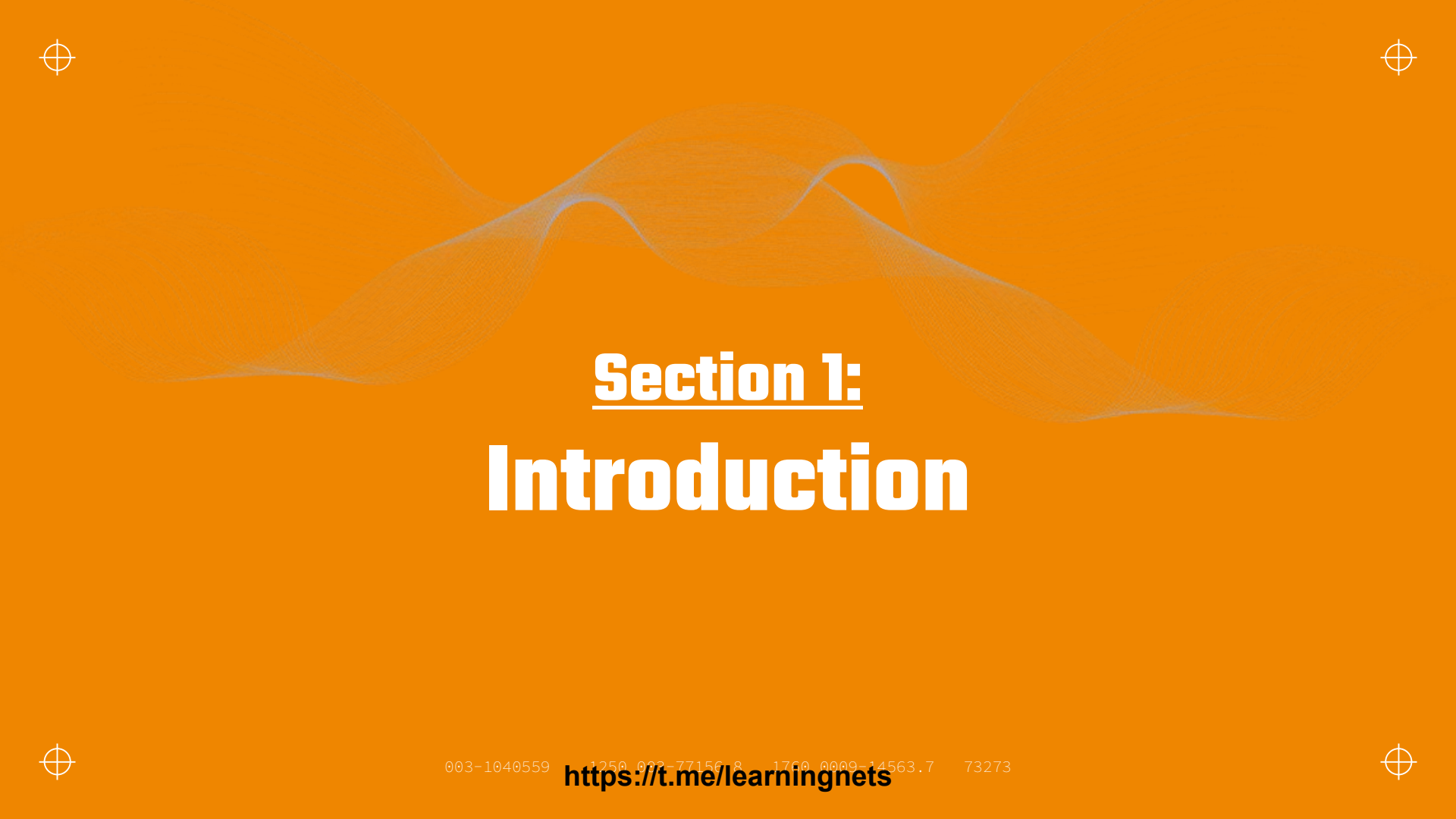




Section 1: Introduction

003-1040559 1250 002-7715618 1750 0009-14563.7 73273

<https://t.me/learningnets>

INTRODUCTION

Section 1:

About the exam & course setup



About the Machine Learning Engineer ASSOCIATE Certification



Why getting certified?

- ✓ Impactful way to advance career
- ✓ Positioning as an expert
- ✓ Future proof + great job opportunities.

What is covered?

- ✓ AWS Certified Machine Learning Engineer Associate
- ✓ <https://aws.amazon.com/certification/certified-machine-learning-engineer-associate/>

Demos

- ✓ Not needed for the exam.
- ✓ Help with memorizing.
- ✓ Give you practical foundation.

Goal

- ✓ Clear exam with ease.
- ✓ Knowledge for working with AWS

Passing Score

- ✓ 720 / 1000
- ✓ Goal: Achieve a score of 850+



Master the Exam

Free Trial Account

Not needed for the exam.
Help with memorizing
Give you a practical knowledge.

Exam Overview

<https://aws.amazon.com/certification/certified-data-engineer-associate>

Exam Duration

☐ Time: 170min

Exam Questions

☐ 85 questions ☐ Multiple Select, Multiple Choice

Scenario-based questions – find the best solution

A data scientist is tasked with predicting house prices based on features like the number of rooms, size of the house, and location. They decide to use a supervised learning model. Which of the following models would be most appropriate for this task?

- ☐ K-means clustering
- ☒ Linear regression
- ☐ Principal component analysis (PCA)
- ☐ Random Cut Forest (RCF)



Recipe to clear the exam



Lectures

Step-by-step incl. Demos
~ 30-60 min / day

Quizzes

Practice and test your knowledge

Slides

Repeat and go through important points

Practice Test

Evaluate knowledge & weaknesses
Eliminate weaknesses

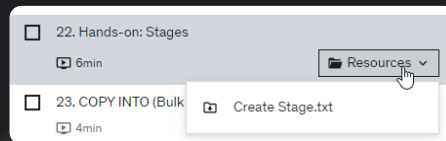
Book Exam

Confident & Prepared

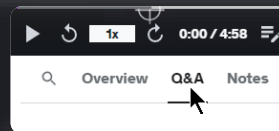
<https://t.me/learningnets>

Final Tips

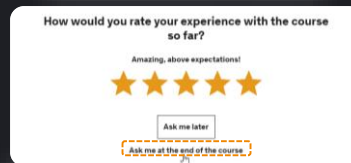
Resources



Q&A Section



Reviews



Connect & Congratulate



<https://t.me/learningnets>

Section 2:

SageMaker: Basics & Setup



Amazon SageMaker

003-1040559 1250 002-77156.8 1760 0009-14563.7 73273

<https://t.me/learningnets>

Overview of Amazon SageMaker



Amazon SageMaker is a fully managed service that brings together a broad set of tools to enable high-performance, low-cost machine learning (ML) for any use case.

Data Preparation

Model Building

Training

Hyperparameter tuning

Deployment



Studio



Notebooks



Console



Ground truth

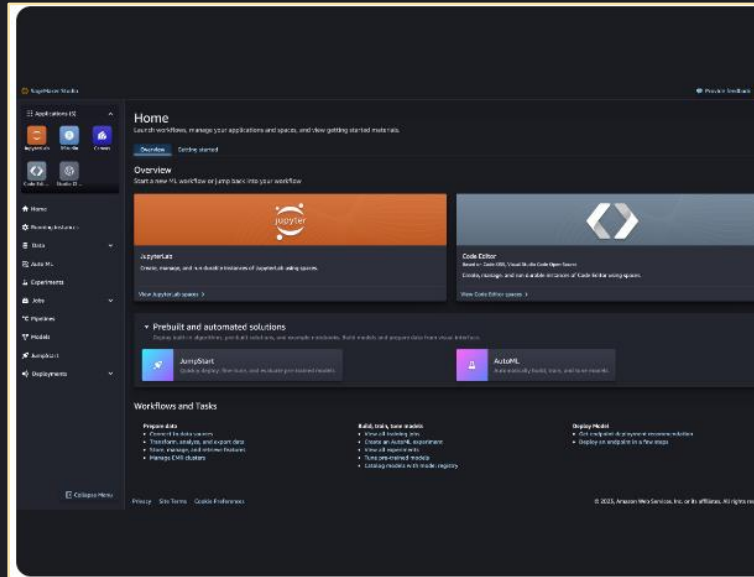
<https://t.me/learningnets>

Amazon SageMaker Studio



Key capabilities:

- ❑ Supports various integrated development environments
- ❑ Ability to scale underlying compute resources as needed
- ❑ Access from any device without the need to download sensitive ML artifacts to your local machine



SageMaker Notebooks



❑ Supports comprehensive ML tasks.

- Preparing petabyte scale data
- Training
- Debugging
- Deploying Models

❑ Build unified analytics and ML workflows

- Interactive Spark jobs
- Monitor and debug jobs

❑ Contains popular ML packages

- Deep learning framework
- Widely-used Python packages

❑ Scale your compute resources

- Compute-optimized
- GPU-accelerated instances

Amazon SageMaker Console

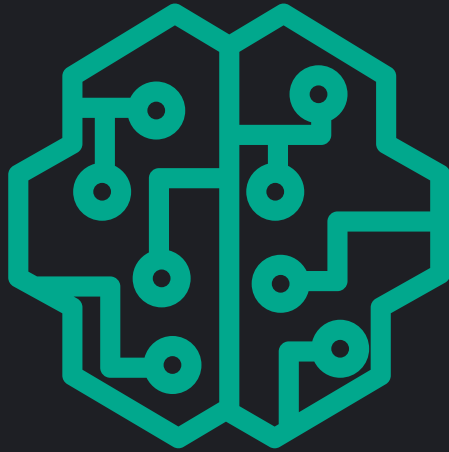


☐ Code

☐ Build

☐ Manage

SageMaker's Role in the ML Lifecycle



- ❖ A graphical user interface (GUI)-based environment
- ❖ Integrating and managing algorithms
- ❖ High-performing MLOps tools
- ❖ Pre-trained models

Conclusion



- ❑ Robust and scalable environment for managing the entire ML lifecycle.



- ❑ Streamline your ML development and deployment processes.



An Overview of Amazon SageMaker Notebooks and Their Role in the ML Workflow

<https://t.me/learningnets>



01

02

03

04

05

06

SageMaker Notebooks

Introduction:

What are Amazon SageMaker Notebooks?

- Fully managed Jupyter notebooks for ML tasks.
- Amazon SageMaker manages all infrastructure for Jupyter notebooks, including servers, environment setup, and software installation.
- Provides an interactive environment to write and execute code, visualize data, and develop ML models.

SageMaker Notebooks

Key Features:

On-Demand Instances

⇒ Launch notebook instances on-demand with various instance types

Persistent Storage

⇒ Automatic saving of notebooks and data.

Pre-installed Libraries

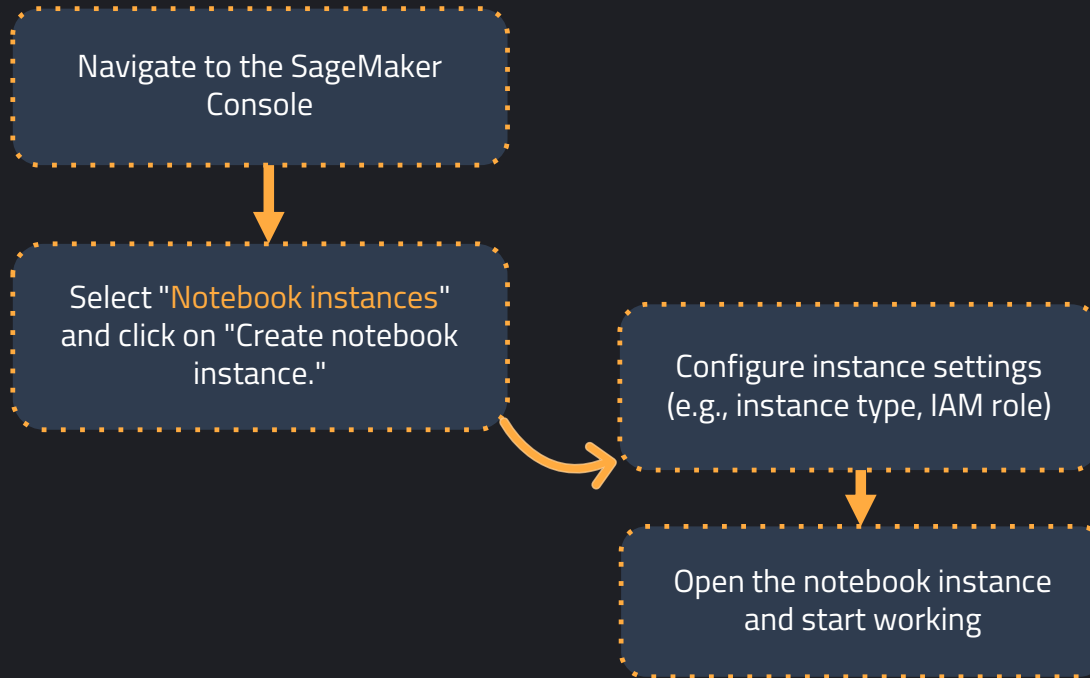
⇒ Popular ML like TensorFlow, PyTorch, Scikit-learn.

Built-In Collaboration

⇒ Share notebooks for easy collaboration.

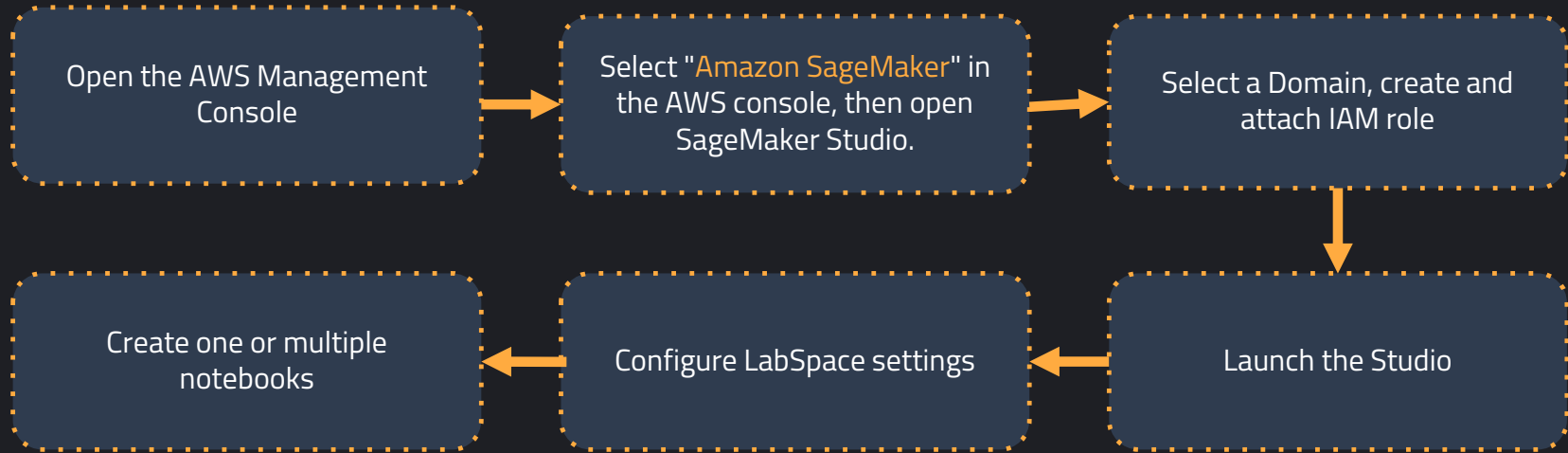
Setting Up Notebooks

Using SageMaker Instances:



Setting Up Notebooks

Using SageMaker Studio:



SageMaker Notebook Instances

- SageMaker Notebook Instances run independently with a chosen instance type, operating in standalone mode.
- SageMaker Notebook Instances are fixed and independent

VS

SageMaker Studio Notebooks

- SageMaker Studio Notebooks allow managing multiple notebooks, switching tasks, and accessing multiple SageMaker services from a single interface.
- **Studio Notebooks** are more integrated



SageMaker Notebooks

Use in the ML Workflow:

Data Preparation

Import and preprocess datasets using built-in tools and libraries.

Model Building

Write and experiment with ML models using Python libraries.

Training and Tuning

Use SageMaker's distributed training capabilities.

Deployment

Deploy models for real-time inference or batch processing.

Benefits of Using SageMaker Notebooks

Scalability

- Automatically adjust compute resources.



Cost-

Effectiveness

- Pay only for what you use.



Ease of Use

- Simple, user-friendly interface with minimal setup.



Integration

- Seamless integration with other AWS services.



Section 3:

SageMaker: Data Ingestion & Feature Engineering



Data Preparation With SageMaker Data Wrangler

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Amazon SageMaker – Data Wrangler

- It **simplifies data preparation process** through its user-friendly interface.

Capabilities

Data Import

- ⇒ Data Sources: **S3, Redshift, EMR, Feature Store, Athena, JDBC...**
- ⇒ Direct data integration with Amazon SageMaker.

Data Preparation

- ⇒ **Pre-built data transformations** like one-hot encoding, normalization...
- ⇒ **Custom transformations** using Pandas, PySpark.
- ⇒ **Feature engineering** such as feature scaling, binning...

Amazon SageMaker – Data Wrangler

Capabilities

Data Exploration
&
Visualization

- ⇒ Offers **pre-configured templates**: Histograms, line plots, bar charts...
- ⇒ **Data quality insights**: Provides summary statistics.

Data Flow &
Transformation
Pipelines

- ⇒ **Interactive data flow**: Build visual data processing pipelines.
- ⇒ **Track transformations**: Every transformation action logged.
- ⇒ **Custom data flows**: You can export pipelines as Python scripts.

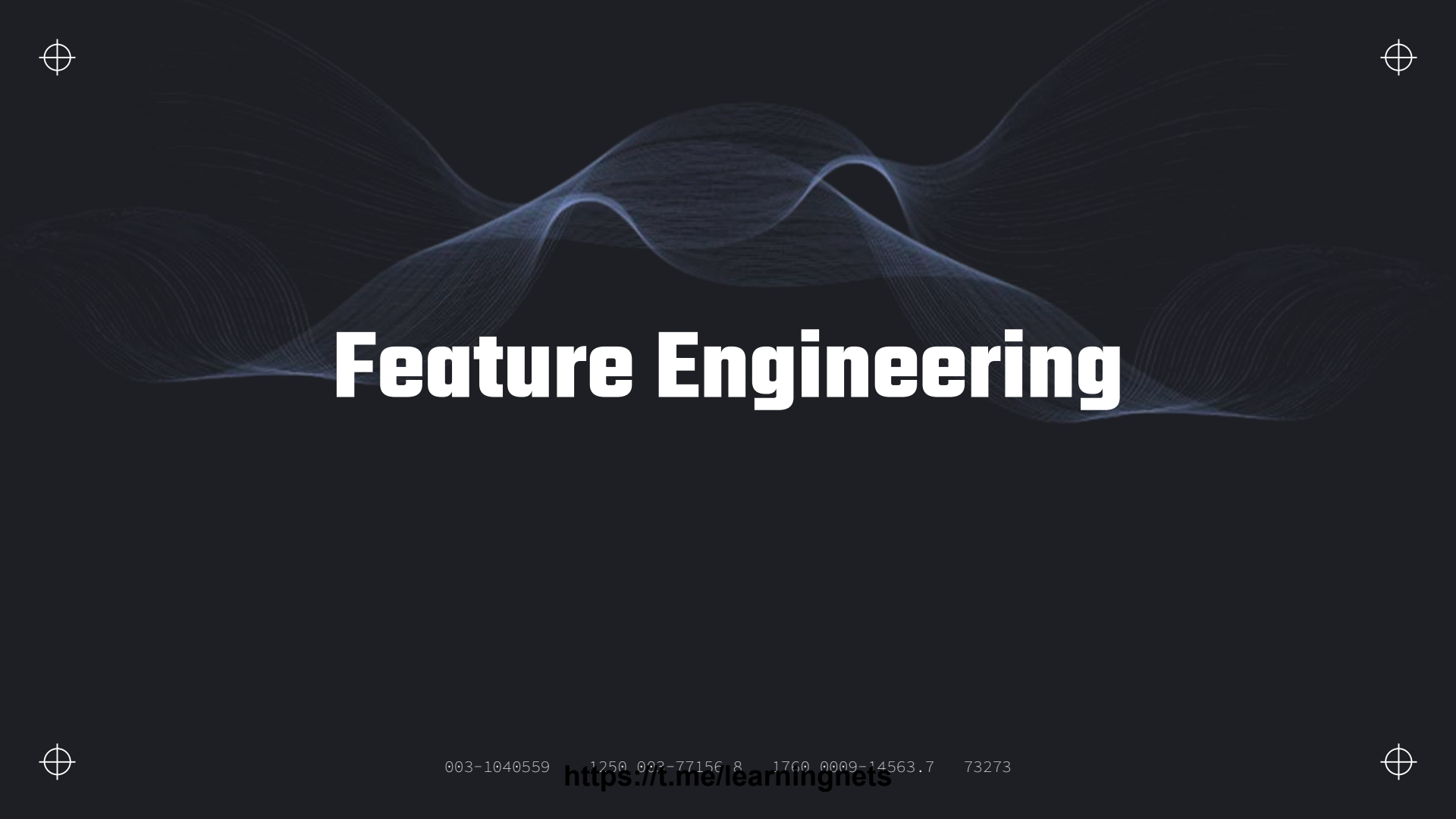
Data Export
&
Scalability

- ⇒ Export data into **Amazon S3, SageMaker Feature Store...**
- ⇒ **Batch** and **real-time** transformations.

Amazon SageMaker – Data Wrangler

Quick Model

- You can quickly test your data.
- Automatically trains/tests the data and provides you insights:
 - Model summary
 - Feature summary
 - Confusion matrix



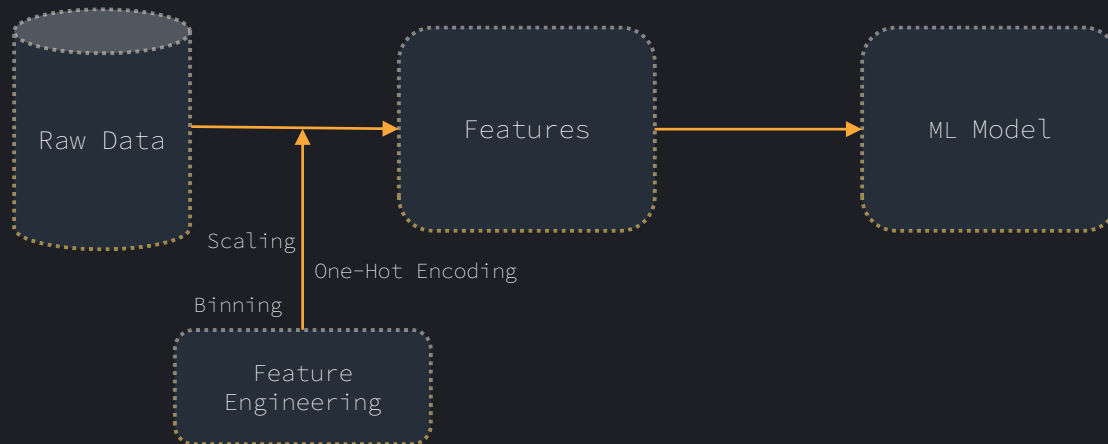
Feature Engineering

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Feature Engineering

- **Features** are individual measurable properties of the data
- Feature Engineering is process of **transforming raw data** into **meaningful features**.
- Organize the data to extract the maximum value.



Feature Engineering

Handling Missing Values

- ⇒ Some data can be missing and should be filled properly.
- ⇒ Filling missing values with **mean, median, mode**.
- ⇒ **Interpolation**.

Scaling & Normalization

- ⇒ Data can be in different scale or unit.
- ⇒ **Min-Max scaling**.
- ⇒ **Standardization**.

Encoding Categorical Variables

- ⇒ Convert categorical features to numerical form.
- ⇒ **One-Hot Encoding**.
- ⇒ **Label Encoding**.

Feature Creation & Transformation

- ⇒ Combine existing features to get more.
- ⇒ Apply transformations to current features like log, square root.

Feature Selection

- ⇒ Select the important features.
- ⇒ **Recursive Feature Elimination (RFE)**.



Amazon SageMaker Feature Store

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Amazon SageMaker – Feature Store

- Simplifies creating, sharing and managing data features.
- Acts as a centralized repository.
- Streamline and standardize feature engineering workflows.
- Query features using Amazon Athena or other SQL-based tools.

Online and Offline Store

- Online store
⇒ Real-time apps, low latency

- Offline store
⇒ Historical data analysis.
⇒ Data stored in Amazon S3.

Amazon SageMaker – Feature Store

- Data can be ingested through from many sources.
 - Clickstreams, service logs, sensors, EMR, Glue, Kinesis, Kafka, Lambda, Athena...
- Provides **feature versioning**.
- Features can be grouped within **Feature Groups**.

Id	Time	Feature1	Feature2

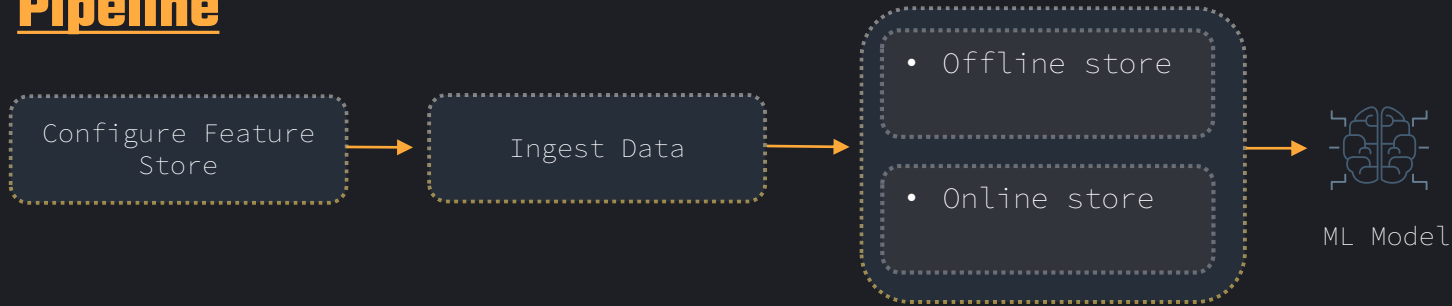
Group 1

Id	Time	Feature1	Feature2

Group 2

Amazon SageMaker – Feature Store

Pipeline



Lineage Tracking

- ⇒ Provides built-in lineage tracking with **SageMaker Lineage**.
- ⇒ Track pipeline executions and visualize data flow in **SageMaker Studio**.

Point-In-Time Query

- ⇒ Retrieve feature values **at a specific point in the past**.
- ⇒ Used for **historical data**; financial records, medical informations...



Amazon SageMaker Ground Truth

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

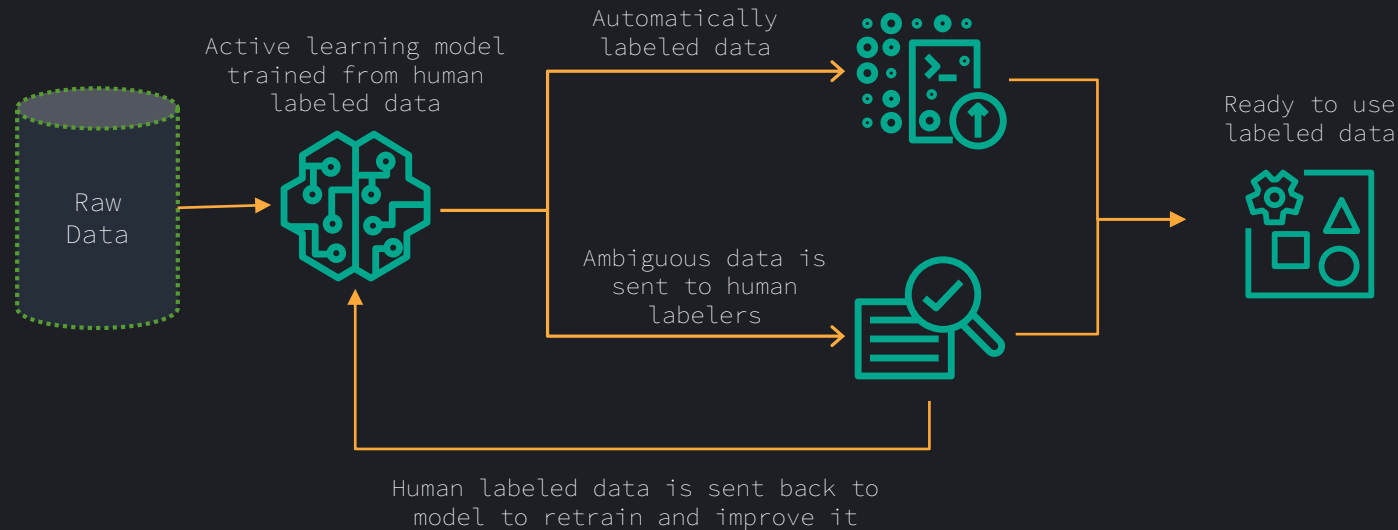
<https://t.me/learningnets>

Amazon SageMaker – Ground Truth

- Provides tools and workflows for **labeling data**.
- Labeling can be done **manually** or **semi-automatically**.
- **User-friendly interface** for labeling tasks
 - Image and video, text, 3D Point Cloud.

Amazon SageMaker – Ground Truth

Active Learning



Amazon SageMaker – Ground Truth

Human-in-the-loop

- ⇒ Data generation, annotation, model review and evaluation
- ⇒ Fine tune models by leveraging human feedback

Automating human-in-the-loop tasks

- ⇒ Automation to accelerate data generation, annotation and model review.

Human workforce options

- ⇒ Amazon Mechanical Turk.
- ⇒ Private workforce.
- ⇒ Third-party vendors.

Integration

- ⇒ Amazon SageMaker, Amazon S3, AWS Lambda...

Amazon SageMaker – Ground Truth

Use Cases

Evaluating and Red Teaming

- ⇒ Detect vulnerabilities & biases
- ⇒ Reduce bias and toxicity in predictions

Comparison and Ranking Data

- ⇒ Rank and classify model responses.
- ⇒ Improved fine-tuning opportunities

Training FMs with human-generated data

- ⇒ Text summarization, Q&A, generating captions can be used to train FMs

Scalability

Accuracy

Cost-Efficiency

Section 4: SageMaker: Training & Hyperparameter Tuning

003-1040559 1250.000-77156.8 1700.0009-14563.7 73273

<https://t.me/learningnets>



SageMaker JumpStart

003-1040559 1250 032-77156.8 1760 0009-14563.7 73273

<https://t.me/learningnets>

Amazon SageMaker Jumpstart



Access `FM Models`, `built-in algorithms`, and `pre-built ML Solutions`

⇒ Reduces time and effort for model development

⇒ Supports popular frameworks like TensorFlow, PyTorch

Features



Pre-trained
Models



Pre-built
Solutions



Customization

<https://t.me/learningnets>

Amazon SageMaker Jumpstart



Pre-Trained Models

Foundation Models

⇒ Large AI models pre-trained on vast amounts of data

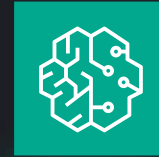
Text generation
Models

⇒ Text analysis and classification models

Image generation
Models

⇒ Image classification and segmentation models

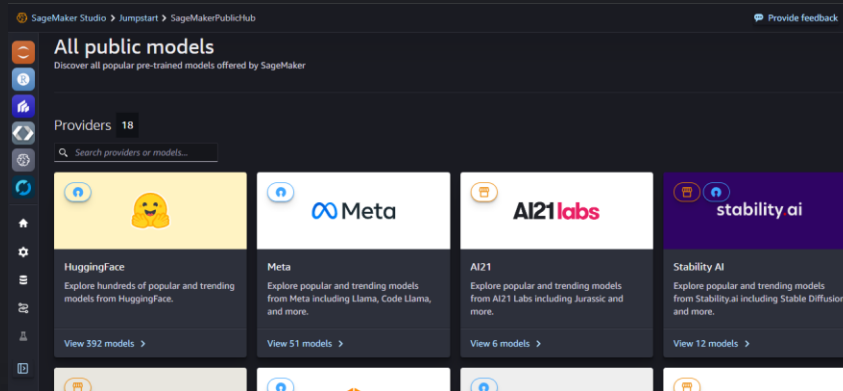
Amazon SageMaker Jumpstart



Deploying Pre-Trained Models

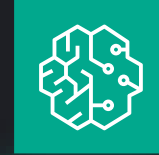
Deployment Options

- SageMaker Studio
 - ⇒ Deploy models directly from the SageMaker Studio Interface(UI)



<https://t.me/learningnets>

Amazon SageMaker Jumpstart



Deploying Pre-Trained Models

Deployment Options

- SageMaker Python SDK
 - ⇒ Deploy models programmatically using the SageMaker Python SDK(API)

```
[ ]: import sagemaker
from sagemaker.xgboost import XGBoost
from sagemaker import get_execution_role

# Set up the SageMaker session
sagemaker_session = sagemaker.Session()

# Get the role ARN
role = get_execution_role()

# Specify sample data locations
train_data = 's3://sagemaker-example-data/train.csv'
validation_data = 's3://sagemaker-example-data/validation.csv'

# Set up the XGBoost estimator
xgb_model = XGBoost(
    entry_point='xgboost_script.py',
    framework_version='1.5-1',
    hyperparameters={
        'max_depth': 5,
        'eta': 0.2,
        'objective': 'binary:logistic',
        'num_round': 100
    },
    role=role,
    instance_count=1
```

<https://t.me/learningnets>

Amazon SageMaker Jumpstart

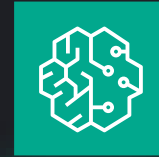


Customizing Pre-Trained Models

Customizing Options

- Prompt Engineering
 - ⇒ Craft the input prompts used to fine-tune pre-trained models
- Fine-Tuning
 - ⇒ Pre-built models can be fine-tuned on custom datasets to adapt them to specific use cases
- Hyperparameter Tuning
 - ⇒ Adjust hyperparameters and retrain models

Amazon SageMaker Jumpstart

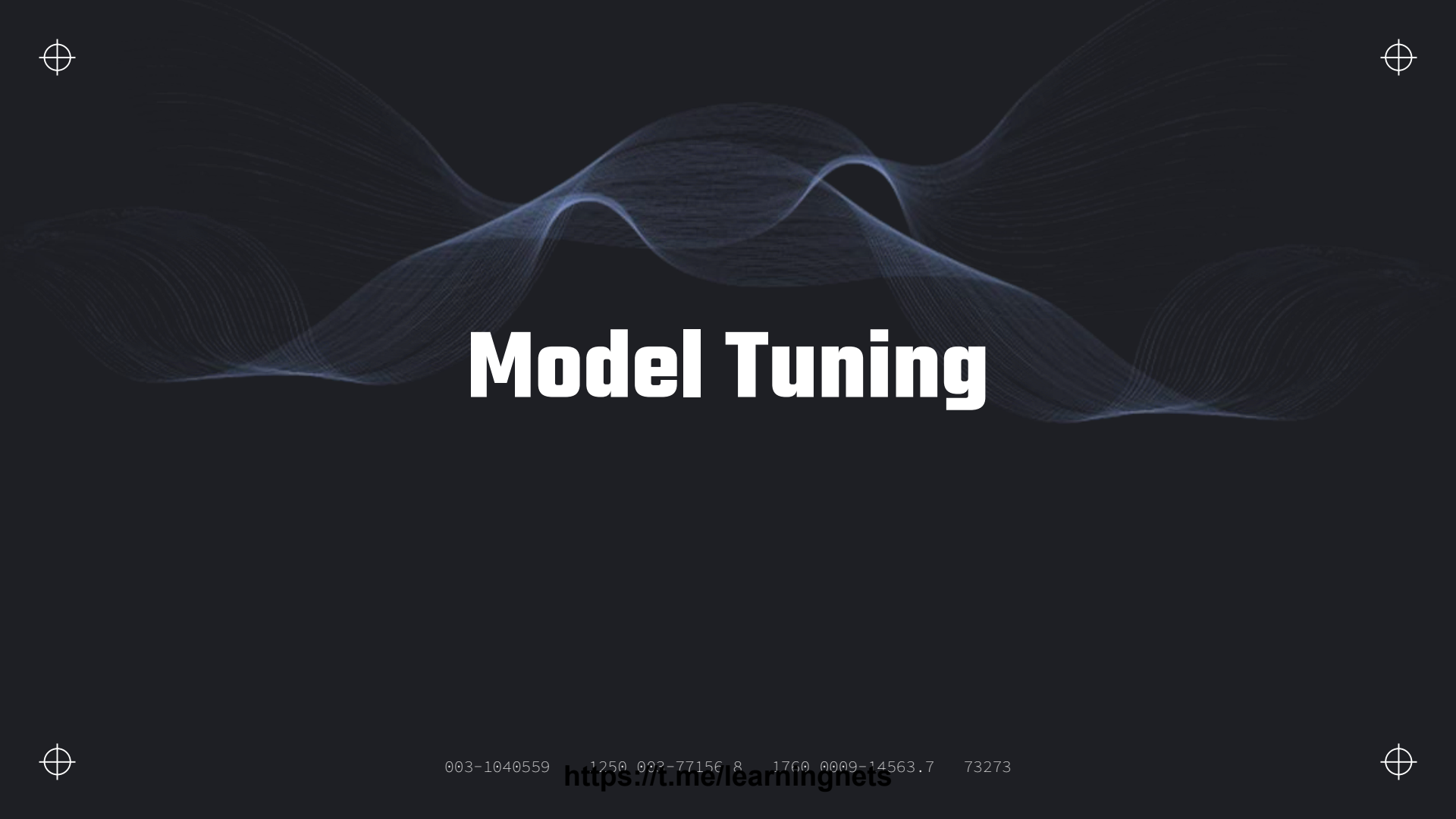


Pre-built Solutions

- End-to-End `pre-configured solutions` for different ML use cases
⇒ E.g. solutions for fraud detection, demand forecasting, etc.

Solution Components

- Pre-built Models
- Data Processing Pipelines
- Deployment Configurations



Model Tuning

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Model Tuning



Hyperparameters

- `External configurations` that `control` how a ML model learns
 - ⇒ Must be set before training begins

Hyperparameters Tuning

- Tune hyperparameters in your algorithm

Importance of Hyperparameter Tuning

Improves Model Performance

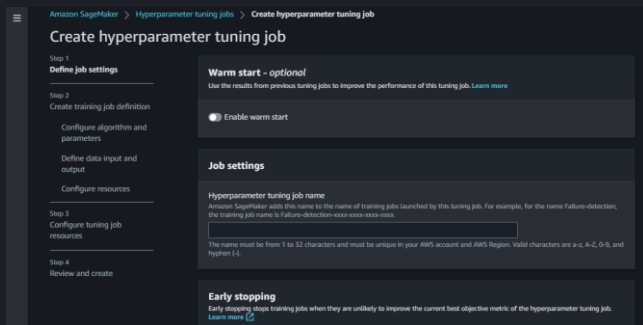
Minimizes overfitting/Underfitting

Accelerates Model development

Model Tuning

SageMaker Automatic Model Tuning

- Automates the process of hyperparameter tuning for ML models
- Finds the best version of a model
- Runs multiple training jobs with different hyperparameter combinations



How it works

- Uses algorithm and hyperparameter ranges you specify
- Chooses hyperparameter combinations that create the best model
- Performance measured by your chosen metric

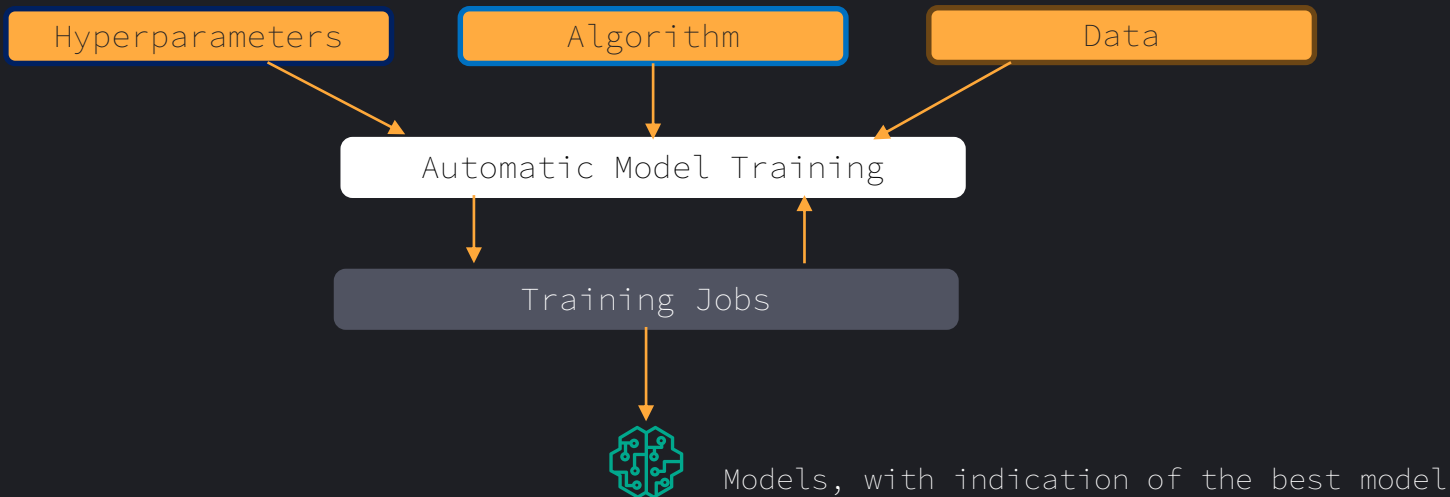
Cost Optimization

- Amazon EC2 Spot instances for training jobs

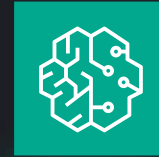
<https://t.me/learningnets>

Model Tuning

SageMaker Automatic Model Tuning



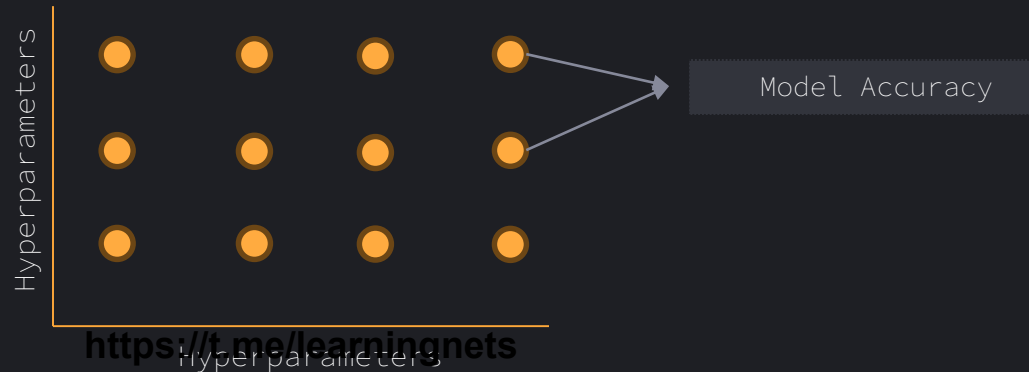
Amazon SageMaker



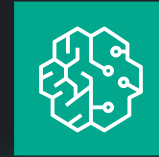
Optimization Techniques

Grid Search

- Tests **all** possible combinations of specified hyperparameter values
- Creates a **'grid'** of possibilities
- Thorough but can be computationally **intensive**



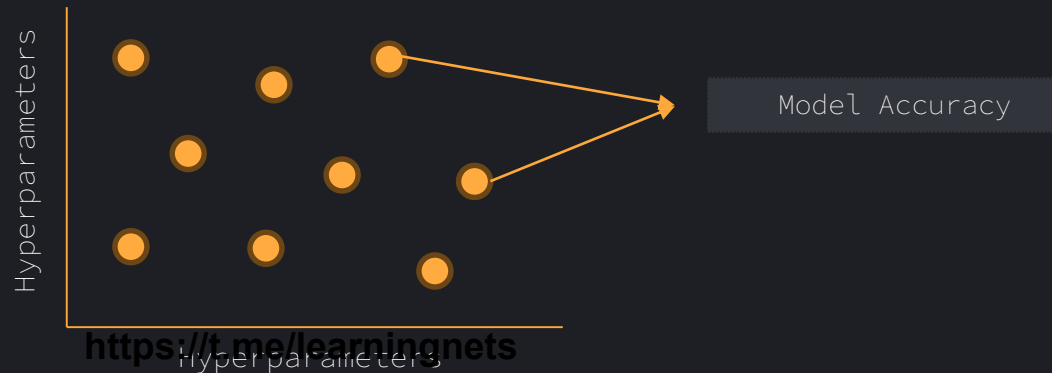
Amazon SageMaker



Optimization Techniques

Random Search

- Randomly samples set of hyperparameters within a specified range
- Doesn't depend on results of previous training jobs
- Faster and less resource-intensive than grid search



Amazon SageMaker



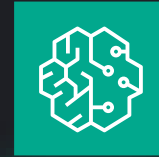
Optimization Techniques

Bayesian Optimization

- Uses prior evaluation results to inform future hyperparameter choices
- **Reduces** the number of evaluations needed to find optimal hyperparameters
- More **efficient** than **random or grid search** by focusing on promising areas



Amazon SageMaker



Optimization Techniques

Hyperband

- Allocates a fixed resources for hyperparameter configurations
- Automatically stops underperforming configurations
- Balances exploration and exploitation for efficient tuning



Using Script Mode

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Script Mode



Script Mode in SageMaker

- Write and manage custom training and inference code
- Supports prebuilt frameworks e.g. TensorFlow, and PyTorch
- Removes the complexities of creating custom Docker containers

```
import tensorflow as tf
from tensorflow.keras.datasets import cifar10
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense

# Load the CIFAR-10 dataset
(x_train, y_train), (x_test, y_test) = cifar10.load_data()

# Normalize the data
x_train = x_train / 255.0
x_test = x_test / 255.0

# Define the model architecture
model = Sequential([
    Conv2D(32, (3, 3), activation='relu', input_shape=(32, 32, 3)),
    MaxPooling2D((2, 2)),
    Flatten(),
    Dense(10, activation='softmax')
])

# Compile the model
model.compile(optimizer='adam', loss='sparse_categorical_crossentropy', metrics=['accuracy'])

# Train the model
model.fit(x_train, y_train, epochs=5, batch_size=32, validation_data=(x_test, y_test))
```

<https://t.me/learningnets>

Script Mode

Key Concepts of Script Mode

Pre-built Framework Containers

SageMaker provides **optimized containers** for popular frameworks

Entry Point Script

Custom training code written in **Python** and stored in a **.py** file

Customization

Developers can pass **custom code**, **dependencies**, and **libraries**

Modularization

Allows for **modularizing** training jobs, models, and inference steps

Dependency Management

Developers can include **custom libraries** or **dependencies** in the script



Script Mode



Process of Running Script Mode in SageMaker

- 1 — Define the Estimator
Create an estimator by specifying the entry point for the custom training script
- 2 — Train the Model
Using the fit() method, the model is trained
- 3 — Deploy the Model
After training, the trained model is deployed to a real-time inference endpoint

```
estimator = SKLearn(entry_point='train.py',  
                    source_dir='code',  
                    framework_version='0.23-1',  
                    instance_type='ml.c5.xlarge',  
                    hyperparameters=custom_hyperparameters,  
                    role=role)
```

```
estimator.fit(  
    inputs={'train': train_s3_uri, 'test': test_s3_uri})
```

```
predictor = estimator.deploy(  
    initial_instance_count=1,  
    instance_type='ml.m5.xlarge')  
  
predictor.predict(test_data)
```

Amazon SageMaker



Use Cases for Script Mode

Custom Algorithms

Developers can implement algorithms that aren't natively supported by SageMaker

Modular Code

Developers can modularize their code to keep the architecture clean

Bring Your Own Libraries

SageMaker allows custom libraries and external dependencies to be included

Amazon SageMaker

Example Workflow



1

Setup

Create an IAM role with S3 access, set up a SageMaker notebook instance



2

Write Code

Create Python scripts for training and inference, modularize your code



3

Train

Define a SageMaker Estimator with the necessary parameters



4

Deploy

Deploy the trained model using the `.deploy()` method

<https://t.me/learningnets>



Custom Docker Containers

003-1040559 1250.002-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Custom Docker Containers



Packages your code, libraries, dependencies, and even the operating system—into a single Docker image

How it works

- ⇒ Dockerfile that defines everything and how to execute your code.
- ⇒ Container must follow SageMaker's requirements
- ⇒ Once your Docker image is built, you push it to Amazon Elastic Container Registry (ECR)

Use Cases

- ⇒ To have full control or meet requirements that SageMaker doesn't support natively

The Trade-Off

- ⇒ You manage everything all the complexities



Distributed Training

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Distributed Training



- Train large models across multiple devices or nodes
- Improves efficiency and reduces training time

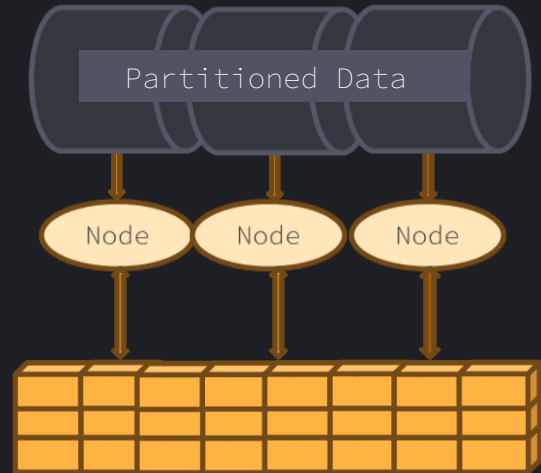


<https://t.me/learningnets>

Distributed Training

Data Parallelism

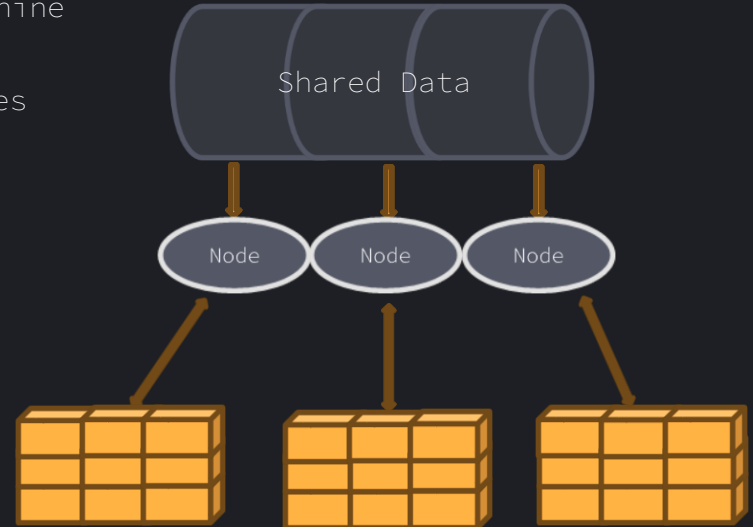
- Splits dataset across different machines(workers)
- Each worker trains the model on its piece of the data
- Best for Large datasets



Distributed Training

Model Parallelism

- Ideal for large models that can't fit on a single machine
- The Model itself is split the across different machines
- Each machine handles its portion of the model



Distributed Training

Data Parallelism in Script Mode



1

Define Job

Set the number of instances you want to use

2

Data Splitting

SageMaker partitions the dataset across workers

3

Gradient Sync

SageMaker synchronizes gradients across instances.

