


Version

▼

 Filter by title

- Microsoft HPC Pack 2019
- > Planning and evaluation
- > Get started
- > Tutorials
- > Node deployment
- > Cluster management
- > Job management
- ▼ HPC Pack commands
 - HPC Pack commands
 - > cluscfg
 - clusrun
 - etl2clog
 - etl2otf
 - hpcazurelog
 - > hpccred
 - > hpcfile
 - hpcfutil
 - > hpcpack
 - hpcsync
 - hpctrace
 - > job
 - > jobtemplate
 - ▼ mpiexec
 - mpiexec**
 - Environment variables
 - mpipingpong
 - mpicsync
 - > node
 - > task
 - > test
- > Reference

 Download PDF

[Learn](#) / [HPC Pack](#) / [HPC Pack 2019](#) /

⊕ ⋮

mpiexec

Article • 02/09/2022 • [6 contributors](#)

 [Feedback](#)

In this article

- [Syntax](#)
- [Parameters](#)
- [Remarks](#)
- [Examples](#)

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 {<num>}]
[/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
/genv <env_var_name> <env_var_value>	<p>Sets an environment variable to the specified value for the applications in all sections of the mpiexec command. 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 /genv parameter specifies.</p> <p>To specify the values of multiple environment variables, specify the /genv parameter multiple times, once for each environment variable.</p> <p>For a list of built-in environment variables that you can set with the /genv parameter, see MPICH environment variables.</p>
/gpath <path1>[;<path2>...]	Specifies one or more paths to search on the target node for the application in all sections of the mpiexec command. To specify multiple paths, separate

	<p>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.</p> <p>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 /g parameter specifies.</p>
/gwdir <working_dir>	<p>Specifies the working directory to use for the applications in all sections of mpexec command. This directory can have a local or a remote path. You c also specify this parameter as /gdir.</p> <p>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 /g parameter specifies.</p>
/ghost <host_name>	<p>Starts the applications in all sections of the mpexec 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.</p> <p>If the task that runs the mpexec 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</p> <p>If you specify the /ghosts, /gmachinefile, or /np parameter with a numeri value, you cannot also specify the /ghost parameter.</p> <p>If you specify the host for a specific section of the mpexeccommand 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.</p>
/gmachinefile <file_name>	<p>Reads from the specified file a list of nodes on which to run the applicatio all sections of the mpexec 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.</p> <p>You can designate lines or parts of lines in the file as comments by includir number sign (#) at the start of the comment. The comment then extends t end of the line. Empty lines in the file are ignored.</p> <p>The file must exist on each node that is allocated to the task that runs the mpexec command.</p> <p>If the task that runs the mpexec 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.</p> <p>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 specif the nodes.</p> <p>If you specify the /ghost parameter, you cannot also specify the /gmachin parameter.</p> <p>If you specify the file that lists the nodes on which to run the application fr specific section of the mpexec command by using the /machinefile paran and globally for the applications in all sections of the mpexec command b using the /gmachinefile parameter, the setting that the /machinefile parameter specifies for the section overrides the setting that the /gmachinefile parameter specifies.</p>
/configfile <file_name>	<p>Reads the remaining parameters for the mpexec command line from the specified file. The lines in the file are command-line sections in the followin</p>

	<p>format:</p> <p>[<section_parameters>] <application> [<application_parameters>]</p> <p>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 designate lines or parts of lines in the file as comments by including a number sign (#) at the start of the comment. The comment then extends to end of the line. Empty lines in the file are ignored.</p> <p>If you run the mpixec command as a task in a job, the file must exist on each node that is allocated to the task. If you run the mpixec command by using the clusrun command, the file must exist on each node on which clusrun runs mpixec.</p>
/np {<num> *}	<p>Specifies the number of processes on which to run the application that is specified in the same section of the mpixec command as the /np parameter. If you specify an asterisk (*), the MPI application uses all available cores unless more than one section is specified, in which case the MPI application uses remaining available cores. The absence of the /np parameter is equivalent to /np *.</p> <p>You can also specify this parameter as /n.</p> <p>If you specify the /hosts parameter, you cannot also specify the /np or /n parameter.</p>
/machinefile <file_name>	<p>Reads from the specified file a list of nodes on which to run the application that is specified in the same section of the mpixec 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 allocated to run on that node.</p> <p>You can designate lines or parts of lines in the file as comments by including a number sign (#) at the start of the comment. The comment then extends to the end of the line. Empty lines in the file are ignored.</p> <p>The file must exist on each node that is allocated to the task that runs the mpixec command.</p> <p>If the task that runs the mpixec command is not allocated to one of the nodes listed in the file that the /machinefile 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.</p> <p>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 of the nodes in the file. If you specify a value for the /np option that is less than the sum of the number of cores that are specified for all of the nodes in the file, 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 specifies the nodes.</p> <p>If you specify the /host or /hosts parameter, you cannot also specify the /machinefile parameter.</p> <p>If you specify the file that lists the nodes on which to run the application for a specific section of the mpixec command by using the /machinefile parameter and globally for the applications in all sections of the mpixec command by using the /gmachinefile parameter, the setting that the /machinefile parameter specifies for the section overrides the setting that the /gmachinefile parameter specifies.</p>
/host <host_name>	<p>Starts the application that is specified in the same section of the mpixec command as the /host parameter on the specified node. If you also specify the /np * parameter or do not specify the /np parameter, the application uses one core on that node.</p> <p>If the task that runs the mpixec command is not allocated to the node that the /host parameter specifies, the command fails. Specify the resources and nodes for the task such that you ensure that the task is allocated to the node.</p>

