DRS Batch Builder User Guide

This User Guide describes how to use DRS Batch Builder -- a desktop application that simplifies the process of DRS batch deposit. Batch Builder currently works with these batch genres:

- **Still images**
- **Page-turned objects**
- **Container batches**, including:
  - Dark PDS document containers
  - Opaque containers
- **PDF batches**

**Need help?** To report a problem or ask a question about Batch Builder, use the feedback form:

http://nrs.harvard.edu/urn-3:hul.ois:bbhelp

to contact the Batch Builder support team. If you are reporting a problem, please provide a detailed description of the problem and a copy of any Batch Builder error or warning messages.

Batch Builder is developed and supported by the Harvard University Library Office for Information Systems (OIS).

1. **Getting Started**

   1.1 Installing Batch Builder ............................................................ 4
   1.2 Setting options ........................................................................... 4
   1.3 User interface basics ................................................................. 6
   1.4 Batch creation overview ........................................................... 7
   1.5 About file names ...................................................................... 9
   1.6 About directory names and structure ....................................... 10
   1.7 Batch preparation checklist .................................................... 12
   1.8 Supporting documentation ..................................................... 13

2. **Creating a General Image Batch** ........................................ 14

   2.1 Accepted formats .................................................................. 14
   2.2 File name rules ...................................................................... 15
   2.3 Batch directory rules .............................................................. 15
   2.4 Procedure to create an image batch .................................... 16

3. **Creating a Page-Turned Batch** .......................................... 21
3.1 Accepted formats .......................... 22
3.2 File name rules ......................... 22
3.3 Batch directory rules ................. 24
3.4 About the PDS METS file......... 25
3.5 Procedure to create a page-turned batch .......................... 25

4. Creating a Container Batch .............. 31
   4.1 Internal ZIP file details -- Dark PDS document containers .................. 31
   4.2 Internal ZIP file details -- Opaque containers .................................. 32
   4.3 ZIP file accepted formats .................................................. 34
   4.4 ZIP file naming rules .................................................. 34
   4.5 ZIP file size .................................................. 35
   4.6 ZIP container batch directory rules ............................................ 35
   4.7 Procedure to create a container batch ........................................... 36
   4.8 Brightening dark PDS document containers .......................... 42

5. Creating a PDF Batch ..................... 43
   5.1 Accepted formats .................................................. 43
   5.2 File name rules .................................................. 44
   5.3 Batch directory rules .................................................. 44
   5.4 Procedure to create a PDF batch ................................................. 45

6. Uploading Batches to DRS ................. 51

7. Expressing Relationships ................... 52
   7.1 Image relationships .................. 52
   7.2 Page-turned object relationships .................................................. 54

8. Using External Mapping Files ............ 56
   8.1 Owner supplied name mapping file .................................................. 56
   8.2 Page sequence mapping file .................................................. 57

9. Working with URNs .......................... 58
   9.1 Assigning URN resource name patterns .................................................. 58
   9.2 Using URNs .................................................. 59

10. Reports and Messages .................... 61
   10.1 Summary report .................. 61
   10.2 Processing and error messages .................................................. 62

11. Metadata Reference ......................... 64
   11.1 Administrative properties .................................................. 64

Revised: March 18, 2010
11.2 Global properties .......................................................................................... 67
11.3 Directory-level properties ........................................................................... 70
11.4 File-level properties .................................................................................. 74

12. Command Line Reference ........................................................................... 75
   12.1 Generic command line syntax ................................................................. 75
   12.2 Actions .................................................................................................... 75
   12.3 Switches .................................................................................................. 76

User Guide Revision History ............................................................................. 78
1. **Getting Started**

1.1 **Installing Batch Builder**

Batch Builder is a java-based application compatible with Windows, Mac and Linux operating systems.

The Batch Builder application package includes a graphical user interface and a separate command line tool (for automated deposit workflows). The application package is bundled as a zip file, which must be unzipped and installed on the desktop. The Batch Builder zip archive can be downloaded from the [DRS Depositors iSite](Harvard access only).

Note: Batch Builder requires that version 1.5.0 or higher of the Java Runtime Environment (JRE) be installed on the desktop.

1. Copy the Batch Builder zip archive to your file system.
2. Unzip the archive and extract the files to a directory of your choice. The installer will copy the files to a “batchbuilder-1.x” directory.
3. To start the graphical user interface, double click the launch (executable) file (look for the Batch Builder icon). See the [Command Line Reference](#) section for instructions on using the command line tool.
4. You can also create a shortcut / alias on your desktop to BatchBuilder launch (executable) file for ease of use.

Consult the `release.txt` file in the archive for information on recent updates to the application. A PDF version of this User Guide (bb-userguide.pdf) is also included in the archive.

1.2 **Setting options**

There are a few Batch Builder features that are controlled by using View > Options on the main menu. These options will affect all projects.
Batch Builder User Guide

Batch Builder Client Default Options

- **Auto-increment new batch directory names.** (Unselected by default.) When activated, if you create a batch directory name that ends with a number, the next time you click Batch->New.. Batch Builder will use the same directory name but will increment the number to the next value.

- **Default to create directory structure for a new batch.** (Selected by default.) When checked the checkbox “create directories from batch template” in the “new batch” dialog box is checked by default. See “Create a batch” section for more information.

- **Enable METS file creation.** (Selected by default.) When activated, Batch Builder will create a simple PDS METS file during generation of a page-turned batch. De-select this option if you plan to supply an externally-created METS file.

- **Ignore file validation errors.** (Unselected by default.) When activated, this option forces Batch Builder to create a batch even though JHOVE validation has detected errors in one or more files in the batch.

  Note: this option should be activated when generating PDF batches (many PDFs do not strictly adhere to the standard and will generate errors if the option is unchecked).

- **Open last project on application startup.** (Unselected by default.) When activated, on Batch Builder startup the most recently used project will open automatically.

- **Save reports as XML.** (Selected by default.) When activated, Batch Builder will generate batch summary reports in text and xml formats. Reports are stored in the [project]/sync/[batch directory] folder on your file system.

Revised: March 18, 2010
- Validate contents of zip containers. (Selected by default.) Deselect this option to prevent Batch Builder from validating the contents of ZIP containers.

### 1.3 User interface basics

The Batch Builder interface includes a menu bar, a tool bar, and four display panes. In the following screen-shot of the interface, these components are labeled:

- **Message panel** displays informational and error messages as you use the client.
- **Configuration panel** displays components of the currently active project.
- **Content panel** displays metadata input forms and contents of reports that are selected in one of the left panels.
- **Project directory panel** displays components of the batch directories associated with this project.

Revised: March 18, 2010
Below are more details about the **project directory** panel.

Batch icon (blue B and orange x) indicates a batch.xml file has been generated

Red batch icon indicates a batch.xml file does not yet exist

Crossed out batch icon indicates deleted batch directory (caused by residual metadata about this batch in the report directory)

As you make changes to a project, generate batch directories and batch.xml files, you may need to refresh this panel in order to update the display of files and folders. To do this, from the main menu select View > Refresh file system panel.

### 1.4 Batch creation overview

The goal of using Batch Builder is to prepare a group of digital files for deposit to DRS. The output of Batch Builder is called a “batch”. A batch consists of:

- A set of “batch” directories on your local file system, populated with digital files.

- A batch control file (batch.xml), which describes the batch of files and instructs the DRS loader to perform these actions: add files to DRS, create URNs, and create relationships between deposited files.

- A PDS METS file (for page-turned object batches only). The METS file is a structural metadata file that identifies all the components of a document, describes its structure, and allows for page-turning navigation.

Once the batch is prepared, the deposit agent uses a secure FTP client to upload the batch directories and xml file(s) to a DRS drop box. The upload step is performed outside of Batch Builder.

Here is an overview of batch preparation using Batch Builder.

1. **Create a project.** A project is a collection of metadata and directory settings that can be used over and over to create batches with the same characteristics.
For example, if you deposit both image and page-turned batches, you would need to create separate projects for each. Or, in a high volume deposit operation that handles digital materials for many libraries, you might create a separate project for each DRS owner code you work with.

2. **Create a batch template** for the project. In this batch template you will define the batch directory structure and assign any global or directory-level metadata values that will be applied to digital files in the batch.

3. **Create batch directories** in the project. At this point, you provide a name for the batch and Batch Builder will generate the directory structure on your file system (based on the batch template). You then populate these directories with the digital files that will be part of this batch.

4. **Generate a batch control file.** A batch control file (batch.xml) is added to the top-level batch directory. To this file, Batch Builder adds batch-level administrative metadata as recorded in the project’s batch template. Batch Builder then traverses the batch directory structure and:

   - **Extracts technical metadata** from each digital file in the batch and adds this metadata to the batch.xml file. (For this activity, Batch Builder uses JHOVE - a format validation module developed by Harvard and JSTOR.)

     Note: when JHOVE finds a file to be invalid, it cannot extract the file’s technical metadata, which means the metadata properties required by Batch Builder cannot be calculated. If JHOVE is unable to extract technical metadata for some digital files in the batch, you can supply the missing values in the batch template and these will substitute for the metadata values that JHOVE cannot generate. See the “Set by JHOVE” portions of the Metadata Reference for more information.

   - **Determines administrative metadata** for each digital file based on the names and directory locations of the files and adds this metadata to the batch.xml file. See the Metadata Reference for more information.

   - **Determines an owner supplied name for each file**, either by using original file names or an external mapping file, and adds these to the batch.xml file. See About file names for more information.

   - **Creates an MD5 signature** for each file and adds this to the batch.xml file. The DRS loader uses the MD5 signature to verify that each transferred file arrives intact.

   - **Generates URN requests** for deliverable files in the batch and adds these to the batch.xml file. See Working with URNs for more information.

   - **Generates file relationship metadata** and adds this to the batch.xml file. Note: creation of relationship metadata is triggered by a nested batch directory structure and files with matching ownerSuppliedNames. See Expressing Relationships for more information.

   - **Generates a PDS METS file** (for page-turned batches only). The METS file (usually called mets.xml) will be added to the top level batch directory. Note: generation of

Revised: March 18, 2010
METS xml is optional and can be controlled by setting a Batch Builder option. See Creating a Page-Turned Object Batch for more information.

As the batch is processed, messages will display in the Batch Builder message window. For a successful batch, the final message will be:

INFO   - Creation of batch.xml complete for batch [batch name]

Batch Builder will also create a report that summarizes the results of the batch.xml generation process.

Any “Error” messages that appear usually indicate that generation of the batch.xml will fail. You will need to fix the errors and re-generate the batch.

Note: When generating a PDF batch check “Ignore validation errors” in the Options dialog – otherwise the batch may fail.

5. Upload the batch directories to a DRS dropbox. This step must be performed outside of Batch Builder. The depositor must use a secure FTP client to transfer the batch directory with its batch control file (and METS file if included) to the appropriate DRS drop box. See Uploading batches to DRS for more information.

1.5 About file names

This topic describes file naming requirements for digital files in batches handled by Batch Builder. Also included are descriptions of file names in DRS storage and the purpose of owner supplied name.

File names on disk

For all files in a batch:
• Maximum number of characters per file name is 100. OIS recommends that file names be kept to 64 characters or less, and that the complete_directory_path + file_name for each file be kept to 255 characters or less.

• Valid characters in file names are letters, digits, '.', underscores ('_'), and hyphens ('-').

Related digital files (e.g., a production master .tif file and its related deliverable .jpg file) can share the same file name (with a different format extension of course).

File names in DRS

The file name of a digital file will be changed once the file is in DRS storage. At the point of deposit, DRS assigns each file a numeric DRS identifier. The file name will be the DRS ID followed by the format extension (e.g., 5844020.tif, 5844022.jpg).

The original file name is preserved in DRS metadata as “supplied file name.” The DRS load report that is sent after a deposit is processed will associate the file’s DRS ID with the original file name. Also, if you use Batch Builder’s default method for assigning owner supplied name, the original file name (minus format extension) will be preserved in DRS metadata as the ownerSuppliedName.

Owner supplied name

Each file stored in the DRS must be assigned an owner supplied name (ownerSuppliedName is the name of the corresponding element in DRS batch metadata). This name serves as a unique identifier that links deposited files with local information about those files. This name can be some type of record ID (e.g., an OLIVIA record ID), a local accession number or other curatorially-significant name.

By default Batch Builder will use the names of the files on disk to determine the DRS ownerSuppliedName values. For example, an image file called image998.tif will be assigned an ownerSuppliedName of image998.

Within a DRS owner code, the owner supplied name must be unique. More than one digital file within an owner code can have the same owner-supplied name if the role/purpose/quality values for the files are different. For example, an archival TIFF and deliverable JPEG with the same base file name will be assigned the same owner supplied name by Batch Builder:

<table>
<thead>
<tr>
<th>File name on disk</th>
<th>Owner supplied name</th>
</tr>
</thead>
<tbody>
<tr>
<td>image998.tif</td>
<td>image998</td>
</tr>
<tr>
<td>image998.jp</td>
<td>image998</td>
</tr>
</tbody>
</table>

Note: Batch Builder can be configured to use an external ownerSuppliedName mapping file when the default method of assigning ownerSuppliedName based on file name on disk is not desired. See Using external mapping files for more information.

1.6 About directory names and structure

All files in a batch must be located within a set of batch directories on the local file system and this set of directories must be viewable and editable by Batch Builder. This topic describes Batch Builder requirements for batch directory structure and naming.
**Batch directory structure**

All files within a batch must be located in a batch directory. The top level of this directory should contain only the batch control file (batch.xml). For page-turned object batches, the top-level directory should also contain the PDS METS file.

Besides the batch.xml and METS file, no other file should be stored in the top-level batch directory. Batch Builder will ignore any other files at this level.

All other files in the batch must be located in an appropriate subdirectory under the top level directory. Batch subdirectories are named to indicate the role or type of file they contain (e.g., deliverable, archive_master, etc.). Batch subdirectory structure allows Batch Builder to:

- **Infer the type of file contained in the subdirectory.** For example, a subdirectory called deliverable is understood to contain files meant for delivery to users.

