

AWS re:Inforce

JUNE 13 - 14, 2023 | ANAHEIM, CA

APS208

Securely build generative AI applications and control data with Amazon Bedrock

Dr. Andrew Kane

WW Tech Leader – AI Services,
Amazon Web Services

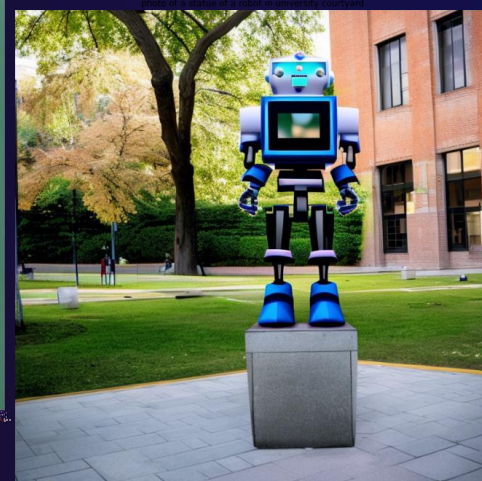
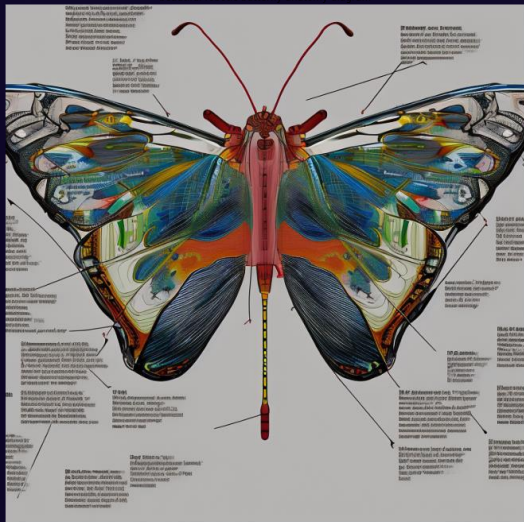
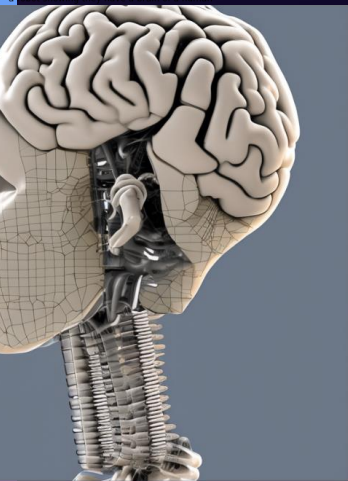
Mark Ryland

Director, Office of the CISO,
Amazon Web Services



Today's agenda

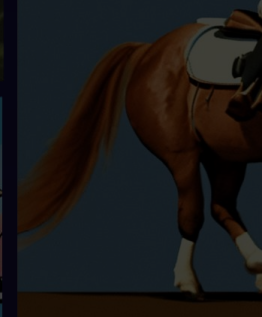
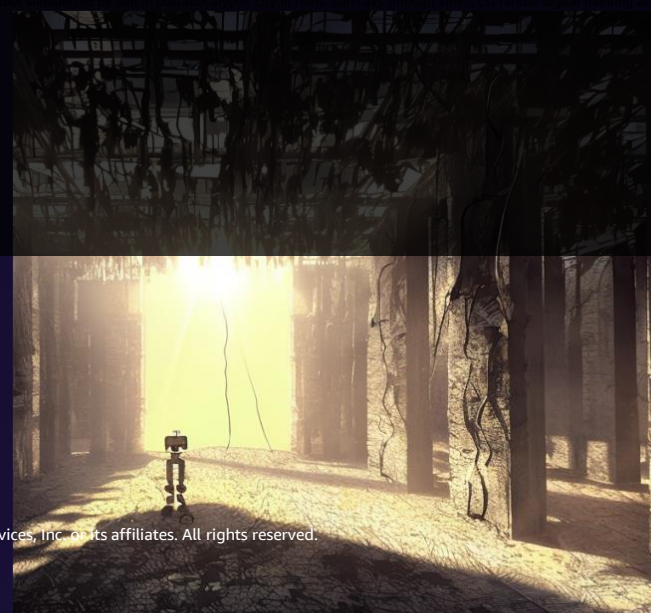
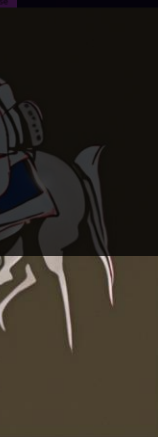
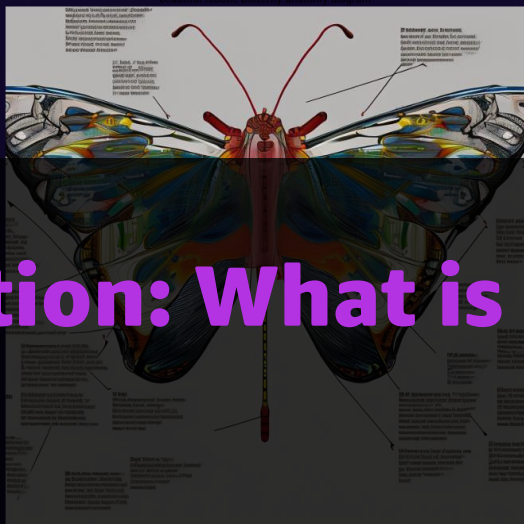
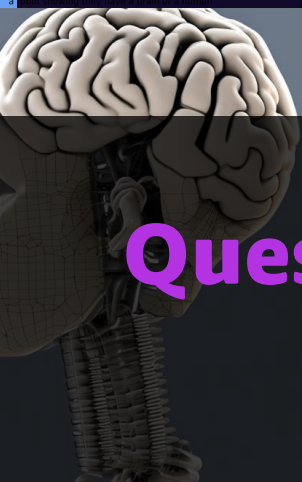
- Generative AI and foundation models (FMs)
- Introducing Amazon Bedrock
- Data privacy and security
- Model tenancy
- Client connectivity
- Access management
- Security in the models: challenges and opportunities
- Amazon SageMaker and FMs



© 2023, Amazon Web Services, Inc. or its affiliates.

© 2023, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Question: What is generative artificial intelligence (AI)?



Question: What is generative artificial intelligence (AI)?

