



---

# ScienceLogic Libraries and Execution Environments

SL1 version 11.1.0, rev1

---

# Table of Contents

<b>Introduction</b> .....	<b>3</b>
What is a ScienceLogic Library? .....	3
What is an Execution Environment? .....	4
<b>Managing ScienceLogic Libraries</b> .....	<b>6</b>
Viewing the List of ScienceLogic Libraries .....	6
Searching and Filtering the List of ScienceLogic Libraries .....	7
Special Characters .....	8
Downloading a ScienceLogic Library .....	11
Importing a ScienceLogic Library .....	12
Creating a ScienceLogic Library .....	12
Important Notes on Creating ScienceLogic Libraries .....	12
Library File Type: py_package .....	13
Library File Type: py_directory .....	14
Deleting a ScienceLogic Library .....	14
<b>Managing Execution Environments</b> .....	<b>15</b>
Viewing the List of Execution Environments .....	15
Searching and Filtering the List of Execution Environments .....	16
Special Characters .....	17
Creating an Execution Environment .....	20
Editing an Execution Environment .....	22
Copying an Execution Environment .....	23
Deleting an Execution Environment .....	23

---

# Chapter

# 1

## Introduction

---

### Overview

This manual describes how to create and manage ScienceLogic Libraries and Execution Environments.

The following sections provide an overview of ScienceLogic Libraries and Execution Environments:

<a href="#">What is a ScienceLogic Library?</a> .....	3
<a href="#">What is an Execution Environment?</a> .....	4

---

### What is a ScienceLogic Library?

Some Dynamic Applications, credential tests, and Run Book Actions in SL1 collect data or complete automated actions by executing one or more blocks of Python code, called **snippets**. When this happens, SL1 passes credentials and other configuration information to the snippet, and at the end of execution, the snippet either passes collected data back to SL1 or runs an action.

A **ScienceLogic Library** is a package consisting of metadata and Python files that can be used by the Dynamic Applications, credential tests, and Run Book Actions that use snippets. ScienceLogic Libraries can be included in a PowerPack along with the Dynamic Applications, credential tests, and Run Book Actions that they support.

These are a few reasons you might want to create and use a ScienceLogic library in your PowerPack:

- You have a few short Python functions or classes you want to share between snippets (py\_directory).
- You have a Python package you created that you want to share between snippets (py\_package).
- You have a third-party package that needs to be available during snippet execution (py\_package).

ScienceLogic Libraries have two formats. The type that you use depends on your Python expertise and your requirements:

- *py\_directory (.tar format)*. This file type includes metadata and a directory of Python source files. This format is most useful for users who are not experienced Python developers or who do not want the overhead of Python packages.
- *py\_package (.tar format)*. This file type includes metadata and Python packages. This format requires an understanding of Python packaging tools. While this format is more complex, it also allows you to more easily take advantage of unit tests, automatic versioning, and other common packaging practices.

ScienceLogic Libraries can be installed on SL1 in three ways:

- Some libraries are included when you install SL1 or run system updates.
- Libraries that are included in a PowerPack are installed on SL1 when you install the PowerPack.
- Libraries can be manually imported into SL1.

You can view all of the installed ScienceLogic Libraries on the **ScienceLogic Library Manager** page (System > Customize > ScienceLogic Libraries).

For details about how to manage libraries in SL1, see the [Managing ScienceLogic Libraries](#) chapter.

---

## What is an Execution Environment?

To use a ScienceLogic Library, you must align it with an execution environment.

In the SL1 user interface, an **execution environment** is a list of one or more ScienceLogic libraries. When a Dynamic Application snippet, Run Book Automation snippet, or credential test is executed, the execution environment defines a virtual Python environment that is deployed on-demand and includes all of the ScienceLogic libraries aligned with it for use during snippet execution.

A ScienceLogic Library can be associated with multiple execution environments, allowing you to share code between snippets, Dynamic Applications, and PowerPacks. A ScienceLogic Library can be aligned with only a single execution environment at a time.

Execution environments must be specified when creating or editing the following:

- Snippet or internal collection Dynamic Applications
- Credential tests
- Run Book Action policies that execute a snippet

When SL1 runs a snippet embedded in a Dynamic Application, credential test, or a Run Book Action, it automatically deploys the runtime environment defined by the execution environment. The runtime environment includes each library aligned with the execution environment, plus system libraries that are provided by SL1.

If you do not specify the execution environment for a Dynamic Application, credential test, or Run Book Action, SL1 aligns them with a system-default execution environment. This default execution environment is defined in each SL1 System update.

When you build a PowerPack, it automatically includes the execution environments that are aligned with the PowerPack's Dynamic Applications, credential tests, and Run Book Actions that execute snippets.

For details about how to create and manage execution environments in SL1, see the [Managing Execution Environments](#) chapter.

