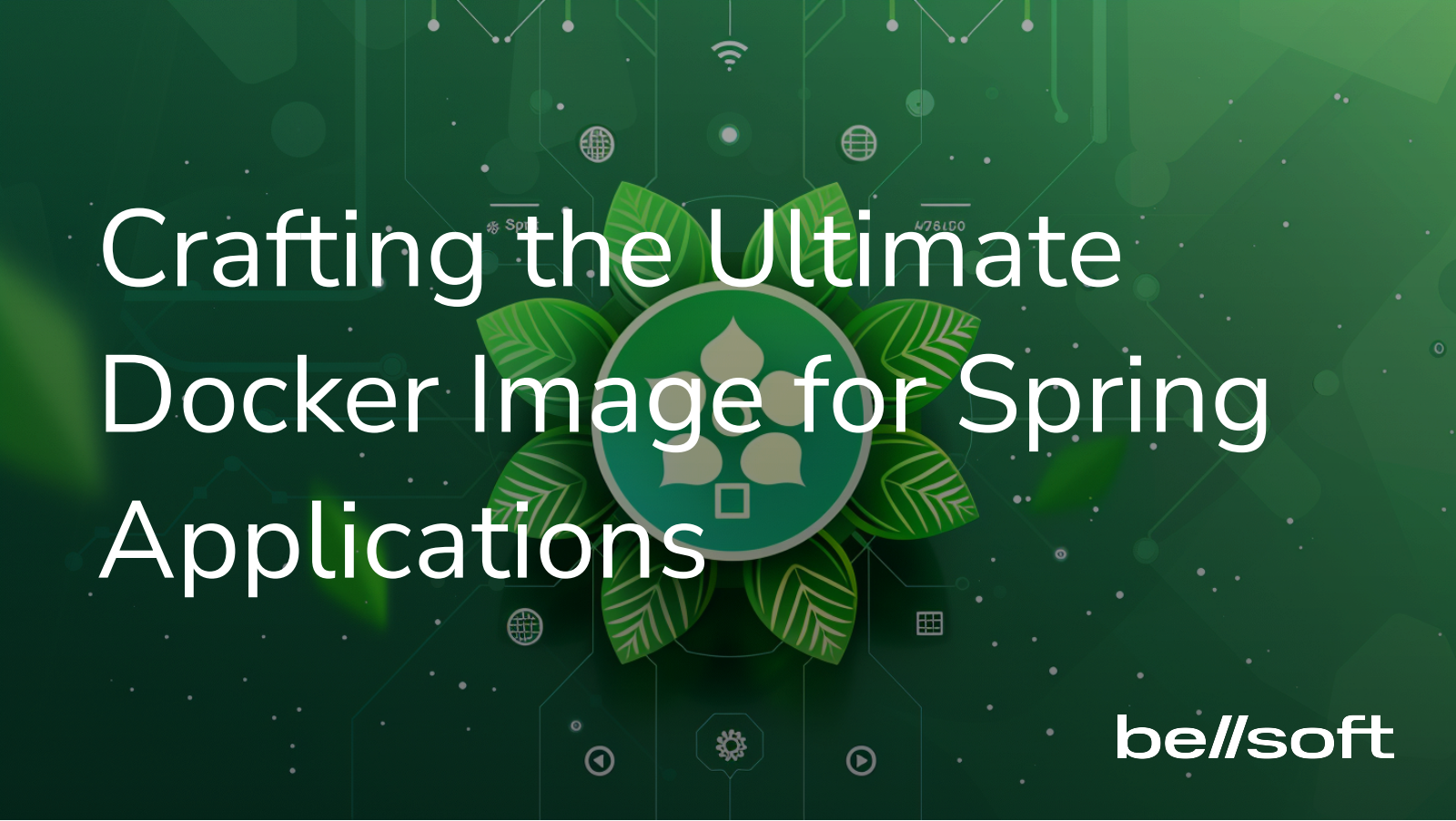

Crafting the Ultimate Docker Image for Spring Applications
bellsoft

whoami

Pasha Finkelshteyn @asm0dey.site @asm0di0

whoami

- Pasha Finkelshteyn

whoami

- Pasha Finkelshteyn
- Dev 🥑 at BellSoft

whoami

- Pasha Finkelshteyn
- Dev 🥑 at BellSoft
- ≈10 years in JVM. Mostly ☕ and

whoami

- Pasha Finkelshteyn
- Dev 🥑 at BellSoft
- ≈10 years in JVM. Mostly ☕ and
- And 🍃

whoami

- Pasha Finkelshteyn
- Dev 🥑 at BellSoft
- ≈10 years in JVM. Mostly ☕ and
- And 🌱
- 🐦 asm0di0

whoami

- Pasha Finkelshteyn
- Dev 🥑 at BellSoft
- ≈10 years in JVM. Mostly ☕ and
- And 🍃
- asm0di0
- @asm0dey@fosstodon.org

BellSoft

- Vendor of Liberica JDK
- Contributor to the OpenJDK
- Author of ARM32 support in JDK
- Own base images
- Own Linux: Alpaquita

Liberica is the JDK officially recommended by 

BellSoft

- Vendor of Liberica JDK
- Contributor to the OpenJDK
- Author of ARM32 support in JDK
- Own base images
- Own Linux: Alpaquita

Liberica is the JDK officially recommended by

We know our stuff!

So, what is ultimate?

- Smallest
- Fatsest startup
- Something inbetween

So, let’s look at all of them!

How do we create an image?

Let’s start from trivial.

```
1   FROM bellsoft/liberica-runtime-container:jdk-21.0.3_10-musl
2
3   COPY . /app
4   RUN cd /app && ./gradlew build
5   ENTRYPOINT     java -jar /app/build/libs/spring-petclinic*.jar
```

How do we create an image?

Let’s start from trivial.

```
1   FROM bellsoft/liberica-runtime-container:jdk-21.0.3_10-musl
2
3   COPY . /app
4   RUN cd /app && ./gradlew build
5   ENTRYPOINT     java -jar /app/build/libs/spring-petclinic*.jar
```

How do we create an image?

Let’s start from trivial.

```
1   FROM bellsoft/liberica-runtime-container:jdk-21.0.3_10-musl
2
3   COPY . /app
4   RUN cd /app && ./gradlew build
5   ENTRYPOINT     java -jar /app/build/libs/spring-petclinic*.jar
```

How do we create an image?

Let’s start from trivial.

```
1  FROM bellsoft/liberica-runtime-container:jdk-21.0.3_10-musl
2
3  COPY . /app
4  RUN cd /app && ./gradlew build
5  ENTRYPOINT     java -jar /app/build/libs/spring-petclinic*.jar
```

How do we create an image?

Let's start from trivial.

```
1  FROM bellsoft/liberica-runtime-container:jdk-21.0.3_10-musl
2
3  COPY . /app
4  RUN cd /app && ./gradlew build
5  ENTRYPOINT     java -jar /app/build/libs/spring-petclinic*.jar
```

`Dockerfile` directives

1. Each directive creates a layer of the image.
2. Layers are *immutable*
3. Some layers are zero-sized

`Dockerfile` directives

1. Each directive creates a layer of the image.
2. Layers are *immutable*
3. Some layers are zero-sized

```
1  FROM bash
2
3  COPY . /app
4  RUN rm -rf /app
```

`Dockerfile` directives

1. Each directive creates a layer of the image.
2. Layers are *immutable*
3. Some layers are zero-sized

```
1  FROM bash
2
3  COPY . /app
4  RUN rm -rf /app
```

`Dockerfile` directives

1. Each directive creates a layer of the image.
2. Layers are *immutable*
3. Some layers are zero-sized (for example `RUN rm -rf /app`)

```
1  FROM bash
2
3  COPY . /app
4  RUN rm -rf /app
```

`Dockerfile` directives

1. Each directive creates a layer of the image.
2. Layers are *immutable*
3. Some layers are zero-sized (for example `RUN rm -rf /app`)

```
1  FROM bash
2
3  COPY . /app
4  RUN rm -rf /app
```

4. We would like image to be light, but it's not :(

Our `Dockerfile`

```
1  FROM bellsoft/liberica-runtime-container:jdk-21.0.3_10-musl
2
3  COPY . /app
4  RUN cd /app && ./gradlew build
5  ENTRYPOINT java -jar /app/build/libs/spring-petclinic*.jar
```

Our `Dockerfile`

```
1  FROM bellsoft/liberica-runtime-container:jdk-21.0.3_10-musl
2
3  COPY . /app
4  RUN cd /app && ./gradlew build
5  ENTRYPOINT java -jar /app/build/libs/spring-petclinic*.jar
```

Our `Dockerfile`

```
1  FROM bellsoft/liberica-runtime-container:jdk-21.0.3_10-musl
2
3  COPY . /app
4  RUN cd /app && ./gradlew build
5  ENTRYPOINT java -jar /app/build/libs/spring-petclinic*.jar
```

