SPB2832LV

vmware® EXPLORE

SpringOne

Scaling Your Spring Boot App to Zero

From Hero to Zero

DaShaun Carter, Spring Developer Advocate, VMware

Thomas Risberg, Staff Engineer, VMware

#vmwareexplore #SPB2832LV

Required Disclaimer

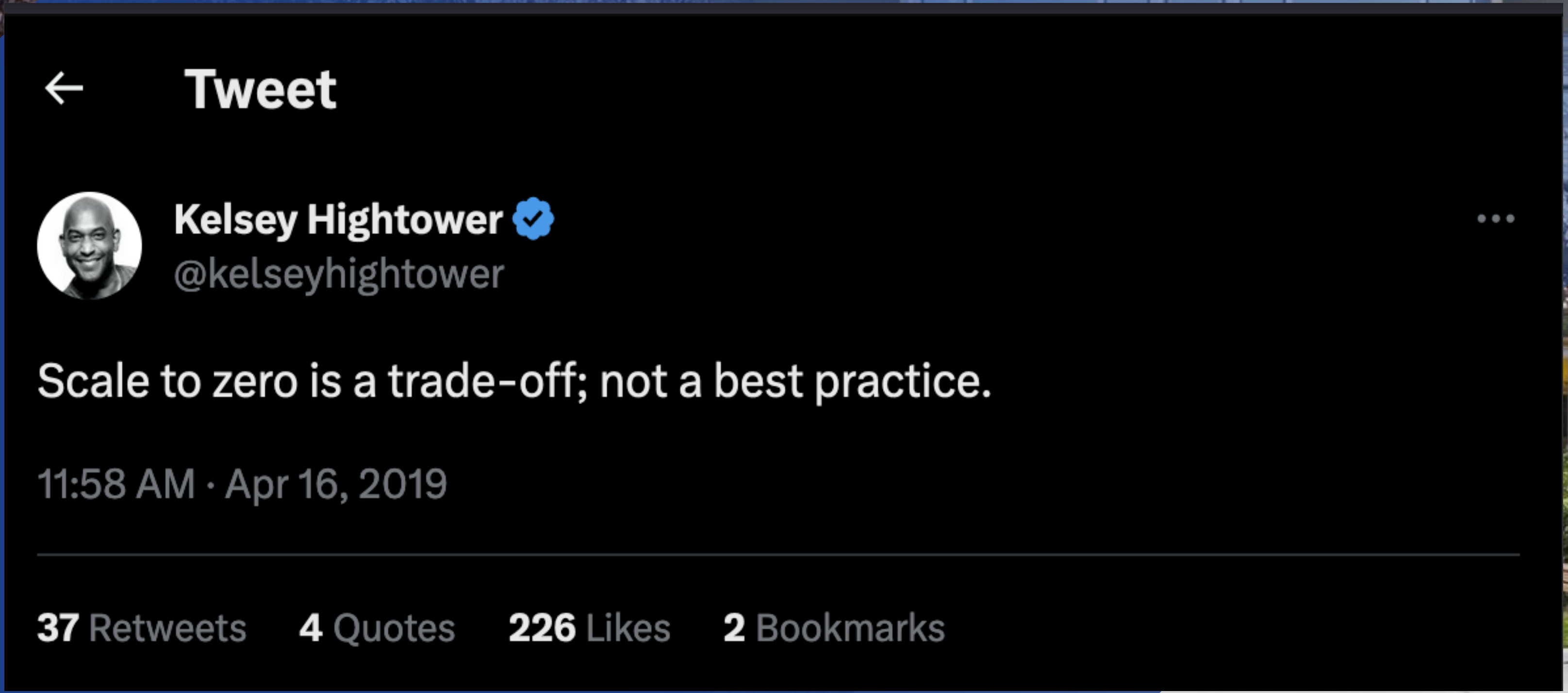
This presentation may contain product features or functionality that are currently under development.

This overview of new technology represents no commitment from VMware to deliver these features in any generally available product.

Features are subject to change, and must not be included in contracts, purchase orders, or sales agreements of any kind.

Technical feasibility and market demand will affect final delivery.

Pricing and packaging for any new features/functionality/technology discussed or presented, have not been determined.



Tweet

Kelsey Hightower
@kelseyhightower

Scale to zero is a trade-off; not a best practice.

