## Obtain the SAS-wsm Utility

If you have not already done so, download the latest version of SAS-wsm from the GitHub project page, and save SAS-wsm to the local machine where you will be controlling services. Then, complete the following steps to obtain the SAS-wsm utility:

1. Navigate to https://github.com/sassoftware/sas-wsm/releases.
2. Select the **Assets** drop-down list under the most recent release near the top of the page.
3. Download the Source code in either a .zip or .tar.gz format.

   Note: The .zip format is recommended for Microsoft Windows.
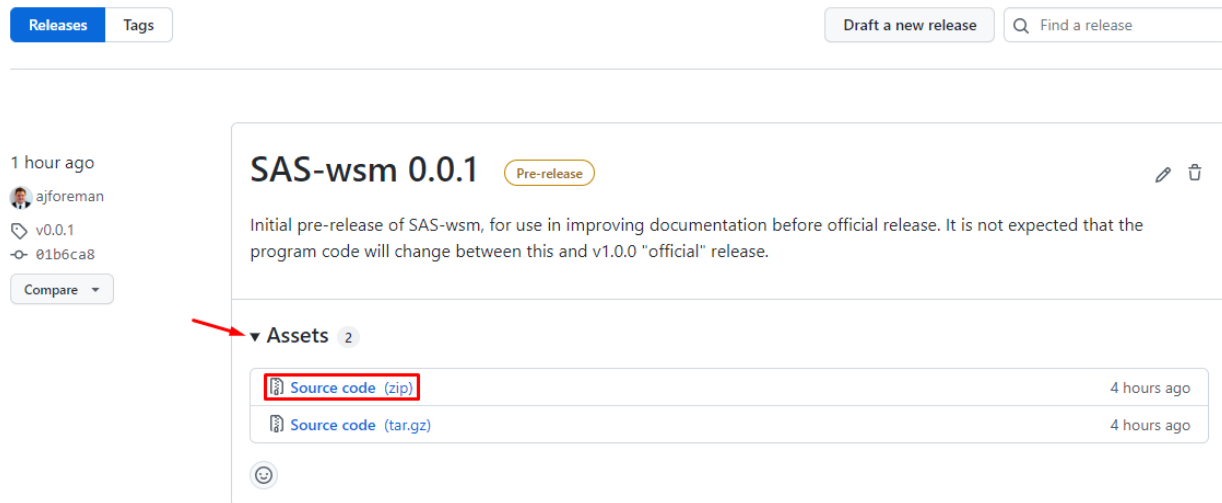
4. Move the downloaded archive to the location that you would like to run SAS-wsm from.
5. Extract the downloaded files using your preferred utility.

Basic Configuration

To run SAS-wsm for basic `start`, `stop`, and `status` commands, you can use the SAS-wsm.ps1 script without any further configuration.

Note: Refer to the "Running SAS-wsm" section for details about how to run the SAS-wsm script.

Information about PowerShell .ps1 Files

Some Windows administrators restrict the ability to run PowerShell script (.ps1) files.

You should verify that your account has the correct permissions to run a .ps1 file since this ability to run .ps1 files might be restricted on a per-file level or blocked via GPO. You should discuss with your system administrators and security teams if you cannot run .ps1 files.

Advanced Configuration

This section explains how to configure various optional components of SAS-wsm. However, note that the content of this section and the custom configuration parameters that it outlines are not necessary for basic usage in most environments.

Advanced configuration tasks are designed for administrators who want to define custom service sets, custom service configuration files, or otherwise want greater control over the actions performed during a SAS-wsm run.

Understanding Your Deployment

In order to correctly configure SAS-wsm custom parameters, you must understand your current SAS deployment. This section provides guidance about how to locate current SAS services and their correct start-up order.

