

NAME

DamScan – Daminion database analyzer

SYNOPSIS

```
Python DamScan.py [--ini INI_FILE] [-f] [-i] [-g]
    [-t [{Event, Place, GPS, People, Keywords, Categories,
    Collections} [...]]]
    [-b [SEPARATOR [SEPARATOR ...]]]
    [-x EXFILE | -y ONLYFILE] [-a ACKFILE]
    [-l] [-c DBNAME] [-s SERVER] [-p PORT] [-u USER]
    [-o OUTFILE]
    [-v[v]] [-h] [--version]
```

DESCRIPTION

Daminion digital asset management (DAM) system is a great tool for assigning metadata (tags) to your digital assets and for sorting and searching the items. Daminion also allows you to link or group associated items together, but there are no built-in tools for checking the consistency of the meta data for the linked or grouped items. **DamScan.py** solves this problem and reports inconsistencies in metadata for Daminion server and standalone catalogs.

The **DamScan.py** tool analyzes a Daminion catalog for potential inconsistencies in tagging between linked or grouped media items. It checks the values of these tags: *Place*, *GPS*, *Event*, *People*, *Categories*, *Keywords* and *Collections*. The command line parameter **-t**, as well as the `TagValues` key in the `INI_FILE` allow you to restrict the verification to fewer tags.

It is also possible to verify, if the name parts of the files associated to linked or grouped media items are fully or partially identical. The command line options provide the capability to configure the process of the analysis as well as contents and format of its output.

For single value tags (*Place*, *GPS* and *Event*) and the program reports both file names and both tag values. For the multi value tags (*People*, *Categories* and *Keywords*) the program will report both filenames and which tag values **are missing** from the first file. If file name analysis is requested (option **-b**), the program compares if the specified part of the file names are identical and if not, report the full filenames of both media items.

The output is tab separated so it can be directly pasted or (when using option **-o**) imported into a spreadsheet for further formatting, sorting and analyzing.

By default the program analyzes the media items that are linked together and compares the tag values for each linked pair. Alternatively, when specifying the **-g** option, the behavior changes so that each image in a group (also called a stack) is compared with the top item.

Options:

--ini *INI_FILE*

A configuration file containing values for the command line options. If an option is specified both in the INI file and on the command line, the command line parameters have a precedence. See details in the section CONFIGURATION PARAMETERS.

-f, --fullpath Display the full path and not only the file name

-i, --id Display the Daminion Item Id after the filename

-g, --group Compare tags between grouped images instead of linked images, if this option is not specified, links are used

-t, --tags [{Event,Place,GPS,People,Keywords,Categories,Collections} [{Event ... }]]
Specify the tag categories to be checked. If the **-t** option is not specified all of the tag categories listed above are compared. Allowed values for the list are *Event*, *Place*, *GPS*, *People*, *Keywords*, *Categories* and *Collections*. Multiple values are separated by spaces (' ').

Specifying **-t** without any tag category disables the tag value checking. This is only useful with the option **-b** to check only for consistent file names.

-b, --basename [*SEPARATOR* [*SEPARATOR* ...]]

Compare the base names (the name part excluding the extension, such as .JPG, .CR2, .TIF ...) of the file names associated to all linked or grouped (option **-g**) media items. If the names do not match, they are reported. If no *SEPARATOR* values are specified only the file extension is excluded from the comparison.

If one or more *SEPARATOR* values are specified (as single characters like '_' or as character strings like 'BW'), the file name is scanned from right to the first occurrence of the first *SEPARATOR* and only the characters on the left are considered for comparison or for subsequent search for the next *SEPARATOR*, if specified. If the *SEPARATOR* is not found from the name, the process continues with the next specified *SEPARATOR*, if any.

If you want to eliminate multiple occurrences of the same *SEPARATOR*, you need to specify the *SEPARATOR* multiple times (e.g. **-b** _ _ makes the file IMG_1234_bw_lowres.JPG to match IMG_1234.JPG).

If the name used for comparison would become shorter than 8 characters, the current and remaining *SEPARATOR*s will be ignored. This will avoid situation where specifying '_' as a *SEPARATOR* IMG_0000.JPG would match IMG_9999.JPG.

If the **-b** option is not specified, only tag values are checked.

Note

- The resulting base name for comparison can be different depending on the order of the *SEPARATORS*.
- A hyphen ('-') is allowed as a *SEPARATOR* alone. Multi character *SEPARATORS* that start with a hyphen are not allowed, because they can be confused with the program options.

