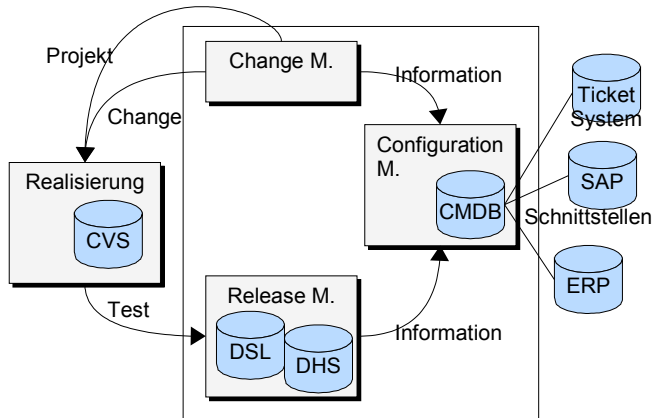


Konfigurationsmanagement	2
Grafik Konfigurationsmanagement.....	2
Beschreibung Datenbanken.....	2
Definition Konfiguration.....	2
Zweck Konfigurationsmanagement.....	2
Funktionen im ITSM.....	2
Operationelle Funktionen.....	2
Konfigurationseinheiten	3
Definition Konfigurationseinheit.....	3
CIs Hardware.....	3
CIs Software.....	3
CIs Dokumentation.....	3
Dokument-Verwaltung.....	3
Metadaten	4
Definition Metadaten.....	4
Verknüpfungsarten.....	4
Attribute Hardware.....	4
Attribute Software / Dokumente.....	4
Lebenszyklus-Statistiken.....	4
Implementation Konfigurationsmanagement	5
Vorgehen KM-Einführung.....	5
Planung.....	5
Festlegung der Konventionen.....	5
Inventur.....	5
Verwaltung Konfigurationssystem	6
Kriterien für KM-Tool.....	6
Filesystem.....	6
Dokumentenmanagement-System DMS.....	6
Konfigurationsmanagement-Datenbank CMDB.....	6
Umsysteme.....	6
Konfigurationsmanagement-Prozess	7
Konfigurationsmanager.....	7
Kernprozesse.....	7
Grafik KM-Prozess.....	7
Beschreibung KM-Prozess.....	7
Grafik Entwicklungsprozess.....	7
Software Version Manager SVM.....	7
Changemanagement-Prozess	8
Change Manager.....	8
Request for Change RfC.....	8
Grafik ChM-Prozess.....	8
Beschreibung ChM-Prozess.....	8
Beurteilungskriterien Kategorie.....	8
Prioritätsstufen.....	8
Entscheidungs-Raster.....	8
Erfolgskontrolle.....	8

Grafik Konfigurationsmanagement



Beschreibung Datenbanken

- CMDB:** Configuration Management Database
Informationen über im Betrieb stehende CI
Database = Datensammlung, nicht zwingend Datenbank
- DSL:** Definitive Software Library
aktuell gültige Software
- DHS:** Definitive Hardware Store
Hardware Ersatzteile mit Basiskonfiguration

Definition Konfiguration

- Gesamtheit aller Bestandteile eines IT-Systems, evtl. aufgeteilt in Gruppen, mit ihren Attributen und Verknüpfungen (Inventar)
- Speicherung auf CMDB
- Darstellung mit Configuration Management Plan, Logisches Modell der IT-Infrastruktur, z.B. ERM
- Configuration Baseline: gesicherter Ist-Zustand zu einem definierten Zeitpunkt, Hauptrelease

Zweck Konfigurationsmanagement

- Erfassung und eindeutige Bezeichnung aller CIs
- aktuelle und korrekte Informationen über alle CIs
- Transparenz über Stati und Abhängigkeiten
- Nachvollziehbarkeit vorgenommener Änderungen
- Wiederherstellbarkeit früherer Konfigurationen
- Kontrolle der eingesetzten Vermögenswerte
- verbesserte Revisionsfähigkeit
- Grundlage für wirtschaftliche und qualitativ hochstehende Informatikdienstleistungen

Funktionen im ITSM

Bereitstellung von Informationen für andere Service-Management-Prozesse.

