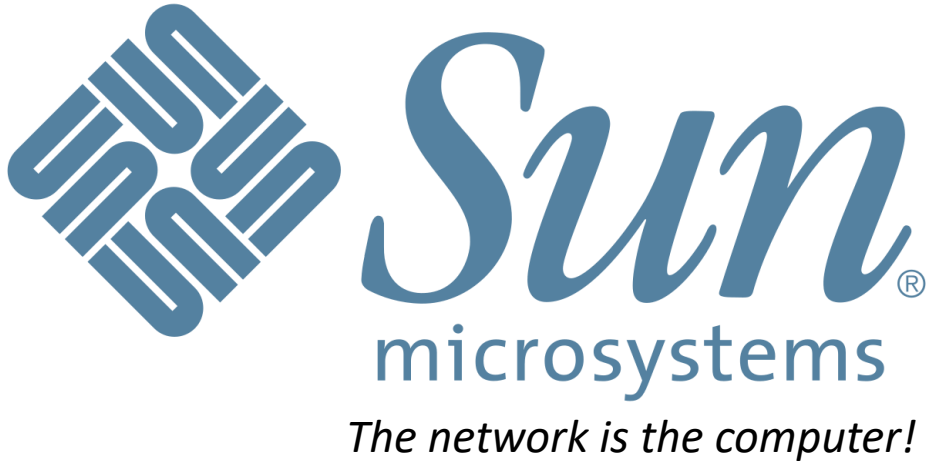
GraalVM
How We Turned Our Research into a Product

Thomas Wuerthinger
GraalVM Founder and Project Lead
VP at Oracle
https://thomaswue.dev

Educational Background

- PhD from JKU Linz university in Austria
- Sun Microsystems collaboration
- Sun Labs internships in the Maxine RVM team
- Google internship in the V8 JavaScript team
- In 2011: Oracle or Google? => Oracle Labs



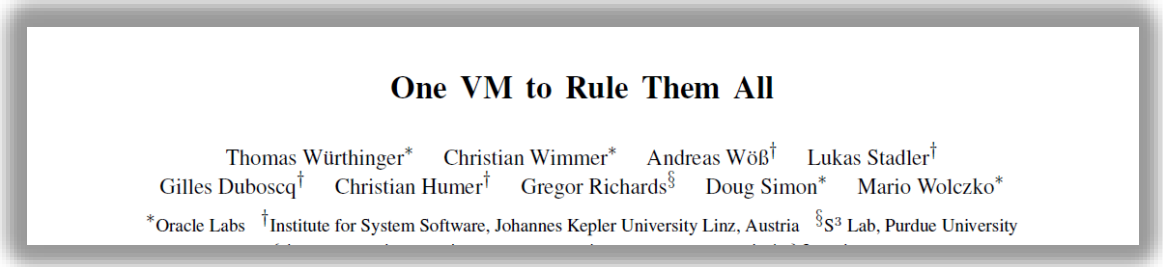
- Moved through all stages: researcher, manager, director, VP

Java History

- Based on programming language research started in 1985 at Sun Microsystems under the name “Oak”
- Renamed and published as Java 1.0 in 1996

- Current estimate: 60 billion Java Virtual Machines running (https://www.oracle.com/java)

- **2013**
  Vision Paper

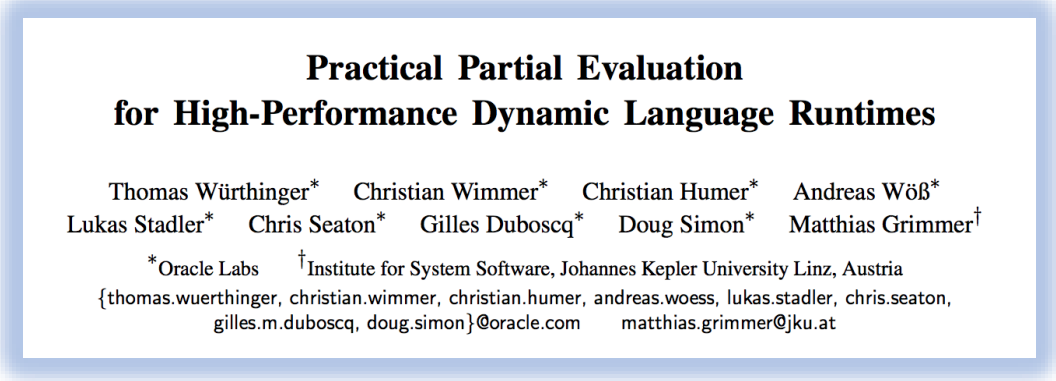


One VM to Rule Them All

Thomas Würthinger* Christian Wimmer* Andreas Wöß† Lukas Stadler†
Gilles Duboscq† Christian Humer† Gregor Richards§ Doug Simon* Mario Wolczko*

