

Deployment management

ITIL®4 Practice Guide

[AXELOS.com](https://www.axelos.com)

Contents

1	About this document	3
2	General information	4
3	Value Streams and Processes	11
4	Organizations and people	17
5	Information and technology	23
6	Partners and suppliers	25
7	Important reminder	27
8	Acknowledgments	28

1 About this document

This document provides practical guidance for the deployment management practice. It is split into five main sections, covering:

- general information about the practice
- the practice's processes and activities and their roles in the service value chain
- the organizations and people involved in the practice
- the information and technology supporting the practice
- considerations for partners and suppliers for the practice.

1.1 ITIL® 4 QUALIFICATION SCHEME

Selected content of this document is examinable as a part of the following syllabus:

- ITIL Specialist: Create, deliver and support
- ITIL Specialist: High Velocity IT

Please refer to the syllabus documents for details.

2 General information

2.1 PURPOSE AND DESCRIPTION

Key message

The purpose of the deployment management practice is to move new or changed hardware, software, documentation, processes, or any other component to live environments. It may also be involved in deploying components to other environments for testing or staging.

The deployment management practice is responsible for moving a service or service component into a designated environment. This practice enables the deployment or removal of service components from or to different environments, including development, integration, live, production, test, or staging environments.

The practice is usually applied to digital and physical IT components, including software, hardware, documentation, licences, and data, within the agreed scope of environments controlled by the organization.

2.2 TERMS AND CONCEPTS

2.2.1 Environments

The deployment management practice enables the transition of products, services, and service components between the environments.

Definition: Environment

A subset of the IT infrastructure that is used for a particular purpose.

A service component's lifecycle may vary depending on its type and the sourcing approach. The number and purpose of controlled environments within the organization may also vary. Table 2.1 provides a list of example environments for an organization that develops software.

Table 2.1 List of example environments for an organization that develops software

Environment	Purpose
Development/Integration	Developing and integrating software
Test	Testing service components
Staging	Testing releases including products, services and other configuration items
Live/Production	Delivering IT services to service consumers

For products and components sourced outside the organization, development environments can be out of the organization's control. For products and services delivered to service consumers outside of the organization, control over the live environment can be limited. Other variations are possible.

2.2.2 Continuous integration, continuous delivery, and continuous deployment (CI/CD)

The key concepts for deployment in Agile and DevOps are:

- **Continuous integration** Integrating, building, and testing code within the software development environment.
- **Continuous delivery** Continuous delivery means that built software can be released to production at any time. Frequent deployments are possible, but deployment decisions are taken case by case, usually because organizations prefer a slower rate of deployment.
- **Continuous deployment** Changes go through the pipeline and are automatically put into the production environment, enabling multiple production deployments per day. Continuous deployment relies on continuous delivery.

These approaches are supported by the software development and management, service validation and testing, deployment management, infrastructure and platform management, and release management practices. These practices involve specific skills, processes, procedures, automation tools, and agreements with third parties. They enable the continuous pipeline for integration, delivery, and deployment. This would also affect the design of other practices, such as service configuration management, monitoring and event management, incident management, and others.

2.3 SCOPE

The scope of the deployment management practice includes:

- the effective transition of products, services, and service components between controlled environments, such as the development, live, test, and staging environments.
- the effective removal of products, services, and service components from designated environments.

These additions, modifications, and removals can be part of authorized changes/releases triggered by:

- new/changed service requirements
- new features/releases
- technical and operational changes
- third-party change requirements
- service retirements and removals
- support/troubleshooting
- service requests.

Several activities and areas of responsibility are not included in the deployment management practice, although they are still closely related to deployment. These are listed in Table 2.2, along with references to the practices in which they can be found. It is important to remember that ITIL practices are merely collections of tools to use in the context of value streams; they should be combined as necessary, depending on the situation.

Table 2.2 Deployment-related activities described in other practice guides

Activity	Practice guide
Authorizing changes/releases	Change enablement
Making services and components in the live environment available to users	Release management
Developing software	Software development and management
Developing and building infrastructure components	Infrastructure and platform management
Preparing and maintaining target environments for deployments	
Providing IT assets to be deployed	IT asset management
Maintaining authorized repositories of service components	
Testing and validating services and service components	Service validation and testing
Naming, versioning, and controlling the service components	Service configuration management

2.4 PRACTICE SUCCESS FACTORS

Definition: Practice success factor

A complex functional component of a practice that is required for the practice to fulfil its purpose.

