Liberica JDK Performance Edition Overview



Liberica JDK Revision 1.0 February 2025

be//soft

Copyright © BellSoft Corporation 2018-2025.

BellSoft software contains open source software. Additional information about third party code is available at <u>https://bell-sw.com/third_party_licenses</u>. You can also get more information on how to get a copy of source code by contacting <u>info@bell-sw.com</u>.

THIS INFORMATION MAY CHANGE WITHOUT NOTICE. TO THE EXTENT PERMITTED BY APPLICABLE LAW, BELLSOFT PROVIDES THIS DOCUMENTATION "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT WILL BELLSOFT BE LIABLE TO YOU OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, LOST INVESTMENT, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF BELLSOFT IS EXPRESSLY ADVISED IN ADVANCE OF THE POSSIBILITY OF SUCH LOSS OR DAMAGE.

The use of any software product referenced in this document is governed by the applicable license agreement, which is not modified in any way by the terms of this notice.

Alpaquita, Liberica and BellSoft are trademarks or registered trademarks of BellSoft Corporation. The registered trademark Linux® is used pursuant to a sublicense from the Linux Foundation, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis. Java and OpenJDK are trademarks or registered trademarks of Oracle and/or its affiliates. Other trademarks are the property of their respective owners and are used only for identification purposes.

Contents

1. Introduction	4
2. Features and Enhancements of Liberica JDK	
Performance Edition	5
3. Installing Liberica JDK Performance Edition	6

6

4. Changes to JVM Runtime Options in Liberica JDK

Performance Edition	
Changes to runtime options in JDK 11	7
Added options in liberica-perf 11 based on JVM 17	7
Removed and renamed options in liberica-perf 11 based on JVM 17	10
Added options in liberica-perf 11 based on JVM 21	15
Removed and renamed options in liberica-perf 11 based on JVM 21	18
Changes to Runtime Options in JDK 8	25
Added runtime options	25

33

1. Introduction

Liberica JDK Performance Edition (or liberica-perf) brings some of the performance of JVM 17 and JVM 21 to projects that can't afford complete migration out of JDK 8 or JDK 11. Liberica JDK Performance Edition is available for all platforms the Liberica JDK 8 and 11 are available for.



2. Features and Enhancements of Liberica JDK Performance Edition

Liberica JDK Performance Edition includes the following features:

- Z Garbage Collector (new in JDK 8, improved in JDK 11): a scalable, low latency garbage collector.
- Compact Strings (new in JDK 8): a space-efficient internal representation of strings, which reduces memory footprint and garbage collection activity; it's enabled by default.
- Unified JVM Logging (new in JDK 8): replaces JDK options that print details about the JVM with -Xlog options.

In addition, Liberica JDK Performance Edition includes the following enhancements:

- Garbage-First (G1) Garbage Collector: targeted for multiprocessor machines scaling to a large amount of memory. This is the default garbage collector for all versions of Liberica JDK Performance Edition.
- G1 String Deduplication: reduces the memory footprint of String objects on the Java heap by taking advantage of the fact that many String objects are identical. It's disabled by default, but you can enable it with the -XX:+UseStringDeduplication option.
- Class Data Sharing (CDS): helps reduce the startup time and memory footprint between multiple JVMs. It's enabled by default in Liberica JDK Performance Edition. To disable it, see <u>Manually</u> <u>Controlling Class Data Sharing</u>.
- Enhanced Java Flight Recorder: a tool for collecting diagnostic and profiling data for a running Java application.

Note:

Concurrent Mark Sweep Garbage Collector is absent in liberica-perf 8 and 11.

3. Installing Liberica JDK Performance Edition

Installation procedure for Liberica JDK Performance Edition is similar to Liberica JDK. For detailed Liberica JDK installation and configuration instructions, see *Installation Guide* for the corresponding <u>Liberica JDK version</u>.

4. Changes to JVM RuntimeOptions in Liberica JDKPerformance Edition

Several JVM runtime options were added from JDK 17 and JDK 21, some were removed. See the tables below for additional information.

Changes to runtime options in JDK 11

Added options in liberica-perf 11 based on JVM 17

The following is a list of options available in liberica-perf 11, but absent in JDK 11. For more information on these parameters, see <u>the official documentation</u>.

New option name	Description
AdjustStackSizeForTLS	Increases the thread stack size to include space for glibc static thread-local storage (TLS) if true.
AllowRedefinitionToAddDeleteMethods	Allows redefinition to add and delete private static or final methods for compatibility with old releases.
ArchiveClassesAtExit	Specifies the path and name of the dynamic archive file.
AsyncLogBufferSize	Specifies the memory in bytes for the intermediate buffer.