Locate Current SAS Services

To view the SAS services that are deployed on your machine, open the Windows Services Manager by completing the following steps:

1. Open the start menu (or press Windows+R).
2. Enter **services.msc**.
3. Then, press the Enter key.
4. Scroll through the list of Windows Services until you find the section of services that begin with SAS, as shown in the following image:
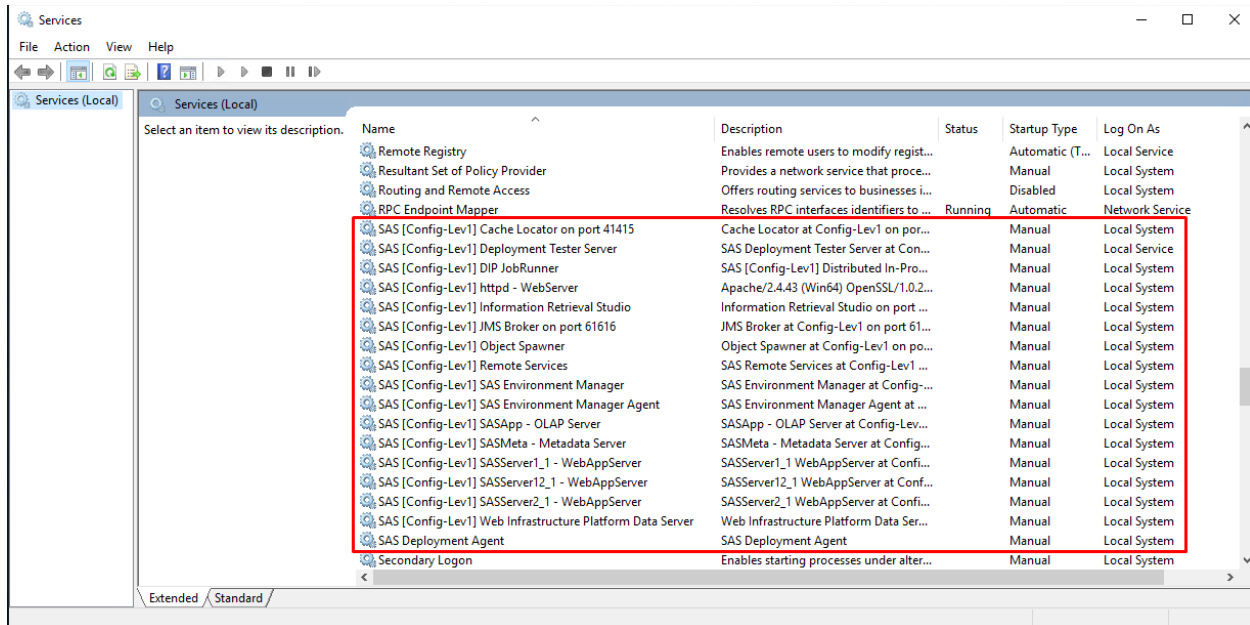
Determine the Correct Start-up Order of SAS Services

Once you locate the SAS services that are deployed on your machine, it is critical that you determine the correct order to start those services. Some SAS services have dependencies on other services being

available, which might result in start-up errors if those dependencies are not running when you start a later service.

If you have ever restarted SAS services on your machine without a full machine reboot, you have likely already determined the start-up order. You should use the documented start-up order for configuring SAS-wsm.

However, if you do not know the correct start-up order, see the following documentation for more information: [Overview of Sever Operation](#)

Default Start-up Order
The following are the typical critical SAS services, which are usually started in the following order:

1. SAS® Metadata Server
2. SAS® Web Infrastructure Platform Data Server
3. additional SAS® Data Servers, if configured
4. SAS® Object Spawner
5. SAS® Cache Locator
6. SAS® JMS Broker
7. SAS® Web Server (httpd)
8. SAS® Web Application Server SASServer1_1
9. additional SAS Web Application Servers, if configured (such as SASServer2_1, SASServer12_1, and so on)

Most other SAS services can be started at any time. The recommended order is to start any additional Metadata-tier resources after the SAS Metadata Server, Compute-tier services after the SAS Object Spawner, and mid-tier (and all other) services after the SAS Web Application Servers.

If you are unsure about when to start a SAS service, contact SAS Technical Support for assistance.

Locate the SAS Configuration Directory

When you deploy SAS software in a
Planned Deployment, a SAS configuration
directory is created to store configuration
information about various SAS components.
SAS-wsm must know where this directory
is located at in order to reference component
log files and verify start-up states.

