Welcome

EIC EDGE – Week 3

CI/CD and the Cloud

Why study these subjects?

In modern software engineering, our ability to quickly deploy incremental innovation, ensure its quality, and scale to meet customer demand proves critical to our success

- Cloud is everywhere and it’s not going away
- As with many topics in technology, there are multiple options and multiple dimensions to those options
- Building a deeper understanding of Cloud and its offerings helps prepare you for modern IT
- Included in that is the importance of learning about the build and operation of CI/CD pipelines using Cloud native technologies

My pledge to you

I will...

- Make this interactive
- Ask you questions
- Ensure everyone can speak
- Use an on-screen timer

Objectives

At the end of this course, you will be able to:

- Speak to available Cloud native capabilities utilized to build out complex infrastructure environments for hosting and execution of our workloads
- Demonstrate the ability to create simple Cloud resources in a modern Cloud Service Provider (CSP)
- Use Cloud native technologies to build and operate simple CI/CD pipelines

Agenda

- **Day One**
- Hosting Applications in the Cloud

- **Day Two**
- Cloud Native Options in AWS, Part 1

- **Day Three**
- Cloud Native Options in AWS, Part 2

Agenda

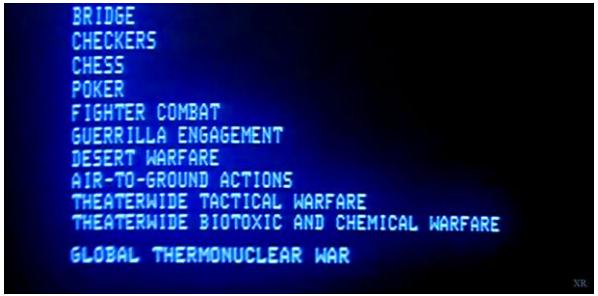
- **Day Four**
- AWS CodeBuild

- **Day Five**
- AWS CodePipeline

How we’re going to work together

- Lab work will take place in individual workspaces – isolated, browser-based virtual machines
- Classroom labs may include individual work as well as group discussion and work
- I welcome being interrupted – if you need more info, or clarification, or anything else, just break in and ask. I am here to help you.

Would you like to play a game?



- To make it interesting, let's add a competition factor to our time together ☺
- For questions posed to the larger group, 1 point will be awarded for each answer provided by a participant
- 2 points will be awarded for acting as a spokesperson for a breakout room during one of our group discussions
- There will be a rubric provided for the capstone (targeted for week 4) which will define a points breakdown for the implemented solution

Open Discussion

What is “the Cloud”?

How does “the Cloud” factor into modern IT?

Hosting Applications in the Cloud

Infrastructure Options

Infrastructure Options

Infrastructure is the hardware & software that run our IT workloads and that provide our business users and customers a way to interface with the applications required to complete their daily jobs

What Are the Options?

On-Premise (in a Data Center)

Public Cloud

At the Edge

Hybrid Cloud

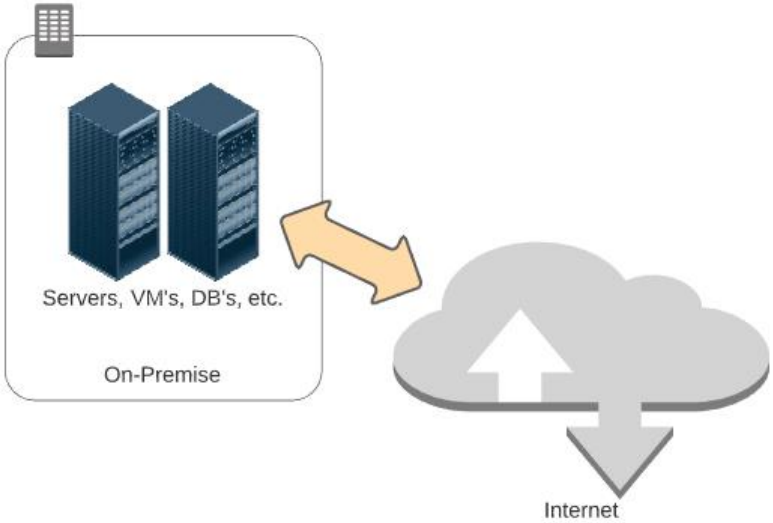
What do they all mean?

On-Premise

Can mean a few different things:

- In a wholly-owned Data Center
- In a COLO (or co-location Data Center)
- Sometimes called a “private cloud”

On-Premise



On-Premise

Why and What?

- How infrastructure has traditionally been done
- With this model, companies try and estimate current & future hardware capacity needed to support business operations

On-Premise

Why and What?

- Stakeholders plan out expected levels of consumption for the next 3 – 5 years (capacity to handle current volumes as well as expected growth)
- Some critical workloads may not be suitable for anything but a physical and directly-managed implementation (e.g., mainframe)

On-Premise – Discussion

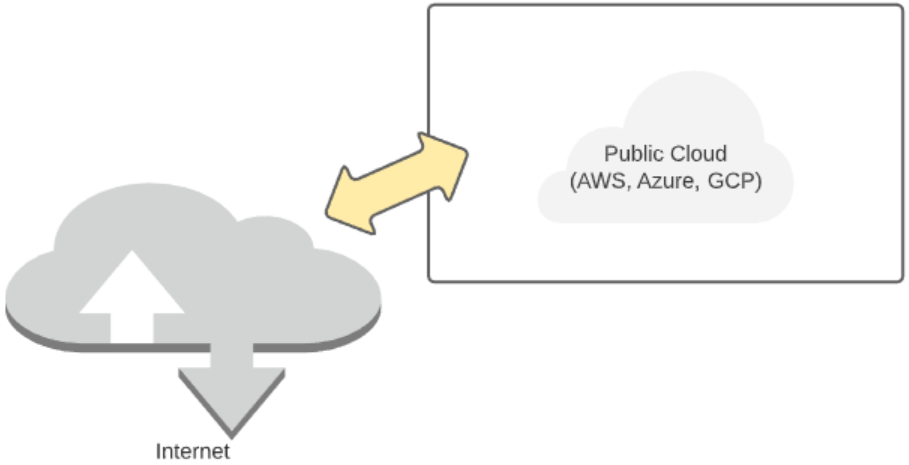
Pros?

- Discrete capacity planning (even if that planning was off)
- Some workloads (e.g., mainframes and certain legacy systems) are tailor-made for a physical data center
- With a move to COLO's, companies could begin to share expenditure

Cons?

- Sometimes difficult to know what is needed and when it is needed – if the plan was off (or unexpected spikes in demand occurred), difficult to adjust quickly
- Some workloads are just as effective (if not more so) in a virtual vs. physical implementation
- Harder to control costs and plan for costs – CAPEX vs. OPEX

Public Cloud



Public Cloud

Why and What?

- Platform using the standard “Cloud computing model” to provide infrastructure and application services
- Accessed and integrated via the Internet
- May provide a few different types of services – IaaS, PaaS, etc.

Public Cloud

Why and What?

- Usually supports a subscription or “pay as you go” (on-demand) pricing model
- Largest players in this space include Azure, AWS and GCP

Public Cloud - Discussion

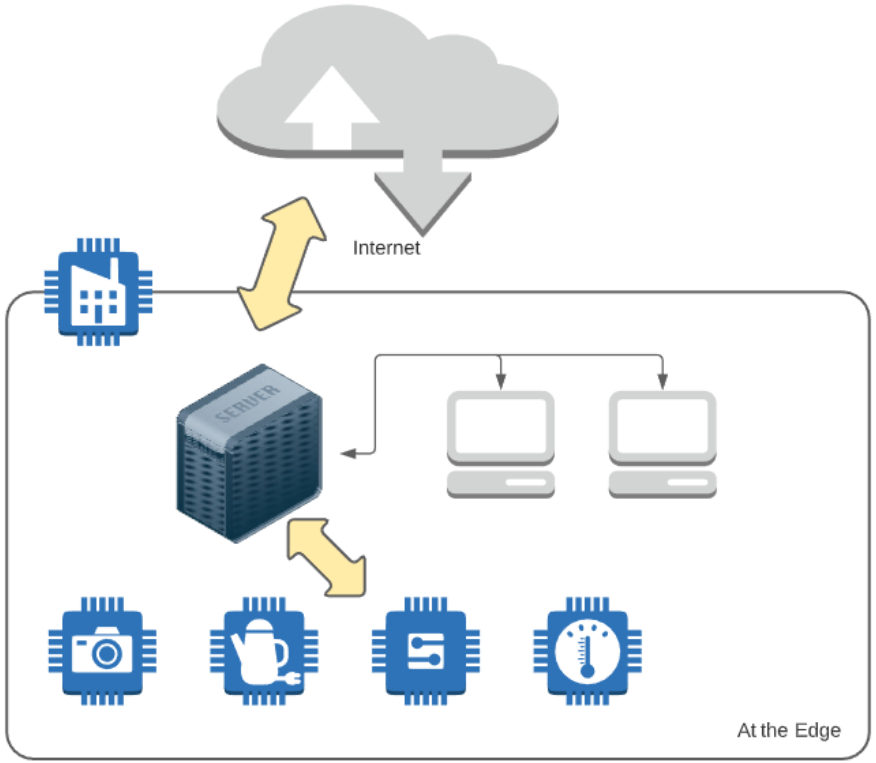
Pros?

- Flexibility and elasticity in capacity planning – enables automated schedule-based or metrics-based adjustments to capacity when required
- In some cases, managed services can be leveraged reducing operations overhead
- Because services are PAYG (pay as you go), you're only charged for what you use, and those expenses are OPEX

Cons?

- Requires enough historical data for schedule-based planning or the right configuration for metrics-based planning
- With managed services you lose some levels of granular control
- Because of the flexibility/elasticity, it can be difficult to budget and, if Cloud services are not managed/monitored, costs can be high

At the Edge



At the Edge

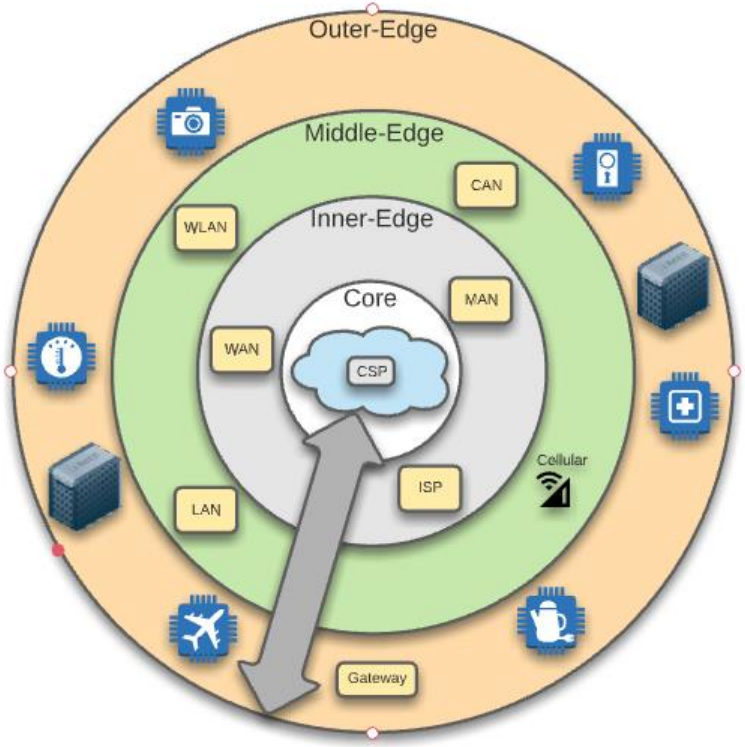
Can include 3 distinct layers:

Inner or Near Edge

Middle Edge

Outer or Far Edge

At the Edge – Layers



CSP – Cloud Service Provider
WAN – Wide Area Network
ISP – Internet Service Provider
MAN – Metropolitan Area Network
LAN – Local Area Network
WLAN – Wireless Local Area Network
CAN – Campus Area Network

At the Edge

Why and What?

