

[Use Cases and Requirements for Batch Import and Export](#)

[High Level Import Export Use Cases](#)

[User Stories](#)

[Technical \(customization and extensibility\)](#)

[Non-technical \(usability\)](#)

[Import/ Export Sources](#)

[Metadata](#)

[Files](#)

[Whole](#)

[Requirements](#)

[Candidate Specifications](#)

Use Cases and Requirements for Batch Import and Export

High Level Use Cases

Use Case	User	Notes during WG Unconference
Export metadata and files from non-Fedora/Hyrax repository for Import to another Hyrax/Fedora repository	CoSector-University of London National Library of Medicine University of York	UMichigan is doing their own batch export / import. You

	University of Michigan UCLA	can see their code here: (we talked as a group about it) See here: https://github.com/mlibrary/deepblue/tree/master/lib/tasks the two base classes are yaml_populate.rb New_content_service.rb (base for creating)
Export metadata and files from repository for import to another system	CoSector-University of London Durham University Indiana University Northwestern University UC San Diego UC Santa Barbara University of Utah UCLA	(presumably any system BUT HYRax/Fedora since that is above) -- a more general output mechanism For example OCFL could be an export ...and an import
Import Newspapers - NDNP created	BPL University of Utah	
Import Newspapers that are non-NDNP created	University of Utah UCLA	
Export Newspapers - NDNP created	BPL University of Utah	
Export Newspapers that are non-NDNP	University of Utah	

<p>Migration Export from one Hyrax version for import into an newer version</p>	<p>Indiana University Northwestern University UC Santa Barbara University of Utah WGBH University of York University of Michigan</p>	
<p>Import batches from ProQuest</p>	<p>George Washington University (ETDs) UC Santa Barbara (maybe → ETDs) UNC-Chapel Hill University of Utah (ETDs) UU</p>	
<p>Import from another system:</p> <ul style="list-style-type: none"> ● CONTENTdm ● Sufia ● EPrints ● Fedora3 ● Millennium (MARC) ● Alma (MARCXML) ● ProQuest (ETDs) ● DSpace 	<p>Durham University (EPrints, Sufia) Indiana University(fedora3) National Library of Medicine (fedora3) Northwestern University(Alma) UC San Diego (Millenium) UC Santa Barbara (Alma) University of Utah (Alma) University of York(fedora3) George Washington University (ProQuest ETDs) University of Michigan(DSpace) UCLA (DSpace) Oxford (Fedora 3) UVic, IUPUI, CSU Fresno (Content DM)</p>	<p>Oregon State University has done an export from DSpace using BagIt. They didn't use browse-everything in that case, but they recommend that as a better approach to others.</p>
<p>Import metadata and files to create new objects (that are not in a repository) in Hyrax, including objects nested in other objects</p>	<p>Duke University Durham University George Washington University</p>	

	Indiana University National Library of Medicine Northwestern University UC San Diego UC Santa Barbara UNC-Chapel Hill UU University of York Oregon State University University of Oregon	
Batch Update - export metadata from Hyrax for offline remediation and subsequent import back into Hyrax to update existing objects.	Duke University Indiana University Linn-Benton Northwestern University UC San Diego UC Santa Barbara UNC-Chapel Hill University of Utah UU Oregon State University	
Import/Export Bagit bags	University of Utah	

User Stories

Technical (customization and extensibility)

As a developer/maintainer of the batch importer...

Story	Requirement
I want failures to happen cleanly, so that I don't end up with partial records, or have to trawl through log files to find out what did and didn't import	Error handling Monitoring and reporting Rollback
I want imports to be idempotent so that recovering from an error can be as simple as running the same import again.	Idempotence
I want to be able to automate ingests from scripts / rails console	Automation
I want to be able to run ingests over HTTP/S from a different server without having to manually SFTP files around	API Remote deposit
I want to support multiple Work types (models), and for it to be easy to extend the base import/exporter for different models and metadata sets	Extensible
I want to be able to easily extend to different import/export sources	Extensible
I want a no-op option so that I can test ingests without actually running the ingest, eg. Is the source data valid? Do the files exist?	Pre-import testing/validation

If I process this, will anything bad happen?	
I want to be able to export what I imported	Round-tripping
I want to have a simple RESTful way of doing an import/export	API
I want to be able to see logs of what was done, whether the import was a success or a failure	Logging

Non-technical (usability)

As a user of a batch import GUI...