Defining instances using TensorFlow Scripts

```
estimator = TensorFlow(entry_point='train.py',  
                        instance_type='ml.p3.xlarge',  
                        instance_count=4,  
                        role=role,)
```

Amazon SageMaker



Model Parallelism in Script Mode

SageMaker's **Model Parallelism Library** automates model partitioning

The library helps you to:

- Automatically Splits your model/data into smaller parts
- Assign each part to a separate device or node
- Coordinate the communication between devices/nodes

```
estimator = PyTorch(entry_point='train.py',  
                    instance_type='ml.p4d.24xlarge',  
                    instance_count=2,  
                    distribution={'model_parallel': {'enabled': True}}))
```

<https://t.me/learningnets>

Distributed Training

Combining Data and Model Parallelism

Hybrid Parallelism configures both data and model parallelism

⇒ allows you to train massive models on vast datasets efficiently

Hybrid Parallelism Set Up Example

```
estimator = PyTorch(entry_point='train.py',  
                    instance_count=8,  
                    instance_type='ml.p4d.24xlarge',  
                    distribution={  
                        'model_parallel': {'enabled': True},  
                        'mpi': {'enabled': True}  
                    })
```

Here the model is split across GPUs, the data is split across multiple instances



Section 5: SageMaker: Experiment Tracking & Debugging

003-1040559 1250.000-77156.8 1750.0009-14563.7 73273

<https://t.me/learningnets>



Amazon SageMaker Experiments

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Amazon SageMaker – Experiments

- Organize, track and compare ML experiments.
- Capture the details of each run, compare and determine.

Experiments

⇒ Group of related model training sessions.

Trials

⇒ Each individual run within an experiment.
⇒ Tracks key performance metrics like accuracy, loss...

Trial Components

⇒ Stages of trials.
⇒ Includes tasks like data preprocessing, model training & evaluation.
⇒ SageMaker logs metadata for each component.

Amazon SageMaker – Experiments

Trackers

- ⇒ Automatically handles tracking.
- ⇒ Every input and output is logged.

Comparison

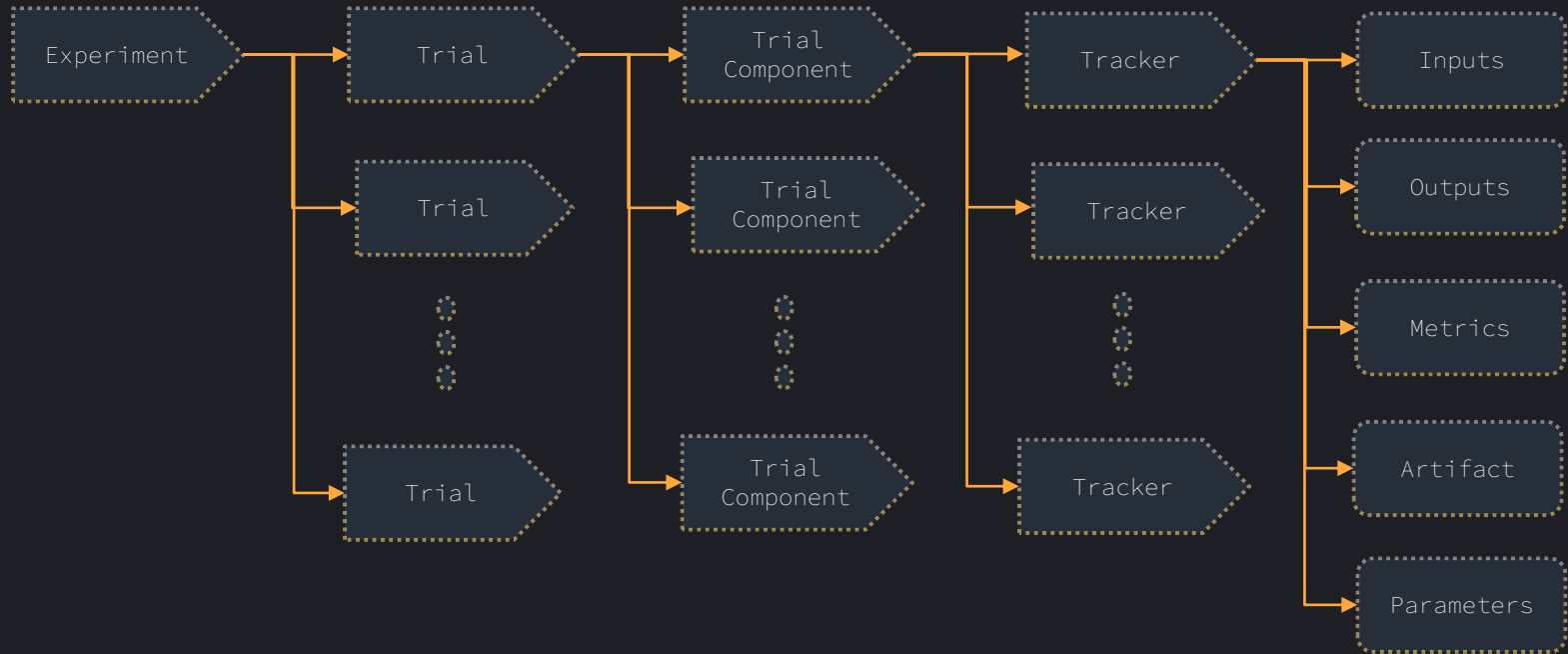
- ⇒ Provides **APIs** and **UI within SageMaker Studio** to compare key metrics.

Reproducibility

Version Control

Automation

Amazon SageMaker – Experiments





SageMaker Neo

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

What does SageMaker Neo do?

Optimizes
models

For efficient inference
in cloud and edge environments

Minimal
Accuracy Impact

Improve your model's performance
by up to 25 times—without losing accuracy.

Key Features of SageMaker Neo

Framework Compatibility

Supports frameworks like TensorFlow, PyTorch, MXNet, ONNX, and XGBoost

Hardware Support

Optimizes for hardware from Intel, NVIDIA, Arm, Qualcomm, etc

Compilation Jobs

Generates artifacts that can be deployed on specific hardware

Cross-platform Deployment

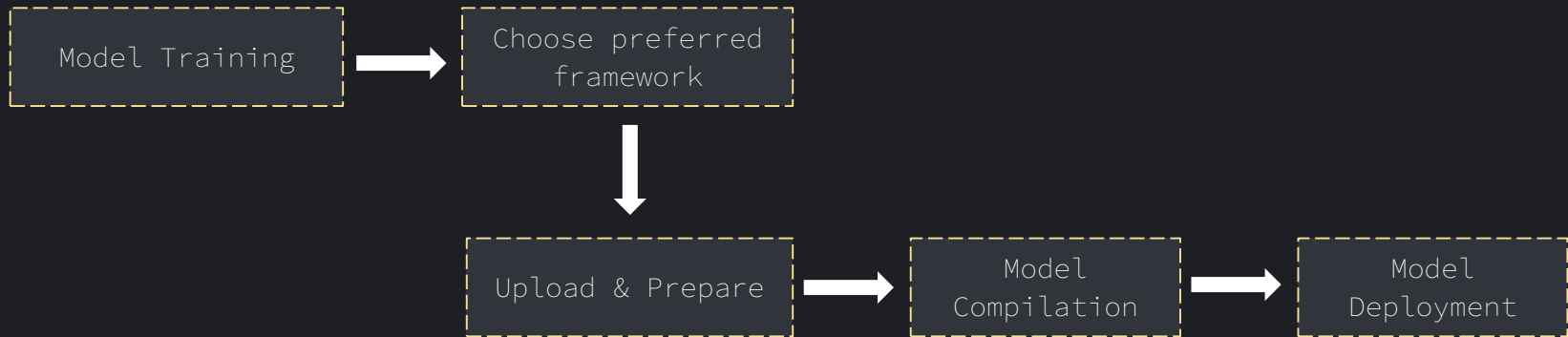
Compile once deploy to cloud or edge seamlessly

Preserving Accuracy

Optimize model's performance without impacting accuracy.

Overview of SageMaker Neo

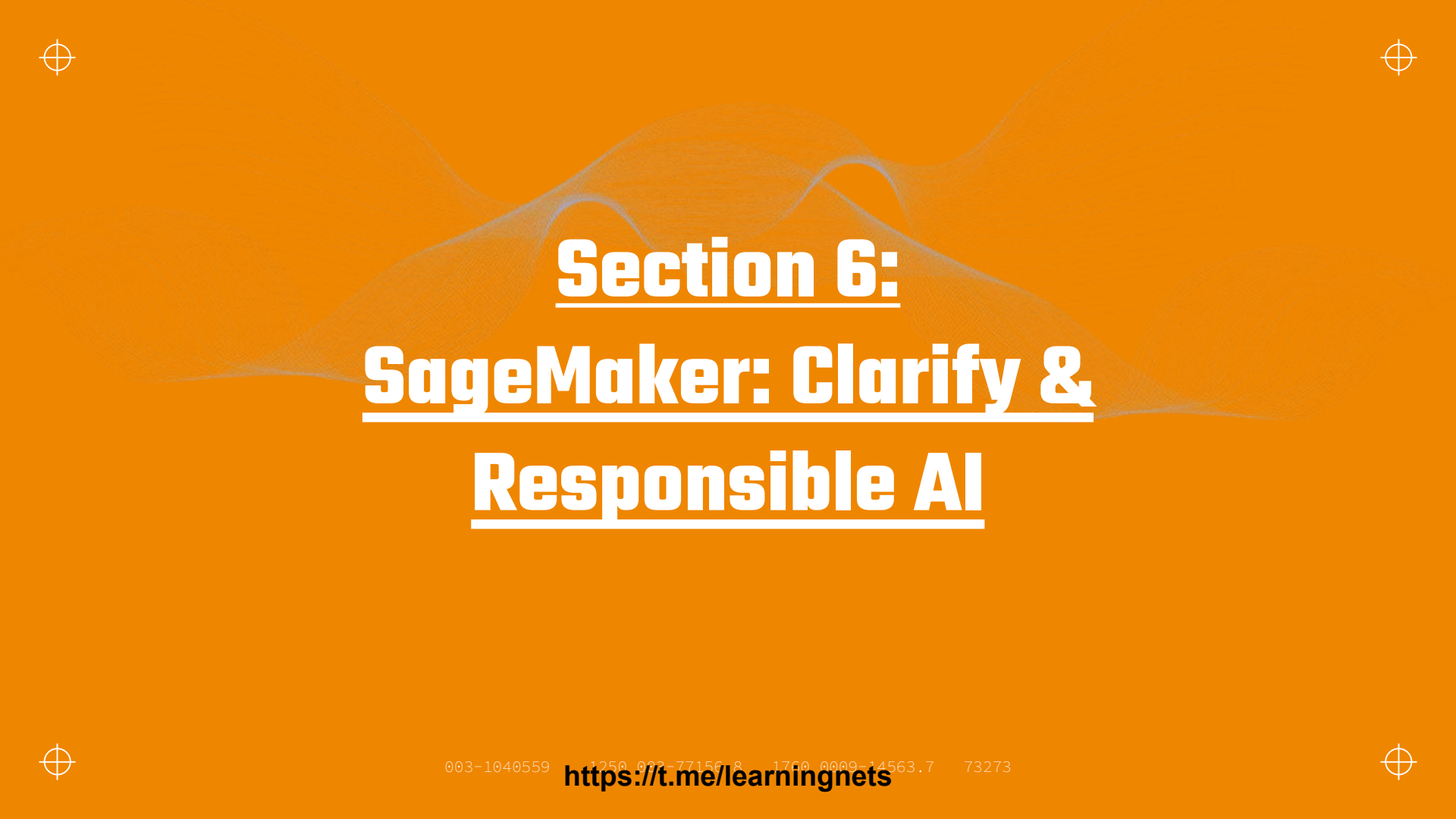
- ❑ A Deep Learning Model Compiler
- ❑ Hardware-Optimized Executable Format
- ❑ Simplified Deployment





Conclusion

- ❑ Neo is a compiler that makes your machine learning models run faster, across a variety of platforms, whether it's the cloud or edge devices.
- ❑ It saves you time, effort, and cost, and allows you to focus on what matters - delivering real value from your models.



Section 6: **SageMaker: Clarify &** **Responsible AI**



Amazon SageMaker Clarify

003-1040559 1250 032-77156.8 1760 0009-14563.7 73273

<https://t.me/learningnets>



Amazon SageMaker – Clarify

- Detect and mitigate bias in ML models.
- Addresses to fairness, accountability, transparency and explainability.
- Meet regulatory requirements with explainable, responsible and fair solutions.
- Integrates with other SageMaker tools for continuous fairness and transparency.

Amazon SageMaker – Clarify

Bias Detection

- Identify **bias** without needing scripts.
- **Automatically** run analyses against key features.
- Generates detailed **visual reports**.
- Offers **pre-training** and **post-training** analyze.
- Amazon SageMaker Data Wrangler integration:
 - **Random Undersampling**: Reduce the number of examples.
 - **Random Oversampling**: Increase the number of examples.
 - **SMOTE** (Synthetic Minority Over-sampling Technique): Generate new data by interpolating.

Amazon SageMaker – Clarify

Explainability

- Important for model debugging, regulatory compliance and trust building.

SHAP
(SHapley Additive
exPlanations)

- ⇒ **Feature attribution:** Calculates the contribution of each feature.
- ⇒ **Global and Local Explanations:** Individual and overall explanations.

SageMaker Experiment
Integration

- ⇒ Provide scores detailing for which feature contributed the most.

Aggregated Feature
Importance Charts

- ⇒ Outputs feature importance charts for tabular datasets.

Amazon SageMaker – Clarify

Transparency

- ⇒ Reports to meet regulatory and compliance.
- ⇒ Beneficial for healthcare, finance, insurance...

Foundation Model Evaluation

- ⇒ Evaluate foundational models.
- ⇒ Accuracy, robustness and toxicity.
- ⇒ Allows human-based evaluations.

Compliance Support

- ⇒ Detect potential bias and other risks.
- ⇒ Compliance programs aligned with ISO 42001.

Monitoring Deployed Models

- ⇒ Works within SageMaker Monitor.
- ⇒ Tracks and notifies users for changes in model behavior.



Section 7: **SageMaker: Debugging &** **Deployment**



003-1040559 1250 002-7715618 1750 0009-14563.7 73273

<https://t.me/learningnets>

SageMaker Debugger

Introduction:

- SageMaker Debugger enables real-time **monitoring** and **debugging** of machine learning models during training.

Alarm Notifications

Corrective Action

Visualization

- Automatically monitors training jobs and provides detailed performance reports.
- Collects and analyzes metrics in real-time.

SageMaker Debugger

Amazon SageMaker Debugger Features:



Capture Data from ML
Training Jobs

Apache MXNet
TensorFlow
PyTorch
XGBoost



Real-time monitoring

Debug and profile
data while training
is ongoing



Save time and cost

Find issues early, stop
training, fix issues and
accelerate prototyping

Setting Up SageMaker Debugger

Defining Debug
Rules

Choose from a library of built-in rules or create your own custom rules to match your training goals.

Configure the
Debugger Hook

Add the Debugger hook configuration to your training script.

Launch the
Training Job

Run your training normally, and Debugger will automatically capture and log metrics for analysis.

Monitor and
Analyze

Use SageMaker Studio or the SageMaker Console to monitor and analyze the captured tensors in real-time.



Identifying and Resolving Training Issues

- **Vanishing or Exploding Gradients.**
 - Debugger can spot these problems and suggest solutions like adjusting your learning rates, using gradient clipping, or even tweaking your model architecture.
- **Overfitting**
 - Debugger can monitor your validation metrics and prompt you to take action
- **Underfitting**
 - Debugger can detect this by monitoring loss curves, giving you insights that could lead to improvements in model complexity or data quality.

Leveraging **Debugger** During Model Training

- Benefits of using SageMaker Debugger during model training.

Real-Time Monitoring

- **Detects** and **addresses** issues early in the training process

Ensuring Model Quality

- **Monitors** key metrics during training, giving you real-time insights into your model's performance.

Continuous Improvement.

- **Track** and **analyze** your model's behavior during training, identify potential issues, and make adjustments.



SageMaker Debugger

Conclusion:

- Profiling is essential to improve training performance and reduce costs.
- SageMaker Debugger provides automated, scalable tools to monitor system and framework
- With visualization tools and best practices, you can efficiently optimize resource utilization during model training.



Model Deployment Strategies in SageMaker

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>



Deployment Strategies in SageMaker

Deploy machine-learning models to endpoints

⇒ Get predictions without worrying about underlying infrastructure

Different deployment options for different workloads



<https://t.me/learningnets>



Model Deployment

Deployment Strategy Options

Real Time Inference

Serverless Inference

Batch Transform

Asynchronous Inference



Model Deployment Strategies

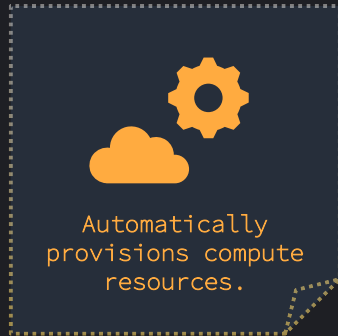
Deploying With Real-Time Inference

- For low-latency predictions with immediate response.
- Use Cases:
 - Real-time recommendation
 - Fraud detection

Model Deployment Strategies

Deploying with Serverless Inference

Features:



Use case:

Unpredictable, fluctuating traffic peaks

Model Deployment Strategies

Batch Transform

- Process very large datasets for predictions without needing a constant endpoint

Preprocess
datasets

⇒ Before training your machine learning

⇒ Run inference on large datasets without an active endpoint

Run inference on
large datasets

⇒ Large datasets in batches using processing job

Example

⇒ Run predictions on an entire dataset

Model Deployment Strategies

Deploying with Asynchronous Inference

- Queue requests and process them asynchronously
- For tasks with large data sizes and long processing durations
- Auto-scaling possible

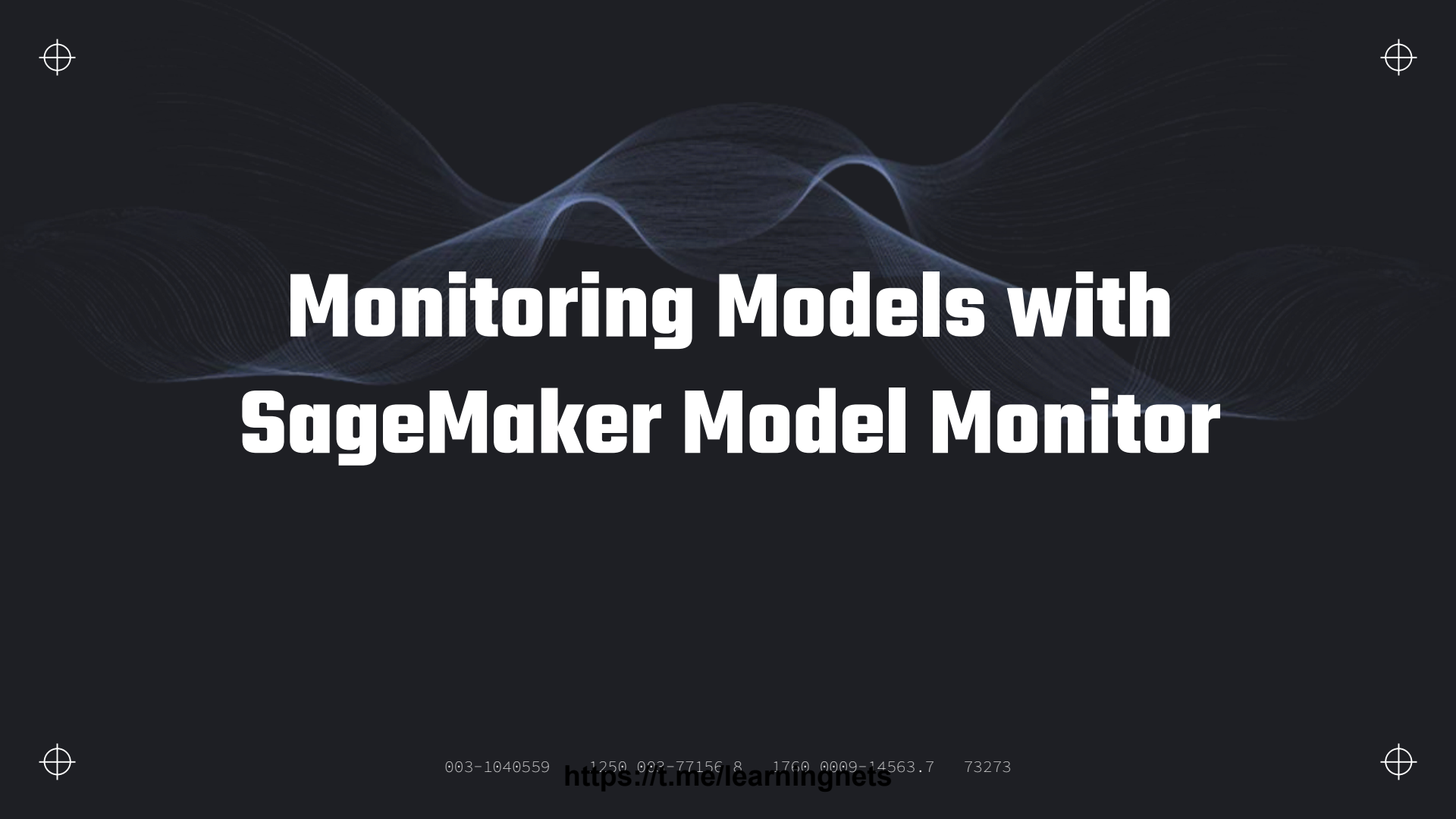


Conclusion

- Different deployment options for different needs.
- Helps to optimize performance and cost efficiency.

Section 8:

SageMaker: Monitoring Models



Monitoring Models with SageMaker Model Monitor

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Monitoring Models with SageMaker Model Monitor

What does model monitoring do?

- Continuously monitors models in production

Detects

Data Drift

Model Quality Drift

Bias Drift

Feature Attribution Drift

- Alert when there is deviation

We need it so that we can

Retrain models

Update training data

Update model

Monitoring Models with SageMaker Model Monitor



SageMaker Model Monitor

- Monitors the quality of Amazon SageMaker machine learning models



You can set up

- Continuous monitoring with a real-time endpoint
- Continuous monitoring with a batch transform
- On-schedule monitoring

Monitoring Models with SageMaker Model Monitor

- Prebuilt monitoring capabilities
- Supports custom monitoring capabilities

Available types of monitoring

Data Quality

Monitor drift in
data quality

Model Quality

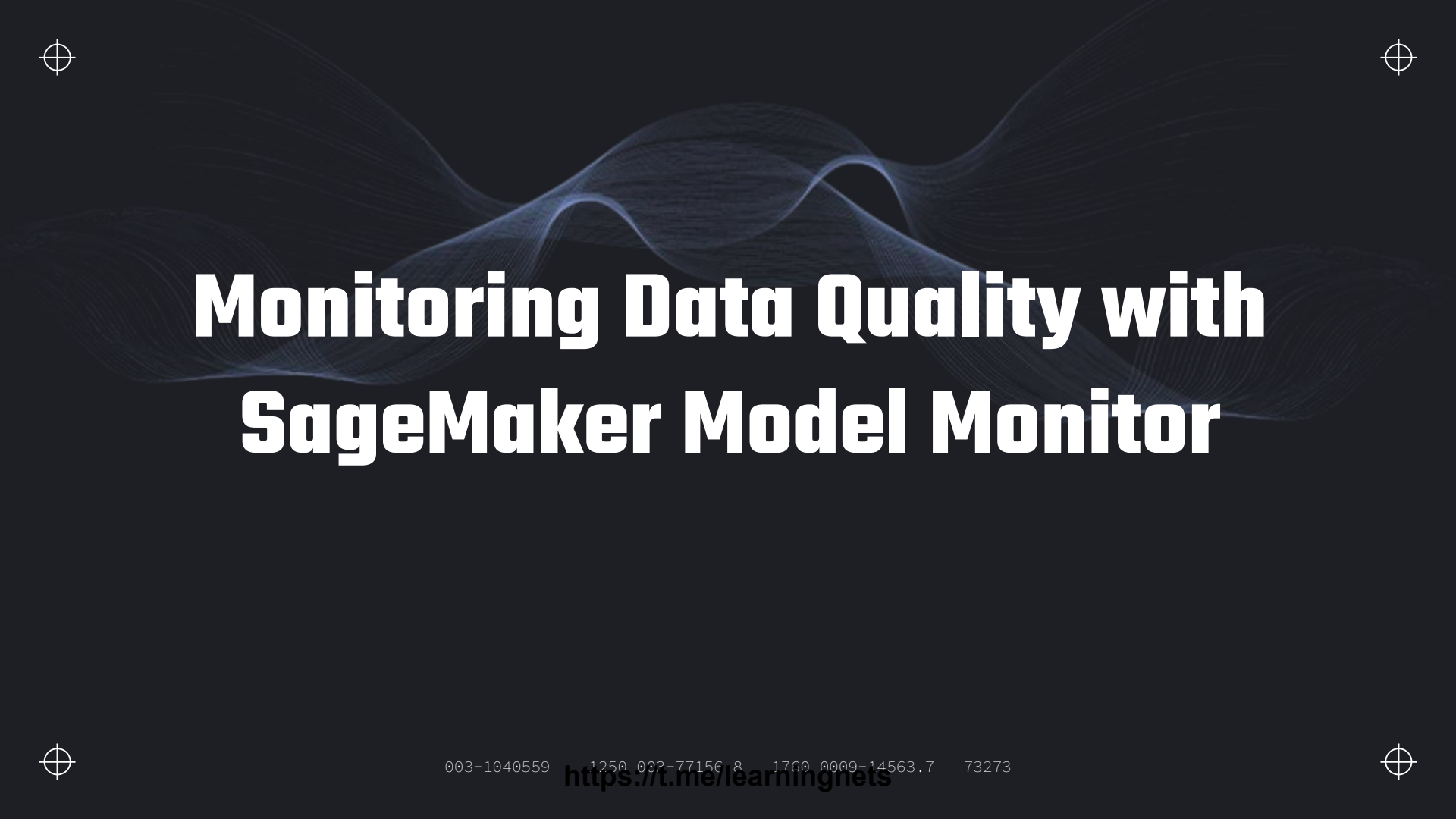
Monitor drift in
model quality

Model Bias

Monitor bias in
your model's
predictions

Model
Explainability

Monitor drift in
feature
attribution



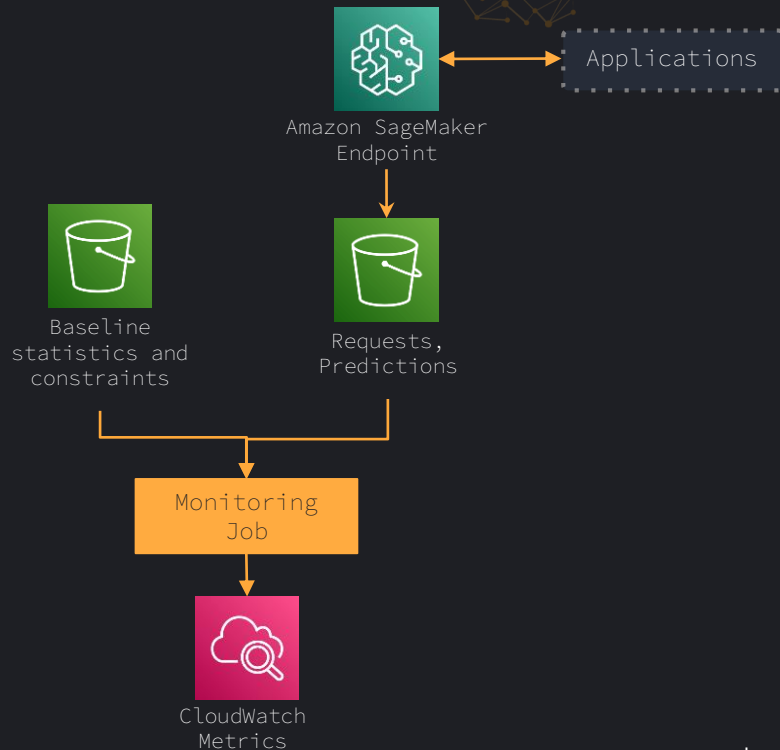
Monitoring Data Quality with SageMaker Model Monitor

003-1040559 1250 032-77156.8 1760 0009-14563.7 73273

<https://t.me/learningnets>

Monitoring Data Quality with SageMaker Model Monitor

- Notifies you when data quality issues arise
- Real-life data is not carefully available like training datasets
- Thus, the model begins to lose accuracy in its predictions
- Model Monitor detects data drift and creates an alert



Monitoring Models with SageMaker Model Monitor

How to monitor data quality

Step 1

Enable data capture

- Real-time endpoint: capture data from `requests` and `model predictions`
- Batch transform: capture data from `batch transform inputs` and `outputs`

Step 2

Create a baseline

- Run a baseline job that analyzes an input dataset that you provide
- Uses `Deequ`

Monitoring Models with SageMaker Model Monitor

How to monitor data quality

Step 3

Define and schedule model quality monitoring jobs

- What data to collect
- How often to collect it
- How to analyze it
- Which reports to produce

Step 4

View data quality metrics



Monitoring Models with SageMaker Model Monitor

How to monitor data quality

Step 5

Integrate data quality monitoring with Amazon CloudWatch

Step 6

Interpret the results of a monitoring job



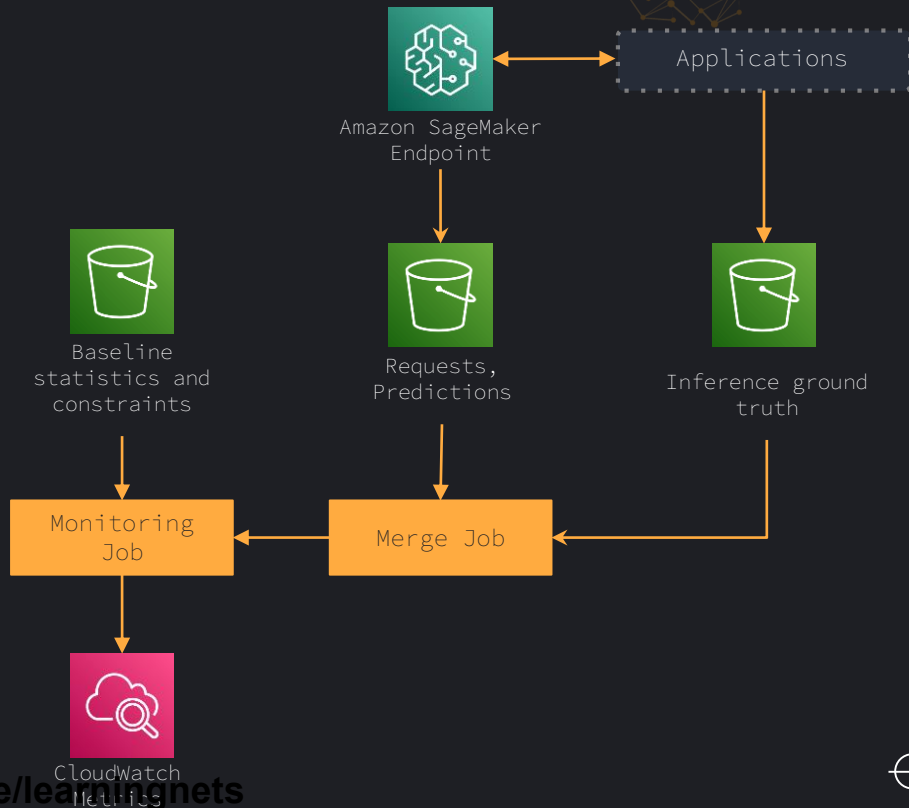
Monitoring Model Quality with SageMaker Model Monitor

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Monitoring Model Quality with SageMaker Model Monitor

- Monitors the performance of a model
- Merges captured data with actual labels
- Compares the predictions with the actual labels
- Notifies when there are deviations





Monitoring Model Quality with SageMaker Model Monitor



Model monitor metrics

- Depend on the ML problem type

Regression

Mean Absolute Error
Mean Squared Error
Square root of MSE

Binary classification

Confusion matrix
Recall
Precision
Accuracy

Multiclass classification

Confusion matrix
Weighted Recall
Weighted Precision
Accuracy



Monitoring Model Quality with SageMaker Model Monitor

How to monitor model quality

Step 1

Enable data capture

- Real-time endpoint: capture data from `requests` and `model predictions`
- Batch transform: capture data from `batch transform inputs` and `outputs`

Step 2

Create a baseline

- Create a baseline from the dataset that was used to train the model

Monitoring Model Quality with SageMaker Model Monitor

How to monitor model quality

Step 3

Define and schedule model quality monitoring jobs

- What data to collect
- How often to collect it
- How to analyze it
- Which reports to produce

Step 4

Ingest Ground Truth labels

- Ingest Ground Truth labels from a real-time inference endpoint or batch transform job



Monitoring Model Quality with SageMaker Model Monitor



How to monitor model quality

Step 5

Integrate model quality monitoring with Amazon CloudWatch

Step 6

Interpret the results of a monitoring job



Section 9:

SageMaker: Pipelines & Model Registry



SageMaker Pipelines

003-1040559 1250 032-77156.8 1760 0009-14563.7 73273

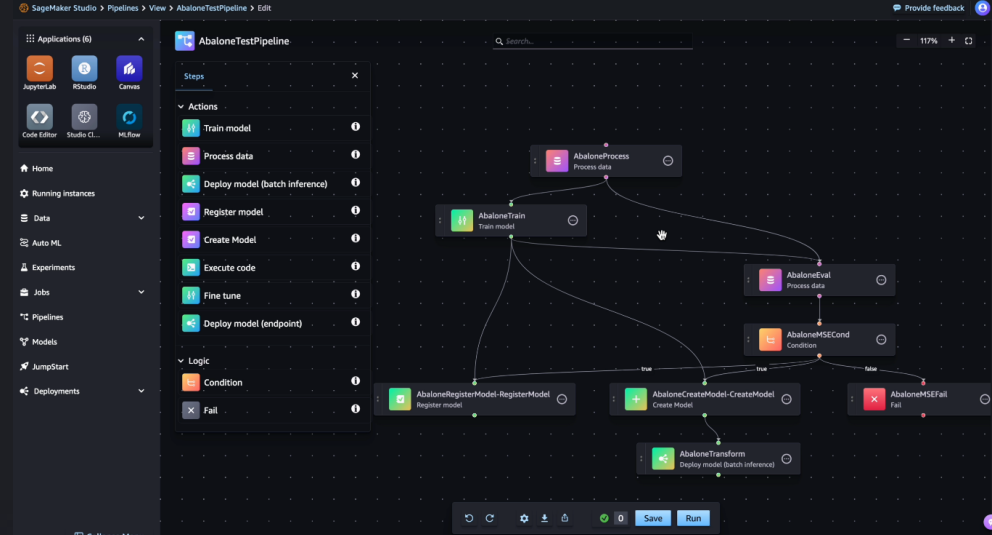
<https://t.me/learningnets>

SageMaker Pipelines

- Server-less workflow orchestration service
- Create, manage, and automate machine learning (ML) workflows
- Is Scalable

Is defined using

- Drag-and-drop UI
- Pipelines SDK
- Directed Acyclic Graph (DAG)
JSON definition



SageMaker Pipelines

Pipeline

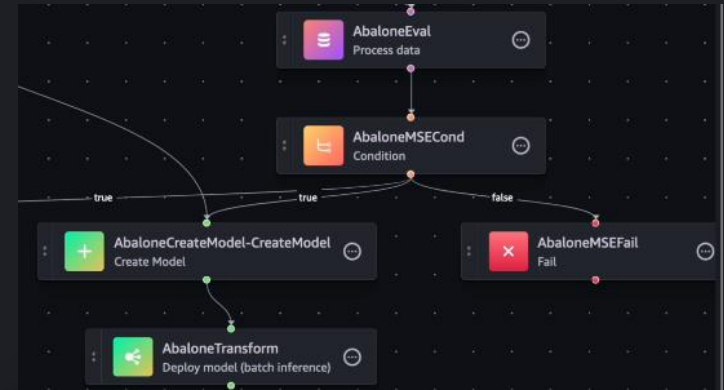
⇒ Is a blueprint

Pipeline Structure

- Is composed of a **name**, **steps**, and **parameters**

Pipeline name

- Must be unique within an (account, region) pair



SageMaker Pipelines

Pipeline steps

Defines

- Actions that the pipeline takes

Can be

Execute code

Create model

Register model

...

- Relationships between steps



SageMaker Pipelines



Pipeline parameters

- Provides a way to introduce variables
- Have a default value
- Must be defined in your pipeline definition

Parameter types

ParameterString

ParameterInteger

ParameterFloat

ParameterBoolean



SageMaker Pipelines

Why Use AWS SageMaker Pipelines?

