Principles in Cloud Architecture Design - Operational Excellence

Cloud Infrastructure Engineering

**Nanyang Technological University
& Skills Union - 2022/2023**

Course Content

- Quick Check-In
- Dive into what Operational Excellence on the cloud means and the best practices in terms of Operational Excellence
- Build operationally excellent architectures

Time	What	How or Why
7:15pm - 7:40pm	Part 1 - Presentation	Operational Excellence Principles
7:40pm - 8:00pm	Part 2 - Activity	
8:00pm - 8:10pm	Break	
8:10pm - 8:30pm	Part 3 - Presentation	Best Practices for Operational Excellence
8:30pm - 8:50pm	Part 4 - Activity	
8:50pm - 10:00pm	Summary & Assignments	

Recap

- Performance Design Principles
    - Implement Cloud Financial Management
    - Adopt a Consumption Model
    - Measure Overall Efficiency
    - Stop Spending Money on Undifferentiated Heavy Lifting
    - Analyze & Attribute Expenditure
- Best Practices
    - Practice Cloud Financial Management
    - Expenditure and usage awareness
    - Cost-effective resources
    - Manage demand and supply resources
    - Optimize over time

Self Study Check-In

Q1) Which will you choose, small & frequent changes or big less often changes?

Q2) How will you anticipate failure?

Overview

Overview

This module focuses on the **operational excellence** pillar and how to apply it as the foundation of your well-architected solutions.

Operational excellence is challenging to achieve in environments where operations is perceived as a function isolated and distinct from the lines of business and development teams that it supports.

By adopting the practices here, you can build architectures that **provide insight to their status**, are **enabled for effective and efficient operation** and **event response**, and can **continue to improve and support your business goals**.

Operational Excellence Principles

Op Excellence Principles Summary

- Perform Operations as Code

- Make Frequent, Small, Reversible Changes

- Refine Operations Procedures Frequently

- Anticipate Failure

- Learn From All Operational Failures

Perform Operations as Code

In the cloud, you can apply the **same engineering discipline** that you use for application code to your entire environment.

You can **define your entire workload** (applications, infrastructure) as code and **update it with code**.

You can **implement your operations procedures as code** and **automate** their execution by triggering them in response to events.

By performing operations as code, you limit human error and enable consistent responses to events.

