

13.4 Umsetzung, Tests und Rollback

“ Kurz erklärt

Nach der Genehmigung beginnt die eigentliche Umsetzung eines Changes.

Ziel ist es, Änderungen kontrolliert, nachvollziehbar und mit möglichst geringem Risiko einzuführen.

Dazu gehören insbesondere sorgfältige Tests, eine geplante Durchführung, eine Überprüfung des Ergebnisses sowie ein vorbereiteter Rollback für den Fall unerwarteter Probleme.

Warum Tests wichtig sind

Auch sorgfältig geplante Änderungen können unerwartete Auswirkungen haben.

Beispiele:

- Anwendungen starten nicht mehr,
- Benutzer können sich nicht anmelden,
- Netzwerkverbindungen funktionieren nicht,
- Zertifikate werden nicht akzeptiert,
- Datenbanken reagieren langsamer.

Tests helfen, solche Probleme möglichst früh zu erkennen.

Testumgebungen

Änderungen sollten nach Möglichkeit zuerst in einer Testumgebung geprüft werden.

Typische Umgebungen:

- Entwicklungsumgebung (Development)
- Testumgebung (Test)
- Integrationsumgebung
- Staging
- Produktivsystem

Je ähnlicher die Testumgebung der Produktivumgebung ist,

desto aussagekräftiger sind die Testergebnisse.

Warum nicht direkt in der Produktion testen?

Tests im Produktivsystem können zu:

- Serviceunterbrechungen,
- Datenverlust,
- Sicherheitsproblemen,
- unerwarteten Ausfällen

führen.

Deshalb sollten Änderungen möglichst vorab getestet werden.

Arten von Tests

Je nach Change können unterschiedliche Tests erforderlich sein.

Beispiele:

- Funktionstest,
- Integrationstest,
- Lasttest,
- Sicherheitstest,
- Regressionstest,
- Benutzertest,
- Wiederherstellungstest.

Nicht jeder Change benötigt alle Testarten.

Funktionstest

Beim Funktionstest wird geprüft,

ob die neue oder geänderte Funktion wie vorgesehen arbeitet.

Beispiel:

Nach einer VPN-Aktualisierung wird getestet,

ob sich Benutzer erfolgreich verbinden können.

Regressionstest

Ein Regressionstest überprüft,

ob bestehende Funktionen nach einer Änderung weiterhin ordnungsgemäß arbeiten.

Beispiel:

Nach einem Datenbankupdate funktionieren weiterhin:

- Anmeldung,
- Berichtswesen,
- Druckfunktionen,
- API-Schnittstellen.

Regressionstests verhindern unbeabsichtigte Nebenwirkungen.

Pilotbetrieb

Bei größeren Änderungen erfolgt häufig zunächst ein Pilotbetrieb.

Dabei wird die Änderung nur für einen kleinen Benutzerkreis eingeführt.

Vorteile:

- geringeres Risiko,
- frühes Feedback,
- Fehler lassen sich leichter erkennen,
- Auswirkungen bleiben begrenzt.

Nach erfolgreichem Pilotbetrieb erfolgt die Einführung für alle Benutzer.

Wartungsfenster

Produktive Änderungen werden häufig innerhalb geplanter Wartungsfenster umgesetzt.

Vorteile:

- geringere Auswirkungen,
- ausreichende Zeit für Tests,
- genügend Zeit für Rollback,
- bessere Planbarkeit.

Größere Änderungen erfolgen häufig außerhalb der Hauptarbeitszeiten.

Durchführung des Changes

Während der Umsetzung sollten alle Schritte dokumentiert werden.

Typische Inhalte:

- Beginn,
- durchgeführte Arbeiten,
- Besonderheiten,
- Testergebnisse,
- Abweichungen,
- Abschlusszeit.

Eine vollständige Dokumentation erleichtert spätere Analysen.

Monitoring nach der Umsetzung

Nach Abschluss überwacht das Monitoring,

ob der Service ordnungsgemäß arbeitet.

Beispiele:

- Erreichbarkeit,
- CPU-Auslastung,
- Speicherverbrauch,
- Fehlermeldungen,
- Antwortzeiten,
- Netzwerkstatus.

Monitoring bestätigt den Erfolg des Changes.

Was ist ein Rollback?

Ein Rollback stellt den vorherigen Zustand wieder her,

wenn der Change nicht erfolgreich war.

Das Ziel besteht darin,

die Auswirkungen eines fehlgeschlagenen Changes möglichst gering zu halten.

Wann wird ein Rollback durchgeführt?

Beispiele:

- kritische Anwendungen funktionieren nicht,
- Service ist nicht verfügbar,
- Sicherheitsprobleme entstehen,

- Performance verschlechtert sich erheblich,
- unerwartete Fehler treten auf.

Nicht jeder Fehler führt sofort zu einem Rollback.

Die Entscheidung richtet sich nach Risiko und Auswirkungen.

Rollback-Plan

Ein Rollback sollte bereits vor Beginn des Changes vorbereitet werden.

Typische Inhalte:

- Auslösekriterien,
- Verantwortliche,
- Reihenfolge der Schritte,
- Wiederherstellung aus Backup,
- Zeitbedarf,
- Kommunikation.

Ein Rollback darf nicht erst während einer Störung geplant werden.

Beispiel eines Rollbacks



Ein klar definierter Ablauf verkürzt die Wiederherstellungszeit.

Backups vor einem Change

Vor vielen Änderungen wird ein Backup erstellt.

