Concepts

Cold Start

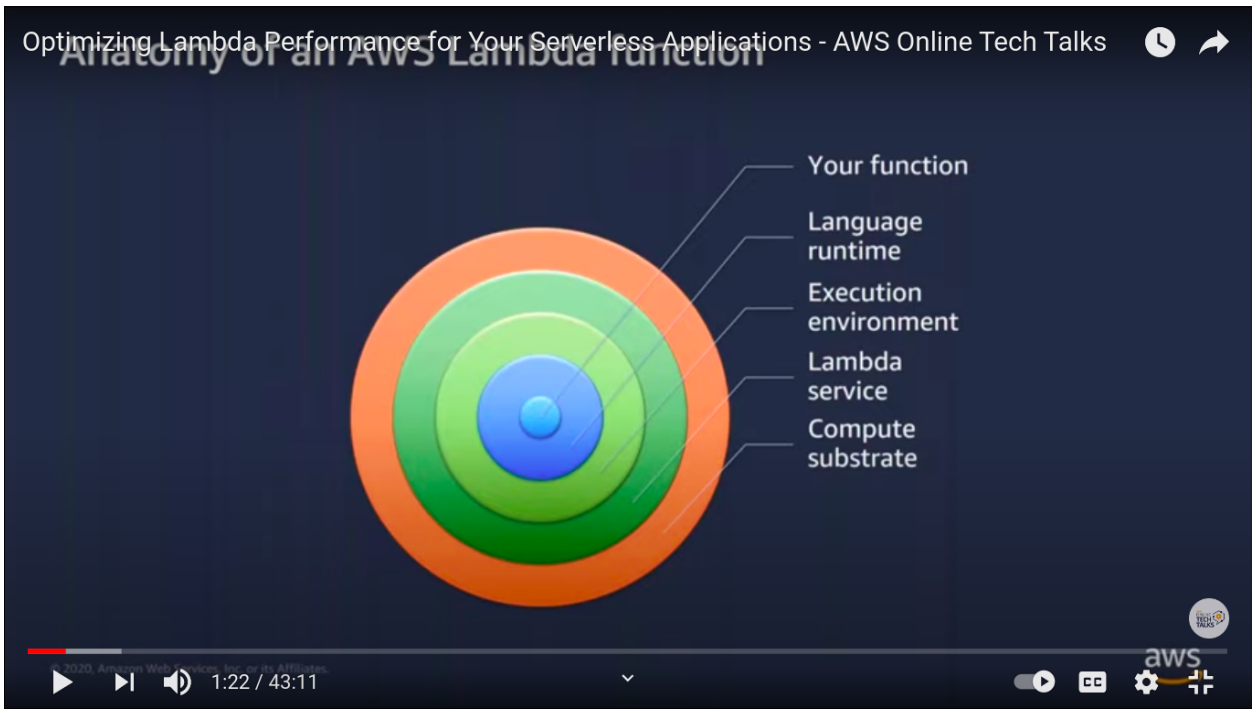
Memory and Profiling

Architecture of Lambda /Functions

Best Practices

How Lambda works

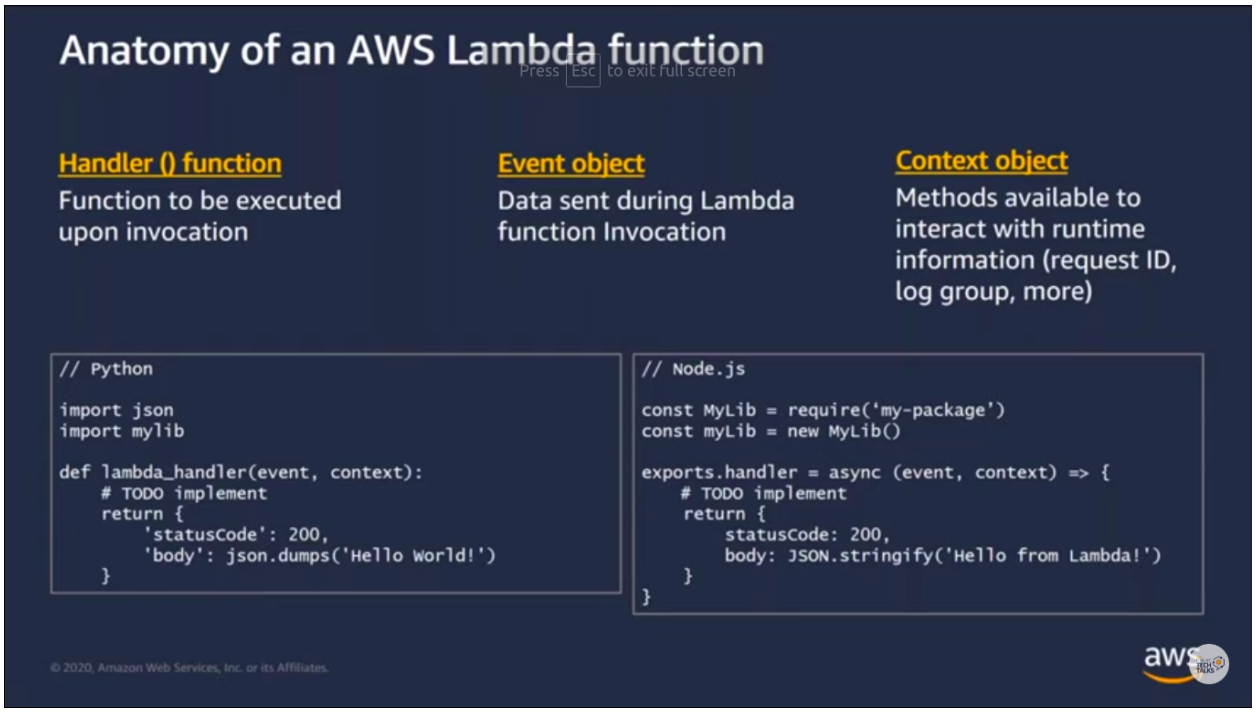
Pic Source : You Tube



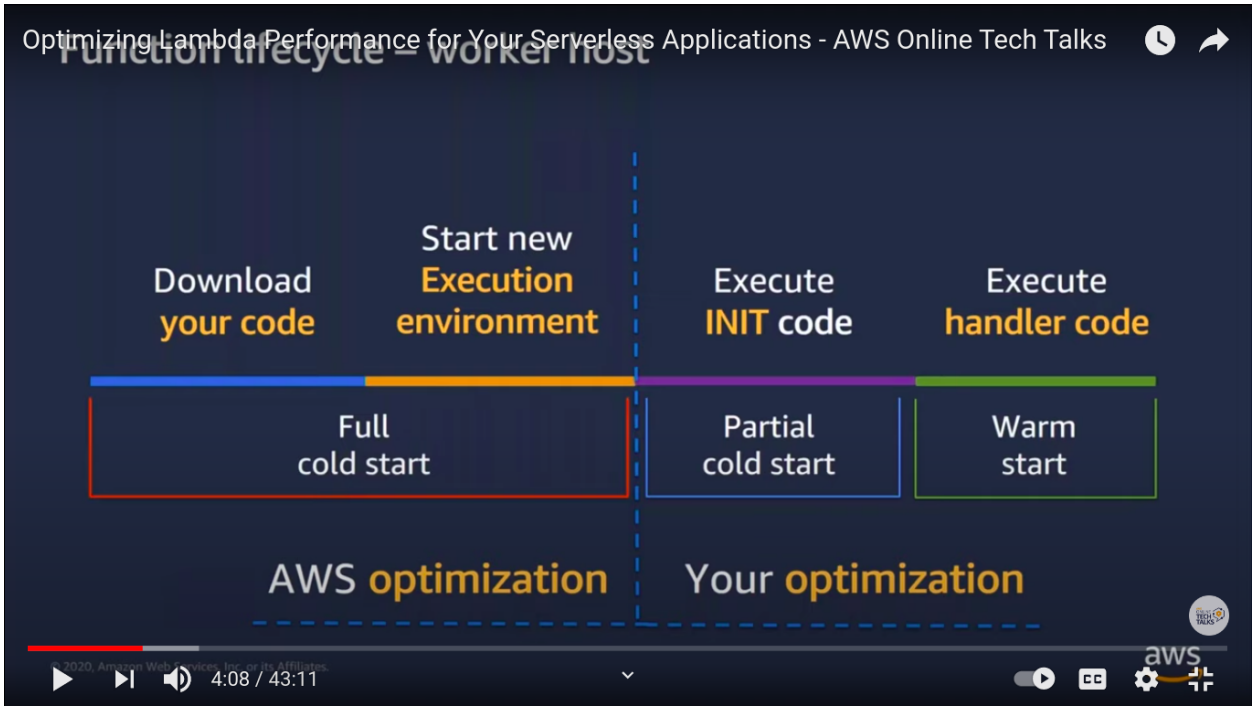
Out of these 5 layers following layers have direct impact on execution