A PSF is more than a task or activity, as it includes components of all four dimensions of service management. The nature of the activities and resources of PSFs within a practice may differ, but together they ensure that the practice is effective.

The deployment management practice includes the following PSFs:

- establishing and maintaining effective approaches to the deployment of services and service components across the organization
- ensuring the effective deployment of services and service components in the context of the organization's value streams.

2.4.1 Establishing and maintaining effective approaches to the deployment of services and service components across the organization

The deployment management practice includes defining and agreeing a model or several models to use when deploying products, services, and components. These models may use one deployment approach or combine deployment approaches, depending on their specific services and requirements and the sizes, types, and impacts of the service components that are being deployed.

Models can be defined for deploying services or service components of similar types. Such deployment models could be defined based on several factors, including:

- automation considerations
- costs/resource limitations
- expected frequency of the deployments
- rate of customer requirements change
- rate of technology change
- risks of components flaws
- source of the components
- user adoption behaviours and preferences
- visibility of the technology change to service consumers

Based on these and other relevant considerations, organizations define a set of models for the deployment of different service components. These models may describe different solutions in all four dimensions of service management. Table 2.3 outlines some example models.

Table 2.3 Example models for the deployment of different service components.

Deployment model applicability	Organizations and people	Information and technology	Value streams and processes	Partners and suppliers
Hardware components of services provided to external service consumers	A service provider should arrange a delivery team for the transportation and installation of the components	A range of tools can be used to automate the procurement, invoicing, user communication, and scheduling of the installation of hardware	An installation order can be triggered by new or changed value streams that include clear authorizations to procure and install new hardware	Third-party shipping, delivery, and installation service providers can be employed, as agreed between the parties
Hardware components of services obtained from a vendor	According to the delivery and installation clause in the vendor contract, the responsibilities for obtaining hardware and ensuring its correct	Vendor catalogues may be used for ordering the components, as well as to store and provide up-to-date installation manuals. A configuration management tool	Vendor activities, such as invoicing and shipping, should be accounted for during the value stream design; interfaces between parties need to be founded in the contracts	

installation should be should be populated
clearly defined with documentation
supplied with the
hardware, including
records and
documents, such as
warranty certificates,
maintenance
schedules, and so on

Software components of services provided to external service consumers	The service provider can have staff perform roadshows to service consumers to promote new software components and facilitate change awareness	An automated deployment toolset is utilized to make software available for use or ordering	Service providers can implement additional controls before a component is deployed, such as quality assurance, security, or commercial; it is crucial to account for such controls in partially- or fully-automated deployment pipelines	Partners can be engaged in deployment, such as bespoke testing of the software made available by the vendor prior to its deployment to the consumer environment.
Software components of services developed in house	DevOps teams are likely to perform the deployment of software	The continual integration and continual deployment pipeline toolset can be used to deploy software to a controlled environment	Service provider organizations have to establish organizational controls over the course of deployment, ensuring that controls are not excessive	Third parties can action some steps of the deployment model; for example, manual environment configuration activities

Deployment models also define the flow of deployment through controlled environments, responsibilities of the involved parties, triggers for deployment, and interactions with other practices' activities in the context of value streams.

These models may be flexible enough to adapt to changing circumstances, such as the scale, urgency, or complexity of the deployment.

Deployment models, and the deployment management practice in general, should be a subject to continual improvement with an aim to eliminate waste and increase effectiveness and efficiency.

2.4.2 Ensuring the effective deployment of services and service components in the context of the organization's value streams

Ensuring effective deployment requires orchestrating resources in all four dimensions of service management.

The effectiveness and efficiency of the deployment is significantly dependent on, and can be considerably impacted by, the availability of the relevant resources, skills, technology, tools and infrastructure. The effective use of technology and automation in deployment can improve the consistency, agility, and efficiency of the practice.

For changes/releases to be successful, it is crucial that the changed/released service's or service component's integrity is maintained throughout the transition process. Any unauthorized change through manual, process, or technology errors can negatively impact the objectives and outcomes of the changes and releases, often significantly impacting the organization.

The success of service transitions depends on the effective and efficient management of changes and releases, which in turn depends on timely deployments that align with requirements and objectives. Alignment of the deployment to the change and release requirements, as well as key aspects such as schedule and cost, must be managed effectively.

