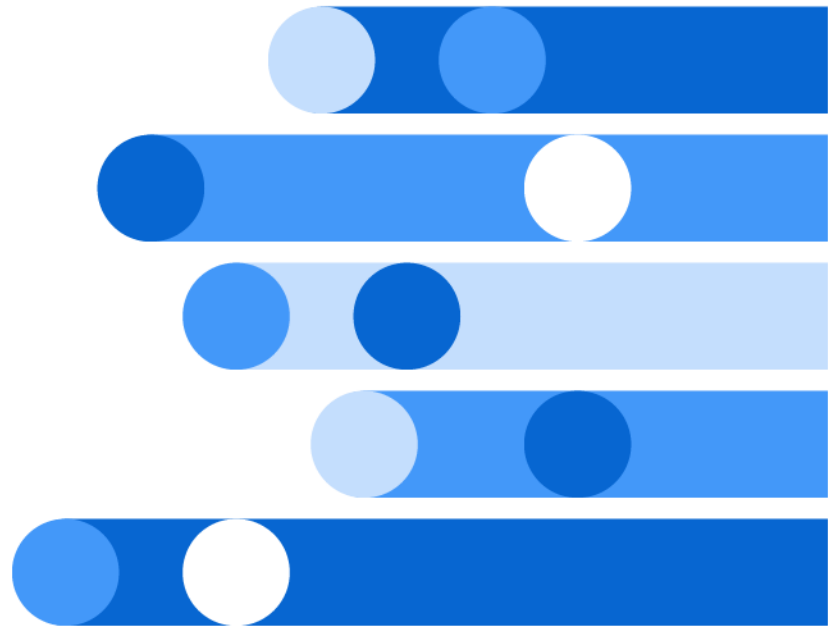




Using the batch Plug-In for the SAS[®] Viya[®] Platform Command-Line Interface

2020.1 - 2025.10*



* This document might apply to additional versions of the software. Open this document in [SAS Help Center](#) and click on the version in the banner to see all available versions.



The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2024. *Using the batch Plug-In for the SAS® Viya® Platform Command-Line Interface*. Cary, NC: SAS Institute Inc.

Using the batch Plug-In for the SAS® Viya® Platform Command-Line Interface

Copyright © 2024, SAS Institute Inc., Cary, NC, USA

All Rights Reserved. Produced in the United States of America.

For a hard copy book: No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

For a web download or e-book: Your use of this publication shall be governed by the terms established by the vendor at the time you acquire this publication.

The scanning, uploading, and distribution of this book via the Internet or any other means without the permission of the publisher is illegal and punishable by law. Please purchase only authorized electronic editions and do not participate in or encourage electronic piracy of copyrighted materials. Your support of others' rights is appreciated.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government. Use, duplication, or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a), and DFAR 227.7202-4, and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007). If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation. The Government's rights in Software and documentation shall be only those set forth in this Agreement.

SAS Institute Inc., SAS Campus Drive, Cary, NC 27513-2414

October 2025

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

v_001-P1:gsub

Contents

Chapter 1 / Getting Started with the batch Plug-In	1
Introduction	1
Set Up the batch Plug-in	4
Working with the Command Line Interface	6
Configuring the SAS Viya Platform for Use with Python, R, and Other Open Source Languages	7
Chapter 2 / Submitting Programs and Commands	9
Submit SAS Programs in Batch	9
Run Commands in Batch	16
Run a SAS Session Interactively	20
Chapter 3 / Managing Jobs, Contexts, and File Sets	23
Manage Jobs	24
Manage Contexts	30
Manage File Sets	39
Chapter 4 / Reference	43
Available Default Contexts	43
Optional Environment Variables for Options	44
Output Specifications and File Locations	44
Runtime Variables	46

Getting Started with the batch Plug-In

Introduction	1
Why Use the batch Plug-in?	1
Who Uses the batch Plug-in?	2
Process	2
Set Up the batch Plug-in	4
About Installation	4
Enable the Ability to Submit Commands in Batch or to Run Interactively	4
Working with the Command Line Interface	6
Command Line Interface Information	6
Tips for Debugging	6
Configuring the SAS Viya Platform for Use with Python, R, and Other Open Source Languages	7

Introduction

Why Use the batch Plug-in?

The batch plug-in to the sas-viya command line interface enables you to use the SAS Batch Server to process programs and commands. For details, see [“Concepts” in SAS Viya Platform: Programming Run-Time Servers](#).

The batch plug-in provides the following advantages:

Asynchronous processing

It enables you to submit programs or commands from a command line to a SAS Viya platform environment running in a Kubernetes cluster for batch processing. You do not have to remain connected to the cluster after submitting the program. These are examples of supported program and commands:

- SAS programs
- Python, R, and other open source programs
- shell commands

See [“Submit SAS Programs in Batch” on page 9](#) and [“Run Commands in Batch” on page 16](#).

Synchronous processing

The plug-in also enables you to start an interactive SAS session in a Kubernetes pod. You can submit programs and commands and wait for the results. You can optionally watch the output, too. See [“Run a SAS Session Interactively” on page 20](#).

Who Uses the batch Plug-in?

The following groups use the batch plug-in:

Programmers

submit SAS jobs and monitor the status of SAS job submissions. When the SAS Viya platform environment is configured to do so, programmers can also submit Python, R, and other open source programs. See [Working with the Command Line Interface](#), [Submitting Programs and Commands](#), [Manage Jobs](#), and [Manage File Sets](#).

SAS administrators

create and manage batch contexts for use by programmers. Also, to enable batch and interactive processing with appropriate protections, a SAS administrator must modify the authorization rules in order to grant permission to a user or group of users. See [Setup the batch Plug-in](#) and [Managing Jobs, Contexts, and File Sets](#).

Note: Kubernetes administrators configure the SAS Viya platform environment so that programmers can submit Python, R, and other open source programs. See [“Configuring the SAS Viya Platform for Use with Python, R, and Other Open Source Languages” on page 7](#).

Process

The plug-in runs SAS programs in a job directory in the execution pod, and it uses a file set as a local holding area for the directory. Here is the basic process for using the batch plug-in:

- 1 Choose a context in which to run the command, program, or session. A context is a specification that contains the information (such as user identity and SAS options) that is needed to run the server for command, program, or session

context. You can use the `sas-viya batch contexts create` command or SAS Environment Manager to create a context, or you can use the default context.

- 2 Use the `sas-viya batch jobs` command to submit a SAS program (`sas-viya batch jobs submit-pgm`), submit a command (`sas-viya batch jobs submit-cmd`), or start an interactive SAS session (`sas-viya batch jobs run-saslm`).

IMPORTANT In release 2020.1.1 and later, SAS program (`sas-viya batch jobs submit-pgm`) runs in lockdown mode. Because submit a command (`sas-viya batch jobs submit-cmd`) and start an interactive session (`sas-viya batch jobs run-saslm`) run without restriction, these commands are disabled by default. You must change the rules in order to enable selected users to run these commands. For more information, see [“Enable the Ability to Submit Commands in Batch or to Run Interactively”](#).

See [“LOCKDOWN System Option”](#) in *SAS Viya Platform: Programming Run-Time Servers* for information about using lockdown mode.

- 3 When you submit the program or command, the batch plug-in creates a file set. This file set contains the program or command and any additional files that you have added to the file set (such as input data sets or autoexec files). You see a message such as The file set "JOB_20200818_132833_572_1" was created.
- 4 After the file set is created, the program (or command) and associated files are downloaded to a job directory in the execution pod in which the program runs. You see a message such as The job was submitted. ID: "dda4268e-0b92-4524-94ce-9b8fa6d07c26" Workload Orchestrator job ID: "613".

You can use the `BATCHJOBDIR` or `GRIDWORK` environment variable to reference the job directory in your SAS program or command script. If you are running a command, the path to the job directory is also added to the `PATH` environment variable so that an uploaded script file can be used from the file set.

- 5 After the job is submitted, it is included in the list of all jobs that have been submitted. You can use the `sas-viya batch jobs list` command to view the list of jobs. A job remains in the list until you retrieve its results or delete it.
- 6 After the job completes, the output and program log are created in the job directory and then uploaded to the file set. You can use the `--wait` options when submitting a program or command to specify different ways of waiting for and handling the job results.
- 7 Use the `sas-viya batch jobs get-results` command to view the results of the program or command. When you run this command, the files in the file set are downloaded to a local directory, and the file set and job are deleted.

Set Up the batch Plug-in

About Installation

The batch plug-in to sas-viya is installed with the sas-viya command-line interface (CLI). For detailed information about how to install the CLI and its plug-ins, see [“Command-Line Interface: Instructions when Downloading the CLI” in SAS Viya Platform: Using the Command-Line Interface](#).

