
AWS
re:Invent
NOV. 27 – DEC. 1, 2023 | LAS VEGAS, NV

CON307

Reducing AWS Fargate startup times by lazy loading container images

Himanshu Kohli
(he/him)
Senior Product Manager, Containers
Amazon Web Services

Olly Pomeroy
(he/him)
Senior Developer Advocate, Containers
Amazon Web Services

AWS Fargate

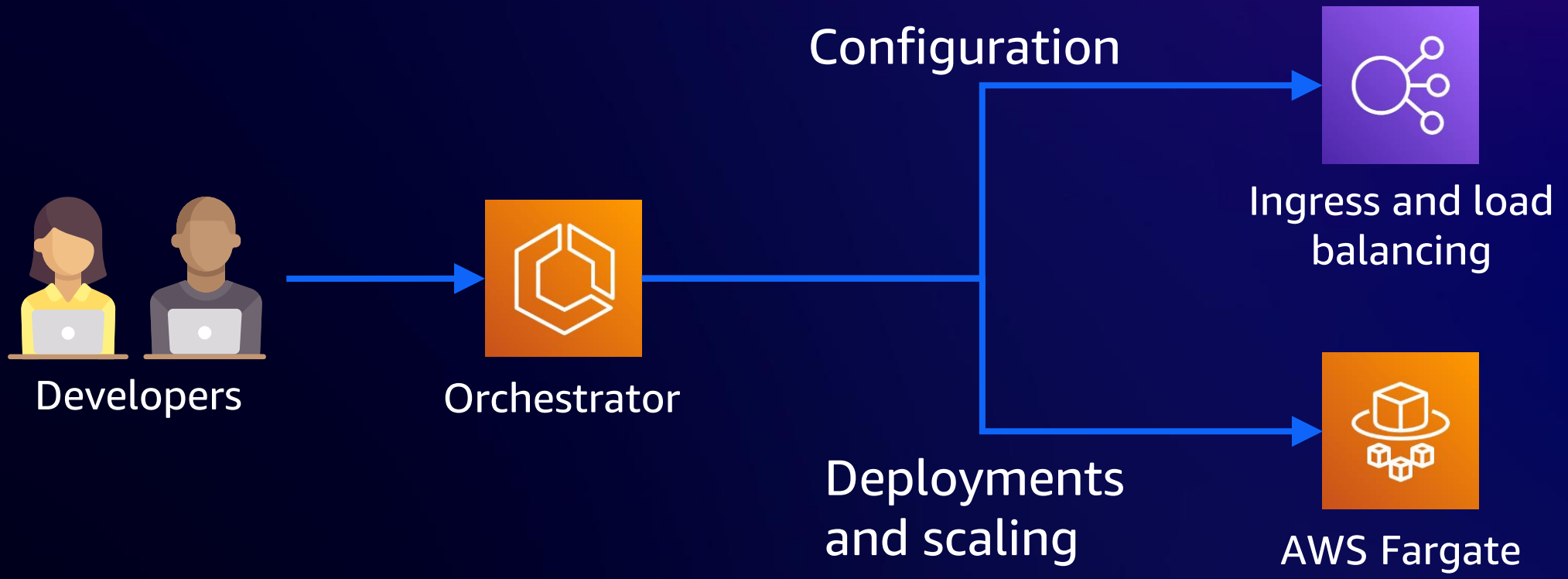
Fargate is a serverless compute engine for containers

- **Managed by AWS**: No AMIs to maintain, no Amazon EC2 instances to provision, scale, or manage

- **Secure**: Isolated, patched, and compliant for running the most sensitive workloads