- **Interpret the relationships between files in the batch.** For example, a subdirectory called deliverable nested under a subdirectory called archival_master indicates to Batch Builder that the deliverable files were derived from the archival master files. If these subdirectories are parallel and not nested, no derivation is assumed. See Expressing Relationships for more information.

- **Assign user-supplied metadata.** In Batch Builder, a directory of files is the smallest unit at which the depositor can assign metadata.

Depending on the batch, some subdirectories will not be needed. For example, a page-turned batch without any OCR text can omit the ocr subdirectories. Batch Builder will ignore empty subdirectories.

A batch can contain more than one of the same type of batch subdirectory. For example, a batch can contain subdirectories named deliverable_1, deliverable_2 and deliverable_3. This can be useful for grouping together files with the same metadata requirements (e.g., the same quality or purpose value).

**Batch directory names**

For the top-level batch directory and all of its subdirectories:

- Maximum directory name length is 100 characters.

- Valid directory name characters are letters, numbers, underscores (‘_’) and hyphens (‘-’).

Batch Builder has no requirements for the name of the top-level batch directory but has very specific requirements for naming of its subdirectories.

Within the top-level batch directory, all digital files must be contained within subdirectories that are named to indicate their role or type. Batch subdirectory names must begin with a pre-defined prefix. The following table lists a few sample directory name prefixes and how files in these subdirectories will be interpreted by Batch Builder.

<table>
<thead>
<tr>
<th>Batch subdirectory name</th>
<th>Files in this directory …</th>
</tr>
</thead>
</table>

Revised: March 18, 2010
<table>
<thead>
<tr>
<th>archival_master</th>
<th>are images that will be given the role “ARCHIVAL_MASTER”</th>
</tr>
</thead>
<tbody>
<tr>
<td>production_master</td>
<td>are images that will be given the role “PRODUCTION_MASTER”</td>
</tr>
<tr>
<td>deliverable</td>
<td>are images that will be given the role “DELIVERABLE”</td>
</tr>
</tbody>
</table>

Consult the requirements of general image batches, page-turned object batches or container batches for detailed subdirectory naming requirements.

Classifying an image as an archival master, production master or deliverable is a local decision. These classifications currently have no bearing upon preservation services in the DRS. General practices to date have been to use the term archival master to designate the highest quality or least processed versions of images; to use production master to designate the images that have been optimized to generate deliverables, particularly through batch automation; and to use deliverable to designate the images optimized for rendering on given delivery systems (e.g., web browsers).

**Note:** Only the first portion (the prefix) of batch subdirectory names is prescribed. Batch Builder will allow the depositor to append additional information to subdirectory names as long as the entire name uses valid characters and does not exceed 100 characters in length. For example, these subdirectory names are valid:

- deliverable_1
- deliverable_screen
- deliverable_thumb

### 1.7 Batch preparation checklist

Depositors should consult this checklist as they prepare to use Batch Builder.

- **Prepare new depositors.** If you are new to the DRS batch deposit process, you may need to request one or more DRS drop box accounts and perform some test deposits. See [How to become a DRS Deposit Agent](#) on the OIS web site.

- **Determine project administrative metadata.** You will need to know the DRS owner code, DRS billing code, success and failure email addresses, URN authority path, and URN resource name pattern. These are part of administrative properties in Batch Builder. For more information, see [How to become a DRS Deposit Agent](#) in Batch Builder.

- **Understand DRS file naming options.** Consult [About file names](#) for information on how to name digital files on disk, what DRS does with file names of deposited files, and the purpose of owner supplied name.

- **Review batch workflows.** For a summary of batch creation using Batch Builder, see [Batch creation overview](#). See also detailed procedures for general image batches, page-turned object batches, container batches and PDF batches. Each of these batch genres has specific file name and batch directory requirements.

- **Check Batch Builder options.** There are a few options that impact every project you create in Batch Builder. See [Setting options](#) for more information.
1.8 Supporting documentation

This User Guide assumes that Batch Builder users are generally familiar with DRS, NRS and PDS operations. For more information, see the online documentation on the OIS web site:

DRS and PDS information:  http://hul.harvard.edu/ois/systems/drs/

NRS information:  http://hul.harvard.edu/ois/systems/nrs_ams/
2. Creating a General Image Batch

2.1 Accepted formats
2.2 File name rules
2.3 Batch directory rules
2.4 Procedure to create an image batch

A general image batch consists of:

- zero or more archival master images
- zero or more production master images (also called archival production images)
- zero or more delivery images
- zero or more external ICC profiles
- zero or more target files
- zero or more text files associated with target images

All files in the batch must share a single DRS owner code and a single DRS billing code.

Only the delivery images (files in a deliverable subdirectory) will be assigned a persistent identifier (also known as a URN or “Universal Resource Name”). NRS URNs can be added or modified after DRS deposit using the NRS Maintenance System.

2.1 Accepted formats

For digital files in general image batches, the following DRS-supported file formats can be used:

- GIF image files (file extension: gif)
- JPEG image files (file extension: jpg)
- JP2 image files (file extension: jp2)
- TIFF image files (file extension: tif or tiff)
- PhotoCD image files (file extension: pcd)

Note: although Batch Builder will accept photoCD files, the JHOVE module cannot automatically extract technical metadata for this format.

- Plain text files (file extension: txt or tdf)

  Only US-ASCII and UTF-8 encodings are supported

- ICC files (file extension: icm or icc)
The file extensions noted above are mandatory.

Any other file formats not listed above that are included in the batch will be ignored by Batch Builder. These ignored files will not be described anywhere in the generated batch.xml file, and they will not be deleted from the batch directory.

### 2.2 File name rules

This topic describes specific file name requirements for general image batches. See [About file names](#) for general Batch Builder requirements.

The file-naming scheme for files in general image batches is:

```
{ownerSuppliedName}_-_\{shortDescription\}.\{extension\}
```

where:

- `{ownerSuppliedName}` is the item control name that associates the digital file with its analog counterpart. This may be an OLIVIA ID, accession number or other curatorially-significant name. In the case of target files or ICC profiles this could be a locally-meaningful name, e.g. Adobe_RGB_1998, PrimeScan_PrimescanFujiNegColor or it8_20000714. Batch Builder will use this value for the batch.xml's ownerSuppliedName value (you can also define ownerSuppliedName using a [mapping file](#)). Valid characters to use for the `{ownerSuppliedName}` are letters, digits, '-', underscores ('_'), and hyphens ('-').

- `_-_` character sequence (one underscore, one hyphen, one underscore) is used to separate the `{ownerSuppliedName}` from the `{shortDescription}`, if the `{shortDescription}` is present.

- `{shortDescription}` is an optional locally-meaningful description of the file. This portion of the file name will not be used for any Batch Builder metadata generation. It is for depositors who want to embed information in the file names that is useful for their workflow like: thumb, prodarc, large, etc. Valid characters to use for the `{shortDescription}` are letters, digits, underscores ('_'), and hyphens ('-'). Do not use the character sequence `_-_` as part of the `{shortDescription}`.

- `{extension}` is one of the valid file extensions listed at the beginning of this topic.

### 2.3 Batch directory rules

This topic describes specific requirements for subdirectory names and structure in general image batches. See [About directory names and structure](#) for general Batch Builder requirements.

Within the top-level batch directory, all digital files must be contained within subdirectories that are named to indicate their role or type. Batch Builder requires that subdirectory names begin with a pre-defined prefix. The following table lists valid prefixes for general image batches and how files in these subdirectories will be interpreted by Batch Builder.

<table>
<thead>
<tr>
<th>Batch subdirectory name</th>
<th>Files in this directory …</th>
</tr>
</thead>
<tbody>
<tr>
<td>archival_master</td>
<td>are images that will be given the role “ARCHIVAL_MASTER”</td>
</tr>
<tr>
<td>production_master</td>
<td>are images that will be given the role “PRODUCTION_MASTER”</td>
</tr>
<tr>
<td>deliverable</td>
<td>are images that will be given the role “DELIVERABLE”</td>
</tr>
<tr>
<td>icc</td>
<td>are ICC profiles</td>
</tr>
</tbody>
</table>

Revised: March 18, 2010
As long as the first part of a batch subdirectory’s name follows the rules above, you can append additional information. Batch Builder would consider the following subdirectory names to be valid:

- archival_master-1
- archival_master_20061205
- archival_masterPCCR

Any extensions to a subdirectory’s name must contain only letters, numbers, underscores or hyphens and must not exceed 100 characters in length.

For the archival_master, production_master and deliverable directories, when a “parent” subdirectory A contains a “child” subdirectory B, Batch Builder will infer that the digital files in B are derived from the digital files in A.

When anicc or target directory is contained within one of the image subdirectories (archival_master, production_master, deliverable), Batch Builder will infer that the ICC and/or target files should be associated with all files in the image subdirectory. If the batch has text files that need to be associated with the target image files, put them in the target directory along with the target files.

Below is a sample batch directory for a general image batch that includes a deliverable file derived from a production master, which in turn was derived from an archival master. The archival master has a related target file with a related text file as well as a related ICC file. The production master also has a related ICC file.

```
C:\myBatches\batchDir\archival_master\1.tif
    target\target1.tif
    target1.tdf
    icc\sepia.icm
    production_master\1.jpg
    deliverable\1.jpg
    icc\redtint.icm
```

### 2.4 Procedure to create an image batch

Follow these steps to create a general image batch. The steps in Part I (Create the project and batch template) need only be done once. (A project is a template from which you can prepare multiple batches for DRS deposit.) If you have already created the project and batch template, proceed to Part II (Create a batch).

Note: to save project values at any point, from the main menu select **Project > Save.**
Part I: Create the project and batch template

1. From the Batch Builder main menu, select **Project > New**.

2. Complete the New Project form and click OK.

   ![New Project form for general image batches](image)

   **Project name**: this name displays in the Batch Builder title bar when the project is open. This value can be changed later by accessing the Project Properties node in the Configuration panel.

   **Project directory**: path and directory name in which project files are stored. To locate or create the directory, click the ellipses (…) button to browse your file system. This value cannot be changed once a project is created.

   Best practice note: new users should consider including the word “project” in their project directory names to make it easier to differentiate these from the underlying batch directories.

   **Batch genre**: select “generalimage” as the genre. This value cannot be changed once the project is created.

   **Project description**: an internal free text note field associated with the project. This value can be changed later by accessing the Project Properties node in the Configuration panel.

3. **Add administrative metadata**. In the Configuration panel on left, select Administrative Properties. This panel records project-level administrative metadata. This metadata will be used in every batch.xml file generated by this project.

   ![Administrative properties panel](image)

   **Administrative properties panel**
Mouse over any of the field labels to display a definition. Or, consult Administrative properties in the Metadata Reference section for more information.

An asterisk indicates required fields. Batch Builder will provide warnings in the message pane if required fields are missing, but does not validate the contents of these fields. For example, if you supply an invalid Owner Code, this value will be added to the batch.xml file and at deposit, the entire batch will be rejected by DRS.

4. [Optional] Add global metadata that will be applied to every digital file in batches created by this project. In the Configuration panel, select Global Properties under the Batch Template node.

Consult Global properties in the Metadata Reference section for more information.

If you plan to accept the JHOVE-supplied metadata and do not need to include optional global metadata, skip this step.

5. Add batch directories to the template. In the Configuration panel, right click on Batch Template and select Add directory. In the pop-up window, choose a directory name from the list. Directory name choices on the list are determined by the batch genre.

[Optional] You can modify a directory name by appending custom text after the predefined name. Type the custom text in the box to the right of the directory list.

Click OK to add the directory to the template. Repeat this step to add all directories needed for this project.

If you plan on defining file relationships in batches created by this project, the directories that you add must be nested. For example, to nest a deliverable directory under an archival_master directory, create the archival_master directory first, right
click on it, select Add directory and select deliverable. See Expressing Relationships for more information.

6. [Optional] **Add directory-specific metadata** that will be applied to every digital file in a specific batch directory. In the Configuration panel, select a directory to display its metadata properties.

![Directory-specific metadata properties](image)

Consult Directory-level properties in the Metadata Reference section for more information on this metadata.

If you plan to accept the administrative metadata inferred by Batch Builder and the technical metadata supplied by JHOVE, skip this step.

**Part II: Create a batch**

7. **Create the batch directory and subdirectories.** From the main menu, select **Batch > New**. In the New Batch window, supply a top-level directory name.

![Creating batch directories on disk](image)

Batch Builder has no specific requirements for name of the top-level batch directory, but the name must be no longer than 100 characters and must consist only of letters, numbers, underscores ("_") and hyphens ("-").

By default, the ‘Create directories from batch template’ checkbox will be selected (indicating that Batch Builder will create subdirectories based on directory names defined in the batch template). De-select this option if you want to create the directories outside of Batch Builder.

8. In the Project Directory pane, an entry for your new batch will appear, flagged with a red b.
9. **Move digital files into the batch directories.** This step must be performed outside of Batch Builder.

   In Batch Builder, you will be able to see these files in the Project Directory panel if you refresh the display. From the main menu, select View > Refresh file system panel.

10. **Generate the batch xml file.** Right click on the batch and select “Create batch.xml file” (or from the main menu, select Batch > Create batch.xml file).

    As Batch Builder processes the batch, status messages will display in the message window. If batch generation is successful, the final message will be:

    INFO   - Creation of batch.xml complete for batch image-batch-1

    In the Project Directory pane, the “b” icon next to the batch directory changes to a green b with an orange x to indicate the batch xml file has been generated.

    If any required metadata is missing, or if any digital files are invalid, Batch Builder will display an error message and the batch.xml file will not be generated. You will need to fix the errors and re-generate the batch.

11. [Optional] **Review the summary report** for this batch.xml generation process. You can view this report by selecting the batch in the ‘project directory’ panel and clicking the Reports node.

    Note that on your file system, this report is saved in the Sync directory for the project and not in the batch directory:

    [project name]/sync/[batch name]/reports/

    The summary report is not deposited to the DRS, so must not be placed within a batch directory.

