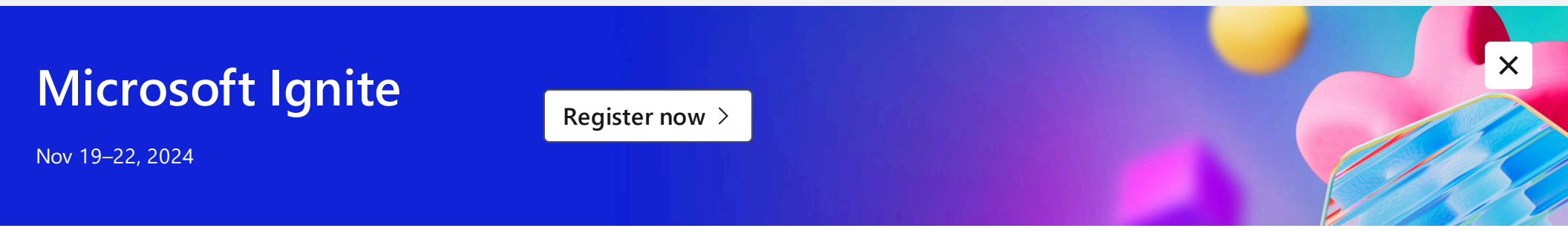
We use optional cookies to improve your experience on our websites, such as through social media connections, and to display personalized advertising based on your online activity. If you reject optional cookies, only cookies necessary to provide you the services will be used. You may change your selection by clicking "Manage Cookies" at the bottom of the page. Privacy Statement Third-Party Cookies

Accept Reject Manage cookies

Microsoft Ignite

Nov 19–22, 2024

Register now



mpiexec

Article • 02/09/2022 • 6 contributors

Feedback

In this article

Starts one or more Message Passing Interface (MPI) applications on an HPC cluster.

For examples of how to use this command, see Examples.

Syntax

Copy

```
mpiexec [/genv <env_var_name> <env_var_value>]
[/gpath <path1>[;<path2>...]] [/gwdir <working_dir>] [/ghost <host_name>]
[/gmachinefile <file_name>] [/genvlist <env_var1>[,<env_var2>...]]
[/cores <num>] [/affinity] [/affinity_layout <algorithm>[:<target>]] [/exitcodes
[/port <port>] [/timeout <seconds>] [/job <string>] [/lines]
[/t [(<filter1>,<filter2>...)]] [/tracefile <file_name>]
[/tracemax <max_size>] [/debug [{0|1|2|3}]] [/logfile <log_file_name>] [/np {<nu
[/machinefile <file_name>] [/host <host_name>]
[/hosts <num> <host1> [num1] <host2> [num2] ... <hostn> [numn]]
[/wdir <working_dir>] [/env <env_var_name> <env_var_value>]
[/path <path1>[;<path2>...]] <application1> [<application1_paramteters>]
[: [<section2_parameters>] <application2> [<application2_parameters>]
[: [<section3_ parameters>] <application3> [<application3_parameters>]...]]

mpiexec /configfile <file_name>

mpiexec {/? | /help | /?? | /help2 | /??? | /help3}
```