- **On-demand pricing**: Pay for what you need

Workloads are scheduled by a container orchestrator on to AWS Fargate



Customers are standardizing on AWS Fargate

Frontend and backend API services

Queue-based message or data processing

Modernizing legacy Java/.NET workloads

CPU-based ML training and inferencing

Batch and parallel processing

Data processing (ETL) pipelines

Improving the scale and performance of AWS Fargate

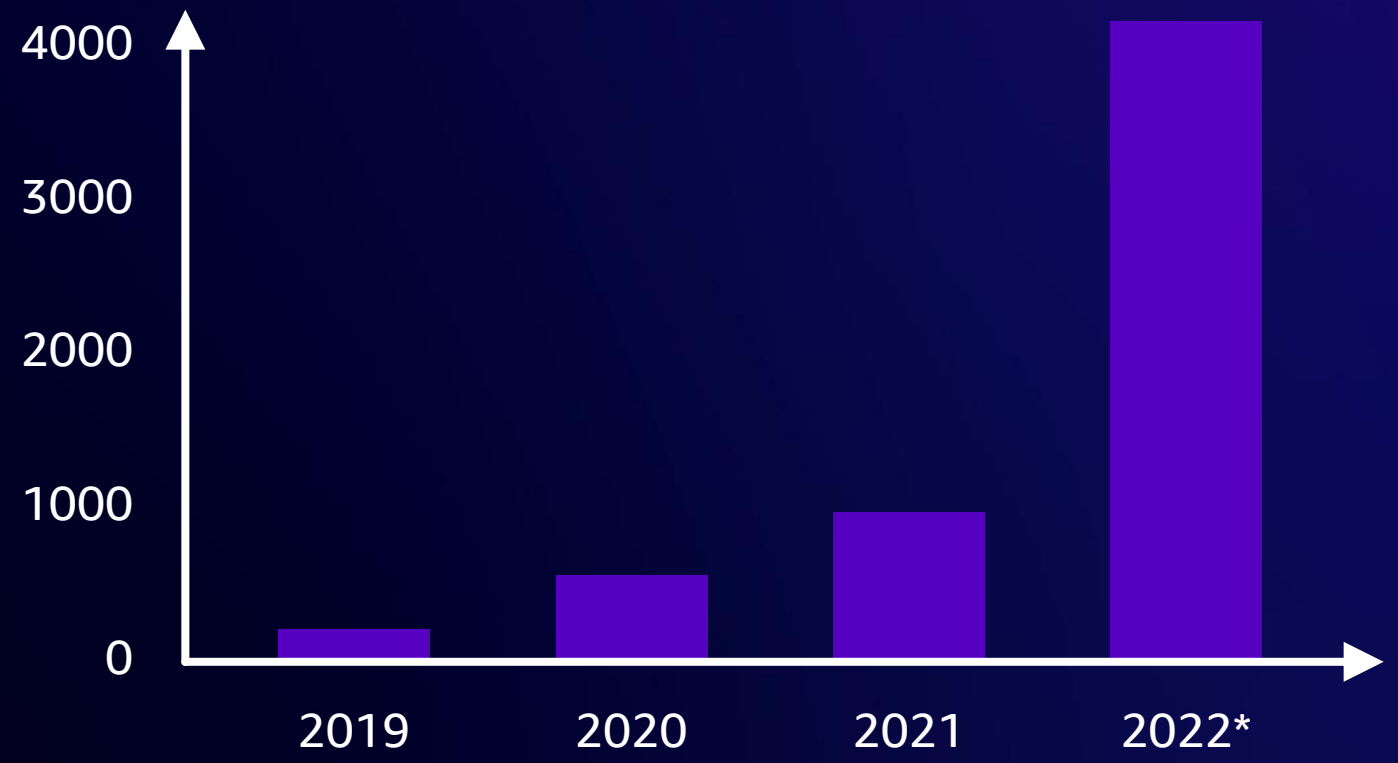
Increase the number of concurrent tasks 

Decrease the time to scale out 

Increase the size of a task 

Improve the task startup time 

Increasing the number of concurrent tasks



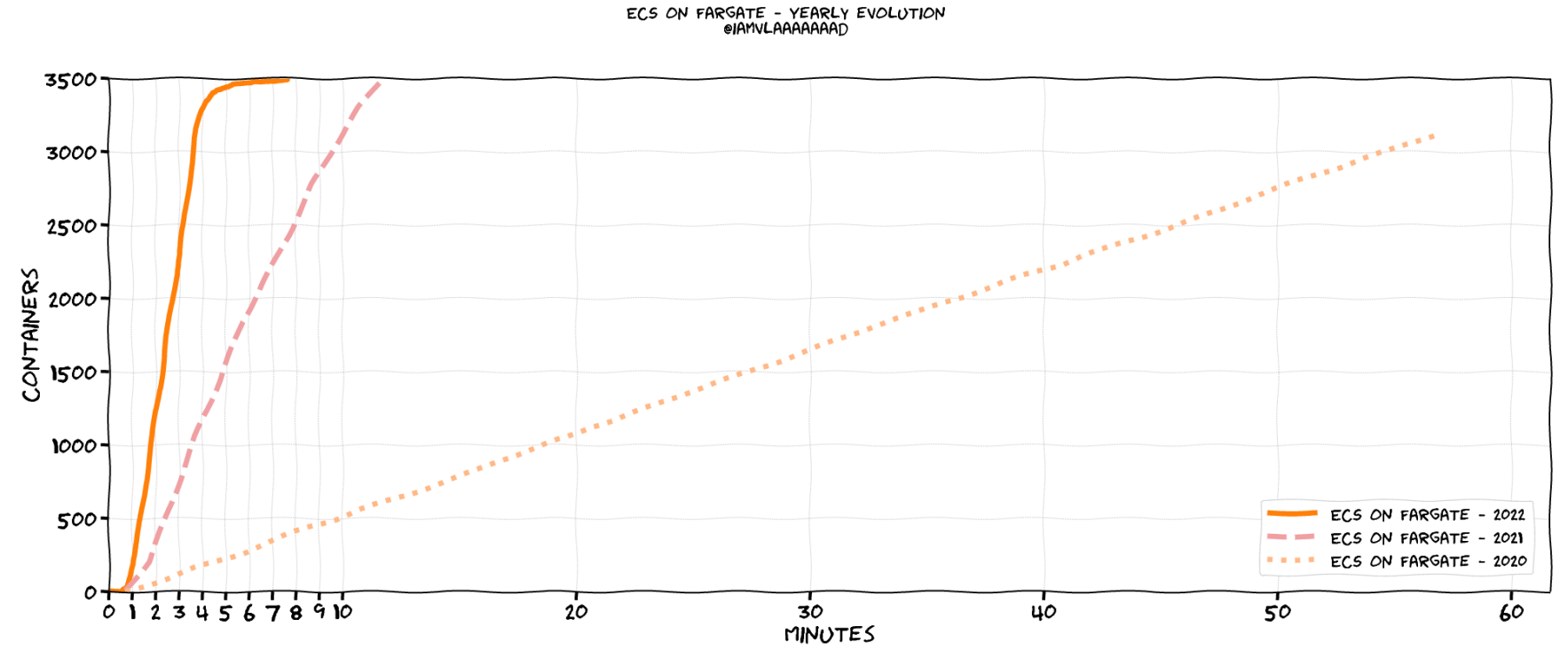
* Based on AWS account soft limits with a 1vCPU ECS task
https://aws.amazon.com/blogs/containers/migrating-fargate-service-quotas-to-vcpu-based-quotas/

Increasing the size of a task

CPU value	Memory value
.25 vCPU	Up to 2 GB
.5 vCPU	Up to 4 GB
1 vCPU	Up to 8 GB
2 vCPU	Up to 16GB
4 vCPU	Up to 30GB
8 vCPU	**Up to 60GB**
16 vCPU	**Up to 120GB**

https://aws.amazon.com/about-aws/whats-new/2022/09/aws-fargate-increases-compute-memory-resource-configurations-4x/

Decreasing the time to scale out



Launch up to 500 tasks in an Amazon ECS service in under a minute.

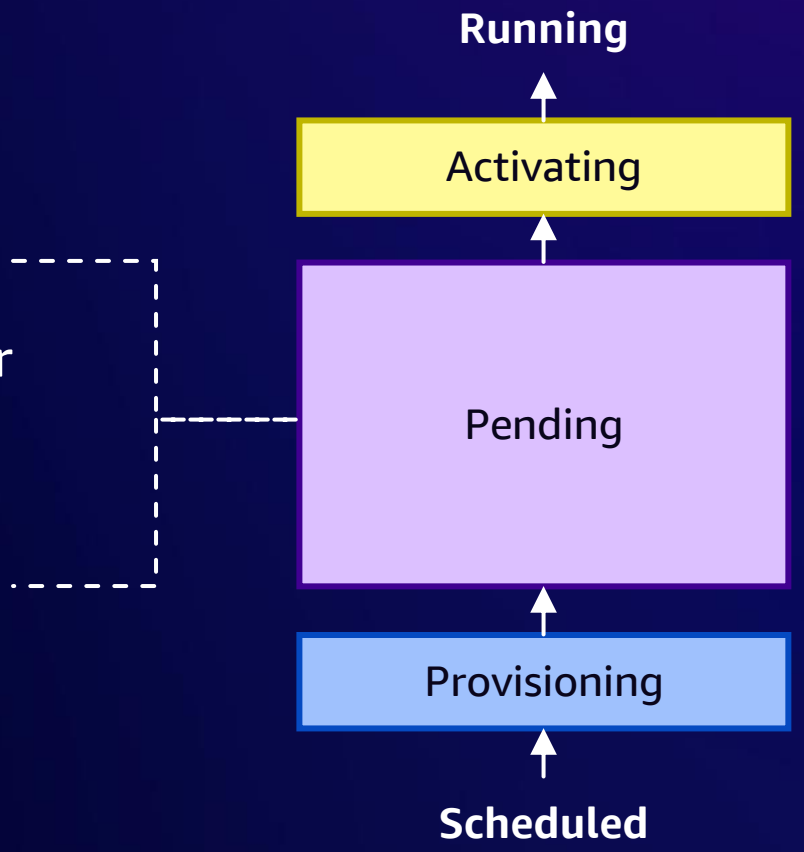
16X faster than in 2021!

https://www.vladionescu.me/posts/scaling-containers-on-aws-in-2022/

https://aws.amazon.com/blogs/containers/under-the-hood-amazon-elastic-container-service-and-aws-fargate-increase-task-launch-rates/

Improving AWS Fargate task startup times

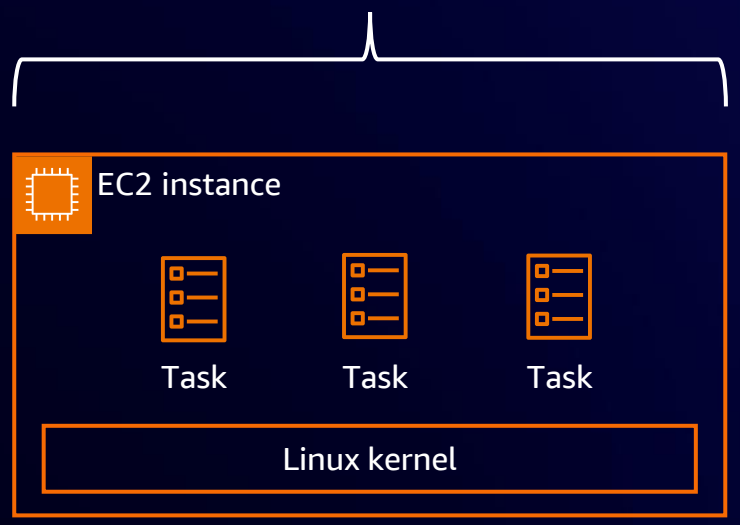
Downloading and extracting the container image(s) has the largest impact on task launch times on AWS Fargate