11:58 AM · Apr 16, 2019

37 Retweets 4 Quotes 226 Likes 2 Bookmarks

https://twitter.com/kelseyhightower/status/1118181786914136064

Options for scaling apps to zero on Kubernetes

- Knative

  https://knative.dev

- KEDA

  https://keda.sh

Options for reducing startup time for Spring Boot apps

- Native compile with GraalVM

  https://www.graalvm.org
- Coordinated Restore at Checkpoint

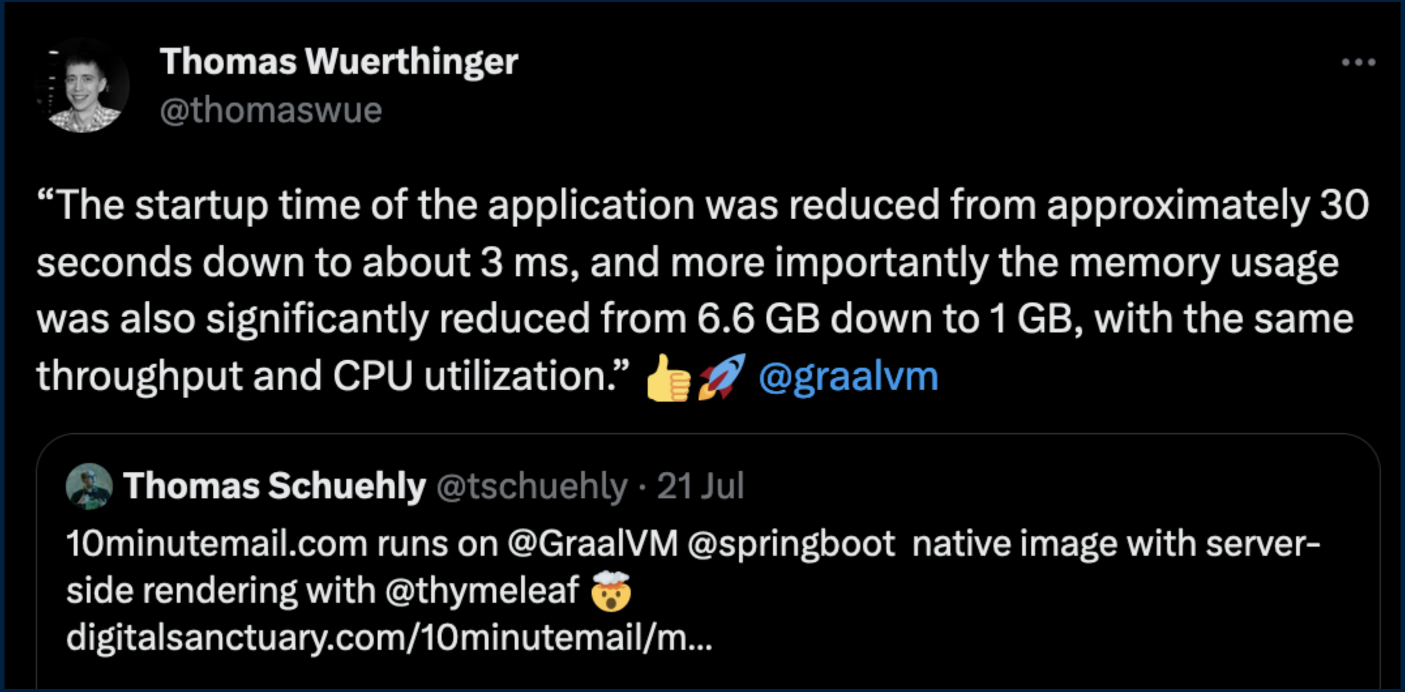
  https://openjdk.org/projects/crac

Native builds with GraalVM

Native builds with GraalVM

- Well supported since Spring Framework 6.0 and Spring Boot 3.0
- Continuously improved in collaboration with GraalVM team at Oracle

Why?

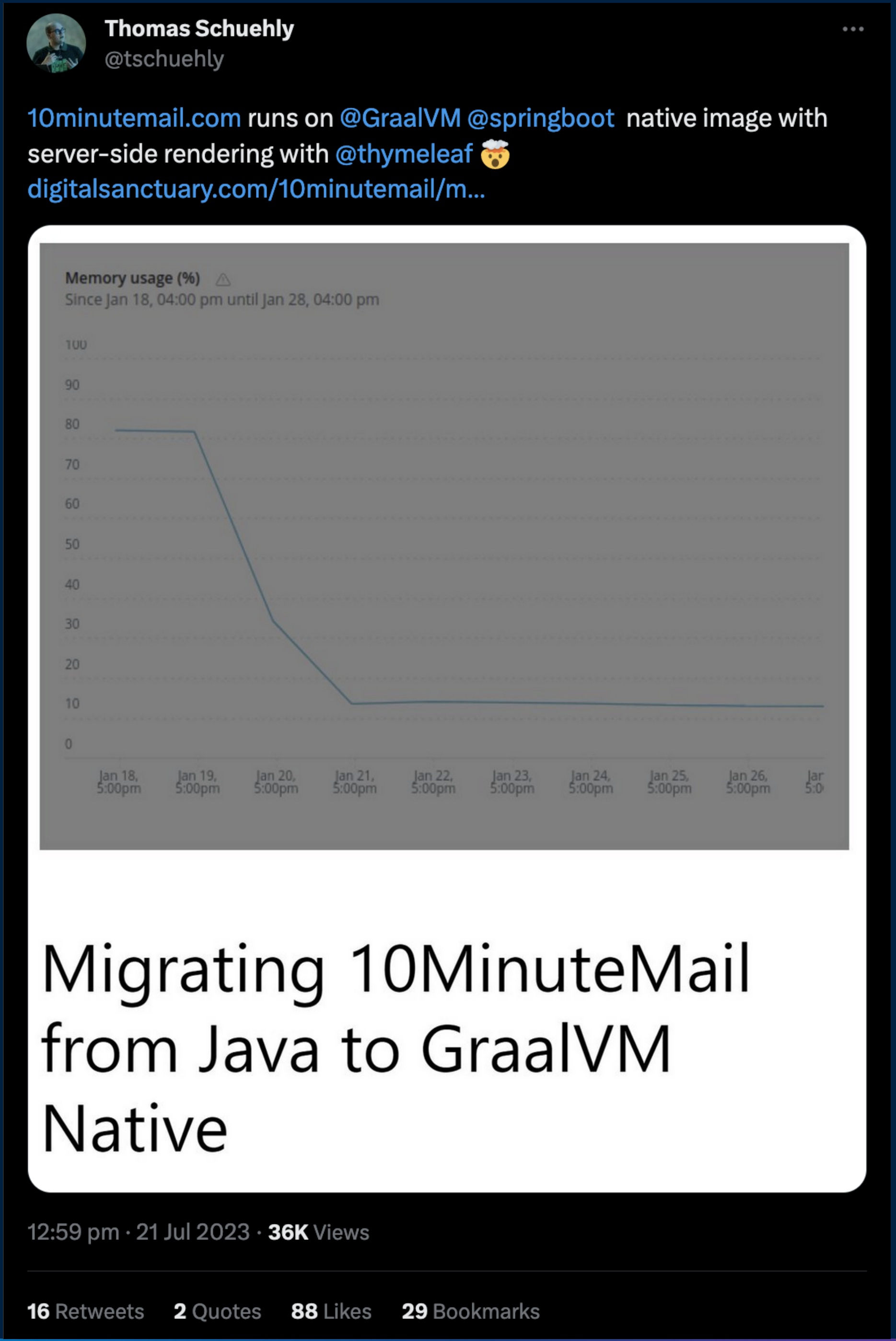


Thomas Wuerthinger
@thomaswue

"The startup time of the application was reduced from approximately 30 seconds down to about 3 ms, and more importantly the memory usage was also significantly reduced from 6.6 GB down to 1 GB, with the same throughput and CPU utilization." 👍🚀 @graalvm