Perform Operations as Code

CONTINUOUS IT STACK

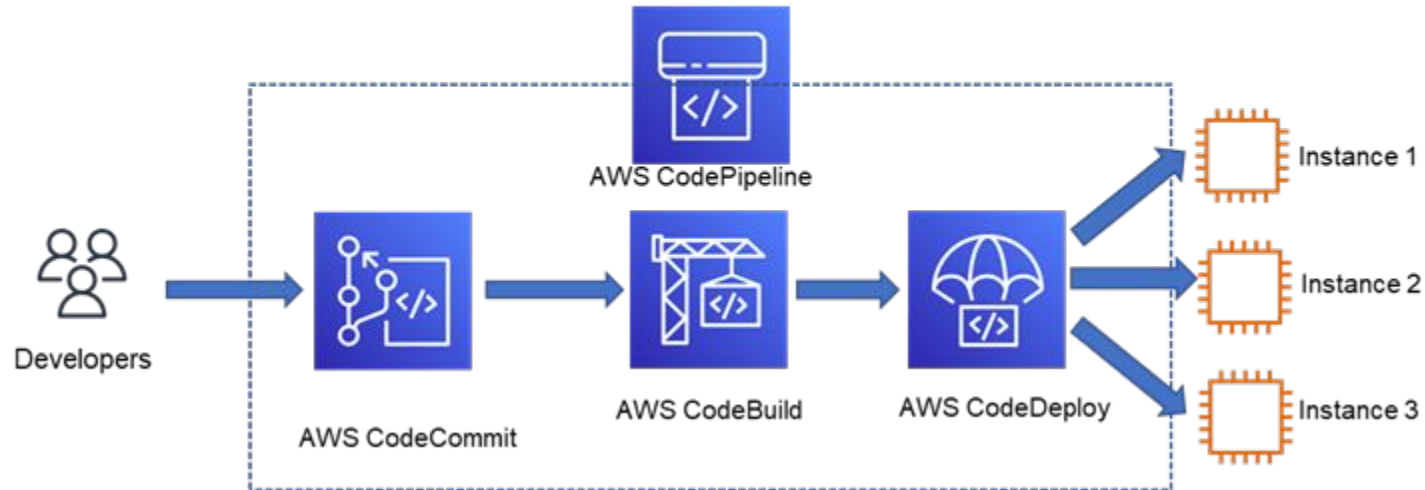|				
1		**INFRASTRUCTURE as code**	In the world of IT operations, infrastructure as code is responsible for on-boarding, provisioning, and licensing of infrastructure and services necessary to deploy and operate the deployment pipeline. This includes containers.	**TOOLS:**
2		**CONFIGURATION as code**	Once infrastructure is prepped, it must be configured. Configuration as Code uses declarative or imperative methods to automatically configure services in the deployment pipeline.	**TOOLS:** Puppet SALTSTACK
3		**PIPELINE as code**	The pipeline drives the deployment of an application from start to finish. This layer of the stack is what defines per-application pipelines and enables continuous operation of a comprehensive data path that secures and scales an application.	**TOOLS:**
4		**OPERATIONS as code**	Operations as code is concerned with post-deployment operations. This includes capabilities like auto-scaling, monitoring and alerting, and resource discovery. These are repeatable workflows (processes) that comprise a variety of operational tasks for which automation makes sense.	**TOOLS:**

Make Frequent, Small, Reversible Changes

Design workloads to allow components to be updated regularly.

Make changes in **small increments** that can be **reversed** if they **fail** (without affecting customers when possible).

Make Frequent, Small, Reversible Changes

Make frequent, small, reversible changes

Developers → AWS CodeCommit → AWS CodeBuild → AWS CodeDeploy → Instance 1, Instance 2, Instance 3

AWS CodePipeline

Make Frequent, Small, Reversible Changes

Company's Environments	
Scratch	Prod Simulation
Dev	Prod
UAT	

Refine Operations Procedures Frequently

As you use operations procedures, **look for opportunities to improve** them.

As you evolve your workload, **evolve your procedures appropriately**.

Set up regular game days to review and validate that all procedures are effective and that teams are familiar with them.

Refine Operations Procedures Frequently

Think OpsWorks - https://aws.amazon.com/opsworks/

Considerations:

    Chef & Puppet vs Alternatives like Ansible?

Anticipate Failure

Perform "pre-mortem" exercises to identify potential sources of failure so that they can be removed or mitigated.

Test your failure scenarios and validate your understanding of their impact.

Test your response procedures to ensure that they are effective, and that teams are familiar with their execution.

Set up regular game days to **test workloads** and **team responses** to simulated events.

Anticipate Failure

Game day on AWS: [Game day - AWS Well-Architected Framework (amazon.com)](#)

Anticipate Failure

Learn From All Operational Failures

Drive improvement through lessons learned from all operational events and failures.

Share what is learned across teams and through the entire organization.

Op Excellence Principles Summary

- Perform Operations as Code

- Make Frequent, Small, Reversible Changes

- Refine Operations Procedures Frequently

- Anticipate Failure

- Learn From All Operational Failures

Activity - Understanding Op Excellence

In this activity, gather into your own group and each group should take on one or two research problem.

Ensure all research problems are taken and presented by the end of this section.

Activity - Understanding Cost Optimization

- Perform Operations as Code

- Make Frequent, Small, Reversible Changes

- Refine Operations Procedures Frequently

- Anticipate Failure

- Learn From All Operational Failures

Best Practices For Operational Excellence

Summary

There are four best practice areas for operational excellence in the cloud:

- Organization
- Prepare
- Operate
- Evolve

Organization

Your teams need to have a **shared understanding of your entire workload**, their role in it, and shared business goals to set the priorities that will enable business success.

