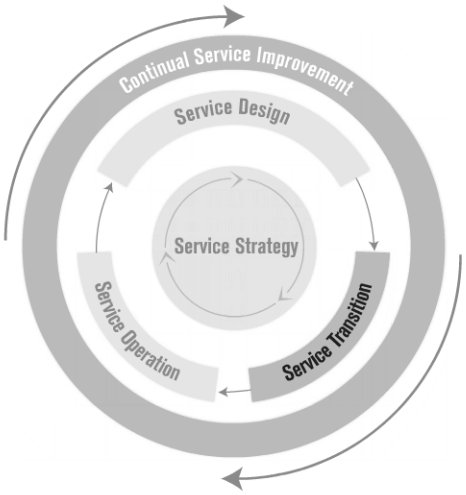
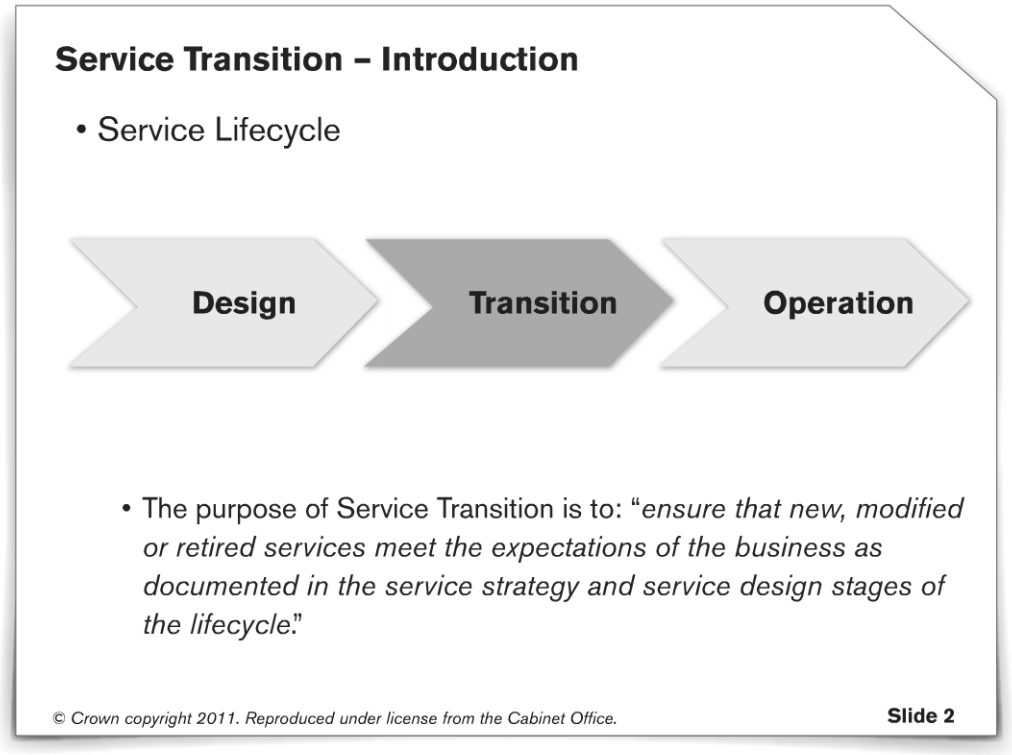
4
Service Transition





Service Transition – Introduction

Business environments are in a constant state of transition, and as such, so are the services required from IT. To ensure these services are being designed and implemented properly, an effective Service Transition process is critical.

Service Transition takes its shape and input from the strategy set by the organization, and also from the new (or changed) services it is attempting to bring into live operation (i.e., by the output of the Service Design stage). Therefore, its very nature is dependent upon its relationship with "upstream areas."

Before designing a new service (or redesigning an existing service), the vision for the service's purpose must be first agreed upon and accepted. Without a vision of the service's purpose, that service will always remain undelivered.

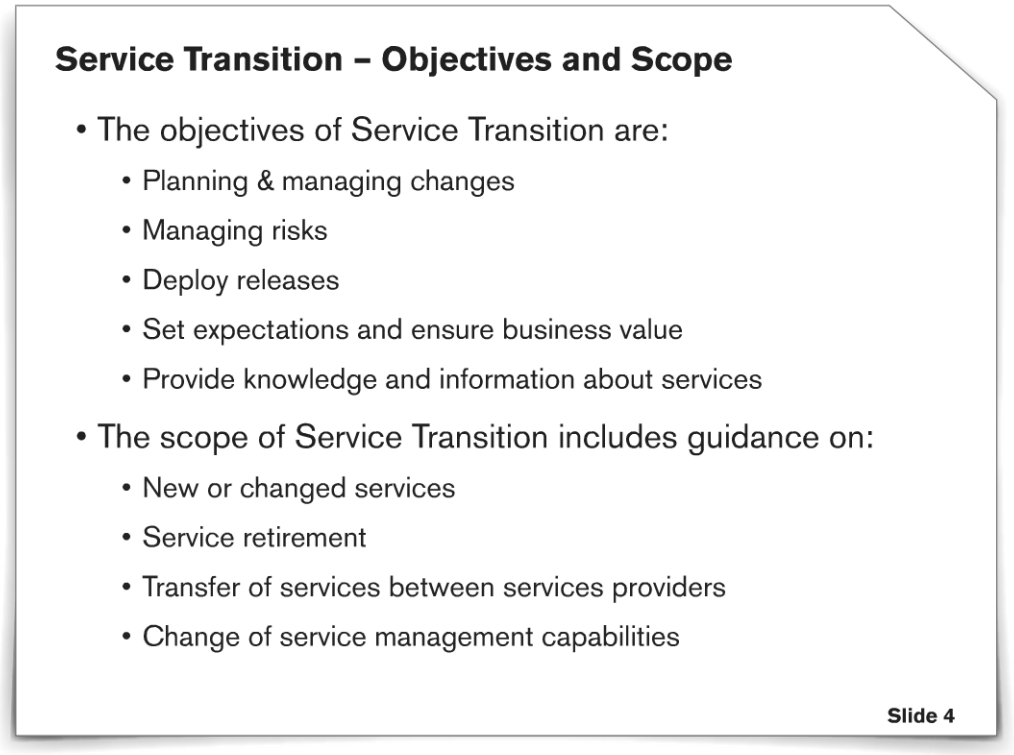
Since the inception of ITIL, the need for ongoing management of the services has been firmly established. This has been recognized as the *core* of IT Service Management–providing and supporting the "business as usual" delivery of the organization's requirements from IT.

Service Transition – "The real world"

Research often shows that the majority of IT projects do not yield the desired results or outputs. Either budgets are exceeded by large amounts or schedules are overrun by months or even years. In addition, each year think tanks such as Gartner Research and others report that roughly 80% of the incidents experienced by IT customers are caused by failed changes and activities within IT, at terrible expense.

These statistics plainly show that IT is actually quite "broken" and that IT is its own "biggest problem." Clearly then, most IT organizations could benefit from any method that brings about stability, improves quality, increases efficiency and effectiveness and reduces the costs of IT.

Service Transition is particularly critical as functional and technical errors not found during this phase will result in significantly higher impact levels to the business and/or IT infrastructure. This inevitably will cost much more to fix once the service is in operation.



Service Transition – Objectives and Scope

The success of Service Transition is in the ability of Service Operation (the day-to-day management of services) to support the business processes via the installed service base. The *mechanism* for achieving the goal is secondary and adaptive—and this applies whether an organization is transitioning service designs into business support or components and materials into products.

Effective Service Transition ensures that meeting business need, cost, and efficiency are achieved with minimal risk, maximum optimization and the highest degree of confidence possible. The Purpose of Service Transition is to:

- Set customer expectations on how the performance and use of the new or changed services may be used to enable positive business change.
- Reduce variations in the predicted and actual performance of the transitioned services.
- Reduce the Known Errors and minimize the risks from transitioning the new or changed services into production.

The **objectives** are to:

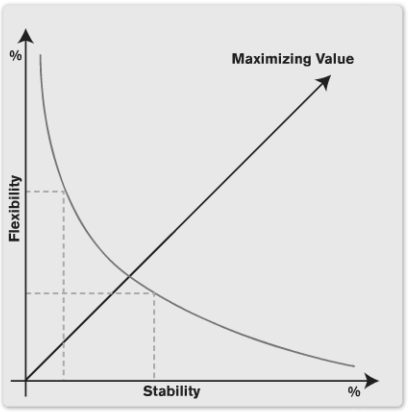
- Plan and manage service changes efficiently and effectively
- Manage risks relating to new, changed or retired services
- Successfully deploy service releases into supported environments
- Set correct expectations on the performance and use of new or changed services

- Ensure that service changes create the expected business value
- Provide good-quality knowledge and information about services and service assets.

ITIL Service Transition provides guidance for the development and improvement of capabilities for transitioning new and changed services into supported environments, including release planning, building, testing, evaluation and deployment. The publication also considers service retirement and transfer of services between service providers. The guidance focuses on how to ensure that the requirements from service strategy, developed in service design, are effectively realized in service operation while controlling the risks of failure and subsequent disruption.

Service Transition – Value to the Business

- Business is in constant Change
- Service Transition provides:
  - More accurate estimations for projects
  - More successful changes
  - Clear way of working
  - Reuse of resources
  - Reduced delays
  - Reduced conflicts
  - Improved expectations
  - Increase confidence
  - Maintainable and cost-effective services
  - Better controle on service assets



Slide 5

Service Transition – Value to the Business

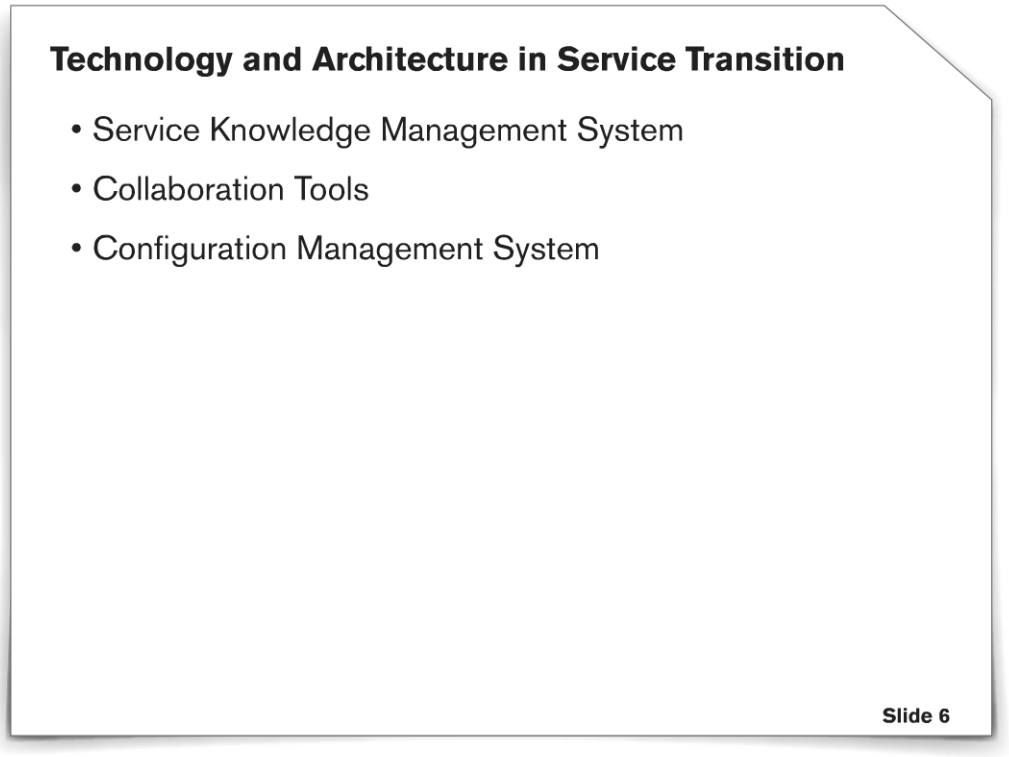
Effective Service Transition may significantly improve a Service Provider's ability to handle high volumes of change and releases across its customer base. It enables the Service Provider to:

- Align the new or changed service with the customer's business requirements and business operations
- Ensure that customers and users can use the new or changed service in a way that maximizes value to the business operations

Specifically, Service Transition adds value to the business by improving:

- Enable projects to estimate the cost, timing, resource requirement and risks associated with the service transition stage more accurately
- Result in higher volumes of successful change
- Be easier for people to adopt and follow
- Enable service transition assets to be shared and re-used across projects and services
- Reduce delays from unexpected clashes and dependencies – for example, if multiple projects need to use the same test environment at the same time
- Reduce the effort spent on managing the service transition test and pilot environments
- Improve expectation setting for all stakeholders involved in service transition including customers, users, suppliers, partners and projects

- Increase confidence that the new or changed service can be delivered to specification without unexpectedly affecting other services or stakeholders
- Ensure that new or changed services will be maintainable and cost-effective
- Improve control of service assets and configurations.



Technology and Architecture in Service Transition

Technology has a major role to play in Service Transition, and this should be designed in, and mechanisms must be in place for maintaining and maximizing benefit from that technology.

Service Transition is supported by technology through:

- Enterprise-wide tools that support the broader systems and processes within which Service Transition delivers support
- Enterprise frameworks that provide integration capabilities to integrate and link in the Configuration Management Database (CMDB) or tools
- System, network and application management tools
- Service dashboards and reporting tools

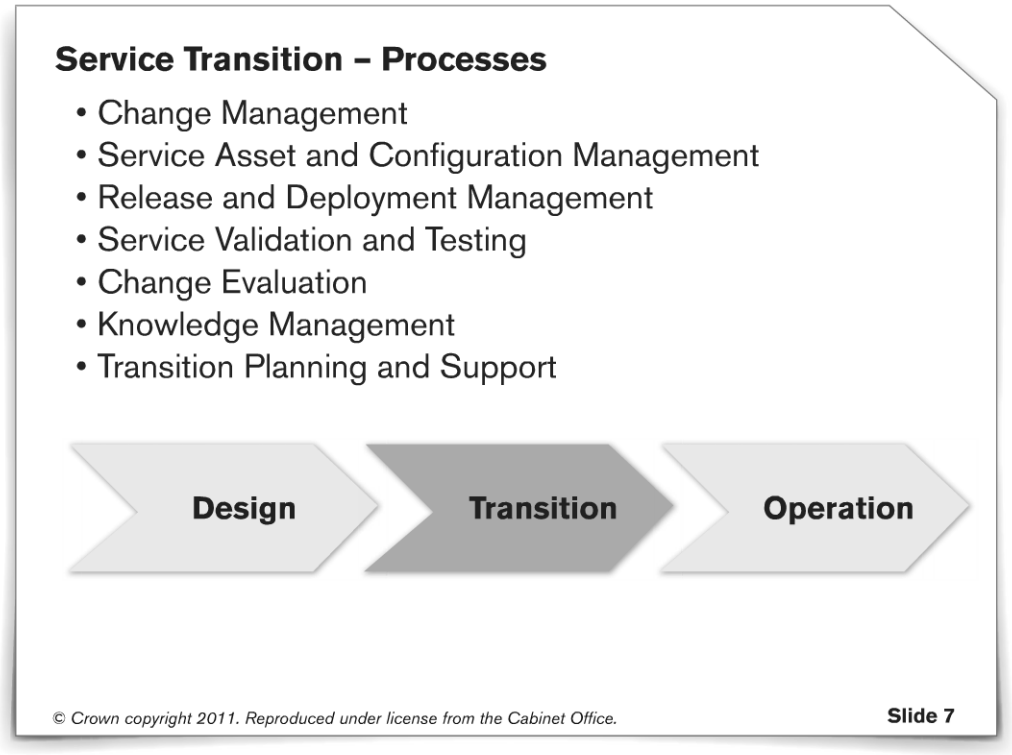
Tools targeted more specifically at supporting Service Transition or parts of Service Transition:

- Service Knowledge Management System
- Collaborative, content management, workflow tools
- Data mining tools
- Extract, load and transform data tools
- Measurement and reporting systems
- Test management and testing tools

- Database and test data management tools
- Copying and publishing tools
- Release and deployment technology
- Deployment and logistics systems and tools

Knowledge Management benefits a lot from technology; the Service Knowledge Management System is a good example. Knowledge Management tools may include functionality for document management, records management or content management.

Collaboration is the process of sharing tacit knowledge and working together to accomplish stated goals and objectives. Examples of tools for collaboration are shared calendars, discussions, instant messaging and email. Other methods are **communities**, such as portal sites and **workflow management** tools.



Service Transition – Processes

The **processes** and **activities** in the list below comprise both lifecycle processes and those almost wholly contained within Service Transition:

- **Change Management**: In place to respond to changing customer requirements and business and IT Requests for Change, while maximizing value and reducing incidents, disruptions and re-work
- **Service Asset and Configuration Management**: A process to manage Service Assets and Configuration Items in order to support the other Service Management processes and the business
- **Release and Deployment Management**: Aims to build, test and deliver the capability to provide the services specified by Service Design and that will accomplish the stakeholders' requirement and deliver the intended objective
- **Service Testing and Validation**: Attempts to establish a degree of confidence that a new or changed service will deliver the value and outcomes required of it through testing and validation.
- **Change Evaluation**: Provides a consistent and standardized means of determining the performance of a Service Change in the context of existing and proposed services and IT Infrastructure. This process is out of scope for the ITIL Foundation exam.
- **Knowledge Management**: Enables organizations to improve the quality of management

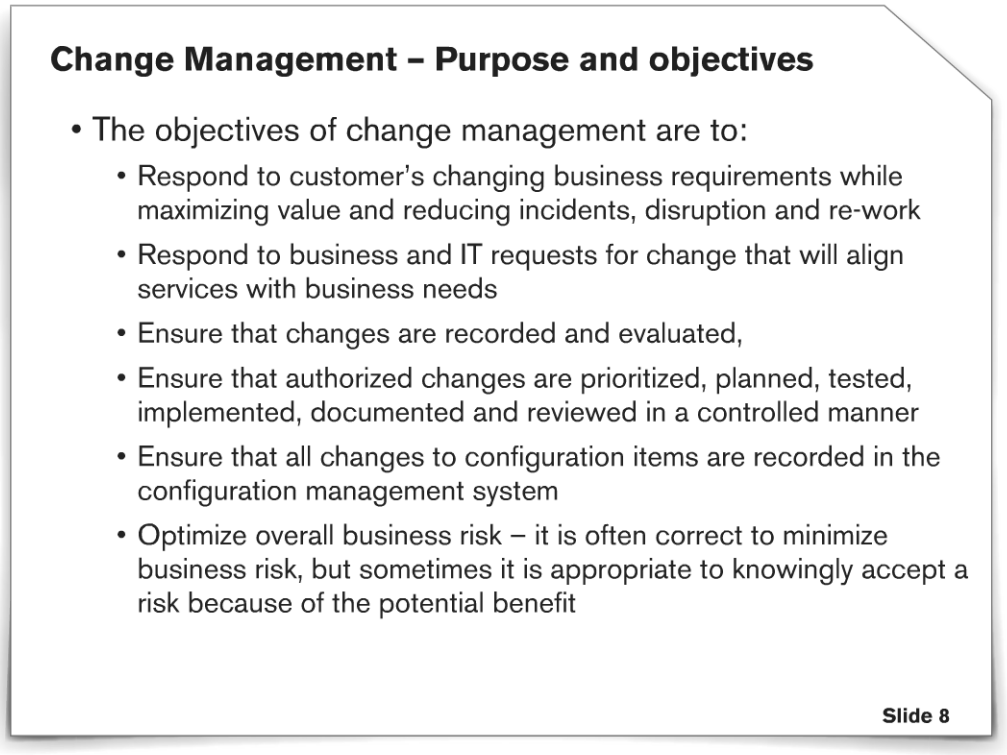
decision-making by ensuring that reliable and secure information and data is available throughout the Service Lifecycle

- **Transition Planning and Support**: Helps a Service Provider's ability to handle high volumes of change and releases across its customer base and aligns the Service Transition plans with customer, supplies and business change project plans. This process is out of scope for the ITIL Foundation exam.