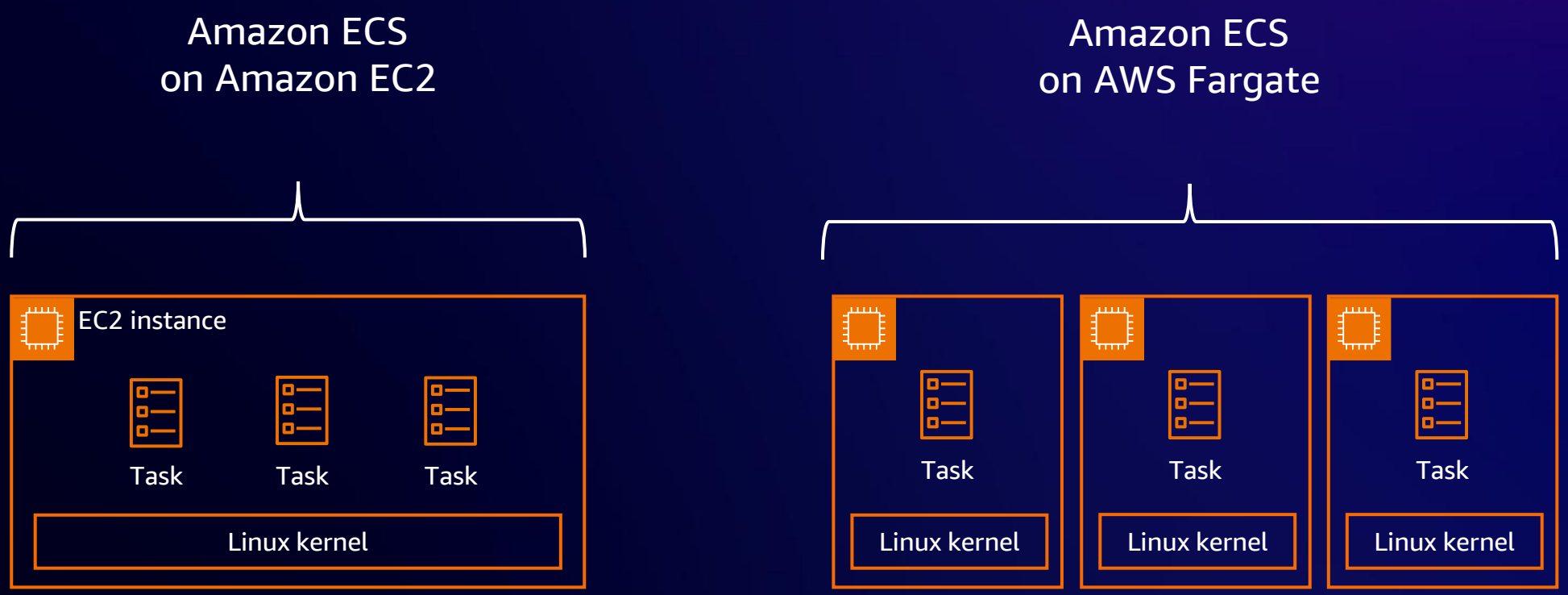
https://docs.aws.amazon.com/AmazonECS/latest/developerguide/task-lifecycle.html

AWS Fargate architecture

AWS Fargate has a strong task isolation boundary

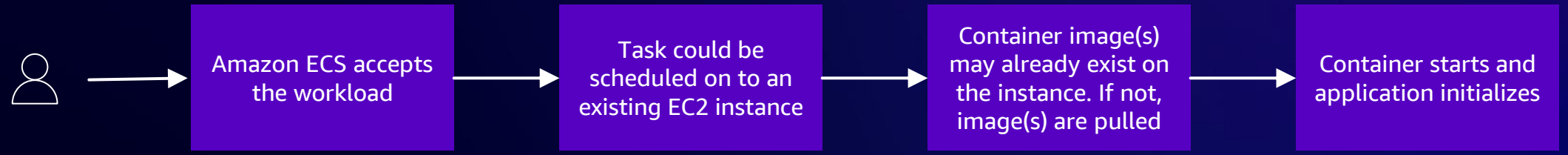


AWS Fargate has a strong task isolation boundary



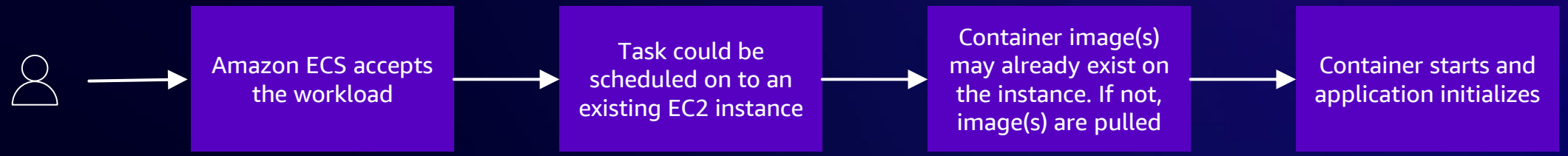
The trade-off for a strong isolation boundary

Deploying an Amazon ECS task on Amazon EC2

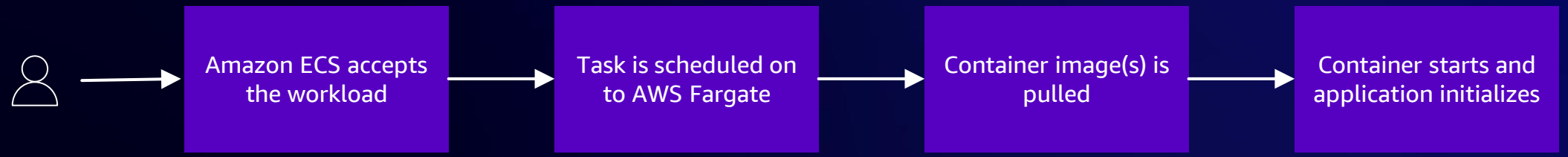


The trade-off for a strong isolation boundary

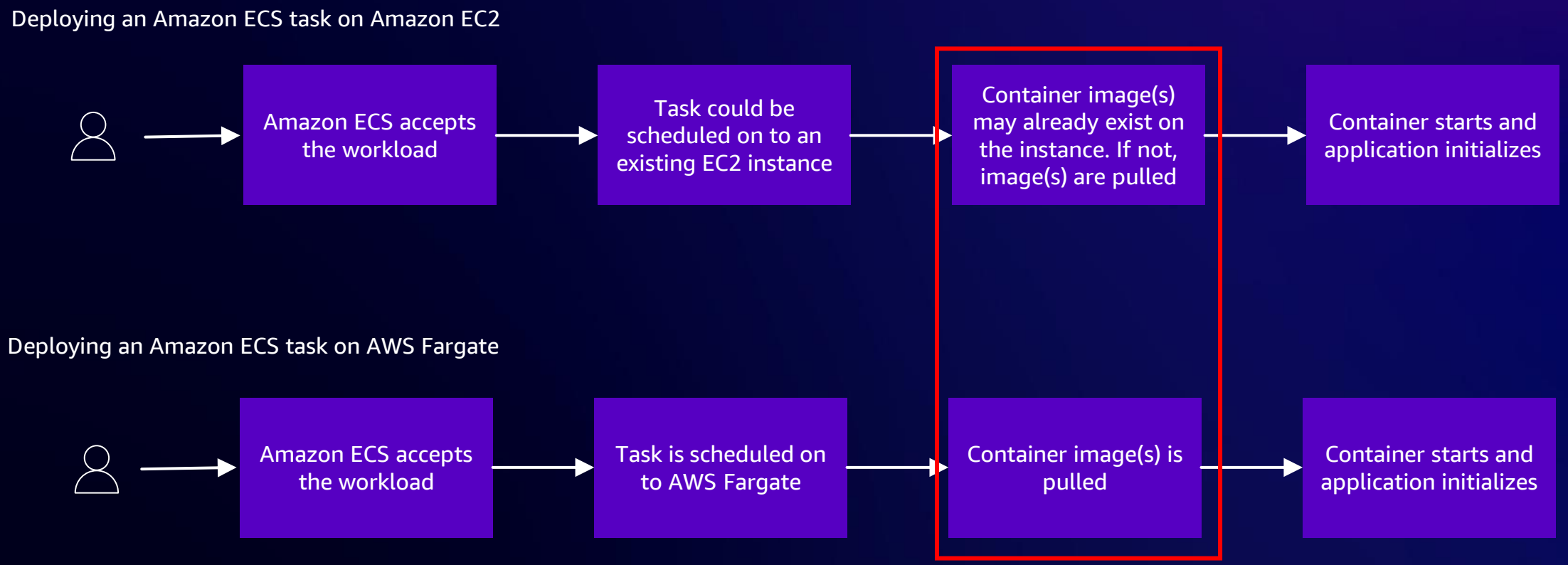
Deploying an Amazon ECS task on Amazon EC2



