer in the event the user chooses subsequently to delete, rename or move files or break the transfer into multiple SIPs. A copy of the METS file is automatically added to any SIP generated from the transfer.
Reformat metadata files [Collapse] Process transfer JSON metadata	Transforms JSON metadata included with transfer to metadata.csv

Quarantine [Collapse] Workflow decision - send transfer to quarantine Move to quarantine Remove from quarantine	Quarantines the transfer for a set duration, to allow virus definitions to update, before virus scan.
Scan for viruses [Collapse] Scan for viruses	Uses ClamAV to scan for viruses and other malware. If a virus is found, the transfer is automatically placed in */sharedDirectoryStructure/failed/* and all processing on the transfer is stopped.
Generate transfer structure report [Collapse] Generate transfer structure report Move to generate transfer tree	Uses Tree to create transfer structure report (original order report) in txt format
Clean up names [Collapse] Sanitize object's file and directory names Sanitize Transfer name	Some file systems do not support unicode or other special characters in filenames. This micro-service removes prohibited characters and replaces them with dashes. Original filenames are preserved in the PREMIS metadata.
Identify file format [Collapse] Move to select file ID tool Select file format identification command Determine which files to identify Identify file format	Identifies formats of the objects in the transfer using either FIDO or file extension based on user choice. Format types are managed in the Format Policy Registry. This micro-service can be skipped and done in Ingest instead.
Extract packages [Collapse] Move to processing directory Move to extract packages Extract packages? (preconfigured choice)	Extracts objects from any zipped files or other packages. Extracts attachments from maildir transfers. User can preconfigure choices about whether to extract and whether to keep one or both of extracted object contents and/or the package itself.

Delete package after extraction? (preconfigured choice) Extract contents from compressed archives	
Update METS.xml document [Collapse] Add processed structMap to METS.xml document	Creates a structMap in the transfer METS to reflect structure of transfer
Characterize and extract metadata [Collapse] Characterize and extract metadata Load labels from metadata/file_labels.csv	Characterizes formats and extracts object metadata using File Information Tool Set (FITS), FFprobe, Exiftool and/or Mediainfo
Validation [Collapse] Validate formats	Validates formats using JHOVE.
Examine Contents [Collapse] Examine contents	Runs Bulk Extractor on transfer contents and creates reports
Complete transfer [Collapse] Index transfer contents Create transfer metadata XML Move to SIP creation directory for completed transfers	Indexes transfer contents, creates transfer METS XML, then marks the transfer as complete.
Create SIP from Transfer [Collapse]	This is the approval step that moves the transfer to the SIP packaging micro-services (Ingest) if user chooses to Create single SIP and continue processing. User can also choose to Send transfer to backlog at this time, which creates an associated PREMIS event.

Check transfer directory for objects Updating transfer file index Create placement in backlog PREMIS events Move transfer to backlog Check transfer directory for objects Load options to create SIP	

Ingest

Micro-service	Description
Approve SIP creation [Collapse] Approve SIP creation Create removal from backlog PREMIS events	Applies to SIPs that have been created from backlog. Approves creation of SIP, and creates PREMIS event for removal from backlog.
Verify transfer compliance [Collapse] Verify mets_structmap.xml compliance	Verifies the METS from the transfer.
Verify SIP compliance [Collapse] Set file permissions Move to processing directory Verify SIP compliance	Verifies that the SIP conforms to the folder structure required for processing in Archivematica. The structure is as follows: */logs/*, */metadata/*, */metadata/submissionDocumentation/*, */objects/*.
Rename SIP directory with SIP UUID [Collapse] Rename SIP directory with SIP UUID Check if SIP is from Maildir Transfer	Directly associates the SIP with its metadata by appending the SIP UUID to the SIP directory name and checks if SIP is from Maildir transfer type to determine workflow.

Include default SIP processingMCP.xml [Collapse] Include default SIP processingMCP.xml	Copies the processing configuration file added to the transfer in **Include default Transfer processingMCP.xml**, above, to the SIP.
Remove cache files [Collapse] Remove cache files	Removes any thumbs.db files.
Clean up names [Collapse] Load Dublin Core metadata from disk Sanitize SIP name Set file permissions	Some file systems do not support unicode or other special characters in filenames. This micro-service removes prohibited characters and replaces them with dashes. Original filenames are preserved in the PREMIS metadata.
Normalize [Collapse] Identify manually normalized files Check for Service directory Check for Access directory Set remove preservation and access normalized files to renormalize link. Grant normalization options for no pre-existing DIP Move to workFlowDecisions-createDip directory Find options to normalize as Set resume link after tool selected Move to select file ID tool Select pre-normalize file format identification commant Identify file format Resume after normalization file identification tool selected Normalize Move to processing	Determines which normalization options are available for the SIP and presents them to the user as choices. Normalizes (i.e. generates preservation and/or access copies) based on selection. Thumbnail files are also generated during this micro-service.