- It’s about bringing the power of Cloud computing to you
- Enables additional processing closer to the sources of data while still supporting the offload of higher order processing to the Cloud
- Often involves setting up “Cloud-in-a-box” facilities on-premise

At the Edge

Why and What?

- IoT (Internet of Things) is a good example – devices in a facility reading massive amounts of data can incorporate processing at the edge to improve overall efficiency
- Helps inject lower latency, increased security and improved bandwidth into systems used to aggregate critical data for an enterprise

At the Edge - Discussion

Pros?

- Allows distribution of processing power across a larger surface area
- Can be used to bring critical latency, security and bandwidth improvements to specific types of business workflows
- Efficiencies gained “at the edge” can help with managing the cost of processing data

Cons?

- Requires more infrastructure and more configuration to support that distribution
- Increased distribution of processing power and activity can expand attack surface and requires the right configuration to ensure optimal interaction between system components (i.e., increased complexity)
- More components “at the edge” can lead to increased infrastructure costs

Hybrid Cloud

Why and What?

- In many ways, an amalgamation of the other options
- Supports distribution of system processing across on-premise infrastructure and the public Cloud

Hybrid Cloud

Why and What?

- Allows an enterprise to keep workloads that are best-suited for on-premise running on-premise while allowing migration of components that can move to the public Cloud
- Can help make an enterprise’s move to the Cloud more gradual and planful

Hybrid Cloud - Discussion

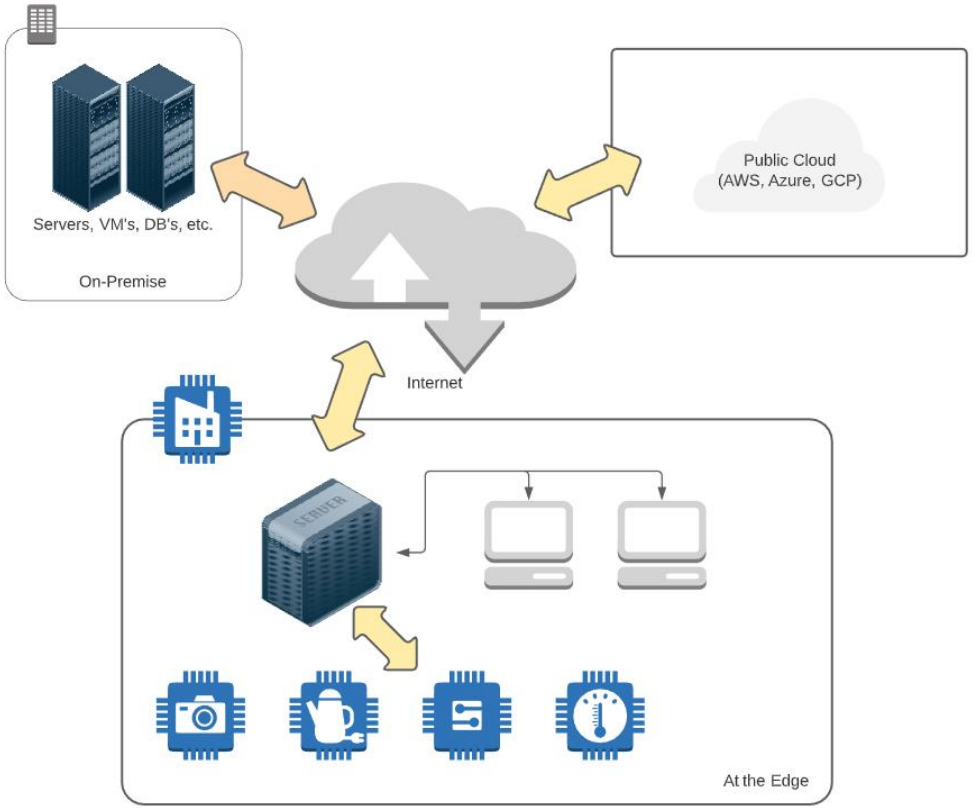
Pros?

- Allows distribution of processing power across a larger surface area
- Can allow a move to the Cloud to be more gradual and allow an enterprise to target optimal deployment platform while making the move
- The ability to support a gradual move enables an enterprise to assess and understand Cloud costs over time

Cons?

- Requires more infrastructure and more configuration to support that distribution
- As with Edge, can lead to increased complexity, often including required setup and maintenance of dedicated, secure connectivity between a data center and the Cloud
- If not managed optimally, costs can be higher due to need to pay for Cloud usage and data center (CAPEX + OPEX)

Hybrid Cloud



LAB:

Infrastructure Options

Scenario: Your company (a global leader in FinTech) is currently hosting all infrastructure used to power the business in an on-premise Data Center. This includes a mainframe for primary business functions (customer management, account management, accounts payable, accounts receivable), several Web Apps (for customer interaction), several Web APIs providing backend data and functionality to the UIs, and a system used to manage data feeds from several security cameras used at corporate offices for observation and security.

As a member of the technical staff, you have been asked to provide thoughts and recommendations on moving from the Data Center to the Cloud.

In your assigned breakout room, discuss as a group and be prepared to provide the following: 1) Potential options for infrastructure in the Cloud for the different types of workload, and 2) considerations that the company should keep in mind as they make the move to ensure awareness and proactive planning.

Nominate someone (or volunteer) to share your group's ideas.

Knowledge Check

With this infrastructure option, stakeholders try to estimate hardware & software needs for the next 3 – 5 years:

A. Public Cloud
B. At the Edge
C. Hybrid
D. On-Premise

Knowledge Check

This infrastructure option focuses on bringing additional power closer to the data sources and application users:

A. Public Cloud
B. At the Edge
C. Hybrid
D. On-Premise

Knowledge Check

This infrastructure option seeks to build a solution by combining the other options into a larger solution:

A. Public Cloud
B. At the Edge
C. Hybrid
D. On-Premise

Application Hosting

Application Hosting

By Application Hosting, we mean the target infrastructure and runtime platform used for deployment and execution of an application or system; can include compute (CPU and server resources), storage, network, data and operating system

Application Hosting – An “Interesting” Example?

Here’s an example of someone thinking “outside-of-the-box” when it comes to application hosting!

https://mashable.com/article/pregnancy-test-doom/

What Are the Hosting Options with Cloud?

- IaaS
- PaaS
- Serverless / FaaS
- SaaS
- Containers

What do they all mean?

Infrastructure-as-a-Service (IaaS)

- Involves the building out (and management) of virtual instances of:
  - Compute
  - Network
  - Storage

- Akin to spinning up a server (physical or virtual) in your location or data center complete with disks and required network connectivity

Infrastructure-as-a-Service (IaaS)

- The difference is in the where – instead of in your data center, it is created in a data center managed by one of the public Cloud providers

- Your organization is responsible for patching the OS, ensuring all appropriate security updates are applied and that the right controls are in place to govern interaction between this set of components and other infrastructure

DEMO:

IaaS

Platform-as-a-Service (PaaS)

- Involves leveraging managed services from a public Cloud provider
- With this model, an enterprise can focus on management of their application and data vs. focusing on management of the underlying infrastructure
- Patching and security of the infrastructure used to back the managed services falls to the CSP (Cloud Service Provider)

Platform-as-a-Service (PaaS)

- Many managed services support automatic scale up or down depending on demand to help ensure sufficient capacity is in place
- Can be considered synonymous with the term “Cloud native”

DEMO:

PaaS

Serverless / Functions-as-a-Service (FaaS)

- Also represents a type of managed service provided by the CSP
- Cost structure is usually consumption-based (i.e., you only pay for what you use)
- Supports many different coding paradigms (C#/.NET, NodeJS, Python, etc.)

Serverless / Functions-as-a-Service (FaaS)

- Typically, with Serverless (and PaaS), the consumer is only concerned with the application code and data – elements of the CSP's "backbone" used to support are managed by the CSP

- Includes more sophisticated automated scaling capabilities – built for Internet scale

DEMO:

FaaS

Software-as-a-Service (SaaS)

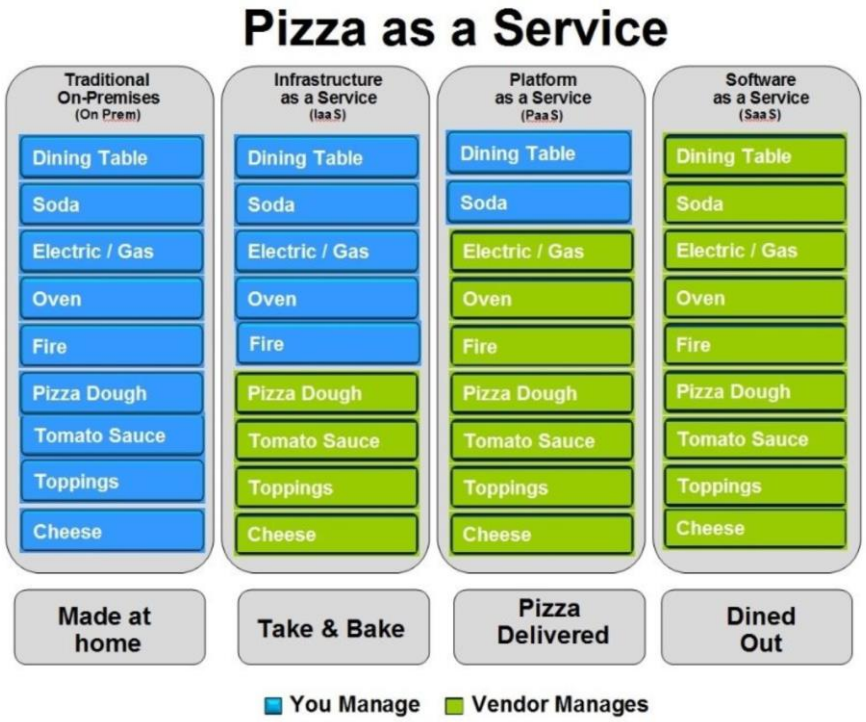
- Subscription-based application services
- Licensed for utilization over the Internet / online rather than for download and install on a server or client machine
- Fully-hosted and fully-managed by a 3rd party

Software-as-a-Service (SaaS)