-x, --exclude *EXFILE*

The option **-x** specifies a configuration file that contains tag values that will be excluded from comparison. The file format is similar to the standard .INI files. See details in the section CONFIGURATION PARAMETERS. Either **-x** or **-y** parameter can be specified, not both. If neither option is specified all tag values are checked.

-y, --only *ONLYFILE*

The option **-y** specifies a configuration file that contains the only tag values that will be used for comparison. The file format is similar to the standard .INI files. See details in the section CONFIGURATION PARAMETERS. Either **-x** or **-y** parameter can be specified, not both. If neither option is specified all tag values are checked.

Using the option **-y** with the same file as used with the option **-x**, makes it possible to verify what was or will be excluded.

-a, --acknowledged *ACKFILE*

The option **-a** specifies a configuration file that contains acknowledged differences between linked or grouped media items. The differences listed in this file are excluded from the output. The output of the program, when used with the **-i** and **-o** option, can be used for a future run of the program with the **-a** option as an *ACKFILE*.

-l, --sqlite Use a standalone (based on SQLite) catalog instead of server catalog (based on Postgresql)**-c, --catalog** *CATALOG*

The **-c** option specifies a Daminion catalog name. If not specified the default Daminion catalog (*NetCatalog* or *DaminionCatalog.dmc*) is used. For standalone catalogs the full path and filename (including *.dmc*) must be specified.

-s, --server *SERVER*

Postgres database server (**Not** the Daminion Server). If *SERVER* is not specified, localhost will be used. You can verify the *SERVER* and *PORT* settings in the Daminion Server Administration panel.

-p --port *PORT*

Postgres database server port for the catalog. If not specified, the Daminion default 5432 will be used.

-u --user *USER*

Postgres database user/password (**Not** Daminion catalog user). If not specified the installation default *postgres/postgres* will be used.

-o --output *OUTFILE*

Write the report to an *OUTFILE*. If **-o** is not specified the output will be printed on the screen. Verbose messages (**-v**) are never directed to *OUTFILE*. To print the output on screen, when an output file specified in an INI file, a special value *<stdout>* can be used.

-v, --verbose Verbose output. Specifying the option **-v** displays a running counter, and the current Item Id and filename on the screen.

If a second v is added to the option (**-vv**) also information of the linked or grouped pairs is printed. This output is always displayed on the screen (stdout) and not directed to the *OUTFILE* specified with the option **-o**.

-h, --help Show help message and exit

--version Display version information and exit.

DIAGNOSTICS

Errors and warnings are logged to the standard error stream and the diagnostic output to the standard output or the specified *OUTFILE*. If **-v** is not used, then no output means that no discrepancies were detected.

DamScan.py terminates with zero exit status if it was able to scan through the whole catalog.

Only the specified tag categories (Event, Place, GPS, People, Keywords, Categories) are checked, not all tag categories that are supported by Daminion. The other tag categories not in the list (e.g. Media Format, Rating, Project etc.) are ignored by **DamScan.py**.

When importing the output file into Excel, you have to select in import wizard at Step 1 **File origin:** 65001 : *Unicode (UTF-8)*. This will import the accented and diacritic letters correctly.

Using **-g** option requires time on larger catalogs, because the scanning is n^2 dependent on the number of items in catalog. For example, for 150.000 items the analysis needs 1–2 hours, when the database and all executables where stored on an SSD and twice as much time on a hard disk.

ENVIRONMENT

Python

Install Python 3.x from <http://www.python.org>. After you have downloaded Python package right click the package and select "*Run as administrator*". In the installation dialog select Customized installation. In the customized configuration tick to include Python in the PATH and select installation for all users. Other options can be left to defaults.

To activate the PATH settings, Windows should be restarted before starting Python for the first time.

psycpg2

After installing Python start an elevated command window (*Run as Administrator*), because the Postgres support package will be installed in the Program Files directory. Enter commands

```
C:> python -m pip install -U pip setuptools
C:> python -m pip install psycpg2
```

CONFIGURATION PARAMETERS

INI_FILE

The file consists of two sections: Database and Session. Each line consists of one parameter, '=' sign and the parameter value. It's recommended to use full pathnames, so the INI file works independent of directory, where the **DamScan** program is run. The allowed parameters and there default values are:

[Database]

- | | |
|----------------|--|
| SQLite | Use a standalone (based on SQLite) catalog instead of server catalog, command line option -l/--sqlite . Allowed values: True/False. If defined True this cannot be reset on command line. |
| Catalog | The Daminion catalog name (-c/--catalog). If SQLite is True, the default is DaminionCatalog.dmc, else the default is NetCatalog. |
| Server | Postgres database server (-s/--server). The default is localhost. |
| Port | Postgres database server port (-p/--port). The defaults is 5432. |
| User | Postgres database user/password (-u/--user). The default is postgres/postgres. |