For more information about aligning execution environments with Dynamic Applications, credential tests, or Run Book Actions, see the following manuals:

- *Dynamic Application Development*
- *Snippet Dynamic Application Development*
- *Internal Collection Dynamic Application Development*
- *Discovery and Credentials*
- *Run Book Automation*

## Managing ScienceLogic Libraries

---

### Overview

The following sections describe how to manage ScienceLogic Libraries in SL1:

<i>Viewing the List of ScienceLogic Libraries</i> .....	6
<i>Searching and Filtering the List of ScienceLogic Libraries</i> .....	7
Special Characters .....	8
<i>Downloading a ScienceLogic Library</i> .....	11
<i>Importing a ScienceLogic Library</i> .....	12
<i>Creating a ScienceLogic Library</i> .....	12
<i>Important Notes on Creating ScienceLogic Libraries</i> .....	12
<i>Library File Type: py_package</i> .....	13
<i>Library File Type: py_directory</i> .....	14
<i>Deleting a ScienceLogic Library</i> .....	14

---

### Viewing the List of ScienceLogic Libraries

The **ScienceLogic Library Manager** page (System > Customize > ScienceLogic Libraries) lets you view a list of all of the ScienceLogic Libraries available for use and download those libraries for your own use. From this page, you can also import new libraries, delete existing libraries, or view and edit a list of Execution Environments that you can then associate with a library.

## Searching and Filtering the List of ScienceLogic Libraries

The Filter-While-You-Type fields appear as a row of blank fields at the top of the list. These fields let you filter the items that appear in the list.

The list is dynamically updated as you select each filter. For each filter, you must make a selection from a drop-down menu or type text to match against. SL1 will search for entries that match the text, including partial matches. Text matches are not case-sensitive, and you can use special characters in each text field.

By default, the cursor is placed in the first Filter-While-You-Type field. You can use the <Tab> key or your mouse to move your cursor through the fields.

You can filter by one or more of the following parameters. Only items that meet all of the filter criteria are displayed on the page.

- **Library Name.** You can enter text to match, including *special characters*, and the **ScienceLogic Library Manager** page will display only ScienceLogic Libraries that have a matching name.
- **Version.** You can enter text to match, including *special characters* and the **ScienceLogic Library Manager** page will display only ScienceLogic Libraries that have a matching version number.
- **System.** You can enter text to match, including *special characters* and the **ScienceLogic Library Manager** page will display only ScienceLogic Libraries that have a matching System value.
- **Type.** You can enter text to match, including *special characters* and the **ScienceLogic Library Manager** page will display only ScienceLogic Libraries that have a matching Type value.
- **Dependencies.** You can enter text to match, including *special characters* and the **ScienceLogic Library Manager** page will display only ScienceLogic Libraries that have matching dependencies.
- **Description.** You can enter text to match, including *special characters* and the **ScienceLogic Library Manager** page will display only ScienceLogic Libraries that have matching descriptions.
- **Edited By.** You can enter text to match, including *special characters* and the **ScienceLogic Library Manager** page will display only ScienceLogic Libraries that were created or last edited by a matching username.
- **Last Edited.** You can select from a list of time periods, and the **ScienceLogic Library Manager** page will display only ScienceLogic Libraries that have been created or edited within that time period.

## Special Characters

You can include the following special characters to filter by each column except those that display date and time:

**NOTE:** When searching for a string, SL1 will match substrings by default, even if you do not include any special characters. For example, searching for "hel" will match both "hello" and "helicopter". When searching for a numeric value, SL1 will not match a substring unless you use a special character.

### String and Numeric

- , (comma). Specifies an "OR" operation. Works for string and numeric values. For example:  
"dell, micro" matches all values that contain the string "dell" OR the string "micro".
- & (ampersand). Specifies an "AND" operation. Works for string and numeric values. For example:  
"dell & micro" matches all values that contain both the string "dell" AND the string "micro", in any order.
- ! (exclamation point). Specifies a "not" operation. Works for string and numeric values. For example:

**NOTE:** You can also use the "!" character in combination with the arithmetical special characters (min-max, >, <, >=, <=, =) described below.

- \* (asterisk). Specifies a "match zero or more" operation. Works for string and numeric values. For a string, matches any string that matches the text before and after the asterisk. For a number, matches any number that contains the text. For example:  
"hel\*er" would match "helpers" and "helicopter" but not "hello".  
"325\*" would match "325", "32561", and "325000".  
"\*000" would match "1000", "25000", and "10500000".
- ? (question mark). Specifies "match any one character". Works for string and numeric values. For example:  
"!?ver" would match the strings "oliver", "levers", and "lover", but not "believer".  
"135?" would match the numbers "1350", "1354", and "1359", but not "135" or "13502"



## String

- `^` (caret). For strings only. Specifies "match the beginning". Matches any string that begins with the specified string. For example:

`^sci` would match "scientific" and "sciencelogic", but not "conscious".

`^happy$` would match only the string "happy", with no characters before or after.

`!^micro` would match all values that do not start with "micro".

`!^$` would match all values that are not null.

`!^` would match null values.

- `$` (dollar sign). For strings only. Specifies "match the ending". Matches any string that ends with the specified string. For example:

`ter$` would match the string "renter" but not the string "terrific".

`^happy$` would match only the string "happy", with no characters before or after.

`!fer$` would match all values that do not end with "fer".

`!^$` would match all values that are not null.

`!$` would match null values.

**NOTE:** You can use both `^` and `$` if you want to match an entire string and only that string. For example, `^tern$` would match the strings "tern" or "Tern" or "TERN"; it would not match the strings "terne" or "cistern".

## Numeric

- `min-max`. Matches numeric values only. Specifies any value between the minimum value and the maximum value, including the minimum and the maximum. For example:

`1-5` would match 1, 2, 3, 4, and 5.