*Oracle Labs †Institute for System Software, Johannes Kepler University Linz, Austria §S3 Lab, Purdue University

- **2017**
  Result Paper



Practical Partial Evaluation
for High-Performance Dynamic Language Runtimes

Thomas Würthinger* Christian Wimmer* Christian Humer* Andreas Wöß*
Lukas Stadler* Chris Seaton* Gilles Duboscq* Doug Simon* Matthias Grimmer†

*Oracle Labs †Institute for System Software, Johannes Kepler University Linz, Austria
{thomas.wuerthinger, christian.wimmer, christian.humer, andreas.woess, lukas.stadler, chris.seaton, gilles.m.duboscq, doug.simon}@oracle.com matthias.grimmer@jku.at

- **2018**
  Open Source Launch

GraalVM™

- **2019**
  Commercial Launch

ORACLE®
GraalVM

A Just-In-Time Compiler For Java in Java!

- Security
  - "NSA advises organizations to consider making a strategic shift from programming languages that provide little or no inherent memory protection, such as C/C++, to a memory safe language when possible."

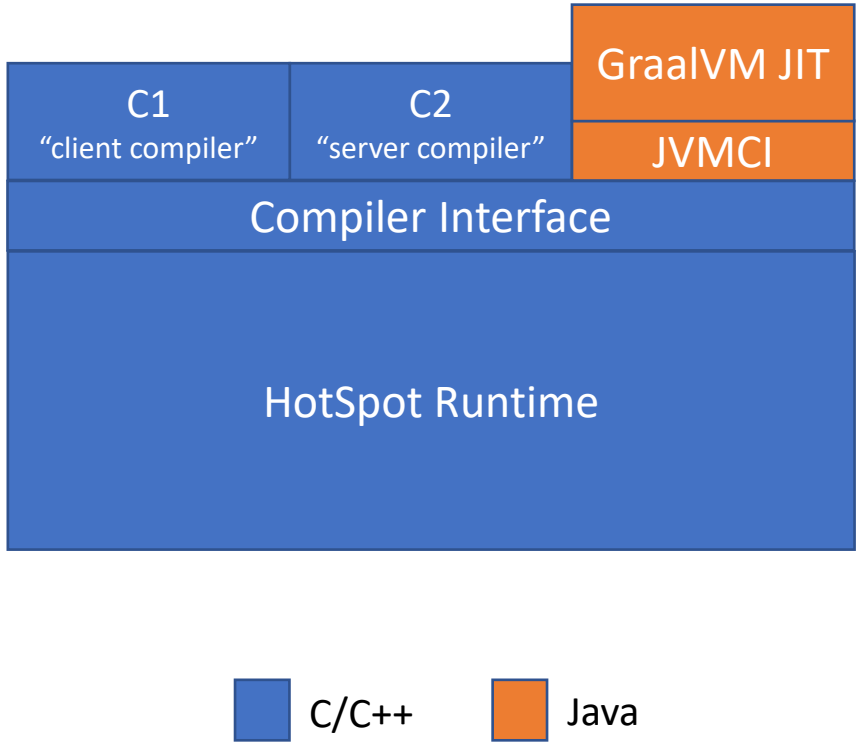
  https://media.defense.gov/2022/Nov/10/2003112742/-1/-1/0/CSI_SOFTWARE_MEMORY_SAFETY.PDF
- Robustness

  try { … } catch(Exception e) { … }
- Developer productivity

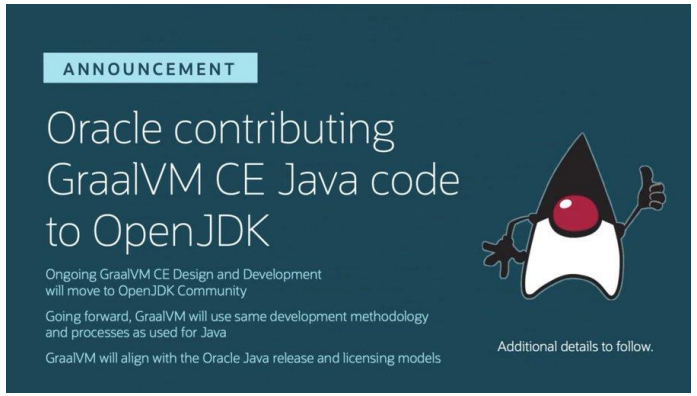
  Modern IDEs, refactoring, style and bug checking
- Dogfooding

  The team writing the Java JIT is programming Java everyday.