	<p>If you specify the /hosts, /machinefile, or /np parameter with a numeric value, you cannot also specify the /host parameter.</p> <p>If you specify the host for a specific section of the mpexec command by using the /host parameter and globally by using the /ghost parameter, the setting that the /host parameter specifies for the section overrides the setting that the /ghost parameter specifies.</p>
<code>/hosts <num> <host1> [num1] <host2> [num2] ... <hostn> [numn]</code>	<p>Specifies the number of nodes on which you want to run the application task specified in the same section of the mpexec command as the /hosts parameter, then lists the names of each node and optionally lists the number of processes to run on that node.</p> <p>If the task that runs the mpexec 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.</p> <p>If you specify the /host or /machinefile parameter, you cannot also specify the /hosts parameter.</p>
<code>/cores <num></code>	<p>Specifies the numbers of cores to use on each node. This option overrides the number of cores that are specified for each node in the /hosts parameter or the file that is specified by the /machinefile parameter. This override applies to all sections of the mpexec command.</p> <p>You can also specify this parameter as /c.</p>
<code>/affinity</code>	<p>Sets the affinity mask for each of the processes that the mpexec command starts to a single core.</p> <p>This setting should only be used if a job is using nodes exclusively. If jobs are sharing a node (not running exclusively), this affinity setting mechanism may 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.</p> <p>For more information about affinity, see Performance Tuning a Windows HPC Cluster for Parallel Applications.</p>
<code>/affinity_layout</code>	<p>Specifies the algorithm used to distribute rank processes to the compute nodes. Specifying a different algorithm will change the cores that each rank process runs on. Changing the target allows larger units of cores to be assigned per rank process.</p> <p>Target is typically used only for multithreaded applications.</p> <p>Following are the values for the algorithm:</p> <p>Disabled = 0 - Does not assign affinity to any process.</p> <p>Spread = 1 - Distributes the processes as far as possible (default).</p> <p>Sequential = 2 - Distributes the processes per core sequentially.</p> <p>Balanced = 3 - Distribute the processes over the available NUMA nodes.</p> <p>Following are the values for the target:</p> <p>L - Assigns each process to a logical core.</p> <p>P - Assigns each process to a physical core.</p> <p>N - Assigns each process to a NUMA node.</p> <p>This parameter was introduced in HPC Pack 2012 and is not supported in previous versions.</p>
<code>/wdir <working_dir></code>	<p>Specifies the working directory to use for the application specified in the same section of the mpexec command as the /wdir parameter. This directory can have a local or a remote path. You can also specify this parameter as /dir.</p>