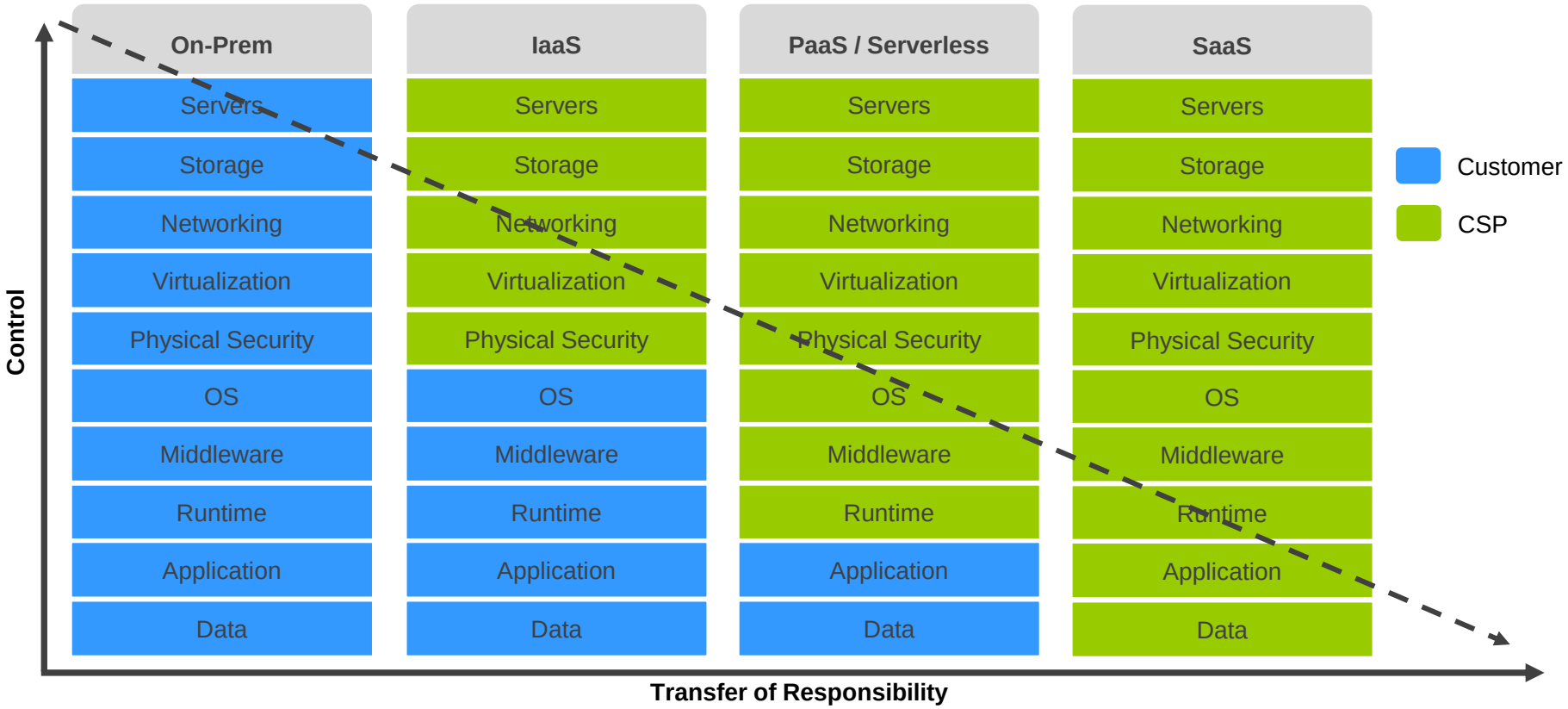
- Of those discussed, often the cheapest option for service consumers
- However, also offers minimal (or no) control, outside of exposed configuration capabilities

Pizza-as-a-Service

From a LinkedIn post by Albert Barron from IBM (https://www.linkedin.com/pulse/20140730172610-9679881-pizza-as-a-service/**)**



Side-by-Side Comparison



Containerization in the Cloud

- One option includes standing up VM’s (IaaS) and installing / managing a Kubernetes cluster on those machines or
- Another option includes leveraging a managed service (PaaS) provided by the CSP
- Options in AWS include Elastic Container Service (ECS), Elastic Container Registry (ECR), and EKS (Elastic Kubernetes Service)

DEMO:

Containers

Which One is Better?

- The answer is “it depends”
- It depends on the type of application
- It depends on the enterprise

Which One is Better?

- It depends on the skillset and expertise within the organization
- It depends on whether you have budget and opportunity to modernize an application environment (in some cases)
- The best option might be a combination of multiple approaches – right tool for the right job

LAB:

Application Hosting

Scenario: Your company (a global leader in FinTech) is currently hosting all infrastructure used to power the business in an on-premise Data Center. This includes a mainframe for primary business functions (customer management, account management, accounts payable, accounts receivable), several Web Apps (for customer interaction), several Web APIs providing backend data and functionality to the UIs, and a system used to manage data feeds from several security cameras used at corporate offices for observation and security.

As a member of the technical staff, you have been asked to provide thoughts and recommendations on moving from the Data Center to the Cloud.

In your assigned breakout room, discuss as a group and be prepared to provide the following: 1) Recommendations for the types of hosting that could be used for the various application components, and 2) the reasoning behind your recommendations – i.e., what are some benefits you would expect the company to receive by implementing your recommendations.

Nominate someone (or volunteer) to share your group's ideas.

Knowledge Check

With this application hosting option, you primarily only pay for what you actually consume:

A. IaaS
B. PaaS
C. Serverless/FaaS
D. SaaS

Knowledge Check

This application hosting option is used to license certain “commodity” functionality like email, CRM, etc. over the Internet:

A. IaaS
B. PaaS
C. Serverless/FaaS
D. SaaS

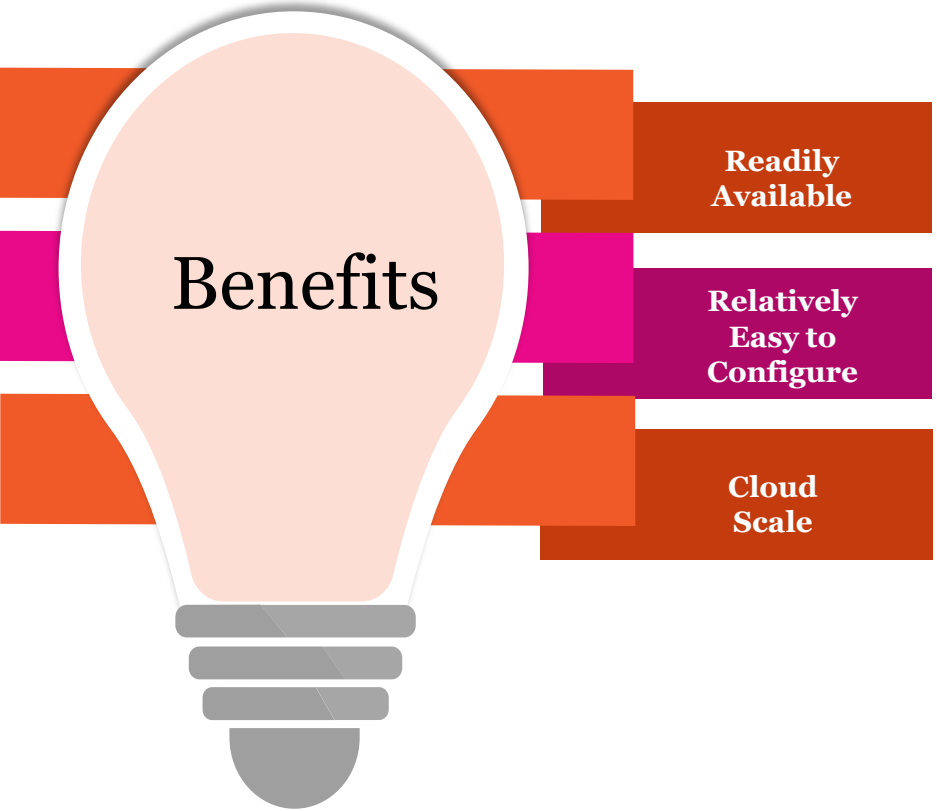
Knowledge Check

Of the application hosting options outlined below, which of them provides the combination of greater control but less transfer of responsibility:

A. IaaS
B. PaaS
C. Serverless/FaaS
D. SaaS

Cloud Native Options in AWS

Cloud Native Services



The offers much in the way of capability and services

Cloud Native Services

- Sophisticated services that bring business value
- Enable expansion of the services ecosystem of your enterprise into new areas of differentiation and segmentation
- Key thing to remember – just because a service is available does not mean that you must use it (or even should use it)
- There should be an architectural vision in place for leverage of the Cloud that is directly aligned with business value

Cloud Native Services

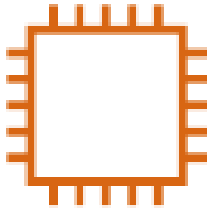
- Why? Because Cloud services will have  associated
- The  must be accounted for and justified as operating expense against business drivers
- With good alignment and a good plan, sophisticated Cloud services can help accelerate business mission

Developing for Cloud

- Compute
- Storage
- Networking
- Database

Compute in AWS

Includes



EC2

AWS Lambda

Elastic Beanstalk

LAB:

EC2

Execute the tutorial available at
https://www.edureka.co/blog/ec2-aws-tutorial-elastic-compute-cloud/

LAB:

AWS Lambda

Execute the tutorial available at
https://docs.aws.amazon.com/lambda/latest/dg/services-apigateway-tutorial.html

Storage in AWS

Includes

Simple Storage Service (S3)

Elastic Block Store (EBS)

AWS Snowball

LAB:

AWS S3

Execute the tutorial available at
https://docs.aws.amazon.com/AmazonS3/latest/userguide/HostingWebsiteOnS3Setup.html

Networking in AWS

Includes

Virtual Private Cloud (VPC)/Subnets

API Gateway

Route 53

Database in AWS

Includes

Relational Database Service (RDS)

DynamoDB

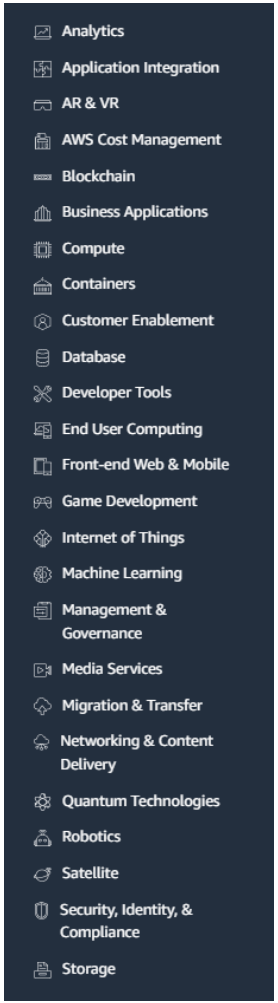
Aurora

LAB:

Amazon RDS

Execute the tutorial available at
https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/TUT_WebAppWithRDS.html

Other in AWS


Analytics
Application Integration
AR & VR
AWS Cost Management
Blockchain
Business Applications
Compute
Containers
Customer Enablement
Database
Developer Tools
End User Computing
Front-end Web & Mobile
Game Development
Internet of Things
Machine Learning
Management & Governance
Media Services
Migration & Transfer
Networking & Content Delivery
Quantum Technologies
Robotics
Satellite
Security, Identity, & Compliance
Storage

Developing on Cloud

- CloudFormation
- Cloud9
- AWS Control Tower

LAB:

AWS CloudFormation

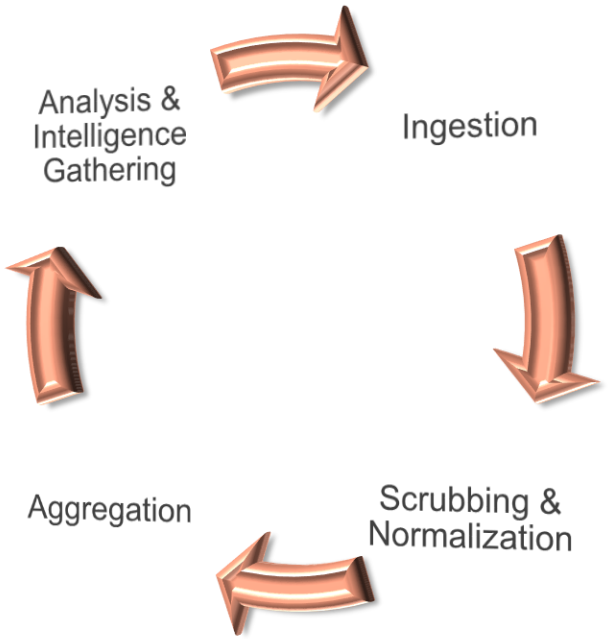
Execute the tutorial available at
https://www.beabetterdev.com/2020/12/06/aws-cloudformation-tutorial/

TBD

TODO: Add Knowledge Check questions

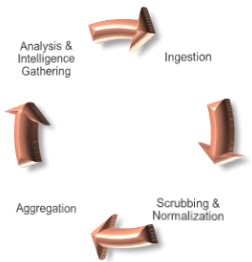
Data Management Lifecycle

Data Management – Stages



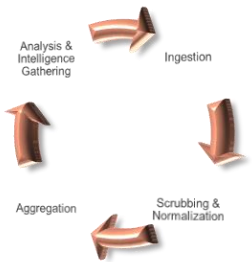
Data Ingestion

- Could be via message exchange or streaming
- Depending on size/scope, may translate to LARGE amounts of incoming data



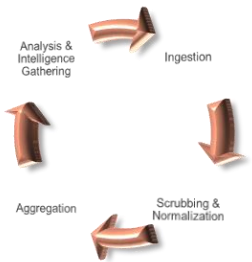
Data Ingestion

- Because of potential scale, bandwidth may be a concern
- Depending on application, latency may also be a concern
- Data may require translation (e.g., from low-level bytes to object or JSON)



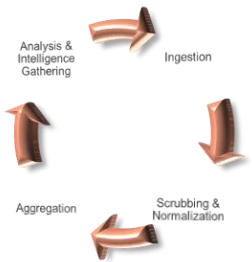
Data Ingestion

- Event hubs or streaming analytics platforms support ingestion at scale
- Provide time and context-aware processing for correct sequencing
- Data may flow through intermediate storage on way to final processing
- Depending on sensitivity of data, could require robust security at each stop



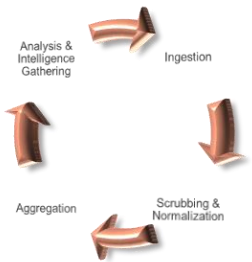
Data Ingestion – What About the Edge?

- Edge components (e.g., gateways) can help optimize
- Preliminary processing at the edge can be used to filter what really matters
- Potential for bundling or compressing data for transmit to cloud
- Can help with bandwidth or latency issues



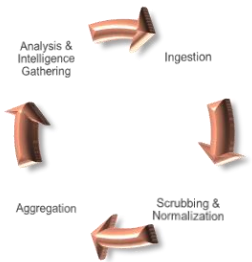
Data Scrubbing & Normalization

- Depending on payload, some portions of the data may not be needed
- Or some portions might contain sensitive detail
- Those parts not needed or sensitive can be “scrubbed” to exclude



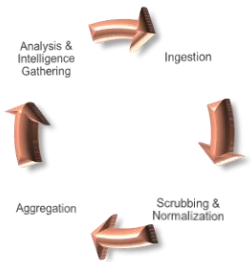
Data Scrubbing & Normalization

- Represents another potential optimization that can preserve storage
- In other cases, similar data may be coming in multiple, disparate formats



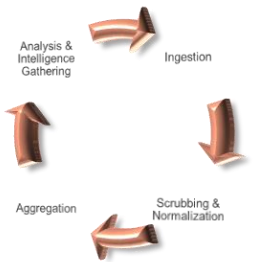
Data Scrubbing & Normalization

- Normalization can bring consistency to the disparate content
- By normalizing, becomes a single dataset for comprehensive analysis
- Normalization may happen as part of ingestion or as part of a separate step



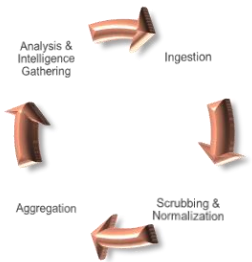
Data Scrubbing & Normalization – What About the Edge?

- Depending on complexity, may execute faster closer to the data
- Might involve proprietary algorithms best kept within full control
- Allows addressing of sensitive data before routed to Cloud
- Can also provide additional optimization (relative to bandwidth)



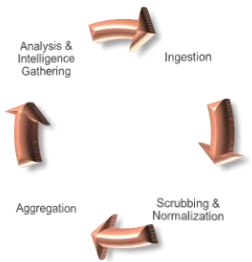
Data Aggregation

- Helps provide full picture of data from multiple streams
- May also be used to enrich with info from other data sources
- Data will be stored in persistent storage for downstream analysis & reporting



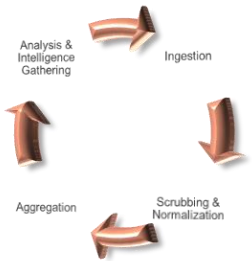
Data Aggregation

- In statistical analysis, the larger the sample size, the more accurate the inference
- To manage costs, large sets of data may leverage different types of storage:
  - Hot storage – most recent data and most relevant for current analysis
  - Cool storage – data not actively used but potentially relevant (short-term trends)
  - Cold or archive storage – data kept for historical purposes and long-term trending
- Security of the stored data and encryption at rest become critical



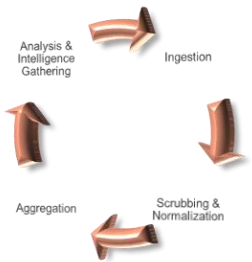
Data Aggregation – What About the Edge?

- Provides an additional layer of storage
- Data not transmitted to Cloud (due to optimizations) may still be valuable to keep
- Enables storage of sensitive data in “raw” format in controlled environment
- Can help balance costs against short to mid-term retention requirements



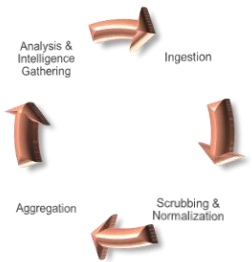
Data Analysis & Intelligence Gathering

- In the digital age, data is the competitive edge
- Companies that manage their data as a critical asset succeed
- Keys:
  - Aggregating efficiently
  - Analyzing effectively



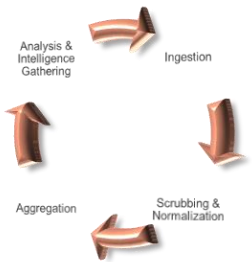
Data Analysis & Intelligence Gathering

- Goal is to identify and leverage the most important data points
- Importance is measured by business value-driven decision-making
- What can I learn about today's customers, scenarios, or business cases?
- What can I effectively predict about tomorrow?



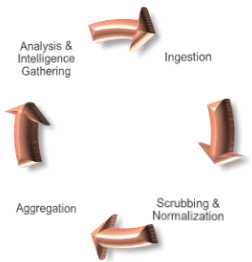
Data Analysis & Intelligence Gathering

- Requires balancing of competing concerns:
  - To increase quality of intelligence, more data is required (sometimes MUCH more)
  - But massive datasets can be complex to manage and process



Data Analysis & Intelligence Gathering

- Enter ML / AI:
  - Algorithms are used to build mathematical models from existing data
  - Results in a mathematical “trajectory” (and confidence level)
  - Algorithms can be configured to learn and improve over time
- Hyperscale available in the Cloud brings near-limitless power to bear



Overview - What is AI?

It is the quest to build machines that can reason, learn, and act intelligently, and it has barely begun

MIT

Artificial intelligence (AI) applies advanced analysis and logic-based techniques, including machine learning, to interpret events, support and automate decisions and take action

Gartner

AI is computer programming that learns and adapts. It can't solve every problem, but its potential to improve our lives is profound

Google

It is the capability of a computer system to mimic human-like cognitive functions such as learning and problem-solving

Microsoft

The Turing Test

The **Turing test**, originally called the **imitation game** by Alan Turing in 1950,[2] is a test of a machine's ability to exhibit intelligent behaviour equivalent to, or indistinguishable from, that of a human. Turing proposed that a human evaluator would judge natural language conversations between a human and a machine designed to generate human-like responses. The evaluator would be aware that one of the two partners in conversation is a machine, and all participants would be separated from one another. The conversation would be limited to a text-only channel such as a computer keyboard and screen so the result would not depend on the machine's ability to render words as speech.[3] If the evaluator cannot reliably tell the machine from the human, the machine is said to have passed the test. The test results do not depend on the machine's ability to give correct answers to questions, only how closely its answers resemble those a human would give.

https://en.wikipedia.org/wiki/Turing_test

Machine Learning Examples

- See https://www.businessinsider.com/shane-wighton-robotic-basketball-hoop-cant-miss-2020-5
- See https://breakingdefense.com/2020/08/ai-slays-top-f-16-pilot-in-darpa-dogfight-simulation/
- See https://www.schwab.com/automated-investing/what-is-a-robo-advisor

LAB:

Responsible AI

Responsible AI is a discipline that seeks to ensure we are developing and using Machine Learning/AI in a manner that eliminates unfair bias and that strikes the proper balance between efficiency and benefits to humankind.

Pick one of the examples from the previous slide. In your assigned breakout room, discuss as a group and be prepared to provide the following: 1) what are some benefits that you could see for the greater good from the given application of intelligence in technology, and 2) what are some potential ways that the technology could be abused for less noble purposes.

Nominate someone (or volunteer) to share your group's ideas.

Types of Machine Learning

Machine Learning

Supervised

What is Supervised Learning?