2.5 KEY METRICS

The effectiveness and performance of the ITIL practices should be assessed within the context of the value streams to which each practice contributes. As with the performance of any tool, the practice's performance can only be assessed within the context of its application. However, tools can differ greatly in design and quality, and these differences define a tool's potential or capability to be effective when used according to its purpose. Further guidance on metrics, key performance indicators (KPIs), and other techniques that can help with this can be found in the measurement and reporting practice guide.

Key metrics for deployment management practice are mapped to its PSFs. They can be used as KPIs in the context of value streams to assess the contribution of deployment management to the effectiveness and efficiency of those value streams. Some examples of key metrics are given in Table 2.4.

Table 2.4 Examples of metrics for the practice success factors

Practice success factors	Key metrics
Establishing and maintaining effective approaches to the deployment of services and service components across the organization	<ul style="list-style-type: none">● Level of stakeholders' satisfaction with the rate of change of products and services supported by deployments● Rate of adoption of the agreed approach to deployment across the organization● Level of key partners' and service consumers' alignment with deployment approaches● Number of audit findings and external compliance issues caused by deployments

Ensuring effective deployment of services and service components in the context of the organization's value streams	<ul style="list-style-type: none">● Level of stakeholders' satisfaction with lead time to deploy● Percentage of successful deployments/number of deployment errors/failures● Number/percentage of incidents related to deployments● Timeliness/adherence to deployments schedule● Deployment backlog throughput● Level of stakeholders' satisfaction with quality of deployments
---	---

The correct aggregation of metrics into complex indicators will make it easier to use the data for the ongoing management of value streams, and for the periodic assessment and continual improvement of the deployment management practice. There is no single best solution. Metrics will be based on the overall service strategy and priorities of an organization, as well as on the goals of the value streams to which the practice contributes.

3 Value Streams and Processes

3.1 VALUE STREAM CONTRIBUTION

Like any other ITIL management practice, the deployment management practice contributes to multiple value streams. It is important to remember that a value stream is never formed from a single practice. The deployment management practice combines with other practices to provide high-quality services to consumers. The main value chain activities to which the practice contributes are:

- Obtain and build
- Design and transition

The contribution of the deployment management practice to the service value chain is shown in Figure 3.1.

