

Ported binaries on HPE Integrity NonStop servers, Java SE 8

NSJ 8—an enterprise-class environment for server-side Java applications



HPE recommends Java SE 8.

All the goodness of NSJ 7 and more
NSJ 8 provides all features of the previous release of Java SE on the NonStop viz. NSJ 7 such as 64-bit mode, large-heap space, reduced “stop-the-world” application pause time by making use of new GC algorithms (parallel GC, CMS GC, G1 GC), thread pre-emption, NSJ Infrastructure, JToolkit 2.0 and others. Refer to **NSJ 7 data sheet** (document number 4AA5-0726ENW) for the complete description of those features. NSJ 8 provides certain new features, which are elaborated in this data sheet.

This enterprise-class software environment enables development of Java applications that take advantage of the mission-critical features of HPE NonStop servers.

Meet the demands of your enterprise applications

The need for always-on and highly scalable systems and services has increased multiple times in the recent time period with the pervasive worldwide web being used for almost every purpose in our daily lives. And this situation will only become more demanding in the coming years due to the explosive growth expected in the **Internet of Things (IoT)**. Therefore, application developers must design for catering to multiple challenges when creating business-critical enterprise applications. However, the ever-decreasing time-to-market often means that attention to design and development for availability and resilience, linear scalability, and absolute data integrity are compromised. A platform automatically providing these critical

features will help streamline development workflows and free uptime that can be devoted to the functional aspects of application development.

HPE NSJ 8 software is an enterprise-class software platform that supports 64-bit applications, PUT library, and faster garbage collection (GC) algorithms. Together with **HPE NonStop** Servlets for JavaServer Pages, HPE JToolkit for HPE NonStop servers, Java Database Connectivity (JDBC) drivers for HPE NonStop SQL, HPE NonStop Application Server for Java, and HPE NonStop Message Queue software, this solution provides nearly all the advantages of the Write-Once-Run-Anywhere (WORA) concept of Java, enabling development of Java applications that take advantage of the mission-critical features of HPE Integrity NonStop servers.





This enables application developers to concentrate on their business workflow while the availability, scalability, and data integrity needs are taken care of by HPE NonStop servers.

This data sheet describes the features offered in NSJ 8 Update 4 (NSJ 8 U4), NSJI, HPjmeter, and TZUpdater software in addition to those that were offered in the NSJ 7 release. Please refer to **[NSJ 7 data sheet](#)** (document number 4AA5-0726ENW) for a full description of all the NSJ 7 and JToolkit features.

Compatible with Java SE 8.0 specification

This is a certified implementation of the Java SE 8.0 and implements all application programming interfaces (APIs) and tools that are defined for a headless¹ configuration of Java SE 8.0. To view a complete list of features in Java SE 8.0 refer to **[openjdk.java.net/projects/jdk8/features](#)**² NSJ 8 U4 is based on Oracle Java SE 8 Update 172.

New features in NSJ 8 U4 release

Dynamic Thread Pre-emption (aka dynamic thread slicing)

Thread pre-emption feature was first introduced in NSJ 7 U2 release. It provided a mechanism for application developers to optimize the performance of their

multi-threaded applications by configuring a timer which will guard how long a particular thread is able to use the services of a CPU.

In NSJ 8 U4 release the feature is enhanced to make it “dynamic” by employing an algorithm which is based on a set of user configurable parameters. This feature is called Dynamic Thread Pre-emption (or) Dynamic Thread Slicing. The amount of time granted to an application thread can be changed dynamically at run time using the parameters provided by the algorithm. Previously, the time for which a thread was allowed to occupy a CPU was fixed for the entire duration of the execution of the thread.

Security key enhancements

NSJ 8 U4 supports Diffie-Hellman key exchange algorithm prime modulus up to 8192 bits and Digital Signature Algorithm (DSA) key length up to 3072 bits.

Enhanced KeyStore Mechanism

NSJ 8 U4 brings a new security property namely `jceks.key.serialFilter`. If this filter is configured, the Java Cryptographic Extensions Key Store (JCEKS) KeyStore uses it during the deserialization of the encrypted Key object stored inside a `SecretKeyEntry`. If it is not configured or if the filter result is `UNDECIDED` or if the system property `jdk.serialFilter` is also supplied, then the filter configured by `jdk.serialFilter` is consulted.

¹ Headless mode is a system configuration in which the display device, keyboard, or mouse is lacking. For further information, refer to: **[oracle.com/technetwork/articles/javase/index-140348.html](#)**

² Features of Oracle Java SE 8 are also listed in the OpenJDK webpage.





New security property to control crypto policy

The JCE jurisdiction policy files used by the JDK can now be controlled via a new Security property `crypto.policy`. If the new Security property (`crypto.policy`) has been set in the `java.security` file or is set dynamically using the `Security.setProperty()` call before the Java Cryptographic Extensions (JCE) framework has been initialized, that setting will be honored. In older releases, JCE jurisdiction files had to be downloaded and installed separately to allow unlimited cryptography to be used by the JDK. The download and install steps are no longer necessary.

TLS session hash and extended master secret extension

NSJ 8 U4 release adds support for IETF RFC 7627—Transport Layer Security (TLS) Session Hash and Extended Master Secret Extension. By setting a new System Property `jdk.tls.allowLegacyResumption` an application can reject abbreviated handshaking when the session hash and extended master secret extension are not negotiated. By setting another new property `jdk.tls.allowLegacyMasterSecret` to false, the application can reject new connections that do not support the session hash and extended master secret extension functionality.

Features continuing from the previous NSJ 8 releases

Process timer granularity (NonStop X only)

The process timer can be set to a minimum value of 32 microseconds from NSJ 8 U4 release onwards. Previously the minimum value was 1 millisecond. The default value is still 1 millisecond. This feature is available only in NonStop X and not in NonStop i or Virtualized NonStop systems.

Compiler process

New option `UseCompilerProcess` is provided from NSJ 8 U2. If enabled the hotspot compiler thread is converted into a different process and is run in a different IPU compared to the application thread. This improves performance as the contention for CPU time between the application thread and the hotspot compiler thread is reduced.

NSJPS

A new option `-pid` is provided for `nsjps` tool. This option will list the java process specified by `<pid>`. Also due to the introduction of Compiler Process, two new options, viz. `-jh` and `-cp` have been added.

For more information specific to the new features in NSJ 8 U4 please refer to the Release Notes for Oracle Java SE 8 Updates 151, 152, 161, 162, 171 and 172 at

[JDK 8 Update Release Notes](#)



ExitOnOutOfMemoryError

If this option is enabled, the JVM exits on the first occurrence of an out-of-memory error. It can be used if user prefers to restart an instance of the JVM rather than handling out of memory errors.

CrashOnOutOfMemoryError

If this option is enabled, when an out-of-memory error occurs, the JVM crashes and produces text and binary saveabend files (if saveabend option is enabled).

Pre-compilation enhancement

Applications can benefit by compiling certain methods at their very first invocation itself (instead of using the interpreter). This feature enables a user to create a list of such methods prior to deployment of the application in production. The “compilealso” advanced compiler command enables this feature. To enable a user to create this list, a new CompileAlsoProfile command line option is provided. Using this the application developer can generate a file, which can be specified as advanced compile command file, containing the list of such methods. This will improve the application’s performance by compiling those methods the very first time the JVM executes them.

The following JDK tools are supported

jmap: Prints java object and heap information.

jinfo: Prints the java configuration information.

jstack: Stack traces of Java threads for a given process.

jstatd: Monitors creation/termination of JVM.

Lambda expressions

Lambda expression is a new and important feature included in Java SE 8. It provides facilities to do functional programming within the object-oriented Java language. Specifically, lambda expressions let one express instances of single-method classes more compactly thus providing a clear and concise way to represent one method interfaces using expressions and hence simplifying development. Lambda

expressions also improve the collection libraries making it easier to iterate through, filter, and extract data from a collection.

An article explaining lambda expressions with code snippets can be found [here](#). An “Oracle by example (OBE)” tutorial can be found at oracle.com/webfolder/technetwork/tutorials/obe/java/lambda-quickstart/index.html.

jdeps command

The jdeps command shows the package-level or class-level dependencies of Java class files. The input class can be a path name to a .class file, a directory, a JAR file, or it can be a fully qualified class name to analyze all class files. Using the command options output can be directed to system output or to a file.

jjs command

NSJ 8 contains Nashorn engine that interprets and executes JavaScript programs on the JVM. Prior to NSJ 8, JavaScript was executed by using the Mozilla Rhino engine. The Nashorn engine offers better performance than the Rhino engine and is compliant to the JavaScript specifications.

Exit of PermGen and the advent of Metaspace

Until NSJ 7 PermGen (Permanent Generation) was used to govern the amount of memory allocated and used for storing class metadata and classloaders. From NSJ 8, the class metadata are stored in the native JVM memory itself. This is called the Metaspace. The amount of memory allocated for class metadata is specified using the parameter MaxMetaspaceSize. If defaulted, then the Metaspace grows till the full JVM heap memory has been consumed. Once the memory occupied reaches the value set for MaxMetaspaceSize, garbage collection is started. Therefore, proper monitoring of Metaspace consumption is required to optimize the frequency of garbage collection runs. Metaspace usage can be monitored via the verbose GC log output.



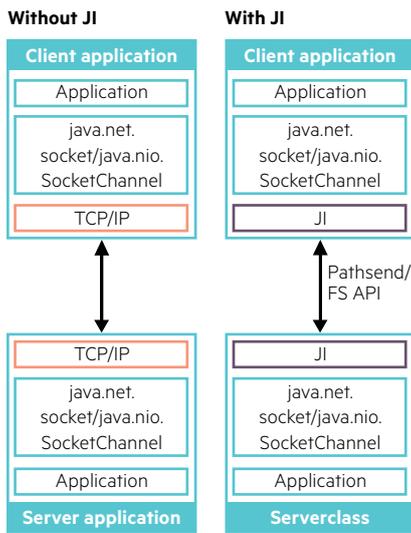


Figure 1. How JI is used to abstract the FS and TS/MP APIs

Supported by a broad range of development tools

Applications designed to run in the HPE NSJ environment can be built using many popular Java development tools. For example, you can take advantage of the HPE NonStop Development Environment for Eclipse. In addition a wide selection of commercial and open source development tools are available on Microsoft® Windows® and Linux® OS.

Garbage First Garbage Collection (G1 GC) Algorithm

NSJ 7 U2 release provided Garbage First GC (aka G1 GC) algorithm as a beta version. From NSJ 8 onwards, this algorithm is certified for production use. G1 GC is expected to have lower application pause time compared to parallel and CMS GC algorithms.

From NSJ 8, the mechanism used for starting a child process up is changed to Tdm_spawn mechanism. Before NSJ 8 it was Fork/Exec mechanism. Tdm_spawn mechanism is faster than Fork/Exec. The process launch mechanism to be used is configurable.

HPjmeter 4.4

This release of NSJ brings HPjmeter version 4.4. This version provides the ability to monitor the Metaspaces related information. This version is also capable of showing PermGen-related information if used with NSJ 7. Garbage collection log files generated using -XverboseGC or -Xloggc in NSJ 8 can be loaded from this version of HPjmeter.

TZUpdater 2.1.2

NSJ 8 U2 had brought version 2.1.1 of the TZUpdater software to NonStop. NSJ 8 U4 brings TZUpdater version 2.1.2. TZUpdater helps download and apply the latest world time zone data from the IANA's Time Zone Database. A significant improvement in this version of TZUpdater is that there is no need to upgrade the software version to support new time zone data. Dynamic update of time zones can be done from the CLI by specifying the URL from where to download the data.

NSJ 8 U4 contains IANA time zone data version 2018c. For more information, refer to **Timezone Data Versions in the JRE Software**.

NSJ Infrastructure (NSJI) enhancements after NSJ 7 was declared MATURE

NSJI has been enhanced with the features:

DDL2BEAN Converter

A new tool called “DDL2BEAN CLI tool” is released as part of the NSJI bouquet. This is an offline program which reads DDL dictionary files and generates one or more JavaBeans with the required annotations. The DDL definitions that are converted to JavaBeans are Field Definition, Group Definition, Reference Definition, Records, and Enumeration. The DDL2BEAN tool supports both DDL and DDL2 dictionaries. The generated beans will have annotations required to serialize/de-serialize the JavaBeans to/from binary format. The beans can be used in conjunction with the NSJI library to communicate with NonStop processes.

From L17.08, this tool allows the users to choose the DDL objects that should be converted to JavaBeans. It also has been enhanced to generate JavaBeans with self-contained code for serialization and de-serialization.

Large message transfer

From L17.08 NSJI supports message sizes from 32 KB to 2 MB in a single chunk of data transfer.

Table 1. Migration of applications from NSJ 6 to NSJ 8

Type of application	Porting changes for HPE NSJ 8 U3	
	32-bit JDK	64-bit JDK
Pure Java application	No change	No change
Non-threaded native library	Recompilation may be needed	Recompile as 64-bit application
Threaded native library	Recompile by specifying _PUT_MODEL and link with ZPUTDLL or XPUTDLL (for NonStop X)	Recompile as 64-bit application by specifying _PUT_MODEL and link with YPUTDLL or WPUTDLL (for NonStop X)
Non-threaded JNI invocation application	Recompile and link with ZPUTDLL or XPUTDLL (for NonStop X)	Recompile as 64-bit application and link with YPUTDLL or WPUTDLL (for NonStop X)
Threaded JNI invocation application	Recompile by specifying _PUT_MODEL and link with ZPUTDLL or XPUTDLL (for NonStop X)	Recompile as 64-bit application by specifying _PUT_MODEL and link with YPUTDLL or WPUTDLL (for NonStop X)





JMX based management and dynamic configuration

From L17.08 NSJ can be managed using JMX. Configuration parameter values can be changed at run time without needing to restart the application that is using the NSJ library.

NIO.2 APIs

From L17.08 NSJ supports standard NIO.2 APIs which allow users to perform asynchronous I/O operations on sockets. All socket related operations work in non-blocking mode.

Compatible with Java technology

HPE NSJ 8 software is compatible with Java SE 8.0 Java Development Kit (JDK 8). It includes the following standard Java components:

- All core Java APIs of JDK 8
- JDK 8 tools, except tools that require a GUI functionality

Compatibility with versions earlier than HPE NSJ 8

HPE NSJ 8 is compatible with earlier JDK versions and so a pure Java application developed for pre-HPE NSJ 8 JDK can be run on HPE NSJ 8 JDK without any

modification to the application. For other types of applications written to run on NSJ 6, there may be some changes needed. Table 1 gives more details.

Please note that there is no change required for applications written for and running on NSJ 7.

Related HPE offerings

Customer technical training

Gain the skills you need with training from Hewlett Packard Enterprise. Accelerate your technology transition, improve operational performance, and get the best return on your HPE investment. Training is available when and where you need it, through flexible delivery options and a global training capability. hpe.com/ww/learnnonstop

HPE Pointnext services

HPE Pointnext leverages our strength in infrastructure, partner ecosystems and the end-to-end lifecycle experience, to accelerate powerful, scalable IT solutions to provide you the assistance for faster time to value. HPE Pointnext provides a comprehensive portfolio including Advisory and Transformational, Professional, and Operational Services to help accelerate your digital transformation.



Technical specifications

System requirements

Hardware	HPE Integrity NonStop X servers HPE Virtualized NonStop X servers HPE Converged Virtualized NonStop X servers HPE Integrity NonStop i BladeSystem servers
Software	HPE NonStop OS, RVU L16.05, or later (for HPE Integrity NonStop X servers) HPE NonStop OS, RVU L17.02, or later (for HPE Virtualized NonStop servers) HPE NonStop OS, RVU J06.20, or later (for HPE Integrity NonStop i BladeSystem servers) The Dynamic Thread Pre-emption feature made available from NSJ 8 U4 release requires Posix User Thread Model library (PUT library) version T1280 L02 or later. This PUT library version is available on the SUT DVD from the RVU L18.08 onwards.

Operational Services

- **Datacenter Care:** HPE's most comprehensive support solution tailored to meet your specific data center support requirements. It offers a wide choice of proactive and reactive service levels to cover requirements ranging from the most basic to the most business-critical environments. **HPE Datacenter Care** Service is designed to scale to any size and type of data center environment while providing a single point of contact for all your support needs for HPE as well as selected multivendor products.
- **Critical Service:** High-performance reactive and proactive support designed to minimize downtime. It offers an assigned support team, which includes an Account Support Manager (ASM). This service offers access to HPE's Global NonStop Solution Center, 24x7 hardware and software support, six-hour Call-to-Repair commitment, enhanced parts inventory, and accelerated escalation management.
- **Proactive 24:** Provides proactive and reactive support delivered under the direction of an ASM. It offers 24x7 hardware support with four-hour on-site response, 24x7 software support with two-hour response and flexible call submittal.

- **Foundation Care:** Support for **HPE servers, storage, networking** hardware, and software to meet your availability requirements with a variety of coverage levels and response times.

Advisory & Transformation Services—HPE Pointnext designs the transformation and builds a road map tuned to your unique challenges including **hybrid cloud**, Workload and Application Migration, **Big Data**, and the **Intelligent edge**. HPE leverages proven architectures and blueprints, integrates HPE Enterprise Group and partner products and solutions, and engages Professional and Operational Services teams of HPE Pointnext as needed.

Professional Services—HPE Pointnext creates and integrates configurations that get the most out of software and hardware, and works with your preferred technologies to deliver the optimal solution. Services provided by the HPE Pointnext team, certified channel partners, or specialist delivery partners include installation and deployment services, mission-critical and technical services, and education services.

Learn more at
hpe.com/info/nonstop

Optimize your IT investment strategy with new ways to acquire, pay for, and use technology, in lock-step with your business and transformation goals.

hpe.com/solutions/hpefinancialservices





 Make the right purchase decision. [Click here to chat with our presales specialists.](#)

 **Share now**

 **Get updates**



© Copyright 2016–2019 Hewlett Packard Enterprise Development LP. The information contained herein is subject to change without notice. The only warranties for Hewlett Packard Enterprise products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Hewlett Packard Enterprise shall not be liable for technical or editorial errors or omissions contained herein.

Microsoft and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. Oracle, Java, and Java Compatible logo are registered trademarks of Oracle and/or its affiliates. Linux is the registered trademark of Linus Torvalds in the U.S. and other countries. All other third-party marks are property of their respective owners.