endlich GmbH & Co. KG

WHITEPAPER

Configuration Management Database (CMDB)
- ein Praxisleitfaden zur erfolgreichen Einführung

Inhaltsverzeichnis

<u>EINLEITUNG</u>	2
DIE IDEALWELT UND IHRE STOLPERFALLEN	2
CONFIGURATION PLANNING (MANAGEMENT & PLANUNG)	3
CONFIGURATION IDENTIFICATION (IDENTIFIKATION DER KONFIGURATION)	4
CONFIGURATION CONTROL (STEUERUNG & KONTROLLE)	6
CONFIGURATION STATUS ACCOUNTING & REPORTING (STATUSNACHWEIS & REPORTING)	7
CONFIGURATION VERIFICATION & AUDIT (VERIFIZIERUNG UND AUDITIERUNG)	8
TOOLUNTERSTÜTZUNG / -ASPEKTE – WAS IHRE CMDB LEISTEN SOLLTE!	9
FLEXIBEL ERWEITERN UND MODIFIZIEREN	9
VISUALISIEREN SIE IHRE DATEN	9
DATEN EINFACH INTEGRIEREN, ABGLEICHEN UND SYNCHRONISIEREN	10
STANDARD-SCHNITTSTELLEN	10
FÖDERIERUNG	11
RECONCILIATION	11
DISCOVERY UND DEPENDENCY MAPPING	12
GENERISCHE SCHNITTSTELLE	13
DATEN AUSGEBEN / REPORTEN	14
WEITERE WICHTIGE KEY-FEATURES	14
<u>FAZIT</u>	<u>15</u>
AUTOREN	16

UNSERE EINSATZBEREICHE IM BEREICH ITSM UND CONFIGURATION MANAGEMENT 17

Einleitung

Das Wissen über vorhandene IT-Infrastrukturen und ihren Beziehungen zum Business wird immer wichtiger und ist mittlerweile die Grundvoraussetzung für viele unternehmensrelevante Entscheidungen und Investitionen.

Das Framework ITIL (IT Infrastructure Library) propagiert die Configuration Management Database (kurz: CMDB) bzw. das Configuration Management System (kurz: CMS) bereits seit vielen Jahren als das prozessunterstützende Querschnittstool. Und weitere Rahmenwerke wie COBIT, BSI Grundschutz-Katalog oder ISO-2000/27001 erfordern eine umfassende IT-Dokumentation mit einer CMDB als zentrales Fundament der ITSM-Prozesse.

In der Praxis erreichen aber viele Configuration Management Projekte ihr Ziel nicht. Oft sind die Unternehmen unzufrieden, weil die angestrebte Transparenz über die IT-Systeme nicht mit der erforderlichen Qualität realisiert werden konnte.

Dieses Whitepaper fasst alles Wissenswerte zum Thema CMDB zusammen. Es stellt einen Leitfaden für CMDB-Projekte dar, zeigt die Herausforderungen und Stolpersteine auf und gibt praxisorientierte Tipps zur erfolgreichen Einführung.

Die Idealwelt und Ihre Stolperfallen

Besonders ehrgeizig und meist nicht erfolgreich waren bis vor einigen Jahren jene ITSM-Projekte, die der Empfehlung von ITIL V2 gefolgt sind und eine zentrale CMDB mit einem einheitlichen Datenmodell für alle Konfigurationsdaten aus unterschiedlichen Quellen eingeführt haben. Viele dezentrale Excel-Listen oder Tools wurden in eine Datenbank migriert und somit ein überadministrierter Datenfriedhof geschaffen. Schnell stieg die Unzufriedenheit und die ITSM-Prozesse – allen voran Change und Incident Management – konnten keinen oder nur einen eingeschränkten Nutzen aus der CMDB ziehen.

Die ITIL-Autoren haben die Problematik erkannt und mit ITIL V3 den Begriff des Configuration Management Systems (CMS) eingeführt. Das CMS stellt ein logisches Modell aus mehreren CMDBs und Werkzeugen zur Datenintegration, -verarbeitung und -präsentation von Services und Infrastrukturinformationen dar. Aber auch mit dem neuen Konzept stellte sich nicht immer der Erfolg ein: die manuelle Datenpflege wird zwar vielfach unterstützt durch Automatisierung mit Discovery Tools (Scanner), aber nicht alle Netzwerkbereiche im Unternehmen können gescanned werden. Die Integration von System Management Tools oder anderer Inventory Systeme treibt den Aufwand in die Höhe und es entstehen neue Fehlerquellen und Dubletten.

