Machine Learning Lens

AWS Well-Architected Framework

Gilles-Kuessan Satchivi, Solutions Architect

June 17, 2021

Agenda

- Introduction to AWS Well Architected Framework
- Well Architected Machine Learning Lens Deep Dive
- Next Steps Resources

When you look at the system your team is building, can you answer the question:

“Are you Well-Architected?”

Are you Well-Architected?

- Operations
- Security
- Reliability
- Performance efficiency
- Cost optimization

What is the AWS Well-Architected Framework?

Pillars

Design principles

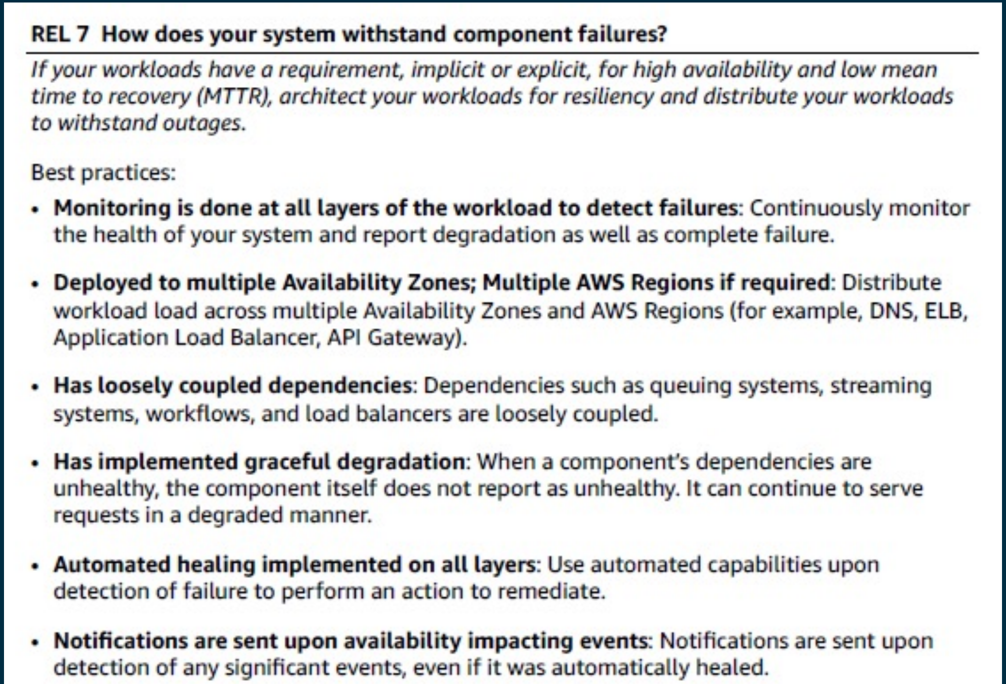
Questions

Why AWS Well-Architected Framework?

- Build and deploy faster
- Lower or mitigate risks
- Make informed decisions
- Learn AWS best practices

Well Architected Framework

Failure management



REL 7 How does your system withstand component failures?

If your workloads have a requirement, implicit or explicit, for high availability and low mean time to recovery (MTTR), architect your workloads for resiliency and distribute your workloads to withstand outages.

Best practices:

- **Monitoring is done at all layers of the workload to detect failures**: Continuously monitor the health of your system and report degradation as well as complete failure.
- **Deployed to multiple Availability Zones; Multiple AWS Regions if required**: Distribute workload load across multiple Availability Zones and AWS Regions (for example, DNS, ELB, Application Load Balancer, API Gateway).
- **Has loosely coupled dependencies**: Dependencies such as queuing systems, streaming systems, workflows, and load balancers are loosely coupled.
- **Has implemented graceful degradation**: When a component's dependencies are unhealthy, the component itself does not report as unhealthy. It can continue to serve requests in a degraded manner.
- **Automated healing implemented on all layers**: Use automated capabilities upon detection of failure to perform an action to remediate.
- **Notifications are sent upon availability impacting events**: Notifications are sent upon detection of any significant events, even if it was automatically healed.

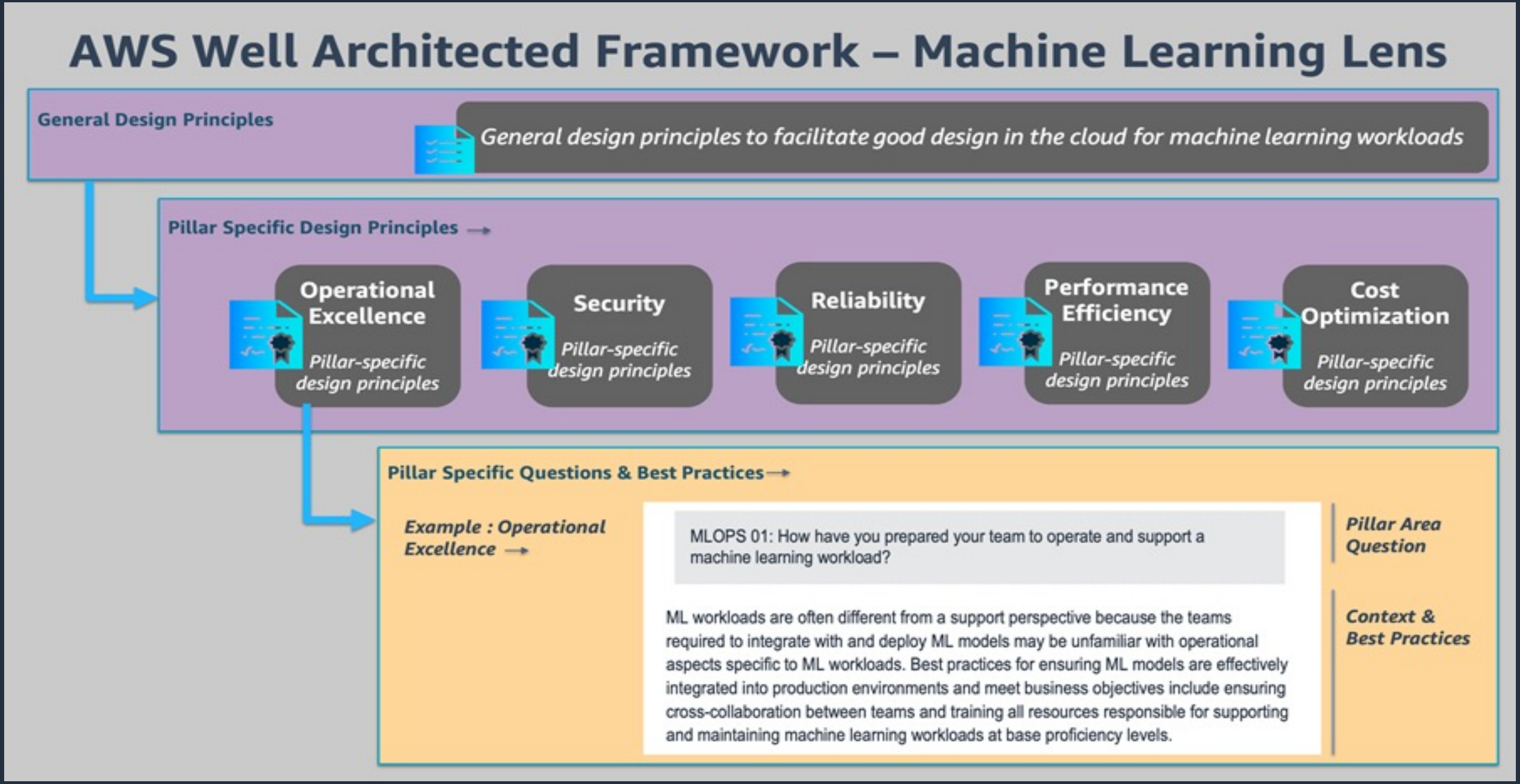
Pillar area

Question

Context

Best practices

ML Lens - Structure



AWS Well Architected Framework – Machine Learning Lens

General Design Principles

General design principles to facilitate good design in the cloud for machine learning workloads

Pillar Specific Design Principles →

Operational Excellence
Pillar-specific design principles

Security
Pillar-specific design principles

Reliability
Pillar-specific design principles

Performance Efficiency
Pillar-specific design principles

Cost Optimization
Pillar-specific design principles

Pillar Specific Questions & Best Practices →

Example : Operational Excellence →

MLOPS 01: How have you prepared your team to operate and support a machine learning workload?

Pillar Area Question

ML workloads are often different from a support perspective because the teams required to integrate with and deploy ML models may be unfamiliar with operational aspects specific to ML workloads. Best practices for ensuring ML models are effectively integrated into production environments and meet business objectives include ensuring cross-collaboration between teams and training all resources responsible for supporting and maintaining machine learning workloads at base proficiency levels.

