

+1Copybook Command Line Options for SQL Server

March 2021

Project Directory

Most users will create a new directory as their project directory. +1Copybook configuration files, like `database_id.txt`, and “touch” files are found in the project directory.

For small projects, a `copybook.txt` file and data file(s) can be found in the project directory. The `copybook.txt` file can contain one or more COBOL copybooks. At the top of each COBOL copybook, you need to add a line like `ABC TABLE` (or `TABLE ABC`) where `ABC` is the table name to be created in the SQL Server database. If a data file for the copybook exists, a copy of the data file can be stored in `ABC.txt` in the project directory.

For larger projects, the `COPYBOOKS` and `DATA` subdirectories in the project directory can be created to better organize the project.

The `COPYBOOKS` directory contains one or more COBOL copybook files. Each file has one COBOL copybook defined. Each copybook needs to start with `table_name TABLE` or `TABLE table_name` to match the SQL Server table you want to create in the database.

The `DATA` directory contains one or more data files to load into the SQL Server database. If a copybook is for the `ABC TABLE`, the data file is named `ABC.txt` or `ABC.deli`.

COPYBOOKS The `COPYBOOKS` directory contains one or more COBOL copybooks. Each copybook has `table_name TABLE` as its first line.

DATA Contains data files for copybooks.

When +1Copybook is run, the following subdirectories are created or overwritten by +1Copybook:

BULK_INSERT Contains generated BULK INSERT scripts to load data into tables.

CREATE_TABLES Contains generated CREATE TABLE scripts.

EXCEL Contains EXCEL loadable files describing the SQL Server tables (-x option).

FORMAT_FILES Contains FORMAT FILES used by BULK INSERT scripts.

html/index.html Open file `index.html` in the `html` directory to display the generated HTML output for conversions. HTML includes tables, sorted tables, copybook mapping, create table script, format files, and bulk insert scripts for each copybook.

The HTML output is very useful and can be viewed by opening the `html/index.html` file using IE or Chrome. To view a file in Chrome or Edge, press Control-O and then select the `index.html` file. The generated `copybooks.txt`, `copybooks.txt.LIST`, `control_files.txt` and `create_tables.txt` files can be ignored and are used by +1Data to generate the HTML output.

Help Output

% cb -h

To run +1Copybook for SQL Server, type:

```
cb -[#]124AbBcCdDEfFhHilLMnOpPsSTvwxy]
```

where

```
-# 99 Load only n rows of data from data files. Used for testing.
-] 99 If table over 1000 columns, split table. First table 99 columns.
-1 Do not print first line of data in copybook mapping.
-2 To debug data, print column name with data values in deli file.
-4 Add SPARSE to each column in the CREATE TABLE script.
-A Data files are in ASCII format (default).
-b Generate output in both upper and lower case.
-B Display copybook.txt lines as they are processed.
-c Data files are in character format.
-C Create tables in SQL Server database.
-d dt Specify a date (dt) for data files. E.g., -d 12252021 for
  DATA/<tablename>_12252021.txt file.
-D Do not convert EBCDIC data into ASCII data stored in .deli files.
-E Data files are in EBCDIC format.
-f fn Specify a file name (fn) that contains a copybook.txt file.
-F Do not include FILLER columns to CREATE TABLE and CONTROL FILE.
-h Display this help message on usage.
-H Display names and basic format for +1Copybook configuration files.
-i Do not add counter 999 to FILLER column names.
-l Generate output in lower case.
-L Load data into SQL Server database using BULK INSERT scripts.
-M Display MAC addresses for your system.
-n Add NOT NULL constraint to each column in the CREATE TABLE script.
-O # Line offset. Specify column number (#) to start reading copybook.
-p COMP-3 Packed Decimals in copybook are not compressed.
-P Change path before running +1Copybook. Use in cronjobs.
-s Use SMALLDATETIME data type for dates.
-S Save one dated version of HTML output in index.html file.
-T fn Convert COBOL copybook files listed in file fn.
-v Display +1Copybook version and expiration date.
-w Do not print copyright notice.
-x Generate Excel output.
-y If copybook column contains DATE, DTE, or DT anywhere in its
  name, define column as a DATE data type.
```