Thomas Schuehly @tschuehly · 21 Jul
10minutemail.com runs on @GraalVM @springboot native image with server-side rendering with @thymeleaf 🤯
digitalsanctuary.com/10minutemail/m...

https://twitter.com/thomaswue/status/1682465475748298755



Thomas Schuehly
@tschuehly

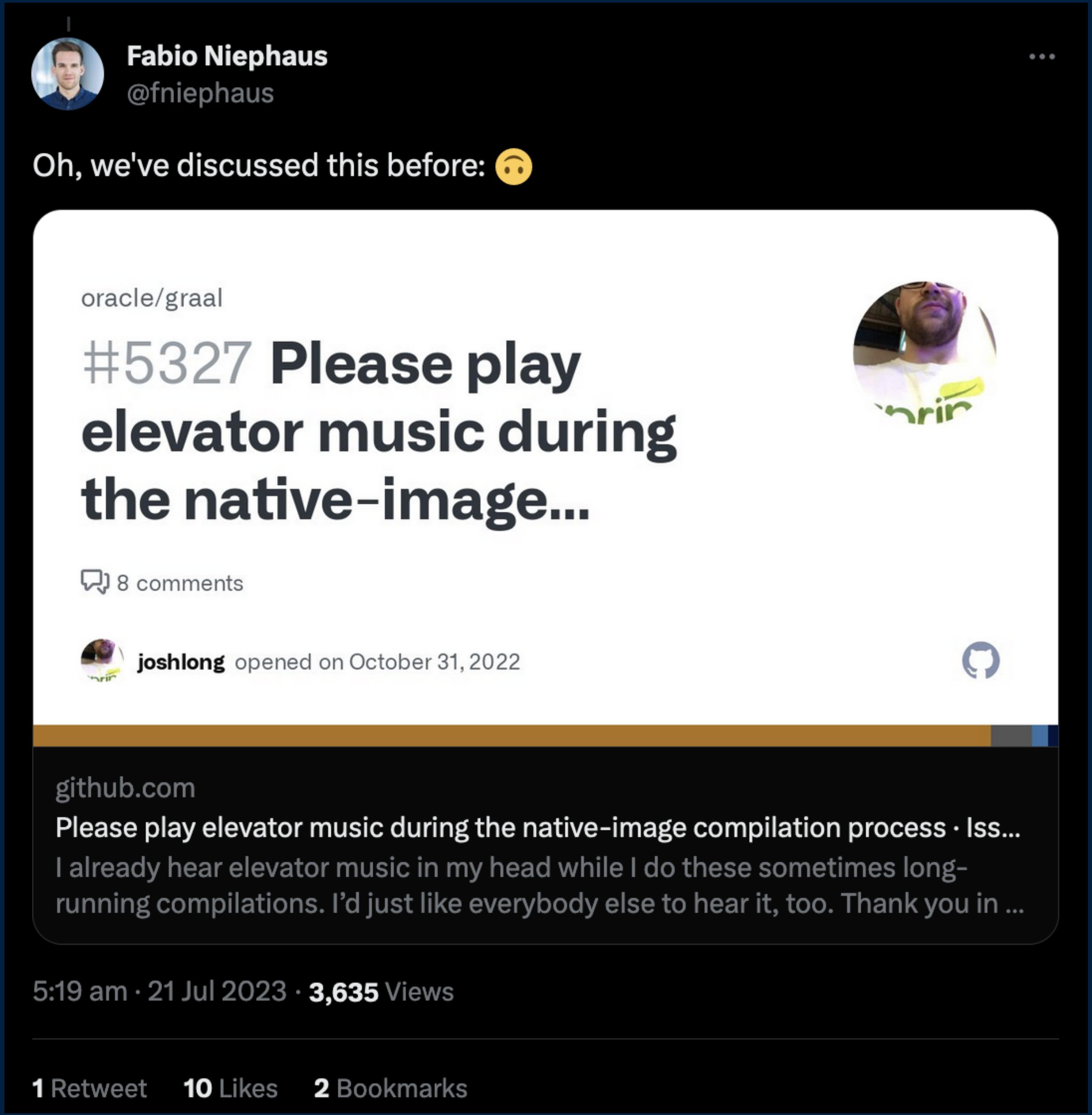
10minutemail.com runs on @GraalVM @springboot native image with server-side rendering with @thymeleaf 🤯
digitalsanctuary.com/10minutemail/m...

Migrating 10MinuteMail from Java to GraalVM Native

12:59 pm · 21 Jul 2023 · **36K** Views

16 Retweets **2** Quotes **88** Likes **29** Bookmarks

Why not?



Fabio Niephaus
@fniephaus

Oh, we've discussed this before: 🙃

oracle/graal

#5327 Please play elevator music during the native-image...

8 comments

joshlong opened on October 31, 2022

github.com

Please play elevator music during the native-image compilation process · Iss...

I already hear elevator music in my head while I do these sometimes long-running compilations. I'd just like everybody else to hear it, too. Thank you in ...

