ITIL v3

Service Design & Service Transition

Phases of Lifecycle in ITIL

1. **Service Strategy** : Describes business goals and customer requirements and how to align objectives of both entities.
2. **Service Design** : Outlines practices for the production of IT policies, architectures and documentation.
3. **Service Transition :** Advises on change management and release practices, and also guides admins through environmental interruptions and changes.
4. **Service Operation :** Offers ways to manage IT services on a daily, monthly and yearly basis.
5. **Continual Service Improvement :** Covers how to introduce improvements and policy updates within the ITIL process framework

Service Design

- **Service Design** provides a blueprint for the services. It not only includes designing of a new service but also devises changes and improvements to existing ones.
- IT Services designed to meet business objectives.
- Services designed to be both fit for purpose and fit for use.
- Cost of ownership planned to achieve return on investment.

Design Coordination

- It deals with maintaining policies, guidelines, standards, and budget for service design activity.
- The central principles in design coordination are balance, prioritization and integration with project management.
- Balance and prioritization address the utility and warranty of a service, as well as the needs of the service throughout its lifecycle.

Design coordination oversees all activity in the service design phase of the service lifecycle. Its aim is to ensure that a holistic, integrated approach is taken to the design of services. This is necessary because of the variety of disciplines involved in Service Design and the need to take a consistent approach.

Design coordination is accountable for the production of the service design package (SDP). The SDP is a comprehensive description of how a new or changed service is to be designed, built, tested, deployed, and operated. The SDP is the handoff from the Service Design phase to the Service Transition phase.

Design coordination handles managing resources needed by the Service Design phase of the lifecycle. This includes:

- Planning to ensure that adequate resources are available
- Design coordination is accountable for the production of the service design package (SDP)
- Scheduling access to resources among the many projects that may be in this phase at any one time

It is accountable for the performance and improvement of the Service Design phase of the lifecycle.

Service Catalogue Management

- This process is responsible for designing service catalogue containing service specific to the customer for which they are willing to pay.
- Service catalog management ensures that an accurate and up-to-date service catalog is available to all parties authorized to see it.

- All parts of IT Service Management, as well as customers and users, use the service catalog. Accuracy and availability are essential

---

- Service catalog management must work closely with service portfolio management as new services move from the pipeline into the catalog and older services are retired.
- It also helps define how services can be requested and what options are available (gold/silver levels, for instance). The service catalog should document all defined services.

The service catalog generally comprises two views:

- a business service view that is visible to the customer
- a technical service view that is visible only to IT personnel.

This enables the customer to choose services based on their business requirements. At the same IT personnel can use their view to determine what technical services they need to support a given business service.

Service Level Management

- The goal of this process is to ensure that quality of the services meet provisioned quality agreement

The service level management (SLM) process focuses on researching and understanding requirements. Areas include:

- Defining, negotiating, agreeing upon and documenting IT service targets
- Monitoring, measuring and reporting on how well the service provider delivered the agreed upon targets

---

- When targets are appropriate and met, then the business and IT have a better chance of becoming aligned.
- Agreed upon targets are often spelled out in service level agreements (SLAs). Monitoring, measuring and reporting on SLA's in this way provides close links to Continual Service Improvement (CSI).
- SLAs are agreements to provide specific services at a defined level of quality (warranty) for a specific price. SLAs typically need negotiation of agreements with other internal organizations (OLA's) or external suppliers (Underpinning Contracts).

Negotiating SLAs to ensure service commitments are met, service level management works with the following warranty processes:

- capacity management
- availability management
- security management
- service continuity management

Service level management is accountable for monitoring conformance to the SLAs and take action if there is a breach of the SLA. This means working with the service desk, incident management, and problem management.

Customer satisfaction is not determined only by SLA performance. Therefore service level management should meet with customers face-to-face on a regular basis. This helps to maintain a positive relationship address any concerns the customer may have.

Capacity Management

- Capacity Management ensures optimal and economic usage of existing resources and future capacity requirement planning.
- Responsible for ensuring that adequate capacity is available at all times to meet the agreed needs of the business in a cost-effective manner
- Includes component, capacity plan, capacity report, capacity management information system, and performance.

Capacity management activities

- Gather the data
- Design a service and reach agreement
- Build the service
- Operation

What is capacity management?

ITIL capacity management is responsible for ensuring that adequate capacity is available at all times to meet the agreed needs of the business in a cost-effective manner. The capacity management process works closely with service level management to ensure that the business' requirements for capacity and performance can be met. Capacity management also serves as a focal point for any capacity issues in IT Service Management. Capacity management supports the service desk and incident and problem management in the resolution of incidents and problems related to capacity.

Successful capacity management requires a thorough understanding of how business demand influences demand for services, and how service demand influences demand on components. This is reflected by the three subprocesses of capacity management: business capacity management, service capacity management, and component capacity management. It is required that capacity management develop a capacity plan, which addresses both current capacity and performance issues, as well as future requirements. The capacity plan should be used throughout IT Service Management for planning and budgeting purposes.

Capacity management is responsible for defining the metrics to be captured during service operation to measure performance and use of capacity. This includes monitoring tools, which can provide input to the event management process. Capacity management may be called upon to perform tactical demand management, which involves using techniques such as differential charging to change users' behavior so that demand does not exceed supply. Other activities of capacity management include sizing (working with developers to understand capacity requirements of new services) and modeling (building statistical representations of systems).

Capacity management definitions

Before implementing capacity management, it's important everyone is on the same page. One way for an organization to accomplish this is to learn and own the definition. Capacity management introduces new ideas and terms that should be discussed before they are implemented, **including component, capacity plan, capacity report, capacity management information system, and performance**.