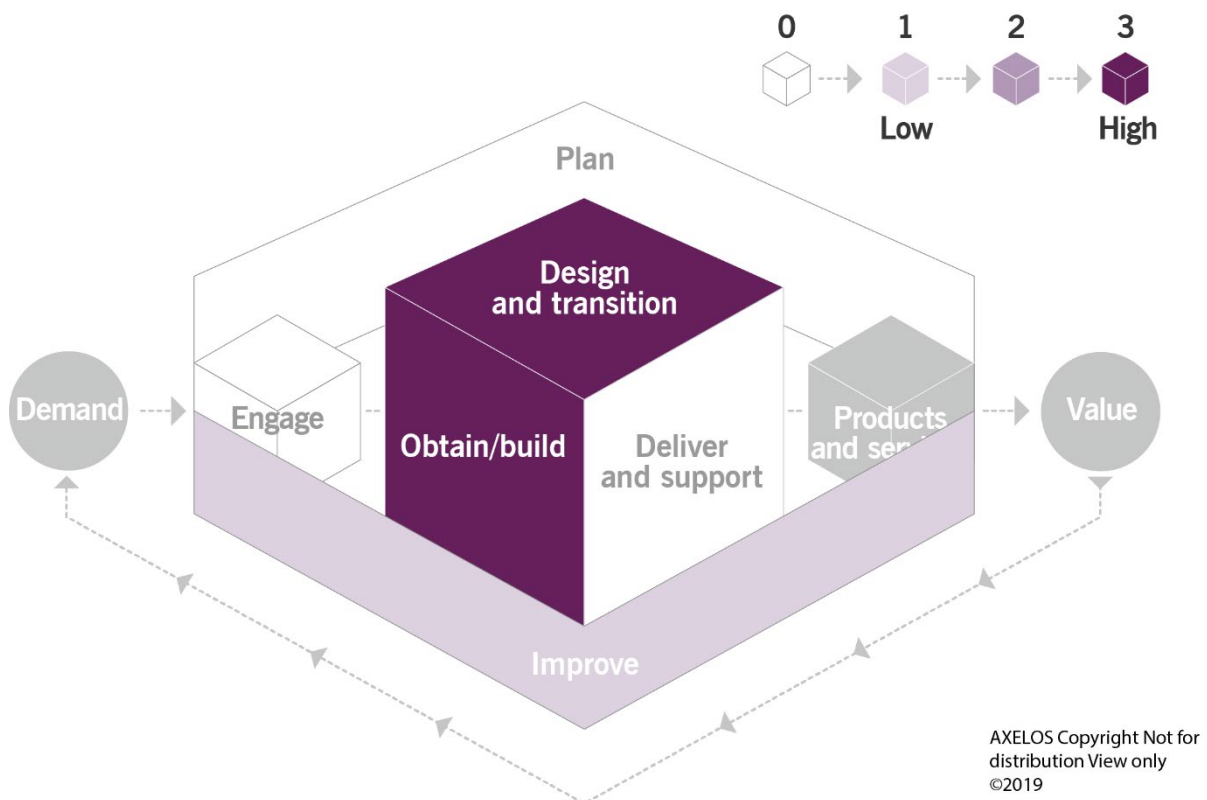


Figure 3.1 Heat map of the contribution of the deployment management practice to value chain activities

3.2 PROCESSES

Each practice may include one or more processes and activities that may be necessary to fulfil the purpose of that practice.

Definition: Process

A set of interrelated or interacting activities that transform inputs into outputs. A process takes one or more defined outputs and turns them into defined outputs. Processes define the sequence of actions and their dependencies.

Deployment management activities form two processes:

- deployment
- deployment models development and review.

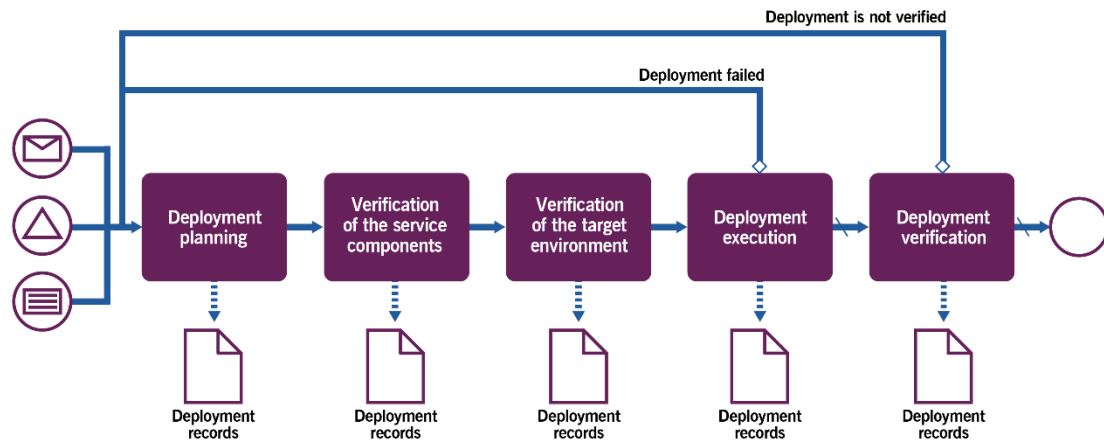
3.2.1 Deployment process

This process includes the activities listed in Table 3.1 and transforms the inputs into outputs.

Table 3.1 Inputs, activities, and outputs of the deployment process

Key inputs	Activities	Key outputs
Deployment requirements and expectations	Deployment planning	Deployed service components/releases
Environment details	Verification of the service components	Deployment records
Service component/release components	Verification of the target environments	Deployment communications
Hardware and software components from the authorized repositories of ITAM and definitive media library	Deployment execution	Feedback and inputs to change enablement, release management, service validation and testing, project management, etc.
Acceptance criteria	Deployment verification	Updates to onboarding procedures, customer knowledge base, service desk data

Figure 3.2 shows a workflow diagram of the process.



AXELOS Copyright Not for
distribution View only ©2019

In an Agile or DevOps environment that has adopted a CI/CD framework, many of these activities will be performed in an automated fashion, without manual intervention.

Table 3.3 provides examples of the process activities.

Table 3.3 Activities of the deployment process

Activity	Manual deployment to a datacenter	Automated deployment of a software component
Deployment planning	After a trigger from deployment (often procurement or the change request initiator), the service provider will schedule the shipping, delivering, verification, storing, and installation of hardware components. This schedule will align with the priorities of other work units for the affected teams and other resources.	Deployments in automated pipelines are triggered by committing all of the necessary pieces of code to a branch of the development version control system that will contain software features that are prepared for deployment.
Verification of the service components	Upon receiving delivered components, the service provider checks the completeness of the inventory, including the documentation, and conducts basic quality checks before accepting the delivery.	The code in the appointed branch is deployed onto a suitable test environment, tested, and any issues are fixed directly in the branch. The 'deploy, test, fix, redeploy, retest' cycle continues until a pre-set quality threshold of automated tests is met.