HotSpot VM with GraalVM JIT Compiler



Recent announcement at JavaOne 2022



What We Learned about Just-in-Time (JIT)
The good, the bad, the ugly

- Great for maximum throughput
- Not ideal for footprint
- Terrible for startup

Let’s go Ahead-of-Time (AOT)!

GraalVM™

JIT

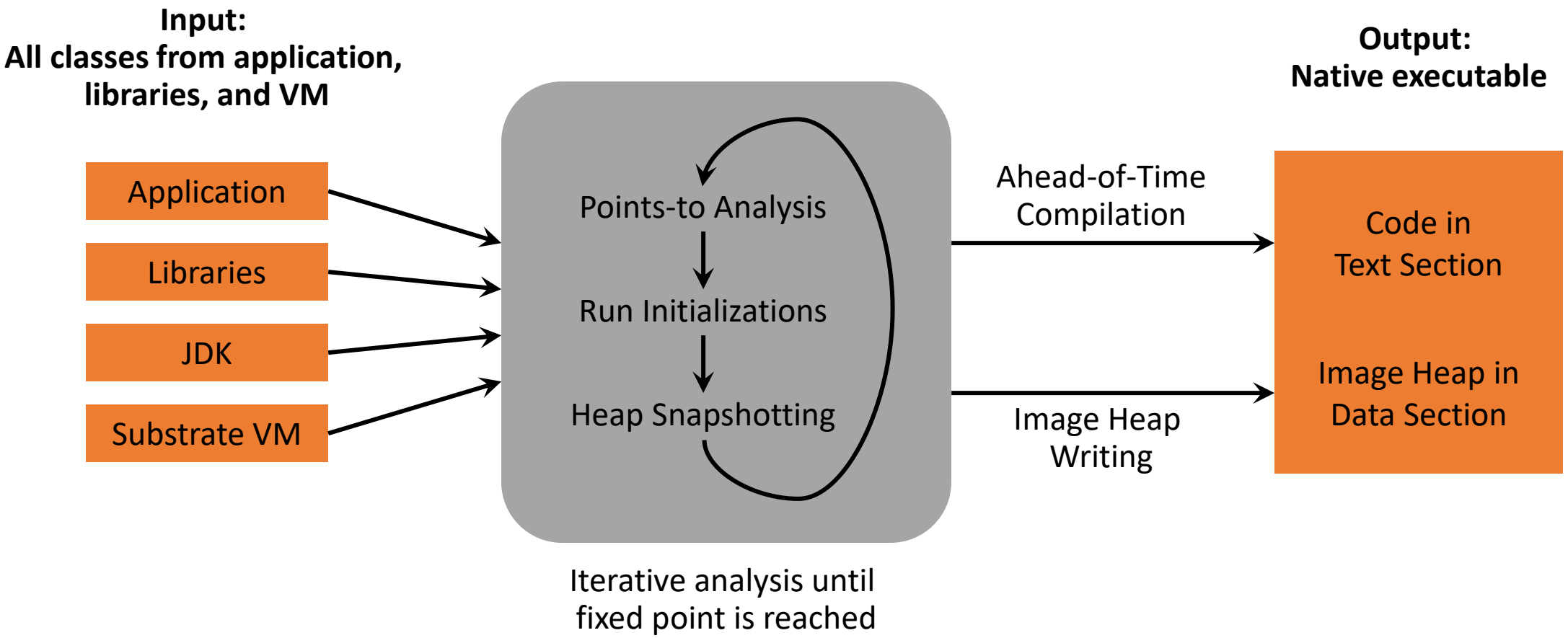
java MyMainClass

OpenJDK™

AOT

native-image MyMainClass
./mymainclass

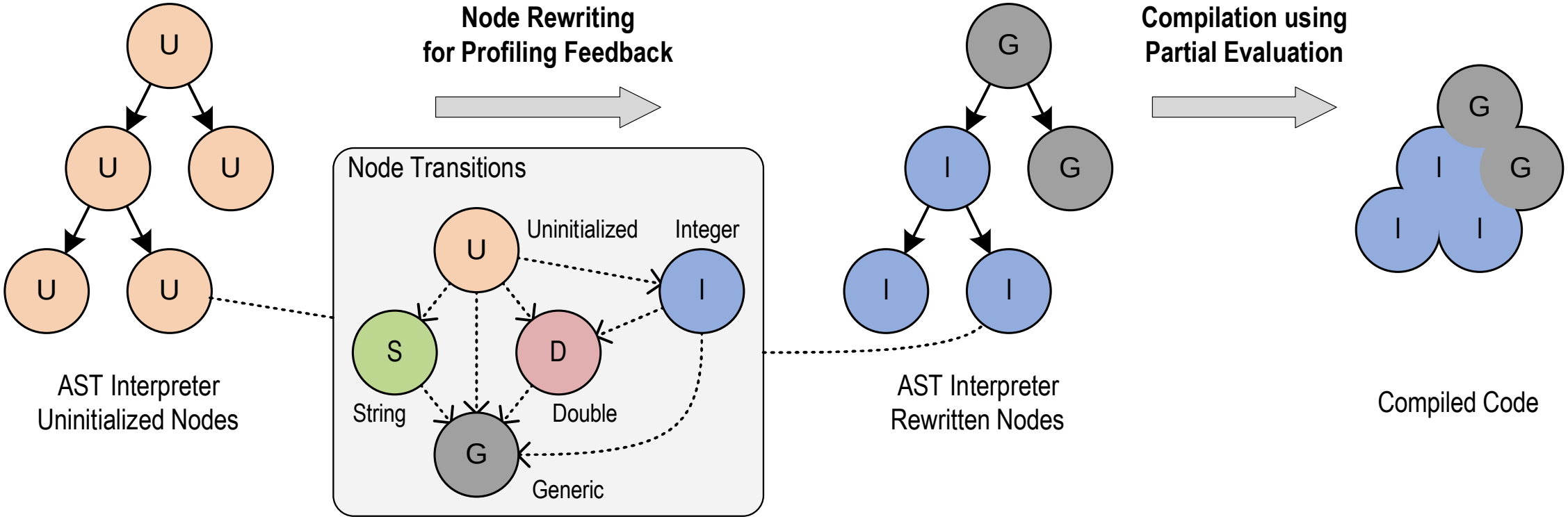
Native Image Architecture

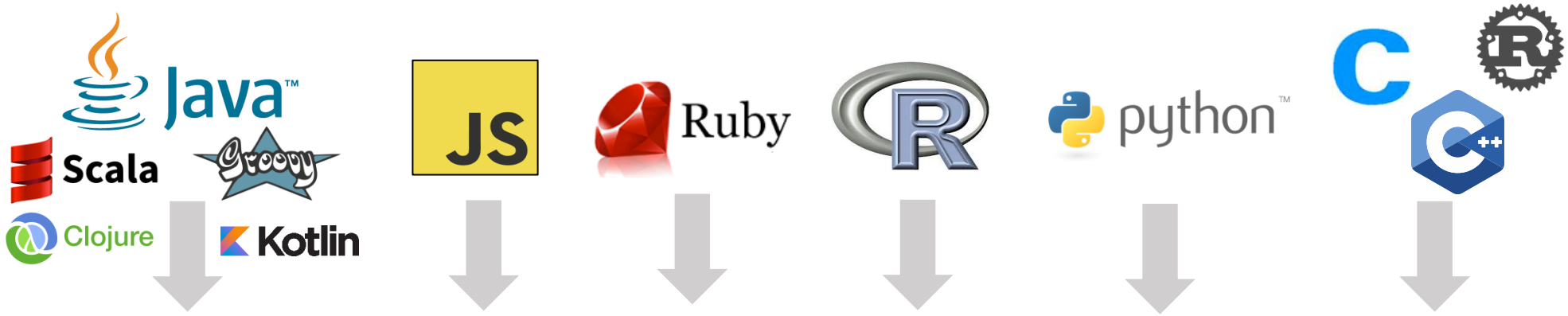


Polyglot


python
TS
R
Java
Swift
HTML
CSS
BASH
THE BOURNE-AGAIN SHELL
[OBJ-C]
POWERED BY
Perl
php

Partial Evaluation of Interpreters





Automatic transform of interpreters to compiler

GraalVM™

Embeddable in native or managed applications

What We Learned about Polyglot

- Primary use case: sandboxing and embedding
- Important internal users

- Great for platforms
- 11 NOV 2022
  GraalVM: running C/C++ application safely in the Java world
  https://www.adyen.com/blog/graalvm-running-c-applications--in-the-cloud