Story	Requirement
I want to see a list of all fields that I can use in the metadata import file and I want to know which are required. I should be able to copy this list and paste it into a text editor or spreadsheet, creating column headers for my import file, or download a template file.	Import configuration and setup Import template
I want to be able to populate certain fields with standard values from dropdowns	Import configuration and setup
I want to be able to choose access permissions	Setup
I want to run a pre-import test to see if my column headers match those allowed by the import tool. I want an on-screen report indicating that either all headers are	Pre-import testing/validation

valid, or a list of invalid headers. I would also like to download this report.	
I want my ingest file to use human-readable labels so I, and collaborators can easily prepare and review the file for ingest. URIs can also be used, either by themselves or combined with labels.	Human-Readable
I want to be able to preview the import	Preview
I want to test that all files in my import file are present in the specified file location	Pre-import testing/validation
I want potential duplicates to be flagged	Identify duplicates
I want support for complex and nested objects	Complex/Nested objects
I want work-work and Collection-work relationships	Relationships
I want support for Authorities, to add them during ingest and to associate imported records with authority records	Authorities
I want to import both a label and a URL for a property, i.e., a related resource with a label and a URL to a finding aid	Complex/Nested metadata
I want a report on success/failure, and progress monitoring	Reporting and Monitoring Download, email reports
I want the option to either use my own file derivatives (specifying use/quality) or have the tool create them automatically upon ingest	Derivative-creation configuration
I want the option to either use pre-minted object IDs or have the tool create them automatically upon ingest	Identifiers

I want to use line breaks in field values, e.g., paragraphs in a description	Text formatting
--	-----------------

Import/ Export Sources

Metadata

- CSV
- TSV
- JSON
- Excel
- METS
- MODS
- MARC
- XML
- FOXML

Files

- local-destination (locally available on the server being imported *to*)
- local-source (locally available on the server being imported *from*)
- local-user (locally on the user's machine)
- remote-networked (available from a network storage service)
- remote-cloud (available in a remote cloud location)

Whole

- Zip

- Bag

Requirements

MUST

- Remote deposit via an API endpoint
- Automatable (via an API call)
- Single import/export format with extensible mappers to/from different formats
- Round-tripping (minimum CSV and JSON)
- Import configuration and setup, eg. boilerplate fields, work type, license, visibility
- Uses existing functionality in Hyrax - does not duplicate any Hyrax features
- Cleanly handle errors to ensure ingest is recoverable
 - note: Hyrax Actor stack does not do this very well, so this may be problematic/need fixing, for example it rolls back database transactions but not object creation
 - This assumes objects are independent, that one failed object will not invalidate another
- Provide a report of success/failure
- Use a supplied identifier as the id, or create a new id if one is not supplied
- Extensible, with clear documentation on how
- Support nested objects
- GUI for preparing, starting and monitoring imports and exports
- Pre-import testing/validation, eg. are files present? is metadata in correct format? can all fields be processed?
- Support relationships between works and collections
- Batch update - for example batch export metadata from Hyrax for offline remediation and subsequent import back into Hyrax to update existing objects; more sophisticated than current batch functionality
- Derivative-creation configuration, eg. do create derivatives, use supplied derivatives
- Support ingest from multiple sources combined,, eg. a CSV containing multiple records, a FITS xml for each record

SHOULD

- Use Hyrax actor stack for creation of objects
- Import template - provides a downloadable template
- Import configuration and setup via GUI
- Rollback the current object on failure and report failed objects
- Be idempotent (running the same import again doesn't change anything) for easier error handling. Running the same import again on a partially completed operation will just finish the operation but not duplicate what was already imported earlier
- Interactive monitoring of current imports/exports
- Support complex metadata, for example
 - Nested metadata - person with name parts
 - Values as URIs to external objects
 - Values as link to internal objects (ie. another repository object)
 - Storing URIs and indexing labels
- Support local and external authorities, for example:
 - URI to term in external scheme
 - Local db entry for an authority term
 - Local repository authority object
- Support custom relationships between works with other works
- Batch delete as an *optional* feature, so a repository can disable it, or only use in staging as admin
 - Nb. it's already there in Hyrax
 - 'mark deleted' rather than real delete (make Hyrax tombstone aware)
- Specify location (remote/local directory, URL, etc.) for each content file (so all files don't have to be in a single directory with unique filenames)
- Support triggering operations in other systems and waiting for them to complete

MAY

- Preview import/export - run a 'no-op' to indicate whether the import will succeed
- Flag potential duplicate records during preview
- Flag potential duplicate records during import

Candidate Specifications

- SWORD
- Fedora Specification (and Import/Export Utility)