A **component** is the underlying structure behind a service. For example, it is the database behind the application or the server underneath the website. It is a component that must be purchased, built, maintained, and monitored. Improving performance often involves a replacement, upgrade, or load balancing of the individual component.

The **capacity plan** contains different scenarios for predicted business demand and offers costed options for delivering the service-level targets as specified. This plan allows service designers to make the best choices about how to provide quality service at an affordable price point.

The **capacity report** is a document that provides other IT management with data regarding service and resource usage and performance. This is used to help other managers make service-level decisions or decisions regarding individual components.

The **capacity management information system (CMIS)** is the virtual repository used to store capacity data. Dashboards are one way to store and report on capacity data.

Performance is how quickly a system responds to requests. For example, how quickly an application processes data and returns a new screen is one indicator of its performance.

The purpose of capacity management

The purpose of capacity management is to determine *how much* capacity should be provided based on the information from demand management regarding what should be provided. In particular, capacity management is concerned with speed and efficiency. If IT capacity forecasts are accurate and the amount of IT capacity in place meets business needs, the capacity management process is a success.

Capacity management activities

This process involves constant measurement, modeling, management, and reporting. More specifically, these activities include:

- Designing a service so that it meets service-level agreement (SLA) objectives once implemented
- Managing resource performance so that services meet SLA objectives
- Assisting with the diagnosis of performance-related incidents and problems
- Creating and maintaining a capacity plan that aligns with the organization's budget cycle, paying particular attention to costs against resources and supply versus demand
- Continually reviewing current service capacity and service performance
- Gathering and assessing data regarding service usage, and documenting new requirements as necessary
- Guiding the implementation of changes related to capacity

In practice, implementing this from scratch would involve the same steps as for other projects. For example, implementation might follow these broad steps:

1. Gather the data Work with business to determine the service-level need. Determine what this means relative to service availability and service capacity. Identify the individual components necessary. Work with demand management resources to predict demand based on user roles. Work with the financial management team to determine the costs.
2. Design a service and reach agreement Once you've identified the services and the level of performance needed, the cost, and the expected demand, you'll be able to work with ITIL service level management to build an SLA that everyone can agree to. You will also have designed a service at this point.
3. Build the service The next step is to build the service. This involves purchasing the components and building the IT infrastructure, processes, and documentation necessary to support the new service/s. Capacity management should continue to monitor the business needs and any new data to ensure that the service being built will have the necessary capacity for quality performance. Financial management will be involved at this stage to facilitate purchasing of components and other resources.
4. Operation Once you have built the service, and everyone agrees it will meet demand, capacity, and availability requirements, it's go-live time. This is when service operation takes over. Capacity management then supports service operation to deliver services that meet targets.

Monitoring and managing services and their individual components are most easily done via monitoring dashboards that provide data on multiple components in one location. Gathering the data manually from each service or component adds to the total time it takes to produce service-capacity reports.

Capacity management processes

This process is built on several sub-processes, including **business capacity management, service capacity management, component capacity management, and capacity management reporting**. These processes share common activities, such as modeling, workload management, analysis, and optimization.'

Business capacity management is the sub-process that turns the needs of the business into IT service requirements. It is involved in service strategy and service design, reviewing the data to ensure that there will be not be any changes in demand before the IT service is implemented. This sub-process works with demand management to ensure that the service is meeting business needs. Other sub-processes make sure that the service meets service-level targets; this sub-process ensures that the service-level targets meet the business needs. A thorough understanding of the business and the service-level agreements is necessary to effectively perform the activities in this sub-process.

Service capacity management is the sub-process that focuses on the operation of the service. Unlike component capacity management, this process focuses solely on the service itself. It ensures that the end-to-end service provided meets agreed-upon service-level targets. For example, this process would monitor, control, and predict a ticketing system to ensure it was up and running efficiently.

Component capacity management focuses on the technology that provides the performance and capacity to the IT service. Components are things like hard disks, phones, and databases. This sub-process requires knowledge of how each component individually contributes to service performance. It manages, controls, and predicts performance usage and capacity of individual components rather than the service as a whole (as seen in service capacity management). The goal of this sub-process is to reduce the total amount of service downtime by monitoring current performance and predicting future performance. Component capacities are designed around service capacities and not the other way around.

Capacity management reporting is the final sub-process. It gathers and then provides other stages with the data related to service capacity, service usage, and service performance. The output of this sub-process is the service capacity report.

Capacity management and other ITIL processes

Capacity management must interface with other processes within ITIL, including demand management, availability management, service-level management, and financial management. When the business has a service need, it comes from demand management. It's then relayed to the business continuity management team, which then translates it into an SLA and capacity terms. Service-level management helps with this. Once the service is deployed, service capacity management and component capacity management come in to keep everything at peak performance. Availability management works hand-in-hand with capacity management to keep services running and prevent downtime. Financial management comes into play when individual components must be estimated, purchased, maintained, and replaced. Not working closely with financial management can result in either untimely drops in uptime or organizational budget losses.

Takeaways

ITIL capacity management is an important one. With it, your organization can save costs by having the data necessary to make decisions regarding service performance. Rather than being based on gut decisions and guesses, you can use gathered component data to make business cases that win over financial management. What's more, this process can identify where performance tuning is a better choice than upgrading, thereby saving the organization money. Other barriers, such as performance bottlenecks and early indicators of performance issues, are identified before they become problems. This maintains uptime and increases customer and end-user satisfaction.

Availability Management

- Availability Management ensures the operative services meet all agreed availability goals.
- Availability management ensures that infrastructure, tools, roles etc. are appropriate for the agreed targets.
- It also works with the design teams to ensure that availability is designed into services.