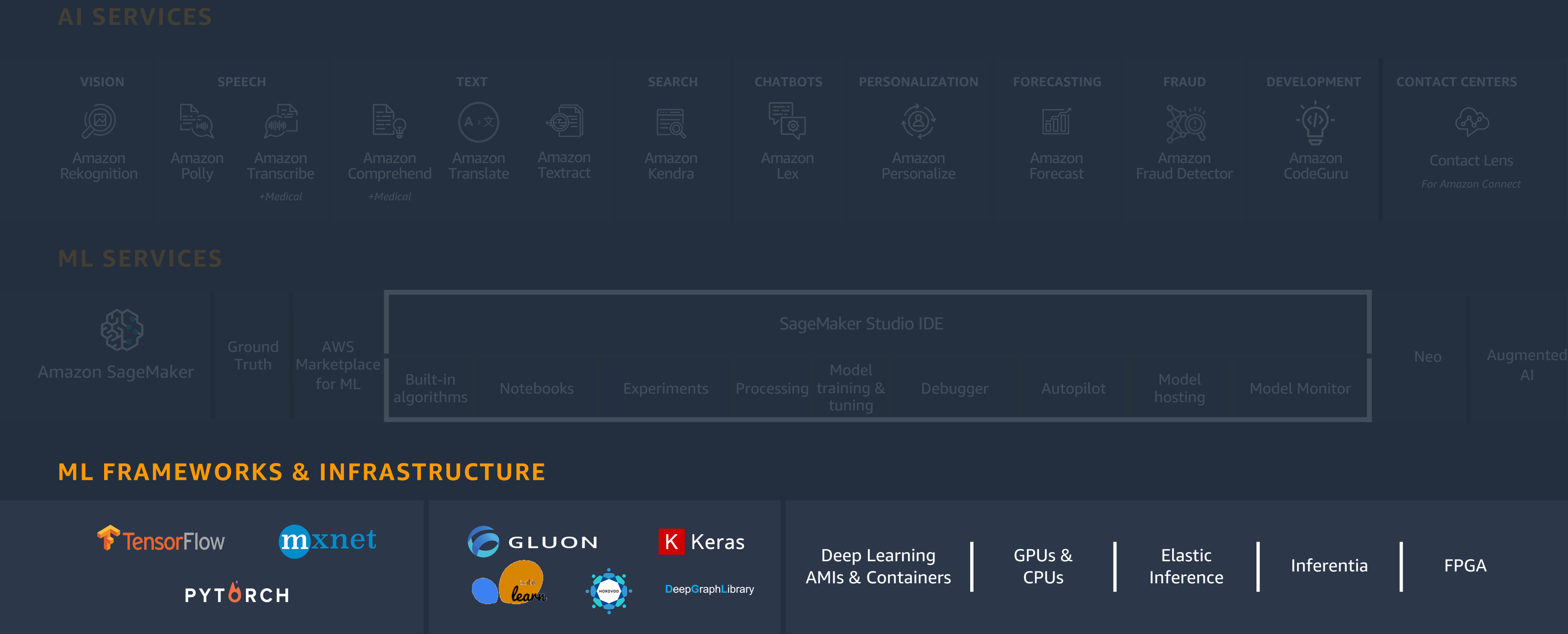
Context & Best Practices

Machine Learning Lens

The AWS ML Stack

The AWS ML Stack

Broadest and most complete set of Machine Learning capabilities



AI SERVICES

VISION: Amazon Rekognition

SPEECH: Amazon Polly; Amazon Transcribe *+Medical*

TEXT: Amazon Comprehend *+Medical*; Amazon Translate; Amazon Textract

SEARCH: Amazon Kendra

CHATBOTS: Amazon Lex

PERSONALIZATION: Amazon Personalize

FORECASTING: Amazon Forecast

FRAUD: Amazon Fraud Detector

DEVELOPMENT: Amazon CodeGuru

CONTACT CENTERS: Contact Lens *For Amazon Connect*

ML SERVICES

Amazon SageMaker

Ground Truth

AWS Marketplace for ML

SageMaker Studio IDE

Built-in algorithms | Notebooks | Experiments | Processing | Model training & tuning | Debugger | Autopilot | Model hosting | Model Monitor

Neo

Augmented AI

ML FRAMEWORKS & INFRASTRUCTURE

TensorFlow

mxnet

PYTORCH

GLUON

Keras

scikit-learn

HOROVOD

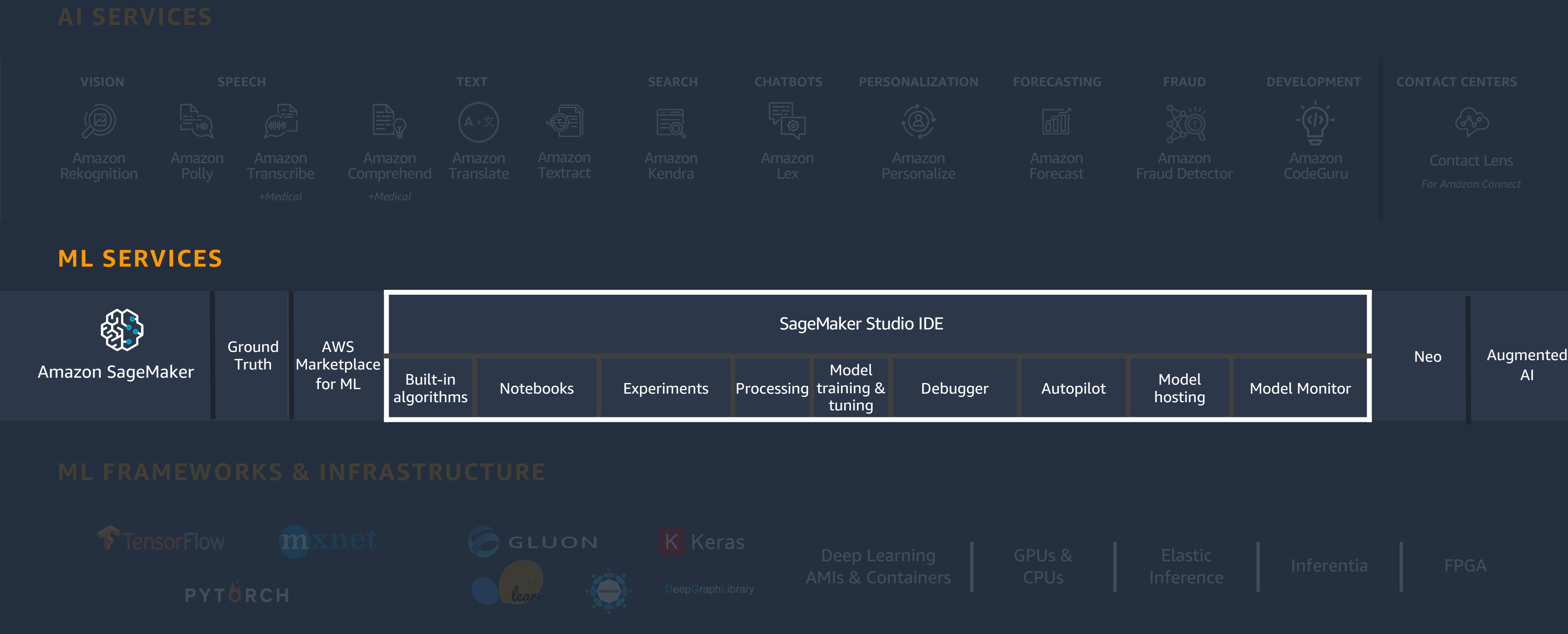
DeepGraphLibrary

Deep Learning AMIs & Containers | GPUs & CPUs | Elastic Inference | Inferentia | FPGA

aws

The AWS ML Stack

Broadest and most complete set of Machine Learning capabilities



AI SERVICES

- VISION: Amazon Rekognition
- SPEECH: Amazon Polly, Amazon Transcribe (+Medical)
- TEXT: Amazon Comprehend (+Medical), Amazon Translate, Amazon Textract
- SEARCH: Amazon Kendra
- CHATBOTS: Amazon Lex
- PERSONALIZATION: Amazon Personalize
- FORECASTING: Amazon Forecast
- FRAUD: Amazon Fraud Detector
- DEVELOPMENT: Amazon CodeGuru
- CONTACT CENTERS: Contact Lens *For Amazon Connect*

ML SERVICES

- Amazon SageMaker
- Ground Truth
- AWS Marketplace for ML
- SageMaker Studio IDE
  - Built-in algorithms
  - Notebooks
  - Experiments
  - Processing
  - Model training & tuning
  - Debugger
  - Autopilot
  - Model hosting
  - Model Monitor
- Neo
- Augmented AI

ML FRAMEWORKS & INFRASTRUCTURE

- TensorFlow
- mxnet
- PYTORCH
- GLUON
- Keras
- learn
- DeepGraphLibrary
- Deep Learning AMIs & Containers
- GPUs & CPUs
- Elastic Inference
- Inferentia
- FPGA