An algorithm uses training data and feedback from human users to learn the relationship between given inputs and a given output

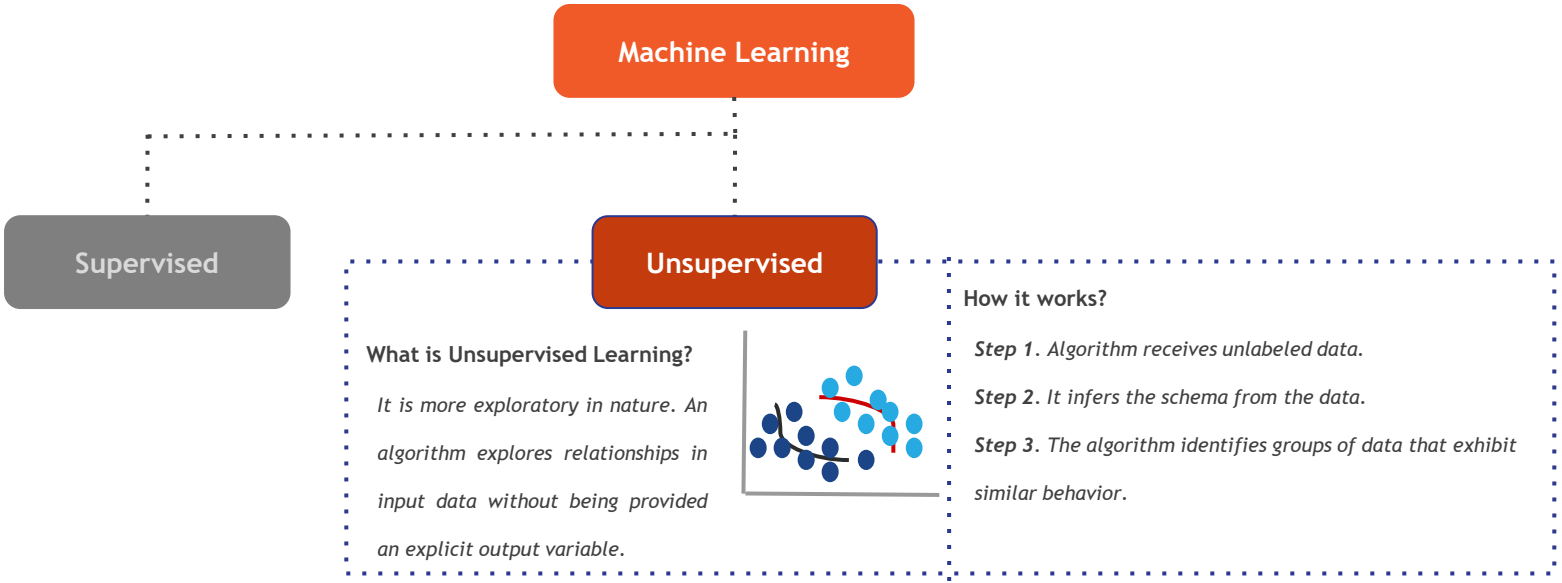
How it works?

***Step 1**. Label input data that defines output variables.*

***Step 2**. Train the algorithm on the data to find the connection between input variables and output.*

***Step 3**. Once training is completed, then apply the algorithm to new data.*

Types of Machine Learning



Types of Machine Learning

Machine Learning

- **Supervised**
- **Unsupervised**
- **Reinforcement**

How it Works?

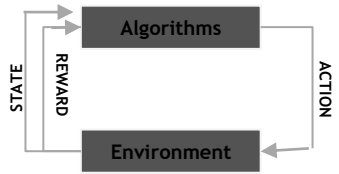
Step 1. *Algorithm takes an action affecting the environment.*

Step 2. *It receives a reward if the action brings the machine a step closer to max. available reward.*

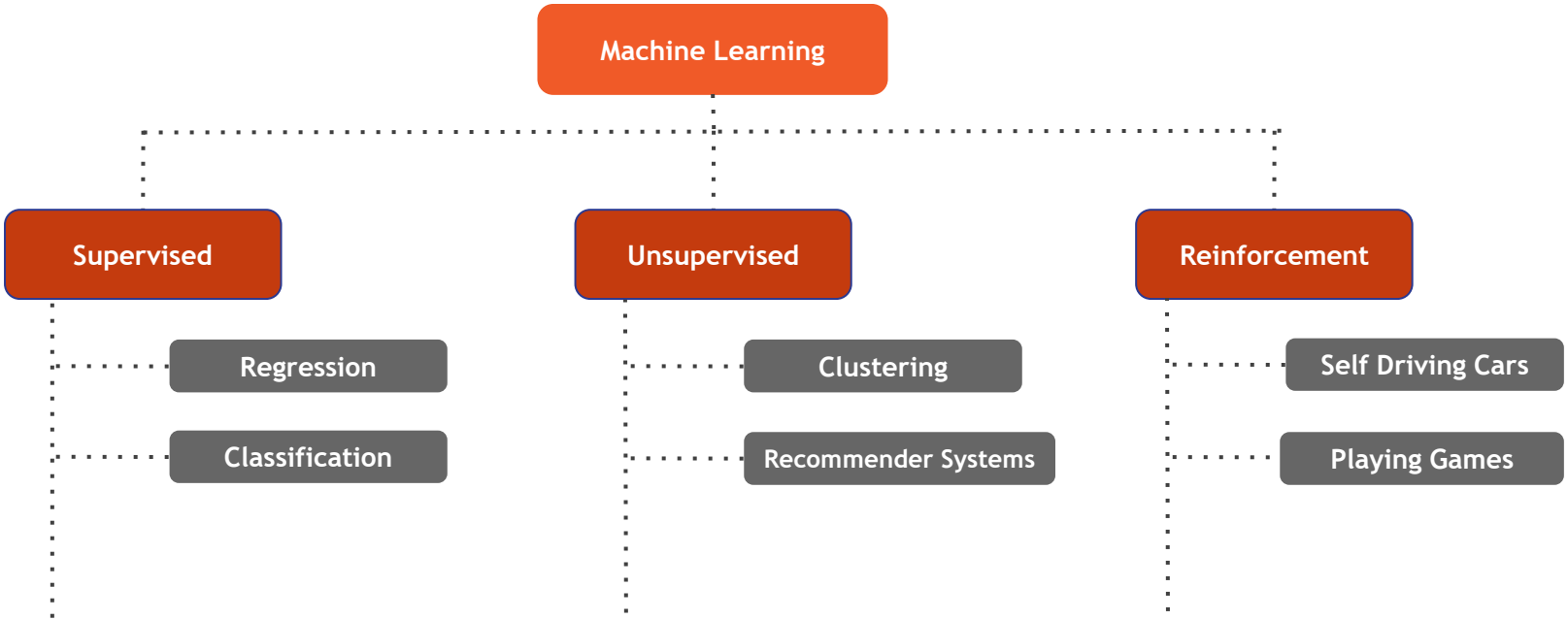
Step 3. *The algorithm optimizes for the best series of actions by correcting itself over time.*

What is Reinforcement Learning?

An algorithm learns to perform task simply by trying to maximize the rewards it would receive for its actions.



Types of Machine Learning



DEMO

Analyze Image

DEMO

Assessing Sentiment

TBD

TODO: Add Knowledge Check questions

Operating in the Cloud

Monitoring in the Cloud

- Monitoring & logging are key considerations in any Cloud environment
- Systems (you hope) will be running around-the-clock – maximizing business benefit
- Unless you want to directly “babysit” those systems around-the-clock, you will need automated monitoring, logging and alerting to notify you of any issues
- Allows you to optimize handling for those exceptional cases when there is a problem

Monitoring in the Cloud

- Capabilities exist to log and aggregate log data from the public Cloud
- There are likely already processes in place to monitor on-premise systems
- The goal is a strategy that allows you to pull that data together so you can analyze and make decisions holistically

Endpoint Protection & Security

- The services exposed by a company used to provide its business value are critical
- The data consumed by a company in the provision of that business value could be very sensitive
- There are multiple regulations in place requiring the protection of sensitive data (e.g., PCI, SOX, HIPAA and GDPR)
- Failure to adhere to those regulations can cost a company significantly – either in actual $’s or in reputation (which can be more damaging)

Endpoint Protection & Security

- The issue is not only one of data security – there are “bad actors” that work to take down sites and services
- One of the ways that service can be hindered is through a DDoS (Distributed Denial of Service) attack
- For DDoS, attackers will attempt to “flood” a service with so much bogus volume that it becomes unable to satisfy real business requests

Endpoint Protection & Security

- Most Cloud platforms provide services to help you protect against a DDoS attack
- Can include API management services (subscriptions, key-based access, throttling)
- Web Application Firewall (or WAF) is another service provide by Cloud platforms to monitor, filter and block (if required) incoming traffic

Endpoint Protection & Security

Potential Challenges

- The threat and regulatory landscapes are constantly evolving
- In a Hybrid Cloud scenario, creating a solution that gives you comprehensive protection across your on-premise resources, your Cloud resources and the connectivity between the two is not trivial
- Optimal application security and infrastructure security requires planning and specialized skillsets
- Good architectural practices (e.g., Least Privilege and Secure-by-Default) can help limit the “blast radius”

DEMO

Operational Management in the Cloud

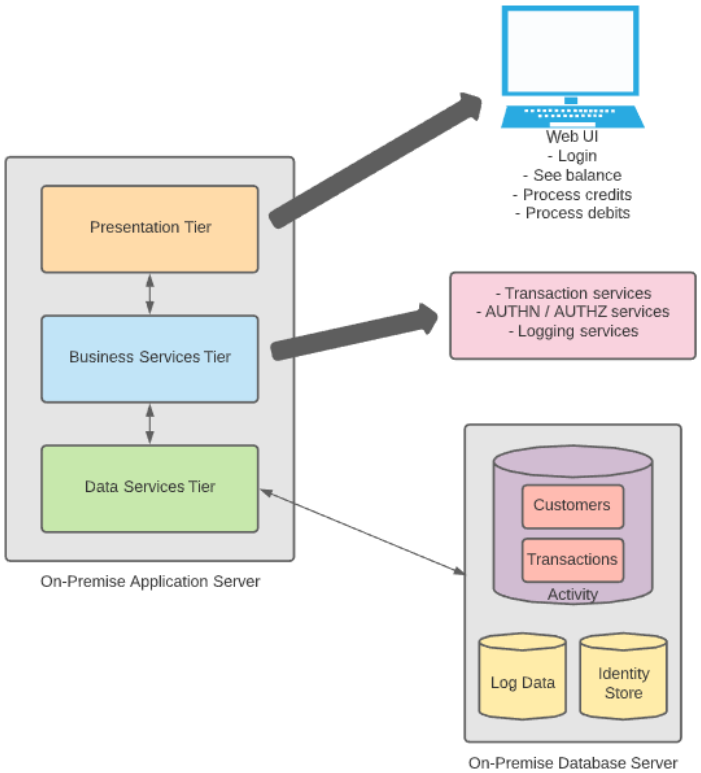
Cloud Modernization

Application Portfolio Assessment

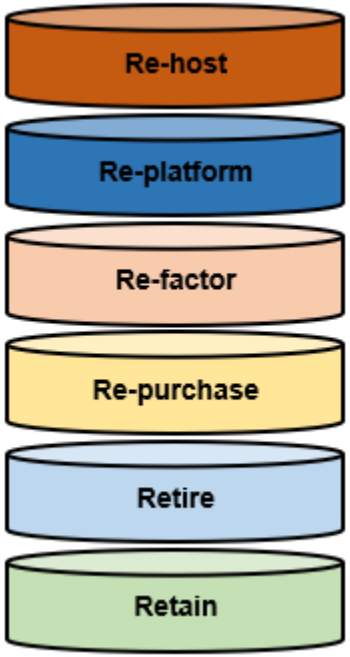
- Will likely include
  - Review of the current state architecture
  - Discussion with stakeholders about roadmap and plans for target state
  - Discussion on expected timing

- The goal is to ensure that migration efforts are focused on the activities that will bring greatest business benefit against optimized cost