Deploying an Amazon ECS task on AWS Fargate

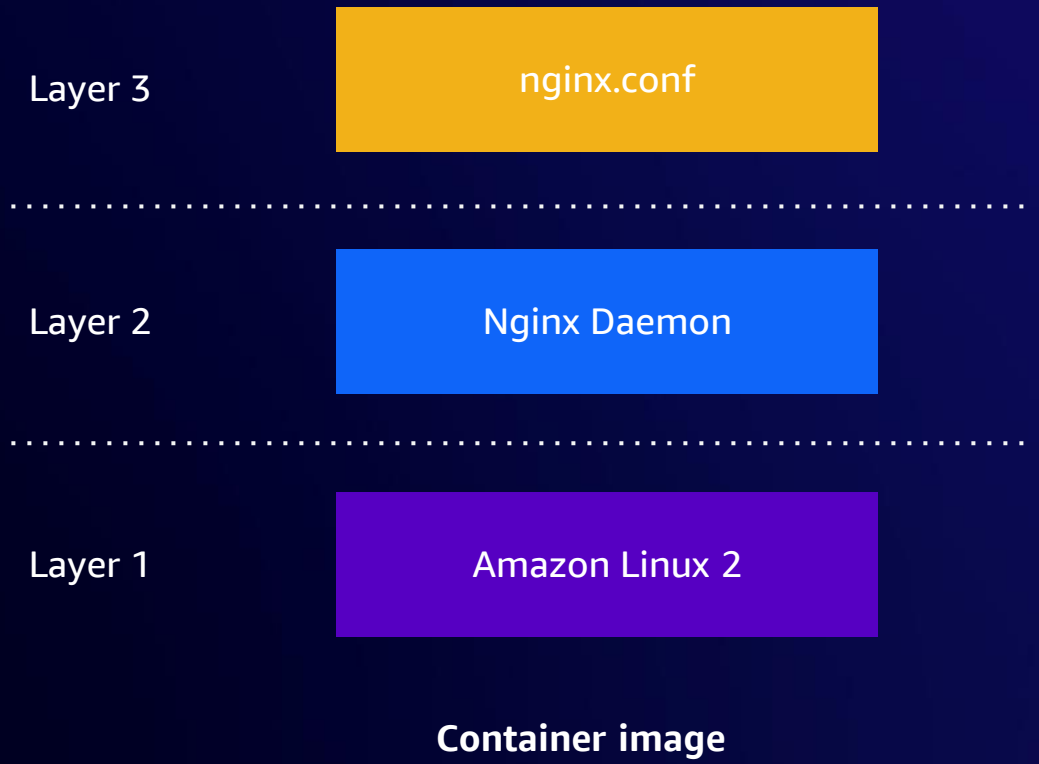


The trade-off for a strong isolation boundary



Reducing the time taken to download a container image

A container image consists of multiple layers overlayed on top of each other at runtime



Each layer corresponds to the Dockerfile line it was built from

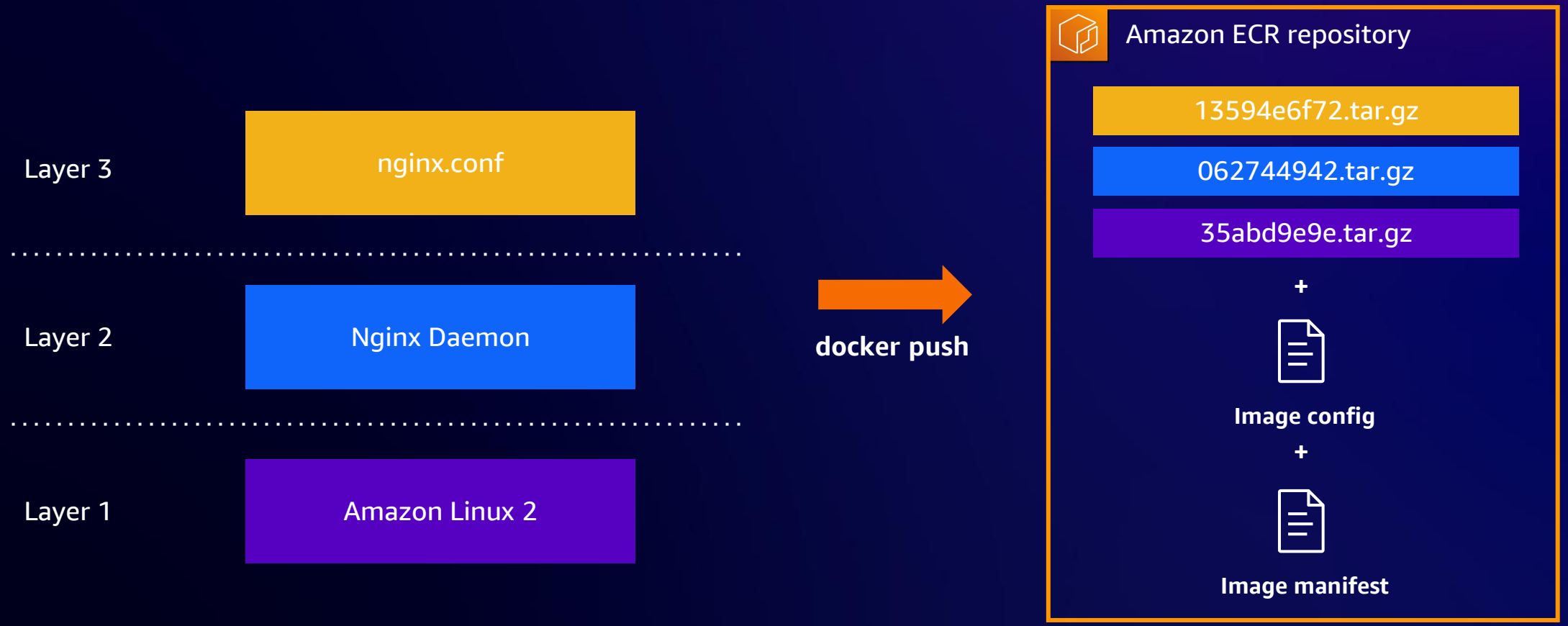
```
FROM amazonlinux:2

RUN amazon-linux-extras install nginx1.12 && \
    yum install nginx -y

COPY nginx.conf nginx.conf
```

Dockerfile

docker build

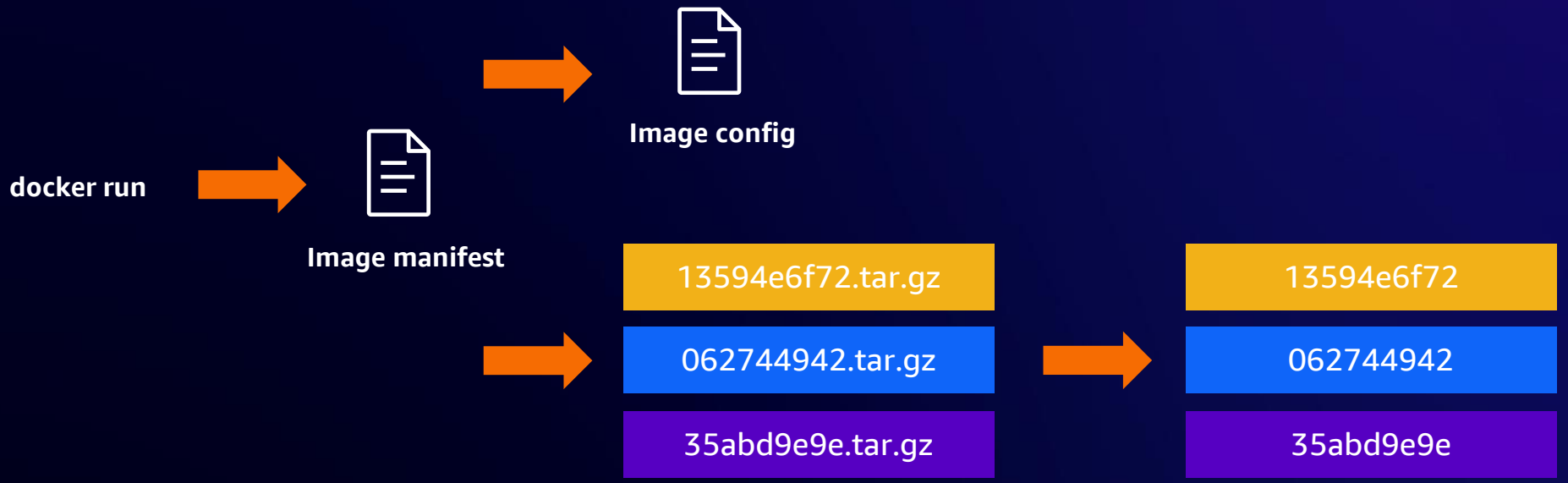
Container image

When an image is pushed to a registry, each layer is compressed and stored individually