Well-defined priorities will maximize the benefits of your efforts.

Evaluate internal and external customer needs involving key stakeholders, including business, development, and operations teams, to determine where to focus efforts.

Organization

Evaluating customer needs will ensure that you have a **thorough understanding of the support that is required to achieve business outcomes**.

Ensure that you are aware of guidelines or obligations defined by your organizational governance and external factors, such as regulatory compliance requirements and industry standards, that may mandate or emphasize specific focus.

Validate that you have mechanisms to **identify changes** to **internal governance and external compliance requirements**.

Review your priorities regularly so that they can be updated as needs change.

Organization

Evaluate **threats to the business** (for example, business risk and liabilities, and information security threats) and maintain this information in a risk registry.

Evaluate the i**mpact of risks, and tradeoffs between competing interests or alternative approaches.**

For example, accelerating speed to market for new features may be emphasized over cost optimization, or you may choose a relational database for non-relational data to simplify the effort to migrate a system without refactoring.

Organization

Manage **benefits and risks** to make **informed decisions** when determining where to focus efforts.

Some risks or choices may be acceptable for a time, it may be possible to mitigate associated risks, or it may become unacceptable to allow a risk to remain, in which case you will take action to address the risk.

Prepare

To prepare for operational excellence, you have to **understand your workloads and their expected behaviors.**

You will then be able design them to provide insight to their status and build the procedures to support them.

Design your workload so that it **provides the information necessary for you to understand its internal state** (for example, metrics, logs, events, and traces) across all components in support of observability and investigating issues.

Prepare

Iterate to **develop the telemetry necessary to monitor the health of your workload, identify when outcomes are at risk, and enable effective responses.**

When instrumenting your workload, capture a broad set of information to enable situational awareness (for example, changes in state, user activity, privilege access, utilization counters), knowing that you can use filters to select the most useful information over time.

Prepare

Adopt approaches that **improve the flow of changes into production and that enable refactoring, fast feedback on quality, and bug fixing**.

These accelerate beneficial changes entering production, limit issues deployed, and enable rapid identification and remediation of issues introduced through deployment activities or discovered in your environments.

Prepare

Using these practices **mitigates the impact of issues introduced through the deployment of changes**.

Plan for **unsuccessful changes** so that you are able to respond faster if necessary and test and validate the changes you make.

Be aware of **planned activities in your environments** so that you can manage the risk of changes impacting planned activities.

Prepare

Emphasize **frequent, small, reversible changes** to limit the scope of change.

This results in **easier troubleshooting** and **faster remediation with the option to rollback a change**.

It also means you are able to get the benefit of valuable changes more frequently.

Operate

Successful operation of a workload is measured by the achievement of business and customer outcomes.

Define expected outcomes, **determine how success will be measured**, and **identify metrics** that will be used in those calculations to **determine** if your workload and operations are successful.

Operational health **includes both the health of the workload** and the **health and success of the operations activities** performed in support of the workload (for example, deployment and incident response).

Operate

Operate

Establish **metrics baselines for improvement, investigation, and intervention, collect and analyze your metrics**, and then **validate** your understanding of operations success and how it changes over time.

Use collected metrics to **determine** if you are satisfying customer and business needs, and identify areas for improvement.

