PRODYNA

AWS

Amazon Web Services

Training

Training Materials

Find docs, examples and practices on my github page:

https://github.com/DennisCreutz/aws-training

© PRODYNA

Agenda

- AWS Introduction
  - First Account
  - Billing
  - IAM
  - CloudWatch
- Networking
  - VPC
  - Subnets
  - Routing
  - Security Groups
  - Network Access Groups
  - NAT
- Cloud Computing
  - EC2
  - Load Balancer
  - Autoscaler
  - Elastic Beanstalk
- Database
  - RDS

© PRODYNA

Agenda

- Serverless
  - Lambda
  - Dynamo DB
  - API Gateway
  - S3
- Terraform Training
- AWS DevOps
  - AWS CloudFormation
  - AWS CodeCommit
  - AWS CodeBuild
  - AWS CodeDeploy
  - AWS CodePipeline

© PRODYNA

Agenda

- Web
  - Amazon Route53
  - Amazon CloudFront
  - Amazon Cognito
  - Amplify
- Containerization
  - ECS
  - EKS
- Multi Account Structure
- Misc

© PRODYNA

Introduction

Dennis Creutz
Managing IT Consultant / Cloud Architect

© PRODYNA

Introduction

- Name
- What you do?
- Experience with Cloud / AWS
- Expectations

© PRODYNA

Scope of this training

The scope of this training:

- Give an overview of the most used AWS services
- Provide basic knowledge and hands-on experience of the most important AWS services
- Give insights into AWS project live
- Serve as the basis for all AWS certificates
- Provide certification tips & insights

Not the scope of this training:

- Prepare for a specific AWS certification
- Learn about <u>all</u> AWS services
- Learn all details about AWS service XY

© PRODYNA

AWS Overview

© PRODYNA

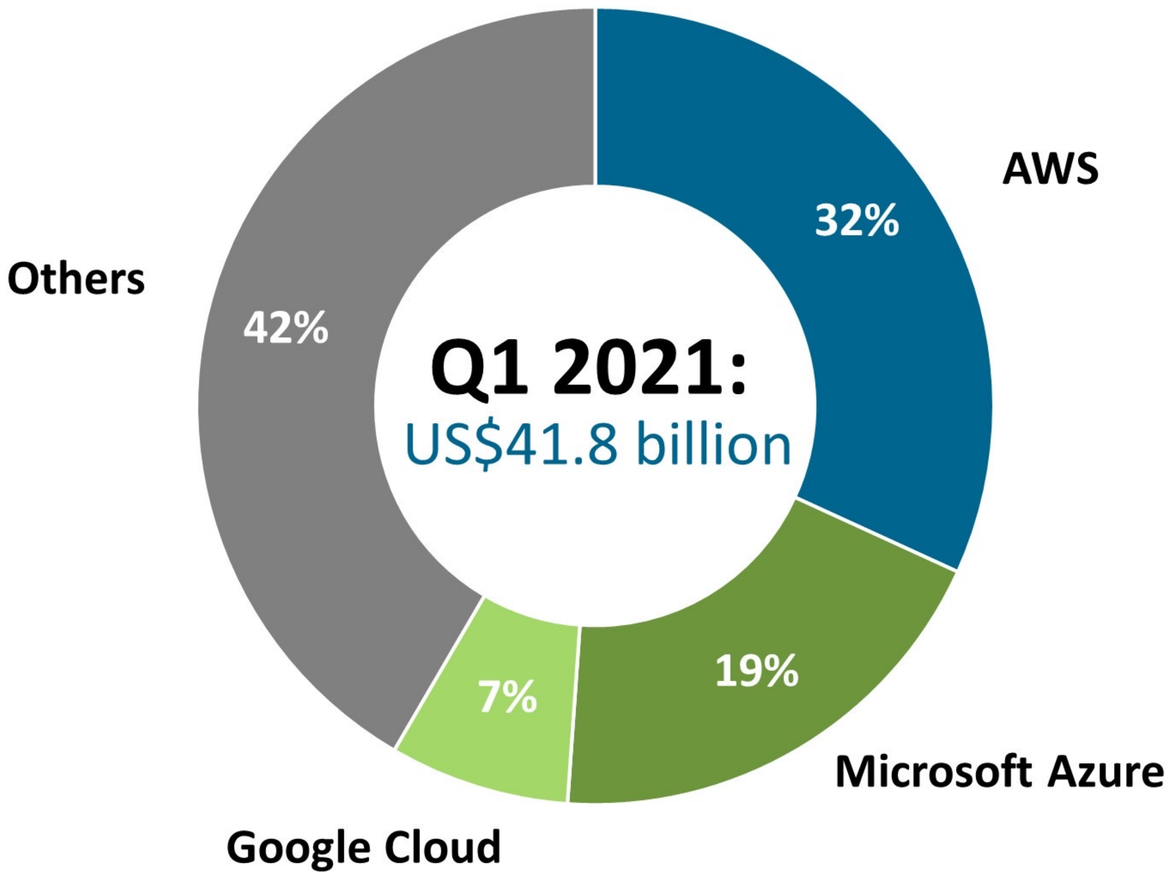
Amazon web services (AWS)

- Launched March 2006
- Subsidiary of Amazon
- Provides on-demand cloud computing platforms
- Mostly paid subscription basis (“Pay as you go”)
- Currently over 200 services

© PRODYNA

The top three cloud service providers accounted for 58% of total cloud spend in Q1 2021

Worldwide cloud infrastructure services spend



Source: Canalys estimates, April 2021

canalys

Adobe

NASA

N26

NETFLIX

slack

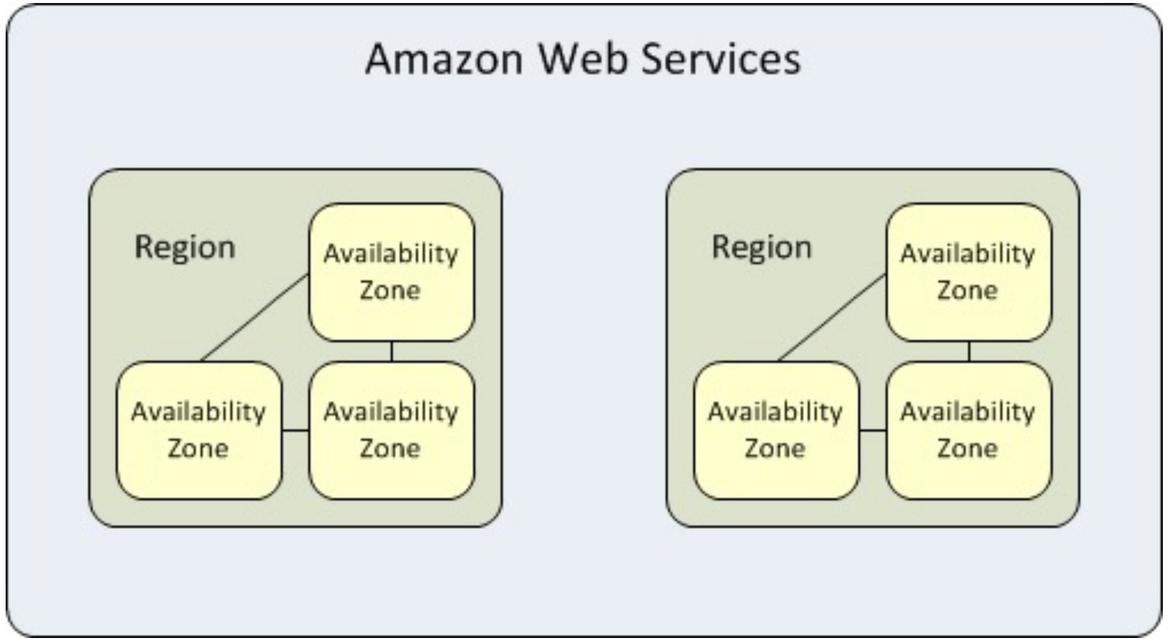
FORTNITE

SAMSUNG

© PRODYNA

Reliability and availability

- Reliability and availability though **Regions** and **Availability Zones**
- Region
  - A physical location around the world where AWS cluster data centers
  - 25 Regions*
- Availability Zones
  - Isolated and physically separate data center in a region
  - 81 Availability Zones*



*as of 2021

© PRODYNA

Regions

Coming Soon

AWS Free Tier

- AWS free tier allows you to use some services (aspects) for free.
- Two types of free tier:
  - 12 Month free: Benefits you get only for the first 12 month after account creation
  - Always free: Benefits you always get (even after the first 12 month)

© PRODYNA

IAM & Billing

© PRODYNA

AWS Identity and Access Management (IAM)

- **Category**: Security, Identity, & Compliance
- "AWS Identity and Access Management (IAM) enables you to manage access to AWS services and resources securely."
- **Pricing Model**: Free
- **Free tier**: Always free

© PRODYNA

IAM

- Users:
  Unique identity recognized by AWS services and applications
- Groups:
  Collection of IAM users
- Policies:
  Set of permissions
- Roles:
  Set of policies for making AWS service requests
  - Not associated with a specific user or group
  - Trusted entities (Users, applications, AWS services) <u>assume</u> roles