[Session]

- | | |
|-----------------|--|
| Fullpath | Display the full path and not only the file name (-f/--fullpath). Allowed values: True/False. The default value is False. If defined True this cannot be reset on command line. |
|-----------------|--|

ID	Display the Daminion Item Id after the filename (-i/--id). Allowed values: <code>True/False</code> . The default value is <code>False</code> . If defined <code>True</code> this cannot be reset on command line.
Group	Compare tags between grouped images (-g/--group) if <code>True</code> . If this option is not specified or <code>False</code> , links are used. Allowed values: <code>True/False</code> . If defined <code>True</code> this cannot be reset on command line.
Tags	Specify the tag categories to be verified (-t/--tags). Allowed values for the list are <code>Event</code> , <code>Place</code> , <code>GPS</code> , <code>People</code> , <code>Keywords</code> , <code>Categories</code> and <code>Collections</code> . Multiple values are separated by spaces (‘ ‘). If the list is empty, no categories are checked. If this parameter is not specified, or it has a special value <code>All</code> , all of the tag categories allowed are checked.
Basename	Compare the base names of the file names associated to all linked or grouped media items (-b/--basename). If the parameter is not specified filenames are not checked. If the parameter value is empty, only the extension is ignored for the comparison, if additional characters or strings separated by a space (‘ ‘) are specified they are used as additional delimiters. See more detailed descriptions of the -b/--basename command line option. The restriction of using multi character <i>SEPARATORS</i> that start with a hyphen doesn’t apply to <code>Basename</code> parameter.
Exclude	Specifies the <i>EXFILE</i> configuration file that contains the tag values that will be excluded from comparison (-x/--exclude).
Only	Specifies the <i>ONLYFILE</i> configuration file that contains the tag values that will be used for comparison (-y/--only). If both <code>Exclude</code> and <code>Only</code> parameters are specified, a warning message is issued and the <i>ONLYFILE</i> is ignored.
Acknowledged	This parameter specifies a configuration file that contains acknowledged differences between linked or grouped media items (-a/--acknowledged).
Verbose	Verbose output. (-v/--verbose or -vv). Allowed values are 0 (default, no output), 1 and 2. If this parameter has value 1 or 2, it cannot be reset to 0 on command line. You can only change between values -v and -vv .
Output	Write the report to the specified file (-o/--output). If the parameter is not specified, the value is empty or the value is a special value <code><stdout></code> the output will be printed on the screen.

Below is an example INI file that corresponds to the defaults and has rest of the parameters commented out.

```

[Database]
SQLite=False
Catalog=NetCatalog
Server=localhost
Port=5432
User= postgres/postgres

[Session]
Fullpath=False
ID=False
Group=False
Tags=Event Place GPS People Categories Keywords Collections
# or
# Tags=All

# Basename=
# strip only extension for file name comparison
# Basename=_ -
# strip extension, up-to last '_' and after that up-to last '-'

# Exclude=tagvalues.ini
# Only=tagvalues.ini
# Acknowledged=ack_pairs_ini.txt
Output=<stdout>
Verbose=0

```

EXFILE/ONLYFILE and TagValuesFile

The configuration parameters in the *EXFILE/ONLYFILE* follow the standard INI file structure. Each tag category can be specified in brackets ([category]) and the tag values below the section. If the tag value is hierarchical, the separator between hierarchy levels is '|'. If you specify only top items of a value hierarchy the filter applies to all child values.

Both the category names and tag values are case sensitive. You can specify also comment lines starting with '#' or ';'.

Below is an example configuration file.

```

[People]
# will be an exact match
Lintula|Juha

[Categories]
# will match to Image|B&W, Image|HDR, Image|Panorama, ...
Image

[Place]
# will match all cities and locations in Germany|Bavaria
Germany|Bavaria

[GPS]
# GPS coordinates not specified
0N 0E 0m

```

If you use the above contents as an EXFILE or TagValuesFile (combined with TagValues=Exclude) the values are excluded from the verification, but all the other tag values are verified. If you use the contents as an ONLYFILE or TagValuesFile combined with TagValues=Include, the values would be the only ones to be verified.

ACKFILE

The configuration parameters in the *ACKFILE* follow the structure of the program's output file. Each line contains an image pair, a tag category and the acknowledged differences. For the single value tags there is no need to specify any values. If the same image pair, tag category combination appears several times, all the listed tag values are taken into consideration.

