AWS re:Invent

NOV. 28 – DEC. 2, 2022 | LAS VEGAS, NV

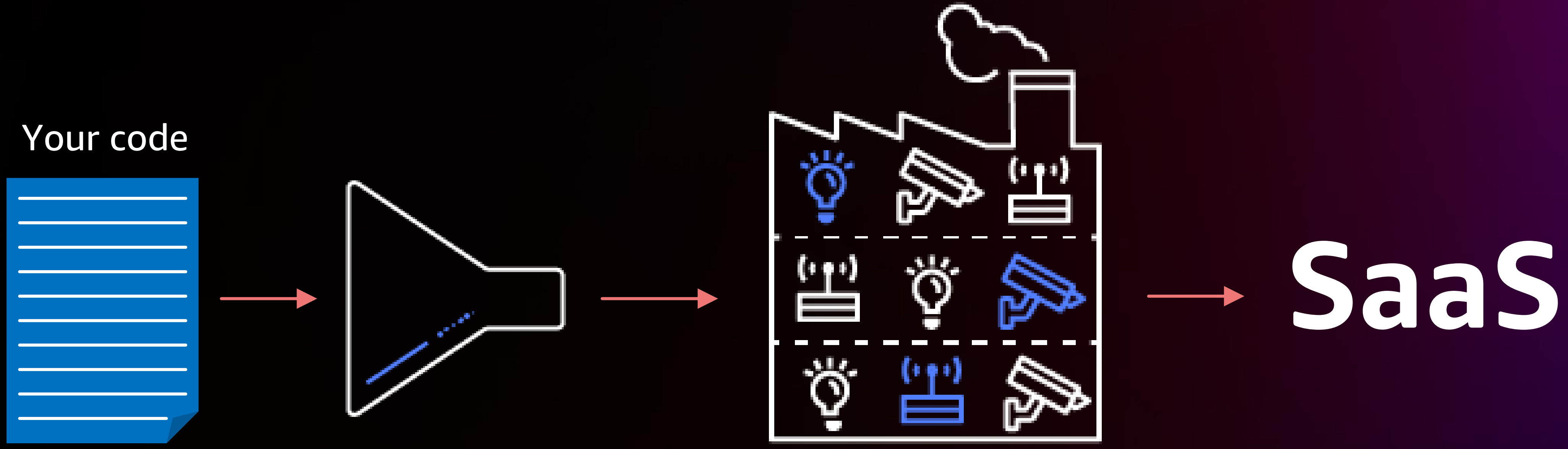
SAS305-R

SaaS architecture patterns: From concept to implementation

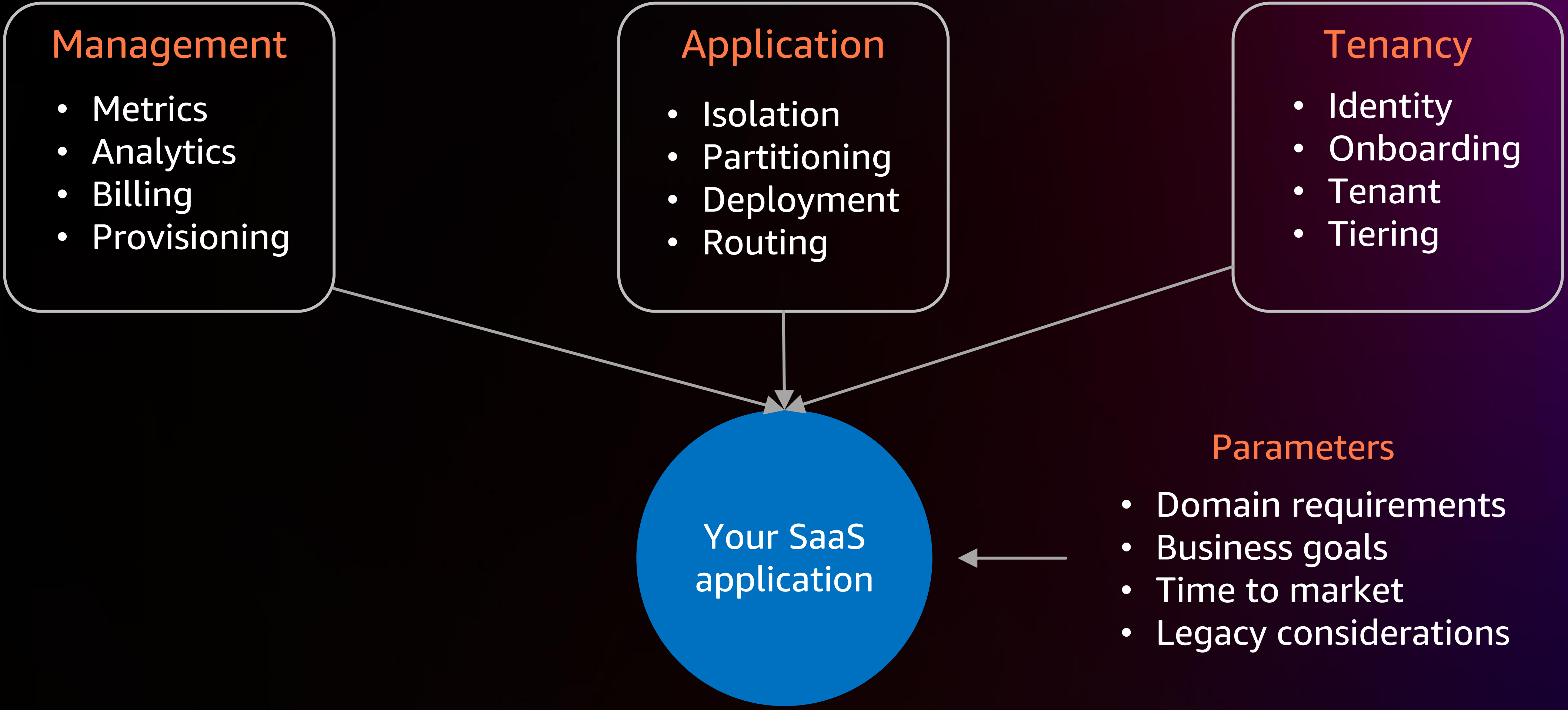
Tod Golding

Senior Principal Solutions Architect, AWS SaaS Factory
AWS

There’s no blueprint for SaaS



Composing SaaS from patterns



Management

- Metrics
- Analytics
- Billing
- Provisioning

Application

- Isolation
- Partitioning
- Deployment
- Routing

Tenancy

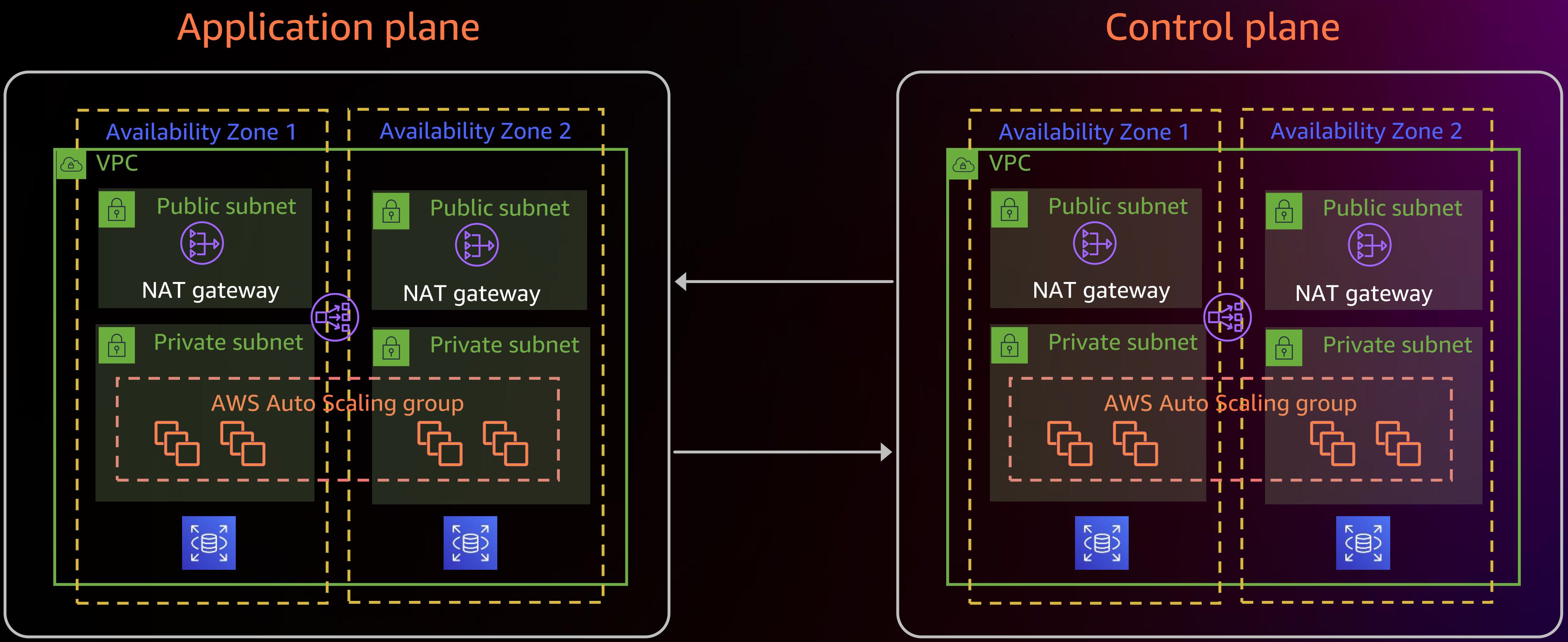
- Identity
- Onboarding
- Tenant
- Tiering

Your SaaS application

Parameters

- Domain requirements
- Business goals
- Time to market
- Legacy considerations

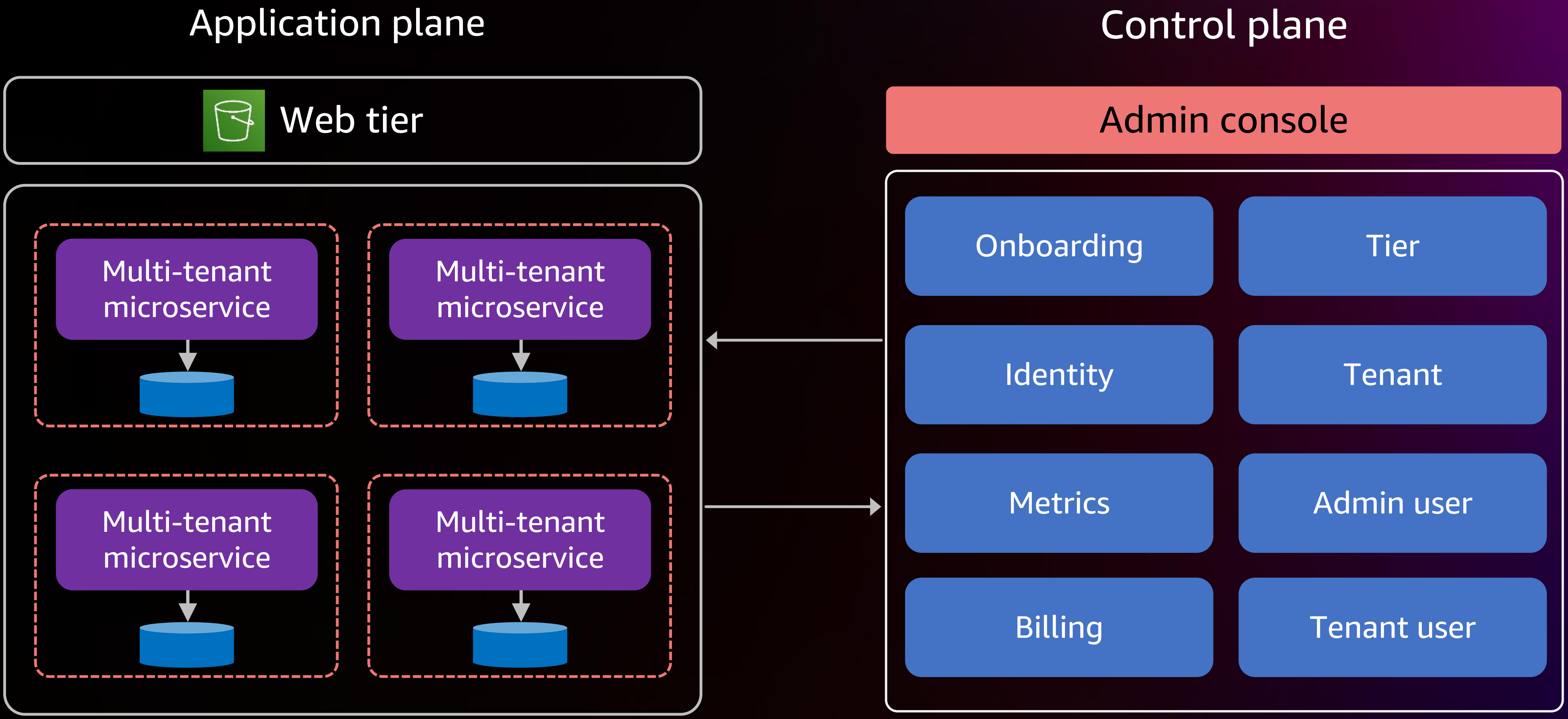
The two halves of SaaS



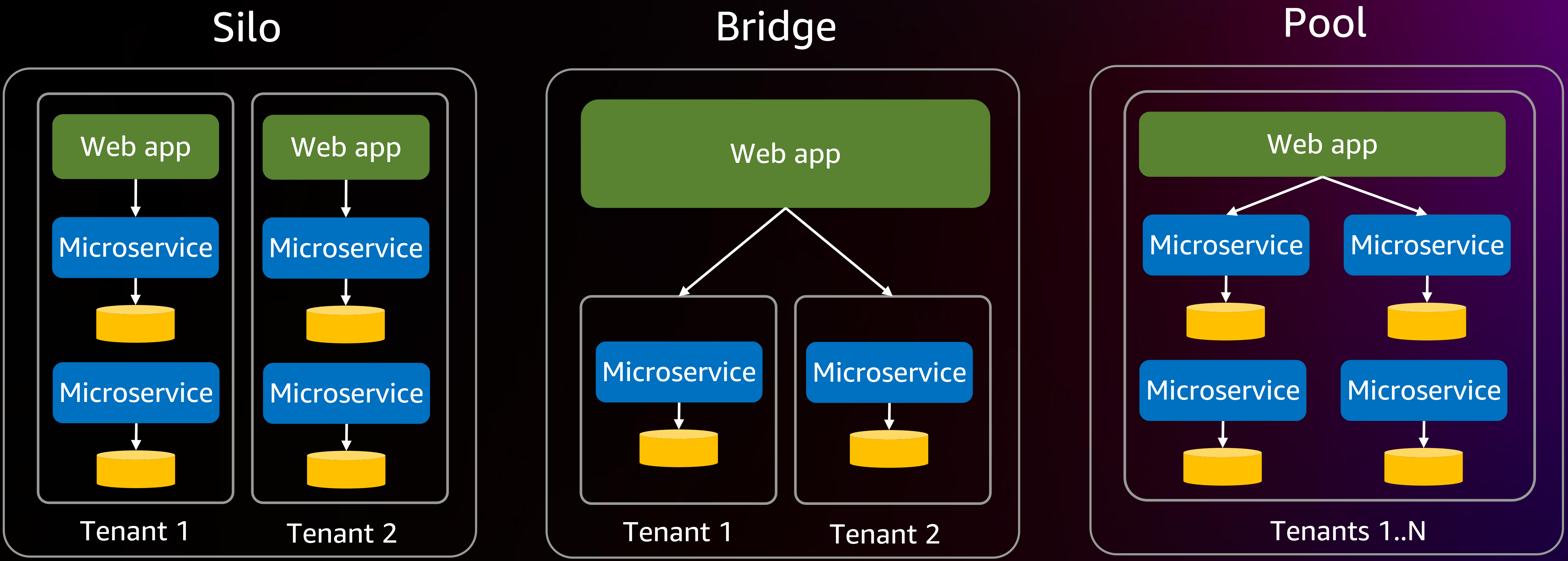
- Multi-tenant environment
- Home for all tenant functionality