5:19 am · 21 Jul 2023 · **3,635** Views

1 Retweet **10** Likes **2** Bookmarks

https://twitter.com/fniephaus/status/1682319334901727232

Key Differences with JVM Deployments

The main differences building GraalVM Native Images vs regular JVM builds are:

- Static analysis performed at build-time
- Code that cannot be reached when the native image is created won’t be part of the executable
- GraalVM is not aware of dynamic elements like reflection, resources, serialization, and dynamic proxies
- The application classpath is fixed at build time
- There is no lazy class loading

Trade-offs: Known Limitations

- GraalVM native images are an evolving technology and not all libraries provide support. The GraalVM community is helping by providing reachability metadata for projects that don't yet ship their own.
- Spring itself relies on the reachability metadata project.
- If you encounter problems please check the Spring Boot with GraalVM page of the Spring Boot wiki.
- You can also contribute issues to the spring-aot-smoke-tests project on GitHub which is used to confirm that common application types are working as expected.
- If you find a library which doesn't work with GraalVM, please raise an issue on the reachability metadata project.
  - https://github.com/oracle/graalvm-reachability-metadata

Trade-offs: Example of runtime config needed at build

TAP workload shows build env set to same as deployment env

```
apiVersion: carto.run/v1alpha1
kind: Workload
metadata:
  name: customer-profile
  labels:
    apps.tanzu.vmware.com/workload-type: web
    app.kubernetes.io/part-of: customer-profile
    apps.tanzu.vmware.com/has-tests: "true"
    apps.tanzu.vmware.com/auto-configure-actuators: "true"
spec:
  build:
    env:
    - name: BP_JVM_VERSION
      value: "17"
    - name: BP_NATIVE_IMAGE
      value: "true"
    - name: BP_MAVEN_ACTIVE_PROFILES
      value: "native"
    - name: SERVICE_BINDING_ROOT
      value: /bindings
    - name: ORG_SPRINGFRAMEWORK_CLOUD_BINDINGS_BOOT_ENABLE
      value: "true"
    - name: MANAGEMENT_HEALTH_PROBES_ENABLED
      value: "true"
    - name: MANAGEMENT_ENDPOINT_HEALTH_PROBES_ADD_ADDITIONAL_PATHS
      value: "true"
    - name: MANAGEMENT_ENDPOINT_HEALTH_SHOW_DETAILS
      value: always
    - name: MANAGEMENT_ENDPOINTS_WEB_BASE_PATH
      value: /actuator
    - name: MANAGEMENT_ENDPOINTS_WEB_EXPOSURE_INCLUDE
      value: '*'
...
```

OpenJDK with **CRaC** (Coordinated Restore at Checkpoint)

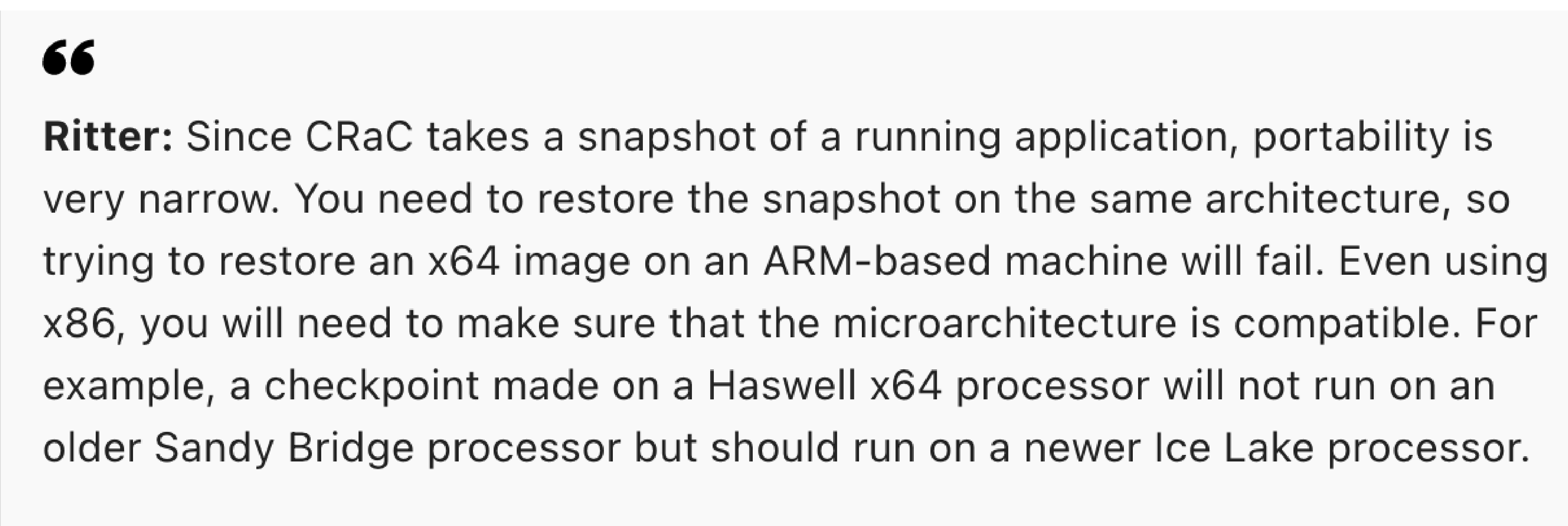
JVM Checkpoint/Restore for OpenJDK with CRaC

- Experimental support in Spring Framework 6.1 and Spring Boot 3.2
- Limited support of features available in Spring eco system
- Any open connections or files must be closed during checkpoint