Once the batch.xml file has been successfully generated, you are ready to upload the batch to DRS. See [Uploading batches to DRS](#) for more information.
3. Creating a Page-Turned Batch

3.1 Accepted formats
3.2 File name rules
3.3 Batch directory rules
3.4 About the PDS METS file
3.5 Procedure to create a page-turned batch

A page-turned object batch consists of at least:

- A single PDS METS file (a structural metadata file that identifies all the components of a document, describes its structure, and allows for page-turning navigation).

- One or more page image files.

- Zero or more text file-versions of the page images.

- Zero or more external ICC profiles

- Zero or more target files

- Zero or more text files associated with target images

In some cases, there are multiple images per page, one of which is an archival image, and one or more delivery images derived from the archival image in other formats. In other cases, there will be a single image file per page. For example, the Page Delivery Service will create delivery page images from JPEG2000 or TIFF masters. For detailed PDS file requirements, consult the PDS section of the OIS web site.

When Batch Builder is set to auto-generate a METS file for a PDS batch (see Procedure to create a page-turned batch) it presumes that all files in this batch make up a single page-turned object. However, if existing METS file is supplied (the Batch Builder auto-generate METS feature is turned off) it is possible to include more than one PDS object per Batch Builder batch. In this case images and text of these PDS objects will inhabit the same batch directories as when a single PDS object is produced. Several METS files will be included in the top batch directory. A batch generated in this way can be deposited to DRS resulting in a single batch deposit of several PDS objects.

All the files in a page-turned batch need to share a single DRS owner code and a single DRS billing code.

During the batch generation process, Batch Builder will request a persistent ID (URN) based on the selected batch genre. If the “Page-turned Object” genre is selected, Batch Builder will request a URN for the METS file only. This URN will resolve to the citation (top) level of the document. If the “Page-turned Object – IDS URN for each image” genre is selected, in addition to a URN for the METS file an IDS URN will be requested for every deliverable image in the batch.

Revised: March 18, 2010
3.1 Accepted formats

For page-turned objects, the following DRS-supported file formats can be used:

- GIF image files (file extension: gif)
- JPEG image files (file extension: jpg)
- JP2 image files (file extension: jp2)
- TIFF image files (file extension: tif or tiff)
- Plain text files (file extension: txt)

Only US-ASCII and UTF-8 encodings are supported.

- XML files (file extension: xml)

Only accepted in the case of the PDS METS file. There can be at most one XML file in the batch and this file must be located in the top-level batch directory.

The file extensions noted above are mandatory.

3.2 File name rules

This topic describes specific file name requirements for page-turned object batches. See About file names for general Batch Builder requirements.

METS file naming

If Batch Builder is configured to automatically generate the METS file, the file naming scheme will be:

{project name}_{batch directory name}_mets.xml

where:

{project name} is the name assigned to the project, as it appears in the Project Properties pane.

{batch directory name} is the name of the top level batch directory, as it appears in the Project Directory pane.

If the METS file is supplied by the depositor, the file name can be any locally meaningful name that follows Batch Builder name rules (only letters, numbers, underscores, hyphens and no more than 100 characters long). The file extension must be .xml.

Like all other files deposited to DRS, the METS file must have an owner supplied name that is unique within an owner’s collection (within a DRS owner code). By default, Batch Builder will generate an owner supplied name for the METS file from its file name on disk. If for some reason the METS file name on disk will not yield a unique owner supplied name, you can use a mapping file to assign a unique owner supplied name to the METS file.

Revised: March 18, 2010
Page image and page text file naming

The file-naming scheme for page image and text files in page-turned object batches is:

{ownerSuppliedName}{separator}{sequenceNumber}.{extension}

or

{ownerSuppliedName}{sequenceNumber}.{extension}

where:

{ownerSuppliedName} is the item control name that associates the digital file with its analog counterpart. This may be an accession number or other curatorially-significant name. Batch Builder will use this value for the batch.xml's ownerSuppliedName value. Valid characters to use for the {ownerSuppliedName} are letters, digits, ',', underscores ('_'), and hyphens ('-').

{separator} is a dash(-) or underscore (_) used to separate the {ownerSuppliedName} from the page sequence number. A dash or underscore is optional if the last character in the ownerSuppliedName is a letter and not a number.

{sequenceNumber} is the numeric value that represents the sequence number of the page within the page-turned document. A sequenceNumber can be composed of any of the following characters: 0123456789.

The sequence number can include leading zeros, for example the third page can be written as: 3 or 03 or 000000003. The page sequence number indicates a page's relative position within a sequence of pages, regardless of the numbering that may appear on the page. If page sequence numbers are supplied in a separate page naming file, this separator and the {sequenceNumber} can be omitted.

{extension} is one of the valid file extensions at the beginning of this topic.

Batch Builder will be able to identify page sequence “1” in these file names:

  page1.tif
  page-1.jpg
  page_1.txt
  page_-_1.tif
  9876page1.jp2
  1980a1.txt
  2005-1.jp2

Batch Builder will not be able to identify page sequence “1” in these file names:

  2001e551.tif
  87051.jpg

Batch Builder will use the file name on disk (minus extension) for the page-turned object’s ownerSuppliedName value in the batch.xml (you can also define ownerSuppliedName using a mapping file).

Revised: March 18, 2010
### 3.3 Batch directory rules

This topic describes specific requirements for subdirectory names and structure in page-turned object batches. See About directory names and structure for general Batch Builder requirements.

Within the top-level batch directory, all digital files must be contained within subdirectories that are named to indicate their role or type. Batch subdirectory names must begin with a pre-defined prefix. The following table lists valid prefixes for page-turned object batches and how files in these subdirectories will be interpreted by Batch Builder.

<table>
<thead>
<tr>
<th>Batch subdirectory name</th>
<th>Files in this directory …</th>
</tr>
</thead>
<tbody>
<tr>
<td>archival_master</td>
<td>are images that will be given the role “ARCHIVAL_MASTER”</td>
</tr>
<tr>
<td>production_master</td>
<td>are images that will be given the role “PRODUCTION_MASTER”</td>
</tr>
<tr>
<td>deliverable</td>
<td>are images that will be given the role “DELIVERABLE”</td>
</tr>
<tr>
<td>ocr_corrected</td>
<td>are corrected OCR text files of page images</td>
</tr>
<tr>
<td>ocr_uncorrected</td>
<td>are uncorrected OCR text files of page images</td>
</tr>
<tr>
<td>keyed_text</td>
<td>are keyed text files of page images</td>
</tr>
<tr>
<td>icc</td>
<td>are ICC profiles</td>
</tr>
<tr>
<td>target</td>
<td>are either target image files or text files associated with target image files</td>
</tr>
</tbody>
</table>

Only the first portion (the prefix) of batch subdirectory names is prescribed. Batch Builder will allow the depositor to append additional information to subdirectory names as long as the entire name uses valid characters and does not exceed 100 characters in length. Batch Builder would consider the following subdirectory names to be valid:

```
archival_master-1
archival_master_20061205
archival_masterPCCR
```

When an ocr_uncorrected, ocr_corrected or keyed_text subdirectory is present in the batch directory, Batch Builder will infer that it contains text files that should be associated with page images.

Below is a sample batch directory for a page-turned batch that includes a PDS METS file, two archival master files, two deliverable files, and two uncorrected OCR files.

```
C:\myBatches\batchDir\mets.xml
archival_master\U123-1.tif
archival_master\U123-2.tif
deliverable\U123-1.jpg
deliverable\U123-2.jpg
ocr_uncorrected\U123-1.txt
ocr_uncorrected\U123-2.txt
```
3.4 About the PDS METS file

The PDS METS file is a structural metadata file that identifies all the components of a page-turned document, describes its structure, and allows for page-turning navigation. A page-turned object batch must contain one METS file (Batch Builder presumes that all files in a batch make up a single page-turned object). The METS file must be located in the top-level batch directory.

Creation of the PDS METS file is optional in Batch Builder. By default, Batch Builder will automatically generate a minimal METS file.

The depositor can turn off the automatic METS generation option within Batch Builder and supply an externally-created METS file. To turn off automatic generation, from the Batch Builder main menu select View > Options and de-select the option “Enable METS file creation”. See the METS file naming section for the rules on file names.

The minimal METS file generated by Batch Builder will describe a simple document (with a citation node and page nodes). It will provide page sequence numbers and the option to include a HOLLIS system number for the document. PDS will use the HOLLIS system number to extract some basic citation metadata (e.g., author, title) from the HOLLIS Catalog.

The minimal METS file generated by Batch Builder will NOT include:

- Physical page numbers or page labels.
- Intermediate (section) nodes or labels.
- Custom citation-level metadata settings for PDF header text, related links or show/hide settings for “Go To” and “View Text” navigation options.

These values can be added after DRS deposit using the PDS Maintenance System.

3.5 Procedure to create a page-turned batch

Follow these steps to create a page-turned object batch. The steps in Part I (Create the project and batch template) need only be done once. (A project is a template from which you can prepare multiple batches for DRS deposit.) If you have already created the project and batch template, proceed to Part II (Create a batch).

Note: to save project values at any point, from the main menu select Project > Save.

Part I: Create the project and batch template

1. Decide on PDS METS file generation. Options are to allow Batch Builder to generate the PDS METS file (the default) or to create the METS file externally and copy it into the top-level batch directory before generating the batch.xml file.

   To deactivate auto-generation of the METS file, from the main menu select View > Options and de-select the option “Enable METS file creation”. This option needs to be set before you generate the batch.xml.

2. From the Batch Builder main menu, select Project > New.

Revised: March 18, 2010
3. Complete the New Project form and click OK.

![New Project form for page-turned batches]

- **Project name:** this name displays in the Batch Builder title bar when the project is open. This value can be changed later by accessing the Project properties node in the Configuration panel.

- **Project directory:** path and directory name in which project files are stored. To locate or create the directory, click the ellipses (…) button to browse your file system. This value cannot be changed once a project is created.

  Best practice note: new users should consider including the word “project” in their project directory names to make it easier to differentiate these from the underlying batch directories.

- **Batch genre:**
  - Select “Page-turned Object” if you want Batch Builder to request a citation (top) level URN only.
  - Select “Page-turned Object – IDS URN for each image” if you want Batch Builder to request a URN for each deliverable image in the batch.

  This value cannot be changed once the project is created.

- **Project description:** an internal free text note field associated with the project. This value can be changed later by accessing the Project properties node in the Configuration panel.

4. **Add administrative metadata.** In the Configuration panel on left, select Administrative Properties. This panel records project-level administrative metadata. This metadata will be used in every batch.xml file generated by this project.

Revised: March 18, 2010
Administrative properties panel

Mouse over any of the field labels to display a definition. Or, consult Administrative properties in the Metadata Reference section for more information.

An asterisk indicates required fields. Batch Builder will provide warnings in the message pane if required fields are missing, but does not validate the contents of these fields. For example, if you supply an invalid Owner Code, this value will be added to the batch.xml file and at deposit, the entire batch will be rejected by DRS.

5. [Optional] **Add global metadata** that will be applied to every digital file in batches created by this project. In the Configuration panel, select Global Properties under the Batch Template node. Add metadata values as needed.

Global properties panel

Consult Global properties in the Metadata Reference section for more information.

If you plan to accept the JHOVE-supplied metadata and do not need to include optional global metadata, skip this step.

6. **Add batch directories to the template.** In the Configuration panel, right click on Batch Template and select “Add Directory”. In the pop-up window, choose a directory name from the list. Directory name choices on the list are determined by the batch genre.
[Optional] You can modify a directory name by appending custom text after the predefined name. Type the custom text in the box to the right of the directory list.

Click OK to add the directory to the template. Repeat this step to add all directories needed for this project.

If you plan on defining file relationships in batches created by this project, the directories that you add must be nested. For example, to nest a deliverable directory under an archival_master directory, create the archival_master directory first, right click on it, select Add Directory and select “deliverable”. See Expressing Relationships for more information.

7. [Optional] **Add directory-specific metadata** that will be applied to every digital file in a specific batch directory. In the Configuration panel, select a directory to display its metadata properties.

Directory-specific metadata properties

Consult [Directory-level properties](#) in the Metadata Reference section for more information on this metadata.

If you plan to accept the administrative metadata inferred by Batch Builder and the technical metadata supplied by JHOVE, skip this step.

Part II: Create a batch

8. **Create the batch directory and subdirectories.** From the main menu, select **Batch > New**. In the New Batch window, supply a top-level directory name.
Creating batch directories on disk

Batch Builder has no specific requirements for name of the top-level batch directory, but the name must be no longer than 100 characters and must consist only of letters, numbers, underscores ('_') and hyphens ('-').

By default, the “Create directories from batch template” checkbox will be selected (indicating that Batch Builder will create subdirectories based on directory names defined in the batch template). De-select this option if you want to create the directories outside of Batch Builder.

9. In the ‘Project Directory’ pane, an entry for your new batch will appear, flagged with a red b.

The red b indicates that a batch.xml file has not yet been generated for this batch directory.

10. **Move digital files into the batch directories.** This step must be performed outside of Batch Builder.

    In Batch Builder, you will be able to see these files in the “Project Directory panel if you refresh the display. From the main menu, select View > Refresh file system panel.

11. **Generate the batch xml file.** Right click on the batch and select “Create batch.xml file” (or from the main menu, select Batch > Create batch.xml file).

    You will be prompted to supply a HOLLIS ID (the HOLLIS system number of the cataloging record that describes the page turned object). PDS will use the HOLLIS system number to extract some descriptive metadata (e.g., author, title) from the HOLLIS Catalog.
Adding a HOLLIS system number

Enter the number and click OK. Or press Cancel to skip this step. If you opt not to add a HOLLIS system number at deposit, you can add it after deposit using the PDS Maintenance System.