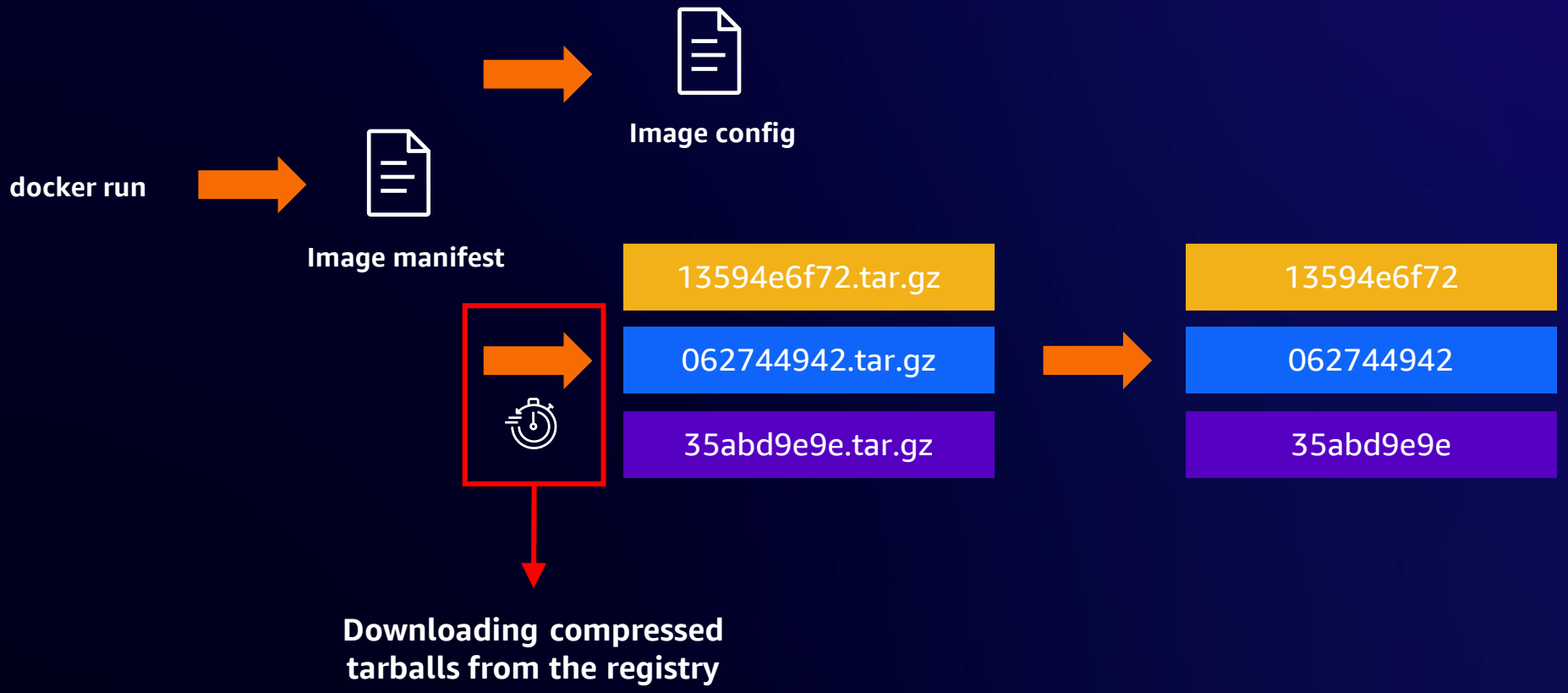


Before starting a container, the container runtime needs a local copy of the container image

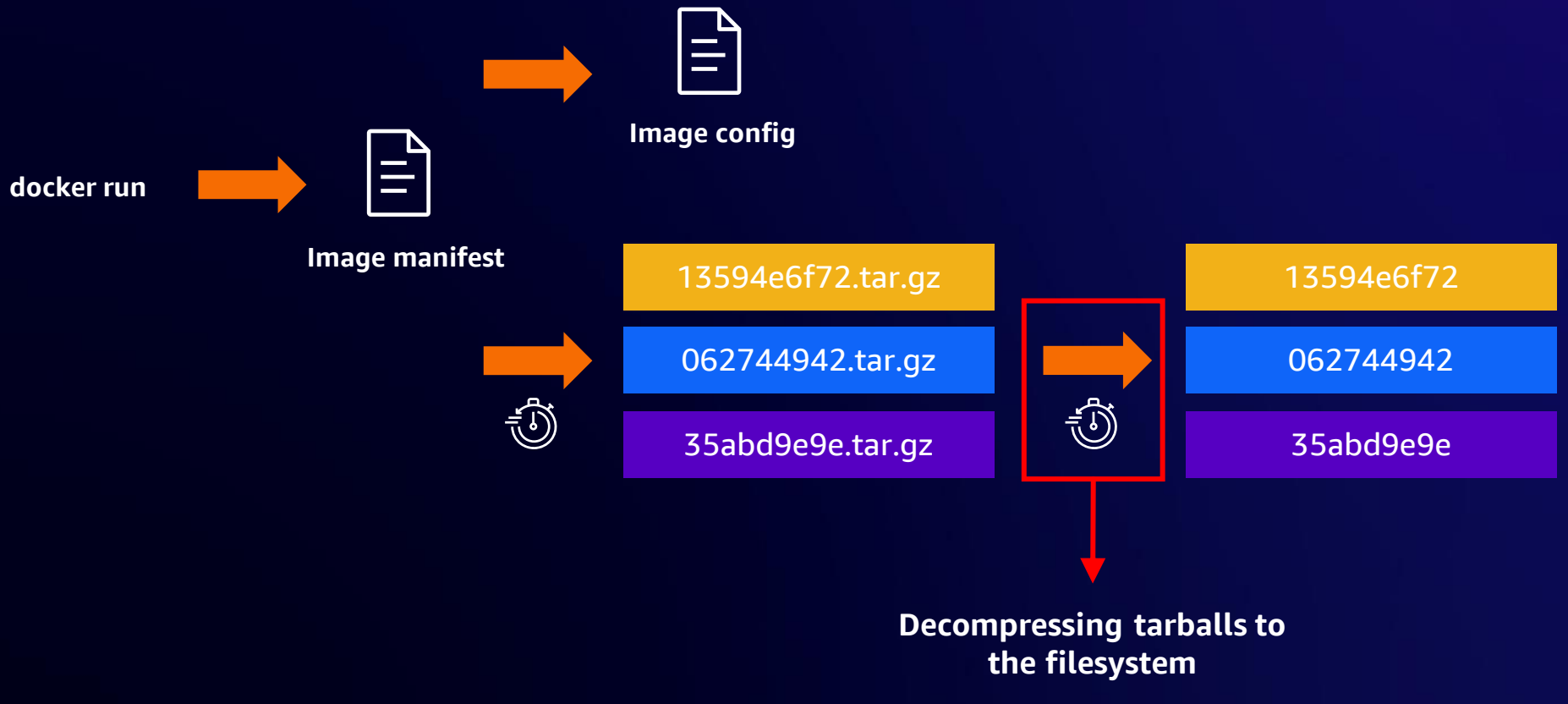
The container runtime first downloads the metadata and then each filesystem layer



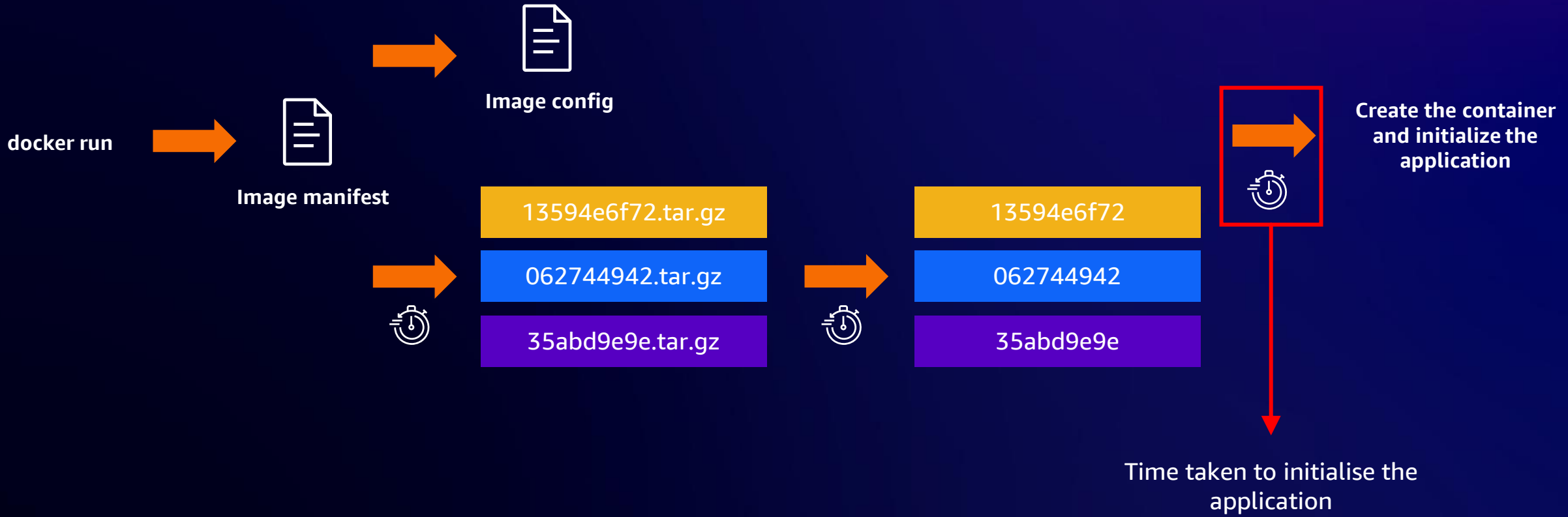
Pulling a container image can be slow, time is lost in downloading the tarballs