Lambda Service
Execution Environment
Your Function Code

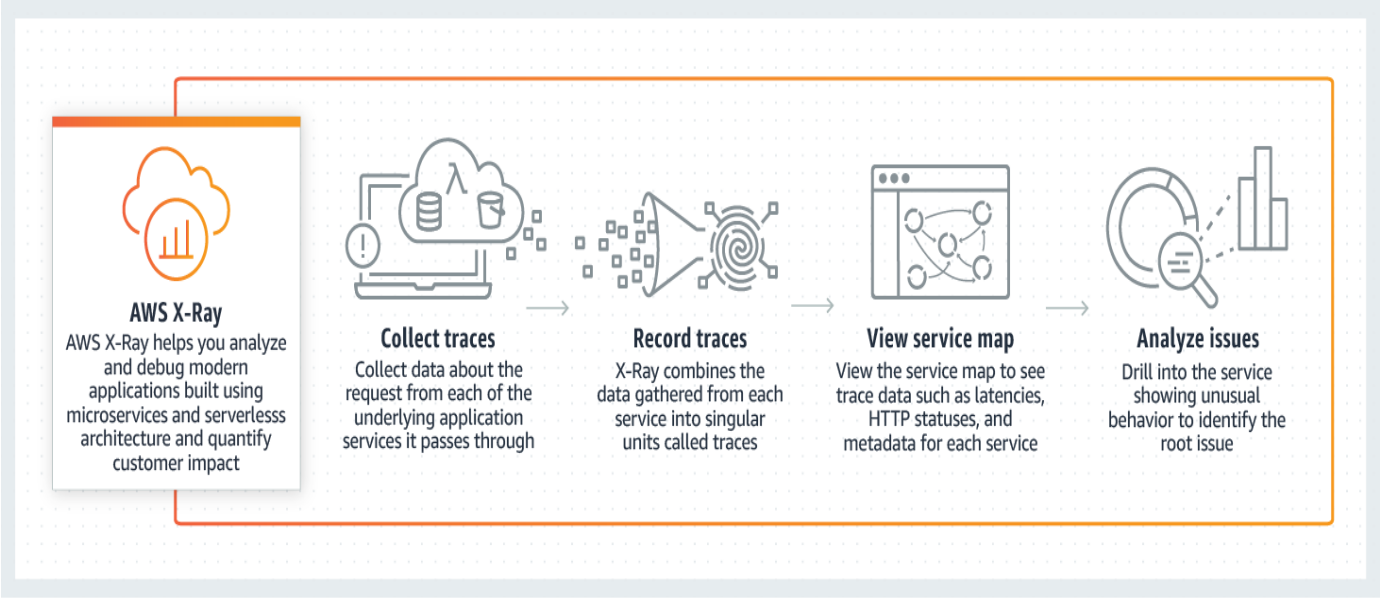
Anatomy of AWS Lambda



Lambda Function LifeCycle



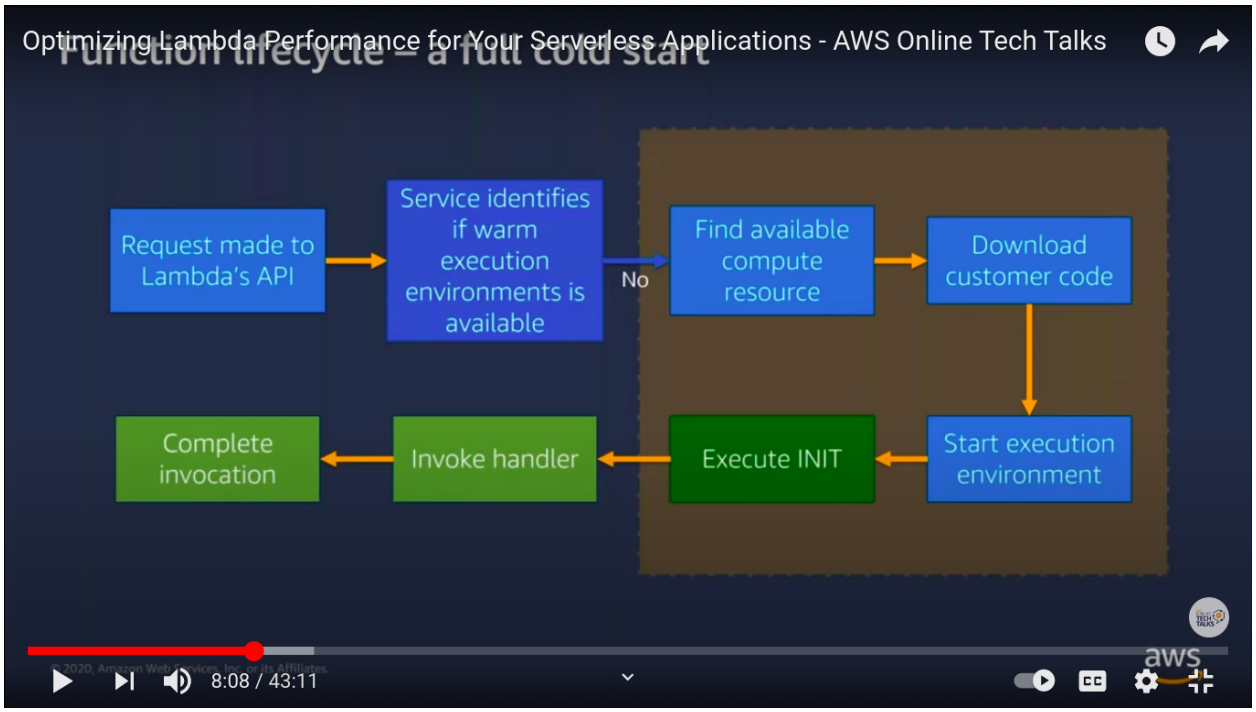
Measuring with AWS X-Ray



Area of Optimization

1. Latency
2. Throughput
3. Cost

Function Lifecycle : Full Cold Start



Cold Start usually take time in rage of 100 ms to 1 sec

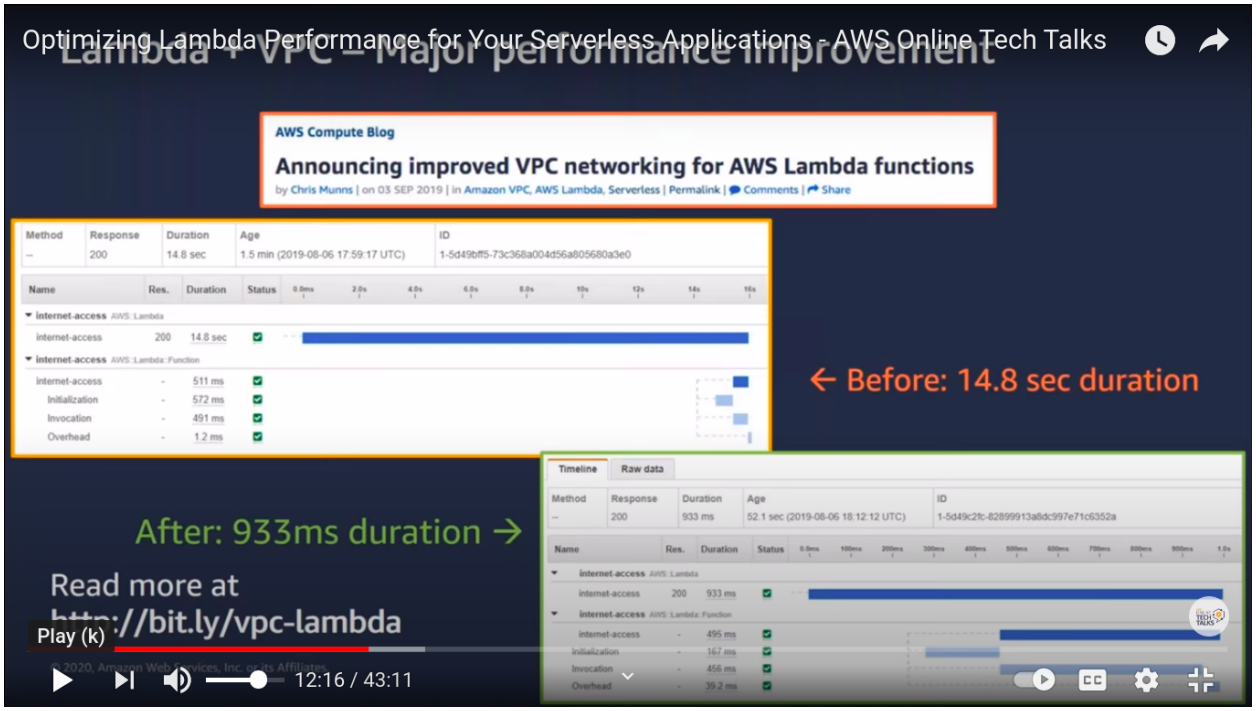
We can not manage Warm environment : Managed by AWS

Pinging function to keep it warm is limited

Cold Start Execution Environment Influenced by

- Memory Allocation
- Function Size
- Frequency of Function Call
- Internal Algorithm

Lambda + VPC improve performance



Cold Start : Static Initializer (INIT)

Facts :

- Managed by Developer
- Code Run Before Handler
- Used to establish connection
- Have much impact on Cold Start
- Occur when Lambda execute first time
- Also execute when Scaling Up lambda

Influenced By :

