ModusToolbox™ Local Content Storage CLI user guide

ModusToolbox™ tools package version 3.1.0

Local Content Storage Manager CLI tool version 1.0

About this document

Scope and purpose

This guide provides information and instructions for using the ModusToolbox™ Local Content Storage (LCS) Manager CLI tool.

Document conventions

Convention	Explanation
Bold	Emphasizes heading levels, column headings, menus and sub-menus
Italics	Denotes file names and paths.
`Courier New`	Denotes APIs, functions, interrupt handlers, events, data types, error handlers, file/folder names, directories, command line inputs, code snippets
File > New	Indicates that a cascading sub-menu opens when you select a menu item

Table of contents

1   Introduction

The LCS Manager CLI tool is a new feature in the ModusToolbox™ 3.1.0 tools package that replaces the older feature called offline content bundle. The LCS feature provides a command-line tool called lcs-manager-cli that allows you to create your own local content on demand. You can decide whether to save the entire database (approximately 5 GB of data) or a subset of data for the BSPs that you specify.

Note: *To use LCS, you must first be able to connect to the Internet in order to download content. Then, you can access the local content when needed without the Internet.*

After acquiring local content, you explicitly set whether to use local or online content. The rest of the ModusToolbox™ ecosystem will use what you select.

1.1   lcs-manager-cli options

The lcs-manager-cli tool is located in the *<path>/ModusToolbox/tools_3.1/lcs-manager-cli/* directory. If you run it with the −h option, you will see the following output:

```
$ ./lcs-manager-cli -h
Usage: <path>\ModusToolbox\tools_3.1\lcs-manager-cli\lcs-manager-cli.exe [options]

Options:
  -?, -h, --help       Displays help on commandline options.
  --help-all           Displays help including Qt specific options.
  -v, --version        Displays version information.
  --verbose            Enable verbose output
  --list               List the content in local content storage
  --add-all            Add all content from remote manifest to local content
                       storage
  --clear-all          Remove all content from local content storage
  --list-bsps          List the BSPs on the watch list
  --add-bsp <name>     Add the BSP to the watch list and add its dependencies
                       to local content storage
  --clear-bsp <name>   Remove the BSP from the watch list and remove its
                       dependencies to local content storage
  --check-for-updates  Check for available updates
  --update <name>      Update a repository and its dependencies.
  --update-existing    Update local content repositories.
```

2   Getting started

2.1   Specify local directory

By default, using lcs-manager-cli commands will create a local content directory in the following location:

<user-home>/.modustoolbox/lcs

To specify an alternate location, set the `MTB_LOCAL_CONTENT_PATH` environment variable to the desired path. Use forward slashes in the path, even on Windows. Do not use a Cygwin style path.

2.2   Create full database

To create a local copy of the entire database, run this command:

```
$HOME/ModusToolbox/tools_3.1/lcs-manager-cli/lcs-manager-cli --add-all
```

This creates a copy of the entire database from the remote manifest. It may take a long time depending on how many repos are in the manifest.

2.3   Create partial database

To create a local copy of a partial database, specify a particular BSPs using this command:

```
$HOME/ModusToolbox/tools_3.1/lcs-manager-cli/lcs-manager-cli --add-bsp <BSP Name>
```

Where `<BSP Name>` is a valid name of a BSP from the remote manifest. The specified BSP will be added to a file named *watchlist.json* in the *lcs* directory, and the BSP and all dependencies will be downloaded into local content storage.

Note: *In some cases when you use the `--add-bsp` command, the tool may not download all the necessary dependencies, resulting in situations where you cannot create the desired application. This can happen if a dependency is referenced by a .mtb file and not in the manifest. To solve this problem, we recommend running lcs-manager-cli with the `--add-all` option to download the full database.*
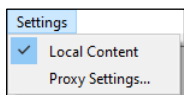
Examples of valid BSP names include CYW943907AEVAL1F, TARGET_PMG1-CY7112, and CY8CKIT-062S2-43012. See [GitHub manifest file](#) for an example of a BSP manifest.

2.4   Use local content with tools

After you have created local content, you can switch to LCS mode in the following ways. When using tools in LCS mode, messages will indicate that the given tool is using manifests from the local content storage location.

2.4.1   GUI option

Various tools that can use local content include the Library Manager, Project Creator, and BSP Assistant. For each of the tools, the GUI has a **Settings** menu item named **Local Content** that will toggle LCS mode on and off.

Setting LCS mode on or off for one tool automatically sets the same mode for other applicable tools.

2.4.2   Environment variable option

If you want to use LCS mode always, including for CLI tools, set the following environment variable:

```
MTB_USE_LOCAL_CONTENT=true
```

The value must be set to "true"; any other value will be ignored and LCS mode will not be enabled. When this variable is set, the GUI **Local Content** option will be enabled by default and cannot be disabled.

3 Other LCS options

3.1 List LCS contents

To list the contents currently included in local content storage, use this command:

```
$HOME/ModusToolbox/tools_3.1/lcs-manager-cli/lcs-manager-cli --list
```

This will display all versions of the BSPs and libraries in local content storage in case (all caps first) and alphabetical order.

3.2 List BSPs

To list only the the included in local content storage in the *watchlist.json* file, use this command:

```
$HOME/ModusToolbox/tools_3.1/lcs-manager-cli/lcs-manager-cli --list-bsps
```

This will display only the BSPs names without versions.

3.3 Remove all local content

To remove all content from local content storage, use this command:

```
$HOME/ModusToolbox/tools_3.1/lcs-manager-cli/lcs-manager-cli --clear-all
```

This will delete all files from the *lcs/git* directory, and remove the list of BSPs from the *watchlist.json* file.

Note: *If you want to recreate local content storage after using this command, you should first delete the entire lcs directory.*

3.4 Remove BSP

To remove only a specified BSP and its dependencies from local content storage, use this command:

```
$HOME/ModusToolbox/tools_3.1/lcs-manager-cli/lcs-manager-cli --clear-bsp <BSPName>
```

Where `<BSPName>` is a valid name of a BSP included in local content storage.

3.5 Check for updates

To check for updates to existing content in local storage, you must be connected to the internet. Then, use this command:

```
$ lcs-manager-cli --check-for-updates
```

3.6 Update existing content

If the check for updates option indicates there are updates to your local content, run this command while connected to the internet:

```
$HOME/ModusToolbox/tools_3.1/lcs-manager-cli/lcs-manager-cli --update-existing
```

Note: *The --update-existing command only brings down new versions of existing assets. It does NOT bring in new assets that have not already been downlaoded unless those assets are related to a BSP in the watchlist (see [--add-bsp](#) for information about setting up a watchlist). If you want to get all the latest content, including new assets and new versions of existing assets, then use the `--add-all` command.*

3.7　　Update repo

If you only want to update a repo and its dependencies rather than everything, run this command while connected to the internet:

```
$HOME/ModusToolbox/tools_3.1/lcs-manager-cli/lcs-manager-cli --update <name>
```

Where `<name>` is a valid name a repo included in local content storage.

Revision history

Revision	Date	Description of change
**	2020-05-23	New document.

Trademarks
All referenced product or service names and trademarks are the property of their respective owners.