GraalVM Native Image for Running Java Applications in the Cloud

- Startup
  - Instant!
- Memory Footprint
  - Up to 2-5x smaller!
- Security
  - Reduced attack surface
- Packaging
  - Compact docker

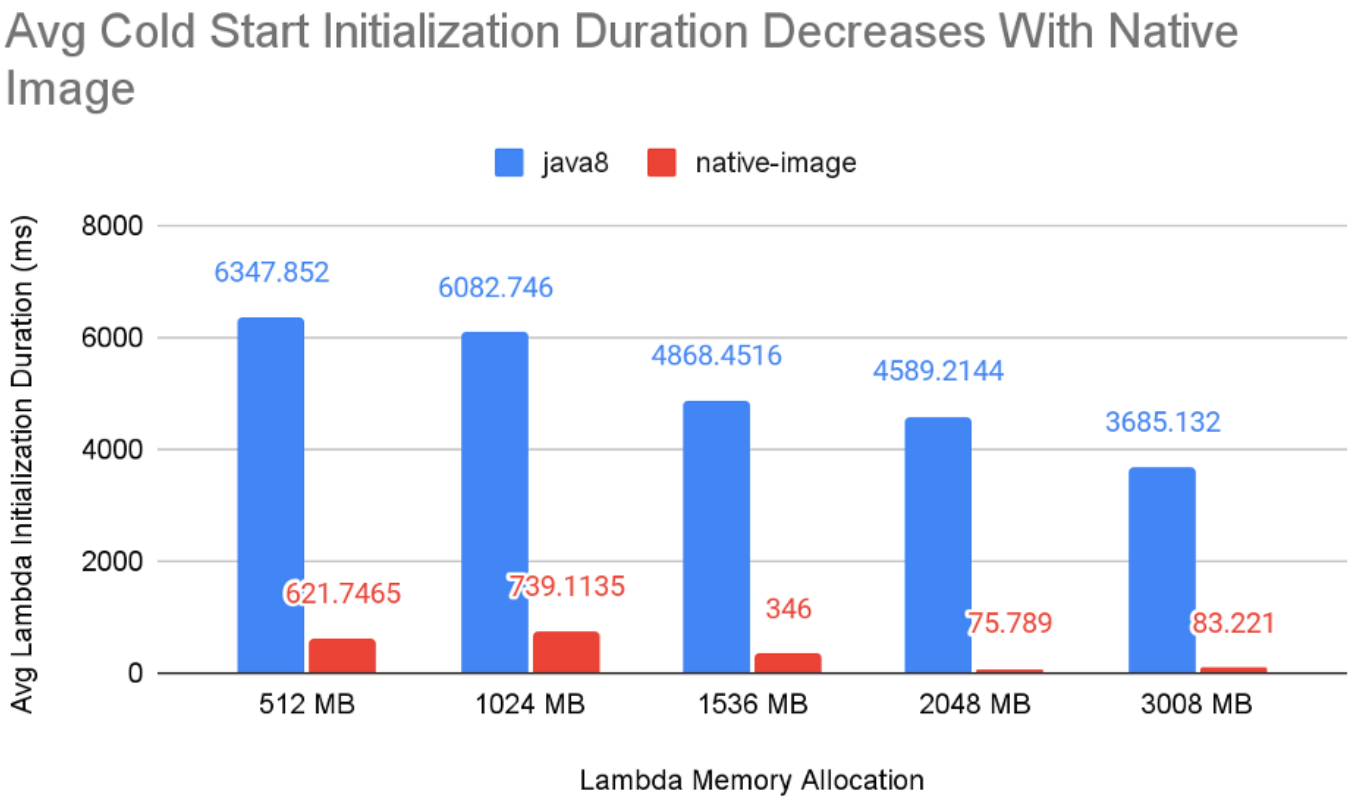
Detailed demos: https://www.youtube.com/watch?v=mhmqomex1zk

Don’t believe us?
Read what the competition says about us!

09 NOV 2022

Improving Developer Productivity at Disney with Serverless and Open Source

https://aws.amazon.com/de/blogs/opensource/improving-developer-productivity-at-disney-with-serverless-and-open-source/



Help I am looking for

- You are running heavy Java workloads where performance matters?
- You want to experiment with the forward-looking features of GraalVM?

Get in contact!

https://thomaswue.dev

thomas.wuerthinger@oracle.com