
AWS
re:Invent
NOV. 28 – DEC. 2, 2022 | LAS VEGAS, NV

SVS320

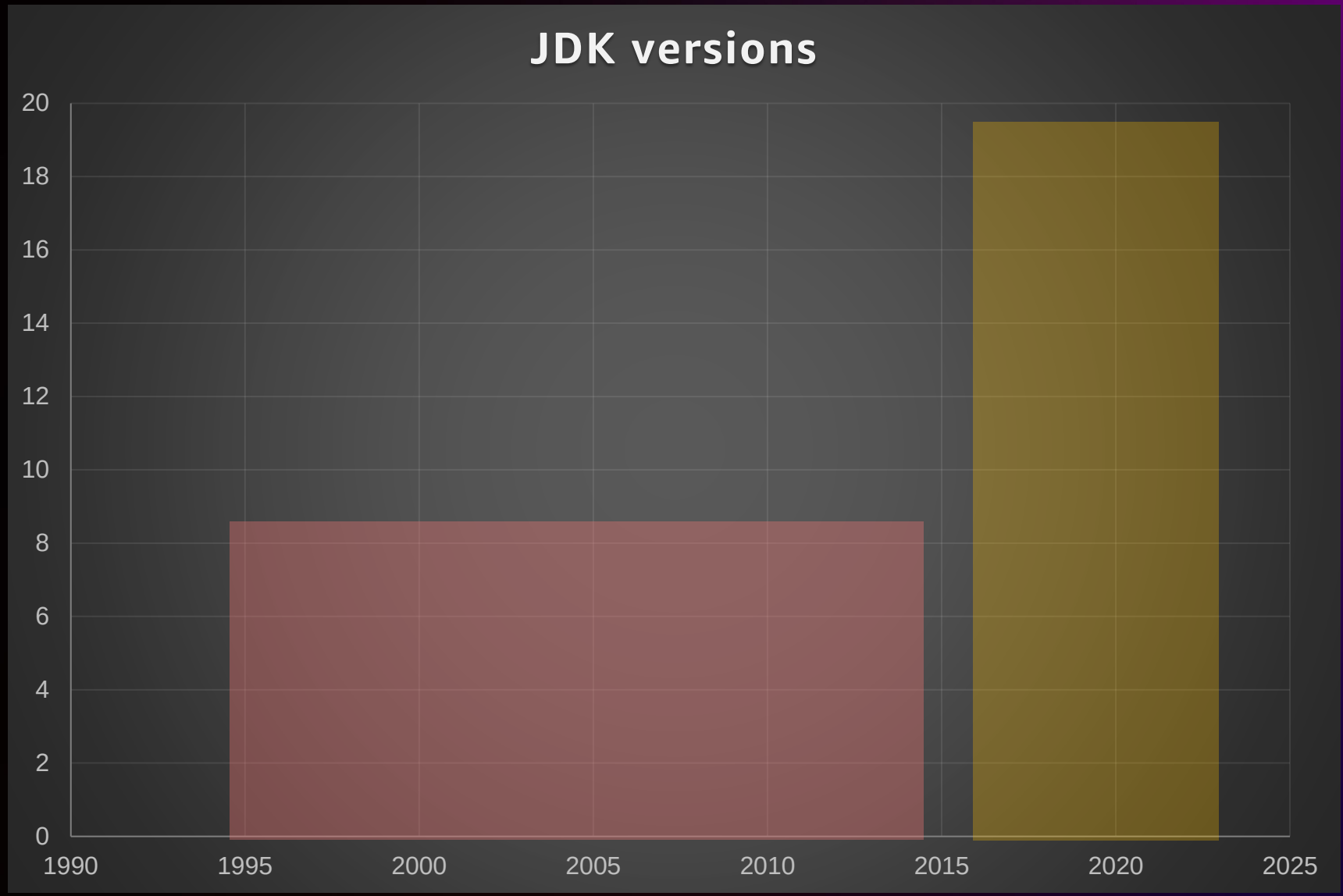
AWS Lambda SnapStart: Fast cold starts for your Java functions

Mark Sailes (he/him)

Senior Specialist SA, Serverless
Amazon Web Services

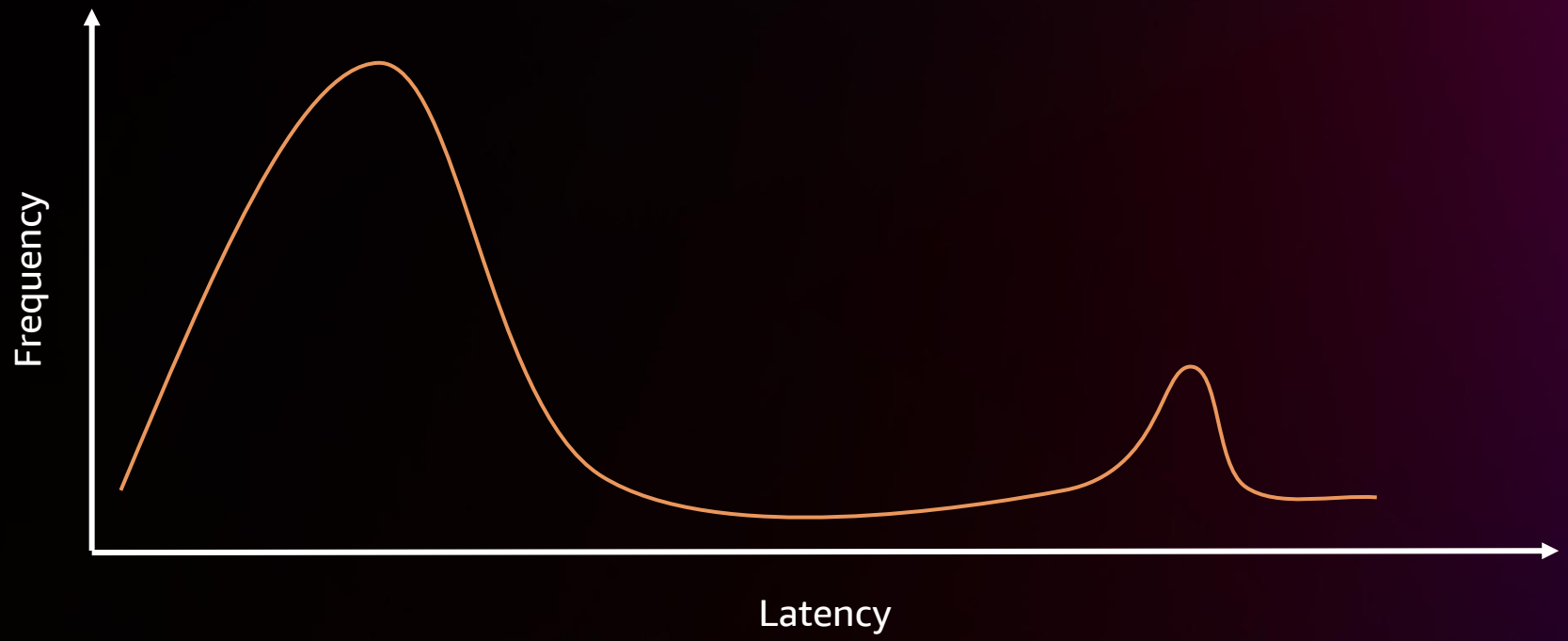
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Java release cadence