Efficiency

Scalability

Collaboration

Model Registry Management

003-1040559 1250 032-77156.8 1760 0009-14563.7 73273

<https://t.me/learningnets>

SageMaker Model Registry Management



Centralized and organized repository for managing machine learning model

Used to

Registering Models

Deploying models

Managing model versions

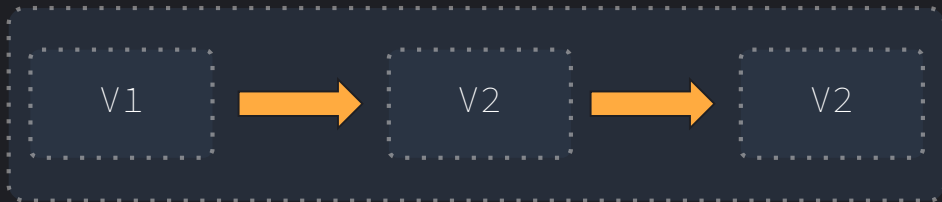
Organizing / Cataloging Models

Associating metadata

Sharing models

SageMaker Model Registry Management

- Model Package
- Model Group



- Approval Status
- Model Lineage



SageMaker Model Registry Management

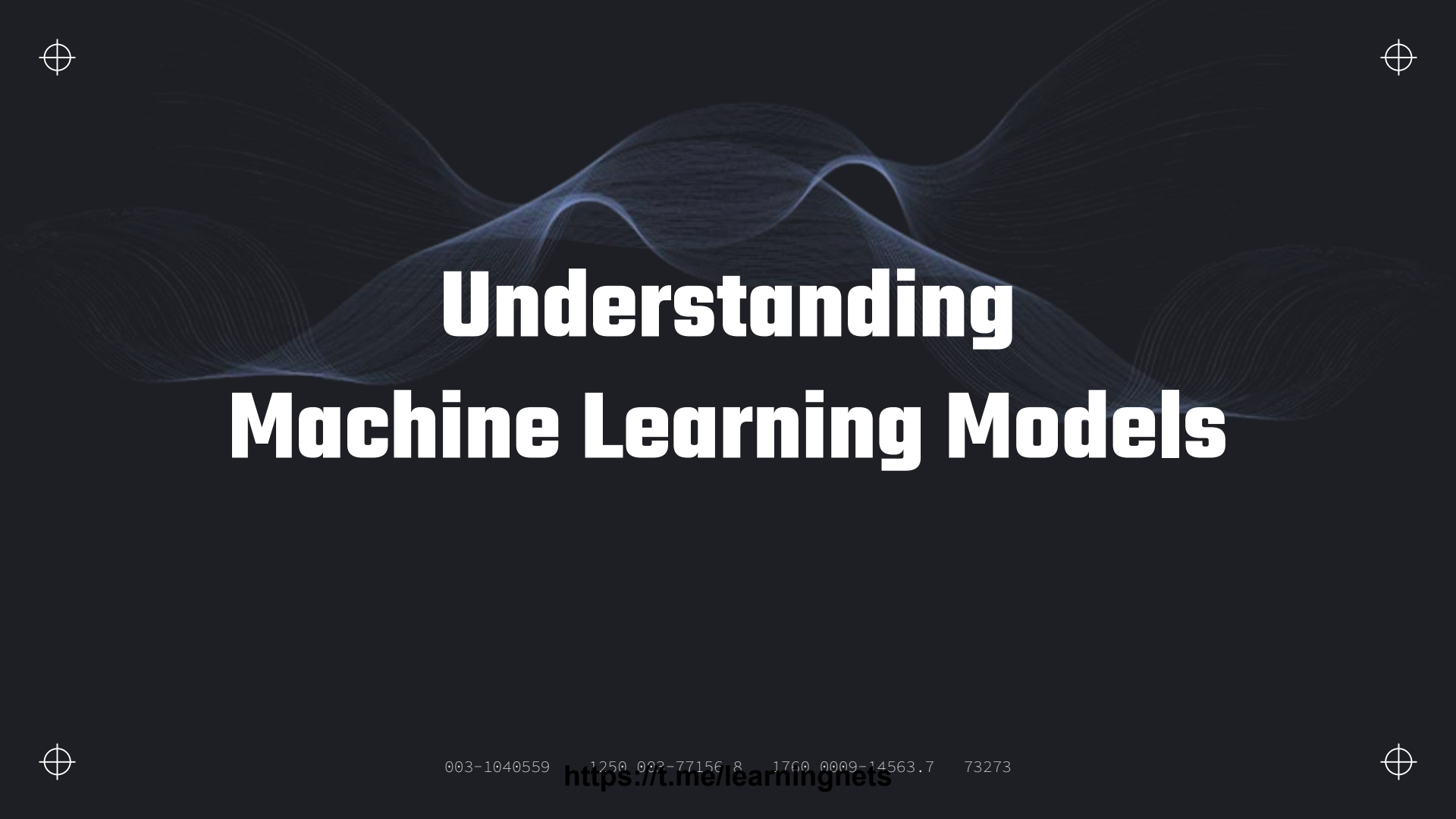


- Model Metrics
- Automation and Integration



Section 10:

Machine Learning Concepts



Understanding Machine Learning Models

003-1040559 1250 032-77156.8 1760 0009-14563.7 73273

<https://t.me/learningnets>

Understanding Machine Learning Models



Supervised Learning Models

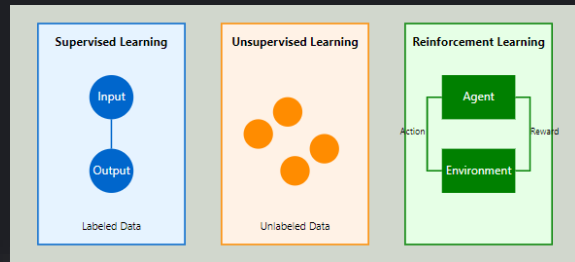
- Models learn from labeled data to predict outcomes
⇒ Use Cases: Classification, Regression

Unsupervised Learning Models

- Models identify patterns in unlabeled data
⇒ Use Cases: Clustering, Dimensionality Reduction

Reinforcement Learning Models

- Models learn through interactions with an environment
⇒ Use Cases: Robotics, Game AI



Understanding Machine Learning Models



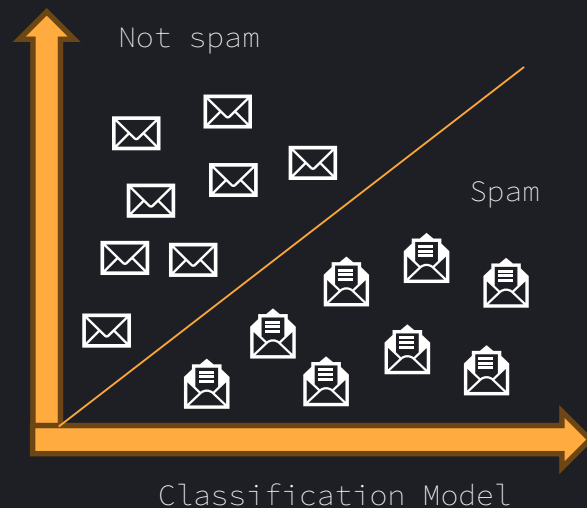
Classification Models

Categorize data into predefined classes/labels

⇒ Suitable for binary and multi-class classification problems

Use Cases:

⇒ Spam detection, sentiment analysis, image classification



Understanding Machine Learning Models



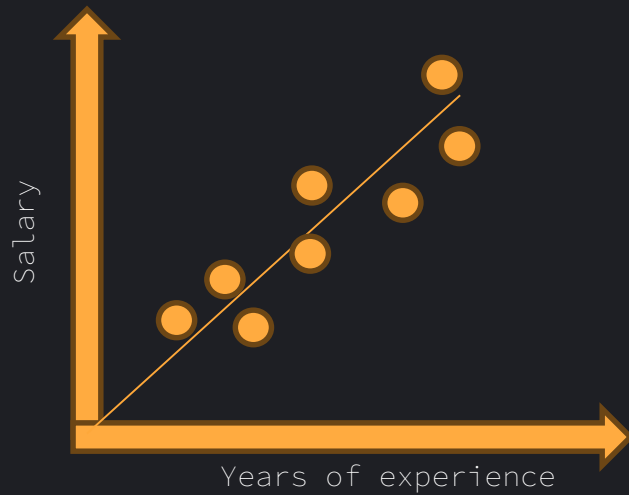
Regression Models

Predict continuous numerical values

⇒ Suitable for high-dimensional and sparse data

Use Cases:

⇒ Price prediction, demand forecasting, risk assessment



Understanding Machine Learning Models

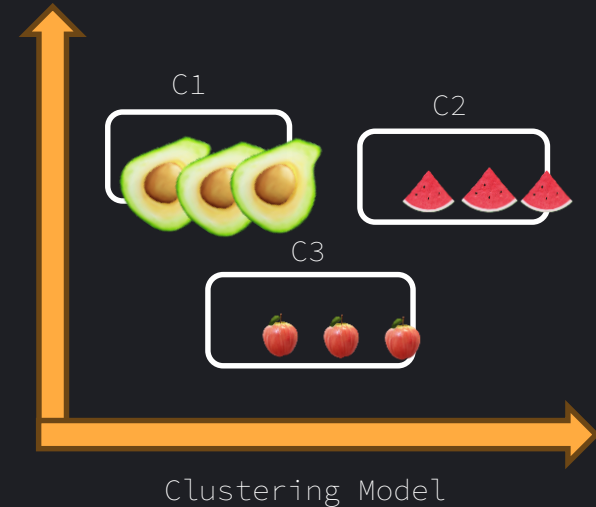
Clustering Models

Group similar data points into distinct clusters

⇒ Automatically selects the optimal number of clusters

Use Cases:

⇒ Customer segmentation, topic modeling, anomaly detection



Amazon SageMaker Algorithms

Built-in algorithms and their Use Cases



Classification

- Linear Learner
- XGBoost

Computer Vision

- Image Classification
- Object Detection
- Semantic Segmentation

Text Classification

- Blazing Text :
Supervised
Unsupervised

Anomaly Detection

- Random Cut Forests
- IP Insights

Topic Modeling

- Latent Dirichlet Allocation
- Neural Topic Model

Forecasting

- DeepAR

Recommendations

- Factorization Machines
 - KNN

Clustering

- K-Means
- K-Nearest Neighbors

<https://t.me/learningnets>

Amazon SageMaker Algorithms



Supervised Learning Algorithms

Algorithms designed for processing data using labeled examples to predict outcomes

Linear Learner	• Predicts dependent variables based on independent variables • E.g. Predict sales of Ice-Creams based on weather
XGBoost	• Predicts outcome by combining estimates from weaker models • E.g. Credit scores
DeepAR	• For time-series forecasting • E.g. weather forecasts

Amazon SageMaker Algorithms



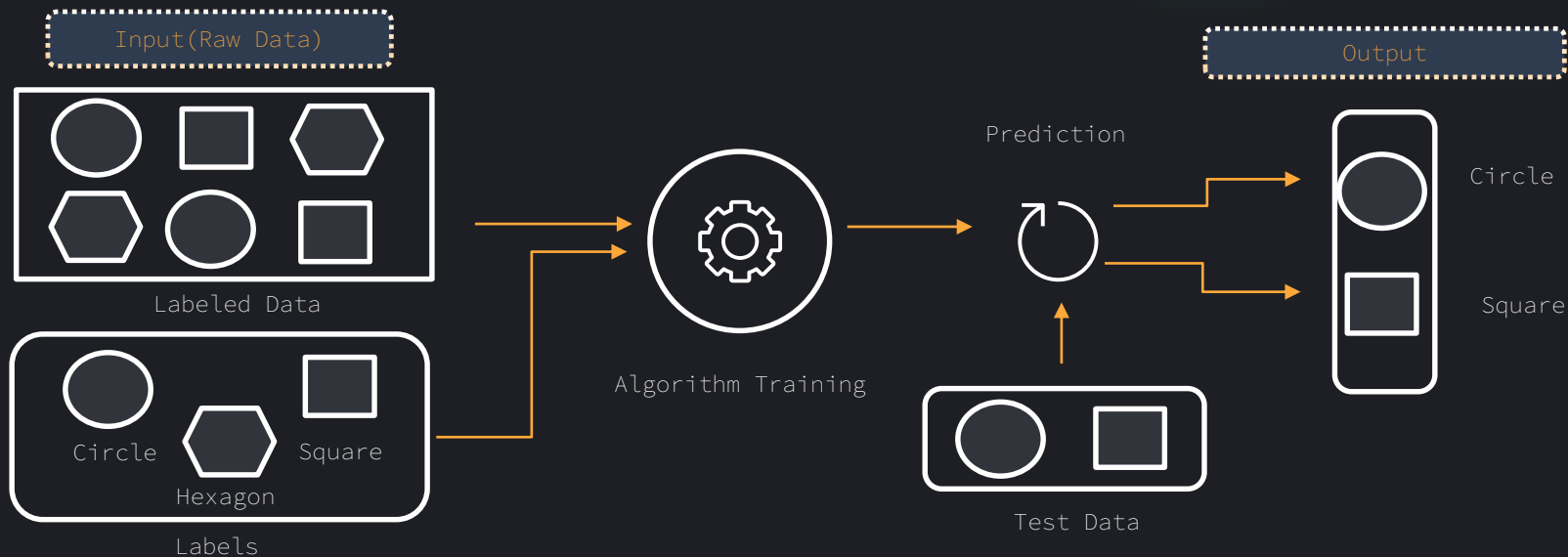
Supervised Learning Algorithms

Algorithms designed for processing data using labeled examples to predict outcomes

Factorization Machines	• For building recommendations • E.g. Song recommendations
KNN (K-Nearest Neighbors)	• Classifies data points based on their neighbors • E.g. finding a product a customer is likely to buy
Object2Vec	• Highly customizable embedding algorithm • Can be used for Multi-label document classification

Amazon SageMaker

Supervised Learning Algorithm Architecture



Amazon SageMaker Algorithms



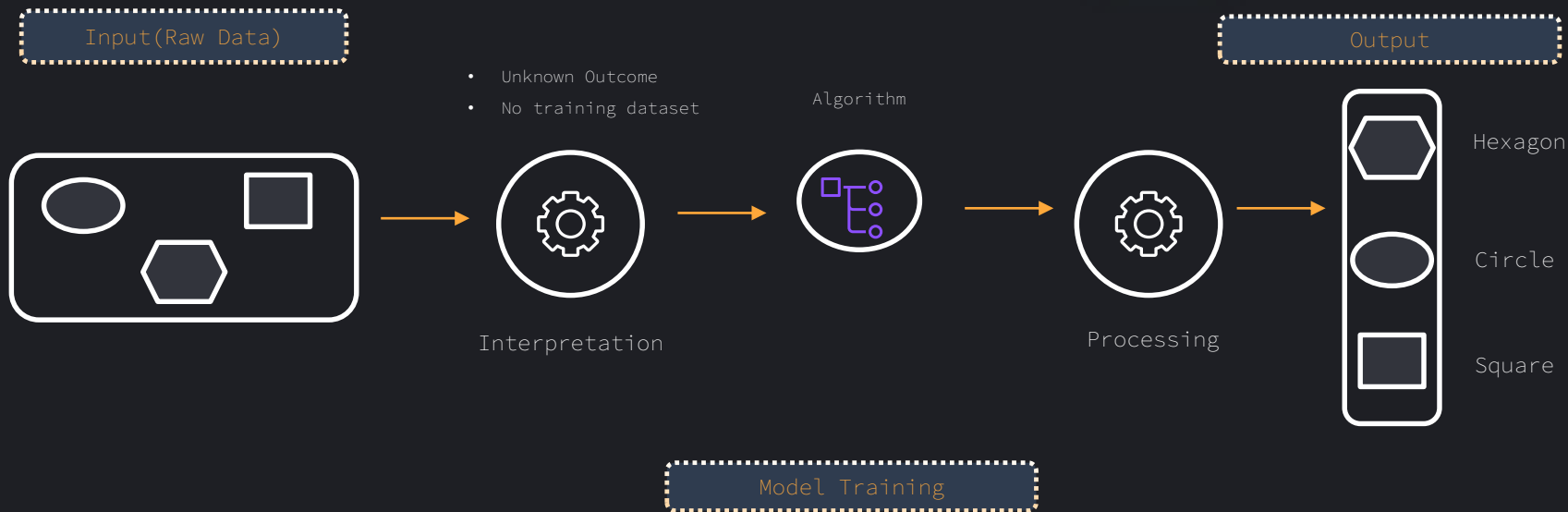
Unsupervised Learning Algorithms

Algorithms designed for processing data to uncover patterns without predefined labels

Principal Component Analysis	• Reduces the dimensionality of large datasets • E.g. simplifying the features of an image dataset
K-Means	• Identifies discrete groupings within data for clustering • E.g. Email spam detection
Random Cut Forest	• Identifies anomalies in a dataset • E.g. Identifying fraudulent transactions
IP Insights	• Detects unusual patterns of IPV4 address usage • Can be used to identify anomalous IP address

Amazon SageMaker

Unsupervised Learning Algorithm Architecture



Amazon SageMaker Algorithms



Textual Analysis Algorithms

Algorithms designed for processing and analyzing text data

BlazingText	• Processes text data quickly for tasks • E.g. automatically categorize customer emails into sub-inquiries
Sequence-to-Sequence	• Transforms one sequence of data into another • E.g. Speech-to-text or text-to-speech
Latent Dirichlet Allocation	• Discovers topics within a collection of documents • Aids in Organizing/Summarizing large text datasets
Neural Topic Model	• Groups documents by topics using neural networks • Can be used to classify or summarize documents based on the topics

Amazon SageMaker



Textual Analysis Algorithms

Input



Algorithm Training

Output



Topics



Key Phrases



Sentiment Analysis

Amazon SageMaker Algorithms



Image Processing Algorithms

Algorithms designed for analyzing and interpreting image data

Image Classification	• Algorithm used to classify images • E.g. simplifying the features of an image dataset
Semantic Segmentation	• Associates a label to every pixel in an image • Used to develop computer vision applications
Object Detection	• Detects and classifies all objects in an image • Uses deep neural network (DNN)

Amazon SageMaker

Image Processing Algorithms



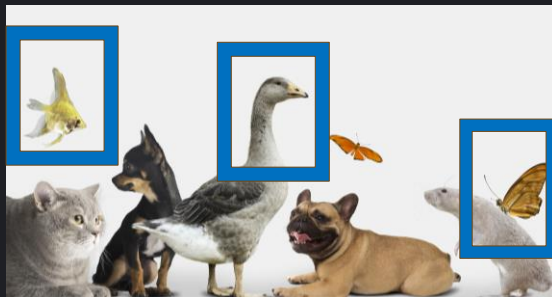
Classification



Deer

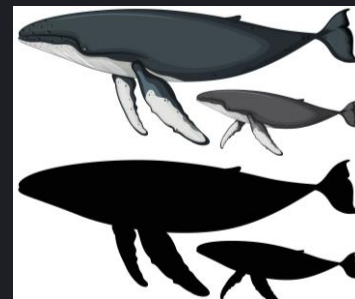
Single Object

Detection



Goldfish, Duck, Butterfly

Segmentation



Big whale, Small Whale

Multiple Objects

Amazon SageMaker



Training Models Using Built-In Algorithms



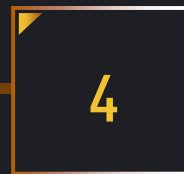
Prepare data



Select algorithm



Set hyperparameters



Train the model



Evaluate performance



Models Performance Evaluation

Model evaluation datasets

- Training Set:
Used to train the model
- Validation set:
Used to fine-tune and improve the model
- Test Set:
Used to assess the model's predictive quality

Models Performance Evaluation

Model Fit

- Overfitting:
Model performs well on training data but poorly on evaluation data
- Underfitting:
Model performs poorly on training data
- Balanced Model:
Model performs well on both training and evaluation datasets

Models Performance Evaluation

Bias

The difference between predicted and actual values

Variance

Dispersion of predicted values across different datasets

Models Performance Evaluation

Classification Problem Metrics

- Accuracy Correct classifications out of total
- Precision Proportion of correct positive predictions
- Recall Proportion of actual positives correctly identified
- F1-Score Harmonic mean of precision and recall
- AUC-ROC Curve Model's ability to distinguish between classes

Models Performance Evaluation

Regression Problem Metrics

- Mean Absolute Error(MAE)

Average absolute prediction error

- Mean Squared Error (MSE)

Average squared prediction error

- Root Mean Squared Error(RMSE)

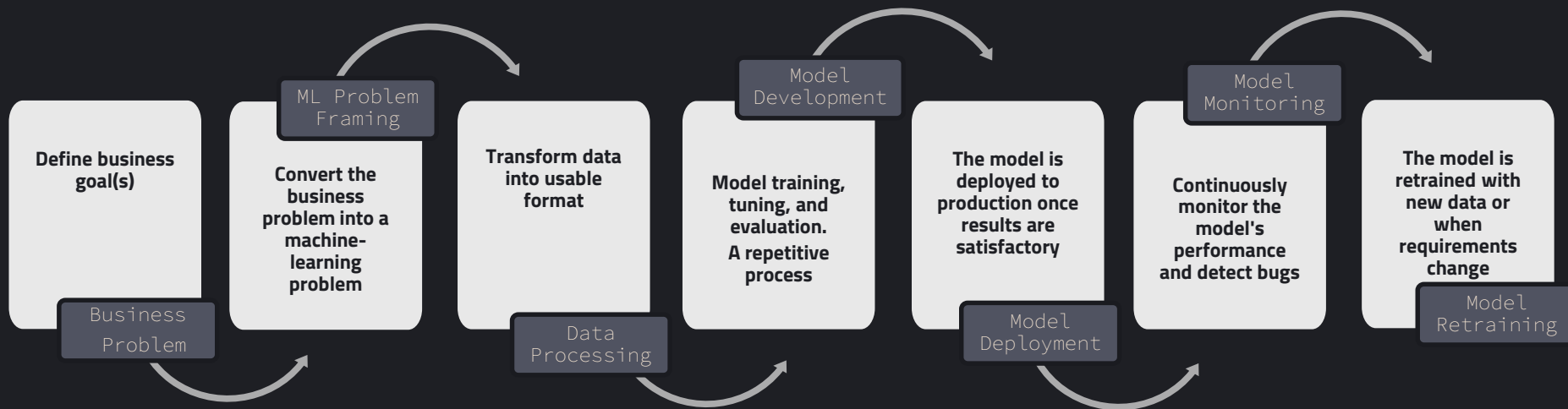
MSE's square root in target units

- R-squared

Variance in target explained by model

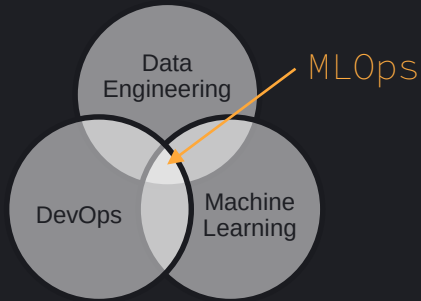
Machine Learning Development Lifecycle

An End-to-End Process for Developing, Deploying, and Maintaining Machine Learning Models



Machine Learning Operations (MLOps)

A set of practices that automate and simplify Machine Learning deployments



Goals:

- Facilitate collaboration
- Minimize model deployment time
- Improve quality metrics
- Governance

Machine Learning Operations (MLOps)

Benefits:

- Productivity
- Reliability
- Repeatability
- Auditability
- Data and model quality

Key Principles:

- Version Control
- Automation
- CI/CD
- Model Governance

Machine Learning Operations (MLOps)

Benefits of MLOps

- Productivity

Provides self-service environments with access to curated datasets

- Reliability

Incorporates CI/CD practices for quick and consistent deployments.

- Repeatability

Ensures a repeatable process

- Auditability

Demonstrates how models are built and deployed

- Quality

Enforces policies against model bias

Machine Learning Operations (MLOps)

Key Principles:

- **Version Control**
Tracks changes to assets for reproducibility and rollback when needed
- **Automation**
Automates various stages of the ML pipeline for repeatability
- **CI/CD**
Continuously tests and deploys assets.
- **Model Governance**
Ensure models are compliant with regulations and policies

Machine Learning Operations (MLOps)

AWS Services for MLOps

- Data Preparation:

SageMaker Data Wrangler

- Feature Store:

Helps create, share, and manage features for ML development

- Model Training:

SageMaker training job feature

- Experiments

SageMaker Experiments

- Processing

SageMaker Processing

Machine Learning Operations (MLOps)

AWS Services for MLOps

- **Model Registry:**
SageMaker Model Registry
- **Deployments**
SageMaker deploys Machine Learning models for predictions
- **Model Monitoring:**
SageMaker Model Monitor tracks the quality of models in production.
- **Pipelines**
SageMaker Model Building Pipelines create end-to-end workflows

Section 11: AWS Machine Learning Services



Amazon Bedrock

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Amazon Bedrock



- Platform for building generative AI application.
- Offers access various **foundational models**.

Foundational Models

⇒ Diverse Model Selection: AI21 Labs, Stable Diffusion, Llama, Amazon Titan, Jurassic, Claude, Command...

Customization & Fine Tuning

⇒ Private Customization: Customize with own data.
⇒ Fine-Tuning: Tune models for specific domains.

Serverless

⇒ It eliminates the server management processes

Data Protection & Privacy

⇒ Prompt and model response secured.
⇒ Data is encrypted in transit and at rest.

Flexible Pricing

⇒ On-Demand Mode: Pay as you go
⇒ Provisioned Throughput Mode: for large and steady workloads.

<https://t.me/learningnets>

Amazon Bedrock

- You can easily get benefits of other AWS Services.
 - Monitoring: AWS CloudWatch
 - Auditing: AWS CloudTrail
 - Storage: Amazon S3
 - Model Development: Amazon SageMaker
- Automation and Orchestration:
 - **Agents for Amazon Bedrock:** Handle complex tasks, use company data, enhance responses, and call APIs automatically.
 - **Knowledge Bases for Amazon Bedrock:** Provide company data, manage data intake and retrieval, support multi-turn conversations.

Amazon Bedrock

Benefits

Efficient Model Building

- ⇒ Rich variety of foundational models with a **single API access**.
- ⇒ **Quick experimentation** and model evaluation with **playgrounds**.

Secure Application Development

- ⇒ Data remains within AWS region **encrypted**.
- ⇒ **AWS IAM** provides fine-grained control over access.

Customizable Experiences

- ⇒ **Automates** complex tasks, integrates with existing data sources.
- ⇒ Users can fine-tune their models.

Amazon Bedrock

Use Cases

Content Creation

- ⇒ Generate dynamic content for various cases.
- ⇒ Personalize user experiences in real-time.

Customer Support

- ⇒ Chatbots and virtual assistants.
- ⇒ Automate repetitive support tasks.

Data Augmentation

- ⇒ Create data for training other ML models.
- ⇒ Enhance datasets.

Product Recommendations

- ⇒ Personalized product suggestions.
- ⇒ Enhance e-commerce platforms with dynamic content.



Amazon Personalize

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Amazon Personalize

- Generate item recommendations for users
- Generates recommendations primarily based on item interaction data
- Interaction data can come from

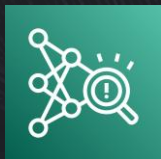
Bulk interaction records

Real-time events

Amazon Personalize

Common Use Cases

- Personalizing a video streaming app
- Adding product recommendations to an ecommerce app
- Adding real-time next best action recommendations to your app
- Creating personalized emails
- Creating a targeted marketing campaign
- Personalizing search results



Amazon Fraud Detector

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Amazon Fraud Detector



- Fully managed service that identifies fraudulent online activities.
 - Payment frauds, fake accounts, bots...
- It uses pre-build machine learning models and offers customization.
- Real-time detection.

Easy Integration

**Automated Model
Training and
Deployment**

**Flexible and
Scalable**

Amazon Fraud Detector

Workflow



Data Ingestion

Feature
Engineering

Model Training

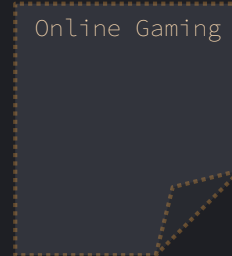
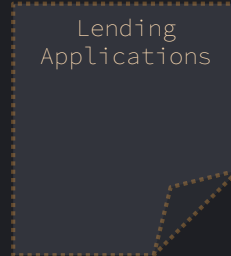
Model
Deployment

Fraud Detection

Amazon Fraud Detector



- Reduced fraud losses.
- Improved customer experience.
- Operational efficiency.





Amazon Augmented AI



Amazon Augmented AI



- Enables a human review of machine learning (ML) systems.
- It makes building and managing human reviews for ML applications easy.
- Provides built-in human review workflows
- Supports custom human review workflows





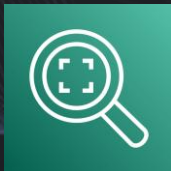
Amazon Augmented AI



Use cases

- Amazon A2I with Amazon Textract
- Amazon A2I with Amazon Rekognition
- Amazon A2I to review real-time ML inferences
- Amazon A2I with Amazon Comprehend
- Amazon A2I with Amazon Transcribe
- Amazon A2I with Amazon Translate
- Amazon A2I to review tabular data





Amazon Rekognition

003-1040559 1250 032-77156 8 1760 0009-14563.7 73273

<https://t.me/learningnets>

Amazon Rekognition

- Is a cloud-based image and video analysis service
- Analyzes any image or video file that's stored in Amazon S3
- It can be used to :

Detect objects

Detect texts

Detect Unsafe content

Compare faces



Amazon Rekognition



Use cases

- Searchable Media Libraries
- Face-Based User Identity Verification
- Face Liveness Detection
- Facial Search
- Unsafe Content Detection
- Detection of Personal Protective Equipment
- Celebrity Recognition
- Text Detection





Amazon Textract

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>



Amazon Textract

- Automatically extract text from any document
⇒ Advanced Optical Character Recognition (OCR)
- Extracts printed text, handwriting, layout elements, and data

Features



Text extraction



Table & form
extraction



Custom queries



Document Analysis



<https://t.me/learningnets>



Amazon Textract

Use cases

- Intelligent Search Index Creation
- NPL integration
- Data capture automation
- Document Classification and extraction

Benefits

- Easy integration with Applications
- Scalability and Cost-effectiveness
- Synchronous and Asynchronous processing



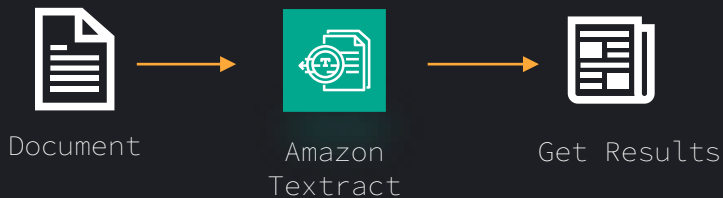
<https://t.me/learningnets>



Amazon Textract

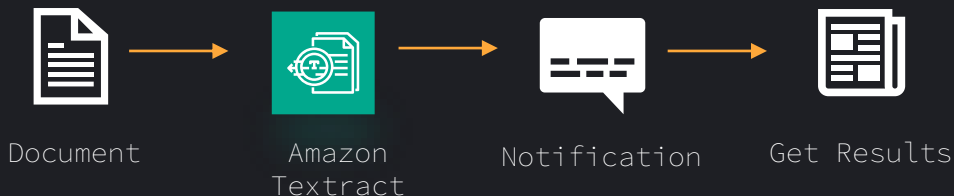
Processing Methods

Synchronous



For single-page documents

Asynchronous



For multi-page documents
Up to 3000 pages





Amazon Textract

Amazon Textract API Operations

AnalyzeDocument

⇒ Extracts relationships among detected text

AnalyzeExpense

⇒ Extracts data from invoices and receipts

AnalyzeID

⇒ Extracts information from ID documents

Analyze Lending
Workflow

⇒ Processes mortgage loan packages automatically



<https://t.me/learningnets>

Abstract blue wavy lines flowing across the top half of the slide.

Amazon Kendra

003-1040559 1250 002-77156.8 1700 0009-14563.7 73273

<https://t.me/learningnets>



Amazon Kendra

- Intelligent search service
⇒ Uses machine learning and NLP
- Provides search functionality to applications
- Amazon Kendra Intelligent Ranking

Features

- ✓ Understands search queries
- ✓ Integrates various data sources
- ✓ Indexes and crawls documents
- ✓ Customizable search experience

<https://t.me/learningnets>



Amazon Kendra

Querying Amazon Kendra

Factoid
questions

- ⇒ simple questions
- ⇒ "What is the capital of France?"

Descriptive
questions

- ⇒ Questions that require more elaborate answers
- ⇒ "How do I setup my Computer"

Keyword and NL
questions

- ⇒ Questions that have complex, conversational content





Amazon Kendra



Amazon Kendra Editions

1

Developer Edition

- Not for production
- Up to 5 indexes
- 10,000 documents or 3 GB text

2

Enterprise Edition

- For production use
- Indexing entire document libraries

Amazon Kendra Components

- **Index** a 'temporary database' that makes documents searchable
- **Data source** a document repository
- **Document Addition API** directly adds documents to an index

<https://t.me/learningnets>



Section 11: Data Ingestion

003-1040559 1250.000-77156.8 1750.0009-14563.7 73273

<https://t.me/learningnets>



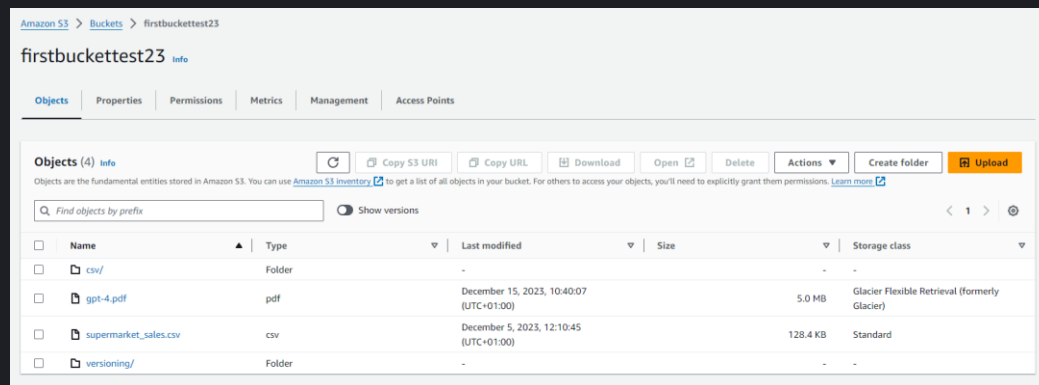
AWS S3 – Storage

AWS S3 - Storage

Main Storage Solution

Data Management

- One of the most important building blocks in AWS
- S3 = "*Simple Storage Service*"
- Cost-effective and simple object storage
- *Buckets* (containers for storage) and *objects* (files)



simple web services interface

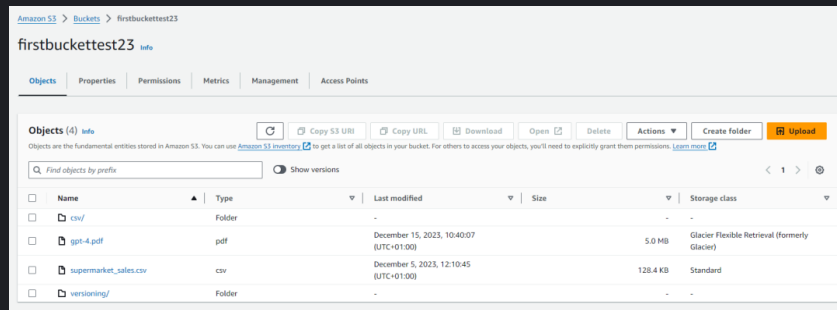
<https://t.me/learningnets>

AWS S3 - Storage

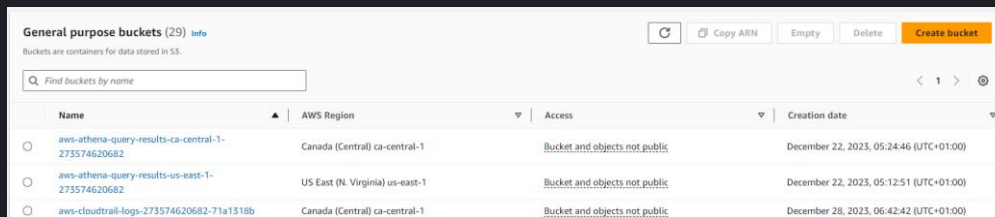
Data Management



- *Buckets* (containers for storage) and *objects* (files)



- Each bucket is created in a specific *region*



AWS S3 - Storage



Rules

General purpose buckets (29) info

Buckets are containers for data stored in S3.

Find buckets by name

Name	AWS Region	Access	Creation date
aws-athena-query-results-ca-central-1-273574620682	Canada (Central) ca-central-1	Bucket and objects not public	December 22, 2023, 05:24:46 (UTC+01:00)
aws-athena-query-results-us-east-1-273574620682	US East (N. Virginia) us-east-1	Bucket and objects not public	December 22, 2023, 05:12:51 (UTC+01:00)
aws-cloudtrail-logs-273574620682-71a1518b	Canada (Central) ca-central-1	Bucket and objects not public	December 28, 2023, 06:42:42 (UTC+01:00)

- Each bucket is created in a specific *region*
- Buckets must have a globally unique name (across all regions, across all accounts)
- Between 3 (min) and 63 (max) characters
- Only lowercase letters, numbers, dots (.), and hyphens (-)
- Must begin and end with a letter or number.
- Not formatted as an IP address (for example, 192.168.5.4)

AWS S3 - Storage



Key

General purpose buckets (29) [info](#)

Buckets are containers for data stored in S3.

Name	AWS Region	Access	Creation date
aws-athena-query-results-ca-central-1-273574620682	Canada (Central) ca-central-1	Bucket and objects not public	December 22, 2023, 05:24:46 (UTC+01:00)
aws-athena-query-results-us-east-1-273574620682	US East (N. Virginia) us-east-1	Bucket and objects not public	December 22, 2023, 05:12:51 (UTC+01:00)
aws-cloudtrail-logs-273574620682-71a1518b	Canada (Central) ca-central-1	Bucket and objects not public	December 28, 2023, 06:42:42 (UTC+01:00)

- Each bucket is created in a specific *region*
- Each object is identified by a unique, user-assigned key



- Upload: `example.txt`



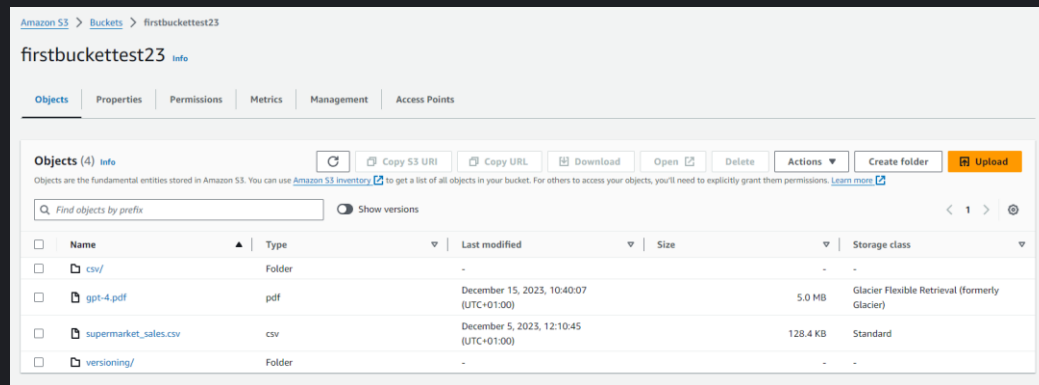
`documents/example.txt`



AWS S3 - Storage

Data Management

- *Buckets* (containers for storage) and *objects* (files)



simple web services interface

Use Cases

- Backup & recovery
- Websites, Applications
- Data archiving
- Data lakes
- ... etc.

<https://t.me/learningnets>

AWS S3 – Storage Classes

Storage Class	Use Case	Durability	Availability
<i>S3 Standard</i>	Frequently accessed data	99.999999999% "11 nines"	99.99%
<i>S3 Intelligent-Tiering</i>	Data with unknown or changing access patterns	99.999999999%	99.90%
<i>S3 Standard-IA</i>	Less frequently accessed data, but requires rapid access when needed	99.999999999%	99.90%
<i>S3 One Zone-IA</i>	Same as Standard-IA, but stored in a single AZ for cost savings	99.999999999%	99.50%
<i>S3 Glacier</i>	Long-term archiving with retrieval times ranging from minutes to hours	99.999999999%	99.99% (after retrieval)
<i>S3 Glacier Deep Archive</i>	Longest-term archiving where retrieval time of 12 hours is acceptable	99.999999999%	99.99% (after retrieval)

Durability

Likelihood of losing an object in a year (1 in 100 billion)

Availability

Percentage of time that the service is operational

Lifecycle Rules

Define change of storage classes over time

Versioning

Allows you to retrieve previous versions of an object



Data Ingestion Methods

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>



Streaming Ingestion

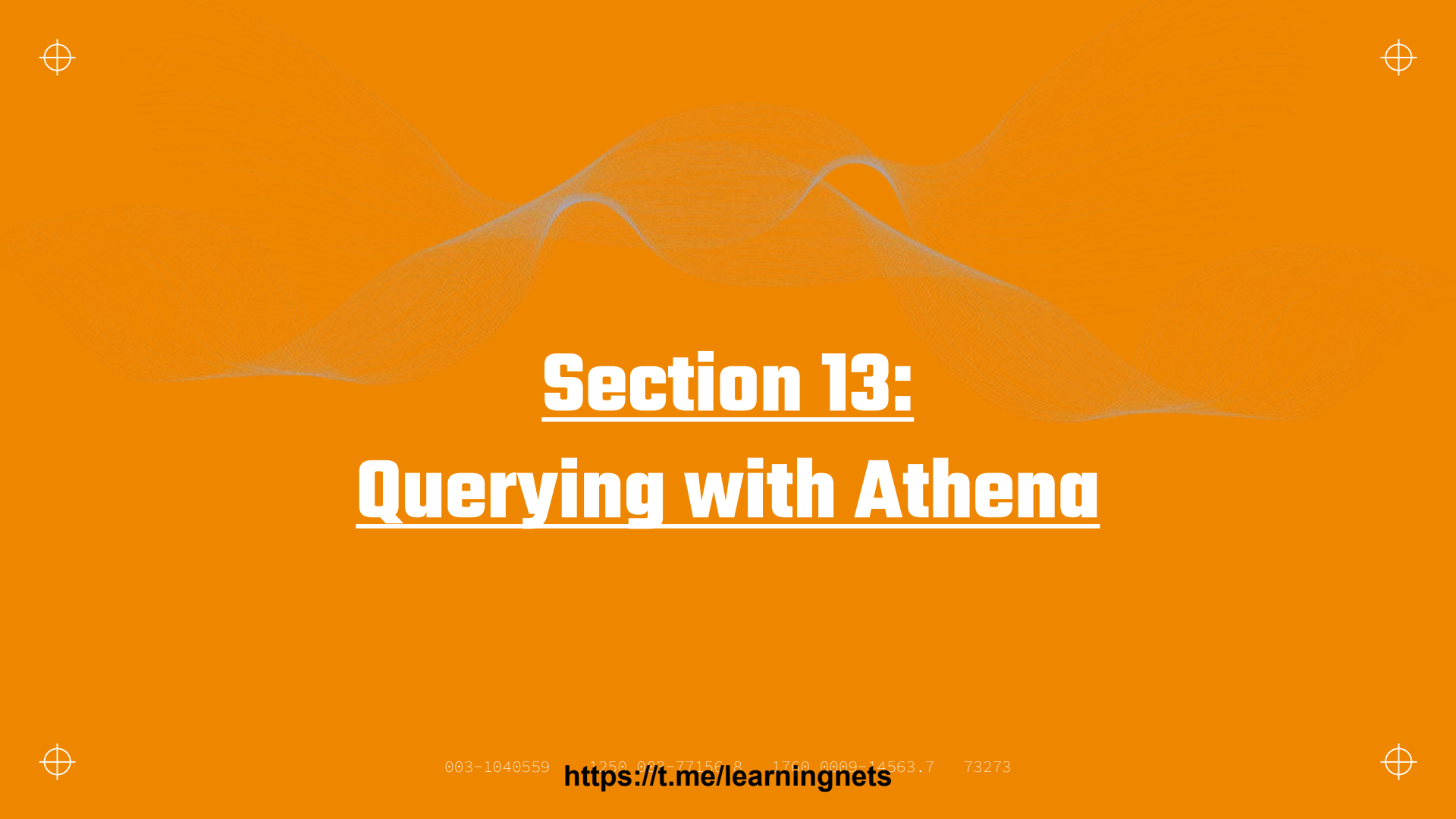
- Enables real-time ingestion
- Ideal for time-sensitive data
- More expensive and intricate
- Implemented using services like Amazon Kinesis for streaming data.

VS

Batch Ingestion

- Ingests data periodically in batches.
- Typically large volumes
- Cost-effective and efficient
- Tools like AWS Glue commonly used



The background of the slide features a solid orange color with a subtle, abstract pattern of wavy, translucent lines in a slightly lighter shade of orange, creating a sense of depth and movement.

Section 13: Querying with Athena

003-1040559 1250.000-77156.8 1700.0009-14563.7 73273

<https://t.me/learningnets>



AWS Athena

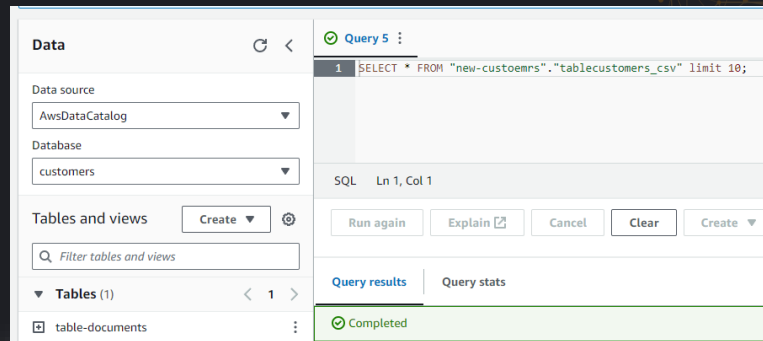
003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>



AWS Athena

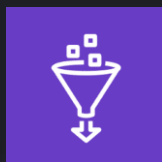
- Interactive query service:
Query files in S3 using SQL
- Serverless:
No infrastructure to manage
Pay-as-you-go



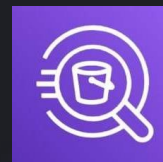
S3 Bucket



Crawler



Data Catalog



Athena



Quicksight





AWS Athena



- **Log Analysis:**
Analyzing log files stored in Amazon S3
- **Ad-Hoc Analysis:**
Ad-hoc queries on data lakes stored in S3
- **Data Lake Analytics:** Building a data lake on Amazon S3 and using Athena to query data
- **Real-Time Analytics:**
Integrating Athena with streaming data sources such as Amazon Kinesis

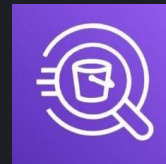




Athena Federated Queries

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>



Federated Query:

- Query data sources **other than S3 buckets** using a data connector.
 - Relational and non-relational data sources
 - Object data sources
 - Custom data sources
- Federated data sources: (built-in examples)
 - Amazon CloudWatch Logs,
 - Amazon DynamoDB,
 - Amazon DocumentDB,
 - Amazon RDS,...
- Work with **multiple sources**
 - Amazon RDS - products table
 - Amazon DocumentDB - detailed customer profile data
 - Amazon DynamoDB - user interactions





Athena Workgroups

003-1040559 1250 032-77156.8 1760 0009-14563.7 73273

<https://t.me/learningnets>



Workgroups:

Isolate queries from other queries in the same account

- Isolate queries for different...

- Teams
- Use cases
- Applications

⇒ Different settings or to track and control cost

- Control...

- Query execution settings
- Access
- Cost
- Type of engine (Athena SQL vs. Apache Spark)

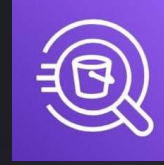
- Up to 1000 workgroups per region

- Each account has primary workgroup





Athena - Cost



Cost:

- Pay for queries run – amount of data scanned
- Cost saving possible with reserved capacity

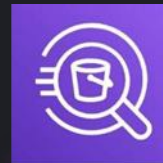




Athena - Performance Optimization



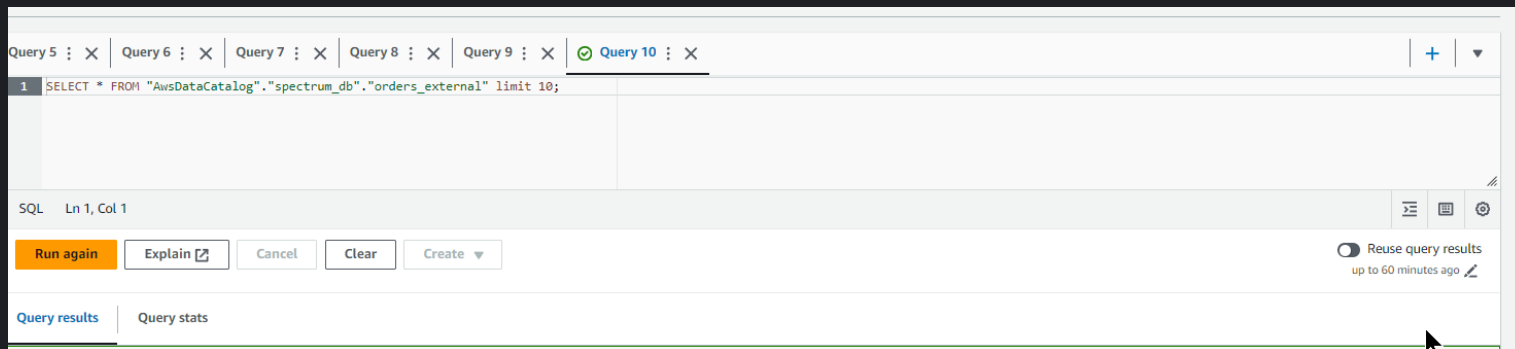
- Use partitions
 - Eliminate data partitions that need to be scanned (pruning)
- Use partition projection
 - Automate partition management
 - Speed up queries for tables with large partitions
- AWS Glue Partition Indexes:
 - Athena retrieves only relevant partitions instead of loading all
 - Optimize query planning and reduce query runtime



Athena – Query Result Reuse

- What it does?

