Genesys Voice Platform

log Section

5/8/2025

log Section

all

Default Value: ../logs/ResourceMgr

Valid Values:

- **stdout** Log events are sent to the Standard output (stdout).
- **stderr** Log events are sent to the Standard error output (stderr).
- **network** Log events are sent to Message Server, which can reside anywhere on the network. Message Server stores the log events in the Log Database. Setting the all log level option to the network output enables an application to send log events of the Standard, Interaction, and Trace levels to Message Server. Debug-level log events are neither sent to Message Server nor stored in the Log Database.
- **memory** Log events are sent to the memory output on the local disk. This is the safest output in terms of the application performance.
- **[filename]** Log events are stored in a file with the specified name. If a path is not specified, the file is created in the application's working directory.
  Changes Take Effect: immediately
  Specifies the outputs to which an application sends all log events. The log output types must be separated by a comma when more than one output is configured.

check-point

Default Value: 1
Valid Values: 0 - 24
Changes Take Effect: immediately

Specifies, in hours, how often the application generates a check point log event, to divide the log into sections of equal time. By

default, the application generates this log event every hour. Setting the option to 0 prevents the generation of check-point events.

compatible-output-priority

Default Value: false
Valid Values: true, false
Changes Take Effect: immediately

Specifies whether the application uses 6.x output logic.

- **true** The log of the level specified by "Log Output Options" is sent to the specified output.
- **false** The log of the level specified by "Log Output Options" and higher levels is sent to the specified output.

debug

Default Value: ../logs/ResourceMgr

Valid Values:

- **stdout** Log events are sent to the Standard output (stdout).
- **stderr** Log events are sent to the Standard error output (stderr).
- **network** Log events are sent to Message Server, which can reside anywhere on the network. Message Server stores the log events in the Log Database. Setting the all log level option to the network output enables an application to send log events of the Standard, Interaction, and Trace levels to Message Server. Debug-level log events are neither sent to Message Server nor stored in the Log Database.
- **memory** Log events are sent to the memory output on the local disk. This is the safest output in terms of the application performance.
- **[filename]** Log events are stored in a file with the specified name. If a path is not specified, the file is created in the application's working directory.
  Changes Take Effect: immediately
  Specifies the outputs to which an application sends the log events of the Debug level and higher (that is, log events of the Standard, Interaction, Trace, and Debug levels). The log output types must be separated by a comma when more than one output is configured.

expire

Default Value: 20

Valid Values:

- **false** No expiration; all generated segments are stored.
- **[number] file or [number]** Sets the maximum number of log files to store. Specify a number from 1-1000.
- **[number] day** Sets the maximum number of days before log files are deleted. Specify a number from 1-100.
  Changes Take Effect: immediately
  Determines whether log files expire. If they do, sets the measurement for determining when they expire, along with the maximum number of files (segments) or days before the files are removed. This option is ignored if log output is not configured to be sent to a log file. Note: If the value of the option is set incorrectly -out of the range of valid values- it will be automatically reset to 10

interaction

Default Value: ../logs/ResourceMgr

Valid Values:

- **stdout** Log events are sent to the Standard output (stdout).
- **stderr** Log events are sent to the Standard error output (stderr).
- **network** Log events are sent to Message Server, which can reside anywhere on the network. Message Server stores the log events in the Log Database. Setting the all log level option to the network output enables an application to send log events of the Standard, Interaction, and Trace levels to Message Server. Debug-level log events are neither sent to Message Server nor stored in the Log Database.
- **memory** Log events are sent to the memory output on the local disk. This is the safest output in terms of the application performance.
- **[filename]** Log events are stored in a file with the specified name. If a path is not specified, the file is created in the application's working directory.
  Changes Take Effect: immediately
  Specifies the outputs to which an application sends the log events of the Interaction level and higher (that is, log events of the Standard and Interaction levels). The log outputs must be separated by a comma when more than one output is configured.

keep-startup-file

Default Value: false

Valid Values:

- **false** No startup segment of the log is kept.
- **true** A startup segment of the log is kept. The size of the segment equals the value of the segment option.
- **[number] KB** Sets the maximum size, in kilobytes, for a startup segment of the log.
- **[number] MB** Sets the maximum size, in megabytes, for a startup segment of the log.
  Changes Take Effect: After restart
  Specifies whether a startup segment of the log, containing the initial T-Server configuration, is to be kept. If it is, this option can be set to true or to a specific size. If set to true, the size of the initial segment will be equal to the size of the regular log segment defined by the segment option. The value of this option will be ignored if segmentation is turned off (that is, if the segment option set to false).

memory

Default Value:
Valid Values: [string] (memory file name)
Changes Take Effect: immediately

Specifies the name of the file to which the application regularly prints a snapshot of the memory output, if it is configured to do this. The new snapshot overwrites the previously written data. If the application terminates abnormally, this file will contain the latest log messages. Memory output is not recommended for processors with a CPU frequency lower than 600 MHz.

memory-storage-size

Default Value:

Valid Values:

- **[number] KB or [number]** The size of the memory output, in kilobytes. The minimum value is 128 KB.
- **[number] MB** The size of the memory output, in megabytes. The maximum value is 64 MB
  Changes Take Effect: When memory output is created
  Specifies the buffer size for log output to the memory, if configured.

message_format

Default Value: short

Valid Values:

- **short** An application uses compressed headers when writing log records in its log file.
- **full** An application uses complete headers when writing log records in its log file.
  Changes Take Effect: immediately
  Specifies the format of log record headers that an application uses when writing logs in the log file. Using compressed log record headers improves application performance and reduces the log file's size. With the value set to short:
- A header of the log file or the log file segment contains information about the application (such as the application name, application type, host type, and time zone), whereas single log records within the file or segment omit this information.
- A log message priority is abbreviated to Std, Int, Trc, or Dbg, for Standard, Interaction, Trace, or Debug messages, respectively.
- The message ID does not contain the prefix GCTI or the application type ID.
  A log record in the full format looks like this:
  2002-05-07T18:11:38.196 Standard localhost cfg_dbserver GCTI-00-05060 Application started
  A log record in the short format looks like this:
  2002-05-07T18:15:33.952 Std 05060 Application started

messagefile

Default Value:
Valid Values: [string].lms (message file name)
Changes Take Effect: Immediately, if an application cannot find its *.lms file at startup

Specifies the file name for application-specific log events. The name must be valid for the operating system on which the application is running. The option value can also contain the absolute path to the application-specific *.lms file. Otherwise, an application looks for the file in its working directory.

print-attributes

Default Value: false
Valid Values: true, false
Changes Take Effect: immediately

Specifies whether the application attaches extended attributes, if any exist, to a log event that it sends to log output. Typically, log events of the Interaction log level and Audit-related log events contain extended attributes. Setting this option to true enables audit capabilities, but negatively affects performance. Genesys recommends enabling this option for Solution Control Server and Configuration Server when using audit tracking. For other applications, refer to Genesys 7.5 Combined Log Events Help to find out whether an application generates Interaction-level and Audit-related log events; if it does, enable the option only when testing new interaction scenarios.

- **true** Attaches extended attributes, if any exist, to a log event sent to log output.
- **false** Does not attach extended attributes to a log event sent to log output.

segment

Default Value: 10000

Valid Values:

- **false** No segmentation is allowed.
- **[number] KB or [number]** Sets the maximum segment size, in kilobytes. The minimum segment size is 100 KB.
- **[number] MB** Sets the maximum segment size, in megabytes.
- **[number] hr** Sets the number of hours for the segment to stay open. The minimum number is 1 hour.
  Changes Take Effect: immediately
  Specifies whether there is a segmentation limit for a log file. If there is, sets the mode of measurement, along with the maximum size. If the current log segment exceeds the size set by this option, the file is closed and a new one is created.