Banking App – Current State



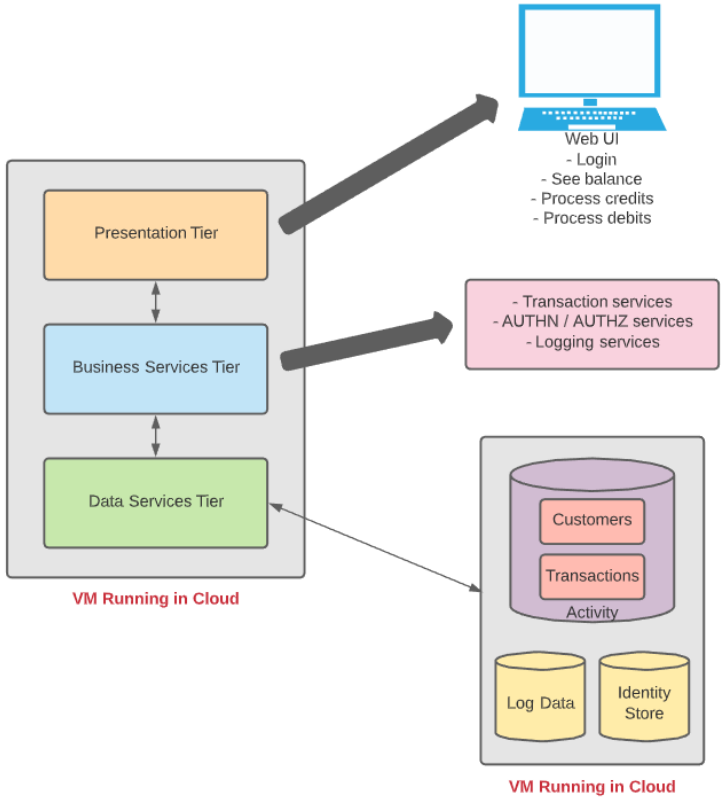
The 6 R's – Application Migration Strategies



- AKA “lift & shift”
- For all intents and purposes, involves recreating the on-premise infrastructure in the Cloud
- Sometimes used to expedite retirement of a data center

- Can be a mechanism to quickly migrate workloads and see immediate cost savings, even without Cloud optimizations
- There are third-party tools available to help automate the migration
- Once the application is in the Cloud, it can be easier to apply Cloud optimizations vs. trying to migrate and optimize at the same time

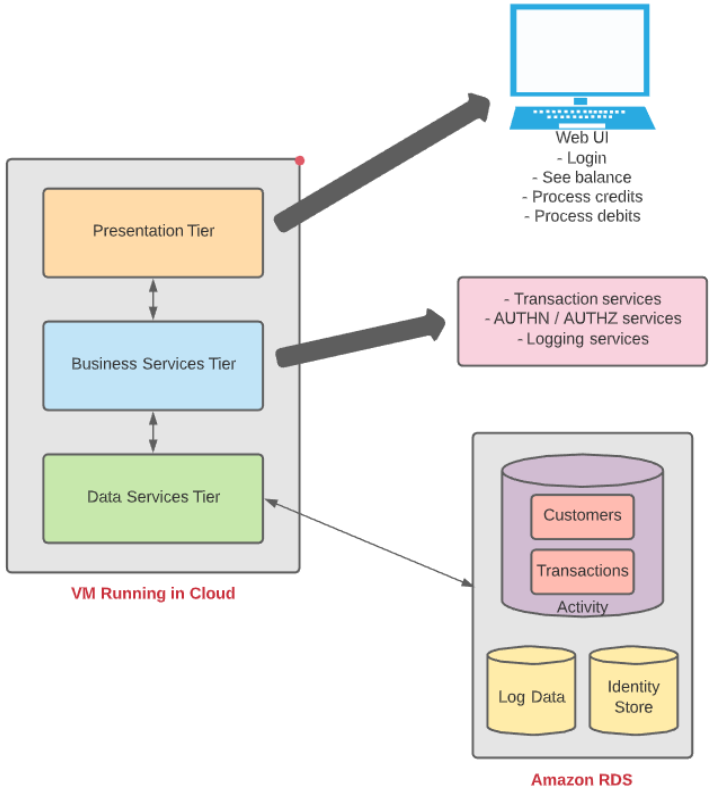

Re-host

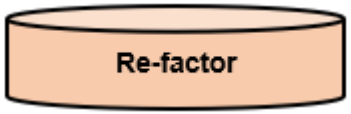

Web UI
- Login
- See balance
- Process credits
- Process debits
Presentation Tier
Business Services Tier
Data Services Tier
- Transaction services
- AUTHN / AUTHZ services
- Logging services
VM Running in Cloud
Customers
Transactions
Activity
Log Data
Identity Store
VM Running in Cloud

- AKA “lift, tinker & shift” or “lift & fiddle”
- Core architecture of the application will not change
- Involves recreating most of the on-premise infrastructure in the Cloud with a few Cloud optimizations

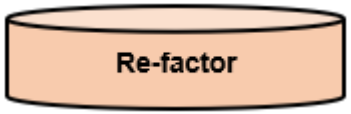
- Those optimizations will involve a move to one or more Cloud native services for a specific, tangible business benefit
- A common example is moving to a managed database (e.g. Relational Database Service, or RDS, in AWS)
- Enables migration speed while providing cost savings or benefit in a targeted portion of the application’s architecture


Re-platform

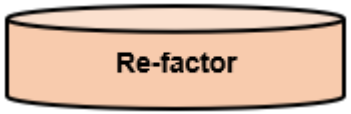

Web UI
- Login
- See balance
- Process credits
- Process debits
Presentation Tier
Business Services Tier
Data Services Tier
VM Running in Cloud
- Transaction services
- AUTHN / AUTHZ services
- Logging services
Customers
Transactions
Activity
Log Data
Identity Store
Amazon RDS



- Involves rearchitecting the application to maximize utilization of Cloud native optimizations
- Likely the most expensive and most complex of the available options
- The application profile needs to fit, and business value must be identified commensurate with the cost required to execute



- Good option if the application can benefit from features, scalability or performance offered by the Cloud
- Examples include rearchitecting a monolith to microservices running in the Cloud or moving an application to serverless technologies for scale



?

To Be Determined

- Good fit for applications that are candidates for moving from on-premise licensing (for installed products) to a SaaS model
- Often applications that have been installed to manage a specific type of business capability
- Examples can include Customer Relationship Management (CRM), Enterprise Resource Planning (ERP), HR or e-mail (among others)

- In some cases, an enterprise may have a “healthy” percentage of legacy applications that are no longer being used or maintained (especially for larger organizations)
- When completing the Application Portfolio Assessment and associated interviews with application owners, look for opportunities to recommend retire of the unused legacy apps
- This type of application can represent a “quick win”

- The associated savings can potentially be advantageously factored into the business case for the Cloud modernization effort
- Finally, retiring unused apps reduces attack surface area from a security perspective

- Applications that must remain in place but that cannot be migrated to the Cloud without major refactor
- This type of application profile may prevent the ability to completely move out of the data center (at least in the interim)
- From a cost perspective, it can be difficult for an enterprise to take on both the operating expense of Cloud while continuing to carry the capital expense of a data center and on-premise infrastructure
- The hybrid model can be a good solution to support where needed

LAB:

Application Hosting

Scenario: Your company (a global leader in FinTech) is currently hosting all infrastructure used to power the business in an on-premise Data Center. This includes a mainframe for primary business functions (customer management, account management, accounts payable, accounts receivable), several Web Apps (for customer interaction), several Web APIs providing backend data and functionality to the UIs, and a system used to manage data feeds from several security cameras used at corporate offices for observation and security.

As a member of the technical staff, you have been asked to provide thoughts and recommendations on moving from the Data Center to the Cloud.

In your assigned breakout room, discuss as a group and be prepared to provide the following: 1) Thoughts & recommendations on which of the 6 R's you'd recommend for each of the high-level infrastructure components used at the company, and 2) any business or technical drivers behind your recommendations.

Nominate someone (or volunteer) to share your group's ideas.

TBD

TODO: Add Knowledge Check questions

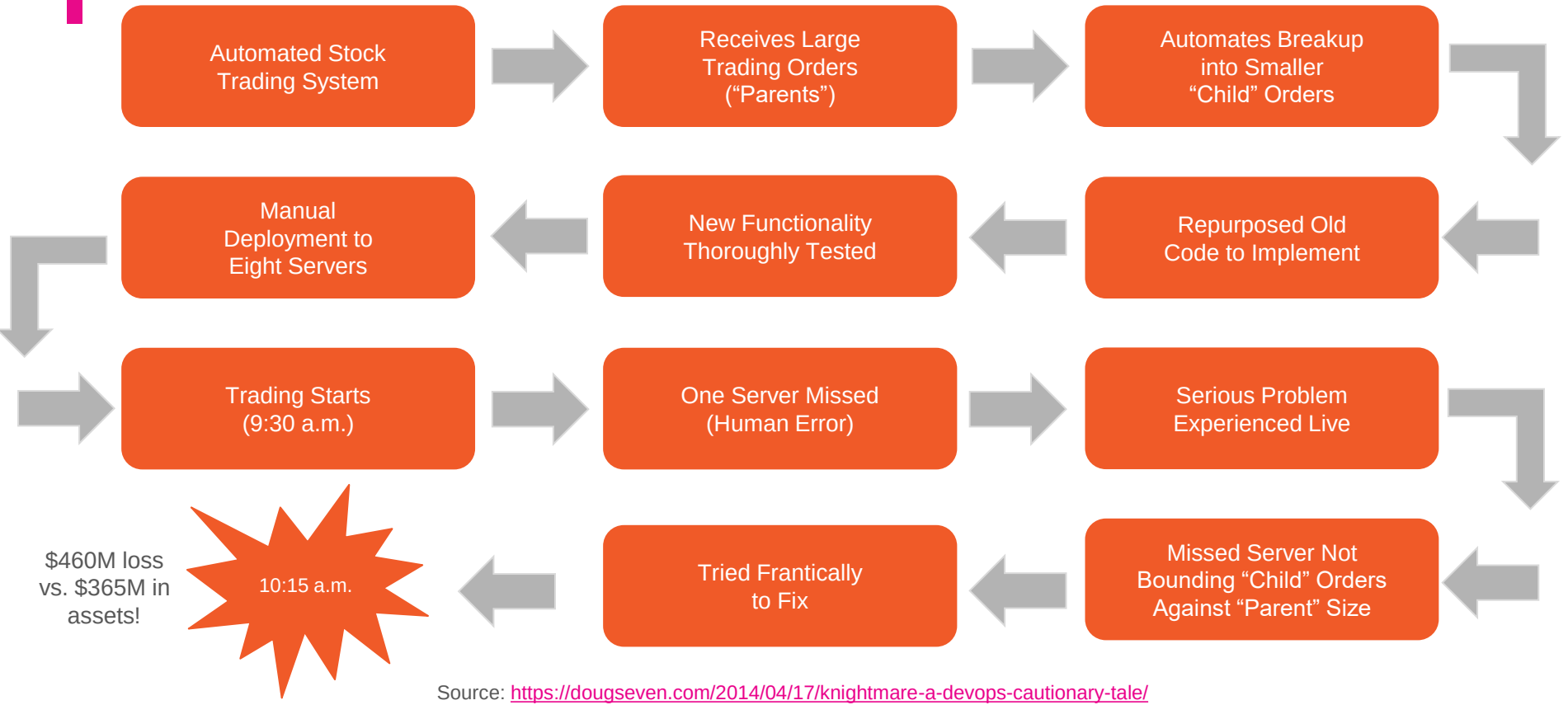
CI/CD in AWS

What a difference 45 minutes can make!!

How Long Does it Take for a Company to Go Bankrupt?

