The batch script

Jonathan Hooper, Leeds. May 2022

Contents

Overview

The batch script provides a convenient way to run the individual steps across multiple modules with one command, while checking for errors from each script.

The original reason for providing this was that running the individual scripts across multiple reading lists in-one-go was too memory-hungry: the batch script allows processing two be split up into individual modules.

Under the hood, this just makes repeated system() calls to the various individual scripts *(see 1_3_the_individual_scripts.pdf).*

The system() function is used rather than include() in order to isolate the memory in each individual script call from each other, and from the parent batch script.

Alternatives

Instead of running the batch script, it is possible to use the individual scripts step-by-step and module-by-module - this may be useful e.g. during development or testing. See 1_3_the_individual_scripts.pdf.

Running the script

Script: **batch.php**

Usage: php batch.php **OPTIONS**

e.g. to fully process (including exporting reports) lists belonging to module PSYC3505:

```
php batch.php -m PSYC3505
```

Options

The script takes three options:

-m	List of module codes to process
-f	Filename of file containing list of module codes to process
-s	List of individual steps to run

Either -m or -f **must** be supplied *not both* - if both are supplied, -f is ignored

-s is optional - if not provided, all steps will be followed

-m option

This is a list of module codes, separated by commas, colons, semi-colons, full-stops or whitespace. e.g. to fully process lists for MODULE_A and MODULE_B:

```
php batch.php -m MODULE_A,MODULE_B
```

Module codes are searched for in the Ex Libris Courses API by ***non-exact*** *match on the* ***searchable_id*** *field*

-f option

This is the name of a file containing a list of module codes, separated by commas, colons, semi-colons, full-stops or whitespace (including newline/carriage-return, so we might typically have a module code on each line).

A possible place to save this file would be **Config/Modules** e.g. to fully process all the modules listed in a file Config/Modules/modcodes.txt:

```
php batch.php -f Config/Modules/modcodes.txt
```

-s option

If present, this is a list (separated by commas, colons, semi-colons, full-stops or whitespace) of individual steps to follow.

Possible steps are:

Step label	Activity	Scripts called
get	Get citations from Alma/Leganto	**getCitationsFromAlma.php**
alma	Enhance citations with Alma bib record data	**enhanceCitationsFromAlma.php**
scopus	Enhance citations with Scopus data	**enhanceCitationsFromScopus.php**

wos	Enhance citations with WoS data	**enhanceCitationsFromWos.php**
viaf	Enhance citations with VIAF data	**enhanceCitationsFromViaf.php**
export	Export spreadsheet reports	**simpleExport.php** **longExport/php**

e.g. to fetch Leganto citations and add data from Alma bib records for PSYC3505:

```
php batch.php -m PSYC3505 -s get,alma
```

Steps are not case-sensitive i.e. "Scopus" or "WoS" or "VIAF" could be used.

Steps are followed in the normal order (i.e. get -> alma -> scopus -> wos -> viaf -> export) and **not** any different order in the -s option. i.e. these are equivalent:

```
php batch.php -m PSYC3505 -s get,alma
php batch.php -m PSYC3505 -s alma,get
```

To fully process reading lists for module PSYC3505 a step-at-a-time - e.g. to allow checking inbetween:

```
php batch.php -m PSYC3505 -s get
php batch.php -m PSYC3505 -s alma
php batch.php -m PSYC3505 -s scopus
php batch.php -m PSYC3505 -s wos
php batch.php -m PSYC3505 -s viaf
php batch.php -m PSYC3505 -s export
```

To separate the Leganto/Alma steps from the searching of external sources, and from the export of reports (this is quite a natural division):

```
php batch.php -m PSYC3505 -s get,alma
php batch.php -m PSYC3505 -s scopus,wos,viaf
php batch.php -m PSYC3505 -s export
```

If -s is absent, all steps are followed in the normal order, i.e. the following single command is equivalent to groups of commands above:

```
php batch.php -m PSYC3505
```

Errors

If there is an error in any individual script, the batch script terminates - it is only possible to do the full process using the batch script if every step completes without error. *(This may be a defect - it may be preferable to complete the full process, and leave the operator to decide whether any errors were serious enough the invalidate the batch).*

Output

If running the complete process (no -s option) or running the export step (-s export) the output is exactly as when running the individual export scripts *(see 1_3_the_individual_scripts.pdf)* i.e. a set of CSV reports in the directory **Data/**

For the previous steps (get, alma, scopus, wos, viaf), the individual scripts read from STDIN and write to STDOUT, so it is up to the batch script to choose file locations:
Individual scripts called from the batch script write to temporary .json files in the directory **Data/tmp/** named for the module and the individual step. If the step completes without error, this temporary file is copied by the batch script to a .json file in the directory **Data/** named for the module.

Subsequent individual scripts called from the batch script read input from the .json file in the directory **Data/** (not from the temporary file in **Data/tmp/**).