spool

Default Value:
Valid Values: [path] (the folder, with the full path to it)
Changes Take Effect: immediately

Specifies the folder, including full path to it, in which an application creates temporary files related to network log output. If you change the option value while the application is running, the change does not affect the currently open network output.

standard

Default Value: ../logs/ResourceMgr

Valid Values:

- **stdout** Log events are sent to the Standard output (stdout).
- **stderr** Log events are sent to the Standard error output (stderr).
- **network** Log events are sent to Message Server, which can reside anywhere on the network. Message Server stores the log events in the Log Database. Setting the all log level option to the network output enables an application to send log events of the Standard, Interaction, and Trace levels to Message Server. Debug-level log events are neither sent to Message Server nor stored in the Log Database.
- **memory** Log events are sent to the memory output on the local disk. This is the safest output in terms of the application performance.
- **[filename]** Log events are stored in a file with the specified name. If a path is not specified, the file is created in the application's working directory.
  Changes Take Effect: immediately
  Specifies the outputs to which an application sends the log events of the Standard level. The log output types must be separated by a comma when more than one output is configured.

time_convert

Default Value: local
Valid Values: local, utc
Changes Take Effect: immediately

Specifies the system in which an application calculates the log record time when generating a log file. The time is converted from the time in seconds since the Epoch (00:00:00 UTC, January 1, 1970).

- **Local Time (local)** The time of log record generation is expressed as a local time, based on the time zone and any seasonal adjustments. Time zone information of the application's host computer is used.
- **Coordinated Universal Time (utc)** The time of log record generation is expressed as Coordinated Universal Time (UTC).

time_format

Default Value: ISO8601
Valid Values: time, locale, ISO8601
Changes Take Effect: immediately

Specifies how to represent, in a log file, the time when an application generates log records.
A log record's time field in the ISO 8601 format looks like this:
2001-07-24T04:58:10.123

- **HH:MM:SS.sss (time)** The time string is formatted according to the HH:MM:SS.sss (hours, minutes, seconds, and milliseconds) format.
- **According to the system's locale (locale)** The time string is formatted according to the system's locale.
- **ISO 8601 format (ISO8601)** The date in the time string is formatted according to the ISO 8601 format. Fractional seconds are given in milliseconds.

trace

Default Value: ../logs/ResourceMgr

Valid Values:

- **stdout** Log events are sent to the Standard output (stdout).
- **stderr** Log events are sent to the Standard error output (stderr).
- **network** Log events are sent to Message Server, which can reside anywhere on the network. Message Server stores the log events in the Log Database. Setting the all log level option to the network output enables an application to send log events of the Standard, Interaction, and Trace levels to Message Server. Debug-level log events are neither sent to Message Server nor stored in the Log Database.
- **memory** Log events are sent to the memory output on the local disk. This is the safest output in terms of the application performance.
- **[filename]** Log events are stored in a file with the specified name. If a path is not specified, the file is created in the application's working directory.
  Changes Take Effect: immediately
  Specifies the outputs to which an application sends the log events of the Trace level and higher (that is, log events of the Standard, Interaction, and Trace levels). The log outputs must be separated by a comma when more than one output is configured.

verbose

Default Value: standard
Valid Values: all, debug, trace, interaction, standard, none
Changes Take Effect: immediately

Determines whether a log output is created. If it is, specifies the minimum level of log events generated. The log events levels, starting with the highest priority level, are standard, interaction, trace, debug and all.

- **all** All log events (that is, log events of the Standard, Trace,Interaction, and Debug levels) are generated.
- **debug** The same as all.
- **trace** Log events of the Trace level and higher (that is, log events of the Standard, Interaction, and Trace levels) are generated, but log events of the Debug level are not generated.
- **interaction** Log events of the Interaction level and higher (that is, log events of the Standard and Interaction levels) are generated, but log events of the Trace and Debug levels are not generated.
- **standard** Log events of the Standard level are generated, but log events of the Interaction, Trace, and Debug levels are not generated.
- **none** No log evenets are generated.