- Creates new content and ideas, including conversations, stories, images, videos, and music
- Powered by large models that are pretrained on vast corpora of data and commonly referred to as foundation models (FMs)

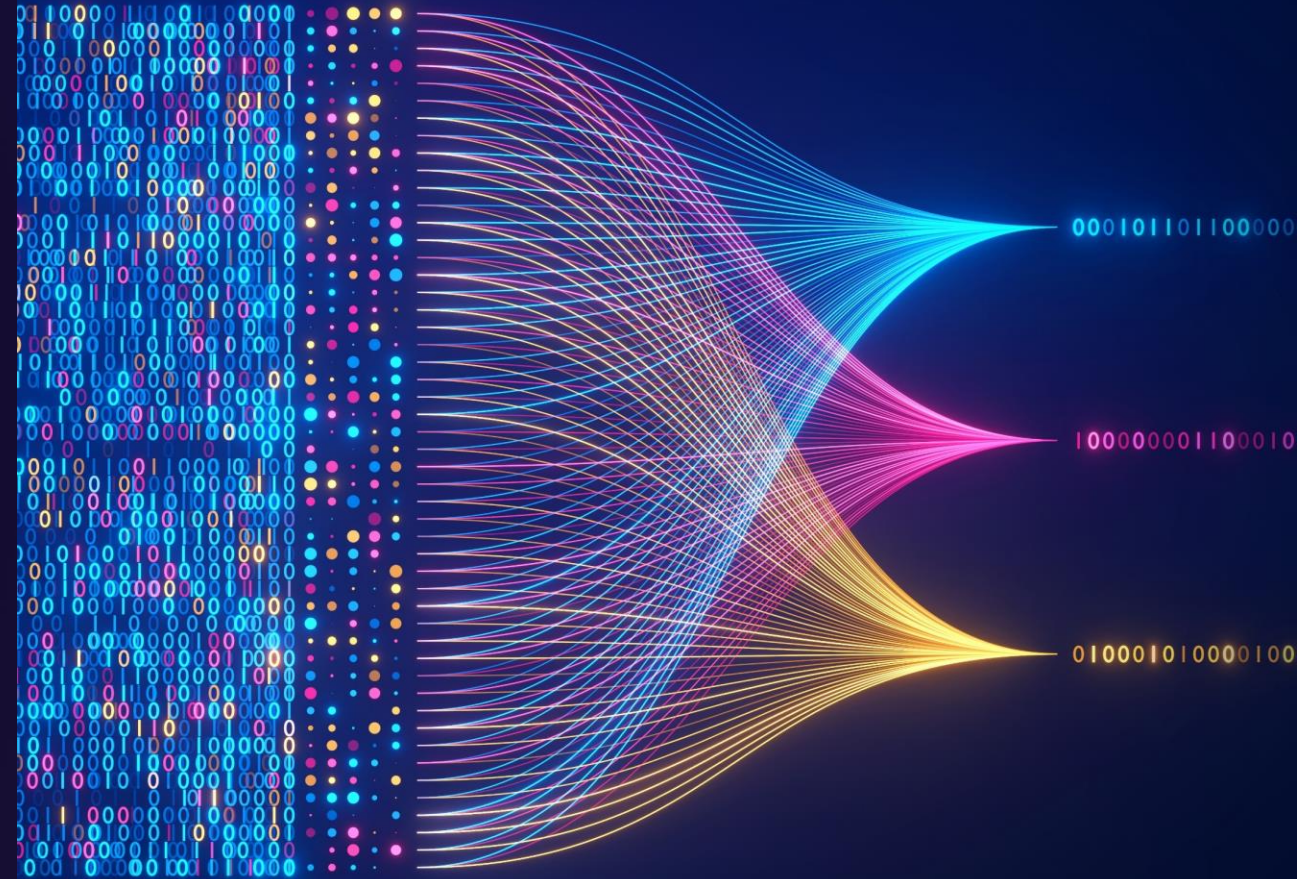
Generative AI is powered by foundation models

Pretrained on vast amounts of unstructured data

Contain large number of parameters that make them capable of learning complex concepts

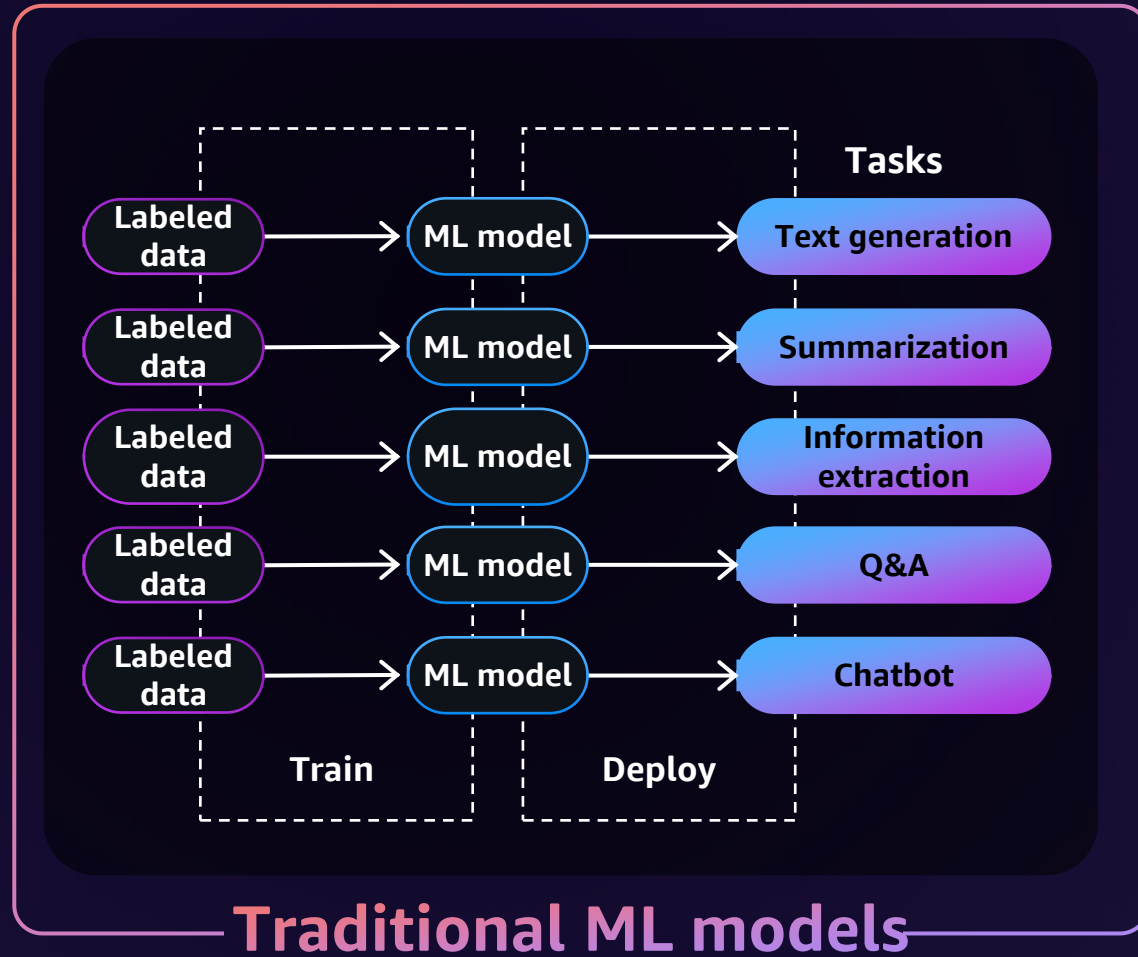
Can be applied in a wide range of contexts