© PRODYNA

Policy language version ->
List of policy statements->

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "FirstStatement",                <- Statement ID (optional)
      "Effect": "Allow",                      <- Indicates if “Allow” or “Deny” statement
      "Action": ["iam:ChangePassword"],       <- List of actions that are allowed or denied, “{service}:{service-action}”
      "Resource": "*"                         <- List of resources this policy applies to, wildcards are possible
    },
    {
      "Sid": "SecondStatement",
      "Effect": "Allow",
      "Action": "s3:ListAllMyBuckets",
      "Resource": "*"
    },
    {
      "Sid": "ThirdStatement",
      "Effect": "Allow",
      "Action": [
        "s3:List*",                           <- Wildcards in service-actions are allowed
        "s3:Get*"
      ],
      "Resource": [
        "arn:aws:s3:::confidential-data",
        "arn:aws:s3:::confidential-data/*"
      ],
      "Condition": {"Bool": {"aws:MultiFactorAuthPresent": "true"}}
    }
  ]
}
```

Policy applies only if
this condition is met ->

© PRODYNA

Amazon CloudWatch

- **Category**: Management & Governance
- "CloudWatch provides you with data and actionable insights to monitor your applications, understand and respond to system-wide performance changes, optimize resource utilization, and get a unified view of operational health. "
- **Pricing Model**: Log Data, Metrics, Dashboards, etc.
- **Free tier**: 5 GB of log data are always free

© PRODYNA

Amazon CloudWatch - Feature

- Logging
- Metrics
- Alarms
- Custom Dashboards
- ...

© PRODYNA

Demo: IAM & Billing

© PRODYNA

Demo & Practice: First AWS Account

© PRODYNA

Practice: First AWS Account

- Form teams around the people with AWS credentials and decide on a team name
- Login with your AWS credentials
- Set region to Frankfurt (eu-central-1)
- Create
  - ...a group „Developers“ with „AdministratorAccess“ policy
  - ... users (programmatic & console access) for each team member with an autogenerated password and add the users to the “Developers” group
  - ...a budget of $10 and alerts on 50%,100% and 150% of the budget to your e-mail addresses
  - ...a tag based resource group for the tag „team“: *YOUR_TEAM_NAME*

© PRODYNA

Practice: First AWS Account

- Create a new policy “S3Creator” with...
  - ... S3 all LIST actions
  - ... S3 all READ actions
  - .... S3 createBucket action
  - For all resources
- Create a new group “S3Creators”, add the “S3Creator” permission, add one user to this group and remove him from the “Developers” Group
- The user in the ”S3Creators” group:
  - Re-Login into AWS console
  - Go to “S3” and create a new S3 bucket with the name of your team
  - Can you create the bucket?
  - Can you delete the bucket?
  - Can you move your user back to the “Developers” group?

© PRODYNA

Questions: IAM & Billing

© PRODYNA

Questions: IAM & Billing

How do you make sure your organization does not exceed its monthly budget?

a. Sign up for the free alert under filing preferences in the AWS Management Console.
b. Set a schedule to regularly review the Billing an Cost Management dashboard each month.
c. Create an email alert in AWS Budget
d. In CloudWatch, create an alarm that triggers each time the limit is exceeded.

Answer: C. Create an email alert in AWS Budget.

AWS Budgets gives you the ability to set custom budgets that alert you when your costs or usage exceed (or are forecasted to exceed) your budgeted amount.

© PRODYNA

Questions: IAM & Billing

What is the relationship between AWS global infrastructure and the concept of high availability?

a. AWS is centrally located in one location and is subject to widespread outages if something happens at that one location.
b. AWS regions and Availability Zones allow for redundant architecture to be placed in isolated parts of the world.
c. Each AWS region handles a different AWS services, and you must use all regions to fully use AWS.
d. None of the above

Answer: B.

As an AWS user, you can create your applications infrastructure and duplicate it. By placing duplicate infrastructure in multiple regions, high availability is created because if one region fails you have a backup (in a another region) to use.

© PRODYNA

Networking

© PRODYNA

Amazon Virtual Private Cloud (VPC)

- **Category:** Network
- "Amazon Virtual Private Cloud (Amazon VPC) lets you provision a logically isolated section of the AWS Cloud where you can launch AWS resources in a virtual network that you define. "
- **Pricing Model:** Free except VPN Connection & NAT Gateway
- **Free tier**: Mostly always free (exception e.g. NAT)

© PRODYNA

Networking

- VPC:
  Logical isolated section of the AWS Cloud where you can launch AWS resources in a virtual network that you define.
  A VPC is assigned to a region.
- Subnet:
  A subnet is a part of a VPC and is assigned to a zone in the region of the VPC.
  The IP adress range of a subnet needs to be a subset of the subnet's VPC IP adress range.

  Example:
  VPC = 192.168.0.0/24
  Subnet = 192.168.0.0/25

© PRODYNA

CIDR Notation

- Number of addresses = $2^{32-\text{Suffix}}$
  E.g.: 192.168.0.0/30 = $2^{32-30} = 2^2 = 4$ addresses
  AWS reserves 5 addresses!
- VPC subnet
  - Minimum suffix is /28
  - Maximum suffix is /16
- Private CIDR's:
  - 10.0.0.0/8 | 10.0.0.0 – 10.255.255.255
  - 172.16.0.0/12 | 172.16.0.0 – 172.31.255.255
  - 192.168.0.0/16 | 192.168.0.0 – 192.168.255.255

© PRODYNA

Basic Network Security

- Security Group:
  - Firewall on instance level
  - Inbound and outbound rules
  - Stateful: Any changes applied to an incoming rule will be automatically applied to the outgoing rule
  - Support allow rules only (by default all rules are denied)
  - All rules applied
- Network Access Control List (ACL):
  - Firewall on subnet level
  - Inbound and outbound rules
  - Stateless
  - Support allow and deny rules
  - All rules processed in order (lower number first) and first rule matches is applied

© PRODYNA

Example Security Group

Type	Protocol	Port Range	Source	Description
HTTP	TCP	80	0.0.0.0/0	
HTTP	TCP	80	::/0	
HTTPS	TCP	443	0.0.0.0/0	
HTTPS	TCP	443	::/0	

© PRODYNA

Example NACL

Rule #	Type	Protocol	Port Range	Source	Allow / Deny
10	SSH (22)	TCP (6)	22	0.0.0.0/0	DENY
100	ALL Traffic	ALL	ALL	0.0.0.0/0	ALLOW
*	ALL Traffic	ALL	ALL	0.0.0.0/0	DENY

© PRODYNA

Network address translation (NAT)

- Used to hide private network from outside
- One Internet-routable IP address of a NAT gateway can be used for an entire private network
- Use case:
  - Secure infrastructure in a private network without direct access from outside
  - Allow request to the outside network over the NAT in a public network
  - Allow only responses from outside
  - No request from outside are possible

© PRODYNA

Demo: Networking

© PRODYNA

Questions: Networking

© PRODYNA

Questions: Networking

Which of the following features is associated with a Subnet in a VPC to protect against Incoming traffic requests?

a. AWS Inspector
b. Subnet Groups