Our `Dockerfile`

```
1  FROM bellsoft/liberica-runtime-container:jdk-21.0.3_10-musl
2
3  COPY . /app
4  RUN cd /app && ./gradlew build
5  ENTRYPOINT java -jar /app/build/libs/spring-petclinic*.jar
```

Result

```
13  <missing>                                   0B        ARG BUNDLE_TYPE
14  <missing>                                   0B        ARG BUNDLE_TYPE
15  <missing>                                   0B        ARG BUNDLE_TYPE
16  <missing>                                   0B        ARG JAVA_RELEAS
17  <missing>                                   7.49MiB   CMD ["/bin/sh"]
18  <missing>                                   0B        |9 APK_REPOS= D
19  <missing>                                   0B        ADD file:a4d77f
20  <missing>                                   0B        LABEL org.openc
21  <missing>                                   0B        LABEL org.openc
22  <missing>                                   0B        LABEL maintaine
23  <missing>                                   0B        ARG APK_REPOS D
24  <missing>                                   0B        ARG APK_REPOS L
25  <missing>                                   0B        ARG APK_REPOS L
26  <missing>                                   0B        ARG APK_REPOS L
27  <missing>                                   0B        ARG LIBC MINIRO
28  <missing>                                   0B        ARG MINIROOTFS
```

Result

```
 9   <missing>                                       0B        LABEL org.openc
10   <missing>                                       0B        LABEL maintaine
11   <missing>                                       0B        ARG BUNDLE_TYPE
12   <missing>                                       0B        ARG BUNDLE_TYPE
13   <missing>                                       0B        ARG BUNDLE_TYPE
14   <missing>                                       0B        ARG BUNDLE_TYPE
15   <missing>                                       0B        ARG BUNDLE_TYPE
16   <missing>                                       0B        ARG JAVA_RELEAS
17   <missing>                                       7.49MiB   CMD ["/bin/sh"]
18   <missing>                                       0B        |9 APK_REPOS= D
19   <missing>                                       0B        ADD file:a4d77f
20   <missing>                                       0B        LABEL org.openc
21   <missing>                                       0B        LABEL org.openc
22   <missing>                                       0B        LABEL maintaine
23   <missing>                                       0B        ARG APK_REPOS D
24   <missing>                                       0B        ARG APK_REPOS L
25   <missing>                                       0B        ARG APK_REPOS L
```

Result

```
3   74a0788411                                                  512.03MiB /bin/sh -c cd /
4   7b82eb08ec                                                  67.41MiB  COPY dir:ef251b
5   e4bcf83d29 bellsoft/liberica-runtime-container:jdk-21.0.3_10-musl 98.32MiB  |15 BUNDLE_TYPE
6   <missing>                                                   0B        ENV JAVA_HOME="
7   <missing>                                                   0B        ENV LANG=en_US.
8   <missing>                                                   0B        LABEL org.openc
9   <missing>                                                   0B        LABEL org.openc
10  <missing>                                                   0B        LABEL maintaine
11  <missing>                                                   0B        ARG BUNDLE_TYPE
12  <missing>                                                   0B        ARG BUNDLE_TYPE
13  <missing>                                                   0B        ARG BUNDLE_TYPE
14  <missing>                                                   0B        ARG BUNDLE_TYPE
15  <missing>                                                   0B        ARG BUNDLE_TYPE
16  <missing>                                                   0B        ARG JAVA_RELEAS
17  <missing>                                                   7.49MiB   CMD ["/bin/sh"]
18  <missing>                                                   0B        |9 APK_REPOS= D
19  <missing>                                                   0B        ADD file:a4d77f
```

Result

```
ID          TAG                                                  SIZE      COMMAND
cda5201e70  dumb:latest                                          0B        CMD ["/bin/sh"
74a0788411                                                       512.03MiB /bin/sh -c cd /
7b82eb08ec                                                       67.41MiB  COPY dir:ef251b
e4bcf83d29  bellsoft/liberica-runtime-container:jdk-21.0.3_10-musl 98.32MiB |15 BUNDLE_TYPE
<missing>                                                        0B        ENV JAVA_HOME="
<missing>                                                        0B        ENV LANG=en_US.
<missing>                                                        0B        LABEL org.openc
<missing>                                                        0B        LABEL org.openc
<missing>                                                        0B        LABEL maintaine
<missing>                                                        0B        ARG BUNDLE_TYPE
<missing>                                                        0B        ARG BUNDLE_TYPE
<missing>                                                        0B        ARG BUNDLE_TYPE
<missing>                                                        0B        ARG BUNDLE_TYPE
<missing>                                                        0B        ARG BUNDLE_TYPE
<missing>                                                        0B        ARG JAVA_RELEAS
```

Result

```
ID          TAG                                                      SIZE       COMMAND
cda5201e70  dumb:latest                                              0B         CMD ["/bin/sh"
74a0788411                                                           512.03MiB  /bin/sh -c cd /
7b82eb08ec                                                           67.41MiB   COPY dir:ef251b
e4bcf83d29  bellsoft/liberica-runtime-container:jdk-21.0.3_10-musl   98.32MiB   |15 BUNDLE_TYPE
<missing>                                                            0B         ENV JAVA_HOME="
<missing>                                                            0B         ENV LANG=en_US.
<missing>                                                            0B         LABEL org.openc
<missing>                                                            0B         LABEL org.openc
<missing>                                                            0B         LABEL maintaine
<missing>                                                            0B         ARG BUNDLE_TYPE
<missing>                                                            0B         ARG BUNDLE_TYPE
<missing>                                                            0B         ARG BUNDLE_TYPE
<missing>                                                            0B         ARG BUNDLE_TYPE
<missing>                                                            0B         ARG BUNDLE_TYPE
<missing>                                                            0B         ARG JAVA_RELEAS
```

Result

```
ID          TAG                                                    SIZE      COMMAND
cda5201e70  dumb:latest                                            0B        CMD ["/bin/sh"
74a0788411                                                         512.03MiB /bin/sh -c cd /
7b82eb08ec                                                         67.41MiB  COPY dir:ef251b
e4bcf83d29  bellsoft/liberica-runtime-container:jdk-21.0.3_10-musl 98.32MiB  |15 BUNDLE_TYPE
<missing>                                                          0B        ENV JAVA_HOME="
<missing>                                                          0B        ENV LANG=en_US.
<missing>                                                          0B        LABEL org.openc
<missing>                                                          0B        LABEL org.openc
<missing>                                                          0B        LABEL maintaine
<missing>                                                          0B        ARG BUNDLE_TYPE
<missing>                                                          0B        ARG BUNDLE_TYPE
<missing>                                                          0B        ARG BUNDLE_TYPE
<missing>                                                          0B        ARG BUNDLE_TYPE
<missing>                                                          0B        ARG BUNDLE_TYPE
<missing>                                                          0B        ARG JAVA_RELEAS
```

Result

```
ID           TAG                                                    SIZE      COMMAND
cda5201e70 dumb:latest                                              0B        CMD ["/bin/sh"
74a0788411                                                          512.03MiB /bin/sh -c cd /
7b82eb08ec                                                          67.41MiB  COPY dir:ef251b
e4bcf83d29 bellsoft/liberica-runtime-container:jdk-21.0.3_10-musl   98.32MiB  |15 BUNDLE_TYPE
<missing>                                                           0B        ENV JAVA_HOME="
<missing>                                                           0B        ENV LANG=en_US.
<missing>                                                           0B        LABEL org.openc
<missing>                                                           0B        LABEL org.openc
<missing>                                                           0B        LABEL maintaine
<missing>                                                           0B        ARG BUNDLE_TYPE
<missing>                                                           0B        ARG BUNDLE_TYPE
<missing>                                                           0B        ARG BUNDLE_TYPE
<missing>                                                           0B        ARG BUNDLE_TYPE
<missing>                                                           0B        ARG BUNDLE_TYPE
<missing>                                                           0B        ARG JAVA_RELEAS
```

579.44 MB are changed on every build!

Why do we care? We care because:

1. `push` takes longer time to start (update is longer)
2. `pull` takes longer (update takes longer & scaling takes longer)

Also, more disk psace is inefficiently used

Optimizing. Round 1.

Let’s build it outside of container

```
FROM bellsoft/liberica-runtime-container:jdk-21.0.3_10-musl

COPY . /app
RUN cd /app && ./gradlew build
ENTRYPOINT java -jar /app/build/libs/spring-petclinic*.jar
```

Optimizing. Round 1.

Let’s build it outside of container

```
1  FROM bellsoft/liberica-runtime-container:jdk-21.0.3_10-musl
2
3  COPY build/libs/spring-petclinic-3.3.0.jar /app/app.jar
4  CMD java -jar /app/app.jar
```

Optimizing. Round 1.