The AWS ML Stack

Broadest and most complete set of Machine Learning capabilities

AI SERVICES

VISION
- Amazon Rekognition

SPEECH
- Amazon Polly
- Amazon Transcribe *+Medical*

TEXT
- Amazon Comprehend *+Medical*
- Amazon Translate
- Amazon Textract

SEARCH
- Amazon Kendra

CHATBOTS
- Amazon Lex

PERSONALIZATION
- Amazon Personalize

FORECASTING
- Amazon Forecast

FRAUD
- Amazon Fraud Detector

DEVELOPMENT
- Amazon CodeGuru

CONTACT CENTERS
- Contact Lens *For Amazon Connect*

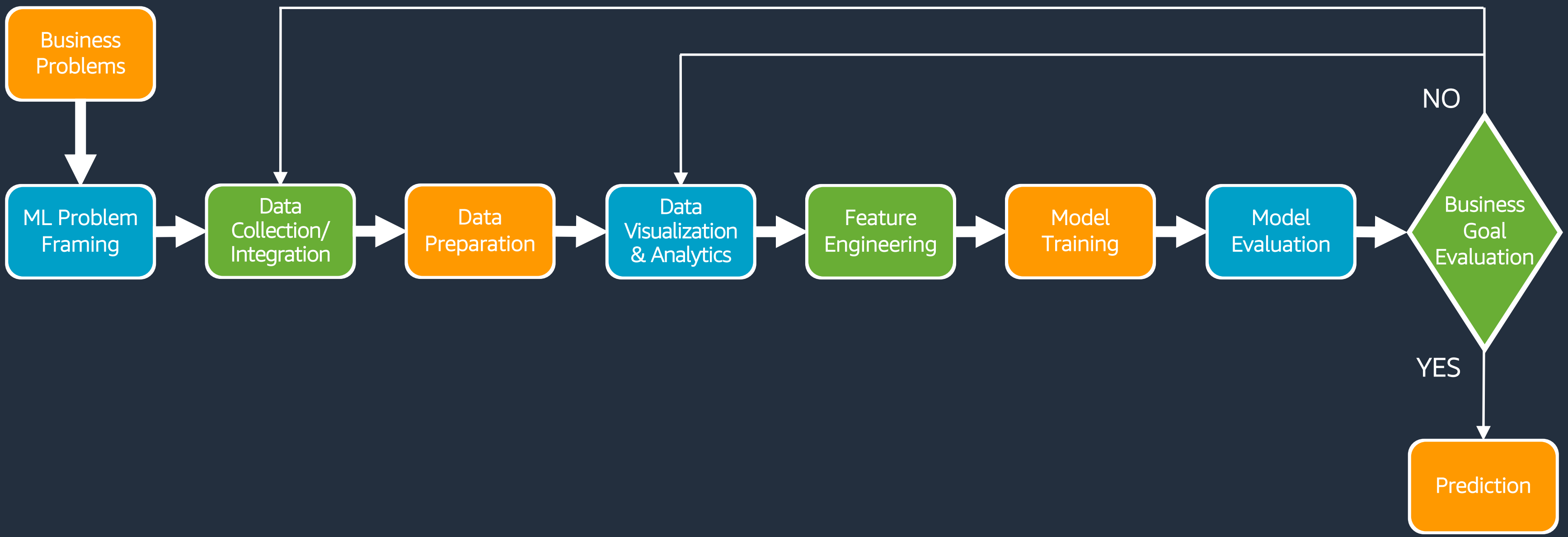
ML SERVICES

- Amazon SageMaker
- Ground Truth
- AWS Marketplace for ML
- SageMaker Studio IDE
  - Built-in algorithms
  - Notebooks
  - Experiments
  - Processing
  - Model training & tuning
  - Debugger
  - Autopilot
  - Model hosting
  - Model Monitor
- Neo
- Augmented AI

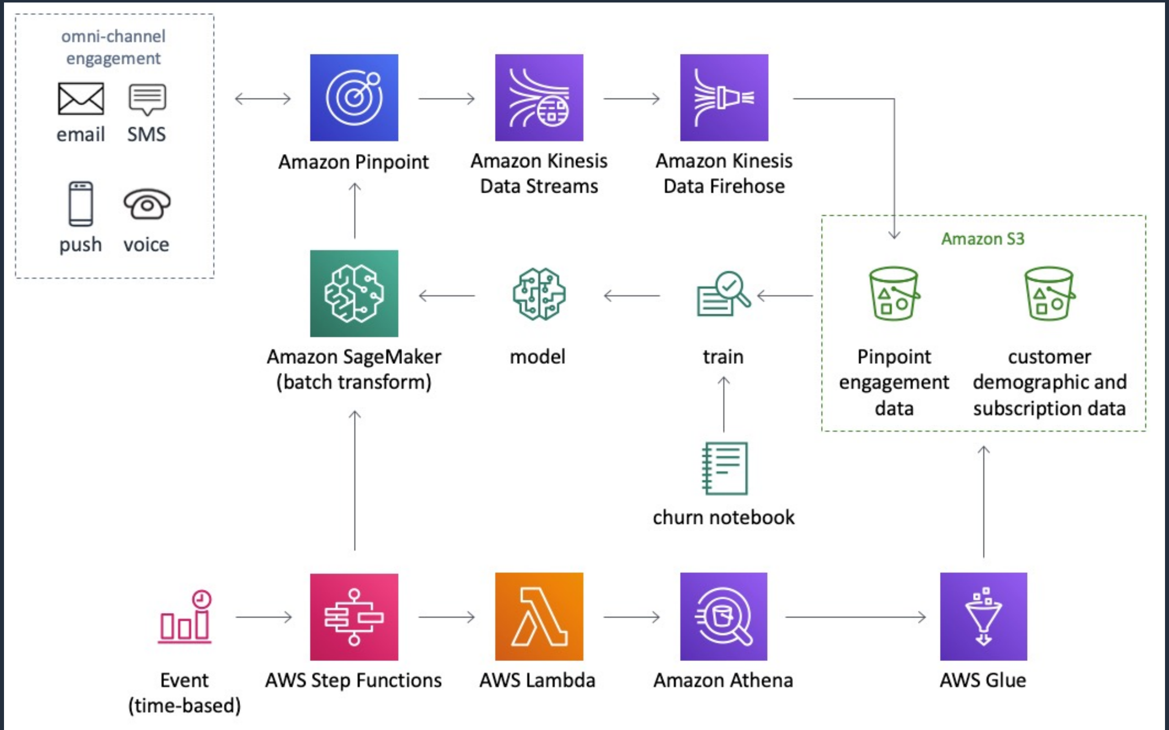
ML FRAMEWORKS & INFRASTRUCTURE

- TensorFlow
- mxnet
- PYTORCH
- GLUON
- K Keras
- learn
- DeepGraphLibrary
- Deep Learning AMIs & Containers
- GPUs & CPUs
- Elastic Inference
- Inferentia
- FPGA

Machine Learning Phases



Use Case: Predictive Segmentation Using Amazon Pinpoint and Amazon SageMaker

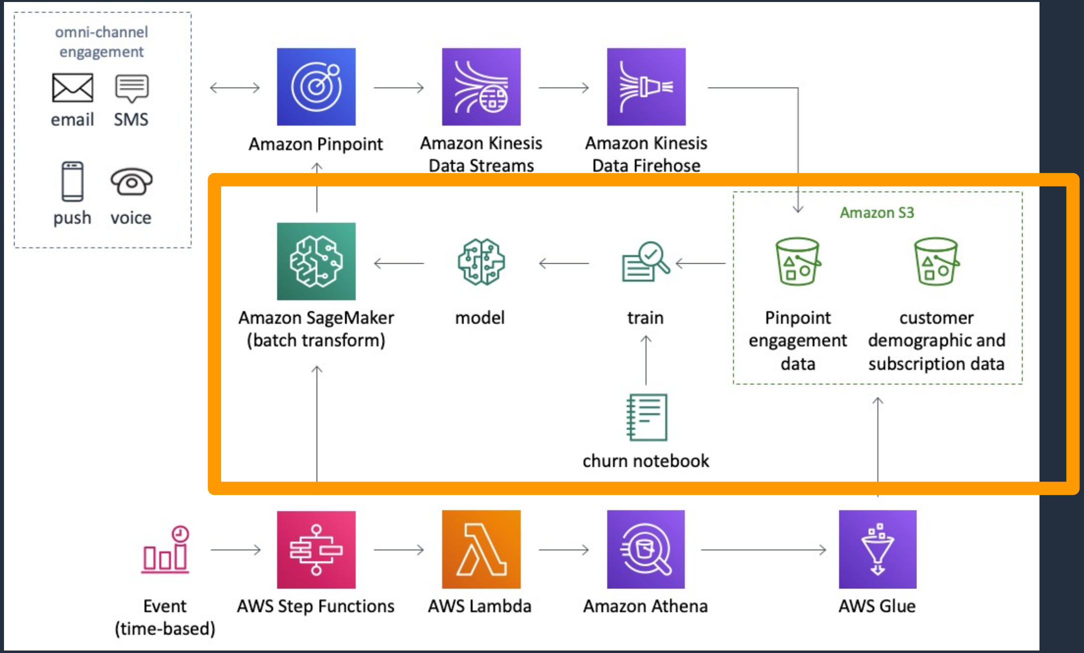


Machine Learning Lens

Security Pillar

- Restrict Access to ML systems
- Ensure Data Governance
- Enforce Data Lineage
- Enforce Regulatory Compliance