Parameters

Expand table

Parameter	Description

Download PDF

/genv <env_var_name> <env_var_value>	Sets an environment variable to the specified value for the applications in sections of the **mpiexec** command. If you specify the value of an environm variable for a specific section by using the **/env** parameter and globally by using the **/genv** parameter, the setting that the **/env** parameter specifies f the section overrides the setting that the **/genv** parameter specifies. To specify the values of multiple environment variables, specify the **/genv** parameter multiple times, once for each environment variable. For a list of built-in environment variables that you can set with the **/genv** parameter, see MPICH environment variables.
/gpath <path1>[;<path2>...]	Specifies one or more paths to search on the target node for the applicatic in all sections of the **mpiexec** command. To specify multiple paths, separat paths with a semicolon. If the paths include spaces, enclose the paths in quotation marks ("). The value you specify for this parameter does not rep and is not appended to the PATH environment variable. If you specify the search path for a specific section by using the **/path** parameter and globally by using the **/gpath** parameter, the setting that th **/path** parameter specifies for the section overrides the setting that the **/gp** parameter specifies.
/gwdir <working_dir>	Specifies the working directory to use for the applications in all sections of **mpiexec** command. This directory can have a local or a remote path. You c also specify this parameter as **/gdir**. If you specify the working directory for a specific section by using the **/wd** parameter and globally by using the **/gwdir** parameter, the setting that th **/wdir** parameter specifies for the section overrides the setting that the **/gw** parameter specifies.
/ghost <host_name>	Starts the applications in all sections of the **mpiexec** command on the spe node. If you also specify the **/np *** parameter or do not specify the **/np** parameter, the application uses one core on that node. If the task that runs the **mpiexec** command is not allocated to the node th the **/ghost** parameter specifies, the command fails. Specify the resources a nodes for the task such that you ensure that the task is allocated to the n If you specify the **/ghosts**, **/gmachinefile**, or **/np** parameter with a numeri value, you cannot also specify the **/ghost** parameter. If you specify the host for a specific section of the **mpiexec**command by us the **/host** parameter and globally by using the **/ghost** parameter, the setti that the **/host** parameter specifies for the section overrides the setting tha **/ghost** parameter specifies.
/gmachinefile <file_name>	Reads from the specified file a list of nodes on which to run the applicatior all sections of the **mpiexec** command. The format for the file is one node r per line, optionally followed by the number of cores on which the applicati allowed to run on that node. You can designate lines or parts of lines in the file as comments by includi number sign (#) at the start of the comment. The comment then extends t end of the line. Empty lines in the file are ignored. The file must exist on each node that is allocated to the task that runs the **mpiexec** command. If the task that runs the **mpiexec** command is not allocated to one of the nodes listed in the file that the **/gmachinefile** parameter specifies, the command fails. Specify the resources and nodes for the task such that you ensure that the task is allocated to all of the nodes in the file. If you also specify the **/np *** option or do not specify the **/np** option, the application uses the sum of the number of cores that are specified for all c nodes in the file. If you specify a value for the **/np** option that is less than sum of the number of cores that are specified for all of the nodes in the fil the application runs only the total number of cores that the **/np** option specifies. Cores on the nodes are used in the order in which the file specifi

	the nodes. If you specify the **/ghost** parameter, you cannot also specify the **/gmachin** parameter. If you specify the file that lists the nodes on which to run the application f specific section of the **mpiexec** command by using the **/machinefile** paran and globally for the applications in all sections of the **mpiexec** command b using the **/gmachinefile** parameter, the setting that the **/machinefile** parameter specifies for the section overrides the setting that the **/gmachinefile** parameter specifies.
/configfile <file_name>	Reads the remaining parameters for the **mpiexec** command line from the specified file. The lines in the file are command-line sections in the followin format: [<section_parameters>] <application> [<application_parameters>] You can include a command in the file that spans multiple lines by ending line of the command except the last line with a backslash (\). You can desig lines or parts of lines in the file as comments by including a number sign (# the start of the comment. The comment then extends to end of the line. Empty lines in the file are ignored. If you run the **mpiexec** command as a task in a job, the file must exist on node that is allocated to the task. If you run the **mpiexec** command by us the **clusrun** command, the file must exist on each node on which **clusrun** **mpiexec**.
/np {<num>|*}	Specifies the number of processes on which to run the application that is specified in the same section of the **mpiexec** command as the **/np** parame If you specify an asterisk (*), the MPI application uses all available cores un more than one section is specified, in which case the MPI application uses remaining available cores. The absence of the **/np** parameter is equivalent **/np ***. You can also specify this parameter as **/n**. If you specify the **/hosts** parameter, you cannot also specify the **/np** or **/n** parameter.
/machinefile <file_name>	Reads from the specified file a list of nodes on which to run the applicatio that is specified in the same section of the **mpiexec** command as the **/machinefile** parameter. The format for the file is one node name per line, optionally followed by the number of cores on which the application is allc to run on that node. You can designate lines or parts of lines in the file as comments by includi number sign (#) at the start of the comment. The comment then extends end of the line. Empty lines in the file are ignored. The file must exist on each node that is allocated to the task that runs the **mpiexec** command. If the task that runs the **mpiexec** command is not allocated to one of the nodes listed in the file that the **/machinefile** parameter specifies, the comr fails. Specify the resources and nodes for the task such that you ensure th the task is allocated to all of the nodes in the file. If you also specify the **/np *** option or do not specify the **/np** option, the application uses the sum of the number of cores that are specified for all c nodes in the file. If you specify a value for the **/np** option that is less than sum of the number of cores that are specified for all of the nodes in the fil the application runs only the total number of cores that the **/np** option specifies. Cores on the nodes are used in the order in which the file specifi the nodes. If you specify the **/host** or **/hosts** parameter, you cannot also specify the **/machinefile** parameter. If you specify the file the lists the nodes on which to run the application fo

