Nitro Release Notes

The Nitro release notes contains the following sections:
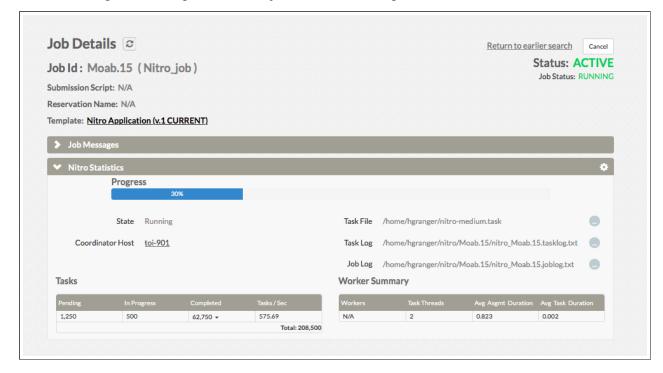
New Features

This section contains a summary of key new features in Nitro.

2.0.0

Nitro Web Services

Nitro Web Services is a new, optional, component that lets you view Nitro job status information in Viewpoint.

The following is an example of a Nitro job shown in Viewpoint.

The information that Nitro Web Services reports to Viewpoint includes the current job's state, task progress (counts of tasks completed, in-progress, failed, timed out, insufficient resources, invalid, etc), task and job log file locations, worker count, average task duration, and average node cpu load.

Added Support for Task Files with Spaces

Nitro now supports two task file environment variables (NITRO_TASK_FILE and NITRO_LONG_TASK_FILE).

- NITRO_TASK_FILE can be used with file names that do not use spaces; but MUST be used if the user submits more than one task file.

  For example:

  ```
  export NITRO_TASK_FILE="myfile.txt"
  export NITRO_TASK_FILE="myfile1.txt myfile2.txt"
  ```

- NITRO_LONG_TASK_FILE can be used with file names that do not contain spaces; but MUST be used if the file name contains spaces. This variable can only contain one file name. You cannot submit multiple file names containing spaces. This is the default variable when using the Viewpoint Nitro template.

  For example:

  ```
  export NITRO_LONG_TASK_FILE="my file name containing spaces.txt"
  ```

Both environment variables can be defined at the same time and all included files will be submitted to Nitro.

Nitro Coordinator Threads

Nitro supports the new configuration options of 'io-threads' and 'pool-threads'.

- io-threads sets the number of threads available to Nitro's message bus to process incoming and outgoing messages.

- pool-threads configures how many threads Nitro uses to process requests to the coordinator from workers. This value is capped at six threads.

Differences

This section contains differences in previously existing features that require a change in configuration or routine.

2.0.1

Nitro Web Services

- Nitro Web Services was refactored to build using pyinstaller and rpmbuild (instead of rpmvenv).

- Nitro Web Services tarball installation is now performed by an 'install' script. The install script allows the installer to designate an alternate install directory (defaults to /opt).

2.0.0

Changed --hostname Command Line Option to --name

The --hostname command line option has been renamed to --name to reduce confusion. This command line option can be used on both the coordinator and the workers. For workers, this is the name the worker uses to reference itself when communicating with the coordinator. This option has no effect on a coordinator.

Added --key-file Command Line Option

Using the --key-file <*file*> creates a passphrase that can be used to authenticate workers to a coordinator. If the file contains newline or tab characters, these will be removed from the passphrase.

Added InsufRes Category to Job Stats and Worker List

The following changes were made to the job stats output and worker list:

- Added InsufRes column to worker list in job log to show which workers are not able to fulfill resource requirements for tasks with cores or memory specifications.

- Added InsufRes row to total task stats in job log.

- Added InsufRes row to task stats reported on stdout when coordinator exits.

- Added InsufRes element to json output through zeromq publish channel.

Added stdout Capture Limit to Config File

Added "task-output-limit" to the nitro.cfg file. This value allows the administrator to set the number of bytes that will be captured by stdout and stderr (each) and written to the task log file. Default is 512 bytes. Nitro captures the stdout and stderr strings output by tasks, but only retains the last n characters as set by the limit to reduce communication and storage overhead.

Installation and Upgrade Information

This section identifies information when installing and upgrading.

In this section:

Licensing Requirement

Beginning with Nitro 2.0, the licensing procedure changed to use an RLM server. You can use an existing RLM server or you can install a new one. Instructions for installing an RLM server and instructions how to obtain and install the Nitro license is available in the System Administrator Guide.

Nitro Installation Bundles

The method to obtain the download files for Nitro has been revised to support the license requirement and the new Nitro Web Services component.

Beginning with Nitro 2.0, the nitro-releated files are bundled together.

- For the manual installation/upgrade method, the Nitro, Nitro Web Services, and the RLM tarball files are packaged together in a Nitro Tarball Bundle.

- For the RPM installation/upgrade method, the Nitro, Nitro Web Services. and the RLM RPM files are packaged together in a Nitro RPM Bundle.

Both bundles are available from the Nitro Download page. The Installation and Upgrade information in the System Administrator Guide has been enhanced to include a prepare the host topic and specific topics for a manual versus an RPM installation.

Known Issues

This section identifies known issues. Known issues are aggregated and grouped by the release version for which they first occurred or where reported. When applicable, each issue description has the associated issue number in parentheses.

2.0.0

No known issues.

Resolved Issues

This section lists issues/key bugs that are fixed. Resolved issues are aggregated and grouped by the release version in which they were resolved. When applicable, each resolved issue has the associated issue number in parentheses.

2.0.0

Duplicate names in workers file cause the coordinator to expect a worker of the same name. This has been fixed to remove duplicate worker node entries and use the first node as the coordinator. (NITRO-223)

Nitrostat failed to handle some special characters in job output. If the contents of stdout or stderr contain tab or newline characters, these characters are now replaced with a space character. (NITRO-221)