- Operates and manages all tenants
- Home to all shared services

A multi-tenant application vs. a SaaS service

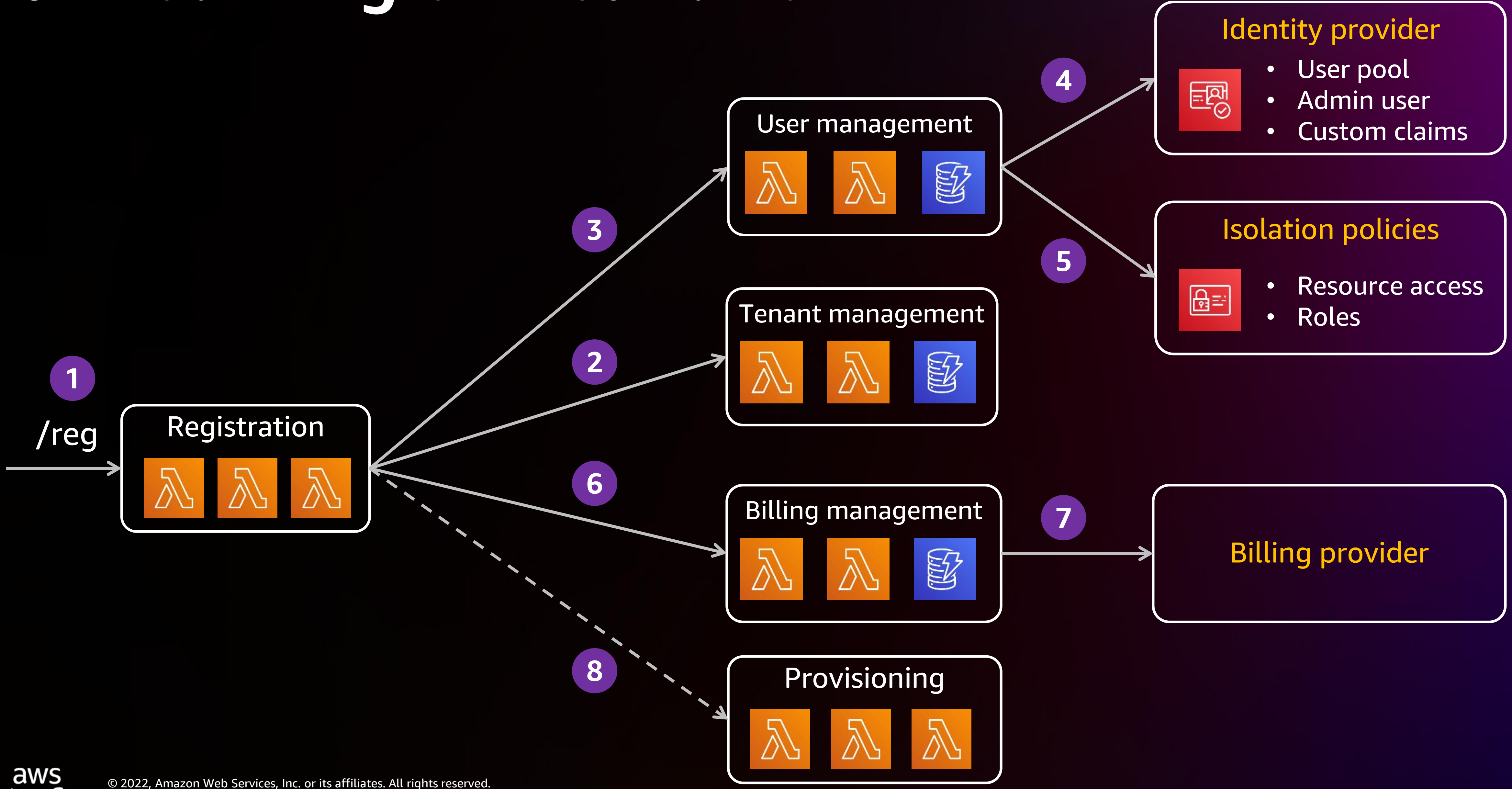


Describing SaaS environments



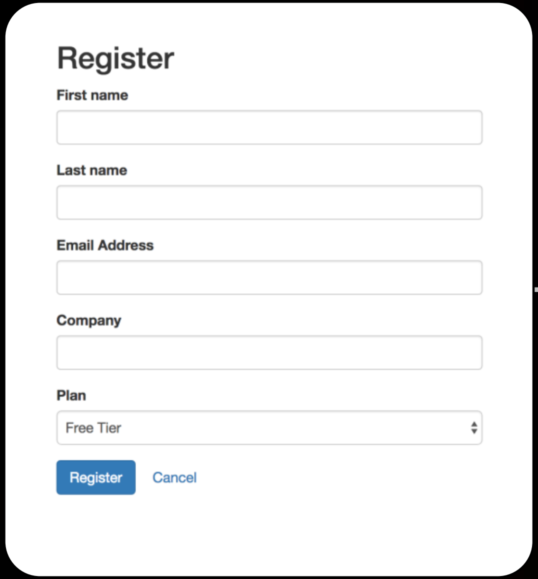
Control plane patterns

Onboarding orchestration



Tier-driven onboarding

1 Tenant registration



Register

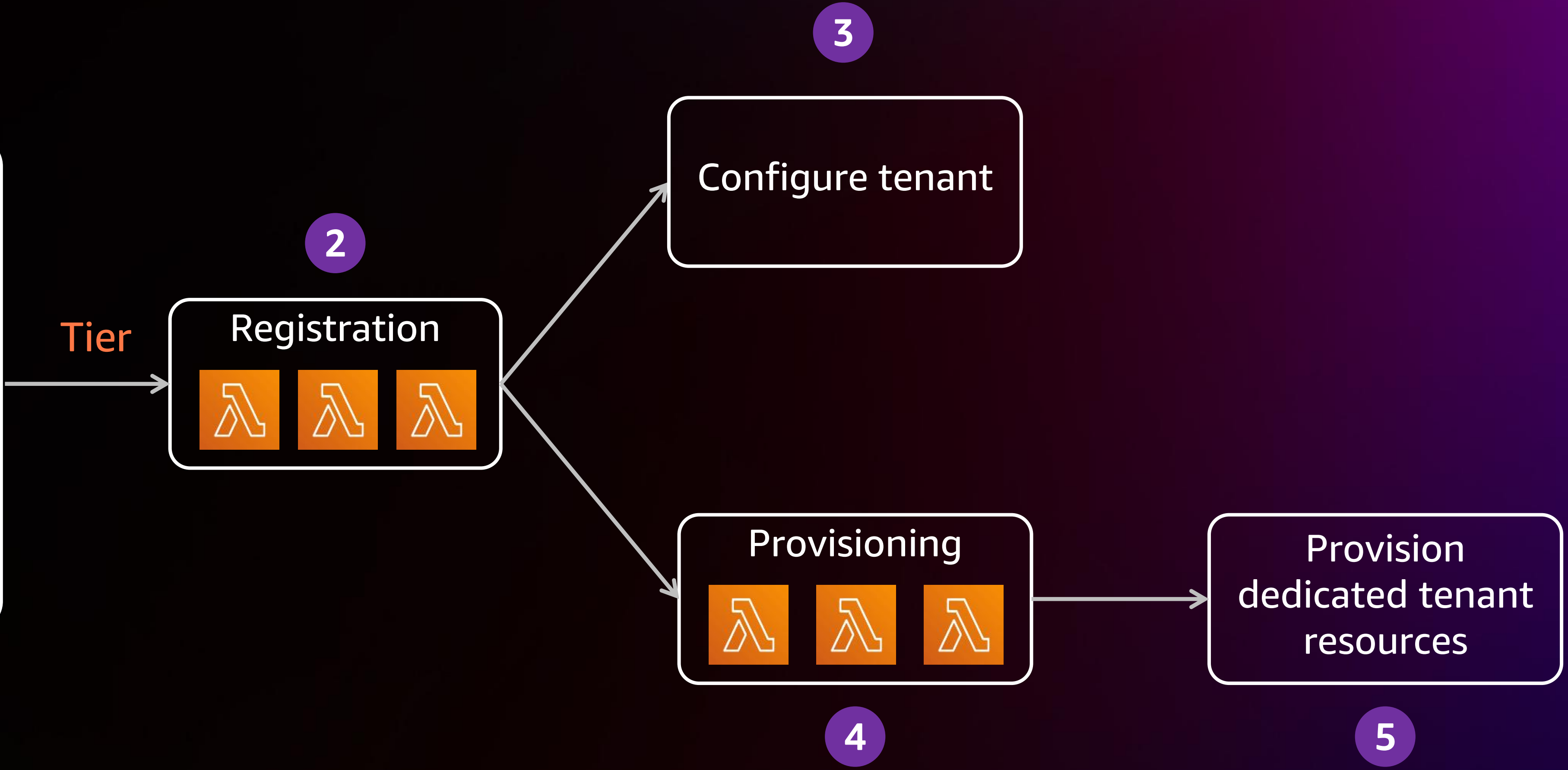
First name

Last name

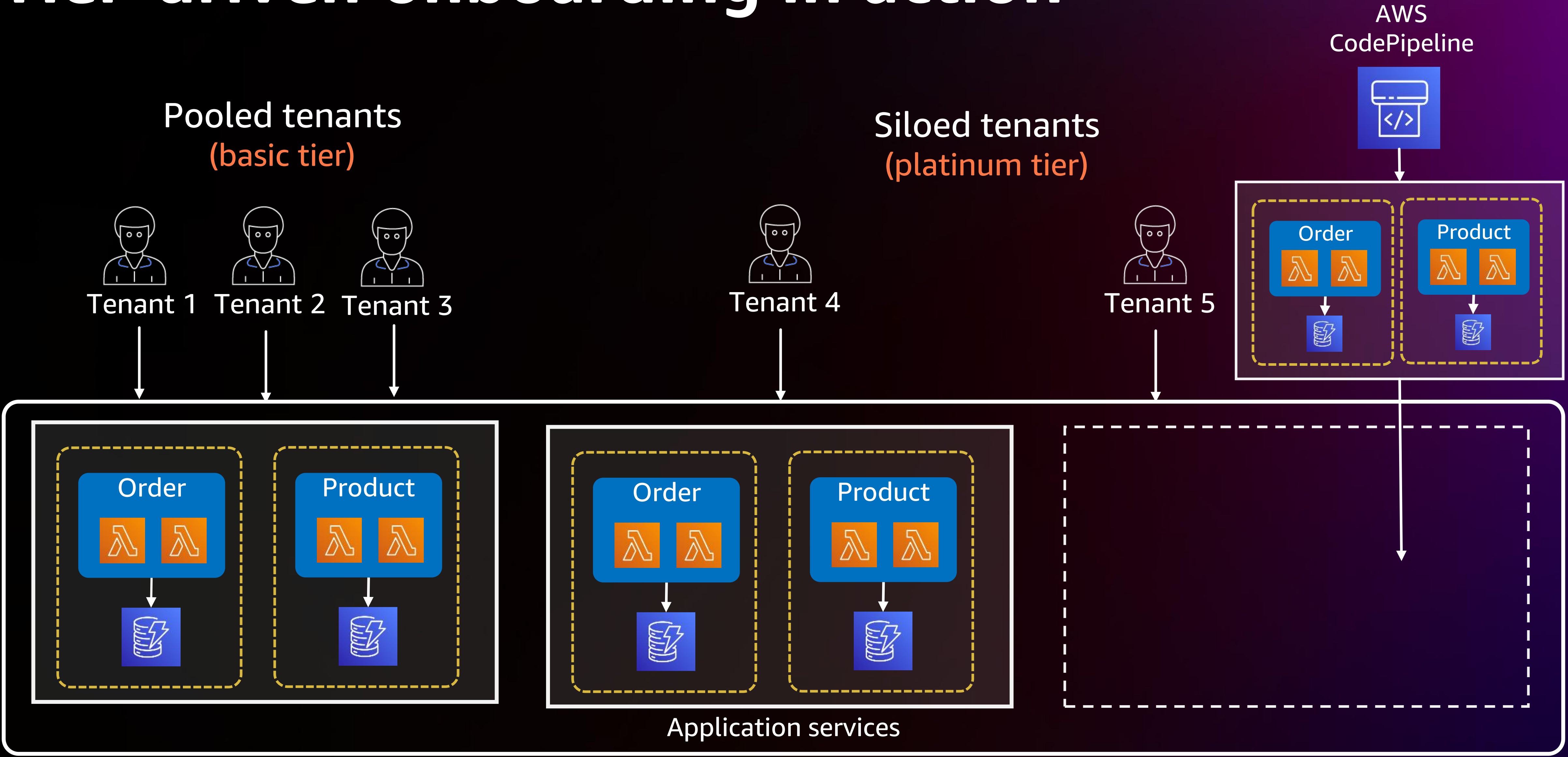
Email Address

Company

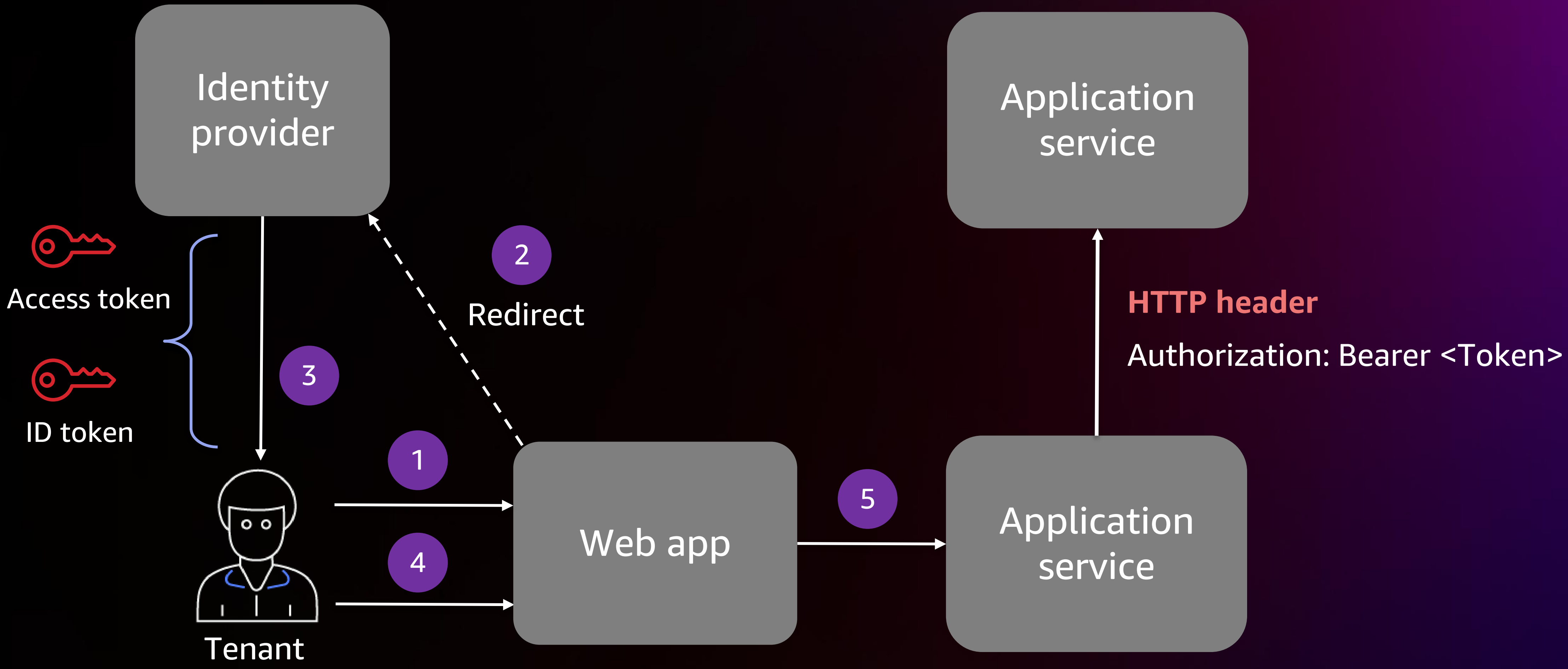
Plan: Free Tier



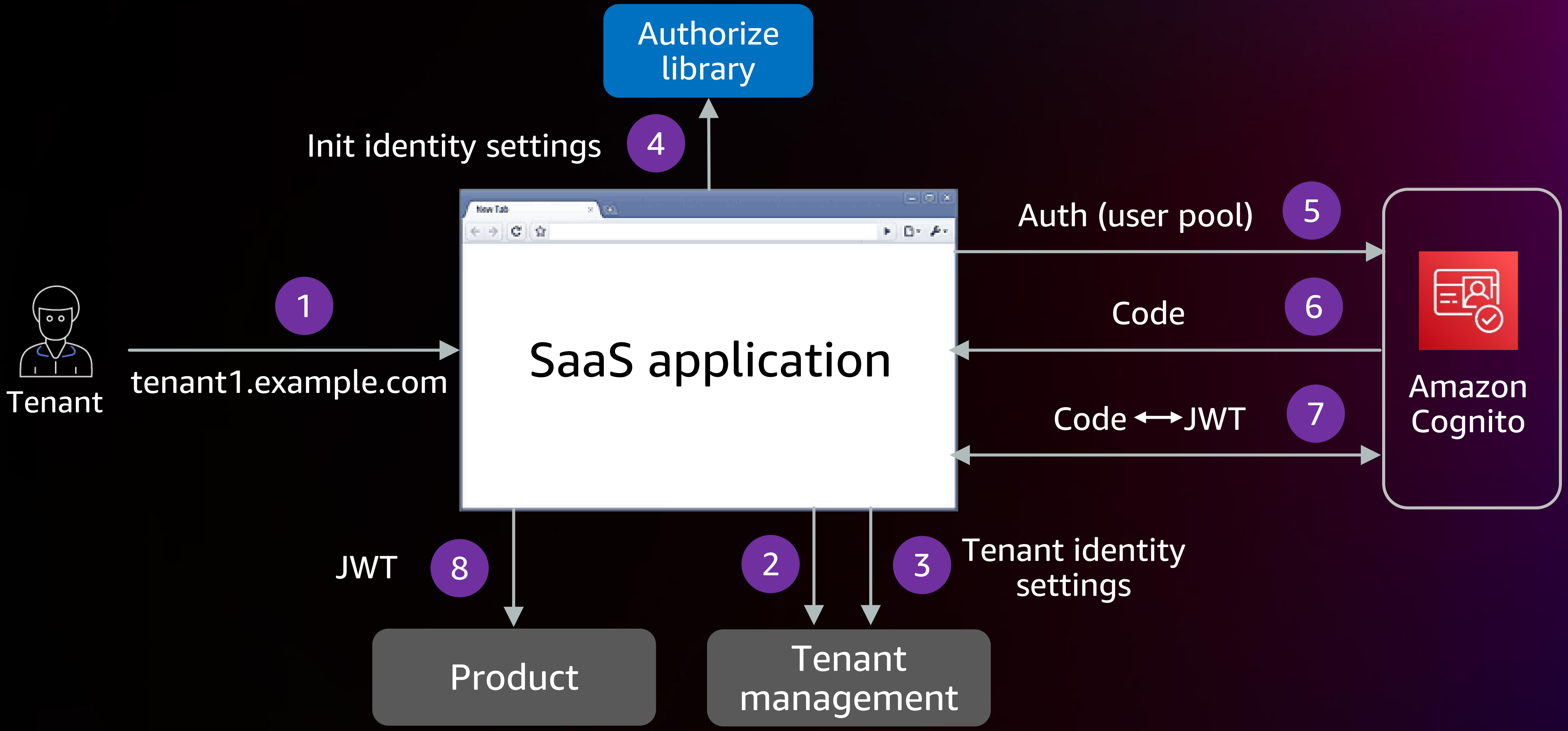
Tier-driven onboarding in action



Tenant-aware identity

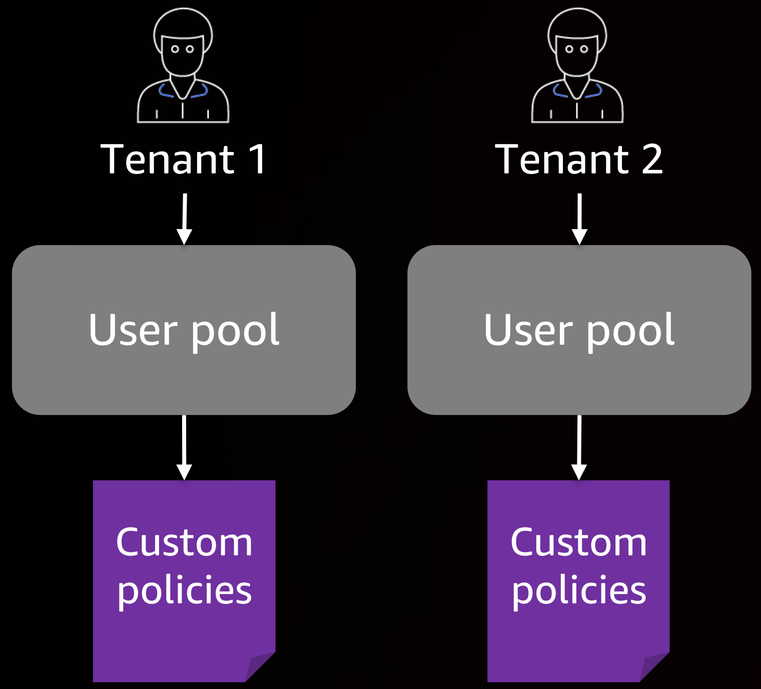


Tenant-aware identity flow



- Origin ⟶ company
- User pool mapping
- App ID

Tenant → User pool mapping

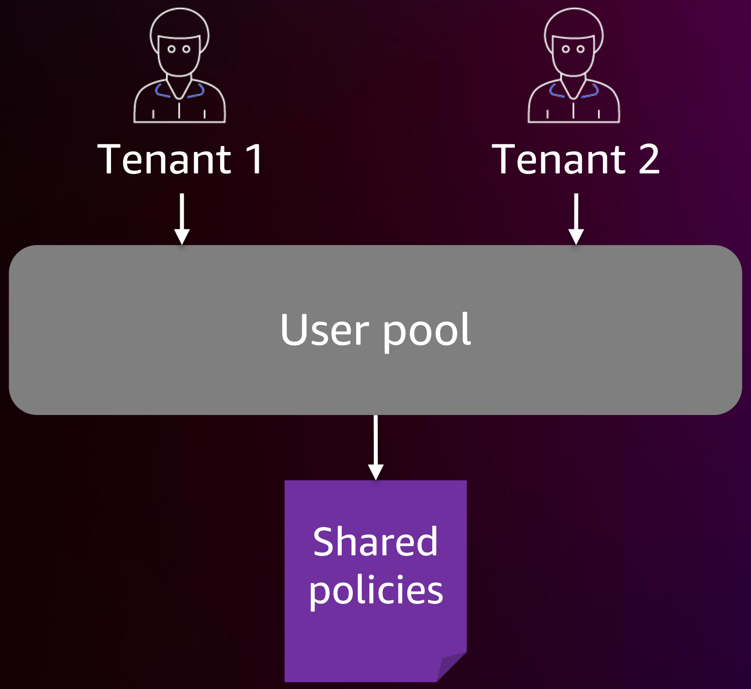


Pros

- Separate policies
- Better isolation

Cons

- Mapping required
- Scale
- Atypical OAuth flow



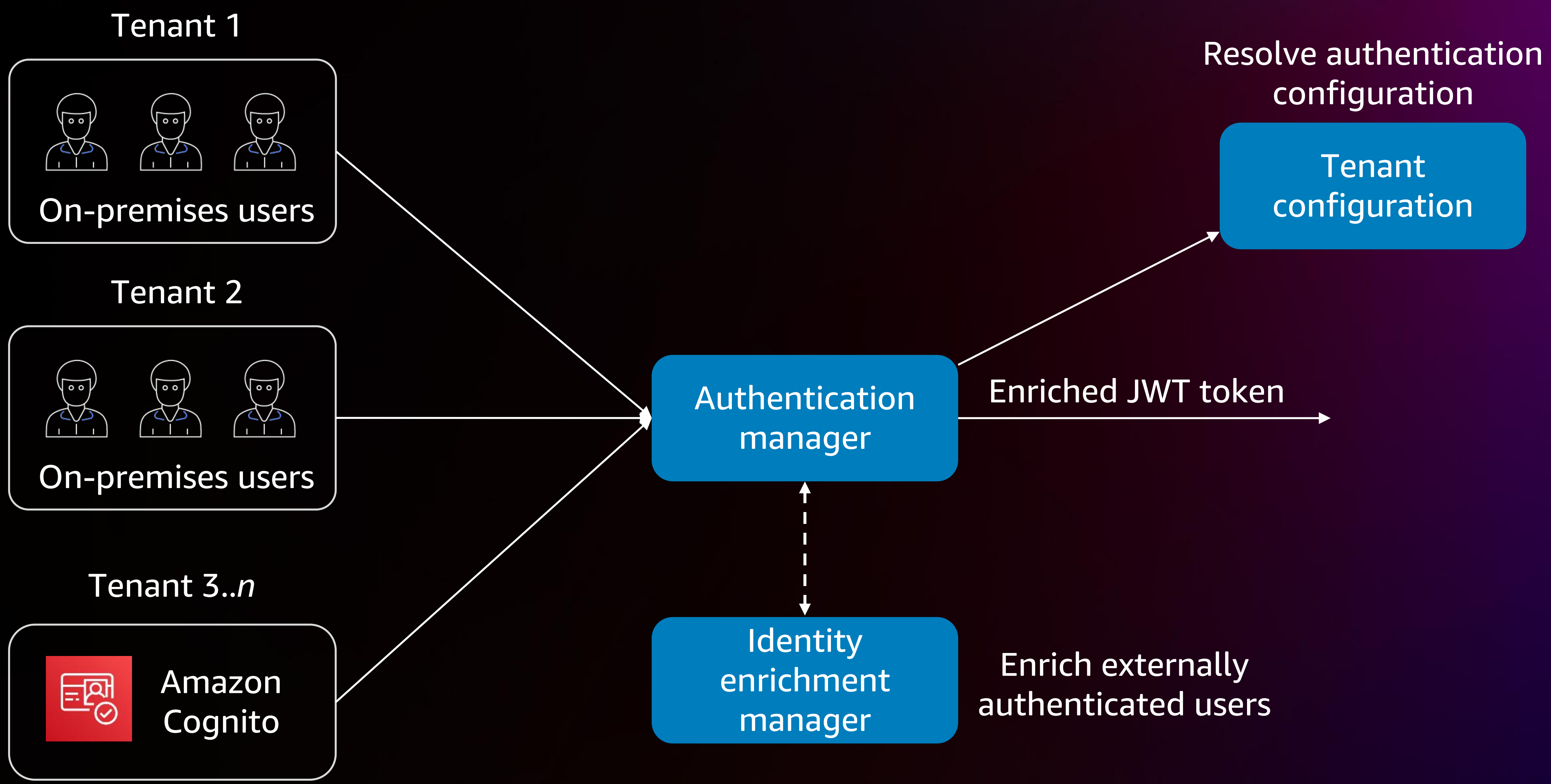
Pros

- No mapping
- Better OAuth flow
- Scale (maybe)