Verification of the target environments	The item is delivered to the installation location, where it is installed with an aim of causing minimal disruption to the service users. The installation location should have sufficient power, back-up power, air-conditioning, and fire protection arrangements. It may be necessary to include target environment checks in the deployment plan.	For an Infrastructure as a Code solutions, the configuration of a virtual environment in which the software should be run also follows an automated pipeline and is deployed to the virtual resources alongside the software code.
Deployment execution	The service provider or an external supplier staff installs and activates the equipment according to the installation instructions, which may include intermediate checks.	Deployment to an environment is automated, but can include additional human interaction steps before the actual deployment to account for business, security, or other non-automated types of verification.
Deployment verification	<p>After the item has been installed, a series of tests is performed to confirm the equipment is functioning.</p> <p>The staff performing the installation notifies those who triggered the deployment of the deployment results.</p>	The version control system sends notifications to the change requestor, such as a product owner, when the deployment is complete.

3.2.2 Deployment models development and review process

This process focuses on the continual improvement of the deployment management practice, deployment models, and deployment procedures. It is either performed regularly or triggered by deployment failures which highlight inefficiencies and other improvement opportunities. Regular reviews may occur every three months or more frequently, depending on the effectiveness of the existing models and procedures.

This process includes the activities listed in Table 3.4 and transforms the inputs into outputs:

Table 3.4 Inputs, activities, and outputs of the deployment models development and review process

Key inputs	Activities	Key outputs
<ul style="list-style-type: none"> • Current deployment models and procedures • Deployment records • Deployment failure reports • Policies and regulatory requirements • Release information 	<ul style="list-style-type: none"> • Deployment model planning • Deployment model implementation • Deployment model testing 	<ul style="list-style-type: none"> • Updated deployment models and procedures • Deployment models and procedures update communications • Change requests

- Configuration information
- IT asset information
- SLAs with consumers and suppliers/partners
- Capacity and performance information
- Continuity policies and plans
- Security policies and plans
- Deployments review and deployment records analysis
- Deployment model improvement initiation
- Deployment model update and communication
- Improvement initiatives
- Deployment review reports
- Updated knowledge management articles
- Lessons learnt

Figure 3.3 shows a workflow diagram of the process.

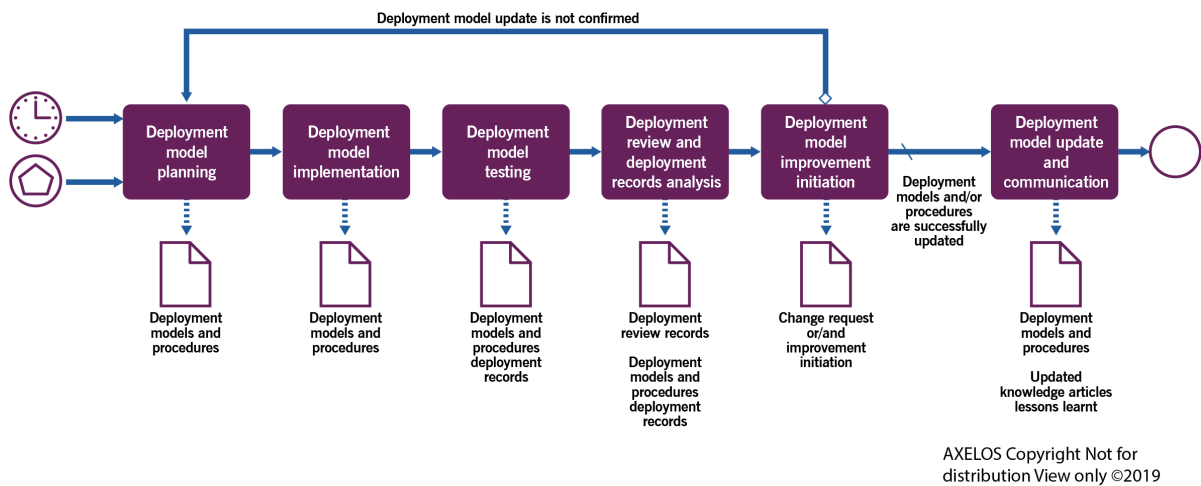


Figure 3.3 Workflow of the deployment models development and review process

Table 3.5 provides examples of the process activities.

