

JEP 422: Linux/RISC-V Port

<i>Author</i>	fyang
<i>Owner</i>	Fei Yang
<i>Type</i>	Feature
<i>Scope</i>	Implementation
<i>Status</i>	Closed / Delivered
<i>Release</i>	19
<i>Component</i>	hotspot
<i>Discussion</i>	riscv dash port dash dev at openjdk dot java dot net
<i>Effort</i>	L
<i>Duration</i>	S
<i>Reviewed by</i>	Aleksey Shipilev
<i>Endorsed by</i>	Vladimir Kozlov
<i>Created</i>	2021/11/08 08:11
<i>Updated</i>	2023/05/12 12:47
<i>Issue</i>	8276797

Summary

Port the JDK to Linux/RISC-V.

Motivation

[RISC-V](#) is a free and open-source RISC instruction set architecture (ISA) designed originally at the University of California, Berkeley, and now developed collaboratively under the sponsorship of [RISC-V International](#). It is already supported by a wide range of language toolchains. With the increasing availability of RISC-V hardware, a port of the JDK would be valuable.

Description

The RISC-V ISA is actually a family of related ISAs, of which there are currently four base ISAs. Each base ISA can be configured with one or more standard extensions.

The Linux/RISC-V port will only support the RV64GV configuration of RISC-V, which is a general-purpose 64-bit ISA that includes vector instructions. In the future we may consider supporting other RISC-V configurations such as, for example, a general-purpose 32-bit configuration (RV32G).

The port will support the following HotSpot subsystems:

- The template interpreter,
- The C1 (client) JIT compiler,
- The C2 (server) JIT compiler, and
- All current mainline GCs, including ZGC and Shenandoah.

The focus of this JEP is not the porting effort itself, which is nearly complete, but rather the integration of the port into the JDK main-line repository.

Currently we have a number of trivial changes in the shared part of JDK. These are mostly `#ifdefs` that include the relevant platform-specific files. There are a few changes of other kinds, but again these are guarded by `#ifdef RISCV`. Thus, the risk to other platforms is low.

There are also changes to the build machinery of the JDK to add appropriate definitions for endianness, word size, and so on. Again, these should not affect other platforms.

The current diff between JDK mainline and RISC-V port project can be obtained by:

```
$ git clone https://github.com/openjdk/riscv-port riscv-port
$ cd riscv-port
$ git diff master..riscv-port
```

An automatically generated webrev for that difference is available [here](#).

Testing

The port passes jtregh tiers 1–4 and jcstress tests on a HiFive Unmatched development board. We also regularly carry out SPECjbb and SPECjvm benchmark tests.

Huawei Technologies, Alibaba, and Red Hat regularly build and test this port on Linux/RISC-V as well as on the currently-supported JDK platforms to ensure that no regressions are introduced which break the existing platforms.

Huawei Technologies is committed to support fully (i.e., regularly update, enhance, and test) the code introduced by this effort.

Risks and Assumptions

There are potentially some issues with the availability of RISC-V hardware for this work, but we hope that it will become widely available in the near future.