Pulling a container image can be slow, time is lost in extracting the tarballs



Once the content exists locally, we can create a container and initialize the application



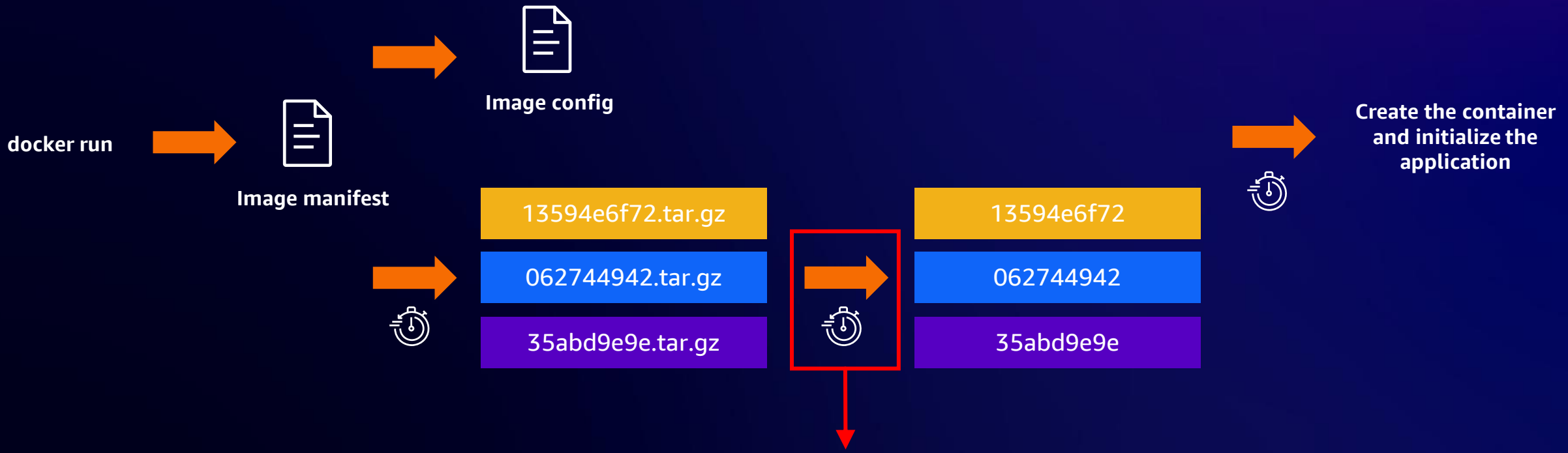
> **Our analysis shows that copying package data accounts for 76% of container start-up time, only 6.4% of the copied data is actually needed for containers to begin useful work.**

Harter et al. study, Slacker: Fast Distribution with Lazy Docker Containers

https://www.usenix.org/conference/fast16/technical-sessions/presentation/harter

As AWS Fargate launch times are heavily impacted by container pull times, what can we optimize?

Is there more efficient way to decompress images?

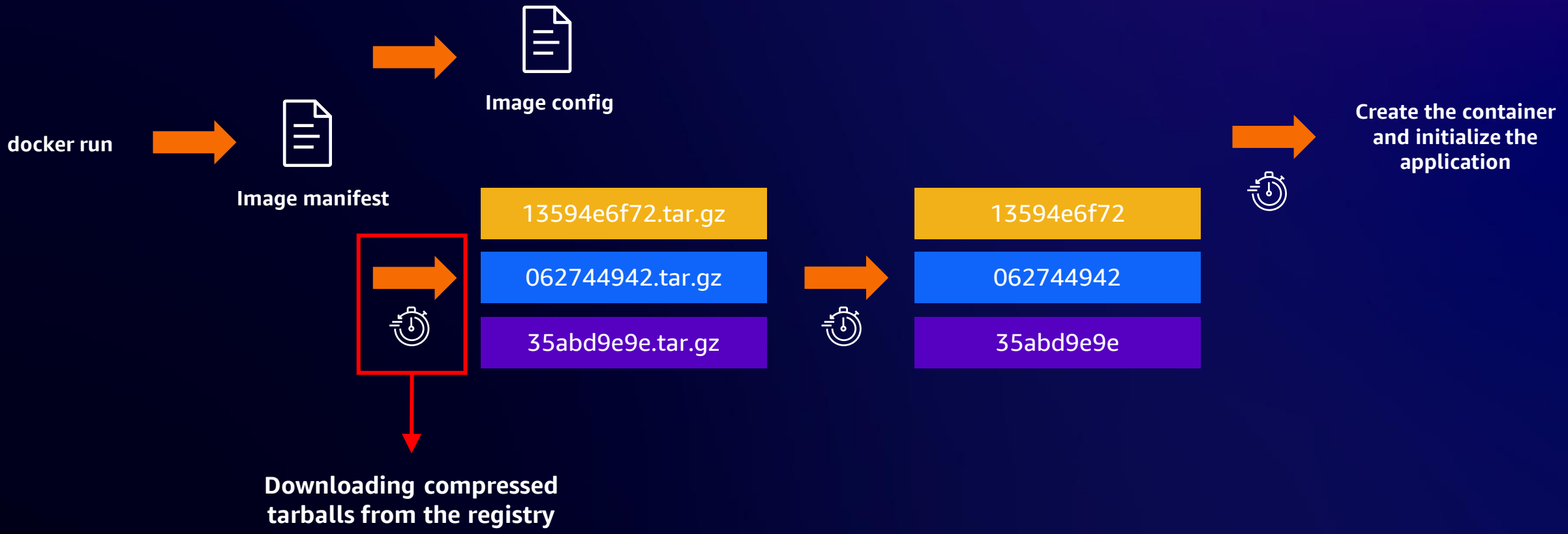


AWS Fargate now supports container images compressed with the zstd compression format

"In our internal testing of zstd compressed container images on AWS Fargate, we have seen up to a 27% reduction in Amazon ECS task or Kubernetes pod startup times."

https://aws.amazon.com/blogs/containers/reducing-aws-fargate-startup-times-with-zstd-compressed-container-images/

Can we reduce how much data we need to download?



The time taken to download a container is directly proportional to the size of the container image

Image	Size (MB)
Amazon Linux 2023	52.4
Amazon Linux 2023 Minimal	35.2

Choose a smaller base image

Image tag	Size (MB)
Full test suite	2560.2
Test suite 1	305.7
Test suite 2	512.1

Remove all unnecessary binaries and libraries from the container

Reduce the number of layers in the container image

Start the workload instantaneously and then lazily load the container image

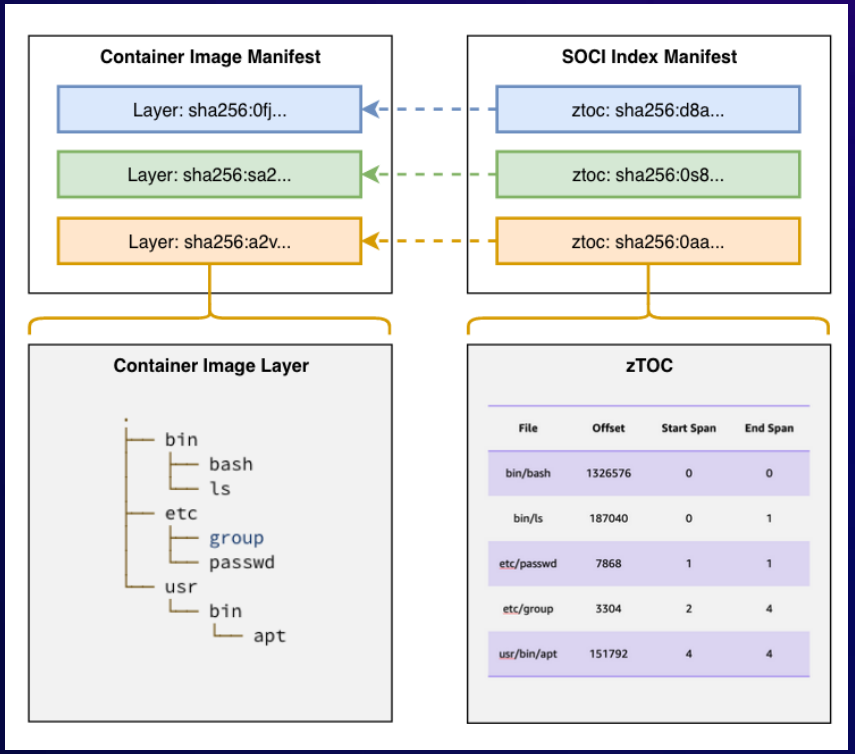
Lazy loading container images is the concept of downloading container image data from the image repository when the application needs it.