Reuses previous results that match your query and max. age



- Benefits?

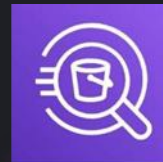
Improve query performance & cost



Athena – Query Result Reuse



- When to use?
 - Query where the source data doesn't change frequently
 - Repeated queries
 - Large datasets with complex queries

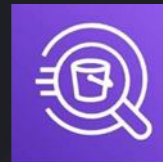




Athena - Performance Optimization



- **Data Compression:**
Reduce file size to speed up queries
- **Format Conversion:**
transform data into an optimized structure such as Apache Parquet or Apache ORC columnar formats



Section 14:

AWS Data Processing Services



Glue Costs

003-1040559 1250.000-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Glue Costs

- Crawlers:
 - Hourly rate based on the number of DPUs used
 - Billed by seconds with with a 10-minute minumum
- What are DPUs?
 - DPUs = Data Processing Units
 - A single DPU provides 4 vCPU and 16 GB of memory
- Data Catalog:
 - o Up to a million objects for free
 - o \$1.00 per 100,000 objects over a million, per month

Glue Costs



- ETL jobs:
 - Hourly rate based on the number of DPUs used
 - Billed by seconds with a 10-minute minimum
 - AWS Glue versions 2.0 and later have a 1-minute minimum
- How many DPUs are used?
 - Apache Spark: Minimum of 2 DPUs – Default: 10 DPUs
 - Spark Streaming: Minimum of 2 DPUs – Default: 2 DPUs
 - Ray job (ML/AI): Minimum of 2 M-DPUs (high memory). Default: 6 M-DPUs
 - Python Shell job (flexible & simple): 1 DPU or 0.0625 DPU. Default 0.0625 DPU
- Cost of DPUs
 - \$0.44 per DPU-Hour (may differ and depend on region)

Glue Costs

- Glue Job Notebooks / Interactive Sessions:
 - Used to interactively develop ETL code in notebooks
 - Based on time session is active and number of DPUs
 - Configurable idle timeouts
 - 1-minute minimum billing
 - Minimum of 2 DPUs – Default: 5 DPUs

Glue Costs

- ETL jobs cost example:

- Apache Spark job
- Runs for 15 minutes
- Uses 6 DPU
- 1 DPU-Hour is \$0.44

⇒ Job ran for 1/4th of an hour and used 6 DPUs

⇒ $6 \text{ DPU} * 1/4 \text{ hour} * \$0.44 = \$0.66.$

- Interactive Session cost example:

- Use a notebook in Glue Studio to interactively develop your ETL code.
- 5 DPU (default)
- Keep the session running for 24 minutes (2/5th of an hour)

⇒ Billed for $5 \text{ DPUs} * 2/5 \text{ hour} \text{ at } \$0.44 \text{ per DPU-Hour} = \$0.88.$



AWS Budgets

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

AWS Budgets

- Alarms:
Set budgets & receive alarms when exceeded
- Actual & Forecasted
Help to manage cost
- Budget Types:
 - Cost budget
 - Usage budget
 - *Saving plans budget*
 - *Reservation plans budget*
- Budgets are free
Two action-enabled budgets are free then it is \$0.10/day



Stateful vs. Stateless

003-1040559 1250 032-77156.8 1760 0009-14563.7 73273

<https://t.me/learningnets>

Stateful vs. Stateless



- **Stateful:**
Systems remember past interactions for influencing future ones.
- **Stateless:**
Systems process each request independently without relying on past interactions.

Stateful vs. Stateless

Data Ingestion Context:

- **Stateful:**
Maintain context for each data ingestion event.
- **Stateless:**
Process each data ingestion event independently.



Stateful vs. Stateless

Data Ingestion in AWS:

- Amazon Kinesis:
Supports both stateful (Data Streams) and stateless (Data Firehose) data processing.
- AWS Data Pipeline:
Orchestrates workflows for both stateful and stateless data ingestion.
- AWS Glue:
Offers stateful or stateless ETL jobs with features like job bookmarks for tracking progress.



Glue Transformations

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

AWS Glue – Extract Transform Load

Extract

- Amazon RDS, Aurora, DynamoDB
- Amazon Redshift
- Amazon S3, Kinesis

Transform

- Filtering: Remove unnecessary data
- Joining: Combine data
- Aggregation: Summarize data.
- FindMatches ML: Identify records that refers same entity.
- Detect PII: Identify and manage sensitive information.

CSV <-> Parquet <-> JSON <-> XML

Load

- Amazon RDS, Aurora, DynamoDB
- Amazon Redshift
- Amazon S3, Kinesis

S3 Target Location

Choose an S3 location in the format s3://bucket/prefix/object/ with a trailing slash (/).

Data Catalog update options [Info](#)

Choose how you want to update the Data Catalog table's schema and partitions. These options will only apply if the Data Catalog table is an S3 backed source.

- ☐ Do not update the Data Catalog
- ☒ Create a table in the Data Catalog and on subsequent runs, update the schema and add new partitions
- ☐ Create a table in the Data Catalog and on subsequent runs, keep existing schema and add new partitions

Database

Choose the database from the AWS Glue Data Catalog.

<https://t.me/learningnets>

AWS Glue – Extract Transform Load

Extract

- Amazon RDS, Aurora, DynamoDB
- Amazon Redshift
- Amazon S3, Kinesis

Transform

- Filtering: Remove unnecessary data
- Joining: Combine data
- Aggregation: Summarize data.
- FindMatches ML: Identify records that refers same entity.
- Detect PII: Identify and manage sensitive information.

CSV <-> Parquet <-> JSON <-> XML

Load

- Amazon RDS, Aurora, DynamoDB
- Amazon Redshift
- Amazon S3, Kinesis

S3 Target Location

Choose an S3 location in the format s3://bucket/prefix/object/ with a trailing slash (/).

Data Catalog update options [Info](#)

Choose how you want to update the Data Catalog table's schema and partitions. These options will only apply if the Data Catalog table is an S3 backed source.

- ☐ Do not update the Data Catalog
- ☒ Create a table in the Data Catalog and on subsequent runs, update the schema and add new partitions
- ☐ Create a table in the Data Catalog and on subsequent runs, keep existing schema and add new partitions

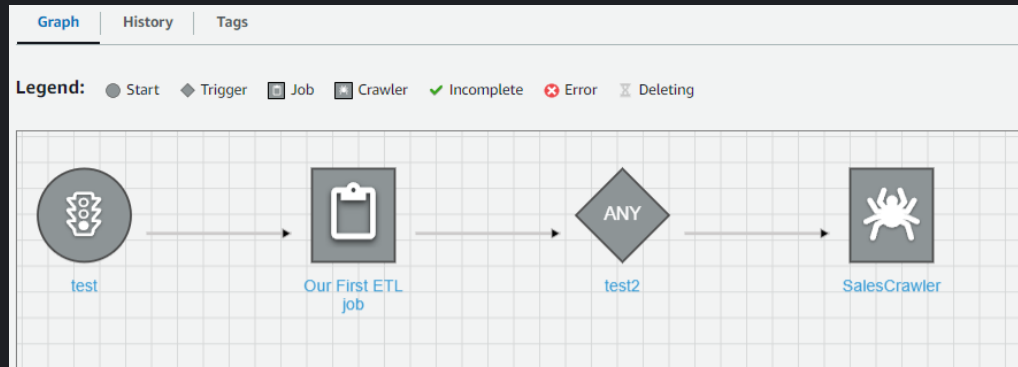
Database

Choose the database from the AWS Glue Data Catalog.



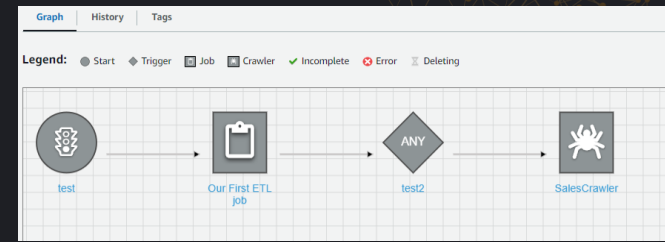
<https://t.me/learningnets>

Glue Workflows



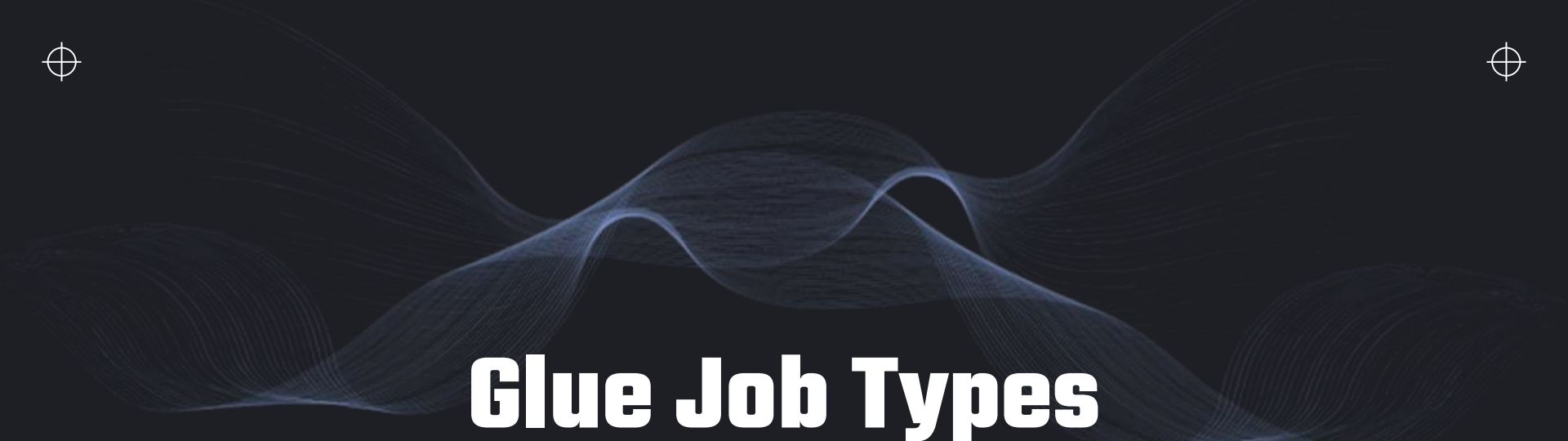


AWS Glue – Glue Workflows



- Orchestrate multi-step data processing jobs, manage executions and monitoring of jobs/crawlers.
- Ideally used for managing AWS Glue operations but also can be leveraged other services.
- Provides visual interface.
- You can create workflows manually or with AWS Glue Blueprints.
- **Triggers** initiate jobs and crawlers.
 - **Schedule Triggers:** Starts the workflow at regular intervals.
 - **On-Demand Triggers:** Manually start the workflow from AWS Console.
 - **EventBridge Event:** Launches the workflow based on specific events captured by Amazon EventBridge.
 - **On-Demand & EventBridge:** Combination of On-Demand and EventBridge rules.
 - **Lambda Function:** With a trigger that invokes Workflow.





Glue Job Types

003-1040559 1250.002-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

AWS Glue

[AWS Glue](#) > Jobs

AWS Glue Studio [Info](#)

Create job [Info](#)

Author in a visual interface focused on data flow.

 **Visual ETL**

Author using an interactive code notebook.

 **Notebook**

Author code with a script editor.

 **Script editor**

Type
The type of ETL job. This is set automatically based on the types of data sources you have selected.

Glue version [Info](#)

Language

Worker type
Set the type of predefined worker that is allowed when a job runs.

Script ✕

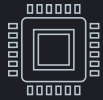
Engine

 **Choose file**

Limited to Python (*.py, *.py3) files only.

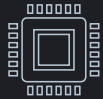
AWS Glue

Job Types



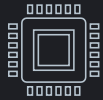
Spark ETL Jobs:

- ⇒ Large-scale data processing.
- ⇒ 2DPU to 100DPU



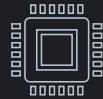
Spark Streaming ETL Jobs:

- ⇒ Analyze data in real-time.
- ⇒ 2DPU to 100DPU



Python Shell Jobs:

- ⇒ Suitable for light-weight tasks.
- ⇒ 0.0625DPU to 1DPU.



Ray Jobs:

- ⇒ It is suitable for parallel processing tasks.

AWS Glue

Execution Types



Standard Execution:

- ⇒ Designed for predictable ETL jobs.
- ⇒ Jobs start running immediately.
- ⇒ Guarantees consistent job execution times.



Flex Execution:

- ⇒ Cost-effective option for less time-sensitive ETL jobs.
- ⇒ Jobs may start with some delay.



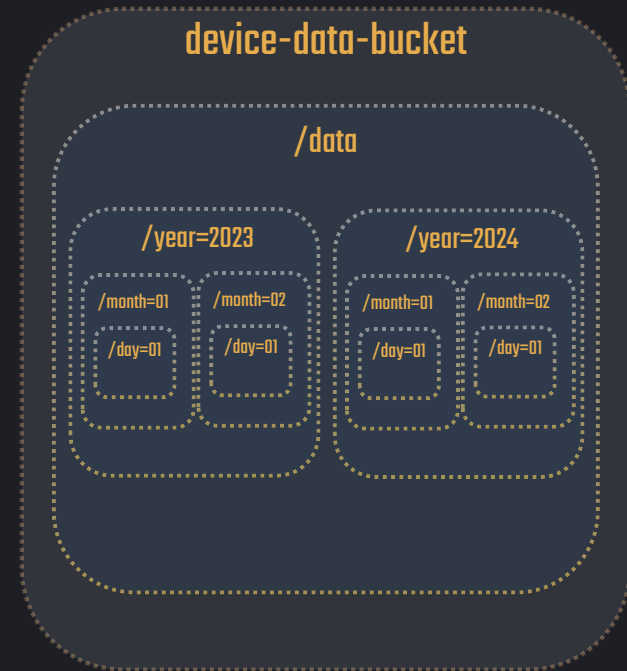
Partitioning

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

AWS Glue – Partitioning

- Enhances the performance of AWS Glue
 - Provides better query performance.
 - Reduces I/O operations.
- AWS Glue can skip over large segments within partitioned data.
- AWS Glue can process each partition independently.
- Provides cost efficiency by reducing query efforts.
- In AWS Glue, define partitioning as part of ETL job scripts. Also possible within Glue Data Catalog.





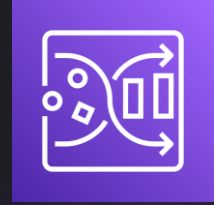
AWS Glue DataBrew

003-1040559 1250 032-77156.8 1760 0009-14563.7 73273

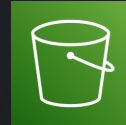
<https://t.me/learningnets>

AWS Glue DataBrew

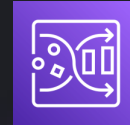
- Data preparation tool with visual interface.
- Cleaning and data format processes.
- Pre-built transformations.
- No coding required.
- Automate data preparations.



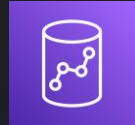
Amazon S3
(Data Lake)



AWS Glue
Data Brew



Amazon
Redshift



AWS Glue DataBrew

The screenshot displays the AWS Glue DataBrew console interface. At the top, it shows 'Sample project - 1' with a 'Create job' button and tabs for 'LINEAGE' and 'ACTIONS'. Below this is a toolbar with various data manipulation icons like 'UNDO', 'REDO', 'FILTER', 'SORT', 'COLUMN', 'FORMAT', 'CLEAN', 'EXTRACT', 'MISSING', 'INVALID', 'DUPLICATES', 'OUTLIERS', 'SPLIT', 'MERGE', 'CREATE', 'FUNCTIONS', 'CONDITIONS', 'NEST-UNNEST', 'PIVOT', 'GROUP', 'JOIN', 'UNION', and 'MORE'. The main area is divided into two panes. The left pane, titled 'Viewing 15 columns 500 rows', shows a table with columns: # year, # assembly_session, # state_code, and ALC state_name. Each column has a distribution chart above it. The right pane, titled 'Recipe (0)', shows a 'Sample recipe - 1' in 'Working version' and a 'Build your recipe' section with the text 'Start applying transformation steps to your data. All your data preparation steps will be tracked in the recipe.' and an 'Add step' button.

# year	# assembly_session	# state_code	ALC state_name
1946	1	2	United States of America
1947	2	2	United States of America
1948	3	2	United States of America
1949	4	2	United States of America
1950	5	2	United States of America
1951	6	2	United States of America
1952	7	2	United States of America
1953	8	2	United States of America
1954	9	2	United States of America
1955	10	2	United States of America
...	...	...	...

<https://t.me/learningnets>

AWS Glue DataBrew - Transformations

Project

Where you configure transformation tasks

Step

Applied transformation to your dataset

Recipe

Set of transformation steps; can be saved and reused

Job

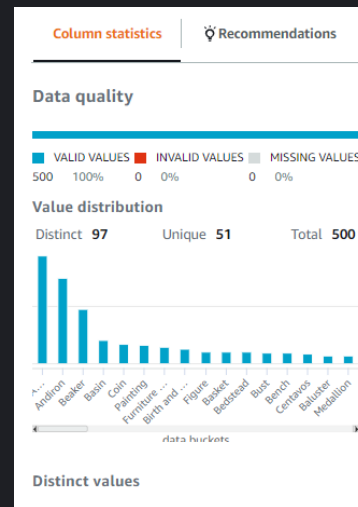
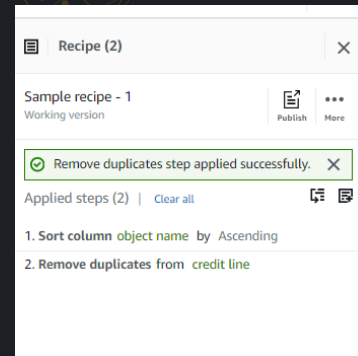
Execution of a recipe on a dataset; output to locations such as S3

Schedule

Schedule jobs to automate transformation

Data Profiling

Understand quality and characteristics of your data



<https://t.me/learningnets>

AWS Glue DataBrew - Transformations

- **NEST_TO_MAP:**
 - convert columns into a map.

Name	Age	City		NEST_TO_MAP
Alice	30	New York	→	{'Name': 'Alice', 'Age': 30, 'City': 'New York'}

- **NEST_TO_ARRAY:**
 - convert columns into an array

Name	Age	City		NEST_TO_ARRAY
Alice	30	New York	→	['Alice', 30, 'New York']

- **NEST_TO_STRUCT**
 - Like NEST_TO_MAP but retains exact data type and order

AWS Glue DataBrew - Transformations

- **UNNEST_ARRAY:**

- Expands array to multiple columns

NEST_TO_ARRAY			
['Alice', 30, 'New York']			



Name	Age	City
Alice	30	New York

AWS Glue DataBrew - Transformations

- **PIVOT**

- Pivot column and pivot values to rotate data from rows into columns

Product	Quarter	Sales
A	Q1	150
A	Q2	200
B	Q1	180
B	Q2	210



Product	Q1	Q2
A	150	200
B	180	210

- **UNPIVOT**

- Pivot column into rows (attribute + value)

Name	Age	City
Frank	40	Miami

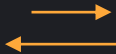


Attribute	Value
Name	Frank
Age	30
City	Miami

AWS Glue DataBrew - Transformations

- TRANSPOSE
 - Switch columns and rows

Name	Age	City
Alice	30	New York
Frank	32	Miami



Attribute	Alice	Frank
Age	30	32
City	New York	Miami

AWS Glue DataBrew - Transformations

Join

⇒ Combine two datasets.

Split

⇒ Split a column into multiple columns based on a delimiter.

Filter

⇒ Apply conditions to keep only specific rows in your dataset.

Sort

⇒ Arrange the rows in your dataset in ascending or descending order.

Date/Time Conversions

⇒ Convert strings to date/time formats or change between different date/time formats.

Count Distinct

⇒ Calculates the number of unique entries in that column.

AWS Lambda



003-1040559 1250 032-77156 8 1760 0009-14563.7 73273

<https://t.me/learningnets>

AWS Lambda



- What is AWS Lambda?
 - Lets you run code without managing servers
⇒ *automatically scaling based on demand*
 - Serverless compute service
⇒ *No need to provision or manage servers*
- Various programming languages
 - Python
 - Java
 - Node.js
 - Go
 - ...

AWS Lambda



Use cases

- Data processing tasks on data in Amazon S3 or DynamoDB
- Event-driven ingestion:
 - S3
 - *DynamoDB*
 - *Kinesis*

⇒ *Process real-time events, such as file upload*
- Automation:

Automate tasks and workflows by triggering Lambda functions in response to events

AWS Lambda

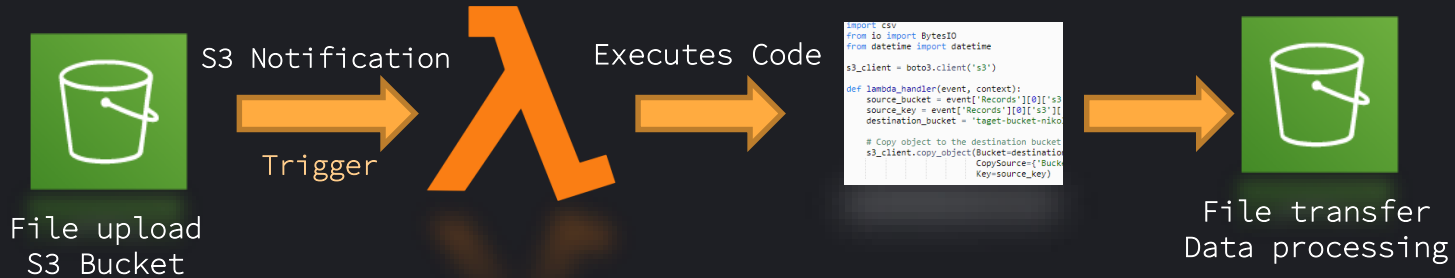


Advantages

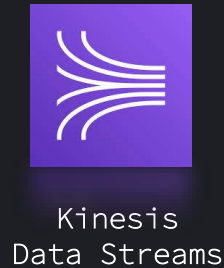
- Scalable: Scales automatically depending on workload
- Cost efficient: Pay only for what you use
- Simplicity: No need to manage infrastructure

Typically stateless: Each invocation is independent and doesn't maintain state

AWS Lambda – S3 Notification



AWS Lambda – Kinesis Data Stream



Event Source
Trigger



Continuously
executes

```
import csv
from io import BytesIO
from datetime import datetime

s3_client = boto3.client('s3')

def lambda_handler(event, context):
    source_bucket = event['Records'][0]['s3']
    source_key = event['Records'][0]['s3']
    destination_bucket = 'target-bucket-miso'

    # Copy object to the destination bucket
    s3_client.copy_object(Bucket=destination
                        CopySource='Bucket=source_key')
```



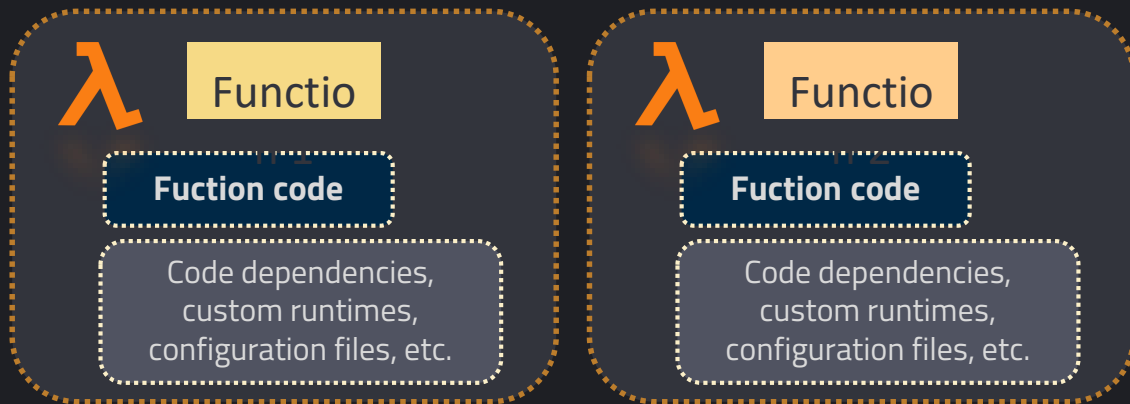
File transfer
Data processing

⇒ Executes in batches

⇒ Automatically scales

Lambda Layers

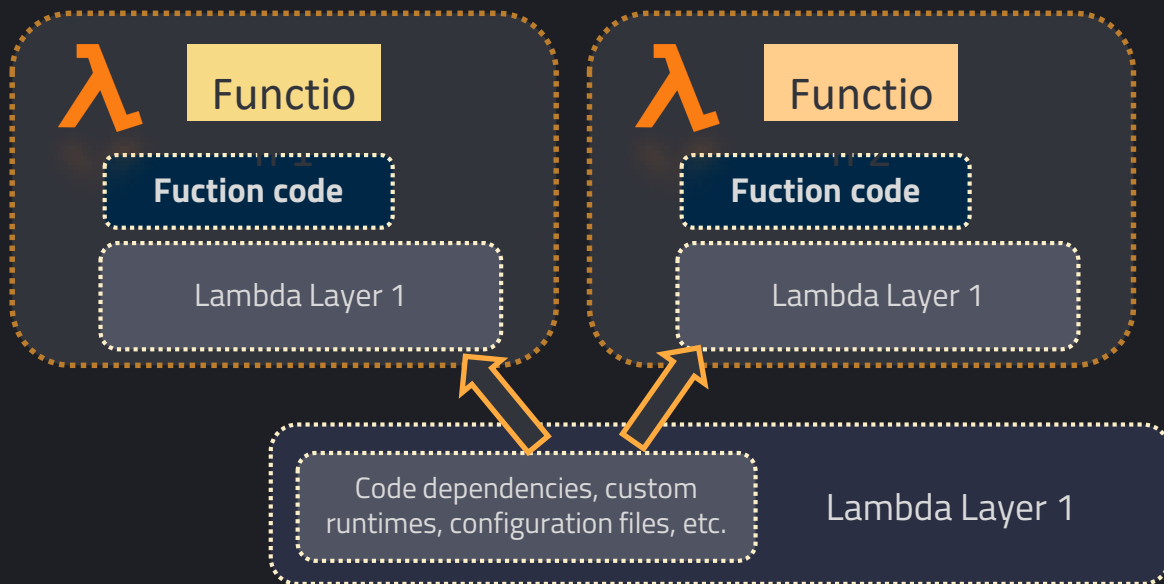
- Contains **supplementary code**
 - library dependencies,
 - custom runtime or
 - configuration file



Lambda function
components:
Without layers

Lambda Layers

1. Package Layer content (.zip)
2. Create the Layer in Lambda
3. Add the Layer to Your Functions
4. The function can access the contents of the layer during runtime



Lambda function
components:
With layers

Benefits

1

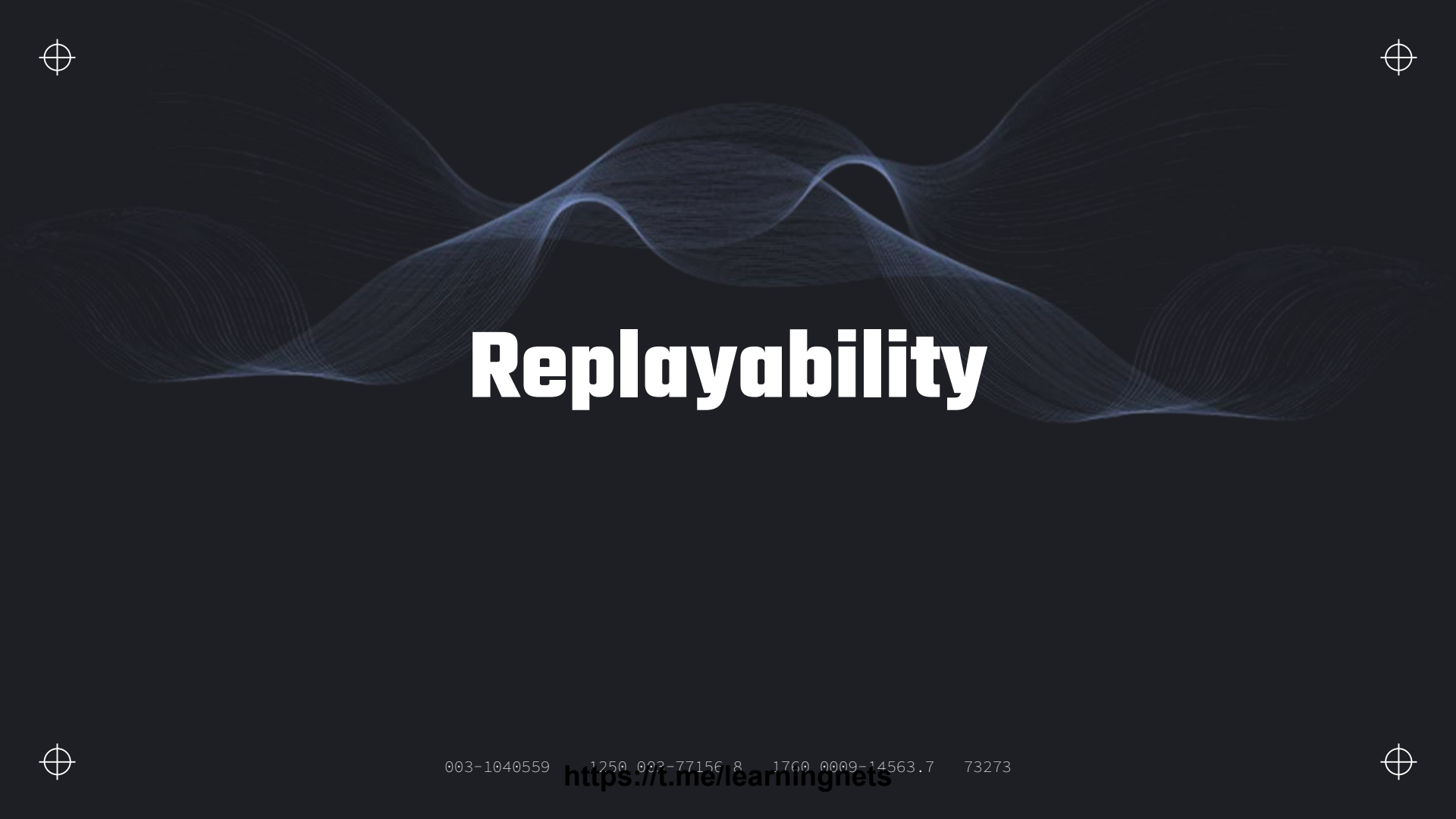
Share Dependencies Across Multiple Functions

2

Separate Core Logic from Dependencies

3

Reduce Deployment Package Size



Replayability

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Replayability

Definition of Replayability:

- Ability to reprocess or re-ingest data that has already been handled.

Why is it important?

- Error Handling: Corrects processing mistakes and recovers lost data.
- Data Consistency: Ensures uniform data across distributed systems.
- Adapting to Changes: Adjusts to schema or source data changes.
- Testing and Development: Facilitates feature testing and debugging without risking live data integrity.

Replayability

Strategies for Implementing Replayability

- **Idempotent Operations:**
Ensure repeated data processing yields consistent results.
- **Logging and Auditing:**
Keep detailed records for tracking and diagnosing issues.
- **Checkpointing:**
Use markers for efficient data process resumption.
- **Backfilling Mechanisms:**
Update historical data with new information as needed.

Replayability

Replayability is an important safety net for data processing.

It ensures systems are resilient, accurate, and adaptable to changes.

Abstract blue wavy lines flowing across the top half of the slide.

Amazon Kinesis

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Amazon Kinesis

Collection of services:

Handling different aspects of **data streaming**

- **Kinesis Data Streams**

Ingest and process large volumes of streaming data

- **Kinesis Firehose**

Fully managed service to deliver streaming data to destinations more easily

⇒ *Amazon S3, Amazon Redshift, Amazon Elasticsearch Service, and Splunk.*

- **Managed Apache Flink**

(formerly Kinesis Data Analytics)

Analyze streaming data in real time using standard SQL queries



Amazon Kinesis

Use cases

Variety of real-time data processing and analytics use cases

1) Real-time analytics

Analyzing streaming data to gain insights

2) IoT Processing

Ingesting and processing data from IoT devices or sensors

3) Security and fraud detection:

Detecting anomalies and responding to security in real time



Amazon Kinesis Data Streams

003-1040559 1250 002-77156.8 1760 0009-14563.7 73273

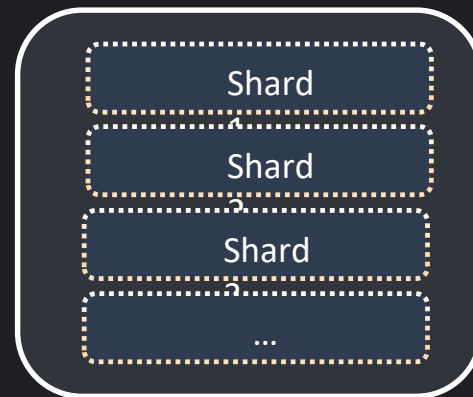
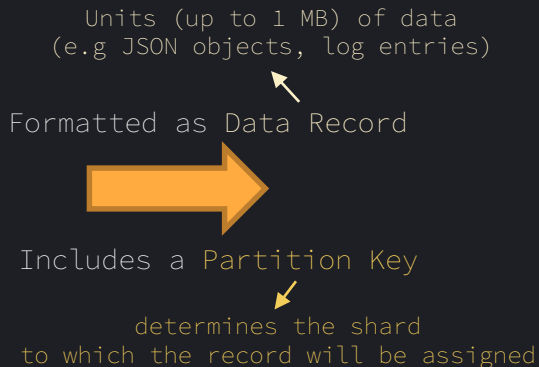
<https://t.me/learningnets>

Amazon Kinesis Data Streams



Producer

Device or application that generates and writes data to Kinesis data stream (data source)



Data Stream

Amazon Kinesis Data Streams

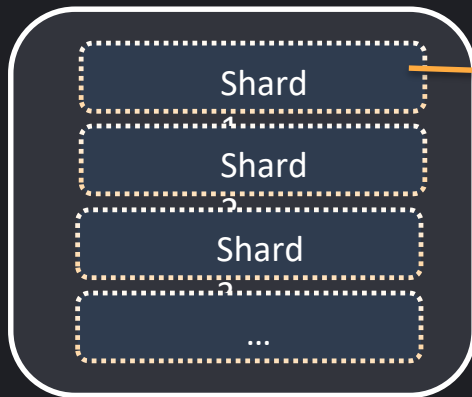


Producer

Writes data to Kinesis data stream

Data Record

Partition Key
determines the shard



Data Stream

Shards

Basic units of capacity

- 1 shard:
- 1 MB/s in (in-throughput)
1000 records/s
 - 2 MB/s out (out-throughput)
- ⇒ Determines its overall capacity for ingesting and processing data.
- ⇒ Efficient parallel processing

Durability

Configurable retention period
(default 24 hours up to 365 days)

Replicated across multiple
Availability Zones

⇒ Resilient to failure

Data is immutable

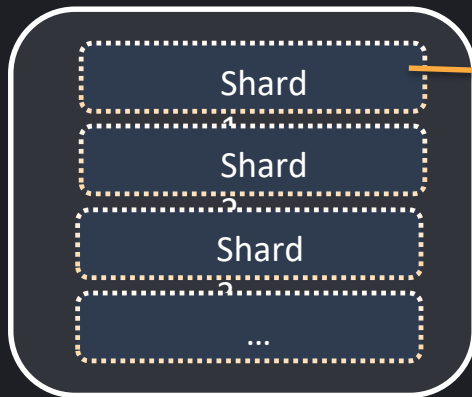
Amazon Kinesis Data Streams



Producer

Writes data to Kinesis data stream

Data Record
Partition Key
determines the shard



Data Stream

Scalability

Add and remove shards dynamically

Auto Scaling to automatically scale

⇒ Elastically scale up and down

Capacity mode

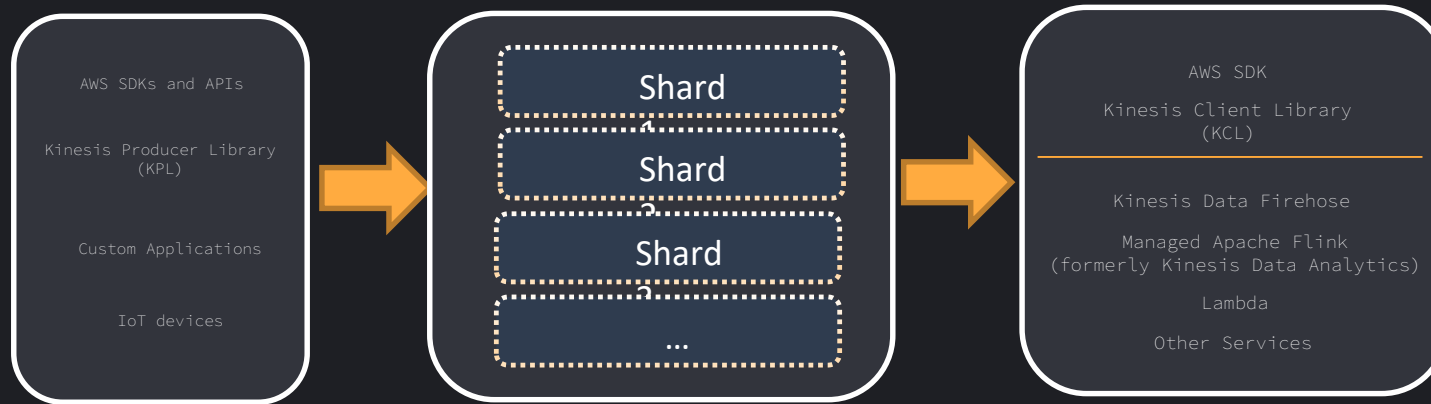
Provisioned mode

Must specify the number of shards
Can increase and decrease the number
Pay hourly rate

On-demand mode

Automatically scales shards based on
throughput peaks over last 30 days
Default: 4 MB/s or 4,000 records/s

Amazon Kinesis Data Streams



Producer

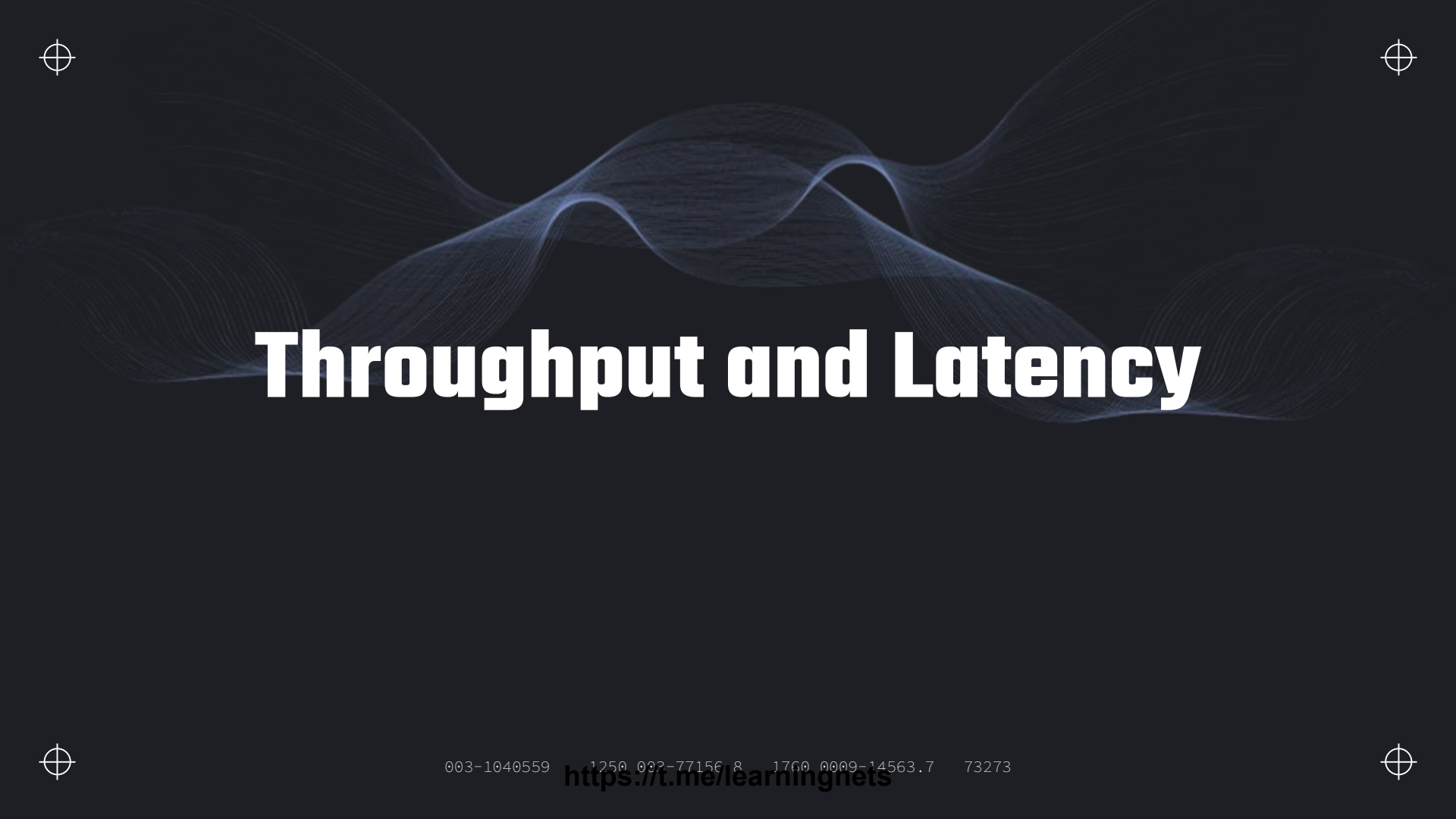
Writes data to Kinesis data stream

Data Stream

Consumer

Gets records from Amazon Kinesis Data Streams and processes them

Kinesis Data Streams application



Throughput and Latency

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Throughput

Throughput in AWS Kinesis Data Streams

- **Definition:**
Volume of data (in MB or records per second) ingested into or retrieved from a Kinesis Data Stream.
- **Measurement Units:**
Units per second (e.g., Mbps, records per second)
- **Real-World:**
Actual rate of data processing, accounting for all factors.
- **Shard-Based Scaling:**
Scalable through number of shards; each shard adds fixed capacity to stream.
- **Proportional Relationship:**
Total stream throughput directly relates to shard count.
- **Optimization Goal:**
Improving capacity to process more data within timeframe for high-volume data

Bandwidth vs. Throughput

Bandwidth in AWS Kinesis Data Streams

- Definition:
The maximum data transfer rate
- Theoretical Upper Limit:
Potential maximum for throughput

Latency

Latency and Propagation Delay

- **Definition Latency:**
The time from initiating a process to the availability of the result.
- **Propagation Delay:**
Specific latency from when a record is written to when it's read by a consumer.
- **Influencing Factor:**
Significantly affected by the polling interval, or how often consumer applications check for new data.

Latency

Latency and Propagation Delay

- **Recommendation:**
Poll each shard once per second per consumer application to balance efficiency and avoid exceeding API limits.
- **Kinesis Client Library (KCL) Defaults:**
Configured to poll every second, keeping average delays below one second.
- **Reducing Delays for Immediate Data Needs:**
Increase the KCL polling frequency for quicker data access, with careful management to avoid API rate limit issues.

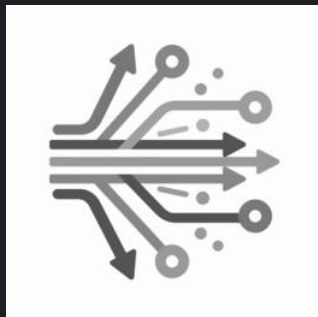


Enhanced Fan-Out

003-1040559 1250 002-77156.8 1760 0009-14563.7 73273

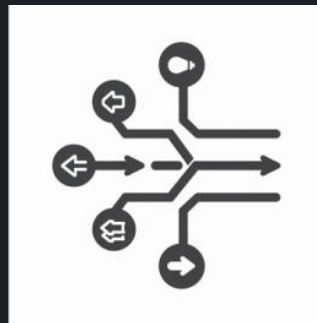
<https://t.me/learningnets>

Fan-Out



- A **single stream** distributes data to **multiple consumers**
- Distribute data to different applications

Fan-In

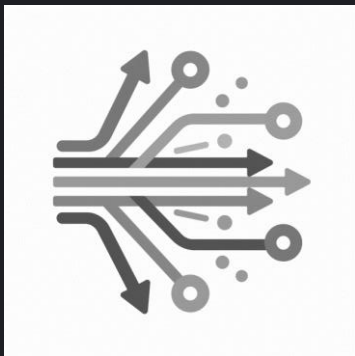


- **Multiple sources** converge towards a **single destination**.
- Destination can be another stream or another storage system.
- Combine data from different sensors into a single stream

VS



Enhanced Fan-Out



- **Traditionally:** Shared Through-put (*standard consumer*)
⇒ Potential bottleneck
- **Solution:** Enhanced fan-out
⇒ Push data through HTTP/2
⇒ Each registered consumer its own dedicated read throughput
(up to 20 consumers)

⇒ Each consumer: Up to 2 MB/second per shard

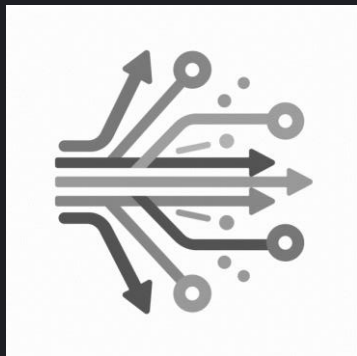
⇒ Add more consumers without creating bottleneck

⇒ E.g. 10 consumers ⇒ 20 MB/s instead of 2 MB/s





Enhanced Fan-Out - Benefits

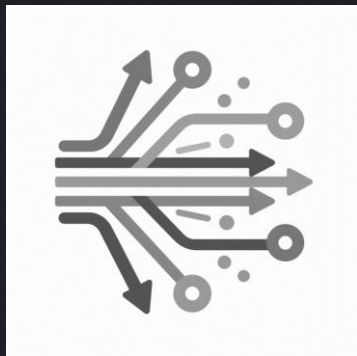


- **Increased Throughput:**
Each consumer gets its own dedicated bandwidth, allowing for faster data processing.
- **Improved Scalability:**
The system can handle more concurrent consumers without performance degradation.
- **Reduced Latency:**
Improved latency (~70ms)
- **Simplified Application Development:**
You don't need to implement complex logic to manage shared throughput between consumers.