Customize FMs using your data for domain specific tasks

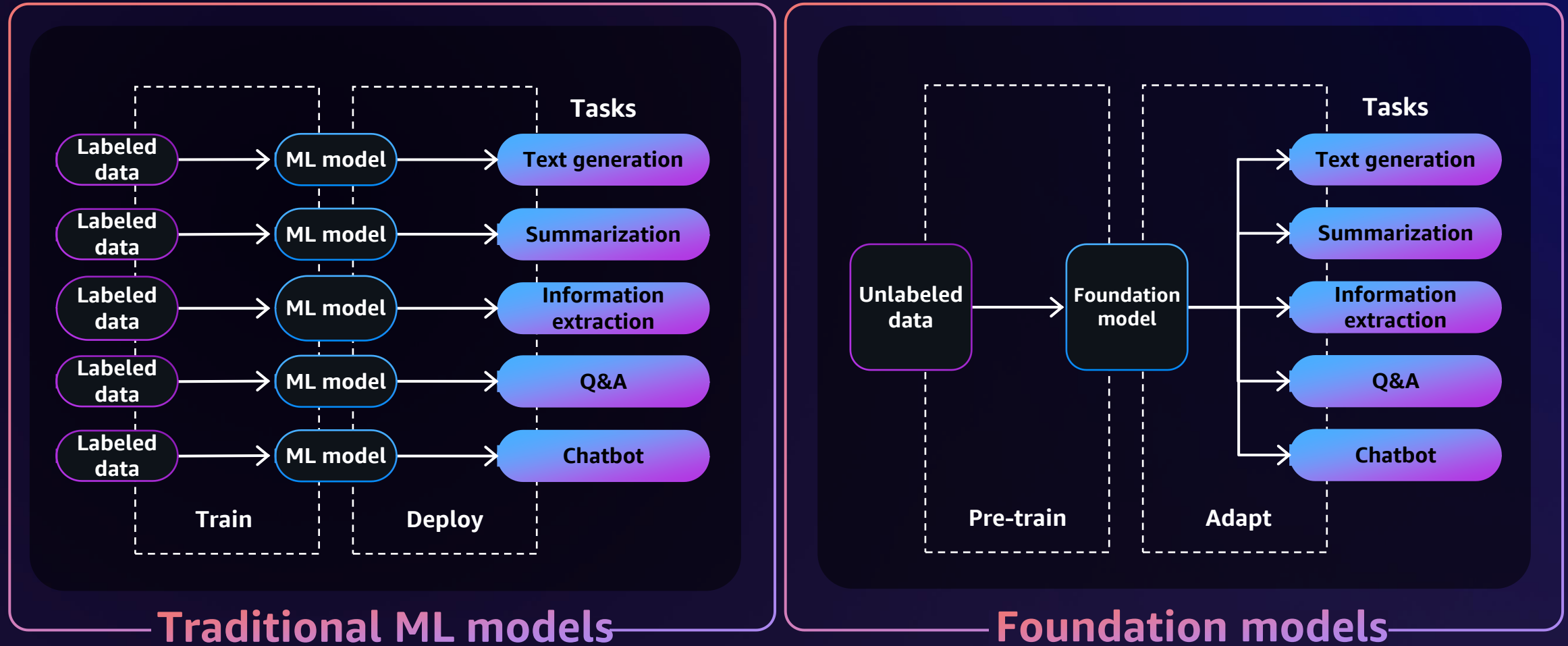


How foundation models differ from other ML models

How foundation models differ from other ML models



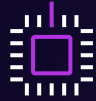
How foundation models differ from other ML models



Unlocking the potential of generative AI



The easiest way to build with FMs



The most price-performant infrastructure



Generative AI-powered applications



Flexibility

PRIVATE BETA

Amazon Bedrock

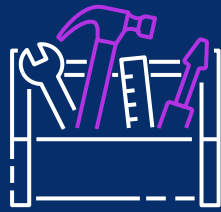
The easiest way to build
and scale generative AI
applications with
foundation models (FMs)



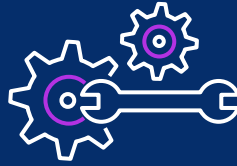
Amazon Bedrock key benefits



Accelerate development of generative AI applications using FMs through an API, without managing infrastructure



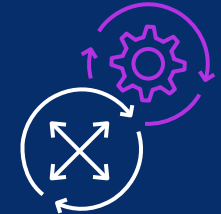
Choose FMs from AI21 Labs, Anthropic, Stability AI, and Amazon to find the right FM for your use case



Privately customize FMs using your organization's data



Enhance your data protection using comprehensive AWS security capabilities



Responsible AI provided by supported model providers; Amazon Titan supports AI best practices

Amazon Bedrock supports leading FMs

The Amazon logo, featuring the word "amazon" in a black, lowercase, sans-serif font with a curved orange arrow underneath it.

Amazon Titan

Amazon Titan FMs are a family of models built by Amazon that are pretrained on large datasets, which makes them powerful, general-purpose models

The AI21 Labs logo, featuring the text "AI21" in black and "labs" in a pink, lowercase, sans-serif font.

Jurassic-2

Multilingual LLMs for text generation in Spanish, French, German, Portuguese, Italian, and Dutch

The Anthropic logo, featuring the word "ANTHROPIC" in a black, uppercase, sans-serif font.

Claude

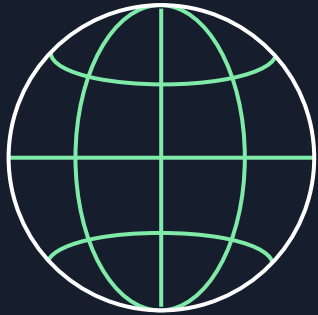
LLM for conversations, question answering, and workflow automation based on research into training honest and responsible AI systems

The Stability AI logo, featuring the text "stability.ai" in a black, lowercase, sans-serif font.

Stable Diffusion

Generation of unique, realistic, high-quality images, art, logos, and designs

Quickly integrate FMs into your applications



Quickly integrate and deploy FMs into your applications and workloads running on AWS using familiar controls and integrations with the depth and breadth of AWS capabilities and services like Amazon SageMaker and Amazon S3

Isolated customization to drive differentiation



Private fine-tuning

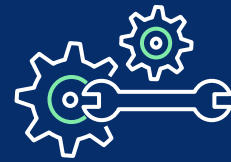
PURPOSE

Maximizing accuracy for specific tasks

DATA NEED

Small number of labeled examples

Isolated customization to drive differentiation



Private fine-tuning

PURPOSE

Maximizing accuracy for specific tasks

DATA NEED

Small number of labeled examples

Data privacy and localization



You are always in control of your data

- Customer data is not used to improve the Amazon Titan models for other customers and is not shared with other foundation model providers
- Customer data (prompts, responses, fine-tuned models) are isolated per customer and remain in the Region where they were created

Data security



You are always in control of your data

- Customer data is always encrypted in transit with a minimum of TLS1.2 and AES-256 encrypted at rest using AWS KMS managed data encryption keys
- Integration with AWS Identity and Access Management Service (IAM) to manage inference access, allow/deny access for specific models, enable AWS Management Console access, etc.
- Use AWS CloudTrail to monitor all API activity and troubleshoot issues as you integrate with applications
- Fine-tuned (customized) models are encrypted and stored using customer AWS KMS key. Only you have access to your customized models