Let’s build it outside of container

```
1  FROM bellsoft/liberica-runtime-container:jdk-21.0.3_10-musl
2
3  COPY build/libs/spring-petclinic-3.3.0.jar /app/app.jar
4  CMD java -jar /app/app.jar
```

Just copying the prebuilt `jar` file

Results

```
ID          TAG                                                    SIZE      COMMAND
cda5201e70  dumb:latest                                            0B        CMD ["/bin/sh" "-c" "/app
74a0788411                                                         512.03MiB /bin/sh -c cd /app && ./g
7b82eb08ec                                                         67.41MiB  COPY dir:ef251bd1e17ac0af
e4bcf83d29  bellsoft/liberica-runtime-container:jdk-21.0.3_10-musl 98.32MiB  |15 BUNDLE_TYPE=jdk CDS=n
<missing>                                                          0B        ENV JAVA_HOME="/usr/lib/j
<missing>                                                          0B        ENV LANG=en_US.UTF-8 LANG
<missing>                                                          0B        LABEL org.opencontainers.
<missing>                                                          0B        LABEL org.opencontainers.
<missing>                                                          0B        LABEL maintainer="$MAINTA
<missing>                                                          0B        ARG BUNDLE_TYPE CDS DESCR
<missing>                                                          0B        ARG BUNDLE_TYPE CDS JAVA_
<missing>                                                          0B        ARG BUNDLE_TYPE CDS JAVA_
<missing>                                                          0B        ARG BUNDLE_TYPE CDS JAVA_
<missing>                                                          0B        ARG BUNDLE_TYPE JAVA_RELE
<missing>                                                          0B        ARG JAVA_RELEASE
<missing>                                                          7.49MiB   CMD ["/bin/sh"]
```

Results

```
ID         TAG                                                    SIZE     COMMAND
cefb30e21d smarter:latest                                         0B       CMD ["/bin/sh" "-c" "java
46d7594000                                                        58.80MiB COPY file:8d96bee95ae5ce43
e4bcf83d29 bellsoft/liberica-runtime-container:jdk-21.0.3_10-musl 98.32MiB |15 BUNDLE_TYPE=jdk CDS=no
<missing>                                                         0B       ENV JAVA_HOME="/usr/lib/jv
<missing>                                                         0B       ENV LANG=en_US.UTF-8 LANGU
<missing>                                                         0B       LABEL org.opencontainers.i
<missing>                                                         0B       LABEL org.opencontainers.i
<missing>                                                         0B       LABEL maintainer="$MAINTAI
<missing>                                                         0B       ARG BUNDLE_TYPE CDS DESCRI
<missing>                                                         0B       ARG BUNDLE_TYPE CDS JAVA_R
<missing>                                                         0B       ARG BUNDLE_TYPE CDS JAVA_R
<missing>                                                         0B       ARG BUNDLE_TYPE CDS JAVA_R
<missing>                                                         0B       ARG BUNDLE_TYPE JAVA_RELEA
<missing>                                                         0B       ARG JAVA_RELEASE
<missing>                                                         7.49MiB  CMD ["/bin/sh"]
```

Pasha Finkelshteyn @asm0dey.site @asm0di0

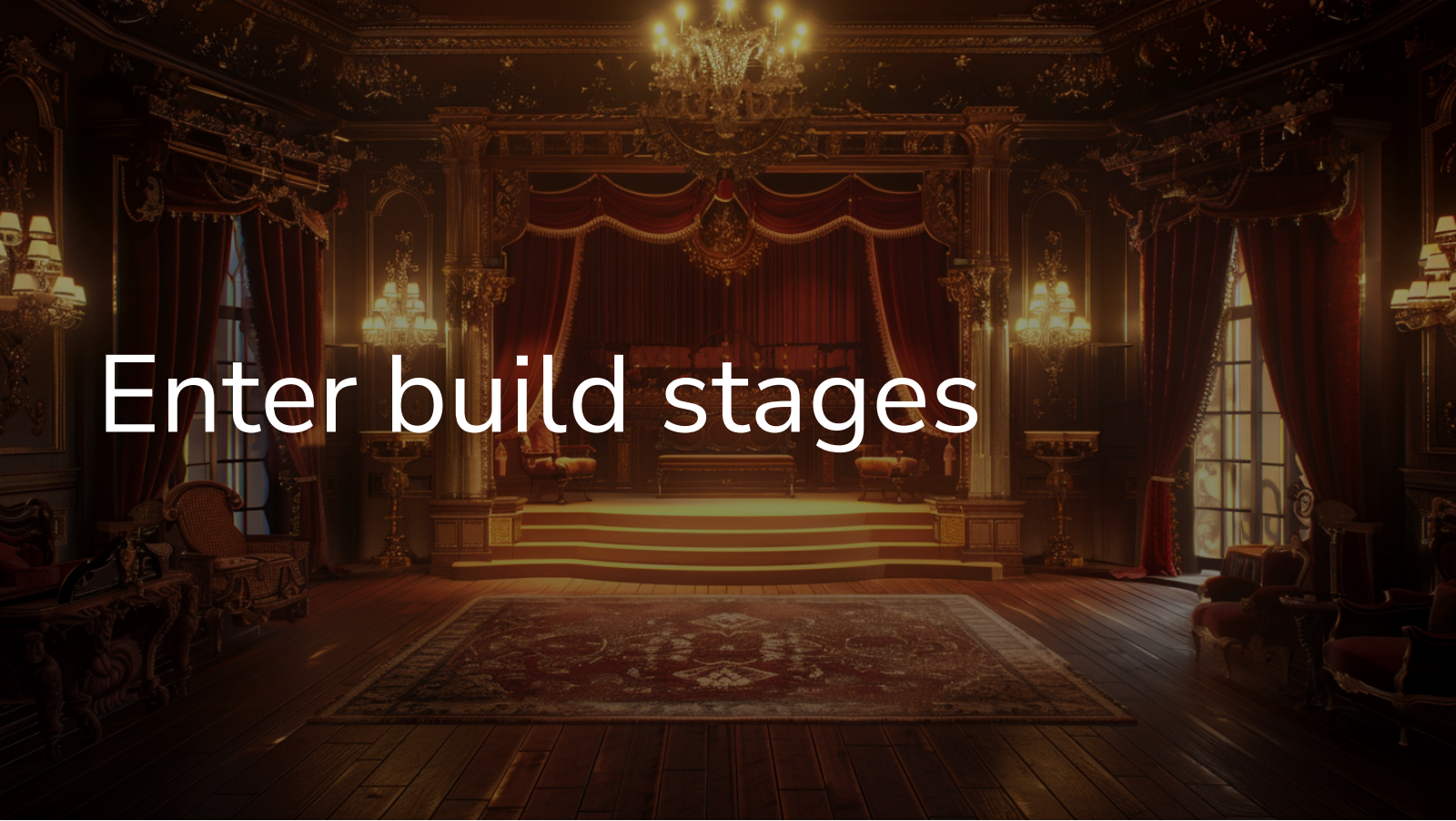
Saved 500+ MB

But the build is not clean now :(

Saved 500+ MB

But the build is not clean now :(

We have to build everything outside, what if environment affects the build?


Enter build stages

Multi-stage builds

Holy grail of pure builds

- Allow clean builds
- Allow optimal packaging
- Allow different base images

Staged example

```
1  FROM bellsoft/liberica-runtime-container:jdk-musl as builder
2
3  COPY . /app
4  RUN cd /app && ./gradlew build -xtest
5
6  FROM bellsoft/liberica-runtime-container:jre-slim-musl as runner
7
8  COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
```

Staged example

```
1  FROM bellsoft/liberica-runtime-container:jdk-musl as builder
2
3  COPY . /app
4  RUN cd /app && ./gradlew build -xtest
5
6  FROM bellsoft/liberica-runtime-container:jre-slim-musl as runner
7
8  COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
```

Staged example

```
1  FROM bellsoft/liberica-runtime-container:jdk-musl as builder
2
3  COPY . /app
4  RUN cd /app && ./gradlew build -xtest
5
6  FROM bellsoft/liberica-runtime-container:jre-slim-musl as runner
7
8  COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
```

Staged example

```
1  FROM bellsoft/liberica-runtime-container:jdk-musl as builder
2
3  COPY . /app
4  RUN cd /app && ./gradlew build -xtest
5
6  FROM bellsoft/liberica-runtime-container:jre-slim-musl as runner
7
8  COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
```

Staged example

```
1  FROM bellsoft/liberica-runtime-container:jdk-musl as builder
2
3  COPY . /app
4  RUN cd /app && ./gradlew build -xtest
5
6  FROM bellsoft/liberica-runtime-container:jre-slim-musl as runner
7
8  COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
```