Part of the process is to identify vital business functions (VBFs) which IT services support. This will help clarify which approach to availability to take:

- prevention (making sure, as far as possible, that unavailability never happens)
- recovery (developing plans to restore service rapidly in the event of an outage)

- Availability management views availability from the user's perspective, from end to end. This means identifying single points of failure and designing resilience into any infrastructure supporting the service.
- Availability management serves as a focal point for all issues in IT Service Management related to availability.
- Availability management handles specifying which metrics to use to measure availability. And, monitors availability to ensure that the SLA targets are met.

IT Service Continuity Management

- This process ensures continuity of IT services regardless of any disaster occurs.
- IT service continuity management (ITSCM) focuses on supporting the overall continuity of the business.
- We define ITSCM as the process responsible for managing risks that could seriously impact IT services.

- Risks so serious they could threaten the very survival of the business.
- This activity is often referred to as disaster recovery (DR). But, the use of the term ITSCM should show that there is a corresponding business continuity management (BCM) process.
- This analysis determines how different types of disruptions impact the business. The business areas determined to suffer the greatest impact need the most focus from the service continuity teams.
- ITSCM is responsible for development and deployment of the service continuity plan.
- This includes regular testing and training of all personnel associated with the plan. ITSCM also works with change management to ensure that continuity plans are updated as the operational model changes.

---

4 Stages in the ITSCM lifecycle process

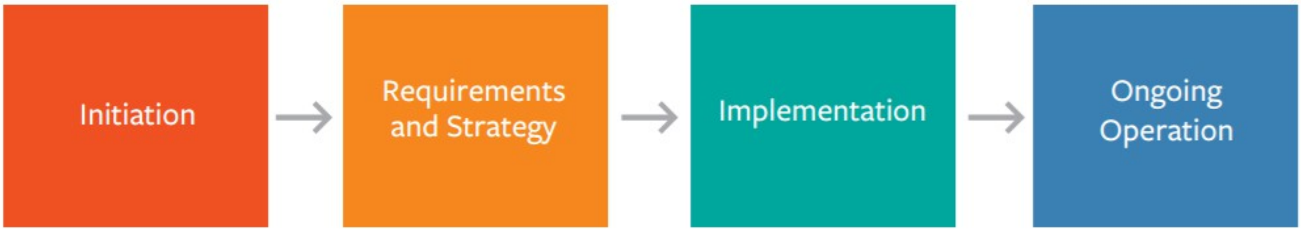


Figure 2. Four Stages of the ITSCM LIfecycle Process

Information Security Management

- This process ensures confidentiality, integrity, availability of data.
- Large organizations typically appoint a Security Manager who is accountable for the ISM process, end-to-end.
- Recommended security controls : *preventative* , *Detective* , corrective measures are in place.

Objectives of Information Security Management

- *Information is available and usable when required, and the systems that provide it can appropriately resist attacks and recover from or prevent failures (availability)*
- *Information is observed by or disclosed to only those who have a right to know (confidentiality)*
- *Information is complete, accurate, and protected against unauthorized modification (integrity)*
- *Business transactions, as well as information exchanges between enterprises or with partners, can be trusted (authenticity and non-repudiation*

ITIL information security management

Today, nearly every major company is in the technology business. Even the largest industrial and mining operations in the world depend heavily on complex IT services (and the hardware, software, networks, people, and processes that comprise them) to turn a profit.

More than ever, that means that IT has to be able to help the business manage risk, ensuring that resources are used responsibly and protected against potential threats or losses.

That's exactly the goal of ITIL Information Security Management, or ISM: to "align IT and business security and ensure that information security is effectively managed in all service and Service Management activities."

Unlike some ITIL processes that are invoked on an as-needed basis, security is not a single step in a service lifecycle. It's a continuous, integral need that requires stringent controls.

A few other helpful definitions as we dive further into ISM are:

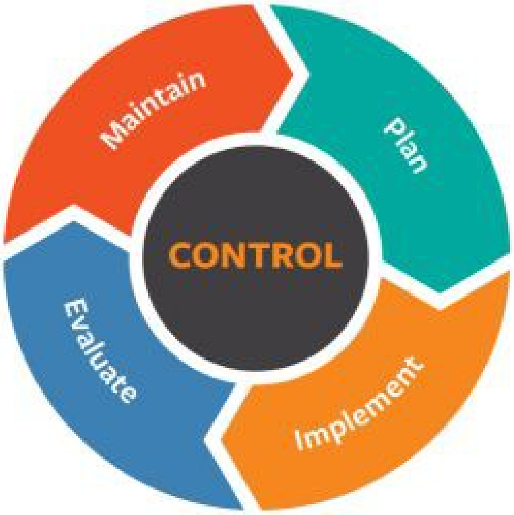
- **Information Security Policy** — An overarching security policy for your company that has the full support of top executive IT and business management. It should include separate policies for use and misuse of assets, access control, password control, email and internet, anti-virus, information classification, document classification, remote access, supplier access to your IT services and information, and asset disposal.

ITIL recommends that you make these policies widely available to all of your users and customers, and that you review and revise them at least every twelve months.

- **Information Security Management System (ISMS)** – This is just a wordy way of referring to the set of policies you put in place to manage security and risk across your company. The most important thing is that you take a calculated and comprehensive approach to designing, implementing, managing, maintaining and enforcing information security processes and controls. ITIL suggests that your ISMS should address what it calls "The Four P's": people, process, products and technology, and partners and suppliers.

Many global IT organizations seek global certification for their ISMS frameworks, which is done through ISO 27001. Typically, an ISMS framework addresses five key elements:

1. Control You should establish management framework for managing information security, preparing and implementing an Information Security Policy, allocating responsibilities, and establishing and controlling documentation.



2. Figure 3. Framework for Managing IT Security
3. Plan In the planning phase of the framework, you will be responsible for gathering and fully understanding the security requirements of the organization — then recommending the appropriate measures to take based on budget, corporate

culture around security, and other factors.