- `-` (dash). Matches numeric values only. A "half open" range. Specifies values including the minimum and greater or including the maximum and lesser. For example:

`1-` matches 1 and greater. So would match 1, 2, 6, 345, etc.

`-5` matches 5 and less. So would match 5, 3, 1, 0, etc.

- `>` (greater than). Matches numeric values only. Specifies any value "greater than". For example:

`>7` would match all values greater than 7.

- < (less than). Matches numeric values only. Specifies any value "less than". For example:  
 "<12" would match all values less than 12.
- >= (greater than or equal to). Matches numeric values only. Specifies any value "greater than or equal to". For example:  
 "=>7" would match all values 7 and greater.
- <= (less than or equal to). Matches numeric values only. Specifies any value "less than or equal to". For example:  
 "=<12" would match all values 12 and less.
- = (equal). Matches numeric values only. For numeric values, allows you to match a negative value. For example:  
 "=-5" would match "-5" instead of being evaluated as the "half open range" as described above.

### Examples

- "!dell" matches all values that do not contain the string "dell".
- "! ^ micro" would match all values that do not start with "micro".
- "!fer\$" would match all values that do not end with "fer".
- "! ^ \$" would match all values that are not null.
- "! ^ " would match null values.
- "! \$" would match null values.
- "!\*" would match null values.
- "happy, !dell" would match values that contain "happy" OR values that do not contain "dell".
- "aio\$". Matches only text that ends with "aio".
- "^ shu". Matches only text that begins with "shu".
- "^ silo\$". Matches only the text "silo", with no characters before or after.
- "!silo". Matches only text that does not contain the characters "silo".
- "! ^ silo". Matches only text that does not start with "silo".
- "!O\$". Matches only text that does not end with "O".
- "! ^ silo\$". Matches only text that is not the exact text "silo", with no characters before or after.
- "! ^ ". Matches null values, typically represented as "--" in most pages.
- "! \$". Matches null values, typically represented as "--" in most pages.
- "! ^ \$". Matches all text that is not null.
- silo, !aggr". Matches text that contains the characters "silo" and also text that does not contain "aggr".
- "silo, O2, !aggr". Matches text that contains "silo" and also text that contains "O2" and also text that does not contain "aggr".

- "silo, 02, !aggr, !01". Matches text that contains "silo" and also text that contains "02" and also text that does not contain "aggr" and also text that does not contain "01".
- "^s\*i!\*o\$". Matches text that contains the letter "s", "i", "l", "o", in that order. Other letters might lie between these letters. For example "sXiXIXo" would match.
- "!^s\*i!\*o\$". Matches all text that does not that contains the letter "s", "i", "l", "o", in that order. Other letters might lie between these letters. For example "sXiXIXo" would not match.
- "!vol&!silo". Matches text that does not contain "vol" AND also does not contain "silo". For example, "volume" would match, because it contains "vol" but not "silo".
- "!vol&02". Matches text that does not contain "vol" AND also contains "02". For example, "happy02" would match, because it does not contain "vol" and it does contain "02".
- "aggr,!vol&02". Matches text that contains "aggr" OR text that does not contain "vol" AND also contains "02".
- "aggr,!vol&!infra". Matches text that contains "aggr" OR text that does not contain "vol" AND does not contain "infra".
- "\*". Matches all text.
- "!\*". Matches null values, typically represented as "--" in most pages.
- "silo". Matches text that contains "silo".
- "!silo". Matches text that does not contain "silo".
- "!^silo\$". Matches all text except the text "silo", with no characters before or after.
- "-3,7-8,11,24,50-". Matches numbers 1, 2, 3, 7, 8, 11, 24, 50, and all numbers greater than 50.
- "-3,7-8,11,24,50-,a". Matches numbers 1, 2, 3, 7, 8, 11, 24, 50, and all numbers greater than 50, and text that includes "a".
- "?n". Matches text that contains any single character and the character "n". For example, this string would match "an", "bn", "cn", "1n", and "2n".
- "n\*SAN". Matches text the contains "n", zero or any number of any characters and then "SAN". For example, the string would match "nSAN", and "nhamburgerSAN".
- "^?n\*SAN\$". Matches text that begins with any single character, is following by "n", and then zero or any number of any characters, and ends in "SAN".

---


## Downloading a ScienceLogic Library

From the **ScienceLogic Library Manager** page, you can download ScienceLogic Libraries for your own use.

**WARNING:** ScienceLogic-authored libraries are copyrighted content and should not be modified without the express permission of ScienceLogic.

Libraries that are included in SL1 are bundled in either a py\_package or a py\_directory format.

To download a ScienceLogic Library:

1. Go to the **ScienceLogic Library Manager** page (System > Customize > ScienceLogic Libraries).
2. Click the save icon () of the ScienceLogic Library you want to download.
3. The library file is downloaded to your computer.

---

## Importing a ScienceLogic Library

In addition to the ScienceLogic Libraries that are included in the latest version of SL1 or included in a PowerPack, you can use the **ScienceLogic Library Manager** page to import additional libraries for your own use.