	<p>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 the /wdir parameter specifies for the section overrides the setting that the /gwdir parameter specifies.</p>
<p>/env <env_var_name> <env_var_value></p>	<p>Sets an environment variable for the application that is specified in the same 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 /genv parameter specifies.</p> <p>To specify the values of multiple environment variables, specify the /env parameter multiple times, once for each environment variable.</p> <p>For a list of built-in environment variables that you can set with the /env parameter, see MPICH environment variables.</p>
<p>/genvlist <env_var1>[, <env_var2>...]</p>	<p>Passes the values of the specified environment variables to the processes that mpiexec starts. The list is a comma-separated list of environment variable names.</p> <p>For a list of built-in environment variables that you can use with the /genvlist parameter, see MPICH environment variables.</p>
<p>/exitcodes</p>	<p>Prints the exit codes of the processes that mpiexec started as part of standard output at the end of the run.</p>
<p>/priority {0 1 2 3 4}</p>	<p>Sets the priority for the processes that mpiexec starts. The default priority is 0. Following are the possible values:</p> <p>0 - Idle</p> <p>1 - Below normal</p> <p>2 - Normal</p> <p>3 - Above normal</p> <p>4 - High</p>
<p>/port <port></p>	<p>Specifies the port on which the <code>smpd.exe</code> process listens. You can also specify this parameter as /p.</p>
<p>/path <path1>[;<path2>...]</p>	<p>Specifies one or more paths to search on the target node for the application 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 the paths include spaces, enclose the paths in quotation marks ("). The value that you specify for this parameter does not replace and is not appended to the <code>PATH</code> environment variable.</p> <p>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 the /path parameter specifies for the section overrides the setting that the /gpath parameter specifies.</p>
<p>/timeout <seconds></p>	<p>Sets the amount of time that job that runs the mpiexec command should run before timing out, in seconds.</p>
<p>/job <string></p>	<p>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 Manager Service.</p>
<p>/lines</p>	<p>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.</p>
<p>/trace [(<filter1>,<filter2>...)]</p>	<p>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 to enable only trace events of interest. List the event filter names, or their equivalent hexadecimal values, by using a comma-separated list enclosed in brackets.</p>

parentheses. **Important:** While this type of tracing can be useful for a very simple test application, in most practical applications, this method generates such a large trace file that it is not very useful. To control tracing and create manageable trace files, use an Event Tracing for Windows (ETW) tool such as 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 each node. Use the **/tracefile** parameter to specify an alternative trace file.

Following are the event filter names and equivalent hexadecimal values that 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>	<p>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_identifier>. You can also specify this parameter as /tf.</p> <p>This parameter is deprecated as of HPC Pack 2008 R2 with Service Pack 3 (SP3).</p>
/tracemax <max_size>	<p>Specifies the maximum size of the trace log file, in megabytes. You must have at least <i>max_size</i> megabytes of free space available on the drive that you specify for the trace file.</p>

	<p>The binary tracing data is written by using a circular buffer, so when the data exceeds the maximum size of the file, the tracing data is overwritten starting at the beginning of the file. As a result, the tracing log file always contains the most recent <i>max_size</i> megabytes of tracing data from the MPI job.</p> <p>Each binary record in the trace log file has a time stamp, so that log file viewers can display the information in chronological order regardless of the wrapping that occurs when the tracing data is overwritten.</p> <p>The default value for this parameter is 10240. Specify 0 to allow the creation of a trace log file of unrestricted size.</p> <p>You can also specify this parameter as /tm.</p> <p>This parameter was introduced in HPC Pack 2008 R2. It is deprecated as of HPC Pack 2008 R2 with Service Pack 3 (SP3).</p>
/debug [{0 1 2 3}]	<p>Prints the specified level of debugging information to standard error. If you specify the /debug option without a value, mpiexec behaves as if you specified a value of 2. Following are the possible debugging levels:</p> <p>0 - No debugging information is displayed.</p> <p>1 - Information about errors is displayed.</p> <p>2 - Extensive debugging information is displayed.</p> <p>3 - Both errors and extensive debugging information are displayed.</p> <p>You can also specify this parameter as /d.</p>
/logfile <log_file_name>	<p>Specifies the name of file to use for the log file, including the path.</p>
<sectionN_parameters>	<p>Specifies parameters that apply only to the section of the mpiexec command in which they occur, and includes the following parameters:</p> <ul style="list-style-type: none">- /env- /path- /wdir- /host- /hosts- /machinefile- /np <p>The Syntax section explicitly lists these parameters for the first section, and uses the <sectionN_parameters> placeholder for the additional sections. For more information about sections in an mpiexec command, see the description for the <applicationN> [<applicationN_parameters>] parameter.</p>
<applicationN> [<applicationN_parameters>]	<p>Specifies the MPI applications that you want to start and any necessary parameters for those applications. You can specify more than one application by including multiple colon-separated sections in the mpiexec command.</p> <p>If you include multiple sections to specify multiple applications, the applications need to have compatible requirements for the ranks on which the applications 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 to run on rank 0 and a subordinate process that does not need to run on rank 0.</p>
/pwd <string>	<p>Authenticates the user with the provided password. This option is only valid when MS-MPI Launch Service is being used.</p>
/saveCreds	<p>Notifies the Launch Service to save credentials. This option is only valid when /pwd is provided.</p> <p>After a successful invocation of /saveCreds, it is not necessary to provide the password with /pwd unless the password is changed.</p>
/unicode	<p>Switches mpiexec output to unicode stream. This only affects the output of mpiexec. Unicode path and executables are supported with or without this option.</p>

/?	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 and 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 mpiexec 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.