Leaving +1Copybook. Have a nice day.

```
*****
```

```
Copyright (C) 2005-2021 +1 Software Engineering LLC
All Rights Reserved
```

% cb -H

SQL Server configuration files for +1Copybook:

```
acronymns.txt - [Acronym Name]
copybook.txt - Contains one or more COBOL copybooks.
database_id.txt - [Database Id]
date.txt - [MMDDYYYY] - Date of data file, e.g., 12252021
dos_paths.txt - Line 1:[Project Directory] Line 2:[Data Directory]
sparse.txt - [Table Name] - Add SPARSE to each column in CREATE TABLE script.
```

Leaving +1Copybook. Have a nice day.

```
Copyright (C) 2005-2021 +1 Software Engineering LLC
All Rights Reserved
```

Commonly Used Options

- b | -l** **Generate column names in both upper and lower case (-b) or lower case only (-l).**
Default: Column names are in upper case.
- C** **Create tables in the database identified in the database_id.txt file.**
- d date** **Specify date of data files to load.**
Using `-d 12252021` will search for data file `CPY123_12252021.txt`.
- E** **Data files are in EBCDIC format.**
`+1Copybook` will convert EBCDIC data into ASCII data and store the ASCII data in `*.deli` files (delimiter files).
- f name** **Specify a copybook name to convert.**
Converts the named COBOL copybook into CREATE TABLE and BULK INSERT scripts.
- L** **Load the data into database identified in database_id.txt file using BULK INSERT scripts.**
- n** **Add NOT NULL constraint to each column in the CREATE TABLE script.**
- s** **Use SMALLDATETIME instead of DATETIME for dates.**
- y** **If copybook column contains DATE, DTE, or DT anywhere in its name, define column as a DATE data type.**

Examples

- `% cb`
`-- Converts all copybooks found in copybook.txt file. If copybook.txt file not found in project directory, converts all copybooks found in the COPYBOOKS directory.`
- `% cb -C`
`-- Converts all copybooks and creates tables in SQL Server database identified in database_id.txt file.`
- `% cb -C -L`
`-- Converts all copybooks and creates then loads data into SQL Server database identified in database_id.txt file.`
- `% cb -f FF0123A`
`-- Converts copybook FF0123A found in a file named FF0123A or FF0123A.txt.`
- `% cb -b -n -y`
`-- Creates scripts in upper and lower case, assigns copybook columns with DATE, DTE, or DT in column name to a DATE data type, and appends NOT NULL to every column.`
- `% cb -C -4 -E -L`
`-- Creates tables adding SPARSE attribute to CREATE TABLE columns, converts EBCDIC into ASCII data, then loads data into the database identified in database_id.txt file.`

All Options Defined

- # 99 Load 99 rows of data from data files.**
Generates the first 99 rows of data from each data file used for testing.

- J 99 If table over 1024 columns, split table. First table is defined with 99 columns.**
If a copybook defines over 1024 columns, which is SQL Server's maximum, split table into table A with 99 columns and table B with the remaining columns. While there are limitations, SQL Server can support up to 30,000 columns using wide tables.

- 1 Do not print first line of data in copybook map.**
Default: If data file exists, first line of data is displayed in HTML copybook map.

- 2 To debug data, write "column name=data value" for each value in deli files.**
Default: Only data is found in data files.

- 4 Add SPARSE to each column in the CREATE TABLE script.**
SPARSE is needed when the number of columns in a SQL Server database exceeds the maximum of 1024 columns.

- A Data files are in ASCII format (default).**
A value of -123.45 may look like 12345{ in the data file. The decimal point is implied.

- b Generate column names in both upper and lower case.**
Default: Column names are in upper case.

- B Display copybook.txt lines as they are processed.**
Default: Copybook.txt lines are not displayed as they are processed.