Change Management, Service Asset and Configuration Management and Knowledge Management are used throughout the Service Lifecycle, but are addressed in this module since they are central to effective Service Transition. The other processes and activities are mostly contained within the Service Transition phase of the lifecycle, but also are used throughout other phases (e.g., evaluation of design, performance testing within operations).

Two specific **activities** that are important to Service Transition:

- **Organizational and Stakeholder Change**: Reflecting the holistic nature of change that Service Transition must be based on, organizations do not transform their IT service by only changing the IT services. Modern innovations mean that the organization itself will also inevitably change to make use of the new (and changed) Services available
- **Communications**: One of the major traditional weaknesses in Service Transition has been the inability to deliver sufficient prompt understanding of the implications, benefits and usage of IT services. So, Service Transition must take on this communication role to ensure that the business, the end user community and the IT staff are all aware of the services available, why they are important, how they interrelate with and impact other services, and how they are supported.

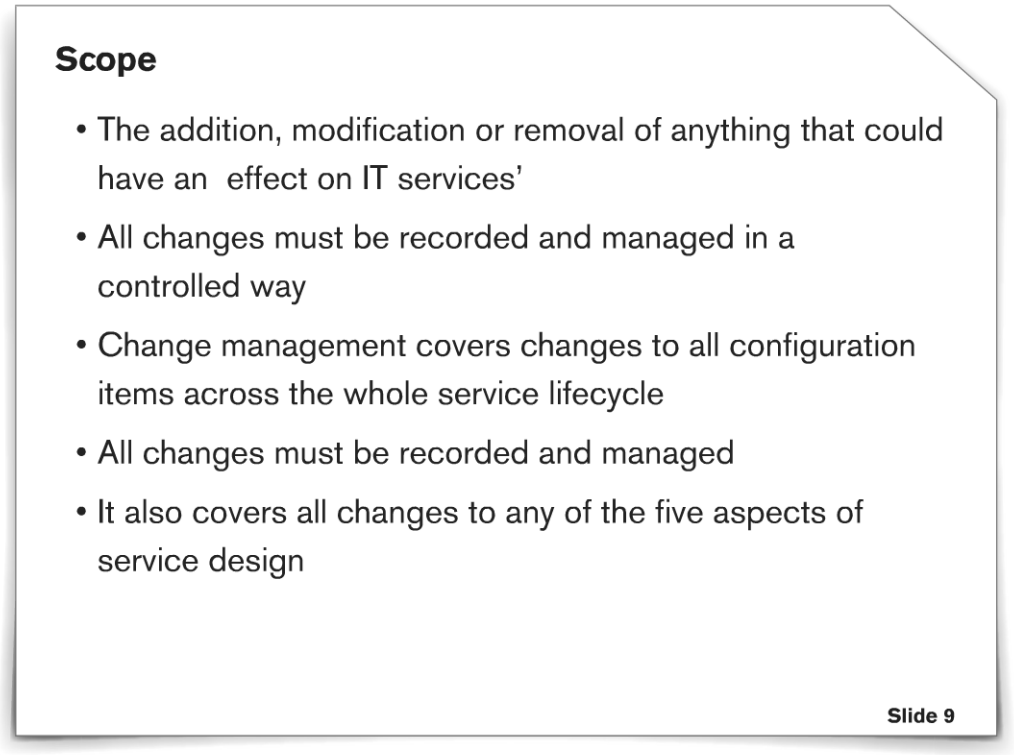


Change Management – Purpose and objectives

The purpose of the change management process is to control the lifecycle of all changes, enabling beneficial changes to be made with minimum disruption to IT services.

The objectives of change management are to:

- Respond to the customer's changing business requirements while maximizing value and reducing incidents, disruption and re-work.
- Respond to the business and IT requests for change that will align the services with the business needs.
- Ensure that changes are recorded and evaluated, and that authorized changes are prioritized, planned, tested, implemented, documented and reviewed in a controlled manner.
- Ensure that all changes to configuration items are recorded in the configuration management system.
- Optimize overall business risk – it is often correct to minimize business risk, but sometimes it is appropriate to knowingly accept a risk because of the potential benefit.



Scope

Change can be defined in many ways. The ITIL definition of a change is 'the addition, modification or removal of anything that could have an effect on IT services'. The scope should include changes to all architectures, processes, tools, metrics and documentation, as well as changes to IT services and other configuration items.

All changes must be recorded and managed in a controlled way. The scope of change management covers changes to all configuration items across the whole service lifecycle, whether these CIs are physical assets such as servers or networks, virtual assets such as virtual servers or virtual storage, or other types of asset such as agreements or contracts.

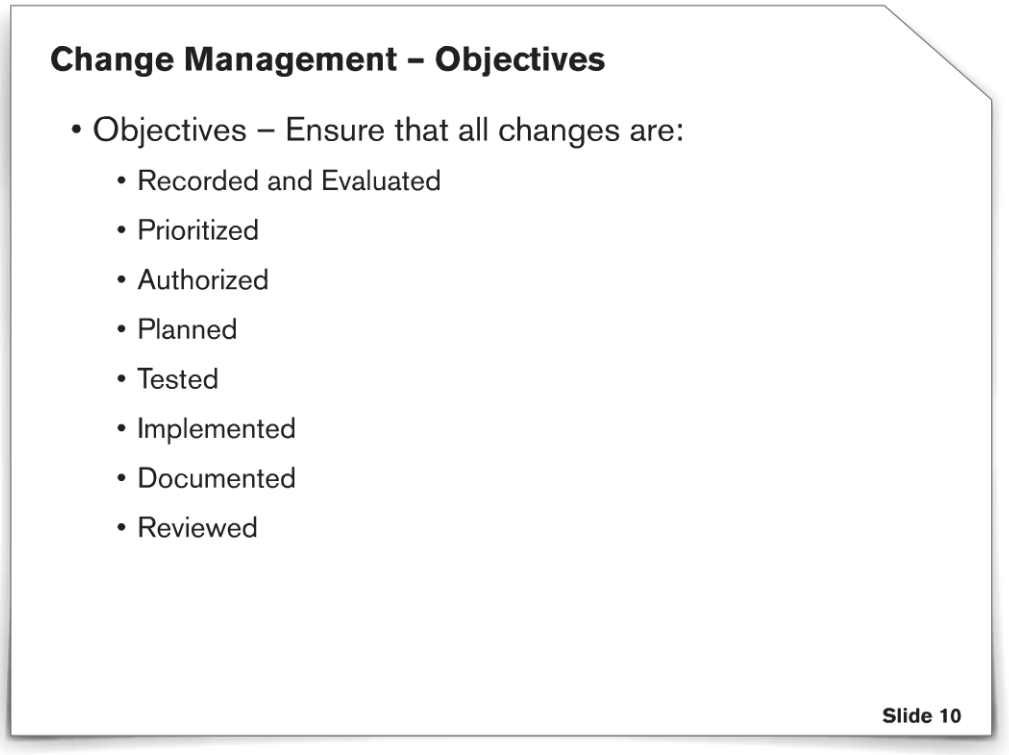
It also covers all changes to any of the five aspects of service design:

- Service solutions for new or changed services, including all of the functional requirements, resources and capabilities needed and agreed
- Management information systems and tools, especially the service portfolio, for the management and control of services through their lifecycle
- Technology architectures and management architectures required to provide the services
- Processes needed to design, transition, operate and improve the services
- Measurement systems, methods and metrics for the services, the architectures, their constituent components and the processes.

Each organization should define the changes that lie outside the scope of its change management process. Typically these might include:

- Changes with significantly wider impacts than service changes, e.g. departmental organization, policies and business operations – these changes would produce RFCs to generate consequential service changes.
- Changes at an operational level such as repair to printers or other routine service components.

Change management is not responsible for coordinating all of the service management processes to ensure the smooth implementation of projects. This activity is carried out by transition planning and support.

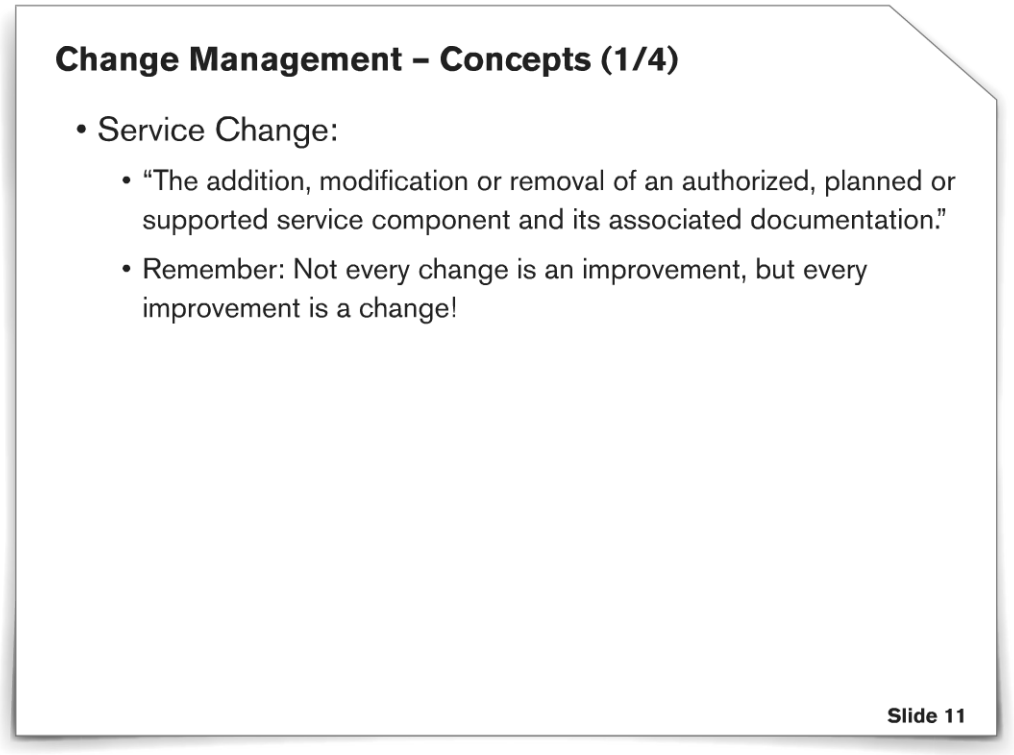


Change Management – Objectives

The **objectives** of the Change Management process are to ensure that changes are recorded and then evaluated, authorized, prioritized, planned, tested, implemented, documented and reviewed in a controlled manner.

The **purpose** of the Change Management process is to ensure that:

- Standardized methods and procedures are used for efficient and prompt handling of all changes
- All changes to Service Assets and Configuration Items are recorded in the Configuration Management System
- Overall business risk is minimized, optimized and understood



Service Management – Concepts (1/4)

To understand the process of Change Management, it is important to understand the following definition:

Service Change:

"Change" can be defined in many ways; however the definition of a **Service Change** is very specific:*"The addition, modification or removal of an authorized, planned or supportedservice or service component and its associated documentation*"

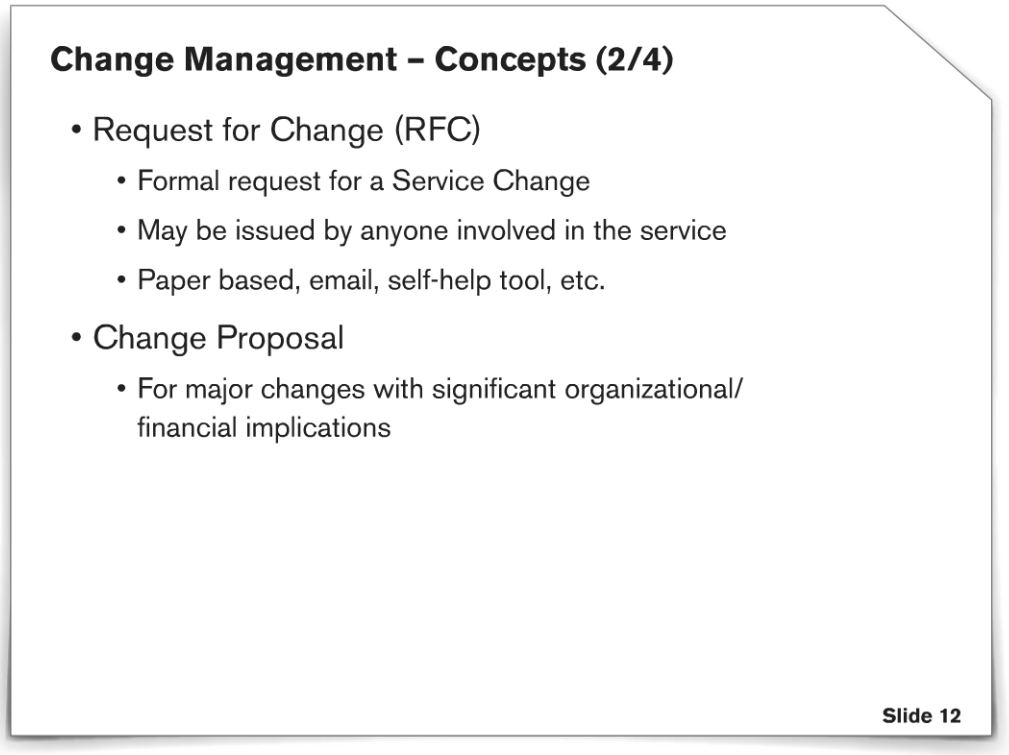
This definition sets the scope for Change Management and reveals some important aspects:

- Not only is modification considered a change, but also addition or removal
- Live services, as well as those that are proposed or being built, are all within the scope of Change Management
- Documentation that supports a service is also within scope

Changes arise for a variety of reasons:

Proactively: Seeking business benefits such as reducing costs or improving services or increasing the ease and effectiveness of support

Reactively: As a means of resolving errors and of adapting to changing circumstances, supporting new product lines, etc.



Change Management – Concepts (2/4)

Changes should be managed to:

- Optimize risk exposure (supporting the risk profile required by the business)
- Minimize the severity of any impact and disruption
- Be successful at the first attempt

To understand the process of Change Management, it is important to understand the following definitions:

Request for Change (RFC):

The Request for Change (RFC) is the formal request for a change and may be issued by anyone involved in the service, including an end-user, IT staff or management from business or IT. The RFC is the starting point for the Change Management process and is often restricted to a certain format or required information.

Procedures for documenting and receiving RFCs should be decided. All RFCs received should be logged and assigned a unique identification number (ideally in chronological sequence). Where Requests for Change are submitted in response to a trigger such as a resolution to a Problem Record (PR), it is important that the reference number of the triggering document is retained to provide traceability.

Change Management – Concepts (3/4)

- A change proposal is used to communicate a high-level description of the change. The change proposal is normally created by the service portfolio management process and is passed to change management for authorization.
- In some organizations, change proposals may be created by a program management office or by individual projects.

Slide 13

Change Management – Concepts (3/4)

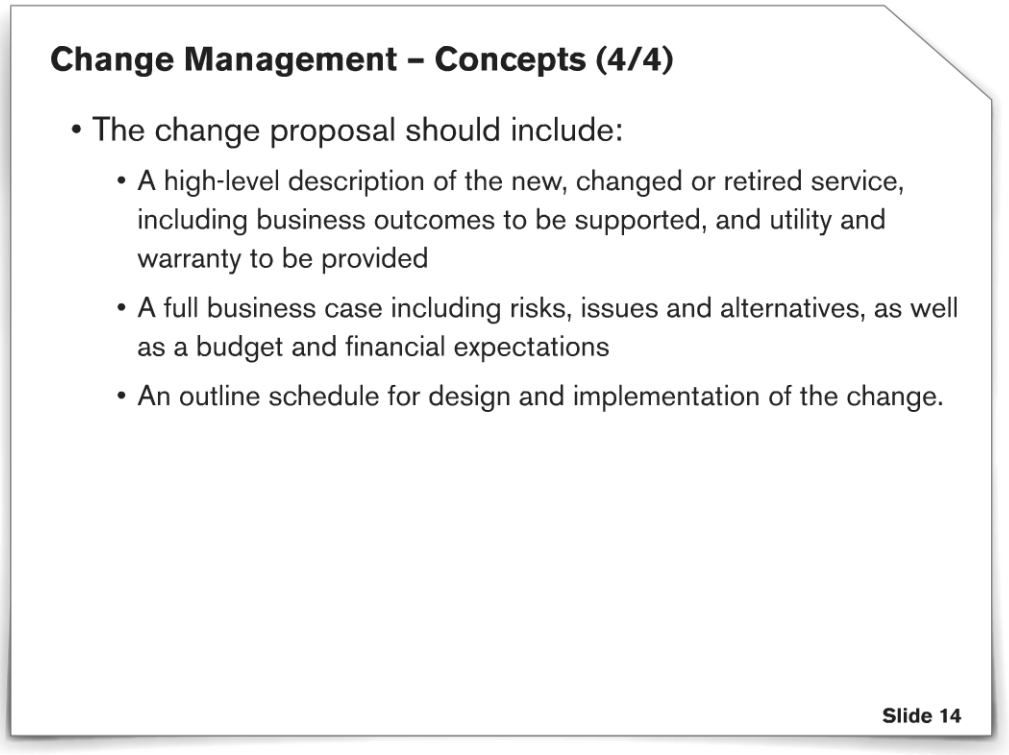
The procedures should stipulate that, as changes are logged, Change Management should briefly consider each request and filter out any that seem to be:

- Totally impractical
- Repeats of earlier RFCs which are accepted, rejected or still under consideration
- Incomplete submissions (e.g., inadequate description, without necessary budgetary approval)

Requests that are denied should be returned to the initiator, together with brief details of the reason for the rejection. All of these items should be recorded in the log. A right of appeal against rejection should exist, via normal management channels, and should be incorporated within the procedures.