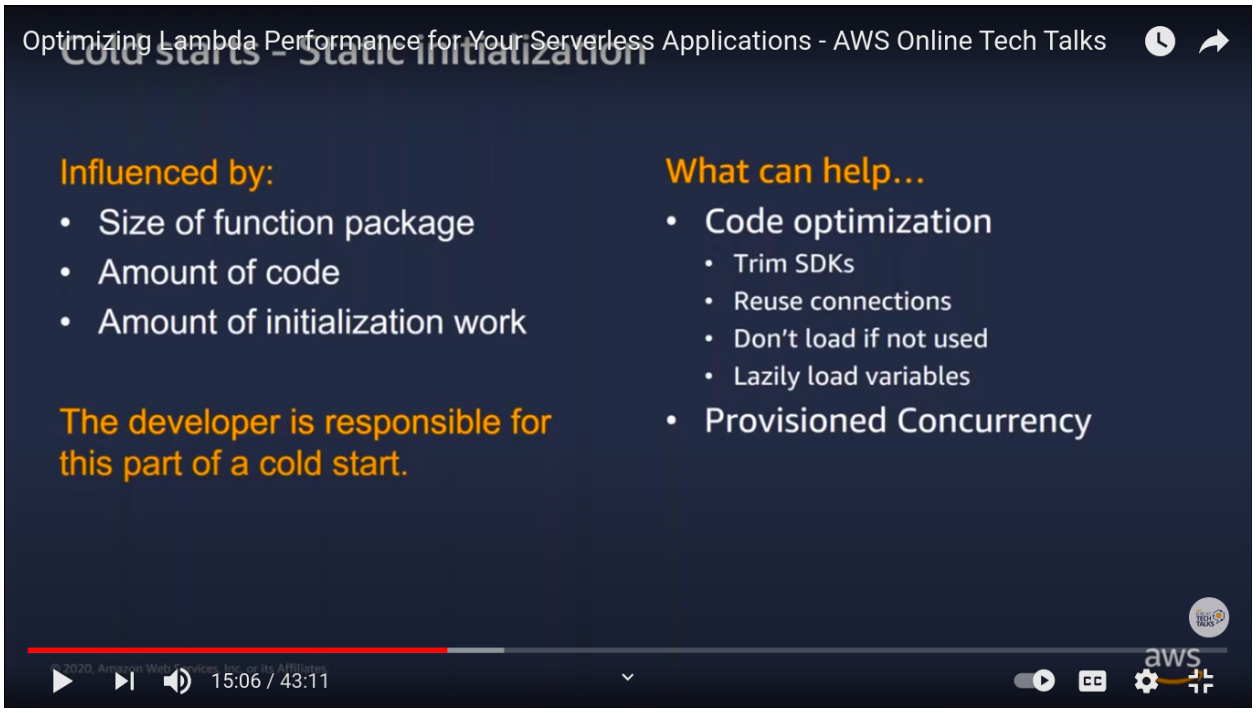
- ❖ Size of Function package
- ❖ Function Code
- ❖ Initialization work

What can help :