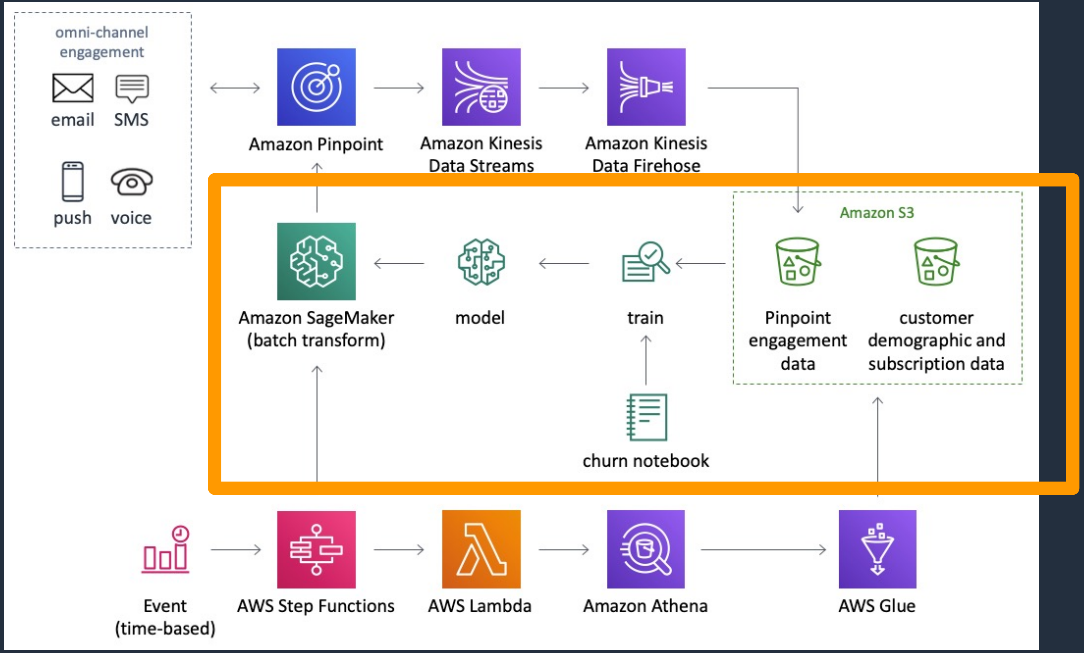
Security Pillar



How do you control access to your ML workload?

- Enforce least privileged based access
- Secure access to hosted model endpoint

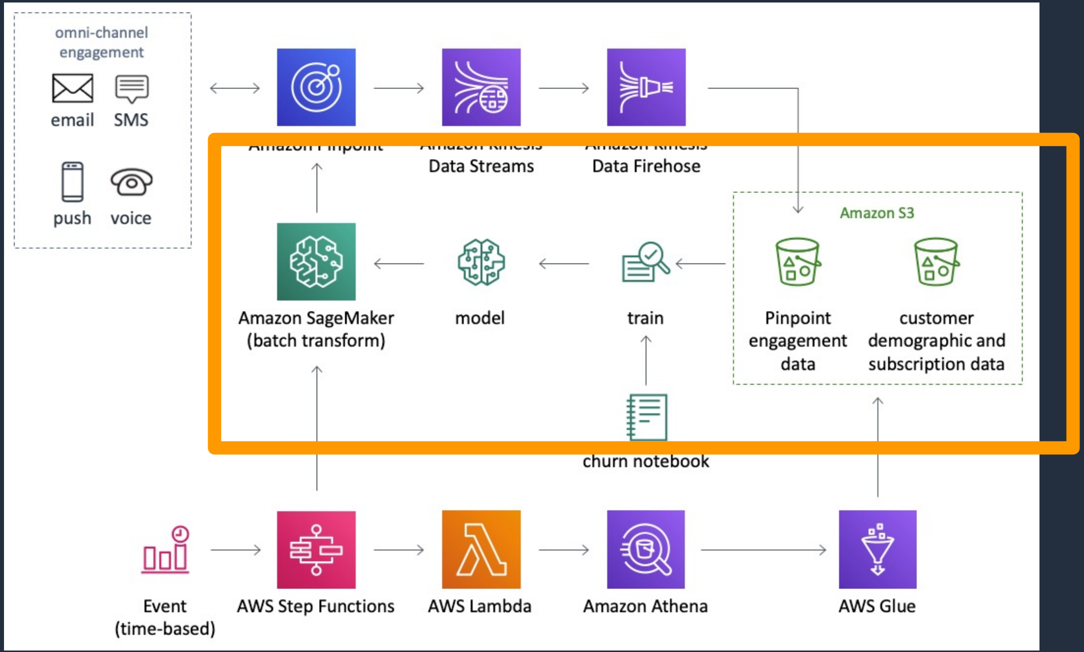
Security Pillar



How are you protecting trained ML models?

- **Enforce data classification**
- **Centralized datalake**
- **Data encryption**
- **Least privilege based access**

Security Pillar



How are you protecting and monitoring access to sensitive data used in your ML workloads?

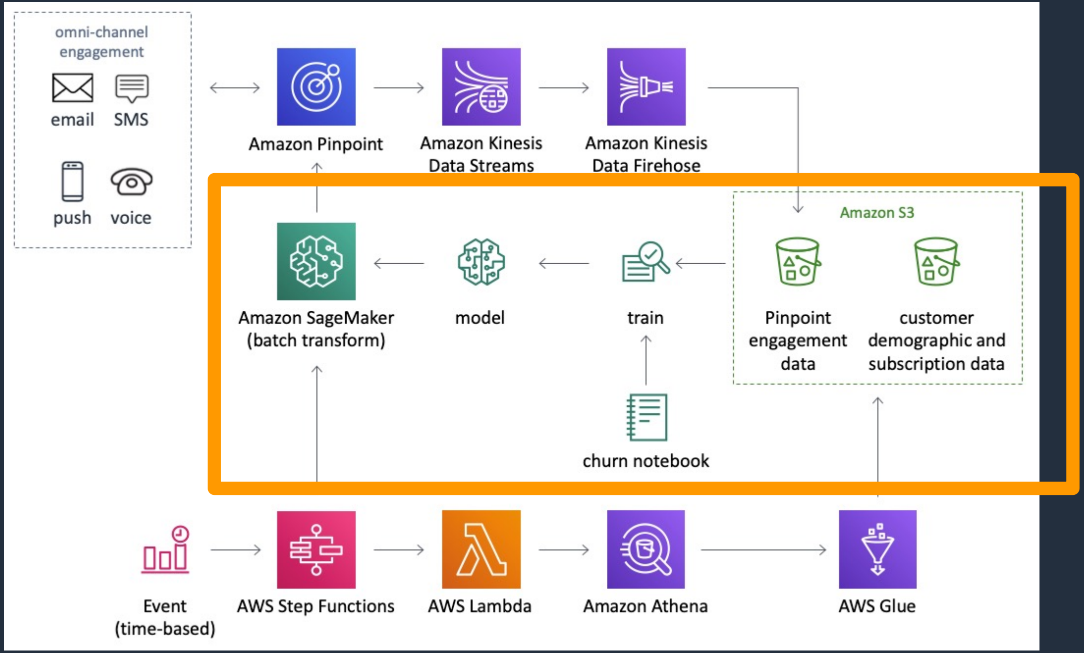
- **Secure model artifacts**
- **Secure hosted model**
- **Controlled external access to hosted model**

Machine Learning Lens

Cost Optimization Pillar

- Use managed services to reduce cost of ownership
- Experiment with small datasets
- Right size training and model hosting instances
- Account for inference architecture based on consumption patterns
- Define overall ROI and opportunity cost

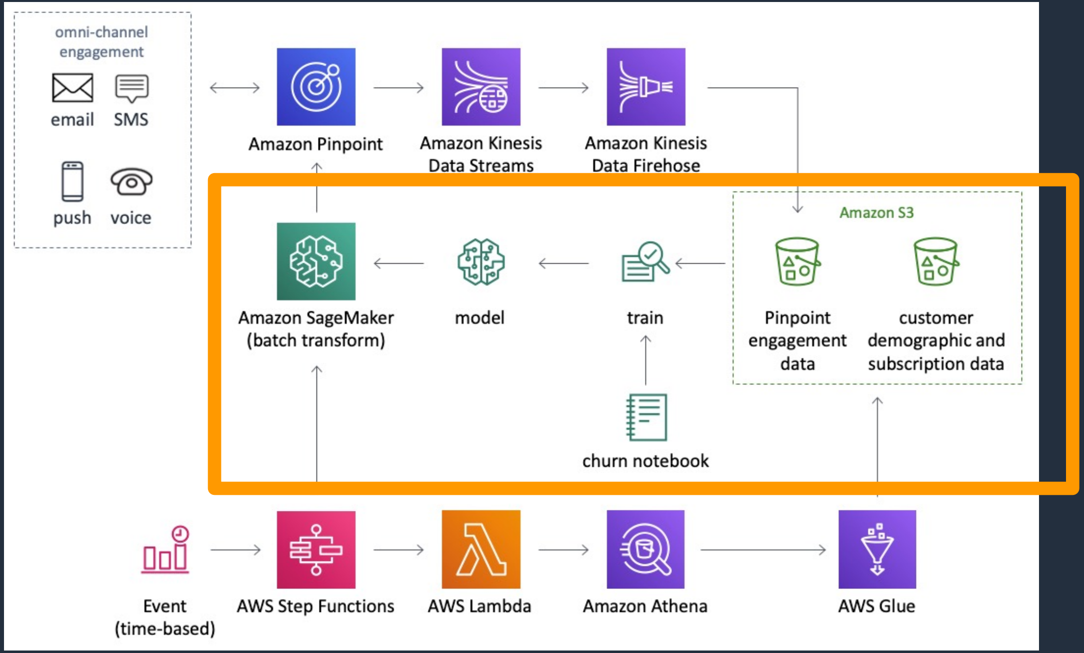
Cost Optimization Pillar



How do you optimize data labeling costs?