Change Proposal: For those cases where a change has significant organization and/or financial implications, an RFC may not be sufficient, and a Change Proposal must be submitted. This Change Proposal requires the **full description** of the change, together with a business and financial justification for the change. It requires sign-off by appropriate levels of business management.

Major changes that involve significant cost, risk or organizational impact will usually be initiated through the service portfolio management process. Before the new or changed service is chartered it is important that the change is reviewed for its potential impact on other services, on shared resources, and on the change schedule.



Change Management – Concepts (4/4)

Change proposals are submitted to change management before chartering new or changed services in order to ensure that potential conflicts for resources or other issues are identified. Authorization of the change proposal does not authorize implementation of the change but simply allows the service to be chartered so that service design activity can commence.

A change proposal is used to communicate a high-level description of the change. This change proposal is normally created by the service portfolio management process and is passed to change management for authorization. In some organizations, change proposals may be created by a programme management office or by individual projects. The change proposal should include:

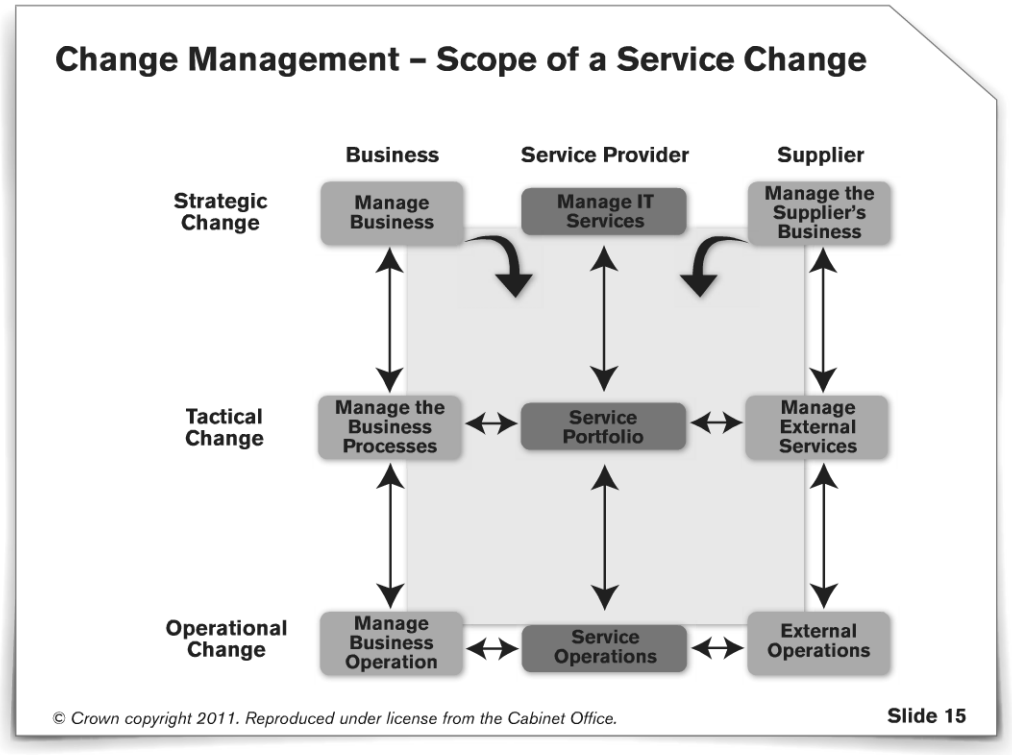
A high-level description of the new, changed or retired service, including business outcomes to be supported, and utility and warranty to be provided

A full business case including risks, issues and alternatives, as well as a budget and financial expectations

An outline schedule for design and implementation of the change.

Change management reviews the change proposal and the current change schedule, identifies any potential conflicts or issues and responds to the change proposal by either authorizing it or documenting the issues that need to be resolved. When the change proposal is authorized, the change schedule is updated to include outline implementation dates for the proposed change.

After the new or changed service is chartered, RFCs will be used in the normal way to request authorization for specific changes. These RFCs will be associated with the change proposal so that change management has a view of the overall strategic intent and can prioritize and review the RFCs appropriately.



Change Management – Scope of a Service Change

The **scope** of Change Management covers changes to baseline Service Assets and Configuration Items across the entire Service Lifecycle.

The figure above illustrates a typical scope for the Service Change Management process for an IT department and how it interfaces with the business and suppliers at strategic, tactical and operational levels. It covers interfaces to internal and external Service Providers where there are shared assets and Configuration Items that must be governed by Change Management.

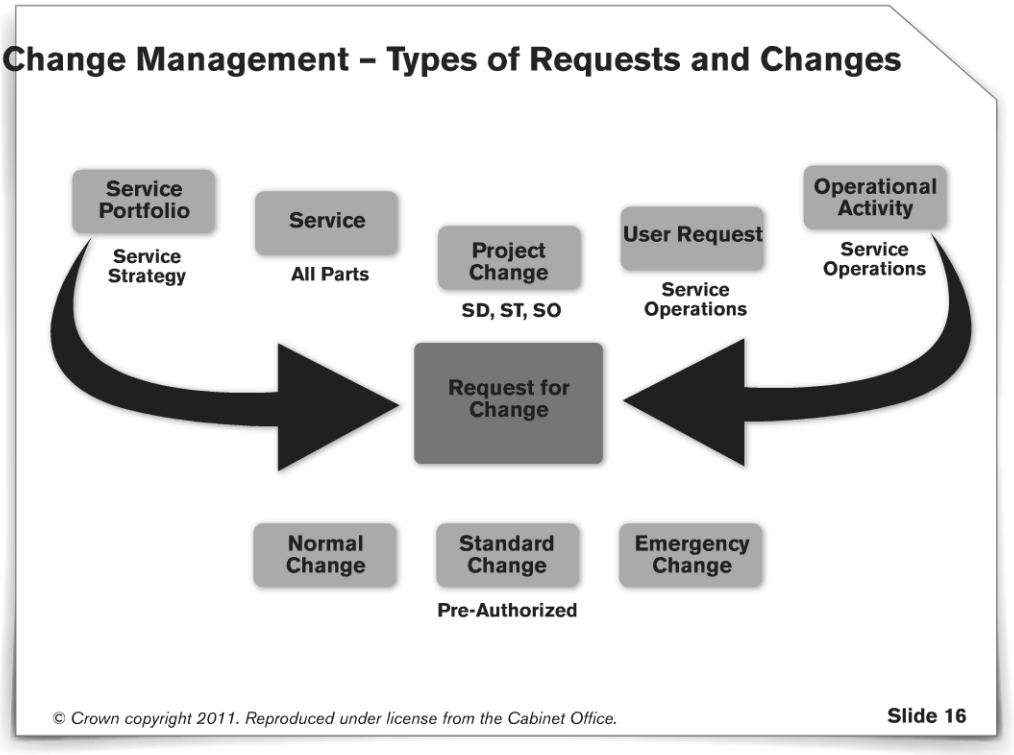
Change Management must interface with business Change Management (the left side of the figure), and also with supplier Change Management (the right side of the figure). The supplier may be an external party with its own Change Management process, or it could be an internal development project with specific change mechanisms.

The Service Portfolio provides a clear definition of all current, planned and retired services (discussed in further detail in the strategy module). Understanding the Service Portfolio helps all parties involved in the Service Transition to understand the potential impact of the new or changed service on current services and other new or changed services.

Strategic changes are introduced via Service Strategy and business relationship management.

Tactical changes to a service will be brought in via Service Design, Continual Service Improvement and the Service Level Management process.

Operational changes, such as corrective change or resolving errors detected in services, will be initiated from Service Operations and may route via support or external suppliers into a formal RFC.



Change Management – Types of Requests and Changes

A change request is a formal communication seeking an alteration to one or more Configuration Items. This could take several forms, e.g., "Request for Change" document, Service Desk call, Project Initiation Document. Different types of change may require different types of change request. An organization needs to ensure that appropriate procedures and forms are available to cover the anticipated requests. Avoiding a bureaucratic approach to documenting a minor change removes some of the cultural barriers to adopting the Change Management process.

In the example given in the above diagram, we see how different changes may be initiated by various parties, users and activities involved in Service Management at different lifecycle stages of services.

Simple authorization should be used as much as possible, both through the standard change procedure and through the authorization of minor changes by Change Management staff.

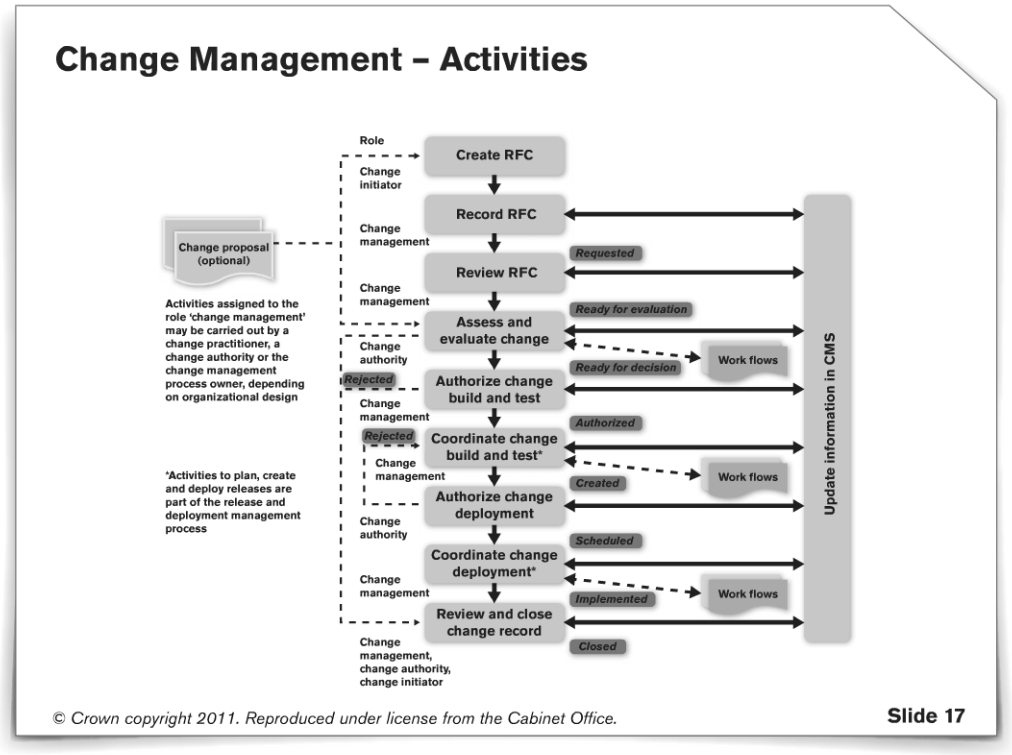
Normal Changes (such as changes to Service Portfolios, service definitions, project changes, user access, operational changes) must follow the "normal" Change Management process, and may be further categorized as changes that are *major*, *minor*, or *significant* in nature and they must be referred to the various levels of management for the final authorization.

Emergency Changes are sometimes required. Procedures should be devised to deal with them quickly, without sacrificing normal management controls.

The Emergency Change designation is reserved for changes intended to repair an error in an IT service that is negatively impacting the business to a high degree. Changes intended to introduce immediately required business improvements are handled as **Normal Changes**, assessed as having the highest urgency.

Defined authorization levels will exist for an emergency change, and the levels of delegated authority must be clearly documented and understood.

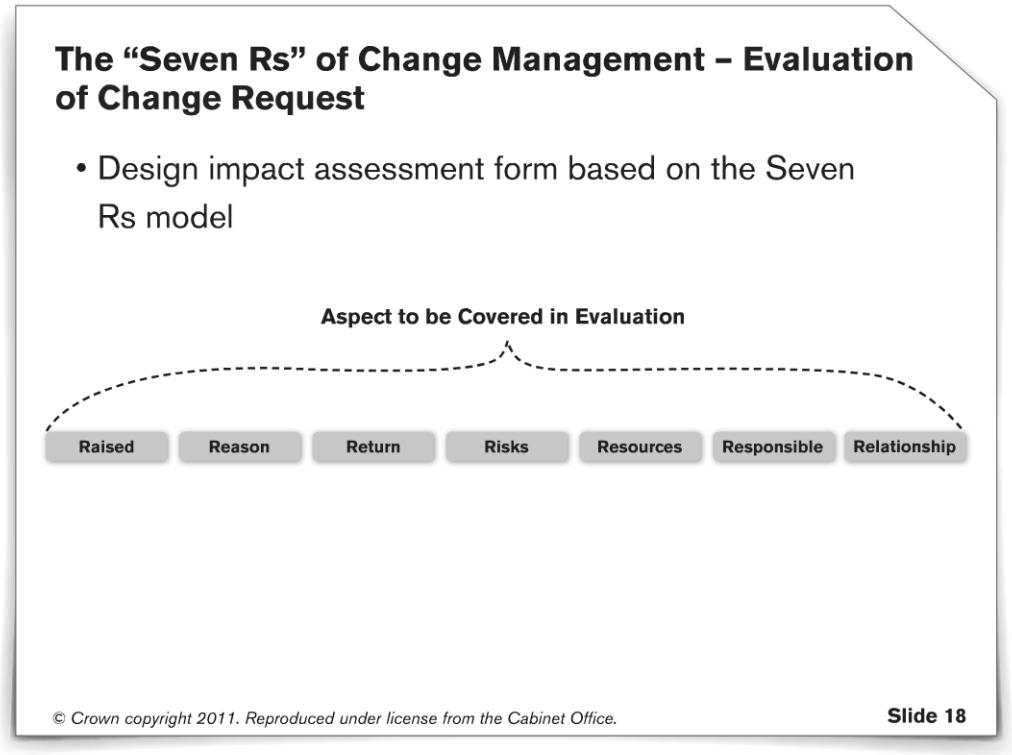
As much testing of the Emergency Change as possible should be carried out. Completely untested changes should not be implemented if at all avoidable. Clearly, if a change goes wrong, the cost is usually greater than that of adequate testing. Consideration should be given to how much it would cost to fully test all changes *with* the cost of the change failing factored by the anticipated likelihood of its failure.



Change Management – Activities

- Create and record the RFC
- Review the RFC
  - Filter changes (e.g. incomplete or wrongly routed changes)
- Assess and evaluate the change
  - Establish the appropriate level of change authority
  - Establish relevant areas of interest (i.e. who should be involved in the CAB)
  - Evaluate the business justification, impact, cost, benefits, risks and predicted performance of the change
  - Submit a request for evaluation to initiate activity from the change evaluation process
- Authorize the change
  - Obtain authorization/rejection
  - Communicate the decision with all stakeholders, in particular the initiator of the request for change
- Plan updates
- Coordinate change implementation

- Review and close change
  - Collate the change documentation, e.g. baselines and evaluation reports
  - Review the change(s) and change documentation
  - Ensure that details of lessons learned have been entered into the service knowledge management system
  - Close the change document when all actions are completed.

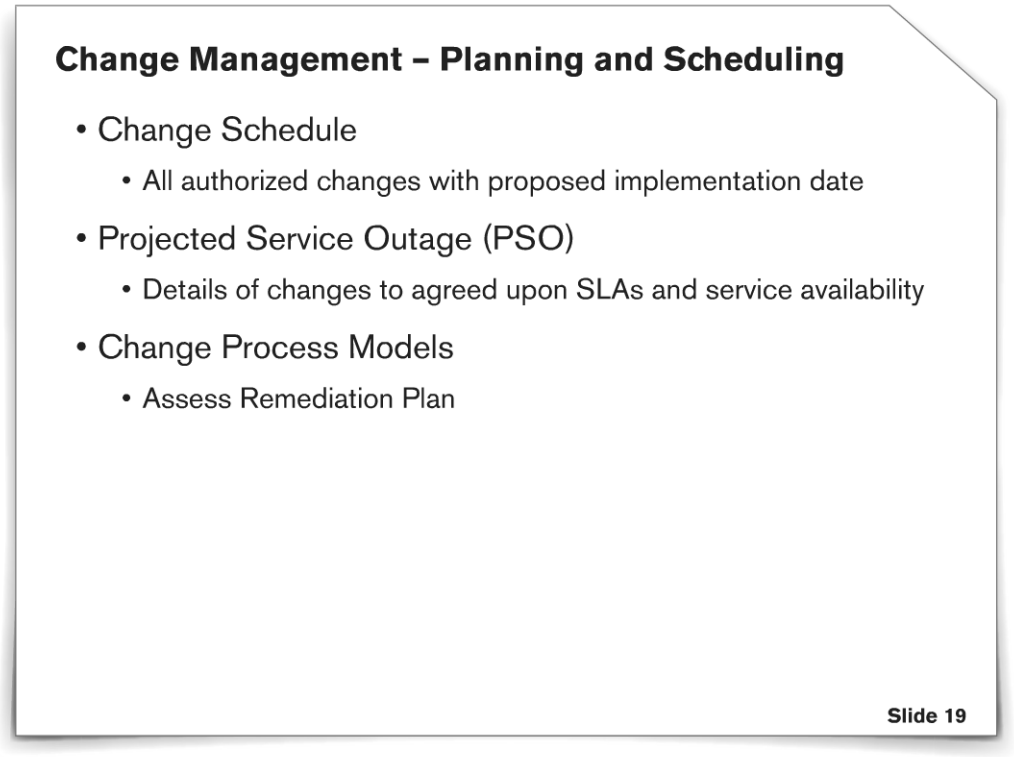


The "Seven Rs" of Change Management – Evaluation of Change Request

Every Request for Change that enters the Change Management process, must be evaluated. The potential impact of failed changes to services, service assets (and their configurations) must be considered. Within ITIL, the "Seven Rs" provide generic questions that are a good starting point for the evaluation of a request. These are:

- Who **raised** the change?
- What is the **reason** for the change?
- What is the **return** required from this change?
- What are the **risks** involved in the change?
- What **resources** are required to deliver the change?
- Who is **responsible** for the build, test and implementation of the change?
- What is the **relationship** between this change and other changes?