If you are not sure of this location, complete the
following steps:

1. Open the properties of a SAS Web Application
   Server from **services.msc**.
2. In the **Path to executable** property,
   there should be a file path similar to the one in
   the image to the right, containing
   a folder named **Lev*N*** (where *N* is a number).
   This Lev*N* folder is the root of your SAS
   configuration directory.

Copy or make note of this path as you will
need to reference it in the next section.
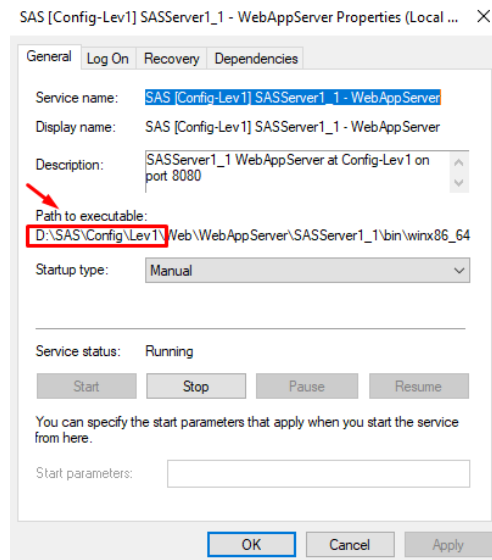
Create a Custom Service Configuration File

A custom service configuration file can be used to start or stop only a subset of SAS services. When you
run SAS-wsm with a custom service configuration file, only the services that are defined in the file have
their state modified.

This capability might be desirable to configure for administrators who perform partial restarts in
production environments. For example, you could make a custom service configuration file that starts or
stops only the Middle-Tier services, which allows users who are running SAS® Enterprise Guide or batch
jobs to continue running code without interruption.

The example-servers.cfg file that is delivered with SAS-wsm is an example of a custom service
configuration file that would start or stop only the Middle-Tier services.

A custom service configuration file can also be used to start or stop all SAS services in a non-default
order. However, SAS does not recommend that you follow this non-default order. You should use the
default service order for full-environment restarts.

Custom Service Configuration: Define Start-up Order

Refer to the "Determine Correct Start-up Order of SAS Services" section if you have not already done so.

Then, complete the following steps to define the desired custom service configuration file:

1. Using the correct determined SAS services start-up order, edit **example-servers.cfg** or create a
   new configuration file.
2. Use the **Display name** property of the services, as shown in **services.msc**.

Note: For accuracy, it is recommended that you copy the service names from **services.msc** by right-clicking the **name** of the service ► select **Properties** ► copy and paste the name from the **Display Name** property:
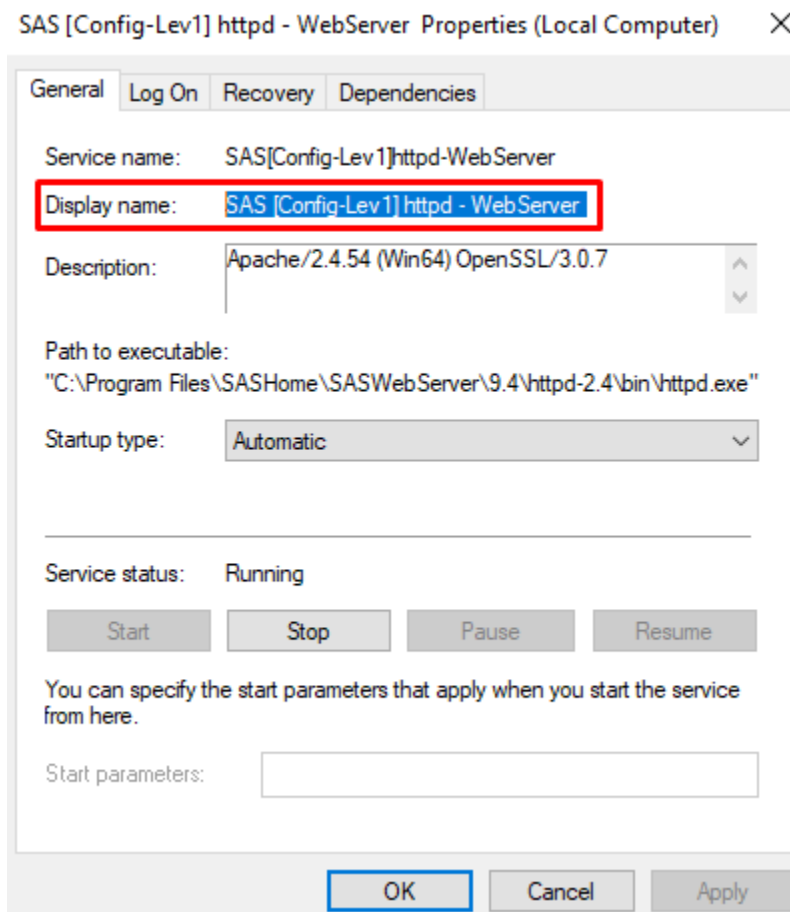
Be sure to use the **Display Name** field and not the **Service Name** field. These values might vary for some services, such as the httpd Web Server. SAS-wsm will not be able to provide Windows with the correct information to locate a service if the specified name does not match a service's Display Name.