- **UI based annotation tool**
- **Managed service for annotation**
- **Combination of manual and machine learning for annotation**

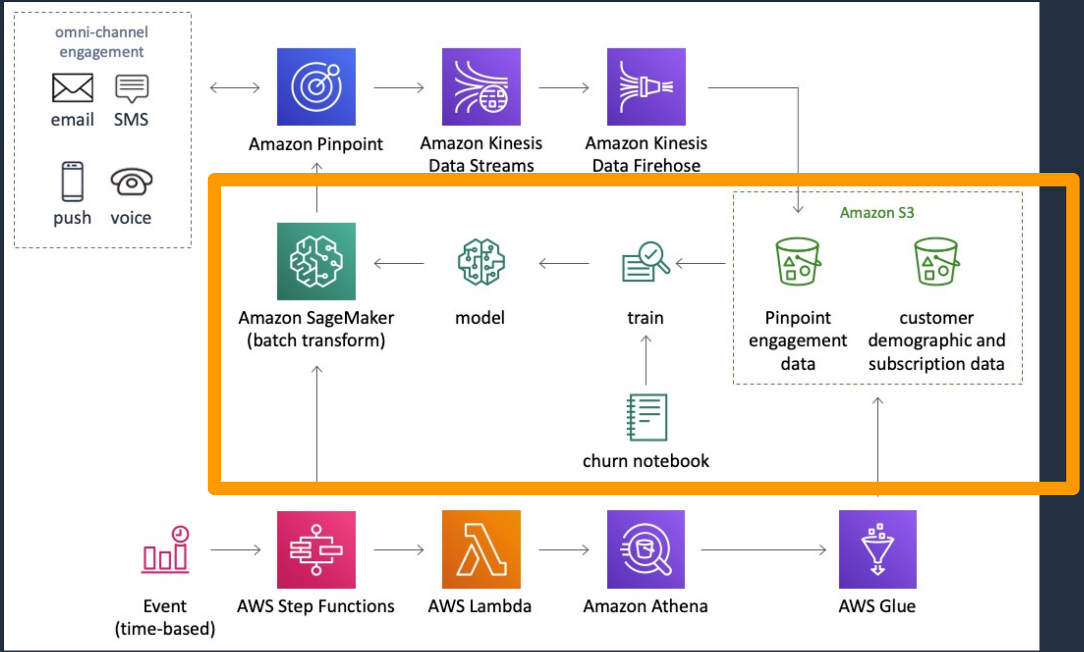
Cost Optimization Pillar



How do you optimize costs during ML experimentation?

- **Managed notebooks**
- **Local experimentation**
- **Explore AWS Marketplace for machine learning**

Cost Optimization Pillar



How do you optimize cost for ML Inference?

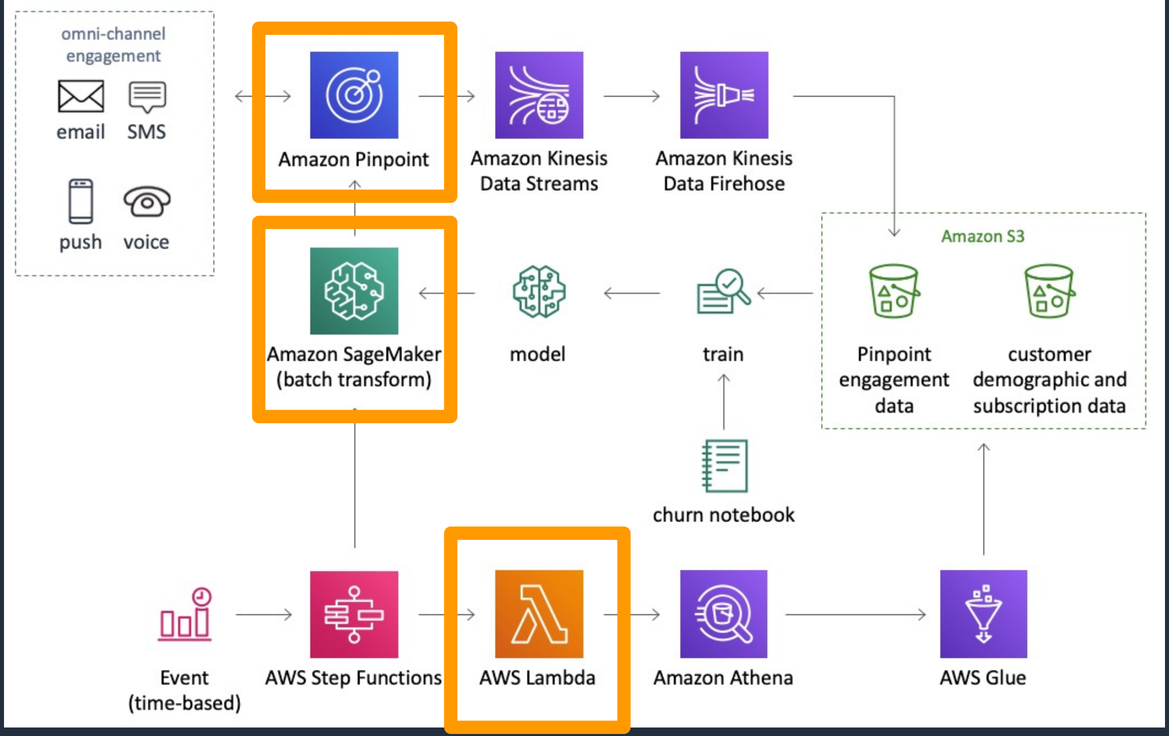
- **Right size the hosting cluster**
- **Autoscale**
- **Differentiate between CPU vs GPU needs**
- **Real-time vs on-demand inference architecture**

Machine Learning Lens

Performance Efficiency Pillar

- Selection
- Review
- Monitoring
- Tradeoffs

Performance Efficiency Pillar

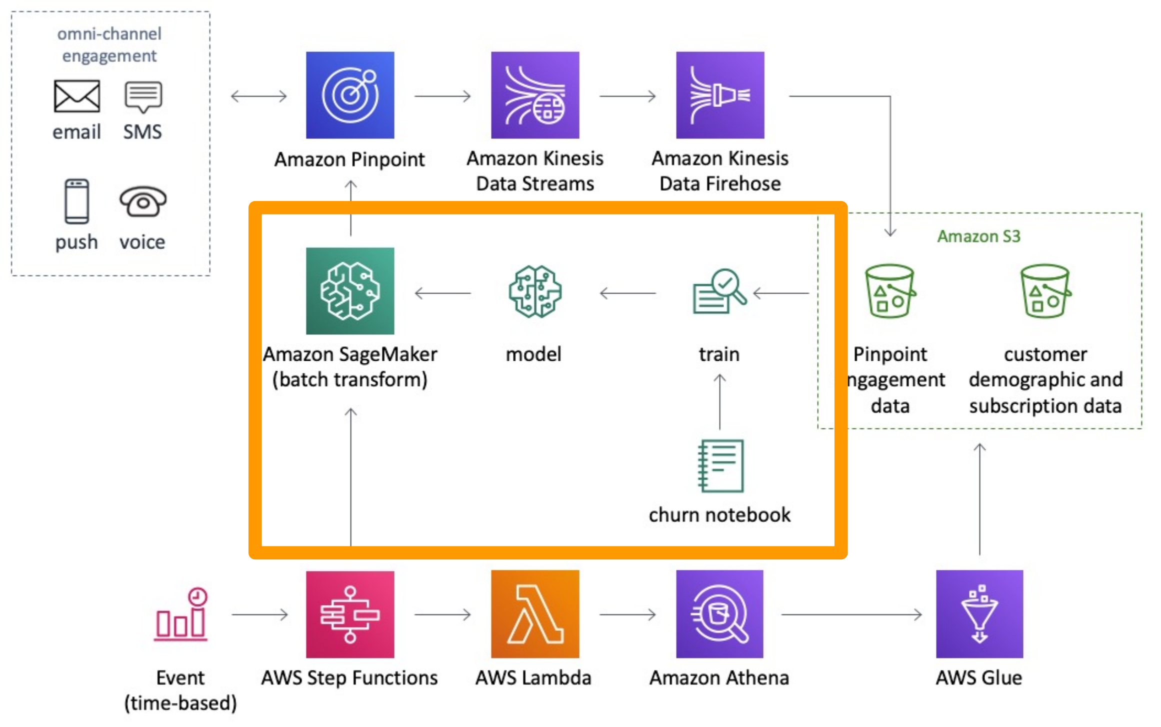


Optimize compute for your ML workload

Considerations:

- Managed Services vs.
- Layer 2 ML Services vs.
- Serverless

Performance Efficiency Pillar

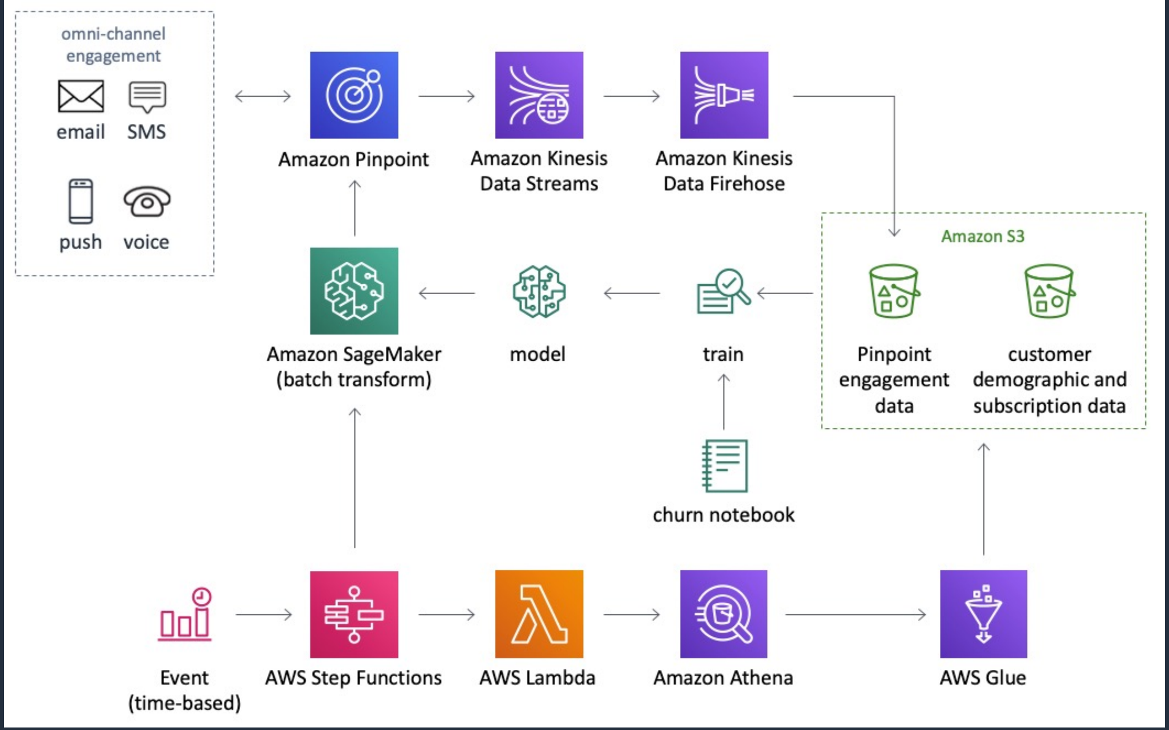


Continuously monitor and measure system performance:

Considerations:

- What are your goals for monitoring?
- What resources will you monitor?
- Who should be notified when something goes wrong?

Performance Efficiency Pillar



Continuously Review:

Considerations:

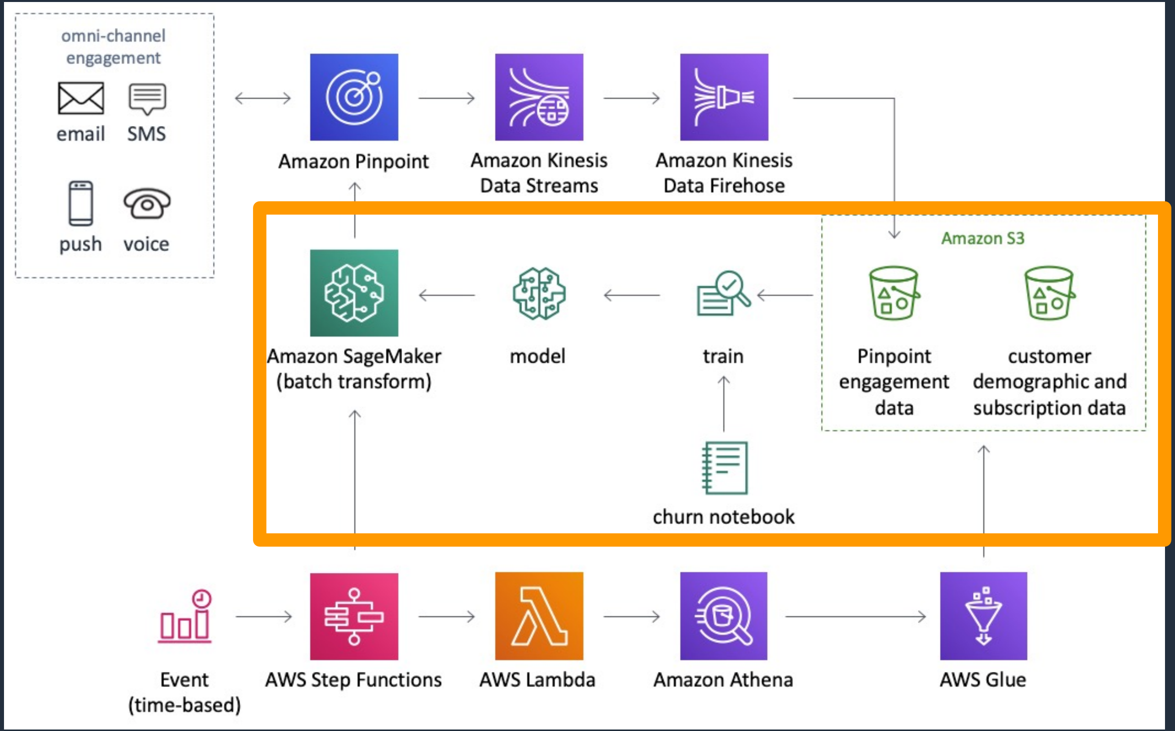
- Continuous Improvement Is Critical
- Cost optimization is not a task, it’s a way of life
- Continuous Reviews are part of Operational Excellence

Machine Learning Lens

Operational Excellence Pillar

- Establish cross functional teams
- Identify the end-to-end architecture and operational model early
- Continuously monitor and measure ML workloads
- Establish a model retraining strategy
- Document machine learning discovery activities and findings
- Version machine learning inputs and artifacts
- Automate machine learning deployment pipelines

Operational Excellence Pillar



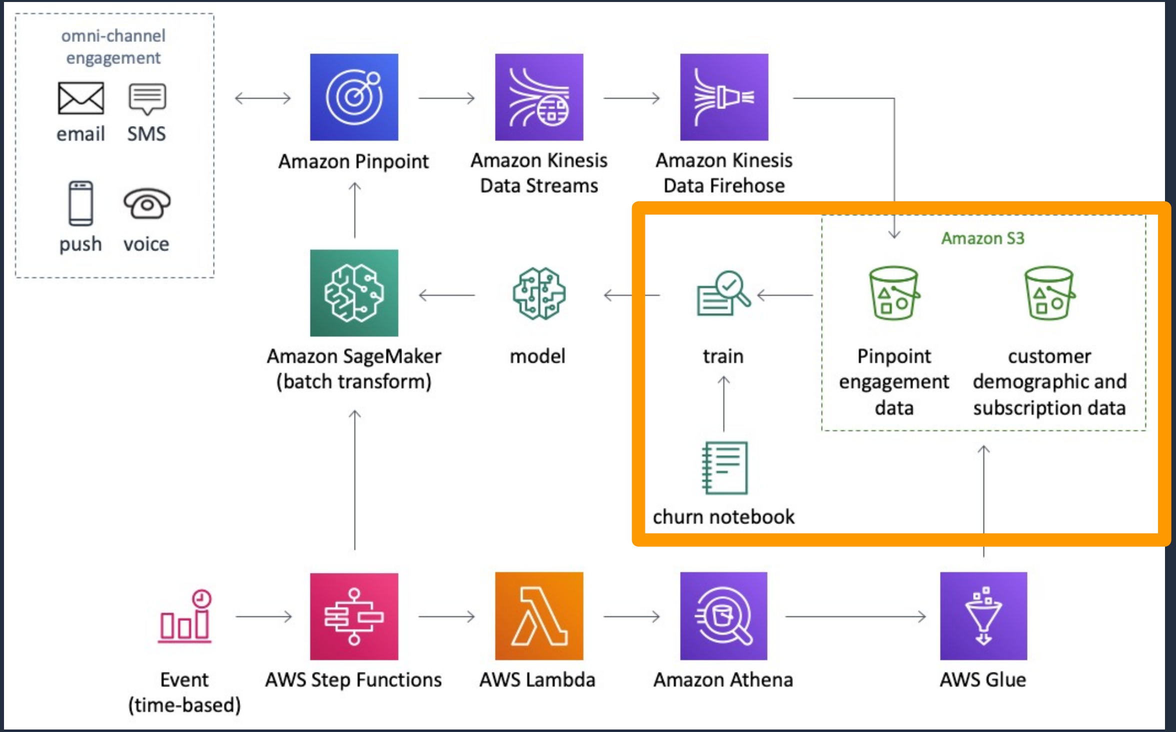
How have you prepared your team to operate and support a machine learning workload?