Trade-offs: Portability Limitations

InfoQ interview with Simon Ritter from Azul

https://www.infoq.com/news/2023/06/crac-cracks-mainstream-adoption/



“

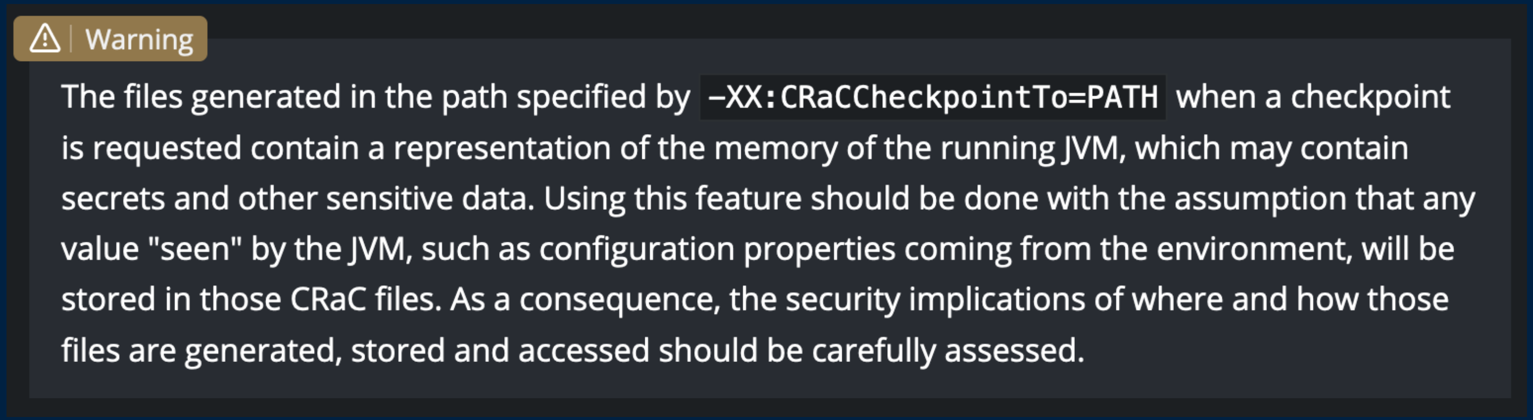
Ritter: Since CRaC takes a snapshot of a running application, portability is very narrow. You need to restore the snapshot on the same architecture, so trying to restore an x64 image on an ARM-based machine will fail. Even using x86, you will need to make sure that the microarchitecture is compatible. For example, a checkpoint made on a Haswell x64 processor will not run on an older Sandy Bridge processor but should run on a newer Ice Lake processor.

Trade-offs: Framework Support Limitations

Spring Framework 6.1 - JVM Checkpoint Restore

Using this feature requires:

- A checkpoint/restore enabled JVM (Linux only for now).
- The presence in the classpath of the org.crac:crac library.
- Specifying the required java command line parameters like -XX:CRaCCheckpointTo=PATH or -XX:CRaCRestoreFrom=PATH.
- Conceptually, checkpoint and restore match with Spring Lifecycle contract for individual beans.



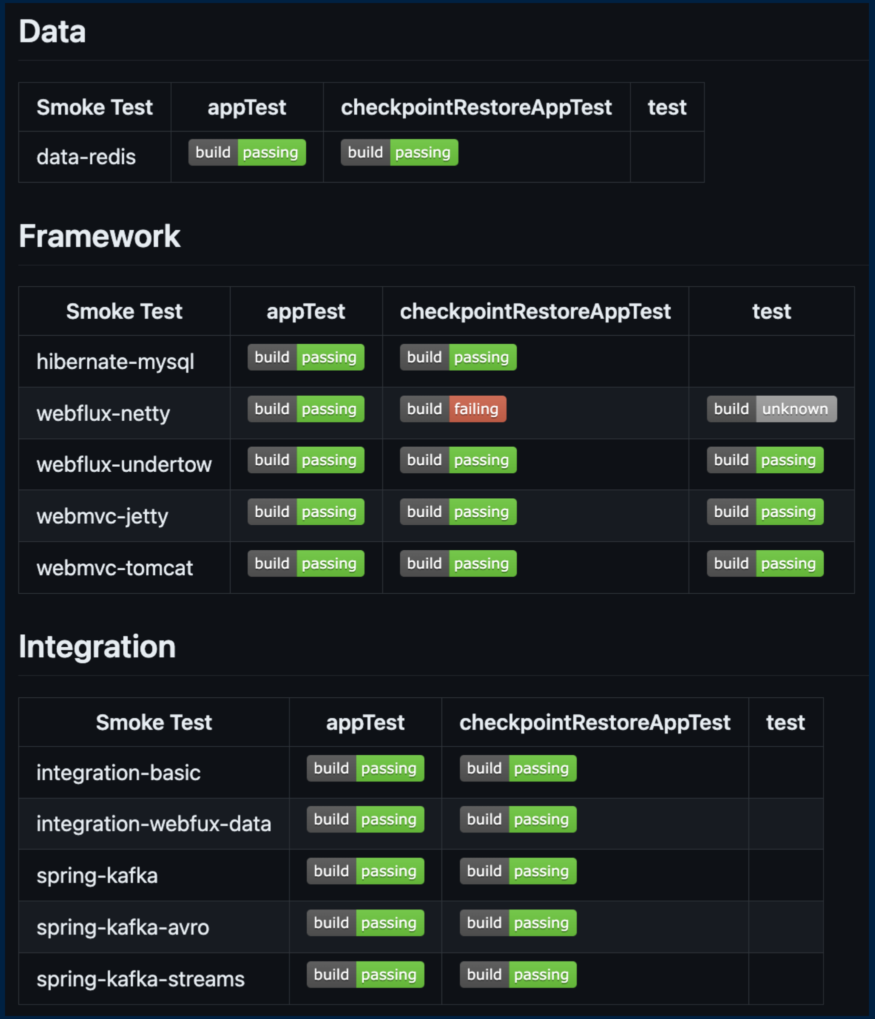
⚠ | Warning

The files generated in the path specified by `-XX:CRaCCheckpointTo=PATH` when a checkpoint is requested contain a representation of the memory of the running JVM, which may contain secrets and other sensitive data. Using this feature should be done with the assumption that any value "seen" by the JVM, such as configuration properties coming from the environment, will be stored in those CRaC files. As a consequence, the security implications of where and how those files are generated, stored and accessed should be carefully assessed.