Staged example

```
1  FROM bellsoft/liberica-runtime-container:jdk-musl as builder
2
3  COPY . /app
4  RUN cd /app && ./gradlew build -xtest
5
6  FROM bellsoft/liberica-runtime-container:jre-slim-musl as runner
7
8  COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
```

Result

```
ID          TAG                                                    SIZE      COMMAND
6ea7ad11af smarter:latest                                         0B        CMD ["/bin/sh" "-c
d6eb71a5df                                                        58.80MiB  COPY file:8d96bee9
c6ede8dac2 bellsoft/liberica-runtime-container:jdk-21.0.3_10-musl 98.32MiB |15 BUNDLE_TYPE=jc
<missing>                                                         0B        ENV JAVA_HOME="/us
<missing>                                                         0B        ENV LANG=en_US.UTF
<missing>                                                         0B        LABEL org.opencont
<missing>                                                         0B        LABEL org.opencont
<missing>                                                         0B        LABEL maintainer="
<missing>                                                         0B        ARG BUNDLE_TYPE CD
<missing>                                                         0B        ARG BUNDLE_TYPE CD
<missing>                                                         0B        ARG BUNDLE_TYPE CD
<missing>                                                         0B        ARG BUNDLE_TYPE CD
<missing>                                                         0B        ARG BUNDLE_TYPE JA
<missing>                                                         0B        ARG JAVA_RELEASE
<missing>                                                         7.49MiB   CMD ["/bin/sh"]
```

Result

```
ID          TAG                                                    SIZE      COMMAND
6ea7ad11af  smarter:latest                                         0B        CMD ["/bin/sh" "-c
d6eb71a5df                                                         58.80MiB  COPY file:8d96bee9
c6ede8dac2  bellsoft/liberica-runtime-container:jdk-21.0.3_10-musl 98.32MiB |15 BUNDLE_TYPE=jd
<missing>                                                          0B        ENV JAVA_HOME="/us
<missing>                                                          0B        ENV LANG=en_US.UTF
<missing>                                                          0B        LABEL org.opencont
<missing>                                                          0B        LABEL org.opencont
<missing>                                                          0B        LABEL maintainer="
<missing>                                                          0B        ARG BUNDLE_TYPE CD
<missing>                                                          0B        ARG BUNDLE_TYPE CD
<missing>                                                          0B        ARG BUNDLE_TYPE CD
<missing>                                                          0B        ARG BUNDLE_TYPE CD
<missing>                                                          0B        ARG BUNDLE_TYPE JA
<missing>                                                          0B        ARG JAVA_RELEASE
<missing>                                                          7.49MiB   CMD ["/bin/sh"]
```

Pasha Finkelshteyn @asm0dey.site @asm0di0

That’s all folks!

Pasha Finkelshteyn @asm0dey.site @asm0di0

That’s all folks!

Or is it?

Pasha Finkelshteyn @asm0dey.site @asm0di0

What’s these 59 MiB?

We have to pull 59 MiB of petclinic on every small commit!

Is there a way to optimize it?

Layers!

```
1  FROM bellsoft/liberica-runtime-container:jdk-musl as builder
2
3  COPY . /app
4  RUN cd /app && ./gradlew build -xtest
5
6  FROM bellsoft/liberica-runtime-container:jre-slim-musl as optimizer
7
8  COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
9  WORKDIR /app
10 RUN java -Djarmode=tools -jar /app/app.jar extract --layers --launcher
```

- Build image

Layers!

```
1  FROM bellsoft/liberica-runtime-container:jdk-musl as builder
2
3  COPY . /app
4  RUN cd /app && ./gradlew build -xtest
5
6  FROM bellsoft/liberica-runtime-container:jre-slim-musl as optimizer
7
8  COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
9  WORKDIR /app
10 RUN java -Djarmode=tools -jar /app/app.jar extract --layers --launcher
11
```

- Build image
- Introduce new "optimizer" stage

Layers!

```
4   RUN cd /app && ./gradlew build -xtest
5
6   FROM bellsoft/liberica-runtime-container:jre-slim-musl as optimizer
7
8   COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
9   WORKDIR /app
10  RUN java -Djarmode=tools -jar /app/app.jar extract --layers --launcher
11
12  FROM bellsoft/liberica-runtime-container:jre-slim-musl as runner
13
14  ENTRYPOINT ["java", "org.springframework.boot.loader.launch.JarLauncher"]
```

- Build image
- Introduce new "optimizer" stage

Layers!

```
6   FROM bellsoft/liberica-runtime-container:jre-slim-musl as optimizer
7
8   COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
9   WORKDIR /app
10  RUN java -Djarmode=tools -jar /app/app.jar extract --layers --launcher
11
12  FROM bellsoft/liberica-runtime-container:jre-slim-musl as runner
13
14  ENTRYPOINT ["java", "org.springframework.boot.loader.launch.JarLauncher"]
15  COPY --from=optimizer /app/app/dependencies/ ./
```

- Build image
- Introduce new "optimizer" stage
- Extract the jar to layered structure

Layers!

```
 8  COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
 9  WORKDIR /app
10  RUN java -Djarmode=tools -jar /app/app.jar extract --layers --launcher
11
12  FROM bellsoft/liberica-runtime-container:jre-slim-musl as runner
13
14  ENTRYPOINT ["java", "org.springframework.boot.loader.launch.JarLauncher"]
15  COPY --from=optimizer /app/app/dependencies/ ./
16  COPY --from=optimizer /app/app/spring-boot-loader/ ./
17  COPY --from=optimizer /app/app/snapshot-dependencies/ ./
```

- Build image
- Introduce new "optimizer" stage
- Extract the jar to layered structure

Layers!

```
9   WORKDIR /app
10  RUN java -Djarmode=tools -jar /app/app.jar extract --layers --launcher
11
12  FROM bellsoft/liberica-runtime-container:jre-slim-musl as runner
13
14  ENTRYPOINT ["java", "org.springframework.boot.loader.launch.JarLauncher"]
15  COPY --from=optimizer /app/app/dependencies/ ./
16  COPY --from=optimizer /app/app/spring-boot-loader/ ./
17  COPY --from=optimizer /app/app/snapshot-dependencies/ ./
18  COPY --from=optimizer /app/app/application/ ./
```

- Build image
- Introduce new "optimizer" stage
- Extract the jar to layered structure
- Copy layers

Layers!

```
9   WORKDIR /app
10  RUN java -Djarmode=tools -jar /app/app.jar extract --layers --launcher
11
12  FROM bellsoft/liberica-runtime-container:jre-slim-musl as runner
13
14  ENTRYPOINT ["java", "org.springframework.boot.loader.launch.JarLauncher"]
15  COPY --from=optimizer /app/app/dependencies/ ./
16  COPY --from=optimizer /app/app/spring-boot-loader/ ./
17  COPY --from=optimizer /app/app/snapshot-dependencies/ ./
18  COPY --from=optimizer /app/app/application/ ./
```

- Build image
- Introduce new "optimizer" stage
- Extract the jar to layered structure
- Copy layers

Layers

Because Spring Boot jar is complex!

```
Usage:
  java -Djarmode=tools -jar my-app.jar

Available commands:
  extract      Extract the contents from the jar
  list-layers  List layers from the jar that can be extracted
```

Layers

```
app
├── application
│   ├── BOOT-INF
│   │   ├── classes
│   │   │   ├── application-mysql.properties
│   │   │   ├── application-postgres.properties
│   │   │   ├── application.properties
│   │   │   ├── banner.txt
│   │   │   ├── db
│   │   │   │   ├── ...
│   │   │   ├── org
```

Layers

```
app
├── application
|   ├── BOOT-INF
|   |   ├── classes
|   |   |   ├── application-mysql.properties
|   |   |   ├── application-postgres.properties
|   |   |   ├── application.properties
|   |   |   ├── banner.txt
|   |   |   ├── db
|   |   |   |   ├── ...
|   |   |   ├── org
```

Layers

```
├── application
│   ├── BOOT-INF
│   │   ├── classes
│   │   │   ├── application-mysql.properties
│   │   │   ├── application-postgres.properties
│   │   │   ├── application.properties
│   │   │   ├── banner.txt
│   │   │   ├── db
│   │   │   │   ├── ...
│   │   │   ├── org
│   │   │   │   └── springframework
```

Layers

```
 9  |   |   |   ├── db
10  |   |   |   |   ├── ...
11  |   |   |   ├── org
12  |   |   |   |   └── springframework
13  |   |   |   |       ├── aop
14  |   |   |   |       |   └── aspectj
15  |   |   |   |       |       └── annotation
16  |   |   |   |       |           └── AnnotationAwareAspectJAutoProxyCreator__BeanDefinitions
17  |   |   |   |       ├── com/github/asm0dey/ ...
18  |   |   |   ├── static
19  |   |   |   ....
```