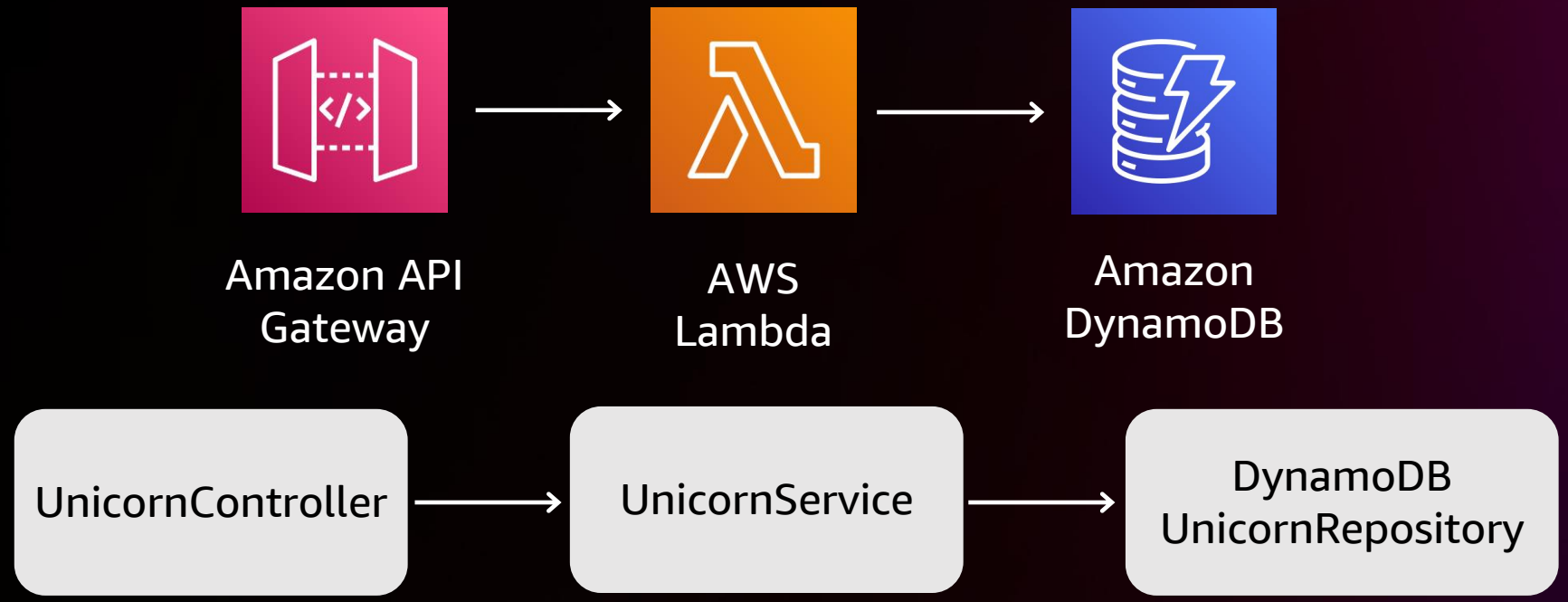
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Lambda function performance



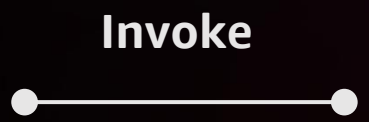
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Example Spring Boot Function



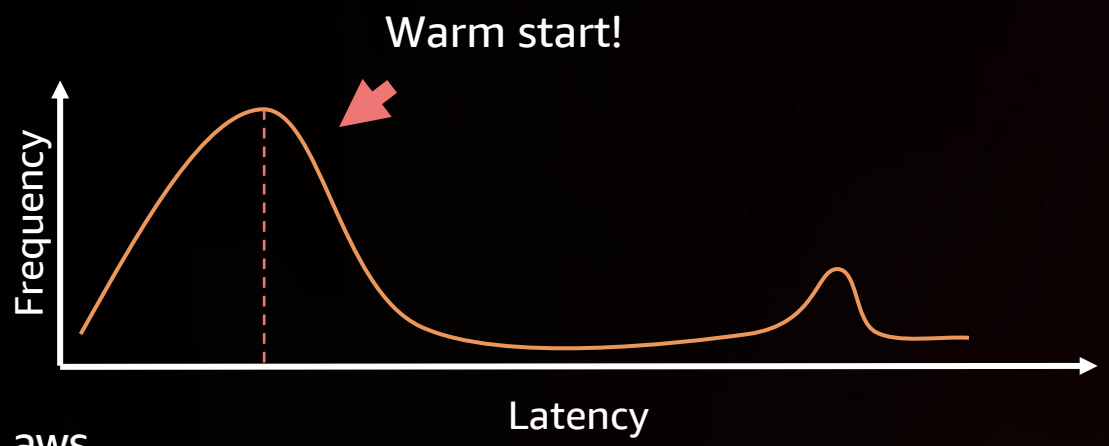
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Execution environment lifecycle