Trade-offs: Limited Eco System Support

Any technology requiring open connections need to add lifecycle support

- The number of supported Spring projects/technologies is growing …



Data

Smoke Test	appTest	checkpointRestoreAppTest	test
data-redis	build passing	build passing	

Framework

Smoke Test	appTest	checkpointRestoreAppTest	test
hibernate-mysql	build passing	build passing	
webflux-netty	build passing	build failing	build unknown
webflux-undertow	build passing	build passing	build passing
webmvc-jetty	build passing	build passing	build passing
webmvc-tomcat	build passing	build passing	build passing

Integration

Smoke Test	appTest	checkpointRestoreAppTest	test
integration-basic	build passing	build passing	
integration-webflux-data	build passing	build passing	
spring-kafka	build passing	build passing	
spring-kafka-avro	build passing	build passing	
spring-kafka-streams	build passing	build passing	

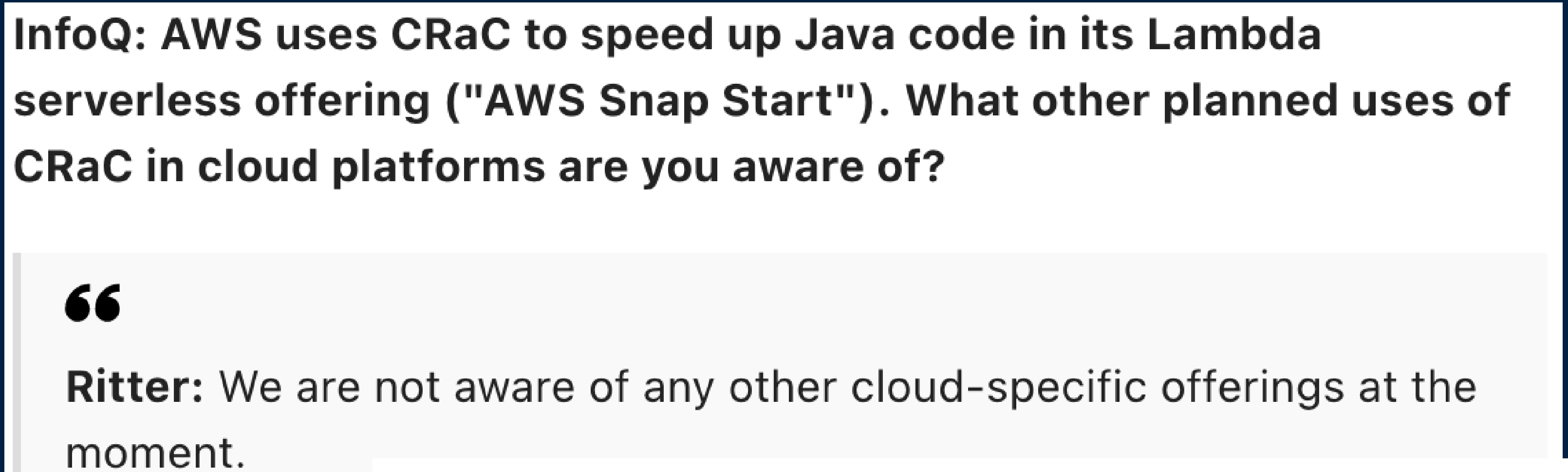
https://github.com/spring-projects/spring-checkpoint-restore-smoke-tests/blob/main/STATUS.adoc

Trade-offs: Limited Cloud Platform Support

Best practices?

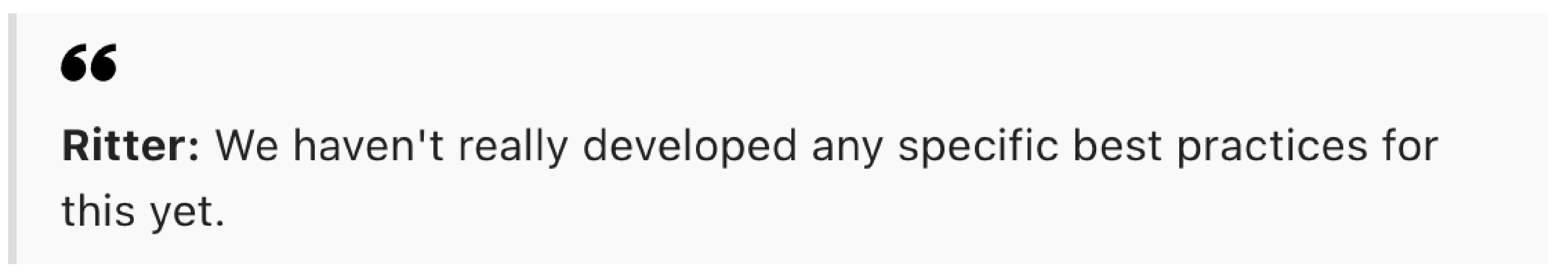
InfoQ interview with Simon Ritter from Azul

https://www.infoq.com/news/2023/06/crac-cracks-mainstream-adoption/



InfoQ: AWS uses CRaC to speed up Java code in its Lambda serverless offering ("AWS Snap Start"). What other planned uses of CRaC in cloud platforms are you aware of?

> **Ritter:** We are not aware of any other cloud-specific offerings at the moment.



> **Ritter:** We haven't really developed any specific best practices for this yet.

