



## Using the Specify Command Line Data Export Program

The Specify **Command Line Data Export Program** (ExpCmdLine) is a software tool which enables database administrators to automate the process of producing exported collection records to update a Specify Web Portal, or to create tab-delimited text files of exported collection data for other uses.

Specify has a flexible mechanism for identifying which data fields will be included in an exported data file. Setting up a pipeline for exporting data includes a few steps and some additional software tools. The Specify **Schema Mapper** is an embedded tool accessible from the “System” menu for manager-level users. It enables the creation of an ‘**Export Data Map**’, which is a custom mapping between the data fields used in a Specify database and the data concepts one wishes to include and populate in an export file. The Specify **Data Exporter** application, a separate program included with Specify, is an interactive tool which uses an Export Data Map to copy the data in the mapped fields to a single table within the Specify MySQL database. This flat table in MySQL is an ‘index’ or ‘cache’ copy of the export data. Once an internal cache is created, the **Data Exporter** can either export the data as a flat, tab-delimited, text file (e.g. for GBIF’s IPT software to create a DarwinCore Archive data package), or data can be exported and associated configuration files packaged inside of a .zip file which can be saved and used for updating a Specify Web Portal.

The **Command Line Data Export Program**, included in the Specify installation, is similar in function to the Specify **Data Exporter**; it uses Export Data Maps to update a flat table internal cache and can also export that data to a tab-delimited test file or to a Web Portal zip package. The main difference between the two applications is that the Data Exporter is interactive whereas the Command Line Data Exporter can perform these actions in a “headless” environment which can be scripted.

- The first time an Export Data Map is made with the **Schema Mapper** function, Specify’s interactive **Data Exporter** application (not **ExpCmdLine**) must be used to create the initial MySQL cache copy and to export the zip file package which is used by the Specify Web Portal. (See separate documentation “Using the Specify Web Portal” available on the Specify Software Project web site for description of the steps for creating an Export Data Map and exporting data the first time.)

After the first export, **ExpCmdLine** replaces the need to use the interactive Specify Data Exporter application for pushing recurring data updates from a Specify database. Web Portal updates can then be automated by using **ExpCmdLine** in a scripted, scheduled job from a server account.

- If the Specify data mapping is changed because the Specify Schema Mapper was used to add/remove/shift/change mapped concepts, or to change titles for output fields, or

to change record selection conditions, then the internal, flat table, MySQL cache or index must be rebuilt using the Specify **Data Exporter** tool (not **ExpCmdLine**). If an internal cache rebuild is necessary **ExpCmdLine** will fail and not complete a Portal update. After a rebuild of the internal cache, this tool can then again be used for scripting recurring updates to update the internal cache and separately to export the Web Portal zip package by using the appropriate argument to the `-a` parameter.

**ExpCmdLine** must be run inside of a shell or terminal window. The program is located in the Specify installation directory under `../Specify/bin`).

- A Specify manager-level user account is required to use this tool.

The **ExpCmdLine** program can be installed and run from a server computer but it requires access to several files which must be copied to the server. The full Specify package does not need to be installed on the server.

Files needed on the server to run **ExpCmdLine** include:

- Specify.jar
- Specify installation config directory along with its contents (`../Specify/config/`).
- user.properties

These can be copied from a Specify workstation installation configured to use that collection. The first two are located in the Specify software installation directory, user.properties is located in a user directory on a workstation configured to access Specify (see file location below).

Although specify.jar and /config directory of files can be placed anywhere on the machine running **ExpCmdLine**, a convenient place to put them would be in a new directory called /specify on the server along with the copy of the user.properties file.

The `-w` parameter tells **ExpCmdLine** where to look for the Specify.jar file and the /config directory--where they have been copied.

- **ExpCmdLine** requires access to the encrypted user account password key inside the user.properties file and it must be for a Specify user account at the Specify Manager level. The encrypted password key can be copied from a workstation where a Specify manager level user was originally created and edited into the user.properties file as a new line.

On Windows 7, the user.properties file for a Specify user is located in:

`C:\Users\[UserName]\AppData\Local\Specify\user.properties`

On Linux, the directory holding the user.properties file is:

`~/Specify/user.properties`

where ~ is the home directory of the Linux user.

The encryption key entry in the user.properties file, looks like this:

```
fishes_manuser1_master.path=419D54F662B15FE67B8ADB62A5CB23A587CA1DA802EC8813
```

In this example, 'fishes' is the name of the database and 'manuser1' is user name for a Specify manager-level user.

## Using the ExpCmdLine Program

Parameters and arguments of **ExpCmdLine**:

Parameter	Argument	Required
-u	The username of a (manager-level) Specify user	Yes
-p	The user's password	Yes
-d	The database name	Yes
-m	The name of the data mapping (contained within the Specify database)	Yes
-a	The action to be performed. Options: "Update", "ExportToTabDelim", "ExportForWebPortal"	Yes
-l	The name of the log file	No, defaults to stdout
-h	IP name or address of the MySQL database server	No, defaults to localhost
-o	The name of the output file	No
-w	Path to: "../Specify" installation directory	Yes

## Example Usage

**Commands and parameters are shown in bold.** Replace the arguments in *Courier font text* with arguments appropriate for your Specify configuration. If a space exists within an argument, it must be enclosed in single or double ASCII quotes:

e.g.: `-w "C:/Program Files/Specify/bin"`

### ➤ **Update the internal Specify, MySQL cache (index):**

```
ExpCmdLine -u username -p password -d database name -m name  
of mapping -a Update -l name of log file -h IP name of  
database server -w Specify installation directory path
```

➤ **Solr index, package, and export the cache to the web portal:**

```
ExpCmdLine -u username -p password -d database name -m  
name of mapping -a ExportForWebPortal -l name of log file  
-h IP name of database server -o name of output file -w  
Specify installation directory path
```

➤ **Export the contents of the MySQL cache/index to a tab-delimited text file:**

```
ExpCmdLine -u username -p password -d database name -m name  
of mapping -a ExportToTabDelim -l name of log file -h IP  
name of database server -o name of output file -w Specify  
installation directory path
```

The spelling of the Data Exporter command differs by operating system. Substitute “**ExpCmdLine**” in the examples above with the appropriate version shown below. The path to the Specify installation directory (./Specify) must be specified with the **-w** parameter, even if **ExpCmdLine** itself is located in OS path or in the “current” directory.

Operating System	Command
Linux	./ExpCmdLine.sh
Mac OS X	./ExpCmdLine
Windows	ExpCmdLine

Use **ExpCmdLine** with a scripting language to automate and schedule the execution of internal data cache rebuild and data export operations.