Die Praxis zeigt zwar, dass Aufbau, Funktionen und Schwerpunkte des Configuration Managements an die Größe und Branche des Unternehmens angepasst werden sollte. Es gibt aber trotzdem Richtlinien, die grundsätzlich in allen CMDB-Projekten beachtet werden sollten.

"Pläne sind nichts. Planung ist alles", sagte schon Dwight D. Eisenhower. Planung ist im Configuration Management alles. Je detaillierter Sie sich in der "Configuration Planning"-Phase Gedanken machen, desto schneller werden Sie das anvisierte Ziel erreichen.

Das Ziel der Planungsphase ist die Definition der Strategie, Ziele, Scope, Prozesse, KPIs, Rollen und Schnittstellen zu den Stakeholdern des Configuration Managements (Change, Incident, Problem Management etc.). Die Planungsphase ist nicht nur zu Beginn des Projektes durchzuführen, sondern auch für spätere Weiterentwicklungen gemäß zukünftiger Anforderungen aus den ITSM-Prozessen.

- Unklare Inhalte, keine gemeinsame Vision: Häufig wird zuerst ein CMDB-Tool beschafft und alles andere erst im Laufe des Projektes definiert.
- Zu großer Umfang mit unrealistischen Terminen: Mit dem großen "Rundumschlag" soll alles auf einmal abgedeckt werden.
 Falsch gesetzte Prioritäten: Ohne ausreichende Zielorientierung

Stolperfallen



3

- können keine sinnvollen Prioritäten gewählt werden.
- Unklare Rollenverteilung und Prozessschnittstellen: Zuständigkeiten sind das A und O des Configuration Managements.

Viele Organisationen scheitern bereits an der Definition des Umfangs des Configuration Managements und den entsprechenden Prioritäten - nach dem Motto "wir integrieren alles". Wichtig ist aber ein iteratives Vorgehen nach dem Motto "Think big, but start small!". Ein kleines, schnelles und qualitätsorientiertes Projekt mit spezifischen, klar abgegrenzten und messbaren Zielen ist ein guter Ausgangspunkt für die nächsten Iterationen. Das schafft schnelle Erfolge und Sie sammeln wertvolle Erfahrungen:

- a. Integrieren Sie hierbei früh Ihre ITSM-Prozesspartner, finden Sie aktive Treiber des Projekts und definieren Sie ausführlich messbare Ziele, Scope (oder Nichtim-Scope), Prioritäten, eine gemeinsame Roadmap und eine langfristige Vision.
- b. Ziele, Zweck und Nutzen einer CMDB müssen hierbei stets außerhalb des Configuration Managements liegen bzw. aus anderen Disziplinen abgeleitet werden.
- c. Die CMDB stellt vornehmlich ein "Informationsbecken" für angrenzende ITSM-Prozesse dar. Somit hat eine CMDB alleine keinen Nutzenwert, vielmehr steht und fällt eine funktionierende CMDB mit dem In- und Output angrenzender ITSM-Disziplinen, wie z.B. dem Change Management.
- d. Legen Sie viel Wert auf die Vereinbarung von Prozessschnittstellen und saubere definierte (Pflege-)Verantwortlichkeiten.

Oftmals herrscht die Meinung, dass eine CMDB außerhalb des Datacenters keine Rolle spielt. Aber das ist grundlegend falsch. Planen und priorisieren Sie die Inhalte einer CMDB immer mit Bezug auf den Business Impact (welche Elemente sind Business-kritisch, wie sind die Zusammenhänge untereinander und wie beeinflussen diese ihr Business). Mit dem Top-Down-Ansatz (Service für Service, Applikation für Applikation) erhalten Sie einen sofortigen Mehrwert für Ihr Business und der Rahmen bleibt handhabbar. Elemente in Test- und Entwicklungslaboren, in denen Infrastrukturen schnell auf- und abgebaut werden, können so zunächst vernachlässigt werden.

Achten Sie außerdem auf eine eindeutige Dokumentation der Rollen- und Aufgabendefinitionen. Besonders wichtig ist die Besetzung der Rolle des CI-Owners je CI-Klasse. Verpflichten Sie ggf. auch externe Service Provider vertraglich zur Dokumentation der IT-Infrastrukturen in Ihrer CMDB. Nur so kann Ihre Strategie aufgehen.