Exploring best practices

Will evolve as we learn more

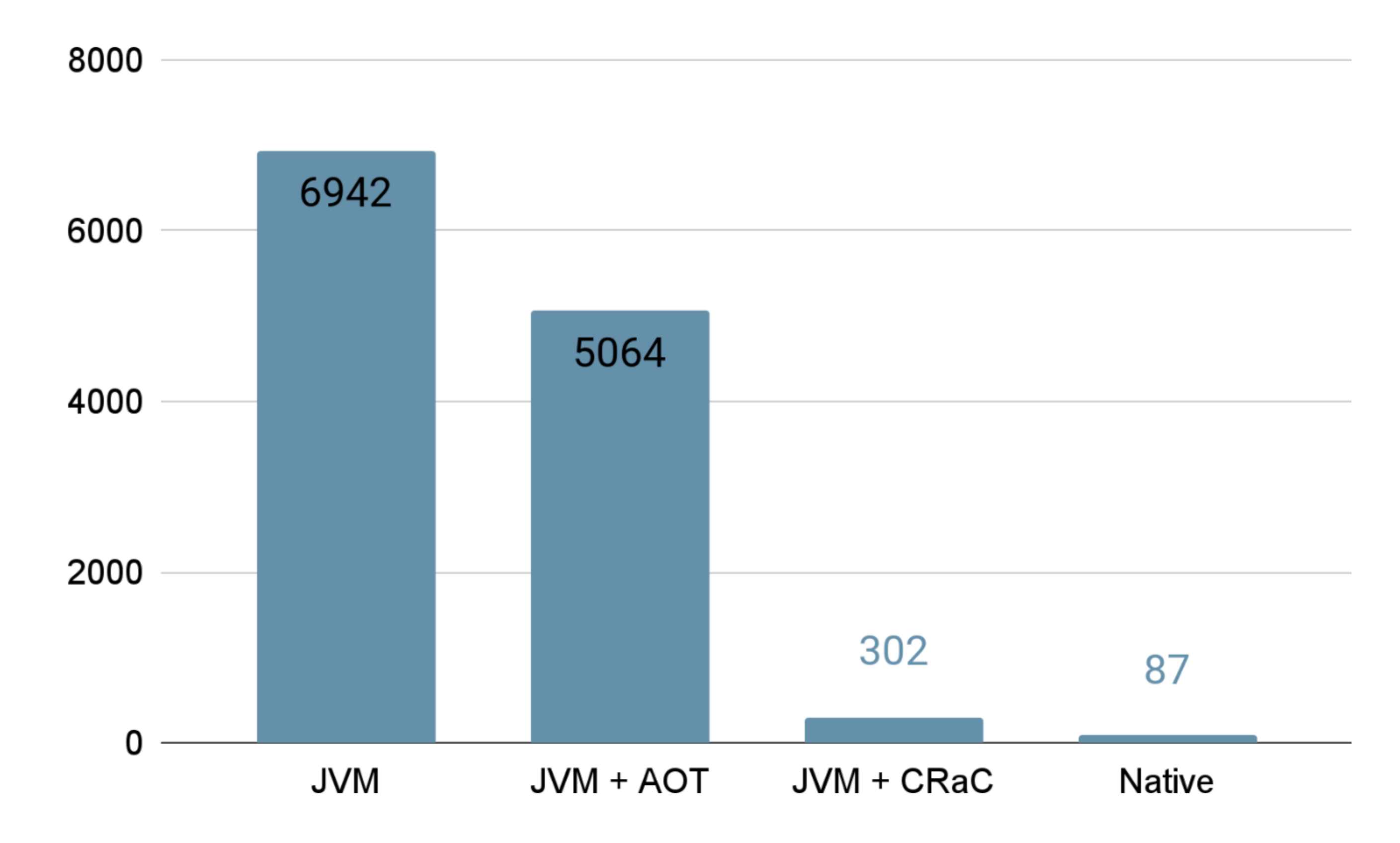
When do we take the checkpoint for apps running in Kubernetes?

- Checkpoint on demand on first start
  - ✓ write files to a directory per node
- Checkpoint on demand with a separate job
  - ✓ needs uniform nodes or node affinity to select compatible nodes
- Checkpoint automatically on startup
  - ✓ use `-Dspring.context.checkpoint=onRefresh`
  - ✓ not a fully warmed-up JVM
  - ✓ write files to directory per node

General Scale to Zero Considerations

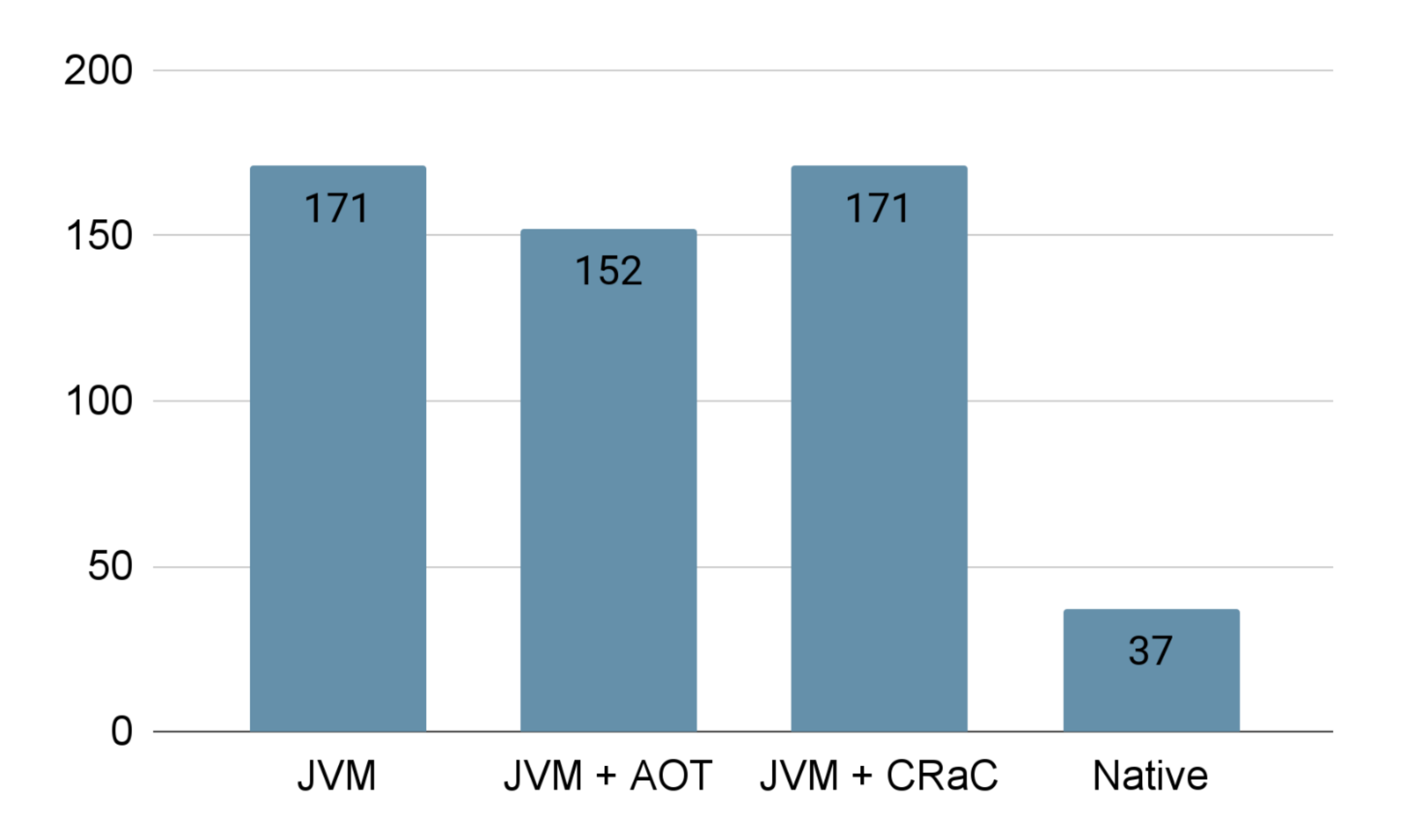
Container start to application ready (milliseconds)

