

CHANGE MANAGEMENT EXPLAINED: CHANGE IN SERVICE MANAGEMENT, DEVOPS & MORE



Translating customer requirements into actual products and [services](#) is one of the main value streams of [service management](#). And the activities behind the building, testing and deployment of these products and services are usually enabled and controlled by the change management practice.

According to [VeriSM](#), change management is normally implemented as a process that:

1. Reviews and approves (or rejects) a proposed change
2. Manages it through its development and deployment

At the heart of controlling changes is [risk management](#), to protect the service provider and its customers from unnecessary negative impact of changes, including service degradation and regulatory penalties.

Successful service management requires proactively managing system changes associated with configurations, resource provisioning, and service management. These changes are not always planned, often emerging as unforeseen consequences of a service disruption. Technology dependencies in a complex IT infrastructure environment can cause the impact of a small change in the infrastructure to escalate across the IT environment and impact multiple users at scale.

Organizations therefore need to adopt prudent change management strategies, solutions, and practices that help manage service and component changes, and mitigate the associated risks. Let's

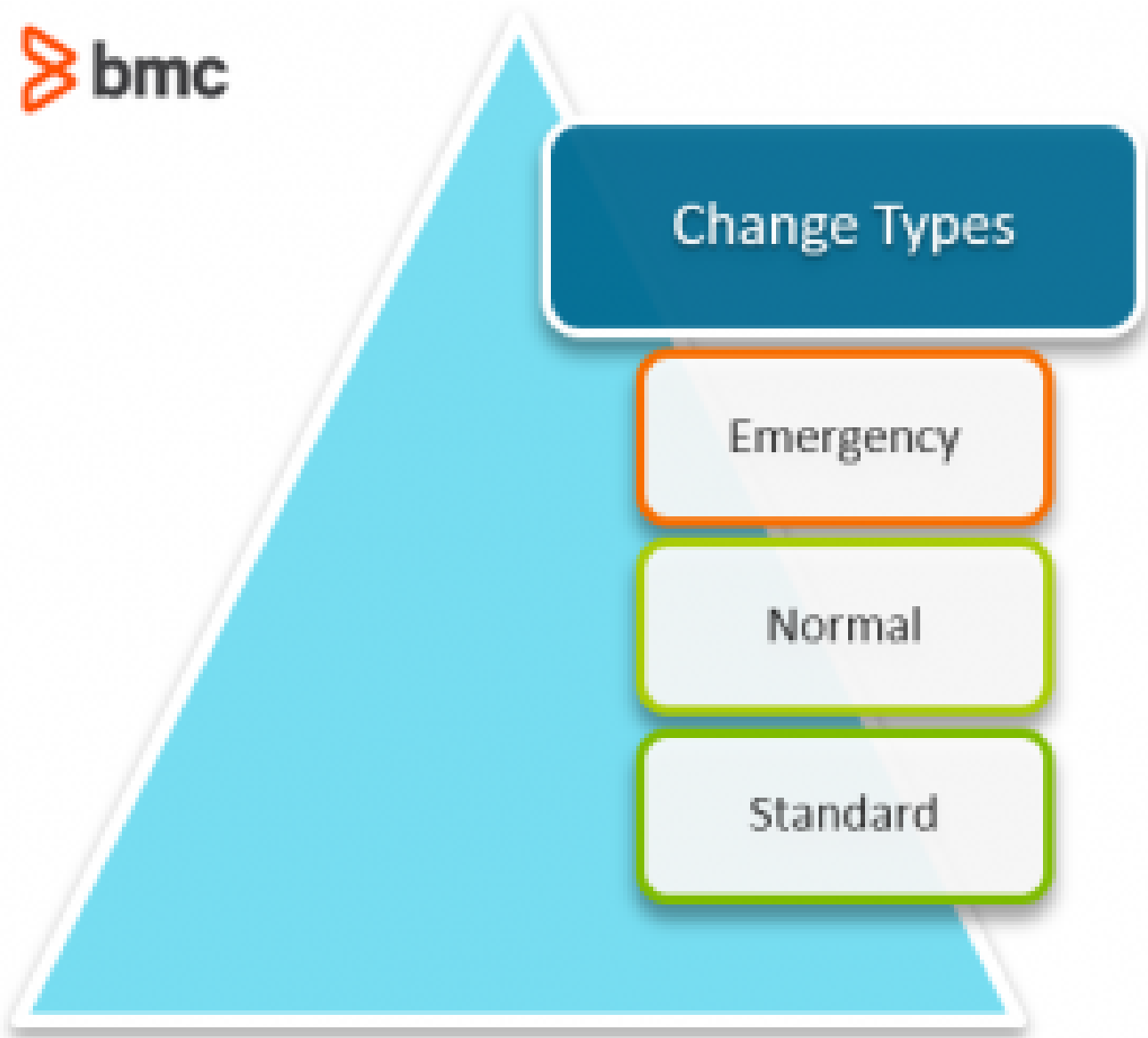
take a look at change management activities, tools, and best practice approaches.

What is change?

The ITIL® 4 change enablement practice defines change as:

“the addition, modification or removal of anything that could have a direct or indirect effect on services.”

Though [service management frameworks](#) vary in best-practice workflows, most follow a standard hierarchy of change:



Standard change

A low-risk change that is pre-authorized and follows documented tasks per a change model, which outlines a repeatable workflow to manage such changes.

Examples include an IT service request from the service desk platform.