4. Implement Next, you'll put the plan into action, making sure that you have the proper safeguards in place to properly enact and enforce your Information Security Policy in the process.
5. Evaluate Once your policies and plans are in place, you need to properly oversee them to ensure that your systems are truly secure and your processes are running in compliance with your policies, SLAs, and other security requirements.
6. Maintain Finally, an effective ISMS means you are continuously improving the entire process — looking for opportunities to revise SLAs, security agreements, the way you monitor and control them, and more.

- A **security management information system** (or SMIS) is simply a tool or repository that stores data that supports your security management practices. It's part of your overall service knowledge management system (or SKMS). Ultimately, it should serve as the primary place for storing things like your security policies and plans, as well as all associated documents, measurements, and plans of action.

Who is responsible for Information Security Management?

Large organizations typically appoint a Security Manager who is accountable for the ISM process, end-to-end. Their job is to make sure that effective security policies are created, shared, and approved, and they are also responsible for overall security operations (from architecture and administration to recovery).

Key activities of Information Security Management (and thus responsibilities of the Security Manager), according to ITIL, include:

1. Creating (and revising as needed) an overall Information Security Policy for your company, and all necessary supporting policies.
2. Communicating, implementing, and enforcing these policies
3. Assessing and classifying all information assets and documentation
4. Implementing (and revising as needed) a set of security controls
5. Monitoring and managing all security breaches and major security incidents
6. Analyzing, reporting, and reducing the volume and impact of severity breaches and incidents
7. Scheduling and completing security reviews, audits, and penetration tests.

Recommended security controls

Because security is a continuous process, you should put in place a set of measures and controls that help minimize both threats and the impact of human errors. ITIL suggests five different types of measures:

First, *preventative* measures are designed to keep a security incident from happening altogether. Much of this practice is focused on access management tasks like assigning appropriate rights and permissions, verifying identification, and ensuring that unauthorized people cannot access your information and systems.

Reductive measures seek to reduce the impact of incidents that do occur, like putting contingency plans into place and testing them, for example, or performing automated backups of your critical data and systems.

Detective measures are exactly as they are named: controls put in place to identify a risk or threat as quickly as possible. This means putting the best possible monitoring systems in place — including network and systems monitoring tools, alerts, etc.

Repressive measures are like counterattacks. When a potential threat is detected, like a possibly malicious bot continuously trying to log in with an assortment of username and password combinations, automatically blocking further attempts from that IP address (or temporarily locking the usernames associated with the login attempts) is a great example of a repressive measure.

Finally, *corrective measures* seek to repair any damage caused by an error or incident. Restoring a backup is a top example.

Measuring your effectiveness

ITIL recommends a wide variety of metrics and KPIs you can use to keep track of how efficient and effective your ISM process and activities are. Key examples include:

- Percentage decrease in security breaches reported to your Service Desk, or in the impact of breaches and incidents
- Increase in support of your security procedures by senior management, and in conformance to your policies across the company
- The number of improvements suggested or made to your security procedures
- Increased awareness of your security policies across the organization

Key recommendations

First, ensure that you secure adequate support from senior leadership in the executive suite. Without buy-in, your efforts to create (and enforce) strong security policies could prove futile. To improve executive commitment, it's helpful to make sure your security strategy focused on business priorities and the IT services they count on, not just technology.

Second, don't forget to gather information from across your business as you create your security strategy and ISMS. Talking with and gathering data from every aspect of the organization is essential, to make sure you properly understand and address all risks, requirements, and priorities.

Don't forget the education, either. As you define your security requirements and create policies, making sure employees (and suppliers, etc.) are aware of them is critical. Setting proper expectations upfront will go a long way to widespread adoption and compliance.

Finally, remember that as your business evolves, so will the potential risks and security needs. Remember to regularly re-evaluate your policies and systems to ensure you are keeping up with new requirements (and even new threats from hackers as they become more sophisticated).

Supplier Management

- This process ensures supplier relationship & performance and also ensures management of right and relevant contracts with supplier
- Supplier management works with third parties, such as suppliers, to negotiate contracts for products or services.
- Supplier management monitor conformance to the contract conditions and address any breaches. At renewal, supplier management will determine whether to renew, renegotiate, or end the contract.

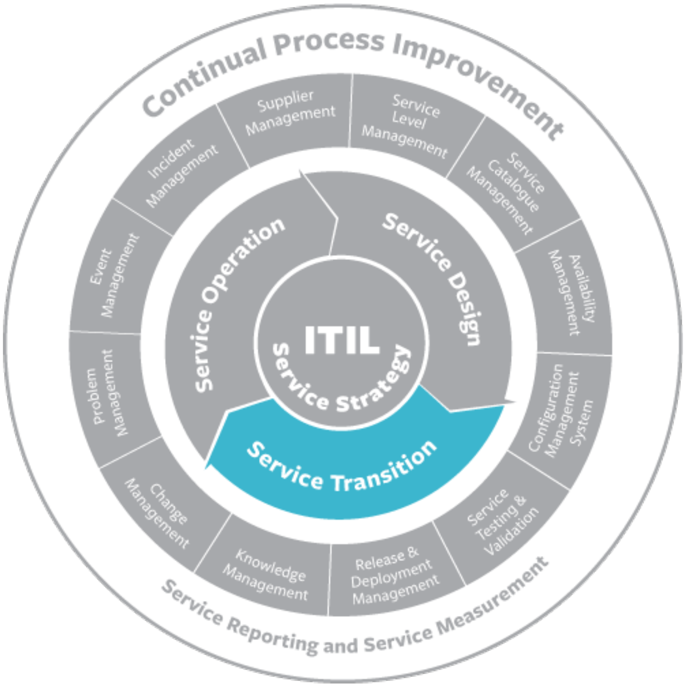
- The objectives of supplier management is to ensure alignment of contracts with the needs of the business.
- It is also responsible for ensuring suppliers are meeting their commitments. The supplier and contract management information system (SCMIS) holds supplier and contract details.