Batch Builder will start processing the batch. Status messages will display in the message window as processing proceeds. If batch generation is successful, the final message will be:

INFO - Creation of batch.xml complete for batch pagebatch-20070130a

In the ‘Project Directory’ pane, the ‘b’ icon next to the batch directory changes to a green b with an orange x to indicate the batch xml file has been generated.

If any required metadata is missing, or if any digital files are invalid, Batch Builder will display an error message and the batch.xml file will not be generated. You will need to fix the errors and re-generate the batch.

12. [Optional] Review the summary report for this batch.xml generation process. You can view this report by selecting the batch in the Project Directory panel and clicking the Reports node.

Note that on your file system, this report is saved in the Sync directory for the project and not in the batch directory:

[project name]/sync/[batch name]/reports/

The summary report is not deposited to DRS, so must not be placed within a batch directory.

Once the batch.xml and mets.xml have been successfully generated, you are ready to upload the batch to DRS. See Uploading batches to DRS for more information.
4. Creating a Container Batch

4.1 Internal ZIP file details: Dark PDS document containers
4.2 Internal ZIP file details: Opaque containers
4.3 ZIP file accepted formats
4.4 ZIP file naming rules
4.5 ZIP file size
4.6 ZIP container batch directory rules
4.7 Procedure to create a container batch
4.8 Brightening dark PDS document containers

A container batch can consist of:

- A single ZIP container file
- Multiple ZIP container files

Two types of container batches can be created:

A “dark PDS document” container batch – for the purpose of depositing a page-turned object as a “dark” object (an object not yet accessible to users). Such dark objects can later be retrieved from DRS and “brightened” (made accessible) when needed. See the section Brightening dark PDS document containers for more information.

An “opaque” container batch – for the purpose of storing in DRS groups of files in formats that are currently not explicitly supported by DRS or/and cannot be adequately processed locally (assessed or appraised, cataloged or described, etc.) in advance of deposit.

The ZIP files in a container batch are not assigned URNs. Although URNs are not assigned to ZIP containers, as long as the restriction flag is set to ‘R’ or ‘P’ these can be delivered by the new File Delivery Service (FDS) using FDS URLs with the following syntax:

http://fds.lib.harvard.edu/fds/deliver/[DRS ID],
e.g: http://fds.lib.harvard.edu/fds/deliver/13790530/

See the OIS web site for more about FDS.

Important note: Contact the DRS support team before starting a project involving container batches. Container batches often require more guidance than other types of batches.

4.1 Internal ZIP file details -- Dark PDS document containers

A dark PDS Document ZIP file holds the directory structure and files for a PDS page-turned object batch. The format and structure of the files within the ZIP container must follow the guidelines for page-turned objects as described in Creating a Page-Turned Batch. This means using batch directory structure and file naming practices similar to the following example:

/deliverable/page_1.jp2
/deliverable/page_2.jp2
/deliverable/ . . .
Using Batch Builder to generate the page-turned object batch (before zipping it) is recommended, but not required. If you do use Batch Builder for this purpose, the resulting batch.xml can be discarded. The resulting mets.xml file should be included in the ZIP container if it contains descriptive or structural metadata.

4.2 Internal ZIP file details -- Opaque containers

This section describes the requirements for content of an opaque container ZIP file. Topics in this section include:

- File and directory structure
- Content files
- DRS documentation files

File and directory structure

Opaque containers can contain two types of files – content files (those files being deposited for preservation), and DRS documentation files. See Content files and DRS Documentation Files for more information.

Before zipping the files up, the files should be arranged under one of two directories: either content or documentation. The DRS documentation files should be placed in the documentation directory, all other files should be placed in the data directory.

Example 1: Contents of an opaque container ZIP file storing a presentation

```
content\the_presentation.ppt
content\the_presentation_audio.mp3
content\the_presentation_video.mpeg
content\the_presentation_handout.pdf
content\the_presentation_notes.doc
content\drs_documentation.txt
```

Example 2: Contents of an opaque container ZIP file storing a faculty member's research

```
content\projectx\2005_002.xls
content\projectx\runDemo.exe
content\projectx\logo.bmp
content\projecty\docs\original_grant.pdf
content\projecty\images\logo.bmp
content\misc.pdf
content\temp\test1.jpg
documentation\drs_documentation.pdf
documentation\drs_documentation_license.odt
documentation\drs_documentation_original_order.ods
```
Content files

To the extent possible, care should be taken to only include in a single opaque container those files that logically make up an object.

Try to include within the same opaque container, files that:

- Are related by derivative relationships (e.g. master and use copies)
- Have display or rendering dependencies (e.g. style sheets, scripts and images should be included in the same opaque container as web pages dependent on them)
- Require the same descriptive metadata (are part of the same work with the same bibliographic record)

Deposit in separate opaque containers:

- Files which are not related or dependent on each other contextually or structurally
- Files that constitute several independent logical objects

Omit from opaque containers:

- Content not intended for long-term preservation

Although it is expected that the content of opaque containers might not be fully appraised or processed prior to deposit into the DRS, to the extent possible depositors should perform an initial weeding out of unwanted material. This is especially important in the case of donated hard drives which may contain applications, system files, cached data, etc.

<table>
<thead>
<tr>
<th>Examples of opaque container designs</th>
<th>Assessment of design</th>
</tr>
</thead>
<tbody>
<tr>
<td>A ZIP file containing one video game object (multiple files related to a single game)</td>
<td>Very good. If the DRS ever supports video game objects the files in this ZIP file could be expanded into a fully supported object.</td>
</tr>
<tr>
<td>A ZIP file containing one database object (multiple files related to a single database)</td>
<td>Very good. Once the DRS supports database objects the files in this ZIP file could be expanded into a fully supported object.</td>
</tr>
<tr>
<td>A ZIP file containing one set of RAW camera image files</td>
<td>OK. If you can envision separating them in the future it would be better to only group the ones together that are related to each other. Also if there are other files related to these RAW files (derivative images, etc.) it may be better to group them together so that you don't have to combine them in the future.</td>
</tr>
<tr>
<td>A ZIP file containing one set of word processing files</td>
<td>OK. Same reason as for the RAW camera image files.</td>
</tr>
<tr>
<td>A ZIP file containing one audio file</td>
<td>Good. If there are other files related to this audio file (other derivative audio files, transcripts, etc.) it may be better to group them together so that you don't have to combine them in the future.</td>
</tr>
</tbody>
</table>
DRS documentation files

OIS encourages the creation and deposit of documentation within opaque container ZIP files to increase the chances that this content can be migrated into fully supportable DRS objects in the future, and so that all information known about this content can be made available to future stewards of this material.

The file names of all documentation files should start with the prefix drs_documentation. The files should be located within a directory called documentation, as shown in the File and directory structure topic. These documentation files can be one or more text, PDF or spreadsheet files. The recommended formats for these files are:

- If text: plain text (ASCII or UTF-8 encoding) (*.txt) or OpenDocument Text (*.odt)
- If PDF: PDF/A-1a (*.pdf)
- If spreadsheet: OpenDocument Spreadsheet (*.ods)

These files may include information about file relationships within an opaque container, documentation of original order, licenses or donor agreements, etc. Files describing original order or rights should be named according to the following guidelines.

Documenting original order

In keeping with archival practice, depositors are encouraged to document the original order and structure of files as they are submitted to them, especially when large file sets are split into separate opaque containers because of the size restriction (5 GB per ZIP file). The DRS documentation file that documents the original order should conform to the recommended formats listed above, and be deposited with all relevant opaque containers (those split from the original order). The file names of these files should start with the prefix drs_documentation_original_order.

Documenting rights

Donor agreements and other licenses should conform to the DRS documentation files recommended formats, and be stored with the relevant opaque container(s). The file names of these files should start with the prefix drs_documentation_license.

4.3 ZIP file accepted formats

For opaque and dark container batches, the accepted format at present is ZIP format (a valid ZIP file with a .zip extension).

4.4 ZIP file naming rules

This topic describes specific file name requirements for ZIP container file. See About file names for general Batch Builder requirements.

The file-naming scheme for zip files in container batches is:
{ownerSuppliedName}₋₋{shortDescription}.{extension}

or

{ownerSuppliedName}.{extension}

where:

{ownerSuppliedName} is the item control name that associates the digital file with its analog counterpart. This may be an accession number or other curatorially-significant name. Batch Builder will use this value for the batch.xml's ownerSuppliedName value. Valid characters to use for the {ownerSuppliedName} are letters, digits, ',', underscores ('_'), and hyphens ('-').

₋₋ character sequence (one underscore, one hyphen, one underscore) is used to separate the {ownerSuppliedName} from the {shortDescription}, if the {shortDescription} is present. You can also define ownerSuppliedName using a mapping file. See 6.1 Owner supplied name mapping file in DRS Batch Builder User Guide for details on creating a mapping file.

OIS recommends that file names be kept to 64 characters or less for the ZIP file.

4.5 ZIP file size

The size of each ZIP file is limited to 4GB or less.

If depositors must split related content into separate opaque containers because of this restriction, it is recommended that an original order documentation file be included with each opaque container. See DRS documentation files for more information on original order documentation files.

4.6 ZIP container batch directory rules

These are the specific requirements for subdirectory names and structure in dark PDS document and opaque container batches. See About directory names and structure for general Batch Builder requirements.

Within the top-level batch directory, the container zip file must be stored within a subdirectory named with the prefix “container”. Here is an example of a Batch Builder project directory path for a single container object (zip file):

\projectDirectory\batchDirectory\container\book.zip

When a container subdirectory is present under the batch directory, Batch Builder will infer that it contains ZIP files and will assign the role “container” to those ZIP files.

A single “container” subdirectory can contain more than one ZIP file. And, there can be more than one subdirectory with prefix “container” in a single container batch.

Only the first portion (the prefix) of batch subdirectory names is prescribed. Batch Builder will allow the depositor to append additional information to subdirectory names as long as the entire name uses valid characters and does not exceed 100 characters in length. Batch Builder would consider the following subdirectory names to be valid:

Revised: March 18, 2010
Below is a sample project directory structure for a dark PDS document container batch. The batch directory (“batch1”) contains two container subdirectories; the first (“container-series1”) holds two ZIP files while the other (“container-series2”) holds one ZIP file.

```
\projectDirectory\batch1\container-series1\        \container-series2\          
  \  \    \   
  book1.zip  book3.zip
  book2.zip
```

### 4.7 Procedure to create a container batch

Follow these steps to create a container batch. The steps in Part I (Create the project and batch template) need only be done once per project (or once per project workflow if a project has several different workflows). A project is a template from which you can prepare multiple batches for DRS deposit. If you have already created the project and batch template, proceed to Part II (Create a batch).

Note: to save project values at any point, from the main menu select **Project > Save**.

**Part I: Create the project and batch template**

1. **Decide on validation of ZIP container contents.** By default, Batch Builder will validate the contents of the ZIP container and report errors related to the files.

   To deactivate ZIP container validation, from the main menu select **View > Options** and de-select the option “Validate contents of zip container objects”. This option needs to be set before you generate the batch.xml.

2. From the Batch Builder main menu, select **Project > New**.

3. Complete the New Project form and click OK.
Project name: this name displays in the Batch Builder title bar when the project is open. This value can be changed later by accessing the Project properties node in the Configuration panel.

Project directory: path and directory name in which project files are stored. To locate or create the directory, click the ellipses (...) button to browse your file system. This value cannot be changed once a project is created.

Batch genre: select ‘Container’. This value cannot be changed once the project is created.

Project description: an internal free text note field associated with the project. This value can be changed later by accessing the Project properties node in the Configuration panel.

4. Add administrative metadata. In the Configuration panel on left, select Administrative Properties. This panel records project-level administrative metadata. This metadata will be used in every batch.xml file generated by this project.

Administrative properties panel

Mouse over any of the field labels to display a definition. Or, consult Administrative properties in the Metadata Reference section for more information.

An asterisk indicates required fields. Batch Builder will provide warnings in the message pane if required fields are missing, but will not validate the contents of these fields. For

Revised: March 18, 2010
example, if you supply an invalid Owner Code, this value will be added to the batch.xml file and, at deposit, the entire batch will be rejected by DRS.

**Note for container batches:** Although a container batch will not be assigned a URN, Batch Builder requires that all batch genres contain values for URN authority path and URN resource name pattern in their administrative metadata.

5. **[Optional] Add global metadata** that will be applied to every digital file in batches created by this project. In the Configuration panel, select Global Properties under the Batch Template node. Add metadata values as needed.

6. **Add batch directories to the template.** In the Configuration panel, right click on Batch Template and select "Add Directory". In the pop-up window, choose a directory name from the list. For the container batches the only choice will be “container”.

[Optional] You can modify a directory name by appending custom text after the predefined name. Type the custom text in the box to the right of the directory list.

Click OK to add the directory to the template. Repeat this step if you need additional “container” directories for the project.

Revised: March 18, 2010
7. **Add directory-specific metadata** that will be applied to every digital file in a specific batch directory. In the Configuration panel, select a directory to display its metadata properties.

For container directories, these metadata values are required:

- Role: container (set by default)
- Purpose: NA (set by default)
- Quality: NA (set by default)
- Access Flag: N (set by default)
- Usage Class: LOWUSE (set by default)
- Profile: “Opaque” or “Kress” (set by user). The rest of the profiles “Google” and “WAX” are for internal OIS use.
- Use Opaque for opaque containers, Kress for Dark PDS Documents.
- Version: 1.0 (set by user)

Values for Profile and Version must be set by the operator for each individual “container” subdirectory defined in the project.
Part II: Create a batch

8. **Create the batch directory and subdirectories.** From the main menu, select **Batch > New.** In the New Batch window, supply a top-level directory name.

   ![New Batch Window](image)

   **Creating batch directories on disk**

   Batch Builder has no specific requirements for name of the top-level batch directory, but the name must be no longer than 100 characters and must consist only of letters, numbers, underscores ('_') and hyphens ('-').

   By default, the "Create directories from batch template" checkbox will be selected (indicating that Batch Builder will create subdirectories based on directory names defined in the batch template). De-select this option if you want to create the directories outside of Batch Builder.