New option name	Description
C1InlineStackLimit	Limits stack frame size impact for inlining decisions for C1 JIT compiler.
C1MaxInlineLevel	Specifies the maximum number of nested calls that are inlined by C1 JIT compiler.
C1MaxInlineSize	Specifies the maximum bytecode size of a method to be inlined by C1 JIT compiler.
C1MaxRecursiveInlineLevel	Specifies the maximum number of nested recursive calls that are inlined by C1 JIT compiler.
C1MaxTrivialSize	Specifies the maximum bytecode size of a trivial method to be inlined by C1 JIT compiler.
CompilationMode	Sets C1 JIT compiler compilation mode, such as default, quick-only, high-only, high- only-quick-internal.
DynamicDumpSharedSpaces	Dynamic archive
G1PeriodicGCInterval	Determines a minimum interval in ms at which G1 Garbage Collector considers performing a garbage collection.
G1PeriodicGCInvokesConcurrent	If set, periodic garbage collections trigger a concurrent marking or continue the existing collection cycle, otherwise trigger a Full GC.
G1PeriodicGCSystemLoadThreshold	Sets a threshold for the current system load as returned by the hosts getloadavg() call to determine whether a periodic garbage collection should be triggered.

New option name	Description
G1RSetRegionEntries	Sets the max number of regions.
G1RSetSparseRegionEntries	Sets the eager reclaim threshold. If set the humongous objects are reclaimed only if their remembered set has less than the specified number of entries.
HeapDumpGzipLevel	Sets the level of compressing the heap dump.
MetaspaceReclaimPolicy	Reclaims memory after purging and returns it to the Operating System. This switch controls fine-tuning of metaspace reclamation (balanced aggressive none).
MinHeapSize	Sets the minimum size (in bytes) of the memory allocation pool.
RecordDynamicDumpInfo	Records a separate dynamically-generated archive on top of the default system for each application.
ShowCodeDetailsInExceptionMessages	Enables printing of improved NullPointerException messages.
SweeperThreshold	Sets the cache sweeper threshold value.
UseContainerCpuShares	Takes cpu.shares into account when determining how many threads to allocate to the various thread pools.
UseEmptySlotsInSupers	Allows allocating fields in empty slots of super- classes.
UseNotificationThread	If set uses Notification Thread.

New option name	Description
ZMarkStackSpaceLimit	Sets the maximum number of bytes allocated for mark stacks.
ZProactive	Enables proactive GC cycles.
ZUncommit	Uncommit memory if it has been unused for the specified amount of time (in seconds).
ZUncommitDelay	Uncommit memory if it has been unused for the specified amount of time (in seconds).

Removed and renamed options in liberica-perf 11 based on JVM 17

The table below lists options unavailable in liberica-perf 11 and the options that were renamed.

JDK 11 options	Resolution in liberica-perf 11
AggressiveOpts	Removed
AllowJNIEnvProxy	Removed
AllowNonVirtualCalls	Removed
AssertOnSuspendWaitFailure	Removed
AssumeMP	Removed
BindGCTaskThreadsToCPUs	Removed
BranchOnRegister	Removed

JDK 11 options	Resolution in liberica-perf 11
BytecodeVerificationLocal	Moved to the diagnostic flags category. Use with -XX:+UnlockDiagnosticVMOptions
BytecodeVerificationRemote	Moved to the diagnostic flags category. Use with -XX:+UnlockDiagnosticVMOptions
CMS*	CMS GC is not supported in liberica-perf 11
CalculateClassFingerprint	AOT is not supported in liberica-perf 11
CompactFields	Removed
CompilationPolicyChoice	Removed
CompilerThreadHintNoPreempt	Removed
Debugging	Removed
DeoptimizeRandom	Moved to the non-product category. Not available in release build.
ErrorReportServer	Removed
FLSAlwaysCoalesceLarge	CMS GC is not supported in liberica-perf 11
FLSCoalescePolicy	CMS GC is not supported in liberica-perf 11
FLSLargestBlockCoalesceProximity	CMS GC is not supported in liberica-perf 11
FailOverToOldVerifier	Removed
FieldsAllocationStyle	Removed
ForceNUMA	Removed

JDK 11 options	Resolution in liberica-perf 11
G1RSetRegionEntries	Removed
G1RSetScanBlockSize	Removed
G1RSetSparseRegionEntries	Removed
GCLockerInvokesConcurrent	Removed
GCTaskTimeStampEntries	Removed
InitialBootClassLoaderMetaspaceSize	Removed
InsertMemBarAfterArraycopy	Removed
LIRFillDelaySlots	Removed
MonitorBound	Removed
MonitorInUseLists	Removed
NeedsDeoptSuspend	Removed
OldPLABWeight	Removed
ParGCDesiredObjsFromOverflowList	Removed
ParGCTrimOverflow	Removed
ParGCUseLocalOverflow	Removed
PrintJNIResolving	Functionality moved to unified logging. Use -Xlog:jni+resolve instead