To import a library:

1. Go to the **ScienceLogic Library Manager** page (System > Customize > ScienceLogic Libraries).
2. Click the **[Actions]** menu and then select *Import ScienceLogic Library*.
3. The **Import ScienceLogic Library** dialog box appears. Use the **[Browse]** button to navigate to the ScienceLogic Library file you want to import and then click the **[Import]** button.
4. The imported ScienceLogic Library appears on the **ScienceLogic Library Manager** page.

---

## Creating a ScienceLogic Library

You can create your own library to import into SL1 in either a `py_directory` or `py_package` library file saved in uncompressed `.tar` format. This section describes the requirements for a ScienceLogic compatible library.

## Important Notes on Creating ScienceLogic Libraries

The following are important items to note when creating a ScienceLogic library:

- Library files must be saved in uncompressed `.tar` format.
- For best results, use GNU tar rather than bsdtar.
- Libraries must be Python 2 or Python 2/3 compatible. Currently, SL1 does not support Python 3 libraries.
- For a `py_directory` library, the library name comes from the name of the directory that you saved.
- You can only include wheel files (`.whl`) as additional dependencies. You cannot include any additional packages in your library file that require compilation or linking because Python build tools are not included in the SL1 platform.

- Any libraries that are bundled as dependencies in your `py_package` file will not be used if a newer version of that library exists on the appliance *unless* you have a restriction in your package specifying a given version to use.

For example, if your library "my-library" requires the Python community package "requests" version 2.25.1, you would create the library "requests" version "2.25.1" first. You would then import the requests library into SL1 and align it and "my-library" to your Execution Environment. At runtime, both libraries, at those specific versions, will be used.

For an alternative example, if your library "my-library" requires the Python community package "requests", but the version is not important, you would include the "requests" package as a "whl" file in your "my-library" library, adding it to the "manifest.txt" file. After aligning "my-library" to your Execution Environment, it will be available at runtime. The "requests" package included in the "manifest.txt" will be available during the install of "my-library", but may not be used if a newer acceptable package is found for install.

**TIP:** A best practice is to version your libraries. Each time you make modifications to a library, create a new version and a new execution environment that uses that version. In that way, you can use different versions in different Dynamic Applications for testing purposes, for example.

## Library File Type: `py_package`

The `py_package` file type is the preferred library file type used to bundle Python packages and metadata files. A `py_package` must have the metadata files described below along with the wheel file(s) and must be saved in uncompressed `.tar` format. You must create this tar file with Red Hat Enterprise Linux or a similar operating system to ensure compatibility. Tar files created on MacOS are not supported.

**NOTE:** All metadata files included in the package must be at the root level of the directory.

A `py_package` tar file must include the following files:

- `__init__.py`. This file is required for the directory to be used as a Python package. It identifies this as a package.
- `name.txt`. This text file specifies the name of the library and must match the name of the package being installed.
- `version.txt`. This text file specifies the version of the library and must match the version of the package being installed.
- `type.txt`. This text file specifies that the library is a `py_package` type library.
- `manifest.txt`. This text file includes the Python package name and version (as specified in the `name.txt` and `version.txt` files), followed by a list of additional supporting files that must be deployed to the internal package repository to use the library. For example, if the library is dependent on another package, that other package must be listed in the manifest file.

**NOTE:** The package name of the library must be the first entry in the manifest file.

- Wheel (.whl) file. This is a standard wheel file that you create. See Python documentation for more information about creating wheel files. Wheel files must be Python 2 or Python 2/3 compatible and must be created in Red Hat Enterprise Linux or a similar operating system. You may include multiple wheel files in the `py_package`.

**CAUTION:** Libraries that are Python 3 only are not currently supported.

## Library File Type: `py_directory`

The `py_directory` file type is used to share short code snippets, for example, between Dynamic Applications or SL1 stacks. Files must be organized in a directory. You must then save the directory in uncompressed `.tar` format in Red Hat Enterprise Linux or a similar operating system. Tar files created in MacOS are not supported.

**NOTE:** The name of the library comes from the name of the directory.

A `py_directory` tar file can include the following files:

- `__init__.py`. This file is required for the directory to be imported as a Python package.
- `version.txt`. This text file contains a version string specifying the version of the library.
- `dependencies.txt`. This text file includes a comma-separated list of other libraries on which the library is dependent.
- `description.txt`. This text file includes a single-line description of the library's intended use.
- Python bytecode (.pyc) files or other source files. These are the additional relevant files that you might want to include in the library.

---

## Deleting a ScienceLogic Library

The **ScienceLogic Library Manager** page allows you to delete ScienceLogic Libraries that are not currently in use.

SL1 will not allow you to delete ScienceLogic Libraries that are aligned to execution environments that are being used by any Dynamic Applications, credential tests, or Run Book Actions. You also cannot delete libraries that are aligned to the "System" environment.

To delete a library:

1. Go to the **ScienceLogic Library Manager** page (System > Customize > ScienceLogic Libraries).
2. Select the ScienceLogic Library that you want to delete.
3. Click the **Select Actions** menu in the lower right of the page and select *DELETE ScienceLogic Library*, and then click the **[Go]** button.
4. The selected ScienceLogic Library is deleted from the **ScienceLogic Library Manager** page.

## Managing Execution Environments

---

### Overview

Execution environments let you share code between snippets, Dynamic Applications, and PowerPacks. Execution environments have a global unique identifier (GUID), a name, and list of associated libraries. The libraries associated with an execution environment provide a way to bundle reusable code in a self-contained environment available in snippets. You can use different execution environments for different Dynamic Application requirements.