Layers

```
12  |   |   |   |   └── springframework
13  |   |   |   |       ├── aop
14  |   |   |   |       |   └── aspectj
15  |   |   |   |       |       └── annotation
16  |   |   |   |       |           └── AnnotationAwareAspectJAutoProxyCreator__BeanDefinitions
17  |   |   |   |       ├── com/github/asm0dey/ ...
18  |   |   |   ├── static
19  |   |   |   ....
20  |   |   ├── classpath.idx
21  |   |   └── layers.idx
22  |   └── META-INF
```

Layers

```
16  |   |   |   |       |           └── AnnotationAwareAspectJAutoProxyCreator__BeanDefinitions
17  |   |   |   |       ├── com/github/asm0dey/ ...
18  |   |   |   ├── static
19  |   |   |   ....
20  |   |   ├── classpath.idx
21  |   |   └── layers.idx
22  |   └── META-INF
23  |       ├── MANIFEST.MF
24  |       ├── native-image
25  |       |   ├── ch.qos.logback
26  |       |   |   └── logback-classic
```

Layers

```
17  |   |   |   |       ├── com/github/asm0dey/ ...
18  |   |   |   ├── static
19  |   |   |   ....
20  |   |   ├── classpath.idx
21  |   |   └── layers.idx
22  |   └── META-INF
23  |       ├── MANIFEST.MF
24  |       ├── native-image
25  |       |   ├── ch.qos.logback
26  |       |   |   └── logback-classic
27  |       |   |       └── 1.5.6
```

Layers

```
22  |   └── META-INF
23  |       ├── MANIFEST.MF
24  |       ├── native-image
25  |       |   ├── ch.qos.logback
26  |       |   |   └── logback-classic
27  |       |   |       └── 1.5.6
28  |       |   |           ├── reflect-config.json
29  |       |   |           └── resource-config.json
30  |       |   ├── ...
31  |       ├── sbom
32  |       |   └── application.cdx.json
```

Layers

```
27  |       |   |       └── 1.5.6
28  |       |   |           ├── reflect-config.json
29  |       |   |           └── resource-config.json
30  |       |   ├── ...
31  |       ├── sbom
32  |       |   └── application.cdx.json
33  |       └── services
34  |           └── java.nio.file.spi.FileSystemProvider
35  ├── dependencies
36  |   └── BOOT-INF
37  |       └── lib
```

Layers

```
30  |       |   ├── ...
31  |       ├── sbom
32  |       |   └── application.cdx.json
33  |       └── services
34  |           └── java.nio.file.spi.FileSystemProvider
35  ├── dependencies
36  |   └── BOOT-INF
37  |       └── lib
38  |           ├── angus-activation-2.0.2.jar
39  |           ├── ...
40  ├── snapshot-dependencies
```

Layers

```
33  |       └── services
34  |           └── java.nio.file.spi.FileSystemProvider
35  ├── dependencies
36  |   └── BOOT-INF
37  |       └── lib
38  |           ├── angus-activation-2.0.2.jar
39  |           ├── ...
40  ├── snapshot-dependencies
41  └── spring-boot-loader
42      └── org
43          └── springframework
```

Layers

```
35  ├── dependencies
36  │   └── BOOT-INF
37  │       └── lib
38  │           ├── angus-activation-2.0.2.jar
39  │           ├── ...
40  ├── snapshot-dependencies
41  └── spring-boot-loader
42      └── org
43          └── springframework
44              └── boot
45                  └── loader
```

Layers

```
36  |   └── BOOT-INF
37  |       └── lib
38  |           ├── angus-activation-2.0.2.jar
39  |           ├── ...
40  ├── snapshot-dependencies
41  └── spring-boot-loader
42      └── org
43          └── springframework
44              └── boot
45                  └── loader
46                      ├── jar
```

Layers

```
145                             ├── ZipContent$Loader.class
146                             ├── ZipContent$Source.class
147                             ├── ZipContent.class
148                             ├── ZipDataDescriptorRecord.class
149                             ├── ZipEndOfCentralDirectoryRecord$Located.class
150                             ├── ZipEndOfCentralDirectoryRecord.class
151                             ├── ZipLocalFileHeaderRecord.class
152                             ├── ZipString$CompareType.class
153                             └── ZipString.class
154
155   212 directories, 525 files
```

Layered image structure

```
ID           TAG                                                SIZE      COMMAND
618743f6a2 layers:latest                                        2.96MiB   COPY dir:e0faa63b96544
9cca3273f0                                                      0B        COPY dir:acd0d0ac1f7df
fe123ee16e                                                      382.56kiB COPY dir:01225d3c4ef6c
dede9bac3d                                                      57.69MiB  COPY dir:755815d928fd9
65e3fb4cbf                                                      0B        ENTRYPOINT ["java" "or
7130fa5864 bellsoft/liberica-runtime-container:jre-slim-musl 126.80MiB |18 CDS=no DESCRIPTION
<missing>                                                       0B        ENV JAVA_HOME="/usr/li
<missing>                                                       0B        ENV LANG=en_US.UTF-8 L
<missing>                                                       0B        LABEL org.opencontaine
```

Layered image structure

```
ID         TAG                                                SIZE      COMMAND
618743f6a2 layers:latest                                      2.96MiB   COPY dir:e0faa63b96544
9cca3273f0                                                    0B        COPY dir:acd0d0ac1f7df
fe123ee16e                                                    382.56kiB COPY dir:01225d3c4ef6c
dede9bac3d                                                    57.69MiB  COPY dir:755815d928fd9
65e3fb4cbf                                                    0B        ENTRYPOINT ["java" "or
7130fa5864 bellsoft/liberica-runtime-container:jre-slim-musl 126.80MiB |18 CDS=no DESCRIPTION
<missing>                                                     0B        ENV JAVA_HOME="/usr/li
<missing>                                                     0B        ENV LANG=en_US.UTF-8 L
<missing>                                                     0B        LABEL org.opencontaine
```

- Dependencies: ~57.7MiB

Layered image structure

```
1   ID         TAG                                                SIZE      COMMAND
2   618743f6a2 layers:latest                                      2.96MiB   COPY dir:e0faa63b96544
3   9cca3273f0                                                    0B        COPY dir:acd0d0ac1f7df
4   fe123ee16e                                                    382.56kiB COPY dir:01225d3c4ef6c
5   dede9bac3d                                                    57.69MiB  COPY dir:755815d928fd9
6   65e3fb4cbf                                                    0B        ENTRYPOINT ["java" "or
7   7130fa5864 bellsoft/liberica-runtime-container:jre-slim-musl 126.80MiB |18 CDS=no DESCRIPTION
8   <missing>                                                     0B        ENV JAVA_HOME="/usr/li
9   <missing>                                                     0B        ENV LANG=en_US.UTF-8 L
10  <missing>                                                     0B        LABEL org.opencontaine
```

- Dependencies: ~57.7MiB
- Launcher: 382.56kiB

Layered image structure

```
ID           TAG                                                SIZE      COMMAND
618743f6a2   layers:latest                                      2.96MiB   COPY dir:e0faa63b96544
9cca3273f0                                                      0B        COPY dir:acd0d0ac1f7df
fe123ee16e                                                      382.56kiB COPY dir:01225d3c4ef6c
dede9bac3d                                                      57.69MiB  COPY dir:755815d928fd9
65e3fb4cbf                                                      0B        ENTRYPOINT ["java" "or
7130fa5864   bellsoft/liberica-runtime-container:jre-slim-musl  126.80MiB |18 CDS=no DESCRIPTION
<missing>                                                       0B        ENV JAVA_HOME="/usr/li
<missing>                                                       0B        ENV LANG=en_US.UTF-8 L
<missing>                                                       0B        LABEL org.opencontaine
```

- Dependencies: ~57.7MiB
- Launcher: 382.56kiB

Layered image structure

```
ID         TAG                                                SIZE      COMMAND
618743f6a2 layers:latest                                      2.96MiB   COPY dir:e0faa63b96544
9cca3273f0                                                    0B        COPY dir:acd0d0ac1f7df
fe123ee16e                                                    382.56kiB COPY dir:01225d3c4ef6c
dede9bac3d                                                    57.69MiB  COPY dir:755815d928fd9
65e3fb4cbf                                                    0B        ENTRYPOINT ["java" "or
7130fa5864 bellsoft/liberica-runtime-container:jre-slim-musl 126.80MiB |18 CDS=no DESCRIPTION
<missing>                                                     0B        ENV JAVA_HOME="/usr/li
<missing>                                                     0B        ENV LANG=en_US.UTF-8 L
<missing>                                                     0B        LABEL org.opencontaine
```

- Dependencies: ~57.7MiB
- Launcher: 382.56kiB
- Application: ~3MiB

Layered image structure

```
ID         TAG                                                SIZE      COMMAND
618743f6a2 layers:latest                                      2.96MiB   COPY dir:e0faa63b9654
9cca3273f0                                                    0B        COPY dir:acd0d0ac1f7c
fe123ee16e                                                    382.56kiB COPY dir:01225d3c4ef6
dede9bac3d                                                    57.69MiB  COPY dir:755815d928fc
65e3fb4cbf                                                    0B        ENTRYPOINT ["java" "c
7130fa5864 bellsoft/liberica-runtime-container:jre-slim-musl 126.80MiB |18 CDS=no DESCRIPTIC
<missing>                                                     0B        ENV JAVA_HOME="/usr/l
<missing>                                                     0B        ENV LANG=en_US.UTF-8
<missing>                                                     0B        LABEL org.opencontai
```

- Dependencies: ~57.7MiB
- Launcher: 382.56kiB
- Application: ~3MiB

Together: ~61MiB

61.0MiB > 58.8Mib!