Many organizations develop specific impact assessment forms to prompt the impact assessors about specific types of change. This may help with the learning process, particularly for new services or when implementing a formal impact assessment step for the first time.



Change Management – Planning and Scheduling

Careful planning of changes will ensure that there is no ambiguity about what tasks are included in the Change Management process or other processes and how processes interface with any suppliers or projects that are providing a change or release.

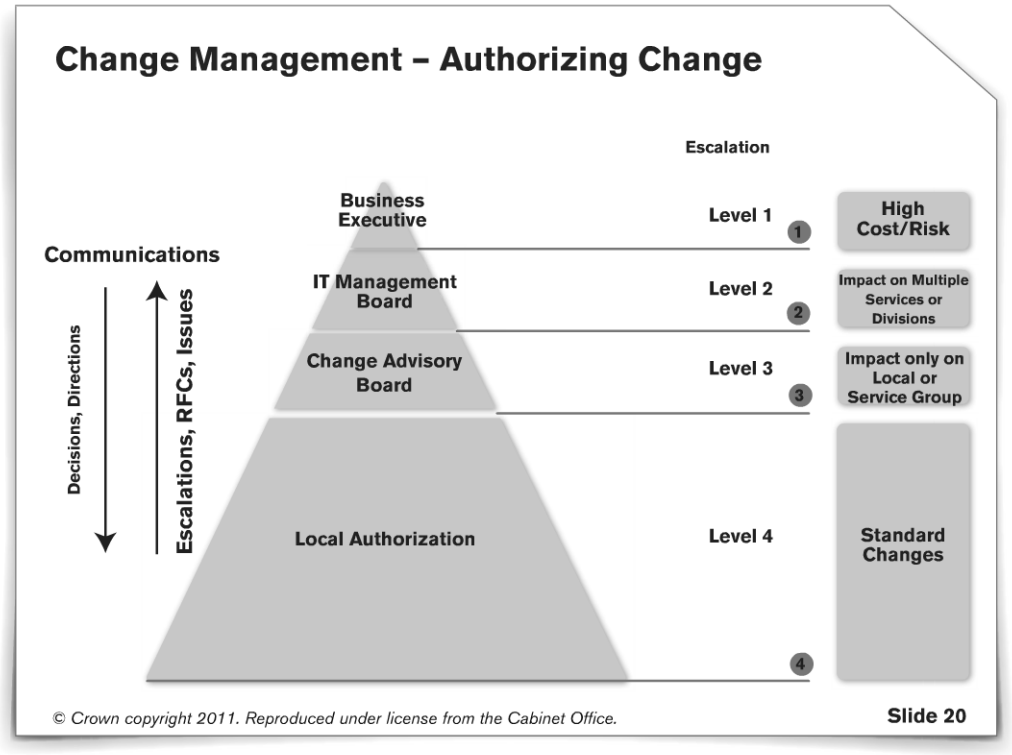
It is strongly recommended that Change Management schedule changes to meet business rather than IT needs (e.g., avoid making changes during critical business periods, even though these periods may be more convenient for IT).

Wherever possible, Change Management should schedule authorized changes into target Release or Deployment Packages and recommend the allocation of resources accordingly.

Change Management coordinates the production and distribution of a **Schedule of Changes** (SC) and **Projected Service Outage** (PSO). The SC contains details of all the changes authorized for implementation and proposed implementation dates for the changes. The PSO contains details of changes to agreed upon SLAs and service availability because of the currently planned SC, in addition to planned downtime from other causes such as planned maintenance and data backups. These documents are agreed to with the relevant customers within the business, with Service Level Management, with the Service Desk and with Availability Management. Once agreed upon, the Service Desk should communicate any planned additional downtime to the user community at large, using the most effective methods available.

Organizations may find it helpful to predefine change process models and apply them to appropriate changes when they occur. This set of predefined steps that should be taken when dealing with specific types of changes in an agreed upon way. This could include the steps that should be taken to handle the change, including handling issues, the chronological order of these steps and any dependencies, who is responsible (who should do what), the timeframes and thresholds for completing the actions and any escalation procedures if needed.

No change should be approved without having explicitly addressed the question of what to do if it is not successful. Ideally, there will be a back-out plan, which will restore the organization to its initial situation, often through the reloading of a baselined set of CIs, especially software and data. However, not all changes are reversible, in which case an alternative approach to remediation is required. This remediation may require a revisiting of the change itself in the event of failure, or may be so severe that it requires invoking the organization's business continuity plan. Only by considering what remediation options are available before instigating a change, and by establishing that the remediation is viable (e.g., it is successful when tested), can the risk of the proposed change be determined and the appropriate decisions taken.



Change Management – Authorizing Change

For each change, formal authorization is obtained from a Change Authority that may be a role, person or a group of people. The levels of authorization for a particular type of change should be judged by the type, size or risk of the change (e.g., changes in a large enterprise that affect several distributed sites may need to be authorized by a higher level Change Authority such as a global CAB or by the Board of Directors).

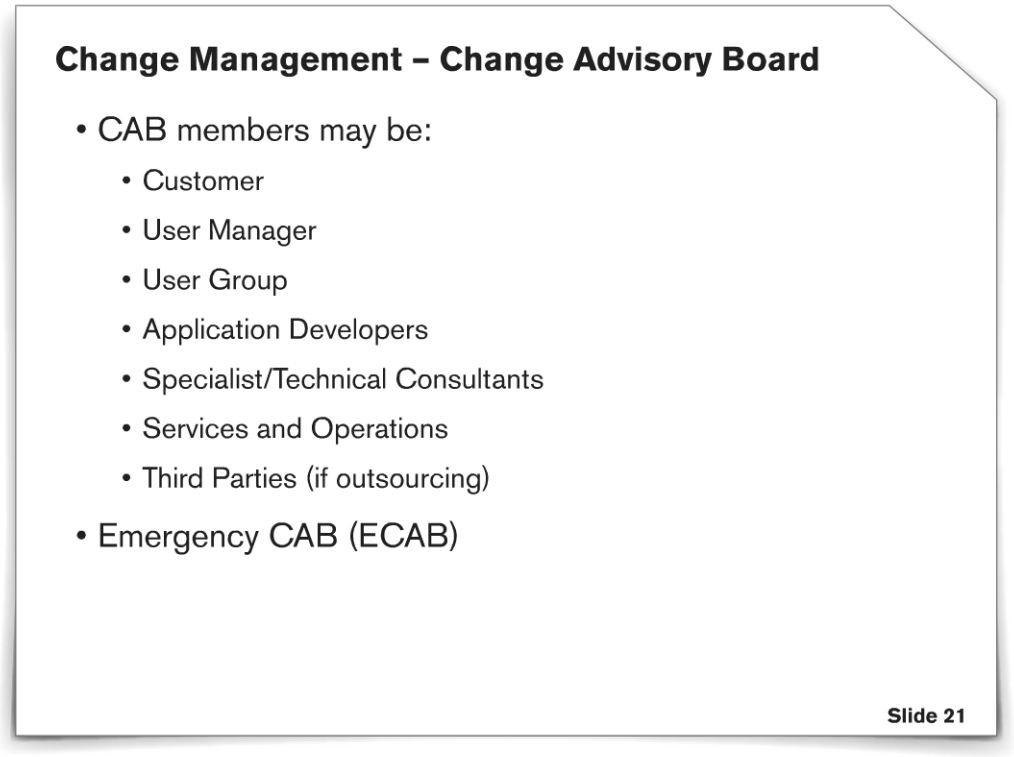
If change assessment at levels 2, 3, or 4 detects higher levels of risk, the authorization request is escalated to the appropriate higher level for the assessed level of risk. The use of delegated authority from higher levels to local levels must be accompanied by trust in the judgment, access to the appropriate information and supported by management. The level at which change is authorized should rest where accountability for accepting risk and remediation exist.

While the responsibility for change authorization lies with the change manager, he still must gain:

- Financial Approval: What is it going to cost? And what is the cost of not doing it?
- Business Approval: What are the consequences to the business? And of not doing it?
- Technology Approval: What are the consequences to the infrastructure? And of not doing it?

It is important that the following considerations are taken into account:

- The implications of performing the change, as well as the impacts of NOT implementing the change
- The importance of empowering the change manager as its primary role is to protect the integrity of the IT infrastructure



Change Management – Change Advisory Board

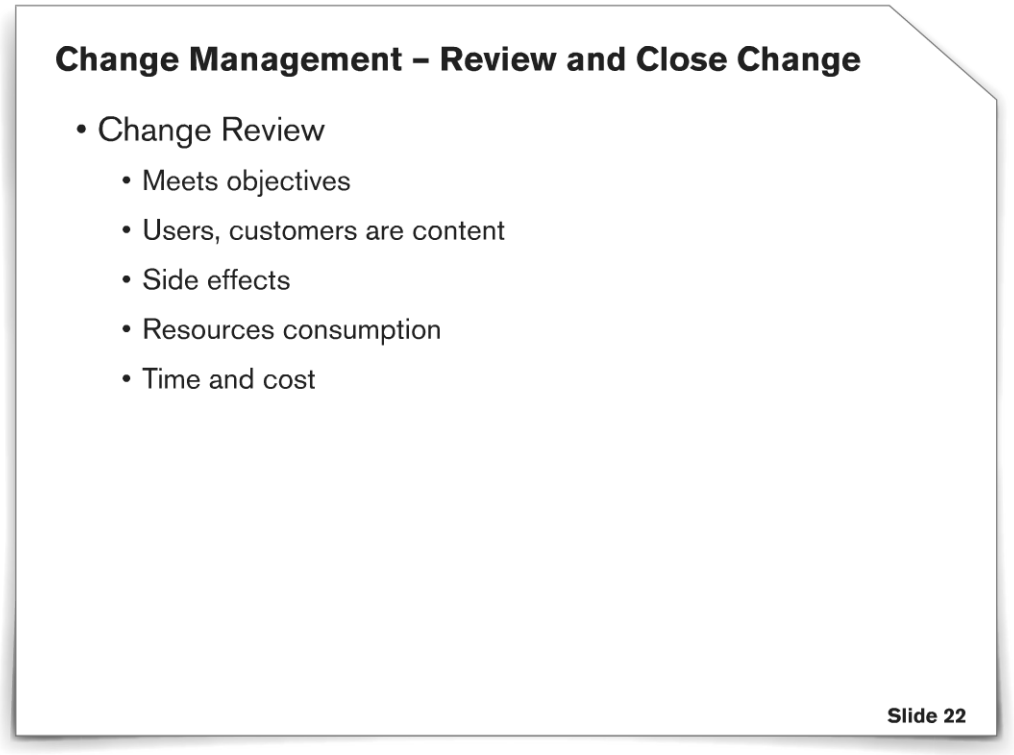
The **Change Advisory Board** (CAB) is a body that exists to support the authorization of changes and to assist Change Management in the assessment and prioritization of changes. CAB members should be chosen who are capable of ensuring that all changes within the scope of the CAB are adequately assessed from both a business and a technical viewpoint. The CAB members may differ from change to change.

The CAB may be asked to consider and recommend the adoption or rejection of changes appropriate for higher level authorization, at which point recommendations should be submitted to the appropriate Change Authority. To achieve this, the CAB must include people with a clear understanding across the whole range of stakeholder needs. The change manager will typically chair the CAB.

When the need for an emergency change arises (instances in which there may not be time to convene the full CAB), it is necessary to identify a smaller organization with authority to make emergency decisions. This body is the Emergency Change Advisory Board (ECAB).

Many organizations are conducting CABs electronically without frequent face-to-face meetings. There are benefits and problems from such an approach.

Much of the assessment and referral activities may be handled electronically via support tools or email. However, in complex, high-risk or high-impact cases, a formal "live" CAB meeting may be necessary.

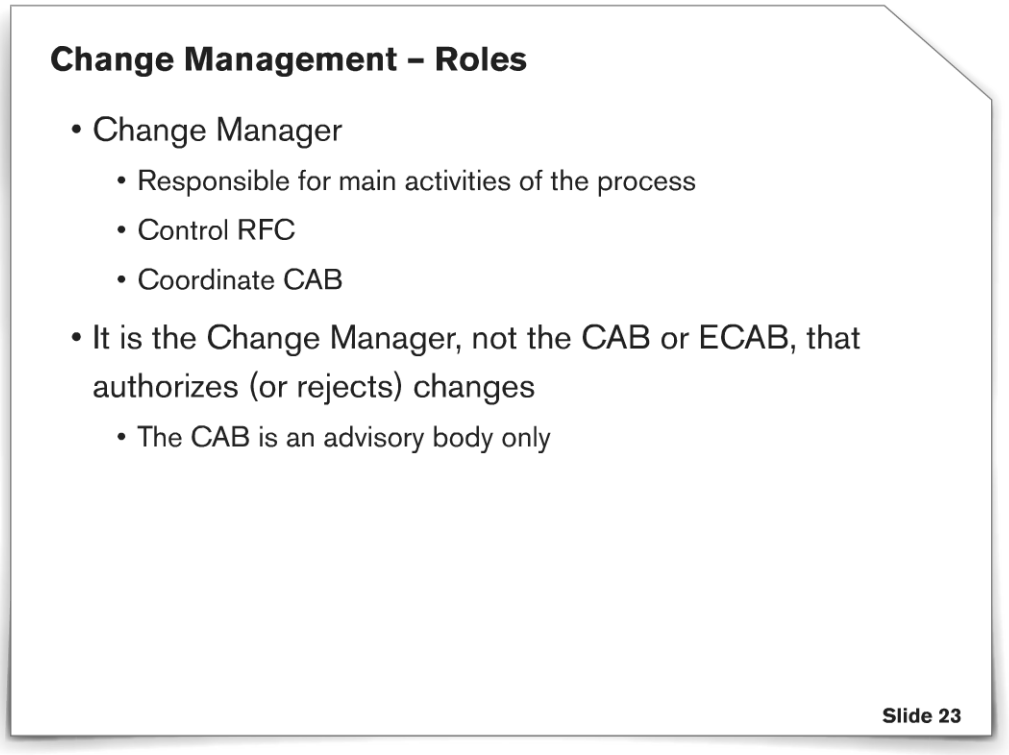


Change Management – Review and Close Change

A **Change Review** (also known as a "Post Implementation Review") should be carried out to confirm that the change has met its objectives, that the initiator and stakeholders are content with the results, and that there have been no unexpected side effects. Lessons learned should be fed back into future changes.

There is a significantly different approach and profile between:

- **Review of a *Service* Change**, which is immediately visible to the customer and scheduled for discussion at the next Service Level Management review meeting
- **Review of an *Infrastructure* Change**, which is concerned with *how* IT delivers rather than *what* IT delivers, which will be (nearly) invisible to the customer



Change Management – Roles

Responsibilities for reviewing and approving assets and CIs across the Service Lifecycle and during deployment must be defined and allocated to individuals with appropriate skills and authority.

The main duties of the **Change Manager**, some of which may be delegated, are listed below:

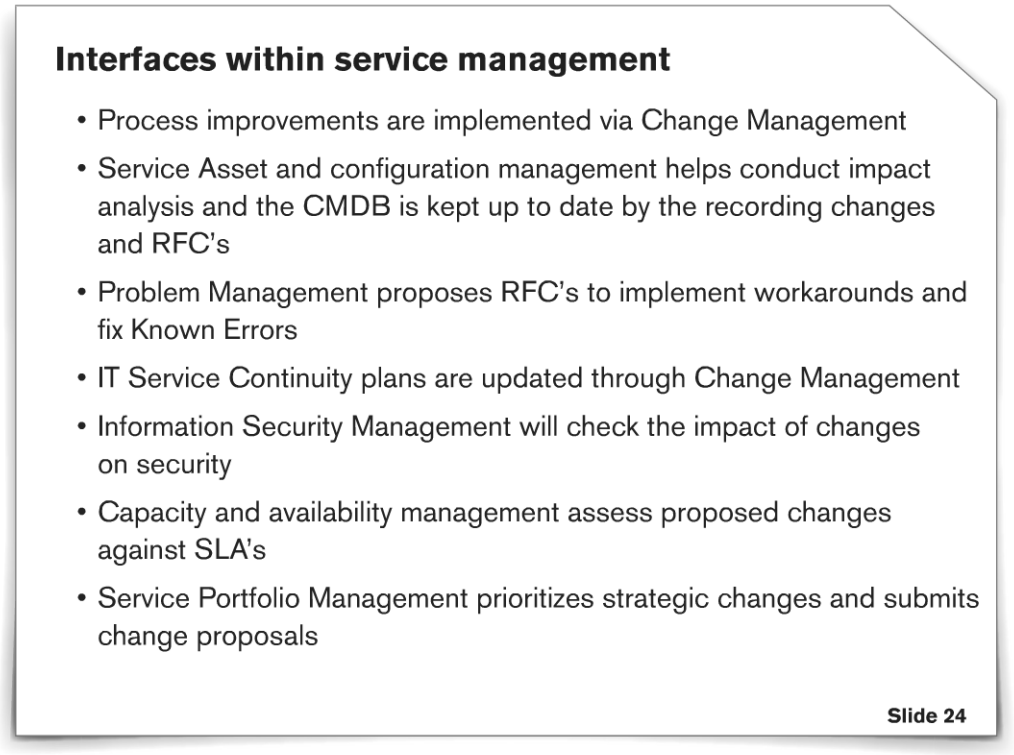
- Receive, log and allocate a priority to all RFCs, in collaboration with the initiator. Reject any RFCs that are impractical.
- Table all RFCs for a CAB meeting, issue an agenda and circulate all RFCs to CAB members in advance of meetings to allow prior consideration
- Decide which people will attend which meetings, who gets specific RFCs depending on the nature of the RFC, what is to be changed and people's areas of expertise
- Convene urgent CAB or ECAB meetings for all urgent RFCs
- Chair all CAB and ECAB meetings

After consideration of the advice given by the CAB or ECAB, authorize acceptable changes

- Issue Change Schedules via the Service Desk.
- Communicate with all necessary parties to coordinate change building, testing and implementation, in accordance with schedules.
- Update the change log with all progress that occurs, including any actions to correct

Problems and/or to take opportunities to improve service quality.