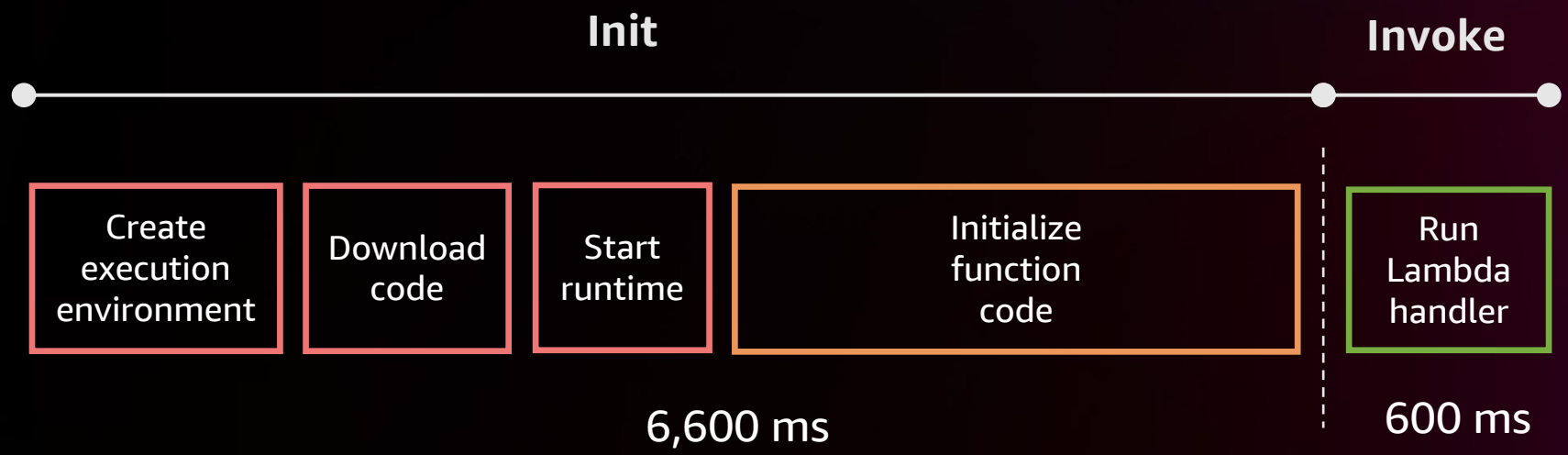
Run Lambda handler

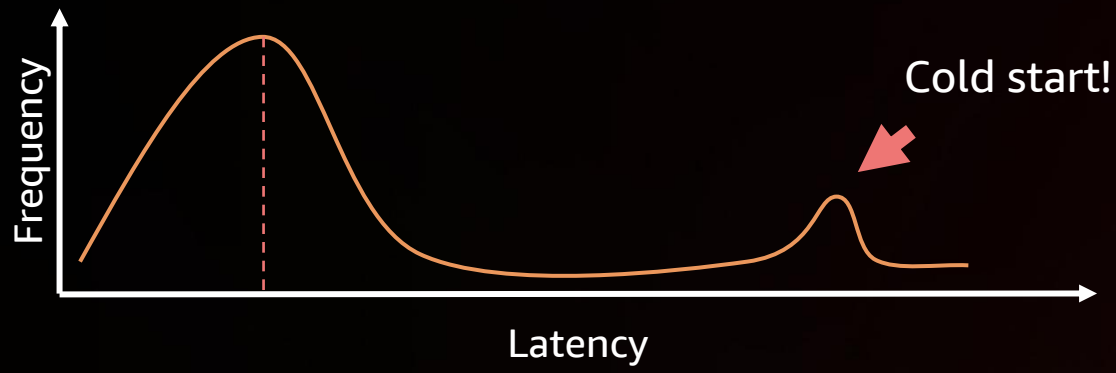
~10 ms



aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

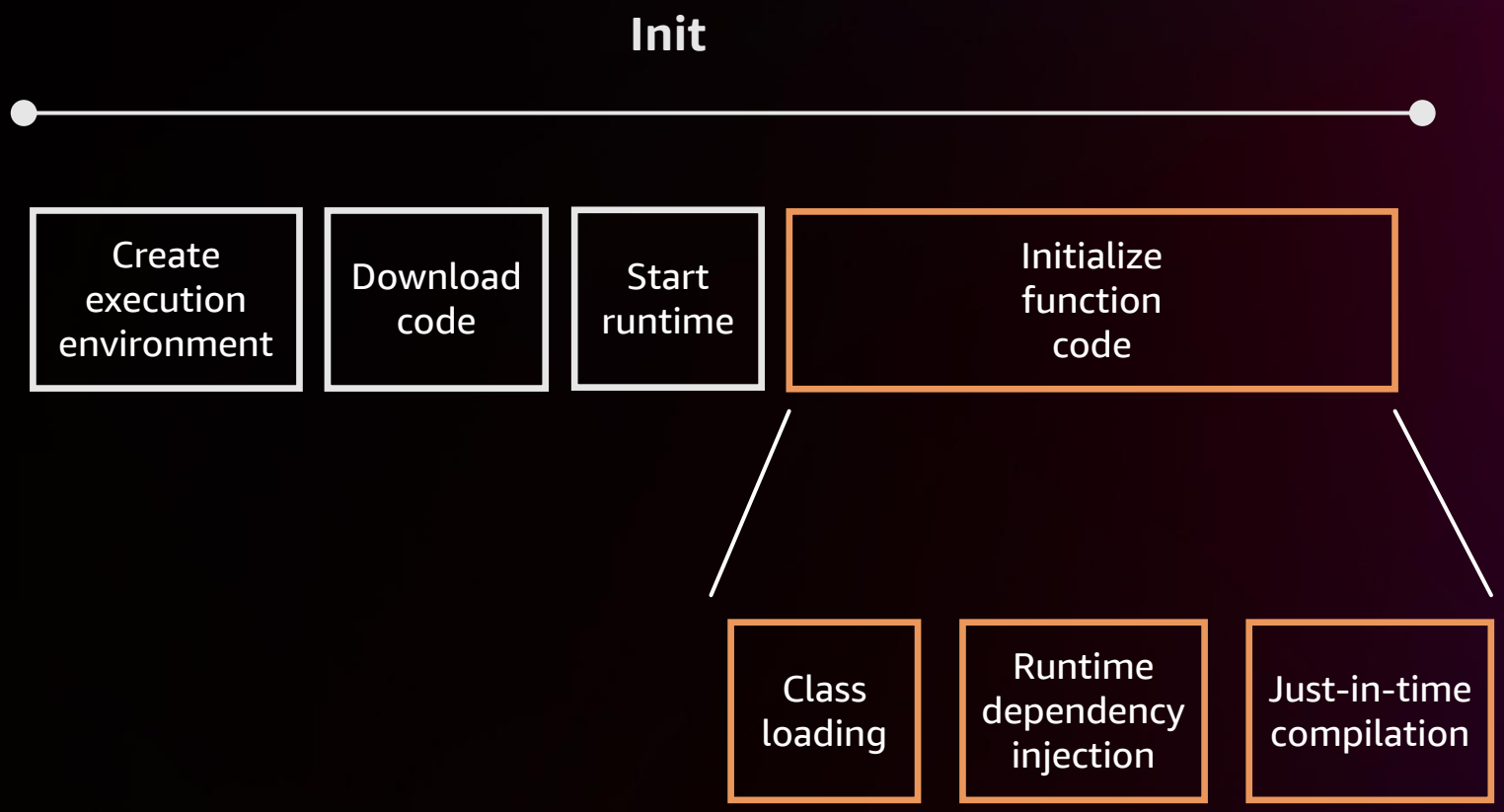
Invocation model





© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Java initialization



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

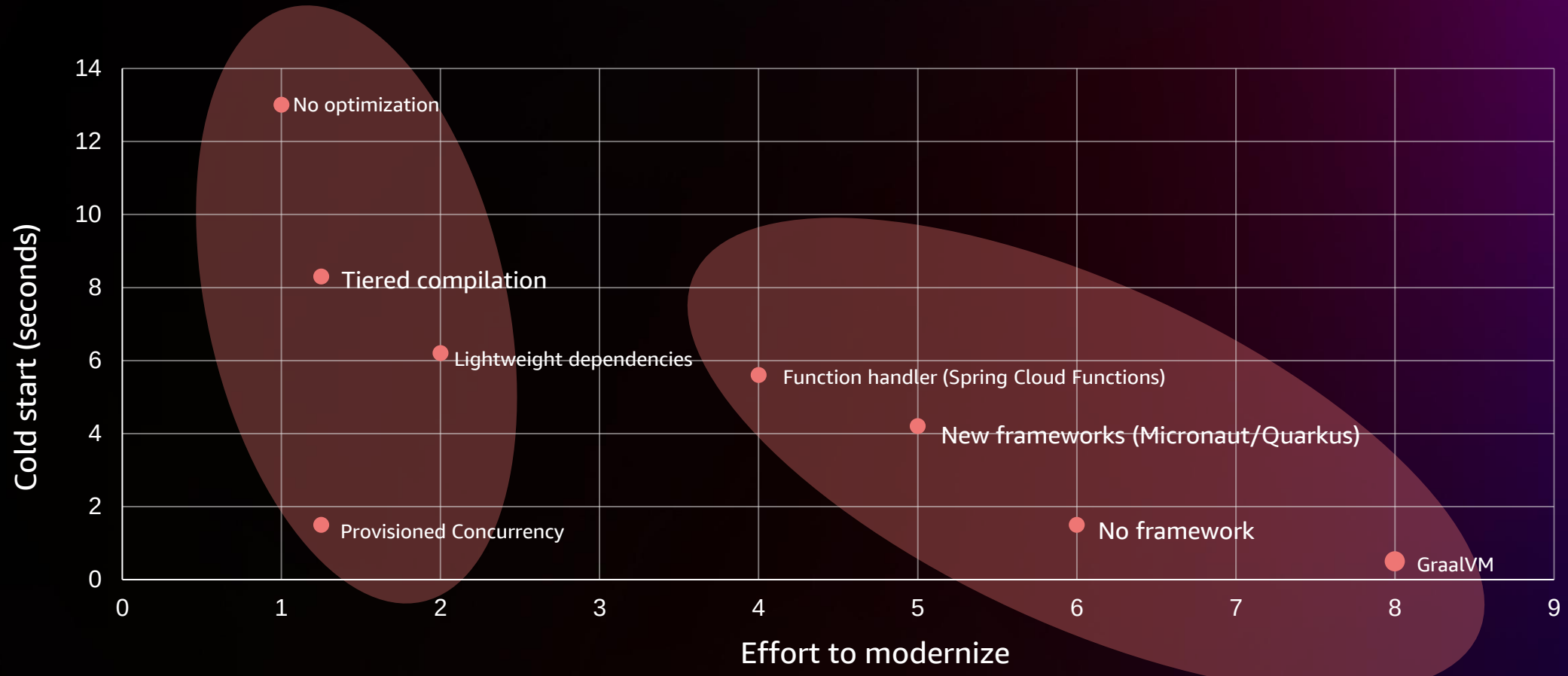
Current solutions

Class loading	Runtime dependency injection	Just-in-time compilation
Lightweight dependencies • Jackson-jr • Slf4j-simple	• Micronaut • Quarkus • Dagger • No DI	• More memory • Tiered compilation • AOT GraalVM native-image

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Cold start optimizations for a Spring Boot Migration



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.



