LEARNING UNIT 04

INFORMATION TECHNOLOGY SERVICE MANAGEMENT (ITSM):

INTEGRATING ITIL® AND DEVOPS

4.1 INTRODUCTION

This learning unit introduces you to Information Technology Service Management's (ITSM's) integrating ITIL® and DevOps based on the prescribed textbook, which is Kaiser, A.K. (2018). *Reinventing ITIL® in the Age of DevOps: Innovative Techniques to Make Processes Agile and Relevant*. New York: Apress. This book is freely available online from Unisa's e-library – Safari Books Online (O'Reilly) database at https://learning.oreilly.com/library/view/reinventing-itil-in/9781484239766/.

This learning unit analyses ITIL® and DevOps, compares and contrasts their strengths and weaknesses to demonstrate where and how they can be beneficially integrated and begins the process of integrating ITIL® and DevOps by identifying the ITIL® processes and phases that need to be changed and those that do not need to be changed. It also discusses how IT people are structured for successful outcomes since people are required to develop, manage and maintain IT systems, analyses the ITIL® and DevOps team structures and explains how they can be successfully integrated, examines how configurations are managed in a DevOps Project and explains incident, problem, change and release management adaption.
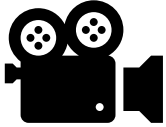

TAKE NOTE

Although Kaiser (2018) refers to the previous ITIL® version 3 structure and ITIL® is now on ITIL® version 4, this module focuses on those relevant aspects of ITL® 3 that remain in ITIL® 4 as practices and require careful integration to be effective. In addition, many organisations that you interact with may still have many aspects of ITIL® 3 embedded; therefore, you must understand how DevOps can be integrated with both ITL® 3 and ITIL® 4 processes and practices. Furthermore, ITIL® 4 states that it can be integrated with DevOps and Kaiser (2018) provides important knowledge about how to practically do this.
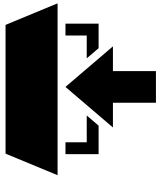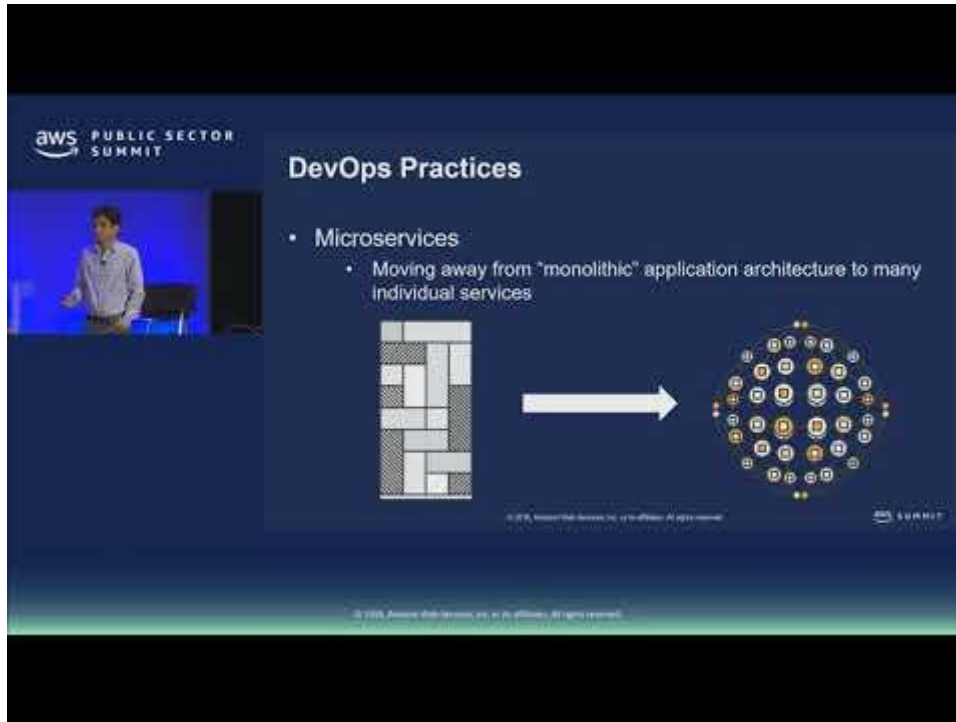