Batch-level summary report

See 3_1_export_scripts.pdf for more details

The short report (**simpleExport.php**) normally produces a **Summary.CSV** spreadsheet listing reading lists for just those modules it was called for. But when processing a batch of modules we want a **single** Summary.CSV including **all** the modules in the batch.

The batch script uses simpleExport.php's **-i** (initialise) and **-a** (append) *options (see document "The export scripts")* to produce a Summary.CSV report for the whole batch.

Putting it together

What happens under-the-hood when calling the batch script with all steps, for more than one module:

```
php batch.php -m PSYC3505,BLGY3135
```

- **php simpleExport.php -i**
  Create an empty Data/Summary.CSV file with column headings
- **php getCitationsByModule.php -m PSYC3505 >Data/tmp/PSYC3505_L.json**
  Write reading list citations to temporary .json file
- **Copy("Data/tmp/PSYC3505_L.json", "Data/PSYC3505.json")**
  Ready to be passed to next script
- **php enhanceCitationsFromAlma.php <Data/PSYC3505.json >Data/tmp/PSYC3505_A.json**
  Take citations from previous call, enhance with Alma data, and save to temporary .json file
- **Copy("Data/tmp/PSYC3505_A.json", "Data/PSYC3505.json")**
  Ready to be passed to next script
- **php enhanceCitationsFromScopus.php <Data/PSYC3505.json >Data/tmp/PSYC3505_S.json**
  Take citations from previous call, enhance with Scopus data, and save to temporary .json file
- **Copy("Data/tmp/PSYC3505_S.json", "Data/PSYC3505.json")**
  Ready to be passed to next script
- **php enhanceCitationsFromWos.php <Data/PSYC3505.json >Data/tmp/PSYC3505_W.json**
  Take citations from previous call, enhance with WoS data, and save to temporary .json file

- **Copy("Data/tmp/PSYC3505_W.json", "Data/PSYC3505.json")**
  Ready to be passed to next script
- **php enhanceCitationsFromViaf.php <Data/PSYC3505.json >Data/tmp/PSYC3505_V.json**
  Take citations from previous call, enhance with VIAF data, and save to temporary .json file
- **Copy("Data/tmp/PSYC3505_A.json", "Data/PSYC3505.json")**
  Ready to be passed to next scripts
- **php simpleExport.php -a <Data/PSYC3505.json**
  Create short report(s) **Data/LISTCODE.CSV** and append summary to **Data/Summary.CSV**
  LISTCODE is the code of the reading list belonging to PSYC3505. There will be more than one file if PSYC3505 has more than one reading list.
- **php longExport.php -a <Data/PSYC3505.json**
  Create long report(s) **Data/LISTCODE_LONG.CSV**
- **php getCitationsByModule.php -m BLGY3135 >Data/tmp/BLGY3135_L.json**
  Write reading list citations to temporary .json file
- **Copy("Data/tmp/BLGY3135_L.json", "Data/BLGY3135.json")**
  Ready to be passed to next script
- **php enhanceCitationsFromAlma.php <Data/BLGY3135.json >Data/tmp/BLGY3135_A.json**
  Take citations from previous call, enhance with Alma data, and save to temporary .json file
- **Copy("Data/tmp/BLGY3135_A.json", "Data/BLGY3135.json")**
  Ready to be passed to next script
- **php enhanceCitationsFromScopus.php <Data/BLGY3135.json >Data/tmp/BLGY3135_S.json**
  Take citations from previous call, enhance with Scopus data, and save to temporary .json file
- **Copy("Data/tmp/BLGY3135_S.json", "Data/BLGY3135.json")**
  Ready to be passed to next script
- **php enhanceCitationsFromWos.php <Data/BLGY3135.json >Data/tmp/BLGY3135_W.json**
  Take citations from previous call, enhance with WoS data, and save to temporary .json file
- **Copy("Data/tmp/BLGY3135_W.json", "Data/BLGY3135.json")**
  Ready to be passed to next script
- **php enhanceCitationsFromViaf.php <Data/BLGY3135.json >Data/tmp/BLGY3135_V.json**
  Take citations from previous call, enhance with VIAF data, and save to temporary .json file
- **Copy("Data/tmp/BLGY3135_A.json", "Data/BLGY3135.json")**
  Ready to be passed to next scripts
- **php simpleExport.php -a <Data/BLGY3135.json**
  Create short report(s) **Data/LISTCODE.CSV** and append summary to **Data/Summary.CSV**
- **php longExport.php -a <Data/PSYC3505.json**
  Create long report(s) **Data/LISTCODE_LONG.CSV**

Final outputs in the above example:

- **Data/Summary.CSV**
  Report summarising data for PSYC3505 and BLGY3135

- For each reading list belonging to PSYC3505 or BLGY3135:

  - **Data/{LISTCODE}.CSV** - short report

  - **Data/{LISTCODE}_LONG.CSV** - long report

- **Data/PSYC3505.json** - raw data from module PSYC3505, with all enhacements from Alma, Scopus, WoS and VIAF

- **Data/BLGY3135.json** - raw data from module BLGY3135, with all enhacements from Alma, Scopus, WoS and VIAF