NEW

AWS Lambda SnapStart

Up to 10x faster startup performance

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

SnapStart overview

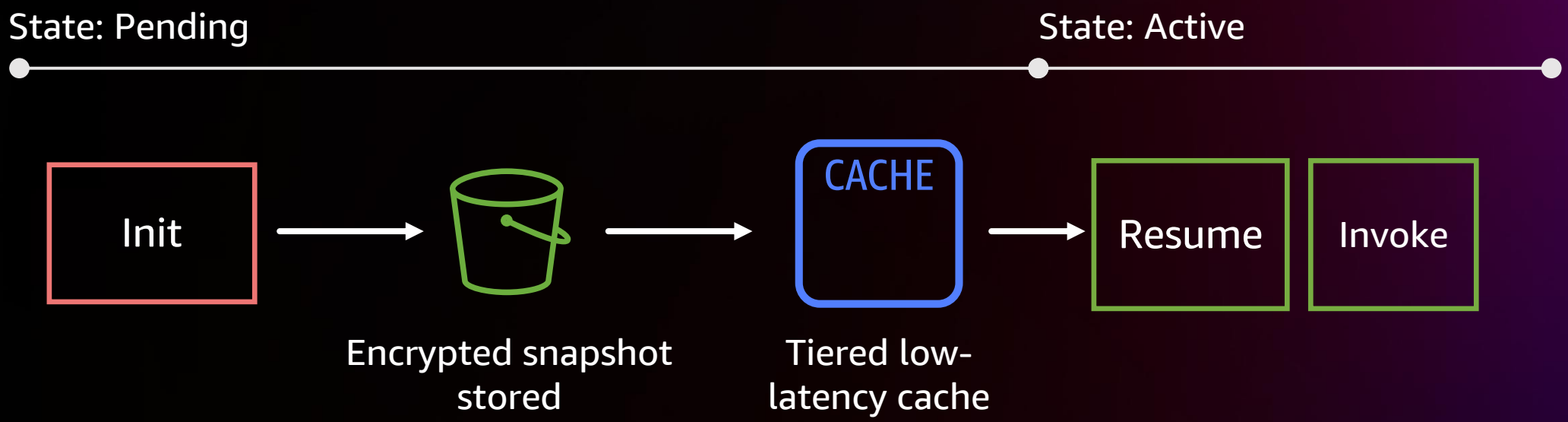
publish-version



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

SnapStart overview

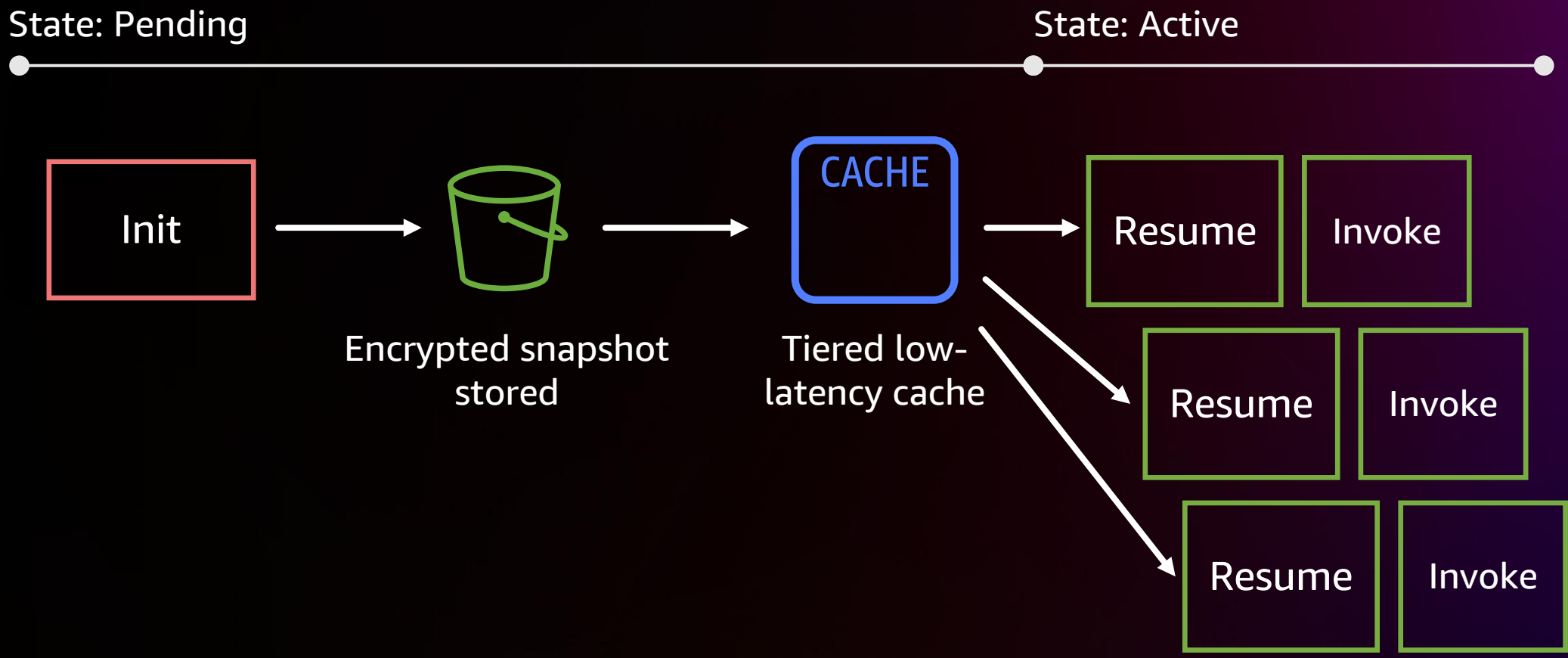
publish-version



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

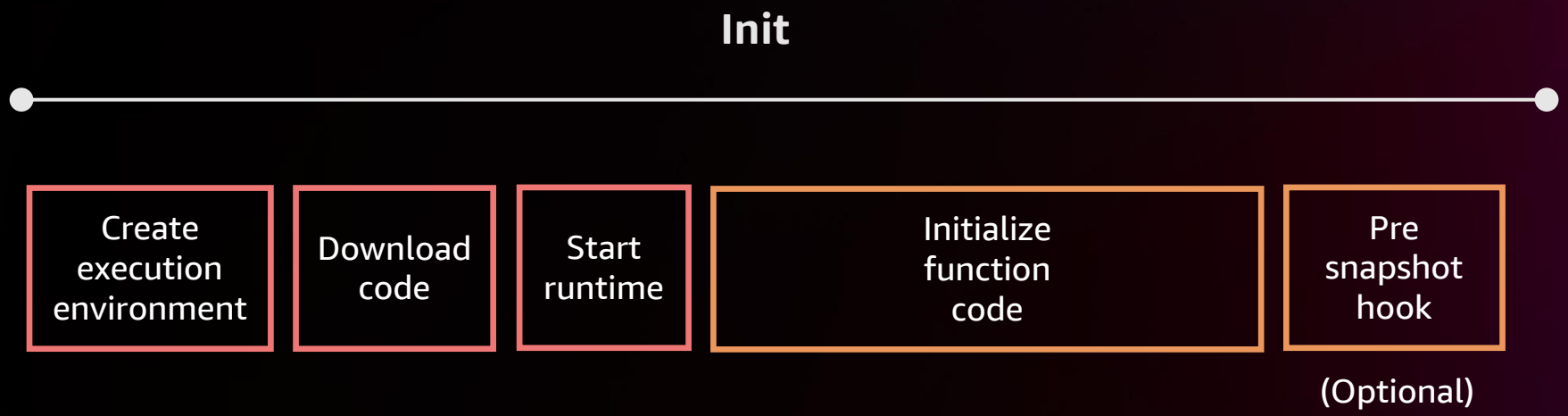
SnapStart overview

publish-version



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Invocation model



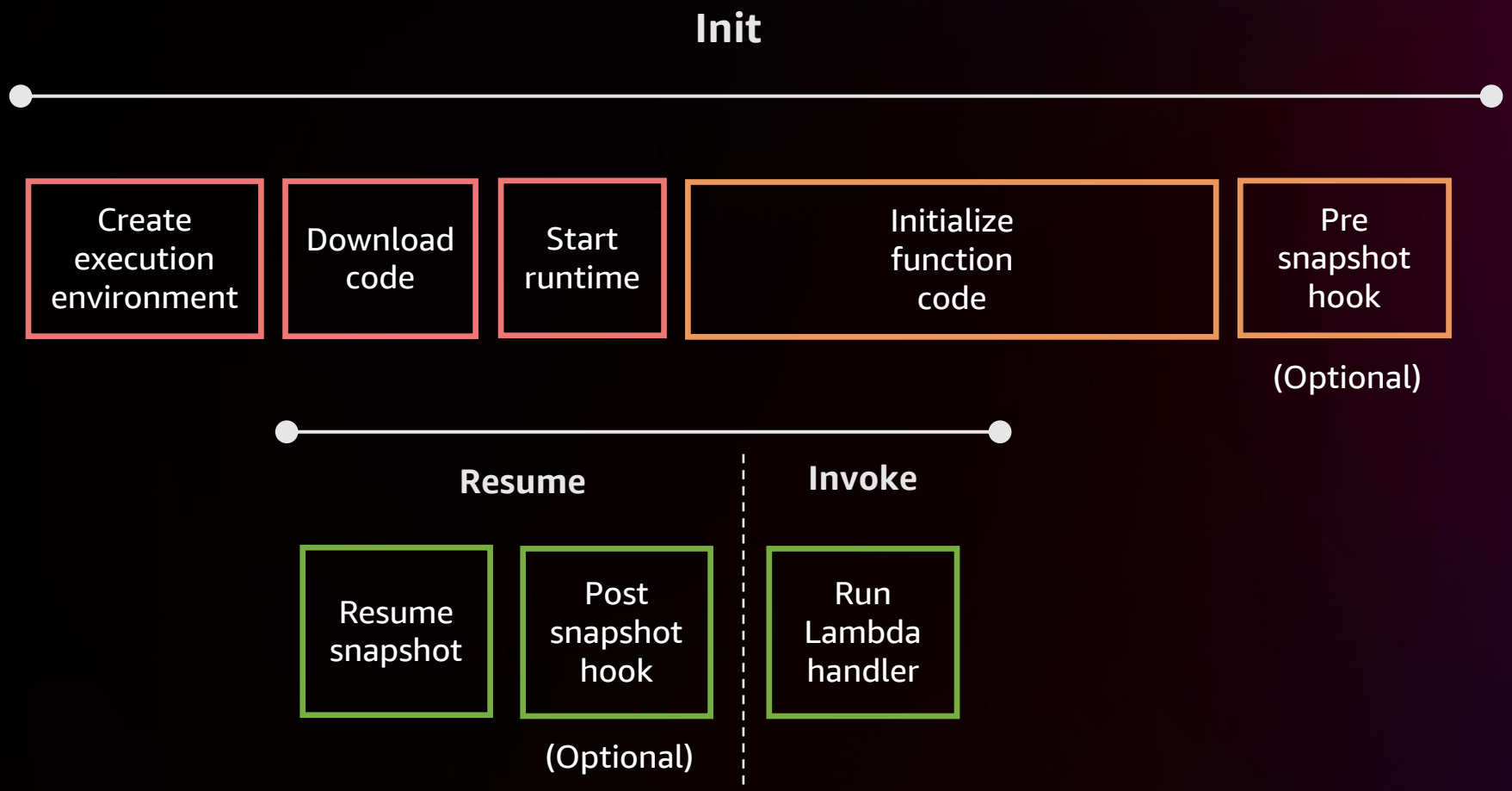
Firecracker

microVM snapshot technology

aws

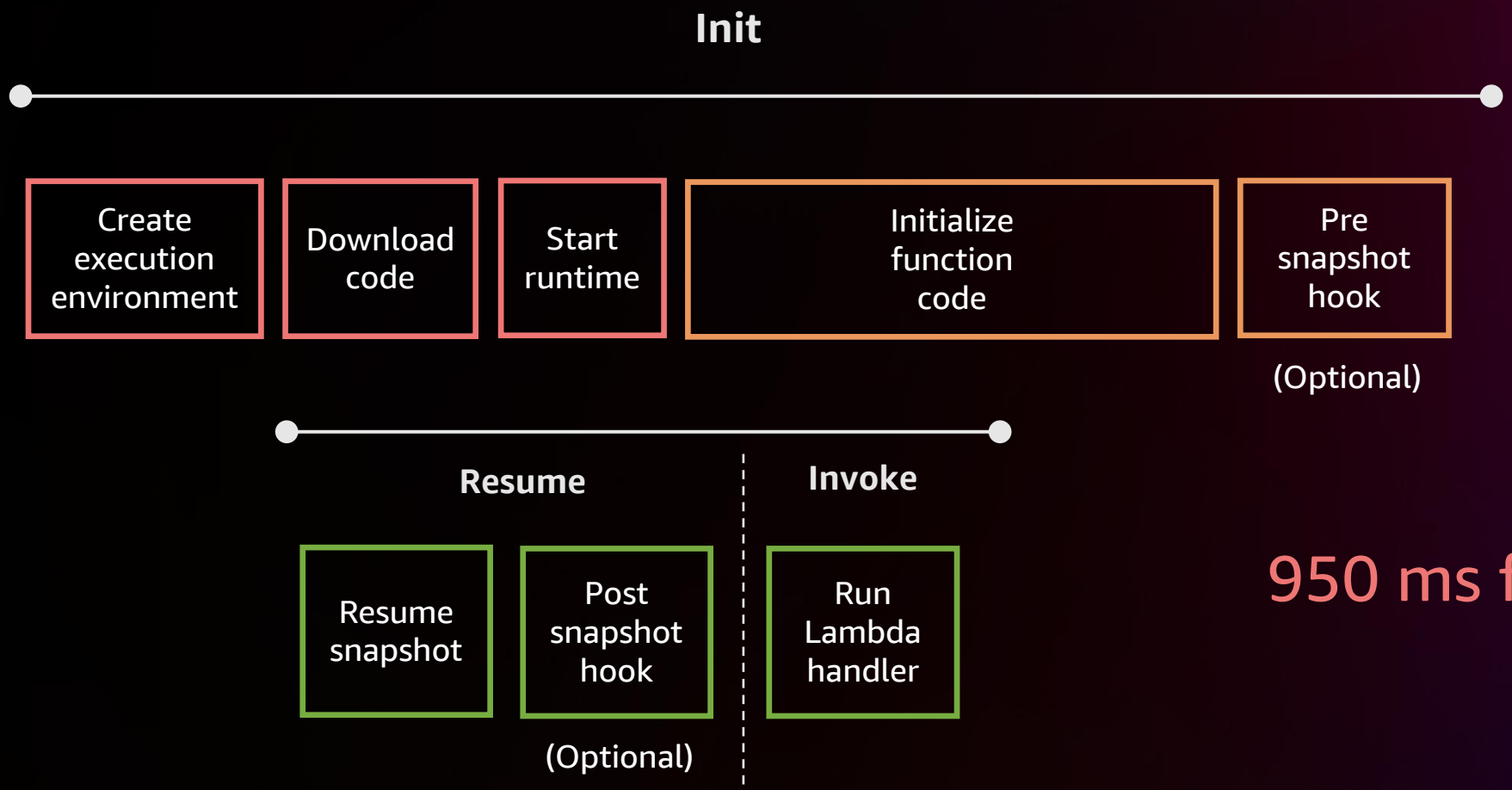
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Invocation model



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Invocation model



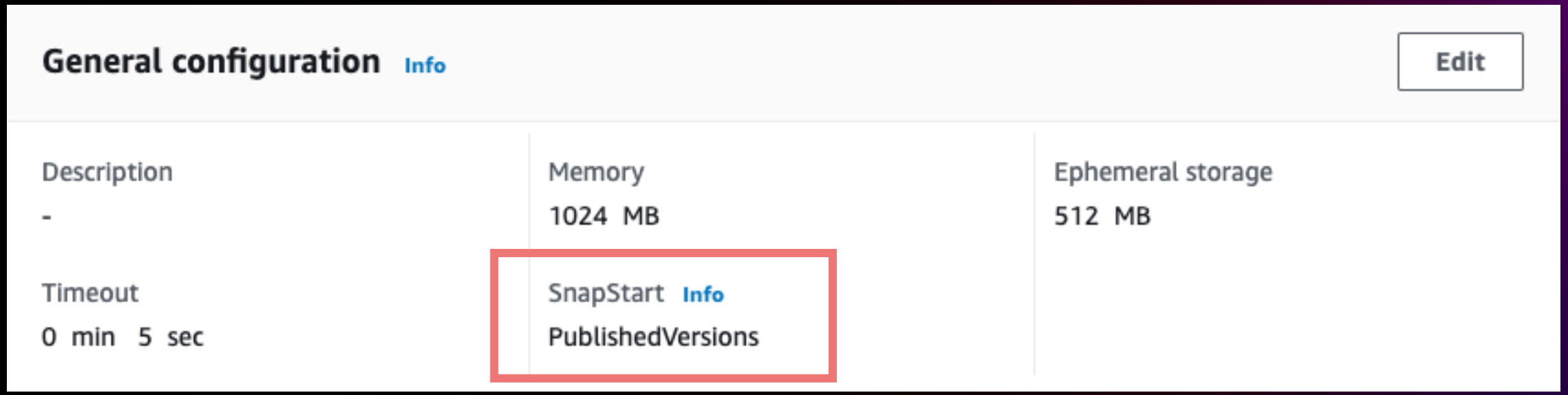
950 ms first invoke

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Configuring SnapStart

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Enabling SnapStart



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM

```yaml
Transform: AWS::Serverless-2016-10-31
Description: Lambda Function with SnapStart and versions.

Resources:
  HelloWorldFunction:
    Type: AWS::Serverless::Function
    Properties:
      CodeUri: <code-location>
      Handler: <handler>
      Runtime: java11
      AutoPublishAlias: live
      SnapStart:
        ApplyOn: PublishedVersions
```

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS CDK

```
CfnFunction.Builder.create(this, "HelloWorldFunction")
        .code(CodeProperty.builder()
            .s3Bucket("BUCKET_NAME")
            .s3Key("OBJECT_KEY")
            .build())
        .handler("HANDLER")
        .runtime(Runtime.JAVA_11)
        .snapStart(SnapStartProperty.builder()
            .applyOn("PublishedVersions")
            .build())
        .build();
```