What are we optimizing?

Pull size!

Pull size here is usually only around 3MiB!

We just reinvented how the BellSoft's buildpack works!

And it is amazing

Pasha Finkelshteyn @asm0dey.site @asm0di0

Diversion: buildpacks

https://paketo.io/

Trivial usage:

```
/usr/sbin/pack build petclinic \
  --builder bellsoft/buildpacks.builder \
  --path . \
  -e BP_JVM_VERSION=21
```

Diversion: buildpacks

https://paketo.io/

Trivial usage:

```
/usr/sbin/pack build petclinic \
  --builder bellsoft/buildpacks.builder \
  --path . \
  -e BP_JVM_VERSION=21
```

Diversion: buildpacks

https://paketo.io/

Trivial usage:

```
/usr/sbin/pack build petclinic \
  --builder bellsoft/buildpacks.builder \
  --path . \
  -e BP_JVM_VERSION=21
```

Diversion: buildpacks

https://paketo.io/

Trivial usage:

```
/usr/sbin/pack build petclinic \
  --builder bellsoft/buildpacks.builder \
  --path . \
  -e BP_JVM_VERSION=21
```

Result

```
ID          TAG               SIZE      COMMAND
2774e3f214  petclinic:latest  0B        Buildpacks Process Types
<missing>                     1.44kiB   Buildpacks Launcher Config
<missing>                     2.44MiB   Buildpacks Application Launcher
<missing>                     59.17MiB  Application Layer
<missing>                     729.21kiB Software Bill-of-Materials
<missing>                     97.87MiB  Layer: 'jre', Created by buildpack: bellsoft/buildpacl
<missing>                     200.00B   Layer: 'java-security-properties', Created by buildpac
<missing>                     4.27MiB   Layer: 'helper', Created by buildpack: bellsoft/buildp
<missing>                     2.97kiB
<missing>                     7.48MiB
```

Result

```
ID           TAG               SIZE      COMMAND
2774e3f214 petclinic:latest 0B        Buildpacks Process Types
<missing>                  1.44kiB   Buildpacks Launcher Config
<missing>                  2.44MiB   Buildpacks Application Launcher
<missing>                  59.17MiB  Application Layer
<missing>                  729.21kiB Software Bill-of-Materials
<missing>                  97.87MiB  Layer: 'jre', Created by buildpack: bellsoft/buildpacl
<missing>                  200.00B   Layer: 'java-security-properties', Created by buildpac
<missing>                  4.27MiB   Layer: 'helper', Created by buildpack: bellsoft/buildp
<missing>                  2.97kiB
<missing>                  7.48MiB
```

Result

```
ID           TAG               SIZE      COMMAND
2774e3f214 petclinic:latest 0B        Buildpacks Process Types
<missing>                  1.44kiB   Buildpacks Launcher Config
<missing>                  2.44MiB   Buildpacks Application Launcher
<missing>                  59.17MiB  Application Layer
<missing>                  729.21kiB Software Bill-of-Materials
<missing>                  97.87MiB  Layer: 'jre', Created by buildpack: bellsoft/buildpacl
<missing>                  200.00B   Layer: 'java-security-properties', Created by buildpac
<missing>                  4.27MiB   Layer: 'helper', Created by buildpack: bellsoft/buildp
<missing>                  2.97kiB
<missing>                  7.48MiB
```

Result

```
ID           TAG               SIZE      COMMAND
2774e3f214 petclinic:latest 0B        Buildpacks Process Types
<missing>                  1.44kiB   Buildpacks Launcher Config
<missing>                  2.44MiB   Buildpacks Application Launcher
<missing>                  59.17MiB  Application Layer
<missing>                  729.21kiB Software Bill-of-Materials
<missing>                  97.87MiB  Layer: 'jre', Created by buildpack: bellsoft/buildpacl
<missing>                  200.00B   Layer: 'java-security-properties', Created by buildpac
<missing>                  4.27MiB   Layer: 'helper', Created by buildpack: bellsoft/buildp
<missing>                  2.97kiB
<missing>                  7.48MiB
```

Buildpacks

Why do we need them?

- Buildpacks transform your application source code into container images
- The Paketo open source project provides production-ready buildpacks for the most popular languages and frameworks
- Use Paketo Buildpacks to easily build your apps and keep them updated

- Reminds of s2i images
- Build better than default
- Spring-aware

Is this all?

We've optimized pull/push size, right?

Optimization is multidimensional

When startup time is more important...

There are several solutions for the Spring application startup time

1. Class Data Sharing (CDS)
2. Coordinated Restore at Checkpoint
3. Native Image

Class Data Sharing (CDS)

- **When**: Java 5 (!)
- **Why**: reduce the startup and memory footprint of multiple JVM instances running on the same host
- **How**: stores data in an archive

How does it help us?

Class Data Sharing (CDS)

- **When**: Java 5 (!)
- **Why**: reduce the startup and memory footprint of multiple JVM instances running on the same host
- **How**: stores data in an archive

How does it help us?

It does not! 

Class Data Sharing (CDS)

- **When**: Java 5 (!)
- **Why**: reduce the startup and memory footprint of multiple JVM instances running on the same host
- **How**: stores data in an archive

How does it help us?

It does not!  But

Application Class Data Sharing

- **When**: Java 10
- **Why**: add application classes to the archive

And then:

- **When**: Java 13, JEP 350
- **Why**: allow addition of classes into the archive upon app exit!

And this helps! How?

How to use AppCDS with Spring?

- `-XX:ArchiveClassesAtExit=application.jsa` to create an archive
- `-Dspring.context.exit=onRefresh` to start and immediately exit the application

NB:

1. Use the same JDK
2. Use the same arguments

And even better!

Spring AOT

Add `-Dspring.aot.enabled=true`

Even more classes!!!

Practice

```
1  FROM bellsoft/liberica-runtime-container:jdk-musl as builder
2
3  COPY . /app
4  RUN cd /app && ./gradlew build -xtest
5
6  FROM bellsoft/liberica-runtime-container:jre-slim-musl as optimizer
7
8  COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
9  WORKDIR /app
10 RUN java -Djarmode=tools -jar /app/app.jar extract --layers --launcher
11
12 FROM bellsoft/liberica-runtime-container:jre-slim-musl as runner
13
14 ENTRYPOINT ["java", "org.springframework.boot.loader.launch.JarLauncher"]
15 COPY --from=optimizer /app/app/dependencies/ ./
16 COPY --from=optimizer /app/app/spring-boot-loader/ ./
17 COPY --from=optimizer /app/app/snapshot-dependencies/ ./
18 COPY --from=optimizer /app/app/application/ ./
```

Practice

```
1  FROM bellsoft/liberica-runtime-container:jdk-musl as builder
2
3  COPY . /app
4  RUN cd /app && ./gradlew build -xtest
5
6  FROM bellsoft/liberica-runtime-container:jre-slim-musl as optimizer
7
8  COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
9  WORKDIR /app
10 RUN java -Djarmode=tools -jar /app/app.jar extract --layers --launcher
11
12 FROM bellsoft/liberica-runtime-container:jre-slim-musl as runner
13
14 ENTRYPOINT ["java", "org.springframework.boot.loader.launch.JarLauncher"]
15 COPY --from=optimizer /app/app/dependencies/ ./
16 COPY --from=optimizer /app/app/spring-boot-loader/ ./
17 COPY --from=optimizer /app/app/snapshot-dependencies/ ./
18 COPY --from=optimizer /app/app/application/ ./
```