IMPORTANT As of release 2020.1.1, some commands are disabled by default. For more information, see [“Enable the Ability to Submit Commands in Batch or to Run Interactively”](#).

Enable the Ability to Submit Commands in Batch or to Run Interactively

In release 2020.1.1 and later, the commands that are used to submit commands in batch (using `sas-viya batch jobs submit-cmd`) and to start interactive processing (using `sas-viya batch jobs run-saslm`) are run without restriction. Therefore, as a protection, these commands are disabled for all users by default. If a user tries to run a command that they have not been granted permission to run, this message appears: `The current user is not authorized to perform this command.`

To enable batch and interactive processing with appropriate protections, a SAS administrator must modify the authorization rules in order to grant permission to a user or group of users.

Complete these steps:

- 1 In SAS Environment Manager, select  **Rules**.
- 2 On the Rules page, search for the text `/batch/jobs`.
- 3 To enable the ability to submit a command using the `sas-viya batch jobs submit-cmd` command, locate the two rules that control this capability. Both rules contain these parameters:

Object URI
`/batch/jobs`

Setting

Prohibit

Description

Prohibits everyone the ability to run commands in batch

- 4 Right-click on one of the two rules and select **Edit**.
- 5 In the Edit Rule window, select **Grant** in the **Rule type** field.
- 6 In the **Principal type** field, select the types of users who should have the ability to run commands. Note that if you select **Group** in this field, you can specify only a single user group.
- 7 Select **Save** to save the rule changes.
- 8 Repeat steps 4 through 6 to change the other rule that controls the ability to submit a command.
- 9 To enable the ability to start an interactive SAS session using the `sas-viya batch jobs run-saslm` command, locate the two rules that control this capability. Both rules contain these parameters:

Object URI

/batch/jobs

Setting

Prohibit

Description

Prohibits everyone the ability to run SAS interactively

- 10 Right-click on one of the two rules and select **Edit**.
- 11 In the Edit Rule window, select **Grant** in the **Rule type** field.
- 12 In the **Principal type** field, select the types of users who should have the ability to start an interactive SAS session. Note that if you select **Group** in this field, you can specify only a single user group.
- 13 Select **Save** to save the rule changes.
- 14 Repeat steps 10 through 13 to change the other rule that controls the ability to start an interactive SAS session.

See [“Rules Page” in SAS Environment Manager: User’s Guide](#) for more information about rules.

See [“LOCKDOWN System Option” in SAS Viya Platform: Programming Run-Time Servers](#) for information about using lockdown mode.

Working with the Command Line Interface

Command Line Interface Information

In order to use the sas-viya batch plug-in, it's important that you are familiar with the following information:

- the [CLI syntax](#) and the integrated help within the CLI
- key points and details about [using the CLI](#)

Tips for Debugging

You can specify the global option `--verbose` to run a command in debug mode. For more information, see the [CLI syntax](#) and the integrated help within the CLI.

The following information is returned:

- REST API calls
- the debug output from the sas-batch-cli service
- a log named `SASBatch.<user>.<datetime>.<hostname>.<process-id>.log` is created and returned with the other files in the file set
- If the `--watch-output` option is also specified, a log from the STDIO redirection utility named `redir.<user>.<datetime>.<hostname>.<process-id>.log` is created and returned with the other files in the file set.

Configuring the SAS Viya Platform for Use with Python, R, and Other Open Source Languages

You can use the `sas-viya batch job` subcommand `submit-cmd` to run Python and R programs. First, you must use SAS Configurator for Open Source to enable the SAS Viya platform for use with Python, R, and other open source languages.

- This application facilitates the download and management of Python and R from source.
- It provides a build of Python or R in a PVC. The PVC and the build that it contains are then referenced by pods that require Python or R for their operations.

For detailed information about deployment, configuration (including security), and enablement of Python and R, see [SAS Viya Platform: Integration with External Languages](#).

Note: For information about the `sas-viya batch job` command, see “[Run Commands in Batch](#)”.

Submitting Programs and Commands

Submit SAS Programs in Batch	9
Syntax for Commands to Submit a SAS Program in Batch	9
Examples of Batch CLI Commands That Submit a SAS Program in Batch	14
Run Commands in Batch	16
About This Command	16
Syntax for Commands to Run in Batch	16
Examples of Batch CLI Commands That Submit Commands	19
Run a SAS Session Interactively	20
Syntax to Run a SAS Session Interactively	20
Example: Interactive SAS Session	21

Submit SAS Programs in Batch

.....

Note: As of release 2020.1.1, this command runs in lockdown mode. See [“LOCKDOWN System Option” in SAS Viya Platform: Programming Run-Time Servers](#) for information about using lockdown mode.

.....

Syntax for Commands to Submit a SAS Program in Batch

Here is the syntax to submit a SAS program to run in batch:

```
sas-viya batch jobs submit-pgm
--context context
--job-file file_name
```

```

--job-name name
--job-option option
[--queue-name queue_name]
[--rem-list-path path]
[--rem-log-path path]
[--rem-output-dir directory]
--rem-pgm-path | --pgm-path program
[--restart-datastep | --restart-label | --restart-job]
[--results-dir directory]
[--sas-option options ]
[--wait | --wait-log-list | --wait-no-results | --wait-results | --wait-results-no-log]
[--wait-timeout timeout-value]
[--watch-output]

```

--context *context*
 specifies the batch context in which the program is run. Valid aliases for this option are --ctxt, --cxt, and --c.

Note: You can use the default context. See [“Available Default Contexts”](#).

--job-file *file_name*
 specifies a local file (such as an input data set) that is stored in the file set and then copied to the batch job directory, where it can then be used by the SAS program. To specify multiple files, use multiple instances of --job-file *name*. Valid aliases for this option are --jobfile and --file.

Note: This option should be used for small files. Large files should be in a volume mounted to the batch server pod.

--job-name *name*
 specifies the name of the job as it appears in the cluster. If you do not specify this option, the SAS program name is used. For example, if you submit a program named `myprogram.sas`, the default job name is `myprogram`. Valid aliases for this option are --jobname and --name.

--job-option *option*
 specifies options that you want to pass with the job to the cluster. To specify multiple options, use multiple instances of --job-option *option*. Valid aliases for this option are --joboption and --jobopt.

--queue-name *queue_name*
 specifies the name of the SAS Workload Orchestrator queue to use when submitting the batch job. This option is ignored if a queue is already specified in the batch service context or in the launcher context identified in that batch service context. A valid alias for this option is --queue.

This option is available as of release 2021.2.1.

--rem-list-path *path*
 specifies the file name and, optionally, the path to use for the output SAS list file:

- If the path is not specified, the SAS list file name is based on the SAS program name.

- If the specified path is only a file name, the SAS list file is placed in the file set or the specified output directory.
- If the specified path includes a directory, the path must be accessible by the batch environment.

For more information, see [“Input and Output File Storage” in SAS Viya Platform: Programming Run-Time Servers](#). For details, see [“Output Specifications and File Locations”](#) and [“Runtime Variables”](#).

Valid aliases for this option are `--remlistpath` and `--remlist`.

This option is available as of version 1.9.0 of the batch CLI. The reusable batch server feature is available as of version 1.25.0 of the batch service.

`--rem-log-path path`

specifies the file name and, optionally, the path to use for the output SAS log file:

- If the path is not specified, the SAS log file name is based on the SAS program name.
- If the specified path is only a file name, the SAS log file is placed in the file set or specified output directory.
- If the specified path includes a directory, the path must be accessible by the Batch environment.

For more information, see [“Input and Output File Storage” in SAS Viya Platform: Programming Run-Time Servers](#). For details, see [“Output Specifications and File Locations”](#) and [“Runtime Variables”](#).

Valid aliases for this option are `--remlogpath` and `--remlog`.

This option is available as of version 1.9.0 of the batch CLI. The reusable batch server feature is available as of version 1.25.0 of the batch service.

`--rem-output-dir directory`

specifies a directory in the server that is used to store job output. The default is to upload the files to the file set.

For more information, see [“Input and Output File Storage” in SAS Viya Platform: Programming Run-Time Servers](#). For details, see [“Output Specifications and File Locations”](#) and [“Runtime Variables”](#).

Valid aliases for this option are `--remoutputdir` and `--remoutdir`.

This option is available as of version 1.9.0 of the batch CLI. The reusable batch server feature is available as of version 1.25.0 of the batch service.