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

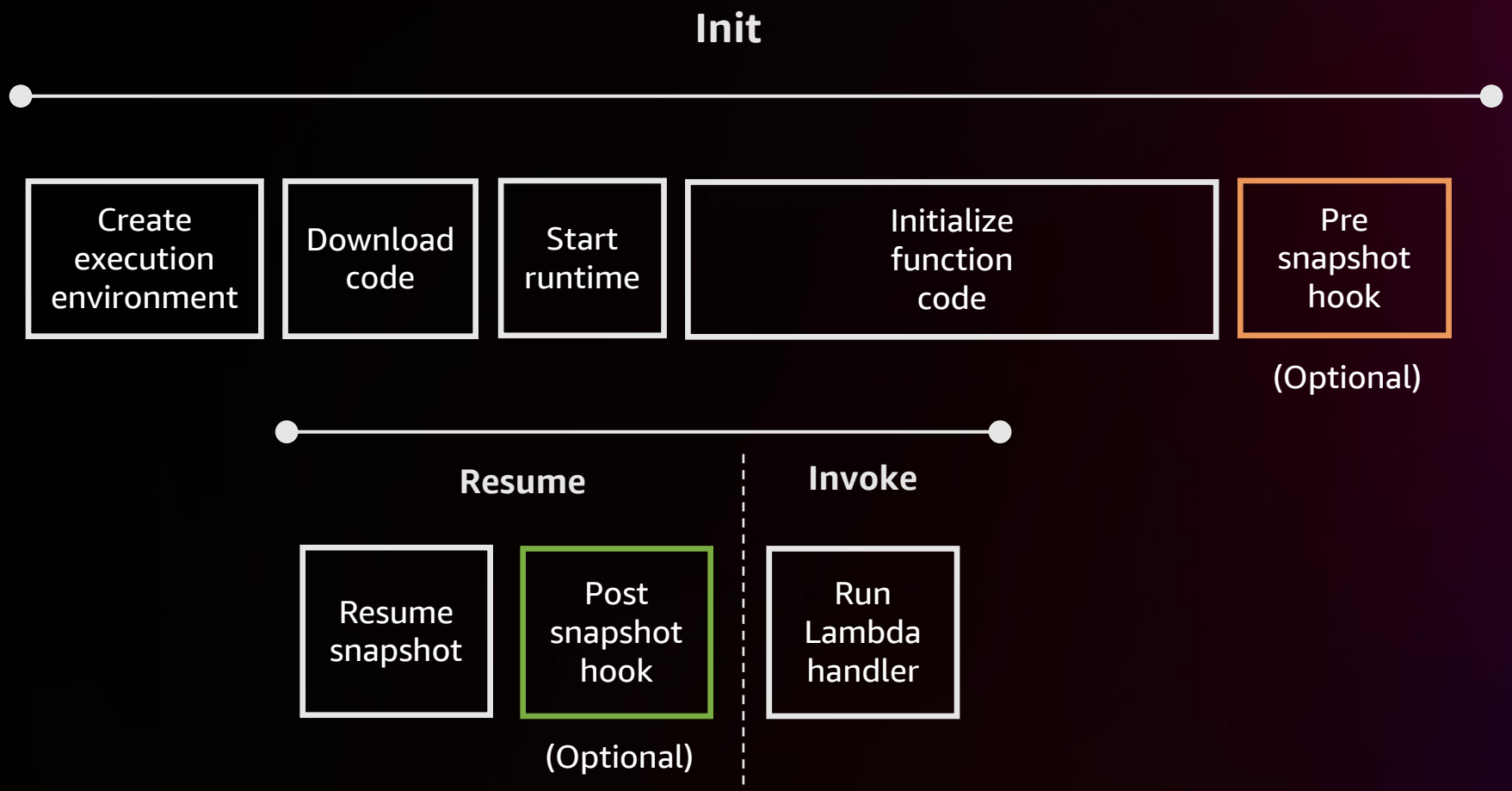
Create a version



arn:aws:lambda:eu-west-1:123456789012:function:unicorn-store-spring:prod	Alias
arn:aws:lambda:eu-west-1:123456789012:function:unicorn-store-spring:1	Version
arn:aws:lambda:eu-west-1:123456789012:function:unicorn-store-spring	$LATEST

aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Hooks detail



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Example

```
import org.crac.Core;
import org.crac.Resource;

public class HelloHandler implements RequestHandler<String, String>, Resource {

    public HelloHandler() {
        Core.getGlobalContext().register(this);
    }

    public String handleRequest(String name, Context context) {
        return "Hello " + name;
    }

    public void beforeCheckpoint() {
        System.out.println("Before Snapshot");
    }

    public void afterRestore() {
        System.out.println("After Resume");
    }
}
```

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

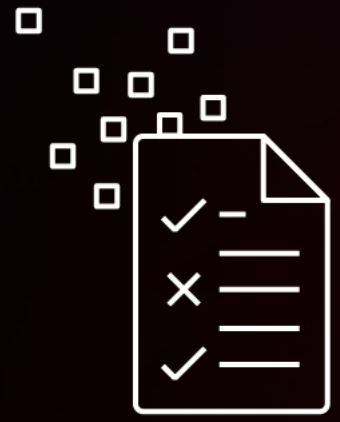
Considerations

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Considerations



Network connections



Ephemeral data



Uniqueness

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Network connections



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Network connections

Resume



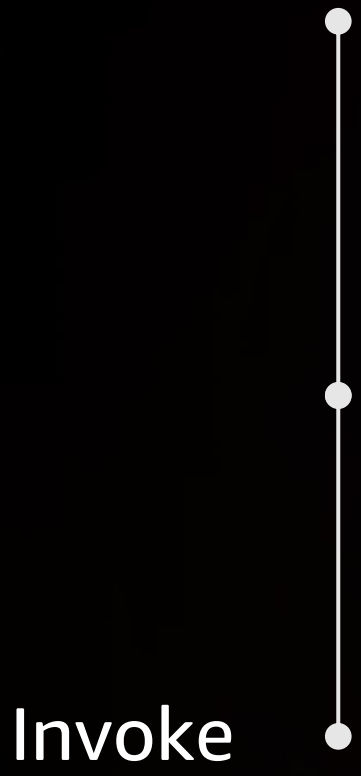
AWS
Lambda



Amazon Relational
Database Service
(Amazon RDS)

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Network connections





AWS
Lambda





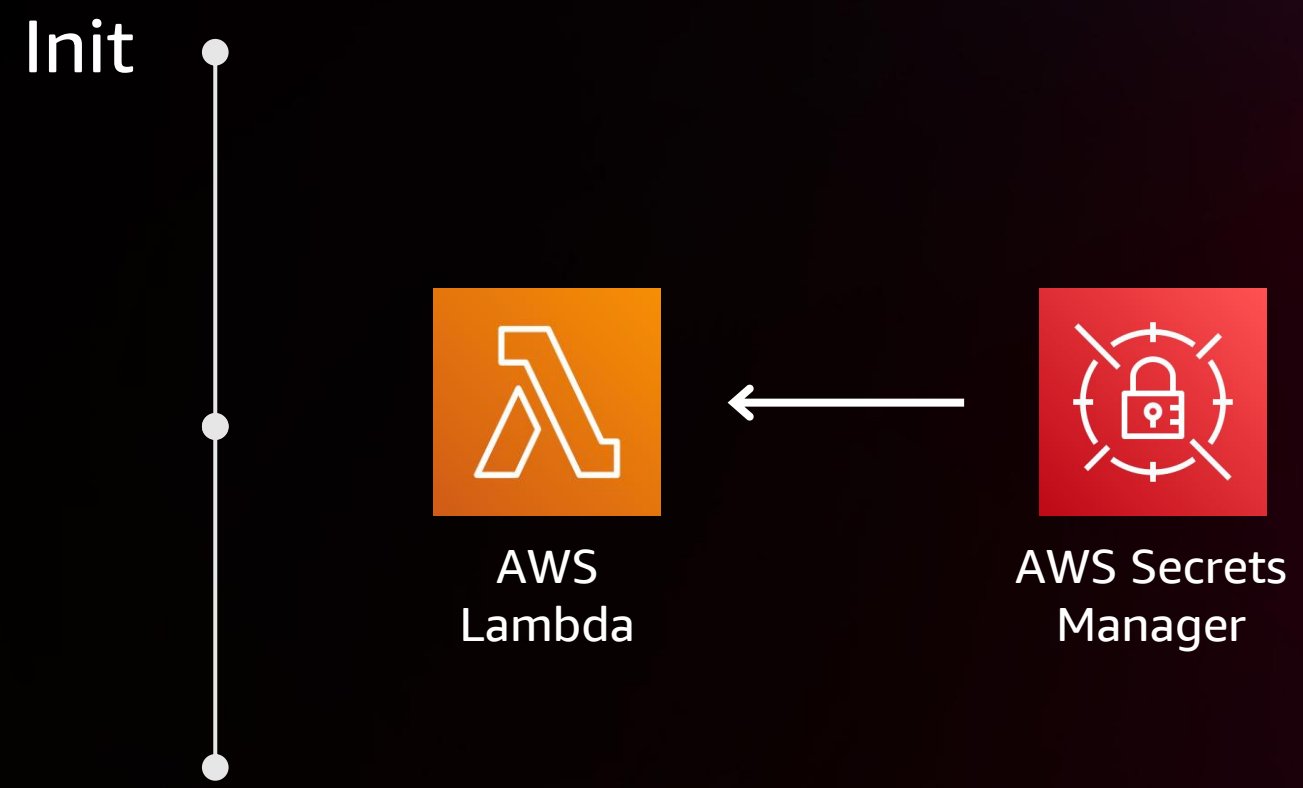
Amazon Relational
Database Service
(Amazon RDS)

```
Connection Failed, Retrying
```