You should comply with the following formatting when defining services:

- Place the first service to start on line 1, the second service to start on line 2, and so on.
- Provide the full display names exactly as they occur in **services.msc** properties (one per line).
- Do not add numeric formatting characters, comments, blank lines, or other superfluous text.

Custom Service Configuration: Stop Order

In most cases, the stop order for SAS services should be the opposite of the start order.

SAS-wsm reads a specified custom service configuration file backward (from the last line of the file to the first line of the file) when performing a stop operation. This process results in the stop order being the opposite of the start order that was defined.

Define SAS Configuration Directory Path

SAS-wsm must know where the SAS configuration directory is located at in order to reference component log files and verify start-up states. By default, SAS-wsm attempts to determine this location by reading the properties of the following common Windows Services:

- SAS Metadata Server
- SAS JMS Broker
- SAS Cache Locator

In rare instances, it might not be possible for these properties to be used for locating the SAS configuration directory, or these services might not be deployed on a machine. If SAS-wsm fails to locate the SAS configuration directory, SAS-wsm displays a `WARN` message to the user. This message does not terminate the operation of the script, but the message prevents SAS-wsm from validating that various services have fully started.

If an administrator wants to specify the SAS configuration directory path manually, or if there are multiple SAS configuration directory paths on the host but the administrator wants to use only one path, the paths can be explicitly defined when running SAS-wsm.

To skip the SAS configuration directory path location logic, define the path manually with the **-saslev** parameter. Provide the full path to the SAS configuration directory, including the Lev*N*.

See the "Running SAS-wsm" section for an example of using the explicit SAS configuration directory path parameter.

Testing Your SAS-wsm Configuration

You should test the utility to verify that it can successfully start, stop, and check the status of your SAS services.

If any errors occur while SAS-wsm is running, there might be a problem with your definitions.

If no errors occur but some services are not controlled as expected, verify that the Windows services appear in the service list output by SAS-wsm. If you are using a custom service configuration file, ensure that the correct Display Name was copied from services.msc.

Testing Considerations

You can perform a status check on a running system, which will not impact the state of SAS services.

A stop or start check causes SAS services to become unavailable while they are stopped. However, if there are current SAS sessions, users could lose their unsaved work. Please ensure that there are no active users on the system before testing a stop-and-start operation!

For command usage and examples, see the "Running SAS-wsm" section.

Running SAS-wsm

To run SAS-wsm, execute the **SAS-wsm.ps1** script that you previously modified from a PowerShell session. Then, provide the script with the parameter for the action that you want to perform (such as start, stop, or status).

Optionally, you might specify the path to a custom start-up configuration file and/or an explicit definition of your SAS configuration directory path. If you do not specify these items, the program logic uses the default recommended start-up order and attempts to automatically determine the SAS configuration directory path.