	specific section of the **mpiexec** command by using the **/machinefile** paran and globally for the applications in all sections of the **mpiexec** command b using the **/gmachinefile** parameter, the setting that the **/machinefile** parameter specifies for the section overrides the setting that the **/gmachinefile** parameter specifies.
/host <host_name>	Starts the application that is specified in the same section of the **mpiexec** command as the **/host** parameter on the specified node. If you also specify * parameter or do not specify the **/np** parameter, the application uses one on that node. If the task that runs the **mpiexec** command is not allocated to the node th the **/host** parameter specifies, the command fails. Specify the resources an nodes for the task such that you ensure that the task is allocated to the n If you specify the **/hosts**, **/machinefile**, or **/np** parameter with a numeric v you cannot also specify the **/host** parameter. If you specify the host for a specific section of the **mpiexec** command by u the **/host** parameter and globally by using the **/ghost** parameter, the setti that the **/host** parameter specifies for the section overrides the setting tha **/ghost** parameter specifies.
/hosts <num> <host1> [num1] <host2> [num2] ... <hostn> [numn]	Specifies the number of nodes on which you want to run the application th specified in the same section of the **mpiexec** command as the **/hosts** parameter, then lists the names of each node and optionally lists the num of processes to run on that node. If the task that runs the **mpiexec** command is not allocated to one of the nodes that the **/hosts** parameter specifies, the command fails. Specify the resources and nodes for the task such that you ensure that the task is allocated to all of the nodes in the list. If you specify the **/host** or **/machinefile** parameter, you cannot also specif **/hosts** parameter.
/cores <num>	Specifies the numbers of cores to use on each node. This option overrides number of cores that are specified for each node in the **/hosts** parameter the file that is specified by the **/machinefile** parameter. This override appli all sections of the **mpiexec** command. You can also specify this parameter as **/c**.
/affinity	Sets the affinity mask for each of the processes that the **mpiexec** comman starts to a single core. This setting should only be used if a job is using nodes exclusively. If jobs a sharing a node (not running exclusively), this affinity setting mechanism m assign the same affinity mask to all jobs on the node. If you want to share nodes, you must either set affinity explicitly, or allow the Job Scheduler to manage affinity. For more information, see job scheduler affinity settings For more information about affinity, see Performance Tuning a Windows H Cluster for Parallel Applications .
/affinity_layout	Specifies the algorithm used to distribute rank processes to the compute Specifying a different algorithm will change the cores that each rank proce runs on. Changing the target allows larger units of cores to be assigned p rank process. Target is typically used only for multithreaded applications. Following are the values for the algorithm: Disabled = 0 - Does not assign affinity to any process. Spread = 1 - Distributes the processes as far as possible (default). Sequential = 2 - Distributes the processes per core sequentially. Balanced = 3 - Distribute the processes over the available NUMA nodes.

	Following are the values for the target: L - Assigns each process to a logical core. P - Assigns each process to a physical core. N - Assigns each process to a NUMA node. This parameter was introduced in HPC Pack 2012 and is not supported in previous versions.
/wdir <working_dir>	Specifies the working directory to use for the application specified in the sa section of the **mpiexec** command as the **/wdir** parameter. This directory ca have a local or a remote path. You can also specify this parameter as **/dir**. If you specify the working directory for a specific section by using the **/wd** parameter and globally by using the **/gwdir** parameter, the setting that th **/wdir** parameter specifies for the section overrides the setting that the **/gw** parameter specifies.
/env <env_var_name> <env_var_value>	Sets an environment variable for the application that is specified in the sar section of the **mpiexec** command as the **/env** parameter. If you specify the value of an environment variable for a specific section by using the **/env** parameter and globally by using the **/genv** parameter, the setting that the **/env** parameter specifies for the section overrides the setting that the **/ger** parameter specifies. To specify the values of multiple environment variables, specify the **/env** parameter multiple times, once for each environment variable. For a list of built-in environment variables that you can set with the **/env** parameter, see MPICH environment variables.
/genvlist <env_var1>[, <env_var2>...]	Passes the values of the specified environment variables to the processes t **mpiexec** starts. The list is a comma-separated list of environment variable names. For a list of built-in environment variables that you can use with the **/genv** parameter, see MPICH environment variables.
/exitcodes	Prints the exit codes of the processes that **mpiexec** started as part of stan output at the end of the run.
/priority {0|1|2|3|4}	Sets the priority for the processes that **mpiexec** starts. The default priority Following are the possible values: 0 - Idle 1 - Below normal 2 - Normal 3 - Above normal 4 - High
/port <port>	Specifies the port on which the smpd.exe process listens. You can also spec this parameter as **/p**.
/path <path1>[;<path2>...]	Specifies one or more paths to search on the target node for the applicatio that is specified in the same section of the **mpiexec** command as the **/path** parameter. To specify multiple paths, separate paths with a semicolon. If th paths include spaces, enclose the paths in quotation marks ("). The value t you specify for this parameter does not replace and is not appended to th PATH environment variable. If you specify the search path for a specific section by using the **/path** parameter and globally by using the **/gpath** parameter, the setting that th **/path** parameter specifies for the section overrides the setting that the **/gp** parameter specifies.