aws

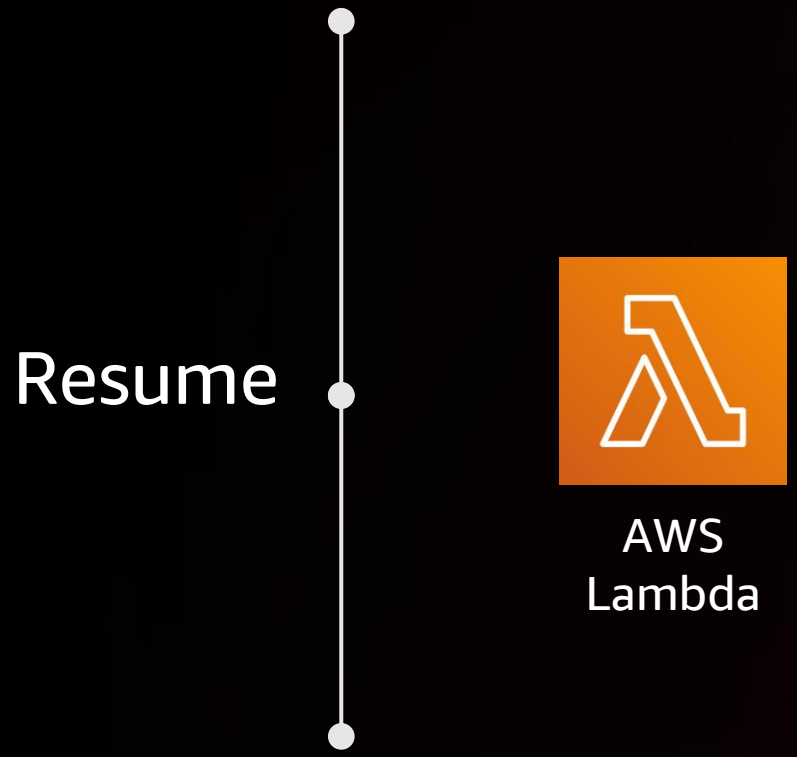
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Ephemeral data



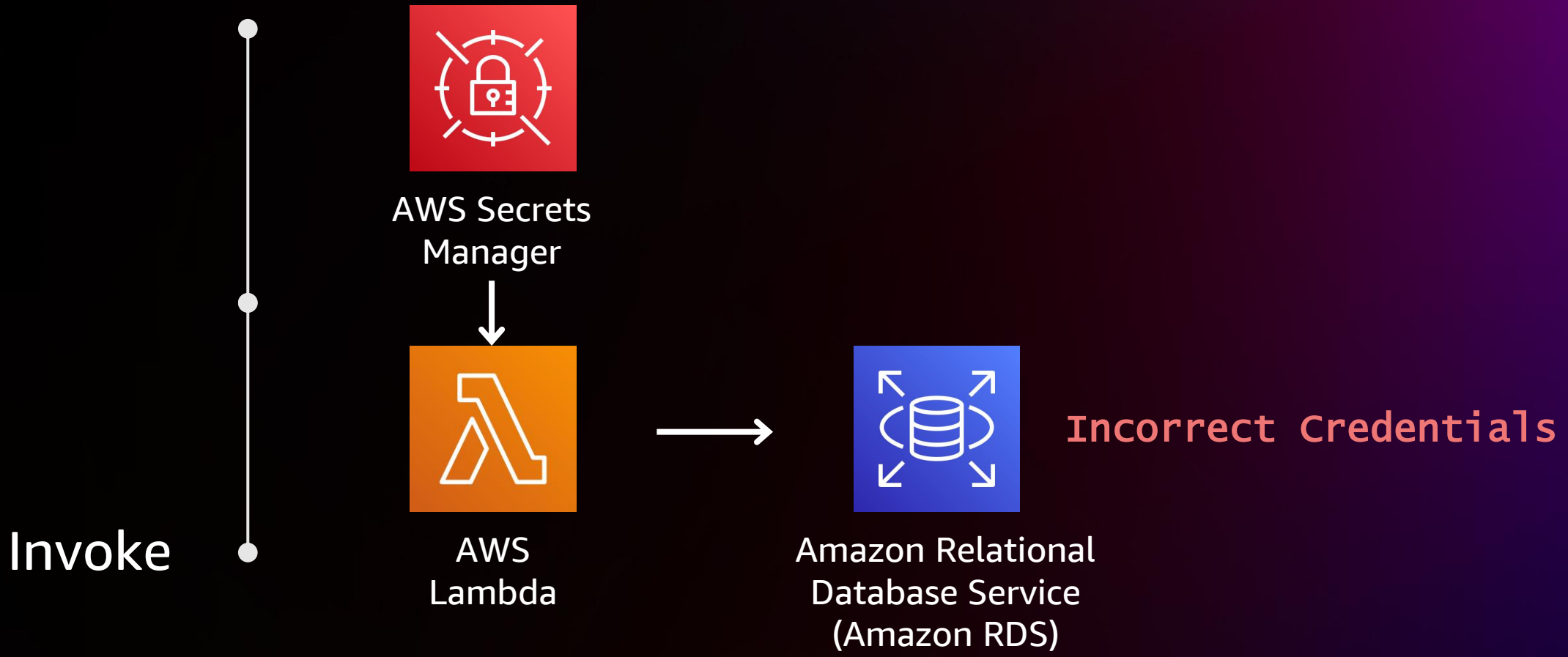
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Ephemeral data



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Ephemeral data



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Uniqueness

AWS tested

Updated

- Amazon Linux 2 and OpenSSL (1.0.2) resilient to snapshot operations
  - OpenSSL
  - /dev/random
  - /dev/urandom

Verified

- Java 11 runtime validated
  - java.security.SecureRandom

Please verify uniqueness if you're using other methods

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Scanning tool + Runtime hooks