> **Watch this YouTube video:**
>
> Use the following link to watch a video about Amazon's DevOps culture: https://www.youtube.com/watch?v=mBU3AJ3j1rg (47:01)
>
> **Purpose of the video:**
>
> This video provides practical insight into how DevOps is practiced in an international technology giant.
>
>

4.2 PRACTICES/PROCESSES AND TEAM INTEGRATION

Let us look at how the relevant ITIL® 3 processes map to the ITIL® 4 practices. Following are the most relevant aspects for integration with DevOps. From the ITIL® 3 Service Strategy lifecycle, Strategy Management for IT Services, Service Portfolio Management and Financial Management for IT Services have the same or similar descriptions as ITIL® 4 practices. Demand Management is now part of Capacity and Performance Management in ITIL® 4 and Business Relationship Management is now part of Portfolio Management in ITIL® 4.

From the ITIL® 3 Service Design lifecycle, Availability Management, Service Catalogue Management, IT Service Continuity Management, Information Security Management and Supplier Management have the same or similar descriptions as ITIL® 4 practices. Design

Coordination is now part of Service Design in ITIL® 4, Service Level Management is now part of Service Design in ITIL® 4 and Capacity Management is now part of Capacity and Performance Management in ITIL® 4.

From the ITIL® 3 Service Transition lifecycle, Service Asset and Configuration Management, Service Validation and Testing and Knowledge Management have the same or similar descriptions as ITIL® 4 practices. Transition Planning and Support is now part of Project Management in ITIL® 4, Change Management is now part of Change Enablement in ITIL® 4, Release and Deployment Management is now part of Deployment Management in ITIL® 4 and Change Evaluation is now part of Change Enablement in ITIL® 4.

From the ITIL® 3 Service Operation lifecycle, Problem Management has the same or similar description as an ITIL® 4 practices. Event Management is now part of Monitoring and Event Management in ITIL® 4, Incident Management is now part of Service Desk in ITIL® 4, Request Fulfilment is now part of Service Desk in ITIL® 4 and Access Management is now part of Information Security Management in ITIL® 4.

The ITIL® 3 Continual Service Improvement lifecycle has the same or similar description as an ITIL® 4 practices.

Let us discuss these aspects now in terms of integration with DevOps, using the ITIL® 4 descriptions. Strategy Management (Strategy Management for IT Services ITIL® 3) end-to-end would not be required in DevOps before designing, developing, deploying and maintaining the new IT service. Portfolio Management (Service Portfolio Management ITIL® 3) should not be changed in DevOps since it is a crucial decision-making and due diligence practice to determine which services to build, the required investment and return on investment. Service Financial Management (Financial Management for IT Services ITIL® 3) would also be similar for ITIL® and DevOps. Capacity and Performance Management (Demand Management ITIL® 3) would require a lean version and input at each sprint planning session to continually assess any demand changes to be incorporated in subsequent DevOps sprints. Portfolio Management (Business Relationship Management ITIL® 3) would be fulfilled by the product owner's role in the DevOps team who provides input during scrum sessions.

Service Design (Design Coordination ITIL® 3) would be altered to accommodate uncertain design outcomes through flexible requirements and having multiple design options in DevOps. Service Catalogue Management (Service Catalogue Management ITIL® 3) has many benefits and would be used by DevOps in a similar way as it is in ITIL®. DevOps would use Service Design (Service Level Management ITIL® 3) in a similar way as it is in ITIL® and Availability Management

(Availability Management ITIL® 3) in a similar way as it is in ITIL® for the customer and DevOps teams. DevOps would use Capacity and Performance Management (Capacity Management ITIL® 3) in a similar way as it is in ITIL® and Service Continuity Management (IT Service Continuity Management ITIL® 3) in a similar way as it is in ITIL®. Information Security Management (Information Security Management ITIL® 3) would be altered to include DevSecOps or Rugged DevOps security practices to incorporate security throughout the development processes. DevOps would use Supplier Management (Supplier Management ITIL® 3) in a similar way as it is in ITIL®.