`--rem-pgm-path | --pgm-path program`

specifies the path and file name of a SAS program that you want to run.

If the path to the program is on the client machine, specify the `--pgm-path` option. The specified program is uploaded to the file set so that it can be downloaded to the job directory and accessed on the execution pod. Valid aliases for this option are `--pgmpath` and `--pgm`.

If the program is directly accessible by SAS in the Kubernetes pod, specify the `--rem-pgm-path` option. Because the program is directly accessible by the

execution pod, it does not need to be added to the file set. Valid aliases for this option are `--rempgmpath` and `--rempgm`.

`--restart-datastep` | `--restart-label` | `--restart-job`
specifies that the SAS program can be restarted from either a SAS DATA step checkpoint or a label that is included in the SAS program.

As of release 2021.2.4, specifies that a job can be restarted, from the beginning, without using a SAS DATA step or label checkpoint.

Valid aliases for `--restart-label` are `--restartlabel` and `--lrestart`.

Valid aliases for `--restart-job` are `--restartjob` and `--jrestart`.

Valid aliases for `--restart-datastep` are `--restartdatastep` and `--drestart`.

Note: As of release 2021.2.4, the `--restart` alias is renamed to `--drestart`.

`--results-dir` *directory*
specifies the directory to store the results of commands or SAS programs if you also specified a `--wait` option that returns results. If you do not specify this option, the default value is used, which is the current directory. Valid aliases for this option are `--resultsdir` and `--dir`.

`--sas-option` *options*
specifies any SAS program options to be applied to the SAS program. If you are specifying options that contain quotation marks, you can enclose all the options in parentheses before enclosing them in quotation marks. To specify multiple program options, you can either use multiple instances of `--sas-option` or specify all the options inside a set of quotation marks. Both of these forms are valid:

- `--sas-option option-1 --sas-option option-2`
- `--sas-option "option-1 option-2"`

Valid aliases for this option are `--sasoption` and `--sasopt`.

`--wait`
specifies that the batch plug-in waits until the job has completed running (either successfully or with an error).

This option cannot be used with any of the other wait options (`--wait-log-list`, `--wait-no-results`, `--wait-results`, `--wait-results-no-log`).

`--wait-log-list`
specifies that the batch plug-in waits until the job has completed running (either successfully or with an error). It then downloads the job information to the results directory, copies the log and list files to the current directory, and then deletes the job information from the results directory. If the job does not complete, it must be ended manually.

This option cannot be used with any of the other wait options (`--wait`, `--wait-no-results`, `--wait-results`, `--wait-results-no-log`).

A valid alias for this option is `--waitloglist`.

--wait-no-results

specifies that the batch plug-in waits until the job has completed running (either successfully or with an error). It does not return results, the job log, or the output listing, but deletes the results, the job log, the output listing, along with the job directory. If the job does not complete, it must be ended manually.

This option cannot be used with any of the other wait options (`--wait`, `--wait-log-list`, `--wait-results`, `--wait-results-no-log`).

A valid alias for this option is `--waitnoresults`.

--wait-results

specifies that the batch plug-in waits until the job has completed running (either successfully or with an error). It then saves the results of the processing to the results directory and copies the job log and output listing to the current directory. If the job does not complete, it must be ended manually.

This option cannot be used with any of the other wait options (`--wait`, `--wait-log-list`, `--wait-no-results`, `--wait-results-no-log`).

A valid alias for this option is `--waitresults`.

--wait-results-no-log

specifies that the batch plug-in waits until the job has completed running (either successfully or with an error). It then downloads the job information to the results directory but does not copy the log and list files to the current directory. If the job does not complete, it must be ended manually.

This option cannot be used with any of the other wait options (`--wait`, `--wait-log-list`, `--wait-no-results`, `--wait-results`).

A valid alias for this option is `--waitresultsnolog`.

--wait-timeout *timeout-value*

specifies that the batch plug-in waits until either the job has completed running (either successfully or with an error) or *timeout-value* (in seconds) is reached. If you use this option with one of the wait options, it enables you to specify how long to wait for results. The default value is 0 (zero), which specifies that the results are returned immediately.

Valid aliases for this option are `--waittimeout` and `--timeout`.

--watch-output

specifies that the log and output submitted by the `sas-viya batch` command are displayed on your local machine. A valid alias for this option is `--watchoutput`.

Examples of Batch CLI Commands That Submit a SAS Program in Batch

Example: Submit a Program

This example submits the program `mypgm` to run in the default context.

```
sas-viya batch jobs submit-pgm --pgm-path /User/myuser/Programs/mypgm.sas --  
context default
```

Here is the output from the command submission:

```
>>> File set created, ID="JOB_20200817_191949_741_1"  
>>> Uploading "mypgm.sas"  
>>> Job submitted, ID="40f55940-f387-4a17-af2a960b19", Workload Job ID="613".
```

Example: Add Input Data to the File Set

In this example, the SAS program `mysasprogram.sas` uses the `mydata.csv` file as input data:

```
filename csv "!BATCHJOBDIR/mydata.csv";  
/* Import the CSV file */  
proc import datafile=csv out=work.mysasdata dbms=csv;  
  getnames=yes;  
run;
```

You can use the `--job-file` option to upload `mydata.csv` to the file set so that it can be used by `mysasprogram`.

```
sas-viya batch jobs submit-pgm --pgmpath /User/myuser/Programs/mysasprogram.sas  
--context my-batch-context --job-file /cas-shared/mydata.csv
```

Example: Specify an Autoexec File

In this example, the file `myauto.sas` is used as an autoexec file for the SAS session. The `--job-file` option is used to upload the `myauto.sas` file to the file set, where it is then copied to the batch job directory. It is accessible as `!BATCHJOBDIR`. The `--sasoption` option is used to specify the `-autoexec` SAS option that uses the `myauto.sas` file.

```
sas-viya batch jobs submit-pgm --context default --pgm-path testauto.sas -file  
myauto.sas --sasoption '-autoexec !BATCHJOBDIR/myauto.sas'
```

Example: Return Results and Discard the Log

In this example, only the results of the program are returned, and the program log is discarded.

```
sas-viya batch jobs submit-pgm --pgm-path /User/myuser/Programs/mypgmam.sas --
context my-batch-context --job-file /cas-shared/mydata --wait-results-no-log
```

Example: Submit a Program and Wait for the Results

In this example, the program uses `testEnvVars.sas` file as input data. The job waits until the program completes running and then saves the results.

```
sas-viya batch jobs submit-pgm --context default --pgm-path testEnvVars.sas --
wait-results
```

Here is the output:

```
>>> The file set "JOB_20220209_195209_155_1" was created.
>>> Uploading "testEnvVars.sas".
>>> The job was submitted. ID: "07ef25c4-560d-4278-84ae-33240147ff90" Workload Orchestrator job ID: "5"
>>> Waiting for the job to complete...
>>> Job started.
<<< Job ended.
<<< Downloading the files in the file set to the following path: "./JOB_20220209_195209_155_1".
<<< Downloading "testEnvVars.log".
<<< Downloading "testEnvVars.sas".
<<< Copying the log and list files from "./JOB_20220209_195209_155_1" to the current directory.
<<< Copying "testEnvVars.log".
<<< Deleting the batch job.
<<< Deleting the file set.
```

Example: Specify a Location That Is Mounted in a Pod

In this example, the command specifies the location for the program and its output and logs. The example assumes that these locations are mounted in the pod (`/path/in/pod/for/`). This is done by modifying the Kubernetes pod template.

Note: The SAS Batch context specifies the SAS Launcher context. The Launcher context specifies the Kubernetes pod template.

```
sas-viya batch jobs submit-pgm -c myBatchContext --rem-pgm-path /
path/in/pod/for/programs/myProgram.sas --rem-log-path /path/in/pod/for/output/
myProgram-@STARTTIME@.log --rem-list-path /path/in/pod/for/output/myProgram-
@STARTTIME@.lst
```

where *myBatchContext* is the SAS Batch context.

Note: The example uses the runtime variable @STARTTIME@. See [“Runtime Variables” on page 46](#).

Run Commands in Batch

About This Command

The `sas-viya batch jobs submit-cmd` command enables you to submit commands to the pod.

Note: Commands can include Python and R programs. For details, see [“Configuring the SAS Viya Platform for Use with Python, R, and Other Open Source Languages”](#) and [“Available Default Contexts”](#).

IMPORTANT As of release 2020.1.1, because the `sas-viya batch jobs submit-cmd` command runs without restriction, the command is disabled for all users by default. If a user tries to run this command without being granted permission, this message appears: The current user is not authorized to perform this command.