c. Security Groups
d. NACL

Answer D. NACL

A network access control list (ACL) is an optional layer of security for your VPC that acts as a firewall for controlling traffic in and out of one or more subnets. You might set up network ACLs with rules similar to your security groups in order to add an additional layer of security to your VPC.

© PRODYNA

Cloud Computing & Database

© PRODYNA

Elastic Compute Cloud (EC2)

- **Category:** Compute
- "Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides secure, resizable compute capacity in the cloud."
- **Pricing Model:** Compute capacity in seconds / minutes
- **Free tier**: 750 Hours t2.micro for 12 Month

© PRODYNA

EC2 Provisioning Types

- On-Demand
  - Pay per hour or second
  - No longer-term commitments or upfront payments are needed
- Spot
  - Request spare Amazon EC2 computing capacity
  - Bidding system
    - AWS sets prices for Spot instances
    - Prices continuously changes
    - Bid higher or equals = Instance starts
    - Lower bid = Instances shutdown (2 min. time to clean up)
  - Up to 90% discount

© PRODYNA

EC2 Provisioning Types

- Reserved
  - Purchase instance for 1 or 3 years
  - Up to 75% discount
- Dedicated Host
  - Physical EC2 server dedicated for your use
  - On-demand or reserved

<u>Very important for all AWS certificates!</u>

© PRODYNA

EC2 Instance Types

- MANY instance types
- Categories:
  - General purpose
  - Compute optimized
  - Memory optimized
  - Accelerated Computing
  - Storage Optimized
- All types: https://aws.amazon.com/ec2/instance-types/
- General purpose instances (e.g. “t3” instances) are normally a good starting point

© PRODYNA

Elastic Load Balancing (ELB)

- **Category:** Compute
- "Elastic Load Balancing automatically distributes incoming application traffic across multiple targets, such as Amazon EC2 instances, containers, IP addresses, and Lambda functions."
- **Pricing Model:** Hours running & capacity units used
- **Free tier**: 750 Hours for 12 Month

© PRODYNA

Elastic Load Balancer

- Application Load Balancer
  - Layer 7 (HTTP / HTTPS)
  - Host and path based routing possible
- Network Load Balancer
  - Layer 4 (TCP / UDP)
  - Capable of handling millions of requests per second while maintaining ultra-low latencies.
- Classic Load Balancer
  - Layer 4 and 7
  - Just forwards requests

© PRODYNA

Elastic Load Balancer

- Target Group
  A group of instances the ELB can route traffic to.
  An instance must pass a health check first to get registered to the target group.
- Listener
  Define the protocol and port, where the load balancer listens for incoming connections.

© PRODYNA

AWS - Auto Scaling

- **Category:** Compute
- "AWS Auto Scaling monitors your applications and automatically adjusts capacity to maintain steady, predictable performance at the lowest possible cost."
- **Pricing Model:** Free
- **Free tier**: Always

© PRODYNA

Auto Scaling

- Launch Template
  - Template to automatically create EC2 instances
  - Launch **Configuration**: Are like Launch Templates without versioning. Kind of deprecated.
- Auto Scaling Group
  - Defines minimum, maximum and desired capacity
  - Can use metrics to scale
  - Uses the launch template to create new instances

© PRODYNA

Demo: Computing & Database I

© PRODYNA

Amazon Relational Database Service (RDS)

- **Category:** Database
- "Amazon Relational Database Service (Amazon RDS) makes it easy to set up, operate, and scale a relational database in the cloud. It provides cost-efficient and resizable capacity while automating time-consuming administration tasks (...).
- **Pricing Model:** Hour
- **Free tier**: 750 hours (no Aurora)

© PRODYNA

RDS Database Types

- PostgreSQL
- MySQL
- MariaDB
- Oracle
- MicrosoftSQL Server

© PRODYNA

RDS -Features

- Simple integration with other AWS services (e.g. CloudWatch Logs & Metrics)
- Multi-Availability Zone (Multi-AZ) data replication
- Encryption
- Can create Read Replicas to unload read load from the main instance
- Cross-Region Read Replicas = Nearly instant disaster recovery
- Can utilise the AWS Database Migration Service (DMS)
- Many more...

© PRODYNA

Amazon Aurora

- Relational database built for the cloud
- MySQL and PostgreSQL-compatible
- Up to five times faster than standard MySQL databases
- Up to three times faster than standard PostgreSQL databases

© PRODYNA

Demo: Computing & Database II

© PRODYNA

Questions: Computing & Database

© PRODYNA

Questions: Cloud Computing & Database

You have EC2 instances running at 90% utilization and you expect this to continue for at least a year.
What type of EC2 instance would you choose to ensure your cost stay at a minimum?

a. Dedicated host instances
b. On-demand instances
c. Spot instances
d. Reserved instances

Answer: D. Reserved instances

Reserved instances are the best choice for instances with continuous usage and offer a reduced cost because you purchase the instance for the entire year.

© PRODYNA

Questions: Cloud Computing & Database

There is a requirement hosting a set of servers in the Cloud for a short period of 3 months. Which of the following types of instances should be chosen to be cost effective?

a. Spot Instances
b. On-Demand
c. No Upfront costs Reserved
d. Partial Upfront costs Reserved

Answer: B. On-Demand

Since the requirement is just for 3 months, then the best cost effective option is to use On-Demand Instances.

© PRODYNA

Questions: Cloud Computing & Database

The firm you work for is considering migrating to AWS. They are concerned about cost and the initial investment needed. Which of the following features of AWS pricing helps lower the initial investment amount needed? Choose 2 answers from the options given below:

a. The ability to choose the lowest cost vendor.

b. The ability to pay as you go

c. No upfront costs

d. Discounts for upfront payments

© PRODYNA

Questions: Cloud Computing & Database

The firm you work for is considering migrating to AWS. They are concerned about cost and the initial investment needed. Which of the following features of AWS pricing helps lower the initial investment amount needed?
Choose 2 answers from the options given below:

a. The ability to choose the lowest cost vendor.
b. The ability to pay as you go
c. No upfront costs needed
d. Discounts for upfront payments

Answers: B and C

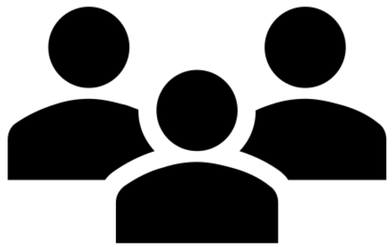
The best features of moving to the AWS Cloud is: No upfront cost and the ability to pay as you go where the customer only pays for the resources needed.
You can get discount for upfront payments but the questions was about lower the initial investment.

© PRODYNA

Practice: Computing & Database

© PRODYNA

Practice 1.1: Cloud Computing & Database

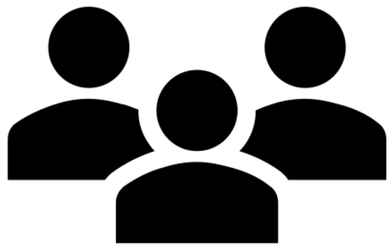


1. Create a VPC with 3 public and 3 private subnets
2. Create an RDS MySQL database in the private subnets
3. Create an EC2 bastion host in one public subnet
4. Try to connect through SSH tunnel:
   ssh -N -L localPort:rdsHost:remotePort ec2-user@remoteHost -i ~/path/to/key