- Störungsmanagement:
HW/SW eines Benutzers kann sofort identifiziert werden
- Problemmanagement:
betroffene Benutzer / Prozesse, wichtig für Priorisierung
- Änderungsmanagement:
betroffene CIs mit aktuellen Statusinformationen
- Releasemanagement:
aktuelle Versionen

Operationelle Funktionen

- neue CI registrieren
- alte CI archivieren
- CI verwalten:
 - Attribute (Status, Standort, Eigentümer) ändern
 - Abhängigkeiten ändern
- CI ein- und auschecken
- Variante erstellen (für Emergency)
- Versionen vergleichen und mergen
- CMDB mit dem wirklichen System vergleichen
- Konfigurations-Reports erstellen

Konfigurationseinheiten

KONF

Definition Konfigurationseinheit

Kleinste einzeln auswechselbare Einheit einer Konfiguration, Configuration Item CI / KE. Identifiziert durch Name, Typ, Version.

- Hardware
- Software
- Dokumente

Detaillierungsgrad entsprechend den Bedürfnissen des Unternehmens.

CIs Hardware

- Workstation: Bildschirm, PC, Tastatur, Maus
- Server
- Peripheriegeräte: Drucker
- Netzwerkkomponenten

CIs Software

- Sourcecode
- ausführbare Programme
- Programm-Bibliotheken
- Online-Hilfe
- GUIs
- Compiler, Linker, Generatoren, Installationsprogramm
- Demonstrationsprogramm
- Datenbanken
- Data Dictionary
- Office-Paket: Textverarbeitung, Tabellenkalkulation, Präsentation, Kommunikation, Organisation

CIs Dokumentation

- Projektdokumentation: Auftrag, Arbeitspakete, Pflichtenheft, Offerte, Projektpläne
- Entwicklungsdokumentation: Spezifikationen, Ziele, Anforderungen, Use Case Diagramme
- Testdokumentation
- Systemdokumentation: Betriebshandbuch
- Benutzerdokumentation: Benutzerhandbuch, Anwendungsbeschreibung, Schulungsunterlagen
- Informationen über Benutzer, Mitarbeiter, Ressourcen
- Formulare: Request for Change RfC

Dokument-Verwaltung

- AKV definieren
- Namenskonventionen festlegen und kommunizieren
- Berechtigungen vergeben
- Dokument-Header: Version, Änderung, Datum, Wer
- Status: Entwurf, Final Draft, Freigegeben
- Benutzer über Änderungen informieren, evtl. automatische Alerts

Definition Metadaten

Daten über Daten, beschreiben Eigenschaften von Datensätzen hinsichtlich Inhalt, Struktur und Form.

Kategorien:

- inhaltsabhängig: Abstract, Schlagwörter
- identifizierend: Autor, Titel, ID, Version, Objektart
- administrativ: Status, Speicherort, Erstellungsdatum

Zweck:

- Daten finden
- Zugang
- Beziehung zu anderen Objekten
- Einordnung

Verknüpfungsarten

Verknüpfung:

- verbunden mit
- benutzt von

- Teil von
- Kopie von
- Variante: identische Funktionalität für unterschiedliche Schnittstelle

- Version: unterschiedliche Funktionalität

Beispiel:

- Terminal mit LAN
- Programm verwendet Subroutine
- Motherboard Teil von PC
- Software-Verteilung
- andere Sprache, Peripheriegeräte, Betriebssystem, wird parallel entwickelt
- wird durch neue Version abgelöst

Attribute Hardware

- Inventarnummer (eindeutig)
- Seriennummer
- Bezeichnung
- Kategorie
- Typ
- Status
- letzte Änderung
- Speicher
- Standort
- spezifische Attribute, z.B. Netzwerkkarte, Treiber, CPU, Takt in Mhz, RAM in MB

Attribute Software / Dokumente

- Name
- Dateityp
- Dateiversion
- Beschreibung
- Sprache
- Speicherort
- Status
- Grösse
- erstellt
- verändert
- letzter Zugriff

Lebenszyklus-Stati

Hardware:

- geplant
- bestellt
- geliefert
- getestet
- installiert
- deinstalliert
- entsorgt

Software:

- geplant
- in Entwicklung / in Bearbeitung
- bereit für Test
- getestet
- bereit für Roll-out / akzeptiert
- produktiv
- archiviert

Vorgehen KM-Einführung

- Planung
- Inventur
- Evaluation eines geeigneten Tools
- Installation CMDB, DSL, DHS und unterstützende Tools
- CI mit Attributen in CMDB aufnehmen (gruppenweise, Statusänderungen während Implementation beachten)
- Ausbildung aller Benutzer
- Benutzergruppen einrichten, Benutzerrechte erteilen
- Umstellung der Arbeitsprozesse
- laufende Richtigkeits- und Erfolgskontrolle

Planung

- Bewusstseinsentwicklung, Akzeptanz beim Management
- Ist-Analyse, Bedarfsanalyse, Ziele festlegen
- Prozesse neu strukturieren
- Konventionen festlegen
- Kontrollverantwortungen zuteilen
- Implementationsplan erstellen
- Betroffene informieren

Festlegung der Konventionen

- CI-Gruppen definieren
- Detaillierungsgrad festlegen
- Attribute pro CI-Gruppe definieren
- Namenskonvention für CIs definieren: kurze und aussagekräftige Namen, mehrstufige Versionsnummern
- Varianten-Erfassung definieren: ab wann anderer CI
- Lebenszyklus-Stati definieren
- Regeln für Lebenszyklus-Übergänge definieren

Inventur

Physische Namenszuteilung zu allen CI.

Erhebungstechniken:

- Erfassung vor Ort
- automatisches Scanning-Tool
- Dokumenten-Studium: Rechnungen, Verträge, Bestellungen

Identifikation:

- Hardwarekleber
- Kommentarzeile im Programmkopf

Kriterien für KM-Tool

- Integration in bestehende Entwicklungs- und Testumgebung
- Unterstützung der üblichen Standards (z.B. ITIL)
- Unterstützung anderer Prozesse wie Incident, Problem, Change und Release Management
- automatischer Import von CIs
- Überwachung, Controlling, Reporting
- einfache Bedienbarkeit, Übersichtlichkeit
- Anpassung an neue Anforderungen
- Kosten bei der Anschaffung und während des Betriebs

üblich ist oft ein Mix der Systeme

Filesystem

Ordnerstruktur auf Server mit Vereinbarung (Konvention) über Ablageort und Namen, v.a. für Dokumente.

Vorteile:

- rasch realisierbar
- keine Investitionskosten
- einfache Handhabung

Nachteile:

- Aktualisierung aufwändig
- fehleranfällig
- geringer Zugriffsschutz
- Versionierung manuell
- Auswertungen von Hand

Dokumentenmanagement-System DMS

System zur Ablage und Verwaltung von Dokumenten und Software.

Vorteile:

- automatische Versionierung
- Historisierung
- Zugriffsmechanismen auf Dokumentenebene
- Check-in / Check-out
- Attributerfassung

Nachteile:

- Verknüpfungen schwierig
- geringe Integration in Entwicklungsumgebung und Softwareverteilung
- Auswertungen aufwändig

Konfigurationsmanagement-Datenbank CMDB

Datenbank mit allen Informationen zu einer Konfiguration, mit Attributen der CIs und ihren Verknüpfungen.

Vorteile:

- Check-in / Check-out
- autom. Export / Import
- Relationen
- schnelle und flexible Auswertungen
- Integration in ITIL-Prozesse
- skalierbar
- Berechtigungen auf Attributebene

Nachteile:

- teuer
- komplex
- speicherintensiv
- braucht Fachpersonal

Umsysteme

- Element Manager: verwaltet System-oder Netzwerkkomponenten
- Enterprise Management System: Integration der diversen Element Management-Systeme auf einer zentralen Plattform meldet Konfigurationsänderungen, Zustands- und Fehlermeldungen, Alarme
- Enterprise Resource Planning ERP: übernimmt interne Kostenverrechnung, dient als Masterdatenbank