Project Management (Transition Planning and Support ITIL® 3) would be replaced by the DevOps Agile practices throughout. DevOps would use Service Validation and Testing (Service Validation and Testing ITIL® 3) in a similar way as it is in ITIL®, except that in DevOps there is enhanced testing automation capabilities. DevOps would alter Change Enablement (Change Evaluation ITIL® 3) by being part of the DevOps change and release management processes and Knowledge Management (Knowledge Management ITIL® 3) by being less formal with more code-based comments.

Due to the additional detail required for the following processes and practices, the following are discussed in Section 4.3 "CONFIGURATION, INCIDENT AND PROBLEM INTEGRATION": Service Configuration Management (Service Asset and Configuration Management ITIL® 3), Service Desk (Incident Management ITIL® 3) and Problem management (Problem Management ITIL® 3) and the following are discussed in Section 4.4 "CHANGE AND RELEASE INTEGRATION": Change Enablement (Change Management ITIL® 3) and Deployment Management (Release and Deployment Management ITIL® 3).

DevOps would use Monitoring and Event Management (Event Management ITIL® 3) in a similar way as it is in ITIL®, Service Desk (Request Fulfilment ITIL® 3) in a similar way as it is in ITIL®, and Information Security Management (Access Management ITIL® 3) in a similar way as it is in ITIL®.

DevOps would use Continual Improvement (Continual Service Improvement ITIL® 3) and the seven-step improvement process in a similar way as it is in ITIL®.

Generally, ITIL® structures its teams around their processes and practices where people with similar skillsets are grouped together (in types of knowledge or practice silos); for example, the database team or network team. Typically, people in these teams remain in these teams unless they are needed on a specific project where they are utilised and when the project is done (and the service or product developed is handed over to operations), they return to their original team.

This is contrasted with a DevOps team member who only belongs and remains in that DevOps team forever (for as long as a product or service is used). Moving to a DevOps team structure would mean a DevOps team (with development and operational team members) for each significant product or service, no hierarchical project structures with mini-hierarchies such as a development team, testing team and database team, scrum framework teams with flat hierarchies, joint and shared responsibility for delivery/non-delivery, extensive collaboration, no project manager, self-supervised, a scrum master and a product owner or customer representative who is the sole owner of the product backlog and priorities. In addition, DevOps development processes and cycles are consistent and predictable and enable customer expectation management.

A DevOps team typically has three groups. The first is the Scrum team, the second is the application management function relating directly to ITIL® and manages and maintains the application and the third are the team member focusing on the non-functional, such as IT security, tools automation and configuration and service management who connect the DevOps team and the customer organization. Outside of each DevOps team are shared teams that provide shared services to multiple DevOps teams, such as IT facilities management and IT infrastructure teams.

TAKE NOTE

Although Kaiser (2018) refers to the previous ITIL® version 3 structure and ITIL® is now on ITIL® version 4, this module focuses on those relevant aspects of ITL® 3 that remain in ITIL® 4 as practices and require careful integration to be effective. In addition, many organisations that you interact with may still have many aspects of ITIL® 3 embedded; therefore, it is important to understand how DevOps can be integrated with both ITIL® 3 and ITIL® 4 processes and practices. Furthermore, ITIL® 4 states that it can be integrated with DevOps and Kaiser (2018) provides important knowledge about how to practically do this.

Watch this YouTube video:

Use the following link to watch a video about balancing DevOps and ITIL®: https://www.youtube.com/watch?v=-3DgUxMtDRo (38:42)
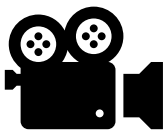
Purpose of the video:

To provide understanding about how to balance DevOps and ITIL® in the industry.

How can DevOps help ITIL®?