Practice

```
1  FROM bellsoft/liberica-runtime-container:jdk-musl as builder
2
3  COPY . /app
4  RUN cd /app && ./gradlew build -xtest
5
6  FROM bellsoft/liberica-runtime-container:jre-slim-musl as optimizer
7
8  COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
9  WORKDIR /app
10 RUN java -Djarmode=tools -jar /app/app.jar extract --layers --launcher
11
12 FROM bellsoft/liberica-runtime-container:jre-cds-slim-musl as runner
13
14 ENTRYPOINT ["java", "org.springframework.boot.loader.launch.JarLauncher"]
15 COPY --from=optimizer /app/app/dependencies/ ./
16 COPY --from=optimizer /app/app/spring-boot-loader/ ./
17 COPY --from=optimizer /app/app/snapshot-dependencies/ ./
18 COPY --from=optimizer /app/app/application/ ./
```

Practice

```
1  FROM bellsoft/liberica-runtime-container:jdk-musl as builder
2
3  COPY . /app
4  RUN cd /app && ./gradlew build -xtest
5
6  FROM bellsoft/liberica-runtime-container:jre-slim-musl as optimizer
7
8  COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
9  WORKDIR /app
10 RUN java -Djarmode=tools -jar /app/app.jar extract --layers --launcher
11
12 FROM bellsoft/liberica-runtime-container:jre-cds-slim-musl as runner
13
14 ENTRYPOINT ["java", "org.springframework.boot.loader.launch.JarLauncher"]
15 COPY --from=optimizer /app/app/dependencies/ ./
16 COPY --from=optimizer /app/app/spring-boot-loader/ ./
17 COPY --from=optimizer /app/app/snapshot-dependencies/ ./
18 COPY --from=optimizer /app/app/application/ ./
```

Practice

```
#omitted
FROM bellsoft/liberica-runtime-container:jre-cds-slim-musl as runner

ENTRYPOINT ["java", "org.springframework.boot.loader.launch.JarLauncher"]
COPY --from=optimizer /app/app/dependencies/ ./
COPY --from=optimizer /app/app/spring-boot-loader/ ./
COPY --from=optimizer /app/app/snapshot-dependencies/ ./
COPY --from=optimizer /app/app/application/ ./
RUN java -Dspring.aot.enabled=true \
  -XX:ArchiveClassesAtExit=./application.jsa \
  -Dspring.context.exit=onRefresh \
  org.springframework.boot.loader.launch.JarLauncher
```

Practice

```
#omitted
FROM bellsoft/liberica-runtime-container:jre-cds-slim-musl as runner

ENTRYPOINT ["java", "org.springframework.boot.loader.launch.JarLauncher"]
COPY --from=optimizer /app/app/dependencies/ ./
COPY --from=optimizer /app/app/spring-boot-loader/ ./
COPY --from=optimizer /app/app/snapshot-dependencies/ ./
COPY --from=optimizer /app/app/application/ ./
RUN java -Dspring.aot.enabled=true \
  -XX:ArchiveClassesAtExit=./application.jsa \
  -Dspring.context.exit=onRefresh \
  org.springframework.boot.loader.launch.JarLauncher
```

Practice

```
#omitted
FROM bellsoft/liberica-runtime-container:jre-cds-slim-musl as runner

ENTRYPOINT ["java", \
            "-Dspring.aot.enabled=true", \
            "-XX:SharedArchiveFile=application.jsa", \
            "org.springframework.boot.loader.launch.JarLauncher"]
COPY --from=optimizer /app/app/dependencies / ./
COPY --from=optimizer /app/app/spring-boot-loader/ ./
COPY --from=optimizer /app/app/snapshot-dependencies/ ./
COPY --from=optimizer /app/app/application/ ./
RUN java -Dspring.aot.enabled=true \
  -XX:ArchiveClassesAtExit=./application.jsa \
  -Dspring.context.exit=onRefresh \
  org.springframework.boot.loader.launch.JarLauncher
```

Practice

```
#omitted
FROM bellsoft/liberica-runtime-container:jre-cds-slim-musl as runner

ENTRYPOINT ["java", \
            "-Dspring.aot.enabled=true", \
            "-XX:SharedArchiveFile=application.jsa", \
            "org.springframework.boot.loader.launch.JarLauncher"]
COPY --from=optimizer /app/app/dependencies / ./
COPY --from=optimizer /app/app/spring-boot-loader/ ./
COPY --from=optimizer /app/app/snapshot-dependencies/ ./
COPY --from=optimizer /app/app/application/ ./
RUN java -Dspring.aot.enabled=true \
  -XX:ArchiveClassesAtExit=./application.jsa \
  -Dspring.context.exit=onRefresh \
  org.springframework.boot.loader.launch.JarLauncher
```

What does it cost ?

```
1   -r--r--r--         0:0      81 MB  ├── application.jsa
```

Which is not small at all!

What does it cost ?

```
1   -r--r--r--         0:0      81 MB  ├── application.jsa
```

Which is not small at all!

But you're trading pull speed for startup speed

How much faster?

Up to 50-60%

Pushing further with CRaC

CRaC: Coordinated Restore at Checkpoint

> The CRaC (Coordinated Restore at Checkpoint) Project researches coordination of Java programs with mechanisms to checkpoint (make an image of, snapshot) a Java instance while it is executing. Restoring from the image could be a solution to some of the problems with the start-up and warm-up times. The primary aim of the Project is to develop a new standard mechanism-agnostic API to notify Java programs about the checkpoint and restore events. Other research activities will include, but will not be limited to, integration with existing checkpoint/restore mechanisms and development of new ones, changes to JVM and JDK to make images smaller and ensure they are correct.

https://openjdk.org/projects/crac/

In a perfect world it should be:

```
FROM bellsoft/liberica-runtime-container:jdk-musl as builder

COPY . /app
RUN cd /app && ./gradlew build

FROM bellsoft/liberica-runtime-container:jre-crac-slim as optimizer

COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
WORKDIR /app
RUN java -Dspring.context.checkpoint=onRefresh -XX:CRaCCheckpointTo=/checkpoint -jar /app/ap

FROM bellsoft/liberica-runtime-container:jre-crac-slim as runner

ENTRYPOINT java -XX:CRaCRestoreFrom=/checkpoint
COPY --from=optimizer /app/app.jar /app/app.jar
COPY --from=optimizer /checkpoint /checkpoint
```

In a perfect world it should be:

```
1   FROM bellsoft/liberica-runtime-container:jdk-musl as builder
2
3   COPY . /app
4   RUN cd /app && ./gradlew build
5
6   FROM bellsoft/liberica-runtime-container:jre-crac-slim as optimizer
7
8   COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
9   WORKDIR /app
10  RUN java -Dspring.context.checkpoint=onRefresh -XX:CRaCCheckpointTo=/checkpoint -jar /app/ap
11
12  FROM bellsoft/liberica-runtime-container:jre-crac-slim as runner
13
14  ENTRYPOINT java -XX:CRaCRestoreFrom=/checkpoint
15  COPY --from=optimizer /app/app.jar /app/app.jar
16  COPY --from=optimizer /checkpoint /checkpoint
```

In a perfect world it should be:

```
1   FROM bellsoft/liberica-runtime-container:jdk-musl as builder
2
3   COPY . /app
4   RUN cd /app && ./gradlew build
5
6   FROM bellsoft/liberica-runtime-container:jre-crac-slim as optimizer
7
8   COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
9   WORKDIR /app
10  RUN java -Dspring.context.checkpoint=onRefresh -XX:CRaCCheckpointTo=/checkpoint -jar /app/ap
11
12  FROM bellsoft/liberica-runtime-container:jre-crac-slim as runner
13
14  ENTRYPOINT java -XX:CRaCRestoreFrom=/checkpoint
15  COPY --from=optimizer /app/app.jar /app/app.jar
16  COPY --from=optimizer /checkpoint /checkpoint
```

But in reality

```
#12 6.051       Suppressed: java.lang.RuntimeException: Native checkpoint failed.
#12 6.051               at java.base/jdk.crac.Core.translateJVMExceptions(Core.java:114) ~[r
#12 6.051               at java.base/jdk.crac.Core.checkpointRestore1(Core.java:192) ~[na:na
#12 6.051               at java.base/jdk.crac.Core.checkpointRestore(Core.java:299) ~[na:na]
#12 6.051               at java.base/jdk.crac.Core.checkpointRestore(Core.java:278) ~[na:na]
#12 6.051               at java.base/javax.crac.Core.checkpointRestore(Core.java:73) ~[na:na
#12 6.051               at java.base/jdk.internal.reflect.DirectMethodHandleAccessor.invoke(
#12 6.051               at java.base/java.lang.reflect.Method.invoke(Method.java:580) ~[na:n
#12 6.051               at org.crac.Core$Compat.checkpointRestore(Core.java:141) ~[crac-1.4.
#12 6.051                ... 17 common frames omitted
```

CRaC is hard!