Service Transition

- Service Transition manages transition of a new or changed service. It ensures all changes to the service management processes are carried out in coordinated way

- Ensure that service can be managed, operated and supported in accordance with the requirements and constraints specified in service design

ITIL service transition helps plan and manage the change of state of a service in its lifecycle.



Managing risk for new, changed and retired services protects the product environment. This helps the business deliver value to itself and its customers.

Curating service knowledge helps all stakeholders make informed, reliable decisions and support challenges with service delivery. Both managing service risk and curating service knowledge are integral to service transition.

During service transition, the following organizational elements need support:

- Service strategy
- People
- Process
- Technology
- Suppliers of the service
- Organizational culture
- Governance
- Risk

No change is without risk. In fact, change can create extra risk. When transitioning services, focus on communication planning for awareness and compliance. One of the biggest challenges in service transition is changing people's behavior to accommodate a new or different service. People have a psychological need to feel safe and comfortable with changes to them and around them.

Transition Planning and Support

- This process deals with management and control of transition plan.
- At any time, there will be several projects passing through the service transition phase of the lifecycle.
- It is the responsibility of transition planning and support to coordinate service transition activities for all these projects.

Specifically, the responsibilities of transition planning and support include:

- Work with capacity management to ensure that adequate resources are available
- Where there is contention for resources, develop a schedule that meets the requirements of the stakeholders
- Ensure that all parties use a standard, reusable process framework.
- Monitor and improve the performance of the Service Transition lifecycle phase.

Change Management

This process ensures manage and control change management process. It also prevents any unauthorized changes from occurring.

ITIL change evaluation analyzes changes before they move to the next phase in their lifecycle. The lifecycle of a change includes several points at which a go/no-go decision needs to be made:

- Authorization to build and test
- Authorization to check software into the definitive media library (DML)
- Authorization to deploy

We should evaluate all changes. But, for significant changes a formal evaluation process should be invoked. Each organization must define for itself what "significant change" is.

The evaluation should include:

- Evaluating the intended effects of the change
- As far as possible, anticipating any unintended effects of the change
- Identifying risks
- Presenting a recommendation to change management on whether to proceed to the next stage

The change management process can make the go/no-go decision on proceeding to the next stage.

Service Asset and Configuration Management (SACM)

- It maintains database for configuration items such as servers, switches, routers etc.
- SACM is a combination of two important processes:
    - **Asset management** which addresses the assets you use to deliver IT services.
    - **Configuration management** which tracks the configurations of and relationships between the various components (configuration items or CIs) of your various IT services

According to ITIL, SACM is:

The process responsible for ensuring that the assets required to deliver services are properly controlled, and that accurate and reliable information about those assets is available when and where it is needed. This information includes details of how the assets have been configured and the relationships between assets.

This critical practice spans the entire service lifecycle. Nailing SACM is important to the health of both your individual services and your entire IT organization — and as such, it's often one of the first ITIL processes implemented in top IT organizations.

Goals of SACM

At its core, SACM is about ensuring that you are able to identify and control all assets across your infrastructure, and can manage their integrity through effective recording, reporting, and auditing.

More specifically, ITIL dictates that the objectives of SACM are to:

- Ensure that assets under the control of the IT organization are identified, controlled and properly cared for throughout their lifecycle.
- Identify, control, record, report, audit and verify services and other configuration items (CIs), including versions, baselines, constituent components, their attributes and relationships.
- Account for, manage and protect the integrity of CIs through the service lifecycle by working with change management to ensure that only authorized components are used and only authorized changes are made.
- Ensure the integrity of CIs and configurations required to control the services by establishing and maintaining an accurate and complete configuration management system (CMS).
- Maintain accurate configuration information on the historical, planned, and current state of services and other CIs.
- Support efficient and effective service management processes by providing accurate configuration information to enable people to make decisions at the right time — for example, to authorize changes and releases, or to resolve incidents and problems.

SACM definitions Before we dive deeper, it's helpful to get a few definitions out of the way that you'll see regularly throughout this process and beyond. There are a ton of definitions, but we'll limit it to the most important ones for a basic functional understanding of the ITIL Asset and Configuration Management.

Configuration Management System (CMS) A **Configuration Management System** is basically just a set of tools (like databases, files, etc.) that are used to gather, update, and analyze data about all of your configuration items and their relationships. A CMS may also include information about:

- Incidents
- Problems
- Known errors
- Changes
- Releases
- Sometimes even corporate data about employees, suppliers, locations and business units, customers, and users

This is different from a CMDB.

Configuration Management Database (CMDB) A CMDB is a database that stores configuration records. One or more CMDB's may be part of the overarching CMS.

Configuration records **Config records** describe your CIs by recording information about:

- Attributes, like name, location, version number, etc.
- Relationships among your CIs

Configuration items (CIs) CIs are simply any component that needs to be managed in order to deliver an IT service. A server, a virtual server, or even the configuration of an application could be considered a CI, for example.

There are different types of CIs, including:

- **Service lifecycle CIs** such as business cases, service management plans, service lifecycle plans, service design package, release and change plans, and test plans, etc.
- **Organizational CIs** like the organization's business strategy, regulatory requirements that need to be managed to, etc.
- **External CIs** such as external customer requirements and agreements, releases from suppliers or sub-contractors, and external services.
- **Interface CIs** that are required to deliver the end-to-end service across a service provider interface (SPI), such as an escalation document that specifies how incidents will be transferred between two service providers.

Service assets **Service assets** can be any resources or capabilities that contribute to the delivery of a service. Unlike CIs, which can be specifically managed by IT, service assets can't always. The knowledge of an IT worker, for example, is a service asset. The overall ability for a team to respond to an incident or problem is a service asset.