- Review all implemented changes to ensure that they have met their objectives. Refer back any that have been backed out or have failed.
- Review all outstanding RFCs awaiting consideration or awaiting action.
- Analyze change records to determine any trends or apparent Problems that occur. Seek rectification with relevant parties.
- Close RFCs.



Interfaces within service management

All service management processes may require change management, for example to implement process improvements. Many service management processes will also be involved in the impact assessment and implementation of service changes, as discussed below.

Service asset and configuration management

The configuration management system provides reliable, quick and easy access to accurate configuration information to enable stakeholders and staff to assess the impact of proposed changes and to track change work flow. This information enables the correct CI versions to be released to the appropriate party or into the correct environment. As changes are implemented, the configuration management information is updated. The CMS may also identify related CIs that will be affected by the change, but not included in the original request, or similar CIs that would benefit from similar changes.

Problem management

Problem management is another key process, as changes are often required to implement workarounds and to fix known errors. Problem management is one of the major sources of RFCs and is also often a major contributor to CAB discussion.

IT service continuity management

IT service continuity management has many procedures and plans, which should be updated via change management to ensure that they are accurate and up to date, and that stakeholders are aware of changes. Every change should be assessed for its impact on IT service continuity arrangements. For a standard change this will be done at the time the change model is authorized; for normal and emergency changes the assessment will be done as part of change assessment.

Information security management

Information security management interfaces with change management, since changes required by security will be implemented through the change management process and security will be a key contributor to CAB discussion on many services. Every significant change will be assessed for its potential impact on information security management.

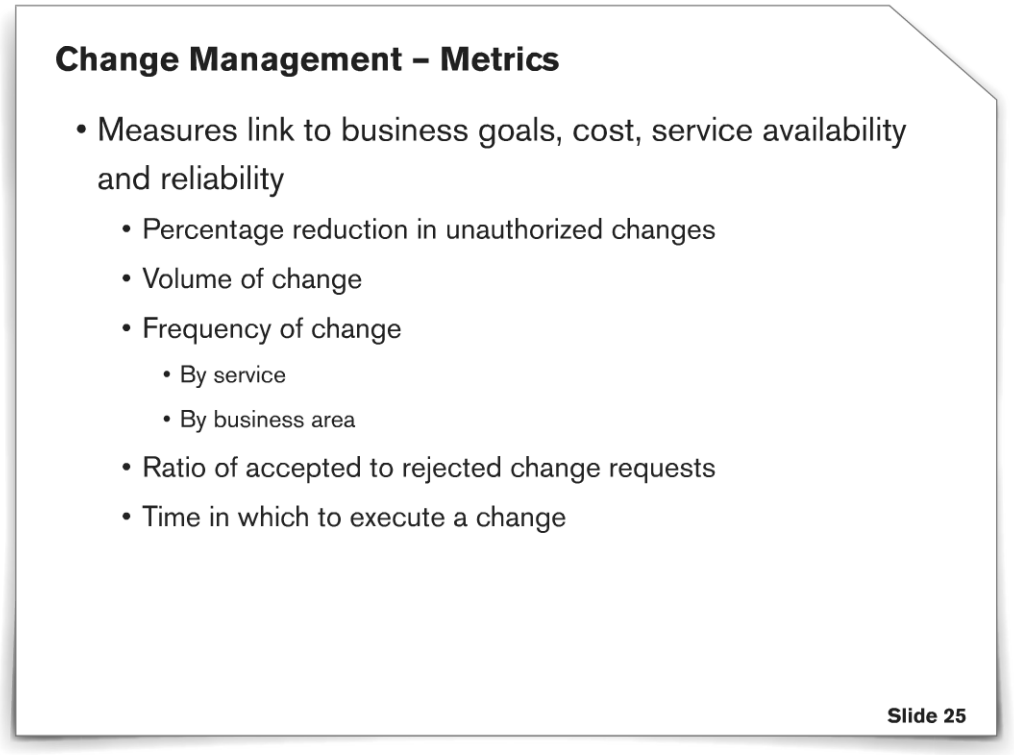
Capacity management and demand management

Capacity management and demand management are critical aspects of change management. Poorly managed demand is a source of cost and risk for service providers because there is always a level of uncertainty associated with the demand for services. Capacity management has an important role in assessing proposed changes – not only the individual changes but the total impact of changes on service capacity. Changes arising from capacity management, including those set out in the capacity plan, will be initiated as RFCs through the change process.

Service portfolio management

The service portfolio management process prioritizes and charters strategic changes, and submits change proposals for these. Change proposals will be a significant input to long-term planning for the change schedule, and will also be a key input to help change management review and authorize related RFCs.

Some change requests will require analysis by the service portfolio management process, potentially adding to the service pipeline. Each organization should define criteria for deciding whether these requests are managed as part of the change management process or are passed to service portfolio management.

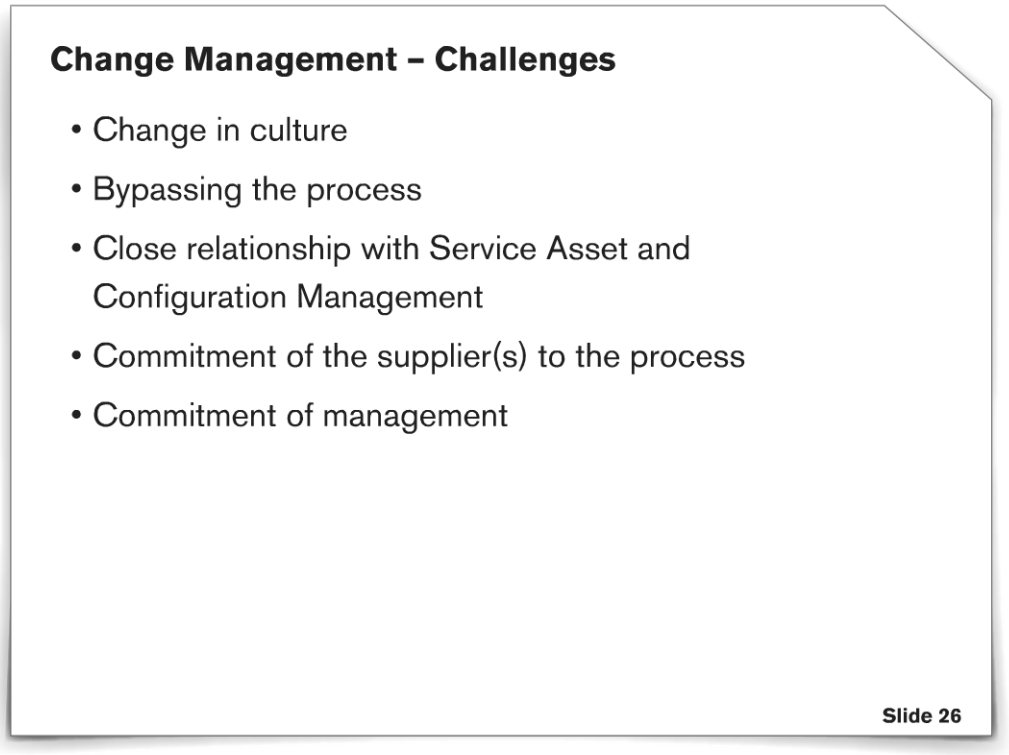


Change Management – Metrics

While it is relatively easy to count the number of incidents that eventually generate changes, it is infinitely more valuable to look at the underlying cause of such changes and to identify trends. It would be better still to measure the impact of changes and to demonstrate reduced disruption over time due to the introduction of Change Management, and also to measure the speed and effectiveness with which the IT infrastructure responds to identified business needs.

The Key Performance Indicators for Change Management are:

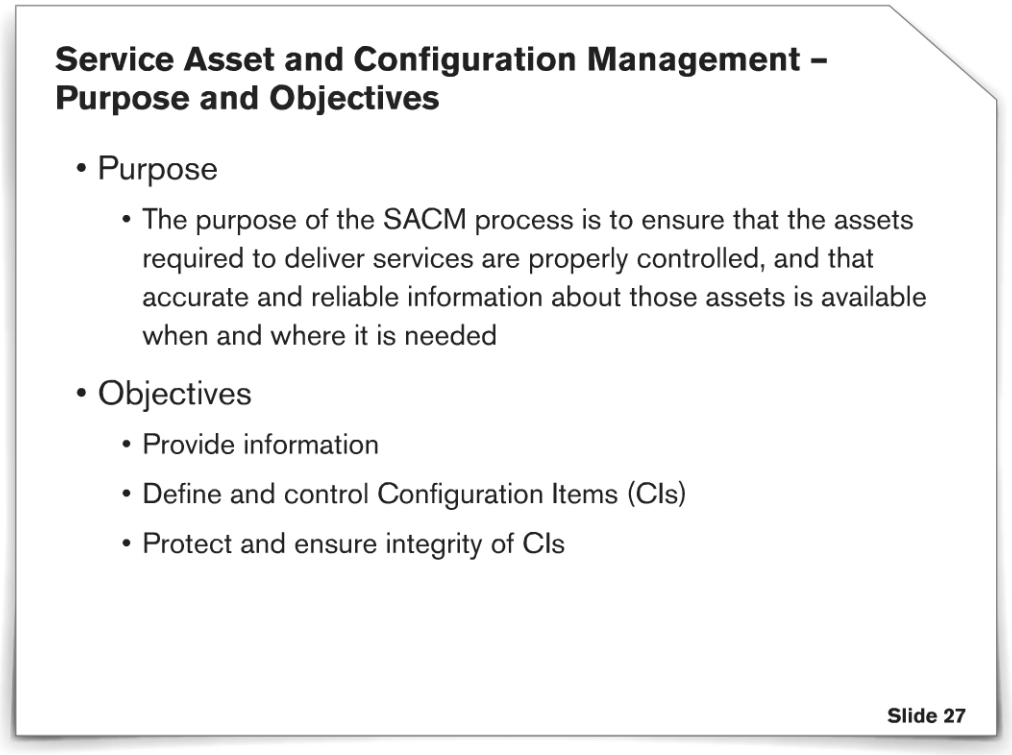
- Reduction in the number of changes where remediation is invoked
- Reduction in the number of failed changes
- Average time to implement based on urgency, priority, or change type
- Incidents attributable to changes
- Accuracy percentage in change estimate
- Number of RFCs (accepted or rejected)
- Number of emergency changes as a percentage of the total



Change Management – Challenges

Challenges affecting Change Management:

- Change in culture: A central process comes into place that influences everyone's activities
- Bypassing: Projects dodging the Change Management process
- Must have a close relationship with Service Asset and Configuration Management: To execute a controlled change, all data MUST be reliable. Change Management relies on configuration data, whereas Service Asset and Configuration Management relies on Change Management for the information on changes
- Commitment of the supplier(s) to the process
- Commitment of management

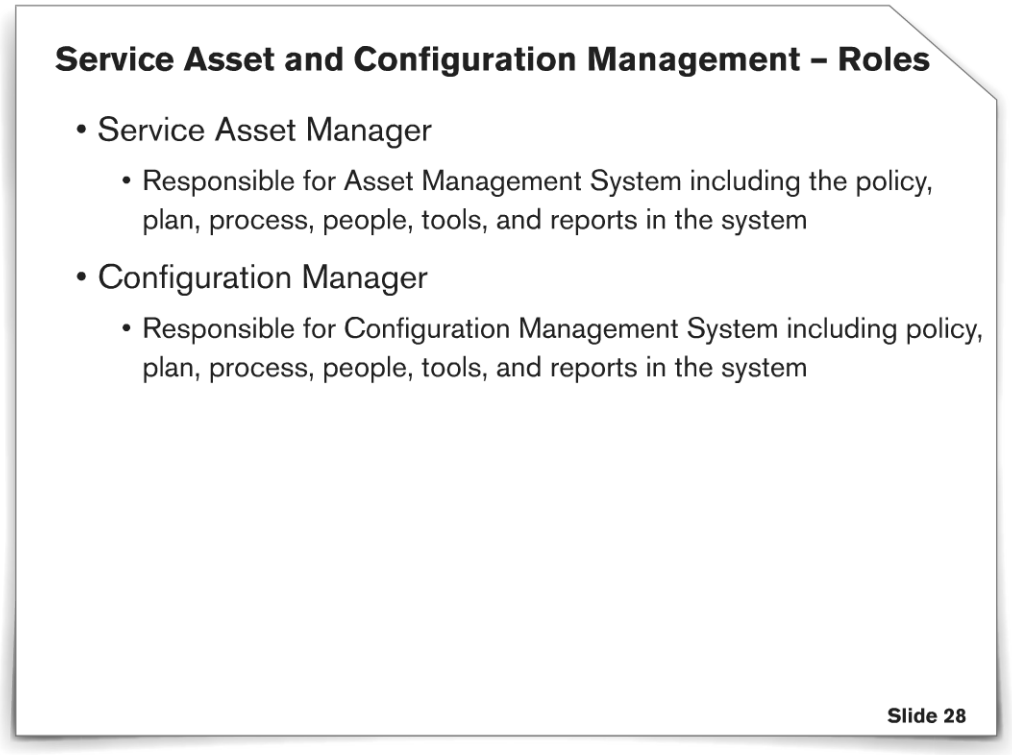


Service Asset and Configuration Management – Purpose and Objectives

PURPOSE: The purpose of the SACM process is to ensure 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.

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.



Service Asset and Configuration – Roles

Service Asset Manager

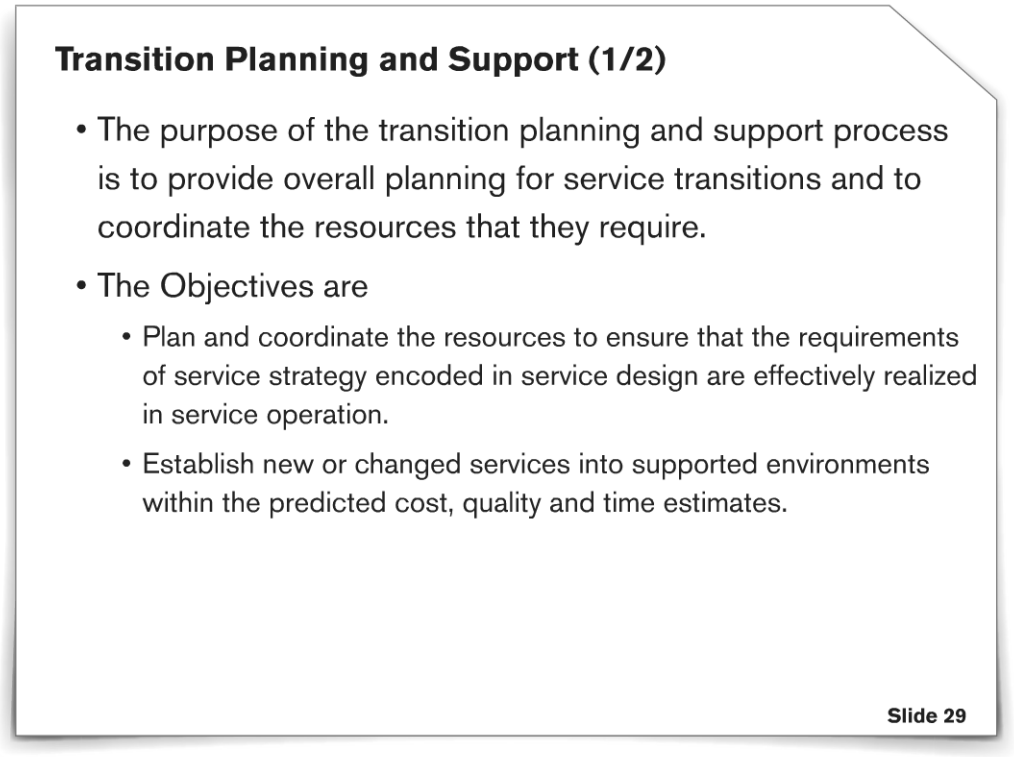
The Service Asset Manager has the following responsibilities:

- Works to the overall objectives agreed upon with the IT Services Manager; implements the organization's Service Asset Management policy and standards
- Evaluates existing Asset Management Systems, plans, implements and manages changed systems, and provides reporting on progress against plan
- Agrees upon the scope of the Asset Management processes. Develops and manages Asset Management standards, plans and procedures and manages implementation
- Mounts an awareness campaign on procedures, controls changes to Asset Management methods and processes and communicates these to staff before implementation. The Service Asset Manager also oversees implementation of new Asset Management Systems.
- Manages the evaluation of proprietary Asset Management tools and recommends suitable tools
- Agrees upon which assets will be uniquely identified with naming conventions and compliance

Configuration Manager

The Service Configuration Manager has the following responsibilities:

- Works to the overall objectives agreed upon with the IT Services Manager; implements the organization's Configuration Management policy and standards
- Evaluates existing Configuration Management Systems, plans, implements and manages changed systems, and provides reporting on progress against plan
- Agrees upon scope of the Configuration Management processes. Develops and manages Configuration Management standards, plans and procedures and manages implementation
- Mounts an awareness campaign on procedures, controls changes to Configuration Management methods and processes and communicates these to staff before being implemented. The Configuration Manager also oversees implementation of new Configuration Management Systems
- Manages the evaluation of proprietary Configuration Management tools and recommends suitable tools
- Agrees upon assets to be uniquely identified with naming conventions and compliance
- Plans population of the CMS. Manages CMS, central libraries, tools, common codes and data. Ensures regular housekeeping of the CMS
- Provides reports, including management reports, impact analysis reports and Configuration status reports



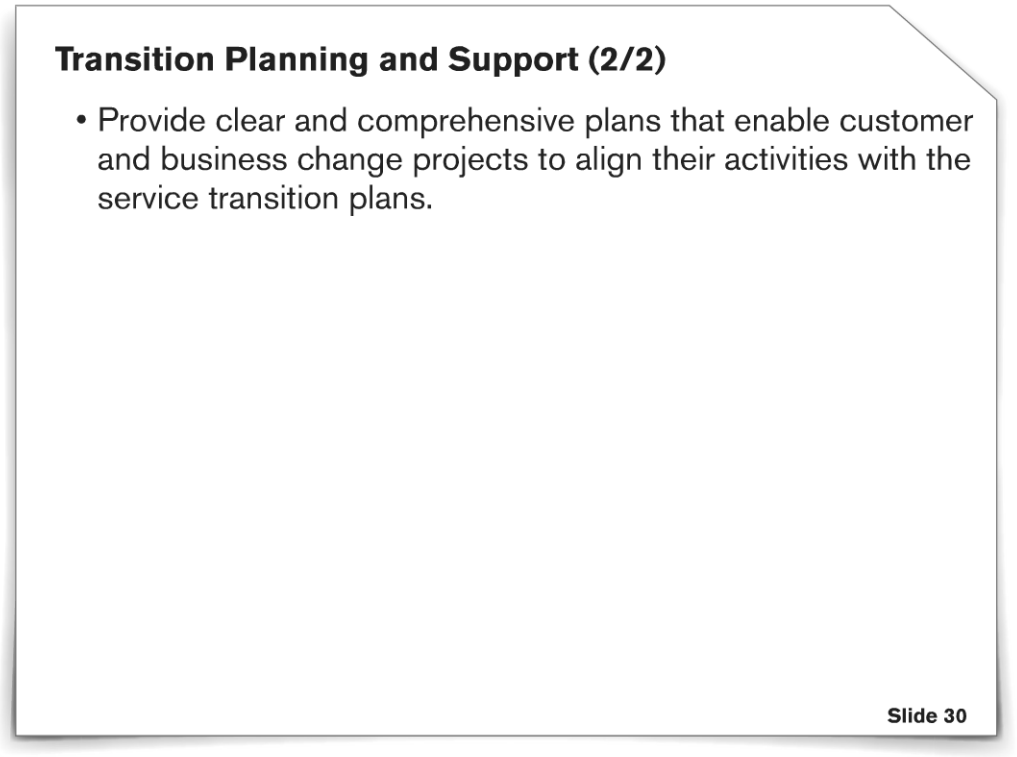
Transition Planning and Support (1/2)

Purpose and objectives

The purpose of the transition planning and support process is to provide overall planning for service transitions and to coordinate the resources that they require.