To enable this command, a SAS administrator must modify the authorization rules in order to grant permission to a user or group of users. For more information, see [“Enable the Ability to Submit Commands in Batch or to Run Interactively”](#).

Syntax for Commands to Run in Batch

Here is the syntax for the commands to run in a specified context in batch:

```
sas-viya batch jobs submit-cmd
--command command
--context context
[--job-file file_name]
[--job-name name]
[--job-option option]
[--queue-name queue_name]
[--rem-log-path path]
```

```
[--rem-output-dir directory]
[--results-dir directory]
[--wait | --wait-log-list | --wait-no-results | --wait-results | --wait-results-no-log]
[--wait-timeout timeout-value]
[--watch-output]
```

`--command command`
 specifies any command that you want to run in the specified context. A valid alias for this option is `--cmd`.

`--context context`
 specifies the batch context in which the command is run. Valid aliases for this option are `--ctxt`, `--cxt`, and `--c`.

Note: As of release 2021.2.6, you can use the `default-cmd` context. See [“Available Default Contexts”](#).

`--job-file file_name`
 specifies a local file that is stored in the file set and then copied to the batch job directory, where it is available for use by the command. The path to the job directory is added to the PATH environment variable so that an uploaded script file can be used from the file set. To specify multiple files, use multiple instances of `--job-file name`. Valid aliases for this option are `--jobfile` and `--file`.

`--job-name name`
 specifies the name of the job as it appears in the cluster. If you do not specify this option, the name SUBMITCMD is used. Valid aliases for this option are `--jobname` and `--name`.

`--job-option option`
 specifies options that you want to pass with the job to the cluster. To specify multiple options, use multiple instances of `--job-option option`. Valid aliases for this option are `--joboption` and `--jobopt`.

`--queue-name queue_name`
 specifies the name of the SAS Workload Orchestrator queue to use when submitting the batch job. This option is ignored if a queue is already specified in the batch service context or in the launcher context identified in that batch service context. A valid alias for this option is `--queue`.

This option is available as of release 2021.2.1.

`--rem-log-path path`
 specifies the file name and, optionally, the path to use for the output SAS log file:

- If the path is not specified, the log file name is “submitcmd.log”.
- If the specified path is only a file name, the SAS log file is placed in the file set or specified output directory.
- If the specified path includes a directory, the path must be accessible by the Batch environment.

For details, see [“Output Specifications and File Locations”](#) and [“Runtime Variables”](#).

Valid aliases for this option are `--remlogpath` and `--remlog`.

This option is available as of version 1.9.0 of the batch CLI. The reusable batch server feature is available as of version 1.25.0 of the batch service.

`--rem-output-dir directory`

specifies a directory in the server that is used to store job output. The default is to upload the files to the file set.

For more information, see [“Input and Output File Storage” in SAS Viya Platform: Programming Run-Time Servers](#). For details, see [“Output Specifications and File Locations”](#) and [“Runtime Variables”](#).

Valid aliases for this option are `--remoutputdir` and `--remoutdir`.

This option is available as of version 1.9.0 of the batch CLI. The reusable batch server feature is available as of version 1.25.0 of the batch service.

`--results-dir directory`

specifies the directory to store the results of commands or SAS programs if you also specified a `--wait` option that returns results. If you do not specify this option, the default value is used, which is the current directory. Valid aliases for this option are `--resultsdir` and `--dir`.

`--wait`

specifies that the batch plug-in waits until the job has completed running (either successfully or with an error).

This option cannot be used with any of the other wait options (`--wait-log-list`, `--wait-no-results`, `--wait-results`, `--wait-results-no-log`).

`--wait-log-list`

specifies that the batch plug-in waits until the job has completed running (either successfully or with an error). It then downloads the job information to the results directory, copies the log and list files to the current directory, and then deletes the job information from the results directory. If the job does not complete, it must be ended manually.

This option cannot be used with any of the other wait options (`--wait`, `--wait-no-results`, `--wait-results`, `--wait-results-no-log`).

A valid alias for this option is `--waitloglist`.

`--wait-no-results`

specifies that the batch plug-in waits until the job has completed running (either successfully or with an error). It does not return results, the job log, or the output listing, but deletes the results, the job log, the output listing, along with the job directory. If the job does not complete, it must be ended manually.

This option cannot be used with any of the other wait options (`--wait`, `--wait-log-list`, `--wait-results`, `--wait-results-no-log`).

A valid alias for this option is `--waitnoresults`.

`--wait-results`

specifies that the batch plug-in waits until the job has completed running (either successfully or with an error). It then saves the results of the processing to the results directory and copies the job log and output listing to the current directory. If the job does not complete, it must be ended manually.

This option cannot be used with any of the other wait options (`--wait`, `--wait-log-list`, `--wait-no-results`, `--wait-results-no-log`).

A valid alias for this option is `--waitresults`.

`--wait-results-no-log`

specifies that the batch plug-in waits until the job has completed running (either successfully or with an error). It then downloads the job information to the results directory but does not copy the log and list files to the current directory. If the job does not complete, it must be ended manually.

This option cannot be used with any of the other wait options (`--wait`, `--wait-log-list`, `--wait-no-results`, `--wait-results`).

A valid alias for this option is `--waitresultsnolog`.

`--wait-timeout timeout-value`

specifies that the batch plug-in waits until either the job has completed running (either successfully or with an error) or *timeout-value* (in seconds) is reached. If you use this option with one of the wait options, it enables you to specify how long to wait for results. The default value is 0 (zero), which specifies that the results are returned immediately.

Valid aliases for this option are `--waittimeout` and `--timeout`.

`--watch-output`

specifies that the stdout | stderr from the command is displayed on your local machine. A valid alias for this option is `.-watchoutput`.

Examples of Batch CLI Commands That Submit Commands

Also see [“Examples of Batch CLI Commands That Submit a SAS Program in Batch” on page 14.](#)

Example: Run a Command in Batch

Here is an example of a command to submit the command `hostname` in the context `default-cmd`, and to save the results of the job after it completes.

```
sas-viya batch jobs submit-cmd --context default-cmd --cmd "hostname" --wait-results
```

Example: Run a Python Program in Batch

Here is an example of a command to submit the Python program in the file `hello.py` in the context `default-cmd`.

```
sas-viya batch jobs submit-cmd --context default-cmd --jobfile hello.py --command "/sas-pyconfig-mount-path/python-profile-name/bin/python3 hello.py"
```

Where:

- *sas-pyconfig-mount-path* is the sas-pyconfig PVC volume mount path defined in the configmap from SAS Configurator for Open Source (sas-pyconfig-parameters). The default path is `/opt/sas/viya/home/sas-pyconfig`.
- *python-profile-name* is the Python profile name defined in the configmap from SAS Configurator for Open Source. The default name is `default_py`.
- `hello.py` is a local text file that contains the following Python code:

```
print("Hello World!")
```

Run a SAS Session Interactively

IMPORTANT As of release 2020.1.1, because the `sas-viya batch jobs run-saslm` command runs without restriction, the command is disabled for all users by default. If a user tries to run this command without being granted permission, this message appears: The current user is not authorized to perform this command.

To enable this command, a SAS administrator must modify the authorization rules in order to grant permission to a user or group of users. For more information, see [“Enable the Ability to Submit Commands in Batch or to Run Interactively”](#).

Syntax to Run a SAS Session Interactively

Here is the syntax for the command to run an interactive SAS line-mode session from the command line. Input comes from and output goes to your terminal session.

```
sas-viya batch jobs run-saslm
--context context
[--job-name name]
[--job-option option]
[--queue-name queue_name]
```

```
--context context
```

specifies the batch context in which the SAS session is run. Valid aliases for this option are `--ctxt`, `--cxt`, and `--c`.

Note: You can use the default context. See [“Available Default Contexts”](#).

`--job-name name`
 specifies the name of the job as it appears in the cluster. If you do not specify this option, the name RUNSASLM is used. Valid aliases for this option are `--jobname` and `--name`.

`--job-option option`
 specifies options that you want to pass with the job to the cluster. To specify multiple options, use multiple instances of `--job-option option`. Valid aliases for this option are `--joboption` and `--jobopt`.

`--queue-name queue_name`
 specifies the name of the SAS Workload Orchestrator queue to use when submitting the batch job. This option is ignored if a queue is already specified in the batch service context or in the launcher context identified in that batch service context. A valid alias for this option is `--queue`.

This option is available as of release 2021.2.1.

