

**Frequently Asked Questions:**  
ActEV19 SDL  
Validation Phase Processing

Validation steps are colored in *blue* (see Q0001 to have more information about these steps). CLI command are colored in *green*.

**Q001: How does the evaluation process work?**

First, you need to validate your system by submitting a URL pointing to your system on the web platform (<https://actev.nist.gov>).

An automatic process will be in charge of validating your system. It consists of multiple steps to check if your system is downloadable, installable on a fresh instance and runnable on a validation set. Here is a non exhaustive list of the steps:

- **actev\_get\_system**: call **actev get-system** using your URL as well as your credentials to download your system.
- **actev\_validate\_performer\_execution**: check that your container outputs are complete (see Q004 to learn more about container output).
- **actev\_system\_setup**: call **actev system-setup**.
- **actev\_validate\_system**: check that the structure of the CLI, the method signatures of the entry points as well as the structure of the container output have not been modified.
- **actev\_design\_chunks**: call **actev design-chunks**.
- **actev\_experiment\_init**: call **actev experiment-init**.
- **actev\_process\_chunks**: call **actev pre/post/process-chunk** and **reset-chunk** on all the chunks.
- **actev\_merge\_chunks**: call **actev merge-chunks**.
- **actev\_validate\_nist\_execution**: check that your container outputs are similar to the result of the automatic process to make sure that your system is reproducible.

You can check the progress of these steps via the web platform, by checking the status and the logs clicking on the web platform:

System Name:



The screenshot shows a user interface for a system named 'TEST1'. On the left, the system name 'TEST1' is displayed in bold, with 'Type: primary' below it. On the right, there are three buttons: 'Upload' (with an upload icon), 'View Submission (1175)' (in a blue box), and 'Scoring FAIL-5\_of\_17\_actev\_get\_system' (in a light blue box with a red border). A red arrow points from the left towards the 'Scoring' button.

Then, at the end of this process, NIST keeps your system to evaluate it on private data sets.

**Q002: My system is stuck at [actev\\_test\\_cmd](#) step but seems to be valid. What is going on?**

The issue actually comes from the next step: [actev\\_get\\_system](#). Credentials are possibly not correct, NIST is not able to get your system and may wait indefinitely for a manual assistance to get the right credentials. In this case, check if you are able to get your system in local using [actev get-system](#) and submit new credentials.

If the problem persists, contact [diva-nist@nist.gov](mailto:diva-nist@nist.gov).

**Q003: My submission is a zip file and I got a *Permission denied* error on a script that should be executable.**

Sometimes, tools to zip files do not support the permissions conservation and NIST is not able to execute them. If such an error occurred, the best solution is to convert your zip archive into a tar archive.

**Q004: My system doesn't pass the [actev\\_validate\\_performer\\_execution](#) step. How can I resolve this?**

You need to include your container output, which is a trace of the local execution of your system on a validation set, so that NIST can compare their results with yours to make sure that your system is reproducible.

A container output is a directory contained in *diva\_evaluation\_cli/container\_output* and its name is composed of: <validation set name>\_<team name>.

(e.g. ActEV-Eval-CLI-Validation-Set1\_Nist)

Reference container outputs are available, they represent validation sets and are named by <validation set name>. You should use them to validate your system.

A container output directory should contain:

- *output.json* - your output on the validation set used (generated by your system)
- *chunk.json* - the chunk file designed using the validation set (generated by your system)
- *file.json* - the file index used (copied pasted from a reference container output)
- *activity.json* - the activity index used (copied pasted from a reference container output)

**Q005: My system is stuck at [actev\\_validate\\_performer\\_execution](#) step, but seems to be valid. What is the issue?**

The issue comes from the next step: [actev\\_system\\_setup](#). Two possibilities:

- the command `actev system-setup` is waiting for a manual assistance that cannot occur since we use an automatic process: make sure that your script is able to run without any manual actions.
- the command “`actev system-setup`” is constantly restarting the instance, because one of your conditions is not able to stop the setup when an issue occurred:  
Make sure that your system is able to stop and return an error if any issue occurred.

**Q006: My system fails at `actev_process_chunk` step but works fine on my side.**

Make sure you install all the dependencies using the `actev system-setup`, even python packages.

**Q007: My system doesn't pass `actev_validate_nist_execution` step. What is going on?**

Two possibilities:

- the most likely is that something went wrong in the previous steps and did not return any error. The error usually comes from: `actev design-chunks`, `actev experiment-init` or `actev process-chunk`. Send an email to [diva-nist@nist.gov](mailto:diva-nist@nist.gov) if you want the logs to investigate on the issue.
- Your system is not reproducible and output from the NIST execution is too different from yours.