Cons

- No custom policies
- Isolation story

Hybrid identity model



Tenant 1

On-premises users

Tenant 2

On-premises users

Tenant 3..*n*

Amazon Cognito

Authentication manager

Identity enrichment manager

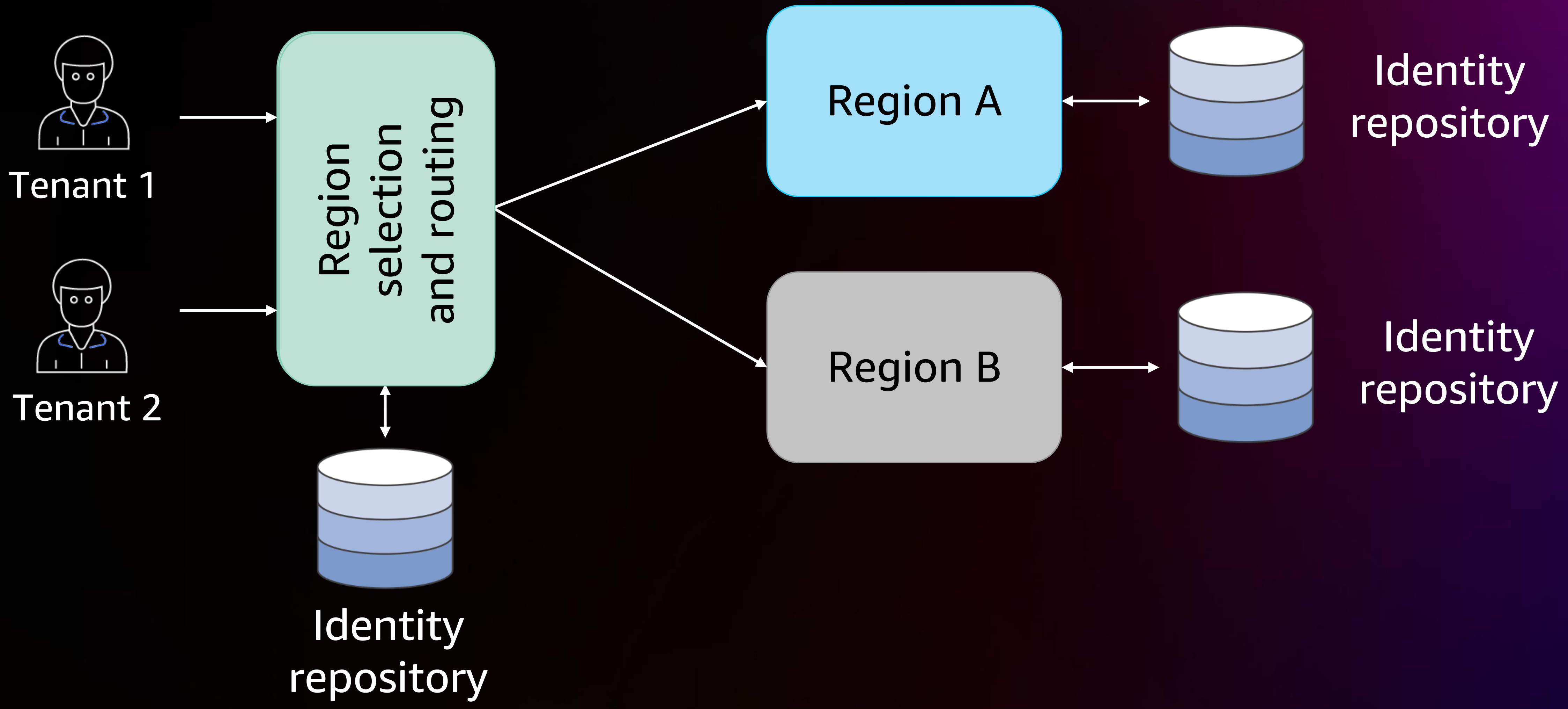
Resolve authentication configuration

Tenant configuration

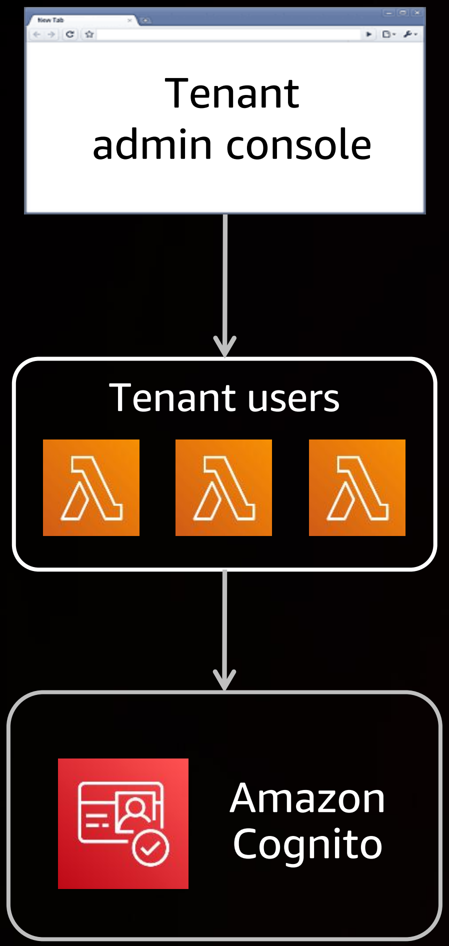
Enriched JWT token

Enrich externally authenticated users

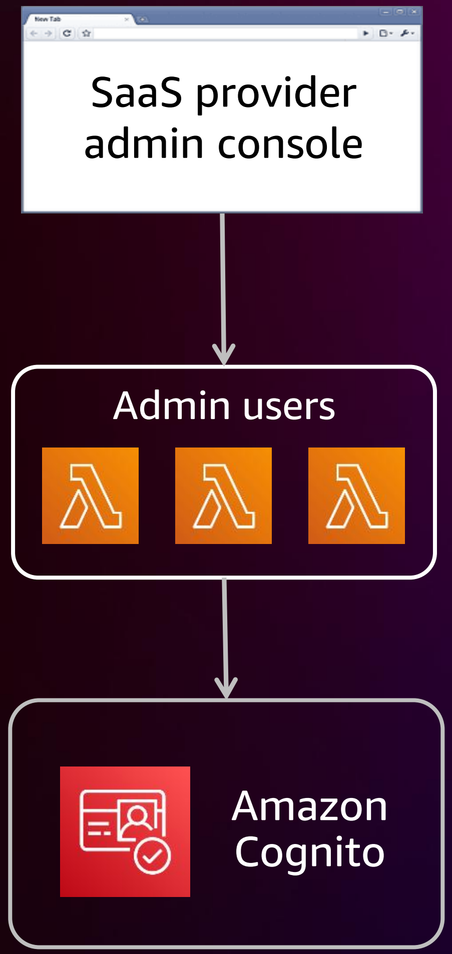
Multi-Region identity



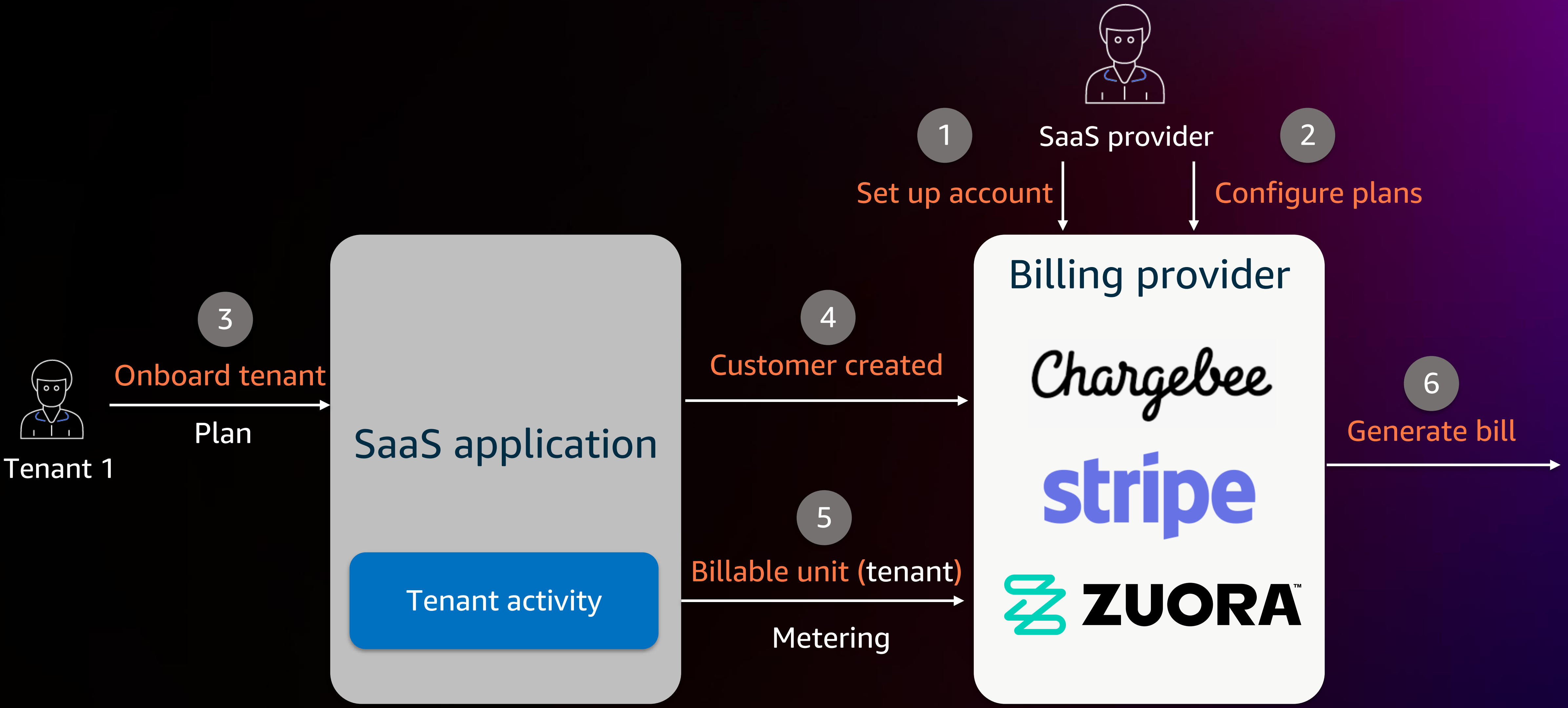
User management



- CRUD operations
- Enable/disable

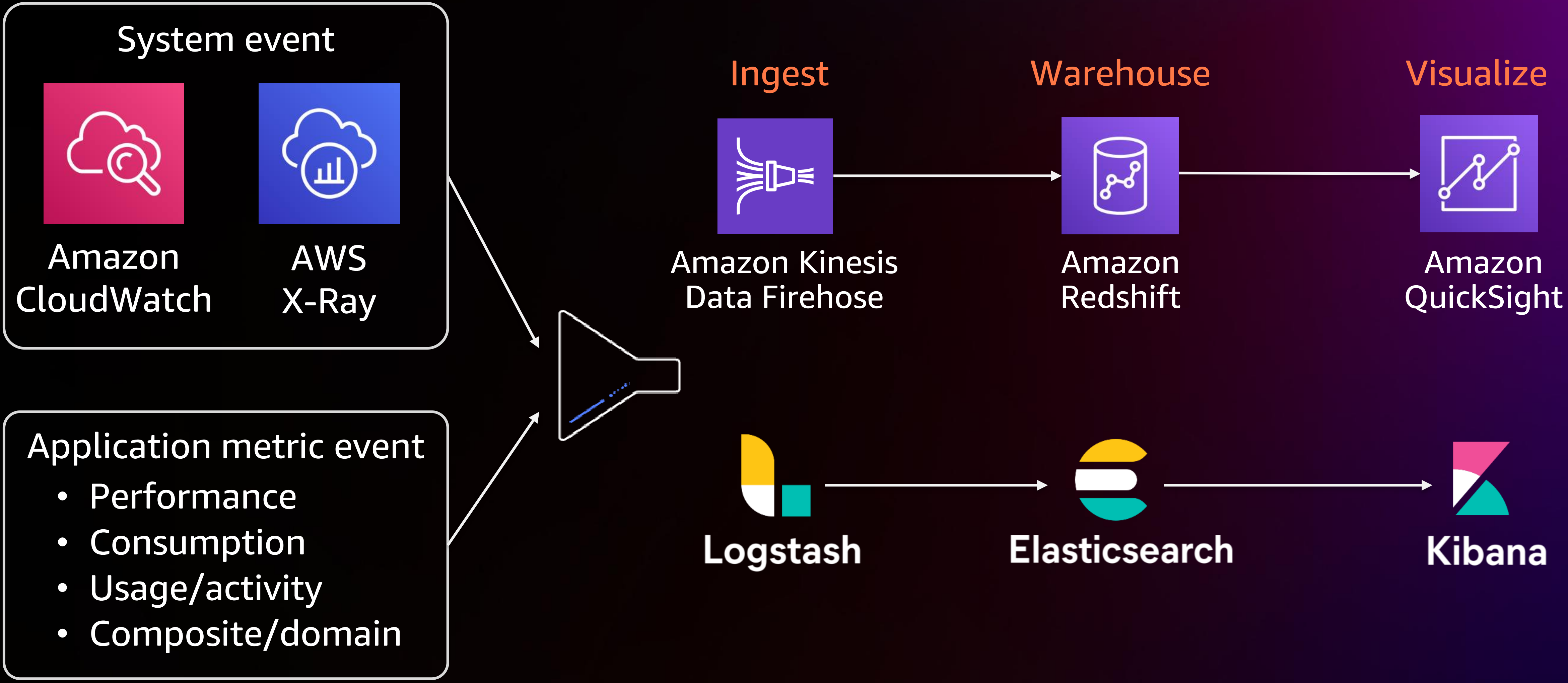


Billing configuration and instrumentation



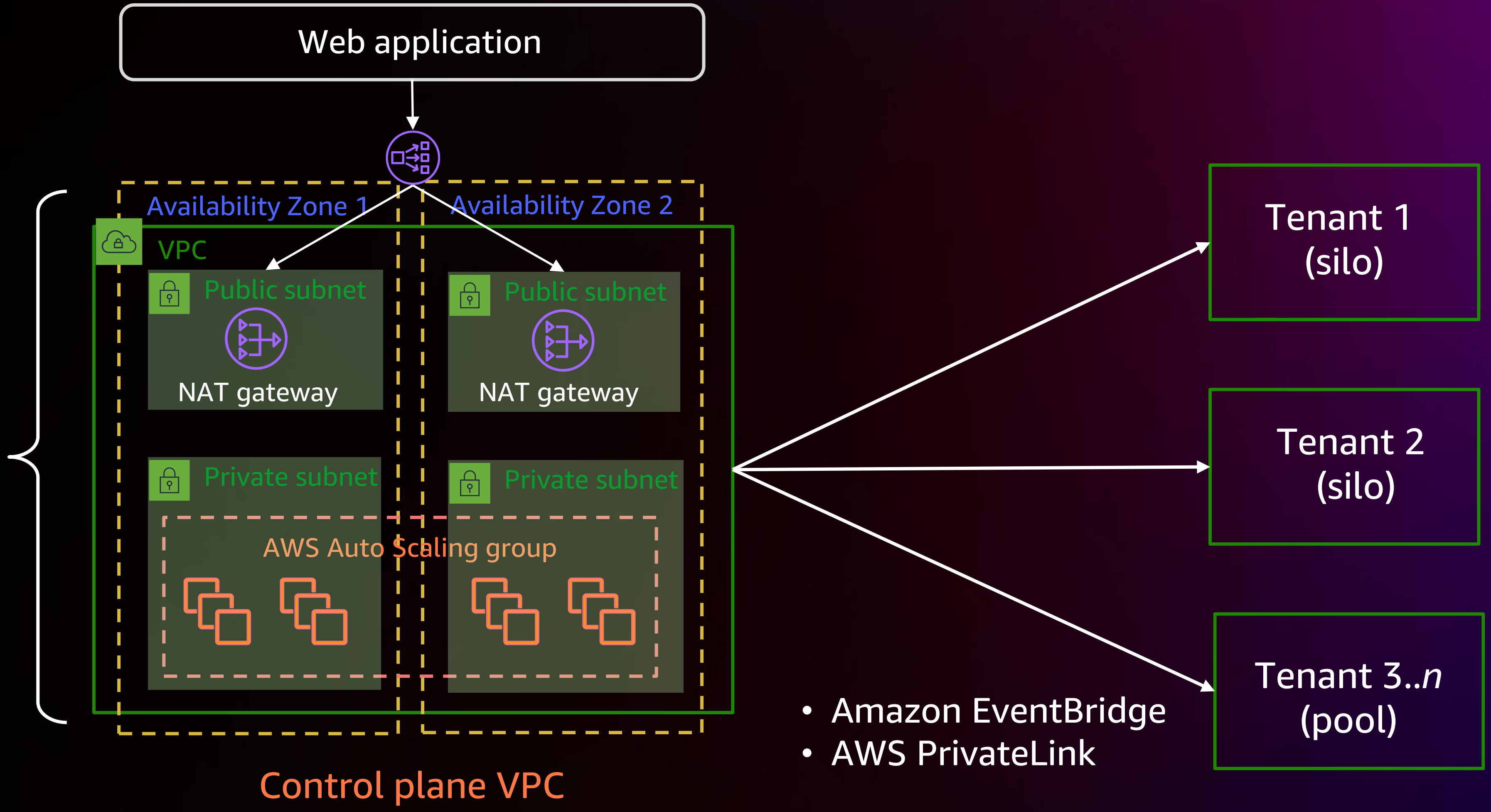
This is not a complete list. To view all AWS Partners for this category, visit AWS Partner Solutions Finder. This list of partners is current as of October 14, 2022.