The following sections describe how to create and manage execution environments in SL1:

<i>Viewing the List of Execution Environments</i> .....	15
<i>Searching and Filtering the List of Execution Environments</i> .....	16
Special Characters .....	17
<i>Creating an Execution Environment</i> .....	20
<i>Editing an Execution Environment</i> .....	22
<i>Copying an Execution Environment</i> .....	23
<i>Deleting an Execution Environment</i> .....	23

---


### Viewing the List of Execution Environments

The **Environment Manager** page (System > Customize > ScienceLogic Libraries > Actions > Execution Environments) displays a list of all existing execution environments.

Environment Manager   Execution Environments Found [30]			New	Reset
Environment Name	Env GUID	Libs		
1. Amazon Web Services	3CD53F11C68808FAB0A59D2E44440F46	6		
2. Cisco: ACI	1C88582E76AADD40EB8C5E6A6F71B64A	3		
3. Cisco: Base Pack	824F19CAC9F7E844121354B6DCF75CF8	1		
4. Cisco: CSP-2100	05E6DD26A71848A26807873D1B72A386	3		
5. Cisco: CUCM Cisco Unified Communications Manager	38EC0135EE530542AED1A9B0483DE094	2		
6. Cisco: IPSLA	8785FFA3A90C659A4DB8C9454685BF57	1		
7. Cisco: UC Ancillary	A5EF0DBB2827E292F6FECB2DF7EF6DC7	1		
8. Cisco: UC VOS Applications	09D4BDBD2E305F1485851D4EA748B6B1	3		
9. Cisco: UCS	B099415941BF458ADB3208F2B0B36249	1		
10. Cisco: Wireless	A216F0F1429CC78049765432B94C4336	1		
11. Citrix: Xen	AF446B05268871CCD12E30C2E6292CE6	1		
12. Dell EMC: Isilon	5BF14AA62DDA747738FC83942E5037EF	6		
13. Dell EMC: XtremIO	71F3772E46DF1B3540C2F627AA5EF84E	5		
14. EM7 Credential Tests	4B28CE11009D21D93889A87C23BAEB67	1		
15. EMC: VMAX	DCC08CE7F1E917705E642F1604857207	5		

For each execution environment, the page displays the following information:

**TIP:** To sort the list of execution environments, click on a column heading. The list will be sorted by the column value, in ascending order. To sort by descending order, click the column heading again.

- **Environment Name.** The name of the execution environment.
- **Env GUID.** The execution environment's globally unique identifier.
- **Libs.** The number of ScienceLogic Libraries aligned with the execution environment. Click the View Aligned Libraries icon () to view the list of ScienceLogic Libraries that are aligned to a particular execution environment.

## Searching and Filtering the List of Execution Environments

The Filter-While-You-Type fields appear as a row of blank fields at the top of the list. These fields let you filter the items that appear in the list.

The list is dynamically updated as you select each filter. For each filter, you must make a selection from a drop-down menu or type text to match against. SL1 will search for entries that match the text, including partial matches. Text matches are not case-sensitive, and you can use special characters in each text field.

By default, the cursor is placed in the first Filter-While-You-Type field. You can use the <Tab> key or your mouse to move your cursor through the fields.

You can filter by one or more of the following parameters. Only items that meet all of the filter criteria are displayed on the page.



- **Environment Name.** You can enter text to match, including *special characters*, and the **Environment Manager** page will display only execution environments that have a matching name.
- **Env GUID.** You can enter text to match, including *special characters*, and the **Environment Manager** page will display only execution environments that have a matching GUID.
- **Libs.** You can enter text to match, including *special characters*, and the **Environment Manager** page will display only execution environments that have a matching number of aligned ScienceLogic Libraries.

## Special Characters

You can include the following special characters to filter by each column except those that display date and time:

**NOTE:** When searching for a string, SL1 will match substrings by default, even if you do not include any special characters. For example, searching for "hel" will match both "hello" and "helicopter". When searching for a numeric value, SL1 will not match a substring unless you use a special character.

### String and Numeric

- , (comma). Specifies an "OR" operation. Works for string and numeric values. For example:  
"dell, micro" matches all values that contain the string "dell" OR the string "micro".
- & (ampersand). Specifies an "AND" operation. Works for string and numeric values. For example:  
"dell & micro" matches all values that contain both the string "dell" AND the string "micro", in any order.
- ! (exclamation point). Specifies a "not" operation. Works for string and numeric values. For example:

**NOTE:** You can also use the "!" character in combination with the arithmetical special characters (min-max, >, <, >=, <=, =) described below.

- \* (asterisk). Specifies a "match zero or more" operation. Works for string and numeric values. For a string, matches any string that matches the text before and after the asterisk. For a number, matches any number that contains the text. For example:  
"hel\*er" would match "helpers" and "helicopter" but not "hello".  
"325\*" would match "325", "32561", and "325000".  
"\*000" would match "1000", "25000", and "10500000".
- ? (question mark). Specifies "match any one character". Works for string and numeric values. For example:  
"!?ver" would match the strings "oliver", "levers", and "lover", but not "believer".  
"135?" would match the numbers "1350", "1354", and "1359", but not "135" or "13502"

## String

- `^` (caret). For strings only. Specifies "match the beginning". Matches any string that begins with the specified string. For example:

`^sci` would match "scientific" and "sciencelogic", but not "conscious".