9. In the 'Project Directory' pane, an entry for your new batch will appear, flagged with a red b.

   ![Project Directory Pane](image)

   The red b indicates that a batch.xml file has not yet been generated for this batch directory.

10. **Move digital files into the batch directories.** This step must be performed outside of Batch Builder.

    In Batch Builder, you will be able to see these files in the "Project Directory" panel if you refresh the display. From the main menu, select View > Refresh file system panel.

11. **Generate the batch xml file.** Right click on the batch and select "Create batch.xml file" (or from the main menu, select Batch > Create batch.xml file).
Batch Builder will start processing the batch. Status messages will display in the message window as processing proceeds. If batch generation is successful, the final message will be:

INFO   - batch.xml written to:  
C:\BatchBuilder\ProjContainerTest\container-batch1\batch.xml

FINISHED - Creation of batch.xml complete for batch:  
container-batch1

In the `Project Directory' pane, the `b' icon next to the batch directory changes to a green b with an orange x to indicate the batch xml file has been generated.

If any required metadata is missing, or if any digital files are invalid, Batch Builder will display an error message and the batch.xml file will not be generated. You will need to fix the errors and re-generate the batch.

12. [Optional] **Review the summary report** for this batch.xml generation process. You can view this report by selecting the batch in the Project Directory panel and clicking the Reports node.

Note that on your file system, this report is saved in the Sync directory for the project and not in the batch directory:

[project name]\sync\[batch name]\reports\ 

Once the batch.xml has been successfully generated, you are ready to upload the batch to DRS. See 4 Uploading batches to DRS in DRS Batch Builder User Guide for more information.

Once the batch.xml has been successfully generated, you are ready to upload the container batch to DRS. See Uploading batches to DRS for more information.
4.8 Brightening dark PDS document containers

Follow this procedure to brighten (make accessible to users) a dark PDS document container stored in the DRS.

1. Retrieve the container from DRS using DRS WebAdmin.

2. Unzip the container and extract its contents to a location on your file system.

   The recommended way to do this is to create a directory that will serve as your Batch Builder project directory and copy the container’s directories into this project directory.

   Below is a sample structure of an unzipped dark PDS document batch.

```plaintext
C:\myProject\batchDir\  
mets.xml
archival_master\  
   U123-1.tif  
   U123-2.tif
deliverable\  
   U123-1.jpg  
   U123-2.jpg
ocr_uncorrected\  
   U123-1.txt  
   U123-2.txt
```

Structure of unzipped batch directory

3. Use Batch Builder to process the page-turned files into a batch for DRS deposit. For details about the steps that follow, consult Procedure to create a page-turned batch.

   a. Select Project > New to create a new project. Enter a project name of your choice. For project directory, navigate your file system and select the project directory created in step 2. Select a page-turned object batch genre. Click OK to continue.

   b. Decide on METS file generation. By default, Batch Builder is configured to generate a simple METS file. If you will be using an externally-created METS file, go to View > Options in Batch Builder and turn off the “Enable METS file creation” option before generating the batch.

      If it is not already there, remember to copy the external METS file into the top level batch directory before generating the batch.

   c. Select Administrative Properties from the Configuration panel and enter or modify administrative metadata for the batch.

   d. Define batch directories in the template that mimic the directory structure of the unzipped container object. Add global and directory-level metadata values as needed.

Revised: March 18, 2010
e. **Generate the batch xml file.** Right click on the batch and select “Create batch.xml file” (or from the main menu, select Batch > Create batch.xml file).

Supply a HOLLIS ID when prompted, or click Cancel to skip this step. HOLLIS ID (if supplied) will be stored in the METS file and used by PDS to extract some descriptive metadata (e.g., author, title) from the HOLLIS Catalog.

f. Once the batch is generated, use a secure FTP client to upload the batch directories to a DRS dropbox. Once you disconnect the SFTP client from the dropbox, the DRS loader will queue the batch for processing.

5. **Creating a PDF Batch**

5.1 **Accepted formats**
5.2 **File name rules**
5.3 **Batch directory rules**
5.4 **Procedure to create a PDF batch**

A PDF batch consists of:

- One or more PDF files put in a directory with the prefix **deliverable** inside the batch directory

All files in the batch must share a single DRS owner code and a single DRS billing code.

All PDF files will be assigned a persistent identifier (also known as a URN or “Universal Resource Name”). NRS URNs can be added or modified after DRS deposit using the NRS Maintenance System.

Note: the NRS URN assigned to a PDF file will automatically resolve to the FDS (File Delivery Service) URL when PDF files are delivered to the public.

See the [OIS web site](#) for more about FDS.

### 5.1 Accepted formats

DRS accepts files in any variation of the PDF format. However, to facilitate the long-term preservation, accessibility and rendering of PDF files, OIS recommends that PDF files be created in or converted to the most preferred format possible:

<table>
<thead>
<tr>
<th>Most Preferred</th>
<th>PDF/A-1a</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PDF/A-1b</td>
</tr>
<tr>
<td></td>
<td>Any valid PDF format that conforms with the recommendations found in the <em>PDF Feature Recommendations</em> section of the <a href="#">Guidelines for PDF Creation</a></td>
</tr>
<tr>
<td>Least Preferred</td>
<td>Invalid PDF</td>
</tr>
</tbody>
</table>
For all PDF files the extension .pdf is mandatory.

For more information on creating PDF files for long term preservation see: Guidelines for PDF Creation.

Any other file formats that are included in the batch will be ignored by Batch Builder. These ignored files will not be described anywhere in the generated batch.xml file, and they will not be deleted from the batch directory.

5.2 File name rules

This topic describes specific file name requirements for PDF batches. See About file names for general Batch Builder requirements.

The file-naming scheme for files in PDF batches is:

{ownerSuppliedName}_-_{shortDescription}.{extension}

where:

{ownerSuppliedName} is the item control name that associates the digital file with a local counterpart (physical object, catalog record, etc.). This may be an accession number or other curatorially-significant name. Batch Builder will use this value for the batch.xml's ownerSuppliedName value (you can also define ownerSuppliedName using a mapping file). Valid characters to use for the {ownerSuppliedName} are letters, digits, '.', underscores ('_'), and hyphens ('-').

_-_ character sequence (one underscore, one hyphen, one underscore) is used to separate the {ownerSuppliedName} from the {shortDescription}, if the {shortDescription} is present.

{shortDescription} is an optional locally-meaningful description of the file. This portion of the file name will not be used for any Batch Builder metadata generation. It is for depositors who want to embed information in the file names that is useful for their workflow. Valid characters to use for the {shortDescription} are letters, digits, underscores ('_'), and hyphens ('-'). Do not use the character sequence _-_ as part of the {shortDescription}.

.{extension} is the valid file extension – which should always be .pdf for PDF batches.

5.3 Batch directory rules

This topic describes specific requirements for subdirectory names and structure in PDF batches. See About directory names and structure for general Batch Builder requirements.

Within the top-level batch directory, all digital files must be contained within one or more subdirectories that are named to indicate their role or type. Batch Builder requires that subdirectory names begin with a pre-defined prefix. For PDF batches the prefix to use is deliverable.
As long as the first part of a batch subdirectory’s name follows the rule above, you can append additional information. Batch Builder would consider the following subdirectory names to be valid:

- deliverable-1
- deliverable_20061205
- deliverablePDF

Any extensions to a subdirectory’s name must contain only letters, numbers, underscores or hyphens and must not exceed 100 characters in length.

Below is a sample batch directory for a PDF batch that includes several PDF files.

```
C:\myBatches\batchDir\deliverable-shortdescription\1.pdf
   2.pdf
   3.pdf
   ...
   etc.
```

### 5.4 Procedure to create a PDF batch

Follow these steps to create a PDF batch. The steps in Part I (Create the project and batch template) need only be done once. (A project is a template from which you can prepare multiple batches for DRS deposit.) If you have already created the project and batch template, proceed to Part II (Create a batch).

Note: to save project values at any point, from the main menu select Project > Save.

Part I: Create the project and batch template

1. From the Batch Builder main menu, select Project > New.

2. Complete the New Project form and click OK.
New Project form

- **Project name**: this name displays in the Batch Builder title bar when the project is open. This value can be changed later by accessing the Project Properties node in the Configuration panel.

- **Project directory**: path and directory name in which project files are stored. To locate or create the directory, click the ellipsis (…) button to browse your file system. This value cannot be changed once a project is created.

  Best practice note: new users should consider including the word “project” in their project directory names to make it easier to differentiate these from the underlying batch directories.

- **Batch genre**: select “PDF” as the genre. This value cannot be changed once the project is created.

- **Project description**: an internal free text note field associated with the project. This value can be changed later by accessing the Project Properties node in the Configuration panel.

3. **Add administrative metadata**. In the Configuration panel on left, select Administrative Properties. This panel records project-level administrative metadata. This metadata will be used in every batch.xml file generated by this project.
Administrative properties panel

Mouse over any of the field labels to display a definition. Or, consult Administrative properties in the Metadata Reference section for more information.

An asterisk indicates required fields. Batch Builder will provide warnings in the message pane if required fields are missing, but does not validate the contents of these fields. For example, if you supply an invalid Owner Code, this value will be added to the batch.xml file and at deposit, the entire batch will be rejected by DRS.

4. [Optional] Add global metadata that will be applied to every digital file in batches created by this project. In the Configuration panel, select Global Properties under the Batch Template node.

Global properties panel

Consult Global properties in the Metadata Reference section for more information.

If you plan to accept the JHOVE-supplied metadata and do not need to include optional global metadata, skip this step.

Revised: March 18, 2010
5. **Add batch directories to the template.** In the Configuration panel, right click on Batch Template and select Add directory. In the pop-up window, choose a directory name from the list. For PDS batches the only name available will be ‘deliverable’.

![New Batch Template Directory]

Directory-specific metadata properties

[Optional] You can modify a directory name by appending custom text after the predefined name. Type the custom text in the box to the right of the directory list.

Click OK to add the directory to the template.

6. **[Optional]** **Add directory-specific metadata** that will be applied to every digital file in a specific batch directory. In the Configuration panel, select a directory to display its metadata properties.

![Configuration]

Directory-specific metadata properties

Consult [Directory-level properties](#) in the Metadata Reference section for more information on this metadata.

If you plan to accept the administrative metadata inferred by Batch Builder and the technical metadata supplied by JHOVE, skip this step.

**Part II: Create a batch**

7. **Create the batch directory and subdirectories.** From the main menu, select **Batch > New**. In the New Batch window, supply a top-level directory name.

Revised: March 18, 2010
Creating batch directories on disk

Batch Builder has no specific requirements for name of the top-level batch directory, but the name must be no longer than 100 characters and must consist only of letters, numbers, underscores ('_') and hyphens ('-').

By default, the ‘Create directories from batch template’ checkbox will be selected (indicating that Batch Builder will create subdirectories based on directory names defined in the batch template). De-select this option if you want to create the directories outside of Batch Builder.

8. In the Project Directory pane, an entry for your new batch will appear, flagged with a red b.

The red b indicates that a batch.xml file has not yet been generated for this batch directory.

9. **Move digital files into the batch directories.** This step must be performed outside of Batch Builder.

   In Batch Builder, you will be able to see these files in the Project Directory panel if you refresh the display. From the main menu, select View > Refresh file system panel.

10. **Generate the batch xml file.** Right click on the batch and select “Create batch.xml file” (or from the main menu, select Batch > Create batch.xml file).

    As Batch Builder processes the batch, status messages will display in the message window. If batch generation is successful, the final message will be:

    INFO   - Creation of batch.xml complete for batch image-batch-1

    In the Project Directory pane, the “b” icon next to the batch directory changes to a green b with an orange x to indicate the batch xml file has been generated.
If any required metadata is missing, or if any digital files are invalid, Batch Builder will display an error message and the batch.xml file will not be generated. You will need to fix the errors and re-generate the batch. If errors are PDF validation errors, check the option to “ignore file validation errors” in the Options dialog (from the main menu, View > Options) and re-run the batch.

11. [Optional] Review the summary report for this batch.xml generation process. You can view this report by selecting the batch in the “project directory” panel and clicking the Reports node.

Note that on your file system, this report is saved in the Sync directory for the project and not in the batch directory:

[project name]\sync\batch name\reports\ 

The summary report is not deposited to the DRS, so must not be placed within a batch directory.

Once the batch.xml file has been successfully generated, you are ready to upload the batch to DRS. See Uploading batches to DRS for more information.
6. Uploading Batches to DRS

Once the batch is prepared, the deposit agent must upload the batch directories and xml file(s) to a DRS drop box. This upload step must be performed outside of Batch Builder.

You will need a DRS dropbox account and a secure FTP client in order to perform an upload to DRS. To request a dropbox, submit a dropbox request.

The procedure for uploading a batch is described in section 2.4 of the DRS Load Manual, which is available from the DRS section of the OIS web site:

http://hul.harvard.edu/ois/systems/drs/load_manual/

While batch deposits can occur at any time, the DRS batch loading service processes batches only during business hours (Monday to Saturday, 8am-8pm). Batches deposited after these hours will be processed the next business day.

Depending on the size of the batch, the upload process may take several minutes to multiple hours to process and report. The status of batches in the queue can be viewed online:

- For production drop boxes: http://drs.harvard.edu:8080/drs/servlet/WebAdminService?page=view_queue

Review and save the loader report. The DRS loader will send out an email message that reports on the success (or failure) of a processed batch. Recipients of this message are identified within the batch.xml file, based on values stored in the administrative properties of the related Batch Builder project.

For more about loader reports, consult section 3.2 of the DRS Load Manual: http://hul.harvard.edu/ois/systems/drs/load_manual/.
7. Expressing Relationships

7.1 Image relationships

Batch Builder supports these types of relationships between files in a general image batch:

- IS_DERIVATIVE_OF
- IS_TARGET_OF
- IS_TDF_FOR
- IS_ICC_OF

Batch builder will automatically infer these relationships between files in a batch based on file names and directory locations.