Seekable OCI (SOCI)

Introducing Seekable OCI (SOCI)

Seekable OCI (SOCI) is a technology open sourced by AWS that enables containers to launch faster **by lazily loading the container image.**

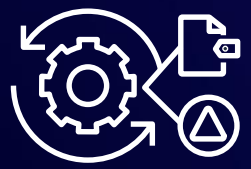
It provides the capability to **extract a subset of an existing container image** instead of downloading the entire container image.



https://aws.amazon.com/about-aws/whats-new/2022/09/introducing-seekable-oci-lazy-loading-container-images/

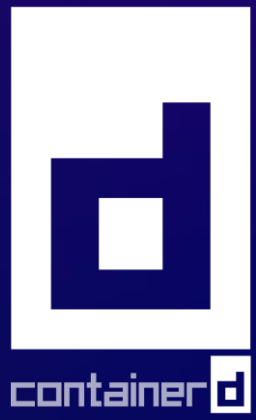
SOCI is easy to use, adopt, and onboard

No changes are required to the application code



Existing container images do not need to be rebuilt or converted

Does not require changes to CI/CD pipelines



Built as a remote snapshotter for containerd

Usage of the containerd logo approved by Jake Holt, AWS Trademark Corporate Counsel

To lazily load a container image we need to know a lot more information about the container image and what is inside

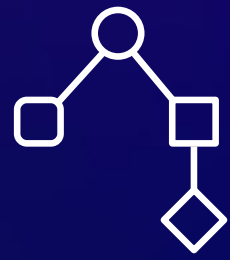
How do I know where in the container image the file is stored?

How do I download only a subset of a compressed tarball?

SOCI index – An inventory of what is inside of a container image

SOCI indexes an existing container image without modifying the image

SOCI stores the index in the container registry alongside the container image

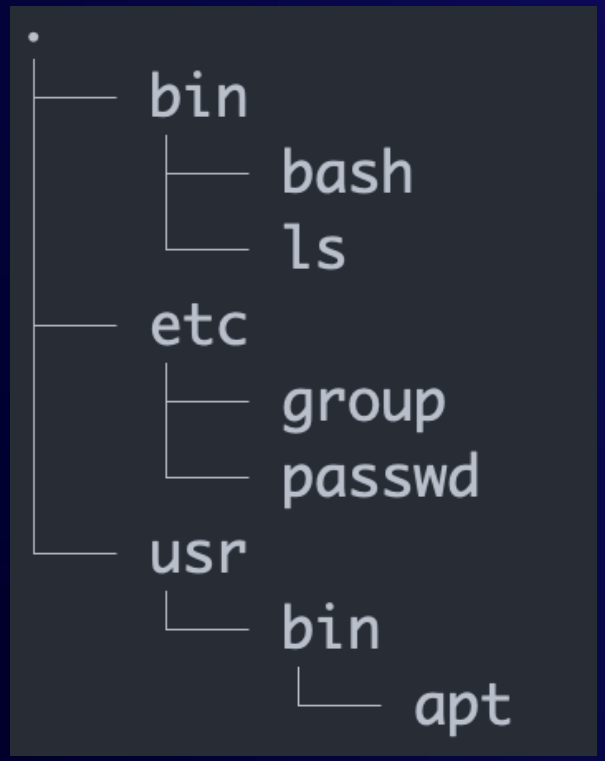


SOCI will leverage the upcoming OCI 1.1 referrers API to allow clients to discover indexes

First, we create a list of all files in a container image layer and their offsets within the tarball

File	Offset
bin/bash	1326576
bin/ls	187040
etc/passwd	7868
etc/group	3304
usr/bin/apt	151792

Second, we carve up the container image tarball into logical chunks known as spans

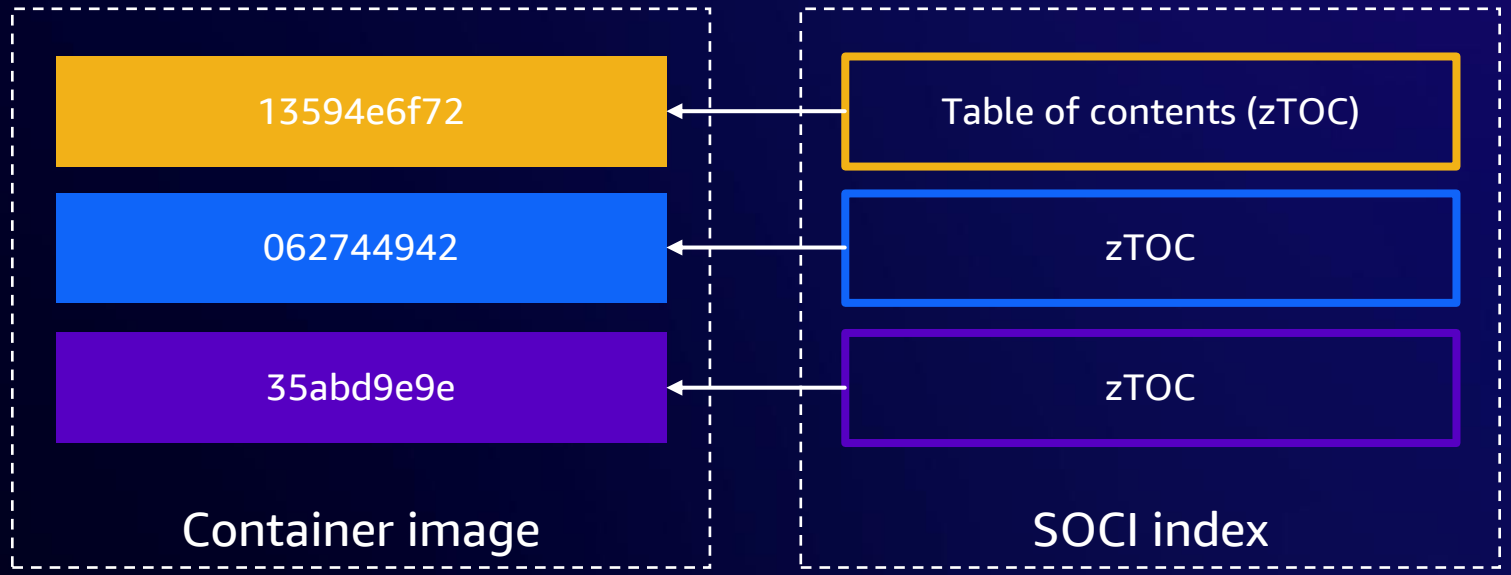


Second, we carve up the container image tarball into logical chunks known as spans

We now know which files are stored in each layer and which span contains the file

File	Offset	Start span	End span
bin/bash	1326576	0	0
bin/ls	187040	0	1
etc/passwd	7868	1	1
etc/group	3304	1	1
usr/bin/apt	151792	2	4