Operate
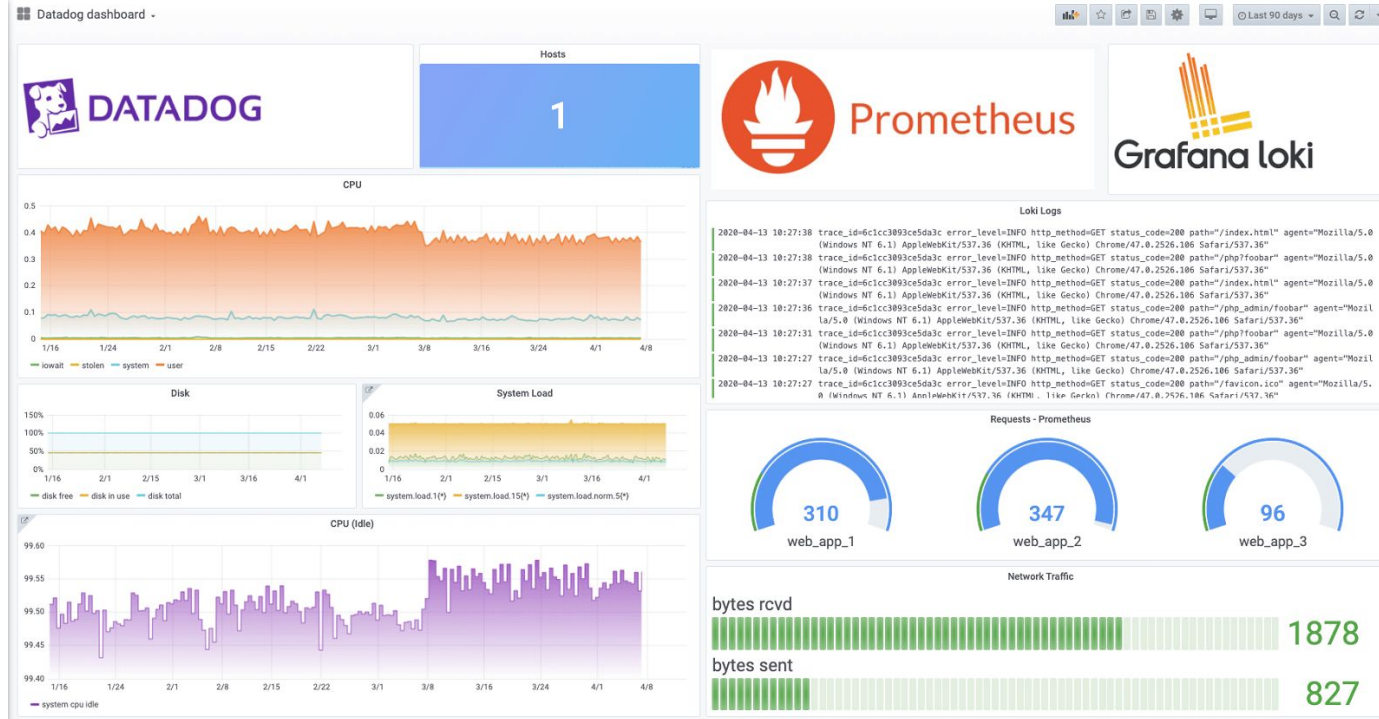
Operate

Efficient and effective management of operational events is required to achieve operational excellence.

Use **established runbooks for well-understood events**, and use playbooks to aid in investigation and resolution of issues.

Prioritize responses to **events** based on their business and customer impact.

Operate

Ensure that **if an alert is raised in response to an event**, there is an **associated process to be executed,** with a specifically identified owner.

Define in advance the **personnel required** to resolve an event and include escalation triggers to engage additional personnel, as it becomes necessary, based on urgency and impact.

Identify and engage individuals with the authority to make a decision on courses of action where there will be a business impact from an event response not previously addressed.