File names

Related files must have the same owner supplied name. By default, Batch Builder derives the owner supplied name from the file name on disk (e.g., file “123.tif” will be assigned the owner supplied name “123”). To relate files, assign matching file names (for example, “123.tif” and “123.jpg” will both be assigned the same owner supplied name “123”).

Or, you can use an external mapping file to assign matching owner supplied names in cases where file names on disk do not match.

Directory locations

The following table describes the directory locations that are necessary for Batch Builder to infer relationships in a general image batch.

<table>
<thead>
<tr>
<th>Case</th>
<th>Relationship</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 files with the same ownerSuppliedName, file A is in a directory whose name starts with archival_master,</td>
<td>B IS_DERIVATIVE_OF A</td>
</tr>
</tbody>
</table>
production_master, or deliverable; **file B** is in a directory whose name starts with archival_master, production_master, or deliverable; B's directory is nested directly under A's directory

**File T** is in a directory whose name starts with target, **file A** is in a directory whose name starts with archival_master, production_master, or deliverable; T's directory is nested directly under A's directory

2 files with the same ownerSuppliedName and in the same directory whose name starts with target; **file F** is a file that ends in .txt or .tdf; **file T** is a file that ends in .tif, .tiff, .jpg, .gif, .jp2, or .pcd

**File I** is in a directory whose name starts with icc; **file A** is in a directory whose name starts with archival_master, production_master, or deliverable; I's directory is nested directly under A's directory

The following diagram illustrates the directory structure and files for a general image batch and the relationships that Batch Builder would derive.

```
batchDirectory/
  archival_master/
    U123.tif
    U124.tif
  target/
    retint.tif
    retint.tdf
  icc/
    sepia.icm
  production_master/
    U123.jpg
    U124.jpg
    icc/
      retint.icm
  deliverable_1/
    U123_-_big.jpg
    U124_-_big.jpg
  deliverable_2/
    U123_-_thumb.jp2
    U124_-_thumb.jp2
```

Batch Builder will infer the following relationships for this general image batch:

1. archival_master/production_master/U123.jpg  IS_DERIVATIVE_OF  archival_master/U123.tif
2. archival_master/production_master/U124.jpg  IS_DERIVATIVE_OF  archival_master/U124.tif
3. archival_master/target/redtint.tif  IS_TARGET_OF  archival_master/U123.tif

Revised: March 18, 2010
4. archival_master/target/redtint.tif  IS_TARGET_OF  archival_master/U124.tif
5. archival_master/target/redtint.tdf  IS_TDF_FOR  archival_master/target/redtint.tif
6. archival_master/icc/sepia.icm  IS_ICC_FOR  archival_master/U123.tif
7. archival_master/icc/sepia.icm  IS_ICC_FOR  archival_master/U124.tif
8. archival_master/production_master/deliverable_1/U123_-_big.jpg  IS_DERIVATIVE_OF  archival_master/production_master/U123.jpg
10. archival_master/production_master/deliverable_2/U123_-_thumb.jpg  IS_DERIVATIVE_OF  archival_master/production_master/U123.jpg
11. archival_master/production_master/deliverable_2/U124_-_thumb.jp2  IS_DERIVATIVE_OF  archival_master/production_master/U124.jpg
12. archival_master/production_master/icc/redtint.icm  IS_ICC_FOR  U123.jpg
13. archival_master/production_master/icc/redtint.icm  IS_ICC_FOR  U124.jpg

7.2 Page-turned object relationships

Batch Builder supports the IS_DERIVATIVE_OF relationship between files in a page-turned object batch. Batch builder will automatically infer this relationship between files in a batch based on file names and directory locations. Only files in archival_master, production_master and deliverable directories can be related.

Note: A different kind of relationship – associating a page image with related page text file -- is accomplished through file names or a page mapping file. See file name rules for page-turned batches for more information.

File names

Related files must have matching file names on disk and these names must include a matching page sequence number. For example, Batch Builder will infer a derivative relationship between the following archival master TIFF and deliverable JPEG:

   U123-1.tif
   U123-1.jpg

If file names on disk do not match, you will need to create an owner supplied name mapping file and also a page sequence mapping file for the files you want to relate.
**Directory locations**

The following table describes the directory locations that are necessary for Batch Builder to infer relationships in a general image batch.

<table>
<thead>
<tr>
<th>Case</th>
<th>Relationship</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 files with the same ownerSuppliedName and page sequence number, file A is in an image directory whose name starts with archival_master, production_master, or deliverable; file B is in an image directory whose name starts with archival_master, production_master, or deliverable; B's directory is nested directly under A's directory</td>
<td>B IS_DERIVATIVE_OF A</td>
</tr>
</tbody>
</table>

The following diagram illustrates the directory structure and files for a page-turned batch and the relationships that Batch Builder would derive.

```
C:\myBatches\batchDir\mets.xml
  \archival_master\U123-1.tif
  \archival_master\U123-2.tif
  \deliverable\U123-1.jpg
  \deliverable\U123-2.jpg
  \ocr_corrected\U123-1.txt
  \ocr_corrected\U123-2.txt
```

Batch Builder will infer the following relationships for this page-turned object batch:

1. archival_master\deliverable\U123-1.jpg IS_DERIVATIVE_OF archival_master\U123-1.tif
2. archival_master\deliverable\U123-2.jpg IS_DERIVATIVE_OF archival_master\U123-2.tif
8. Using External Mapping Files

8.1 Owner supplied name mapping file

Batch Builder can be configured to determine the ownerSuppliedName values from a mapping file rather than by using the file names on disk. This mapping file must be named `filemapping.txt`, and it must be placed in the `sync/{/batchname}/` subdirectory of the project before the batch is generated. (Mapping files are not deposited to DRS, so must not be placed within a batch directory.)

An example of the need for an external mapping file is using Batch Builder to deposit images that will be cataloged in OLIVIA. There is a DRS-OLIVIA linking process for images which requires that their ownerSuppliedName be their OLIVIA ID. But in some digitization workflows, you may not have such control of file names on disk (which would be the default source for ownerSuppliedName). But, the depositor can use a mapping file to assign the OLIVIA ID as ownerSuppliedName.

An ownerSuppliedName mapping file must be a **comma-delimited** text file containing:

```
{filePath},{ownerSuppliedName}
```

where `{filePath}` is a relative path from the top-level batch directory, and `{ownerSuppliedName}` is the ownerSuppliedName value that you want Batch Builder to assign to the file.

Here is a sample batch of OLIVIA images in its batch directory:

```
C:\myBatches\proj2007\batch0130\\deliverable\\
    image23.jp2g
    image39.jp2
    image217.jp2
    image340.jp2
```

The depositor would create an ownerSuppliedName mapping file like the following to map the file names on disk to the image’s OLIVIA ID:

```
deliverable\image23.jp2,W80329_1
deliverable\image39.jp2,W80331_1
deliverable\image217.jp2,W80335_1
deliverable\image340.jp2,W80339_1
```
8.2 Page sequence mapping file

Batch Builder can be configured to determine the sequence number of page image and page text files from a mapping file rather than by using a sequence number appended to the file name on disk. This mapping file must be named `ptomapping.txt`, and it must be placed in the `sync/{/batchname}/` subdirectory of the project before the batch is generated. (Mapping files are not deposited to DRS, so must not be placed within a batch directory.)

An external page mapping file may be needed, for example, if page images come back from a digitization vendor without sequence numbers imbedded in their file names. The depositor can use a page mapping file to associate page sequence number with file name on disk at the point of deposit.

A page sequence mapping file must be a **comma-delimited** text file containing:

```
{filePath},{sequenceNumber}
```

where `{filePath}` is a relative path from the top-level batch directory, and `{sequenceNumber}` is the page sequence number that you want Batch Builder to assign to the page image or page text file.

Here is a sample batch of page images and related page text files in its batch directory:

```
C:\myBatches\proj_PTO_2007\batch0130\deliverable\970143.jp2
970144.jp2
970145.jp2
ocr_uncorrected\970143.txt
970144.txt
970145.txt
```

The depositor would create a page sequence mapping file like the following to map the file names on disk to page sequence numbers in the document:

```
deliverable\970143.jpg,1
deliverable\970144.jpg,2
deliverable\970145.jpg,3
ocr_uncorrected\970143txt,1
ocr_uncorrected\970144txt,2
ocr_uncorrected\970145txt,3
```
9. Working with URNs

9.1 Assigning URN resource name patterns

9.2 Using URNs

Batch Builder will request a persistent identifier (also known as a URN or “Universal Resource Name”) from NRS for selected files in a batch, based on batch genre. URN requests are constructed using the NRS authority path and resource name pattern that are specified in a project’s administrative properties. Batch Builder adds URN requests to the batch.xml. After a batch is processed, the resulting URNs are included in the load report sent by DRS back to the depositor.

For general image batches, URNs will be requested for all deliverable files (any files directly within deliverable subdirectories). The URNs will resolve to the Image Delivery Service (IDS) URLs, e.g.: http://ids.lib.harvard.edu/ids/view/7154181, where http://ids.lib.harvard.edu/ids/view/ is the base URL and 7154181 is the DRS ID of the deliverable file.

For Page-turned batches:

- In the Page-turned Object genre, a URN will be requested for the PDS METS file only (a URN to the METS file will take the user to the top level of the document). This URN will resolve to a Page Delivery Service (PDS) URL, e.g.: http://pds.lib.harvard.edu/pds/view/6773426 where http://pds.lib.harvard.edu/pds/view/ is the base URL and 6773426 is the DRS ID of the METS file.

- In the Page-turned Object – IDS URN for each image genre, URNs will be requested for the PDS METS file and for all deliverable page image files. The URN of the METS file will resolve to a PDS URL while the URNs for the deliverable page images will resolve to IDS URLs.

For PDF batches, URNs will be requested for all files. The URNs will resolve to File Delivery Service (FDS) URLs, e.g.: http://fds.lib.harvard.edu/fds/deliver/13790530 where http://fds.lib.harvard.edu/fds/deliver/ is the base URL and 13790530 is the DRS ID of the deliverable file.

For any single batch, Batch Builder requires that the first part of the URN authority path match the first part of the DRS owner code. For example, the combination of authority path “RAD.ARCH” and owner code “RAD.SCH” would be acceptable, but the combination authority path “RAD.ARCH” and owner code “FHCL.COLL” would not work because “RAD” and “FHCL” do not match. When a batch has mismatching owner code and authority path, Batch Builder will fail to generate the batch.xml file.

NRS URNs can be added or modified after DRS deposit using the NRS Maintenance System. For page-turned objects, URN modifications are also possible using the PDS Maintenance System.

9.1 Assigning URN resource name patterns

The resource name pattern as well as NRS authority path are specified in a project’s administrative properties.
The resource name component of the URN must be unique relative to the authority path. Batch Builder can generate all or part of the resource name automatically using mask fields that will be replaced with actual values by the system (shown in the table below).

Because only one resource name pattern can be specified in Batch Builder and resource names have to be unique, the resource name pattern must include one or more mask fields. The exception to this rule is in the case of batches where only one URN is requested. This would be the case for general image batches that contain only a single deliverable image, or for page-turned batches where only the PDS METS file is assigned a URN. For these batches the user-specified resource name does not need to include mask fields.

Resource names can be composed of alpha-numeric characters (upper and lower case, although all NRS comparisons are case-insensitive) and the following punctuation characters:

\( (\ ) + , - = @ ; $ _ ! * ' : \)

Batch Builder will use \( \{n\} \) as the default resource name mask field (\( \{n\} \) resolves to a unique number). However, NRS supports additional resource name patterns that are described below.

<table>
<thead>
<tr>
<th>Field</th>
<th>Replacement Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>{dd}</td>
<td>01 to 31</td>
<td>Day of the month</td>
</tr>
<tr>
<td>{hh24}</td>
<td>01 to 24</td>
<td>Current hour of day</td>
</tr>
<tr>
<td>{mm}</td>
<td>00 to 59</td>
<td>Current minute of hour</td>
</tr>
<tr>
<td>{mo}</td>
<td>01 to 12</td>
<td>Current month of year</td>
</tr>
<tr>
<td>{ss}</td>
<td>00 to 59</td>
<td>Current second of hour</td>
</tr>
<tr>
<td>{yyyy}</td>
<td>1999 to 9999</td>
<td>Current year</td>
</tr>
<tr>
<td>{n}</td>
<td>0 to 10 to the 27th power</td>
<td>Unique numeric value</td>
</tr>
<tr>
<td>{fileBaseName}</td>
<td>Name of the file excluding the file extension</td>
<td>Base name of file</td>
</tr>
</tbody>
</table>

Here are examples of URN masks and the resulting resource name:

- urn-3:dig:{n} ➞ urn-3:dig:75
- urn-3:FAL:{yyyy}-{n} ➞ urn-3:FAL:1999-76
- urn-3:HCL-DIG:{yyyy}{mo}{dd} ➞ urn-3:HCL-DIG:20000103 (unique only for one file on any given day)
- urn-3:HCL-DIG:{yyyy}{mo}{dd}{n} ➞ urn-3:HCL-DIG:200001032 (always unique)

### 9.2 Using URNs

Each URN assigned by NRS has the form:
urn-3:{authorityPath}:{resourceName}
urn-3:RAD.SCHL:1081404

where "urn-3" is the Harvard namespace identifier, {authorityPath} is the NRS authority path corresponding to the organizational unit responsible for the URN, and {resourceName} is the unique identifier for the digital file.

To make a URN actionable (usable in a browser), you must append it to an NRS URL (http://nrs.harvard.edu). The full working URL for the example URN above would be:

http://nrs.harvard.edu/urn-3:RAD.SCHL:1081404

New URNs are passed to the NRS name resolution server each night. An “active” URN created today will work in a browser tomorrow.
10. Reports and Messages

10.1 Summary report

After Batch Builder successfully generates a batch.xml file, it will create a summary report of the process. This summary report is saved in the Sync directory for the project and not in the batch directory:

[project name]\sync\[batch directory name]\reports\

By default, both a text and xml version of the report will be created. You can turn off creation of the xml report by accessing Batch Builder options (View > Options from the main menu).