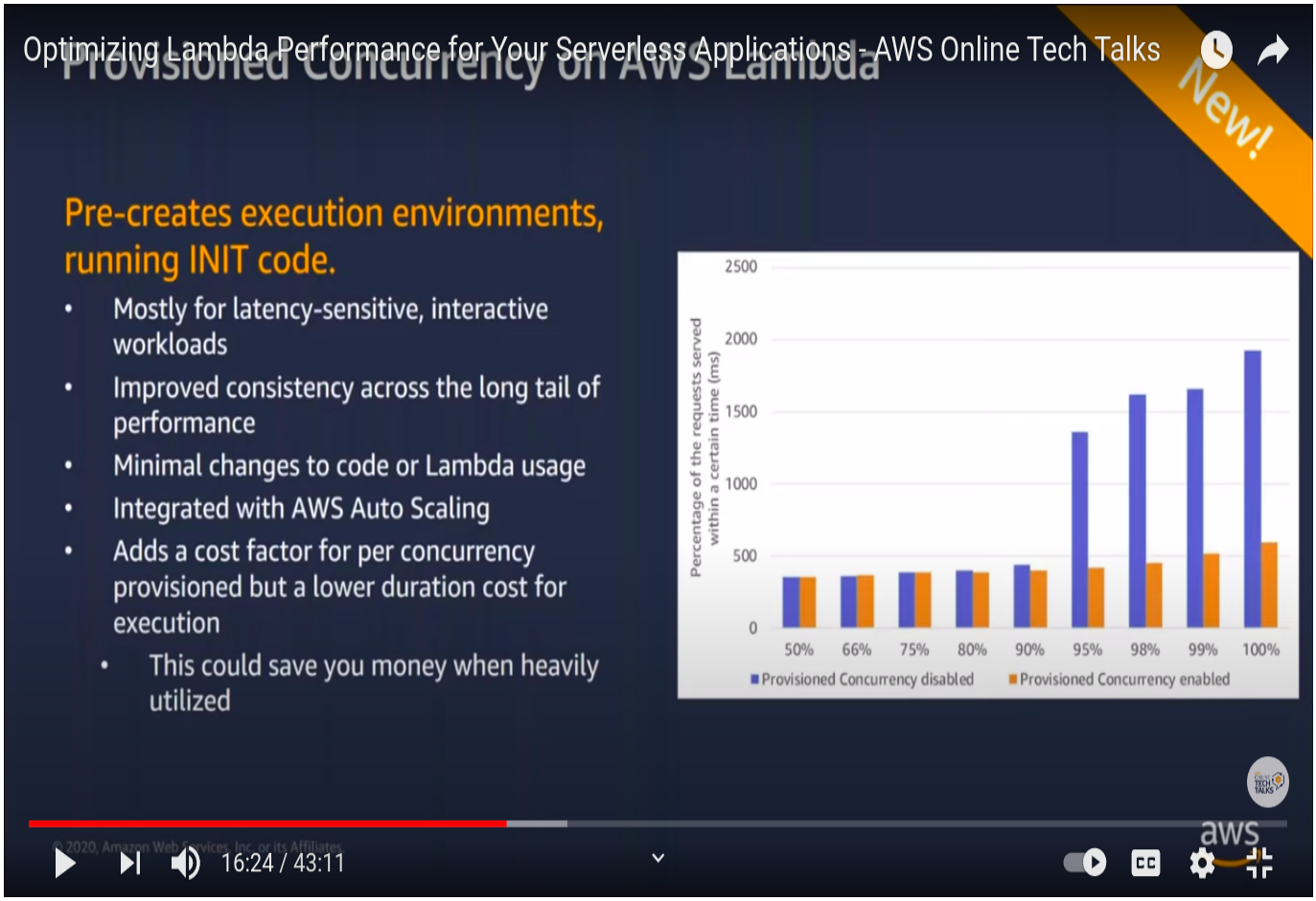
Code Optimization

- Trim SDK
- Reuse Connection
- Don’t Load if not used
- Lazily load variables
-

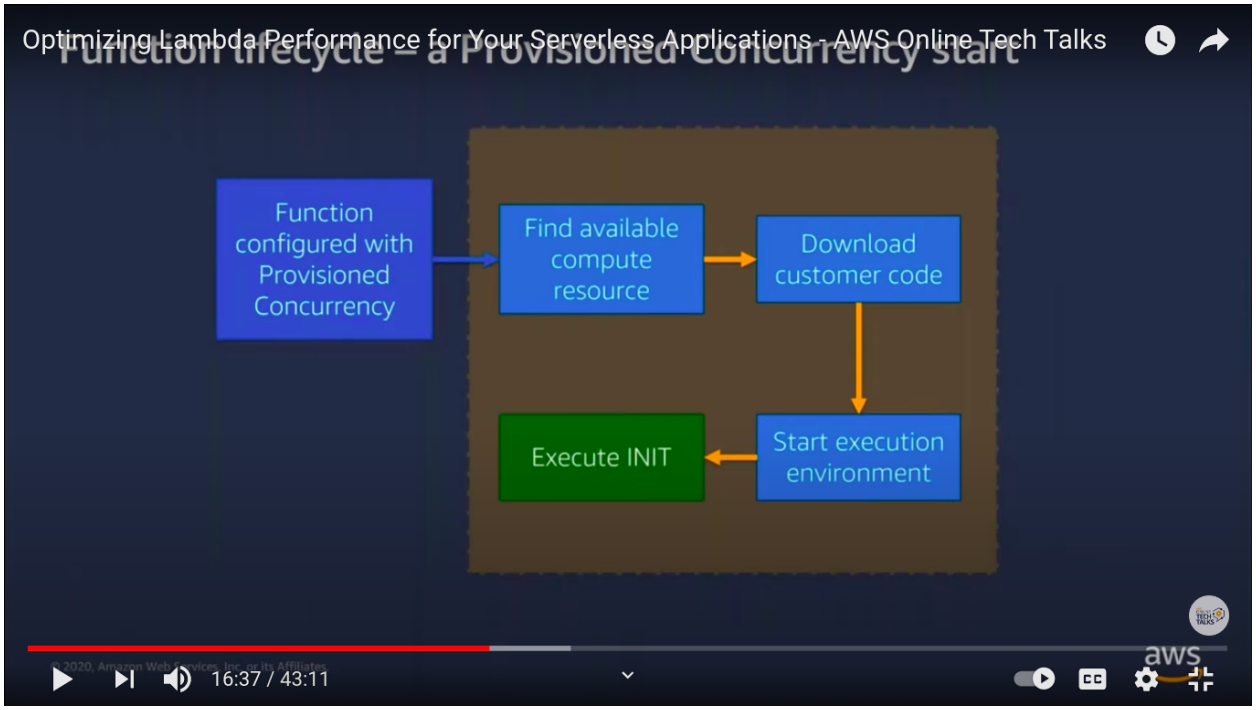
Provisioned Concurrency



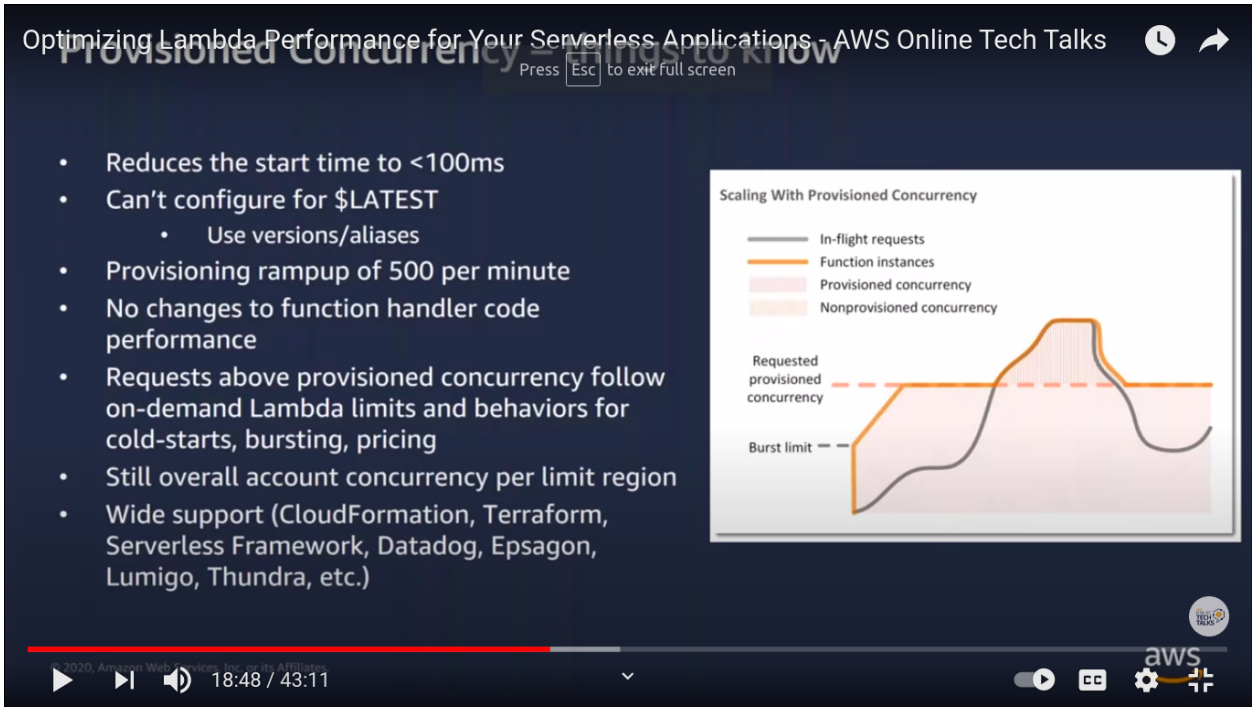