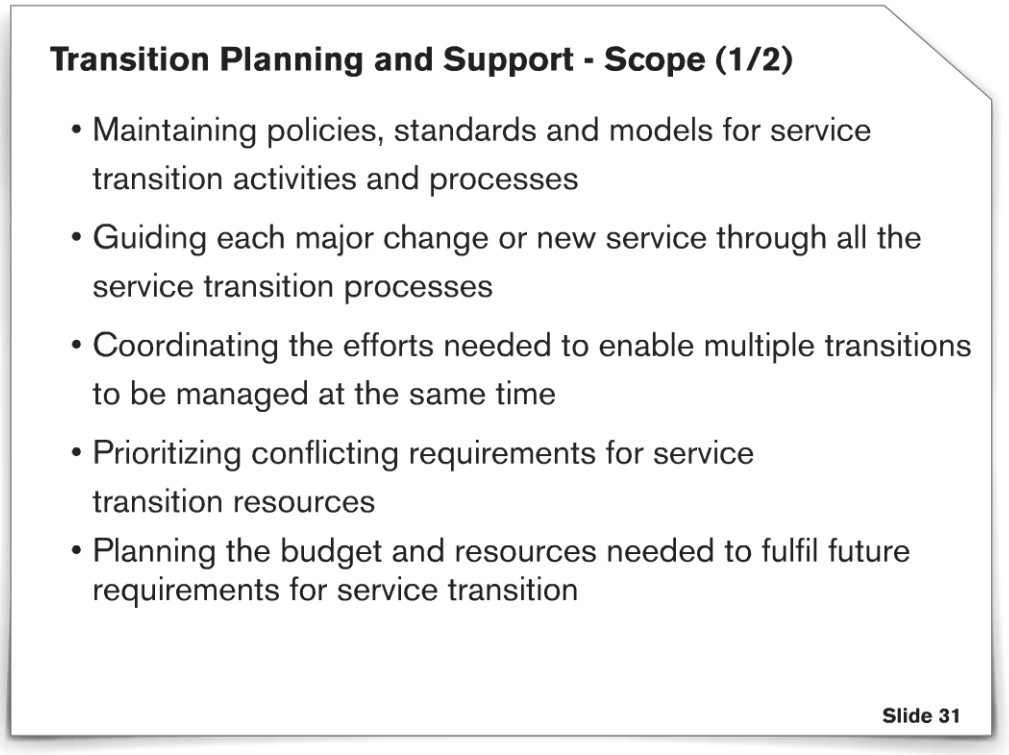
The objectives of transition planning and support are to:

- Plan and coordinate the resources to ensure that the requirements of service strategy encoded in service design are effectively realized in service operation.
- Coordinate activities across projects, suppliers and service teams where required.
- Establish new or changed services into supported environments within the predicted cost, quality and time estimates.
- Establish new or modified management information systems and tools, technology and management architectures, service management processes, and measurement methods and metrics to meet requirements established during the service design stage of the lifecycle.
- Ensure that all parties adopt the common framework of standard re-usable processes and supporting systems in order to improve the effectiveness and efficiency of the integrated planning and coordination activities.



Transition Planning and Support (2/2)

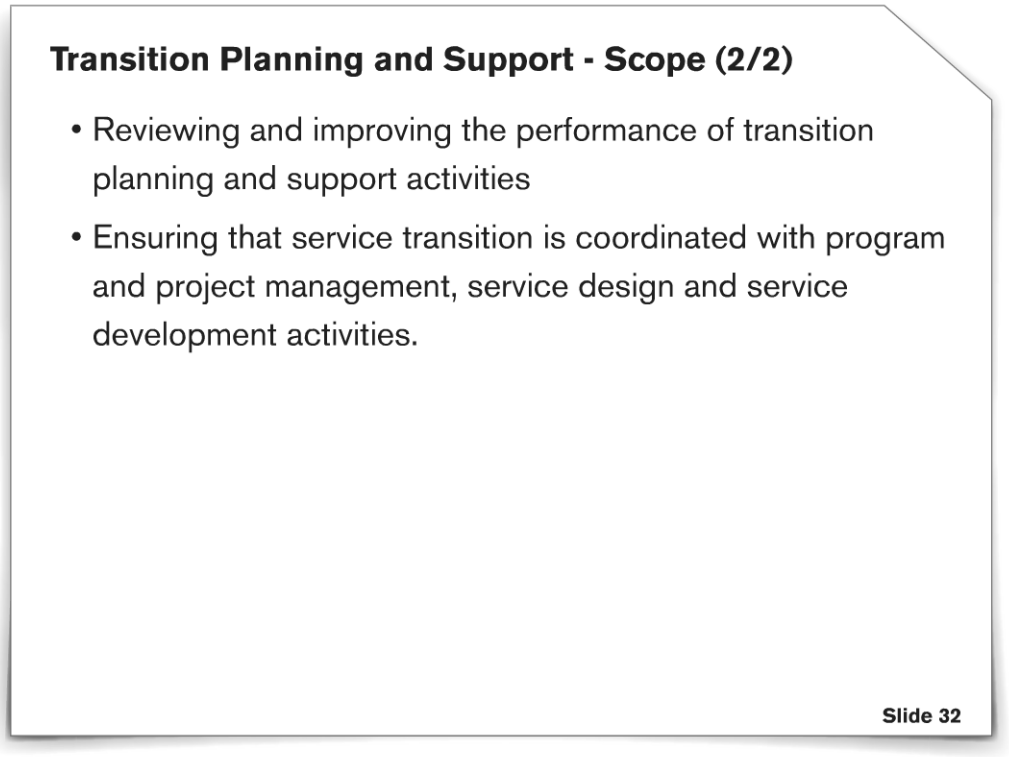
- Provide clear and comprehensive plans that enable customer and business change projects to align their activities with the service transition plans.
- Identify, manage and control risks, to minimize the chance of failure and disruption across transition activities; and ensure that service transition issues, risks and deviations are reported to the appropriate stakeholders and decision makers.
- Monitor and improve the performance of the service transition lifecycle stage.



Transition Planning and Support - Scope (1/2)

The scope of transition planning and support includes:

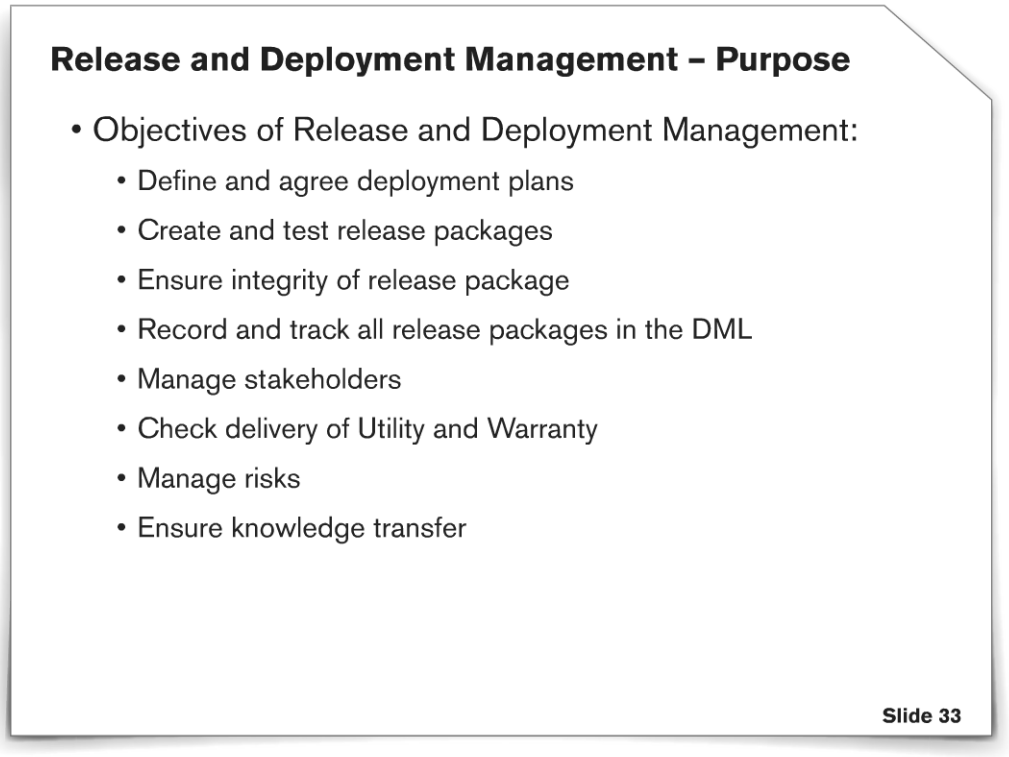
- Maintaining policies, standards and models for service transition activities and processes
- Guiding each major change or new service through all the service transition processes
- Coordinating the efforts needed to enable multiple transitions to be managed at the same time
- Prioritizing conflicting requirements for service transition resources
- Planning the budget and resources needed to fulfil future requirements for service transition



Transition Planning and Support - Scope (2/2)

- Reviewing and improving the performance of transition planning and support activities
- Ensuring that service transition is coordinated with program and project management, service design and service development activities.

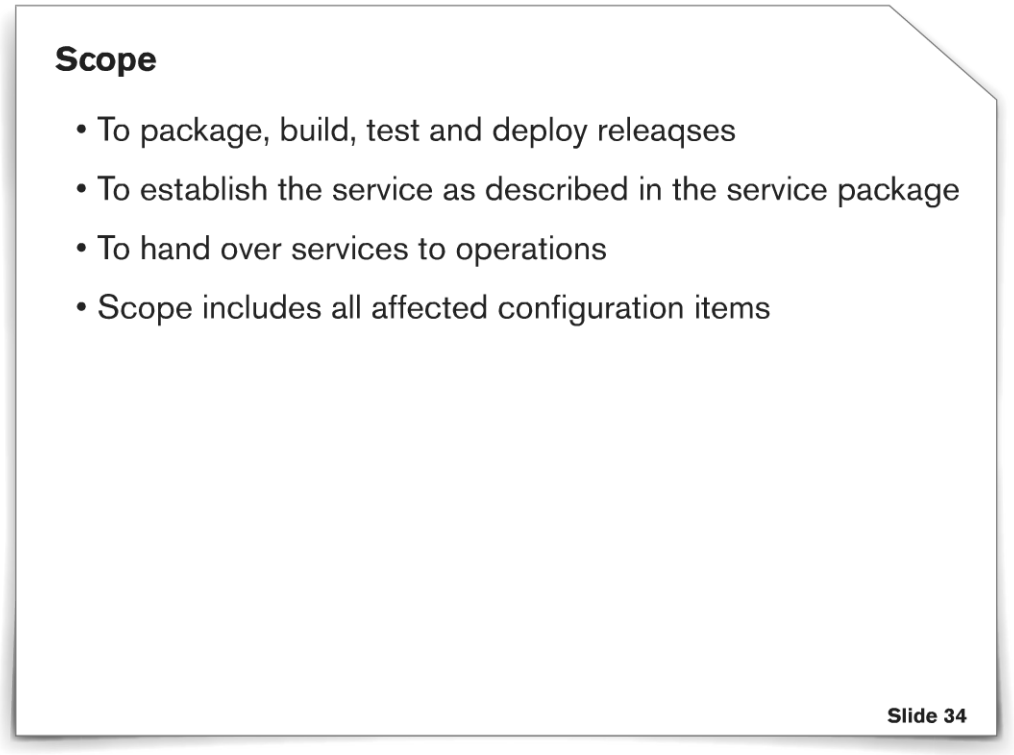
Transition planning and support is not responsible for detailed planning of the build, test and deployment of individual changes or releases; these activities are carried out as part of change management and release and deployment management.



Release and Deployment Management – Purpose

The objectives of release and deployment management are to:

- Define and agree release and deployment management plans with customers and stakeholders
- Create and test release packages that consist of related configuration items that are compatible with each other
- Ensure that the integrity of a release package and its constituent components is maintained
- Ensure that all release packages are stored in a DML and recorded accurately in the CMS
- Deploy release packages from the DML to the live environment following an agreed plan and schedule
- Ensure that releases can be tracked, installed, tested, verified and/or uninstalled or backed out if appropriate
- Ensure that organization and stakeholder change is managed during release and deployment activities
- Ensure that a new or changed service is capable of delivering the agreed utility and warranty
- Record and manage deviations, risks and issues and take necessary corrective action
- Ensure that there is knowledge transfer to enable the customers and users to optimize their use of the service
- Ensure that skills and knowledge are transferred to service operation functions



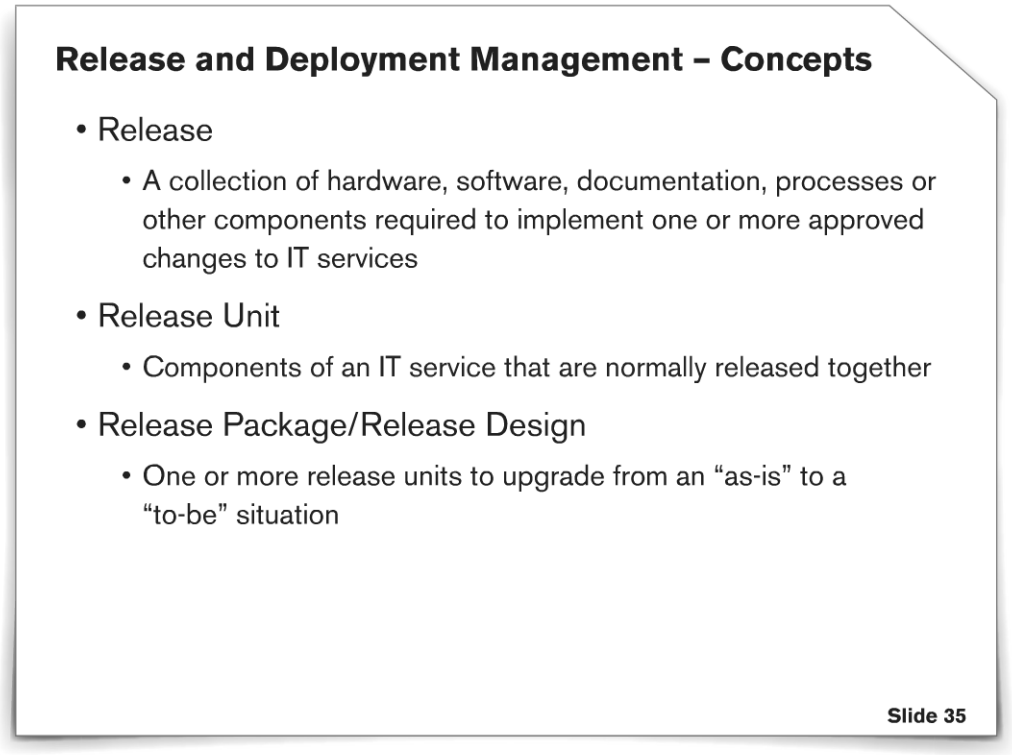
Scope

The scope of release and deployment management includes the processes, systems and functions to package, build, test and deploy a release into live use, establish the service specified in the service design package, and formally hand the service over to the service operation functions. The scope includes all configuration items required to implement a release, for example:

- Physical assets such as a server or network
- Virtual assets such as a virtual server or virtual storage
- Applications and software
- Training for users and IT staff
- Services, including all related contracts and agreements

Although release and deployment management is responsible for ensuring that appropriate testing takes place, the actual testing is carried out as part of the service validation and testing process.

Release and deployment management is not responsible for authoring changes, and requires authorization from change management at various stages in the lifecycle of a release.



Release and Deployment Management – Concepts

Release:

A Release is defined as the collection of hardware, software, documentation, processes or other components required to implement one or more approved changes to IT services.

Release Unit:

A Release Unit describes the portion of a service or IT infrastructure that is normally released together according to the organization's release policy. The unit may vary, depending on the type(s) or item(s) of Service Asset or service component such as software and hardware.