SACM scope If it's an asset that you use during the Service Lifecycle, it typically falls under the scope of SACM. Most things are fair game, including virtual assets, but there are some exceptions to the scope of SACM:

- Service assets that are not CIs are excluded from the scope. For example, the knowledge used by an experienced service desk agent to manage incidents.
- Assets that are not under the control of change management are excluded, such as information stored on a server.
- Non-IT assets in general are excluded. Everything else is fair game, including virtual assets.

In general, the scope of SACM includes management of the complete lifecycle of every CI — including interfaces to internal and external service providers where there are assets and configuration items that need to be controlled.

SACM benefits Benefits of service asset and config management include:

- Better cost management of services
- Improved planning and delivery of changes and releases
- More efficient resolution of incidents and problems, meeting SLAs more frequently
- Less risk of non-compliance to important legal, regulatory, and procedural standards

Along the way, service asset and configuration management will interface with nearly all of your other IT process, including change management, financial management, incident and problem management, and more.

Key activities of SACM There are six main activities in service asset and configuration management:

Planning

For each of the services you offer, ITIL suggests that you create a Service Management Plan — which is essentially just a document that addresses the critical components of a service before you implement it. It typically covers:

- The scope and objective of a service
- The activities and procedures (and even roles and people) required
- The relationship with other processes
- The tools and CIs also involved

Service Management Plans typically go into quite a bit more detail, though — outlining explicitly how the change management and configuration management processes will interface, how and when CI data will be audited, and much more.

Identifying

This is essentially creating a complete "inventory" of all the CI's in your infrastructure. In this activity, you are essentially recording every bit of data about your CIs that is necessary for effective operations — from the version of a piece of hardware or software to all the documentation, configurations, ownership details, etc.

To do so, ITIL recommends that you give all CIs **unique identifiers** (for example, numbers) and record all the relevant attributes of the CI (including the owner). Attributes you may want to capture include:

- The unique CI identifier and CI type. Every CI should have a unique identification number that makes it easily identifiable.
- The name and description of the CI
- Version numbers, since multiple versions of the same CI often exist
- Location and owner information, so you know where to find it
- Current status (ordered, in production, etc.)
- When necessary, supplier information, related documentation, etc.

Control

Here, ITIL recommends that all CIs follow a strict process for being added, changed, and removed from your CMS or CMDB. The goal is to ensure that changes of any kind don't occur without following your approved procedures for a wide variety of processes like license management, change management, version management, and even deployment.

Along the way, you will need to create your own policies and procedures for things like:

- **Controlling software licenses** to avoid noncompliance and prevent financial waste.
- **Controlling access** to facilities and systems.
- **Capturing the baseline of your assets and CIs before releases** to give you an accurate way to verify the success of actual deployments.

Status accounting and reporting

Throughout the lifecycle of every CI, ITIL encourages you to keep track of the complete status — including what changes have been proposed, the status of approved changes, etc.Being able to view and provide status reports gives you important insight into both the current and historical state of your CI's, and can even help you detect unauthorized CI's along the way.Reports typically include things like:

- An inventory of CIs and their baseline configurations
- An itemization of any unauthorized CIs
- Updates on recent changes or exceptions
- An itemization of hardware and software assets

Verifying and auditing

At any given time, you need assurance that your data is accurate on all of your CIs. Regular reviews and audits are essential, and ITIL recommends that you perform them to prevent discrepancies between your actual environment and how it is documented.

In ITIL terminology:

- **Verification** is an ongoing activity, consistently ensuring that the CMDB accurately reflects all of your CIs.
- **Auditing** is a more formal, occasional deep dive to confirm not only that records are accurate, but that processes are being followed and standards (including SLAs, etc.) are being met.

Verifications and audits can be performed anytime, both randomly and according to a planned schedule. Tools are available to automate the process, too — comparing the configuration of designated servers with the master configuration you've recorded, for example. Audits are also often performed before major changes or releases are deployed, to avoid potential incidents or service disruptions.

Managing information

Ensuring the integrity of your configuration and asset data and systems is equally important. As part of the asset and configuration management process, you need to regularly back up the CMS, keep detailed records about archived and historical CI versions, and take appropriate measures to ensure data integrity across the entire lifecycle.

SACM best practices

To get the most from your SACM activities, follow these best practices:

1. Always ensure that changes to every configuration item are authorized by change management, and that all changes and updates also modify the relevant configuration records accordingly.
2. Ensure that you have the right checks and balances in place to prevent unauthorized personnel from moving hardware assets, or making changes to any assets or CIs. When this happens, the CMS becomes out of date — and it's important that your records stay accurate.
3. Stay vigilant about performing regular verifications and occasional audits, too. The goal is to prevent the accuracy of your configuration records, etc. from diminishing over time — and that can happen when you stop paying attention to the process and the results.

Release and Deployment Management

- This process deals with management and control of movement of releases to test and live environment.
- Key phases to release and deployment management :
  - Release and deployment planning
  - Release building and Testing
  - Deployment
  - Reviewing and closing a deployment

- **Major releases.** To qualify as a major release, it should contain new hardware or software. More often than not, a major release equates to introducing completely new functionality. Think v1.0 and v2.0-level releases — they're major milestones.
- **Minor releases** make significant improvements to existing functionality, often packaging together a number of fixes — and are often numbered v1.1, v1.2, v1.3 etc.
- **Emergency releases** are exactly what they sound like. Something bad needs attention ASAP, so you're releasing a temporary fix — and probably numbering the release something like 1.1.1, v1.1.2, v1.1.3, etc.

But first — what's a release?