Data Scientist

Software Engineer

Infrastructure/Operations

Operational Excellence Pillar



How do you know when to retrain ML models with new or updated data?

Strategy:

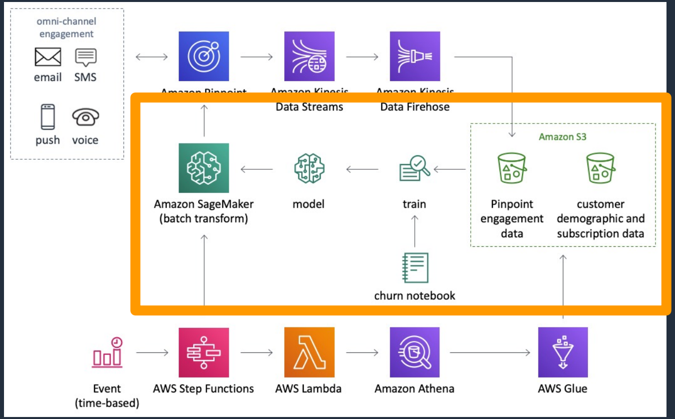
- Metric Driven
- Scheduled/New Data

Retraining Considerations:

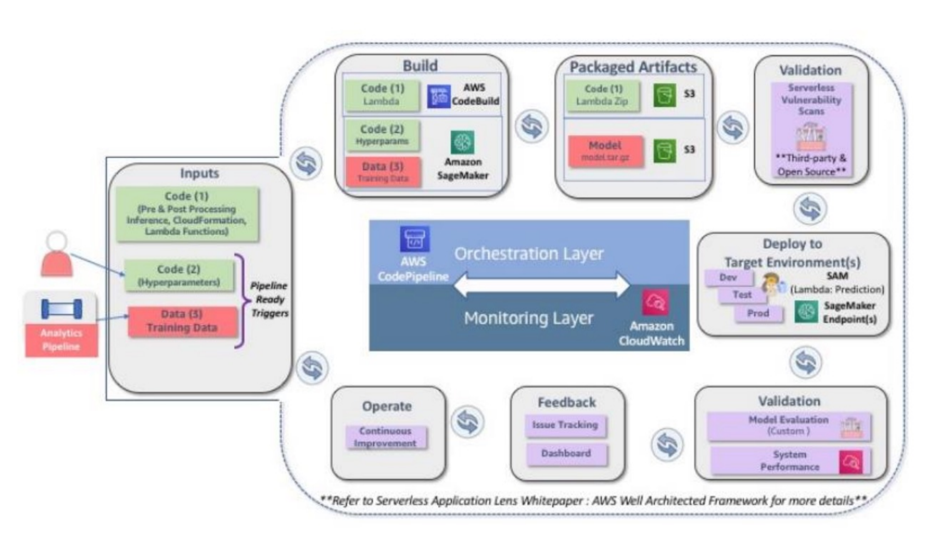
- Versioning
- Automation

AWS Step Functions

Operational Excellence Pillar



How have you automated the development and deployment pipeline for your ML workload?

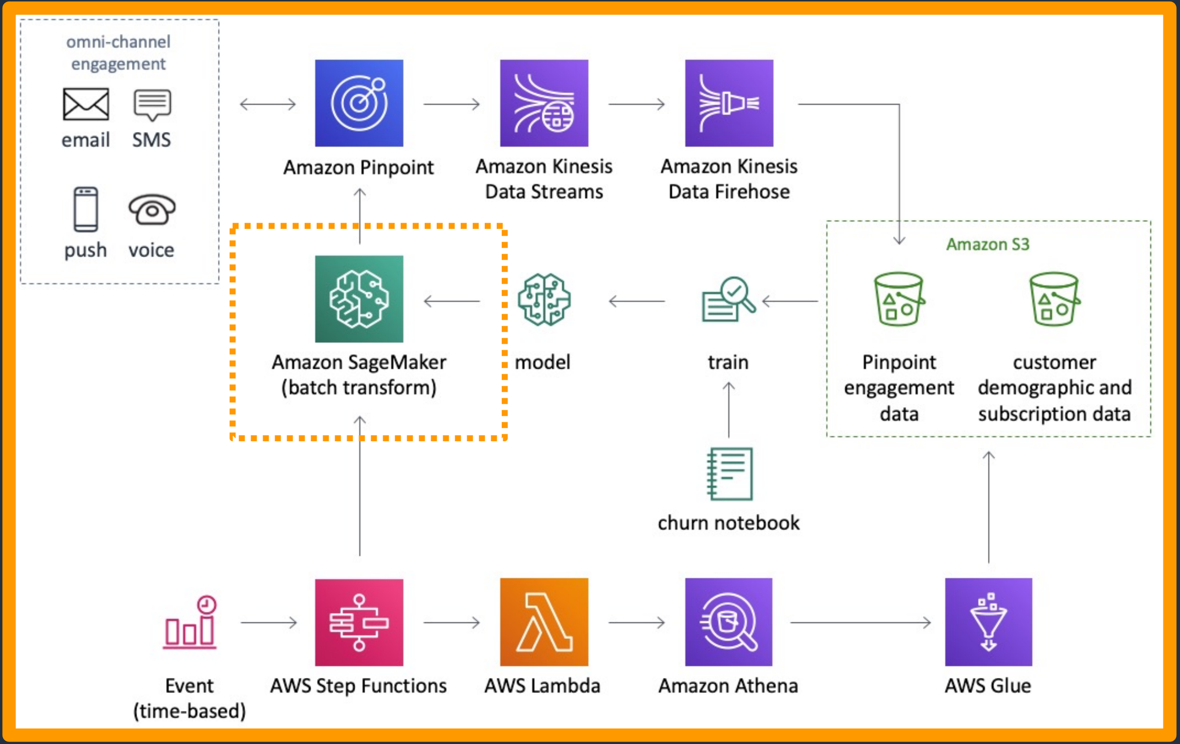


Machine Learning Lens

Reliability Pillar

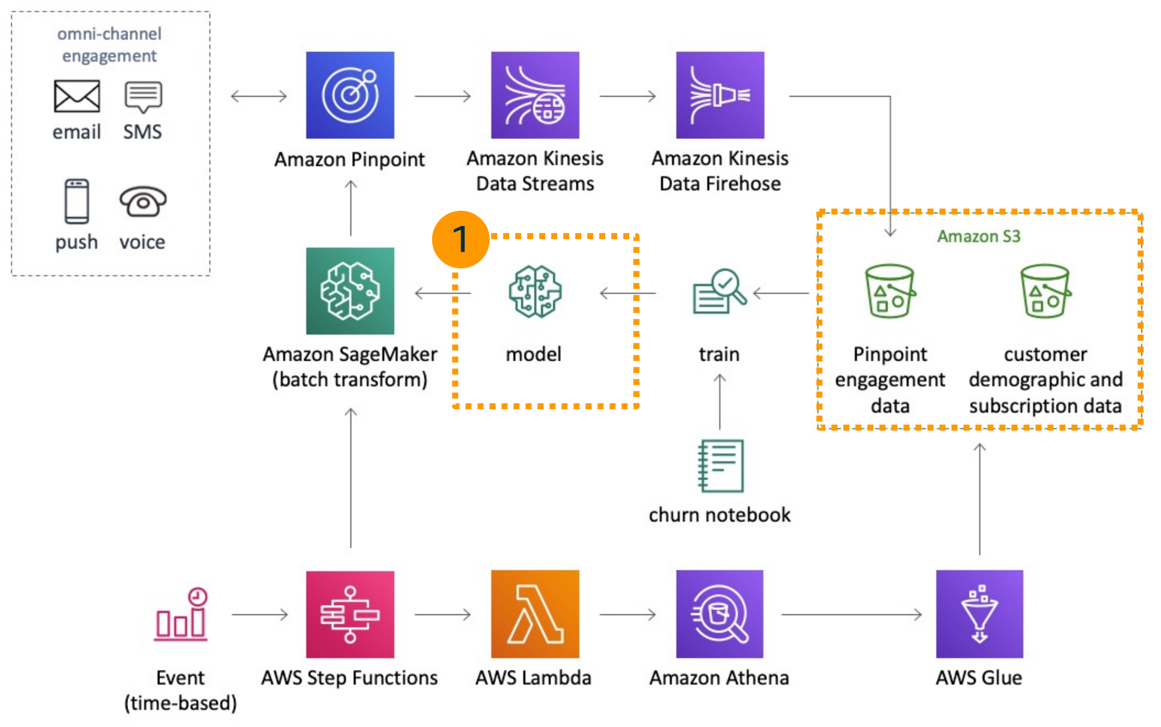
- Manage changes to model inputs through automation
- Train once and deploy across environments

Reliability Pillar



How are changes to ML models coordinated across your workload?

Reliability Pillar



How do you recover from failure or inadvertent loss of a trained ML model?