JDK 11 options	Resolution in liberica-perf 11
PrintSafepointStatistics	Functionality moved to unified logging. Use -Xlog:safepoint+stats instead
PrintSafepointStatisticsCount	Functionality moved to unified logging. Use -Xlog:safepoint+stats instead
PrintSafepointStatisticsTimeout	Functionality moved to unified logging. Use -Xlog:safepoint+stats instead
PrintVMQWaitTime	Removed
ProfileIntervals	Removed
ProfileIntervalsTicks	Removed
ProfileVM	Removed
ProfilerPrintByteCodeStatistics	Removed
ProfilerRecordPC	Removed
ResizeOldPLAB	Removed
ShenandoahSoftMaxHeapSize	Known as SoftMaxHeapSize in liberica-perf 11
StressLdcRewrite	Moved to the diagnostic flags category. Use with -XX:+UnlockDiagnosticVMOptions
SuspendRetryCount	Removed
SuspendRetryDelay	Removed
ThreadLocalHandshakes	Removed

JDK 11 options	Resolution in liberica-perf 11
Tier3AOTBackEdgeThreshold	AOT is not supported in liberica-perf 11
Tier3AOTCompileThreshold	AOT is not supported in liberica-perf 11
Tier3A0TInvocationThreshold	AOT is not supported in liberica-perf 11
Tier3AOTMinInvocationThreshold	AOT is not supported in liberica-perf 11
TraceSuspendWaitFailures	Removed
TransmitErrorReport	Removed
UnlinkSymbolsALot	Removed
UseAdaptiveGCBoundary	Removed
UseCMSBestFit	CMS GC is not supported in liberica-perf 11
UseCMSInitiatingOccupancyOnly	CMS GC is not supported in liberica-perf 11
UseConcMarkSweepGC	CMS GC is not supported in liberica-perf 11
UseGCTaskAffinity	Removed
UseLWPSynchronization	Removed
UseLargePagesInMetaspace	Removed
UseLegacyJNINameEscaping	Removed
UseMembar	Removed
UseOSErrorReporting	Removed

JDK 11 options	Resolution in liberica-perf 11
UseRDPCForConstantTableBase	Removed
VMThreadHintNoPreempt	Removed
VerifyMergedCPBytecodes	Removed
ZMarkStacksMax	Removed
Zpath	Removed
ZStallOnOutOfMemory	Removed
ZStatisticsInterval	Removed

Added options in liberica-perf 11 based on JVM 21

The table below	lists option	s added to	liberica-perf	11 based on	JVM 21.

New option name	Description
AdjustStackSizeForTLS	Increases the thread stack size to include space for glibc static thread-local storage (TLS) if true.
AllowRedefinitionToAddDeleteMethods	Allows redefinition to add and delete private static or final methods for compatibility with old releases.
ArchiveClassesAtExit	Specifies the path and name of the dynamic archive file.

New option name	Description
AsyncLogBufferSize	Specifies the memory in bytes for the intermediate buffer.
AutoCreateSharedArchive	Automatically creates Dynamic CDS Archive File. The specified archive file is created if it does not exist, or if it was generated by a different JDK version.
C1InlineStackLimit	Limits stack frame size impact for inlining decisions for C1 JIT compiler.
C1MaxInlineLevel	Specifies the maximum number of nested calls that are inlined by C1 JIT compiler.
C1MaxInlineSize	Specifies the maximum bytecode size of a method to be inlined by C1 JIT compiler.
C1MaxRecursiveInlineLevel	Specifies the maximum number of nested recursive calls that are inlined by C1 JIT compiler.
C1MaxTrivialSize	Specifies the maximum bytecode size of a trivial method to be inlined by C1 JIT compiler.
CompilationMode	Sets C1 JIT compiler compilation mode, such as default, quick-only, high-only, high-only-quick-internal.
G1PeriodicGCInterval	Determines a minimum interval in ms at which G1 Garbage Collector considers performing a garbage collection.
G1PeriodicGCInvokesConcurrent	If set, periodic garbage collections trigger a concurrent marking or continue the existing collection cycle, otherwise trigger a Full GC.

New option name	Description
G1PeriodicGCSystemLoadThreshold	Sets a threshold for the current system load as returned by the hosts getloadavg() call to determine whether a periodic garbage collection should be triggered.
G1RSetRegionEntries	Sets the max number of regions.
G1RSetSparseRegionEntries	Sets the eager reclaim threshold. If set the humongous objects are reclaimed only if their remembered set has less than the specified number of entries.
GCCardSizeInBytes	Specifies the size of the remembered set entries.
HeapDumpGzipLevel	Sets the level of compressing the heap dump.
MinHeapSize	Sets the minimum size (in bytes) of the memory allocation pool.
RecordDynamicDumpInfo	Records a separate dynamically-generated archive on top of the default system for each application.
ShowCodeDetailsInExceptionMessages	Enables printing of improved NullPointerException messages.
SweeperThreshold	Sets the cache sweeper threshold value.
UseCodeAging	Inserts a counter to measure the age of a method when it is compiled.
UseContainerCpuShares	Takes cpu.shares into account when determining how many threads to allocate to the various thread pools.