`^happy$` would match only the string "happy", with no characters before or after.

`!^micro` would match all values that do not start with "micro".

`!^$` would match all values that are not null.

`!^` would match null values.

- `$` (dollar sign). For strings only. Specifies "match the ending". Matches any string that ends with the specified string. For example:

`ter$` would match the string "renter" but not the string "terrific".

`^happy$` would match only the string "happy", with no characters before or after.

`!fer$` would match all values that do not end with "fer".

`!^$` would match all values that are not null.

`!$` would match null values.

**NOTE:** You can use both `^` and `$` if you want to match an entire string and only that string. For example, `^tern$` would match the strings "tern" or "Tern" or "TERN"; it would not match the strings "terne" or "cistern".

## Numeric

- `min-max`. Matches numeric values only. Specifies any value between the minimum value and the maximum value, including the minimum and the maximum. For example:

`1-5` would match 1, 2, 3, 4, and 5.

- `-` (dash). Matches numeric values only. A "half open" range. Specifies values including the minimum and greater or including the maximum and lesser. For example:

`1-` matches 1 and greater. So would match 1, 2, 6, 345, etc.

`-5` matches 5 and less. So would match 5, 3, 1, 0, etc.

- `>` (greater than). Matches numeric values only. Specifies any value "greater than". For example:

`>7` would match all values greater than 7.

- < (less than). Matches numeric values only. Specifies any value "less than". For example:  
 "<12" would match all values less than 12.
- >= (greater than or equal to). Matches numeric values only. Specifies any value "greater than or equal to". For example:  
 "=>7" would match all values 7 and greater.
- <= (less than or equal to). Matches numeric values only. Specifies any value "less than or equal to". For example:  
 "=<12" would match all values 12 and less.
- = (equal). Matches numeric values only. For numeric values, allows you to match a negative value. For example:  
 "=-5" would match "-5" instead of being evaluated as the "half open range" as described above.

### Examples

- "!dell" matches all values that do not contain the string "dell".
- "! ^ micro" would match all values that do not start with "micro".
- "!fer\$" would match all values that do not end with "fer".
- "! ^ \$" would match all values that are not null.
- "! ^ " would match null values.
- "! \$" would match null values.
- "!\*" would match null values.
- "happy, !dell" would match values that contain "happy" OR values that do not contain "dell".
- "aio\$". Matches only text that ends with "aio".
- "^ shu". Matches only text that begins with "shu".
- "^ silo\$". Matches only the text "silo", with no characters before or after.
- "!silo". Matches only text that does not contain the characters "silo".
- "! ^ silo". Matches only text that does not start with "silo".
- "!O\$". Matches only text that does not end with "O".
- "! ^ silo\$". Matches only text that is not the exact text "silo", with no characters before or after.
- "! ^ ". Matches null values, typically represented as "--" in most pages.
- "! \$". Matches null values, typically represented as "--" in most pages.
- "! ^ \$". Matches all text that is not null.
- silo, !aggr". Matches text that contains the characters "silo" and also text that does not contain "aggr".
- "silo, O2, !aggr". Matches text that contains "silo" and also text that contains "O2" and also text that does not contain "aggr".

- "silo, 02, !aggr, !01". Matches text that contains "silo" and also text that contains "02" and also text that does not contain "aggr" and also text that does not contain "01".
- "^s\*i!\*o\$". Matches text that contains the letter "s", "i", "l", "o", in that order. Other letters might lie between these letters. For example "sXiXo" would match.
- "!^s\*i!\*o\$". Matches all text that does not that contains the letter "s", "i", "l", "o", in that order. Other letters might lie between these letters. For example "sXiXo" would not match.
- "!vol&!silo". Matches text that does not contain "vol" AND also does not contain "silo". For example, "volume" would match, because it contains "vol" but not "silo".
- "!vol&02". Matches text that does not contain "vol" AND also contains "02". For example, "happy02" would match, because it does not contain "vol" and it does contain "02".
- "aggr,!vol&02". Matches text that contains "aggr" OR text that does not contain "vol" AND also contains "02".
- "aggr,!vol&!infra". Matches text that contains "aggr" OR text that does not contain "vol" AND does not contain "infra".
- "\*". Matches all text.
- "!\*". Matches null values, typically represented as "--" in most pages.
- "silo". Matches text that contains "silo".
- "!silo". Matches text that does not contain "silo".
- "!^silo\$". Matches all text except the text "silo", with no characters before or after.
- "-3,7-8,11,24,50-". Matches numbers 1, 2, 3, 7, 8, 11, 24, 50, and all numbers greater than 50.
- "-3,7-8,11,24,50-,a". Matches numbers 1, 2, 3, 7, 8, 11, 24, 50, and all numbers greater than 50, and text that includes "a".
- "?n". Matches text that contains any single character and the character "n". For example, this string would match "an", "bn", "cn", "1n", and "2n".
- "n\*SAN". Matches text the contains "n", zero or any number of any characters and then "SAN". For example, the string would match "nSAN", and "nhamburgerSAN".
- "^?n\*SAN\$". Matches text that begins with any single character, is following by "n", and then zero or any number of any characters, and ends in "SAN".