Metrics instrumentation and aggregation



Application plane integration

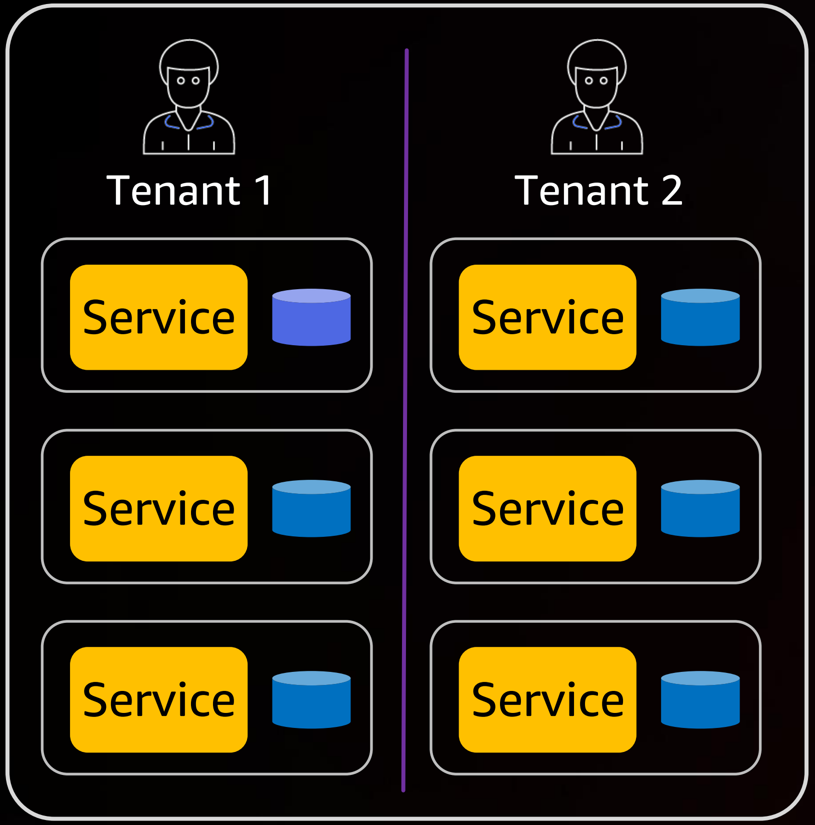
- Onboarding
- Identity
- Provisioning
- Management
- Metrics and analytics
- Tenant-aware logging
- Billing



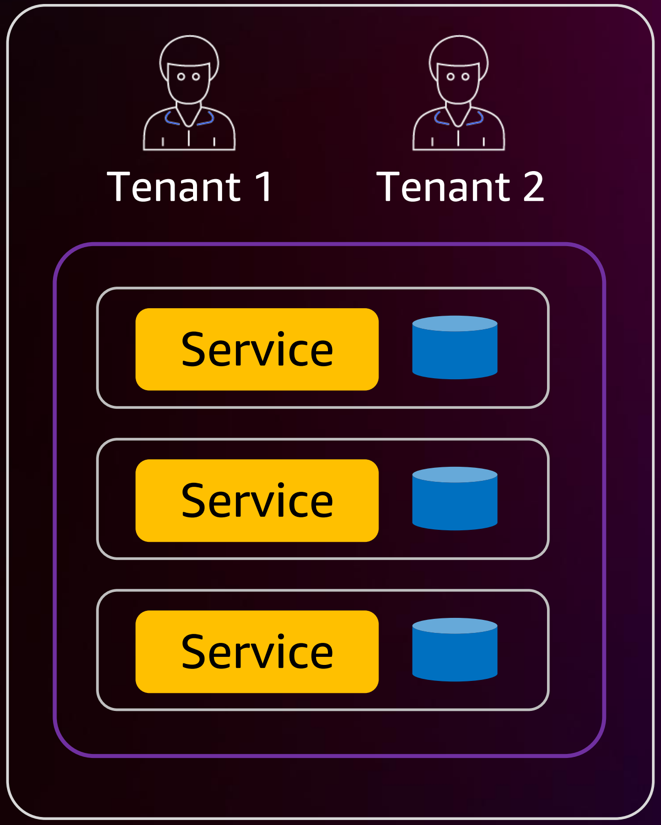
- Amazon EventBridge
- AWS PrivateLink

Application plane patterns

Starting with deployment models

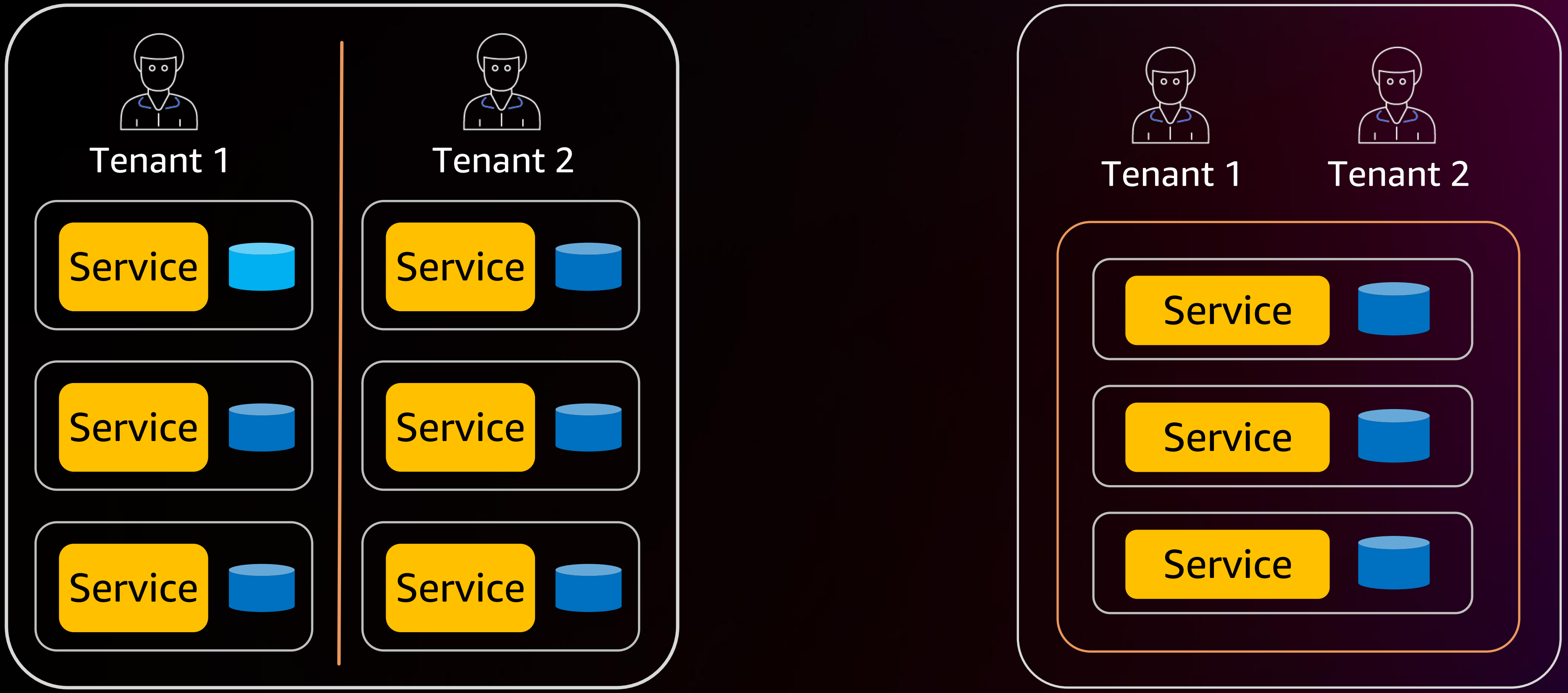


Full stack silo model



Full stack pool model

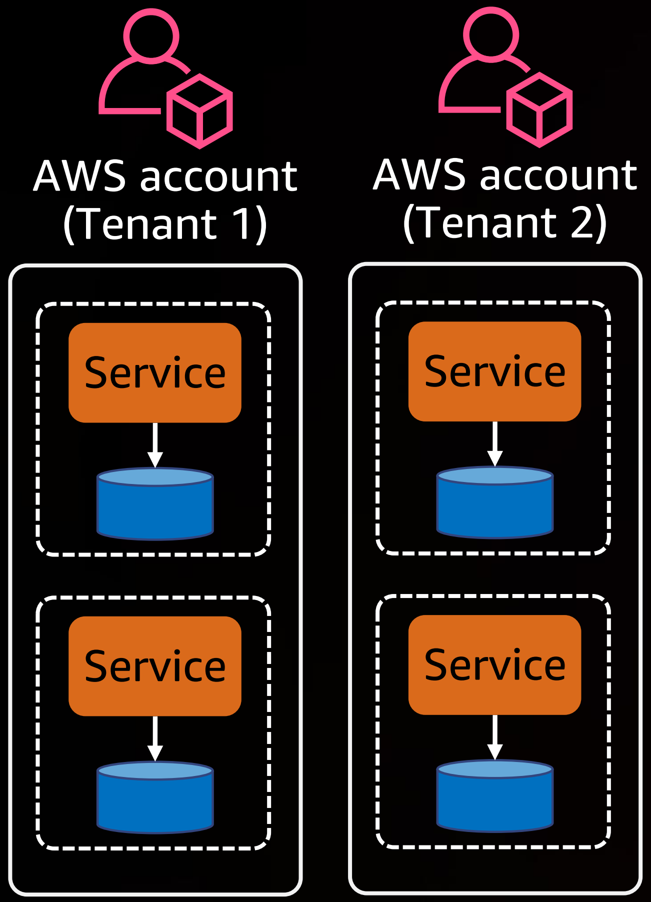
Tier-driven deployment models



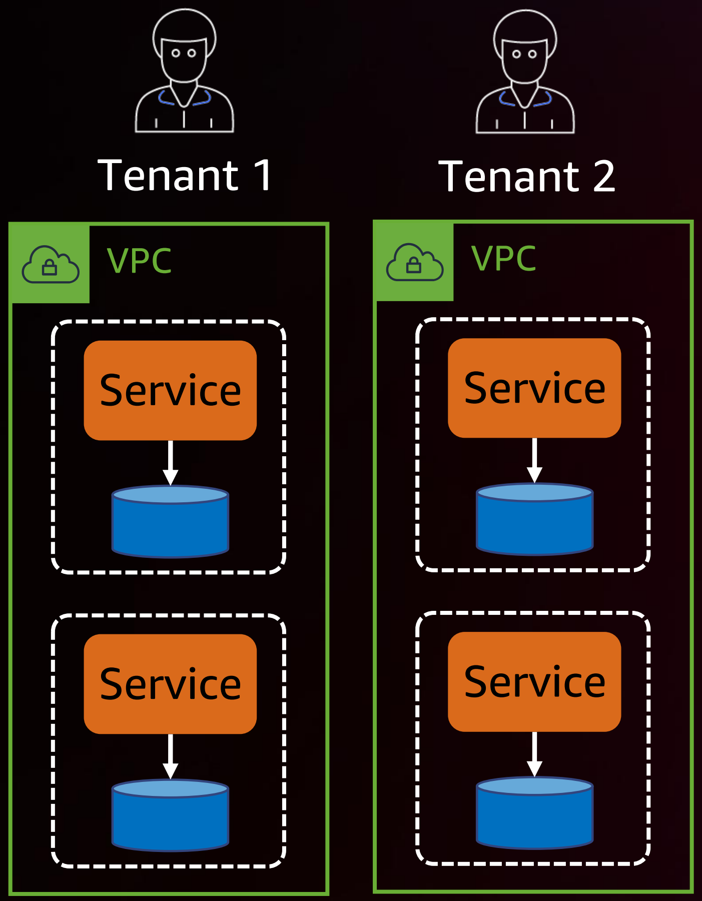
Advanced tier tenants

Basic tier tenants

Full stack silo patterns



Account per tenant

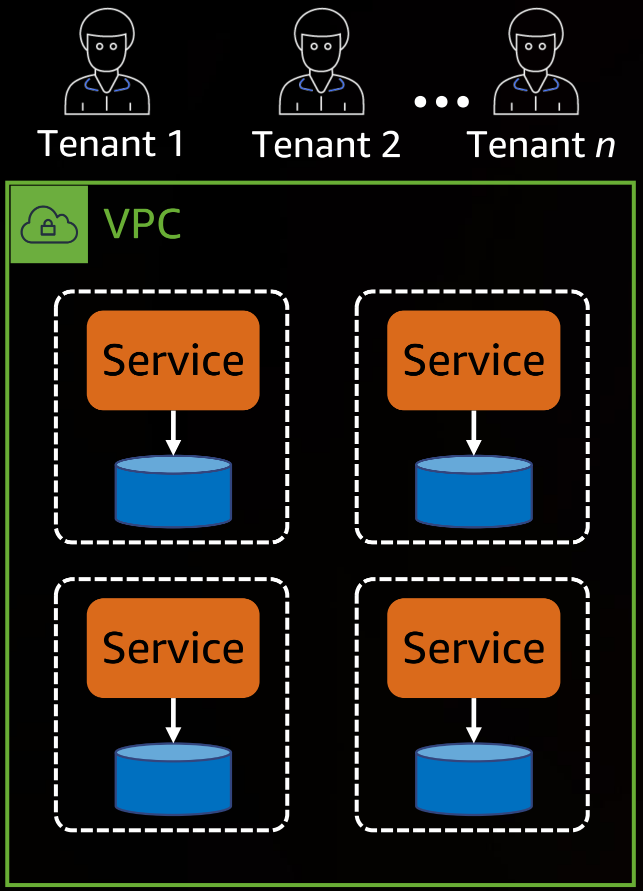


VPC per tenant

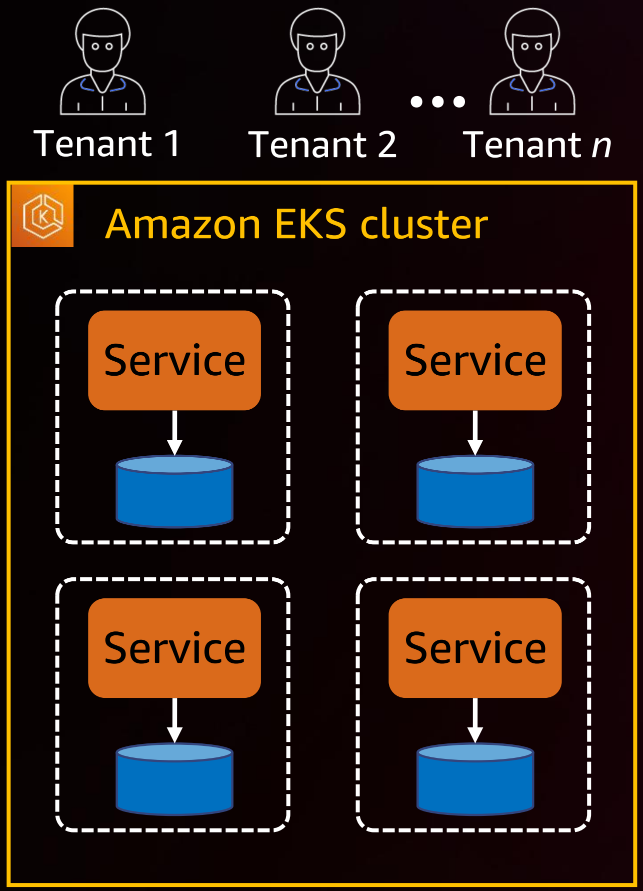
Onboarding	Tier
Identity	Tenant
Metrics	Admin user
Billing	Tenant user