Table 3.5 Activities of the deployment models development and review process

Activity	Description
Deployment model planning	When a product follows a similar low-risk, high-success-rate deployment pattern and there are means to eliminate waste and reduce deployment lead times, the deployment manager may choose to define a new deployment model. The deployment model should reduce the human involvement and control over the deployment.
Deployment model implementation	The deployment manager arranges for appropriate pipeline tools to be configured to support the new model, such as access settings, code support, or branching procedures. Alternatively, if automated deployment tools are not applicable, the deployment manager establishes and communicates adequate guidelines to the teams and parties involved.
Deployment model testing	The deployment manager tests the new deployment model to ensure proper edge-case handling and workflow. Where testing is impossible, the deployment manager oversees the first of the model's live runs.
Deployments review and deployment failure records analysis	The deployment manager, together with service owners and other relevant stakeholders, performs a review of selected deployments or deployment failures. They identify opportunities for the optimization of deployment models and deployment procedures.
Deployment model improvement initiation	The deployment manager registers the improvement initiatives to be processed with the involvement of the continual improvement practice, or initiates a change request, if the deployment models and procedures are included within the scope of change enablement.
Deployment model update and communication	If the deployment model is successfully updated, it is communicated to the relevant stakeholders. This is usually done by the deployment manager and/or the service or resource owner.

4 Organizations and people

4.1 ROLES, COMPETENCES, AND RESPONSIBILITIES

The practice guides do not describe the practice management roles, such as practice owner, practice lead, or practice coach. They focus instead on specialist roles that are specific to each practice. The structure and naming of each role may differ from organization to organization, so any roles defined in ITIL should not be treated as mandatory, or even recommended. Remember, roles are not job titles. One person can take on multiple roles and one role can be assigned to multiple people.

Roles are described in the context of processes and activities. Each role is characterized with a competency profile based on the model shown in Table 4.1.

Table 4.1 Competency codes and profiles

Competency code	Competency profile (activities and skills)
L	Leader Decision-making, delegating, overseeing other activities, providing incentives and motivation, and evaluating outcomes
A	Administrator Assigning and prioritizing tasks, record-keeping, ongoing reporting, and initiating basic improvements
C	Coordinator/communicator Coordinating multiple parties, maintaining communication between stakeholders, and running awareness campaigns
M	Methods and techniques expert Designing and implementing work techniques, documenting procedures, consulting on processes, work analysis, and continual improvement
T	Technical expert Providing technical (IT) expertise and conducting expertise-based assignments

Two practice-specific roles may be found in organizations: deployment manager and deployment practitioner. These roles are often introduced in organizations where the number of deployments is high. In other organizations, these roles might be combined with, or assigned to, other roles carrying related responsibilities in development, operations, IT asset teams, and so on.

4.1.1 Deployment manager role

A deployment manager role calls for a strong knowledge of the organization's business, products and services, technology, platforms, frameworks, and processes. The role requires strong planning and project management skills and the ability and authority to coordinate teamwork. The competency profile for this role is LACM. This role is usually responsible for the planning, management, and coordination of deployment management as a practice as well as the deployment of individual releases, including:

- planning deployments

- ensuring the alignment of deployment plans with change/release plans, requirements, and objectives
- planning, coordinating, and ensuring the availability of the resources needed for the effective completion of deployments
- effectively managing overlaps or conflicts among multiple deployments
- implementing and maintaining effective control and governance to ensure the integrity of components throughout the deployment practice
- managing and/or ensuring effective interfaces between and coordination with other practices and stakeholders
- managing and optimizing deployment resources to ensure optimum levels of availability, capability, and capacity to manage deployments
- monitoring, reporting, analysing, and improving deployment performance against defined KPIs.

In more complex organizations, some of the deployment management responsibilities may be delegated to the role of deployment coordinators, team leaders, or any other similar additional roles.

4.1.2 Deployment practitioner role

A deployment practitioner role calls for strong technical skills and effective teamwork. The competency profile for this role is TAC. This role is usually responsible for effective deployments to the target environments in alignment with applicable requirements, objectives, and targets, including:

- acquiring, maintaining, and continually improving the skills and capabilities required for technical aspects of deployments
- contributing and assisting in deployment planning
- ensuring the integrity of components throughout the deployment practice
- managing and coordinating deployment documentation, records, and communications, including for training purposes
- coordinating with other practices and stakeholders and facilitating interfaces between groups
- verifying and providing feedback on deployments to stakeholders
- contributing to monitoring, reporting, analysing, and improving deployment performance against defined KPIs.