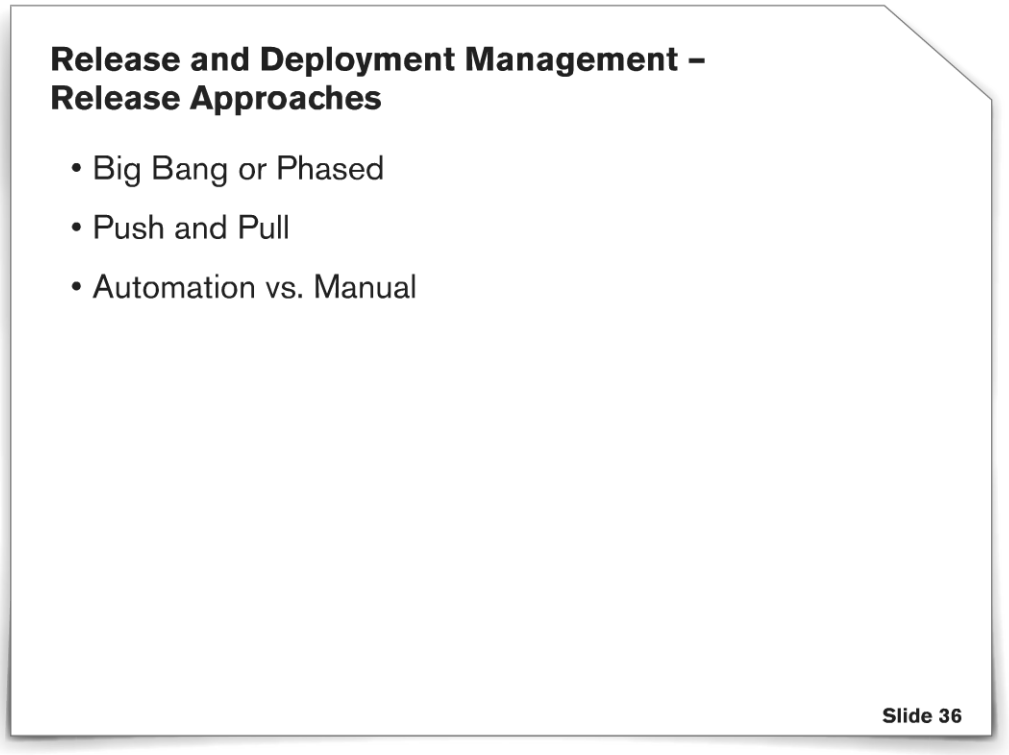
The following factors should be taken into account when deciding the appropriate level for Release Units:

- The ease and amount of change necessary to release and deploy a Release Unit
- The amount of resources and time needed to build, test, distribute and implement a Release Unit
- The complexity of interfaces between the proposed unit and the rest of the services

Release Package/Release Design:

A Release Package is used to upgrade an IT service from a current situation ("as-is") to the desired situation ("to-be"). With proper release planning, the contents of a Release Package are predetermined and the organization works towards developing this Release Package. It is also possible that an organization will work bottom-up to generate updates (release units), and at a certain interval, these release units will be combined into a Release Package.

A Release Package contains everything that is required to move from the current situation to the "to-be" situation, including software, hardware, documentation and perhaps training.



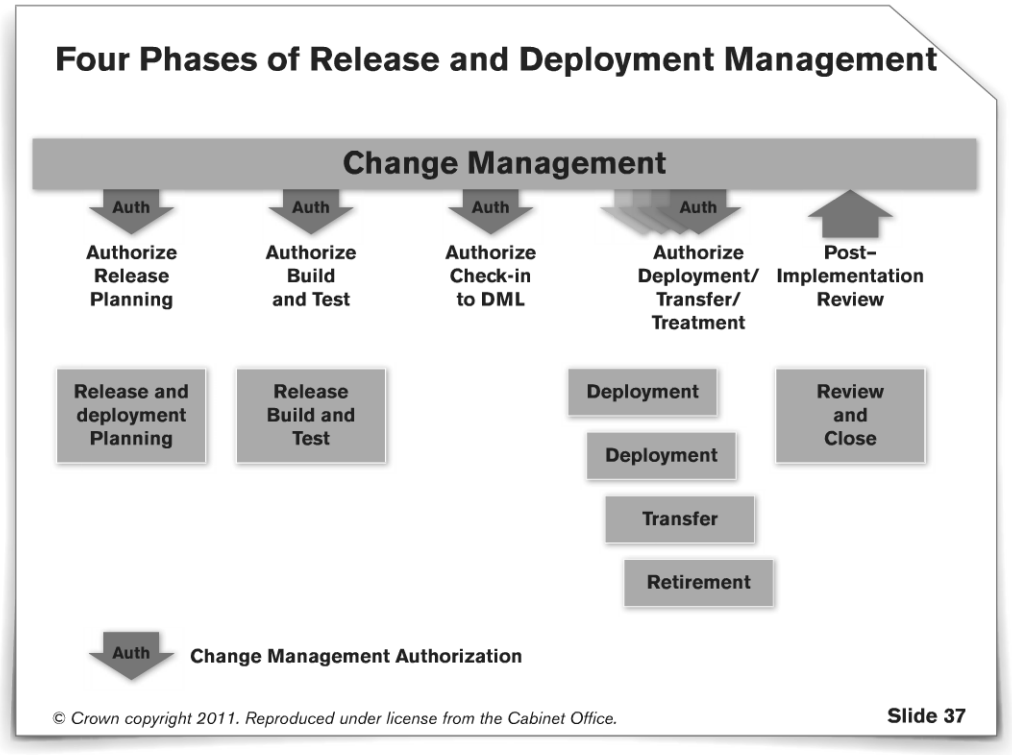
Release and Deployment Management – Release Approaches

The following options can have a significant impact on the release and deployment resources as well as the business outcomes. It is important to understand the Patterns of Business Activity (PBA) and user profiles when planning and designing the releases.

Approaches:

- **Big Bang Option** is used when the new or changed service is deployed to all user areas in one operation. This will often be used when introducing an application change and consistency of service across the organization is considered important
- **Phased Approach** is when the service is initially deployed to a part of the user base and then this operation is repeated for subsequent parts of the user base via a scheduled rollout plan. This will be the case in many scenarios such as in retail organizations, for when new services are being introduced in manageable phases into the stores' environment.
- A **Push Approach** is used where the service component is deployed from the center and pushed out to the target locations. In terms of service deployment, delivering updated service components to all users–either in big-bang or in phased form–constitutes a "push," since the new or changed service is delivered into the users environment at a time not of their choosing.

- A **Pull Approach** is used for software releases where the software is made available in a central location but users are free to pull the software down to their own location at a time of their choosing or when a user workstation restarts. A good example is virus signature updates, which are typically pulled down to update PCs and servers when it best suits the customer. However, at times of extreme virus risk this may be overridden by a release that is pushed to all known users.
- **Automation** will help to ensure repeatability and consistency. The time required to provide a well-designed and efficient automated mechanism may not always be available or viable. If a **manual** mechanism is used it is important to monitor and measure the impact of many repeated manual activities as they are likely to be inefficient and error-prone. Too many manual activities will slow down the release team and create resource/capacity issues that affect the service levels



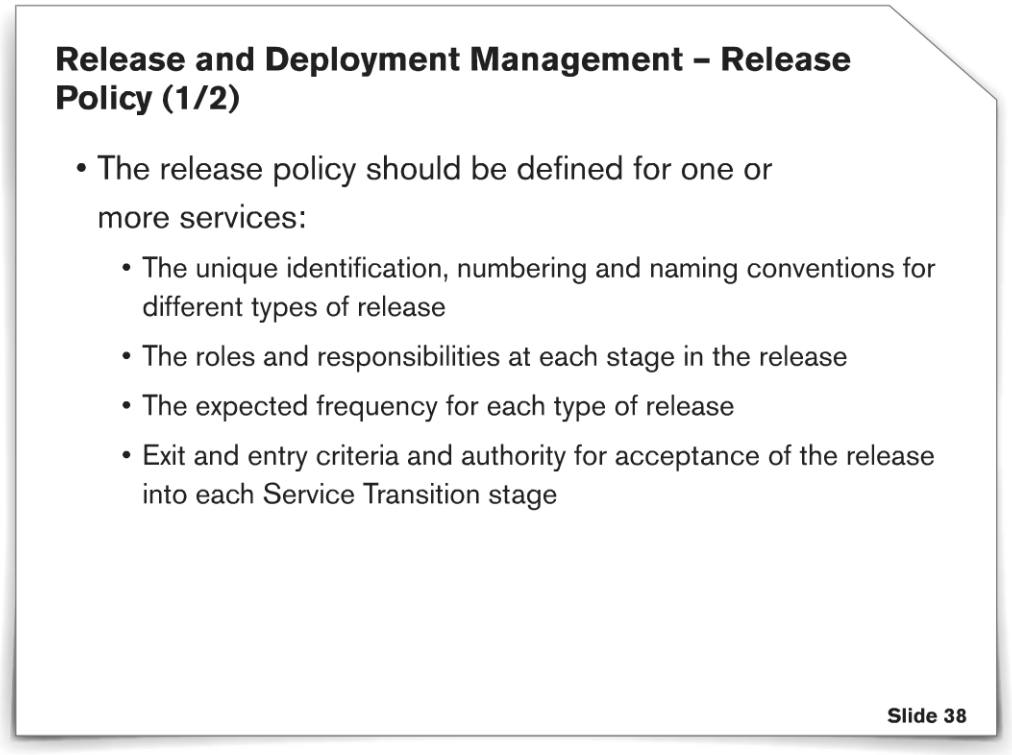
Four Phases of Release and Deployment Management

Release and deployment planning Plans for creating and deploying the release are created. This phase starts with change management authorization to plan a release and ends with change management authorization to create the release.

Release build and test The release package is built, tested and checked into the DML. This phase starts with change management authorization to build the release and ends with change management authorization for the baseline release package to be checked into the DML by service asset and configuration management. This phase only happens once for each release.

Deployment The release package in the DML is deployed to the live environment. This phase starts with change management authorization to deploy the release package to one or more target environments and ends with handover to the service operation functions and early life support. There may be many separate deployment phases for each release, depending on the planned deployment options.

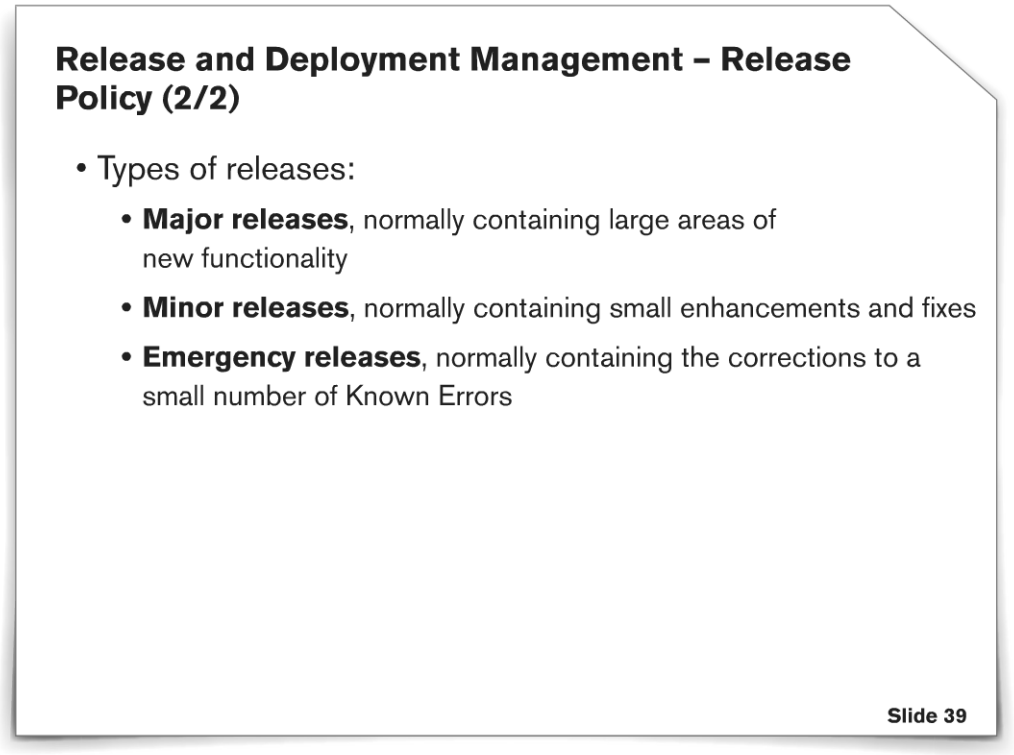
Review and close Experience and feedback are captured, performance targets and achievements are reviewed and lessons are learned.



Release and Deployment Management – Release Policy (1/2)

The release policy should be defined for one or more services and include:

- The unique identification, numbering and naming conventions for different types of release, together with a description
- The roles and responsibilities at each stage in the release and deployment process
- The expected frequency for each type of release
- The approach for accepting and grouping changes into a release (e.g., how enhancements are prioritized for inclusion)
- The mechanism to automate the build, installation and release distribution processes to improve reuse, repeatability and efficiency
- How the configuration baseline for the release is captured and verified against the actual release contents (e.g., hardware, software, documentation and knowledge)
- Exit and entry criteria and authority for acceptance of the release into each Service Transition stage and into the controlled test, training, disaster recovery and production environments
- Criteria and authorization to exit early life support and handover to Service Operations

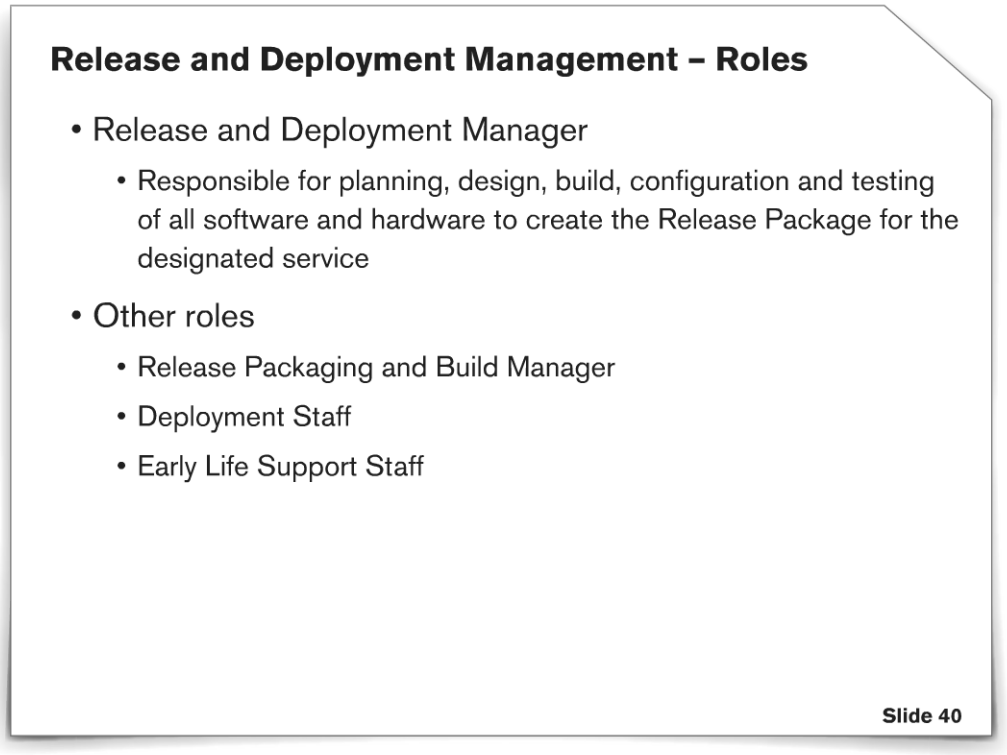


Release and Deployment Management – Release Policy (2/2)

The types of release should be defined as this helps to set customer and stakeholder expectations about the planned releases. A typical example is:

- **Major releases**, normally containing large areas of new functionality, some of which may eliminate temporary fixes to Problems. A major upgrade or release usually supersedes all preceding minor upgrades, releases and emergency fixes.
- **Minor releases**, normally containing small enhancements and fixes, some of which may already have been issued as emergency fixes. A minor upgrade or release usually supersedes all preceding emergency fixes.
- **Emergency releases**, normally containing the corrections to a small number of Known Errors or sometimes an enhancement to meet a high priority business requirement.

A release policy may say, for example, that only strict "emergency fixes" will be issued in between formally planned releases of enhancements and non-urgent corrections.



Release and Deployment Management – Roles

Several roles may be identified within Release and Deployment Management. Whether or not these are present within an organization depends upon the size of the organization and the maturity of the Release and Deployment Management process.

The main role is the **Release and Deployment Manager**, which is the process manager for the entire process.

Other roles are:

Release Packaging and Build Manager, with the following responsibilities:

- Establishes the final release configuration (e.g., knowledge, information, hardware, software and infrastructure)
- Builds the final release delivery
- Tests the final delivery prior to independent testing
- Establishes and reports outstanding Known Errors and work-arounds
- Provides input to the final sign-off implementation process

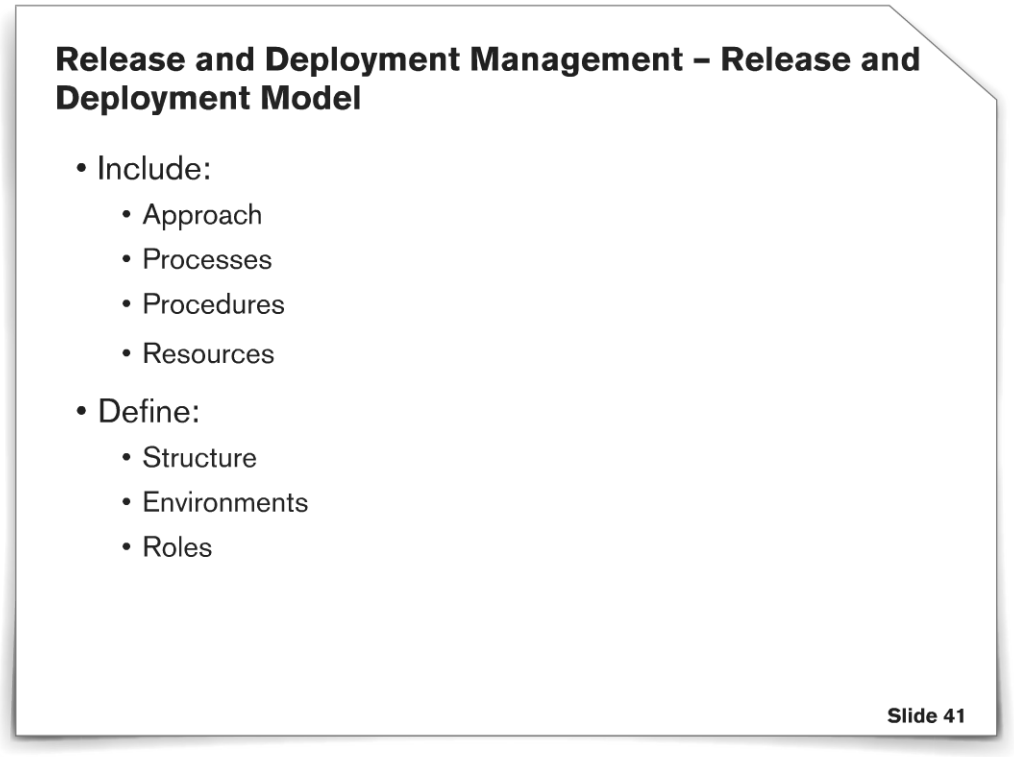
Deployment Staff, with the following responsibilities:

- The final physical delivery of the service implementation
- Coordinates release documentation and communications, including training and customer,

Service Management and technical release notes

- Plans the deployment in conjunction with change and Service Knowledge Management System and SACM (Service Asset Configuration Management System)
- Provides technical and application guidance and support throughout the release process, including Known Errors and work-arounds
- Provides feedback on the effectiveness of the release
- Records metrics for deployment to ensure within agreed upon SLAs

Early Life Support Staff may provide help throughout the release and deployment phase and also for the first period of operation. For example, by solving incidents with the new or changed service, or by perfecting service documentation.