If the circumstances are right, it can happen in just 45 minutes!

Knight Capital Group – What Happened?



Source: https://dougseven.com/2014/04/17/knightmare-a-devops-cautionary-tale/

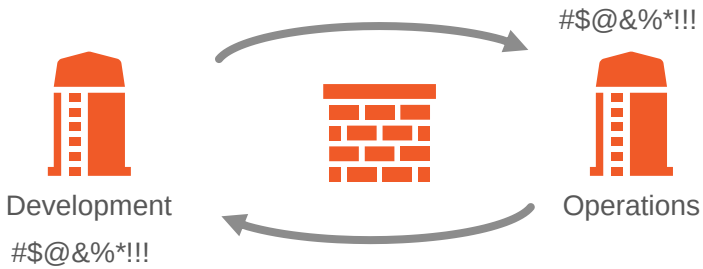
Moment of Reflection

From the brief overview of the Knight Capital Group scenario, can you identify any specific implementation choices that might’ve led to the issues encountered?

Type your answer in chat or come off mute and share

DevOps – What it is & why it’s valuable

Approach Used In the “Olden” Times



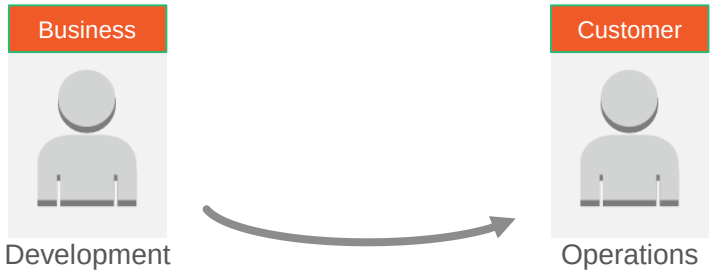
- Development & Operations siloes
- Lack of understanding of the other perspective
- Lack of communication & partnership
- “Fire and forget” engagement

DevOps – The New Way

- Development & Operations work together
- Each understands & appreciates the position of the other
- Good communication, partnership, & collaboration
- Ongoing engagement – continuous improvement

DevOps – The Three Ways

The First Way: Flow/Systems Thinking

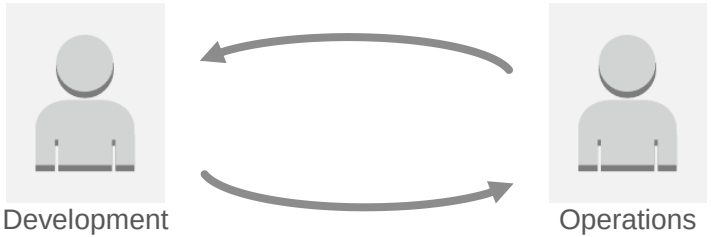


Source: Gene Kim (https://itrevolution.com/the-three-ways-principles-underpinning-devops/)

- Holistic vs. siloed view – success is defined by the performance of the entire system vs. a specific team or individual
- Starts with requirements (defined by the business) and not called “done” until value is delivered to the customer
- Target outcomes:
  - Known defect never passed to downstream teams
  - Favor global performance vs. focusing on local optimization
  - Goal is increased flow (value)
  - Asking ourselves the question: “What don’t I know about our systems?”

DevOps – The Three Ways

The Second Way: Amplify Feedback Loops

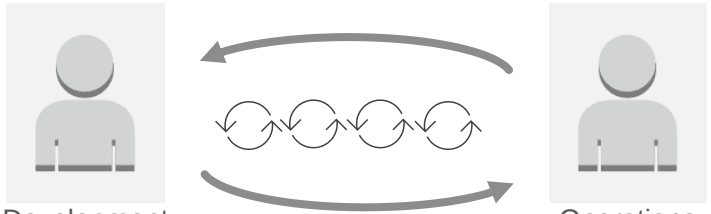


Source: Gene Kim (https://itrevolution.com/the-three-ways-principles-underpinning-devops/)

- Create right-to-left feedback loops
- Shorten & amplify so information needed to make continuous improvements & continuous correction is readily available
- Target outcomes:
  - Understanding all customer perspectives (internal & external)
  - Responding to all customer concerns (internal & external)
  - Knowledge is power!

DevOps – The Three Ways

The Third Way: Culture of Continual Experimentation & Learning

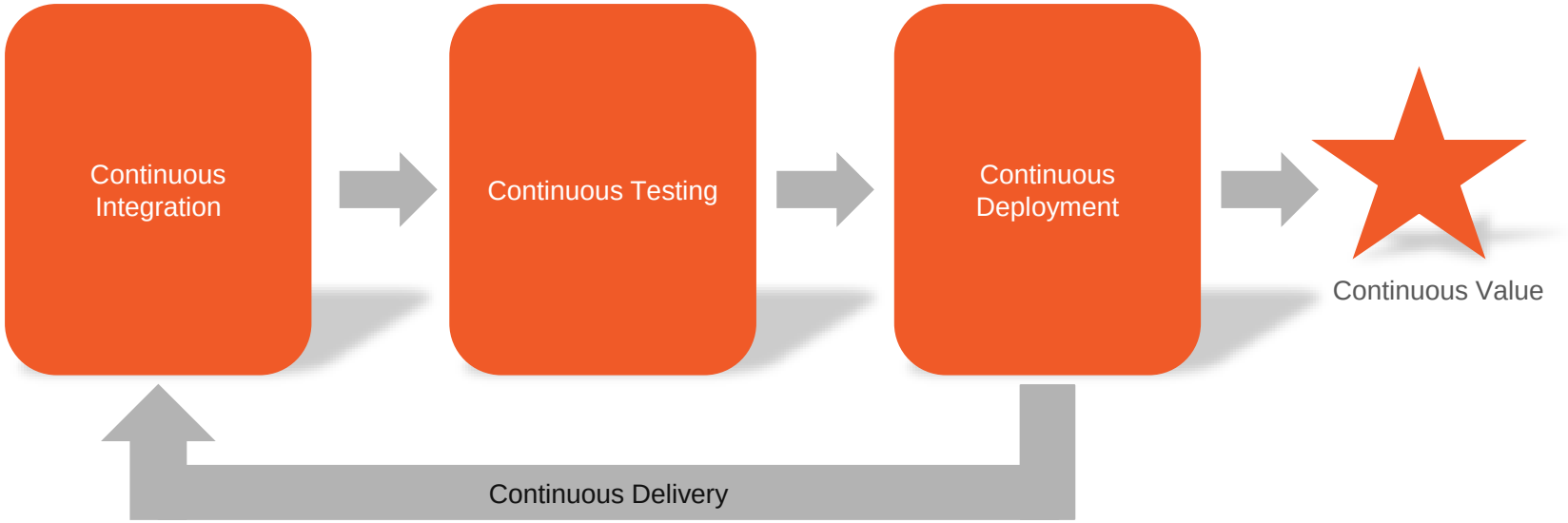


Source: Gene Kim (https://itrevolution.com/the-three-ways-principles-underpinning-devops/)

- Culture that fosters:
  - Continual experimentation, risk-taking, and learning from failure
  - Practice makes perfect!
- Target outcomes:
  - Allocating time for continual improvement (e.g., resolution of technical debt as a standard “category” of work)
  - Taking risks and seeing them pay off – No guts no glory!
  - Expecting failure & building resiliency into the process

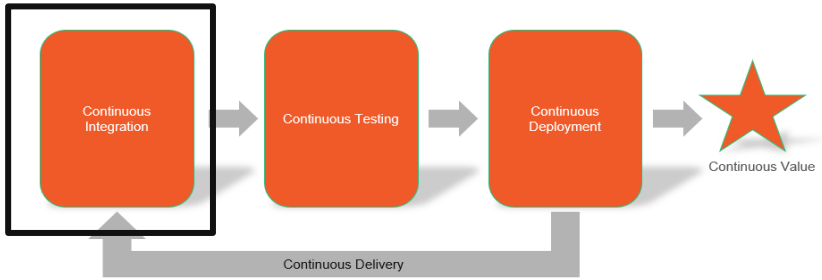
Continuous Delivery

Continuous Delivery



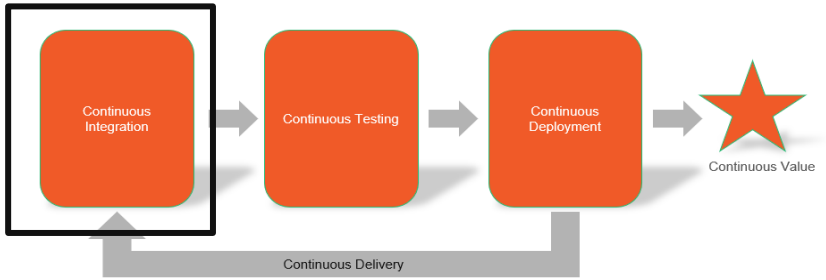
Continuous Integration

Continuous Integration



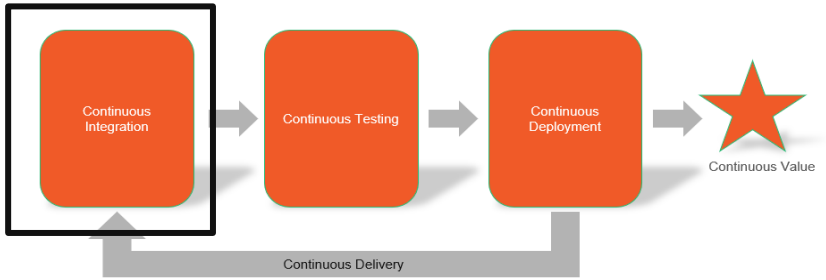
- Changes from individual developers are merged & system is validated “as a whole”
- Code changes are detected & automated processes are executed to assess
- Any issues uncovered are communicated to the developers for remediation (feedback)
- Targeted testing (i.e., unit testing) executed as part of CI flow

Continuous Integration



- Initiation is often event-driven
- Rather than waiting for a “critical mass” of changes before starting our testing, each change can be exercised & validated as it gets persisted to the centralized code repository
- Amplifies the feedback loop rather than burying in a “sea” of other changes
- Sets the stage for multiple types of validation (including security validation, or DevSecOps)

Continuous Integration

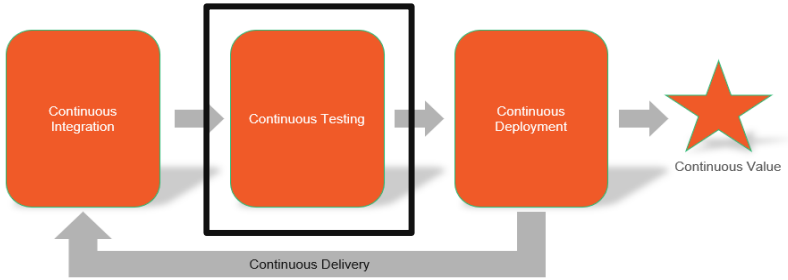


Best practices:

- Minimize dependencies between features – each feature should be able to stand on its own
- Work to drive out manual steps and make automation the priority
- Stabilize data sources & other system dependencies through mocking
- Include assessment of code quality (e.g., cyclomatic complexity) & unit test coverage as key parts of verification workflow