Simply put, a release (also called a release package) is a set of authorized changes to an IT service. That means a release can include hardware and software, documentation, processes, or other components that are essential to successfully implementing an approved change to your IT services.

Most ITIL-abiding organizations create **release policies**, which help to define how releases are numbered, how often they are released, and even how they are released (via a phased rollout versus a "big bang" rollout, for example.)

Your customers expect valuable services — and they expect them without disruption. That makes it critical that every single release be built, tested, and delivered following a rigorous process that ensures quality and minimizes risk.

ITIL's Release and Deployment Management phase is designed for exactly that: ensuring that your releases are deployed into production efficiently and effectively.

As part of your release policy, ITIL encourages creating a system for categorizing your releases.

As you design your releases, you will have several options for how you plan to deploy, as well. Your release policies may or may not specify which approach to take, but regardless, you should pay close attention to the users you are deploying to and the business activities the release will impact to ensure you have the desired impact.

- **Big bang** releases are deployed to all users, all at once.
- **Phased** approach releases are more paced — deployed first to a subset of the broader user base, then deployed more gradually to additional users as part of a scheduled rollout plan.

Releases can also be pulled or pushed. A pull approach means the release is placed in a central location where users can download it at their own will, while a push approach implies that the release would be pushed out from a central location to each user.

Finally, ITIL suggests that you clearly specify whether the release will be deployed automatically (i.e. using installation software, etc.) or manually.

Objectives of release and deployment management

According to ITIL, the objectives of Release and Deployment Management are:

- To create, test, verify, and deploy release packages
- To manage organization and stakeholder change
- To ensure that new or changed services are capable of delivering the agreed utility and warranty
- To record and manage deviations, risks, and issues related to the new or changed service and take necessary corrective action
- *To ensure there is knowledge transfer to enable customers and users to optimize use of services that support their business activities*

Release and deployment management scope

The scope of Release and Deployment Management includes all Configuration Items (CIs) that are required to implement a release, including:

- Virtual and physical assets
- Applications and software
- Training for staff and users
- All related contracts and agreements

Testing that is carried out as part of the Service Validation Process is not considered in scope, and neither is authorizing changes.

The key phases to release and deployment management

Release and Deployment Management is divided into four distinct phases, beginning with thoroughly planning what you will release and how, and concluding after the release is deployed and a post-mortem review is conducted.

Phase 1: Release and deployment planning

A well-thought-out release and deployment plan is just one component of your overall Service Transition plan — but the point is to clearly define a set of guidelines for both what a release will include and how you will deploy it into production. The release and deployment plan is then approved as part of the Change Management process.

At the beginning of the release and deployment-planning phase, change management typically authorizes the planning process to begin for a release. The plan typically addresses:

1. What changes the release will include
2. Who will be affected or impacted by the release
3. What risk the release may introduce, if any
4. The audience for the release (i.e. what users, customers, organizations will be impacted)
5. A clear chain of approval, clarifying which stakeholders may authorize the change request at every stage of the release
6. Ownership, defining the team who is ultimately responsible for the release.
7. Deployment schedule and strategy

It's quite common for building and test plans to also be built in this stage, clearly outlining critical logistics like design specifications, testing procedures, timelines for building and testing, and even pass / fail criteria at each phase of deployment. A pilot rollout may also optionally be planned at this time.

Phase 2: Release building and testing

Once a plan is in place and approved by change management, the responsible teams have to build and test the release, including both the software, documentation, and any other elements the release plan specifies.

At the beginning of this process, documentation is typically created to ensure that developers will be able to build the release package as accurately and efficiently as possible — and throughout the build process, accurate records should be kept so the build process can be repeated, if it becomes necessary.

Most organizations typically follow stringent procedures, or even provide standard templates for building a complete release package. Ensure you are utilizing and following these at every step of the way.

Testing happens throughout the process — from testing any and all input CIs, to testing and rehearsing the services before they are deployed live.

A few things to remember along the way:

- Pilots are a great way to identify and correct any issues with a service before they are deployed to the entire intended audience. This can dramatically reduce risk.
- Some teams also choose to stage a rehearsal, effectively "practicing" as much of a service rollout as possible shortly before a deployment is scheduled.

Phase 3: Deployment

In this phase, the release package is deployed to the live environment, beginning when change management authorizes the release package to be deployed to the target environments. The deployment phase ends with handoff to service operations and early-life support.

Before deploying, ITIL encourages quite a bit of advanced planning and preparation — confirming the target group is ready for the deployment, identifying and attempting to mitigate any potential risks or disruptions, and specifying the order of how each component of the release will be deployed (like financial assets, processes and materials, the actual service release, etc.)

Once a release is deployed, it's critical that you verify that it is operating properly for all stakeholders — and remediate or back out the release as needed should serious problems arise.

After verifying that the release is functioning as planned, ITIL calls for you to transition the new or changed service over to service operations in two stages. First, a formal notification should be issued that the service is now live (at the beginning of early life support or ELS), followed eventually by a formal notification that the service is fully operational and SLA's are being fully enforced.

Phase 4: Reviewing and closing a deployment

Once the release is deployed, it's time to review and learn from the entire process. Feedback is gathered, and evaluations are performed against performance goals — with the results being reviewed and discussed by all involved.

Reviews should be careful and thorough, confirming that all quality requirements have been met, that sufficient knowledge transfer and training were performed, and that any known errors, fixes, and changes have been adequately documented. Additionally, change management should conduct a full Post Implementation Review, or PIR.

A deployment isn't considered "closed" or completed until support has been formally transitioned to Operations.

Service validation and Testing

- This process deals with the quality of services offered.
- Testing can take place at any point in the service lifecycle but, it generally occurs during Service Transition.
- The service validation and testing process plans, conducts and reports on tests of new or changed services.
- The results of testing go to the change evaluation process to support a decision on whether to proceed.