You can also access the summary report for a batch within the Batch Builder interface. In the Project Directory panel on left, open the batch directory of interest and then open the Reports folder. Click a report and its contents will display in the right pane.

The report provides summary information about the batch, including its size and number of files included. The report will also indicate file exceptions which did not prevent generation of the batch. These exceptions include files found invalid by JHOVE for which valid metadata was provided by the depositor, files that were ignored during batch generation, and files in a format different than indicated by their extension.

If Batch Builder finds errors during batch generation, creation of the batch.xml file will fail and a summary report will not be created.
10.2 Processing and error messages

Batch Builder will display processing messages in the message pane as you create batches and generate batch.xml files. There are three types of message:

- **INFO**: an informational message about tasks performed by Batch Builder. Normally, no action by the user is required.

- **WARN**: a warning, usually about missing required file metadata, that may require action by the user. Action may not be required if you have supplied metadata in global properties or directory-level properties to replace metadata that JHOVE cannot extract.

- **ERROR**: a message that Batch Builder has failed a particular task or information about an error condition that will prevent Batch Builder from generating the batch.xml and/or the PDS METS file. The user will need to correct the error and regenerate the batch.

These messages will accumulate in the message pane during your Batch Builder session, but will be discarded when you close the application. During a session, you can clear the message pane by selecting **View > Clear messages** from the main menu.

If you contact OIS to report a problem using Batch Builder, OIS staff may ask to see the processing messages from your session. In an error situation, it is good practice to either keep the client open so messages are available or preserve the processing messages by copying them to a mail message or text file.

Here are some common ERROR messages:

**ERROR - Batch with name: [batch directory name] already exists.**
You are trying to create a new batch using a directory name that is already in use.

**ERROR - File system out of sync with directory structure in user configuration:**
The set of directories defined in your batch template do not match the directories defined on the file system. For example, your batch template has `archival_master` and `deliverable` directories defined but only a `deliverable` directory defined on the file system. To correct the error, add or remove directories in the template or file system so directories match. Or, use the project to create a new batch in which directories match the template.

**ERROR - page order 1 not found. Order values must be sequential and start with 1.**
Batch Builder cannot identify the page sequence numbers associated with page image files and/or page text files in a page-turned batch. File names may not be in the required format or Batch Builder cannot find the page sequence mapping file.

**ERROR - Required fields are missing or invalid:**
**ERROR - Required Administrative Properties are missing: [name of property]**
**ERROR - Property not found: [name of property]**
Required metadata properties are missing or invalid. In most cases, Batch Builder will indicate which properties are at issue. The missing or invalid values could be located in administrative properties, global properties, or directory-level properties.

**ERROR - The first part of the URN Authority path must match the first part of the DRS owner code: Authority Path=… owner name=…**

Revised: March 18, 2010
The project contains mismatching top levels in URN authority path and DRS owner code. For example, the combination of authority path “RAD.ARCH” and owner code “RAD.SCH” would be acceptable, but the combination authority path “RAD.ARCH” and owner code “FHCL.COLL” would not work because “RAD” and “FHCL” do not match.

**ERROR - There are no files in the batch directories**
Your batch directories on disk may be empty – to avoid the error, populate those directories with digital files and regenerate the batch.xml. It is also possible to get this error if the directories defined in the batch template do not match the directories defined on the file system.

**WARN - Jhove: file extension may be wrong but Technical Metadata can still be extracted:**
A JHOVE module warning that the technical metadata for a digital file does not agree with the format indicated by its file extension.

**WARN - Required field [name of field] set to empty**
A required metadata property is empty.

**WARN - Error extracting technical metadata for [path and file name] Reason: [reason for error]**
When JHOVE finds a file to be invalid, it cannot extract the file’s technical metadata, which means the metadata properties required by Batch Builder cannot be calculated. If JHOVE is unable to extract technical metadata for some digital files in the batch, you can supply the missing values in the batch template and these will substitute for the metadata values that JHOVE cannot generate. See the “Set by JHOVE” portions of the Metadata Reference for more information.
11. Metadata Reference

11.1 Administrative properties
11.2 Global properties
11.3 Directory-level properties
11.4 File-level properties

This section provides definitions for the DRS metadata elements that appear in Batch Builder properties panels.

Note that Batch Builder relies on an internal JHOVE module (a format validation module developed by Harvard and JSTOR) to extract technical metadata from digital files in a batch. Consult the JHOVE web site for more information.

For more information about DRS metadata requirements, consult the DRS & Delivery Services Documentation on the OIS web site.

11.1 Administrative properties

Administrative properties are administrative metadata about a batch as a whole. These properties are set by the user and will be applied to every batch.xml file generated by a project. To access these properties, select the ‘Administrative properties’ node in the ‘Configuration’ area of the left pane.

Notes about the administrative properties form: You can mouse over any of the field labels to display a definition. An asterisk indicates required fields. Batch Builder will provide warnings in the message pane if required fields are missing, but can not validate the contents of these fields. For example, if you supply an invalid Owner Code, this value will be added to the batch.xml file and at deposit, the entire batch will be rejected by DRS.
The table below provides definitions of each administrative property. For more information about these properties, see DRS base metadata.

<table>
<thead>
<tr>
<th>Batch Builder Label</th>
<th>batch.xml element</th>
<th>Definition</th>
</tr>
</thead>
</table>
| Batch Name Pattern  | <batch name="" . . ."> | Place-holder values used to generate a unique name for each batch created in a project. Batch name will be referenced in the batch processing notification sent by DRS. Also, Batch Builder uses this name to label the summary report it creates after the batch.xml is generated.

Place-holder elements include:
- \{owner\} DRS owner code
- \{depositAgent\} name of deposit agent, as defined in “Deposit Agent” property
- \{batchDir\} name of top level batch directory
- \{yyyy\} current year
- \{mm\} current month
- \{dd\} current day
- \{hh24\} current hour
- \{mm\} current minute
- \{ss\} current second

You can use some or all of these place-holder elements, or add your own custom text.

Note: a batch directory name that ends in a digit
<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Success Email</td>
<td><code>&lt;emailSuccess&gt;</code></td>
</tr>
<tr>
<td>Failure Email</td>
<td><code>&lt;emailFailure&gt;</code></td>
</tr>
<tr>
<td>Owner Code</td>
<td><code>&lt;owner&gt;</code></td>
</tr>
<tr>
<td>Billing Code</td>
<td><code>&lt;billingCode&gt;</code></td>
</tr>
<tr>
<td>BatchUserVal</td>
<td><code>&lt;batch userval=&quot;...&quot;&gt;</code></td>
</tr>
<tr>
<td>Directive</td>
<td><code>&lt;batch directive=&quot;...&quot;&gt;</code></td>
</tr>
<tr>
<td>URN Authority Path</td>
<td><code>&lt;urnmask&gt;</code></td>
</tr>
</tbody>
</table>

(not a place-holder value) can be auto-incremented with each new batch if the “auto increment new batch directory names” option is selected (see Setting Options). Max length 400 characters.

DRS loader will send load reports to this address when a batch is processed successfully. This is a list to which you can add multiple addresses. Type the full address and press enter to add it to the list. Max length 100 characters.

DRS loader will send failure reports to this address when processing of a batch fails. This is a list to which you can add multiple addresses. Type the full address and press enter to add it to the list. Max length 100 characters.

DRS owner code that will be associated with every file in a batch. For a list of DRS owner codes, see: [http://drs.harvard.edu:8080/drs/owners](http://drs.harvard.edu:8080/drs/owners).

Note: The top level values of owner code and URN authority path must match. E.g., the combination of authority path “RAD.ARCH” and owner code “RAD.SCH” are acceptable, but the combination authority path “RAD.ARCH” and owner code “FHCL.COLL” would not work because “RAD” and “FHCL” do not match. Max length 30 characters.

DRS billing code that will be associated with every file in a batch. Billing codes will be supplied by the curator responsible for the digital files. Max length 100 characters.

Batch-specific free text note that will be included in the batch.xml file generated from this project. Max length 50 characters.

Used to request special processing. Only valid value is PAGEDOBJECT and this is supplied automatically for PTO (page-turned object) batches. Max length 100 characters.

NRS authority path that will be used in every URN request generated in a batch. For a list of existing authority paths, consult the NRS status page: [http://nrs.harvard.edu/urn-3:hul.ois:nrsstatusprod](http://nrs.harvard.edu/urn-3:hul.ois:nrsstatusprod)

Note: Top level values of owner code and URN authority path must match. E.g., the combination of authority path “RAD.ARCH” and owner code “RAD.SCH” are acceptable, but the combination authority path “RAD.ARCH” and owner code
```
<table>
<thead>
<tr>
<th>URN Resource Name Pattern</th>
<th>&lt;urnmask&gt;</th>
<th>Auto-generated urnmask values for the resource name portion of the URN. This pattern will be used in every URN request generated in a batch. See Working with URNs for more information on urnmask values. Max length 256 characters.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Additional Batch Output Directory</td>
<td>n/a</td>
<td>For each batch.xml file generated by this project, a backup copy will be saved to the directory path specified in this field. Max length 512 characters.</td>
</tr>
<tr>
<td>Deposit Agent</td>
<td>n/a</td>
<td>Text identifying the deposit agent or agency associated with a batch. This value is used to populate the {depositAgent} place-holder in the Batch Name Pattern field. Max length 80 characters.</td>
</tr>
</tbody>
</table>
```

### 11.2 Global properties

Global properties are administrative and technical metadata values that will be applied to every digital file in a batch created by a project. Note that it only makes sense to apply file metadata values globally when a batch contains digital files in a single format (for example, a batch containing only TIFF files).

Batch Builder can also assign metadata values at the directory level (to files in a single format). See Directory-level properties for more information.

To access global properties, select the Global properties node in the Configuration area of the left pane. The Global Properties node has two data entry tabs:

- **Optional**: these are optional file metadata values that can be set by the user.
- **Set by JHOVE**: these metadata values are automatically extracted by the JHOVE module during the batch.xml generation step. If JHOVE is unable to extract technical metadata for some digital files in the batch, you can add values in the Set by JHOVE tab that will substitute for the metadata values that JHOVE cannot generate.

If you plan to accept the JHOVE-supplied metadata and do not need to include any of the optional global properties, you can skip this metadata level.

### Optional properties

These are optional file metadata values that can be set by the user. Values supplied in this tab will be applied to every digital file in the batch.
The table below provides definitions of each global property. Maximum length for each property value is 4000 characters unless otherwise specified.

<table>
<thead>
<tr>
<th>Batch Builder Label</th>
<th>batch.xml element</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>File metadata</strong> (see DRS base metadata for more information)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>UserVal</td>
<td><code>&lt;transaction userval=&quot;...&quot;/&gt;</code></td>
<td>Free text that will be associated with each digital file in the batch. Maximum length: 50 characters.</td>
</tr>
<tr>
<td><strong>Image metadata</strong> (see DRS image metadata supplement for more information)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Target Notes</td>
<td><code>&lt;targetnotes&gt;</code></td>
<td>Designates the name of the “internal” target(s) scanned in-frame with the source item.</td>
</tr>
<tr>
<td>History</td>
<td><code>&lt;history&gt;</code></td>
<td>Designates the image change history.</td>
</tr>
<tr>
<td>Source</td>
<td><code>&lt;source&gt;</code></td>
<td>Designates the physical attributes of the source material relevant to interpreting digital image accuracy and/or quality.</td>
</tr>
<tr>
<td>System</td>
<td><code>&lt;system&gt;</code></td>
<td>Designates the manufacturer and model names/numbers for the scanner or digital camera and its associated driver/imaging software.</td>
</tr>
<tr>
<td>Producer</td>
<td><code>&lt;producer&gt;</code></td>
<td>Designates the organization-level producer(s) of the image.</td>
</tr>
<tr>
<td>OptRes</td>
<td><code>&lt;optres&gt;</code></td>
<td>Designates the actual number of photo elements in the scanning sensor (colloquially known as “the maximum optical resolution” for a system). Maximum length: 30 characters.</td>
</tr>
<tr>
<td>Pro Software</td>
<td><code>&lt;prosoftware&gt;</code></td>
<td>Designates the name and version of the image processing software used to edit or transform the...</td>
</tr>
</tbody>
</table>
image data captured at scanning.

| Enhancements | <enhancements> | Designates the settings, or description of their function, used by the prosoftware. |
| Methodology | <methodology> | Designates the methodology and rationale to digitize an object or collection. |

**Application metadata**

| Creator | <creator> | Software used to create an application file. |

## Set by JHOVE properties

These are file metadata properties that are normally extracted by the JHOVE module. If JHOVE is unable to extract technical metadata for some digital files in the batch, you can add values in the Set by JHOVE tab that will substitute for the metadata values that JHOVE cannot generate. Values supplied in this tab will NOT override metadata values successfully extracted by JHOVE.

The table below provides definitions of each global property that is set by JHOVE.

<table>
<thead>
<tr>
<th>Batch Builder Label</th>
<th>batch.xml element</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>File metadata</strong> (see DRS base metadata for more information)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>File Format</td>
<td>&lt;fileformat&gt;</td>
<td>The basic nature of the binary material on disk. It is more specific than the DRS metadata type in many cases. Required.</td>
</tr>
<tr>
<td>Mime Type</td>
<td>&lt;mimetype&gt;</td>
<td>Specifies the mime type of the digital file you are depositing. Required.</td>
</tr>
<tr>
<td><strong>Image metadata</strong> (see DRS image metadata supplement for more information)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Quality Layers</td>
<td>&lt;qualitylayers&gt;</td>
<td>Number of quality layers to which each JPEG2000 image tile has been decomposed. Useful in determining the number of lower quality images that can be extracted from the JPEG2000 image. Maximum length: 30 characters.</td>
</tr>
<tr>
<td>Resolution Levels</td>
<td>&lt;reslevels&gt;</td>
<td>Number of resolution levels to which each JPEG2000 image tile has been decomposed. Useful in determining the size of the smallest</td>
</tr>
</tbody>
</table>
subresolution thumbnail image available in the JPEG2000 image. Maximum length: 30 characters.