directory Create DIP directory Create thumbnails directory Normalize thumbnails Normalize access Normalize preservation Set file permissions Remove files without linking information (failed normalization artifacts etc.) Move to approve normalization directory Approve normalization Load post approve normalization link Set resume link after handling any manually normalized files Move to processing directory Set file permissions Load finished with manual normalized link	
Process manually normalized files [Collapse] Check for manual normalized files	Processed files manually normalized during processing
Add final metadata [Collapse] Move to metadata reminder Reminder: add metadata if desired Set file permissions	Provides the operator a reminder to add metadata through the UI if desired.
Transcribe SIP contents [Collapse] Transcribe SIP contents Transcribe	Runs Tesseract OCR tool on any JPG or TIFF image files in SIP.
Process submission documentation [Collapse]	Processes any submission documentation included in the SIP and adds it to the */objects/* directory.

Copy transfer submission documentation Check for submission documentation Move submission documentation into objects directory Assign file UUIDs to submission documentation Assign checksums and file sizes to submissionDocumentation Sanitize file and directory names in submission documentation Scan for viruses in submission documentation Select file format identification command Identify file format Characterize and extract metadata on submission documentation Remove files without linking information (failed normalization artifacts etc.)	
Process metadata directory [Collapse] Copy transfers metadata and logs Process JSON metadata Move metadata to objects directory Assign file UUIDs to metadata Assign checksums and file sizes to metadata Sanitize file and directory names in metadata Scan for viruses in metadata Identify file format of metadata files Characterize and extract metadata on metadata files Remove empty manual normalization directories	Processes metadata.

Verify checksums [Collapse] Verify checksums generated on ingest	Verifies checksums generated on ingest
Generate AIP METS [Collapse] Generate METS.xml document	Generates Archivematica AIP METS.xml document
Prepare DIP [Collapse] Copy thumbnails to DIP directory Copy METS to DIP directory Copy preconfigured choice XML to DIP directory Generate DIP Copy OCR data to DIP directory Set file permissions	Creates a DIP containing access copies of the objects, thumbnails and a copy of the METS file.
Prepare AIP [Collapse] Check if DIP should be generated Prepare AIP Move to compressionAIPDecisions directory Select compression algorithm Select compression level Compress AIP Copy submission documentation Create AIP pointer file Set bag file permissions Check if AIP is a file or directory Removed bagged files	Creates an AIP in Bagit format. Creates the AIP pointer file. Indexes the AIP, then losslessly compresses it.

Upload DIP [Collapse] Upload DIP	Allows the user to choose to upload the DIP AtoM, CONTENTdm, Archivist's Toolkit. Also allows user to store DIP or reject DIP.
Upload DIP to AtoM [Collapse] Upload DIP Move to the uploadedDIPs directory	The user uploads the DIP to a selected description in AtoM.
Upload DIP to CONTENTdm [Collapse] Upload DIP Restructure DIP for CONTENTdm upload Move to the uploadedDIPs directory	The user creates a DIP suitable for upload to CONTENTdm using Project Client.
Upload DIP to Archivists' Toolkit [Collapse]	The user uploads the DIP metadata to Archivists' Toolkit
Store DIP [Collapse] Upload DIP Retrieve DIP Storage Locations Store DIP location Store DIP Move to the uploadedDIPs directory	The user stores the DIP in a location pre-configured in the Storage Service.
Store AIP [Collapse] Move to the store AIP approval directory Store AIP Retrieve AIP Storage Locations Store AIP location Move to processing directory	Moves the AIP to */sharedDirectoryStructure/www/AIPsStore/* or another specified directory. Before the AIP has been stored, a copy of it is extracted to a local temp directory, where it is subjected to standard BagIt checks: verifyvalid, checkpayloadoxum, verifycomplete, verifypayloadmanifests, verifytagmanifests.

Verify AIP Store the AIP Index AIP Clean up after storing AIP Remove processing directory	

Historical Micro-services in Archivematica Releases

- Archivematica 1.1 Micro-services
- Archivematica 1.0 Micro-services
- Archivematica 0.10 Micro-services
- Archivematica 0.9 Micro-services
- Archivematica 0.8 Micro-services
- Archivematica 0.7.1 Micro-services

Retrieved from "https://wiki.archivematica.org/index.php?title=Micro-services&oldid=10619"

- This page was last modified on 14 August 2015, at 18:06.
- This page has been accessed 50,833 times.
-