Control plane

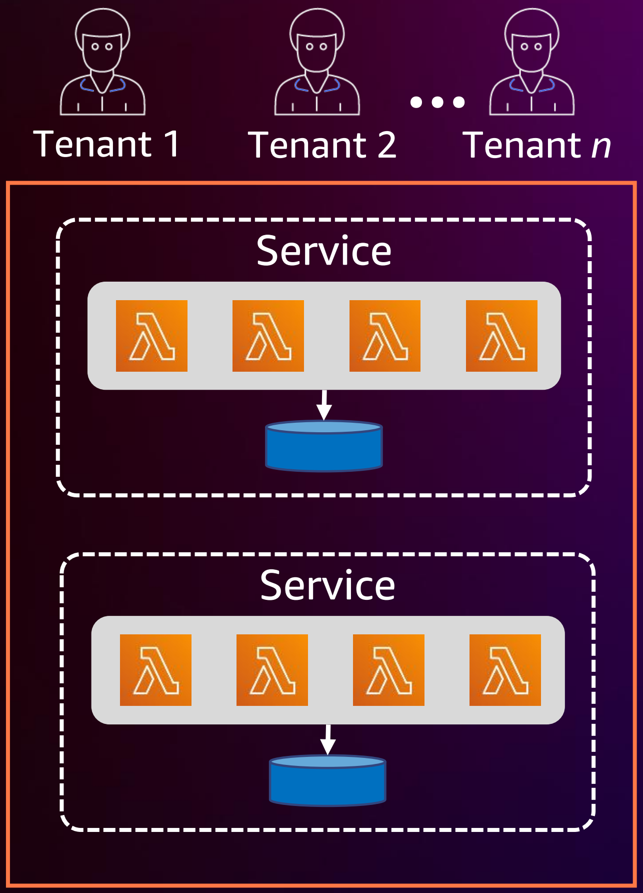
Full stack pool patterns



Pooled VPC

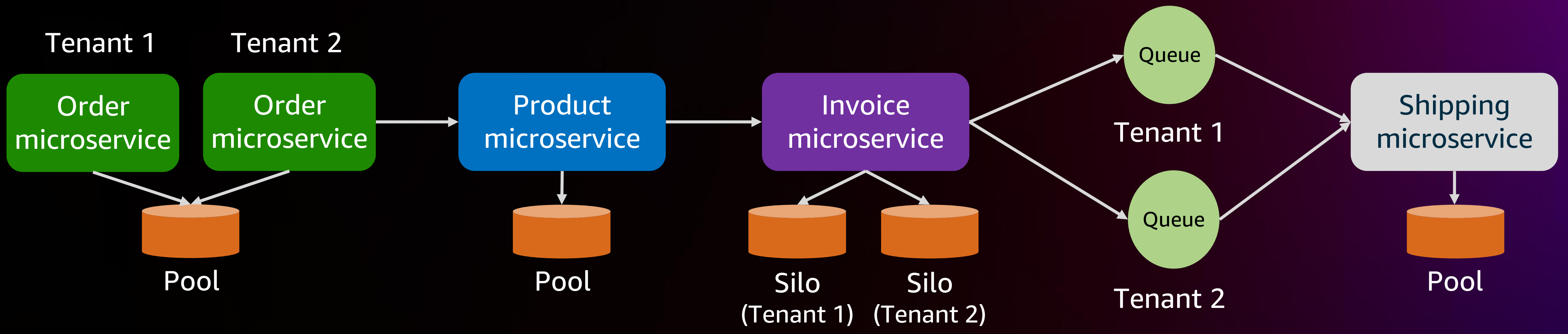


Pooled cluster



Pooled functions

Mixed mode deployment model



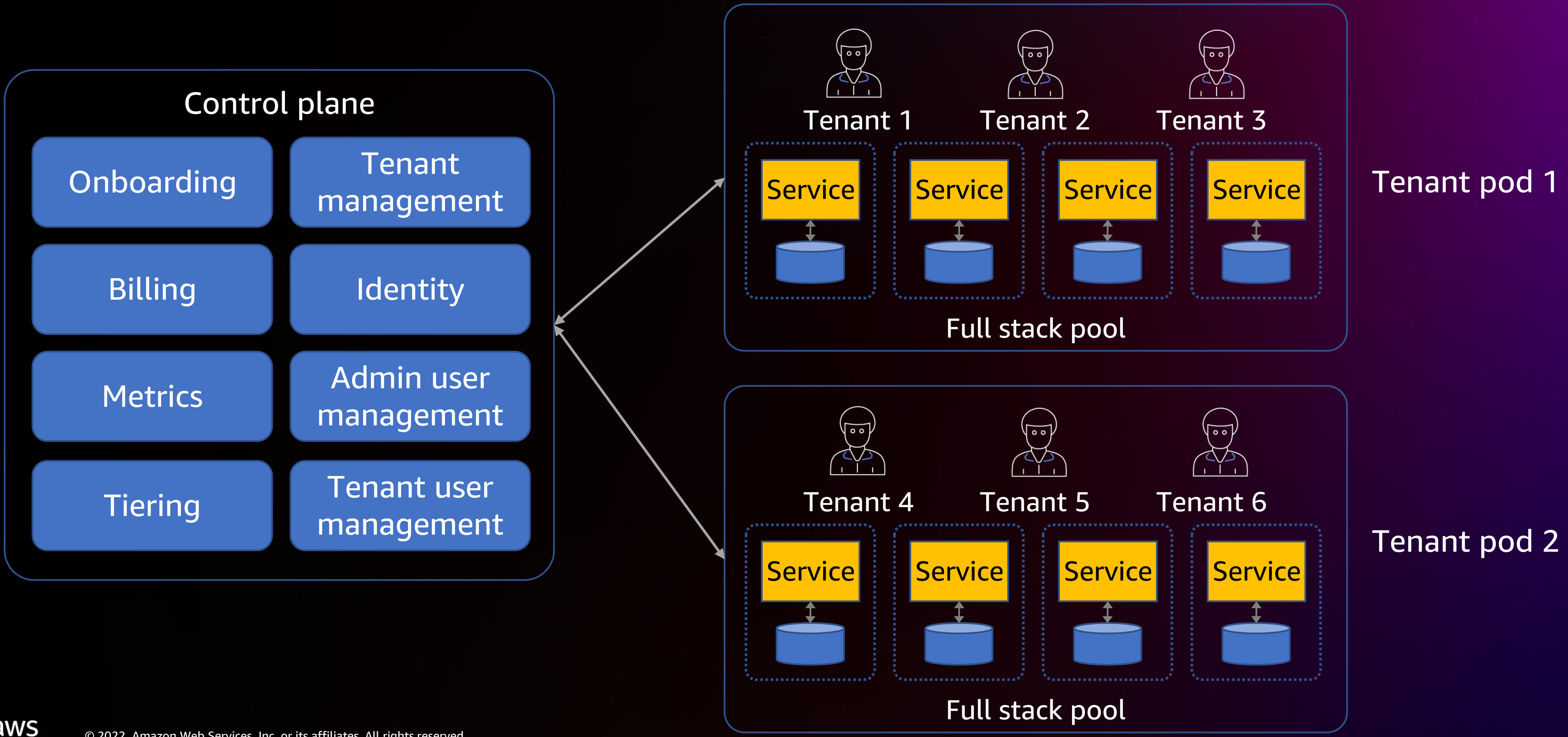
Siloed compute and pooled storage

Pooled compute and pooled storage

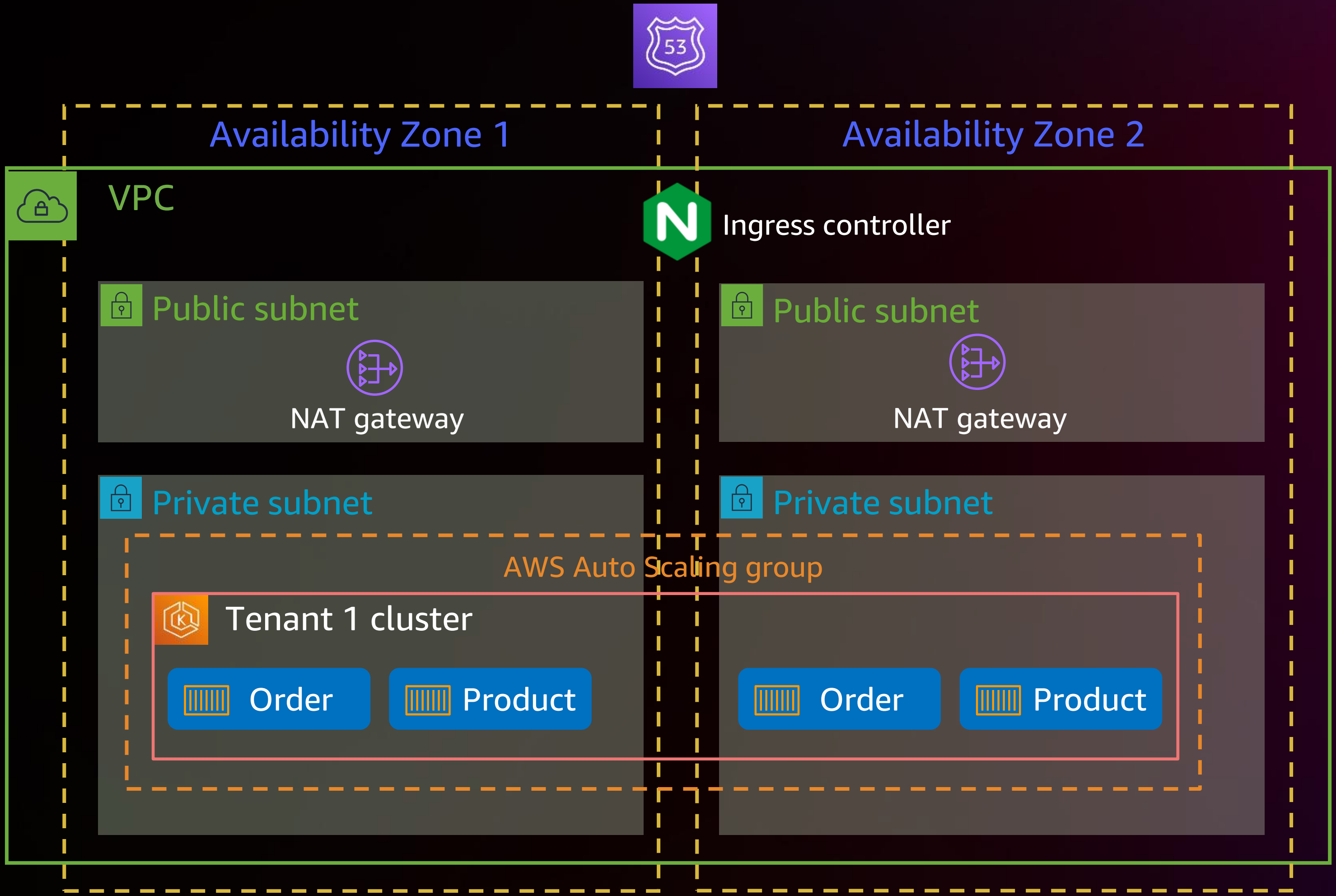
Pooled compute and siloed storage

Pooled compute and storage consuming siloed queues

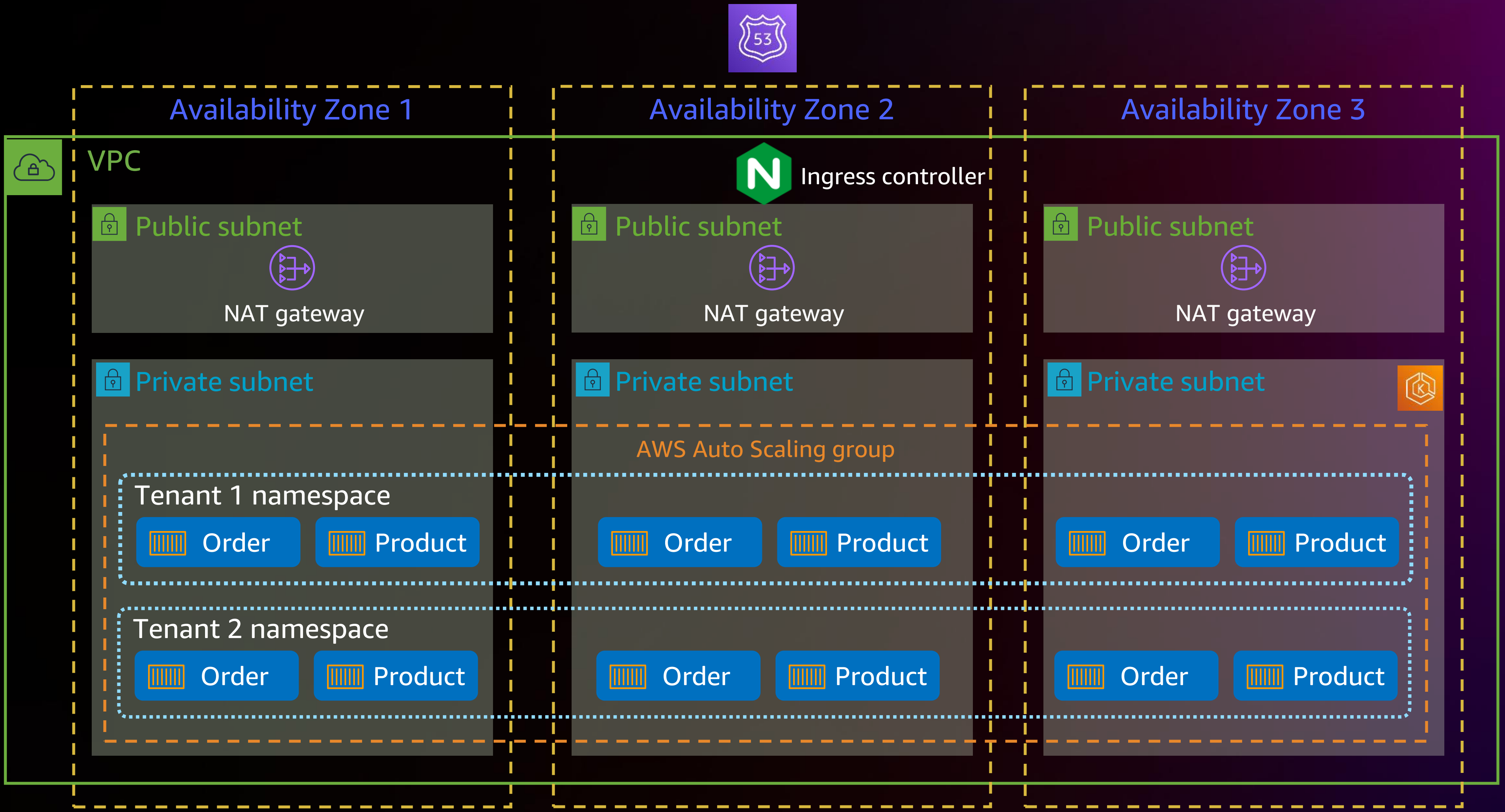
Pod-based deployment model



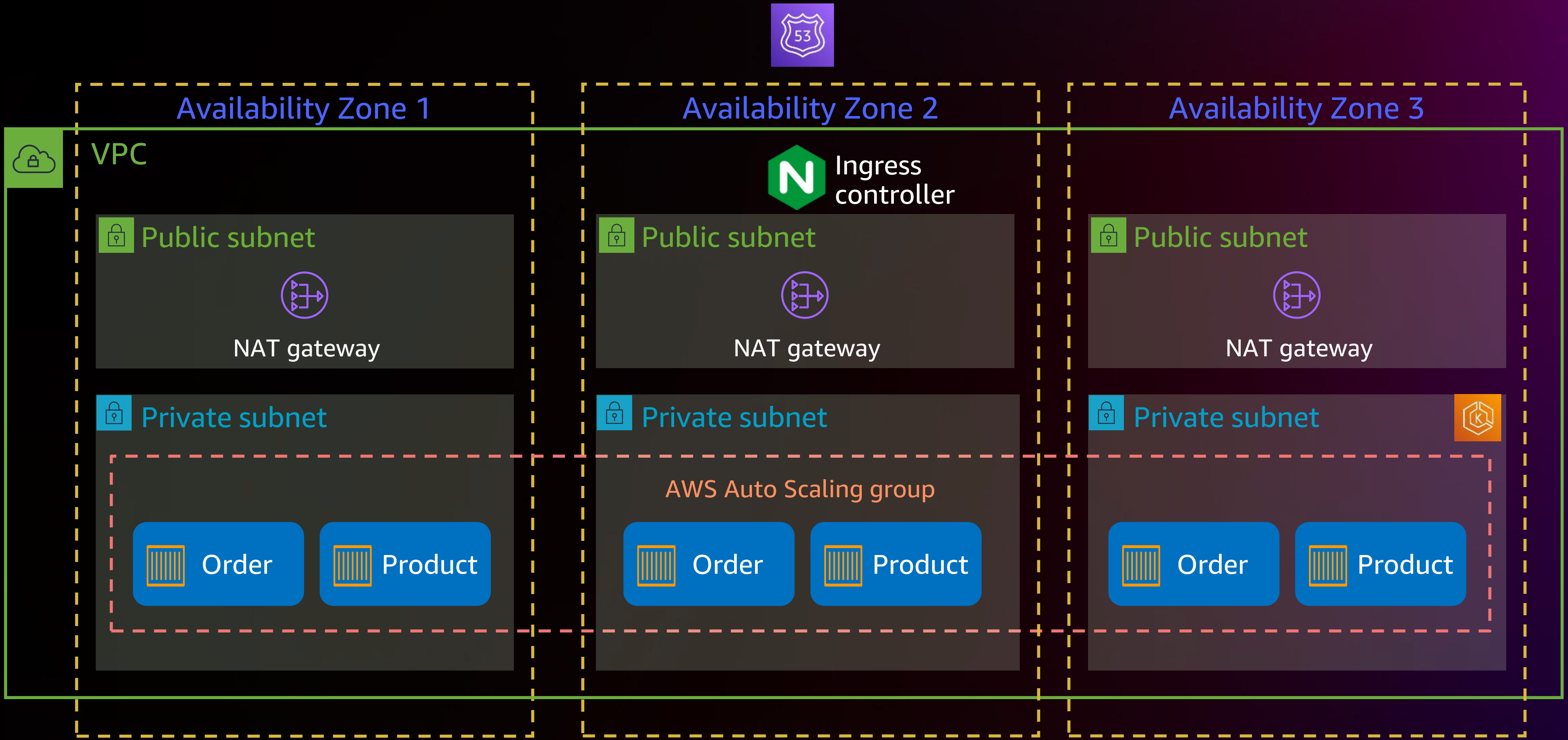
Amazon EKS compute silo: Cluster per tenant