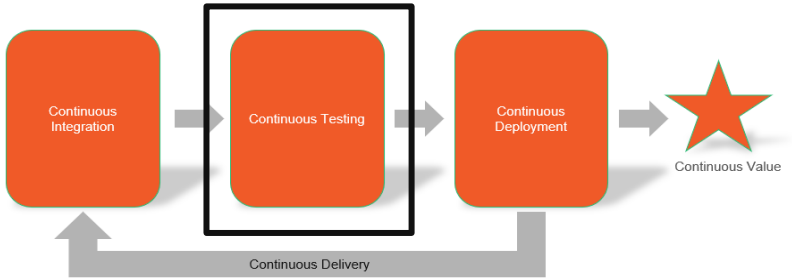
Continuous Testing

Continuous Testing



Common Types	Description
Unit testing	Testing discrete units of code in isolation from other code (e.g., testing a single method). Leverages mocking to stabilize other system dependencies so that tests are repeatable. Code coverage often used to assess quality & sufficiency of unit testing. Should be automated.
Regression testing	Testing that verifies that newly introduced changes for development of a feature do not inadvertently break other, existing features. Automated unit tests can also fill the role of a regression test suite nicely. Quickly confirming no breakages.
Integration testing	Testing that verifies that groups of components & component features operate correctly when combined with other features (e.g., multiple functions or modules that provide a larger “chunk” of functionality). Usually represents a layer above unit testing but below functional testing. Should be automated.
Functional testing	Testing that verifies functional requirements for critical workflows in the software end-to-end. Can be (and probably should be) implemented as automated black box tests, requiring minimal knowledge & minimal assumptions about the inner workings of the software.
Acceptance testing	Testing that verifies the functionality of the software against specific acceptance criteria defining the difference between “good” & “bad”. May include manual testing/utilization by a subset of testers to verify that the software will operate correctly when leveraged by end users in production. AKA does the software do what it is supposed to from the end user’s perspective?
Security testing	Testing that verifies the functionality of the software against critical security requirements as identified & prioritized via threat modeling. Through a combination of static & dynamic testing, assesses the software for security vulnerabilities or deficiencies from a compliance & regulatory perspective (A.K.A. DevSecOps). Examples include SQL Injection and Cross-Site Scripting.
Performance testing	Testing that verifies that the software will meet defined SLAs (Service Level Agreements) when exercised under load or stress. Can also be used to validate the software’s ability to elastically scale (out or in) based on volumes. How does the software perform under load?

Continuous Testing

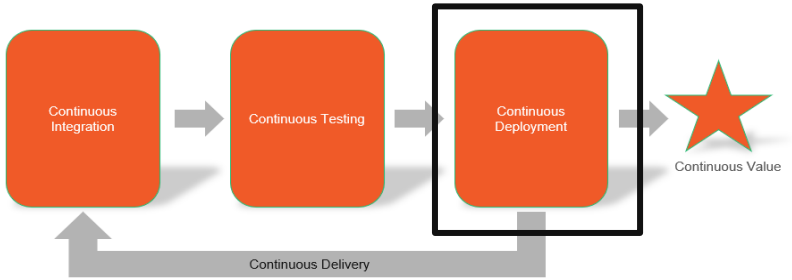


How does Continuous Integration help promote software quality?

- Maximizes the use & benefits of automation to enable “early & often”
- Provides multiple “hooks” within the workflow for plugging in the various types of testing
- Enables integration of testing results into the pipeline as quality gates
- Accelerates test result feedback for visibility & effective disposition

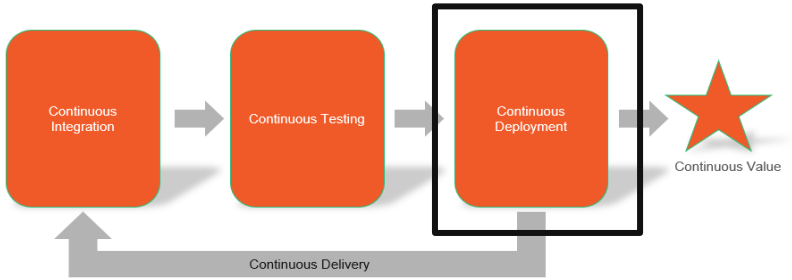
Continuous Deployment

Continuous Deployment



- Does not necessarily mean continuously deployed, but rather continuously *ready* to deploy
- Through a release pipeline, seeks to push latest version of the software through the environments at the frequency required by the business
- Intended to help get innovation “quickly” into the hands of users & customers for serving needs & gathering feedback

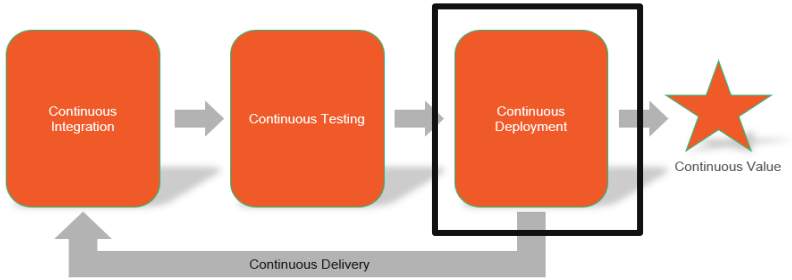
Continuous Deployment



Version management:

- Often leverages semantic versioning (major.minor.patch)
- Often leverages explicit versioning
- Seeks to keep a clear distinction & separation between each version of the software
- Goal is version traceability & visibility when upgrade is possible, or rollback is required

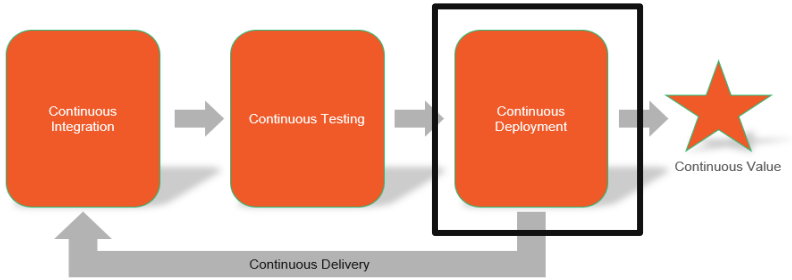
Continuous Deployment



Blue/Green testing:

- Maintains two (or more) identical hosting environments
- Provides the opportunity to run multiple versions of the software in parallel
- Production traffic is managed through routing – switch routing from old to new, & verify
- If issues uncovered, routing can be switched back to previous version for quick rollback

Continuous Deployment



Canary testing:

- Provides the opportunity to run multiple versions of the software in parallel
- Small portion of traffic can be routed to new version to ease in adoption
- Allows verification of new version in lower risk manner (like a canary in a coal mine)
- As confidence in new version builds, more and more traffic can be transitioned over until new version is used exclusively

LAB:

Responsible AI

Using the details of the Knight Capital Group incident (https://dougseven.com/2014/04/17/knightmare-a-devops-cautionary-tale/) and the information reviewed in the past few slides, in your assigned breakout room, discuss the following:

- What are some of the primary issues you see with Knight Capital Group's approach (and that you believe might have led to their "downfall")?
- Where could the principles of CI/CD and DevOps have assisted in preventing those issues?

Nominate someone (or volunteer) to share your group's ideas.

So, What Went Wrong?

Repurpose of legacy code and legacy flags/variables without proper quality checks

Fully manual deployment

No “kill switch” and no clear path to rollback

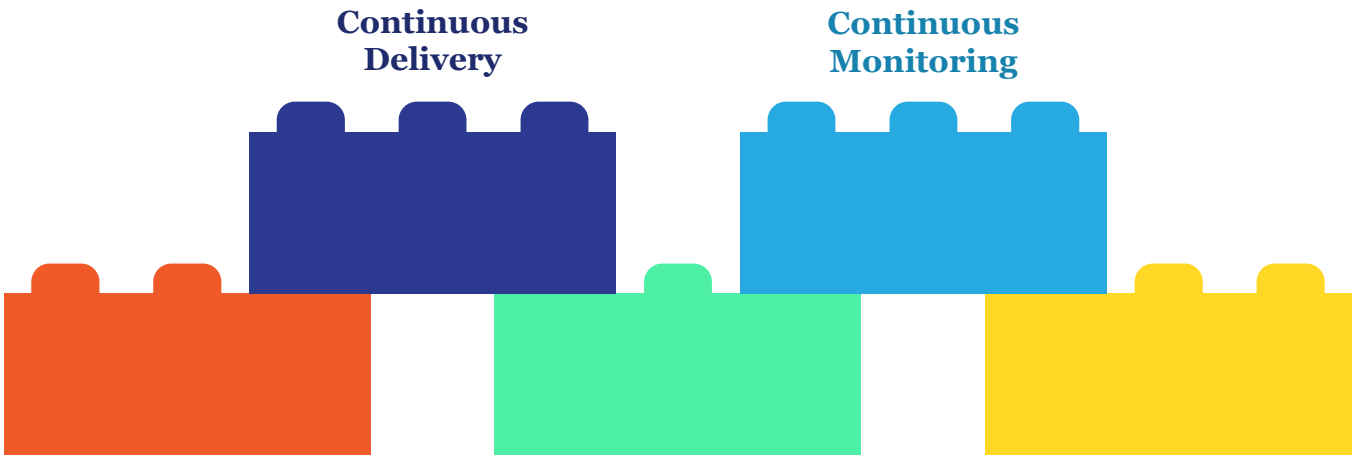
Insufficient observability and monitoring to alert resources to issues

Forced to test and troubleshoot live in Production

Building Blocks & Knight Capital Group

Software remains in a “ready to release” state enabling agility

Copious and effective monitoring alerts us to problems quickly so we can respond quickly



Continuous Integration

Automated testing finds issues quickly and helps ensure quality/reduce fear when releasing

Continuous Deployment

Automated release at the “push of a button” reduces element of human error

Automated Infrastructure

Automated build out of secure runtime environments (and ability to quickly roll back in the event of a discovered issue) drives stability

TBD

TODO: Add Knowledge Check questions

AWS CodeBuild – The What & How

TBD

TODO: Potentially add a few slides outlining/summarizing some of the high-level technical details of AWS CodeBuild. Alternatively, use the upcoming labs/demos (working alongside the participants) as an opportunity to drill into more of the technical details as part of exercise execution.

LAB:

AWS CodeBuild

Execute the tutorial available at
https://docs.aws.amazon.com/codebuild/latest/userguide/getting-started.html

LAB:

AWS CodeBuild – Build Badges

Execute the tutorial available at
https://docs.aws.amazon.com/codebuild/latest/userguide/sample-build-badges.html

DEMO

Docker Custom Image in CodeBuild

AWS CodePipeline – The What & How

TBD

TODO: Potentially add a few slides outlining/summarizing some of the high-level technical details of AWS CodePipeline. Alternatively, use the upcoming labs/demos (working alongside the participants) as an opportunity to drill into more of the technical details as part of exercise execution.

LAB:

AWS CodePipeline

Execute the “Simple Pipeline Walkthrough” tutorial available at https://docs.aws.amazon.com/codepipeline/latest/userguide/tutorials-simple-s3.html

DEMO

Create a Four-Stage Pipeline

LAB:

AWS CodePipeline

Execute the CloudFormation/CodePipeline tutorials available at https://docs.aws.amazon.com/codepipeline/latest/userguide/tutorials-cloudformation.html

Thank you!

If you have additional questions, please reach out to me at: (email address)

DevelopIntelligence
A PLURALSIGHT COMPANY