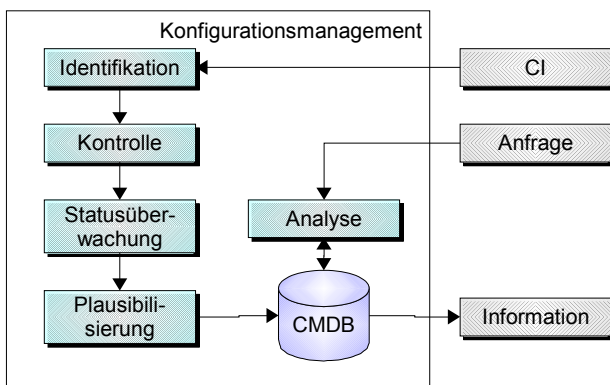
Konfigurationsmanager

- hat die Aufsicht über das Konfigurationsmanagement
- stellt den Konfigurationsmanagement-Prozess sicher
- sorgt für korrekte Nachführung von Änderungen
- erstellt Auswertungen für Projektleitung, Change Management und andere interessierte Stellen
- rapportiert regelmässig an das Management

Kernprozesse

- **Versionierungsprozess:**
Verwaltung der CIs mit Versionskontrolle, Nachvollziehbarkeit von Änderungen
- **Konfigurationsprozess:**
Zusammenstellung und Verwaltung von CIs, gesicherter Konfigurationszustand (Baseline)
- **Releasebildungsprozess:**
Identifizierung und Bereitstellung einer Baseline
Wiederherstellbarkeit früherer Konfigurationen

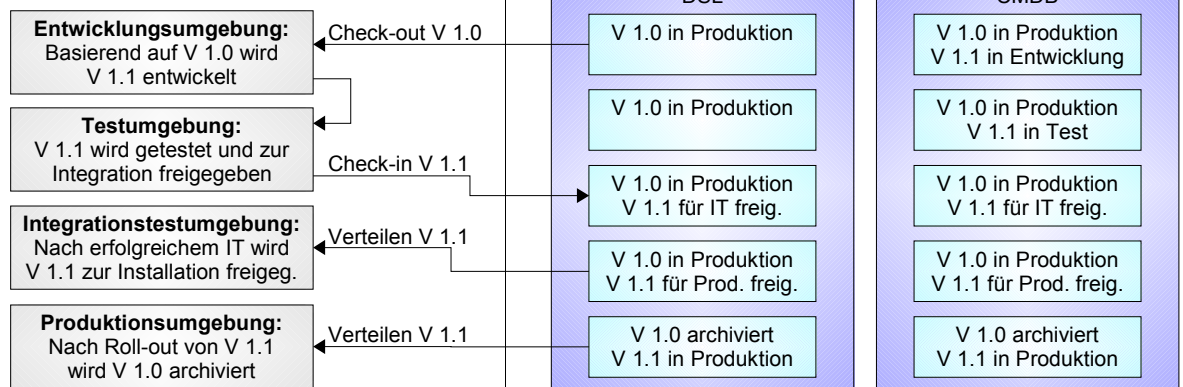
Grafik KM-Prozess



Beschreibung KM-Prozess

- **Identifikation:** eindeutig pro CI, zugehörige Attribute und Verknüpfungen
- **Kontrolle:**
 - autorisierte Verwaltung
 - Journalisierung / Historisierung auf Personenebene
 - Übereinstimmung der CIs mit Realität, Vollständigkeit
 - Versionenkontrolle
- **Statusüberwachung:** zur Verfolgung des Lebenszyklus
- **Plausibilisierung:**
 - Datenintegrität der CIs
 - korrekte Verknüpfungen

Grafik Entwicklungsprozess



Software Version Manager SVM

Programm zur Versionsverwaltung von Quellcode.

- PVCS Version Manager
- Concurrent Versions System CVS

Funktionsweise:

- Speicherung aller Fassungen mit Änderungsprotokollen an zentraler Stelle
- Bearbeitung von lokalen Kopien
- Übernahme sich nicht überschneidender Änderungen in das globale Verzeichnis
- keine logische Prüfung