- Don't forget to change the RDS security group!
- Don't forget to use your team tag:
  "team": *YOUR_TEAM_NAME*

© PRODYNA

Practice 1.2: Cloud Computing & Database



1. Create a Target Group
    1. Use /phpinfo.php as health check
    2. Don't forget to add the Target Group to Auto Scaling Group
2. Create an Application Load Balancer
    1. In all 3 public subnets
    2. With this Target Group
3. Create a Launch Template from shared image "apache_phpinfo"
4. Create an Autoscaling Group with this Launch Template
    1. In all 3 private subnets
    2. Min./Max./Desired Size = 2
5. Wait for everything to come up & access the Webservers over the ALB:
   [ALB_URL]/phpinfo.php

- Don't forget to use your team tag:
  "team": *YOUR_TEAM_NAME*

© PRODYNA

Serverless

© PRODYNA

Do we need dedicated servers?

"30% Of Servers Are Sitting 'Comatose'" - *Uptime Institute*

"(...) servers (...) deliver between five and 15 percent of their maximum computing output on average over the course of the year." - *McKinsey and Company*

"The purchased hardware and software require ongoing support and maintenance, approximately 20 percent of the upfront cost per year. " – *Kenny & Company*

© PRODYNA

Serverless Architecture - Definition

- “A serverless architecture is a way to build and run applications and services without having to manage infrastructure.”
- That means:
  - Zero administration
  - Auto-scaling
  - Pay-per-use

© PRODYNA

Amazon Simple Storage Service (S3)

- **Category:** Storage
- "Amazon Simple Storage Service (Amazon S3) is an object storage service that offers industry-leading scalability, data availability, security, and performance."
- **Pricing Model:** Per GB - higher cost for faster access times / Per request
- **Free tier**: 5 GB standard storage, 20,000 GET Requests, and 2,000 PUT Requests for 12 month

© PRODYNA

Amazon Simple Storage Service - Glacier

- **Category:** Storage
- "Amazon S3 Glacier is a secure, durable, and extremely low-cost cloud storage service for data archiving and long-term backup."
- **Pricing Model:** Very low storage but high request cost (compared to S3)
- **Free tier**: 10 GB data retrievals are always free

© PRODYNA

S3 Storage Classes

	S3 Standard	S3 Intelligent-Tiering*	S3 Standard-IA	S3 One Zone-IA†	S3 Glacier	S3 Glacier Deep Archive
Designed for durability	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)
Designed for availability	99.99%	99.9%	99.9%	99.5%	99.99%	99.99%
Availability SLA	99.9%	99%	99%	99%	99.9%	99.9%
Availability Zones	≥3	≥3	≥3	1	≥3	≥3
Minimum capacity charge per object	N/A	N/A	128KB	128KB	40KB	40KB
Minimum storage duration charge	N/A	N/A	30 days	30 days	90 days	180 days
Retrieval charge	N/A	N/A	per GB retrieved	per GB retrieved	per GB retrieved	per GB retrieved
First byte latency	milliseconds	milliseconds	milliseconds	milliseconds	select minutes or hours	select hours
Storage type	Object	Object	Object	Object	Object	Object
Lifecycle transitions	Yes	Yes	Yes	Yes	Yes	Yes

© PRODYNA

Amazon DynamoDB

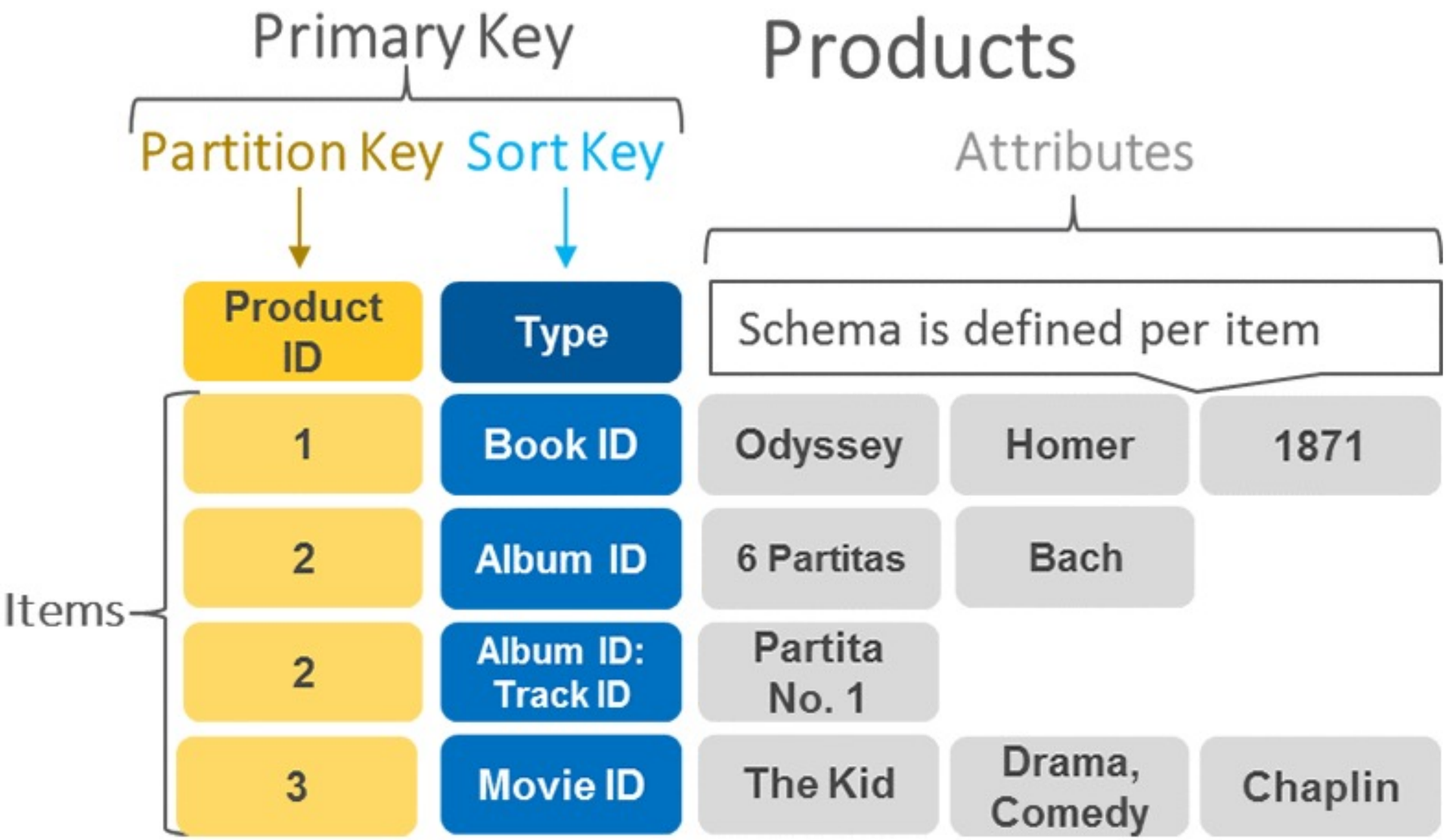
- **Category:** Database
- “Amazon DynamoDB is a key-value and document database (...). It's a fully managed, multiregion, multimaster database with built-in security, backup and restore, and in-memory caching for internet-scale applications.”
- **Pricing Model:** Reading, writing, and storing data
- **Free tier**: 25 GB Storage, 25 Read- and 25 Write-Capacity for 12 month

© PRODYNA

NOSQL

- NoSQL databases stores information in JSON documents...
  - ...instead of columns and rows used by relational databases
- NoSQL tables are build for a specific purpose
- Often sacrifices ACID for speed and scalability
- Concept of “Eventual consistency“
- Doesn’t require normalized data (dynamic “columns”)
  - No database schema
- Mostly SQL-like query language
- SQL = Queries are build for the table
  NoSQL = Table is build for the queries

© PRODYNA


Primary Key
Products
Partition Key
Sort Key
Attributes
Product ID
Type
Schema is defined per item
Items
1
Book ID
Odyssey
Homer
1871
2
Album ID
6 Partitas
Bach
2
Album ID: Track ID
Partita No. 1
3
Movie ID
The Kid
Drama, Comedy
Chaplin

© PRODYNA

	Primary Key		Attributes											
	PK	SK (GSI-1-PK)	GSI-1-SK											
HR OE TABLE	HR-EMPLOYEE1	EMPLOYEE1	Data (Full Name)	StartDate	EndDate	JobID	JobTitle	PhoneNumber	Email	ManagerID	Country	City	Region	Department
			John Smith											
		QUOTA-2017-Q1	Data (Order Totals USD)	EmployeeName										
			50000											
		HR-CONFIDENTIAL	Data (Hire Date)	EmployeeName	Salary	CommissionPct								
			2015-11-08											
		WA|SEATTLE	Data (Desk Location)	EmployeeName										
			B01|F07|A27|R05	John Smith										
		J-AM3	Data (Job Title)	DepartmentID	StartDate	EndDate	JobID							
			Principal Account Manager											
		JH-AM2	Data (Job Title)	DepartmentID	StartDate	EndDate	JobID							
			Senior Account Manager											
		JH-AM1	Data (Job Title)	DepartmentID	StartDate	EndDate	JobID							
			Account Manager											
	HR-REGION1	PNW	Data (Region Name)	RegionName										
			Pacific Northwest Territory											
	HR-COUNTRY1	USA	Data (Country Name)	CountryName	RegionID									
			United States											
	HR-LOCATION1	WA|SEATTLE	Data (City State)	CityName	PostalCode	StreetAddress	StateProvince	CountryID						
			Seattle, Washington											
	HR-JOB1	J-AM3	Data (Job Title)	JobTitle	MinSalary	MaxSalary								
			Principal Account Manager											
	HR-DEPARTMENT1	COMMERCIAL	Data (Department Name)	DepartmentName	ManagerID	City	Location							
			Commercial Sales		EMPLOYEE2									
	OE-CUSTOMER1	CUSTOMER1	Data (Customer Name)	Address	IncomeLevel	PhoneNumber	NLSLanguage	NLSTerritory	CreditLimit	CustEmail	CustLocatic	DateOfBirth	MaritalStatus	Gender
			ACE Building Supplies											
	OE-ORDER1	CUSTOMER1	Data (StatusDate) (GSI-2-SK)	GSI-Bucket (GSI-2-PK)	SalesRepID	AccountManager	OrderMode	OrderTotal	PromotionID					
			OPEN#2018_08_11	RND(0,N)	EMPLOYEE1									
		EMPLOYEE1	Data (StatusDate)	Order Total										
			OPEN#2018_08_11	2500										
		PRODUCT1	Data (StatusDate) (GSI-2-SK)	GSI-Bucket (GSI-2-PK)	OrderQuantity	UnitPrice								
			OPEN#2018_08_11	RND(0,N)										
	OE-PRODUCT1	PRODUCT1	Data (Product Name)	ProductDescription	WAREHOUSE1	WAREHOUSE2	CategoryID	WeightClass	WarranytPeri	SupplierID	ProductSta	ListPrice	MinPrice	CatalogURL
			Quickcrete Cement - 50 lb bag		InventoryQty	InventoryQty								
		PNW	Data (Region Name)	TranslatedName	Description									
			Pacific Northwest	TRANSLATED_NAME										
	OE-WAREHOUSE1	PNW	Data (Warehouse Type)	WarehouseSpec	Location	WHGeoLocation								
			Building Supplies											

PK = Partition Key SK = Sort Key GSI = Global Secondary Index

© PRODYNA

Amazon DynamoDB

- Advantages
  - Highly scalable & fast
  - Cost effective
  - Serverless & truly cloud native
  - Easy to manage
  - Very good integrated in AWS ecosystem
  - Can use Strongly Consistent Reads (if specified)
- Disadvantages
  - Table is very specialised