DevOps provides a culture shift to what ITIL® has been traditionally perceived as
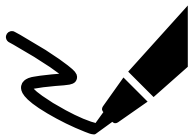
The Three Ways
1. Systems thinking
2. Amplify feedback loops
3. Culture of continual experimentation and learning

There are four main areas within DevOps that work well with ITSM:
1. Extend Development into IT Operations
2. Create IT Operations feedback into Development
3. Embed Development into IT Operations
4. Embed IT Operations into Development
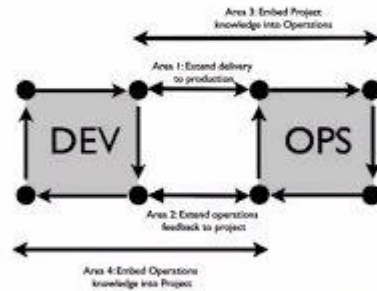
ACTIVITY 4.1: PRACTICES/PROCESSES AND TEAM INTEGRATION

Instructions:

- Go to the **Discussions** tool on the module's myUnisa website.
- Go to the Forum for Learning Unit 04: Specific Questions and Discussions.
- Open the discussion topic: Activity 4.1: Practices/processes and team integration.
- Provide your insight on the questions/tasks that are outlined below. Type your answer to each question in the discussion forum post using 300 words or less.

Purpose:

To provide an opportunity to start developing ideas about how to practically integrate DevOps and ITIL®.

Task:

Study chapters 4 and 5 of the textbook (Kaiser, 2018).

Provide explanatory responses to the following:

        a) In an organisation that you are familiar with (do not use actual names) think about one or two of the IT practices and processes that you have had the most interaction with.

b) Do these IT practices and processes seem to be more ITIL® or DevOps based?

c) What are the main problems evident in these IT practices and processes and do you think a move in the ITIL® or DevOps direction will be beneficial and why?
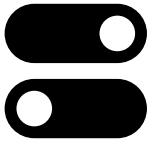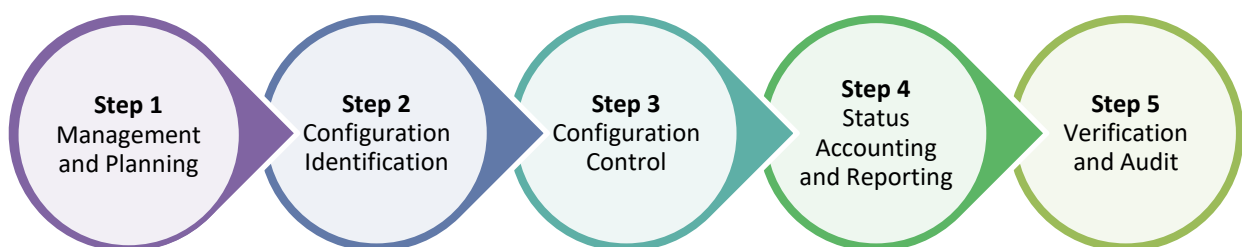
4.3 CONFIGURATION, INCIDENT AND PROBLEM INTEGRATION

Service Configuration Management (Service Asset and Configuration Management ITIL® 3) is the process and practice of creating a detailed and complete list of all the IT services and components and their interdependencies so that the impact (IT and business impacts) of any proposed change is easily and accurately visible and known and outages can be effectively managed. Without a detailed and complete list of all the IT services and components and their interdependencies, chaos will result from any change causing potentially huge financial losses, lawsuits and even irreparable damage to the business.

Service Configuration Management requires databases such as the Configuration Management System (CMS), Configuration Management Databases (CMDBs), Definitive Media Library (DML) and Definitive Spares (DS).

Service Configuration Management involves five general steps:

Step 1 Management and Planning → **Step 2** Configuration Identification → **Step 3** Configuration Control → **Step 4** Status Accounting and Reporting → **Step 5** Verification and Audit
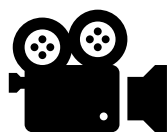
Configuration management is essential in DevOps. DevOps would use service Configuration Management in a similar way as it is in ITIL® with additional automation and Comprehensive Configuration Management (CCM) that includes a configuration management database, source code repository and artefact repository.

DevOps would use Service Desk (Incident Management ITIL® 3) in a similar way as it is in ITIL®, but the DevOps teams are regarded as level 3 support and only address incidents that relate to code and architectural changes, other minor incidents are resolved by the service desk team (level 1) or shared apps teams (level 2). In addition, DevOps incidents form part of the inputs into development sprints or if very urgent and high priority, get dealt with during a sprint and planned sprint items get moved to the next sprint, which supports the main objective of incident management, which is to minimise downtime resulting from service disruptions and has a great impact on customers' perceptions of the services.