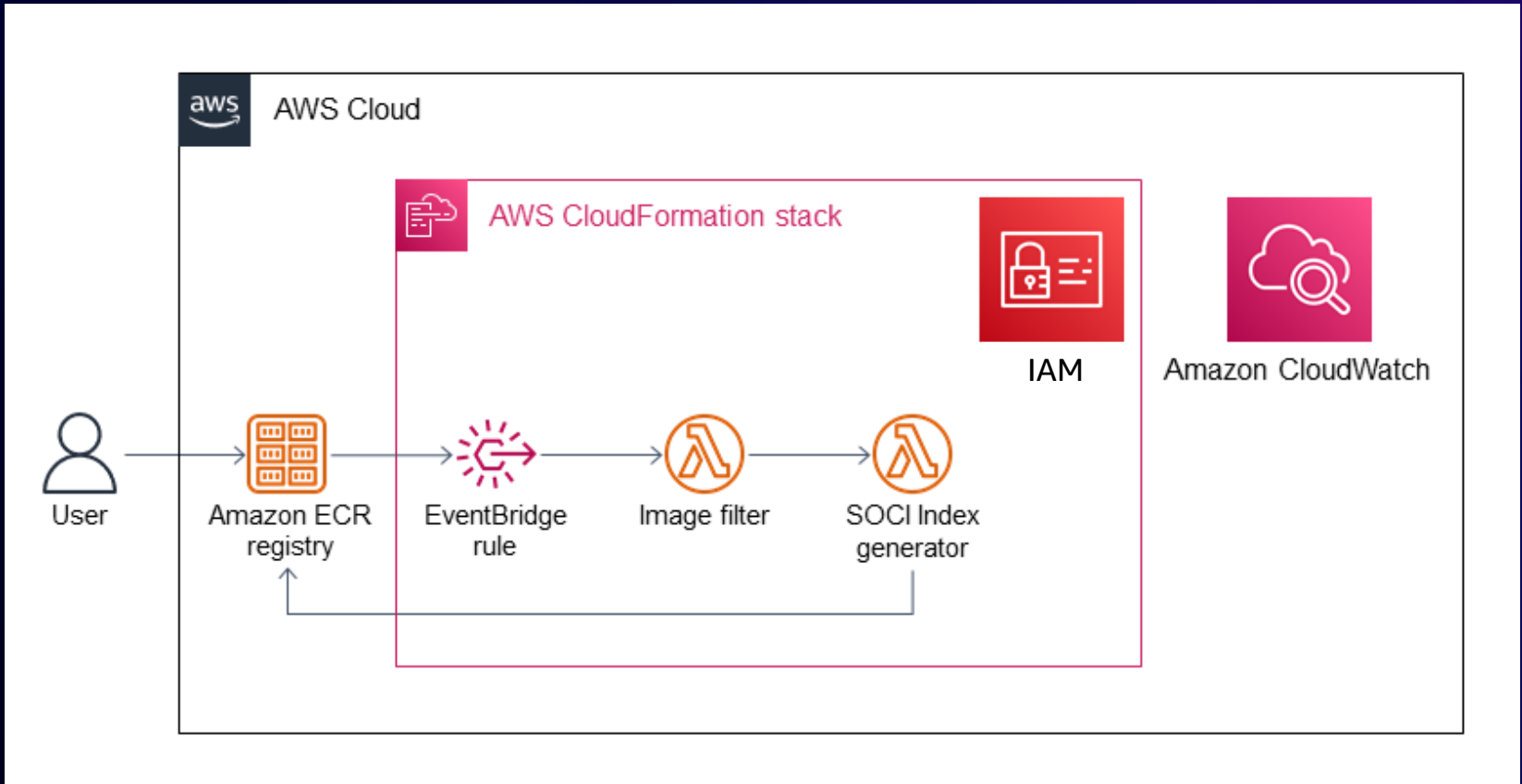
The relationship between a SOCI index and a container image



How to build SOCI indexes

A SOCI index can be created using the SOCI CLI

A SOCI index can be automated using the SOCI index Builder



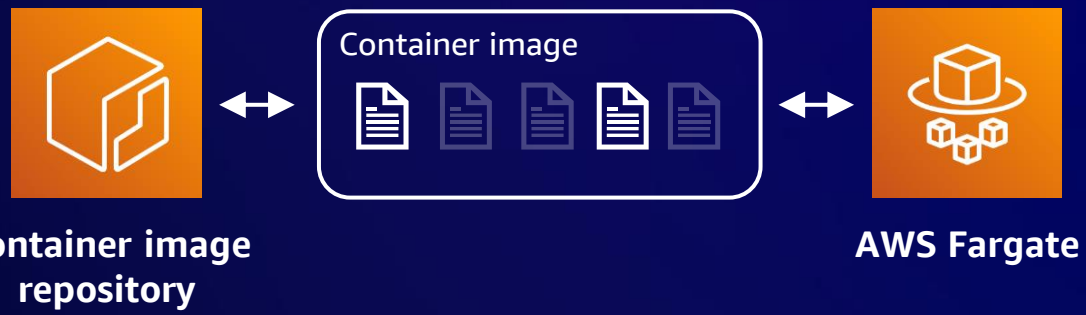
https://aws-ia.github.io/cfn-ecr-aws-soci-index-builder

AWS Fargate and SOCI

Lazy loading container images on AWS Fargate with Seekable OCI (SOCI)

Announced in July 2023

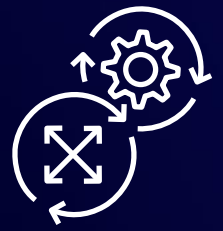
Customers running applications on Amazon ECS with AWS Fargate can now leverage Seekable OCI (SOCI)



https://aws.amazon.com/about-aws/whats-new/2023/07/aws-fargate-container-startup-seekable-oci/

SOCI snapshotter is enabled on Linux tasks launched on AWS Fargate

No additional cost for using SOCI on AWS Fargate



Available for all Linux tasks on platform version 1.4

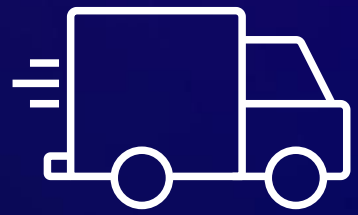
No changes required to task definitions

Easy roll-back! Just delete the SOCI index

Lazy loading has the biggest impact for workloads with large container images

Lazy loading works best for container images greater then 250MB

Does not frequently accesses filesystem metadata



Requires all of the image data very quickly after application start up

The impact of SOCI for AWS Fargate customers

AUTODESK

SOCI is easy to use and improved the startup performance of our time-sensitive workloads by 50%.

SOCI allows us to quickly scale our applications and save on costs by reducing idle compute capacity.

Boaz Brudner,
Head of Innovyze SaaS Engineering, AI and Architecture

flywire

SOCI allowed us to drastically reduce the rollout time for our application updates.

For some of our larger images of over 750MB, SOCI improved the task startup time by more than 60%.

Samuel Burgos,
Sr. Cloud Security Engineer

Virtuoso

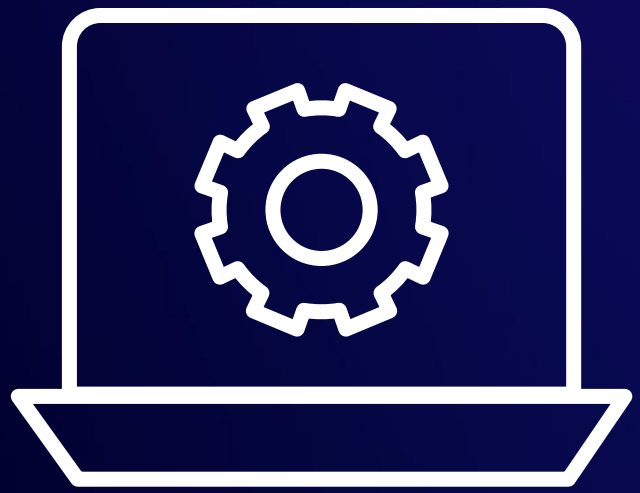
SOCI helped our ECS tasks spin-up 40% faster, allowing us to quickly scale our bursty workloads.

Setting up SOCI was easy and with the quick-start AWS Partner's solution we could leave our build and deployment pipelines untouched.

Mathew Hall,
Head of Site Reliability Engineering

https://aws.amazon.com/blogs/aws/aws-fargate-enables-faster-container-startup-using-seekable-oci/

Demonstration



Lazy Loading Container Images with Seekable OCI and AWS Fargate

Lazy loading on AWS Fargate with Seekable OCI

Learn more about Seekable OCI

AWS Fargate Enables Faster Container Startup using Seekable OCI
https://aws.amazon.com/blogs/aws/aws-fargate-enables-faster-container-startup-using-seekable-oci/

Under the hood: Lazy Loading Container Images with Seekable OCI and AWS Fargate
https://aws.amazon.com/blogs/containers/under-the-hood-lazy-loading-container-images-with-seekable-oci-and-aws-fargate/

Start Spring Boot applications faster on AWS Fargate using SOCI
https://aws.amazon.com/blogs/containers/start-spring-boot-applications-faster-on-aws-fargate-using-soci/

Join us at the Seekable OCI Workshop

CON403 - Seekable OCI (SOCI) for lazy loading container images with Amazon ECS

MGM Grand - Grand 123 - Tuesday - 2pm

Thank you!

Please complete the session survey in the mobile app