    - Table design is very important
  - No joins
  - No multi-region backups
  - AWS specific

Amazon DynamoDB and NoSQL are topics on its own. More informations:
https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/Introduction.html
https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/best-practices.html

© PRODYNA

Demo: S3 & DynamoDB

© PRODYNA

AWS Lambda

- **Category:** Compute
- "AWS Lambda lets you run code without provisioning or managing servers."
- **Pricing Model:** Requests and duration per memory
- **Free tier**: 1,000,000 requests / 3.2 million seconds of compute time are always free

© PRODYNA

AWS Lambda

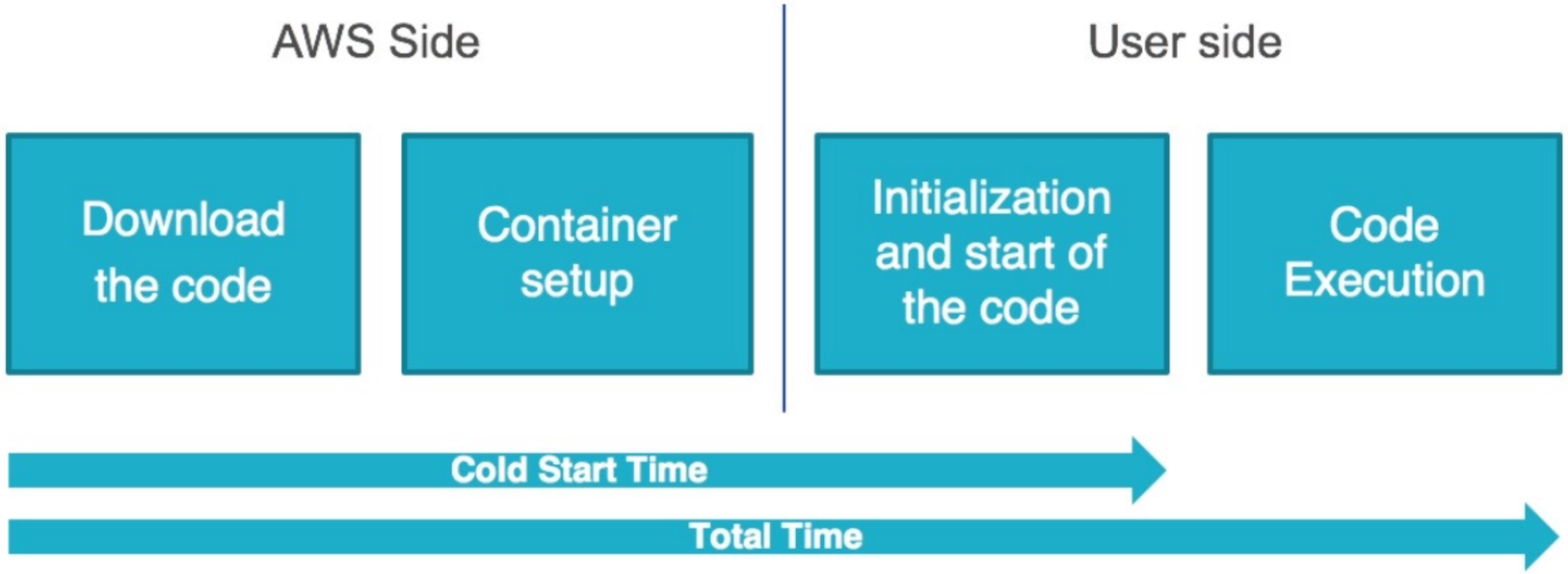
- Natively supports Java, Go, PowerShell, Node.js, C#, Python, and Ruby
  - Any other language over runtime API
- Built-in fault tolerance
- Automatic scaling
- Very good integration with other AWS services

© PRODYNA

AWS Lambda – Cold vs Warm Starts

- Lambda's are containers behind the scenes
- First time call Lambda = Spin up container
  - Takes more time to execute Lambda
  - This is called a "Cold Start"
  - Each call after is faster = "Warm Start"
- Lambda not called for some time = Container shuts down = Next call is "Cold Start"

© PRODYNA



Source : AWS re:Invent 2017 – Become a serverless Black Belt

© PRODYNA

How to prevent cold starts?

Basically two options to prevent cold starts:

- Unofficial: Call your Lambda every X minutes and keep it warm
  - Only slightly increased costs
  - No supported by AWS
  - No guarantee for only warm starts
- Official: Reserved concurrency
  - X Lambdas stay warm every time (guaranteed)
  - You pay (much...) for the reserved concurrency
  - Supported by AWS

© PRODYNA

AWS Lambda – The VPC Problem

Lambdas can't access resources in a VPC (e.g. Amazon RDS database) per default

- Lambda must be placed in a VPC to access VPC resources
- Lambda in VPC = No access to the internet per default
  - Need NAT or VPC Endpoints to access internet & other AWS services (outside the VPC)
- The cold start in a VPC is higher (prior 2019 it was much higher)
- Deleting Lambdas in a VPC can take up to 40 minutes

© PRODYNA

Amazon API Gateway

- **Category:** Network
- "Amazon API Gateway is a fully managed service that makes it easy for developers to create, publish, maintain, monitor, and secure APIs at any scale."
- **Pricing Model:** API calls & Data transferred
- **Free tier**: 1 Million API Calls for 12 month

© PRODYNA

Amazon API Gateway - Features

- Canary deployment
- Caching
- Request throttling
- Generating API SDK
- Generating Swagger & OpenAPI definition
- Authorization (AWS Lambda & AWS Cognito)
- And more...

© PRODYNA

Demo: AWS Lambda & API Gateway

© PRODYNA

But when to use Serverless?

Serverless good for:

- Unknown workloads
- Can't (or don't want to) spend resources to continuously optimize sizes
- Small projects or Proof of Concepts (easy to create & destroy)

Rule of thumb:

The more you know your application workloads the less viable is Serverless

© PRODYNA

Questions: Serverless

© PRODYNA

Questions: Serverless

If you have a set of frequently accessed files that are used on a daily basis, what S3 storage class should you store them in?

A. Fast Access

B. Reduced Redundancy

C. Standard

D. Infrequent Access

© PRODYNA

Questions: Serverless

If you have a set of frequently accessed files that are used on a daily basis, what S3 storage class should you store them in?

A. Fast Access

B. Reduced Redundancy

C. Standard

D. Infrequent Access

Answer: C Standard

© PRODYNA

Questions: Serverless

What is the availability and durability rating of S3 Standard Storage Class?

A. 99.999999999% Durability and 99.99% Availability

B. 99.999999999% Availability and 99.90% Durability

C. 99.999999999% Durability and 99.00% Availability

D. 99.999999999% Availability and 99.99% Durability

© PRODYNA

Questions: Serverless

What is the availability and durability rating of S3 Standard Storage Class?

A. 99.999999999% Durability and 99.99% Availability

B. 99.999999999% Availability and 99.90% Durability

C. 99.999999999% Durability and 99.00% Availability

D. 99.999999999% Availability and 99.99% Durability

Answer: A. 99.999999999% Durability and 99.99% Availability

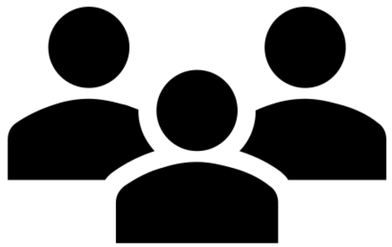
S3 Standard Storage class has a rating of 99.999999999% durability (referred to as 11 nines) and 99.99% availability.

© PRODYNA

Practice: Serverless

© PRODYNA

Practice 2: Serverless – Trump Quoter



Git: https://github.com/DennisCreutz/aws-training/tree/task/serverless

1. Checkout code for the task
2. Create a Lambda function "trump-quoter" with the code from the /app directory
   => Don't forget to use npm install,
   => Use nodejs14.x runtime
3. Create a REST API Gateway with a GET method that calls the Lambda function over Lambda proxy request
   => You need to enable CORS!
   => Please use regional endpoints
4. Create a S3 webhosting bucket with the HTML from the /html directory
   => Bucket name: *yourTeamName-currentDateTime*
   => Enable public access
5. Enter the API Gateway URL and a query (e.g. "Obama") in the input fields and hit "TRUMP QUOTE"

© PRODYNA

DevOps

© PRODYNA

PRODYNA

Terraform

Training

AWS CloudFormation

- **Category:** Management & Governance
- "AWS CloudFormation provides a common language for you to model and provision AWS and third party application resources in your cloud environment."
- **Pricing Model:** Free
- **Free tier**: Always free

© PRODYNA

CloudFormation vs Terraform

	CloudFormation	Terraform
Language	YAML or JSON	HCL or JSON
Provider	AWS	Many
Licence	AWS Service	Open Source
Enterprise Support	Yes	Yes
Remote State Management	Out of the box	Needs to be configured
Modularization	Nested Stacks	Self contained packages
Rolling Updates	Supported	Not Supported
Import Resources	Not Supported	Supported
Remote execution	Yes	Only with Terraform Cloud or in a build pipeline

© PRODYNA

AWS Serverless Application Model (AWS SAM)

The AWS Serverless Application Model (AWS SAM) is an open-source framework that helps you to build serverless applications on AWS.

Components:

- AWS SAM template specification
  - Syntax to describe functions, APIs, permissions, configurations, and events
- AWS SAM command line interface
  - verify SAM template files
  - invoke Lambda functions locally
  - step-through debug Lambda functions
  - package and deploy serverless applications to the AWS Cloud
  - and more...

© PRODYNA

Demo: CloudFormation

© PRODYNA

AWS CodeCommit

- **Category:** Developer Tools
- **"AWS CodeCommit is a fully-managed source control service that hosts secure Git-based repositories."**
- **Pricing Model:** Per active user
- **Free tier**: First 5 active users are always free

© PRODYNA

AWS CodeBuild

- **Category:** Developer Tools
- "AWS CodeBuild is a fully managed continuous integration service that compiles source code, runs tests, and produces software packages that are ready to deploy."
- **Pricing Model:** Per build minute