Beispiele:

- virtuelle Maschine,
- Datenbank,
- Konfiguration,
- Firewall,
- Netzwerkgerät,
- Active Directory.

Backups erleichtern den Rollback erheblich.

Verifizierung

Nach erfolgreicher Umsetzung wird geprüft,

ob der Change tatsächlich erfolgreich war.

Typische Fragen:

- Funktioniert der Service?
- Sind alle Tests erfolgreich?
- Gibt es Fehlermeldungen?
- Ist das Monitoring unauffällig?
- Wurde die Dokumentation aktualisiert?

Erst danach gilt der Change als abgeschlossen.

Post Implementation Review (PIR)

Nach größeren Änderungen wird häufig ein **Post Implementation Review (PIR)** durchgeführt.

Dabei wird bewertet:

- Wurde das Ziel erreicht?
- Gab es Probleme?
- War die Planung ausreichend?
- Hat der Rollback funktioniert?
- Welche Verbesserungen sind möglich?

Der PIR dient der kontinuierlichen Verbesserung zukünftiger Changes.

Dokumentation aktualisieren

Nach erfolgreichem Change sollten unter anderem aktualisiert werden:

- CMDB,
- Netzpläne,
- Runbooks,
- Architekturdiagramme,
- Knowledge Base,
- Betriebshandbücher.

Dadurch bleiben Informationen aktuell.

Praxisbeispiel

Ein Unternehmen aktualisiert seine zentrale Firewall.

Vorher:

- Backup erstellt,
- Rollback vorbereitet,
- Testplan erstellt.

Nach der Aktualisierung:

- Funktionstest erfolgreich,
- Monitoring ohne Auffälligkeiten,
- VPN-Verbindungen geprüft,
- Dokumentation aktualisiert.

Der Change wird anschließend erfolgreich abgeschlossen.

Typische Fehler

Fehler 1

Änderungen werden direkt produktiv getestet.

Fehler 2

Es existiert kein Rollback.

Fehler 3

Backups fehlen.

Fehler 4

Monitoring überprüft den Change nicht.

Fehler 5

Regressionstests werden ausgelassen.

Fehler 6

Dokumentationen werden nicht aktualisiert.

Fehler 7

Benutzer werden nicht informiert.

Fehler 8

Der Change wird zu früh abgeschlossen.

Fehler 9

Erfahrungen werden nicht ausgewertet.

Fehler 10

Testumgebung unterscheidet sich stark von der Produktivumgebung.

Checkliste Umsetzung

- ☐ Genehmigung vorhanden
 - ☐ Testumgebung vorbereitet
 - ☐ Wartungsfenster aktiv
 - ☐ Monitoring eingerichtet
 - ☐ Verantwortlichkeiten bekannt
 - ☐ Dokumentation vorbereitet
 - ☐ Kommunikation erfolgt
 - ☐ Durchführung protokolliert
-

Checkliste Rollback

- ☐ Rollback definiert
 - ☐ Backup vorhanden
 - ☐ Verantwortliche benannt
 - ☐ Zeitbedarf bekannt
 - ☐ Wiederherstellung getestet
 - ☐ Monitoring berücksichtigt
 - ☐ Dokumentation vorhanden
 - ☐ Auslösekriterien definiert
-

Bedeutung für Fachinformatiker für Systemintegration

Fachinformatiker setzen die meisten Changes praktisch um.

Typische Aufgaben:

- Testsysteme vorbereiten,
- Backups erstellen,
- Änderungen durchführen,
- Monitoring überwachen,
- Funktionstests durchführen,
- Rollback vorbereiten,
- Dokumentationen aktualisieren,
- Erfahrungen für zukünftige Changes festhalten.

Eine sorgfältige Umsetzung entscheidet häufig über den Erfolg eines Changes.

Zusammenfassung

“ Genehmigung erhalten

↓

Tests durchführen

↓

Backup erstellen

↓

Change umsetzen



Monitoring überprüfen



Bei Bedarf Rollback durchführen



Dokumentation aktualisieren



Post Implementation Review durchführen

Merksätze

“ Änderungen sollten möglichst nicht direkt in der Produktivumgebung getestet werden.

“ Ein Rollback muss vor Beginn des Changes geplant sein.

“ Monitoring bestätigt den Erfolg einer Änderung.

“ Dokumentationen gehören zum Abschluss jedes Changes.

“ Erfahrungen aus abgeschlossenen Changes verbessern zukünftige Änderungen.

Verwandte Seiten

- 13.1 Ziele und Grundlagen des Change Enablement
- 13.2 Change-Typen und Risikobewertung

- 13.3 Genehmigungen, Change Authority, CAB und Change-Kalender
 - 13.5 Zusammenspiel mit Incident, Problem, Service Configuration Management und Monitoring
 - Service Configuration Management
 - Monitoring and Event Management
 - Continual Improvement
-

Quellen und Versionsstand

Offizielle Grundlagen

- PeopleCert – ITIL Practice Guide: Change Enablement
- ITIL Foundation – Version 5

Einordnung

Die beschriebenen Vorgehensweisen zu Tests, Rollback, Monitoring und Post Implementation Review orientieren sich an den Empfehlungen der ITIL Practice „Change Enablement“. ITIL schreibt keine festen Testverfahren vor, empfiehlt jedoch ausdrücklich eine risikoorientierte Planung, geeignete Testmaßnahmen und eine nachvollziehbare Nachbereitung von Änderungen.

Behandelter Framework-Stand: ITIL Version 5

Zusätzlich berücksichtigt: aktuelle ITIL-4-Practice-Guidance

Fachlicher Stand: August 2026