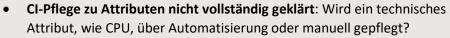
Configuration Identification (Identifikation der Konfiguration)

Das Ziel der Identifikationsphase ist die Definition und Pflege des zugrundeliegenden Konfigurationsmodells (ITIL: Configuration Modell), so dass die CMDB alle Informationen zu den CIs (die IT-Infrastrukturen und die Service-Informationen) aufnehmen kann. Hierzu gehört das Festlegen der CI-Klassen, deren Attribute und Beziehungen untereinander, sowie die Festlegung der Verantwortlichkeiten der Bearbeiter in den unterschiedlichen CI-Lebenszyklusphasen.

Diese Aktivitäten stellen sicher, dass nur autorisierte und identifizierte CIs über ihren gesamten Lebenszyklus erfasst werden. Neben ihrem Status werden Referenzkonfigurationen (Baselines) und Konfigurationskontrollen zur Überprüfung der Vollständigkeit und Aktualität festgelegt.

Stolperfallen

- Unterschiedliches Verständnis von Begrifflichkeiten: Sind Services, Applikationen und Software synonyme Begriffe? Auf gar keinen Fall!
- **Zu hohe Detaillierung**: Insbesondere bei den Attributen wird oft jede noch so unwichtige Nuance in der CMDB geführt.





Zunächst wird in einer Scopedefinition festgelegt, ob und welche CI-Klassen (z.B. Business Prozesse, Anwendungen, Middleware, Server, Netzwerk, SLA/OLA etc.) und CI-Subklassen (z.B. Anwendungsinstanzen, Softwareinstanzen, Oracle Datenbankinstanzen, Server Windows etc.) überhaupt in einer CMDB erfasst werden.

Beginnen Sie mit geschäftskritischen CI-Klassen und Subklassen und modellieren Sie die zulässigen Attribute und die Beziehungen zwischen den Klassen.

Denken Sie dran: je mehr Klassen, Attribute und Beziehungen in der CMDB dokumentiert werden sollen, desto höher ist der Arbeitsaufwand zur Aktualisierung der Daten. Richten Sie den Detailierungsgrad an konkreten Use Cases (Anwendungsfällen) und der Pflegbarkeit aus. Dokumentieren Sie immer Nutzer der Daten und identifizieren Sie Pflegeverantwortliche.

Unterstützung bieten hierbei Frameworks wie das Common Information Model (CIM). Es hilft Ihnen bei der leicht verständlichen Dokumentation des Konfigurationsmodells und der Konzeption von CMDB Schnittstellen.

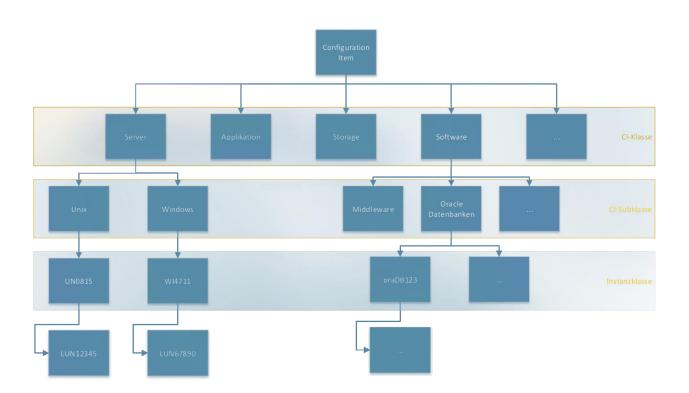


Abbildung 1: Vereinfachtes Beispiel eines Konfigurationsmodells

Achten Sie bei der Entwicklung des Konfigurationsmodells auf eine eindeutige Beschreibung der verwendeten Begrifflichkeiten. Viele Diskussionen können hierdurch vermieden werden.

Für jedes CI-Attribut muss entschieden werden, ob es manuell oder automatisiert gepflegt werden soll. Beides hat seine Vor- und Nachteile und nicht alles ist sinnvoll automatisierbar. Berücksichtigen Sie hierbei Umfang, Change-Frequenz und die relevanten Datenquellen.

Configuration Control (Steuerung & Kontrolle)

Das Ziel der Kontrollphase ist sicherzustellen, dass Veränderungen an der CMDB, also das Hinzufügen, Ändern oder Löschen von CI-Informationen, nur im Rahmen anderer ITSM-Prozesse wie z.B. dem Change Management durchgeführt werden und keine unautorisierten Änderungen geschehen. Damit ist Configuration Control eine Schlüsselaktivität des Configuration Managements.