- **Free tier**: 100 build minutes general1.small are always free

© PRODYNA

AWS CodeBuild

- Pay per minute
- Fully managed
- Elastic
- Custom build environments through Docker images
- Uses buildspec YAML file for building process

© PRODYNA

AWS CodeDeploy

- **Category:** Developer Tools
- "AWS CodeDeploy is a fully managed deployment service that automates software deployments to a variety of compute services as Amazon EC2, AWS Fargate, AWS Lambda, and your on-premises servers."
- **Pricing Model:** Per deployment
- **Free tier**: Deployment to EC2 and Lambda is always free

© PRODYNA

AWS CodeDeploy

- Fully managed deployment & update
- Rollback on failure
- Canary deployment supported
  - Possible: Alarm in X minutes = rollback last version
- Blue/Green deployment supported

© PRODYNA

AWS CodePipeline

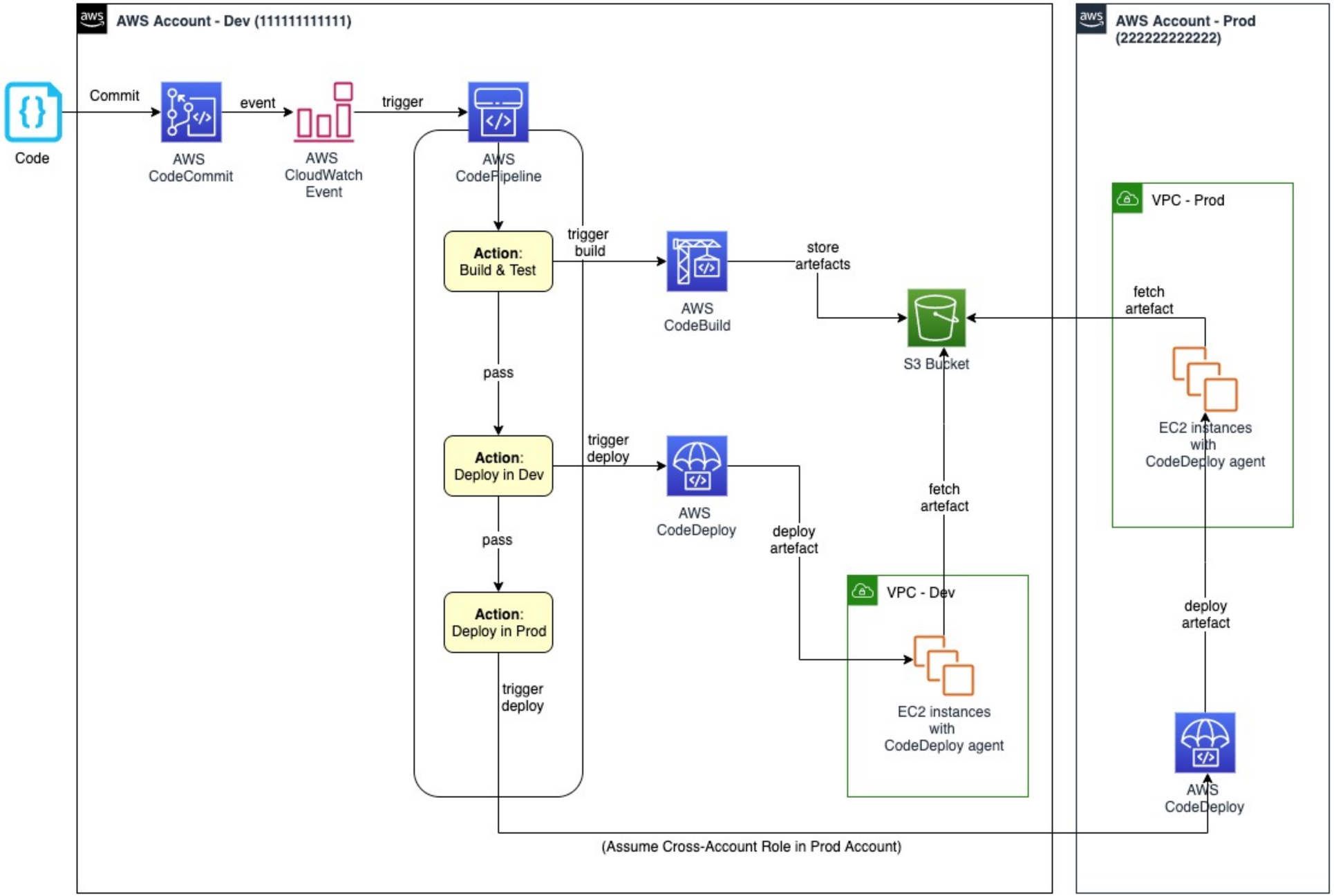
- **Category:** Developer Tools
- "AWS CodePipeline is a fully managed continuous delivery service that helps you automate your release pipelines for fast and reliable application and infrastructure updates."
- **Pricing Model:** Per active pipeline (after 30 days)
- **Free tier**: 1 active pipeline for 12 Month

© PRODYNA

AWS CodePipeline

- Supported Sources:
  - AWS CodeCommit
  - GitHub
  - Amazon S3
  - Amazon ECR
- Build Trigger:
  - (CloudWatch) Schedule
  - On Commit on Branch
  - Webhooks (e.g. Dockerhub)

© PRODYNA



https://aws.amazon.com/blogs/devops/complete-ci-cd-with-aws-codecommit-aws-codebuild-aws-codedeploy-and-aws-codepipeline/

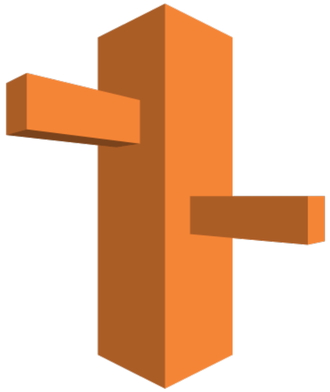
© PRODYNA

Advanced Web Services

© PRODYNA

Amazon Route 53

- **Category: Net**working & Content Delivery
- "You can use Amazon Route 53 to register new domains, transfer existing domains, route traffic for your domains to your AWS and external resources, and monitor the health of your resources."
- **Pricing Model:** Complex
- **Free tier**: No



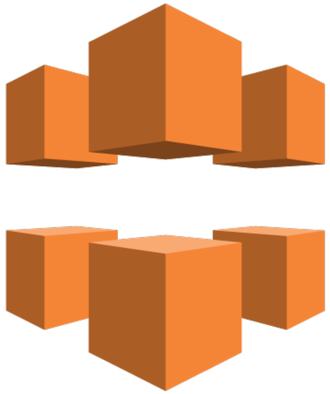
© PRODYNA

Route 53

- Register Domain
- Hosted Zone
- Traffic Flow
- Resolver (e.g. route DNS queries to VPC)

© PRODYNA

Amazon CloudFront



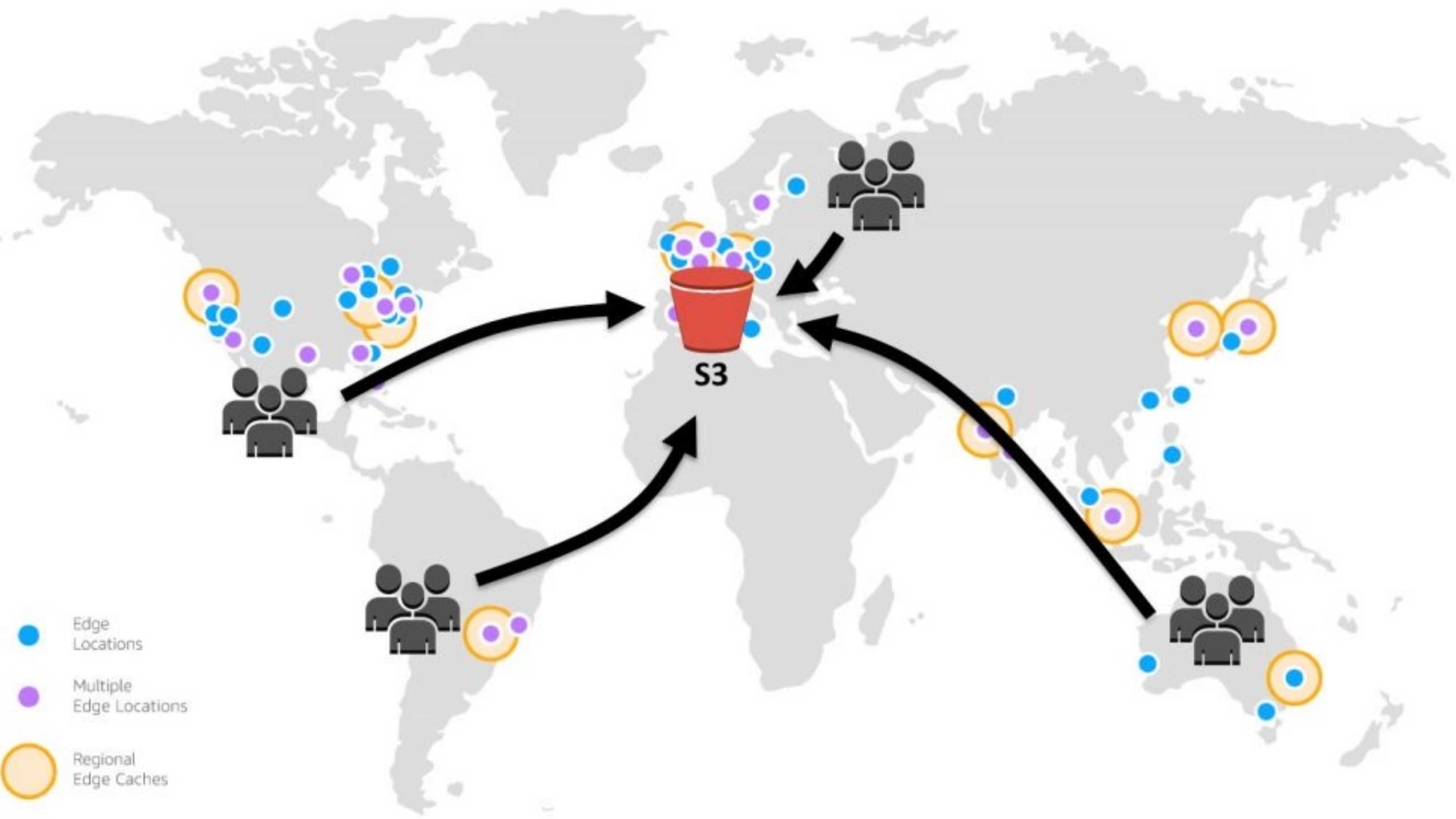
- **Category: Netw**orking & Content Delivery
- "Amazon CloudFront is a fast content delivery network (CDN) service that securely delivers data, videos, applications, and APIs to customers globally with low latency, high transfer speeds, all within a developer-friendly environment."
- **Pricing Model:** Data Transfer Out / Requests
- **Free tier**: 50 GB Transfer Out / 2 Mil. HTTP & HTTPS Requests monthly for 12 month

© PRODYNA

Amazon CloudFront Advantages

- Cache content in edge locations = Quick Content Delivery
- Simple to use
- Content Privacy
- Geo Targeting
- Not very expensive
- Deep integration with other AWS services

© PRODYNA


S3
Edge Locations
Multiple Edge Locations
Regional Edge Caches

Amazon Cognito

- **Category: Sec**urity, Identity, & Compliance
- "Amazon Cognito lets you add user sign-up, sign-in, and access control to your web and mobile apps quickly and easily.
- **Pricing Model:** Pay based on your monthly active users (MAUs)
- **Free tier**: 50,000 MAUs and more are always free

© PRODYNA

Amazon Cognito

- Secure and scalable identity store
  - Cognito User Pool
- Social and enterprise identity federation
  - Sign in through social identity providers such as Apple, Google, Facebook, and Amazon
- Standards-based authentication
  - OAuth 2.0, SAML 2.0, and OpenID Connect
- Security for your apps and user
  - Supports multi-factor authentication and encryption of data-at-rest and in-transit
- Access control for AWS resources
  - Cognito Identity Pools
- Easy integration with your app
  - Easy to configure

© PRODYNA

Amazon Cognito – User Pool vs Identity Pool

- User Pool
  - User management but no authentication
  - Oauth / OpenID Connect Flow:
    - Cognito passes the user's ID and group membership to the app over a JWT
    - App (you) decides on permissions
- Identity Pool
  - No user management just authentication
  - Permissions:
    - Cognito assigns IAM permissions (role) to user
- You can use both together:
  - User pool to manage users to get IAM permissions from the identity pool
- Easy integration with AWS Amplify

© PRODYNA

AWS Amplify

- **Category:** Mobile
- "AWS Amplify is a development platform for building secure, scalable mobile and web applications."
- **Pricing Model:** Build minutes & Hosting space
- **Free tier**: 1000 build minutes per month, 5 GB stored per month, 15 GB served per month for 12 month

© PRODYNA

AWS Amplify

- Develop
  - Install Amplify CLI
  - Setup app. Services
  - Integrate with your app
  - Support for React, Angular, Vue.js, IOS, Android,...
- Deploy
  - Connect your repo.
  - Configure build settings
  - Deploy your app.

© PRODYNA

Demo: Cognito & Amplify

© PRODYNA

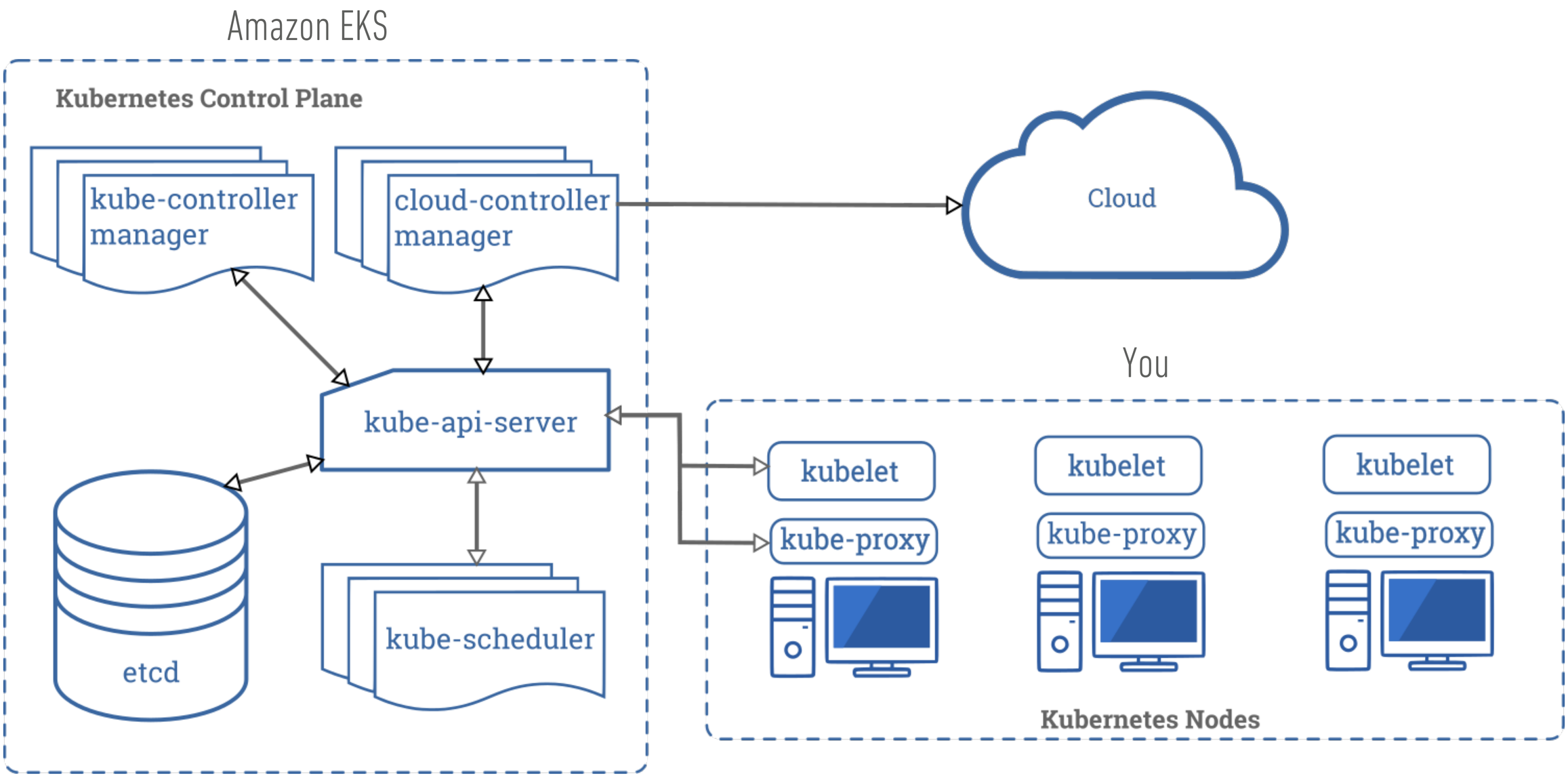
Containerization

© PRODYNA

Amazon Elastic Container Service for Kubernetes (EKS)

- **Category:** Compute
- "Amazon Elastic Container Service for Kubernetes (Amazon EKS) makes it easy to deploy, manage, and scale containerized applications using Kubernetes on AWS."
- **Pricing Model:** Per hour for each Amazon EKS cluster
- **Free tier**: No

© PRODYNA


Amazon EKS
Kubernetes Control Plane
kube-controller manager
cloud-controller manager
Cloud
kube-api-server
etcd
kube-scheduler
You
kubelet
kube-proxy
kubelet
kube-proxy
kubelet
kube-proxy
Kubernetes Nodes

© PRODYNA

Amazon EKS

- You get the control plane from EKS but you need to take care of the worker nodes
- Worker Nodes = Amazon EC2 Instances
- 3 different methods:
  - Self-Managed node groups
  - Managed node groups
  - AWS Fargate

© PRODYNA

Amazon EKS – Self-Managed Node Groups

Configure, manage and create everything yourself

- Provision EC2 instances
- Auto Scaling Groups
- Register as worker nodes to EKS
- Kubernetes API access on the node
- Security (groups)
- Update components (without downtime)

© PRODYNA

Amazon EKS – Self-Managed Node Groups

You can customize everything according to your needs but at the cost of high administrative efforts.

High control
High administration effort

© PRODYNA

Amazon EKS – Managed Node Groups

Automate provisioning and lifecycle management of nodes:

- Creating Autoscaling Group
  - Still need something like Kubernetes Cluster Autoscaler
- Created with latest EKS optimized AMI
  - Easy (manually) triggered AMI update
- Gracefully rotate nodes to update the underlying AMI
- Gracefully draining nodes before termination during a scale down event
- Apply labels to the resulting Kubernetes Node resources

© PRODYNA

Amazon EKS – Managed Node Groups

Limitations:

- Needs Kubernetes >= 1.14
- ~~Only works with Amazons EKS optimized AMI (Amazon Linux OS)~~
- Limited control over the security group rules
  - No option to restrict on CIDR blocks level

Medium control

Medium administration effort

© PRODYNA

Amazon EKS – Join Worker Nodes

To join worker nodes a special ConfigMap needs to be applied:

```yaml
apiVersion: v1
kind: ConfigMap
metadata:
  name: aws-auth
  namespace: kube-system
data:
  mapRoles: |
    - rolearn: <ARN of instance role (not instance profile)>
      username: system:node:{{EC2PrivateDNSName}}
      groups:
        - system:bootstrappers
        - system:nodes
```

Attention: With self managed worker nodes you need to apply this config map yourself!

© PRODYNA

Amazon EKS – Kubernetes RBAC

- When a cluster is created only the user who created the cluster can access it
- You can map IAM users & roles to Kubernetes users (RBAC) over the „aws-auth“ config map, e.g.:

```
mapRoles: |
  - rolearn: arn:aws:iam::XXXXXXX:role/myRole # Role ARN without the path!
    username: user-iam-role
    groups:
      - system:masters
mapUsers: |
  - userarn: arn:aws:iam:: XXXXXXX :user/admin
    username: user-iam-user
    groups:
      - system:masters
```

© PRODYNA

Amazon EKS – Tags

- Required Tag for EKS and Kubernetes to discover and manage networking resources:
  - Tag worker nodes: “**kubernetes.io/cluster/${ClusterName}” : “owned”**
  - Attention: With self managed worker nodes you need to apply this tag yourself!
- Require subnet Tag for internal and external Load Balancer:
  - **1.18 and earlier only**: Allow clusters to use this subnet for LB: **“kubernetes.io/cluster/${ClusterName}” : “shared”**
  - Internal (Private Subnets) LB: **"kubernetes.io/role/internal-elb“: “1“**
  - External (Public Subnets) LB: **"kubernetes.io/role/elb“: “1“**

© PRODYNA

Amazon EKS – AWS Fargate

"AWS Fargate is a serverless compute engine managed by AWS to run container workloads without actively managing servers to run them"

- Just define the container to and AWS will (will figure out how to) run it
- Serverless approach
- No configuration of instance types, autoscaling (rules), OS, patching etc.

© PRODYNA

Amazon EKS – AWS Fargate

Limitations:

- “DaemonSet” are not supported
  - Need to use e.g. side cars
- Privileged containers aren't supported
- Pods can need longer for startup
- No SSH access to Nodes

More informations: https://docs.aws.amazon.com/eks/latest/userguide/fargate.html

Low control

Low administration effort

© PRODYNA

Demo: Amazon EKS

© PRODYNA

Amazon Elastic Container Service (ECS)

- **Category:** Compute
- "Amazon Elastic Container Service (Amazon ECS) is a fully managed container orchestration service."
- **Pricing Model:** Free
- **Free tier**: No

© PRODYNA

Amazon ECS vs Kubernetes

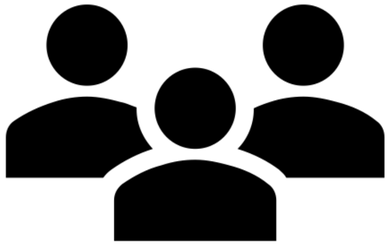
- AWS native
- No control plane fee
- Easy deployment of pods (ECS term: task)
- Supported instance types:
  - EC2
  - Fargate
- Support for IAM roles for tasks
- Bottom line:
  - Need simplicity? Choose ECS
  - Need advanced orchestration? Choose EKS

© PRODYNA

Practice: Containerisation

© PRODYNA

Practice: Amazon EKS I



Time: 90 min

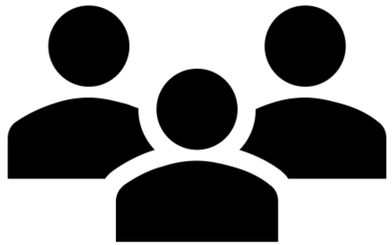
Create a Amazon EKS cluster:

1. You need a VPC with 3 private, 3 public subnets and a NAT
2. Create the EKS cluster
    1. Create cluster role if not already created: https://docs.aws.amazon.com/eks/latest/userguide/service_IAM_role.html#create-service-role
    2. Create a security group
    3. Create the EKS cluster – this will need ~10 minutes
    4. Create Node role https://docs.aws.amazon.com/eks/latest/userguide/create-node-role.html#create-worker-node-role

© PRODYNA

Practice: Amazon EKS I

3. Wait for your cluster and worker nodes to come up
4. Verify that the cluster creator can connect to the cluster:
   a) aws eks --region eu-central-1 update-kubeconfig --name ${clusterName} --profile myAWSProfile
   b) kubectl get nodes



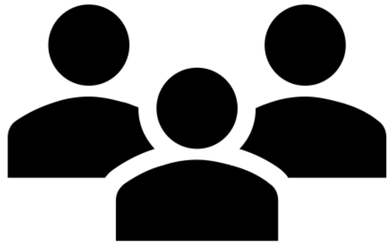
Time: 90 min

© PRODYNA

Practice: Amazon EKS I

5. Add all team member to the cluster's ConfigMap:
   - You can enable access for other user over a IAM role
   - Search in you account in IAM – Roles for a role starting with "AWSReservedSSO_AWSAdministratorAccess" and copy the role ARN without the „ aws-reserved/sso.amazonaws.com/{myRegion}/" part
   - Add the role ARN to the Kubernetes ConfigMap, e.g.:

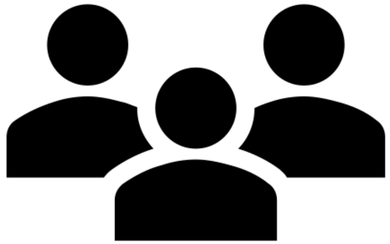
```
mapRoles: |
  - rolearn: myRoleARN
    username: user-over-iam-role
    groups:
      - system:masters
```



Time: 90 min

© PRODYNA

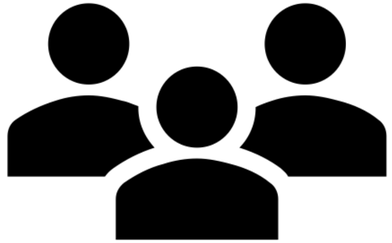
Practice: Amazon EKS II



1. To enable cluster autoscaler AWS EKS need the metric server and the Kubernetes autoscaler
2. Use the commands in the “deployments/metric-server” and “deployments/cluster-autoscaler” directory and apply the “autoscaling.tf”
3. Wait for the metrics to get propagated: kubectl top nodes

© PRODYNA

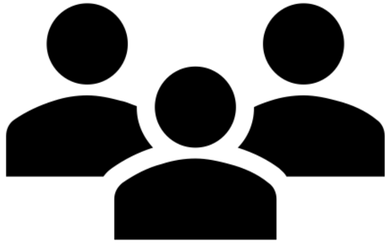
Practice: Amazon EKS II



4. Get current number of worker nodes:
   kubectl top nodes
5. Set the cluster on fire:
   kubectl create deployment autoscaler-demo --image=nginx
   kubectl scale deployment autoscaler-demo --replicas=40
6. Wait and check nodes again:
   kubectl top nodes
   More nodes should be up now
7. Scale down:
   kubectl scale deployment autoscaler-demo --replicas=1

© PRODYNA

Practice: Amazon EKS III

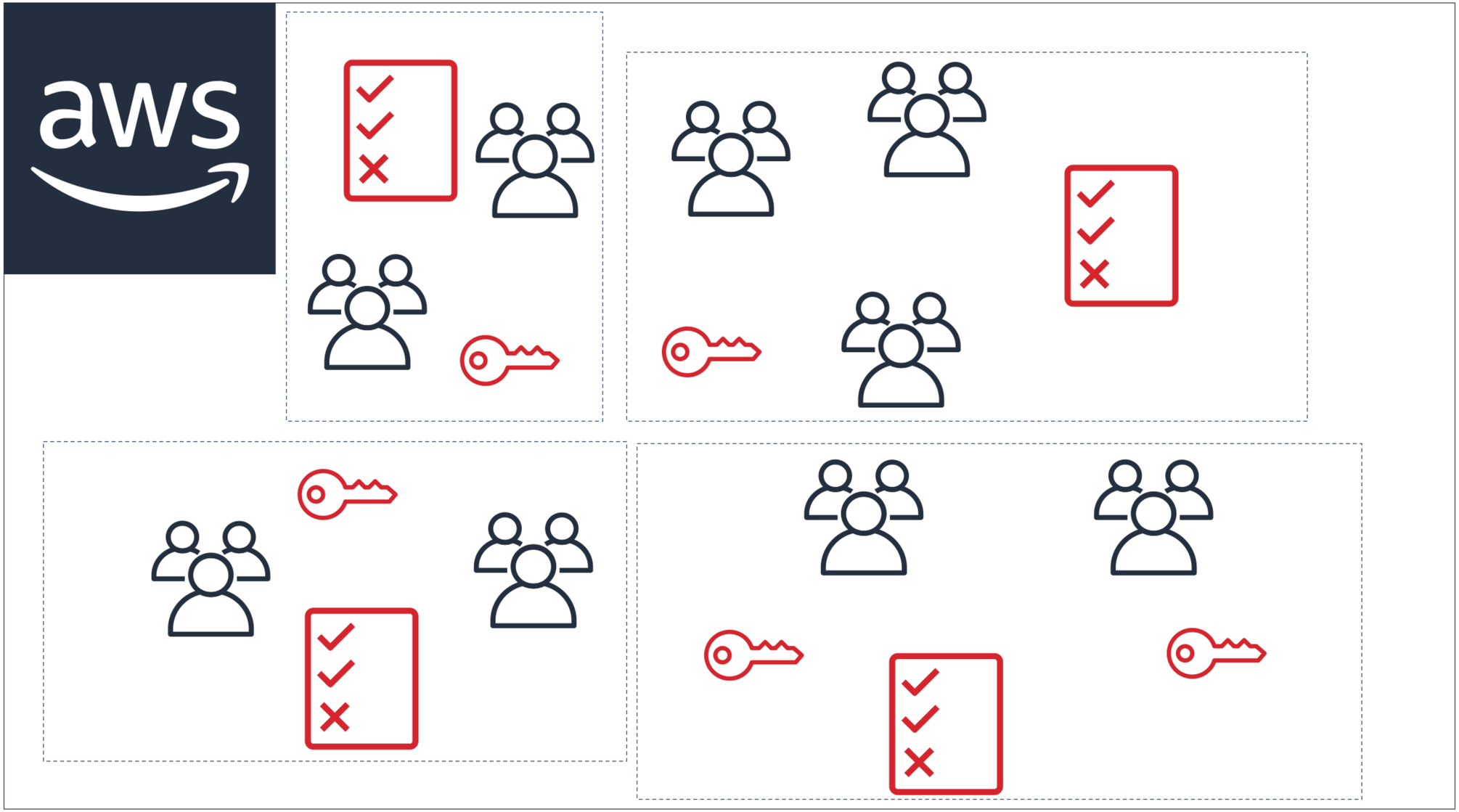


1. Deploy the example deployment and the load balancer service in the „deployments/lb“ directory
2. Wait for the deployment and service to come up and check if a Load Balancer was created in AWS („EC2“->“Load Balancers“)
3. Copy the DNS of the Load Balancer and try to connect, you should see a Hello World message

© PRODYNA

Multi Account Structure

© PRODYNA



© PRODYNA

Problem

How to manage a scalable and secure AWS environment?

© PRODYNA

Solution?

One account with tags & policies?

- Complex and hard to maintain
- Shared resources (e.g. account limits)
- Difficult to separate responsibilities accurately

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:StartInstances",
                "ec2:StopInstances"
            ],
            "Resource": "arn:aws:ec2:*:*:instance/*",
            "Condition": {
                "StringEquals": {"aws:ResourceTag/Owner": "${aws:username}"}
            }
        },
        {
            "Effect": "Allow",
            "Action": "ec2:DescribeInstances",
            "Resource": "*"
        }
    ]
}
```