Configurable security controls

Configurable security controls



Data privacy



Single-tenancy

Multi-tenancy

Model tenancy



Service-owned keys

Customer-managed keys

AWS KMS data encryption



Model fine tuning



Access management

Model tenancy



Single-tenant
endpoint



Multi-tenant
endpoint

Model tenancy



Single-tenant
endpoint



Multi-tenant
endpoint

1. Deployment available to a single customer

Model tenancy



Single-tenant
endpoint

1. Deployment available to a single customer



Multi-tenant
endpoint

1. Deployment available to all customers

Model tenancy



Single-tenant
endpoint

1. Deployment available to a single customer
2. Holds a single version of a baseline 1P/3P model that has been fine-tuned by a customer



Multi-tenant
endpoint

1. Deployment available to all customers

Model tenancy



Single-tenant
endpoint

1. Deployment available to a single customer
2. Holds a single version of a baseline 1P/3P model that has been fine-tuned by a customer



Multi-tenant
endpoint

1. Deployment available to all customers
2. Holds a baseline version of each supported 1P/3P model

Model tenancy



Single-tenant endpoint

1. Deployment available to a single customer
2. Holds a single version of a baseline 1P/3P model that has been fine-tuned by a customer



Multi-tenant endpoint

1. Deployment available to all customers
2. Holds a baseline version of each supported 1P/3P model

3. No inference request's input or output text is used to train any model(s) in the deployment

Model tenancy



Single-tenant endpoint

1. Deployment available to a single customer
2. Holds a single version of a baseline 1P/3P model that has been fine-tuned by a customer



Multi-tenant endpoint

1. Deployment available to all customers
2. Holds a baseline version of each supported 1P/3P model

3. No inference request's input or output text is used to train any model(s) in the deployment
4. Model deployments are inside an AWS account owned and operated by the Bedrock service team

Model tenancy



Single-tenant endpoint

1. Deployment available to a single customer
2. Holds a single version of a baseline 1P/3P model that has been fine-tuned by a customer



Multi-tenant endpoint

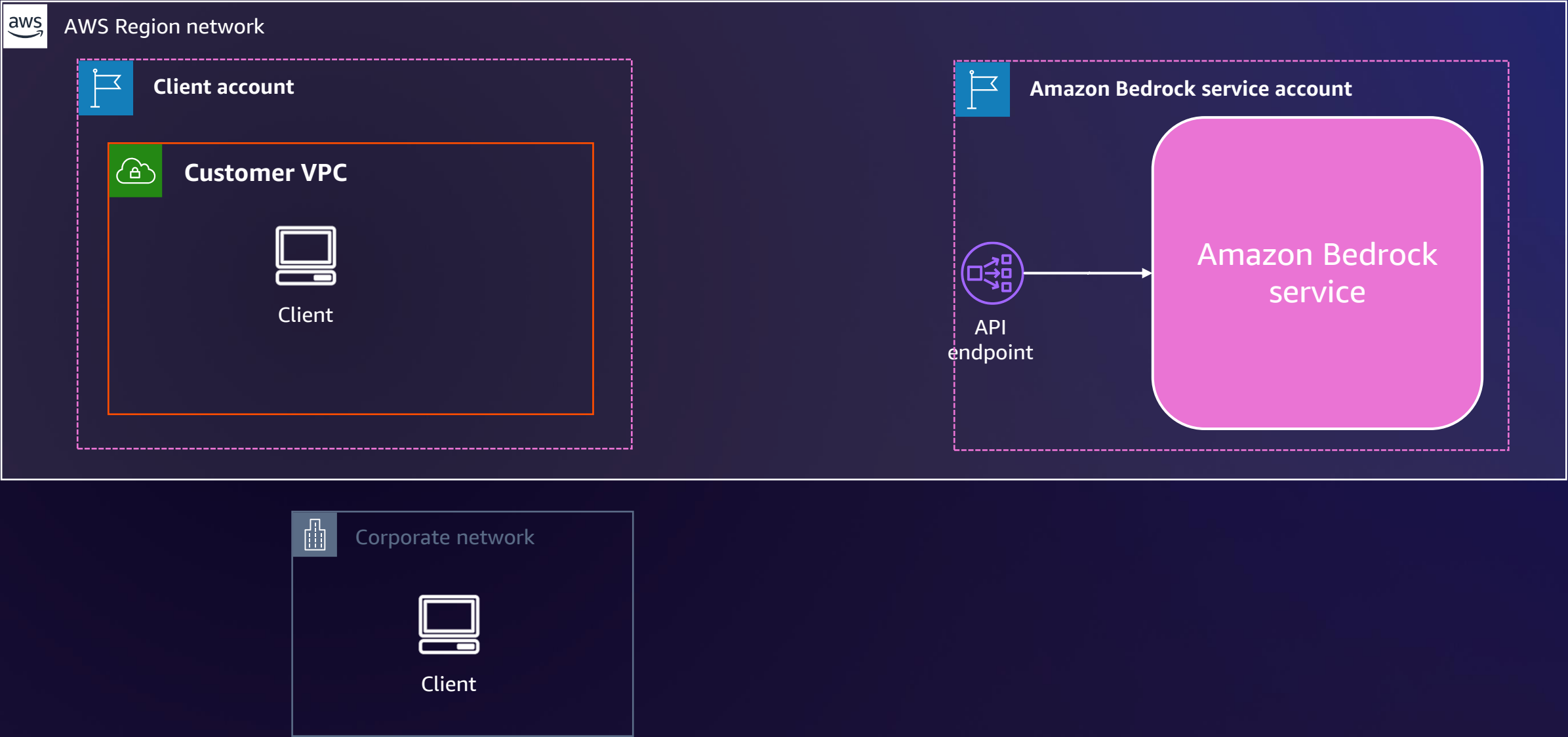
1. Deployment available to all customers
2. Holds a baseline version of each supported 1P/3P model

3. No inference request's input or output text is used to train any model(s) in the deployment
4. Model deployments are inside an AWS account owned and operated by the Bedrock service team
5. Model vendors have no access to any customer data