DevOps would use Problem Management ITIL® 3 in a similar way as it is in ITIL®; however, a dedicated problem manager would be part of the shared team servicing many DevOps teams instead of a problem manager for each DevOps team. In addition, DevOps problem analysis, resolution design and implementation form part of the inputs into development sprints. Notably, problem management aims for prevention as opposed to incident management which aims for correction and a problem is defined as the underlying cause of an incident/s.
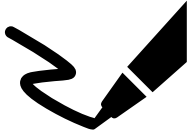
Watch this YouTube video:

Use the following link to watch a video about incident management in DevOps:
https://www.youtube.com/watch?v=NCnSV62nMbE (36:28)

Purpose of the video:

The video demonstrates how DevOps successfully addresses incident management.

WEBINAR

Effective Incident Management:
How to Improve DevOps Efficiency

November 18, 2020 | 2PM EST, 11am PST

Chris Overton
VP Engineering

Paul Rothrock
Customer Engineer

	ACTIVITY 4.2: CONFIGURATION, INCIDENT AND PROBLEM INTEGRATION

Instructions:

- Go to the **Discussions** tool on the module's myUnisa website.

- Go to the Forum for Learning Unit 04: Specific Questions and Discussions.

- Open the discussion topic: Activity 4.2: Configuration, incident and problem integration.

- Provide your insight on the questions/tasks that are outlined below. Type your answer to each question in the discussion forum post using 300 words or less.

Purpose:

To provide an opportunity for you to practically think about configuration, incident and problem management in DevOps.

Task:

Study chapters 6, 7 and 8 of the textbook (Kaiser, 2018).

Provide explanatory responses to the following:

a) Have you experienced a severe system outage in an organisation that you are familiar with (do not use actual names)?

b) How did the IT team react to the incident, was it fast/slow, chaotic/organised or engaging/dismissive?

c) Do you think there are any DevOps ways of working that could have helped?
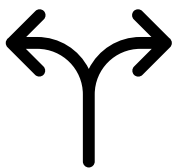
4.4 CHANGE AND RELEASE INTEGRATION

Generally, change management oversees, conducts due diligence, communicates with all stakeholders and authorises or denies all proposed changes to IT services. Change management ensures that all changes follow a set approval process, which is carried out by the change advisory board (CAB). Change management's objective is to enable beneficial changes to be implemented with the least disruption to the IT services.

Change Enablement (Change Management ITIL® 3) would be altered from a sequential process to an iterative process in DevOps, with automated testing, code reviews and deployment to

minimise change-related risks and improve quality. However, within each DevOps iteration, the change management process is sequential, but given the far smaller size and scale of DevOps changes, change management is not cumbersome and obstructive and the changes are far less risky. In DevOps, the change management adaption for continuous delivery requires that change management be initiated before coding begins and approval obtained through a formal CAB and change management provides final approval before deployment. In DevOps, the change management adaption for continuous deployment also requires that change management be initiated before coding begins and approval obtained through a formal CAB. However, the CAB, before coding begins, also provides approval for deployments if the testing is successful. Nevertheless, projects outside of DevOps developments, like datacentre migrations require a full sequential change management process.