Stolperfallen

- Pflege der CMDB-Daten erfolgt nur im Configuration Management durch ein dediziertes "Pflegeteam": Nein, besser nicht! Denken Sie prozessorientiert.
- Mangelhafte Datenqualität: Ist die Datenqualität zu gering, fehlt die Verlässlichkeit der CMDB und das Vertrauen in selbige geht verloren.



• Verifikations- und Kontrollmechanismen zu spät oder gar nicht eingeführt: Planen Sie Verifikations- und Kontrollmechanismen bereits vor Erstbefüllung der CMDB ein.

Die zuverlässige Pflege der CI-Daten ist das wichtigste Erfolgskriterium für die CMDB! Primär sind selbstverständlich die CI-Owner für die Pflege der CIs verantwortlich und in der Praxis werden sogenannte "Configuration Librarians" etabliert. Die Praxis zeigt jedoch, dass es mühsam ist, Änderungen an CIs von "Pflegeteams" einpflegen zu lassen. Denn oft sind diese in den Change Prozess gar nicht involviert und müssen den Informationen bzgl. durchgeführter Änderungen hinterherlaufen.

Change- und Configuration-Management sind deshalb eng miteinander zu verzahnen. Ein Configuration-Management ist auf aktuellen Informationen über Änderungen an der IT-Infrastruktur durch das Change-Management angewiesen. Umgekehrt kann das Change-Management ohne das Configuration Management die Auswirkungen eines Changes auf die IT-Infrastruktur, die IT-Services und damit auf das Business nur schwer beurteilen.

Deshalb:

- Legen Sie höchsten Wert auf die Prozess-Schnittstelle zwischen Configuration Management und Change Management
- Schließen Sie zusätzliche Rollenvereinbarungen ab, z.B. durch
 - Mitgliedschaft des Configuration Managers im Change Control Board
 - das Recht des Change Managers Changes zu stoppen, wenn die CMDB nicht aktuell ist.
- Erstellen Sie Dokumentationsrichtlinien, in denen die Voraussetzungen zur Dokumentation von CI-Klassen, zu dokumentierende Attribute, Nomenklatur, Regeln etc. festgelegt werden.

Im Configuration Control werden zudem Konfigurationsregeln (z.B. Architektur-Richtlinien für erlaubte oder nicht-erlaubte Komponenten) und Datenqualitätsregeln / -kriterien (z.B. fehlende Verlinkung von Applikation zu Datenbank) mit den entsprechenden Fachbereichen erarbeitet.

Eine wichtige Aufgabe innerhalb der Kontrollphase ist die Prüfung der Korrektheit des Datenbestandes anhand von Soll-/Ist-Vergleichen. Dabei wird der durch die Pflegeprozesse definierte Sollbestand mit dem über Discovery Tools gewonnenen Ist-Bestand verglichen. So können unautorisierte Changes erkannt und die Datenqualität sichergestellt werden.

Configuration Status Accounting & Reporting (Statusnachweis & Reporting)

Ziel der Reporting-Phase ist der Nachweis und die kontinuierliche Überwachung des Status aller erfassten CIs über den kompletten CI-Lebenszyklus, also von der Planung bis zur Ausmusterung. Dies ermöglicht auch die Nachverfolgung von Changes, zugehörigen Incidents, Service Requests und Problemen an identifizierten IT-Komponenten.

Basis für die lückenlose Änderungsverfolgung ist die Pflege des Lebenszyklus-Status eines CIs.

Zur Sicherstellung der Nachweisbarkeit müssen die Statusinformationen historisiert werden.

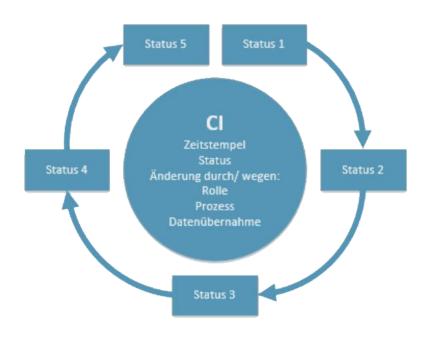


Abbildung 2: Historisierter Nachweis des CI-Status

Für die CI-Lebenszyklusprozesse muss dabei folgendes definiert werden:

- Alle zulässigen Status (Zustände)
- Welche Status für die unterschiedlichen Umgebungen (Test, Integration, Produktion, ...) zulässig sind
- Durch wen oder wie die Vergabe des Status erfolgen darf (eine bestimmte Rolle, ein bestimmter Prozess oder z.B. bei der automatischen Datenübernahme)

Die CMDB dient zudem als Basis für die Erstellung von Referenzkonfigurationen (Configuration Baselines). Auf ihrer Basis kann nach fehlgeschlagenen Changes die frühere Konfiguration einer IT-Landschaft wiederhergestellt werden. Dazu erstellt das Configuration Management zu einem bestimmten Zeitpunkt einen Snapshot der CMDB und legt diesen als Referenzkonfiguration fest.