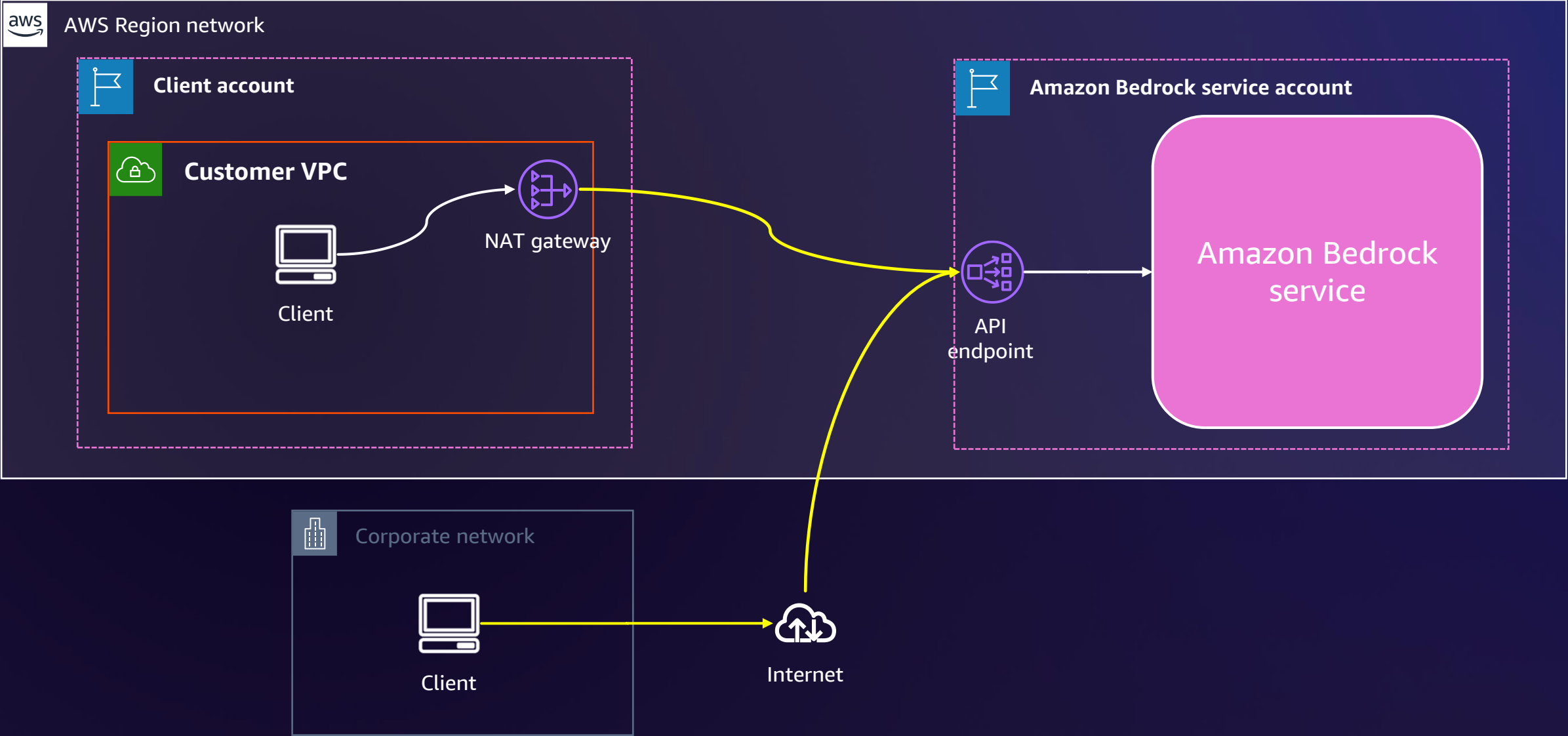
Client connectivity



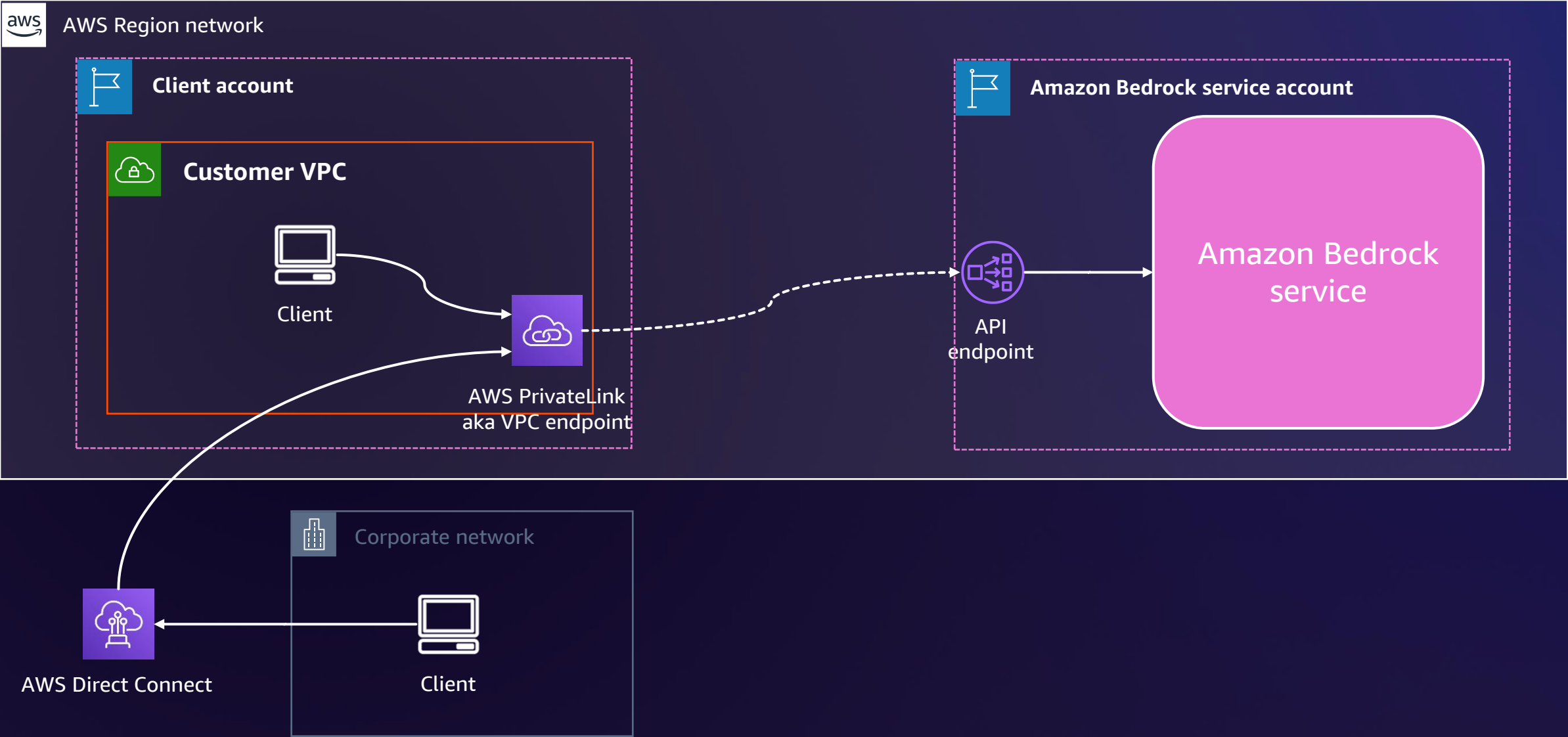
Client connectivity



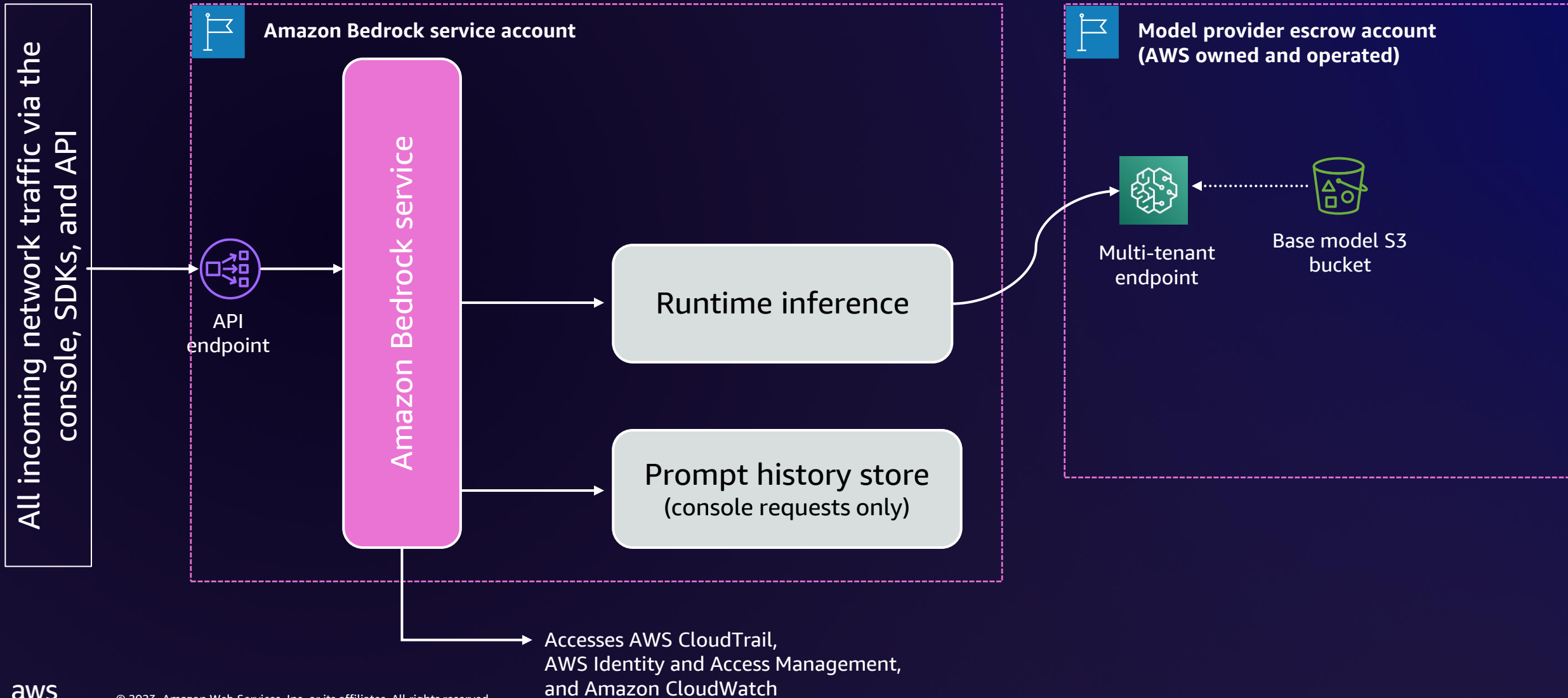
Client connectivity



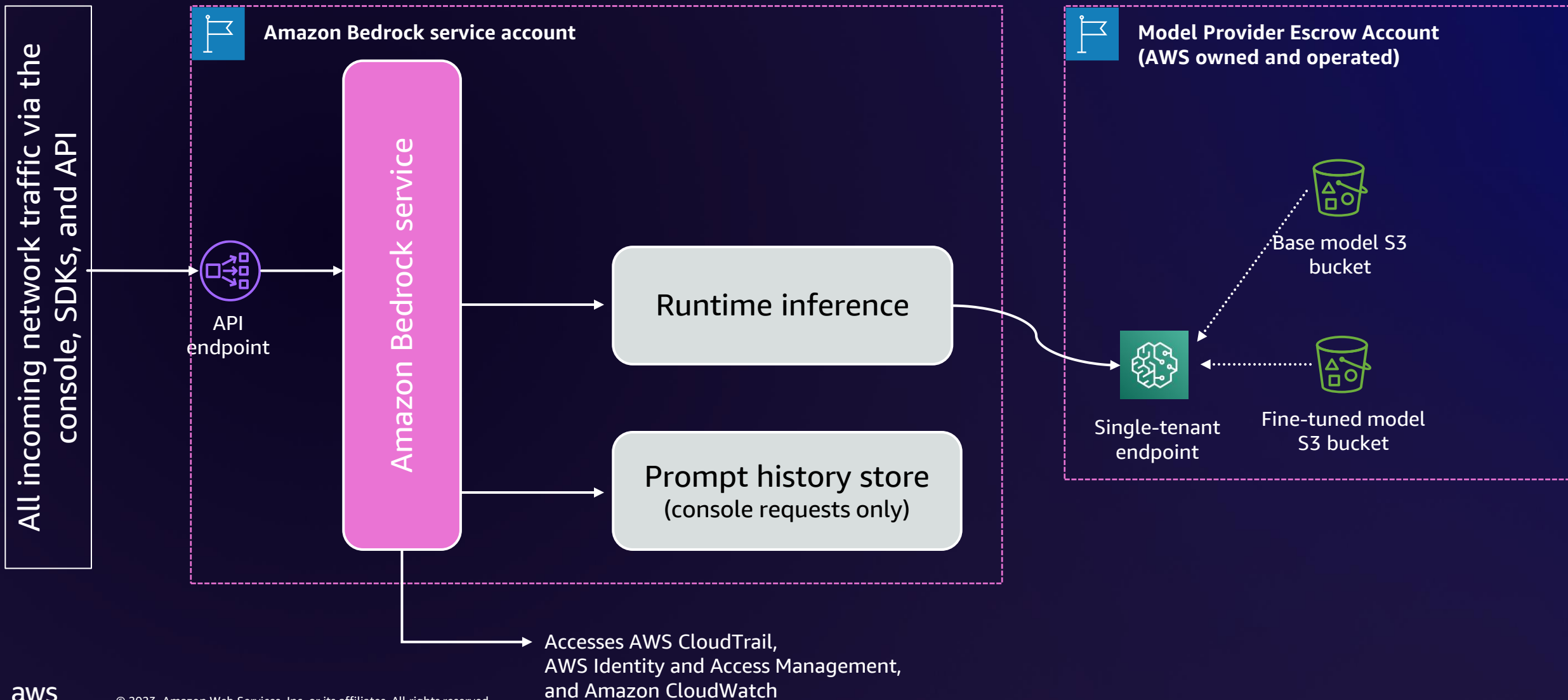
Client connectivity



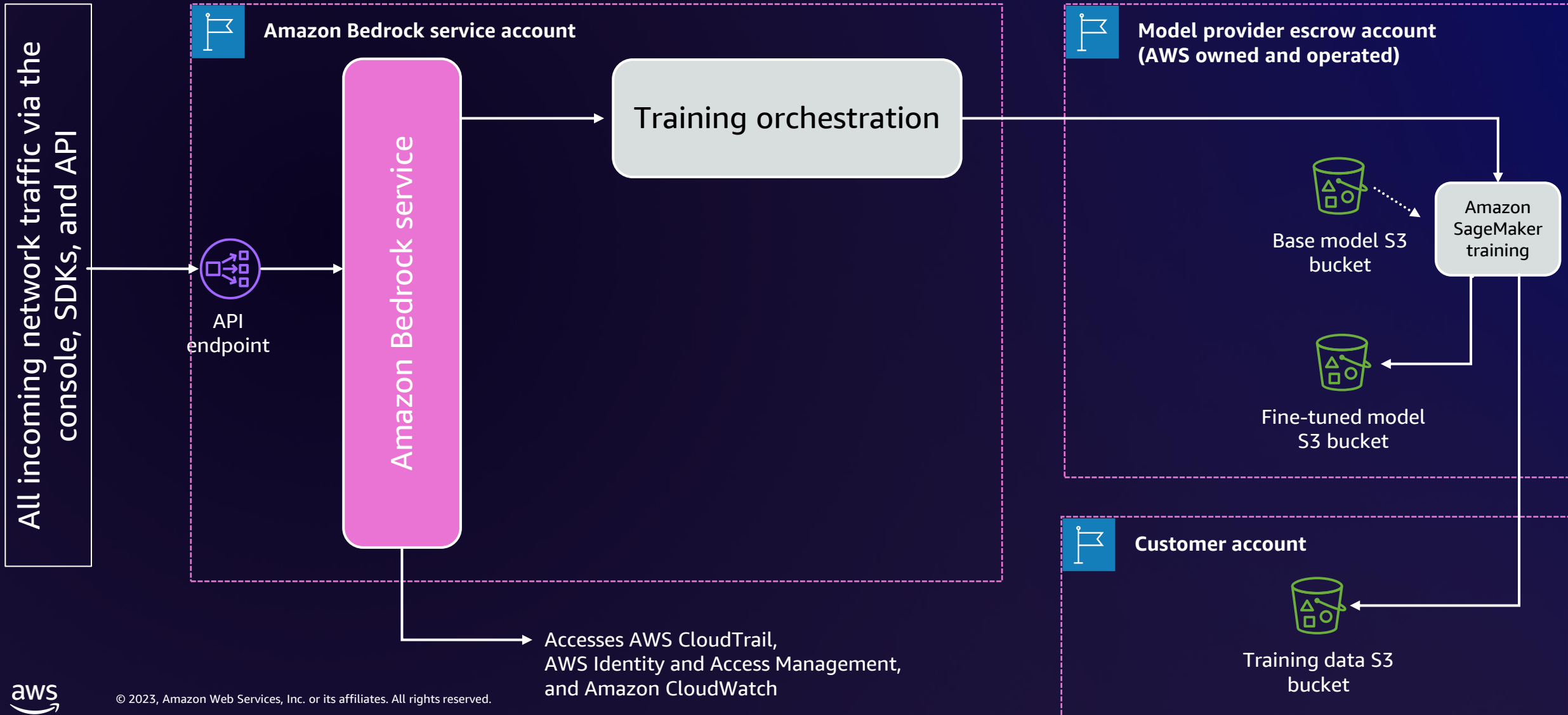
Multi-tenancy inference



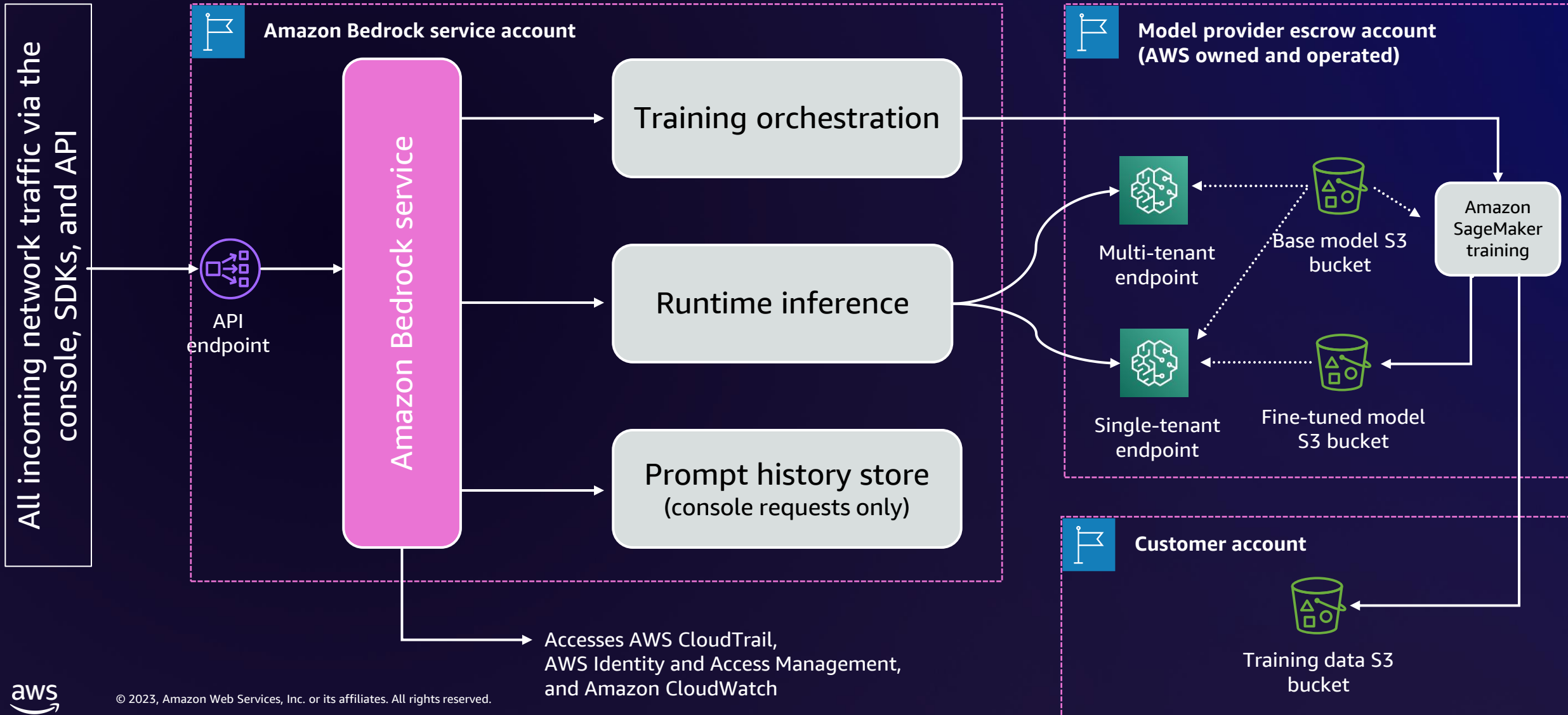
Single-tenancy inference



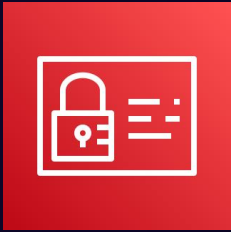
Model fine tuning



Complete architecture overview



AWS Identity and Access Management



IAM

- Identity-based policies
- Actions
- Resources
- Tags (ABAC)

IAM/SCP – Example deny policy

```
{  
  "Version": "2012-10-17",  
  "Statement":  
  {  
    "Sid": "DenyInferenceForModelX",  
    "Effect": "Deny",  
    "Action": "bedrock:InvokeModel",  
    "Resource": "arn:aws:bedrock:::foundation-model/<name-of-model>"  
  }  
}
```

Security in the model: challenges and opportunities

Challenges

- ✓ Use of generative AI and FMs for illegitimate or malicious purposes
- ✓ Prompt manipulation to avoid model protections and filters
- ✓ Risks of erroneous or otherwise undesirable outputs

Opportunities

- ✓ Enhanced security tools and UXs based on FMs
- ✓ Domain-focused FMs and model tuning reduce risks
- ✓ Supporting human judgment rather than closed-loop automation