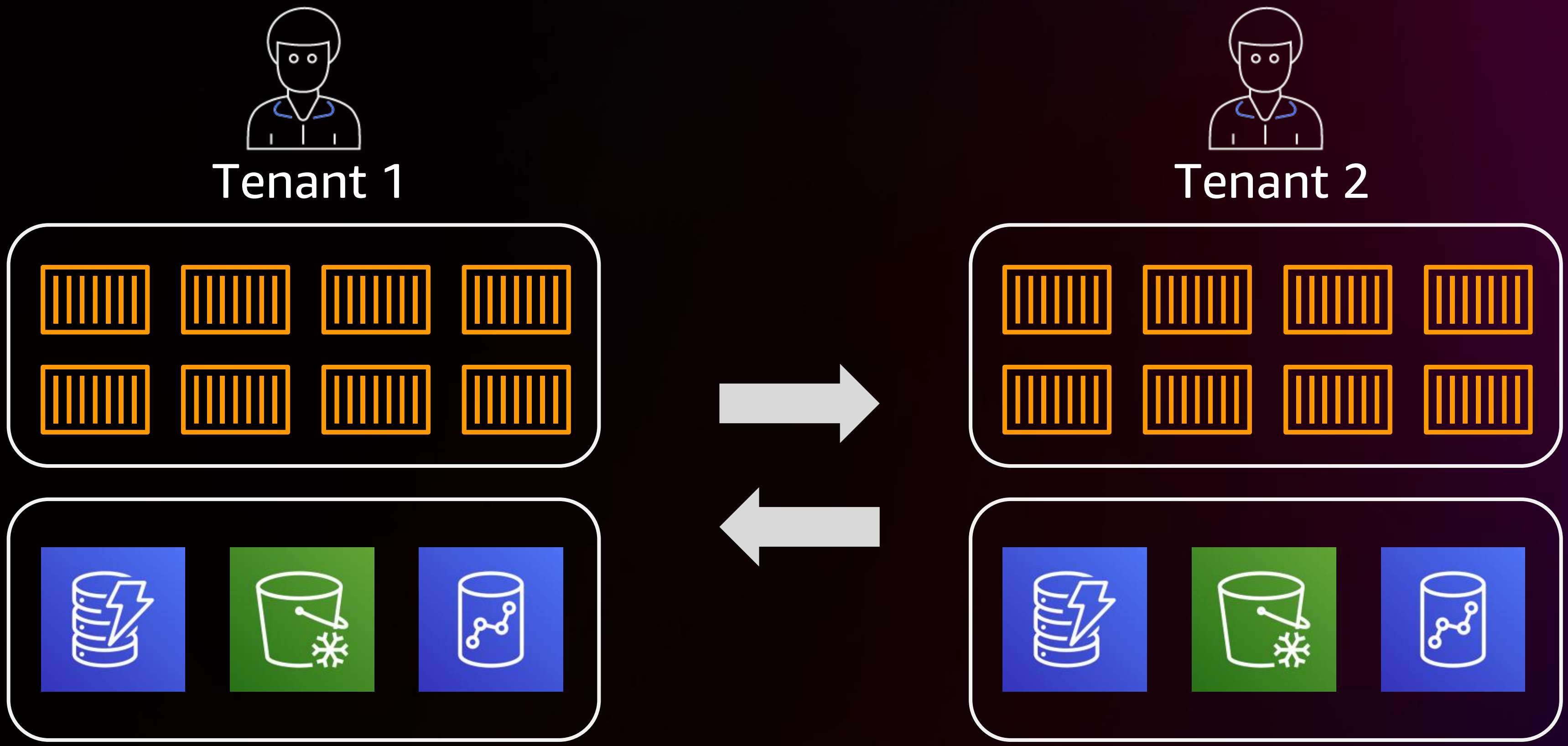
EKS compute silo: Namespace per tenant



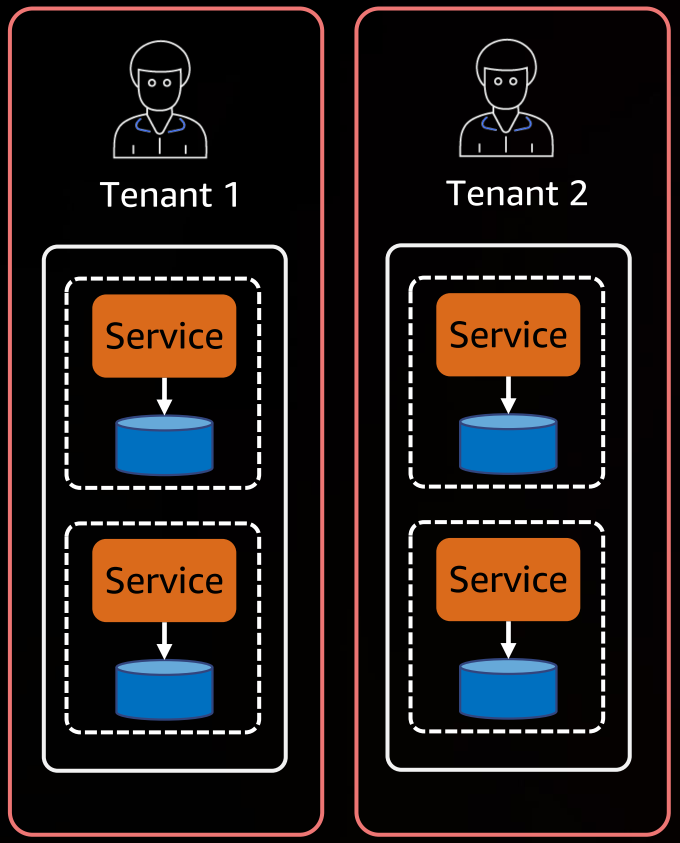
Amazon EKS compute pool



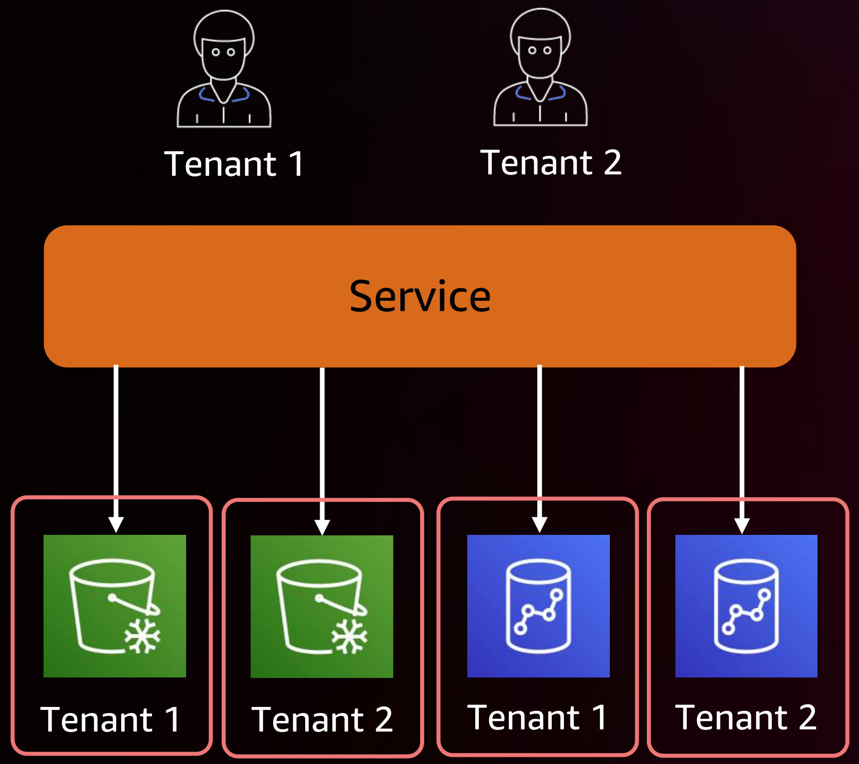
Tenant isolation



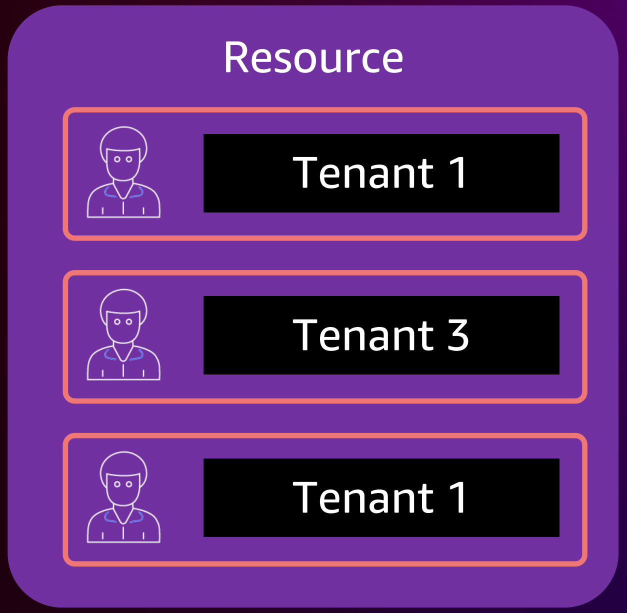
Tenant isolation patterns



Full stack isolation

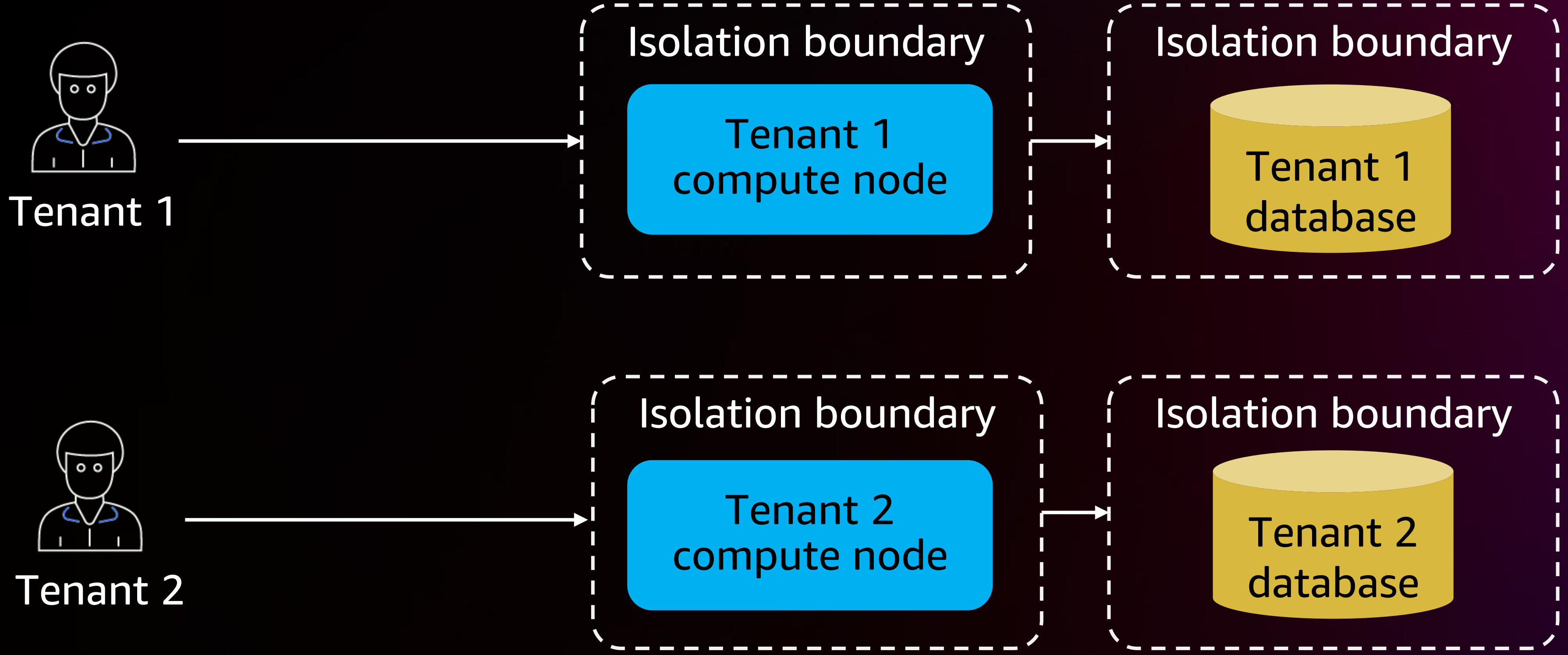


Resource-level isolation



Item-level isolation

Resource-level isolation (silo)