To open a PowerShell session, complete the following steps:

1. Open the Start menu.
2. Scroll through the list of programs for the Windows PowerShell folder.
3. Select the program under that location named Windows PowerShell.

The example commands listed below assume that you have previously navigated to the location of the SAS-wsm.ps1 file in your PowerShell session. If you have not navigated to that location, reference the full path to SAS-wsm.ps1 in your commands (or use the `cd` command to change the current directory of the PowerShell session to where you saved SAS-wsm.ps1).

Similarly, you must provide the full path to the configuration file if the file is not inside the current working directory, as shown in the Advanced: Stop Services example.

Example Commands

Start Services
```
.\SAS-wsm.ps1 -action start
```

Stop Services
```
.\SAS-wsm.ps1 -action stop
```

Check Current Status of Services
```
.\SAS-wsm.ps1 -action status
```

Advanced: Start Services with Explicit SAS Configuration Directory Path definition
```
.\SAS-wsm.ps1 -action start -saslev C:\SAS\SASConfig\Lev1\
```

Advanced: Stop Services using Custom Service Configuration File *example-servers.cfg*
```
.\SAS-wsm.ps1 -action stop -cfg "C:\My Folder\docs\example-servers.cfg"
```

Troubleshooting SAS-wsm

We apologize if you are encountering issues with SAS-wsm! Please know that while we do our best to test all scenarios before releasing updates, the differences in every individual SAS deployment might result in unexpected behavior. We will periodically update this section to document known issues, workarounds, and other potential fixes.

If you encounter problems using SAS-wsm that you are unable to resolve through your own troubleshooting and the information that is contained in this document, consider letting us know via a GitHub issue on the project or opening a track with SAS Technical Support.
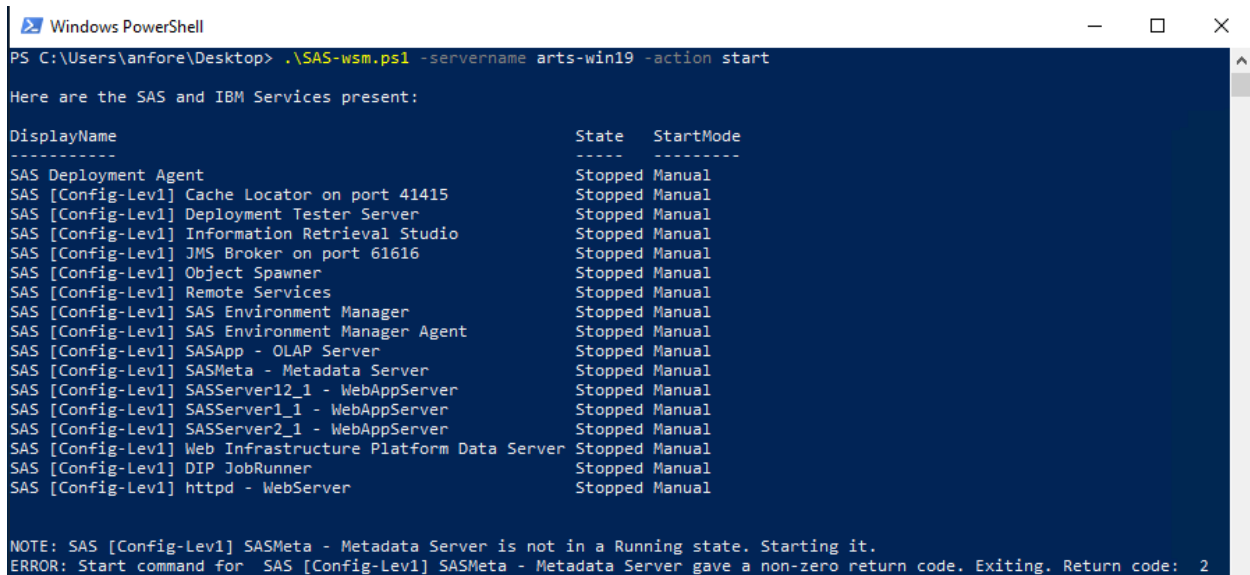
If you encounter a problem but are able to fix it, please consider contributing your fixes to the GitHub project or let us know so that we might resolve the issue in a future release.