- c Data files are in character format.**
A value of -123.45 will look like -123.45 in the data file.

- C Create tables in SQL Server database.**
Generates CREATE TABLE scripts then creates tables in database found in database_id.txt file.

- d date Specify date of data files to load.**
Using -d 12252020, +1Copybook will search for the data file in the following order: FF0001A_12252020.txt (current directory), DATA/FF0001A_12252020.txt (DATA directory), and then DATA/12252020/FF0001A_12252020.txt (DATA/12252020 directory).

- D Do not convert EBCDIC data into ASCII data files stored in deli files.**
This option is the same as using touch file NODELI.

- E Data files are in EBCDIC format.**
+1Copybook will convert EBCDIC data into ASCII data found in *.deli files (delimiter files). This option is the same as using touch file EBCDIC.

- f file** **Specify a file name (file) that contains a copybook.txt file.**
Default: The copybook.txt file name is used to store the copybook.
- F** **Do not include FILLER columns in CREATE TABLE scripts.**
Default: FILLER's are included and are numbered incrementally.
- h** **Display +1Copybook options with brief description.**
- H** **Display names and basic format for +1Copybook configuration files.**
- i** **Do not add counter 999 to FILLER column names.**
- l** **Generate column names in lower case.**
Default: Column names are in upper case.
- L** **Load data into database using generated BULK INSERT scripts.**
Default: Data is not loaded.
- M** **Display MAC addresses defined on system running +1Copybook.**
Default: MAC addresses are not displayed.
- n** **Add NOT NULL constraint to each column in the CREATE TABLE script.**
Default: NOT NULL constraint is not added in CREATE TABLE script.
- O #** **Read Copybook lines starting in position #.**
Default: No offset is used.
- p** **Packed decimal values in data file are not compressed.**
Default: Packed decimal (e.g., COMP-3) are in packed decimal format.
- P path** **Change to specified directory path as part of running +1Copybook.**
Default: +1Copybook runs in the directory it is started in.
Useful if running +1Copybook in a cron job.
- s** **Use SMALLDATETIME data type for dates.**
- S** **Save one dated version of HTML output in index.html file.**
This option will save the current HTML output. Run -s again and the current version will replace the saved version.
- T fn** **Convert copybooks listed in filename fn.**
A "list" is one or more filenames containing copybooks. Using two or more lists, you can convert multiple copybooks and data in parallel on multiple CPU systems.
Default: Copybooks are converted serially based on their order in the copybook.txt file.
- v** **Display +1Copybook version and expiration date.**
Default: Version and expiration dates are not displayed.

- w** **Do not print warning messages or copyright notice on startup.**
Default: Warnings and copyright notice is displayed.
- x** **Generate Excel loadable copybook mapping output.**
Default: Excel output is not created.
Excel files are found in the EXCEL subdirectory.
- y** **If copybook column contains DATE, DTE, or DT anywhere in its name, define column as a DATE data type.**
- Y** **Display list of copybooks already converted using +1Copybook license.**
Default: List of copybooks is not listed.
- H** **Display basic format of configuration files.**
Default: Format for configuration files are not displayed.
- ?**
- h** **Display +1Copybook options with brief description.**

Additional, undocumented options are supported. Contact +1 Software Engineering if desired option isn't found above.

Touch Files

A touch file is the same as specifying the option on the command line when running +1Copybook. By using touch files, you simply don't have to specify the option each time you run +1Copybook.

To create a touch file, run **touch LOWERCASE**. To remove, run **rm LOWERCASE**. To group touch files together when using the ls command, touch files can begin with the '_' character, such as _NOFILLER.

ASCII or _ASCII	Data in character format.
CHAR or _CHAR	Data in character format.
EBCDIC or _EBCDIC	Data in EBCDIC format.
LOWERCASE or _LOWERCASE	Convert column names to lower case (-l).
MIXEDCASE or _MIXEDCASE	Covert column names to be mixed case (-m).
NOFILLER or _NOFILLER	FILLER's will not be included in CREATE TABLE script.
TRUNCATE or _TRUNCATE	Truncate tables before loading data. Default: APPEND.
UPPERCASE or _UPPERCASE	Covert column names to be upper case (default, -u).