Webapp on Azure Container Apps with 1 CPU 2G memory

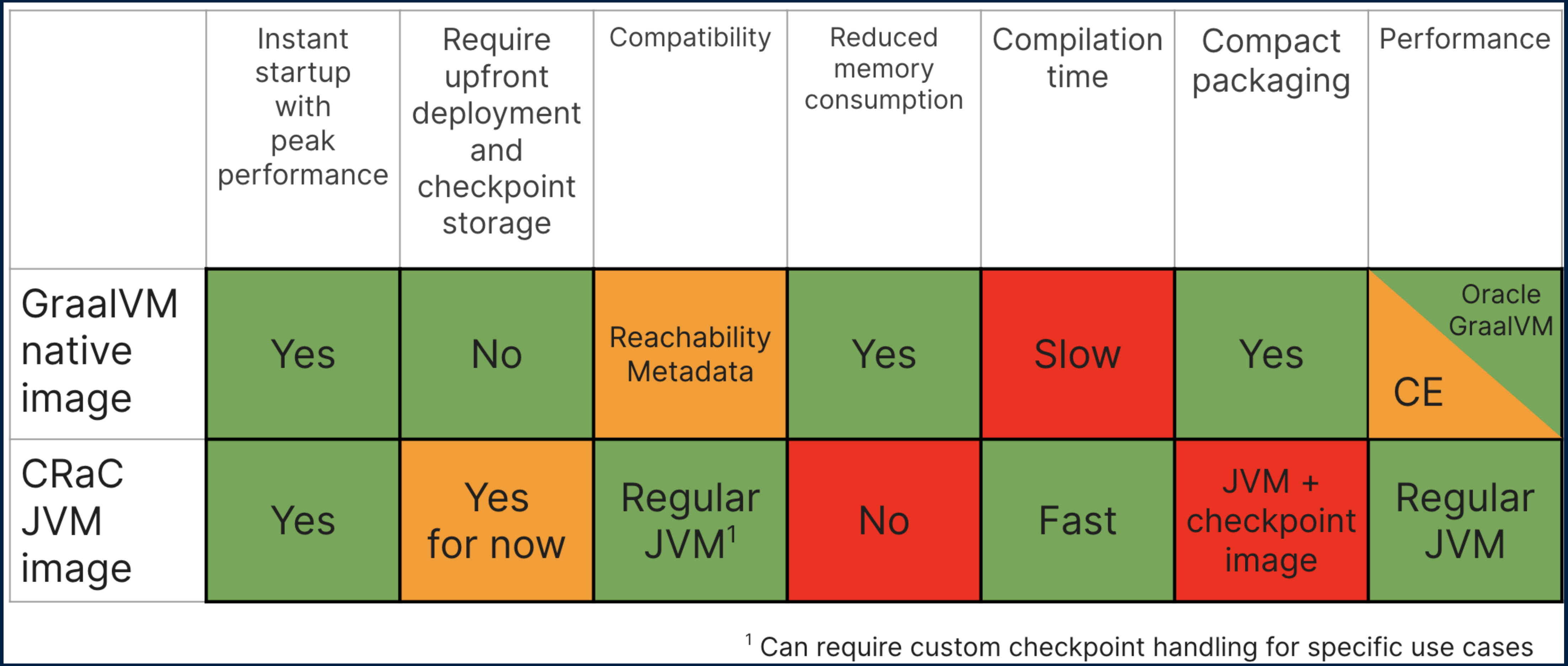


Memory Working Set Bytes (Megabytes)

Webapp on Azure Container Apps with 1 CPU 2G memory



Trade-offs: Summary



	Instant startup with peak performance	Require upfront deployment and checkpoint storage	Compatibility	Reduced memory consumption	Compilation time	Compact packaging	Performance
GraalVM native image	Yes	No	Reachability Metadata	Yes	Slow	Yes	Oracle GraalVM CE
CRaC JVM image	Yes	Yes for now	Regular JVM[1]	No	Fast	JVM + checkpoint image	Regular JVM

[1] Can require custom checkpoint handling for specific use cases

SpringOne at VMware Explore 2023

Notes and demos :: Scaling Your Spring Boot App to Zero

https://github.com/trisberg/springone-explore-2023

Tanzu Academy :: Developer Sandbox

A sandbox cluster with Tanzu Application Platform ready to use

https://tanzu.academy/guides/developer-sandbox

vmware EXPLORE

SpringOne

Stay Connected

Let’s continue the conversation

Follow us on Twitter @dashaun @trisberg

Visit us @ https://spring.io

Dig deeper:

https://github.com/trisberg/springone-explore-2023

vmware EXPLORE

SpringOne

Please take your survey.

vmware EXPLORE

SpringOne

Thank you