Deployment Management (Release and Deployment Management ITIL® 3) would be altered from a sequential process to an iterative process in DevOps, with automation. In DevOps, the release planning, build and test would be iterative, while deployment, review and closure would be iterative (in continuous deployment DevOps) and sequential (in continuous delivery DevOps).
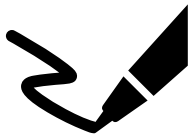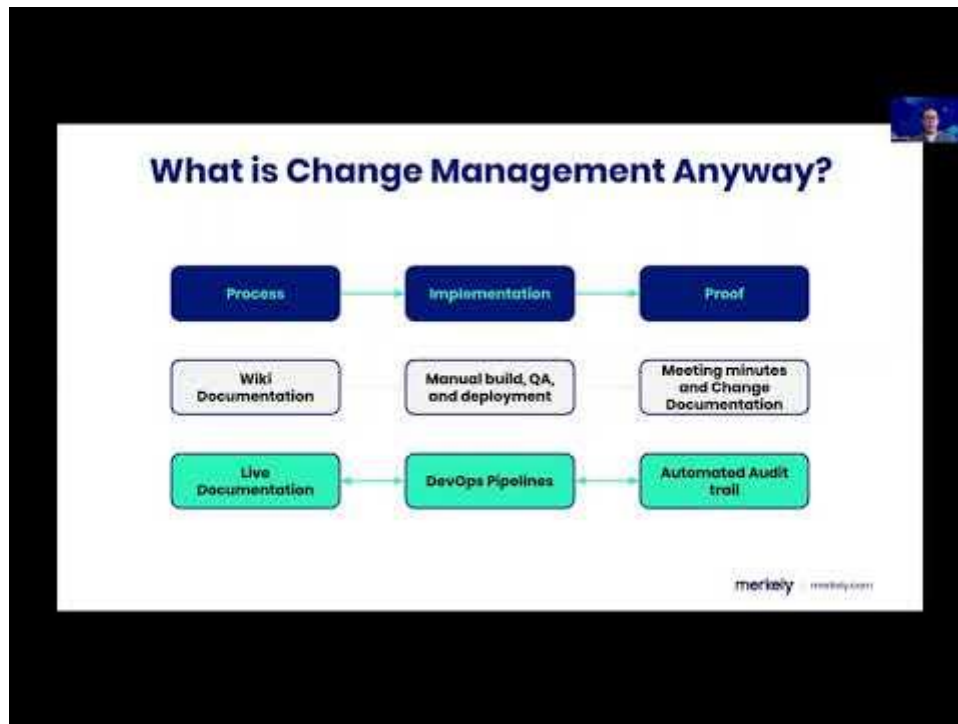
Watch this YouTube video:

Use the following link to watch a video about DevOps and change management: https://www.youtube.com/watch?v=feK9hkJ9p2E (29:18)

Purpose of the video:

This video explains how change management has changed in DevOps and the reasons.

ACTIVITY 4.3: CHANGE AND RELEASE INTEGRATION

Instructions:

- Go to the **Discussions** tool on the module's myUnisa website.
- Go to the Forum for Learning Unit 04: Specific Questions and Discussions.
- Open the discussion topic: Activity 4.3: Change and release integration.
- Provide your insight on the questions/tasks that are outlined below. Type your answer to each question in the discussion forum post using 300 words or less.

Purpose:

To provide an opportunity for you to reflect on change and release integration with DevOps.

Task:

Study chapters 9 and 10 of the textbook (Kaiser, 2018).

Provide explanatory responses to the following:

   a) In an organisation that you are familiar with (do not use actual names), what is your experience and observations of IT system changes?
   b) How could DevOps improve those experiences and observations and why?

4.5 CONCLUSION

This learning unit introduced you to Information Technology Service Management's (ITSM's) integrating ITIL® and DevOps based on the prescribed textbook (Kaiser, 2018).

After reviewing all the topics and completing all the activities, you should be able to create your own summary of the learning unit and use it during your preparation for the examination. In addition, you should now understand how to analyse ITIL® and DevOps together, demonstrate where and how they can be beneficially integrated and begin the process of integrating ITIL® and DevOps by identifying the ITIL® processes and phases that need to be changed and those that do not need to be changed. You should also be able to apply how IT people are structured for successful outcomes since people are required to develop, manage and maintain IT systems, analyse the ITIL® and DevOps team structures and explain how they can be successfully integrated, examine how configurations are managed in a DevOps Project and explain incident, problem, change and release management adaption. We trust that this learning unit significantly improved your understanding of integrating ITIL® and DevOps.

References

Kaiser, A.K. (2018). *Reinventing ITIL® in the Age of DevOps: Innovative Techniques to Make Processes Agile and Relevant*. New York: Apress. https://learning.oreilly.com/library/view/reinventing-itil-in/9781484239766/ (Freely available ONLINE from Unisa's e-library – Safari Books Online (O'Reilly) database).