Enhanced Fan-Out – When to Use?



- Higher Cost
- High number of consumers
- Require reduced latency





Troubleshooting & Performance in Kinesis

003-1040559 1250 032-77156.8 1760 0009-14563.7 73273

<https://t.me/learningnets>

Kinesis Data Stream

Performance Challenges for Producers:

Slow Write Rates

- 1) Problem: Service Limits & Throttling
Solution: Monitor Throughput Exceptions
- 2) Problem: Uneven data distribution to shards - "Hot Shards"
Solution: Effective partition key strategy
- 3) Problem: High throughput and small batches can be inefficient
Solution: Batch records to aggregate multiple records

Kinesis Data Stream

Performance Challenges for Consumers:

Slow Read Rates

- 1) Problem: Hitting shard read limits (per-shard limit)
Solution: Increasing shard count
- 2) Problem: Low value for maximum number of **GetRecords** per call
Solution: System-defaults are generally recommended.
- 3) Problem: Logic inside **ProcessRecords** takes longer than expected.
Solution: Change logic, test with empty records

Kinesis Data Stream

GetRecords returns empty records array

Every call to GetRecords returns a ShardIterator value (must be used in the next iteration).

ShardIterator NULL ⇒ Shard has been closed.

Empty records reasons:

- 1) No more data in the shard.
- 2) No data pointed to by the ShardIterator.

⇒ It is not an issue and usually automatically handled (Kinesis Client Library).

Kinesis Data Stream



Additional Issues

- 1) Problem: Skipped records
Solution: Might be due to unhandled exceptions.
Handle all exceptions in **processRecords**
- 2) Problem: Expired Shard Iterators.
DynamoDB table used by Kinesis does not have enough capacity to store the data. Large number of shards.
Solution: Not called GetRecords for more than 5min.
Increase write capacity to shard to table.

Kinesis Data Stream



Additional Issues

3) Problem: Consumers falling behind

Solution: Increase retention

Monitor **GetRecords.IteratorAgeMilliseconds** or **MillisBehindLatest** metrics

- Spikes ⇒ API failures (transient)
- Steady increases ⇒ Limited resources or processing logic

Kinesis Data Stream

Additional Issues

- 4) Problem: Unauthorized KMS Master Key Permission Error
- Solution: Writing to an encrypted stream without the necessary permissions on the KMS master key. Ensure you have the correct permissions via AWS KMS and IAM policies



Amazon Kinesis Data Firehose

003-1040559 1250 032-77156.8 1760 0009-14563.7 73273

<https://t.me/learningnets>



01

02

03

04

05

06



Overview

01

02

03

04

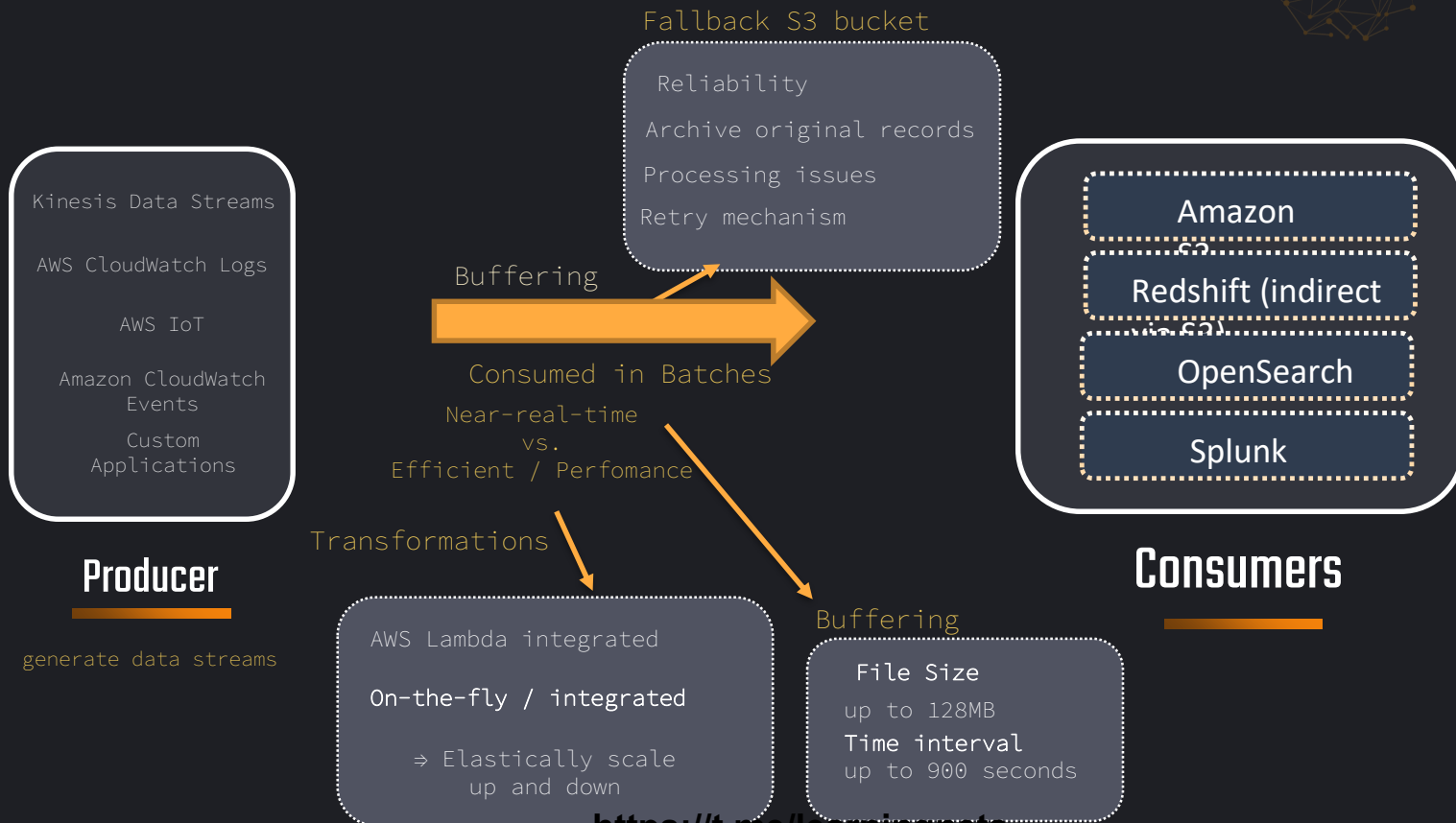
05

06

- Fully Managed Service:
Automates data streaming and loading, reducing the need for manual setup and administration.
- Effortless Data Handling:
Captures, transforms, and loads data streams into AWS data stores and analytic tools with minimal effort.
- Automatic Scaling:
Dynamically adjusts to the volume of incoming data, providing seamless scalability.



Amazon Kinesis Data Firehose





01

02

03

04

05

06



Key Features

01

02

03

04

05

06

- **Near Real-Time Processing:**
Buffers based on time and size, balancing prompt data handling with the efficiency of batch deliveries.
- **Broad Data Format Support:**
Handles multiple data formats and conversions (e.g., JSON to Parquet/ORC).
- **Data Security and Transfer Efficiency:**
Compresses and encrypts data to enhance security during transfer.
- **Real-Time Analytics and Metrics:**
Ideal for scenarios requiring quick data analysis, like log or event data capture.





Pricing

Consumption-Based Pricing:

Costs are tied to the volume of data processed, suitable for various operational scales.

01

02

03

04

05

06



01

02

03

04

05

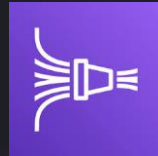
06



Kinesis Data Streams



Kinesis Firehose



VS

- Manual shard setup
- Coding consumers/producers
- Real-time (200ms/70ms latency)
- Data Storage up to 365 days

- Fully managed experience
- Focus on delivery (efficient)
- Near-real-time
- Ease of use

<https://t.me/learningnets>



Managed Service for Apache Flink

003-1040559 1250 032-77156.8 1760 0009-14563.7 73273

<https://t.me/learningnets>

What is Amazon Managed Service for Apache Flink?

- o Fully Managed Service:
For querying and processing of data streams.
- o Real-time streaming processing and analytics:
Scalable real-time analytics using Flink

Real-time monitoring

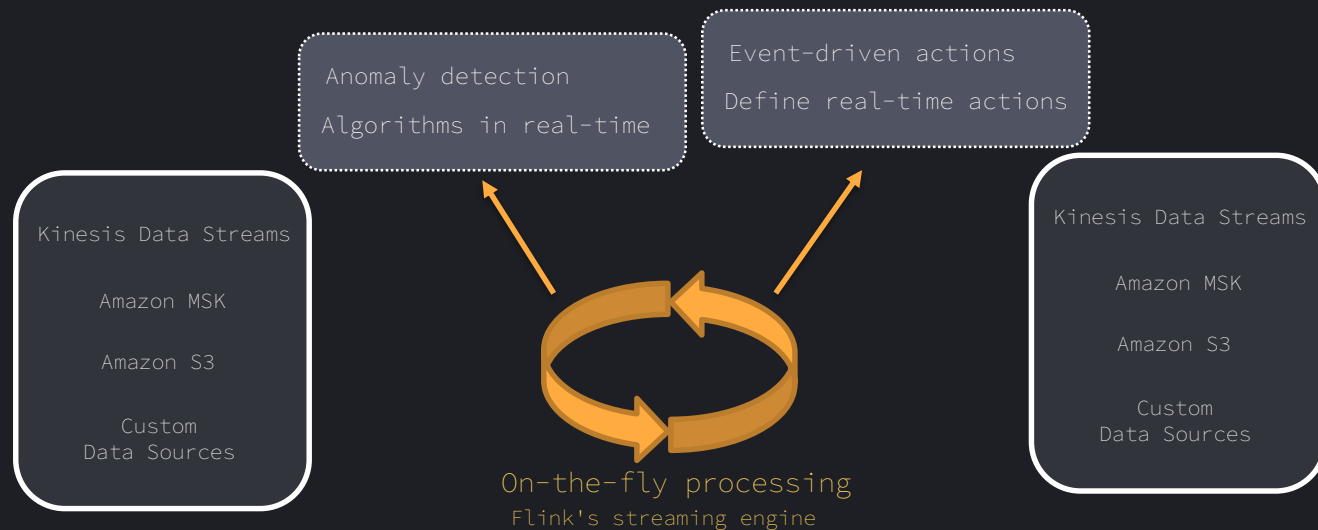
Real-time website traffic

Streaming ETL

SQL
Python
Scala
Java

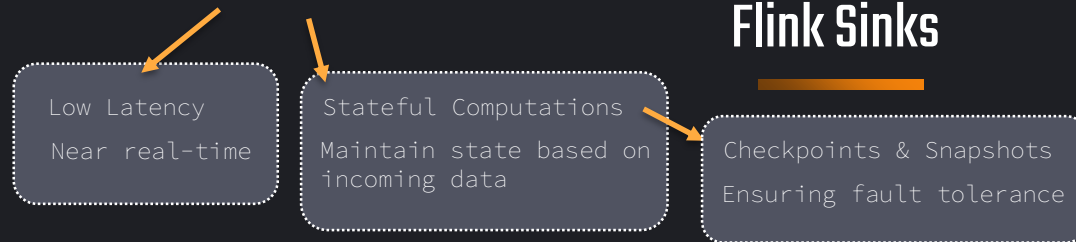
- o Apache Flink under the hood:
Open-source stream processing framework managed by AWS
- o Serverless & Scalable

AMS Flink



Flink Sources

Some data streams
Minimal Code





01

02

03

04

05

06



Pricing

- Pay-as-You-Go
Consumption based
- Kinesis Processing Units (KPU)
Charges are based on the number of KPUs
(1 KPU has 1 vCPU and 4 GB of memory)
- Application Orchestration
Each application requires one additional KPU for orchestration.
- Storage and Backups
per GB per month
- Automatic Scaling
Number of KPUs automatically scaled based on needs.
Manually provisioning possible.
- AMS Flink Studio (Interactive Mode)
Charged for two additional KPUs

01

02

03

04

05

06





Amazon Managed Streaming for Apache Kafka (MSK)



01

02

03

04

05

06



Overview

- Managed Kafka Service
Streamlines setup and management of Apache Kafka.
- Purpose
Enables processing of streaming data in real-time.

01

02

03

04

Architecture

- Kafka Brokers:
Servers that manage the publishing and subscribing of records
- ZooKeeper Nodes:
Coordinates Kafka brokers and manages cluster metadata.

05

06





01

02

03

04

05

06



High Availability & Storage

- Multi-AZ Deployment
Clusters distributed across Availability Zones for fault tolerance.
- EBS Storage
Utilizes Elastic Block Store for durable message storage, supporting seamless data recovery.

02

03

04

Custom Configuration & Scalability

- Message Size Flexibility
Configurable message sizes (up to 10 MB), more flexibility than Kinesis (1 MB limit)
- Scalable Clusters
Easily adjustable broker counts and storage sizes to handle varying workloads.

05

06





01

02

03

04

05

06



Producers & Consumers

- Producers
Applications that send data to MSK clusters.
- Consumers
Applications or services retrieving and processing data from MSK.

02

03

04

Comparison with Amazon Kinesis

- Customization vs. Convenience
MSK offers in-depth configurability; Kinesis offers easier setup with fixed limits.
- Message Handling
MSK supports larger message sizes, critical for specific use cases.

05

06





MSK

- More granular control, more management
- More complex pipelines
- Message size:
Up to 10MB (Default 1 MB)
- Topics & Partitions

Kinesis

- More Managed Experience
- Straight-forward setup
- Message size:
1 MB Limit
- Streams & Shards

VS





MSK

- Encryption
In-flight TLS encryption
OR plain text possible
At Rest:
Supports KMS encryption
- Access Control:
 - Mutual TLS
 - SASL/SCRAM Username/password authentication mechanism, also relying on Kafka ACLs.
 - IAM Access Control authentication and authorization using IAM

Kinesis

- Encryption
TLS encryption in-flight by default
At Rest:
Supports KMS encryption
Straight-forward setup
- Access Control:
 - Uses IAM policies for both authentication and authorization

VS





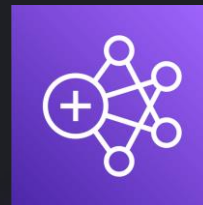
Amazon EMR

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Amazon EMR

- Fast, distributed data processes.
- Uses big data frameworks:
 - Apache Hadoop, Apache Spark
 - For petabyte scale processing
 - Glue is easy to use but for heavy workloads (petabytes) not that suitable
 - Migration from existing on-premise resources



Create cluster [Info](#)

▼ **Name and applications - required** [Info](#)
Name your cluster and choose the applications that you want to install to your cluster.

Name

Amazon EMR release [Info](#)
A release contains a set of applications which can be installed on your cluster.

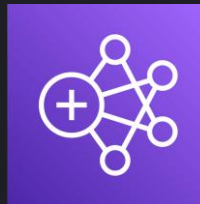
Application bundle

Spark Interactive 	Core Hadoop 	Flink 	HBase 	Presto 	Trino 	Custom 

<https://t.me/learningnets>

Amazon EMR

- Fast, distributed data processes.
- Uses big data frameworks:
 - Apache Hadoop, Apache Spark
- Cluster based architecture.
- Pricing is based on EC2 usage and hourly service rates.
- Security:
 - IAM, VPC, KMS.

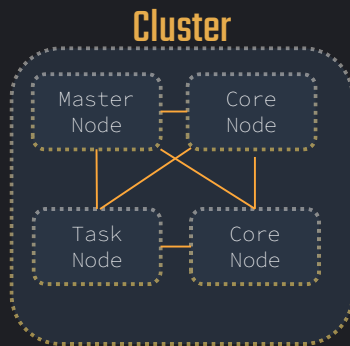
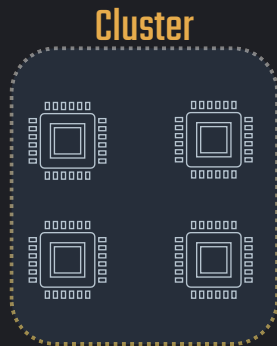


Amazon EMR

What Is Hadoop

- Distributed storage and processing framework.
 - 1) Hadoop Distributed File System (HDFS)
 - 2) MapReduce

EMR Cluster Structure



- **x86-based instance:**
Versatile & traditional choice
- **Graviton-based instance:**
Balance of compute & memory
20% cost savings

Nodes

Master Node: Manages the cluster

Core Nodes: responsible for running tasks and storing data

Task Nodes: responsible for running tasks and storing data



Amazon EMR

Cluster Types & Storage

Amazon EMR

Cluster Types

Transient Cluster



Temporary clusters



After the job cluster terminates



Used for intermittent jobs

Long Running Clusters



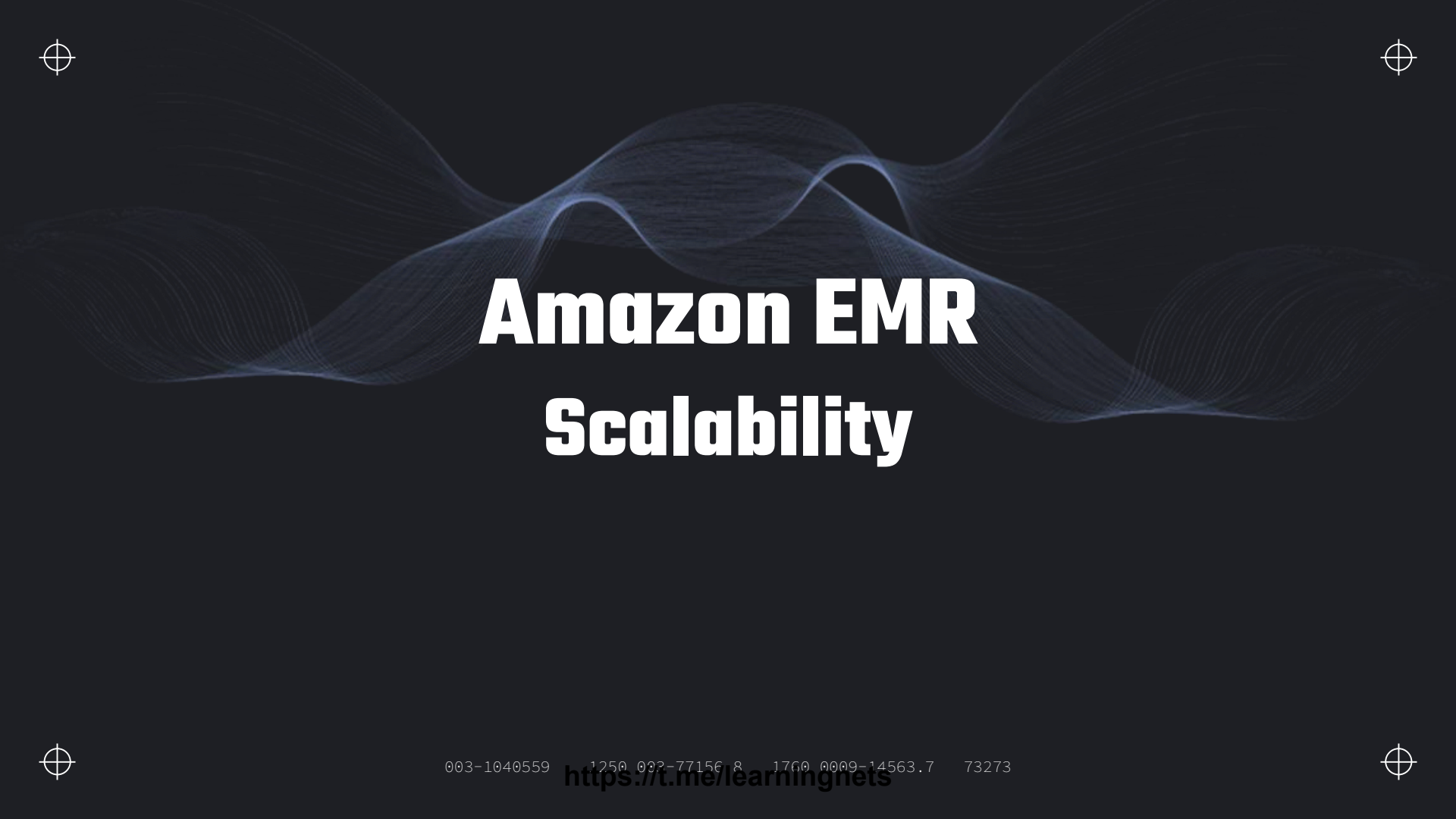
Ongoing processes



Continuously consume and process

Amazon EMR - Storage

- HDFS (Hadoop Distributed File System):
 - Location: Located on the local disks of each node.
 - Use Case: Temporary storage, non-persistent, high throughput.
- EMR File System (EMRFS):
 - Location: HDFS that allows clusters to store S3.
 - Use Case: Persistent, cost-effective, Storing input and output.
- Local File System:
 - Location: Resides on each individual EC2 instance.
 - Use Case: Storing input and output.
- EBS for HDFS:
 - Works differently than its usual.
 - Temporary storage.



Amazon EMR Scalability

003-1040559 1250 032-77156.8 1760 0009-14563.7 73273

<https://t.me/learningnets>

Amazon EMR - Scalability



Manual Scaling

You will be manually scaling the cluster.



Amazon EMR Managed Scaling

EMR takes care of automated scaling policies. Instance groups and instance fleets.



Custom Automatic Scaling

You will be define automated scaling policies. Only for instance groups.

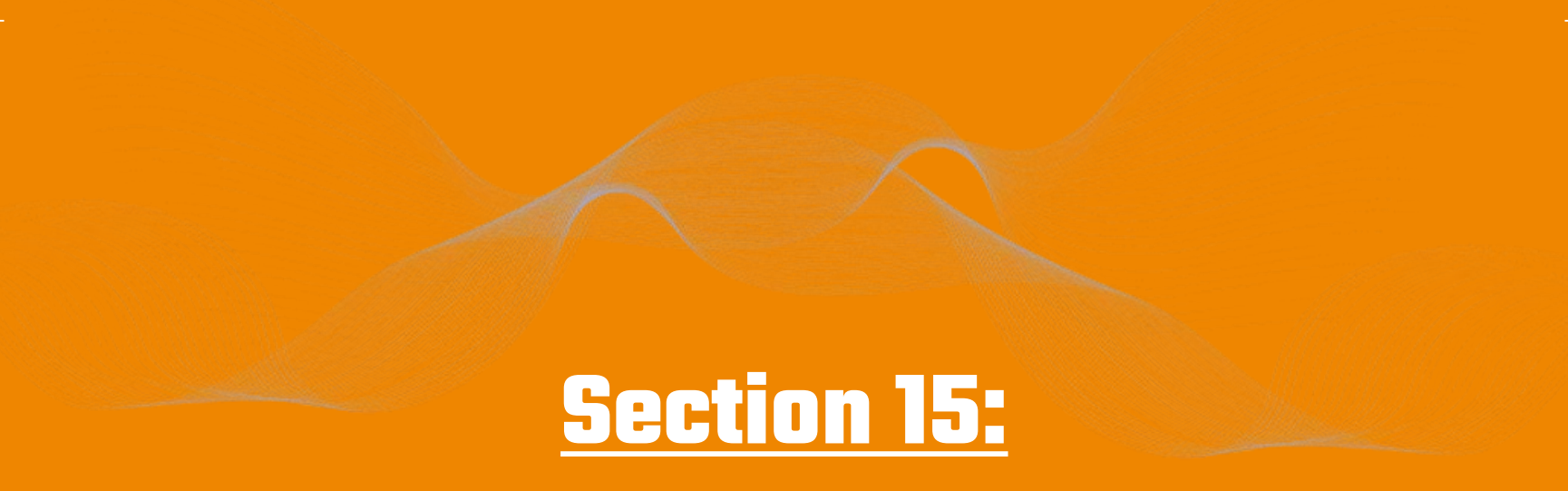
Amazon EMR – Deployment Options

EMR on EKS

- Run data frameworks on Amazon Elastic Kubernetes Service.
- Simplifies container based big data processes.
- Provisioning clusters not needed.

EMR Serverless

- Serverless runtime environment.
- It provides fast job startup and high availability.
- Pay for what you use.



Section 15: AWS Storage Services

003-1040559 1250.000-77156.8 1750.0009-14563.7 73273

<https://t.me/learningnets>

The background is a dark blue-grey gradient. It features several thin, light blue wavy lines that flow across the frame. Scattered throughout are numerous small, light blue dots of varying sizes. Some of these dots are connected by thin vertical lines, creating a sense of depth and structure. In the center, the word "Partitioning" is written in a large, bold, white sans-serif font.

Partitioning

003-1040559 1250 032-77156 8 1760 0009-14563.7 73273

<https://t.me/learningnets>



Overview

- Partitioning data through a combination of bucket and folder structure.



Importance of Folder Structure

Data Management

- Simplifying tasks such as data retention and archival.
 - Facilitating easy archiving or deletion of older data partitions
- Improving query performance by allowing queries to process only relevant data subsets.
 - Reducing the amount of data scanned in queries, leading to cost savings.





Implementation of Partitioning

Organizing Data



- Organizing data into folders and subfolders, including the use of buckets.
- Enabling the use of Glue crawlers to automatically create partition keys.

Partitioning Examples

- Time-based partitioning example with S3 keys.

`2021 /month=11/day=05/filename.txt`

- Introduction of metadata tagging for custom metadata attachment.





Importance of Maintaining **The Glue Catalog** for Managing Metadata and Defining Partitions

<https://t.me/learningnets>

Amazon Athena



Query using Athena.

Amazon Athena



Query using Athena

Data

Data source: AwsDataCatalog

Database: retail_db

Tables and views: Create Filter tables and views

Tables (5):

- nikolai_12343_test
- parquet
- raw_sales
- sales
- sales_sources

Views (0)

Query 1: X Query 2: X Query 3: X Query 4: X

1 SELECT * FROM "retail_db"."parquet" limit 10;

SQL Ln 1, Col 1

Run again Explain Cancel Clear Create

Query results Query stats

Completed

Results (10)

Search rows

#	date	product_id	quantity
1	2023-02-02 22:36:09.000	P319	2
2	2023-03-11 10:02:15.000	P896	20
3	2023-12-14 18:06:50.000	P157	14



Storage Classes and Lifecycle configuration



Storage Classes

003-1040559 1250 032-77156 8 1760 0009-14563.7 73273

<https://t.me/learningnets>

S3 Storage Classes



S3 Standard

S3 Intelligent-Tiering

Amazon S3 Express One Zone

S3 Standard-Infrequent Access

S3 One Zone-Infrequent Access

S3 Glacier Instant Retrieval

S3 Glacier Flexible Retrieval

S3 Glacier Deep Archive

S3 Storage Classes



Amazon S3 Standard (S3 Standard)

- It is best for storing frequently accessed data.
- It delivers low latency and high throughput.
- Is appropriate for

Cloud applications

Dynamic websites

Content distribution

Big data analytics

Mobile and gaming
applications

S3 Storage Classes



S3 Intelligent-Tiering

- Best for unknown or changing access patterns.
- Automatically moves data to the most cost-effective access tier based on access frequency

Frequent
access tier

Infrequent
access tier

Archive
Access tier

S3 Storage Classes



Amazon S3 Express One Zone

- Is a high-performance, single-Availability Zone storage class.
- It can improve data access speeds by 10x and reduce request costs by 50% compared to S3 Standard.
- Data is stored in a different bucket type—an Amazon S3 directory bucket

S3 Storage Classes



S3 Standard-Infrequent Access (S3 Standard-IA)

- Is best for data that is accessed less frequently, but requires rapid access when needed.
- ideal for long-term storage, backups, and as a data store for disaster recovery files.

S3 One Zone-Infrequent Access (S3 One Zone-IA)

- Is best for data that is accessed less frequently, but requires rapid access when needed.
- Stores data in a single AZ.
- Ideal for storing secondary backup copies of on premises data

S3 Storage Classes

S3 Glacier Storage Classes

- Are purpose-built for data archiving.



S3 Glacier Instant
Retrieval storage



S3 Glacier
Flexible Retrieval



S3 Glacier Deep
Archive

S3 Storage Classes



S3 Glacier Instant
Retrieval storage

- Is best for long-lived data that is rarely accessed and requires retrieval in milliseconds.
- Delivers the fastest access to archive storage
- Ideal for archive data that needs immediate access.

S3 Storage Classes



S3 Glacier
Flexible Retrieval

- Is best for archive data that is accessed 1–2 times per year and is retrieved asynchronously.
- It is an ideal solution for backup, disaster recovery, offsite data storage needs
- Configurable retrieval times, from minutes to hours, with free bulk retrievals.

S3 Storage Classes



S3 Glacier Deep
Archive

- Is the cheapest archival option
- Supports long-term retention and digital preservation for data that may be accessed once or twice in a year.
- Objects can be restored within 12 hours.



Lifecycle configuration

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Lifecycle configuration



- Lifecycle configuration is a set of rules that define actions that Amazon S3 applies to a group of objects.

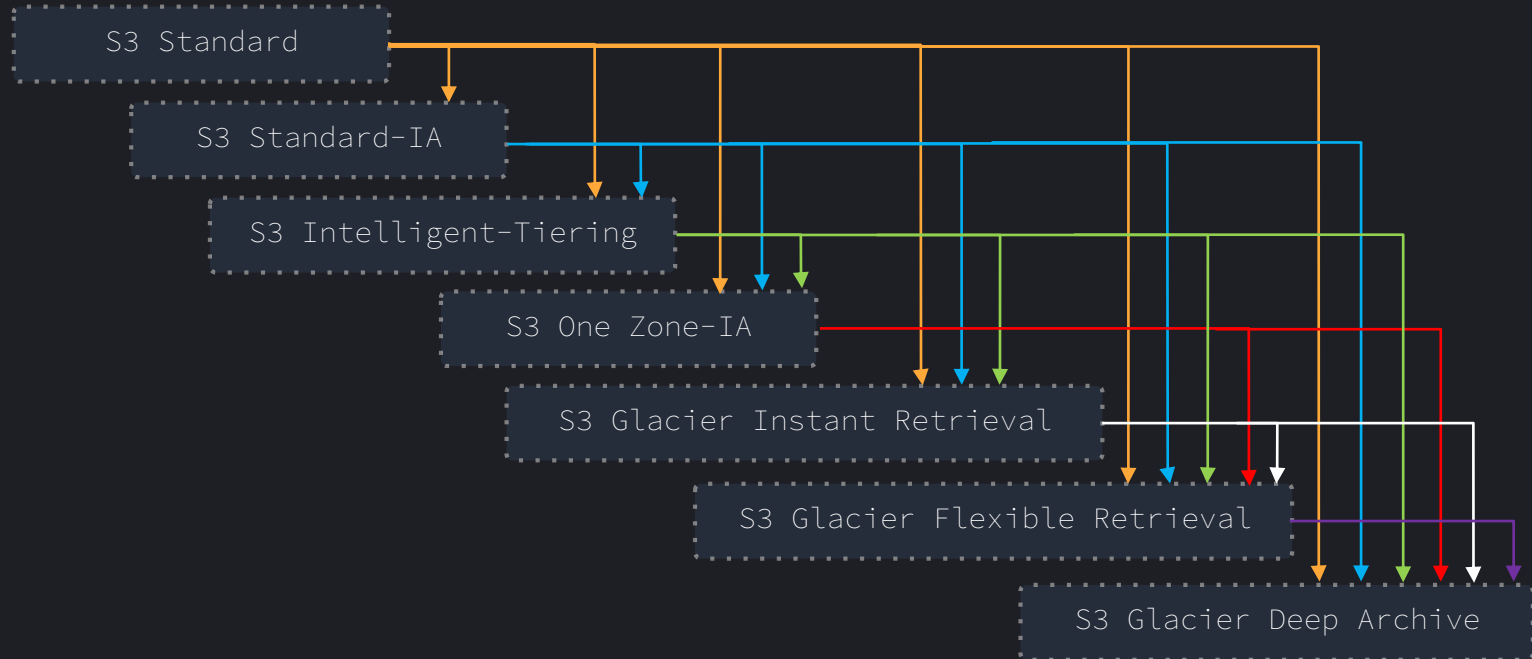
Transition actions

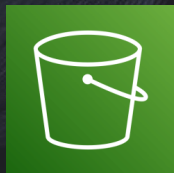
Expiration actions

Lifecycle configuration



Supported lifecycle transitions





Versioning

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

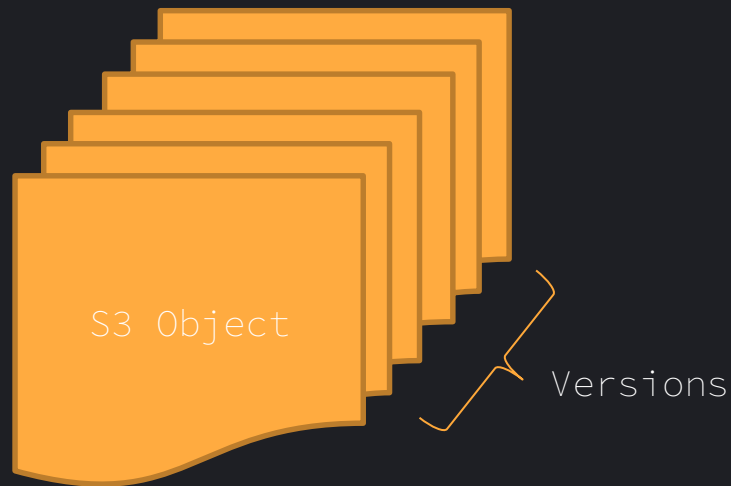
Versioning



- Helps you manage and preserve the versions of objects in your bucket.
- Useful for

Update Safety

Delete Protection



Versioning

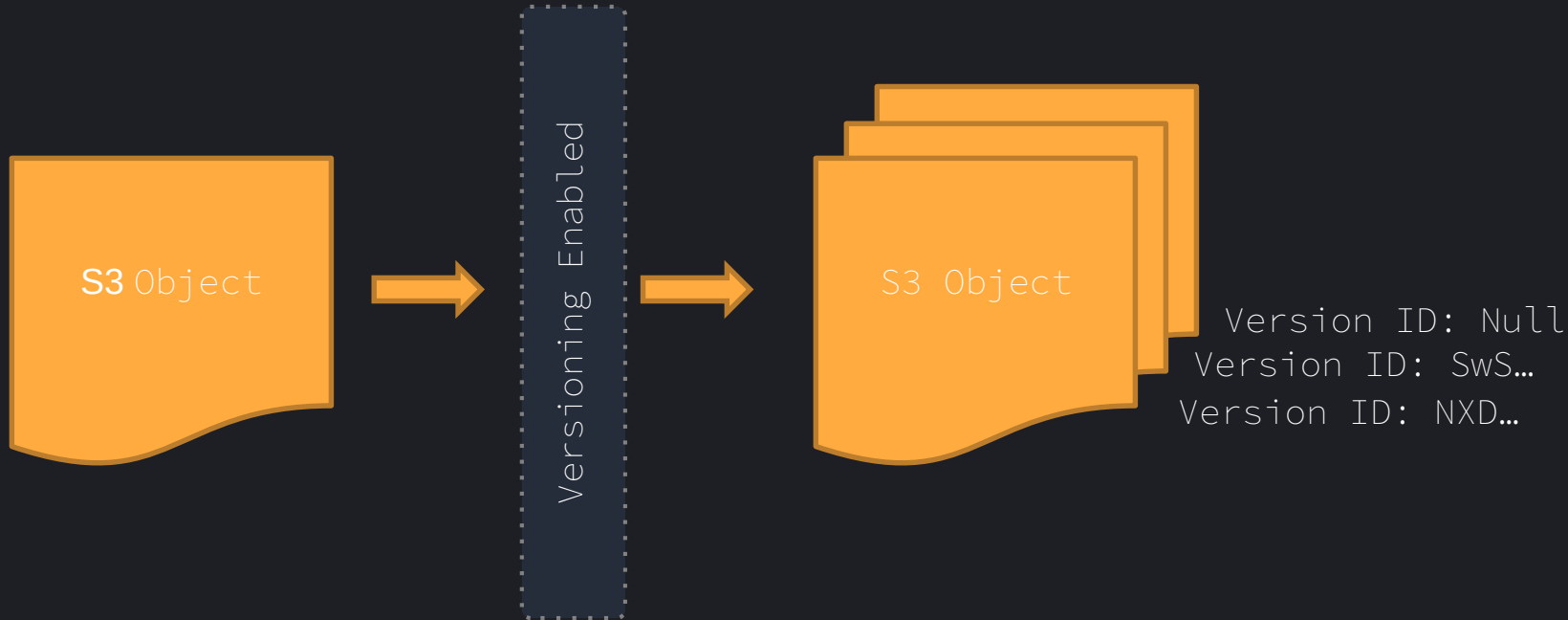
- Is disabled by default
- You can enable s3 bucket versioning using

S3 Console

AWS CLI

AWS SDK

Versioning



Versioning



my-sample-bucket-1234 [Info](#)

[Objects](#)

[Properties](#)

[Permissions](#)

[Metrics](#)

[Management](#)

[Access Points](#)

Objects (3) [Info](#)



Copy S3 URI

Copy URL

Download

Open

Delete

Actions ▾

Create folder

Upload

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

☒ Show versions

< 1 >

☐	Name ▲	Type	Version ID	Last modified	Size	Storage class
☐	home.JPG	Delete marker	zPx3ZdlNLU vhnBHHZsv CY02XyxsP WQHi	May 14, 2024, 14:51:24 (UTC+03:00)	0 B	-
☐	home.JPG	JPG	sh6hwwRAr 5N4az9UhS golCo0PZd wzpor	May 14, 2024, 14:50:43 (UTC+03:00)	36.4 KB	Standard
☐	home.JPG	JPG	null	May 14, 2024, 14:49:45 (UTC+03:00)	36.4 KB	Standard





Encryption and Bucket Policy



Encryption

003-1040559 1250 032-77156 8 1760 0009-14563.7 73273

<https://t.me/learningnets>

Encryption

Encryption in transit



Encryption

Encryption at rest

- Data is encrypted and then stored as it is.
- Can be done at **Server Side** or **Client Side**.
- All buckets have encryption at rest configured by default.



Encryption

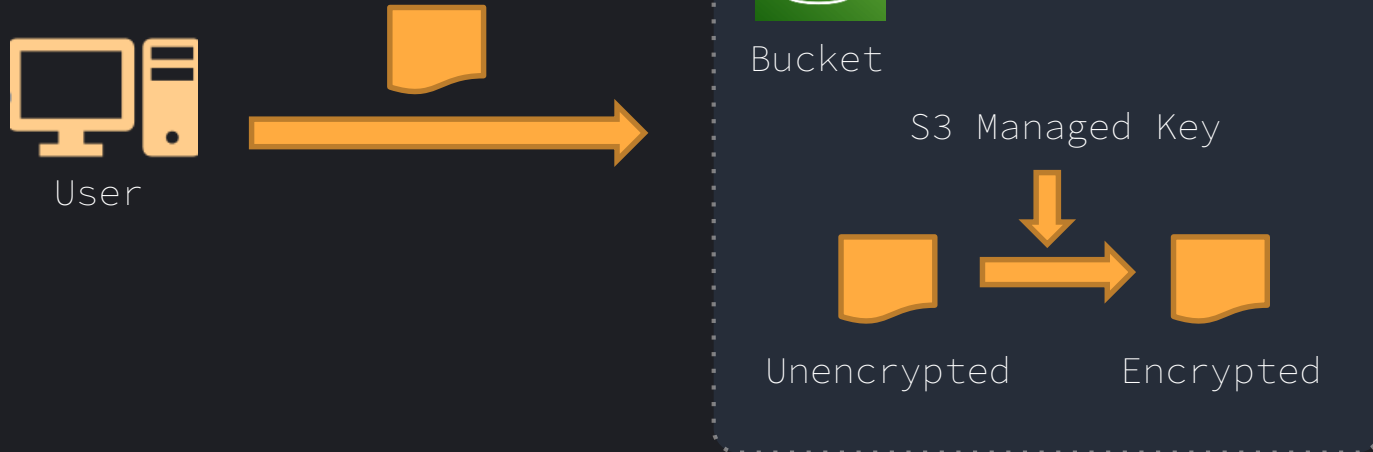


Server Side Encryption methods

- Server-side encryption with Amazon S3 managed keys (SSE-S3)
- Server-side encryption with AWS KMS (SSE-KMS)
- Dual-layer server-side encryption with AWS KMS keys (DSSE-KMS)
- Server-side encryption with customer-provided keys (SSE-C)

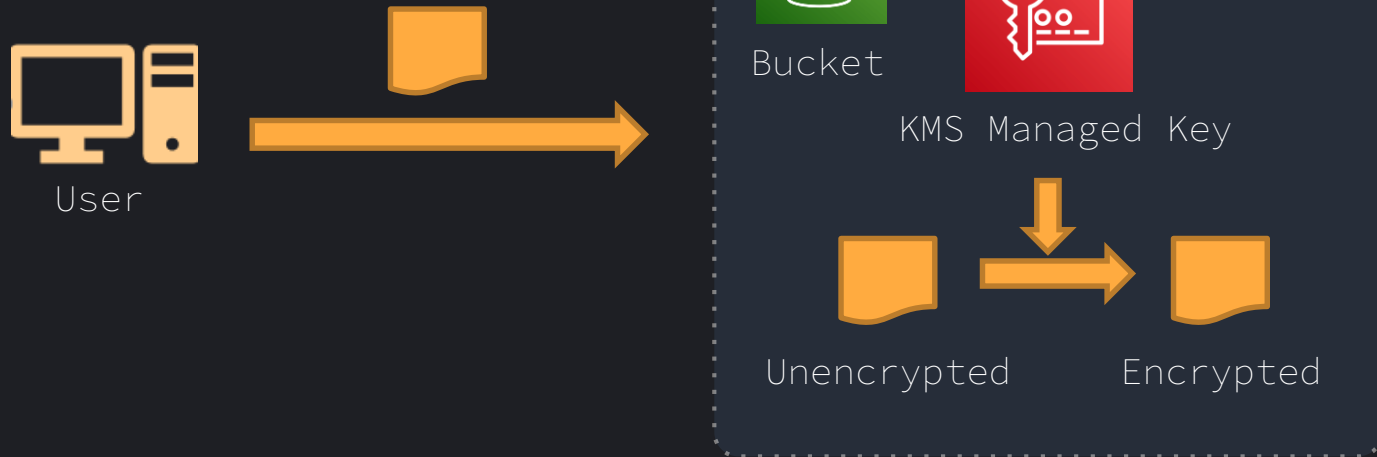
Encryption

Server-side encryption with Amazon S3 managed keys (SSE-S3)



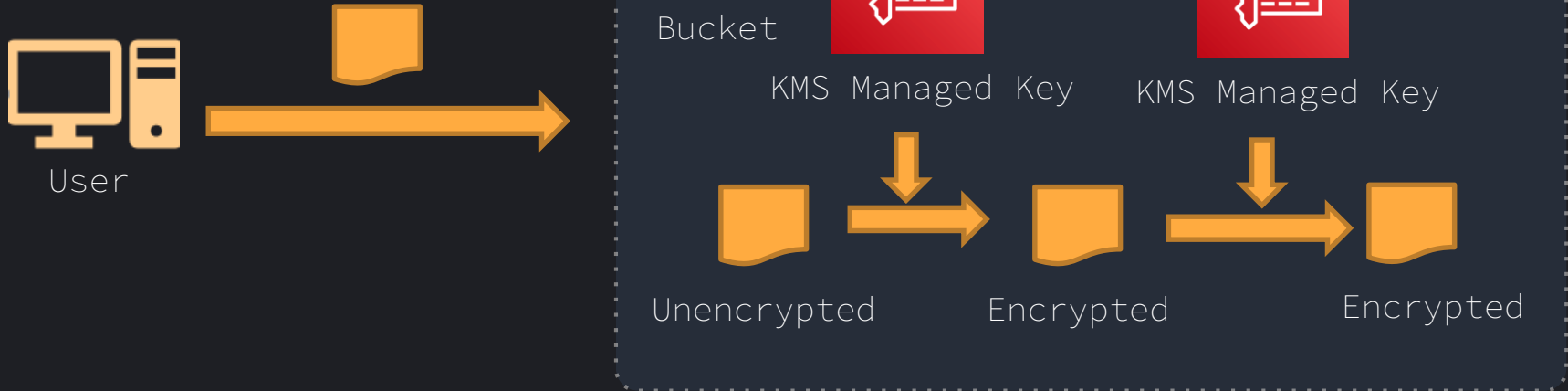
Encryption

Server-side encryption with AWS KMS (SSE-KMS)



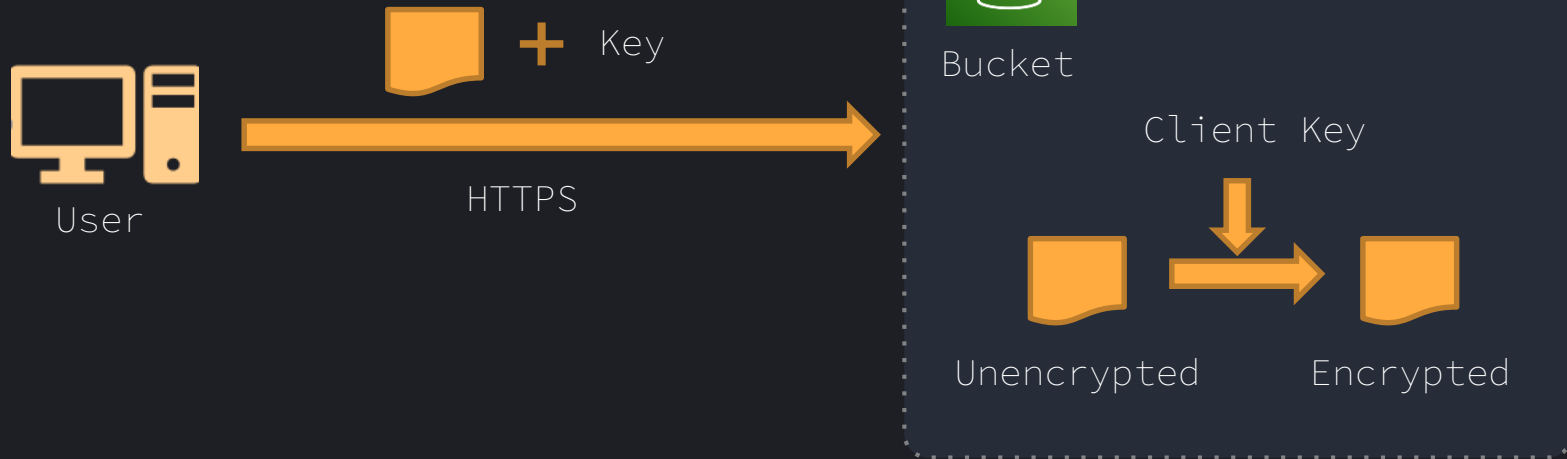
Encryption

Dual-layer server-side encryption with AWS KMS keys (DSSE-KMS)



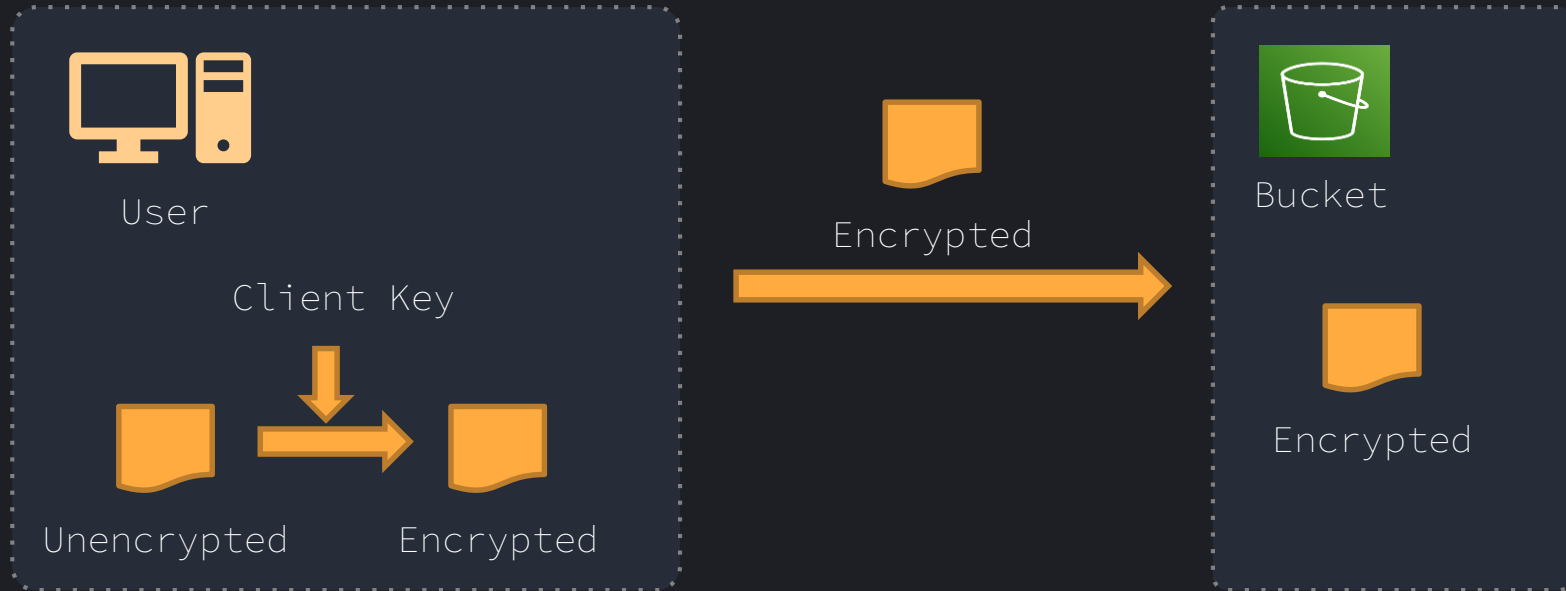
Encryption

Server-side encryption with customer-provided keys (SSE-C)



Encryption

Client-side Encryption





Bucket Policy

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Bucket Policy



- Defines permissions and access control for objects within an Amazon S3 bucket.
- Is written in JSON format.