---

## Creating an Execution Environment

SL1 includes several execution environments from preloaded PowerPacks and system updates for your use. If needed, you can also create new execution environments. After you have successfully created an execution environment, it will appear in the list of execution environments available for aligning with Dynamic Applications, credential tests, and Run Book Actions.

To create an execution environment:

- From the **Environment Manager** page (System > Customize > ScienceLogic Libraries > Actions > Execution Environments), click the **[New]** button.

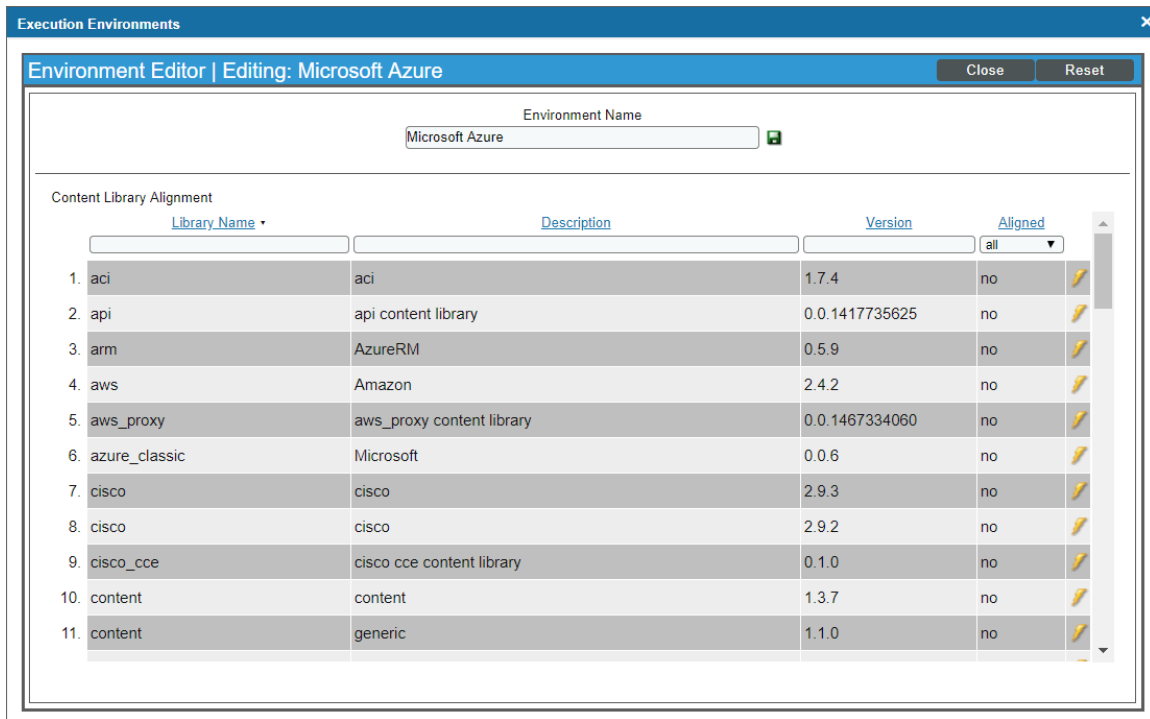
The screenshot shows the 'Environment Manager' interface with the title 'Execution Environments Found [30]'. A red box highlights the 'New' button in the top right corner. Below the header, there is a table with columns for 'Environment Name', 'Env GUID', and 'Libs'. The table lists 15 environments, each with a name, a GUID, and a 'Libs' count. Each row also includes a small icon and a save icon.

	Environment Name	Env GUID	Libs
1.	Amazon Web Services	3CD53F11C68808FAB0A59D2E44440F46	6
2.	Cisco: ACI	1C88582E76AADD40EB8C5E6A6F71B64A	3
3.	Cisco: Base Pack	824F19CAC9F7E844121354B6DCF75CF8	1
4.	Cisco: CSP-2100	05E6DD26A71848A26807873D1B72A386	3
5.	Cisco: CUCM Cisco Unified Communications Manager	38EC0135EE530542AED1A9B0483DE094	2
6.	Cisco: IPSLA	8785FFA3A90C659A4DB8C9454685BF57	1
7.	Cisco: UC Ancillary	A5EF0DBB2827E292F6FECB2DF7EF6DC7	1
8.	Cisco: UC VOS Applications	09D4BDBD2E305F1485851D4EA748B6B1	3
9.	Cisco: UCS	B099415941BF458ADB3208F2B0B36249	1
10.	Cisco: Wireless	A216F0F1429CC78049765432B94C4336	1
11.	Citrix: Xen	AF446B05268871CCD12E30C2E6292CE6	1
12.	Dell EMC: Isilon	5BF14AA62DDA747738FC83942E5037EF	6
13.	Dell EMC: XtremIO	71F3772E46DF1B3540C2F627AA5EF84E	5
14.	EM7 Credential Tests	4B28CE11009D21D93889A87C23BAEB67	1
15.	EMC: VMAX	DCC08CE7F1E917705E642F1604857207	5

- On the **Environment Editor** page, type a name for the new execution environment in the **Environment Name** field, and then click the save icon (📁).

**NOTE:** Each execution environment is assigned an Env GUID, which makes it unique. Since two environments can share a name, it is a best practice to use the Name field to version an environment when updating its contents.