/timeout <seconds>	Sets the amount of time that job that runs the **mpiexec** command should before timing out, in seconds.
/job <string>	Associates an MPI job with a job that is created by the Windows HPC Job Scheduler Service. The string is passed to **mpiexec** by the HPC Node Mana Service.
/lines	Prefixes each line in the output of the **mpiexec** command with the rank of process that generated the line. You can also specify this parameter as **/l**.
/trace [(<filter1>,<filter2>...)]	Traces Microsoft-Message Passing Interface (MS-MPI) events for the application. If you do not specify any filters, the **mpiexec** command traces MS-MPI events for the application. Optionally, you can specify trace filters enable only trace events of interest. List the event filter names, or their equivalent hexadecimal values, by using a comma-separated list enclosed parentheses. **Important:** While this type of tracing can be useful for a very simple test application, in most practical applications, this method generat such a large trace file that it is not very useful. To control tracing and creat manageable trace files, use an Event Tracing for Windows (ETW) tool such Xperf or Logman to turn on tracing only for the part you need. By default, trace logs are written to the directory for the user profile on ea node. Use the **/tracefile** parameter to specify an alternative trace file. Following are the event filter names and equivalent hexadecimal values tha you can include in the list of filters: all (0xffffffff) - All API and communication events api (0x00007fff) - All API events pt2pt (0x00000001) - Point-to-point APIs poll (0x00000002) - Point-to-point polling APIs, such as MPI_Iprobe and MPI_TestXXX coll (0x00000004) - Collective APIs rma (0x00000008) - One-sided APIs comm (0x00000010) - Communication APIs errh (0x00000020) - Error handler APIs group (0x00000040) - Group APIs attr (0x00000080) - Attribute APIs dtype (0x00000100) - Data type APIs io (0x00000200) - Input/output APIs topo (0x00000400) - Topology APIs spawn (0x00000800) - Dynamic process APIs init (0x00001000) - Initialization APIs info (0x00002000) - Information APIs misc (0x00004000) - Miscellaneous APIs interconn (0x000f8000) - All interconnectivity communication icsock (0x00008000) - Socket interconnectivity communication icshm (0x00010000) - Shared memory interconnectivity communication icnd (0x00020000) - NetworkDirect interconnectivity communication You can also specify this parameter as **/t**.

	This parameter is deprecated as of HPC Pack 2008 R2 with Service Pack 3 (SP3).
/tracefile <file_name>	Specifies the name of file to use for the trace log, including the path. The default file is %USERPROFILE%\mpi_trace_*job_identifier.task_identifier.subtask_identifi*. You can also specify this parameter as **/tf**. This parameter is deprecated as of HPC Pack 2008 R2 with Service Pack 3 (SP3).
/tracemax <max_size>	Specifies the maximum size of the trace log file, in megabytes. You must ha at least *max_size* megabytes of free space available on the drive that you specify for the trace file. The binary tracing data is written by using a circular buffer, so when the da exceeds the maximum size of the file, the tracing data is overwritten startin the beginning of the file. As a result, the tracing log file always contains th most recent *max_size* megabytes of tracing data from the MPI job. Each binary record in the trace log file has a time stamp, so that log file vie can display the information in chronological order regardless of the wrapp cause when the tracing data is overwritten. The default value for this parameter is **10240**. Specify **0** to allow the creatic a trace log file of unrestricted size. You can also specify this parameter as **/tm**. This parameter was introduced in HPC Pack 2008 R2. It is deprecated as o HPC Pack 2008 R2 with Service Pack 3 (SP3).
/debug [{0|1|2|3}]	Prints the specified level of debugging information to standard error. If yo specify the **/debug** option without a value, **mpiexec** behaves as if you spe as value of **2**. Following are the possible debugging levels: 0 - No debugging information is displayed. 1 - Information about errors is displayed. 2 - Extensive debugging information is displayed. 3 - Both errors and extensive debugging information are displayed. You can also specify this parameter as **/d**.
/logfile <log_file_name>	Specifies the name of file to use for the log file, including the path.
<section*N*_parameters>	Specifies parameters that apply only to the section of the **mpiexec** comma in which they occur, and includes the following parameters: - **/env** - **/path** - **/wdir** - **/host** - **/hosts** - **/machinefile** - **/np** The Syntax section explicitly lists these parameters for the first section, anc uses the <section*N*_parameters> placeholder for the additional sections. F information about sections in an **mpiexec** command, see the description fo the <application*N*> [<application*N*_parameters>] parameter.
<application*N*> [<application*N*_parameters>]	Specifies the MPI applications that you want to start and any necessary parameters for those applications. You can specify more than one applicat by including multiple colon-separated sections in the **mpiexec** command. If you include multiple sections to specify multiple applications, the applica need to have compatible requirements for the ranks on which the applicat need to run and the roles that the applications serve on those ranks. For