New option name	Description
UseEmptySlotsInSupers	Allows allocating fields in empty slots of super- classes.
UseNotificationThread	If set uses Notification Thread.
ZCollectionIntervalMajor	If set forces GC at a fixed time interval (in seconds).
ZCollectionIntervalMinor	If set forces Minor GC at a fixed time interval (in seconds).
ZCollectionIntervalOnly	If set the GC only uses timers for GC heuristics.
ZGenerational	Uses the Generational ZGC.
ZMarkStackSpaceLimit	Sets the maximum number of bytes allocated for mark stacks.
ZProactive	Enables proactive GC cycles.
ZUncommit	Uncommit memory if it has been unused for the specified amount of time (in seconds).
ZUncommitDelay	Uncommit memory if it has been unused for the specified amount of time (in seconds).
ZYoungCompactionLimit	Sets the maximum allowed garbage in young pages. Default: 25.0 (range between 0-100%).

Removed and renamed options in liberica-perf 11 based on JVM 21

The table below lists options that were added and renamed in liberica-perf 11 based on JVM 21.

JDK 11 options	Resolution in liberica-perf 11
AggressiveOpts	deleted
AliasLevel	deleted
AllowJNIEnvProxy	deleted
AllowNonVirtualCalls	deleted
AlwaysLockClassLoader	deleted
AssertOnSuspendWaitFailure	deleted
AssumeMP	deleted
BiasedLockingBulkRebiasThreshold	deleted
BiasedLockingBulkRevokeThreshold	deleted
BiasedLockingDecayTime	deleted
BiasedLockingStartupDelay	deleted
BindGCTaskThreadsToCPU	deleted
BranchOnRegister	deleted
BytecodeVerificationLocal	moved to diagnostic mode. Use with -XX:+UnlockDiagnosticVMOptions
BytecodeVerificationRemote	moved to diagnostic mode. Use with -XX:+UnlockDiagnosticVMOptions
CMS*	CMS GC was removed along with its parameters

JDK 11 options	Resolution in liberica-perf 11
CalculateClassFingerprint	AOT is not supported in Express builds
CompactFields	deleted
CompilationPolicyChoice	deleted
CompilerThreadHintNoPreempt	deleted
CriticalJNINatives	deleted
Debugging	deleted
DeoptimizeRandom	not available in product(release) builds
DumpSharedSpaces	deleted
ErrorReportServer	deleted
ExtendedDTraceProbes	deleted
FLSAlwaysCoalesceLarge	deleted
FLSCoalescePolicy	deleted
FLSLargestBlockCoalesceProximity	deleted
FailOverToOldVerifier	deleted
FieldsAllocationStyle	deleted
FilterSpuriousWakeups	deleted
ForceNUMA	deleted

JDK 11 options	Resolution in liberica-perf 11
G1ConcRSHotCardLimit	deleted
G1ConcRSLogCacheSize	deleted
G1ConcRefinementGreenZone	deleted
G1ConcRefinementRedZone	deleted
G1ConcRefinementServiceIntervalMillis	deleted
G1ConcRefinementThresholdStep	deleted
G1ConcRefinementYellowZone	deleted
G1RSetScanBlockSize	deleted
G1UseAdaptiveConcRefinement	deleted
GCLockerInvokesConcurrent	deleted
GCTaskTimeStampEntries	deleted
InitialBootClassLoaderMetaspaceSize	deleted
InsertMemBarAfterArraycopy	deleted
LIRFillDelaySlots	deleted
MinInliningThreshold	deleted
MonitorBound	deleted
MonitorInUseLists	deleted

JDK 11 options	Resolution in liberica-perf 11
NeedsDeoptSuspend	deleted
OldPLABWeight	deleted
ParGCDesiredObjsFromOverflowList	deleted
ParGCTrimOverflow	deleted
ParGCUseLocalOverflow	deleted
PreferContainerQuotaForCPUCount	deleted
PrefetchFieldsAhead	deleted
PrintJNIResolving	Moved to unified logging. Use "- Xlog:jni+resolve" instead
PrintSafepointStatistics	Moved to unified logging. Use "- Xlog:safepoint+stats" instead
PrintSafepointStatisticsCount	Moved to unified logging. Use "- Xlog:safepoint+stats" instead
PrintSafepointStatisticsTimeout	Moved to unified logging. Use "- Xlog:safepoint+stats" instead
PrintVMQWaitTime	deleted
ProfileIntervals	deleted
ProfileIntervalsTicks	deleted
ProfileVM	deleted