In some organizational contexts, the deployment practitioner role can be divided into multiple categories and levels based on the types and requirements of the deployments and platforms, the complexity of organization's products and services, and so on.

4.1.3 Roles involved in the deployment management activities

Examples of other roles which can be involved in the deployment management activities are listed in Table 4.1, together with the associated competency profiles and specific skills.

Table 4.1 Examples of roles with responsibility for deployment management activities

Activity	Responsible roles	Competency profile	Specific skills
Deployment process			
Deployment planning	Service owner	ACMT	Understanding the deployment's impact on the service levels, user experiences, and environments
	Product owner		Good communication and cross-team coordination skills
			Good knowledge of deployment models
	Development team member		Understanding of technical service design, supporting infrastructure and platforms, development tools
	Technical specialist		
	Service desk agent		
	Engagement manager		
	Delivery manager		
	Users		
Verification of the service components	Technical specialist	T	Good knowledge of services and components
	Deployment manager		
	Development team member		
	Service owner		
	Product owner		
Verification of the target environments	Technical specialist	TC	Good knowledge of environments and infrastructure
	Deployment manager		

	Development team member		
	Systems administrator		
	Infrastructure team member		
	Service owner		
	Product owner		
Deployment execution	Technical specialist Deployment manager	TM	Understanding of technical service design, supporting infrastructure and platforms, development tools
	Development team member		Good knowledge of deployment models
	Systems administrator		
	Infrastructure team member		
Deployment verification	Technical specialist Deployment manager	TC	Understanding of technical design of services and components
	Development team member		Good knowledge of service performance, service levels, and user experience
	Systems administrator		
	Infrastructure team member		

Service
owner

Product
owner

User

Deployment models development and review process

Deployment model planning	Deployment manager	CAT	Understanding of the service design, resource configuration, and business impact
	Service owner		Good knowledge of existing deployment activities
	Product owner		
Deployment model implementation	Deployment manager	TCL	Knowledge of deployment pipeline tools
	Service owner		Knowledge of the continual improvement and change enablement practices
	Product owner		
Deployment model testing	Deployment manager	TCL	Good knowledge of testing practices across the workflows
	Service owner		Good knowledge of requirements and commitments, service levels
	Product owner		Knowledge of deployment models and methods; analytical skills
Deployment review and deployment records analysis	Deployment manager	TCL	Understanding of the service design, resource configuration, and business impacts
	Service owner		Good knowledge of deployment models
	Product owner		Good knowledge of requirements and commitments, service levels
	Supplier		Knowledge of deployment models and methods; analytical skills

Deployment model improvement initiation	Deployment manager	TMC	Understanding of the service design, resource configuration, business impacts, and service levels
	Service owner		Good knowledge of deployment models, diagnostic tools, and methods
	Product owner		Knowledge of the continual improvement and change enablement practices

Deployment model update and communication	Deployment manager	CA	Knowledge of communication procedures and tools
	Service owner		
	Product owner		
	Service desk agent		

4.2 ORGANIZATIONAL STRUCTURES AND TEAMS

Designated deployment management teams are unusual, except in very large organizations with significant volumes and complexity of deployment. This role is often handled by the technical/operations teams.

In a DevOps environment, deployment is often automated through the continual deployment practice/framework with use of deployment pipelines. However, the role of deployment manager is often still relevant; the deployment manager would own the overall practice and aspects around deployment. This role could be independently established or combined with other relevant and suitable roles, such as release manager.

5 Information and technology

5.1 INFORMATION EXCHANGE

The effectiveness of the deployment management practice is dependent on the quality of the information used. This information includes, but is not limited to, information about:

- authorized repositories of service components and assets, such as IT asset databases and DML
- assets and configurations
- change and release plans
- deployment communications
- deployment documentation and records
- deployment plans
- deployment metrics and reports
- entry, exit, and acceptance criteria for each stage of deployment
- feedback from deployment
- issues and errors identified during deployment
- platforms and environments within deployment's scope
- products and services and their architecture and design
- requirements and expectations about changes and releases
- stakeholder needs, expectations, and contact details.

This information may take various forms. The key inputs and outputs of the practice are listed in Section 3.