	example, you can use two sections to specify a master process that needs run on rank 0 and a subordinate process that does not need to run on ra
/pwd <string>	Authenticates the user with the provided password. This option is only val when MS-MPI Launch Service is being used.
/saveCreds	Notifies the Launch Service to save credentials. This option is only valid wh /pwd is provided. After a successful invocation of /saveCreds, it is not necessary to provide t password with /pwd unless the password is changed.
/unicode	Switches **mpiexec** output to unicode stream. This only affects the output c **mpiexec**. Unicode path and executables are supported with or without thi option.
/?	Displays Help for commonly used parameters of the **mpiexec** command at command prompt. You can also specify this parameter as **/help**.
/??	Displays Help descriptions for all parameters of the **mpiexec** command an examples of the **mpiexec** command at the command prompt. You can also specify this parameter as **/help2**.
/???	Displays Help for the environment variables that you can use with the **mp** command at the command prompt. You can also specify this parameter as **/help3**.

Remarks

In most cases, you should run the **mpiexec** command by specifying it in a task for a job. You can run **mpiexec** directly at a command prompt if the application requires only a single node and you run it on the local computer, instead of specifying nodes with the **/host**, **/hosts**, or **/machinefile** parameters.

If you run the **mpiexec** command by using the **clusrun** command, the **mpiexec** command behaves as if you specified the command locally at the command prompt on each node on which **clusrun** runs the **mpiexec** command. In this case, you again can only run the **mpiexec** command if the application requires only a single node, and only if you do not specify other nodes by using the parameters of the **mpiexec** command.

Examples

To run four application1 processes on the local host with four cores, use either of the following commands:

Copy

```
mpiexec application1
```

Copy

```
mpiexec /np * application1
```

To run one master process and three subordinate processes on the local host with four cores, use the following command, in which the absence of **/np** from the second section is equivalent to **/np** *:

Copy

```
mpiexec /np 1 master : subordinate
```

To run one master process and 31 subordinate processes on the hosts listed in a myhosts.txt file that lists four hosts with eight cores each, use the following command:

Copy

```
mpiexec /gmachinefile myhosts.txt /np 1 master : subordinate
```

To trace the point-to-point and collective MPI calls that the two application1 processes are making and the interconnectivity communication that is generated by MPI, and limit the maximum size of the tracing file to 20,480 megabytes, use the following command:

Copy

```
mpiexec /trace (pt2pt,coll,interconn) /tracemax 20480 /np 2 application1
```

To run the application1 process on the hosts that are listed in a myhosts.txt file and include a connectivity table in the standard output, use the following command:

Copy

```
mpiexec /machinefile myhosts.txt /env MPICH_CONNECTIVITY_TABLE 1 application1
```

Additional references

- mpicsync
- etl2clog
- etl2otf
- clusrun
- Command-Line Syntax Key

Additional resources

Training

Module
Optimizing tightly coupled HPC applications on HBv2, HC, and HB series virtual machines - Training
Learn how to build, run, and tune tightly coupled HPC applications in Azure.

English (United States) Your Privacy Choices Theme

Manage cookies Previous Versions Blog Contribute Privacy Terms of Use Trademarks © Microsoft 2024