Scans for known anti-patterns

- java.util.Random
  - Unsuitable for cryptographic work

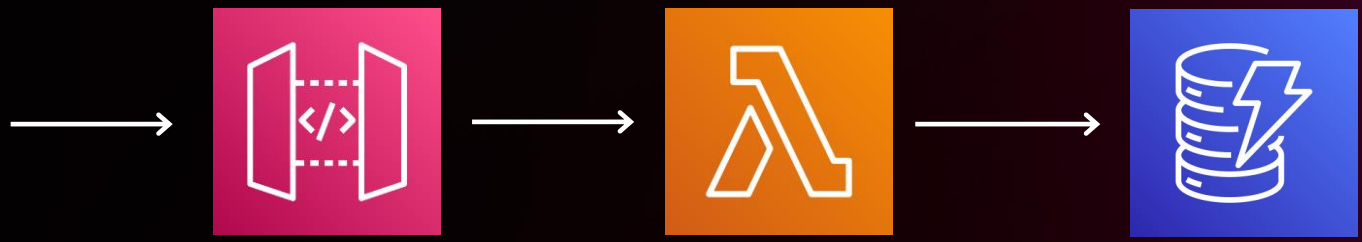


https://github.com/aws/aws-lambda-snapstart-java-rules

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Test

100 requests/second
10 minutes



Amazon API Gateway

AWS Lambda

Amazon DynamoDB

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Results

	P50	P99.9
Without SnapStart	8 ms	5,114 ms

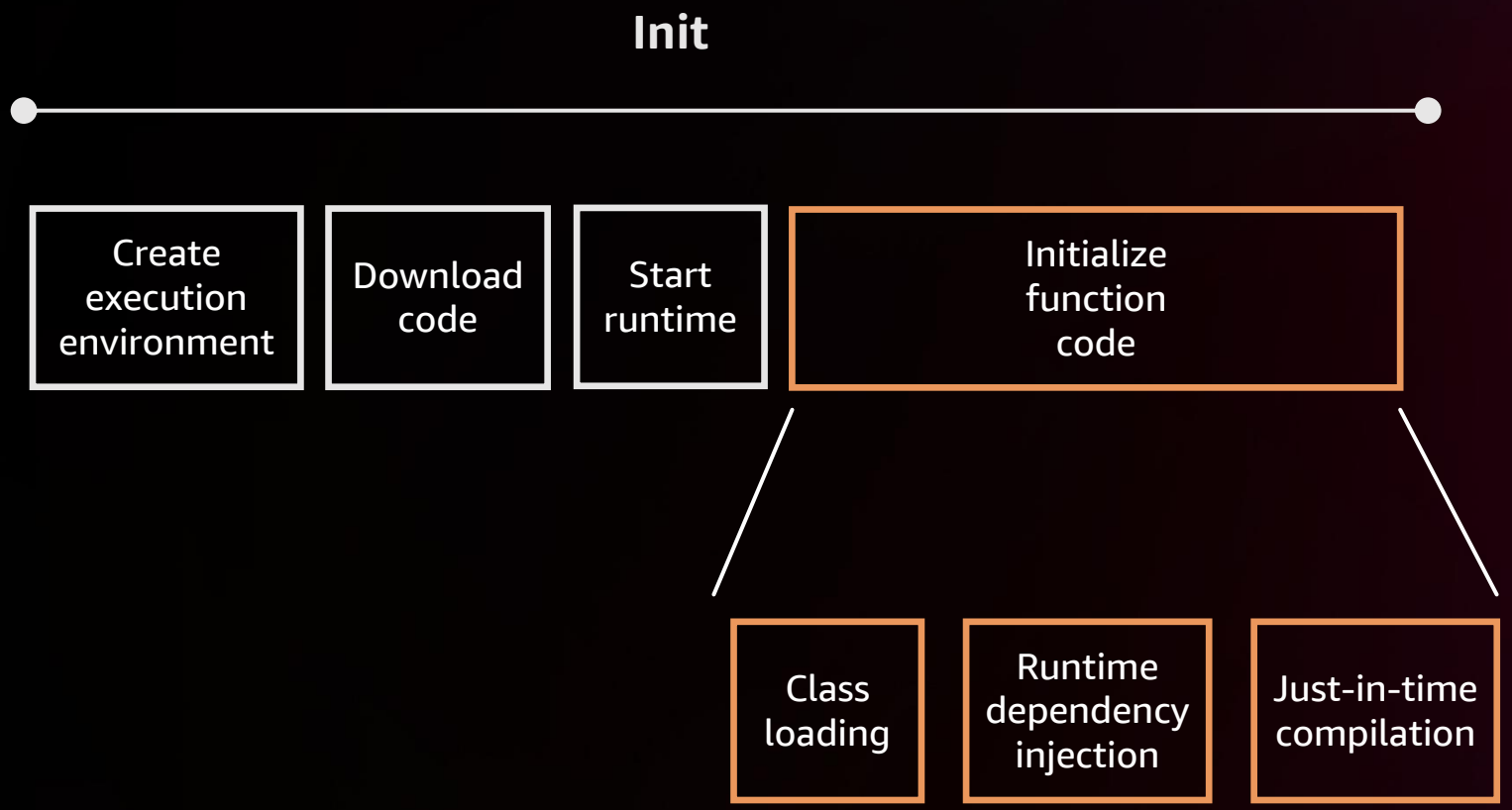
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Results

	P50	P99.9
Without SnapStart	8 ms	5,114 ms
With SnapStart	8 ms	536 ms

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Hooks and best practices



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Priming

Pre snapshot hook

```java
import org.crac.Context;
import org.crac.Core;
import org.crac.Resource;

@Configuration
public class UnicornPrimingResource implements Resource {

    private final UnicornController unicornController;

    public UnicornPrimingResource(UnicornController unicornController) {
        this.unicornController = unicornController;
        Core.getGlobalContext().register(this);
    }

    @Override
    public void beforeCheckpoint(Context<? extends Resource> context) throws Exception {
        unicornController.getUnicorn("123");
    }

    @Override
    public void afterRestore(Context<? extends Resource> context) throws Exception {
        System.out.println("afterRestore hook");
    }
}
```

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Results

	P50	P99.9
Without SnapStart	8 ms	5,114 ms
With SnapStart	8 ms	536 ms
With SnapStart + Priming	8 ms	488 ms

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

“SnapStart requires little to no code changes and delivers significant cold start improvements in numerous Lambda use cases . . . this unlocks the value of serverless on workloads that were previously not suitable for Lambda.”

Marty Andolino

Capital One Retail Bank Technology, VP of Engineering/Divisional Chief Architect

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

“SnapStart has improved the cold start performance for some of our key services by 95% percent.

Over 50% of our AWS Lambda functions in production use the Java runtime, serving as many 6.5M invocations a day.”

Ivonne Roberts

Software Architect, Bill.com

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Lambda Powertools for Java



Tenets
AWS Lambda only
Eases the adoption of best practices
Keep it lean
We strive for backward compatibility
We work backwards from the community
Progressive

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Lambda Powertools for Java

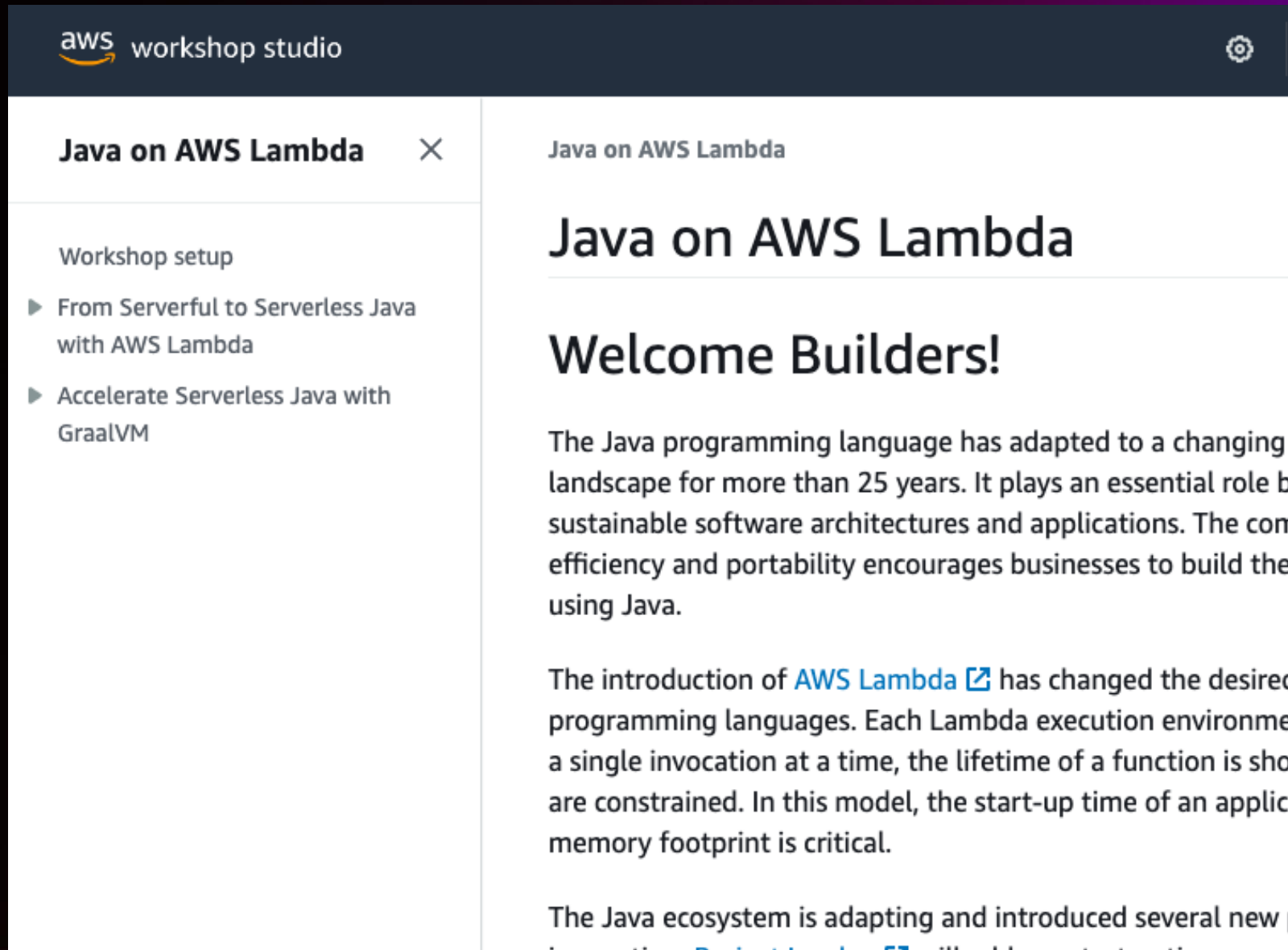
Our customers grew

The community too

- Batch partial failure
- Config management
- Observability
- Handling secrets
- Input/output validation
- Idempotency
- Caching
- Data extraction

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Java on Lambda workshop



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

If you like this . . .

SVS403: Effectively using Java on serverless

Date:
Thursday, December 1

Time:
3:30 PM – 4:30 PM

Location:
Convention Promenade, Palmer 2, Wynn

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Thank you!

Mark Sailes
@MarkSailes3



Please complete the session survey in the **mobile app**

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.