- The service design package (SDP) outlines the tests to perform.
- Working with change evaluation, service validation and testing will:
  - Work with transition planning and support to plan the resources required for testing
  - Plan and design tests

- Schedule tests
- Prepare the test environment
- Perform the tests
- Evaluate exit criteria and report
- Clean up and close tests

Service validation and test will perform different types of tests, as called for in the service design package. Types of tests include:

- Utility testing. Does the service deliver the required functionality?
- Warranty testing. Will the service deliver required levels of availability, capacity, security, and continuity?
- Usability testing. Will the service be usable by all potential users, including those with restricted abilities?

- Contract and regulation testing. Will the service conform to applicable regulatory and contract requirements?
- Operational readiness testing. Are the support functions, including the service desk, staffed and trained to support the new or changed service?

Knowledge Management

This process deals with gathering, storing, analyzing, and sharing knowledge.

What is Knowledge Management? Knowledge management definition: The organization, capture, use, and analysis of the impact of a group's collective knowledge. In the business world, the definition of knowledge management also includes the maintenance of a knowledge base or portal where specific knowledge related to the company is housed. Only a few initiatives are able to truly transform how an organization operates, and knowledge management is one of them.

Why is knowledge management important? The knowledge management (KM) category represents solutions that streamline the process of capturing, distributing, and effectively using knowledge. When an organization is able to easily access, share, and update business knowledge, it can become more productive and cost-efficient. The ability to access the right knowledge at the right time, via a robust knowledge management system, informs accurate decision-making and stimulates collaboration and innovation.

A McKinsey Global Institute Report indicates that a robust knowledge management system can reduce information search time by as much as 35 percent and raise organization-wide productivity by 20 to 25 percent. Findings culled from the International Data Corp also corroborates the value of a knowledge management system, highlighting that Fortune 500 companies lose roughly $31.5 billion a year by failing to share knowledge.

As your enterprise grows, so too will the need to access a reliable knowledge database in order to effectively run your business, serve your clients, and increase revenue. Without a knowledge management system in place, your employees will be forced to learn and relearn processes and information. That's an inefficient and costly practice. Plus, you may also run the risk of losing those processes or information if a knowledge leader or legacy employee leaves your company.

What kind of information is captured in knowledge management? Information captured as part of knowledge management can include:

Documents

1. Company handbooks
2. Benefits breakdown
3. Product FAQs
4. Holiday calendars
5. Release notes
6. Team Data

Strategy

- Competitor briefs
- Product development timelines
- Presentation tactics
- Works in progress
- Best practices
- Organizational Data

Org charts

- Procurement flows
- Individual contract information
- Office location and contact information
- Brand information
- Organizational News

Company media mentions

- Information technology (IT) updates
- All-hands updates
- Upcoming promotions
- NPS scores and insights
- Promotion updates

Types of knowledge management Knowledge is one of your organization's most valuable assets. Storing, growing, and sharing that knowledge is critical to any enterprise.

When looking at it from this perspective, knowledge management's meaning includes the process that helps you acquire, organize, and transfer both explicit knowledge (knowledge that is easy to write down and share), implicit knowledge (applied knowledge), and tacit knowledge (knowledge gained from personal experience) throughout your organization. Learn more about the different types of knowledge management.

Knowledge management system examples Disseminating information throughout your organization is much easier when you have a reliable knowledge management platform to serve a full range of needs – both at a departmental level and a holistic, company-wide level.

Knowledge management platforms (or knowledge bases) are designed with best-in-class features to capture the information you need, verify and organize it, and make it easy to retrieve and share. So, if your definition of "knowledge management" has been limited to formalized assets, chances are your current solution doesn't account for the knowledge that sits below the surface.

Here are some great examples of knowledge management systems: **Document management**: These systems act as centralized digital filing cabinets for company documents. They make retrieving documents easy, support regulatory compliance, and enhance workflow. In addition, when a document management system is enhanced with passwords and backup procedures, document security is enhanced, but not thoroughly protected from outside access. Many typical document management systems have functionality limitations so custom upgrades can increase costs. This type of system does not automatically capture data or analyze it.

Content management: Content management systems are similar to document management systems, but store audio, video, and other media types in addition to documents.

Databases: A database is a computer application that allows people to capture, store, analyze, and interact with data. Databases are indexed in order to make information more accessible. Data stored in databases can be very secure because the system prohibits manipulation. However, they can be volatile and are often costly to design and set up. They also require a high level of skill to use and maintain.

Data warehouses: These enterprise-wide systems pull data from different parts of your organization and can be highly effective for reporting and analysis. They store current, as well as historic data and transform data into meaningful information. However, data warehouses are typically high-maintenance systems which require complex integration in order to provide a unified view of the data.

Intranets: These private computer networks built on searchable platforms can provide an easily accessible resource for information that enhances collaboration and social networking within your enterprise. But intranets do have some risks, including easy access by unauthorized personnel. In addition, they are costly and time-consuming to maintain.

Wikis: These web pages are easy-to-use collaborative tools that allow anyone to publish and store information in a central location. They can be good places to maintain business documents or product catalogues. However, because they can be openly edited, wikis can often include wrong information. In addition, they aren't optimized to show what information within them is being viewed or used or where knowledge gaps exist.

Social networking: Social networking allows people to connect with each other, join groups, contribute information, and discuss issues they are interested in. Social networking can influence organizational knowledge. Knowledge management systems can apply social networking to identify, document, and transfer knowledge.

Benefits of knowledge management The more effectively and efficiently a company shares its information with its employees, the better the business will perform. The benefits of knowledge management include:

- Faster decision-making
- Efficient access to knowledge and information
- Increased collaboration and idea generation

- Enhanced communication throughout your organization
- Improved quality of information and data
- More security for intellectual property
- Optimized training