Configuration Files

A configuration file allows you to customize the generated output.

acronymns.txt	List acronyms one per line. If acronym is seen in COBOL copybook column name when using lower case (-l) or mixed case (-b), the acronym will be capitalized in the column name.
copybook.txt	Contains one or more COBOL copybooks. At the top of each COBOL copybook, add <table_name> TABLE.
database_id.txt	[Database SID] Sets the database id. Default database id is PLUSONE. When the -C or -L option used, specifies the SQL Server database to create and/or load data into.
date.txt	[MMDDYYYY] Specifies the date of the data files to convert. Data files can be stored as DATA/<tablename>_MMDDYYYY.txt files or files found in DATA/MMDDYYYY/<tablename>_MMDDYYYY files.
dos_paths.txt	Specifies DOS paths to project and data directories. First line specifies the DOS path to project directory. Second line specifies the DOS path the data directory.
sparse.txt	Lists table names to append SPARSE to each column name in CREATE TABLE script. One table name per line.

Quick Tutorial

To run +1Copybook on your Windows-based system, you first need to install cygwin Unix. The cygwin download link is: <https://cygwin.com/install.html>

After installing cygwin and +1's cb and de programs onto your Windows system, you can then convert one or more COBOL copybooks into SQL Server tables.

To Convert and Load One COBOL Copybook

% **mkdir my_directory**

% **cd my_directory**

% **cp /path/to/your/copybook.txt copybook.txt** ←copybook.txt file has a copy of your copybook.

% **Edit copybook.txt file.** ← Add **table_name TABLE** before the start of each copybook. You can use any editor, such as vi, notepad, or workpad to add table_name TABLE line.

% **cb** ← Run +1Copybook using default settings.

% **ls** ← Should see BULK_INSERT, CREATE_TABLES, FORMAT_FILES, and html directories.

View html/index.html ← In Chrome, type Control-O to view index.html.

Click on CREATE TABLE ← Click on the CREATE TABLE link.

Create Table ← Copy/paste the CREATE TABLE script from HTML and then run script in SQL Server query window.

Load Data ← Copy/paste the BULK INSERT script and then run script in SQL Server query window.

To Convert and Load Multiple COBOL Copybooks

Adding all copybooks in the COPYBOOKS directory and all data files in the DATA directory is the easiest way to use +1Copybook.

% mkdir my_directory

% cd my_directory

% mkdir COPYBOOKS

% mkdir DATA

% cp /path/to/copybooks/*.txt COPYBOOKS ← Copy all copybooks into COPYBOOKS directory.

% cp /path/to/data/*.txt DATA ← Copy all data files for above copybooks into DATA directory.

% cd COPYBOOKS ← Change to COPYBOOKS directory.

% Edit each copybook file ← Add **table_name TABLE** before the start of each copybook. You can use any editor, such as vi, notepad, or wordpad, to add **table_name TABLE** line to each copybook.

% cd .. ← Change to project directory.

Edit database_id.txt ← Edit file database_id.txt file. Add SQL Server database name to file.

% cb -C -L ← Run +1Copybook. The -C option will create and run the CREATE TABLE scripts and to create tables in the database. The -L option will create and run BULK INSERT scripts to load the data into the database tables. Database to use is identified in database_id.txt file. Use -d option if data file names contain date value. Use -4 to add SPARSE if there are more than 1024 columns being defined.

View html/index.html ← In Chrome or Edge, type Control-O to view index.html. Highly recommended.

View data in SQL Server ← Open SQL Server Management Studio to view created tables and data loaded.

Have questions?

Feel free to call +1 Software Engineering. Your point of contact is:

John Dempsey
+1 Software Engineering LLC
<http://plus1se.com>
805-586-3445
john@plus1se.com