Provisioned Concurrency



Function Lifecycle with Provisioned Concurrency

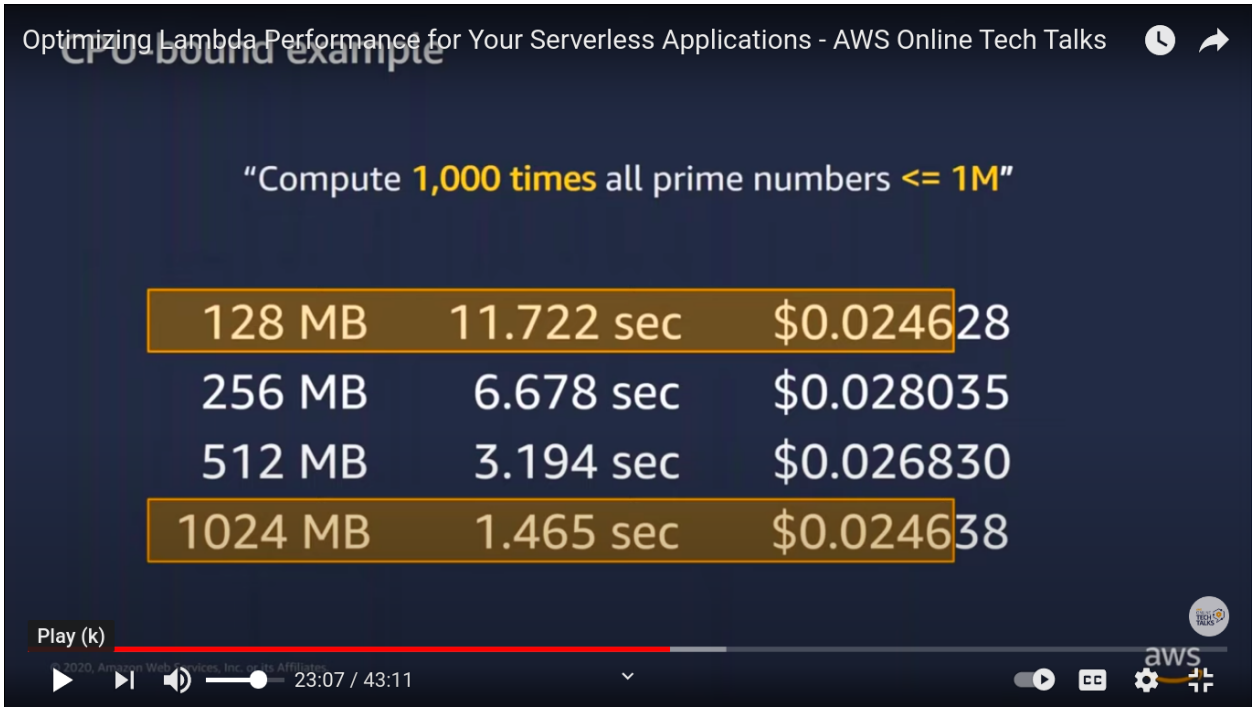


Provisioned Concurrency : Things to Know

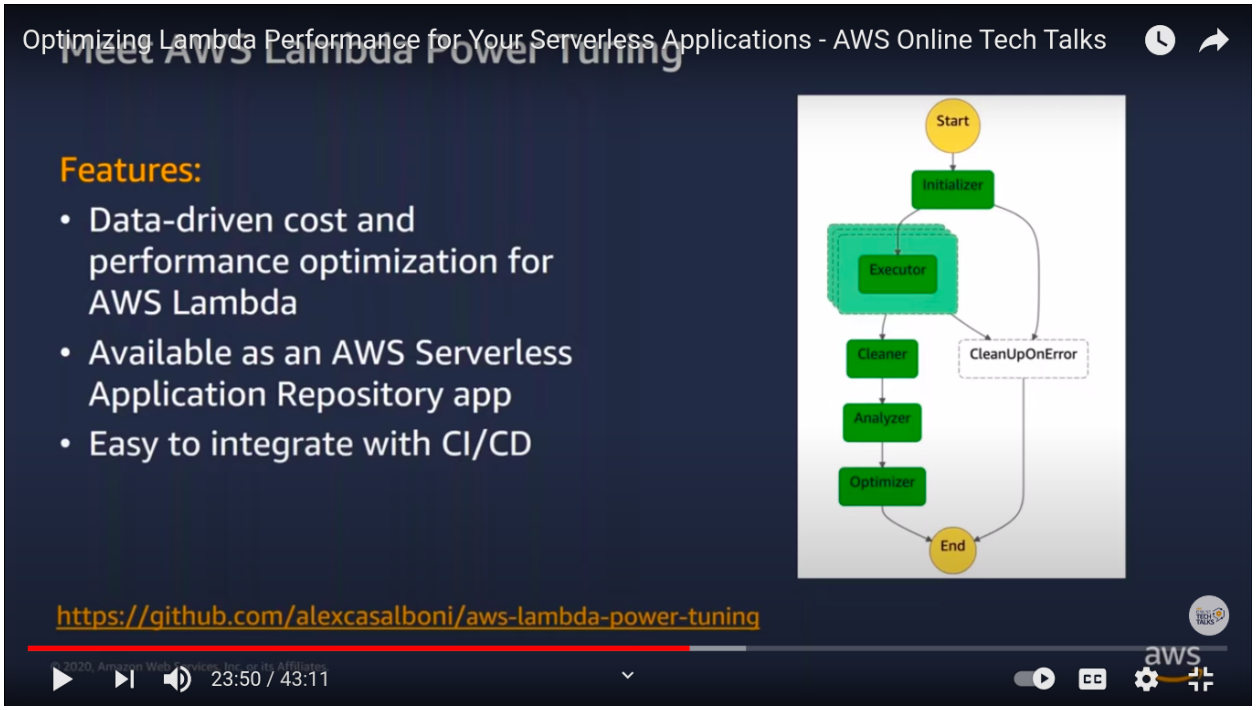


Memory and Profiling