Let’s try to fix it with arcane magic

```
1  FROM bellsoft/liberica-runtime-container:jdk-musl as builder
2
3  COPY . /app
4  RUN cd /app && ./gradlew build -xtest
5
6  FROM bellsoft/liberica-runtime-container:jre-crac-slim as optimizer
7
8  COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
9  WORKDIR /app
10 RUN java -Dspring.context.checkpoint=onRefresh -XX:CRaCCheckpointTo=/checkpoint -jar /app/app.jar
11
12 FROM bellsoft/liberica-runtime-container:jre-crac-slim as runner
13
14 ENTRYPOINT java -XX:CRaCRestoreFrom=/checkpoint
15 COPY --from=optimizer /app/app.jar /app/app.jar
16 COPY --from=optimizer /checkpoint /checkpoint
```

CRaC is hard!

Let’s try to fix it with arcane magic

```
# syntax=docker/dockerfile:1-labs
FROM bellsoft/liberica-runtime-container:jdk-musl as builder

COPY . /app
RUN cd /app && ./gradlew build -xtest

FROM bellsoft/liberica-runtime-container:jre-crac-slim as optimizer

COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
WORKDIR /app
RUN java -Dspring.context.checkpoint=onRefresh -XX:CRaCCheckpointTo=/checkpoint -jar /app/app.jar

FROM bellsoft/liberica-runtime-container:jre-crac-slim as runner

ENTRYPOINT java -XX:CRaCRestoreFrom=/checkpoint
COPY --from=optimizer /app/app.jar /app/app.jar
COPY --from=optimizer /checkpoint /checkpoint
```

CRaC is hard!

Let’s try to fix it with arcane magic

```
# syntax=docker/dockerfile:1-labs
FROM bellsoft/liberica-runtime-container:jdk-musl as builder

COPY . /app
RUN cd /app && ./gradlew build -xtest

FROM bellsoft/liberica-runtime-container:jre-crac-slim as optimizer

COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
WORKDIR /app
RUN java -Dspring.context.checkpoint=onRefresh -XX:CRaCCheckpointTo=/checkpoint -jar /app/app.jar

FROM bellsoft/liberica-runtime-container:jre-crac-slim as runner

ENTRYPOINT java -XX:CRaCRestoreFrom=/checkpoint
COPY --from=optimizer /app/app.jar /app/app.jar
COPY --from=optimizer /checkpoint /checkpoint
```

CRaC is hard!

Let’s try to fix it with arcane magic

```
# syntax=docker/dockerfile:1-labs
FROM bellsoft/liberica-runtime-container:jdk-musl as builder

COPY . /app
RUN cd /app && ./gradlew build -xtest

FROM bellsoft/liberica-runtime-container:jre-crac-slim as optimizer

COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
WORKDIR /app
RUN java -Dspring.context.checkpoint=onRefresh -XX:CRaCCheckpointTo=/checkpoint -jar /app/app.jar

FROM bellsoft/liberica-runtime-container:jre-crac-slim as runner

ENTRYPOINT java -XX:CRaCRestoreFrom=/checkpoint
COPY --from=optimizer /app/app.jar /app/app.jar
COPY --from=optimizer /checkpoint /checkpoint
```

CRaC is hard!

Let’s try to fix it with arcane magic

```
# syntax=docker/dockerfile:1-labs
FROM bellsoft/liberica-runtime-container:jdk-musl as builder

COPY . /app
RUN cd /app && ./gradlew build -xtest

FROM bellsoft/liberica-runtime-container:jre-crac-slim as optimizer

COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
WORKDIR /app
RUN --security=insecure java -Dspring.context.checkpoint=onRefresh \
  -XX:CRaCCheckpointTo=/checkpoint -jar /app/app.jar

FROM bellsoft/liberica-runtime-container:jre-crac-slim as runner

ENTRYPOINT java -XX:CRaCRestoreFrom=/checkpoint
```

CRaC is hard!

Let’s try to fix it with arcane magic

```
# syntax=docker/dockerfile:1-labs
FROM bellsoft/liberica-runtime-container:jdk-musl as builder

COPY . /app
RUN cd /app && ./gradlew build -xtest

FROM bellsoft/liberica-runtime-container:jre-crac-slim as optimizer

COPY --from=builder /app/build/libs/spring-petclinic-3.3.0.jar /app/app.jar
WORKDIR /app
RUN --security=insecure java -Dspring.context.checkpoint=onRefresh \
  -XX:CRaCCheckpointTo=/checkpoint -jar /app/app.jar || true

FROM bellsoft/liberica-runtime-container:jre-crac-slim as runner

ENTRYPOINT java -XX:CRaCRestoreFrom=/checkpoint
```

Pasha

And this is not all!

Did you hear about `buildx` ?

```
docker buildx create --buildkitd-flags \
    '--allow-insecure-entitlement security.insecure' \
    --name insecure-builder
```

And this is not all!

Did you hear about `buildx` ?

```
docker buildx create --buildkitd-flags \
    '--allow-insecure-entitlement security.insecure' \
    --name insecure-builder
```

And this is not all!

Did you hear about `buildx` ?

```
1  docker buildx use insecure-builder
```

And this is not all!

Did you hear about `buildx` ?

```
docker buildx build \
    --allow security.insecure \
    -f Dockerfile.crac \
    -t pet-crac --output type=docker .
```

And this is not all!

Did you hear about `buildx` ?

```
docker buildx build \
    --allow security.insecure \
    -f Dockerfile.crac \
    -t pet-crac --output type=docker .
```

And this is not all!

Did you hear about `buildx` ?

```
docker buildx build \
    --allow security.insecure \
    -f Dockerfile.crac \
    -t pet-crac --output type=docker .
```

And this is not all!

Did you hear about `buildx` ?

```
1  docker run --rm -it --privileged pet-crac
```

And this is not all!

Did you hear about `buildx`?

```
Restarting Spring-managed lifecycle beans after JVM restore
HikariPool-1 - Thread starvation or clock leap detected (housekeeper delta=1d19h37m42s102ms798µs806ns)
Tomcat started on port 8080 (http) with context path '/'
Restored PetClinicApplication in 0.186 seconds (process running for 0.19)
```

And this is not all!

Did you hear about `buildx`?

```
Restarting Spring-managed lifecycle beans after JVM restore
HikariPool-1 - Thread starvation or clock leap detected (housekeeper delta=1d19h37m42s102ms798µs806ns)
Tomcat started on port 8080 (http) with context path '/'
Restored PetClinicApplication in 0.186 seconds (process running for 0.19)
```

Is it a good solution?

It depends

Is it a good solution?

It depends

- If "It Works" is enough for you

Is it a good solution?

It depends

- If "It Works" is enough for you -> YES
- If you need more predictable and maintanable thing

Is it a good solution?

It depends

- If "It Works" is enough for you -> YES
- If you need more predictable and maintanable thing -> NO

How to make it better?

How to make it better?

1. Build JAR (in docker or not)

How to make it better?

1. Build JAR (in docker or not)
2. Create new image that will run the JAR with CRaC arguments in `ENTRYPOINT`

How to make it better?

1. Build JAR (in docker or not)
2. Create new image that will run the JAR with CRaC arguments in `ENTRYPOINT`
3. Run the container with capabilities:

How to make it better?

1. Build JAR (in docker or not)
2. Create new image that will run the JAR with CRaC arguments in `ENTRYPOINT`
3. Run the container with capabilities:
    1. CAP_SYS_PTRACE

How to make it better?

1. Build JAR (in docker or not)
2. Create new image that will run the JAR with CRaC arguments in `ENTRYPOINT`
3. Run the container with capabilities:
    1. CAP_SYS_PTRACE
    2. CAP_CHECKPOINT_RESTORE

How to make it better?

1. Build JAR (in docker or not)
2. Create new image that will run the JAR with CRaC arguments in `ENTRYPOINT`
3. Run the container with capabilities:
    1. CAP_SYS_PTRACE
    2. CAP_CHECKPOINT_RESTORE
4. Container will run and stop

How to make it better?

1. Build JAR (in docker or not)
2. Create new image that will run the JAR with CRaC arguments in `ENTRYPOINT`
3. Run the container with capabilities:
    1. CAP_SYS_PTRACE
    2. CAP_CHECKPOINT_RESTORE
4. Container will run and stop
5. Commit the container like

```
docker commit container-id new-tag
```

Pros and Cons

Pros:

1. Does not require arcane magic
2. Works more predictably
3. Does not depend on unstable features
4. Does not require privileged containers in

Cons:

1. Organization-specific
2. Requires more steps

And now we have all flavours of ultimate docker images!

Quick summary?

1. Use layers for faster deployment
2. Use CDS for faster startup without many compromises
3. Use CRaC for a *lightning-fast* startup (with compromises)
4. Use native image if you're prepared to sacrifice some build time

Thank you!

Find me @

- @asm0dey.site
- @asm0dey@fosstodon.org
- asm0di0
- me@asm0dey.site
- asm0dey

END