Tenant 1

Isolation boundary

Tenant 1 compute node

Isolation boundary

Tenant 1 database

Tenant 2

Isolation boundary

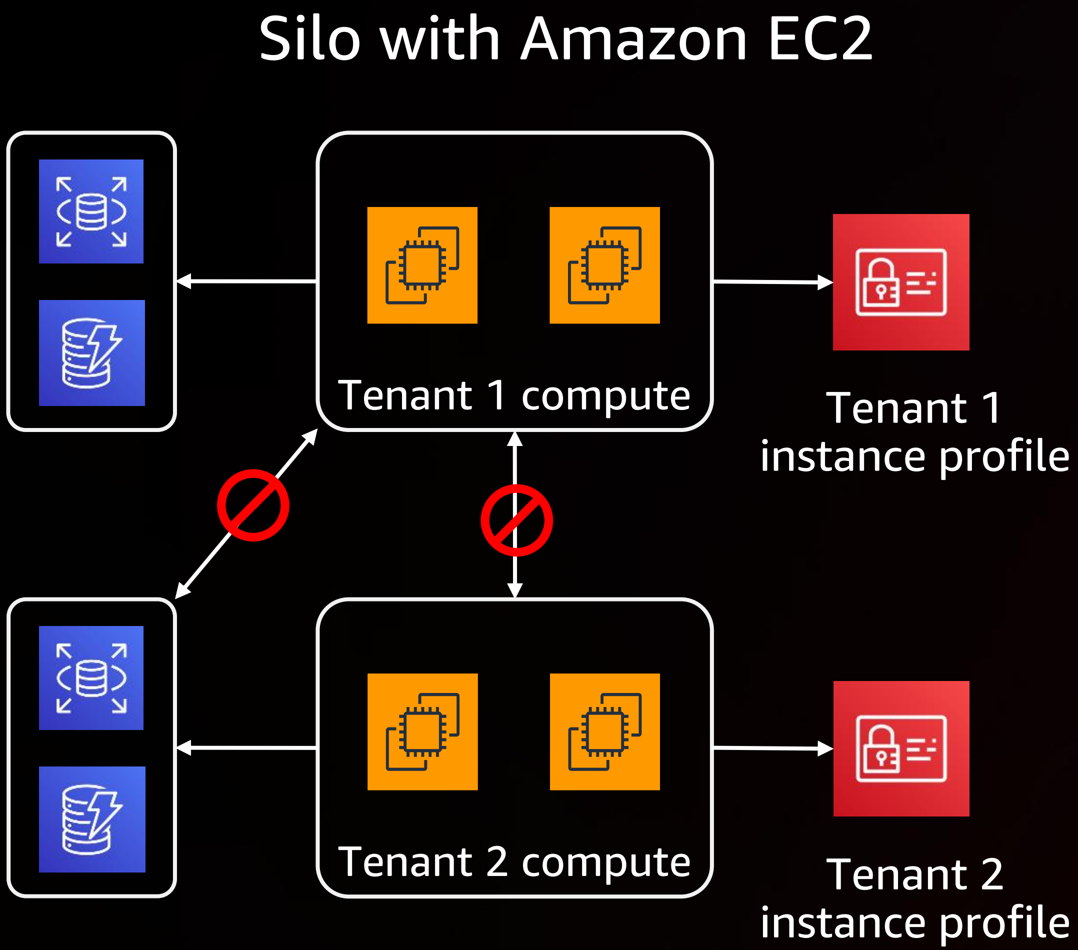
Tenant 2 compute node

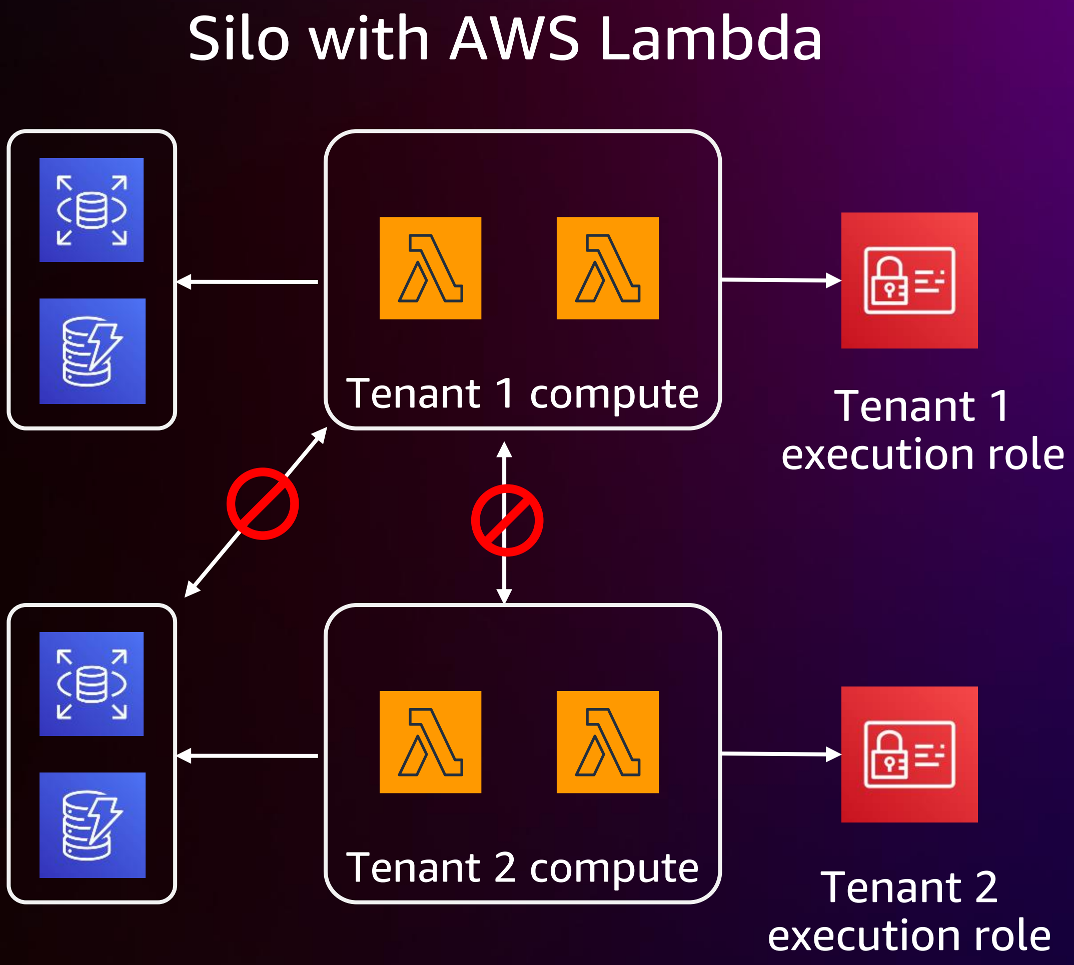
Isolation boundary

Tenant 2 database

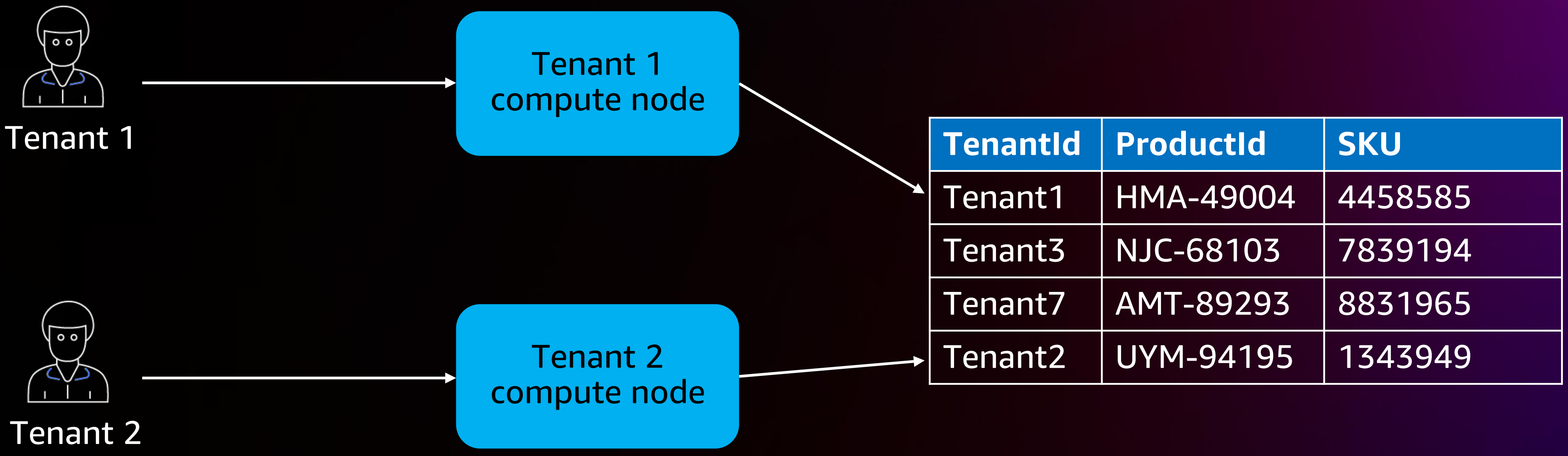
Unit of isolation is an entire resource level

Deployment-driven resource isolation (silo)





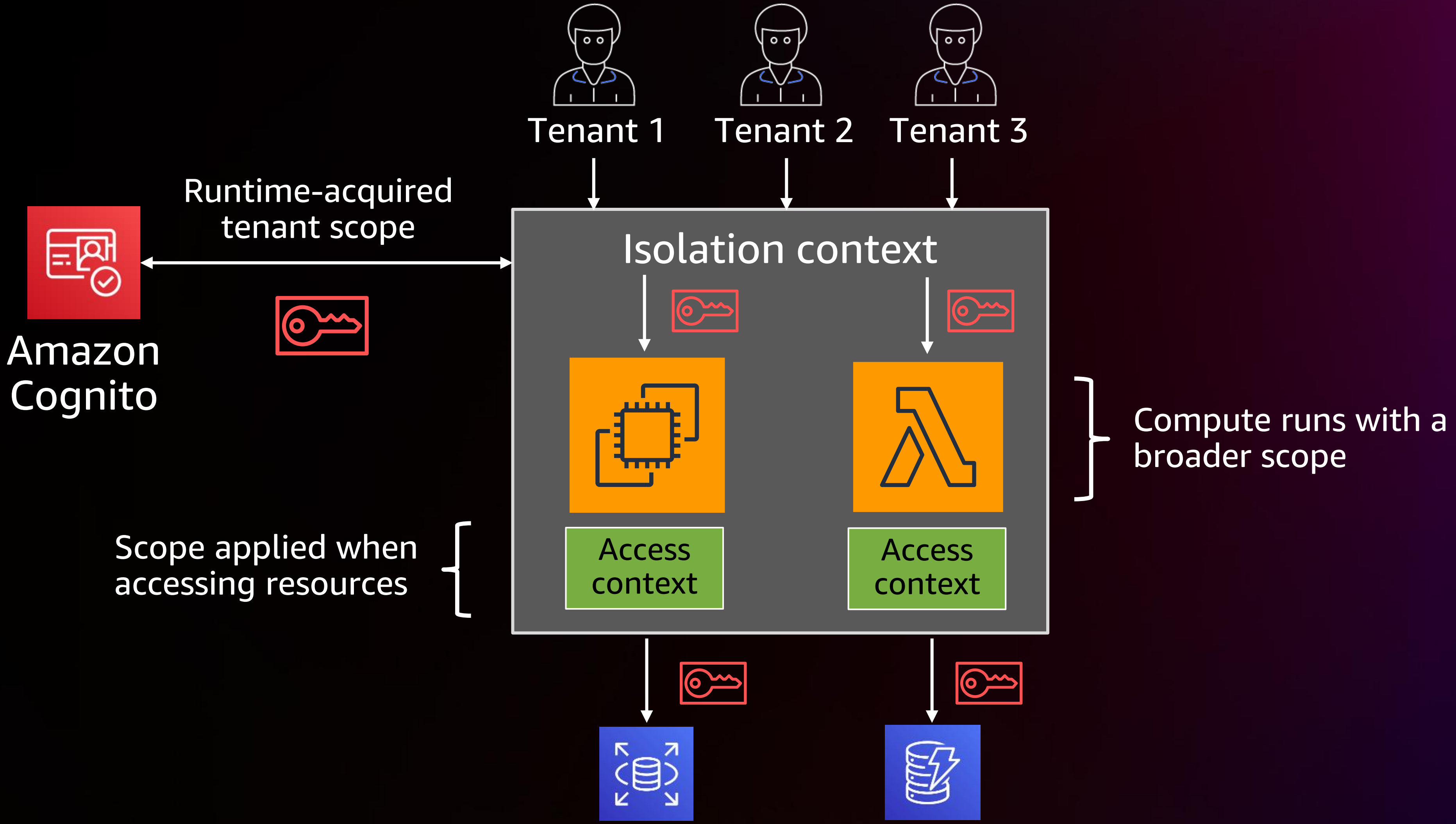
Item-level isolation (pool)



TenantId	ProductId	SKU
Tenant1	HMA-49004	4458585
Tenant3	NJC-68103	7839194
Tenant7	AMT-89293	8831965
Tenant2	UYM-94195	1343949

Unit of isolation is specific tenant item(s) within a resource

Runtime-enforced isolation



Tenant 1

Tenant 2

Tenant 3

Runtime-acquired tenant scope

Amazon Cognito

Isolation context

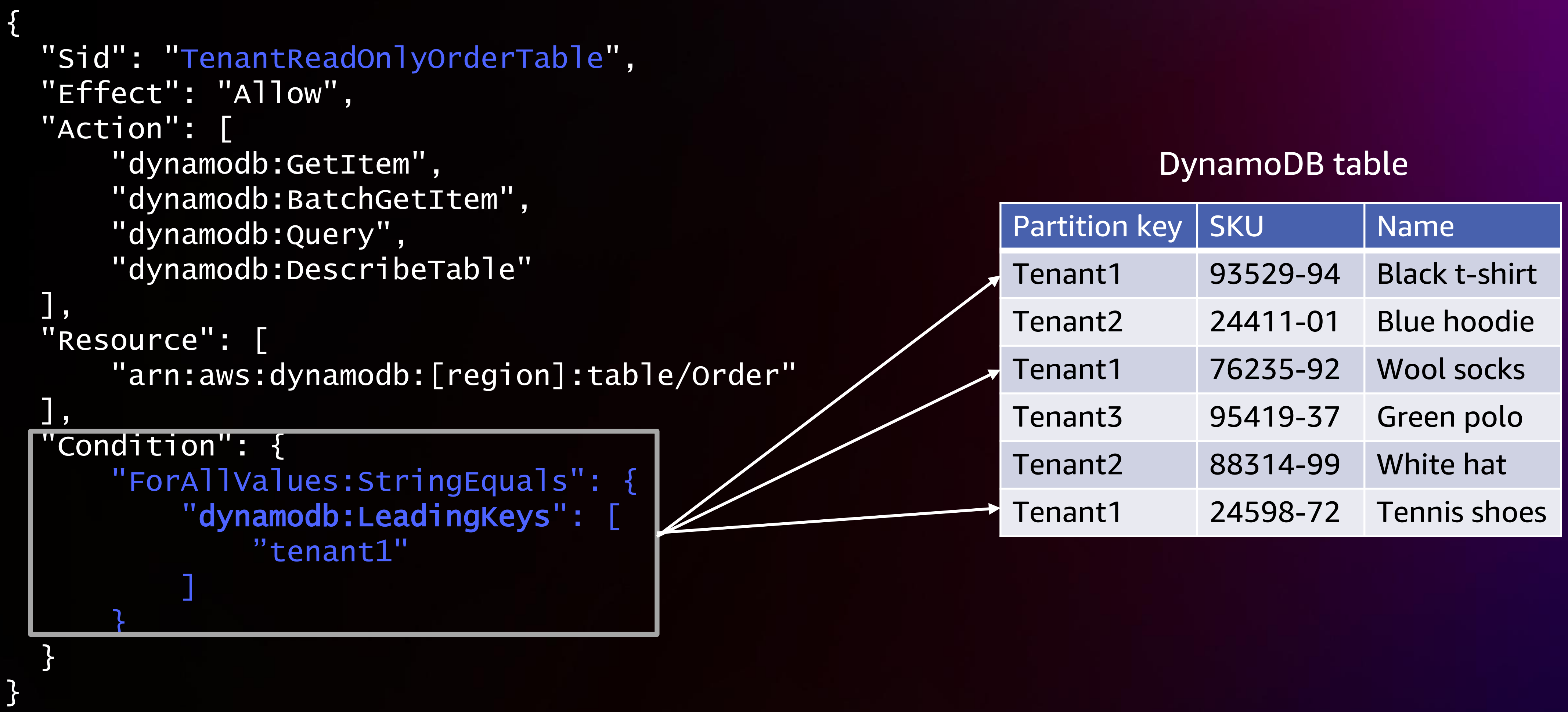
Access context

Access context

Compute runs with a broader scope

Scope applied when accessing resources

Item-level isolation with Amazon DynamoDB

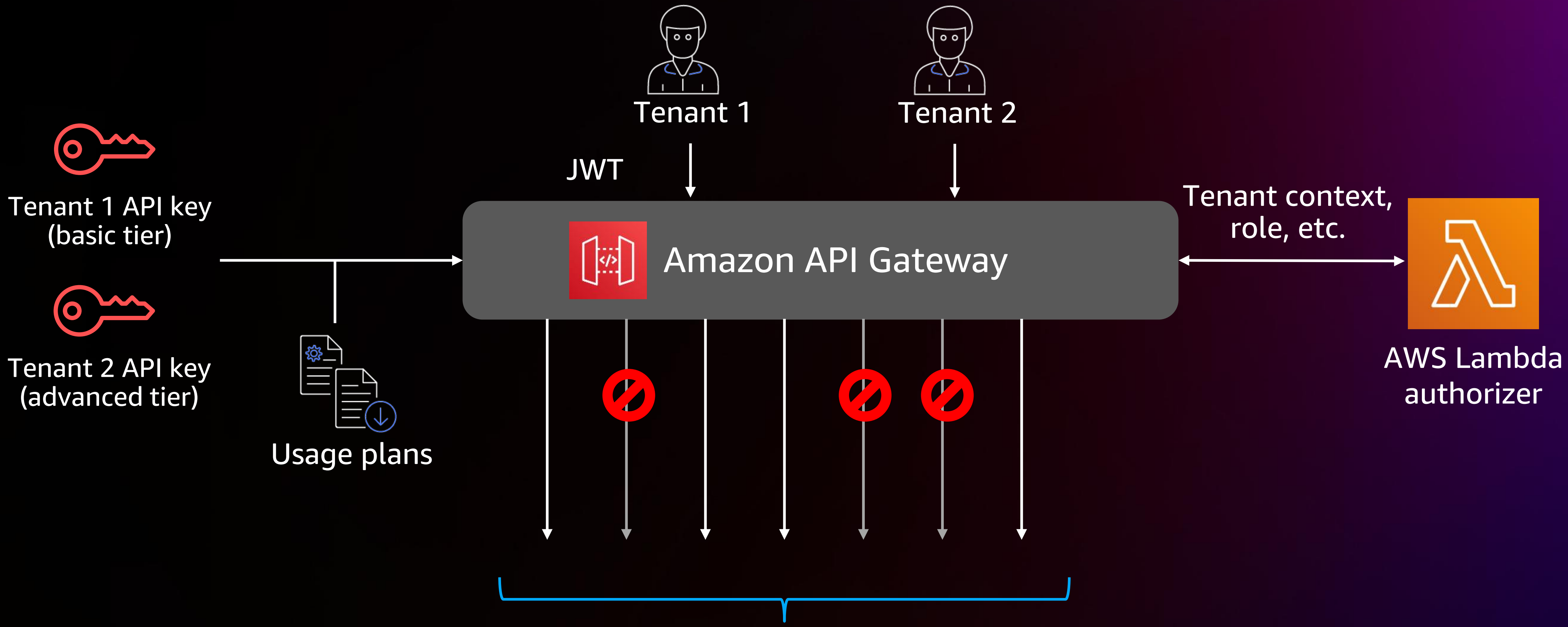


```
{
  "Sid": "TenantReadOnlyOrderTable",
  "Effect": "Allow",
  "Action": [
      "dynamodb:GetItem",
      "dynamodb:BatchGetItem",
      "dynamodb:Query",
      "dynamodb:DescribeTable"
  ],
  "Resource": [
      "arn:aws:dynamodb:[region]:table/Order"
  ],
  "Condition": {
      "ForAllValues:StringEquals": {
          "dynamodb:LeadingKeys": [
              ”tenant1"
          ]
      }
  }
}
```

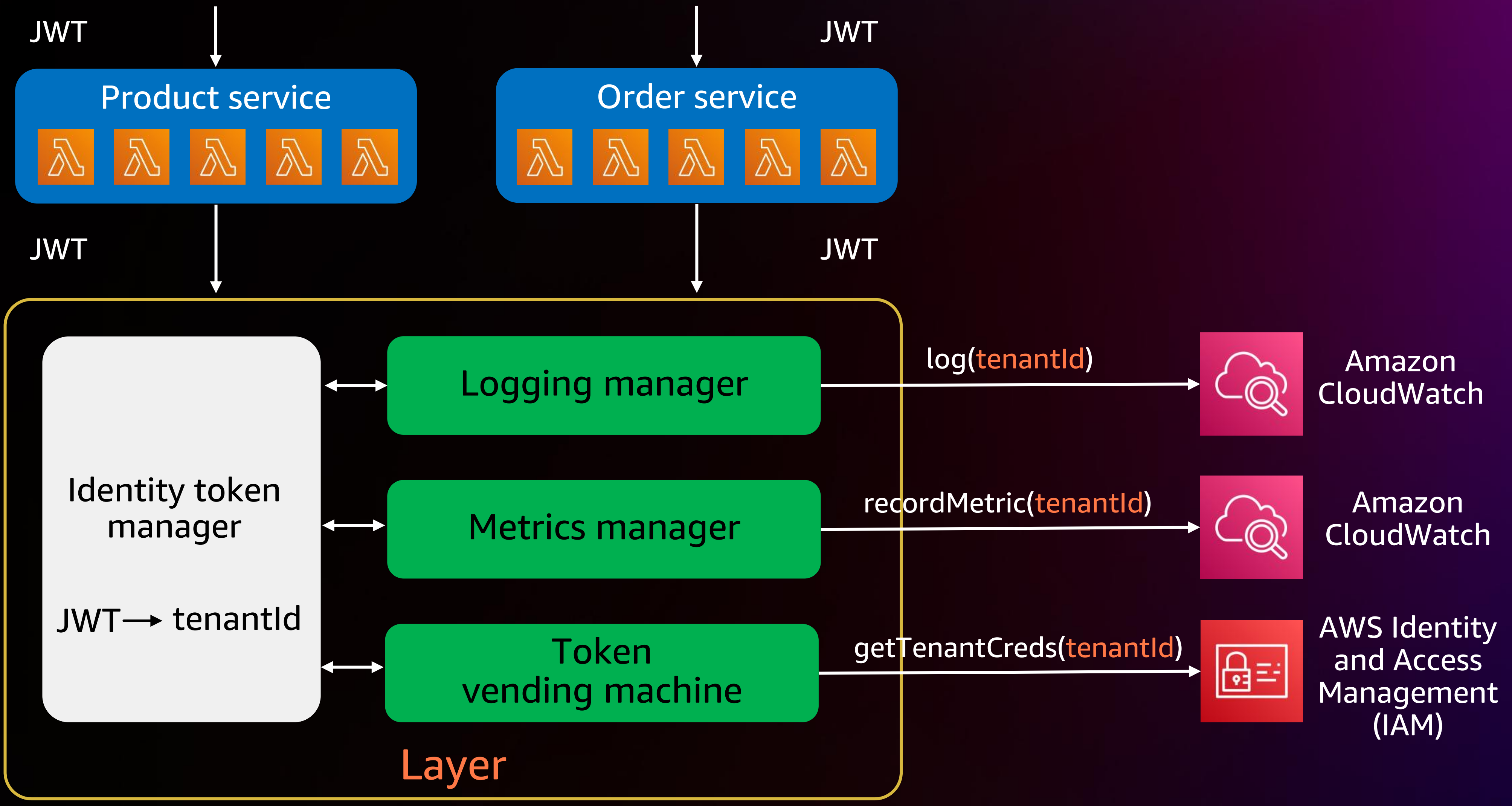
DynamoDB table

Partition key	SKU	Name
Tenant1	93529-94	Black t-shirt
Tenant2	24411-01	Blue hoodie
Tenant1	76235-92	Wool socks
Tenant3	95419-37	Green polo
Tenant2	88314-99	White hat
Tenant1	24598-72	Tennis shoes