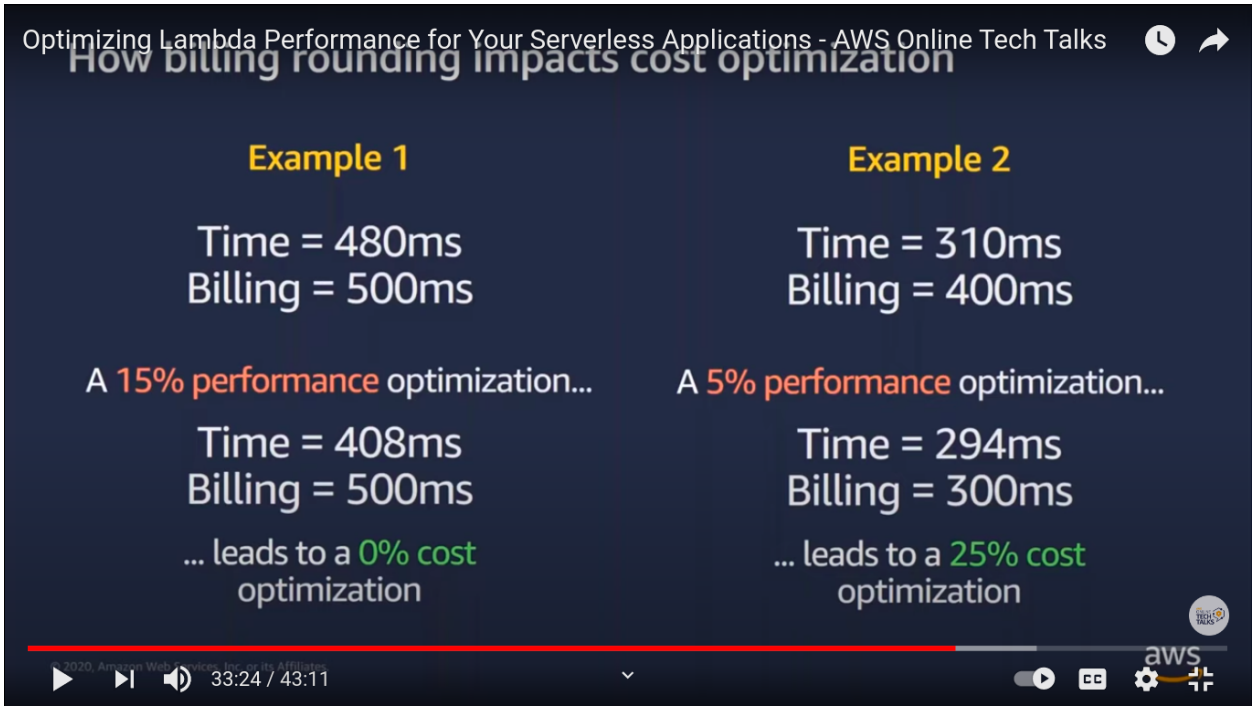
128 MB to 3000 MB



AWS Lambda Power Tunning

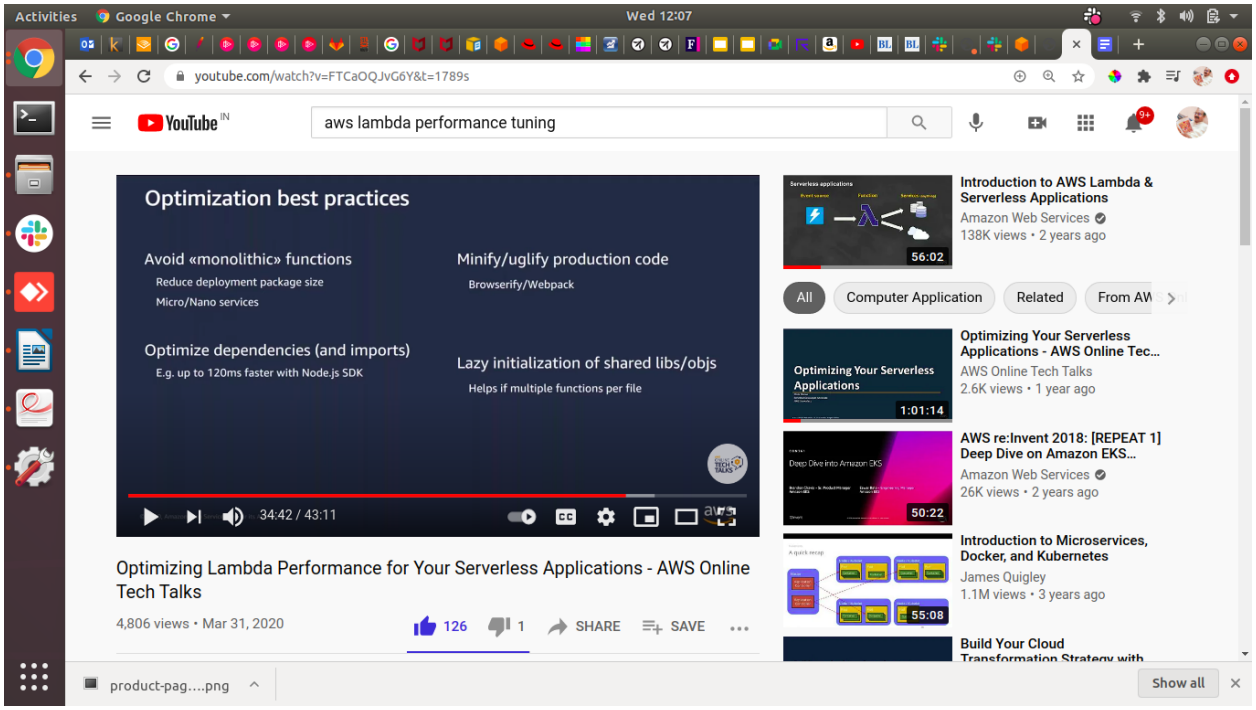


How billing rounding impact Cost Optimization

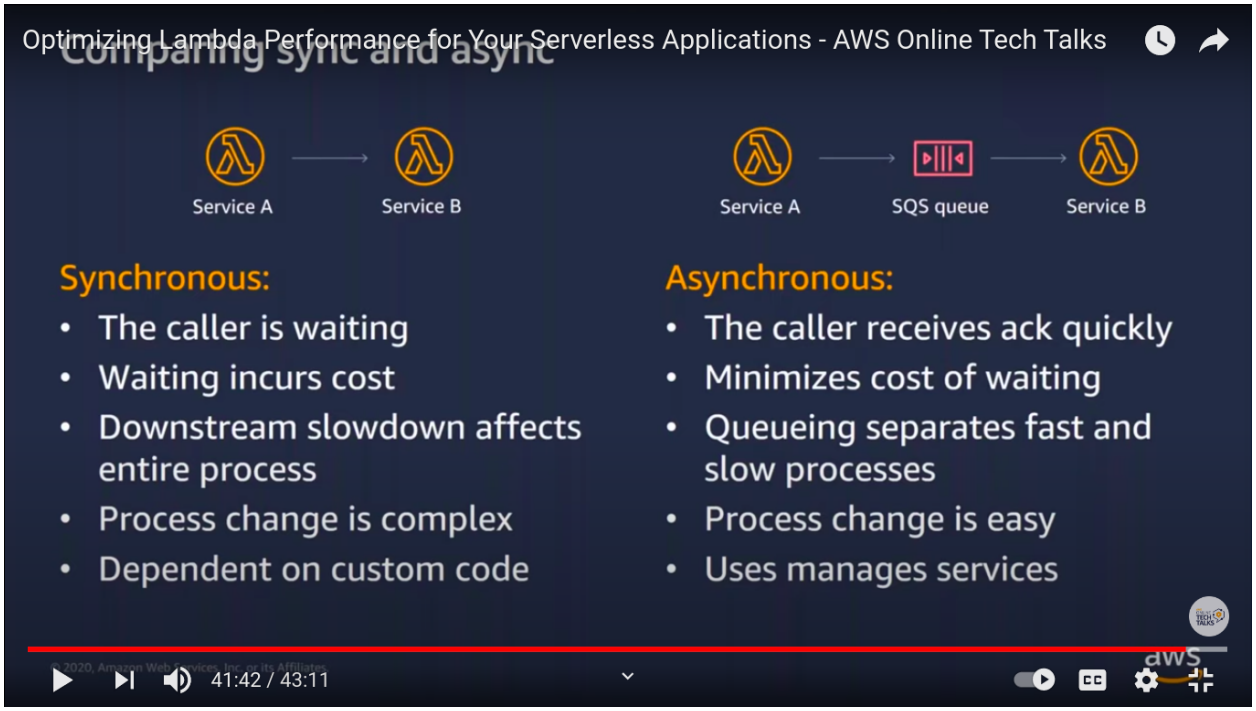


Architecture and Best Practises