Example: Interactive SAS Session

Here is an example of a command to start an interactive SAS session in line mode. It uses the default context.

```
sas-viya batch jobs run-saslm --context default
```

Here is the output of this command. It might take several seconds for the job to process and the prompt to appear.

```
>>> Job submitted, ID="6d96df7c-025d-4cf7-aad6-7259e0caa421".
>>> Waiting for job to complete...
>>> Job started.
NOTE: Copyright (c) 2016 by SAS Institute Inc., Cary, NC, USA.
NOTE: SAS (r) Proprietary Software V.04.00 (TS M0 MBCS3170)
      Licensed to MYCORP-PRODUCTION, Site 12345678.
NOTE: This session is executing on the Linux 3.10.0-957.1.3.el7.x86_64 (LIN
      X64) platform.

NOTE: Additional host information:

      Linux LIN X64 3.10.0-957.1.3.el7.x86_64 #1 SMP Thu Nov 29 14:49:43 UTC 2018
      x86_64 Red Hat Enterprise Linux release 8.2 (Ootpa)

NOTE: SAS initialization used:
      real time          0.04 seconds
      cpu time           0.04 seconds

Group not defined locally. Using GID for restricted options.
User account not defined locally. Using UID for restricted options.
1?
```

After the session prompt appears, enter each SAS statement, one line at a time. Then, enter the `ENDSAS;` statement to end the session.

Managing Jobs, Contexts, and File Sets

Manage Jobs	24
Get Results of Jobs	24
List Jobs	26
Delete a Job	28
Terminate a Job	29
Manage Contexts	30
About This Command	30
Create a Context	30
List Contexts	33
Update Contexts	35
Delete a Context	37
JSON and YAML File Formats	38
Manage File Sets	39
About This Command	39
List File Sets	39
List Files in a File Set	41
Delete a File Set	41

Manage Jobs

Get Results of Jobs

Syntax for Commands to Get Job Results

Use the `sas-viya batch jobs get-results` command to obtain the results for a completed job. If you are not an administrator, you can retrieve the results only for your own jobs. If you are an administrator, you can retrieve the results for all jobs.

After you use this command, any job whose results you retrieved is no longer returned by a `sas-viya batch jobs list` command. The command downloads all the files in the file set, deletes the job, and deletes the file set.

Here is the command syntax:

```
sas-viya batch jobs get-results
[--details]
[--filter filter]
[--job-id job_id]
[--limit maximum_number]
[--results-dir directory]
[--sort-by field_name-1{, field_name-2,...field_name-n}]
[--start record_number]

--details
    shows all the context information.

--filter filter
    lists a subset of the returned objects that match the conditions in the specified
    filter. For more information about filtering, see https://developer.sas.com/rest-apis/.

--job-id job_id
    specifies the ID of the job whose information you want to view. Valid aliases for
    this option are --jobid and --id.

--limit maximum_number
    specifies the maximum number of responses to return at a time. The default is
    to return an unlimited number of results.

--results-dir directory
    specifies the directory in which to store the job results. Valid aliases for this
    option are --resultsdir and --dir.

--sort-by field_name-1{, field_name-2,...field_name-n}
    specifies one or more fields by which the results are sorted.
```

Separate multiple values of *field_name* with commas. By default, the results are sorted in ascending order. Place a ~ character before a *field_name* value in order to sort the field in descending order.

Here are the valid values for *field_name*:

- id
- workloadJobId
- contextId
- filessetId
- processId
- name
- state
- executionHost
- returnCode
- submittedTimeStamp
- startedTimeStamp
- endedTimeStamp
- creationTimeStamp
- modifiedTimeStamp
- createdBy
- modifiedBy

A valid alias for this option is `--sortby`.

`--start record_number`

specifies the record number to start at when returning results. If you do not specify this option, the results are returned starting at the first record.

Example of a Command to Get Job Results

Here is an example of a command to get the results. First, the job is submitted:

```
sas-viya batch jobs submit-pgm --context default --pgm-path testEnvVars.sas
```

This is the output of the command:

```
>>> The file set "JOB_20220209_195011_202_1" was created.
>>> Uploading "testEnvVars.sas".
>>> The job was submitted. ID: "ce362679-0ec5-4052-bbf6-f77277bcb1dF" Workload Orchestrator job ID: "3"
```

Next, the command to get the results of the job:

```
sas-viya batch jobs get-results
```

Here is the output of the command:

ID	Name	Workload	Job ID	Created By	State	Submitted Time	Started
ce362679-0ec5-4052-bbf6-f77277bcb1df	testEnvVars	3		user12	completed	2022-02-09T19:50:14	
2022-02-09T19:50:14	2022-02-09T19:50:32	0					
<<< Downloading the files in the file set to the following path: "./JOB_20220209_195011_202_1".							
<<< Downloading "testEnvVars.log".							
<<< Downloading "testEnvVars.sas".							

Example of a Command to Get Job Results for a Specified Job

Here is an example of a command to get the results of a single job, which is specified by the job ID:

```
sas-viya batch jobs get-results --job-id 40f55940-f387-4a17-alfd-1aaf2a960b19
```

Here is the output of the command:

ID	Name	Workload	Job ID	Created By	State	Submitted Time	Started
40f55940-f387-4a17-alfd-1aaf2a960b19	mypgm	423		myuser	completed	2020-08-17T15:19:52	
2020-08-17T15:19:56	2020-08-17T15:20:09	1					
<<< Downloading the files in the file set to the following path: "./JOB_20200817_191949_741_1".							
<<< Downloading "mypgm.sas"							
<<< Downloading "mypgm.log".							

List Jobs

Syntax for Commands to List Jobs

Use the `sas-viya batch jobs list` command to list jobs. After you submit a command or program, the resulting job remains in the job list until you either get the job results or delete the job.

If you are not an administrator, you can list only your jobs. If you are an administrator, you can list jobs from all users. Here is the command syntax:

```
sas-viya batch jobs list
[--details]
[--filter filter]
[--ids-only]
[--job-id job_id]
[--limit maximum_number]
[--name job_name]
[--sort-by field_name-1{, field_name-2,...field_name-n}]
[--start record_number]
```

- `--details`
shows all the context information.
- `--filter filter`
lists a subset of the returned objects that match the conditions in the specified filter. For more information about filtering, see <https://developer.sas.com/rest-apis/>.
- `--ids-only`
specifies that only the object IDs are returned. Valid aliases for this option are `--idonly` and `--i`.
- `--job-id job_id`
specifies the ID of the job whose information you want to view. Valid aliases for this option are `--jobid` and `--id`.
- `--limit maximum_number`
specifies the maximum number of responses to return at a time. The default is to return an unlimited number of results.
- `--name job_name`
filters the results by the specified job name.
- `--sort-by field_name-1{, field_name-2,...field_name-n}`
specifies one or more fields by which the results are sorted.

Separate multiple values of *field_name* with commas. By default, the results are sorted in ascending order. Place a `~` character before a *field_name* value in order to sort the field in descending order.

Here are the valid values for *field_name*:

- `id`
- `workloadJobId`
- `contextId`
- `filesetId`
- `processId`
- `name`
- `state`
- `executionHost`
- `returnCode`
- `submittedTimeStamp`
- `startedTimeStamp`
- `endedTimeStamp`
- `creationTimeStamp`
- `modifiedTimeStamp`
- `createdBy`
- `modifiedBy`

A valid alias for this option is `--sortby`.

```
--start record_number
```

specifies the record number to start at when returning results. If you do not specify this option, the results are returned starting at the first record.

Examples of Commands to List Jobs

Here is an example of a command to list all jobs. The list is sorted by the time when the job started:

```
sas-viya batch jobs list --sort-by startedTimeStamp
```

Here is the output of the command:

```
40f55940-f387-4a17-afd-1aaf2a960b19 mypgm 567 myuser failed 2020-08-17T15:19:52 2020-08-17T15:19:56
2020-08-17T15:20:09 1
bf4dac59-33f8-4555-be96-c2bcf6337289 mypgm2 568 myuser failed 2020-08-17T15:29:15 2020-08-17T15:29:21
2020-08-17T15:29:34 1
a40cfe27-6903-40b5-a6d4-3e92835264f2 SUBMITCMD 569 myuser pending 2020-08-17T17:58:32
```

Here is an example of a command to list the details of a single job, which is specified by a job ID:

```
sas-viya batch jobs list --job-id 40f55940-f387-4a17-afd-1aaf2a960b19 --
details
```

Here is the output of the command:

ID	Name	Workload	Job ID	Created By	Context ID	File Set
ID	Process ID		State	Execution Host	Return Code	Submitted Time
Time	Ended Time	Creation Time	Modified By	Modified Time		Started
40f55940-f387-4a17-afd-1aaf2a960b19	mypgm	856	myuser	4811b40e-f432-410e-afb5-76b96828066e		
JOB_20200817_191949_741_1	a4cff3e0-afc8-48b4-a33c-390f15d0c401	failed	10.254.6.90	1	2020-08-17T15:19:52	
2020-08-17T15:19:56	2020-08-17T15:20:09	2020-08-17T15:19:52	sas.launcher	2020-08-17T15:20:14		

Delete a Job

Syntax for Commands to Delete a Job

Use the `sas-viya batch jobs delete` command to delete a job. If you are not an administrator, you can delete only your jobs. If you are an administrator, you can delete jobs from any user. Deleting a job does not delete the job's file set.