Release and Deployment Management – Release and Deployment Model

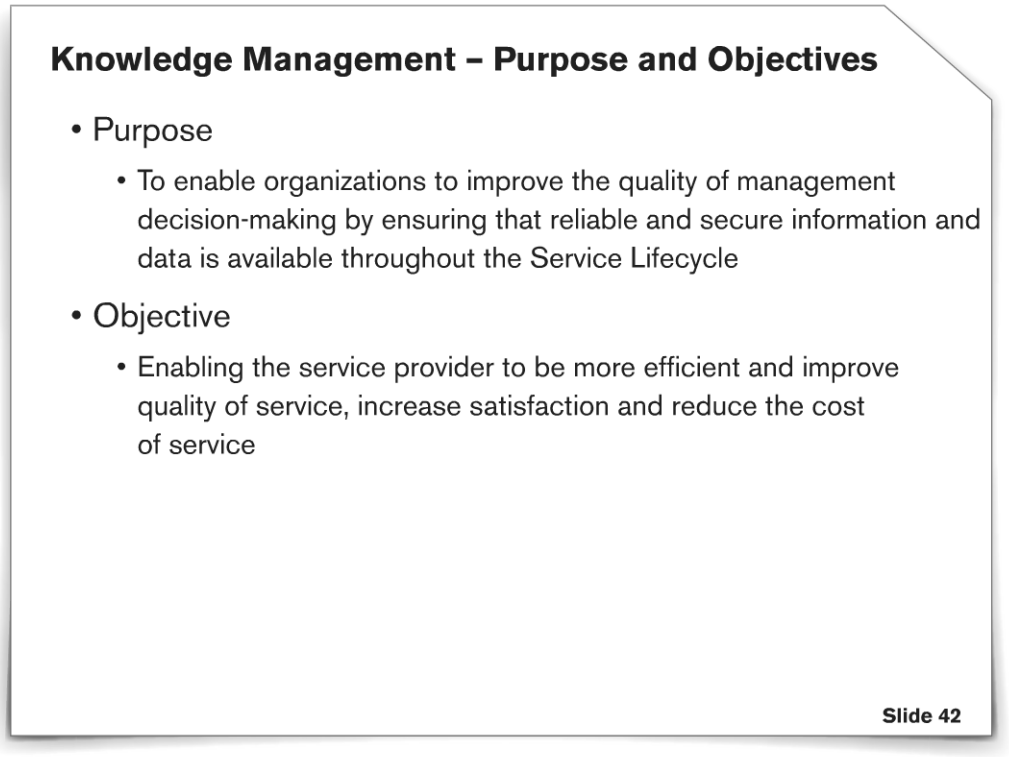
A service may be deployed into the production environment in a number of ways. Service Design will select the most suitable release and deployment models that include the approach, mechanisms, processes, procedures and resources required to build and deploy the release on time and within budget.

The release methods during the early build and test stages may differ significantly from live operations, so plan ahead to ensure that appropriate release methods are adopted at the appropriate time.

Release and deployment models define:

- Release structure, which is the overall structure for building a Release Package and the target environments
- The exit and entry criteria, including mandatory and optional deliverables and documentation for each stage
- Controlled environments required to build and test the release for each release level. There will be multiple logical and physical environments through the Service Transition stage mapped to different physical environments available to the transition team.
- The roles and responsibilities for each Configuration Item at each release level
- The release promotion and configuration baseline model
- Template release and deployment schedules

- Supporting systems, tools and procedures for documenting and tracking all release and deployment activities
- The handover activities and responsibilities for executing the handover for each stage of release and deployment



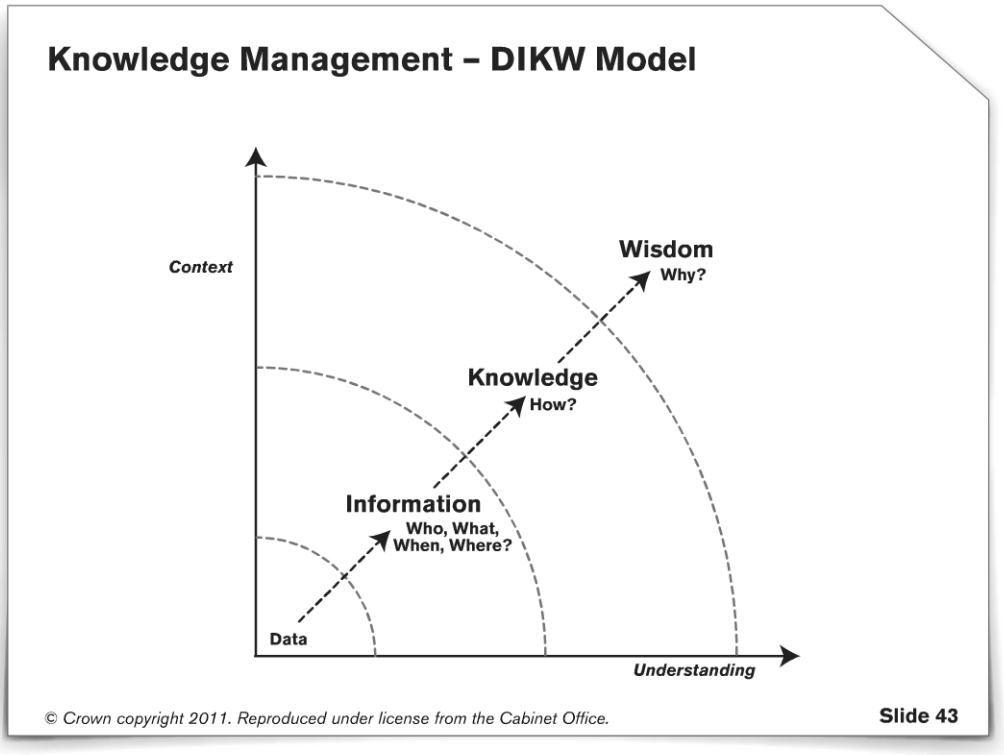
Knowledge Management – Purpose and Objectives

PURPOSE: To enable organizations to improve the quality of management decision-making by ensuring that reliable and secure information and data is available throughout the Service Lifecycle.

The purpose of Knowledge Management is to ensure that the right information is delivered to the appropriate place or competent person at the right time to enable informed decision.

The objectives of Knowledge Management include:

- Enabling the service provider to be more efficient and improve quality of service, increase satisfaction and reduce the cost of service
- Ensuring staff have a clear and common understanding of the value that their services provide to customers and the ways in which benefits are realized from the use of those services
- Ensuring that, at a given time and location, service provider staff have adequate information on:
  - Who is currently using their services
  - The current states of consumption
  - Service delivery constraints
  - Difficulties faced by the customer in fully realizing the benefits expected from the service.



Knowledge Management – Data, Information, Knowledge, Wisdom (DIKW) Model

The DIKW model is found at the basis for Knowledge Management.

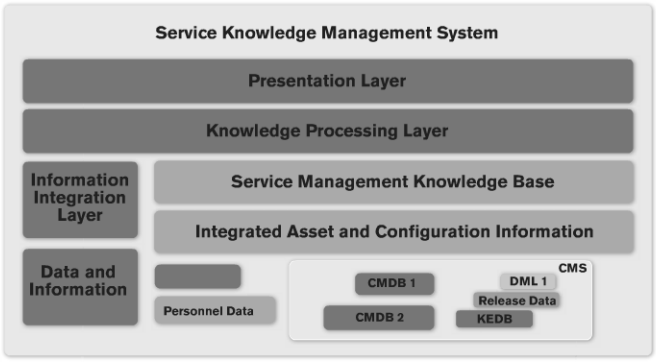
- Data is a set of discrete facts about events (raw measurements). Organizations often capture this in large amounts and put this data in structured databases
- Information is generated when context is added to data and can be stored in semi-structured content, such as documents, e-mail, and multimedia. Simple questions like "Who?, What?, When?, Where?" can be answered when information is available
- Knowledge is composed of the tacit experiences, ideas, insights, values and judgments of individuals. People can gain knowledge from their own expertise, but also from others' expertise or by analyzing data/information. By synthesizing these elements, new knowledge is created. Knowledge helps in answering the "How?" questions
- Wisdom can be seen as the ultimate understanding of the material and allows decisions based on common sense and insight. Using wisdom, the "Why?" questions can be answered

Data is stored in traditional database, like a CMDB, whereas information can be found in systems as the Configuration Management System (CMS). Knowledge is stored in the Service Knowledge Management System (SKMS). Wisdom cannot be stored or managed by a tool.

Many organizations are stuck at gathering data, and do not use it sufficiently. When information, knowledge, and ultimately wisdom can be extracted from the organization, better decisions and improvements can be made. Therefore, proper Knowledge Management is important for all life-cycle stages.

Knowledge Management – Service Knowledge Management System

- The system covers broader aspects of Knowledge Management, creates knowledge from data, and facilitates decision-making.



Slide 44

Knowledge Management – Service Knowledge Management System

The quality and relevance of the knowledge rests in turn upon the accessibility, quality and continued relevance of the underpinning data and information available to service staff. The goal of Knowledge Management enables organizations to improve the quality of management decision-making by ensuring that reliable and secure information and data is available throughout the Service Lifecycle. Specifically within IT Service Management, Knowledge Management's focus will be within the Service Knowledge Management System concerned, as its name implies, with knowledge. Underpinning this knowledge will be a considerable quantity of data that will be held in a central logical repository or Configuration Management System (CMS) and Database (CMDB).

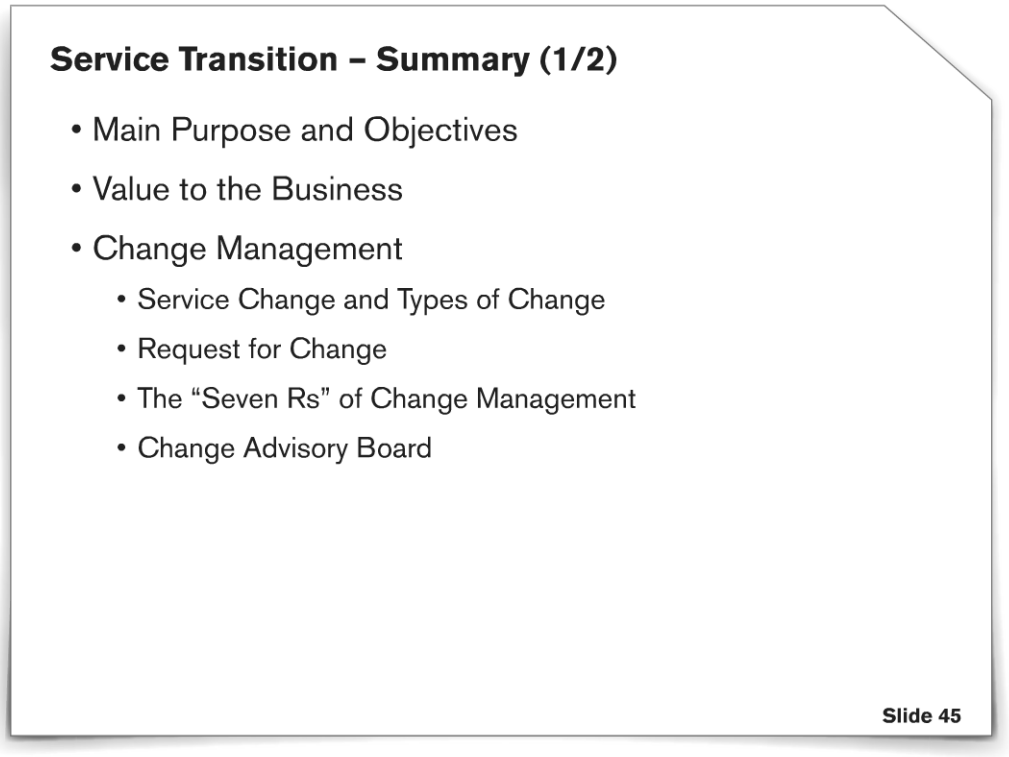
The ability to deliver a quality service or process relies upon the ability of those involved to respond to circumstances—and that in turn rests heavily upon their understanding of the situation, the options and the consequences and benefits (i.e., their knowledge of the situation they are, or may, find themselves in). That knowledge within the Service Transition domain might include:

- Identity of stakeholders
- Acceptable risk levels and performance expectations
- Available resource and timeframes

However, the Service Knowledge Management System (SKMS) is clearly a broader concept that covers a much wider base of knowledge, for example:

- The experience of staff
- Records of peripheral matters (e.g., weather, user numbers and behavior, organization's performance figures)
- Supplier and partner requirements, abilities and expectations
- Typical and anticipated user skill levels

Implementation of a Service Knowledge Management System helps reduce the costs of maintaining and managing the services, both by increasing the efficiency of operational management procedures and by reducing the risks that arise from the lack of proper mechanisms.



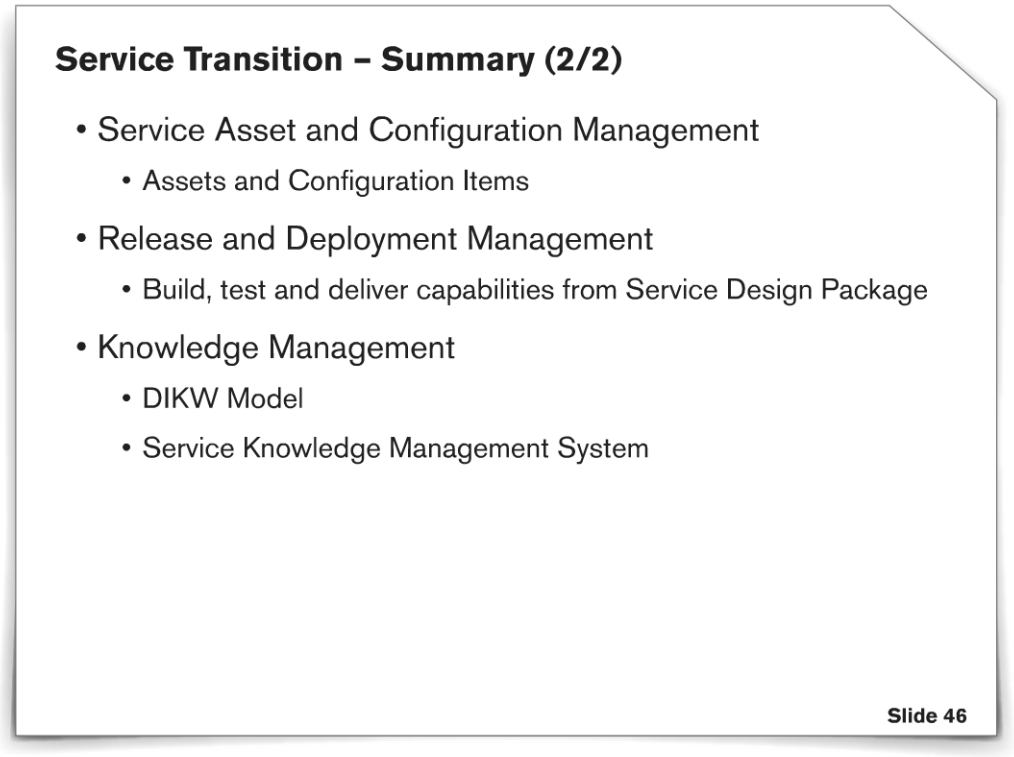
Service Transition – Summary (1/2)

This module presented **Service Transition**.

The **main Purpose** of Service Transition is to introduce new services (or change the existing services) with appropriate balance of speed, cost, safety and focus on customer expectations and requirements. This brings value to the business by balancing the required amount of change with a certain stability of the IT, with reduced risks and a better time to market.

One of the main processes within Service Transition is the Change Management process. Some key concepts are a service change, the several types of change and the Request for Change (RFC). The "Seven Rs" of Change Management indicate what to take into account during evaluation of a Change Request.

The Change Advisory Board (CAB and ECAB) and other levels for change authorization provide insight into the way changes are managed within the organization.



Service Transition – Summary (2/2)

Service Asset and Configuration Management plays an important role throughout the Service Lifecycle. Its objective is to define and control the components of services and infrastructure and maintain accurate Configuration Records. This enables an organization to comply with corporate governance requirements, control its asset base, optimize its costs, manage change and releases effectively and resolve Incidents and Problems faster.

The activities of **Release and Deployment Management** are mainly focused on the Service Transition stage of the Service Lifecycle. These activities aim to build, test and deliver the capability to provide the services specified by Service Design and which, in turn, will accomplish the stakeholders' requirements and deliver the intended objectives.

Knowledge Management also plays an important role throughout the Service Lifecycle. It enables organizations to improve the quality of management decision-making by ensuring that reliable and secure information and data is available throughout the Service Lifecycle.