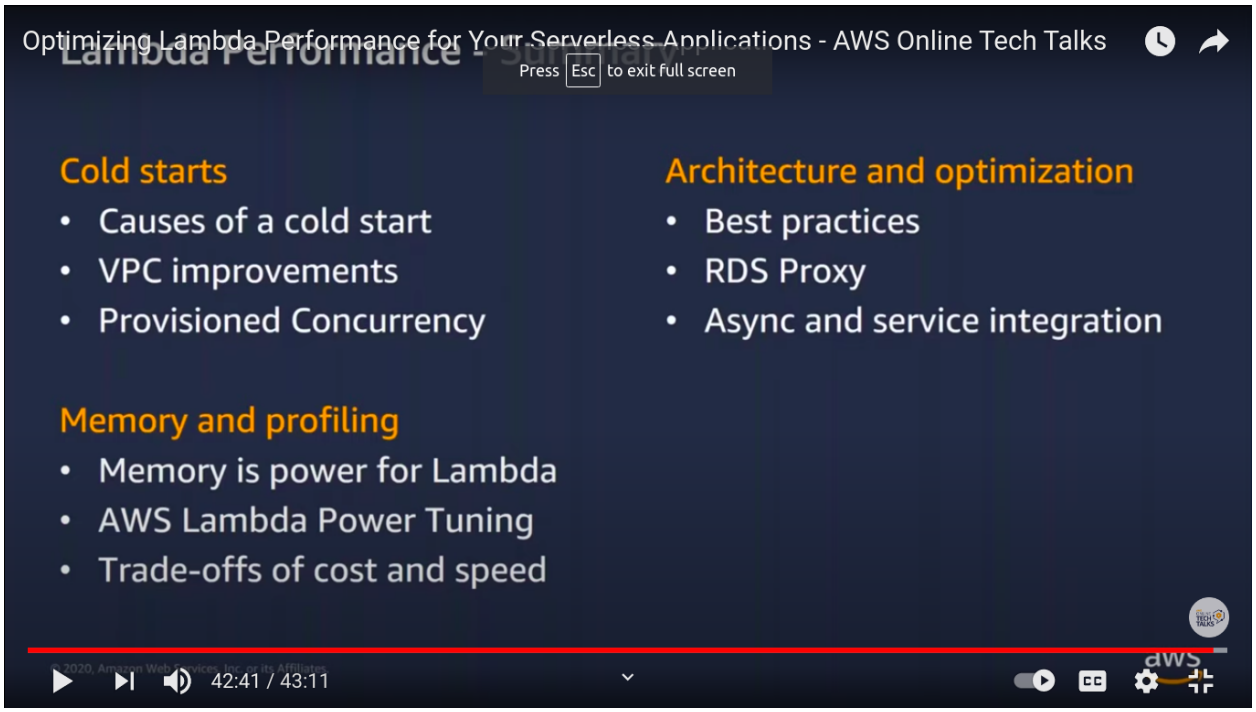
Optimization best practices



Sync Vs Async



Summary :


Optimizing Lambda Performance for Your Serverless Applications - AWS Online Tech Talks
Press Esc to exit full screen
Cold starts
• Causes of a cold start
• VPC improvements
• Provisioned Concurrency
Architecture and optimization
• Best practices
• RDS Proxy
• Async and service integration
Memory and profiling
• Memory is power for Lambda
• AWS Lambda Power Tuning
• Trade-offs of cost and speed
42:41 / 43:11