Here is the command syntax:

```
sas-viya batch jobs delete
--job-id job_id | --name job_name | --uri uri
[--force]
```

`--job-id job_id | --name job_name | --uri uri`

specifies the job to delete by specifying either the job ID, job name, or job URI.

Valid aliases for the `--job-id` option are `--jobid` and `--id`. A valid alias for the `--name` option is `--n`. A valid alias for the `--uri` option is `--u`.

`--force`

specifies that the job is deleted without first displaying a confirmation message.

A valid alias for this option is `--f`.

Example of a Command to Delete Jobs

Here is an example of a command to delete a single job, which is specified by a job ID:

```
sas-viya batch jobs delete --job-id bf4dac59-33f8-4555-be96-c2bcf6337289
```

Here is the output from that command:

```
Are you sure that you want to delete the job "bf4dac59-33f8-4555-be96-c2bcf6337289" (yes/no)?> yes
The job "bf4dac59-33f8-4555-be96-c2bcf6337289" was deleted successfully.
```

Terminate a Job

Syntax for Commands to Terminate a Job

Use the `sas-viya batch jobs kill` command to terminate a job. If you are not an administrator, you can terminate only your jobs. If you are an administrator, you can terminate jobs from all users. Here is the command syntax:

```
sas-viya batch jobs kill
```

```
--job-id job_id | --name job_name | --uri uri
[--force]
```

`--job-id job_id | --name job_name | --uri uri`

specifies the job to terminate by specifying either the job ID, job name, or job URI.

Valid aliases for the `--job-id` option are `--jobid` and `--id`. A valid alias for the `--name` option is `--n`. A valid alias for the `--uri` option is `--u`.

`--force`

specifies that the job is terminated without first displaying a confirmation message. A valid alias for this option is `--f`.

Example of a Command to Terminate a Job

Here is an example of a command to terminate a single job, which is specified by a job ID:

```
sas-viya batch jobs kill --job-id bf4dac59-33f8-4555-be96-c2bcf6337289
```

Manage Contexts

About This Command

Use the `sas-viya batch contexts` command to manage batch contexts. A context is a specification that contains the information (such as user identity and SAS options) that is needed to run the server by using a command, a program, or a session. For more information, see [SAS Batch Contexts](#) and “Introduction to Batch Processing” in *SAS Viya Platform: Programming Run-Time Servers*.

Note:

- By default, you can run this command only if you are a SAS administrator. To enable non-administrators to use this command, adjust the authorization rules. See “General Authorization: Concepts” in *SAS Viya Platform: General Authorization*.
 - Two default contexts are available for use depending on the type of program that you want to run. See “Available Default Contexts”.
 - You can also use SAS Environment Manager to manage these contexts. See “Contexts Page” in *SAS Environment Manager: User's Guide*.
-

Create a Context

Syntax for Commands to Create a Batch Context

Here is the command syntax:

```
sas-viya batch contexts create  
--json-file file_name | --yaml-file file_name
```

```

[--description description]
[--idle-server-time-out "seconds"]
--launcher-context-id context_id | --launcher-context-name context_name
[--maximum-servers "number"]
[--minimum-servers "number"]
--name context_name
[--queue-name queue_name]
[--reusable-server]
[--run-server-as user]
[--sas-options options]

--json-file file_name | --yaml-file file_name
  specifies either a JSON file or a YAML file containing the information that is
  needed to create the context.

```

You can also use a combination of a JSON or YAML file and options to create a context. For example, you could specify a standard set of options in one file, and then specify another file with a unique option.

For file format information, see [“JSON and YAML File Formats”](#).

```

--description description
  specifies the description of the context. A valid alias for this option is --desc.

--idle-server-time-out "seconds"
  specifies the idle time out (in seconds) for the reusable batch server. All
  reusable batch servers that are created beyond the specified minimum number
  of servers (--minimum-servers) terminate after being idle for this specified
  amount of time.

```

- If this value is not specified, the default is 300 seconds or 5 minutes.
- If the context is not for a reusable batch server, this option is ignored.

For more information, see [“Reusable Batch Servers” in SAS Viya Platform: Programming Run-Time Servers](#).

This option is available as of version 1.9.0 of the batch CLI. The reusable batch server feature is available as of version 1.25.0 of the batch service.

```

--launcher-context-id context_id
  specifies the ID of the launcher context that is used to launch batch jobs. The
  value can be provided with this option or via the --json-file or --yaml-file
  options.

--launcher-context-name context_name
  specifies the name of the launcher context that is used to launch batch jobs.
  The value can be provided with this option or via the --json-file or --yaml-
  file options.

--maximum-servers "number"
  specifies the maximum number of reusable batch servers that are allowed to
  run for each combination of user and context.

```

- If this value is not specified, reusable batch servers are added as needed.
- If the context is not for a reusable batch server, this option is ignored.

For more information, see [“Reusable Batch Servers” in SAS Viya Platform: Programming Run-Time Servers](#).

A valid alias for this option is `--max-servers`.

This option is available as of version 1.9.0 of the batch CLI. The reusable batch server feature is available as of version 1.25.0 of the batch service.

`--minimum-servers number`

specifies the minimum number of multi-job servers that must be running.

- If this value is not specified, all reusable batch servers are started as jobs are submitted.
- If this value is specified, a service account must be specified by using the `--run-server-as` option.
- If the context is not for a reusable batch server, this option is ignored.

For more information, see [“Reusable Batch Servers” in SAS Viya Platform: Programming Run-Time Servers](#).

A valid alias for this option is `--min-servers`.

This option is available as of version 1.9.0 of the batch CLI. The reusable batch server feature is available as of version 1.25.0 of the batch service.

`--name context_name`

specifies the name of the batch context. A valid alias for this option is `--n`.

`--queue-name queue_name`

specifies the name of the SAS Workload Orchestrator queue to use when submitting this batch context. A valid alias for this option is `--queue`.

`--reusable-server`

specifies whether the context uses reusable batch servers. For more information, see [“Reusable Batch Servers” in SAS Viya Platform: Programming Run-Time Servers](#).

If this flag is not specified, a batch server is created for each job.

A valid alias for this option is `--reusable`.

This option is available as of version 1.9.0 of the batch CLI. The reusable batch server feature is available as of version 1.25.0 of the batch service.

`--run-server-as user`

specifies the shared service account user that jobs using this context will run under. For information about shared service accounts, see [“SAS Batch Servers Running under a Shared Account” in SAS Viya Platform: Programming Run-Time Servers](#).

Valid aliases for this option are `--run-as` and `--service-account`.

`--sas-options options`

specifies SAS options to add to SAS batch jobs when using this batch context. To specify multiple options, you must enclose the options within quotation marks and separate each option by a space. For example: `--sas-options "option-1 option-2"`.

For information about reusable batch servers, see [“Input and Output File Storage” in SAS Viya Platform: Programming Run-Time Servers](#).

Examples of Commands to Create a Batch Context

Here is an example of a command to create a batch context using a JSON file and specifying the launcher context name:

```
sas-viya batch contexts create --json-file /path/to/JSON/file/containing/
ContextRequest.json --launcher-context-name "My Launcher Name"
```

Here is an example of a command to create a batch context using a YAML file and specifying the launcher context ID:

```
sas-viya batch contexts create --yaml-file /path/to/YAML/file/containing/
ContextRequest.yaml --launcher-context-id "05e5c6bc-ff5d-49af-
bb51-1108a214499c"
```

List Contexts

Syntax for Commands to List Contexts

Here is the syntax to list batch contexts.

The format of the list (text or JSON) is determined by the output type that is selected when you created a profile.

```
sas-viya batch contexts list
[--details]
[--ids-only]
[--limit maximum_number]
[--name context_name | --filter filter]
[--sort-by field_name-1{, field_name-2,...field_name-n}]
[--start record_number]
```

--details
shows all the context information.