JDK 11 options	Resolution in liberica-perf 11
ProfilerPrintByteCodeStatistics	deleted
ProfilerRecordPC	deleted
RefDiscoveryPolicy	deleted
RequireSharedSpaces	deleted
ResizeOldPLAB	deleted
ShenandoahSoftMaxHeapSize	Use SoftMaxHeapSize in jdk express instead
SoftMaxHeapSize	instead of old ShenandoahSoftMaxHeapSize
StressLdcRewrite	moved to diagnostic mode. Use with -XX:+UnlockDiagnosticVMOptions
ThreadLocalHandshakes	deleted
Tier3AOTBackEdgeThreshold	AOT is not supported in Express builds
Tier3AOTCompileThreshold	AOT is not supported in Express builds
Tier3AOTInvocationThreshold	AOT is not supported in Express builds
Tier3A0TMinInvocationThreshold	AOT is not supported in Express builds
TraceSuspendWaitFailures	deleted
TransmitErrorReport	deleted
TrimNativeHeapInterval	moved to be available in product(release) builds

JDK 11 options	Resolution in liberica-perf 11
UnlinkSymbolsALot	deleted
UseAdaptiveGCBoundary	deleted
UseBiasedLocking	deleted
UseCMSBestFit	CMS GC was removed along with its parameters
UseCMSInitiatingOccupancyOnly	CMS GC was removed along with its parameters
UseConcMarkSweepGC	CMS GC was removed along with its parameters
UseGCTaskAffinity	deleted
UseHeavyMonitors	deleted
UseLWPSynchronization	deleted
UseLargePagesInMetaspace	deleted
UseLegacyJNINameEscaping	deleted
UseMembar	deleted
UseOSErrorReporting	deleted
UseRDPCForConstantTableBase	deleted
UseSharedSpaces	deleted

JDK 11 options	Resolution in liberica-perf 11
UseSystemMemoryBarrier	moved to be available in product(release) builds
VMThreadHintNoPreempt	deleted
VerifyMergedCPBytecodes	deleted
ZMarkStacksMax	deleted
Zpath	deleted
ZStallOnOutOfMemory	deleted
ZStatisticsInterval	deleted

Changes to Runtime Options in JDK 8

Added runtime options

The following is a list of options available in liberica-perf 8, but absent in JDK 8.

New options in liberica-perf 8	
AdjustStackSizeForTLS	Increases the thread stack size to include space for glibc static thread-local storage (TLS) if true.
AllocateHeapAt	Specifies a path to the directory where a temporary file is created to use as the backup store for Java Heap.

New options in liberica-perf 8	
AllowRedefinitionToAddDeleteMethods	Allows redefinition to add and delete private static or final methods for compatibility with old releases.
AllowVectorizeOnDemand	Globally suppresses vectorization set in VectorizeMethod.
ArchiveClassesAtExit	Specifies the path and name of the dynamic archive file.
ArrayCopyLoadStoreMaxElem	Specifies the maximum number of arraycopy elements inlined as a sequence of loads/stores.
AsyncLogBufferSize	Specifies the memory in bytes for the intermediate buffer.
C1InlineStackLimit	Limits stack frame size impact for inlining decisions for C1 JIT compiler.
C1MaxInlineLevel	Specifies the maximum number of nested calls that are inlined by C1 JIT compiler.
C1MaxInlineSize	Specifies the maximum bytecode size of a method to be inlined by C1 JIT compiler.
C1MaxRecursiveInlineLevel	Specifies the maximum number of nested recursive calls that are inlined by C1 JIT compiler.
C1MaxTrivialSize	Specifies the maximum bytecode size of a trivial method to be inlined by C1 JIT compiler.

CompactStrings	Disables the Compact Strings feature. By default, this option is enabled. When this option is enabled, Java Strings containing only single-byte characters are internally represented and stored as single-byte-per- character Strings using ISO-8859-1 / Latin-1 encoding.
CompilationMode	Sets C1 JIT compiler compilation mode, such as default, quick-only, high-only, high- only-quick-internal.
CompileThresholdScaling	Provides unified control of first compilation. This option controls when methods are first compiled for both the tiered and the nontiered modes of operation.
CreateCoredumpOnCrash	Creates core/mini dump on VM fatal error.
DoReserveCopyInSuperWord	Creates reserve copy of graph in SuperWord.
DynamicDumpSharedSpaces	Dynamic archive
EnableDynamicAgentLoading	Allows dynamic loading of agents into a running JVM.
ErrorLogTimeout	Timeout, in seconds, to limit the time spent on writing an error log in case of a crash.
ExecutingUnitTests	Specifies whether the JVM can run unit tests or not.
ExtensiveErrorReports	Enables the reporting of more extensive error information in the ErrorFile.

New options in liberica-perf 8	
G1PeriodicGCInterval	Determines a minimum interval in ms at which G1 Garbage Collector considers performing a garbage collection.
G1PeriodicGCInvokesConcurrent	If set, periodic garbage collections trigger a concurrent marking or continue the existing collection cycle, otherwise trigger a Full GC.
G1PeriodicGCSystemLoadThreshold	Sets a threshold for the current system load as returned by the hosts getloadavg() call to determine whether a periodic garbage collection should be triggered.
G1UseAdaptiveIHOP	Controls adaptive calculation of the old generation occupancy to start background work preparing for an old generation collection.
HeapDumpGzipLevel	Sets the compression level of the heap dump.
HeapSearchSteps	Heap allocation steps through preferred address regions to find where it can allocate the heap. Number of steps to take per region.
LoopPercentProfileLimit	Unrolls loop bodies with the percent node count of profile limit.
LoopStripMiningIter	Controls the number of iterations in the inner strip mined loop.
LoopStripMiningIterShortLoop	Controls loop strip mining optimization.
MetaspaceReclaimPolicy	Reclaims memory after purging and returns it to the Operating System. This switch controls fine-tuning of metaspace reclamation (balanced aggressive none).

New options in liberica-perf 8	
MinHeapSize	Sets the minimum size (in bytes) of the memory allocation pool.
NonNMethodCodeHeapSize	Sets the size in bytes of the code segment containing nonmethod code.
NonProfiledCodeHeapSize	Sets the size in bytes of the code segment containing nonprofiled methods.
OptoRegScheduling	Instruction Scheduling before register allocation for pressure.
PreTouchParallelChunkSize	The size of the chunk of memory for each thread when using parallel memory pre-touch.
PrintExtendedThreadInfo	Prints more information in thread dump.
PrintFlagsRanges	Prints the range specified and allows automatic testing of the values.
ProfiledCodeHeapSize	Sets the size in bytes of the code segment containing profiled methods.
RecordDynamicDumpInfo	Records a separate dynamically-generated archive on top of the default system for each application.
RestrictReservedStack	Restricts @ReservedStackAccess to trusted classes.
SegmentedCodeCache	Enables segmentation of the code cache.
SharedArchiveConfigFile	Specifies additional shared data added to the archive file.

New options in liberica-perf 8	
SharedArchiveFile	Specifies the path and name of the class data sharing (CDS) archive file.
SharedSymbolTableBucketSize	Sets the average number of symbols per bucket in shared table.
ShenandoahGCHeuristics	Sets the GC heuristics to use. This option fine- tunes the selected GC mode by choosing when to start the GC, how much to process on each cycle, and what other features to automatically enable.
ShenandoahGCMode	Sets the GC mode to use. Among other things, this defines which barriers are in use.
ShowCodeDetailsInExceptionMessages	Enables printing of improved NullPointerException messages.
ShrinkHeapInSteps	When disabled, informs the GC to shrink the Java heap directly to the target size at the next full GC rather than requiring smaller steps during multiple full GCs.
SoftMaxHeapSize	Sets the soft limit for maximum heap size (in bytes).
StackReservedPages	Sets the number of the reserved zone (reserved to annotated methods) pages of 4KB size. If pages are bigger, reserved zone is aligned respectively.

New options in liberica-perf 8

StartAggressiveSweepingAt	Force stack scanning of active methods to aggressively remove unused code when only the given percentage of the code cache is free. For a segmented code cache, it is the percentage of the non-profiled heap and for a non-segmented code cache, it is the percentage of the total code cache.
SuperWordLoopUnrollAnalysis	Maps the number of unrolls for the main loop via Superword Level Parallelism analysis.
SuperWordReductions	Enables reductions support when using SuperWords.
SweeperThreshold	Sets the cache sweeper threshold value.
UseBASE64Intrinsics	Controls the use of accelerated BASE64 encoding routines for java.util.Base64.
UseCMoveUnconditionally	Generates CMove (scalar and vector) instructions regardless of profitability analysis.
UseCodeAging	Inserts a counter to measure the age of a method when it is compiled.
UseContainerCpuShares	Takes cpu.shares into account when determining how many threads to allocate to the various thread pools.
UseDynamicNumberOfCompilerThreads	Dynamically creates compiler thread up to the limit specified by -XX:CICompilerCount.
UseEmptySlotsInSupers	Allows allocating fields in empty slots of super- classes.

New options in liberica-perf 8	
UseFMA	Enables hardware-based FMA intrinsics for hardware where FMA instructions are available (such as, Intel and ARM64).
UseNotificationThread	If set uses Notification Thread.
UseProfiledLoopPredicate	Moves predicates out of loops based on profiling data.
UseShenandoahGC	Enables the use of the Shenandoah garbage collector.
UseSubwordForMaxVector	Uses Subword Analysis to set the maximum vector size.
UseVectorCmov	Uses Vectorized Cmov.
UseXMMForObjInit	Uses XMM/YMM MOVDQU instruction for Object Initialization.
UseZGC	Enables the use of the Z garbage collector (ZGC).
ZAllocationSpikeTolerance	Sets the allocation spike tolerance for ZGC.
ZCollectionInterval	Sets the maximum interval (in seconds) between two GC cycles when using ZGC.
ZFragmentationLimit	Sets the maximum acceptable heap fragmentation (in percent) for ZGC.
ZMarkStackSpaceLimit	Sets the maximum number of bytes allocated for mark stacks.
ZProactive	Enables proactive GC cycles.

New options in liberica-perf 8	
ZUncommit	Uncommit memory if it has been unused for the specified amount of time (in seconds).
ZUncommitDelay	Uncommit memory if it has been unused for the specified amount of time (in seconds).

Removed and renamed runtime options

The table below lists options unavailable in liberica-perf 8 and the options that were renamed.

JDK 8 options	Resolution in liberica-perf 8
AdaptiveSizePausePolicy	Removed
AdjustConcurrency	Removed
AggressiveOpts	Removed
AllowJNIEnvProxy	Removed
AllowNonVirtualCalls	Removed
AssertOnSuspendWaitFailure	Removed
AssumeMP	Removed
AutoGCSelectPauseMillis	Removed
BackEdgeThreshold	Removed.Use -XX:OnStackReplacePercentage

JDK 8 options	Resolution in liberica-perf 8
BindGCTaskThreadsToCPUs	Removed
BranchOnRegister	Removed
BytecodeVerificationLocal	Is now a diagnostic option
BytecodeVerificationRemote	Is now a diagnostic option
CheckEndorsedAndExtDirs	Removed
ClearFPUAtPark	Removed
CMS*	CMS GC removed as well as its flags
CodeCacheMinimumFreeSpace	Removed
CollectGen0First	Removed
CompactFields	Removed
CompilationPolicyChoice	Removed
CompilerThreadHintNoPreempt	Removed
ConvertSleepToYield	Removed
ConvertYieldToSleep	Removed
CreateMinidumpOnCrash	Removed.UseCreateCoredumpOnCrash instead
Debugging	Removed

JDK 8 options	Resolution in liberica-perf 8
DefaultMaxRAMFraction	Removed. Use MaxRAMFraction instead
DefaultThreadPriority	Removed
DeferPollingPageLoopCount	Removed
DeferThrSuspendLoopCount	Removed
DeoptimizeRandom	Only available in non-product builds
EmitSync	Removed
EnableTracing	Removed
ErrorReportServer	Removed
ExplicitGCInvokesConcurrentAndUnloadsClass es	Removed
FailOverToOldVerifier	Removed
FastTLABRefill	Removed
FenceInstruction	Removed
FieldsAllocationStyle	Removed
FLSAlwaysCoalesceLarge	Removed
FLSCoalescePolicy	Removed
FLSLargestBlockCoalesceProximity	Removed

JDK 8 options	Resolution in liberica-perf 8
ForceNUMA	Removed
G1RSetScanBlockSize	Removed
GCLockerInvokesConcurrent	Removed
GCLogFileSize	Removed
GCTaskTimeStampEntries	Removed
InitialBootClassLoaderMetaspaceSize	Removed
InsertMemBarAfterArraycopy	Removed
JNIDetachReleasesMonitors	Removed
LazyBootClassLoader	Removed
LIRFillDelaySlots	Removed
LogJFR	Removed
MonitorBound	Removed
MonitorInUseLists	Removed
MustCallLoadClassInternal	Removed
NeedsDeoptSuspend	Removed
NmethodSweepCheckInterval	Removed
NmethodSweepFraction	Removed

JDK 8 options	Resolution in liberica-perf 8
NumberOfGCLogFiles	Removed
OldPLABWeight	Removed
ParallelGCVerbose	Removed
ParGCDesiredObjsFromOverflowList	Removed
ParGCTrimOverflow	Removed
ParGCUseLocalOverflow	Removed
PreInflateSpin	Removed
PrintAdaptiveSizePolicy	Removed
PrintClassHistogramAfterFullGC	Removed
PrintClassHistogramBeforeFullGC	Removed
PrintCMSInitiationStatistics	Removed
PrintCMSStatistics	Removed
PrintFLSCensus	Removed
PrintFLSStatistics	Removed
PrintGCApplicationConcurrentTime	Removed
PrintGCApplicationStoppedTime	Removed
PrintGCCause	Removed

JDK 8 options	Resolution in liberica-perf 8
PrintGCDateStamps	Removed
PrintGCID	Removed
PrintGCTaskTimeStamps	Removed
PrintGCTimeStamps	Removed
PrintHeapAtGC	Removed
PrintHeapAtGCExtended	Removed
PrintJNIGCStalls	Removed
PrintJNIResolving	Removed
PrintOldPLAB	Removed
PrintOopAddress	Removed
PrintParallelOldGCPhaseTimes	Removed
PrintPLAB	Removed
PrintPromotionFailure	Removed
PrintReferenceGC	Removed
PrintSafepointStatistics	Removed
PrintSafepointStatisticsCount	Removed
PrintSafepointStatisticsTimeout	Removed

JDK 8 options	Resolution in liberica-perf 8
PrintSharedSpaces	Removed
PrintStringDeduplicationStatistics	Removed
PrintTenuringDistribution	Removed
PrintTLAB	Removed
PrintVMQWaitTime	Removed
ProfileIntervals	Removed
ProfileIntervalsTicks	Removed
ProfilerPrintByteCodeStatistics	Removed
ProfilerRecordPC	Removed
ProfileVM	Removed
ReadPrefetchInstr	Removed
ReflectionWrapResolutionErrors	Removed
ResizeOldPLAB	Removed
SafepointPollOffset	Removed
SafepointSpinBeforeYield	Removed
SharedMiscCodeSize	Removed
SharedMiscDataSize	Removed

JDK 8 options	Resolution in liberica-perf 8
SharedReadOnlySize	Removed
SharedReadWriteSize	Removed
SpecialEncodeISOArray	Moved to diagnostic flags
StarvationMonitorInterval	Removed
StressLdcRewrite	Moved to diagnostic flags
SuspendRetryCount	Removed
SuspendRetryDelay	Removed
SyncFlags	Removed
SyncKnobs	Removed
SyncVerbose	Removed
ThreadSafetyMargin	Removed
TraceBiasedLocking	Removed
TraceClassLoading	Removed
TraceClassLoadingPreorder	Removed
TraceClassPaths	Removed
TraceClassResolution	Removed
TraceClassUnloading	Removed

JDK 8 options	Resolution in liberica-perf 8
TraceDynamicGCThreads	Removed
TraceExceptions	Removed
TraceGen0Time	Removed. Functionality moved to unified logging.
TraceGen1Time	Removed. Functionality moved to unified logging.
TraceLoaderConstraints	Removed
TraceMetadataHumongousAllocation	Removed
TraceMonitorInflation	Removed
TraceParallelOldGCTasks	Removed
TraceRedefineClasses	Removed
TraceSafepointCleanupTime	Removed
TraceSuspendWaitFailures	Removed
TransmitErrorReport	Removed
UnlinkSymbolsALot	Removed
UnlockCommercialFeatures	Removed
Use486InstrsOnly	Removed
UseAdaptiveGCBoundary	Removed

JDK 8 options	Resolution in liberica-perf 8
UseAESIntrinsics	Moved to diagnostic flags
UseAltSigs	Removed
UseAutoGCSelectPolicy	Removed
UseBoundThreads	Removed
UseCMSBestFit	Removed
UseCMSCollectionPassing	Removed
UseCMSCompactAtFullCollection	Removed
UseCMSInitiatingOccupancyOnly	Removed
UseCompilerSafepoints	Removed
UseConcMarkSweepGC	Removed
UseCRC32Intrinsics	Moved to diagnostic flags
UseFastAccessorMethods	Removed
UseFastEmptyMethods	Removed
UseGCLogFileRotation	Removed
UseGCTaskAffinity	Removed
UseGHASHIntrinsics	Moved to diagnostic flags
UseLargePagesInMetaspace	Removed

JDK 8 options	Resolution in liberica-perf 8
UseLegacyJNINameEscaping	Removed
UseLockedTracing	Removed
UseLWPSynchronization	Removed
UseMathExactIntrinsics	Moved to diagnostic flags
UseMembar	Removed
UseMontgomeryMultiplyIntrinsic	Moved to diagnostic flags
UseMontgomerySquareIntrinsic	Moved to diagnostic flags
UseMulAddIntrinsic	Moved to diagnostic flags
UseMultiplyToLenIntrinsic	Moved to diagnostic flags
UseOSErrorReporting	Removed from linux build
UseParallelOldGC	Removed
UseParNewGC	Removed
UseRDPCForConstantTableBase	Removed
UseSHA1Intrinsics	Moved to diagnostic flags
UseSHA256Intrinsics	Moved to diagnostic flags
UseSHA512Intrinsics	Moved to diagnostic flags
UseSquareToLenIntrinsic	Moved to diagnostic flags

JDK 8 options	Resolution in liberica-perf 8
UseVMInterruptibleIO	Removed
VerifyMergedCPBytecodes	Removed
VMThreadHintNoPreempt	Removed
WorkAroundNPTLTimedWaitHang	Removed



Liberica JDK Performance Edition Overview