Tier-based throttling

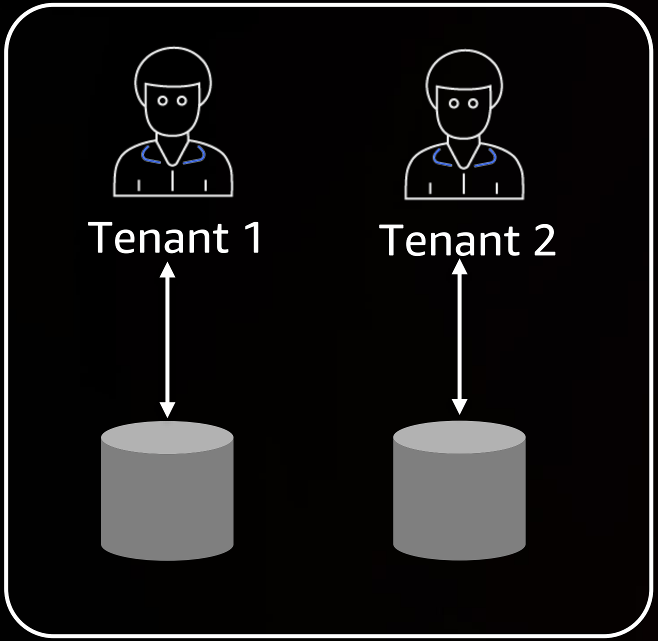


Multi-tenant microservice libraries

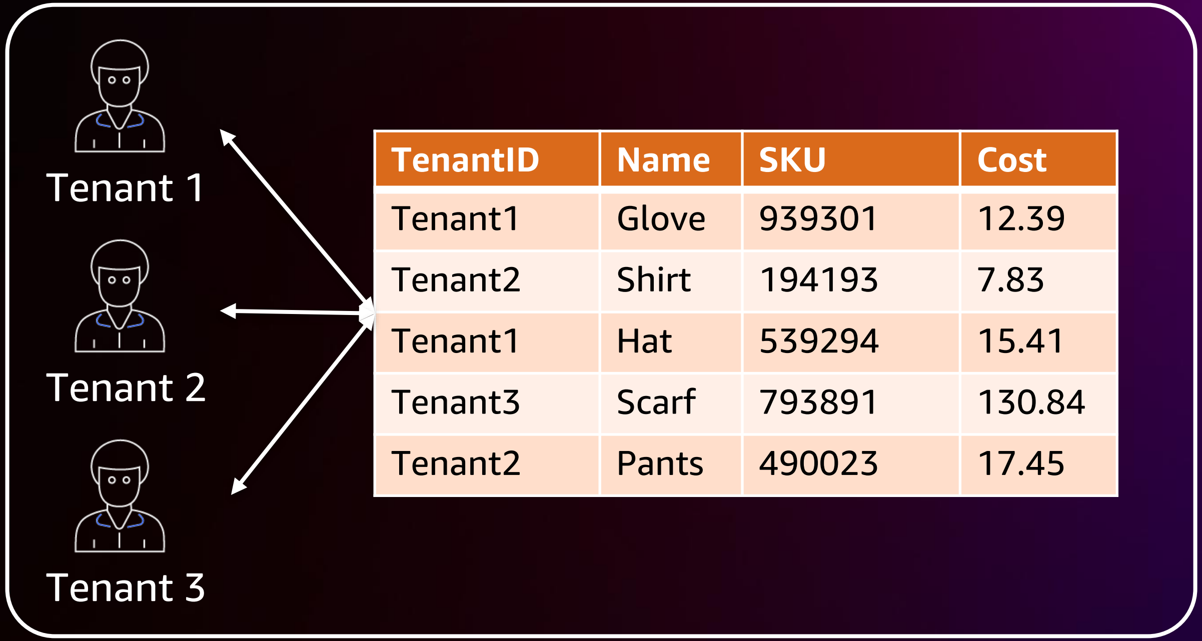


Data partitioning

Silo model



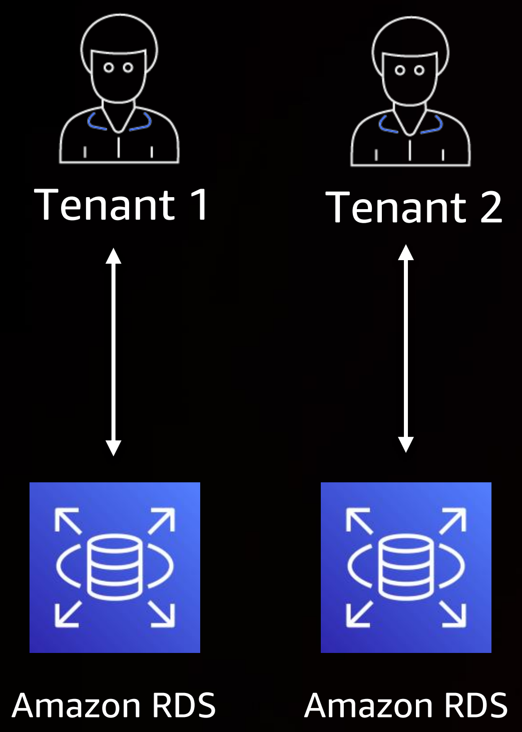
Pool model



TenantID	Name	SKU	Cost
Tenant1	Glove	939301	12.39
Tenant2	Shirt	194193	7.83
Tenant1	Hat	539294	15.41
Tenant3	Scarf	793891	130.84
Tenant2	Pants	490023	17.45

Data partitioning with Amazon RDS

Silo model



Tenant 1

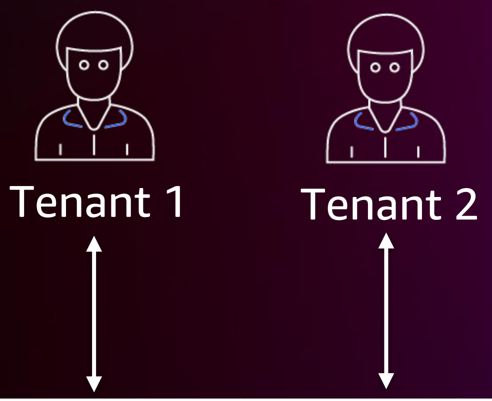
Tenant 2

Amazon RDS

Amazon RDS

Separate instance/database per tenant

Pool model



Tenant 1

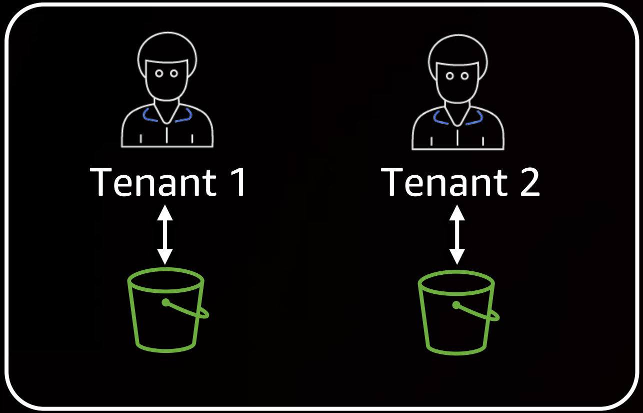
Tenant 2

TenantID	Name	SKU	Cost
Tenant1	Glove	939301	12.39
Tenant2	Shirt	194193	7.83
Tenant1	Hat	539294	15.41
Tenant2	Pants	490023	17.45

One database with tenant-indexed data

Data partitioning with Amazon S3

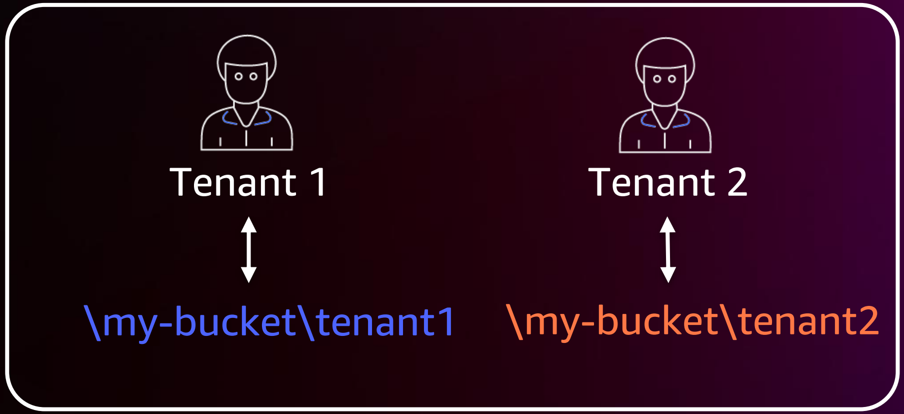
Bucket per tenant



Tenant 1

Tenant 2

Prefix per tenant



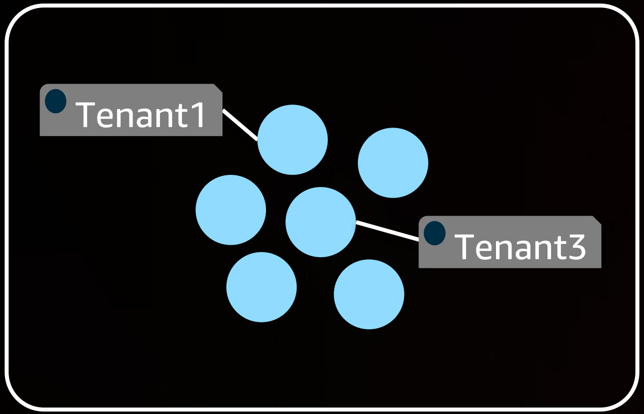
Tenant 1

Tenant 2

\my-bucket\tenant1

\my-bucket\tenant2

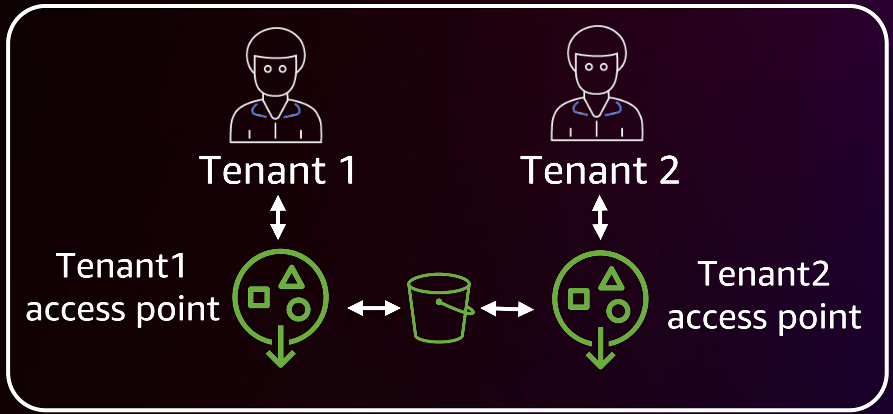
Tag per tenant



Tenant1

Tenant3

Access point per tenant



Tenant 1

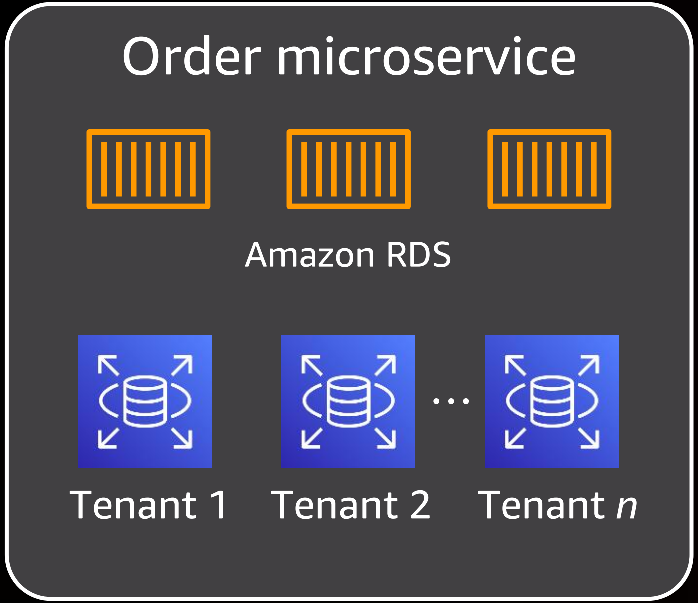
Tenant 2

Tenant1 access point

Tenant2 access point

Domain-driven data partitioning

Partitioning and isolation strategy should be decided on a service-by-service basis

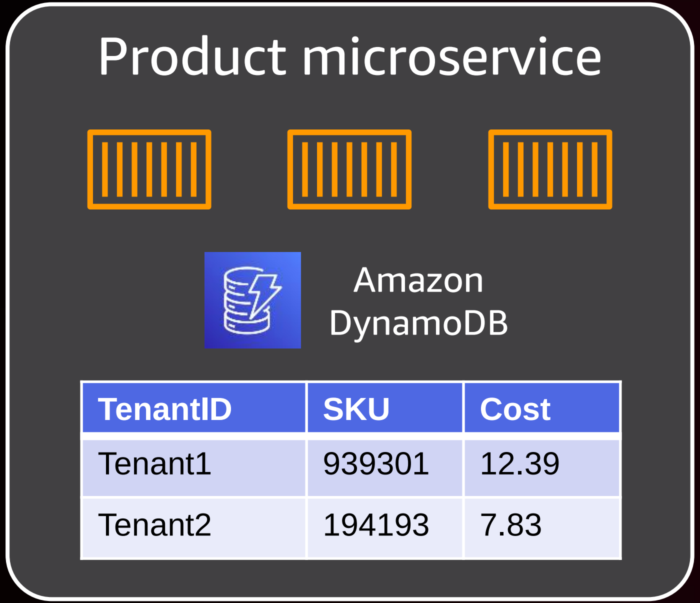


Order microservice

Amazon RDS

Tenant 1 Tenant 2 … Tenant *n*

Database per tenant (silo)

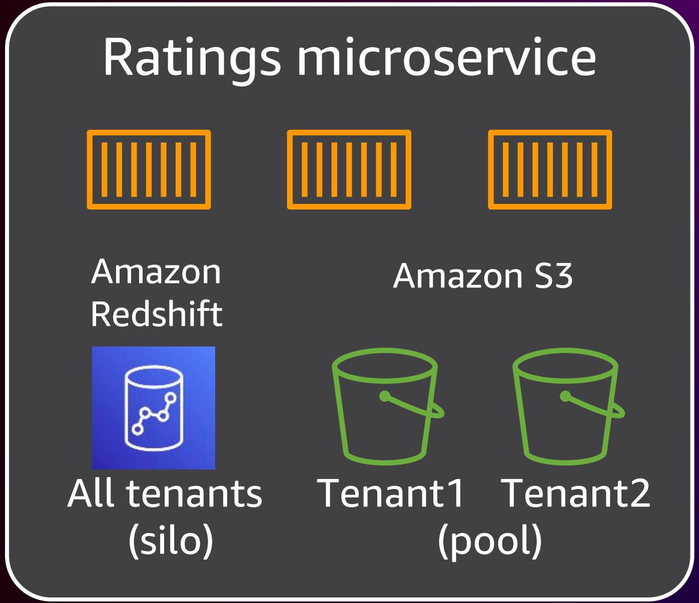


Product microservice

Amazon DynamoDB

TenantID	SKU	Cost
Tenant1	939301	12.39
Tenant2	194193	7.83

Shared database for all tenants (pool)



Ratings microservice

Amazon Redshift

Amazon S3

All tenants (silo)

Tenant1 Tenant2 (pool)

Mixed mode – silo and pool models

Takeaways

- There’s no universal SaaS blueprint
- Multi-tenancy does not require shared infrastructure
- Control plane services are essential to SaaS agility and innovation
- Each AWS stack/service may require a unique approach
- Silo and pool can be applied at the resource/microservice level
- Tiering, performance, and noisy neighbor will shape how and where you apply these patterns
- Find the combination of patterns that best matches your needs

More SaaS sessions

Breakout sessions

- SAS305 – SaaS architecture patterns: From concept to implementation
- SAS405 – SaaS microservices deep dive: Simplifying multi-tenant development
- SAS306 – SaaS migration: Inside a real-world multi-tenant transformation
- SAS302 – Supporting extensibility in SaaS environments
- PEX310 – Optimizing your multi-tenant SaaS architecture

Workshops

- SAS403 – SaaS microservices deep dive: Multi-tenancy meets microservices
- SAS402 – Serverless meets SaaS: Inside a real-world serverless SaaS solution
- SAS401 – Amazon EKS SaaS: Building a working multi-tenant environment

Business session

- PEX209 – Building your SaaS journey on AWS

More SaaS sessions

Chalk talks

- SAS307 – DevOps and SaaS: Applying automation in multi-tenant environments
- SAS303 – SaaS anywhere: Building SaaS solutions that run in hybrid models
- SAS301 – Multi-tenant meets ML: Building ML-based SaaS environments
- SAS304 – Solving the SaaS compliance puzzle
- PEX313 – The SaaS control plane: The heart of SaaS growth
- ARC403 – Amazon EKS SaaS deep dive: Inside a multi-tenant EKS solution
- ARC323 – Designing a multi-tenant SaaS tiering and throttling strategy
- SVS315 – Building multi-tenant applications with AWS Lambda and AWS Fargate

Builders' session

- ARC327 – How to optimize cost in your multi-tenant architecture

Additional resources

1

Subscribe to AWS SaaS Insights

Get monthly emails with bite-size advice and the latest updates

2

Explore the SaaS on AWS hub

Check out the SaaS on AWS page for more resources and insights

3

Discover resources for builders

Access our curated list of SaaS reference solutions, demos, tech events, and more

Thank you!

Tod Golding

todg@amazon.com

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