--filter *filter*
lists a subset of the returned objects that match the conditions in the specified filter. For more information about filtering, see <https://developer.sas.com/rest-apis/>.

--ids-only
specifies that only the object IDs are returned. Valid aliases for this option are **--idsonly** and **--i**.

--limit *maximum_number*
specifies the maximum number of responses to return at a time. The default is to return an unlimited number of results.

--name *context_name*
lists the contexts whose name matches the specified name. A valid alias for this option is **--n**.

`--sort-by field_name-1{, field_name-2,...field_name-n}`
 specifies one or more fields by which the results are sorted.

Separate multiple values of *field_name* with commas. By default, the results are sorted in ascending order. Place a ~ character before a *field_name* value in order to sort the field in descending order.

Here are the valid values for *field_name*:

- id
- name
- description
- launcherContextId
- queueName
- sasOptions
- providerOptions
- requiredResources
- creationTimeStamp
- modifiedTimeStamp
- createdBy
- modifiedBy

A valid alias for this option is `--sortby`.

`--start record_number`
 specifies the record number to start at when returning results. If you do not specify this option, the results are returned starting at the first record. The first context is context #0.

Examples of Commands to List Contexts

Here is an example of a command to list contexts and to sort by the context name:

```
sas-viya batch contexts list --sort-by name
```

Here is an example of a command to list contexts that match a specified filter. The filter returns contexts whose ID is equal to the supplied string.

```
sas-viya batch contexts list --filter "eq(id,'05e5c6bc-ff5d-49af-  
bb51-1108a214499c')"
```

Update Contexts

Syntax for Commands to Update Contexts

Here is the command syntax to update an existing batch context.

```
sas-viya batch contexts update
[--json-file file_name | --yaml-file file_name]
[--id object_id]
[--idle-server-time-out "seconds"]
[--description description]
--launcher-context-id context_id | --launcher-context-name context_name
[--maximum-servers "number"]
[--minimum-servers "number"]
[--name context_name]
[--queue-name queue_name]
[--reusable-server]
[--run-server-as user]
[--sas-options options]
```

--json-file *file_name* | --yaml-file *file_name*
specifies either a JSON file or a YAML file containing the information that is needed to update the context.

You can also use a combination of a JSON or YAML file and options to create a context. For example, you could specify a standard set of options in one file, and then specify another file with a unique option.

For file format information, see [“JSON and YAML File Formats”](#).

--description *description*
specifies the description of the context. A valid alias for this option is --desc.

--id *object_id*
specifies the object ID of the context to be updated. A valid alias for this option is --i.

--idle-server-time-out "*seconds*"
specifies the idle time out (in seconds) for the reusable batch server. All reusable batch servers that are created beyond the specified minimum number of servers (--minimum-servers) terminate after being idle for this specified amount of time.

- If this value is not specified, the default is 300 seconds or 5 minutes.
- If the context is not for a reusable batch server, this option is ignored.

For more information, see [“Reusable Batch Servers” in SAS Viya Platform: Programming Run-Time Servers](#).

This option is available as of version 1.9.0 of the batch CLI. The reusable batch server feature is available as of version 1.25.0 of the batch service.

`--launcher-context-id context_id`
 specifies the ID of the launcher context that is used to launch batch jobs. The value can be provided with this option or via the `--json-file` or `--yaml-file` options.

`--launcher-context-name context_name`
 specifies the name of the launcher context that is used to launch batch jobs. The value can be provided with this option or via the `--json-file` or `--yaml-file` options.

`--maximum-servers "number"`
 specifies the maximum number of reusable batch servers that are allowed to run for each combination of user and context.

- If this value is not specified, reusable batch servers are added as needed.
- If the context is not for a reusable batch server, this option is ignored.

For more information, see [“Reusable Batch Servers” in SAS Viya Platform: Programming Run-Time Servers](#).

A valid alias for this option is `--max-servers`.

This option is available as of version 1.9.0 of the batch CLI. The reusable batch server feature is available as of version 1.25.0 of the batch service.

`--minimum-servers "number"`
 specifies the minimum number of multi-job servers that must be running.

- If this value is not specified, all reusable batch servers are started as jobs are submitted.
- If this value is specified, a service account must be specified by using the `--run-server-as` option.
- If the context is not for a reusable batch server, this option is ignored.

For more information, see [“Reusable Batch Servers” in SAS Viya Platform: Programming Run-Time Servers](#).

A valid alias for this option is `--min-servers`.

This option is available as of version 1.9.0 of the batch CLI. The reusable batch server feature is available as of version 1.25.0 of the batch service.

`--name context_name`
 specifies the name of the batch context. A valid alias for this option is `--n`.

`--queue-name queue_name`
 specifies the name of the SAS Workload Orchestrator queue to use when submitting this batch context. A valid alias for this option is `--queue`.

`--reusable-server`
 specifies whether the context uses reusable batch servers. For more information, see [“Reusable Batch Servers” in SAS Viya Platform: Programming Run-Time Servers](#).

If this flag is not specified, a batch server is created for each job.

A valid alias for this option is `--reusable`.

This option is available as of version 1.9.0 of the batch CLI. The reusable batch server feature is available as of version 1.25.0 of the batch service.

`--run-server-as user`

specifies the shared service account user that jobs using this context will run under. For information about shared service accounts, see [“SAS Batch Servers Running under a Shared Account” in SAS Viya Platform: Programming Run-Time Servers](#).

Valid aliases for this option are `--run-as` and `--service-account`.

`--sas-options options`

specifies SAS options to add to SAS batch jobs when using this batch context. To specify multiple options, you must enclose the options within quotation marks and separate each option by a space. For example: `--sas-options "option-1 option-2"`.

Examples of Commands to Update Contexts

Here is an example of a command to use a JSON file to update a context that is specified by name:

```
sas-viya batch contexts update --json-file /path/to/JSON/file/containing/Context.json --launcher-context-name "My Launcher Name"
```

Here is an example of a command to use a YAML file to update a context that is specified by an ID:

```
sas-viya batch contexts update --yaml-file /path/to/YAML/file/containing/Context.yaml --launcher-context-id "05e5c6bc-ff5d-49af-bb51-1108a214499c"
```

Delete a Context

Syntax for the Commands to Delete a Context

Here is the command syntax to delete a batch context.

```
sas-viya batch contexts delete
--id object_id | --name context_name | --uri uri
[--force]
```

`--id object_id | --name context_name | --uri uri`

specifies the context to delete by specifying either the ID, the name, or the URI of the context. Valid aliases for these options are `--i`, `--n`, and `--u`.

`--force`

specifies that the context is deleted without first displaying a confirmation message. A valid alias for this option is `--f`.

Examples of Commands to Delete a Context

Here is an example of a command to delete a context by specifying the context name:

```
sas-viya batch contexts delete --name "My Batch Context Name"
```

Here is an example of a command to delete a context, specified by an ID, without first displaying a confirmation message:

```
sas-viya batch contexts delete --id 05e5c6bc-ff5d-49af-bb51-1108a214499c --force
```

JSON and YAML File Formats

Formats to Use When Creating a Context

Here is the format of the JSON file to use for creating a context:

```
{
  "name": "context_name"
  "launcherContextId": "ID_of_launcher_context"
  "launcherContextName": "name_of_launcher_context"
  "queueName": "name_of_queue"
  "description": "description"
  "sasOptions": "SAS_command_line_options"
}
```

Here is the format of the YAML file to use for creating a context:

```
name: "context-name"
launcherContextId: "ID_of_launcher_context"
launcherContextName: "name_of_launcher_context"
queueName: "name_of_queue"
description: "description"
sasOptions: "SAS_command_line_options"
```

Formats to Use When Updating a Context

Here is the format of the JSON file to use for updating a context:

```
{
  "name": "context_name"
  "launcherContextId": "ID_of_launcher_context"
  "queueName": "name_of_queue"
}
```

```

    "description": "description"
    "sasOptions": "SAS_command_line_options"
  }

```

Here is the format of the YAML file to use for updating a context:

```

name: "context-name"
launcherContextId: "ID_of_launcher_context"
queueName: "name_of_queue"
description: "description"
sasOptions: "SAS_command_line_options"

```

Manage File Sets

About This Command

Use the `sas-viya batch filesets` command to manage batch plug-in file sets. A file set is created each time that you submit a command, program, or interactive SAS session. It contains the program or command and any additional files that are needed and that you have specified. It serves as a local holding area for the job in the job directory in the execution pod.

An alternative is to use a mounted file system. See [“Input and Output File Storage” in SAS Viya Platform: Programming Run-Time Servers](#).

Note: You can run this command only if you are a SAS administrator.