Normal change

An intermediary or high-risk change that cannot be categorized as an urgent issue or a pre-approved change process. A thorough review process is required before approving such changes.

Emergency change

Urgent changes that may present high-risk consequences if not addressed promptly. An emergency change may need to be resolved to avoid a [major incident](#) or to resume normal operations following the incident occurrence.

Examples include:

- An upgrade to address an active information security threat
- Failover to an alternate data center due to power outages

Change management activities

The main activities involved in managing changes include:

1. **Recording changes.** Identifying the description of the change, who and what is involved, and potentially what is impacted. These records might be maintained in an ITSM solution or a Kanban board.
2. **Planning changes.** Considering resources—time and business needs—to determine the best way and scheduling for successful implementation. Plans would also include testing activities, as well as measures to be taken should the change be unsuccessful including rolling back to previous versions.
3. **Approving changes.** Using appropriate mechanisms to approve the change having considered the plans and risk mitigations. Approval mechanisms include automated workflows, peer reviews, or designated authorities based on the change level and preferred approach.
4. **Communicating changes.** Planned and approved change schedules would need to be communicated to relevant stakeholders to ensure alignment and preparedness for all eventualities.
5. **Review changes.** After execution, conduct a post-implementation review to consider what went well, what went wrong, and what opportunities for improvement exist.

Change management approaches

The approach to handling changes differs from one organization to another and depending on the types of change.

Software based changes can be automated end to end, by taking advantage of CI/CD technologies to execute changes frequently and quickly. For example, [Etsy](#) is known to perform 50 deployments per day through a fully automated and [continuous delivery pipeline](#).

Physical infrastructure changes such as equipment installation may be slower, requiring a staged approach.

When it comes to organizational approaches, here are some common ones:

- Some organizations choose to [freeze changes](#) during certain peak periods to reduce the risks

of service outages and poor customer experience.

- Other organizations have a stringent governance approach to change approval, needing a select group of people (like a Change Advisory Board—CAB for short) to hold scheduled meetings to discuss and approve all major changes.
- Others decentralize change authority depending on service ownership and risk levels.

What matters most in all these approaches is delivery of value for the organization, be it risk mitigation, agility, or reliability.

Prioritizing change

The approval process necessary for a change can depend on decision criteria that's unique for every organization. The decision criteria are based on priority, resource, cost, business need and other factors.

A common approach to prioritize change requests is by considering the impact and urgency.

Impact

Impact evaluates the business impact of a proposed change request. It also accounts for potentially damaging consequences resulting from unsuccessful execution of a change that are not previously considered.

The ranking may range from minor impact to extensive impact.

Urgency

Urgency evaluates the necessary time for a change to realize an Impact. A change request that requires quick implementation or one that must be initiated early to account for prolonged implementation duration is ranked with high urgency.

The ranking may range from Low Urgency to High Urgency.

Priority

Priority indicates the relative importance of a change request is determined by correlating the Impact and Urgency.

Here is an example of a priority matrix:



| | | Impact | | | |
|---------|--------|----------|--------|--------|------|
| | | Priority | Low | Medium | High |
| Urgency | High | Medium | High | High | |
| | Medium | Low | Medium | High | |
| | Low | Low | Low | Medium | |

Basic Impact, Urgency & Priority matrix

(Deep dive into the [impact urgency priority matrix](#) & this [BMC documentation](#).)

DevOps & change management

[DevOps](#) is a philosophy and a movement focusing on organization-wide collaboration to support the delivery of value to the organization and its customers. And at the heart of this collaboration is bringing together developers—who introduce change—and operations, who have to manage the effects of change.

A change management approach that does not have collaboration as one of its core values invariably leads to conflicts between those who desire change versus those who are pro-stability or risk averse. This has played out in many organizations where the change management process aspects (such as CAB meetings) end up becoming a bottleneck rather than a facilitator for change.

Because the digital age has necessitated faster delivery of new features, many organizations now have [digital transformation](#) at the heart of their strategy. And for tech functions, this means adopting technologies such as cloud and CI/CD, and approaches such as [Agile and DevOps](#) to meet these business needs.

The change management approach therefore has to evolve to suit this modern way of working. The change types therefore become a risk-based basis for the organization's change models.

Most standard changes can therefore go the [automation route](#) and require very little interference from upper management. If the change integrates, and all tests pass, then you push it into production.

Only high-risk changes that can severely impact the organization are subjected to appropriate levels of scrutiny during the approval process.

By the time these types of changes are being reviewed by upper management, all the technical teams already collaboratively participated in the planning process and are fully aligned in readiness for the change. This then shortens the time required to approve the change.

Success in change management

What ever the approach, there are four success factors that ITIL 4 suggests any organization practicing change management should aim for:

1. Ensuring that changes are realized in a timely and effective manner.
2. Minimizing the negative impacts of changes.
3. Ensuring stakeholder satisfaction.
4. Meeting change-related governance and compliance requirements.

The rapid evolution of customer needs and technology environments means that no one-size fits all approach can satisfy the needs of effective change management.

Organizations must consider appropriate governance frameworks that control the risks involved during changes, but at the same time do not introduce unnecessary bureaucracy that can prevent or delay value creation through changes.

Related reading

- [BMC Service Management Blog](#)
- [BMC DevOps Blog](#)
- [Roles & Responsibilities in Change Management](#)
- [Change Management in the Cloud](#)
- [Introduction to ECAB: Emergency Change Advisory Board](#)
- [Best Practices for Organizational Change Management](#)