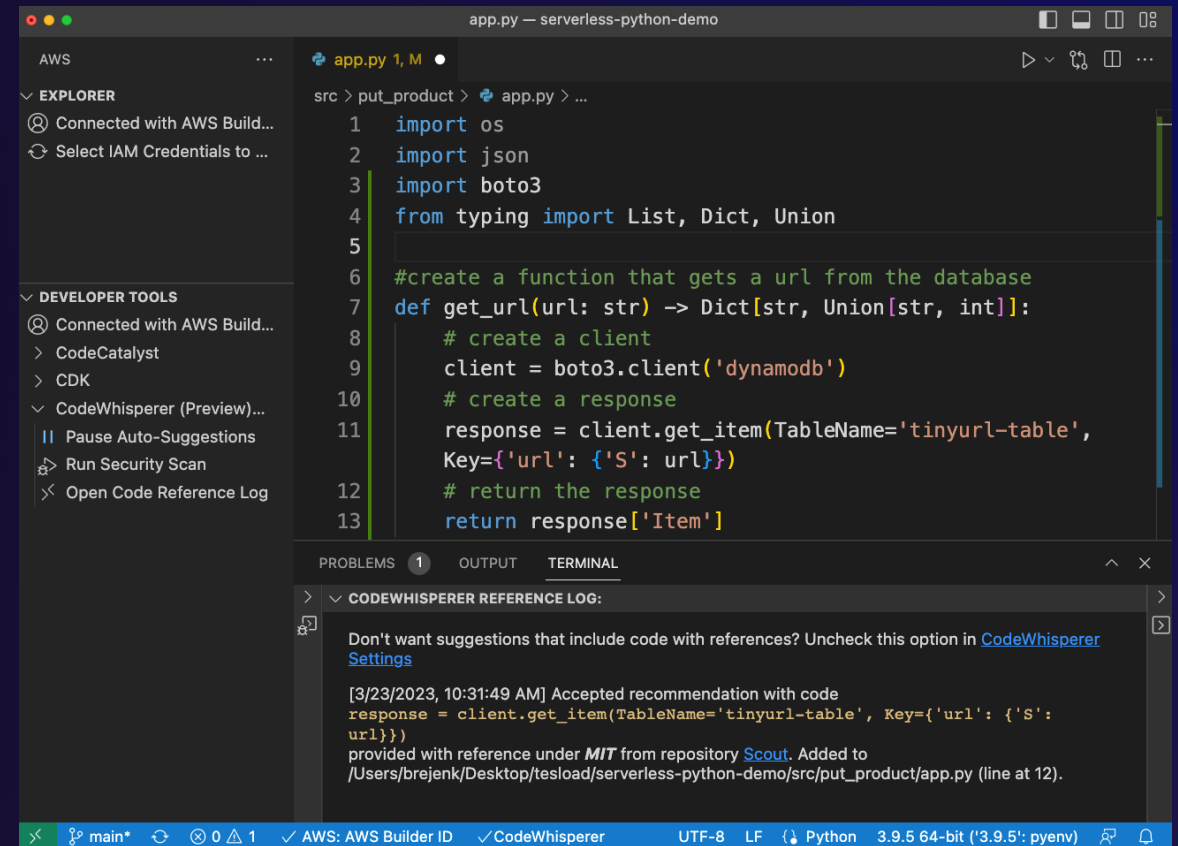
Example: Amazon CodeWhisperer

Reference tracking

- Flags code similar to open-source training data
- Tracks accepted suggestions so that you can provide appropriate attribution
- Enterprise controls to more easily deactivate/filter code suggestions similar to open-source training data

Security scanning

- Scan generated and developer-written code to detect security vulnerabilities
- Receive vulnerability remediation suggestions



The screenshot displays the AWS CodeWhisperer interface within a code editor. The main editor shows a Python file named `app.py` with the following code:

```
1 import os
2 import json
3 import boto3
4 from typing import List, Dict, Union
5
6 #create a function that gets a url from the database
7 def get_url(url: str) -> Dict[str, Union[str, int]]:
8     # create a client
9     client = boto3.client('dynamodb')
10    # create a response
11    response = client.get_item(TableName='tinyurl-table',
12                               Key={'url': {'S': url}})
13    # return the response
14    return response['Item']
```

The interface includes a sidebar with the Explorer and Developer Tools panels. The Explorer panel shows the project structure, and the Developer Tools panel shows the CodeWhisperer (Preview) settings. The CodeWhisperer panel is open, displaying a reference log for the code suggestion on line 12:

```
[3/23/2023, 10:31:49 AM] Accepted recommendation with code
response = client.get_item(TableName='tinyurl-table', Key={'url': {'S':
url}})
provided with reference under MIT from repository Scout. Added to
/Users/brejenk/Desktop/tesload/serverless-python-demo/src/put_product/app.py (line at 12).
```

Build your own FM at scale using Amazon SageMaker



Managed infrastructure

Full control of your model training with managed & most price-performant infrastructure



Efficient distributed training

Complete distributed training up to 40% faster



Debugging and experimentation tools

Capture metrics and profile training jobs in real time to quickly correct performance issues. Track ML model iterations easily.



Price-performant inference

Deploy models in production for any use case with best price-performance



Repeatable and reproducible MLOps

Automate and standardize processes across the ML lifecycle



Governance

Purpose-built governance tools to help you implement ML responsibly



Human-in-the-loop support

Create high quality datasets and align model outputs with human preferences

More models on Amazon SageMaker JumpStart

Publicly available

stability.ai

Models

Text2Image
Upscaling

Tasks

Generate
photo-realistic
images from
text input
Improve quality
of generated
images

Features

Fine-tuning
available with
Stable Diffusion
2.1



Models

AlexaTM
20B

Tasks

Machine
translation
Question
answering
Summarization
Annotation
Data generation



Models

Flan T-5 (8
variants)
DistilGPT2, GPT2
Bloom
(3 variants)

Tasks

Machine
translation
Question
answering
Summarization
Annotation
Data generation

Proprietary models

co:here

Models

Cohere
Command

Tasks

Text generation
Information
extraction
Question
answering
Summarization

Light

Models

Lyra-Fr
10B

Tasks

Text generation
Keyword
extraction
Information
extraction
Question
answering
Summarization
Sentiment
analysis
Classification

AI21labs

Models

Jurassic-2
Jumbo

Tasks

Text generation
Long-form
generation
Summarization
Paraphrasing
Chat
Information
extraction
Question
answering
Classification

How to use JumpStart for building with FMs

1

Choose from more FMs offered by model providers

AI21 labs

Lightn
We bring Light to AI

stability.ai

co:here



alexa

2

Try out model and/or deploy



Try out models via the console



Deploy the model for inference using SageMaker hosting options includes single node

3

Fine tune model and automate ML workflow



Only selected models can be fine-tuned



Automate ML workflow

Data stays in your account including model, instances, logs, model inputs, model outputs

Fully integrated with Amazon SageMaker features

Start your generative AI journey today

1

Start driving
productivity
with Amazon
CodeWhisperer
today

2

Explore FMs
through Amazon
Bedrock and other
FMs on JumpStart

3

Get started on a
PoC for your top
use cases

Thank you!

Dr. Andrew Kane



andrewjkane



@drandrewkane

Mark Ryland



markryland



@mpryland



Please complete
the session survey
in the mobile app