Specifies

- Who (which users or services) can access the bucket and
- What actions they can perform on the bucket and its objects.

```
1  {
2    "Version": "2012-10-17",
3    "Statement": [
4      {
5        "Effect": "Allow",
6        "Principal": "*",
7        "Action": "s3:GetObject",
8        "Resource": "arn:aws:s3:::yourbucketname/*",
9        "Condition": {
10         "IpAddress": {
11           "aws:SourceIp": "19.168.100.0/24"
12         },
13         "Bool": {
14           "aws:SecureTransport": "true"
15         }
16       }
17     ]
18   }
19 }
```




Access Points and Object Lambda

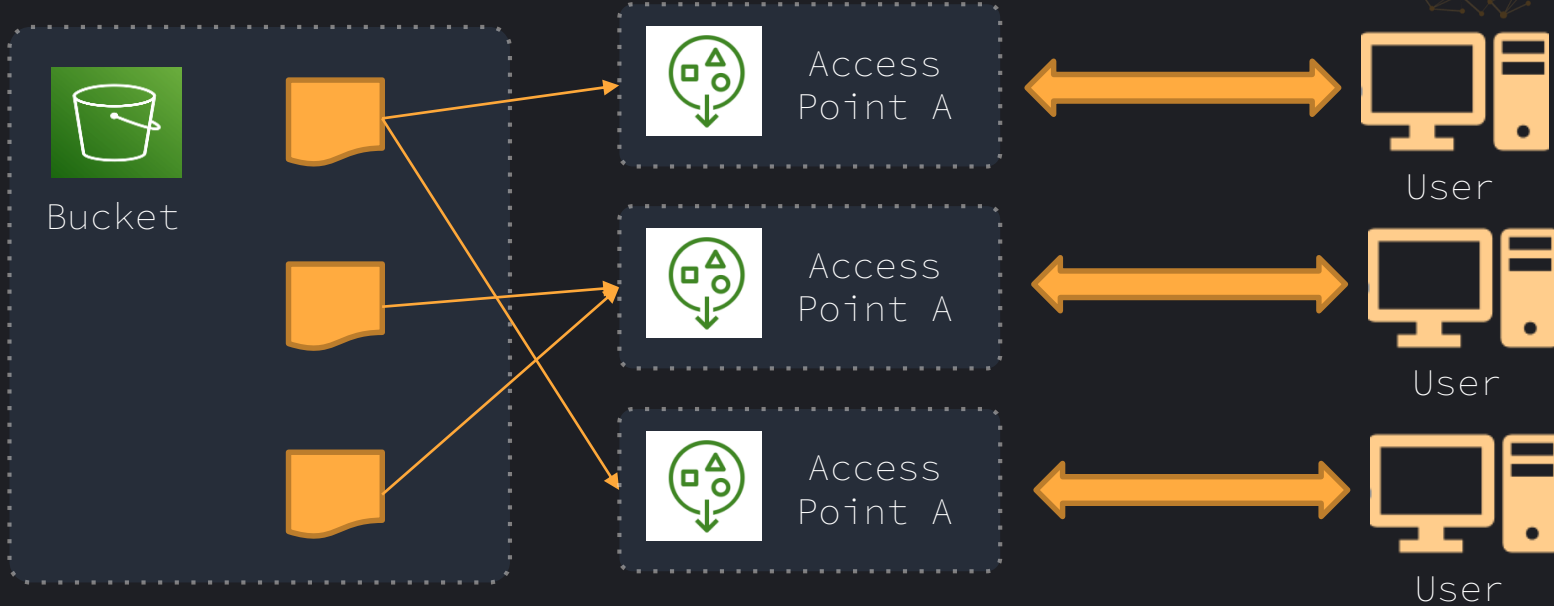


S3 Access Points

003-1040559 1250 032-77156 8 1760 0009-14563.7 73273

<https://t.me/learningnets>

Access Points



- Lets you create customized entry points to a bucket, each with its own unique policies for access control.

Access Points



Each Access Point has

- DNS name (Internet Origin or VPC Origin)
- Access point policy (similar to bucket policy)

Features

Customized
Permissions

Improved Scalability
and Organization

Enhanced
Security



Object Lambda

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Object Lambda



With Object Lambda you can

- Transform data as it is retrieved
- No need to create a separate version of the data

Use Cases

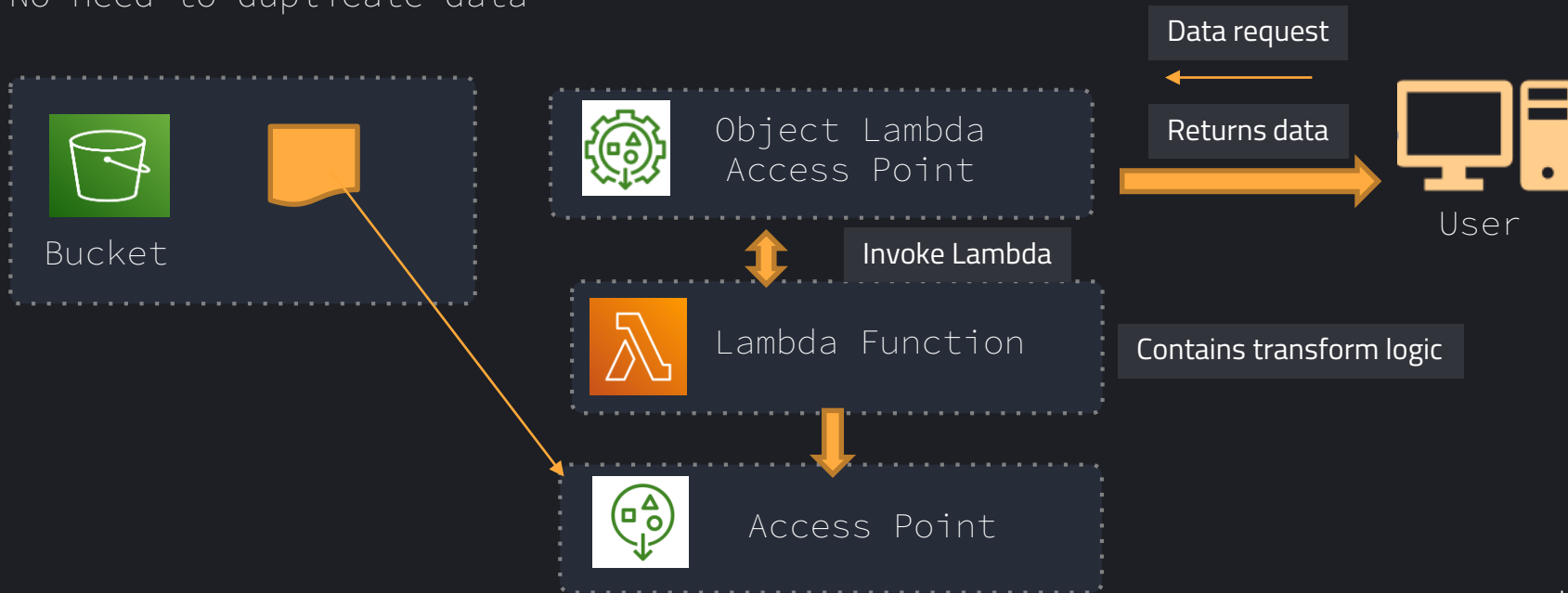
Filtering Sensitive
Information

Converting
Data Formats

Augmenting
Data

Object Lambda

- Add your own code to process data
- No need to duplicate data



Object Lambda



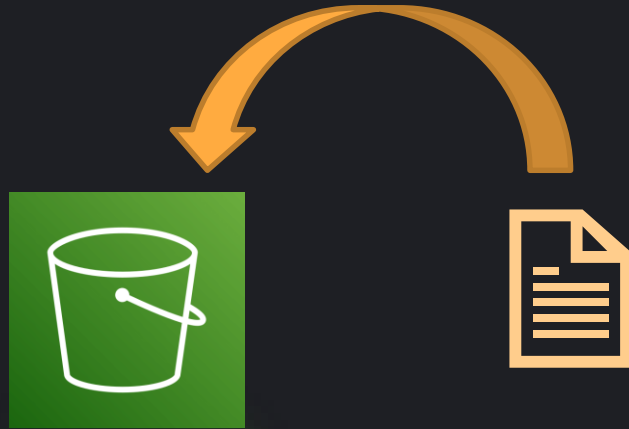
Use Cases

- `Reduct PII` (Personally identifiable information) for analytics
- `Convert data formats` such as XML to JSON
- `Augmenting Data` with information from other services



S3 Event Notification

S3 Event Notification



<https://t.me/learningnets>

S3 Event Notification

The Event can be

- New object created events
- Object removal events
- Restore object events
- Replication events





S3 Event Notification



- S3 Lifecycle expiration events
- S3 Lifecycle transition events
- S3 Intelligent-Tiering automatic archival events
- Object tagging events
- Object ACL PUT events



S3 Event Notification



S3 can send event notification messages to



Simple Notification Service topics



Simple Queue Service queues



Lambda function



EventBridge

S3 Event Notification



The Event Examples

- `s3:ObjectCreated:Put:`
Object is uploaded to the bucket using the PUT method.
- `s3:ObjectCreated:Post:`
Object is uploaded to the bucket using the POST method.
- `s3:ObjectRemoved>Delete:`
Object is deleted from the bucket.

Wildcards

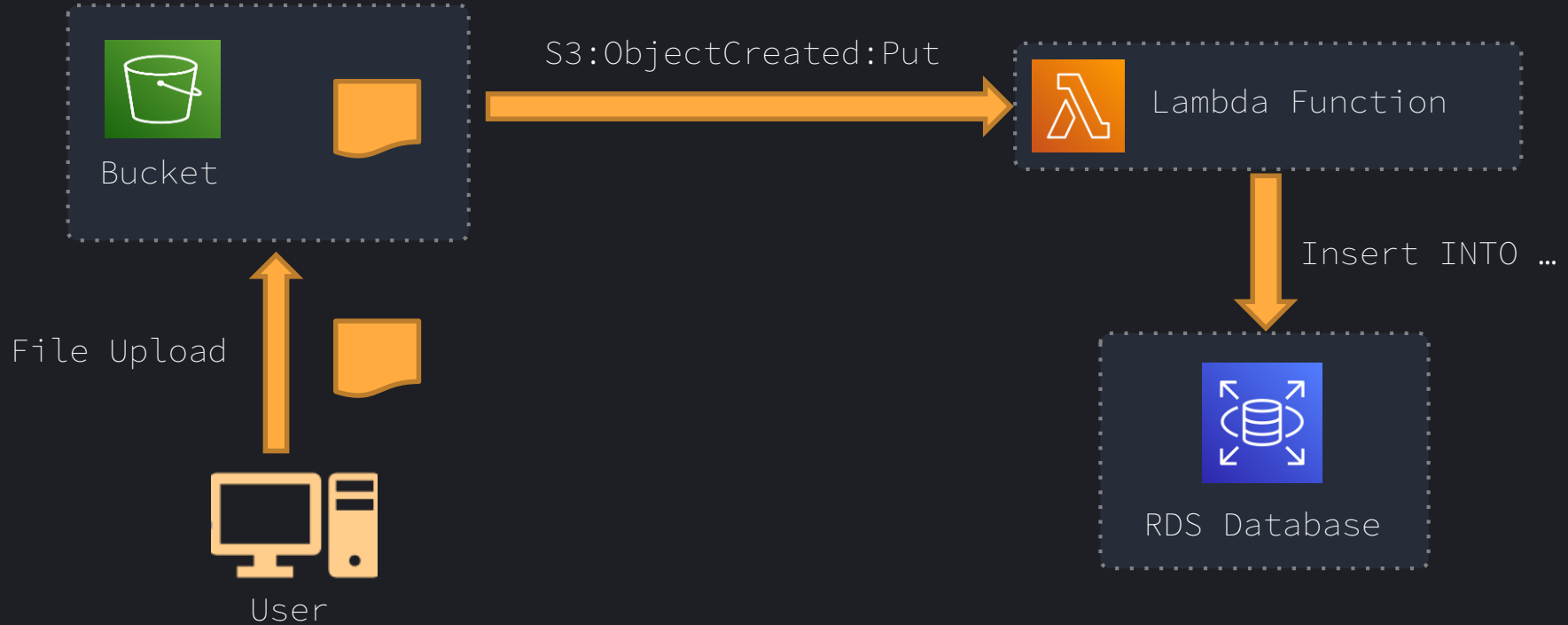
- `s3:ObjectCreated:*`
Captures all object creation events

Prefix and Suffix Filters

Filter to objects in a directory to trigger the event (e.g. uploads/images/)

Filter to file types (e.g. '.jpg') to trigger the event

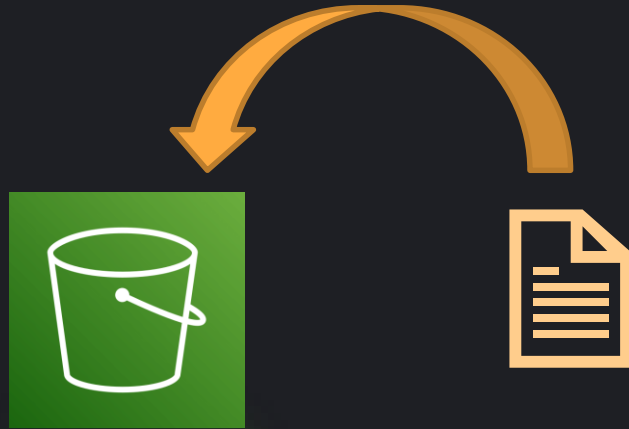
S3 Event Notification





S3 Event Notification

S3 Event Notification



<https://t.me/learningnets>

S3 Event Notification

The Event can be

- New object created events
- Object removal events
- Restore object events
- Replication events





S3 Event Notification



- S3 Lifecycle expiration events
- S3 Lifecycle transition events
- S3 Intelligent-Tiering automatic archival events
- Object tagging events
- Object ACL PUT events



S3 Event Notification



S3 can send event notification messages to



Simple Notification Service topics



Simple Queue Service queues



Lambda function



EventBridge

S3 Event Notification



The Event Examples

- `s3:ObjectCreated:Put:`
Object is uploaded to the bucket using the PUT method.
- `s3:ObjectCreated:Post:`
Object is uploaded to the bucket using the POST method.
- `s3:ObjectRemoved:Delete:`
Object is deleted from the bucket.

Wildcards

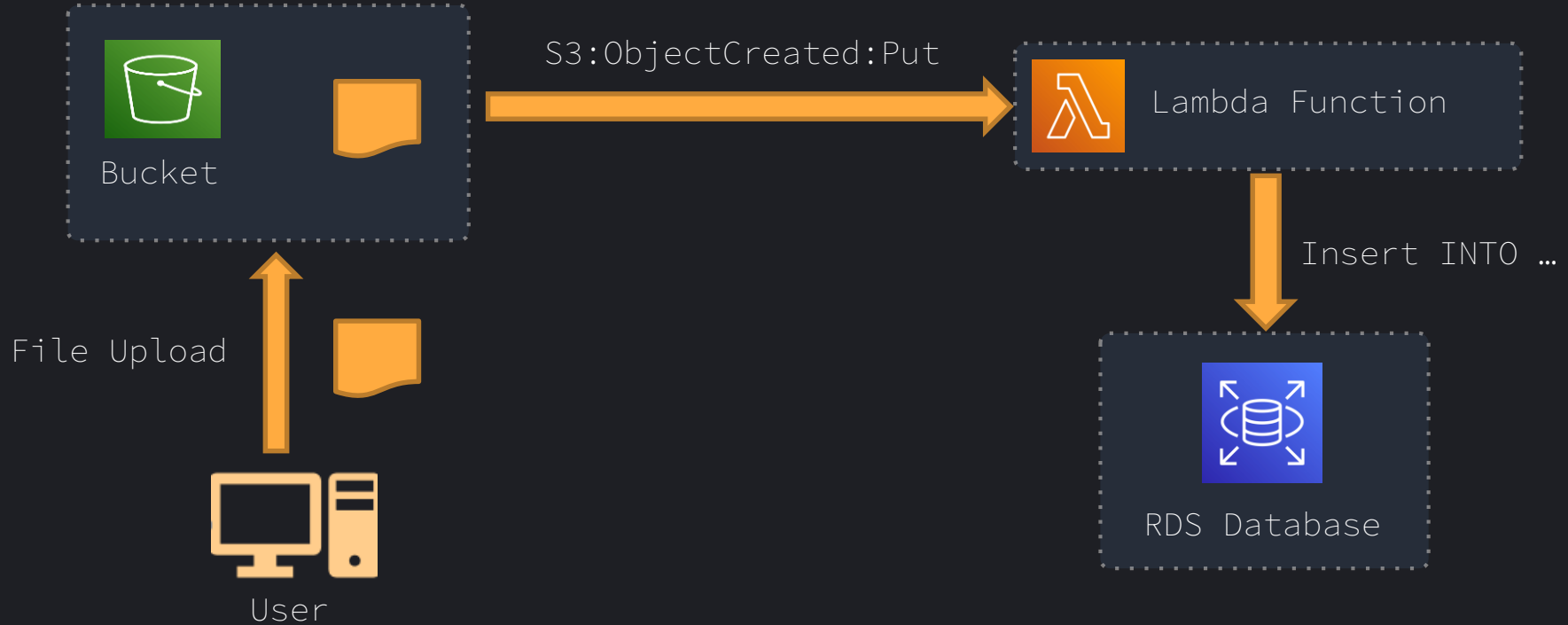
- `s3:ObjectCreated:*`
Captures all object creation events

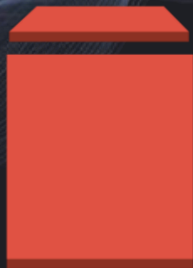
Prefix and Suffix Filters

Filter to objects in a directory to trigger the event (e.g. uploads/images/)

Filter to file types (e.g. '.jpg') to trigger the event

S3 Event Notification





EBS



003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>



AWS EBS

Single availability zone

EC2 instances

- Scalable block solution in AWS
- EBS = Elastic Block Store
- Persistent storage for a variety of data types

Highly scalable

EC2 > Volumes > vol-03aaa55cbe6dbae4e

vol-03aaa55cbe6dbae4e Refresh Actions ▼ Delete Modify

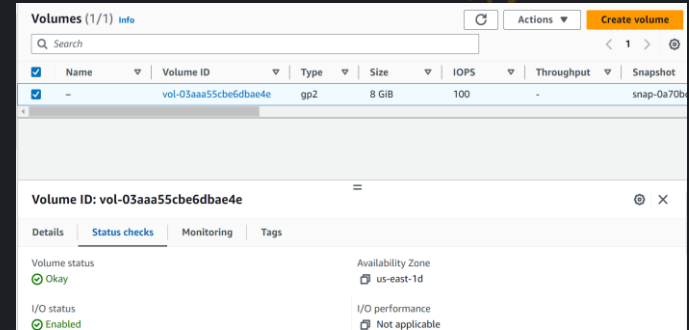
Volume ID  vol-03aaa55cbe6dbae4e	Size  8 GiB	Type gp2	Volume status  Okay
AWS Compute Optimizer finding  Opt-in to AWS Compute Op timizer for recommendations. Learn more 	Volume state  In-use	IOPS 100	Throughput -



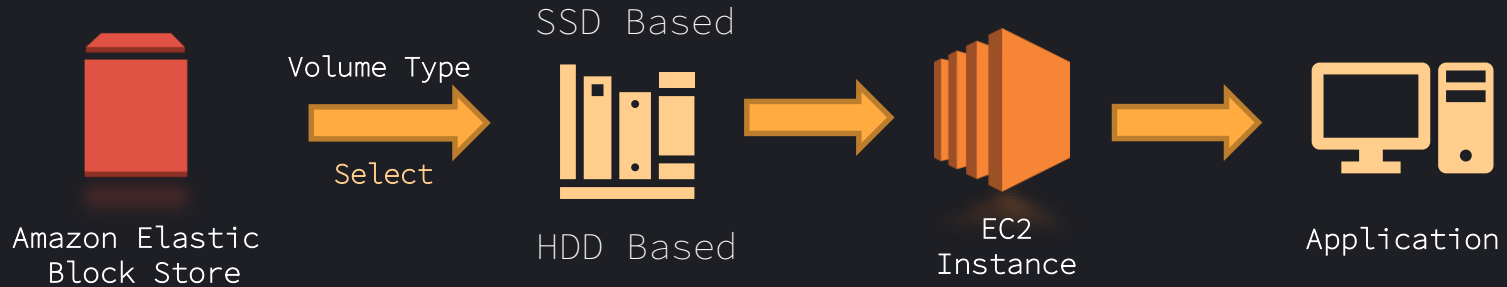
AWS EBS - Storage

Key features

- Scalability
- Durability
- Block-level storage
- Persistent storage
- High performance
- Cost effective



EBS functionality

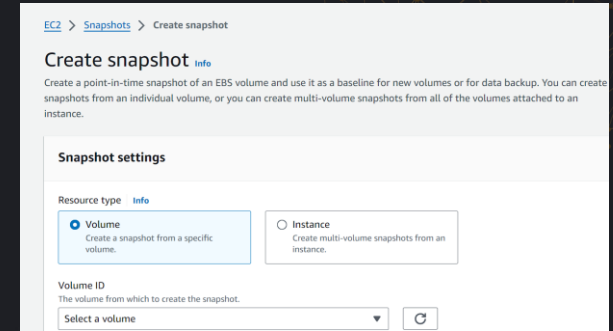


⇒ Bound to a specific AZ

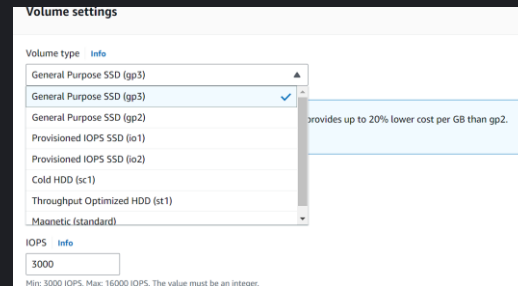
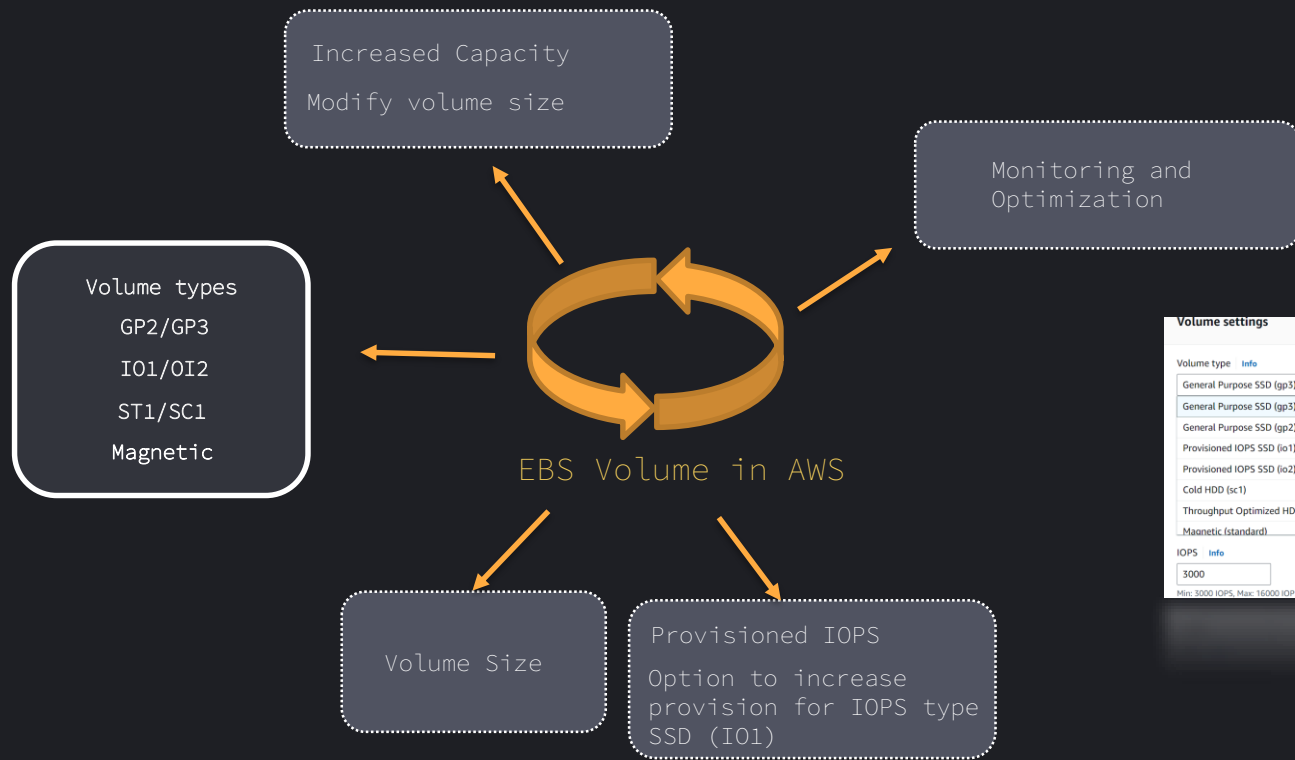
⇒ Use snapshots for different AZ or regions

EBS Snapshots

- **Definition:**
Instantaneous copies of **AWS resources**.
- **Incremental Backups:** Improves efficiency.
- **Block-Level Backup:** Detailed data capturing.
- **Data Consistency:** Before creating a snapshot, AWS ensures data consistency by temporarily pausing I/O operations.
- **S3 Compatible**
- **Lifecycle Management:** Allows you to define lifecycle policies for EBS snapshots, enabling automation of snapshot management tasks.
- **Data Recovery:** Dynamic use of snapshots.
- **Cost Management:** Reasonable cost.



EBS Capacity provisioning

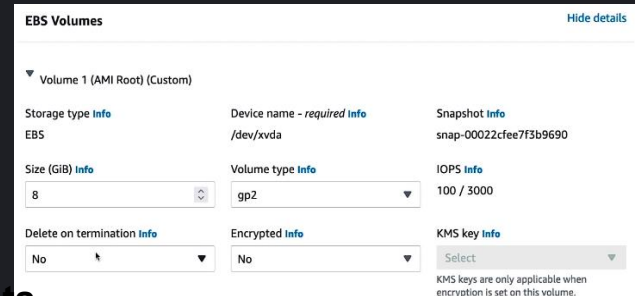


Delete on Termination

Determines whether the volume should be automatically deleted when its associated EC2 instance is terminated.

- **Enabled:**
The volume will be automatically deleted by AWS when the associated EC2 instance is terminated.
- **Disabled:**
The EBS volume will persist even after the associated EC2 instance is terminated.

⇒ Managing the deletion attribute



EBS Volumes [Hide details](#)

▼ Volume 1 (AMI Root) (Custom)

Storage type Info EBS	Device name - required Info /dev/xvda	Snapshot Info snap-00022cfee7f3b9690
Size (GiB) Info 8	Volume type Info gp2	IOPS Info 100 / 3000
Delete on termination Info No	Encrypted Info No	KMS key Info Select

KMS keys are only applicable when encryption is set on this volume.



Amazon EFS

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Elastic File System

Amazon Elastic File System (Amazon EFS) provides serverless, fully elastic **file storage** so that you can **share files** without provisioning or managing storage capacity and performance.

- Multi-AZ Availability
- Scalability
- Shared File System
- Elasticity
- NFSv4.1 protocol
- Performance Mode:
 - General Purpose (broad range)
 - Max I/O (high throughput / IOPS)
- Pay as You Go Pricing



Posix System Standard API

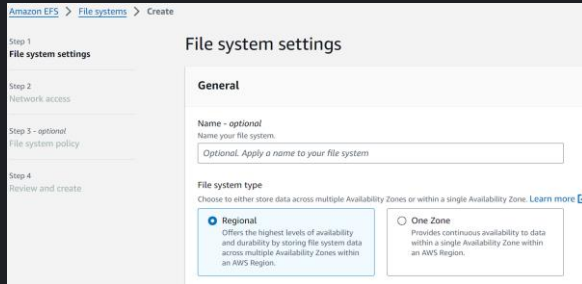
The POSIX (Portable Operating System Interface) standard defines a set of APIs for compatibility between various UNIX-based operating systems.



- Posix APIs Wide range of functionality.
- POSIX simplifies the porting of applications to Linux and fosters interoperability between different platforms.



Performance & Storage classes on EFS



- Performance Scaling
- Performance and Throughput Modes



- Supports 1000 concurrent NFS



- Storage Classes IA / One Zone-IA



- Automatic throughput scaling



- Supports 1000 concurrent NFS

<https://t.me/learningnets>

Performance Mode (General purpose)

- Designed for a wide range of workloads, including latency-sensitive applications and those with mixed read/write operations.
- Offers low latency and good throughput for most use cases.
- Suitable for applications such as web serving, content management, and development environments.
- Automatically scales performance based on the amount of data stored in the file system.

VS

Performance Mode (Max IO)

- Optimized for applications that require the highest levels of aggregate throughput and IOPS.
- Provides higher IOPS and throughput compared to the General Purpose mode, making it suitable for latency-sensitive and I/O-intensive workloads.
- Ideal for applications such as big data analytics, media processing, and database workloads.
- Performance does not scale automatically based on data size; users need to manually adjust provisioned throughput capacity.



Throughput Mode (Bursting throughput)

- Designed for workloads with **unpredictable or spikey access patterns**.
- Provides burst credits that allow the file system to achieve throughput levels higher than its baseline for short periods, enabling **burst workloads** to achieve high performance without provisioning throughput capacity.
- Suitable for applications with **intermittent usage patterns**, such as development and test environments, or applications with **periodic data processing tasks**.

VS

Throughput Mode (Provisioned throughput)

- Designed for applications with **predictable or sustained throughput** requirements.
- Users can provision a **specific amount of throughput** (in MiB/s) for the file system, ensuring consistent performance regardless of workload spikes or burst credits.
- Suitable for applications with **continuous data processing**, high-volume data transfers, or large-scale analytics workloads where predictable performance is critical.



Section 16:

AWS Security and Compliance

AWS Identity and Access Management (IAM)





AWS Identity Access Management (IAM)

Centrally manage access & permissions



Users

Identities that we can attach permissions to

Groups

Collections of users

Roles

Collection of permissions that can be assumed by identities

Policies

Definition of permissions



<https://t.me/learningnets>





AWS Identity Access Management (IAM)

Centrally manage access & permissions



Users

Identities that we can attach permissions to

- Principle of **Least Privilege**:
No access per default
Only grant specific access to what is needed



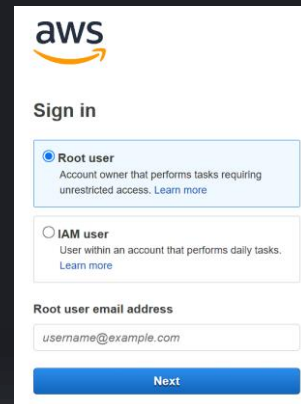
<https://t.me/learningnets>



IAM - Users

Types of users:

- **Root User:**
Initial user with full access and services
Intended for account set up and emergency
- **Standard IAM users:**
Unique set of credentials
Direct access to AWS resources
- **Federated users:**
Authenticated through external identity providers:



aws

Sign in

☒ **Root user**
Account owner that performs tasks requiring unrestricted access. [Learn more](#)

☐ **IAM user**
User within an account that performs daily tasks. [Learn more](#)

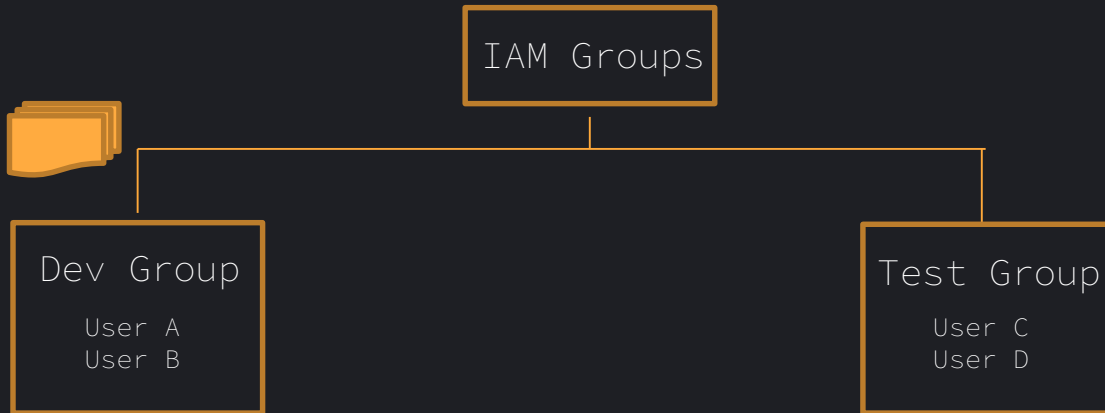
Root user email address
username@example.com

Next

IAM - Groups

Groups:

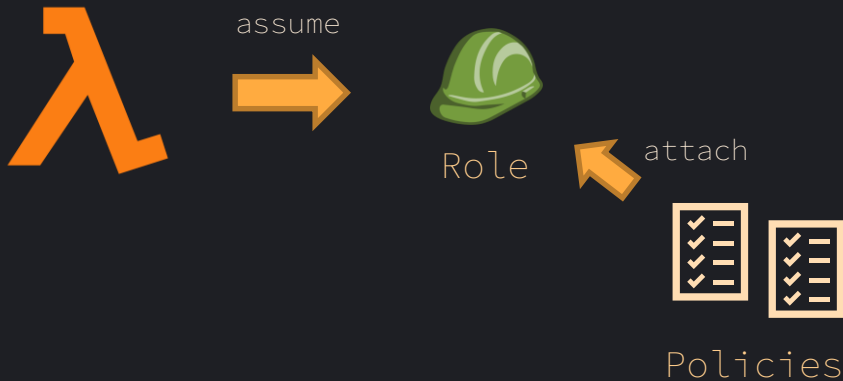
- A collection of users managed as a single entity
- Assign policies to group => all users inherit permissions
- A user can belong to multiple groups
- No credentials associated



IAM - Roles

Roles:

- A combination of permissions that can be assumed
- Attach policies to role
- Role can then be assumed by identities
- Services need to assume roles to perform actions



<https://t.me/learningnets>





IAM - Policies

Policies are documents that define permissions for IAM entities

Managed policies:

- Centrally managed standalone policies

AWS Managed policies:

Created and managed by AWS

Customer Managed policies:

Created and managed by users

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:Get*",
        "s3:List*"
      ],
      "Resource": "*"
    }
  ]
}
```



IAM - Policies

Inline policies:

- Attached only to a single IAM user
- Non-reusable



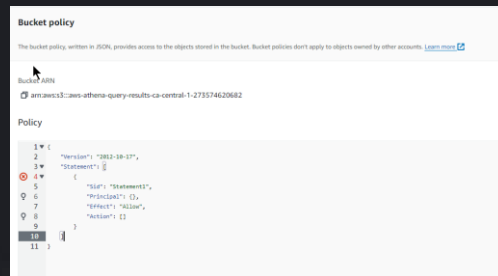
IAM - Policies

Identity-based policies:

- Associated with IAM identities
- Determine what actions can be performed
- Effective to grant identity permissions across different services and resources

Resource-based policies:

- Attached to a resource instead of IAM identity
- Grant or deny permissions on the resource
- Inline policy only



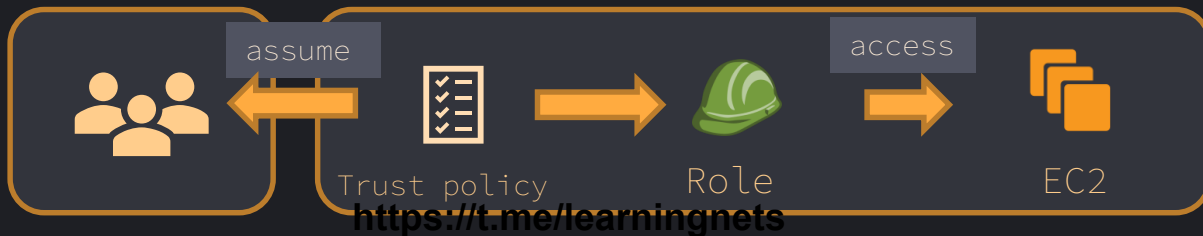
IAM – Trust Policy

Define which entities (accounts, users, or services) are **allowed to assume a role**.

- Type of resource-based policy for IAM roles

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:root"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

- Used for example for cross-account access





AWS KMS

(Key Management Service)

KMS

Overview

- **Manages Encryption Keys:**
To encrypt data in other AWS services
⇒ Used to encrypt & decrypt data
- **Integration:**
Integrates to other services (S3, databases, EBS volumes etc.)
- **API calls:**
Don't store secrets in code
- **Cloud Trail integration:**
Log use of your keys for auditing

Use Cases

- Encrypt data stored in **S3 buckets**
- **Database credentials:**
Encrypt credentials instead of storing them in plain text
<https://t.me/learningnets>

Types of Keys

Symmetric Keys

Default

- **One key** for both encryption and decryption
- Suitable for **high-volume data**.
- **Example:** AES with 256-bit keys
- **At rest & in-transit**

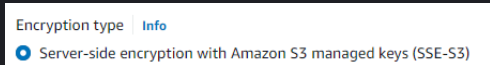
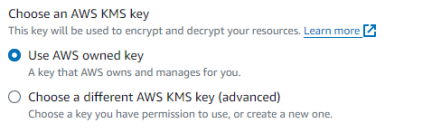
Asymmetric Keys

- Uses **key pair**
- **Public Key:** Encrypt data (can be downloaded)
- **Private Key:** Decrypts data
- Encrypted data must be **shared safely**
- **Sign/Verify** operations
- **Example:** RSA & ECC
<https://t.me/learningnets>

AWS-Managed vs. Customer-Managed

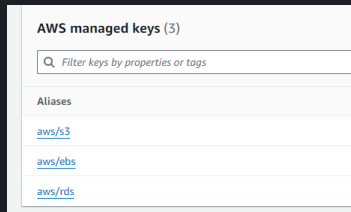
AWS owned keys

- **Controlled and managed by AWS**
- No direct access and lifecycle control



- Owned by service
- Good choice unless you need to audit and manage key

AWS managed keys



- Created & managed by AWS KMS – but customer-specific
- **Cannot control** over its usage and policies, rotation etc.
- Good choice unless you need to control the encrypt key
- **Audit using CloudTrail**

AWS-Managed vs. Customer-Managed

Customer managed keys

Key Management
Service (KMS)



[AWS managed keys](#)

Customer managed keys

- **You create, own, and manage**
- Full control over these KMS keys
- Policies, rotation, encryption

KMS Key Management

Creating Keys

- AWS Management Console, AWS CLI, or SDK
- Selecting the key type
- Key policies

Rotating Keys

- Replaces old keys with new ones
- KMS handles complexities

Managing Keys

- Configuring key policies

Key Rotation

Automatic Rotation

- For keys that **AWS manages**
- **Automatically** rotates the keys every year

Customer Managed

- Users responsibility

Policies

Default Policies

- **Full access to the key to the root user**
- Allows usage of **IAM policies**

Custom Policies

- More complex requirements
- More Granular Control
- **Regulated industries**

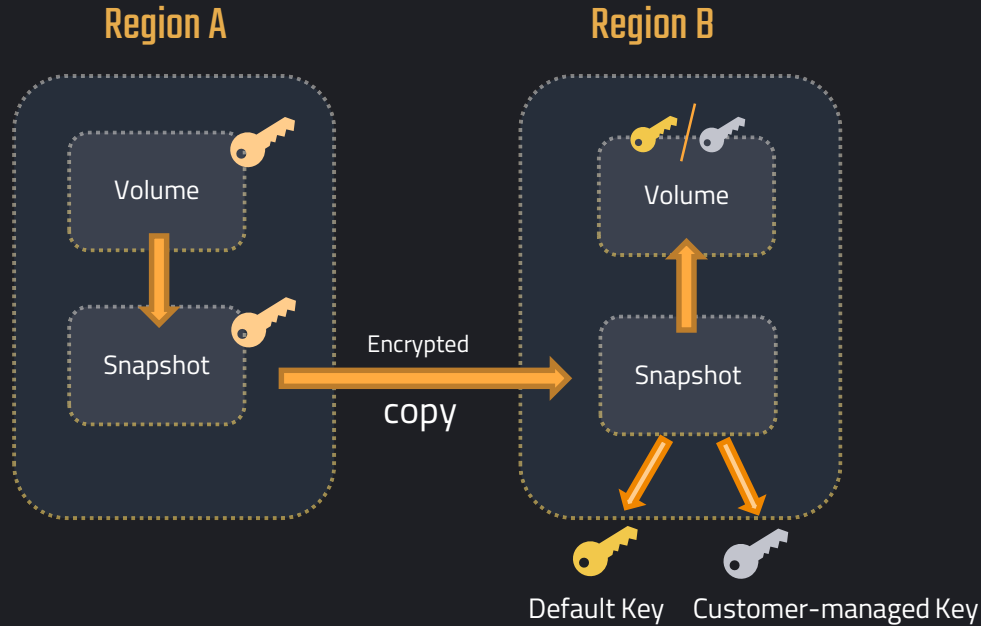
AWS KMS

Pricing

- **Pricing:**
 - \$1.00 per customer-managed key per month;
 - \$0.03 per 10,000 API requests.
- **Key Rotation:**
 - **Automatic:** Free for AWS-managed keys.
 - **Manual:** No extra charge for customer-managed keys; requires setup.
- **Cross-Region Requests:** \$0.01 per 10,000 requests for using a KMS key in a different region.

Cross-region

Keys are bound to the region in which they are created



Multi-Region keys in AWS KMS

- Use keys in different AWS Regions you had the same key
- Each set of *related* multi-Region keys has the same key material and key ID
- Manage each multi-Region key independently
- Create a multi-Region primary key $\Rightarrow$ replicate it into Regions that you select

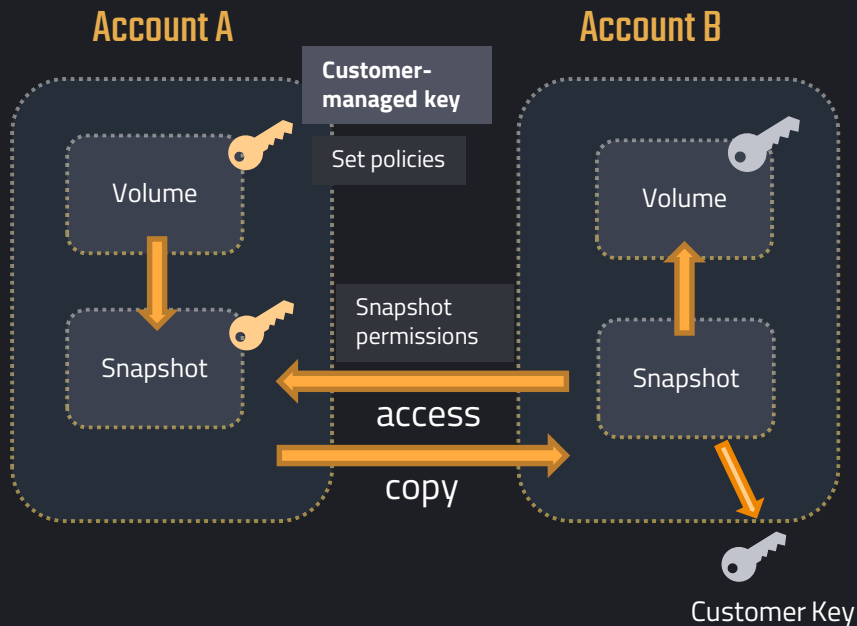
Use Cases

- Disaster recover in multi-region setup
- Data distributed in multiple regions
- Distributed signing applications

Cross-account

Keys can be shared across accounts

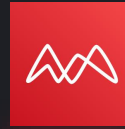
Configurable using policies





AWS Macie

AWS Macie

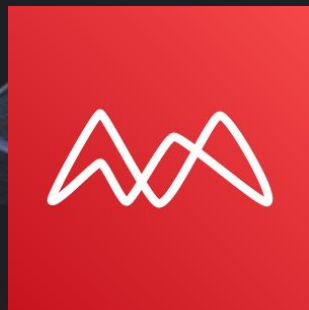


Purpose

Features

Use Cases

- Automatically scans and **classifies sensitive data in Amazon S3**
- **Machine Learning:** Detects sensitive data
 - Personally identifiable information (PII)
 - Financial data
 - Health information
 - *Anomalous access patterns*
- **Automated alerts:**
Detailed alerts when sensitive data or unusual access patterns are detected; Integrates with CloudWatch and other services
- **Comprehensive Dashboard:** Overview of S3 environment
 - Regulatory Compliance
 - Security Monitoring
 - Risk Assessment



AWS Secrets

003-1040559 1250 002-77156 8 1760 0009-14563.7 73273

<https://t.me/learningnets>

AWS Secrets



Purpose

- Manage and retrieve secrets

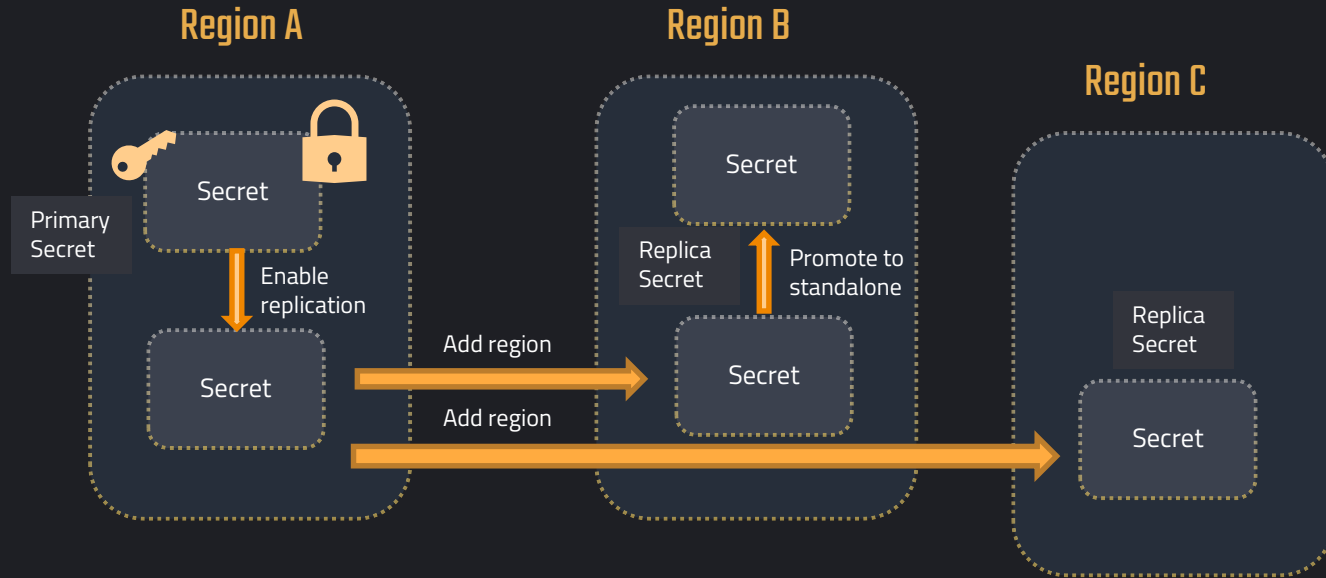
- Database credentials (RDS & Redshift)
- API Keys
- Access Tokens

Features

- **Secrets Management & Storage:** Encrypts and stores secrets with pre-built integration for other services
- **Automatic Rotation:** No need to change code in applications
- **Retrieval:** Secure retrieval via API calls
- **Auditing:** CloudTrail Integration

Cross-region replication

Replicate secrets across multiple AWS Regions



Cross-region replication

- **ARN Consistency:**
ARN remains the same

Primary: `arn:aws:secretsmanager:Region1:123456789012:secret:MySecret-a1b2c3`

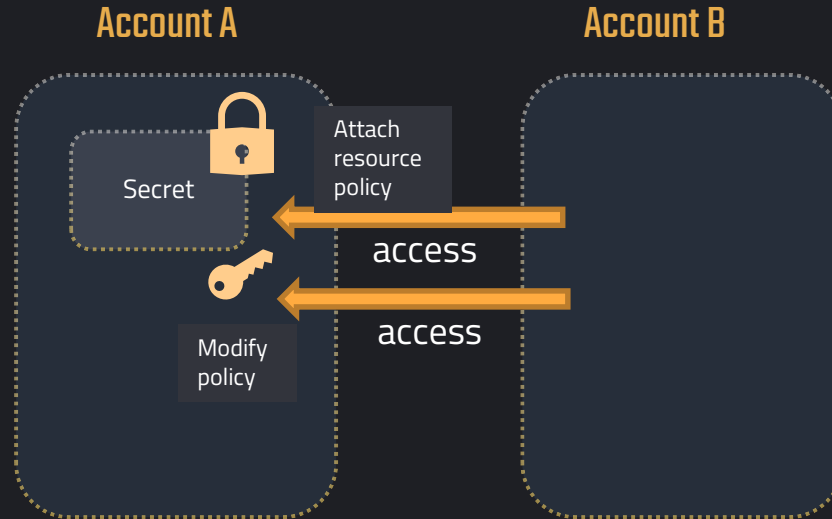
Replica: `arn:aws:secretsmanager:Region2:123456789012:secret:MySecret-a1b2c3`

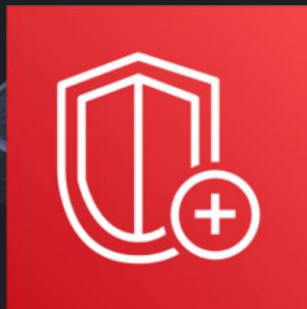
- **Metadata and Data Replication:**
Encrypted secret data, tags, and resource policies are replicated across specified regions.
- **Automatic Rotation:**
If rotation is enabled on the primary secret, the updated secret values automatically propagate to all replicas.

Cross-account

Secrets can be shared across accounts

Configurable using policies



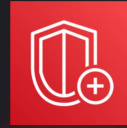


AWS Shield

003-1040559 1250 002-77156 8 1760 0009-14563.7 73273

<https://t.me/learningnets>

AWS Shield



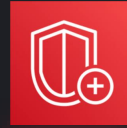
Protect against Distributed Denial of Service (DDoS) attacks

AWS Shield **Standard**

- **Automatically enabled & Free**
- **Protection against most common** network and transport layer DDoS attacks (**96%**)
- Attacks against network and transport layers (layer 3 and 4) and the application layer (layer 7).
- E.g. Slow reads or volumetric attacks
- **AWS Services Covered:** Amazon CloudFront, Elastic Load Balancing (ELB), Amazon Route 53, and more.
- **Visibility and Reporting:** Provides AWS CloudWatch metrics and AWS Health Dashboard notifications during larger attacks.

<https://t.me/learningneta>

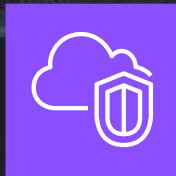
AWS Shield



Protect against Distributed Denial of Service (DDoS) attacks

AWS Shield **Advanced**

- **Enhanced DDoS Protection:** Guards against complex DDoS attacks.
- **Financial Shield:** Protects from attack-related cost spikes.
- **24/7 Expertise:** Access to AWS DDoS Response Team.
- **Attack Insights:** Immediate and detailed attack analysis.
- **Custom Rules:** Personalized protection with AWS WAF.
- **Targeted Defense:** Specific protection for key resources.
- **Premium Service:** Subscription model with advanced features.



Virtual Private Cloud

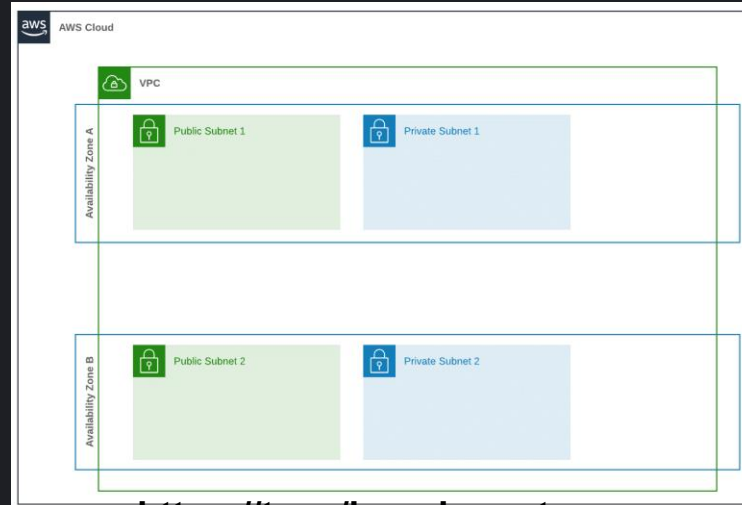


Amazon VPC

Private, secure, isolated network within the AWS cloud to launch your resources.

⇒ *Can be linked to on-premise infrastructure*

⇒ *Regional Service*



<https://t.me/learningnets>





Amazon VPC Subnets



A range of IP addresses in your VPC

⇒ *Zonal Service*

- Types of Subnets
 - *Public Subnets* have access to the internet
 - *Private Subnets* do not have direct access to the internet
 - *VPN-only Subnets* are accessed via a VPN connection
 - *Isolated Subnets* are only accessed by other resources in the same VPC

Subnet Routing



Route Tables are sets of rules that dictate how traffic is routed in your VPC.





Networking Components

003-1040559 1250 032-77156.8 1760 0009-14563.7 73273

<https://t.me/learningnets>



Amazon VPC Networking Components



Internet Gateway



Allows communication between your VPC and the internet.

Egress-Only Internet Gateway



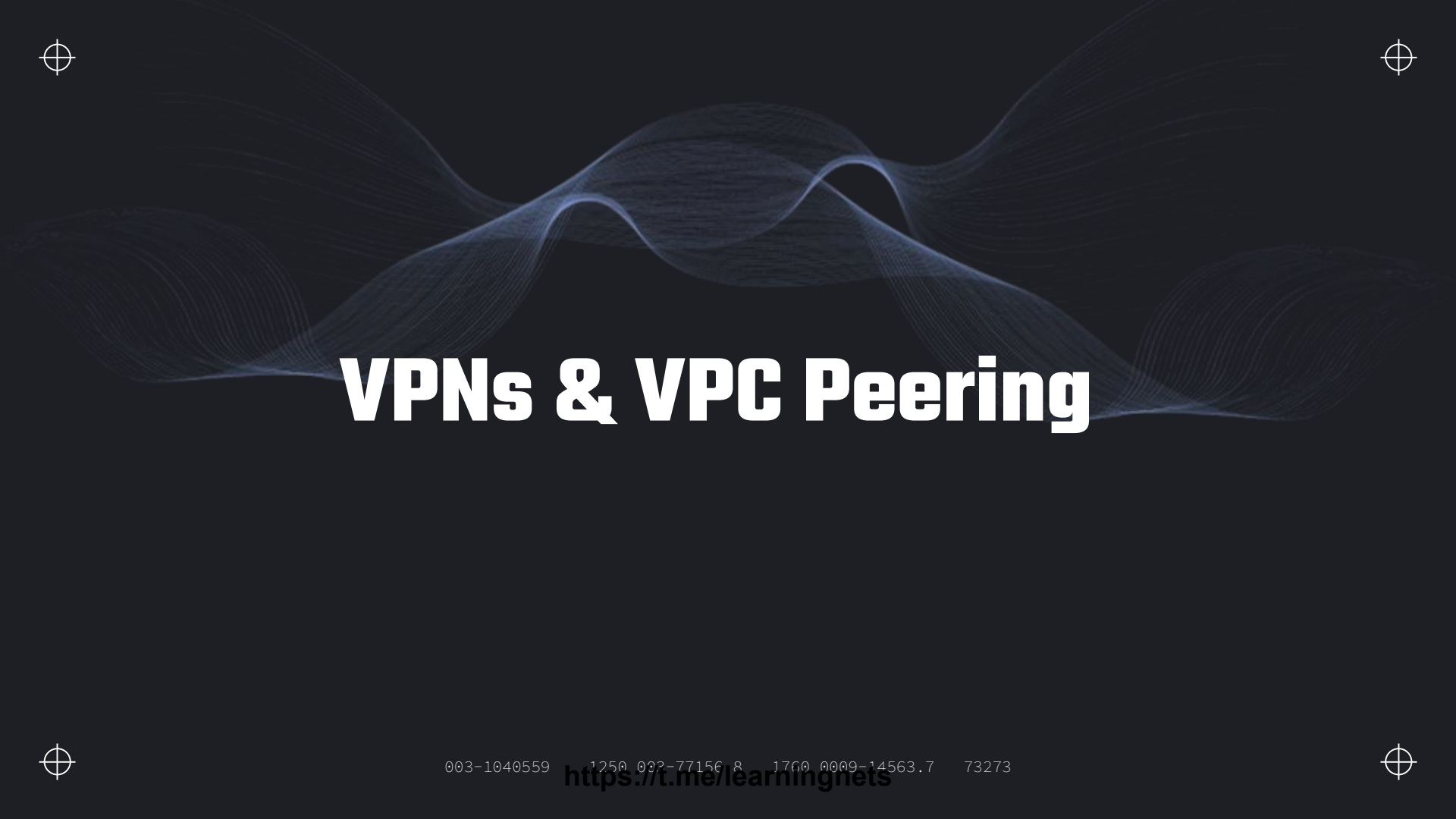
Allows outbound IPv6 communication from VPC instances to the Internet, while blocking inbound IPv6 connections.

NAT Gateway/Instance



Allows resources in private subnets to connect to external destinations but prevents connection requests





VPNs & VPC Peering

003-1040559 1250 002-77156.8 1760 0009-14563.7 73273

<https://t.me/learningnets>



Amazon VPC and Corporate Network

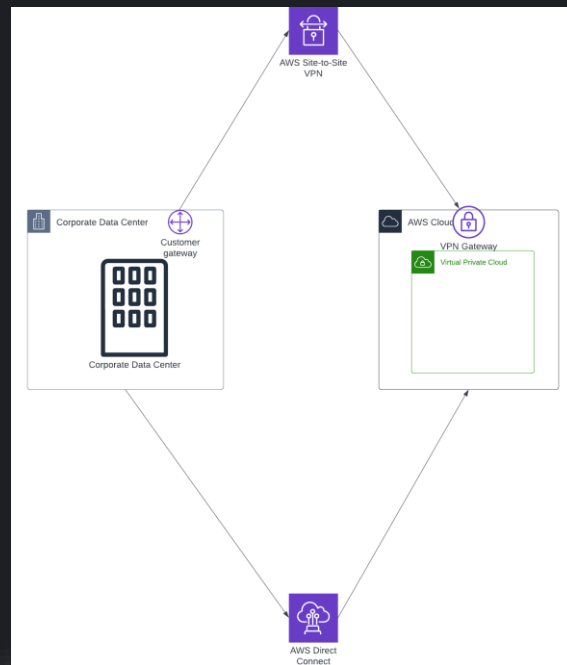
You can connect your VPC to your own corporate data center

Virtual Private Network

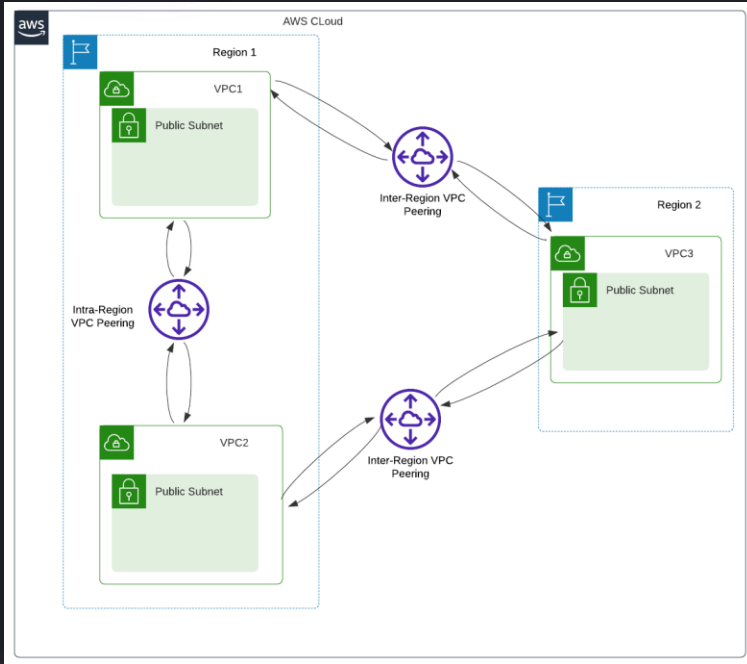
- Secure connection between a VPC and an on-premises network over the internet.

Direct Connect

- Private connectivity between corporate networks and AWS.

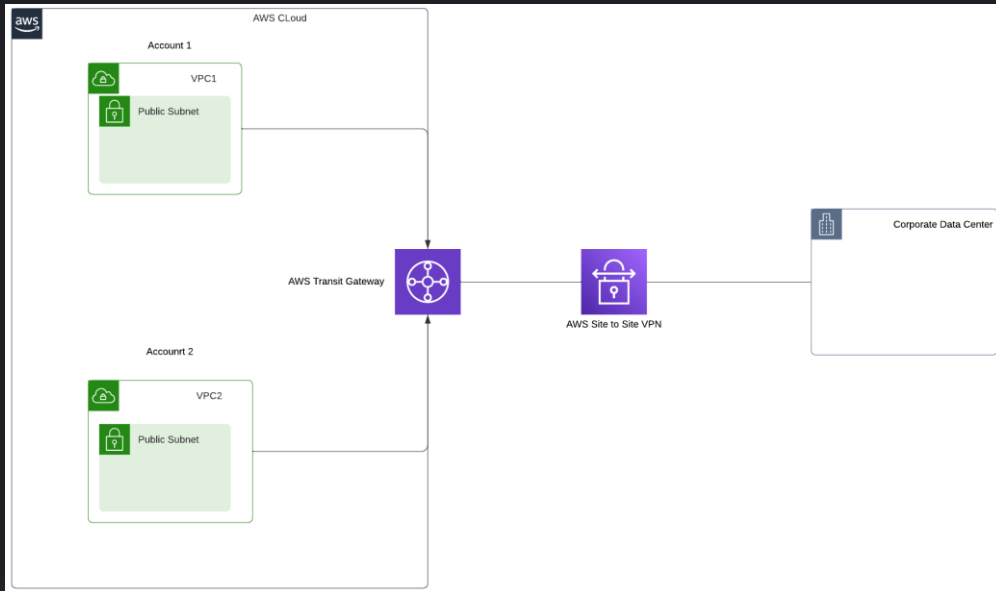


Amazon VPC Peering

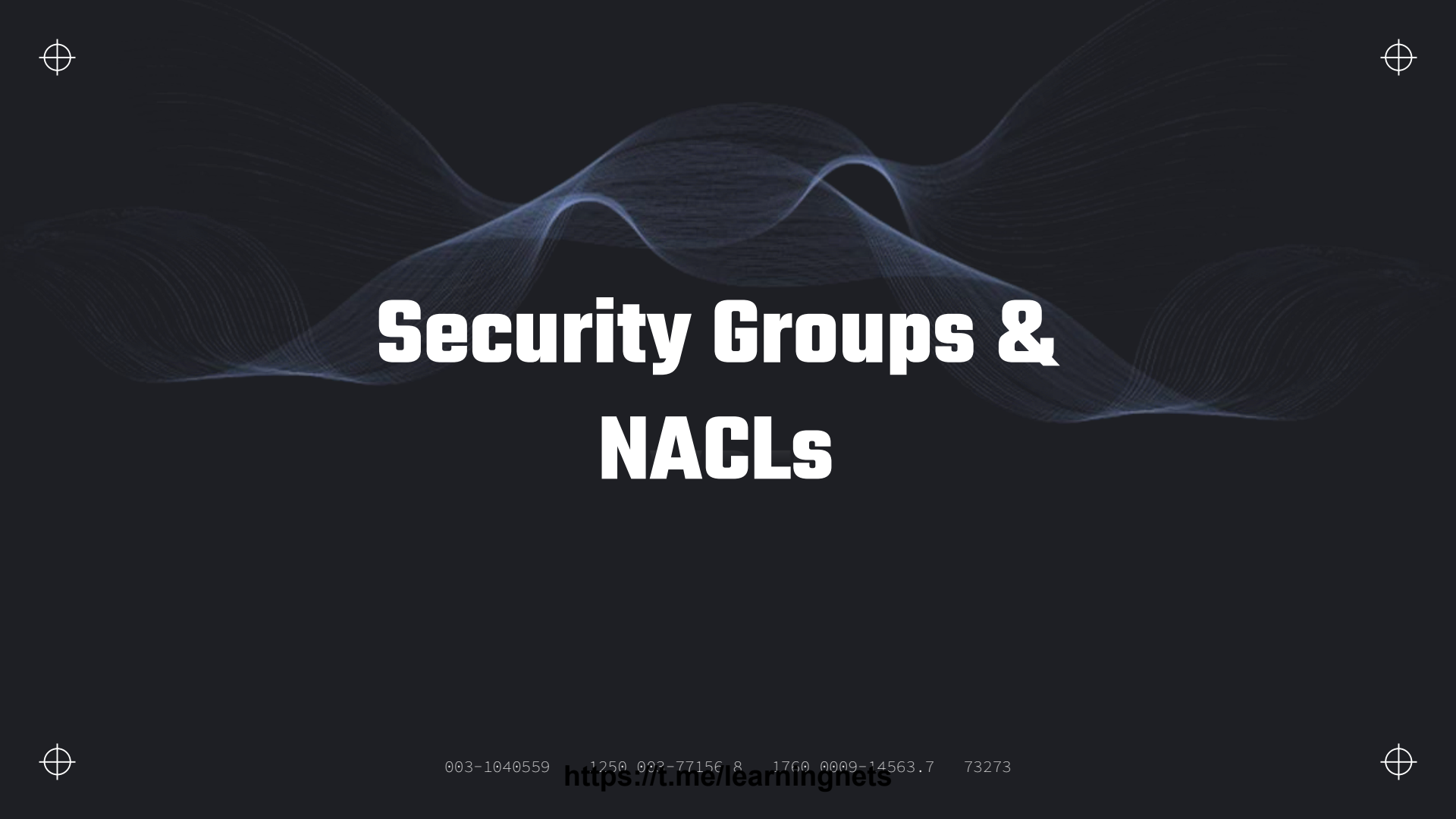


- Enables direct communication between two VPCs
- Intra or Inter Region

Amazon VPC Transit Gateway



- Central hub interconnecting VPCs and on-premises networks.



Security Groups & NACLs

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

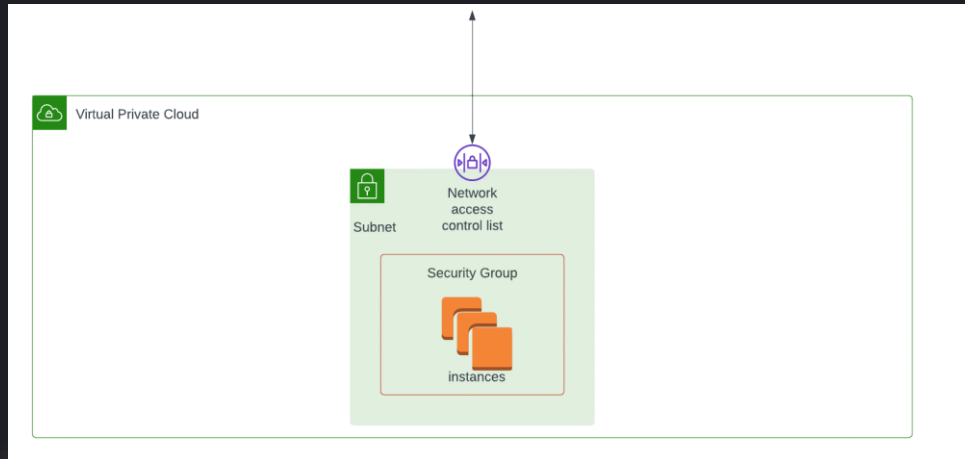
<https://t.me/learningnets>

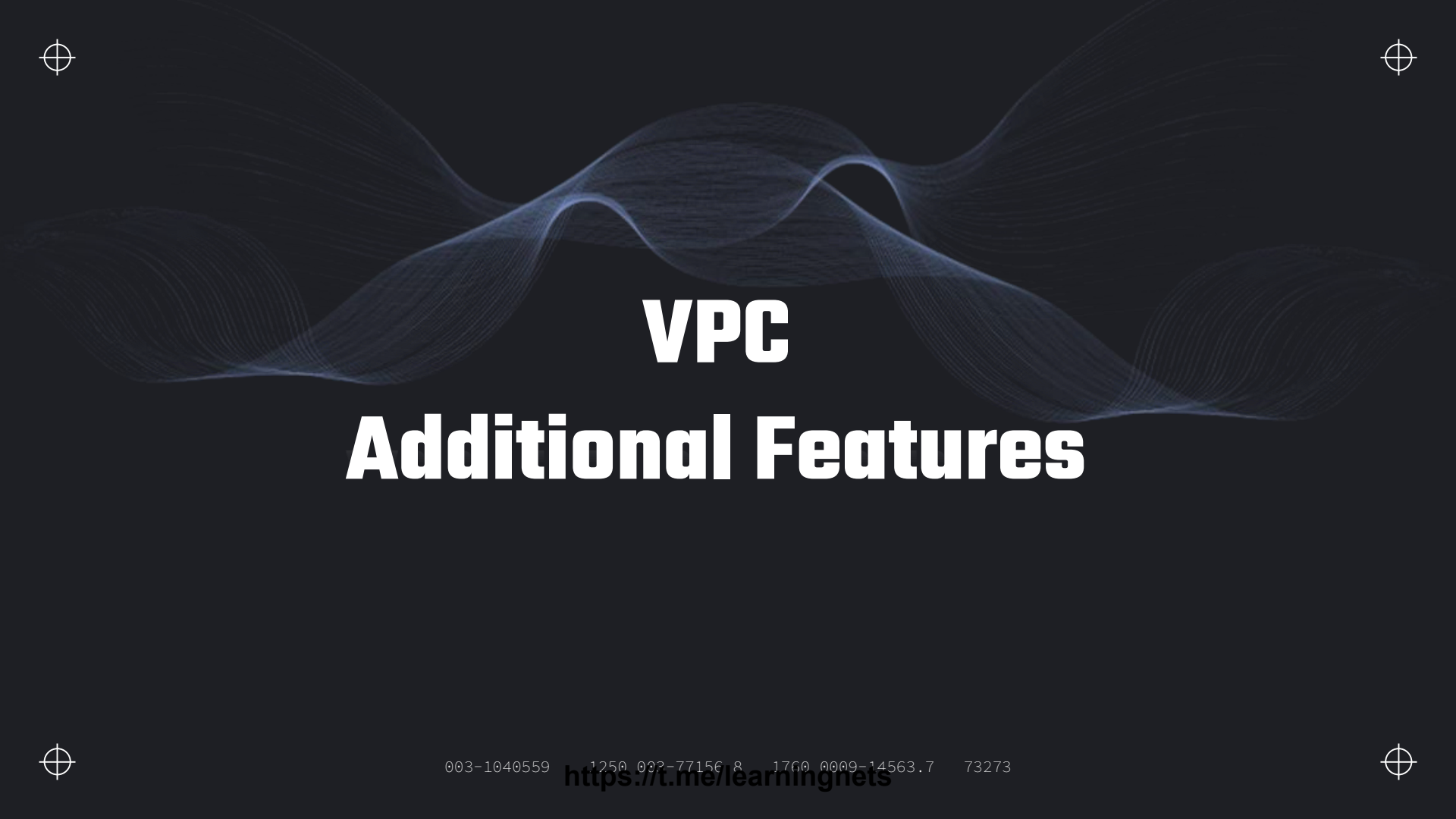


Amazon VPC Security



- Security Groups control inbound or outbound traffic at resource level
- Network Access Control List (NACL)
Control inbound or outbound traffic at the subnet level





VPC

Additional Features

003-1040559 1250 002-77156.8 1760 0009-14563.7 73273

<https://t.me/learningnets>



Amazon VPC Extra Features



VPC Flow Logs: Capture information about IP traffic going to and from network interfaces.

Reachability Analyzer: Analyze network reachability between resources within your VPC and external endpoints

Ephemeral Ports: Temporary ports for outbound communication

VPC Sharing: Share your VPC resources with other AWS accounts in the same AWS Organization,



AWS CloudTrail



003-1040559 1250 002-77156 8 1760 0009-14563.7 73273

<https://t.me/learningnets>



AWS CloudTrail

Audit and Governance Tool

- Records all activities that take place within your AWS account
- Activities are recorded as events
- Enabled by default

The screenshot displays the AWS CloudTrail console. On the left is a navigation sidebar with options like Dashboard, Event history, Insights, Lake, Trails, Settings, Pricing, Documentation, Forums, and FAQs. The main content area is titled 'CloudTrail Insights' and shows a message that insights are not enabled. Below this is the 'Event history' section, which contains a table of recorded events.

Event name	Event time	Event source
ConsoleLogin	April 23, 2024, 23:06:27 (UTC+0...)	signin.amazonaws.com
BackupJobCompleted	April 23, 2024, 08:36:41 (UTC+0...)	backup.amazonaws.com
RecoveryPointCreated	April 23, 2024, 08:30:36 (UTC+0...)	backup.amazonaws.com
BackupJobStarted	April 23, 2024, 08:30:30 (UTC+0...)	backup.amazonaws.com
BackupDeleted	April 23, 2024, 07:01:04 (UTC+0...)	backup.amazonaws.com

At the bottom of the event history section is a link: [View full Event history](#).

CloudTrail interface

<https://t.me/learningnets>



AWS CloudTrail Events

Record(s) of activities

- Types of Events
 - **Management Events:** Captures high-level operations
 - **Data Events:** Captures data-level operations
 - **Insight Events:** Captures unusual activity

Events History



view management events of the past
90 days

<https://t.me/learningnets>





AWS CloudTrail Trails



Captures records of AWS activities and stores in S3

- Trail Types:

- Multi-Region*

- Trail applies to all regions

- Single Region*

- Trail applies to one region

- Organizational*

- Logs events for all accounts in an organization

- Features:

- Multiple Trails Per region creates multiple trails within a single AWS region.



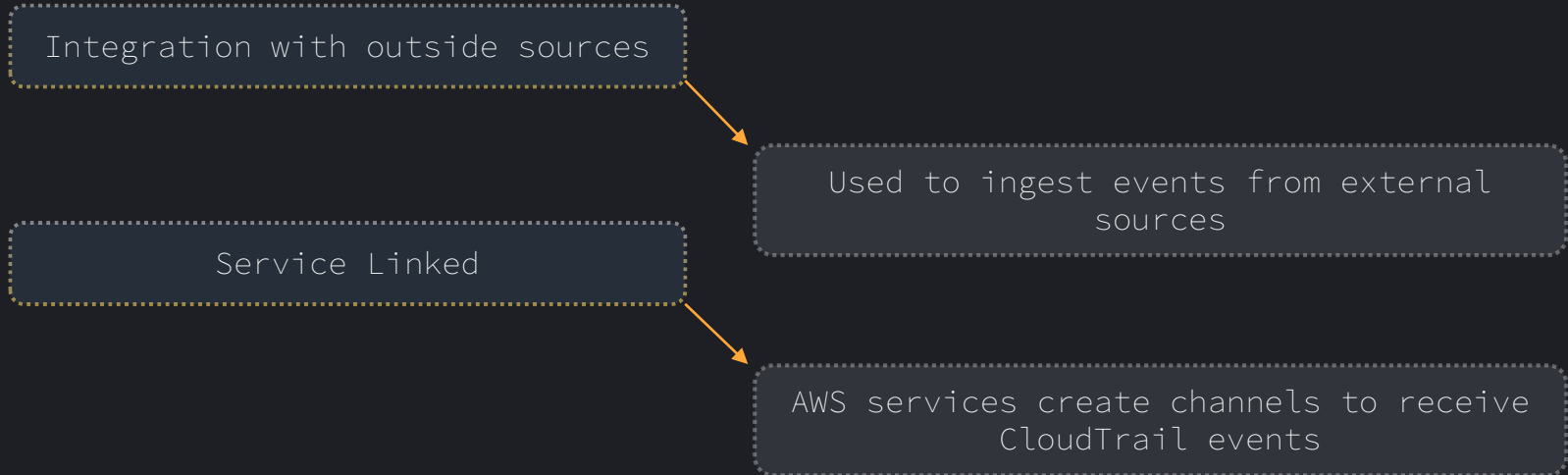


AWS CloudTrail Lake



A Managed data lake for AWS user and API activity

- Lake Channels:





AWS CloudTrail Extras

- CloudTrail allows for deep analysis of event
- Create rules with EventBridge if needed





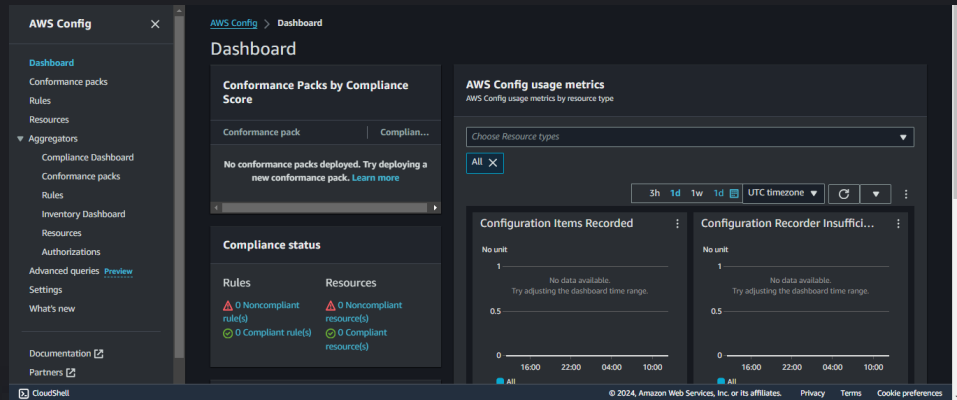
AWS Config



AWS Config

Centralized Configuration
Management

- Assess, audit, and evaluate the configurations of your AWS resources
- Disabled by default
- Generates a configuration item for each resource.



Config interface

<https://t.me/learningnets>





AWS Config Concepts



- *Configuration Item* is the current state of individual AWS resources
- *Configuration Recorder* stores configuration items for resources in your account.
- *Configuration History* is a historical record of configuration changes
- *Configuration Snapshot* is a collection of configuration items
- *Configuration Stream* is an automatically updated list of configuration items for resources recorded by AWS Config.
- *Conformance packs* bundles Config rules, remediation actions, and required AWS resource configurations into a single, reusable package.
- *Discovery* discovers resources in your AWS environment
- *Advanced queries* analyzes real-time and historical resource configurations
- *Resource Relationship* creates a map of relationships between AWS resources.





AWS Config Rules

Evaluates the compliance of your AWS resources against desired configuration

Evaluation results for a Config rule:

Compliant - Resource complies with the rule

Non-compliant - Resource does not comply with the rule

Error - Invalid required or optional parameters

Not Applicable - Filters out resources that the logic of the rule cannot be applied to

- Types of Rules
 - AWS Config Managed Rule
 - AWS Config Custom Rule

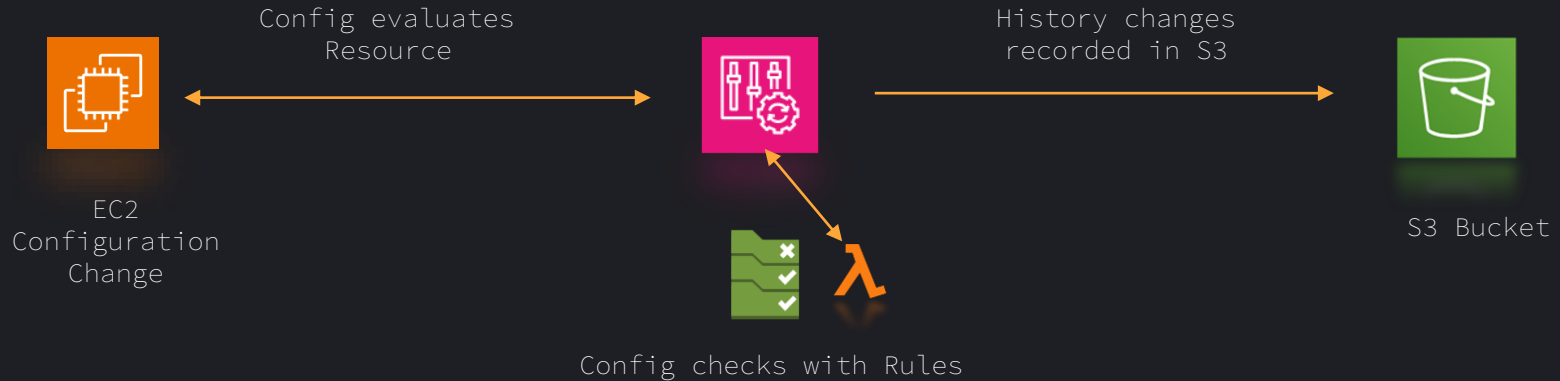
AWS Config Managed Rules

- Pre-defined and customizable rules created by AWS Config.

AWS Config Custom Rules

- Rules you create from scratch.
- Created with lambda functions or Guard

VS





AWS Config Trigger Types



Determines when AWS Config evaluates the rules against your resources.

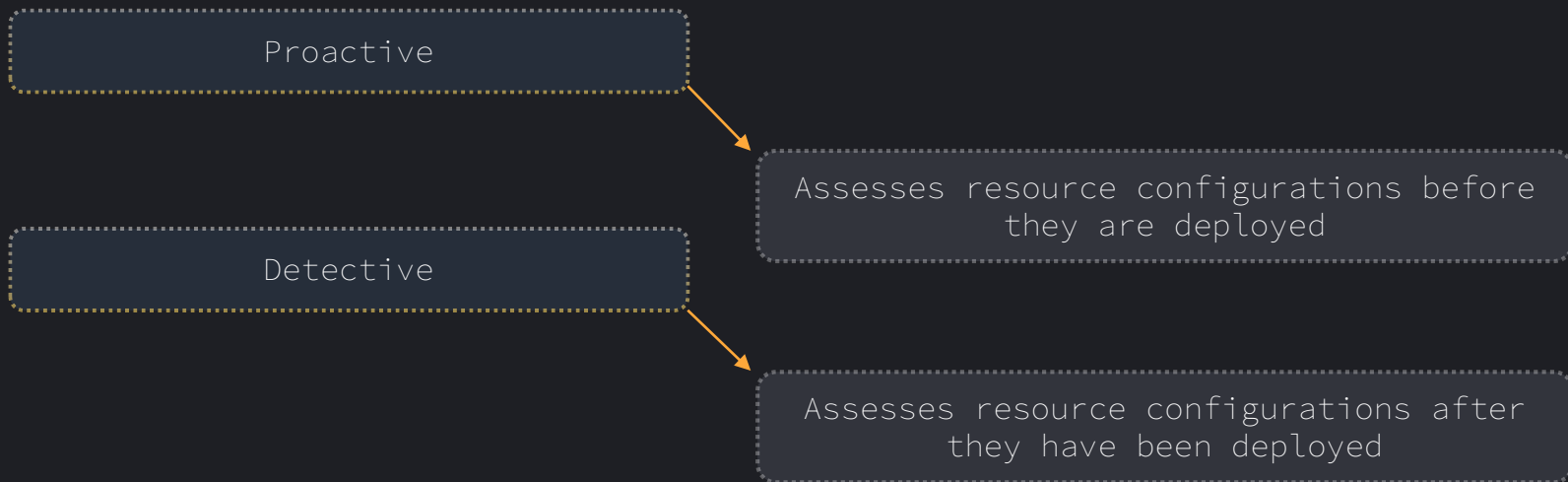
- Trigger Types
 - Configuration changes: A configuration change is detected
 - Periodic: Evaluates at specified intervals
 - Hybrid: Evaluates resource configuration change and chosen frequency





AWS Config Evaluation Modes

- Define when and how resources are evaluated during the resource provisioning process.
- Evaluation Modes:





AWS Config Multi-Account Multi-Region Data Aggregation



Aggregate and centrally manage AWS Config data across multiple AWS accounts and regions

- Concepts
 - **Aggregator**: Collect Config configuration and compliance data from multiple source accounts and regions.
 - **Source Account**: AWS accounts where AWS Config records configuration changes and compliance data for resources
 - **Aggregator Account**: Central hub for aggregating configuration and compliance data from multiple source accounts
 - **Authorization**: Permission granted to an aggregator Account to collect data.





AWS Well-Architected Framework

003-1040559 1250 002-77156.8 1760 0009-14563.7 73273

<https://t.me/learningnets>



AWS Well-Architected Framework



- **Operational Excellence:**
Manage operations to deliver business value and continuously improve processes.
- **Security:**
Protect data and systems; manage access, and respond to security events.
- **Reliability:**
Ensure systems perform as expected, handle changes in demand, and recover from disruptions.
- **Performance Efficiency:**
Use resources efficiently, adapt to changing needs, and leverage new technologies.
- **Cost Optimization:**
Reduce and control costs without sacrificing performance or capacity.





AWS Well-Architected Framework

- Sustainability

Minimize environmental impact by efficiently using resources and reducing carbon emissions.

AWS Well-Architected Tool



003-1040559 1250 032-77156 8 1760 0009-14563.7 73273

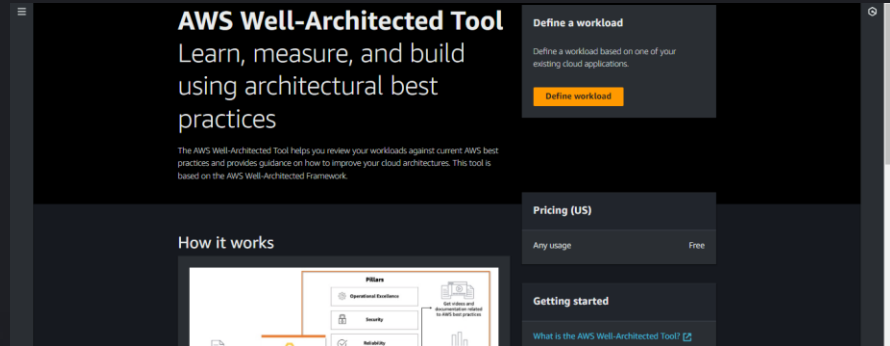
<https://t.me/learningnets>



AWS Well-Architected Tool

Architecture Optimization Tool

- Review workloads against the Well-Architected Framework.
- Questionnaire-Based Assessment



<https://t.me/learningnet> Well-Architected Tool interface



AWS Well-Architected Tool Features



- o **Workload** = Collection of components that add to business value.
- o **Milestones** = crucial stages in your architecture's evolution throughout its lifecycle
- o **Lenses** = Evaluate your architectures against best practices and identify areas of improvement.
 - Lens Catalog (created & maintained by AWS)
 - Custom lenses (user-defined lenses)
- o **High-Risk Issues (HRIs)** = Architectural and operational choices that may negatively impact a business.
- o **Medium risk issues (MRIs)** = Architectural and operational choices that may negatively impact a business but not to the same degree as HRIs.

<https://t.me/learningnets>





AWS Well-Architected Tool Extras



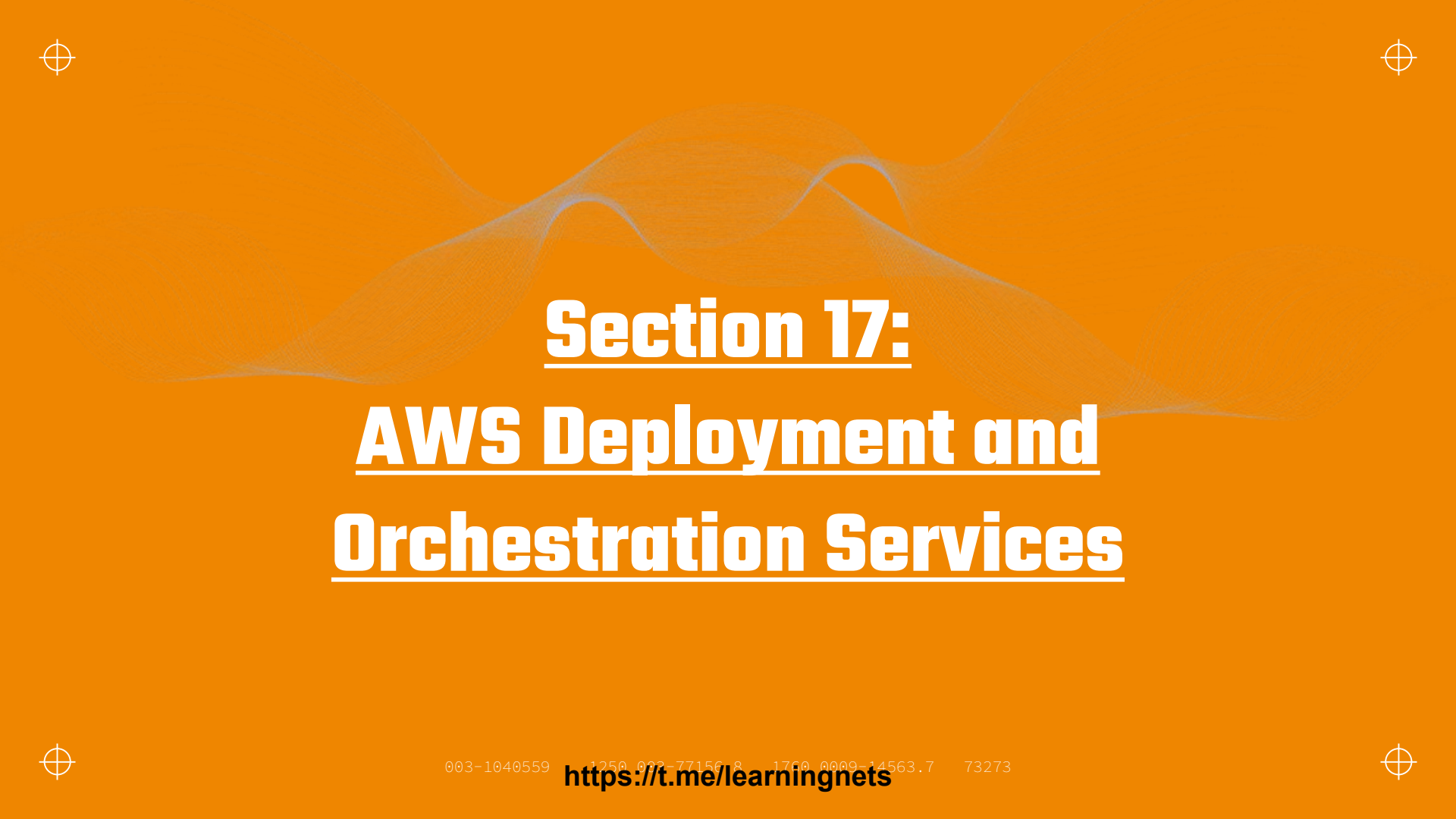
Use Cases

- Continuously improve architectures
- Get architectural guidance
- Enable consistent governance

How it Works

- Define the workload
- Review the workload
- Tool returns feedback





Section 17: **AWS Deployment and** **Orchestration Services**



AWS CloudFormation

003-1040559 1250 032-77156 8 1760 0009-14563.7 73273

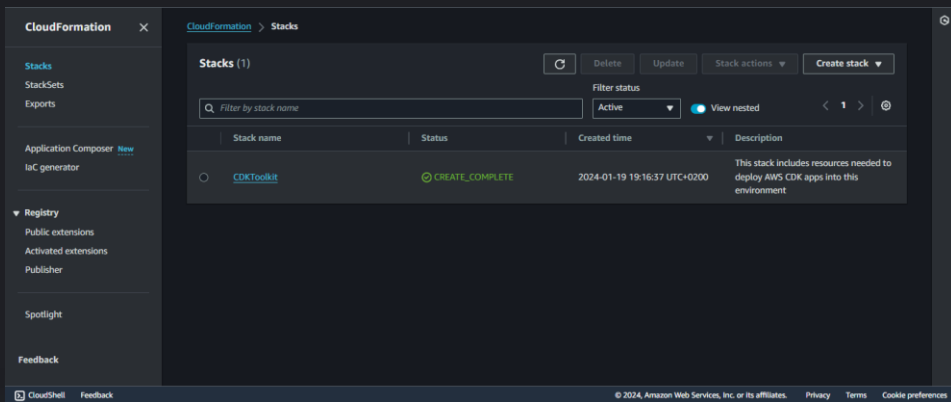
<https://t.me/learningnets>



AWS CloudFormation

Infrastructure Management

- Allows you to define and provision your AWS infrastructure as code
- Use cases:
 - Replicate infrastructure across regions
 - Control and track changes to your infrastructure
 - Simplify infrastructure management



CloudFormation interface

<https://t.me/learningnests>



AWS CloudFormation Templates



Text files that describe the desired state of your AWS infrastructure.