Ein wichtiges Ergebnis dieser Reporting-Phase sind regelmäßige Berichte mit Informationen über den aktuellen Status, den geplanten Status und die Änderungshistorie einzelner CIs oder auch kompletter Services. Je nach Zielgruppe oder Anforderungen sind unterschiedliche Berichte und Dashboards zu erstellen. Diese Komplexität führt in der Praxis häufig zu teuren CMDB-Customizing-Projekten oder dem Einsatz eines zusätzlichen Reporting-Tools.

Hinweis

Unterschätzen Sie das Thema Berichtswesen nicht!

Im Laufe des Projekts werden, vor allem aus angrenzenden ITIL Disziplinen, viele Anforderungen an das Configuration Management gestellt werden. Planen Sie u.U. ein CMDB (inkl. zuliefernder Schnittstellen) übergreifendes Reporting ein.



Configuration Verification & Audit (Verifizierung und Auditierung)

Ziel der Verifizierungsphase ist die Etablierung und Durchführung von regelmäßigen Kontrollmaßnahmen und Audits. Mit diesen wird sichergestellt, dass die Verfahren des Configuration Managements eingehalten werden und die in der CMDB enthaltenen Daten aktuell, vollständig und korrekt sind.

Folgende beispielhafte Vorgaben können für die Verifizierung herangezogen werden:

- Konfigurationsregeln aus dem IT Betrieb zur Identifikation von Konfigurationsmängeln
- Governance: Prüfung der dokumentierten Verantwortlichkeiten (z.B. ist der CI-Owner noch im Unternehmen?)

Eine unterstützende Methode der Verifizierung sind die Visualisierungen von Servicebäumen, die beispielsweise nach erfolgten Changes herangezogen werden können. Configuration-Audits erfolgen zu festgelegten Zeitpunkten (z.B. vor und nach großen Änderungen) oder in regelmäßigen Zeitabständen (z.B. alle sechs Monate). Im Rahmen der Audits wird der CMDB Soll-Datenbestand mit den realen, physischen Gegebenheiten abgeglichen, die Konformität zwischen den dokumentierten Baselines und der tatsächlichen Umgebung geprüft, sowie abgefragt ob die Release und Configuration Dokumentation vor einem Release vorhanden ist.

Stolperfallen

- Unregelmäßige Durchführung von Audits: Häufig wird der Aufwand für Audits gescheut und daher zu selten durchgeführt.
- Auditergebnis "Alte Labels an physischen Geräten":
 Nach der Umbenennung eines Gerätes wurde das Label nicht aktualisiert. Die Seriennummer verbleibt dabei häufig als einziges Kriterium zur eindeutigen Identifizierung.



Toolunterstützung / -aspekte - was Ihre CMDB leisten sollte!

Noch wesentlich mehr als der Prozess ist die Auswahl des passenden Tools höchstindividuell und hängt vom erwarteten Nutzen und den gesetzten Zielen ab. Idealerweise bieten CMDB Tools folgende Standard-Unterstützung:

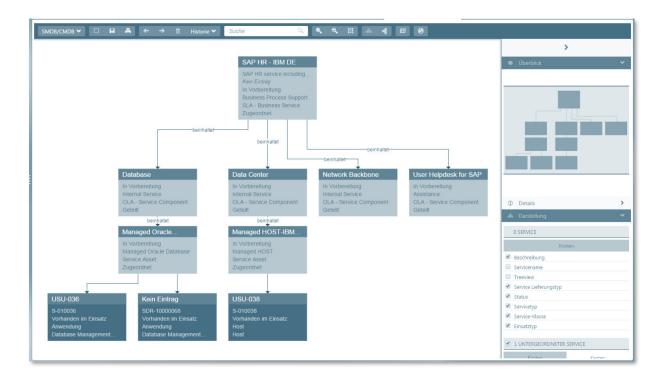
Flexibel erweitern und modifizieren

In der Praxis muss eine CMDB stetig erweitert und angepasst werden. Ihr CMDB Tool sollte deshalb eine größtmögliche Flexibilität in Hinblick auf Anpassungen und Erweiterungen bieten. Hierzu gehören u.a. ein flexibles Datenmodell und die Möglichkeit der einfachen Modifikation von CI-Klassen und -Subklassen, den einzelnen CIs und deren Attributen. So vermeiden Sie teure Customizing-Projekte.