© PRODYNA

Solution

Multi-account AWS environment

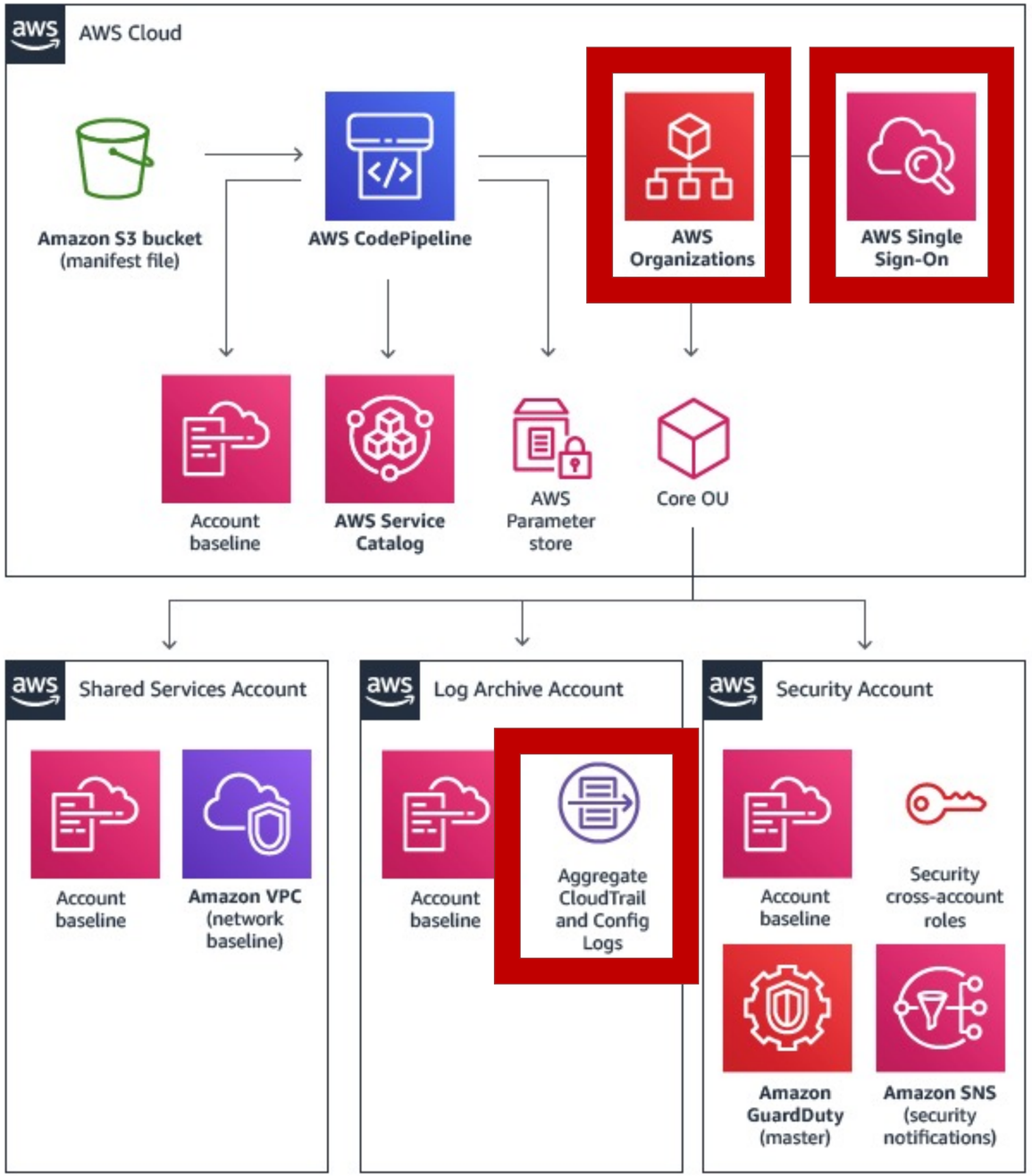
The Landing Zone principal

© PRODYNA

Landing Zone

- Multi-account AWS environment
- Separate resources with AWS accounts
- Every account got a very specific use case
- 4 Core Accounts
  - Master
  - Shared Service
  - Log Archive
  - Security
- Each project and stage is a separate account (e.g. “projectXY-dev”, “projectXY-prod”, “projectA-dev”, etc.)

© PRODYNA


AWS Cloud
Amazon S3 bucket
(manifest file)
AWS CodePipeline
AWS
Organizations
AWS Single
Sign-On
Account
baseline
AWS Service
Catalog
AWS
Parameter
store
Core OU
Shared Services Account
Account
baseline
Amazon VPC
(network
baseline)
Log Archive Account
Account
baseline
Aggregate
CloudTrail
and Config
Logs
Security Account
Account
baseline
Security
cross-account
roles
Amazon
GuardDuty
(master)
Amazon SNS
(security
notifications)

© PRODYNA

Good but setup is very tedious...

© PRODYNA

AWS Control Tower

- **Category:** Management & Governance
- "The easiest way to set up and govern a secure multi-account AWS environment"
- **Pricing Model:** Free
- **Free tier**: Always free

© PRODYNA

AWS Control Tower

- New service (since 06/2019)
- Fully automated creation of the multi-account structure
- Automate ongoing policy management
- View policy-level summaries of your AWS environment
- Offical AWS support

© PRODYNA

DEMO – PRODYNA Account Structure

© PRODYNA

Miscellaneous

© PRODYNA

Amazon Simple Notification Service (SNS)

- **Category:** Application Integration
- "Amazon SNS provides topics for high-throughput, push-based, many-to-many messaging (...)"
- **Pricing Model:** Number of publish / deliver notifications
- **Free tier**: 1,000,000 Requests, 100,000 HTTP notifications and 1,000 email notifications are always free

© PRODYNA

Amazon ElastiCache

- **Category:** Database
- "Amazon ElastiCache offers fully managed Redis and Memcached. Seamlessly deploy, run, and scale popular open source compatible in-memory data stores."
- **Pricing Model:** Hour
- **Free tier**: 750 hours for 12 month

© PRODYNA

Amazon Simple Queue Service (SQS)

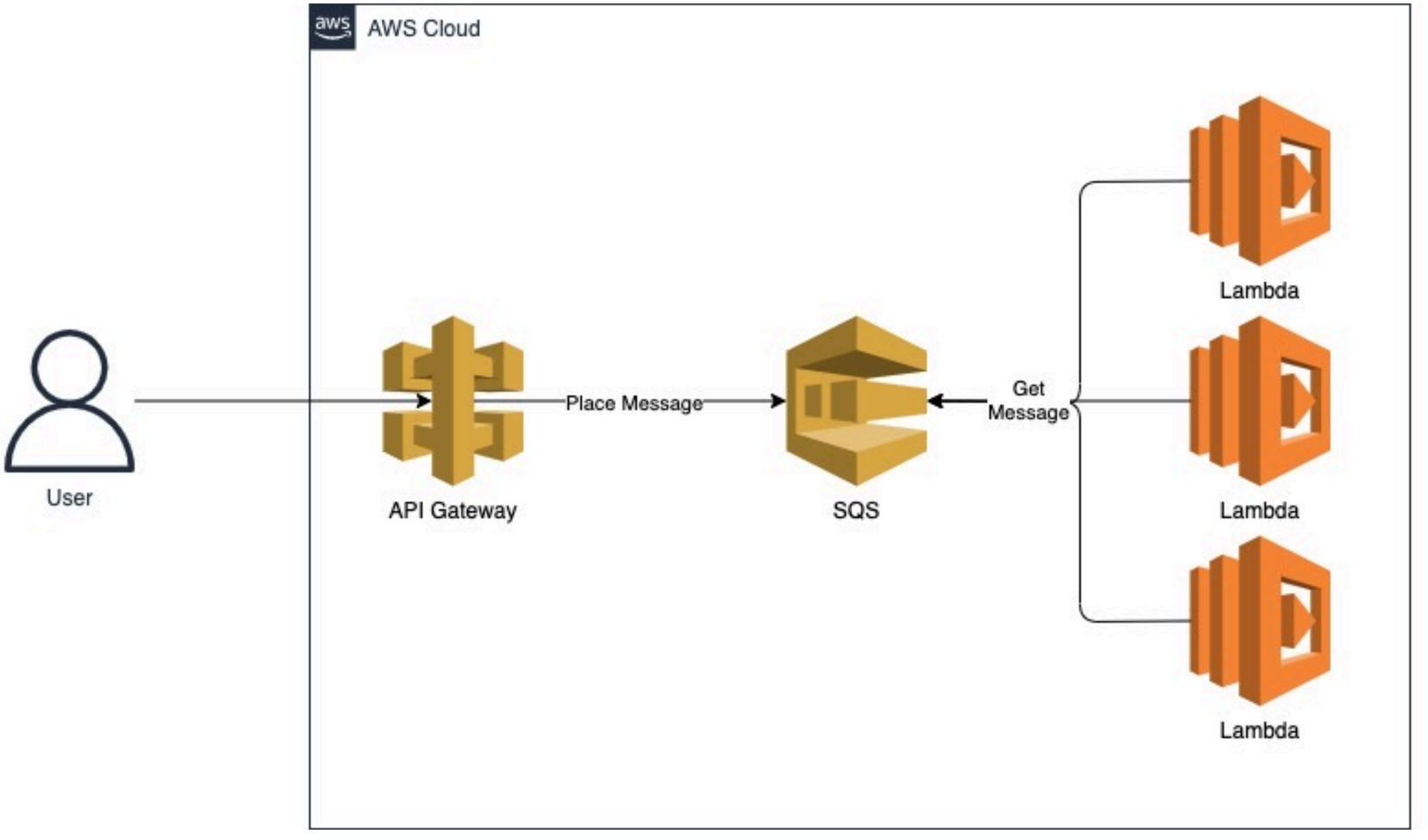
- **Category:** Application Integration
- "Amazon Simple Queue Service (SQS) is a fully managed message queuing service that enables you to decouple and scale microservices, distributed systems, and serverless applications."
- **Pricing Model:** Request
- **Free tier**: 1 Mil. requests are always free

© PRODYNA

Amazon SQS Queue Types

- Standard Queue
  - Nearly unlimited transactions per second
  - Best effort ordering: Order can be different
  - A message is delivered at least once (maybe more!)
  - $0.40 for 1 mil. requests
- FIFO Queue
  - Max. 300 transactions per second
  - First in first out
  - Delivers a message exactly once
  - $0.50 for 1 mil. requests

© PRODYNA


AWS Cloud
User
API Gateway
Place Message
SQS
Get
Message
Lambda
Lambda
Lambda

Any services you want to know about?

© PRODYNA

Last words

© PRODYNA

Professional

Two years of comprehensive experience designing, operating, and troubleshooting solutions using the AWS Cloud

Associate

One year of experience solving problems and implementing solutions using the AWS Cloud

Foundational

Six months of fundamental AWS Cloud and industry knowledge

Architect

- AWS certified Solutions Architect Professional
- AWS certified Solutions Architect Associate

Operations

- AWS certified DevOps Engineer Professional
- AWS certified SysOps Administrator Associate

Developer

- AWS certified Developer Associate

Cloud Practitioner

- AWS certified Cloud Practitioner

Specialty

Technical AWS Cloud experience in the Specialty domain as specified in the **exam guide**

- AWS certified Advanced Networking Specialty
- AWS certified Security Specialty
- AWS certified Machine Learning Specialty
- AWS certified Alexa Skill Builder Specialty
- AWS certified Data Analytics Specialty
- AWS certified Database Specialty

PRODYNA

ANY QUESTIONS?