Administrator's Guide - Filenaming Schemes

Administrator's Guide - Filenaming Schemes

Contents

Overview

Filenaming schemes consist of four main parts: id, volume, page, and recording. Each part has its own format which is described by a sequence of character types or a single character type. The id is the only format that has a fixed length, all others can have arbitrary length which are specified by a single type. A separator character (or sequence) must be specified seperately for use in between parts. Note that the recording separator character can be empty while the score separator character must **not** be empty.

The filenaming formats are then fixed to the following patterns for now:

- Recordings: <id><recording separator character><recording>
- Score Pages: <id><score separator character><volume><score separator character><page>[<score separator character><suffix>]

Static characters used in the id or as separators should be characters that are legal in the filesystem being used by the server and clients. This usually means that "/", "\", and any whitespace are **NOT** valid.

By default, the Z39.50 import before digitization will search using the id format specified above using the any field. For more information see Administrator's Guide - Catalog Integration#Z39.50 Searches

Variations does not deal with the creation of new ids. The filenaming scheme specified will mainly be used for interpretation and validation.

Configuration

The filenaming scheme is configured for each library server through the <FilenameScheme> tag. Each format is configured using the types listed below. Note that id formats are fixed length and can have mixed types while all other formats need to have only a single character type specified like below. The scoreSeparator and recordingSeparator are interpreted as literal strings with the scoreSeparator needing to be non-empty. For the example scheme below, valid filenames include abc0123-a.wav and abc0123-bc.wav for recordings and abc0123-1-1.tiff, abc0123-1-1-a.tiff, and abc0123-03-25.tiff for score images. Invalid filenames would be abc01234-a.wav, abc0123a.wav, abc0123-a-a.tiff, or abc012311.tiff.

Suffixes are only used when adding new files to scores. The suffix allows ordering of files to be inserted. For example, consider a score has three files:

1. abc0123-1-1.djvu
2. abc0123-1-2.djvu
3. abc0123-1-3.djvu

To add three files in order after the 2nd page with the suffixFormat set to "a", one would name these files abc0123-1-2-a.djvu, abc0123-1-2-b.djvu, and abc0123-1-2-c.djvu resulting in the following order:

1. abc0123-1-1.djvu
2. abc0123-1-2.djvu
3. abc0123-1-2-a.djvu
4. abc0123-1-2-b.djvu
5. abc0123-1-2-c.djvu
6. abc0123-1-3.djvu

Example FilenameScheme tag in dmlserver.xml

```
<FilenameScheme
    idFormat="aaadddd"
    volumeFormat="d"
    pageFormat="d"
    recordingFormat="a"
    scoreSeparator="-"
    recordingSeparator="-"
    suffixFormat="a"/>
```

File Character Types

The available character types are shown in the table below:

Type	Placeholder Character	Possible Values
Uppercase Letters	A	A-Z
Lowercase Letters	a	a-z
Decimal	d	0-9
Octal	o	0-7
Binary	b	0-1
Hexadecimal (lowercase letters)	h	0-9,a-f
Hexadecimal (Uppercase letters)	H	0-9,A-F

Note that for validation, Variations is case insensitive. Also note that letter-based character types (Uppercase and Lowercase) do not have an equivalent of zero in number-based character types. This means that leading zeros (in number-based types) in either the volume, page, or recording format do not affect the value while leading a's or A's do have an affect. For example, 002 is equivalent to 2 for a recordingFormat set to "d". If the recording format is set to "a", aac is equivalent to 705 and b is equivalent to 2.