- The **Library Alignment** pane appears, displaying a list of ScienceLogic Libraries that you can align with the new execution environment. Click the align icon (🔧) for a ScienceLogic Library to align that library to the new execution environment.



- Align any additional ScienceLogic libraries to the execution environment as needed.

**TIP:** If necessary, you can click the unalign icon (🔧) for a ScienceLogic Library to remove that library's alignment with the new execution environment.

- When you are finished, click the save icon (💾) again to save the execution environment with the ScienceLogic Library alignment you selected, and then click **[Close]**.

## Editing an Execution Environment

You can edit an execution environment by clicking its wrench icon (🔧). When you do so, you can change the environment's name, align additional ScienceLogic Libraries to the environment, or remove its alignment with one or more libraries.

**WARNING:** Do not edit ScienceLogic-created execution environments. You may make a copy of the execution environment, but ScienceLogic often updates these execution environments in a release, which will overwrite any existing execution environment of the same globally unique identifier (GUID).

To edit an execution Environment:

1. From the **Environment Manager** page (System > Customize > ScienceLogic Libraries > Actions > Execution Environments), click the edit icon (✎) for the execution environment that you want to edit.
2. On the **Environment Editor** page, you can do any of the following:
  - Change the environment's name in the **Environment Name** field.
  - Align additional ScienceLogic Libraries to the execution environment by clicking the appropriate align icons (↗).
  - Remove the execution environment's alignment with one or more libraries by clicking the appropriate unalign icons (↖).
3. When you are finished, click the save icon (💾) and then click **[Close]**.

---

## Copying an Execution Environment

If you want to create a new execution environment that has traits similar to an existing environment, you can make a copy of the existing execution environment that you can then edit as needed. When you copy an existing execution environment, the new environment includes the same libraries as the original environment, but it will have a new globally unique identifier (GUID).

**CAUTION:** Making a copy of an execution environment does not mean there will be a duplicate runtime environment. Environments that share a set of libraries will result in the same virtual runtime environment to optimize space on the Data Collector.

To copy an execution environment:

1. From the **Environment Manager** page (System > Customize > ScienceLogic Libraries > Actions > Execution Environments), click the copy icon (📄) for the execution environment that you want to copy.
2. A confirmation message appears. Click **[OK]** to continue.
3. A copy of the selected execution environment appears on the **Environment Manager** page. By default, it has the same name as the original execution environment, followed by the word "(copy)".

---

## Deleting an Execution Environment

If you no longer need an execution environment and it is not currently aligned with a Dynamic Application, credential test, or Run Book Action, then you can delete it. SL1 will not allow you to delete execution environments that are being used by any Dynamic Applications, credential tests, or Run Book Actions.

**CAUTION:** Do not delete execution environments that are shipped with SL1.

To delete an execution environment:

1. From the **Environment Manager** page (System > Customize > ScienceLogic Libraries > Actions > Execution Environments), click the delete icon (🗑️) for the execution environment that you want to delete.
2. A confirmation message appears. Click **[OK]** to continue.
3. The execution environment is deleted from the **Environment Manager** page.



© 2003 - 2022, ScienceLogic, Inc.

All rights reserved.

#### LIMITATION OF LIABILITY AND GENERAL DISCLAIMER

ALL INFORMATION AVAILABLE IN THIS GUIDE IS PROVIDED "AS IS," WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED. SCIENCELOGIC™ AND ITS SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT.

Although ScienceLogic™ has attempted to provide accurate information on this Site, information on this Site may contain inadvertent technical inaccuracies or typographical errors, and ScienceLogic™ assumes no responsibility for the accuracy of the information. Information may be changed or updated without notice. ScienceLogic™ may also make improvements and / or changes in the products or services described in this Site at any time without notice.

#### Copyrights and Trademarks

ScienceLogic, the ScienceLogic logo, and EM7 are trademarks of ScienceLogic, Inc. in the United States, other countries, or both.

Below is a list of trademarks and service marks that should be credited to ScienceLogic, Inc. The ® and ™ symbols reflect the trademark registration status in the U.S. Patent and Trademark Office and may not be appropriate for materials to be distributed outside the United States.

- ScienceLogic™
- EM7™ and em7™
- Simplify IT™
- Dynamic Application™
- Relational Infrastructure Management™

The absence of a product or service name, slogan or logo from this list does not constitute a waiver of ScienceLogic's trademark or other intellectual property rights concerning that name, slogan, or logo.

Please note that laws concerning use of trademarks or product names vary by country. Always consult a local attorney for additional guidance.

#### Other

If any provision of this agreement shall be unlawful, void, or for any reason unenforceable, then that provision shall be deemed severable from this agreement and shall not affect the validity and enforceability of any remaining provisions. This is the entire agreement between the parties relating to the matters contained herein.

In the U.S. and other jurisdictions, trademark owners have a duty to police the use of their marks. Therefore, if you become aware of any improper use of ScienceLogic Trademarks, including infringement or counterfeiting by third parties, report them to Science Logic's legal department immediately. Report as much detail as possible about the misuse, including the name of the party, contact information, and copies or photographs of the potential misuse to: [legal@sciencelogic.com](mailto:legal@sciencelogic.com)

ScienceLogic

800-SCI-LOGIC (1-800-724-5644)

International: +1-703-354-1010