Changemanagement-Prozess

KONF

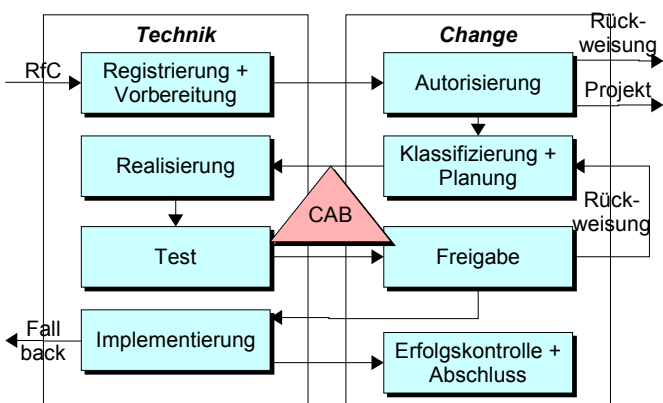
Change Manager

- untersucht die Auswirkungen der geplanten Änderungen
- sorgt für die Aktualisierung der CMDB
- überprüft die formale Korrektheit der Änderungsanträge
- kann Änderungsanträge zurückweisen
- verfolgt die Ausführung genehmigter Änderungsanträge

Request for Change RfC

- Antragsnummer: eindeutig zur Identifikation
- Antragsdatum
- Antragsteller, Produktverantwortlicher
- Änderungsbeschreibung
- zu ändernde Objekte und Abhängigkeiten
- betroffene Konfiguration: Release, Versionen der CI
- Änderungstyp: Fehlerbehebung, Änderung, Erweiterung
- Priorität: Dringlichkeit, Nutzen, Strategie-Wert
- Status
- Kategorie: Auswirkungen, Risiken
- Verweis auf Ticket, Problem, Projekt
- Begründung: Nutzen, Konsequenz bei Nichtdurchführung
- Aufwand, Kosten, Termine

Grafik ChM-Prozess



Beschreibung ChM-Prozess

- Registrierung: eindeutige Änderungsnummer
- Vorbereitung: Prüfung auf Vollständigkeit
- Autorisierung: Machbarkeitsprüfung (evtl. Rückweisung), Aufwandprüfung (evtl. Projektzuweisung)
- Klassifizierung: Kategorie und Priorität aufgrund Dringlichkeit, Auswirkungen, Abhängigkeiten (CMDB)
- Planung: Termine festlegen, Ressourcen bereitstellen, evtl. Koordination zu Releases
- Realisierung: Beschaffung, Parametrierung, Entwicklung
- Test: währenddem CI vor Veränderungen schützen
- Freigabe: Test ok und Richtlinien eingehalten
- Implementation: Zeitfenster mit kleinster Beeinträchtigung
- Erfolgskontrolle: Evaluierung des Erfolgs, Prüfbericht

Beurteilungskriterien Kategorie

- Komplexität: Anzahl betroffene CIs / User
- Erfahrung, Know-how
- Aufwand, Ressourcen
- Einfluss Serviceverfügbarkeit
- Testbarkeit
- Fallback-Möglichkeit

Prioritätsstufen

- 0 = dringend
Ausfall Serviceleistung / ernsthafte Benutzerprobleme
grosser Einfluss auf eine grosse Anzahl Benutzer
- 1 = hoch
Serviceleistung beeinträchtigt
grosser Einfluss auf eine kleine Anzahl Benutzer
geringer Einfluss auf eine grosse Anzahl Benutzer
- 2 = mittel
Serviceleistung gefährdet
geringer Einfluss auf Serviceleistung / Benutzer
Änderung kann nicht bis zum nächsten Release warten
- 3 = niedrig
geringer Einfluss auf Serviceleistung / Benutzer
Änderung kann bis zum nächsten Release warten

Entscheidungs-Raster

Prio.	Kategorie / Auswirkungen / Risiko		
	hoch	mittel	gering
0	Emergency CAB	Emergency CAB	Emergency CAB
1	ChM-Prozess IT-Management	ChM-Prozess CAB	ChM-Prozess Ch-Manager
2	ChM-Prozess IT-Management	ChM-Prozess CAB	ChM-Prozess Ch-Manager
3	Release	Release	Release

Erfolgskontrolle

Post Implementation Review PIR:

- Eintritt des gewünschten Effekts, Erreichung der Ziele
- Zufriedenheit der Benutzer mit dem Vorgehen und dem Ergebnis
- unerwünschte Folgeprobleme
- Übereinstimmung des Ressourcen- und Zeitaufwands mit der Planung
- Dokumentation der Lessons Learned