When contacting SAS Technical Support about SAS-wsm, note that our regular support policies might not apply. Refer to the SUPPORT.md document included with SAS-wsm for more information.

Known Issues

SAS Metadata Server Start-up Permissions

When starting services with SAS-wsm, a Windows account that does not own the files under the SAS Configuration Directory might fail to start SAS Metadata Server. The SAS Metadata Server start-up returns a nonzero exit code, indicating an error, but no SAS Metadata Server log is produced.

Note: The screenshot above contains output from an older version of SAS-wsm. The error message at the bottom remains the same for scenarios where the behavior described here occurs.

In many environments, the permissions to control SAS services are inherited through an administrator's group that various Windows accounts belong to. To access this group in PowerShell, run your PowerShell session as an administrator. There are many methods to achieve to run your session as an administrator; however, the simplest method is to right-click the **Windows PowerShell** program and

select **Run as Administrator**.
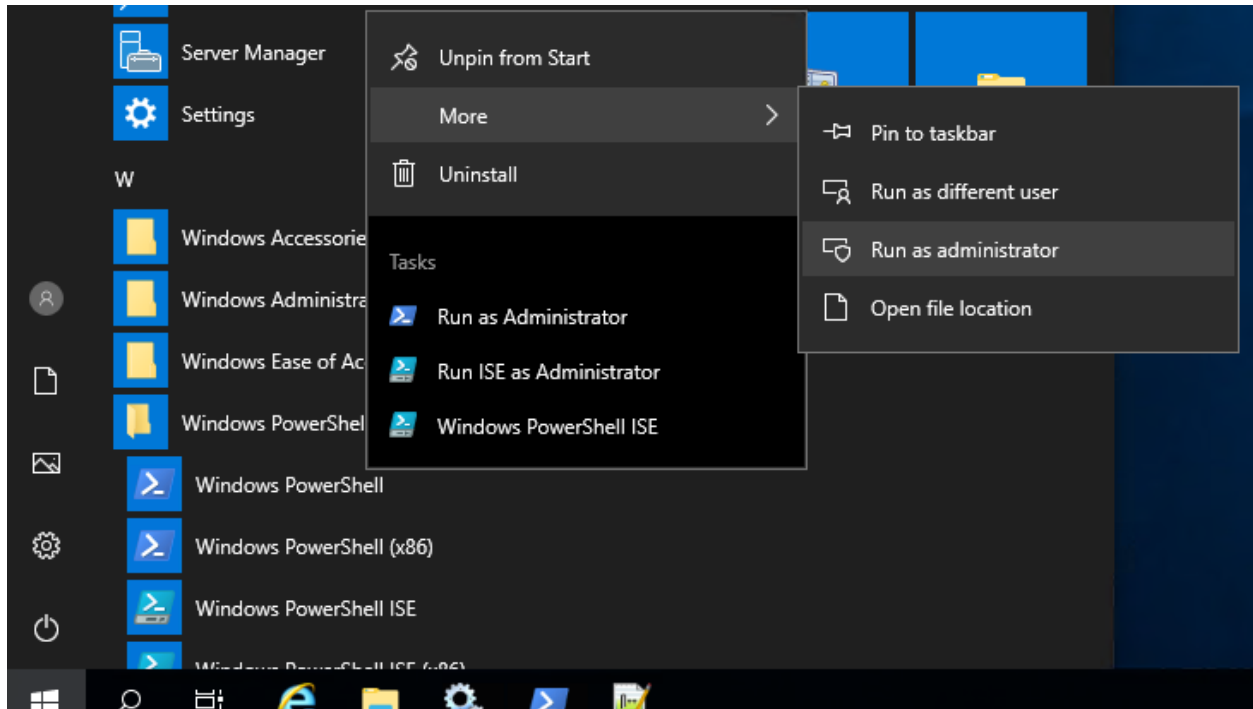
Completing those steps should enable a successful start-up, as shown in the following image: