

# PowerPC/AIX Port

The goal of this project was to provide a full-featured and certifiable version of OpenJDK on the Linux/PowerPC and AIX/PowerPC platforms which can be ultimately integrated into the main OpenJDK development branches. This goal has been reached. Now the project maintains these platforms.

## Ongoing work

SAP is maintaining the linuxppc64le port in the jdk head repository. Both SAP and Eclipse Adoptium are continuously building and testing the linuxppc64le port in the jdk11u and jdk17(u) codelines to avoid any regressions. Further, IBM is maintaining the AIX port and supporting linuxppc64 (big endian) in 8u and 11u. Likewise IBM and Eclipse Adoptium are continuously building and testing the AIX and linuxppc64 code streams to ensure quality.

SAP has contributed implementations of Shenandoah and ZGC for linuxppc64le and more recently Virtual Threads and the Foreign Function Linker for both, linux and AIX.

For currently supported compilers see also [Supported Build Platforms](#).

## Contributions

Contributions to the port are always welcome. We maintain some issues in the OpenJDK Jira, feel free to address any of these:

[PPC: Proposed enhancements](#)

[PPC: Bugs and all other issues](#)

[AIX: Proposed enhancements](#)

[AIX: Bugs and all other issues](#)

## People

See [OpenJDK census](#)

## Supported features

The table lists in which JDK version the support for the feature was added.

	linux big endian	linux little endian	aix big endian
template interpreter	7	7	7
C2 compiler	7	7	7
C1 compiler	9	9	9
Servicability agent	9	9	no
Class Data Sharing	11	11	no
Java Flight Recorder	11	11	19
Epsilon GC	11	11	17
recognizes Power 9	11	11	11
JVMCI / graal	no	no	no
ZGC / Shenandoah	no	17.0.2, 18	no
recognizes Power 10	11.0.10, 16	11.0.10, 16	11.0.10, 16
Loom	21	20 (JEP 436)	21
Panama (FFI)	22	21	22
Vector API (incubator)	no	no	no

## Status

In [2023](#), JDK 20 and 21 have been released. SAP has delivered Panama with JDK 21 for linuxppc64le and the Foreign Linker for the big endian platforms with JDK 22. IBM intensifies its investment in the port.

In [2022](#), JDK 18 and 19 have been released. SAP has ported loom, generational ZGC and generational Shenandoah to linuxppc64le and was working on Panama for linuxppc64le.

In [2021](#), JDK 16 and 17, as well as the security updates for 8, 11, 15, 16 and 17 have been released. In head, support for Concurrent Thread Stack processing and similar improvements needed for Shenandoah and ZGC have been implemented. SAP ported Shenandoah and ZGC to linuxppc64le. The Base64 intrinsics have been optimized further.

SAP passes primary responsibility of the aixppc64 port to IBM and discontinues its active support of aixppc64 and linuxppc64 after 8u301, 11.0.12 and 17. The linuxppc64 port is abandoned in head altogether, as it is superseded by the linuxppc64le port. Linuxppc64 in 8u and 11u is now supported by IBM.

In [2020](#), JDK 14 and 15 as well as the security updates for 8, 11, 13, 14 and 15 have been released for linuxppc64, linuxppc64le and aixppc64. The port has been adapted to ongoing developments in the head revision as the removal of the C2 inline caches. Various problems have been fixed. The new Power 10 processor is recognized and first new instructions are exploited.

SAP does not actively maintain the ports in the long-term-support releases of JDK 13 and 15.

In [2019](#), JDK 12 and JDK 13 as well as the security updates for 8, 11, 12 and 13 have been released for linuxppc64, linuxppc64le and aixppc64. JDK 13 can be compiled with xLC 16 making JEP 347 (C++ 14) possible. A lot has been invested in stability and improving of JDK11. Some new intrinsics and compiler optimizations have been implemented.

In [2018](#), JDK 10 and **JDK 11** have been **released** for linuxppc64, linuxppc64le and aixppc64. Various performance optimizations have been implemented by IBM and SAP. This includes optimizations of the GCs, e.g., cheaper barriers, and various improved or new intrinsics. The linux platforms support Class Data Sharing and Java Flight Recorder since JDK11. Many issues have been fixed. SAP discontinues active support of jdk7u.

In [2017](#) JDK 9 has been released for linuxppc64, linuxppc64le and aixppc64.

In [2016](#) the JDK 7 port has been integrated into the jdk7u main line. [ppc-aix-port/jdk7u](#) is closed now. SAP contributed a row of fixes and continuous improvements to JDK 8. SAP has been working on keeping the three ppc platforms up to date in the jdk 9 project. IBM has tested and optimized JDK 8 and 9 in the Cassandra context and other open source use cases.

In [December 2015](#) SAP finished contributing the C1 compiler to jdk9. The VM now utilizes TieredCompilation. It is not planned to downport this to jdk8u or jdk7u. Alexander Smundak of Google fixed the little endian port in jdk9 to report its platform as 'ppc64le'.

In [March 2015](#) SAP finished contributing optimizations to the compressed oop modes, resulting in measurable performance improvements on ppc.

In [November 2014](#) our ports have successfully passed all the Java SE 8 Test Compatibility Kit (TCK) tests on Fedora 19 Linux/PPC64 and AIX 7.1 /PPC64. This means that our ports are fully compliant with the Java SE 8 specification on the named platforms.

In [July 2014](#) Alexander Smundak of Google contributed the little-endian port of the template interpreter to jdk8u, jdk9 and jdk7u.

In [April 2014](#) we also integrated the little-endian ppc port done by Alexander Smundak of Google into jdk8u, jdk9 and jdk7u. This port not yet supports the template interpreter and must be built setting `--with-jvm-interpreter=cpp`.

In [April 2014](#) we finished the port of the template interpreter for the big-endian platforms. The template interpreter also is ported back to the jdk 7 repository [ppc-aix-port/jdk7u](#).

On [March 27th 2014](#) all our Java 8 changes from the [ppc-aix-port/stage](#) forest have been bulk integrated into [jdk8u/hs-dev](#) and have now finally arrived in [jdk8u/jdk8u-dev](#). The [ppc-aix-port/stage](#) forest has been set to read-only mode leaving [ppc-aix-port/jdk7u](#) as the only remaining, open repository of the project. We will keep this repository open and merge it from time to time with the upstream [ppc-aix-port/jdk7u](#) for downstream users of our port like Linux packagers or the new ppc64le (little-endian) community.

As of [February 2014](#), our port has been integrated into the [jdk9/hs-comp](#) forest from where it will be automatically integrated into all the other Java 9 forests. As a result our [ppc-aix-port/stage-9](#) forest has become obsolete and has been set to read-only mode. The [ppc-aix-port/jdk8](#) forest isn't needed either, because all the Java 8 changes have been reviewed and integrated into our [jdk7u/jdk7u-dev/](#) forest which is planned for integration into [jdk8u-dev](#) (right in time for the 8u20 release).

In [December 2013](#) we completed the C2 compiler and thus reached Milestone M3.1 of the [PowerPC/AIX Port Integration Plan](#). HotSpot can now be built as server VM (with-jvm-variants=server). We also already added the majority of the changes targeted for M3.2 improving stability and performance of the port.

As of [November 2013](#), we reached Milestone 2.3 of our [PowerPC/AIX Port Integration Plan](#). This means that we can now build and use the complete JDK from our staging repository on Linux as well as on AIX. The work on Milestone 3.1 "HotSpot changes for the PPC64 C2 Port" is progressing really well and we hope to have a C2 enabled VM by the end of the year.

<a href="#">blocked URL</a>	Meet the PowerPC/AIX Porting team at the <a href="#">JavaOne2013 BOF "OpenJDK Porting Experiences: The Good, the Bad, and the Downright Ugly [BOF4132]" (HTML-Slides of the talk)</a> .
-----------------------------	---

As of [September 2013](#), we reached Milestone 2.2 of our [PowerPC/AIX Port Integration Plan](#). This means that we can now build and use the (still interpreter-only) HotSpot VM from the staging repository on AIX. We're currently working on getting the whole JDK 8 in the staging repository to build with the new build system on AIX.

As of [August 2013](#), we reached Milestone 2.1 of our [PowerPC/AIX Port Integration Plan](#) (i.e. we now have a buildable and runnable, interpreter-only JDK8 in the staging repository).

As of [June 2013](#), we've got our JEP "[Integrate PowerPC/AIX Port into JDK 8](#)" funded. We agreed on a detailed [PowerPC/AIX Port Integration Plan](#) and created the staging repository <http://hg.openjdk.java.net/ppc-aix-port/stage/> which will be used to collect the officially reviewed changesets of our port while it is regularly synchronized with the current up-stream development repository. Once the complete port is in the staging repository, it will be bulk-integrated up-streams into the latest JDK development repository.

As of [May 2013](#), we have completed the [port of the new build system](#) to AIX.

As of [April 2013](#), the project has finalized its work on the Java 7 port. We've synchronized with the upstream <http://hg.openjdk.java.net/jdk7u> repository at tag 'jdk7u14-b14'. This particularly means that we now have a HotSpot 24 with the new JSR292 implementation in our JDK7 port. We also have a fully functional HotSpot 25 on Linux/PPC64 as well as on AIX in our <http://hg.openjdk.java.net/jdk8/hotspot> repository and on Linux/PPC64 the complete JDK8 can be build with the new build system. Currently we are focusing on completing the JDK8 port with the new build system on AIX.

As of [January 2013](#), the project has reached its third milestone. The port has been stabilize and it successfully passed all the Java SE 7 Test Compatibility Kit (TCK) tests on SLES 11.1 Linux/PPC64 and AIX 5.3/PPC64. This means that the port is now fully compliant with the Java SE 7 specification on the named platforms. We have also initiated and filed "[JEP 175 : Integrate PowerPC/AIX Port into JDK 8](#)". The next steps will be to synchronize our jdk7 repository with jdk7u-dev (we're currently still 7u6 based) and start working full steam on the integration of our changes into jdk8.

As of [December 2012](#), the project has reached its second milestone. A mixed-mode VM (with the C++ Interpreter and the C2 "Server" JIT compiler) is available on both Linux/PPC64 and AIX/PPC64. It successfully runs different benchmarks (JVM98, JBB2005 and DaCapo) and is able to bootstrap itself. On our transitional [ppc-aix-port project page](#) you can download precompiled binaries of this new version of our port which can be used as a bootstrap JDK during the build process. There's also a detailed [README-ppc.html](#) file available which describes the build process on the two porting platforms. The next steps will be to certify our ports on Linux/PPC64 and AIX/PPC64 and integrate them upstream into the main JDK7/8 repositories. Concurrently we are also working on improving the performance which still lags behind that of our closed implementations.

As of [July 2012](#), the project has reached its first milestone. An interpreter only version of the PowerPC port is available on Linux/PPC64. It successfully runs the JVM98 benchmark and is able to bootstrap itself. There's a transitional [ppc-aix-port project page](#) available which contains the nightly build logs of our project repositories on various platforms and a compiled version of the OpenJDK for Linux/PPC64 which can be used as a bootstrap JDK during the build process. The next steps will be to complete the build on AIX (currently only the HotSpot build is working on AIX) and start the port of the C2 server JIT compiler.

## Resources

Port Project	<a href="#">PowerPC/AIX Port</a>
Mailing list:	<a href="#">ppc-aix-port-dev</a>
JEP 175:	<a href="#">PowerPC/AIX Port</a>
Wiki:	<a href="#">PowerPC/AIX Port OpenJDK Wiki Space</a>
Repos:	<a href="https://github.com/openjdk/jdk/">https://github.com/openjdk/jdk/</a> <a href="https://github.com/openjdk/jdk11u-dev/">https://github.com/openjdk/jdk11u-dev/</a> <a href="http://hg.openjdk.java.net/jdk8u">http://hg.openjdk.java.net/jdk8u</a> <a href="http://hg.openjdk.java.net/jdk7u">http://hg.openjdk.java.net/jdk7u</a>

[Navigate space](#)

## Recently Updated

[PowerPC/AIX Port](#)

May 02, 2024 • updated by [Martin Doerr](#) • [view change](#)

[PowerPC/AIX Port](#)

Apr 30, 2024 • updated by [Goetz Lindenmaier](#) • [view change](#)

[Architecture of the OpenJDK PPC Port](#)

Apr 21, 2021 • updated by [Goetz Lindenmaier](#) • [view change](#)

[JDK 9 & 10 and the PowerPC/AIX port](#)

Jan 14, 2019 • updated by [Goetz Lindenmaier](#) • [view change](#)

[JDK 8 and the PowerPC/AIX port](#)

Jan 09, 2019 • updated by [Goetz Lindenmaier](#) • [view change](#)

[JDK 7 and the PowerPC/AIX port](#)

Jan 09, 2019 • updated by [Goetz Lindenmaier](#) • [view change](#)

[Architecture of the OpenJDK PPC Port](#)

Dec 16, 2015 • updated by [Thomas Stuefe](#) • [view change](#)

[PowerPC/AIX Port](#)

Dec 17, 2014 • updated by [Volker Simonis](#) • [view change](#)

[C2 compiler extensions and new optimizations](#)

Oct 01, 2014 • updated by [Goetz Lindenmaier](#) • [view change](#)

[PowerPC/AIX Port Integration Plan](#)

Apr 15, 2014 • updated by [Volker Simonis](#) • [view change](#)

[PowerPC/AIX Port Integration Plan](#)

Feb 05, 2014 • updated by [Azeem Jiva](#) • [view change](#)

[PowerPC/AIX Port Integration Plan](#)

Jan 28, 2014 • updated by [Goetz Lindenmaier](#) • [view change](#)

[PowerPC/AIX Port Integration Plan](#)

Jul 15, 2013 • updated by [Iris Clark](#) • [view change](#)

[C++ interpreter features](#)

May 03, 2013 • updated by [Goetz Lindenmaier](#) • [view change](#)

[PowerPC/AIX Port](#)

Apr 19, 2013 • updated by [Brian Beck](#) • [view change](#)