<table>
<thead>
<tr>
<th>Text metadata</th>
</tr>
</thead>
<tbody>
<tr>
<td>Character Repertoire</td>
</tr>
<tr>
<td>&lt;characterrep&gt;</td>
</tr>
<tr>
<td>Required for page-turned objects.</td>
</tr>
<tr>
<td>Character Map</td>
</tr>
<tr>
<td>&lt;charactermap&gt;</td>
</tr>
<tr>
<td>Required for page-turned objects.</td>
</tr>
<tr>
<td>Text Descriptor</td>
</tr>
<tr>
<td>&lt;descriptor&gt;</td>
</tr>
<tr>
<td>Required for page-turned objects.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Application metadata</th>
</tr>
</thead>
<tbody>
<tr>
<td>App Descriptor</td>
</tr>
<tr>
<td>&lt;descriptor&gt;</td>
</tr>
<tr>
<td>Required for application files.</td>
</tr>
</tbody>
</table>

### 11.3 Directory-level properties

Directory-level properties are metadata values that can be assigned to individual subdirectories in a batch. Metadata assigned at the directory level will be applied to every file in that directory. For files of the same type (e.g., just TIFF files), if you want to specify different metadata for a subset of the files, these files need to be stored in separate batch subdirectories.

To assign directory level metadata values, you need to:

1. Create a project and define a batch template within the project.
2. Add batch subdirectories to the batch template.
3. Select a batch subdirectory and input metadata values into the appropriate form.

For each subdirectory defined in the batch template, there are three tabs of metadata values:

- **Required**: this tab contains required administrative metadata values that will be assigned to every file in the directory. Batch Builder automatically infers most of these values based on the names and directory locations of the files. The user can override these values.

- **Optional**: this tab contains optional administrative metadata values that will be assigned to every file in the directory. Batch Builder does not infer these values – they must be set by the user as needed.

- **Set by JHOVE**: this tab contains metadata values that are automatically extracted by the JHOVE module during the batch.xml generation step. If JHOVE is unable to extract technical metadata for some digital files in the directory, you can add values in the Set by JHOVE tab that will substitute for the metadata values that JHOVE cannot generate.

### Required properties

This tab contains required administrative metadata values that will be assigned to every file in the directory. Batch Builder automatically infers most of these values based on the names and directory locations of the files. The user can override these values.

Revised: March 18, 2010
Note that directory level metadata at Batch Template level is used for METS files in Page Turned Objects genre.

The table below provides definitions of each required directory-level property.

<table>
<thead>
<tr>
<th>Batch Builder Label</th>
<th>batch.xml element</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>File metadata (see DRS base metadata for more information)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Role</td>
<td>&lt;role&gt;</td>
<td>Defines the role of the file as compared to other digital file that are the same logically but a different format. This is automatically calculated using the directory name, but can be overridden.</td>
</tr>
<tr>
<td>Purpose</td>
<td>&lt;purpose&gt;</td>
<td>Defines the purpose of the file as compared to other digital files that are the same logically but in a different format.</td>
</tr>
<tr>
<td>Quality</td>
<td>&lt;quality&gt;</td>
<td>Defines the quality of the file as compared to other digital files that are the same logically but in a different format.</td>
</tr>
<tr>
<td>Access Flag</td>
<td>&lt;access&gt;</td>
<td>This element specifies what bodies can access this digital file if it is deliverable. Valid values are: P for public, R for restricted, N for no access. A public file is available to the world. A restricted file is available to only Harvard. A file marked as having no access is only available to administrators via the DRS Web Admin.</td>
</tr>
<tr>
<td>Usage Class</td>
<td>&lt;usageClass&gt;</td>
<td>Each digital file is stored according to its usage classification as either &quot;high use&quot; or &quot;low use.&quot; Deliverables used for public access should be classified as &quot;HIGHUSE&quot;. Preservation and archival versions, masters that are not being delivered and other &quot;dark files&quot; should be classified as &quot;LOWUSE&quot;.</td>
</tr>
</tbody>
</table>
Profile <profile name="…"/> Name of a profile that identifies form of organization and content found in the container object. Profile is set up by OIS in consultation with local project manager.

Version <profile version="…"/> Version number of a container profile.

Optional properties

This tab contains optional administrative metadata values that will be assigned to every file in the directory. Batch Builder does not infer these values – they must be set by the user as needed.

Note: Some directory types may support only a subset of these properties.

Directory-level properties, optional metadata

The table below provides definitions of each optional directory-level property.

<table>
<thead>
<tr>
<th>Batch Builder Label</th>
<th>batch.xml element</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>File metadata (see DRS base metadata for more information)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>UserVal</td>
<td>&lt;transaction userval=&quot;…&quot;/&gt;</td>
<td>Free text that will be associated with each digital file in the batch. Maximum length: 50 characters.</td>
</tr>
<tr>
<td>Image metadata (see DRS image metadata supplement for more information)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Orientation</td>
<td>&lt;orientation&gt;</td>
<td>Designates the orientation of the image, with respect to the placement of its columns (imagewidth) and rows (imageheight), as it was saved to disk.</td>
</tr>
<tr>
<td>Target Notes</td>
<td>&lt;targetnotes&gt;</td>
<td>Designates the name of the “internal” target(s) scanned in-frame with the source item.</td>
</tr>
<tr>
<td>History</td>
<td>&lt;history&gt;</td>
<td>Designates the image change history.</td>
</tr>
<tr>
<td>Source</td>
<td>&lt;source&gt;</td>
<td>Designates the physical attributes of the source material relevant to interpreting digital image accuracy and/or quality.</td>
</tr>
<tr>
<td>System</td>
<td>&lt;system&gt;</td>
<td>Designates the manufacturer and model names/numbers for the scanner or digital camera and its associated driver/imaging software.</td>
</tr>
<tr>
<td>Producer</td>
<td>&lt;producer&gt;</td>
<td>Designates the organization-level producer(s) of</td>
</tr>
</tbody>
</table>
OptRes | <optres> | Designates the actual number of photo elements in the scanning sensor (colloquially known as “the maximum optical resolution” for a system). Maximum length: 30 characters.
Pro Software | <prosoftware> | Designates the name and version of the image processing software used to edit or transform the image data captured at scanning.
Enhancements | <enhancements> | Designates the settings, or description of their function, used by the prosoftware.
Methodology | <methodology> | Designates the methodology and rationale to create a file or collection.
DRS ID of Target file | <relationshipMap><id> | DRS ID of related target file already in storage.
DRS ID of ICC file | <relationshipMap><id> | DRS ID of related ICC file already in storage.

**Application metadata**

Creator | <creator> | Software used to create an application file.

### Set by JHOVE properties

This tab contains metadata values that are automatically extracted by the JHOVE module during the batch.xml generation step. If JHOVE is unable to extract technical metadata for some digital files in the directory, you can add values in the ‘Set by JHOVE’ tab that will substitute for the metadata values that JHOVE cannot generate. Values supplied in this tab will NOT override metadata values successfully extracted by JHOVE.

**Note:** Some directory types may support only a subset of these properties.

**Directory-level properties, set by JHOVE metadata**

The table below provides definitions of each directory-level property that is set by JHOVE.

<table>
<thead>
<tr>
<th>Batch Builder Label</th>
<th>batch.xml element</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>File metadata</strong> (see DRS base metadata for more information)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### File-level properties

File-level properties are metadata values that can be assigned to individual files in a batch. In Batch Builder, file-level properties are either extracted by the JHOVE module at the point of batch.xml generation or are automatically inferred by the names and directory locations of each file. Many of these properties can be overridden at the directory level by a user entering values in one of Batch Builder’s metadata entry forms.

There are two file-level properties automatically set by Batch Builder that can be overridden by supplying a separate external mapping file: owner supplied name and page sequence number (for page-turned batches). For more information, see [Using External Mapping Files](#).

---

| **File Format** | `<fileformat>` | The basic nature of the binary material on disk. It is more specific than the DRS metadata type in many cases. Required. |
| **Mime Type** | `<mimetype>` | Specifies the mime type of the digital file you are depositing. Required. |
| **Image metadata** | (see DRS image metadata supplement for more information) |
| **Bits Per Sample** | `<bitpersample>` | The number of bits per component for each pixel. This field provides N values depending upon the number of components (aka “channels”) in the image. |
| **Compression** | `<compression>` | Designates the compression scheme used to store the image data. |
| **Photo Interp.** | `<photointerp>` | Designates photometric interpretation, the color space of the decompressed image data. |
| **X Resolution** | `<xres>` | Designates the number of pixels per resunit in the width length. |
| **Y Resolution** | `<yres>` | Designates the number of pixels per resunit in the image length. |
| **Resolution Unit** | `<resunit>` | Designates the intended placement of pixels in the xres and yres dimensions of the printed image. |
| **Quality Layers** | `<qualitylayers>` | Number of quality layers to which each JPEG2000 image tile has been decomposed. Useful in determining the number of lower quality images that can be extracted from the JPEG2000 image. Maximum length: 30 characters. |
| **Resolution Levels** | `<reslevels>` | Number of resolution levels to which each JPEG2000 image tile has been decomposed. Useful in determining the size of the smallest subresolution thumbnail image available in the JPEG2000 image. Maximum length: 30 characters. |
| **Text metadata** |
| **Character Repertoire** | `<characterrep>` |
| **Character Map** | `<charactermap>` |
| **Text Descriptor** | `<descriptor>` |
| **Application metadata** |
| **App Descriptor** | `<descriptor>` |
12. Command Line Reference

12.1 Generic command line syntax

12.2 Actions

12.3 Switches

The Batch Builder installation package includes a command line tool that performs two functions found in the GUI desktop application: (1) create batch directories on disk and (2) generate the batch control file. This tool is designed for use in automated deposit workflows.

The command line tool should be used in conjunction with the GUI. A suggested workflow would be to use the GUI to define the project and batch template, then write a script that:

- Uses the command line tool to create the batch directories on disk,
- Copies files into the directory structure, and
- Uses the command line tool to generate the batch control file. (For page-turned batches, the tool can also generate the METS file.)

12.1 Generic command line syntax

The generic command line syntax is:

```
batchbuildercli -a action [-l] -p project_directory -b batch_directory_name [-hollis hollisID] [-createmets y|n] [-savexml y|n] [-o “propertyname=value;propertyname=value”]
```

Command line actions must be executed in the context of the top level Batch Builder application directory.

12.2 Actions

Two actions can be executed from the command line: create directories on disk (`-a directories`) and generate batch control file (`-a batch`). If both actions are performed in a scripted workflow, the `-a directories` action must occur first.

“Create directories on disk” action

The `-a directories` action creates the directory structure on disk, based on directories defined in the project’s batch template. To view a project’s batch template settings, you can open the project in the GUI or view the proj.conf file in the project directory on your local file system.

Example use:

```
C:\BatchBuilder-1.2>batchbuildercli -a directories -p G:\myprojects\PTOproject1 -b batchA
```

Value of the `-b` switch (batchA) is the name of the batch directory that will be created during execution of this command.
Switches that are applicable to the “create directories on disk” action:

-\p (project path; required)
-\b (batch directory name; required)

“Generate the batch control file” action
The \a batch action generates the batch control file (batch.xml) in the specified batch directory.

Example use:

C:\BatchBuilder-1.2>batchbuildercli-a batch -l -p
G:\myprojects\PTOproject1 -b batchA -hollis 000725134

Switches that are applicable to the “generate batch.xml” action:

-\l (display batch summary report on screen; optional)
-\p (project path; required)
-\b (batch directory name; required)
-\hollis (HOLLIS ID; optional)
-createmets (create METS file; optional)
-savexml (create xml summary report; optional)
-\o (override project properties; optional)

### 12.3 Switches

-\l   – optional – displays the batch summary report to the screen after batch.xml is created. Used only in the context of the \a batch action.

-\p [project path]   – required – indicates the path to the project on the local file system. This path must be created in Batch Builder GUI before using the command line option. The project path cannot be created while executing this command. If path does not already exist the action will fail.

-\b [batchname]   – required – name of batch directory. In the \a directories context, this switch creates the batch directory with the batchname specified. In the \a batch context this switch points to the name of the batch directory where the batch.xml file will be created. In the \a batch context, if the batch directory does not exist the action will fail.

-\hollis [hollisID]   – optional – system number of corresponding record in HOLLIS. The DRS loader will use this number to pull citation information about the page-turned object from HOLLIS. Used in PTO batches only.

-createmets y|n   – optional – in a PTO batch, specifies whether the METS file should be created along with the batch.xml file. Default is yes (if this option is not specified, the METS file will be created). Used in PTO batches only.

-savexml y|n   – optional – create xml version of batch summary report. If this switch is omitted, or if it is set to “no”, a text version of the batch summary report will be written at time of batch creation to the project’s sync\[batchname]\reports directory. If this switch is set to “yes”, text and xml versions will be created.

Revised: March 18, 2010
-o [project property name] – optional – overrides specified project level administrative properties at time of batch creation. All administrative properties are eligible to be overridden. On Linux platforms the -o argument must be wrapped in quotes to ensure that the entire string is passed correctly to Batch Builder. Property names used with this switch are:

- batchName
- successEmail
- failureEmail
- owner
- billingCode
- batchUserval
- directive
- urnAuthorityPath
- resourceNamePattern
- batchFileOutput
- depositAgent

Example use:

C:\BatchBuilder-1.2>batchbuildercli -a batch -l -p c:\myprojects\projectA -b batchA -o "billingCode=hul.arch;successEmail=a@a.com,b@foo.edu"

See the Metadata Reference for descriptions of these administrative properties.
User Guide Revision History

8/3/09: Added information on new batch genres supported by Batch Builder. Updated the Creating a Container Batch section to describe the new container types dark PDS containers and opaque containers. Added a new section Creating a PDF Batch that describes this new batch genre.