Operate

NANYANG TECHNOLOGICAL UNIVERSITY | SINGAPORE
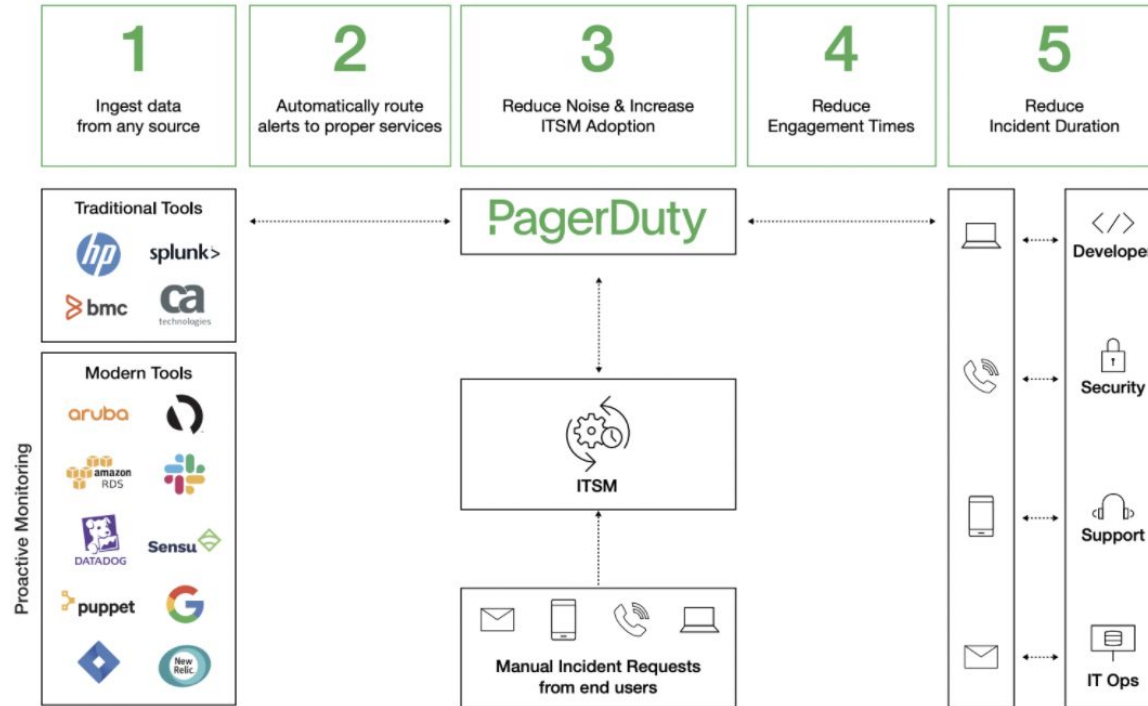
Evolve

You must learn, share, and continuously improve to sustain operational excellence. Dedicate work cycles to making continuous incremental improvements.

Perform post-incident analysis of all customer impacting events. Identify the contributing factors and preventative action to limit or prevent recurrence.

Communicate contributing factors with affected communities as appropriate.

Evolve

Evolve

Regularly **evaluate and prioritize opportunities for improvement** (for example, feature requests, issue remediation, and compliance requirements), including both the workload and operations procedures.

Include feedback loops within your procedures to rapidly identify areas for improvement and **capture learnings** from the execution of operations.

Evolve - Confluence

Evolve

Share lessons learned across teams to share the benefits of those lessons.

Analyze trends within lessons learned and **perform cross-team retrospective analysis of operations metrics** to identify opportunities and methods for improvement.

Implement changes intended to bring about improvement and evaluate the results to determine success.

Summary

There are four best practice areas for operational excellence in the cloud:

- Organization
- Prepare
- Operate
- Evolve

Activity - Understanding Cost Optimization

In this activity, gather into your own group and each group should take on one or two research problem.

Ensure all research problems are taken and presented by the end of this section.

Activity - Understanding Cost Optimization

- How do you determine what your priorities are?
- How do you know that you are ready to support a workload?
- How do you evolve operations?

Activity

Learner:

- Clean up AWS.
- Remove/delete/terminate all service/ resources that created.

Instructor

- Clean up AWS.
- Remove/delete/terminate all service/ resources that created.
- Check the AWS account after learner clean up.

What's Next?

NANYANG TECHNOLOGICAL UNIVERSITY | SINGAPORE