```
1
2 Resources:
3   FirstS3Bucket:
4     Type: AWS::S3::Bucket
5     Properties: {}
6
7
```



YAML Template to create the “FirstS3Bucket” S3 Bucket



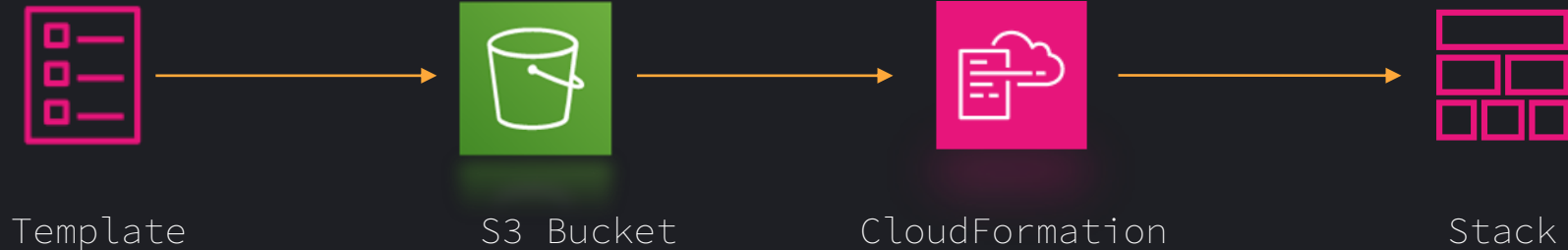


AWS CloudFormation Stacks



A collection of AWS resources created and managed as a single unit

HOW IT WORKS





Introduction to Docker Containers

What is Docker?



- Is a platform for developing, shipping, and running applications in containers
- Packages applications and their dependencies into standardized units called containers.
- Is used to quickly deploy and scale applications.

Why Use Docker Containers?



- Consistency : Applications run consistently across different environments.
- Isolation : Containers are isolated from each other
- Portability : Containers can be easily moved between different systems and environments.
- Scalability : Enables you to easily scale up or down the number of containers as needed.

Docker Use Cases



- Microservices Architecture
- Continuous Integration and Continuous Deployment (CI/CD)
- Hybrid Cloud Environments
- Big Data and Analytics

Docker Components



- Docker Engine
- Docker Image
- Docker Containers
- Docker Registry



Docker Registries



Docker Hub

- Is the most popular container registry
- It hosts millions of pre-built images for various software applications, libraries, and frameworks.
- Docker Hub offers both public and private repositories.

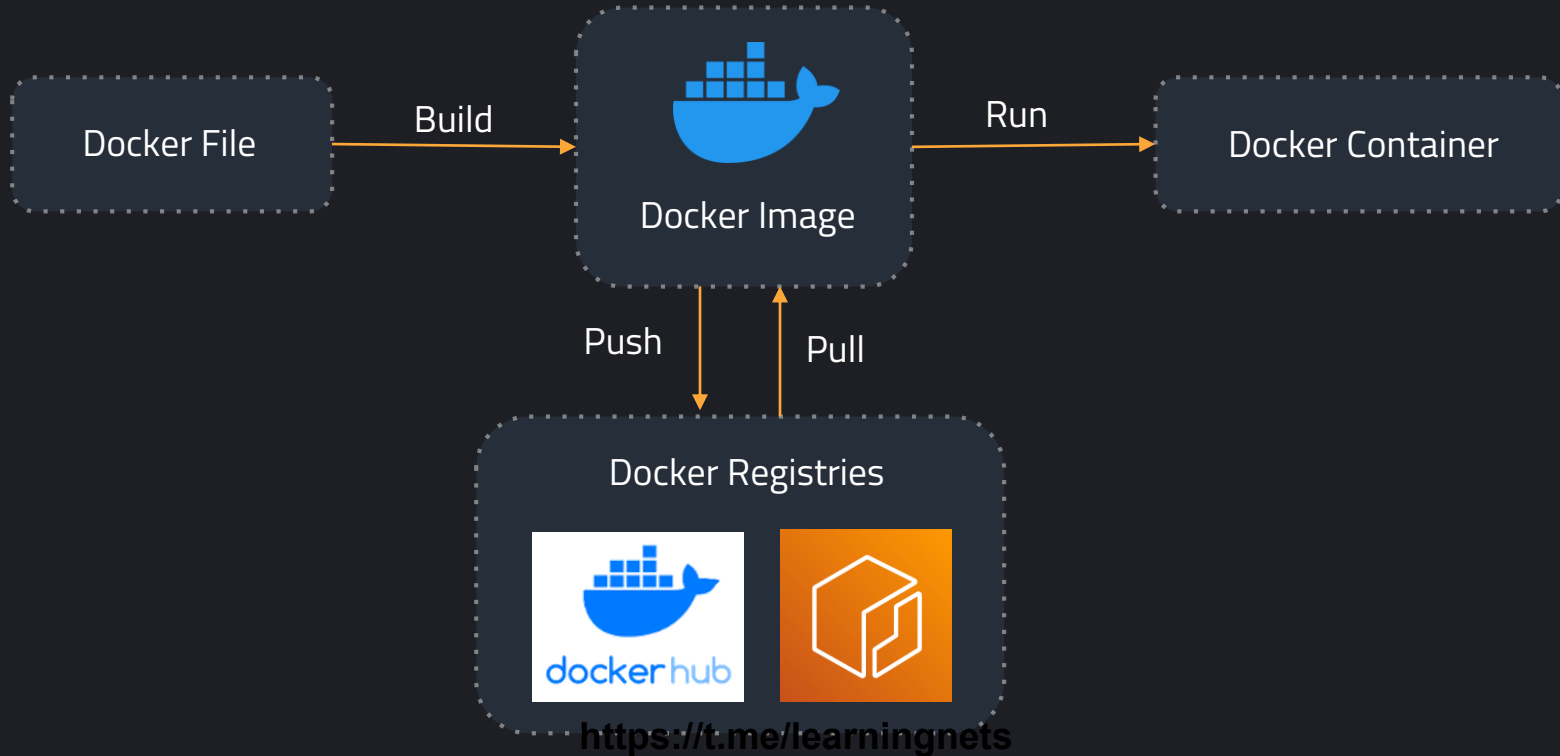


Amazon ECR

- Amazon ECR is a fully managed Docker container registry provided by AWS.
- Integrates seamlessly with other AWS services
- Amazon ECR supports both public and private repositories.



Docker Processes





Elastic Container Service

Elastic Container Service



- Is a fully managed container orchestration service.
- Simplifies the process of container

Deployment

Management

Scaling

- It provides high availability and scalability
- It offers built-in security features
- It's integrated with both AWS and third-party tools.

Elastic Container Service Terms



- Task Definition
 - Is a blueprint for your application
 - It encapsulates all the necessary configuration parameters.
- Cluster
 - Is a logical grouping of container instances
 - Provides a centralized management point

Elastic Container Service Terms



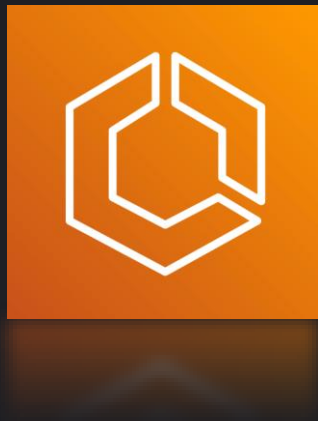
- Task
 - is ideal for short-running jobs
 - It's an instantiation of a Task Definition.
 - Tasks can be scheduled and terminated dynamically based on workload demands.

Elastic Container Service Terms



- Service
 - Is ideal for long-running applications.
 - ECS automatically replaces failed tasks.
 - It's an instantiation of a Task Definition.
- Container Agents
 - Run on each EC2 instance within an ECS cluster.
 - Serve as the communication bridge between the ECS and the container instances.

Amazon ECS launch types



- EC2 launch type
 - You must provision & maintain the infrastructure
 - Suitable for large workloads that must be price optimized.
 - Enables you to use EC2 instances like spot instances and custom instance types.
 - Scaling does not come out of the box.

Amazon ECS launch types



- Fargate launch type
 - You don't need to manage an EC2 infrastructure
 - Requires less effort to set up
 - AWS just runs ECS Tasks for you based on the CPU / RAM you need
 - handles scaling out your capacity
- External launch type
 - Is used to run your containerized applications on your on-premise server or virtual machine (VM)

Task placement strategies and constraints



- Applicable for EC2 launch type launch mode only

Task placement strategies

- is an algorithm for selecting instances for task placement or tasks for termination.
- Available strategies are

Binpack

Spread

Random

Task placement strategies and constraints



- Binpack
 - Tasks are placed on instances that have the least available CPU or memory capacity
 - Helps minimize wasted resources.
 - Beneficial for cost optimization and maximizing the usage of resources.

Task placement strategies and constraints



- Spread
 - Spreads tasks evenly across container instances within the cluster
 - Ensures that no single instance becomes overloaded.
 - Suitable for ensuring even distribution of tasks across instances.

Task placement strategies and constraints



- Random
 - Randomly places tasks onto container instances within the cluster
 - Not suitable for applications with specific performance or availability requirements.
 - It's primarily used when you don't have specific constraints or considerations for task placement.
- It is possible to create a task placement strategy that uses multiple strategies.

Task placement strategies and constraints



- Task placement constraint
 - These are rules that must be met in order to place a task on a container instance.
 - There are two types on constraints types

Distinct Instance

Member of

Task placement strategies and constraints



- Distinct Instance
 - Place each task on a different container instance.
- Member of
 - Place tasks on container instances that satisfy an expression.

Task placement strategies and constraints



- When Amazon ECS places a task, it uses the following process to select the appropriate EC2 Container instance:
 1. CPU, memory, and port requirements
 2. Task Placement Constraints
 3. Task Placement Strategies

IAM Roles for ECS



- Task Execution Role
 - Is an IAM role that ECS uses to manage tasks on your behalf.
- Task Role
 - Enables containers within the task to access AWS resources securely.



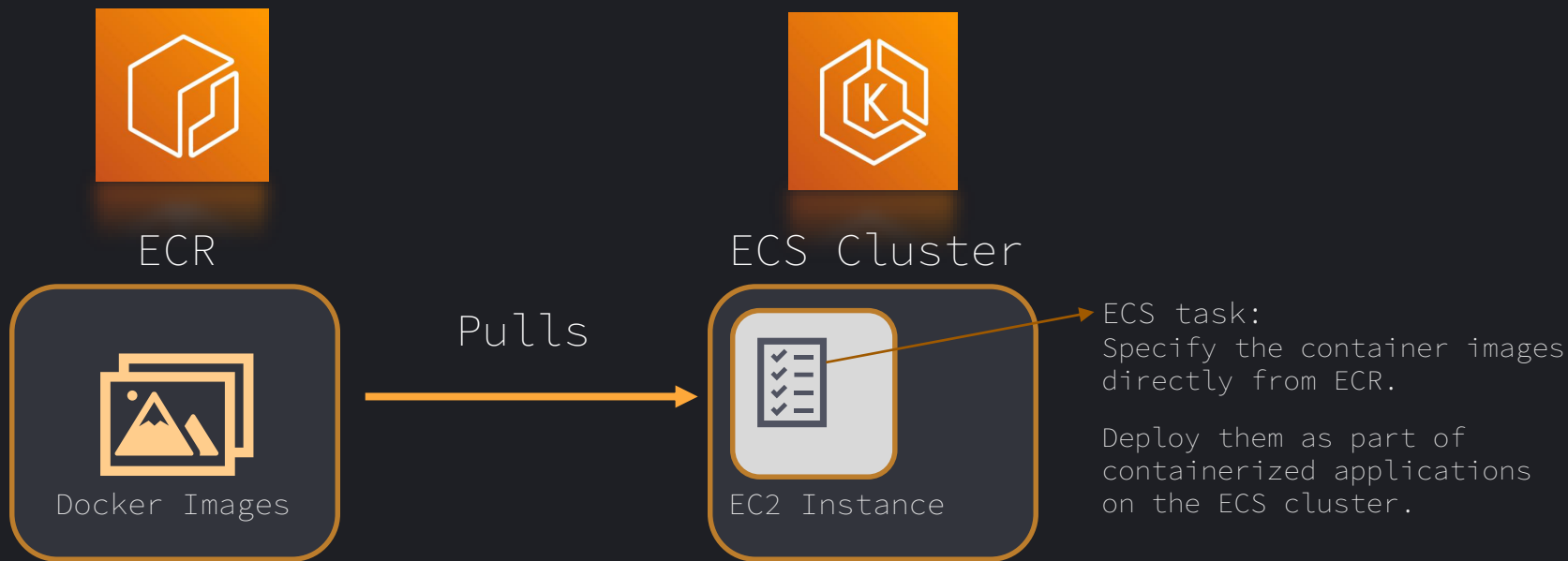
Elastic Container Registry

Amazon Elastic Container Registry



- Amazon ECR is a fully managed Docker container registry provided by AWS.
- It enables you to **store**, manage, and deploy **Docker images** securely.
- It's integrated with other AWS services.
- Is Secure

Amazon Elastic Container Registry



Amazon Elastic Container Registry



- Features
 - Lifecycle policies
 - Image scanning
 - Cross-Region and cross-account replication
 - Versioning
 - Tagging

Amazon Elastic Container Registry



Public Repository

- Are accessible to anyone on the internet.
- Special permission or credential is not required.

Private Repository

- Are only to authorized users.
- Access to private repositories can be controlled using AWS IAM (Identity and Access Management)



Elastic Kubernetes Service

Kubernetes



- Is is an open-source platform designed to automate the

Deployment

Management

Scaling

of containerized applications.

- Google open-sourced the Kubernetes project in 2014
- Suitable for running and managing workloads of all sizes and styles
- It is non cloud native, available on most cloud providers

<https://t.me/learningnets>

Amazon Elastic Kubernetes Service

Managed Kubernetes service to run Kubernetes in the AWS cloud and on-premises data centers.



- It's integrated with other AWS services.
- Provides High Availability
- Scalability
- Security
- Monitoring and Logging

Amazon EKS architecture



Control plane

- Consists of nodes that run the Kubernetes software.
- Manages and orchestrates various components of the Kubernetes cluster.
- Is fully managed by AWS.

Amazon EKS architecture



Compute

- Contains worker machines called nodes.
- Amazon EKS offers the following primary node types.

AWS Fargate

Karpenter

Managed node groups

Self-managed nodes

Amazon EKS architecture



AWS Fargate

- Is a serverless compute engine.
- AWS manages the underlying infrastructure.
- You specify your application's resource needs, and AWS handles the rest.

Amazon EKS architecture



Karpenter

- Best for running containers with a high availability requirement.
- It launches right-sized compute resources in response to changing application load.

Managed node groups

- Create and manage Amazon EC2 instances for you.

Amazon EKS architecture



Self-managed nodes

- Offer full control over your Amazon EC2 instances within an Amazon EKS cluster.
- You are in charge of managing, scaling, and maintaining the nodes.
- Suitable for users who need more control over their nodes.

Section 18: **AWS Monitoring and Cost** **Management Tools**



AWS CloudWatch

003-1040559 1250 002-77156 8 1760 0009-14563.7 73273

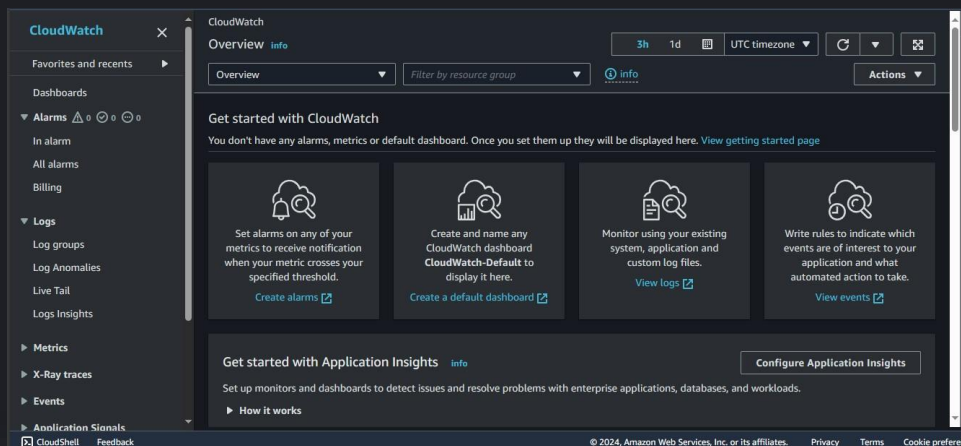
<https://t.me/learningnets>



AWS CloudWatch

Main Monitoring Service

- AWS' proprietary service for monitoring Applications and Resources in real-time
- Metrics are displayed on a Dashboard



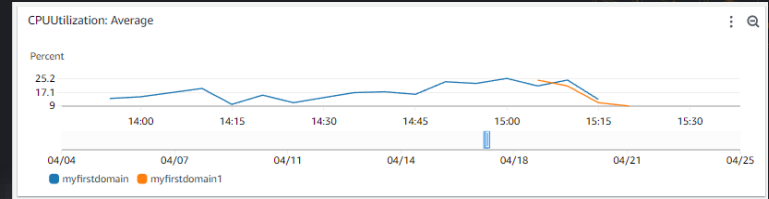
CloudWatch interface

<https://t.me/learningnets>



AWS CloudWatch Metrics

A metric is a numbers to monitor



- Features:

- Name Spaces: Serves as a container for CloudWatch metrics.
- Time Stamps: Every metric should be linked to a timestamp.
- Dimensions: Key/value pair belonging to a metric.
- Statistics: Aggregated data metrics over defined time intervals.
- Period: The length of time associated with a specific statistic.
- Resolution: Level of detail you can see in your data
 - Standard Resolution: data has a one-minute granularity
 - High Resolution: data has a granularity of one second





AWS CloudWatch Metric Streams

- Continuously stream metrics to both AWS & 3rd-party destinations near-real time.
- Kinesis Firehose is used to stream data to AWS destinations.

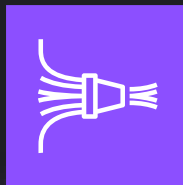
Custom setup with Firehose

Quick S3 setup

Quick AWS partner setup

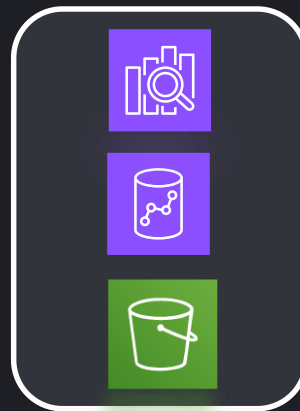


Ingests
metrics



Kinesis Data
Firehose

Sends metrics
to destinations



OpenSearch

Redshift

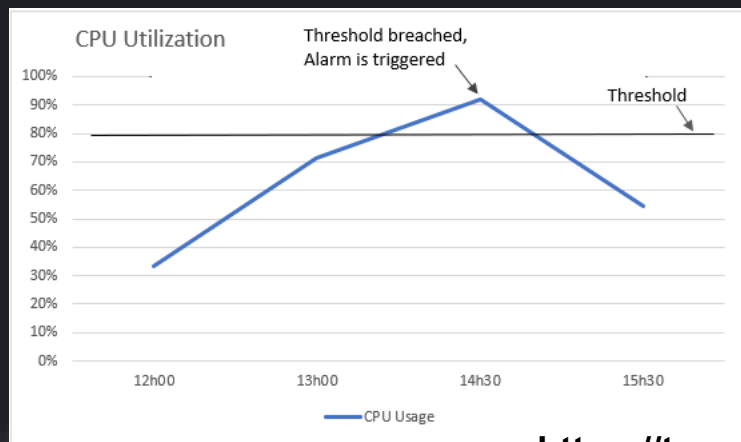
S3



AWS CloudWatch Alarms

Monitor metrics and trigger actions when defined thresholds are breached.

- Types of Alarms:
 - Metric Alarm: Monitors a single metric
 - Composite Alarm: Monitors the state of other alarms





AWS CloudWatch Alarms



- Alarm States:

OK

- Metric is within the defined threshold

ALARM

- Metric is over the defined threshold

INSUFFICIENT_DATA

- Not enough data available to determine the alarm state

- Alarm Actions: Actions an alarm can take when it changes state

- ❑ Amazon SNS: Trigger email, SMS, or push alerts.
- ❑ EC2 Actions: Stop, terminate or reboot EC2 instance.
- ❑ Auto Scaling: Adjust instance count based on load.
- ❑ Lambda: Execute functions for automation.
- ❑ Incident Management: Create urgent incident tickets.





AWS CloudWatch Logs

Collects and consolidates logs from various sources

- **Centralized Logging:** From different services in one location
- **Real-Time Monitoring:** Real-time monitoring of log data





AWS CloudWatch Logs



Collects and consolidates logs from various sources

- **Log streams:** Sequences of log events from a single source
- **Log groups:** Logical containers for log streams
- **Log events:** Records of an activity logged by an application
- **Retention Policy:** Allows you to allocate the period a log is retained
- **Log Insights:** Interactive log analysis tool for querying and visualizing log data stored in CloudWatch Logs
- **S3:** Logs can be sent to S3, Kinesis Data streams/Firehose, and Lambda.

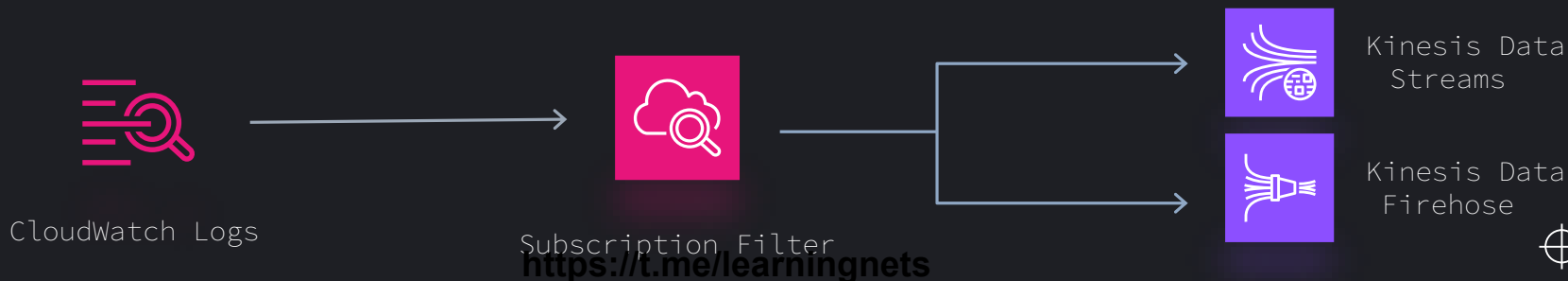




AWS CloudWatch Log Filtering Subscription

Filter log data using a Metric or Subscription filter before sending it to a destination.

- **Metric filter:** Extract data from log events to create custom metrics
- **Subscription filter:** Filter log data being sent to other AWS services

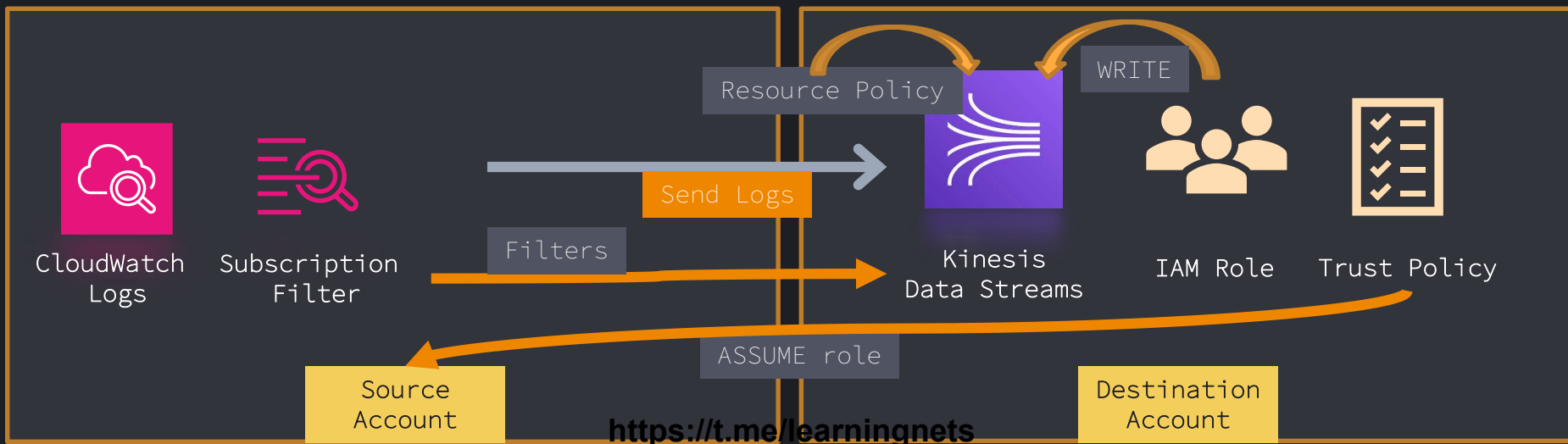




Cross Accounts Access



- 1) Setup Data Stream in Destination Account
- 2) Create IAM role + Trust Policy in Destination Account to write to Stream
- 3) Setup Subscription Filter in Source Account





AWS CloudWatch Logs Agent

- EC2 does not send any data to CloudWatch, to send its logs to CloudWatch – a logs agent is needed.
 - A CloudWatch Logs Agent is a lightweight, standalone agent that can be installed on EC2 instances/On-prem servers
 - It collects and streams log data from EC2 instances/On-prem servers to CloudWatch Logs in near real-time.
- Types of log agents:
 - CloudWatch logs agent
 - CloudWatch Unified logs agent



AWS CloudWatch Logs Agent

Logs Agent

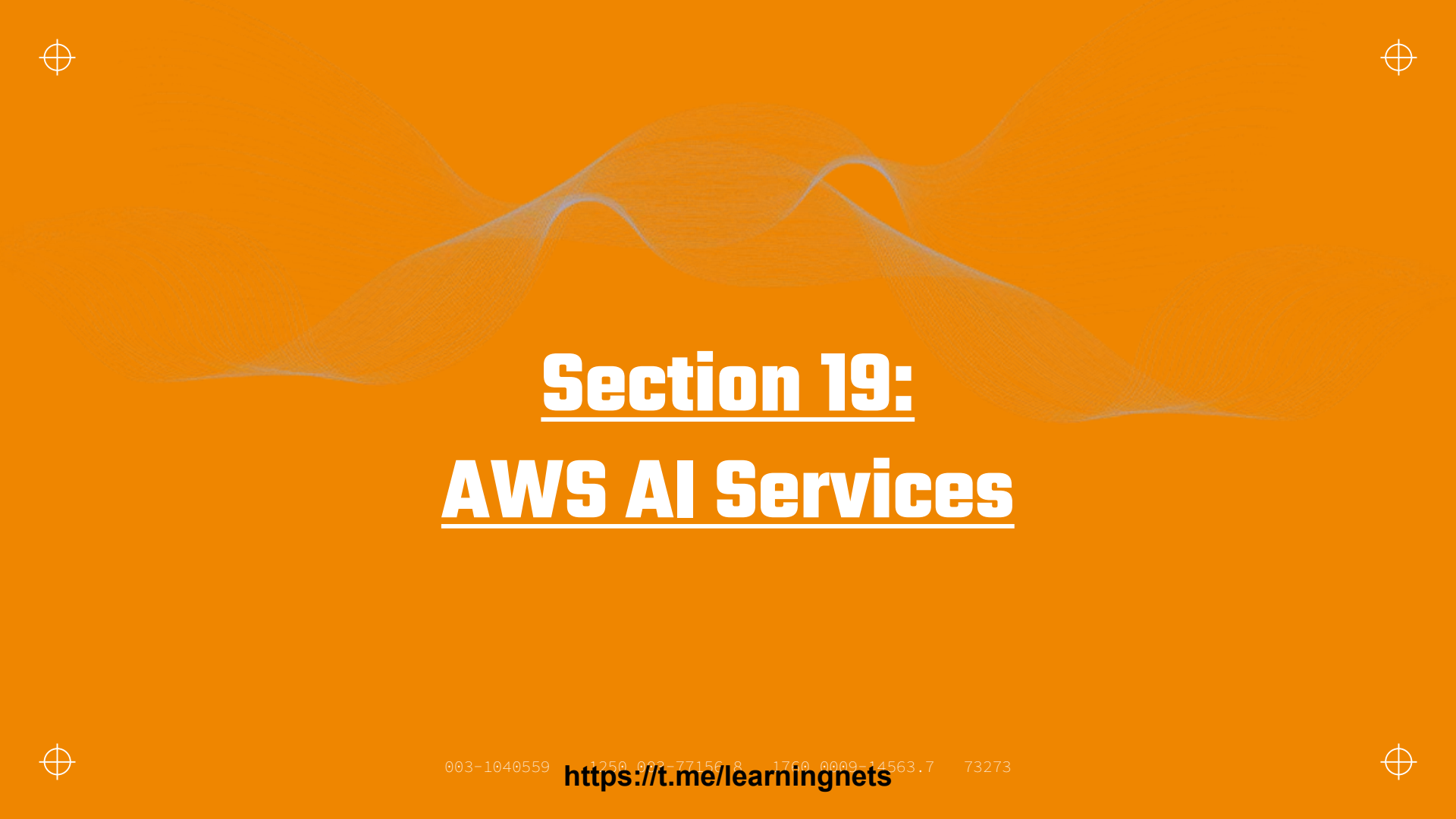
- Older version with limited capabilities
- Collects logs only

Unified CloudWatch agent

- Enhanced logs agent
- Collects logs as well as system-level metrics
- Collects RAM, CPU Utilization, Memory Usage, Disk Space, Network Traffic, and Swap Space metrics

Note: AWS mostly recommends
Unified CloudWatch agent





Section 19: AWS AI Services

003-1040559 1250.000-77156.8 1750.0009-14563.7 73273

<https://t.me/learningnets>



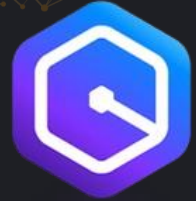
Amazon Q Business

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Amazon Q Business

- AI-powered assistant.
- Intelligent answers, content generation, summaries and task automations.
- Accessible via web interface or APIs.
- Can integrate with other business platforms like Teams, Slack.



Amazon Q Business

Key Features:

Enterprise Integration

⇒ +40 pre-built connectors, Salesforce, Jira, ServiceNow, Zendesk...

User-Friendly Interface

⇒ Web based interface, integration with Microsoft Teams, Slack.

Rapid Deployment

⇒ Quick setup, without any code.

Managed Infrastructure

⇒ need for managing infrastructure.

Access Control

⇒ Respects user permissions within integrated enterprise applications.

Data Integration

⇒ Amazon S3, Salesforce, Oracle, Google Drive, Microsoft 365, and more.

Administrative Controls

⇒ specific guardrails and controls.

<https://t.me/learningnets>

Amazon Q Business

Use Cases:

Content Creation

Marketing & Sales:
Generate blog posts,
social media headlines.

Research:
Summarize academic
papers, create new
sections.

Enterprise Use-Cases

Knowledge Management:
Find specific docs like company
policies.

Support:
Get customer support for common
issues.

Executive Summaries:
Summarize long meetings and
project reports.

Key Insight Generation

Comparative Analysis:
Compare documents,
identify trends.

Market Research:
Analyze market research,
get insights

Amazon Q Business

Technical Details:

- IAM Identity Center:
 - **Purpose:** Manages user access
 - **Function:** Connects existing identity provider. Users can interact with Amazon Q Business.
- Retrieval Augmented Generation (RAG):
 - **Purpose:** Enhances gen AI models with up-to-date information.
 - **Function:** Retrieves relevant data from external sources. Ensures response accuracy.
- Enterprise Data Access Control:
 - **Purpose:** Ensures data security and compliance.
 - **Function:** Respects user permissions and integrates with SAML 2.0 supported identity providers like Microsoft Entra ID.
- Data Integration and Updates:
 - **Purpose:** Connects Amazon Q Business to various enterprise data sources..
 - **Function:** Uses pre-built connectors for easy integration.
- Plugins:
 - **Purpose:** Extends Amazon Q business' functionality
 - **Function:** Allows interaction with popular 3rd party applications like Jira, ServiceNow...



Amazon Transcribe

Automatic Speech Recognition

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Amazon Transcribe

What is Amazon Transcribe?

- **Definition:** Automatic Speech Recognition (ASR) service
- **Key Functions:**
 - Converts audio to text
 - Supports multiple languages and accents

Subtitles for videos

Call recording transcripts

Dictation transcripts

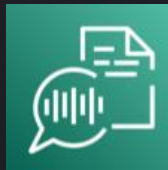
Amazon Transcribe

What is Amazon Transcribe?

- **Definition:** Automatic Speech Recognition (ASR) service
- **Key Functions:**
 - Converts audio to text
 - Supports multiple languages and accents



Input Files
Amazon S3



Amazon Transcribe



Default Transcription
Files Amazon S3

Amazon Transcribe

Key features:

Real-time Transcription

Transcribe audio streams in real-time with low latency

Batch Transcription

Transcribe pre-recorded audio files in batches

Custom Vocabulary

Add industry-specific terms for improved accuracy

Speaker identification

Identify and label different speakers in a conversation

Auto Punctuation

Automatically add punctuation and formatting to transcripts

Multi-language Support

Transcribe audio in multiple languages with high accuracy

Amazon Transcribe

Use Cases:

Subtitling and Closed Captioning

Add subtitles and closed captions to videos for accessibility and global audiences

Call Center Analytics

Analyze call transcripts to improve customer service and agent performance

Meeting Transcription

Automatically transcribe meetings to capture action items and decisions

Content Creation

Transcribe audio from podcasts and videos to generate articles, show notes, and more

Compliance and Regulation

Ensure compliance with regulations by transcribing sensitive audio recordings

Amazon Transcribe

Advantages:

Scalable

Handling varying volumes of audio content
⇒ Easy to scale with your business needs

Accurate

Advanced deep learning algorithms
⇒ high accuracy in transcriptions
⇒ continuously improving with usage

Cost-efficient

Pay-as-you-go model
⇒ significantly reducing costs compared to traditional transcription (manual labor)

Easy Integration

Integrates with existing services
⇒ Quick deployment + minimal disruption

Amazon Transcribe

Advantages

Feature	Amazon Transcribe	Traditional Transcription
Speed	Fast, near-instant processing	Slow, dependent on human effort
Accuracy	High accuracy with continuous learning	Variable accuracy, prone to human error
Cost	Cost-effective, pay-per-use model	Generally higher, fixed costs
Scalability	Highly scalable for large volumes	Difficult to scale
Customization	Supports custom vocabulary	Limited customization options
Integration	Seamless integration with AWS services	Requires manual setup

Amazon Transcribe

Advantages

Upload or Stream Audio

- Users can upload audio files or stream live audio directly to Amazon Transcribe.

Choose Transcription Settings

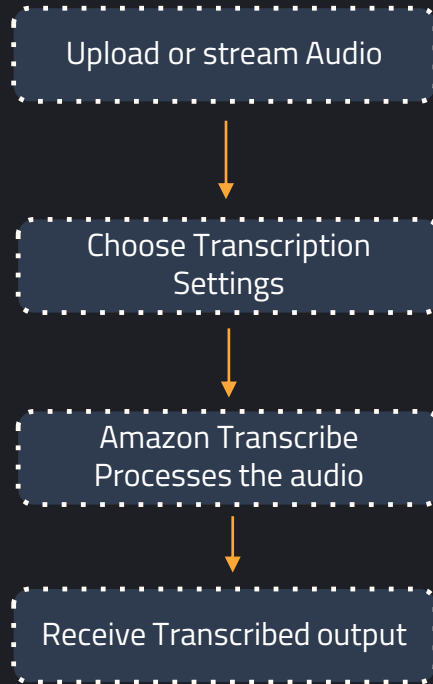
- Configure settings such as language, custom vocabulary, and speaker identification options.

Amazon Transcribe Processes the Audio

- The service uses advanced algorithms to analyze the audio and generate a text transcript.

Receive Transcribed Text

- Users receive the transcribed text in a structured format, ready for use.



Amazon Transcribe

Supported File Formats

Audio Formats:

- MP3
- MP4
- WAV
- FLAC
- OGG
- PCM encoding

AWS Service Integrations

- **Amazon S3:** For storing audio files and transcripts.
- **AWS Lambda:** For automating workflows and processing audio files.
- **Amazon Comprehend:** For analyzing transcribed text to extract insights.
- **Amazon Translate:** For translating transcripts into different languages.



Amazon Transcribe

Compliance and Security

HIPAA Eligible

- Eligible for use in systems covered by the Health Insurance Portability and Accountability Act (HIPAA)
⇒ Ensuring compliance for sensitive healthcare data.

Encryption in Transit and at Rest

- All data transmitted to and from Amazon Transcribe is encrypted using HTTPS.
- Audio files and transcripts are encrypted at rest using AES-256 encryption.

Abstract blue wavy lines, resembling sound waves or smoke, flowing across the top half of the slide.

Amazon Polly

Text-to-Speech

003-1040559 1250.032-77156.8 1760.0009-14563.7 73273

<https://t.me/learningnets>

Amazon Polly

What is Amazon Polly?

- Text-to-Speech(TTS) service.
- **Common Use Cases:**
 - **Media And Entertainment:** Voice-overs for videos and eBooks
 - **Business:** Interactive voice Response(IVR) and automated customer service
 - **Education:** Audio for learning materials

Amazon Polly

Key features:

Natural-Sounding Voices

Lifelike speech synthesis that enhance user experience.

Wide Language Support

Multiple languages and adjustable voices for global reach.

Custom Lexicons

Define pronunciations for specific terms and names.

Speech Marks

Detailed information for synchronization with visual elements.

Real-Time

Generate speech on-the-fly for immediate applications.

Amazon Polly

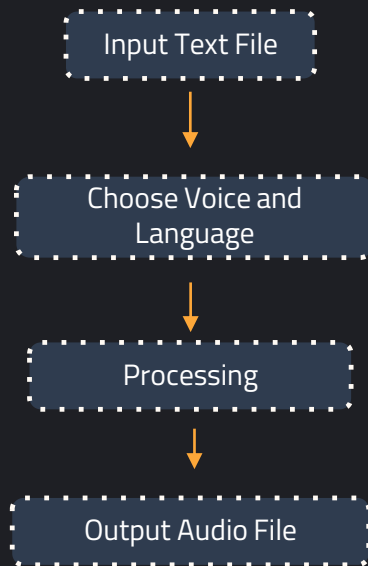
Process of Converting Text to Speech:

Input Text – Start with your written Content

Choose Voice and Language - Select from available options

Process Text - Amazon Polly synthesizes the speech.

Receive Audio Output - Get the final audio file



Amazon Polly

Supported Formats and Integrations

Audio Formats:

- **MP3**

Compressed, widely supported for mobile/web

- **OGG (Vorbis)**

High quality, smaller file sizes

- **PCM(WAV)**

Uncompressed, ideal for IoT devices

AWS Integrations:

- S3: Storage

- Lambda: Processing

- CloudFront: Content delivery



Amazon Polly

Compliance and Security

Data Security:

Supports encryption for data both in transit and at rest.

Compliance:

Amazon Polly is HIPAA-eligible.