The format for a single value tags (Name, Place, GPS or Event) is

```
file1 (id1)<TAB>sep<TAB>file2 (id2)<TAB>Name|Place|GPS|Event
[<TAB>'value1'<TAB>sep<TAB>'value2']
```

and an example

```
160330_4440.JPG (7398)    <>    160330_4440.JPG (6670)    Place
sep is usually <>, < or >, but it could be anything. The vertical bar character ('/')
separates different options and '[text]' optional values.
```

For the multivalue tags (People, Keywords, Categories or Collections) the line is

```
file1 (id1)<TAB>sep<TAB>file2 (id2)
<TAB>People|Keywords|Categories<TAB>'value1'[, 'value2'[, ...]]
```

and an example

```
160330_4456.JPG (6686)    <    160330_4456-001.JPG (7074)    Keywords
'Palazzo|Collicola'
```

If you have hierarchical tags like 'Palazzo|Collicola', use the vertical bar character ('/') as a separator between different hierarchy levels.

If you run **DamScan** using option '**-i -o c:\DamScanResults.txt**', do not change the contents of file *DamScanResults.txt* and run **DamScan** again with option '**-a c:\DamScanResults.txt**', it does not report any mismatches, unless new inconsistent pairs have been added to the DamScan catalog.

EXAMPLES

Examples below assume that you have DamScan.py in your home directory (C:\Users\user) and your local catalog and the configuration files are in the Pictures sub-directory.

```
C:> python DamScan.py -s ServerPC -p 5433
```

Run the analysis of the NetCatalog server catalog, based on linked images. The Postgres database is set up in ServerPC at port #5433.


```
C:> python DamScan.py -v -g -l -c Pictures\DaminionCatalog.dmc -o
Pictures\output.txt
```

Run the analysis of the local catalog DaminionCatalog.dmc in the Pictures directory, based on image groups. Print the results of the analysis in Pictures\output.txt and show basic progress information on the screen.

```
C:> python DamScan.py -t Place GPS -c NewCatalog -p 5433
```

Run the analysis only on Place and GPS tags (not on Event, People, Categories or Keywords) of the NewCatalog server catalog, based on linked images. The Postgres database is set up in localhost at port #5433.

```
C:> python DamScan.py -vv -x Pictures\ExcludeList.ini -c NewCatalog -o
Pictures\output.txt
```

Run the analysis on all tag categories of the NewCatalog server catalog, but exclude the tag values listed in file ExcludeList.ini from comparison. Store the results of the analysis in Pictures\output.txt and show detailed progress information on the screen.

```
C:> python DamScan.py -y Pictures\ExcludeList.ini -c NewCatalog
```

See on your display what was excluded from the report of the previous example.

```
C:> python DamScan.py -b -t -l -g -c Pictures\TestCatalog.dmc
```

Analyze only the filenames of the local TestCatalog.dmc, based on grouped images.

Only the file extension is ignored for the comparison. I.e. IMG_1234.CR2 will be the same as IMG_1234.JPG, but different from IMG_1234_lowers.JPG.

```
C:> python DamScan.py -b _ BW -t -l -g -c Pictures\TestCatalog.dmc
```

Same as the previous example, but more logic for comparing the file names. First the file extension is removed. Then everything from the last '_' and after that everything from the last 'BW'. As a result all files IMG_1234.JPG, IMG_1234_lowers.JPG, IMG_1234BW_lowers.JPG and IMG_1234BW.JPG will match IMG_1234.CR2.

Examples of the use of the option **-b**.

File name	-b	Result	Comment
IMG_1234.JPG	-b _	IMG_1234	IMG is less than 8 characters
IMG_1234_BW.JPG	-b _	IMG_1234	
IMG_1234_BW_lowres.JPG	-b _ BW	IMG_1234_	_: IMG_1234_BW BW: IMG_1234_
IMG_1234_BW_lowres.JPG	-b BW _	IMG_1234	BW: IMG_1234_ _: IMG_1234
IMG_1234_BW_lowres.JPG	-b _ _	IMG_1234	_: IMG_1234_BW _: IMG_1234

```
C:> python DamScan.py --ini Pictures\DamScan.ini -o <stdout>
```

Use the settings stored in an INI file Pictures\DamScan.ini, but override to print the output on screen.

SEE ALSO

[python](#), language description and syntax.

[psycopg2](#), Python-PostgreSQL Database Adapter.

LICENSE

The program is licensed under GPL3.

AUTHOR(S)

Juha Lintula