5.2 AUTOMATION AND TOOLING

In most cases, the deployment management practice can significantly benefit from automation. Deployments in Agile and DevOps environments are predominantly automation- and technology-oriented.

Where automation is possible and effective for deployment, it may involve the solutions outlined in Table 5.1.

Table 5.1 Automation solutions for deployment-management activities

Process activity	Means of automation	Key functionality	Impact on the effectiveness of the practice
In traditional, non-CI-CD environments			
Planning the deployment	Planning tools	Activity planning, scheduling, and tracking	Improved visibility, control, and governance over deployments

Verification of the service components	Service component/release verification using tools/technology	Ability to compare the components on various parameters	Improvement in accuracy and efficiency of verification leading to improved success rate, reduced reworks, quality and overall efficiency of deployments
Verification of the target environment	Platform verification using tools/technology	Ability to check the target platform(s) against set of parameters and attributes	
Deployment execution	Deployment/retirement using tools/technology	Ability to deploy the designated service components/releases to target environment(s) in a scheduled and controlled manner	Improvement in overall effectiveness, efficiency, and consistency of deployments
Deployment verification	Verification of deployments using tools/technology	Ability to verify the deployment and deployed service components against defined acceptance criteria	Improved verification of deployments
In CI/CD environments			
Automated deployments to dev, test, test, staging, and production	Integrated CI/CD tool chains	Schedule/trigger-based, automated deployment of the required components to target environments at each stage.	Effective integration of the release/transition stages for seamless build, integration, testing, and deployment.

6 Partners and suppliers

6.1 SOURCING CONSIDERATIONS FOR DEPLOYMENT PRACTICE

Very few services are delivered using only an organization's own resources. Most, if not all, depend on other services, often provided by third parties outside the organization (see section 2.4 of *ITIL Foundation: ITIL 4 Edition* for a model of a service relationship). Relationships and dependencies introduced by supporting services are described in the ITIL practices for service design, architecture management, and supplier management.

It is important to understand how the organization depends on third-party components and how it aims to establish effective and efficient collaboration with its key suppliers and partners around many activities, including those of the deployment management practice.

In an environment with multiple suppliers, it is important to understand the scope and boundaries of each organization's deployment activities, and how these will interact. Most organizations have a process for deployment, which is often supported by standard tools and detailed procedures to ensure that software is deployed consistently. It is common to have different processes for different environments.

Many areas of the deployment management practice might be enabled by effective sourcing, which could be in terms of people, capabilities, tools, processes, and services.

Deployment management and its PSFs can be enabled and enhanced through selective and judicious sourcing in many forms, including those outlined in Table 6.1.

Table 6.1 Sourcing in the deployment management practice

Sourcing area	Details
People	Where deployment management activities are manual, resources could be sourced from a partner. Key considerations include the schedule of deployments, availability of internal resources, cost, and so on.
Technical/Non-technical skills and capabilities	Sourcing specific skills, including technical (about specific systems, technologies, platforms) and non-technical (planning, governing, and execution capabilities), are useful or even required in many deployment management activities. Key considerations include the variety and complexity of technical/service environments, dynamic technology environments, lack of appropriate internal resources, and so on.
Outsourced deployment management	In certain contexts, it may be necessary or useful to source the entire deployment management practice from a partner.
Tools and technologies for deployment	Several areas of the deployment management practice can be enhanced through the adoption of tools and technologies. Except in minor cases, these technologies, tools, and tool-chains are sourced from specific product/service providers.

7 Important reminder

Most of the content of the practice guides should be taken as a suggestion of areas that an organization might consider when establishing and nurturing their own practices. The practice guides are catalogues of topics that organizations might think about, not a list of answers. When using the content of the practice guides, organizations should always follow the ITIL guiding principles:

- focus on value
- start where you are
- progress iteratively with feedback
- collaborate and promote visibility
- think and work holistically
- keep it simple and practical
- optimize and automate.

More information on the guiding principles and their application can be found in section 4.3 of *ITIL Foundation: ITIL 4 Edition*.

8 Acknowledgments

AXELOS Ltd is grateful to everyone who has contributed to the development of this guidance. These practice guides incorporate an unprecedented level of enthusiasm and feedback from across the ITIL community. In particular, AXELOS would like to thank the following:

8.1 AUTHORS

Vinod Kumar Agrasala, Roman Jouravlev, Konstantin Naryzhny.

8.2 REVIEWERS

Jon Hall, Anton Lykov, Samantha Robertson, Oleg Skrynnik.