Considerations:

- Versioning
- Preventative Controls
- Automation/Orchestration

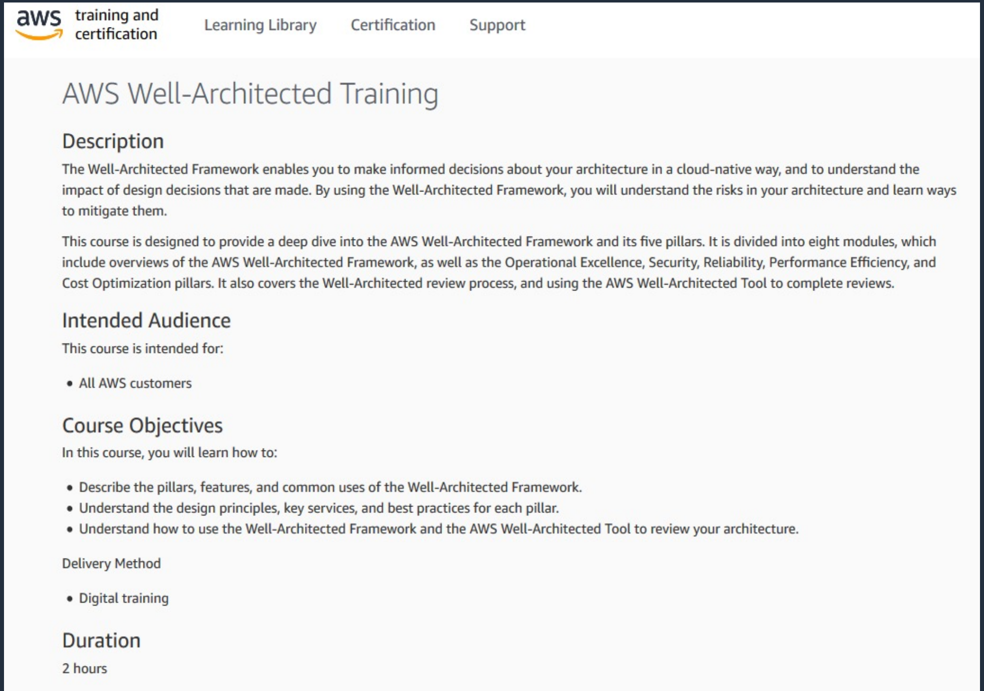
Resources

Training

https://aws.amazon.com/well-architected

AWS Well-Architected

- The Framework
- Operational Excellence
- Security
- Reliability
- Performance Efficiency
- Cost Optimization
- Well-Architected Review
- AWS Well-Architected Tool



AWS Well-Architected Training

Description

The Well-Architected Framework enables you to make informed decisions about your architecture in a cloud-native way, and to understand the impact of design decisions that are made. By using the Well-Architected Framework, you will understand the risks in your architecture and learn ways to mitigate them.

This course is designed to provide a deep dive into the AWS Well-Architected Framework and its five pillars. It is divided into eight modules, which include overviews of the AWS Well-Architected Framework, as well as the Operational Excellence, Security, Reliability, Performance Efficiency, and Cost Optimization pillars. It also covers the Well-Architected review process, and using the AWS Well-Architected Tool to complete reviews.

Intended Audience

This course is intended for:

- All AWS customers

Course Objectives

In this course, you will learn how to:

- Describe the pillars, features, and common uses of the Well-Architected Framework.
- Understand the design principles, key services, and best practices for each pillar.
- Understand how to use the Well-Architected Framework and the AWS Well-Architected Tool to review your architecture.

Delivery Method

- Digital training

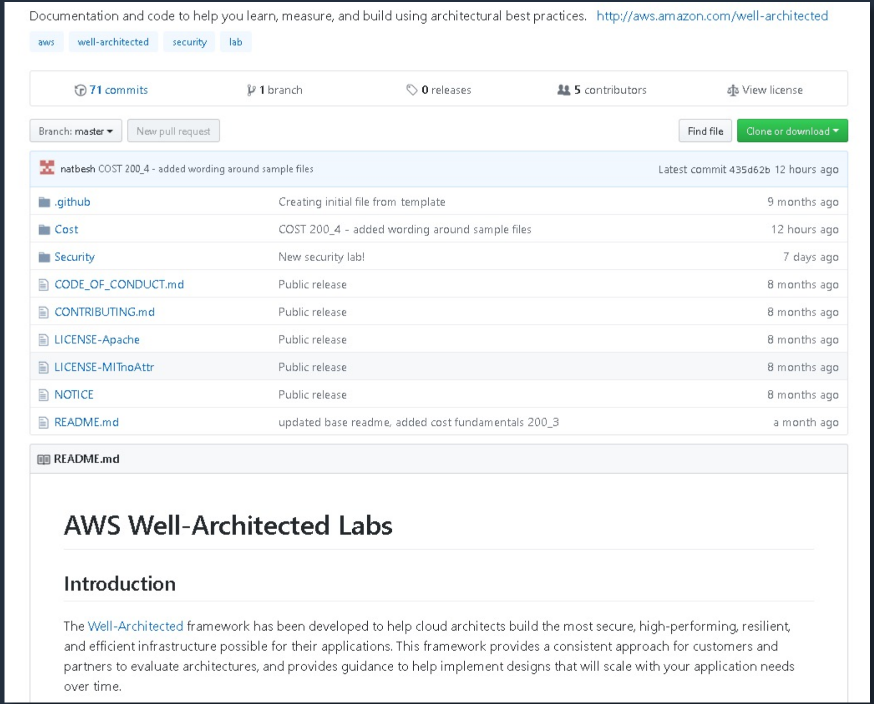
Duration

2 hours

General Well-Architected Labs

AWS Well-Architected

https://github.com/awslabs/aws-well-architected-labs



Documentation and code to help you learn, measure, and build using architectural best practices. http://aws.amazon.com/well-architected

natbesh COST 200_4 - added wording around sample files — Latest commit 435d62b 12 hours ago

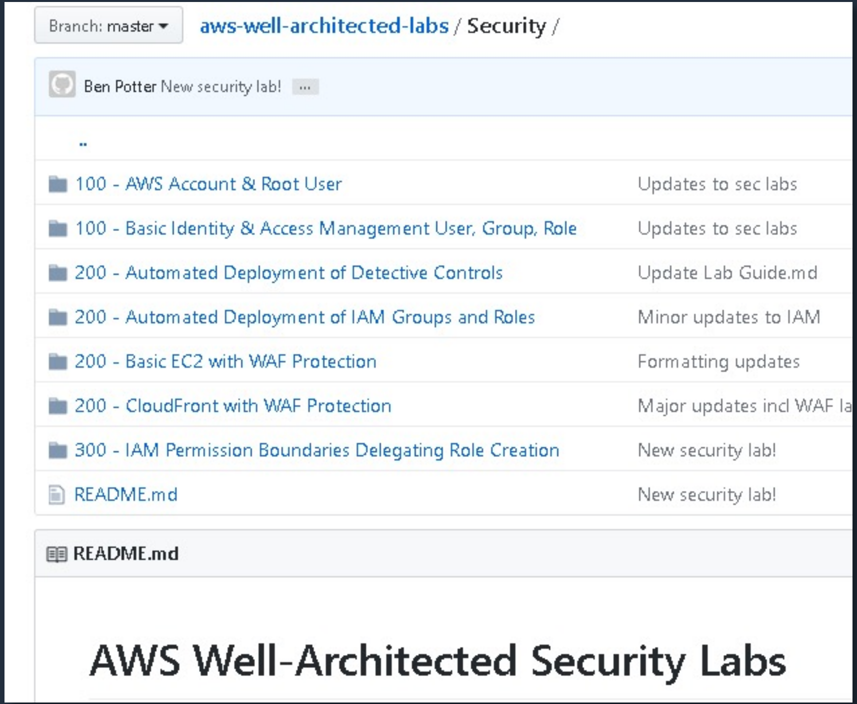
.github	Creating initial file from template	9 months ago
Cost	COST 200_4 - added wording around sample files	12 hours ago
Security	New security lab!	7 days ago
CODE_OF_CONDUCT.md	Public release	8 months ago
CONTRIBUTING.md	Public release	8 months ago
LICENSE-Apache	Public release	8 months ago
LICENSE-MITnoAttr	Public release	8 months ago
NOTICE	Public release	8 months ago
README.md	updated base readme, added cost fundamentals 200_3	a month ago

README.md

AWS Well-Architected Labs

Introduction

The Well-Architected framework has been developed to help cloud architects build the most secure, high-performing, resilient, and efficient infrastructure possible for their applications. This framework provides a consistent approach for customers and partners to evaluate architectures, and provides guidance to help implement designs that will scale with your application needs over time.



aws-well-architected-labs / Security /

Ben Potter New security lab!

..	
100 - AWS Account & Root User	Updates to sec labs
100 - Basic Identity & Access Management User, Group, Role	Updates to sec labs
200 - Automated Deployment of Detective Controls	Update Lab Guide.md
200 - Automated Deployment of IAM Groups and Roles	Minor updates to IAM
200 - Basic EC2 with WAF Protection	Formatting updates
200 - CloudFront with WAF Protection	Major updates incl WAF la
300 - IAM Permission Boundaries Delegating Role Creation	New security lab!
README.md	New security lab!

README.md

AWS Well-Architected Security Labs

aws

General Well-Architected Labs

AWS Well-Architected

https://www.wellarchitectedlabs.com/

Thank you!

Sireesha Muppala (smuppala@amazon.com)
Shelbee Eigenbrode (shelbees@amazon.com)
Christian Williams (wnchris@amazon.com)