Blocks in containers

lessons learned from containerizing Minecraft
Julia Ferraioli
Software Engineer
Open Source @ Google
Empathy in tech advocate
ML enthusiast
I’m addicted to Minecraft
Okay, I’m addicted to Minecraft and Lego
Alright, I’m addicted to Minecraft, Lego, and cloud
Fine! I’m addicted to Minecraft, Lego, cloud, and containers
Let’s talk about containers
14%-36%

Running in production


@juliaferraioli
What’s the barrier here?
We have educational resources
We have cloud provider support
We have meetup upon meetup
Missing the link between here and awesometown
We need a toy... problem
What sort of fully-baked apps do we have...

● ...that **predate** the surge in container popularity?

● ...that are **understood/used** by many?

● ...that are **complex** enough that containerization makes sense

...but **simple** enough to tackle?
Phase 1: experimentation
Uh, hello Minecraft?

What does Minecraft need?

Java. It needs Java.

FROM debian:latest
RUN apt-get update && apt-get install -y curl openjdk-7-jre

ENV MC_VERSION 1.8.9
ENV BASE_URL https://s3.amazonaws.com/Minecraft.Download/versions
ENV SERVER_FILE minecraft_server

RUN mkdir /opt/mc
WORKDIR /opt/mc

RUN curl $BASE_URL/$MC_VERSION/$SERVER_FILE.$MC_VERSION.jar \\
    -o minecraft.jar && chmod u+x minecraft.jar

CMD java -Xmx1024M -Xms1024M -jar minecraft.jar nogui
This is kind of old hat
Where’s the magic?
Phase 2: cautious containerization
Building and testing

➜ ~ docker build -t juliaferraioli/mc:v1 .

➜ ~ docker run -p 25565:25565 -d juliaferraioli/mc:v1
d7d5e58f8833952b51d33c9c63a2b763f1a290d33270245a518d17d689f45f39

➜ ~ docker ps

<table>
<thead>
<tr>
<th>CONTAINER ID</th>
<th>IMAGE</th>
<th>COMMAND</th>
<th>CREATED</th>
<th>STATUS</th>
<th>PORTS</th>
<th>NAMES</th>
</tr>
</thead>
</table>

well, 🖦.
Phase 3: problem identification
I build and run but nothing happens!

`-d` means detached =>

when our CMD exits, the container exits =>

`java -Xmx1024M -Xms1024M -jar minecraft.jar nogui` must have exited
Okay, let’s remove the flag

→ ~ docker run -p 25565:25565 juliaferraioli/mc:v1

[16:33:54] [Server thread/INFO]: Starting minecraft server version 1.9.2
[16:33:54] [Server thread/INFO]: Loading properties
[16:33:54] [Server thread/WARN]: server.properties does not exist
[16:33:54] [Server thread/INFO]: Generating new properties file
[16:33:54] [Server thread/WARN]: Failed to load eula.txt
[16:33:54] [Server thread/INFO]: You need to agree to the EULA in order to run
the server. Go to eula.txt for more info.
[16:33:54] [Server thread/INFO]: Stopping server
Ohhhhhh...

startServer.sh

- If there’s already an accepted EULA...
- or if we accepted through the ENV...
- go ahead and start the server.
- Else, prompt the user to read the EULA and exit the script
Uh, hello Minecraft? (Reprise)

What does Minecraft need?
Java. It needs Java.
And an accepted EULA.

FROM debian:latest
RUN apt-get update & & apt-get install -y curl openjdk-7-jre

ENV MC_VERSION 1.8.9
ENV BASE_URL https://s3.amazonaws.com/Minecraft.Download/versions
ENV SERVER_FILE minecraft_server

RUN mkdir /opt/mc
WORKDIR /opt/mc

COPY startServer.sh /opt/mc/

RUN curl $BASE_URL/$MC_VERSION/$SERVER_FILE.$MC_VERSION.jar \\
    -o minecraft.jar & & chmod u+x minecraft.jar startServer.sh

CMD ./startServer.sh

@juliaferraioli
Lesson(s)
Storage becomes complicated
Storage stays complicated
Ephemerality is hard to grok
Phase 4: containerizing ALL the things!
I’m the conductor of this orchestra
I’m the conductor of this orchestra

→ ~ kubectl run mc --image=gcr.io/octopodes-go-bloop.mc:v2 \ 
  --env="EULA=true" --port=25565

→ ~ kubectl expose deployment mc type=LoadBalancer

→ ~ kubectl describe services mc
Phase 5: freedom!
Breaking out of the machine
Lack of boundaries is freeing
Phase 6: absurdity
Resources

Thanks!
Uh oh!

@juliahferraioli