List File Sets

Syntax for Commands to List File Sets

Here is the command syntax to list file sets:

```

sas-viya batch filesets list
[--filter filter]
[--ids-only]
[--limit maximum_number]
[--sort-by field_name-1{, field_name-2,...field_name-n}]
[--start record_number]

```

`--filter filter`

lists a subset of the returned objects that match the conditions in the specified filter. For more information about filtering, see <https://developer.sas.com/rest-apis/>.

`--ids-only`

specifies that only the object IDs are returned. Valid aliases for this option are `--idsonly` and `--i`.

`--limit maximum_number`

specifies the maximum number of responses to return at a time. The default is to return an unlimited number of results.

`--sort-by field_name-1{, field_name-2,...field_name-n}`

specifies one or more fields by which the results are sorted.

Separate multiple values of *field_name* with commas. By default, the results are sorted in ascending order. Place a `~` character before a *field_name* value in order to sort the field in descending order.

Here are the valid values for *field_name*:

- `id`
- `contextId`
- `creationTimeStamp`
- `modifiedTimeStamp`
- `createdBy`
- `modifiedBy`

A valid alias for this option is `--sortby`.

`--start record_number`

specifies the record number to start at when returning results. If you do not specify this option, the results are returned starting at the first record.

Examples of Commands to List File Sets

Here is an example of a command to list file sets and sort by the file set ID:

```
sas-viya batch filesets list --sort-by id
```

Here is the output of the command:

ID	Context ID	Created By	Creation Time	Modified By	Modified Time
JOB_20200817_191949_741_1	4811b40e-f432-410e-afb5-76b96828066e	myuser	2020-08-17T15:19:49	myuser	
2020-08-17T15:19:49					
JOB_20200817_192913_287_1	4811b40e-f432-410e-afb5-76b96828066e	myuser	2020-08-17T15:29:13	myuser	
2020-08-17T15:29:13					

Here is an example of a command to list the file sets that match a specified filter. The filter returns file sets whose ID is equal to the supplied string.

```
sas-viya batch filesets list --filter "eq(id,'JOB_20200817_191949_741_1')"
```

Here is the output of that command:

ID	Context ID	Created By	Creation Time	Modified By	Modified Time
JOB_20200817_191949_741_1 2020-08-17T15:19:49	4811b40e-f432-410e-afb5-76b96828066e	myuser	2020-08-17T15:19:49	myuser	

List Files in a File Set

Syntax for Commands to List Files in a File Set

Here is the syntax to list the files in a file set.

```
sas-viya batch filesets list-files
--id object_id | --uri uri
```

--id *object_id* | --uri *uri* specifies the file set whose files you want to list by specifying either the ID or the URI of the file set. Valid aliases for these options are --i and --u.

Examples of Commands to List Files in a File Set

Here is an example of a command to list the files in a file set that is specified by the file set's ID:

```
sas-viya batch filesets list-files --id JOB_20200817_191949_741_1
```

Here is the output of the command:

Name	Size	File Set ID	Creation Time	Modified Time	Created By	Modified By
mypgm.sas	23	592b0ff4-8dd5-4d14-9a13-929a07544f88	myuser	2020-08-17T15:19:50	myuser	2020-08-17T15:19:50

Here is an example of a command to list the files in a file set that is specified by the file set's URI:

```
sas-viya batch filesets list-files --uri "/batch/filesets/05e5c6bc-ff5d-49af-
bb51-1108a214499c"
```

Delete a File Set

Syntax for a Command to Delete a File Set

Here is the syntax to delete a file set:

```
sas-viya batch filesets delete
--id object_id | --uri uri
```

`--force`

`--id object_id | --uri uri`
specifies the file set to delete by specifying either the ID or the URI of the file set. Valid aliases for these options are `--i` and `--u`.

`--force`
specifies that the file set is deleted without first displaying a confirmation message. A valid alias for this option is `--f`.

Examples of Commands to Delete a File Set

Here is an example of a command to delete a file set by specifying the file set's URI:

```
sas-viya batch filesets delete --uri "/batch/filesets/05e5c6bc-ff5d-49af-  
bb51-1108a214499c"
```

Here is an example of a command to delete a file set, specified by an ID, without first displaying a confirmation message:

```
sas-viya batch filesets delete --id JOB_20200817_191949_741_1 --force
```

Reference

<i>Available Default Contexts</i>	43
<i>Optional Environment Variables for Options</i>	44
<i>Output Specifications and File Locations</i>	44
<i>Runtime Variables</i>	46

Available Default Contexts

Two contexts are created by default:

`default`

used for running SAS programs via the `sas-viya batch job` subcommands `submit-pgm` or `run-saslm`. See [“Submit SAS Programs in Batch”](#) and [“Run a SAS Session Interactively”](#).

`default-cmd`

used for running arbitrary commands such as Python and R programs via the `sas-viya batch job` subcommand `submit-cmd`. See [“Run Commands in Batch”](#).

Note: This context is available as of release 2021.2.6.

This context starts a pod with the SAS Configurator for Open Source persistent volume claim (PVC) mounted at `/opt/sas/viya/home/sas-pyconfig`.

Note: This context is always available. However, if SAS Configurator for Open Source has not created its PVC, the context cannot mount the PVC. Be sure to deploy SAS Configurator for Open Source. For details, see [“Configuring the SAS Viya Platform for Use with Python, R, and Other Open Source Languages”](#).

Optional Environment Variables for Options

You can choose to set environment variables instead of specifying an option each time you issue a CLI command.

For example, you can set the `SAS_BATCH_CLI_CONTEXT` environment variable to specify the context name to use with the `sas-viya batch jobs` command.

To see available environment variables, view the integrated help for the batch CLI:

```
sas-viya batch --help
```

TIP In the integrated help, environment variables are enclosed in square brackets and prefixed by a '\$'.

Output Specifications and File Locations

When specifying a reusable batch server, you have three options that affect where the output files are located. The location varies depending on which output options you use and whether you specify a file name or full path. The following table shows how the three output options can control the location of output files. You can mix and match how you use the options to meet your needs.

--rem-output-dir	--rem-log-path	--rem-list-path
The output directory option is not specified.	The --rem-log-path option is not specified: ■ The log file is put into the file set. ■ The file uses its default name.	The --rem-list-path option is not specified: ■ The list file is put into the file set. ■ The file uses its default name.
	The --rem-log-path option specifies the file name only: ■ The log file is put into the file set.	The --rem-list-path option specifies the file name only: ■ The list file is put into the file set.

--rem-output-dir	--rem-log-path	--rem-list-path
	■ The file uses its specified name.	■ The file uses its specified name.
	The <code>--rem-log-path</code> option specifies the file name and full path: ■ The log file is put in the full path location that is mounted in the pod. ■ The file uses its specified name.	The <code>--rem-list-path</code> option specifies the file name and full path: ■ The list file is put in the full path location that is mounted in the pod. ■ The file uses its specified name.
The output directory option is specified.	The <code>--rem-log-path</code> option is not specified: ■ The log file is put in the output directory that is mounted in the pod. ■ The file uses its default name.	The <code>--rem-list-path</code> option is not specified: ■ The list file is put in the output directory that is mounted in the pod. ■ The file uses its default name.
	The <code>--rem-log-path</code> option specifies the file name only: ■ The log file is put in the output directory that is mounted in the pod. ■ The file uses its specified name.	The <code>--rem-list-path</code> option specifies the file name only: ■ The list file is put in the output directory that is mounted in the pod. ■ The file uses its specified name.
	The <code>--rem-log-path</code> option specifies the file name and full path: ■ The log file is put in the full path location that is mounted in the pod. ■ The file uses its specified name. ■ The output directory is ignored.	The <code>--rem-list-path</code> option specifies the file name and full path: ■ The list file is put in the full path location that is mounted in the pod. ■ The file uses its specified name. ■ The output directory is ignored.

Runtime Variables

Replaceable parameters are available for use in the output path or file name. These parameters are replaced with runtime values.

- `@STARTTIME@` provides the start-processing time of the batch job in 'YYYY-MM-DD-hh-mm-ss' format. The time is local to the processing host. For an example of its use, see [“Example: Specify a Location That Is Mounted in a Pod”](#) on page 15.
- `@FILESETID@` provides the ID of the file set.
- `@SASPGMNAME@` provides the name of program without the .sas file extension. This parameter is for SAS Batch only.
- `@USERNAME@` provides the value of the USER environment variable at the time of script execution. If this parameter is not set at execution time, the value of the USER environment variable is set to `uid<users-id-value>` where *users-id-value* is an operating system ID value that is associated with the user.