Visualisieren Sie Ihre Daten

Ihre CMDB sollte eine visuelle Darstellung der CIs und deren Abhängigkeiten bieten. Dadurch werden IT-Infrastrukturen mit Ihren IT- und Business Services leichter verständlich und unterstützen bei Impact- und Root-Cause-Analysen.

Zusätzlich ist eine Visualisierung von physischen Geräten (z.B. Rackansichten) ideal für die effiziente Verwaltung und Überwachung von Racks, Server und Netzwerkgeräten in Rechenzentren.



Daten einfach integrieren, abgleichen und synchronisieren

Wie bereits erwähnt ist es nicht zielführend und praktisch unmöglich, sämtliche Konfigurationsdaten aus diversen Werkzeugen in einer zentralen CMDB mit einem einheitlichen Datenmodell zu verwalten. Viel effizienter ist es, Daten dort zu belassen wo sie entstehen und lediglich die wichtigsten Daten über Schnittstellen einzusammeln. Die große Herausforderung beim Einsatz mehrerer Werkzeuge jedoch ist, die jeweiligen Datenbestände synchron zu halten und die CI-Informationen in den verschiedenen Tools zueinander in Beziehung zu setzen. Ihr CMDB Tool sollte deshalb die folgenden Funktionen bieten:

Standard-Schnittstellen

Connectoren zu verbreiteten Tools wie z.B. MS SCCM, vCenter, SAP Solution Manager, Active Directory sollten out-of-the-box vorhanden sein. Massendaten aus individuellen Drittsystemen sollten per Excel-Schnittstelle geladen werden können.

Stolperfallen

Einbindung von Virtuellen und Cloud-Ressourcen:

Viele Unternehmen profitieren mittlerweile von der Flexibilität der Cloud oder den virtuellen Welten. Vergessen Sie dabei nicht, dass auch die dort genutzten Ressourcen in Ihrer CMDB dokumentiert werden müssen. Cloud oder Systems Management Tools bieten sich dabei als Datenlieferant an.



Föderierung

Föderierung erlaubt es, Daten dort zu belassen, wo sie entstehen. Bei diesem Modell werden in einer zentralen CMDB lediglich begrenzte CI-Informationen und Ihre Beziehungen zueinander dokumentiert. Externe Systeme können über Schnittstellen oder Datenbanklinks verbunden sein und weitere CI-Informationen auf Abruf zur Verfügung stellen. Dies können z.B. Inventory Daten, Finanzdaten oder technische Detailinformationen sein.

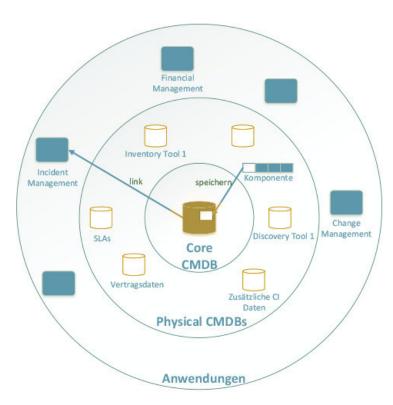


Abbildung 3: Föderation

Damit kann jedes System für sich eigenständig bleiben und ein eigenes unabhängiges Datenmodell besitzen. Insbesondere für große und komplexe Umgebungen ist eine föderierte CMDB empfehlenswert.

Reconciliation

Reconciliation ist immer dann erforderlich, wenn CIs in mehreren Datenquellen mit unterschiedlichen Informationen vorhanden sind. Mittels Reconciliation werden unterschiedliche Datenbestände auf Basis von Validierungsregeln analysiert, synchronisiert und zusammengefasst. Dieser Prozess analysiert bzw. "sichtet" die CI-Daten vor ihrer Aufnahme in die CMDB und verhindert Dubletten. Allerdings darf der Aufwand zur Definition der notwendigen Reconciliation-Regeln nicht vernachlässigt werden.

Idealerweise bietet Ihr CMDB Tool einen Abgleich beliebiger Datenbestände (Ist-, Soll-, Plan- und Historiendaten) an und lässt eine freie Wahl der abzugleichenden Attribute und Beziehungen zu.

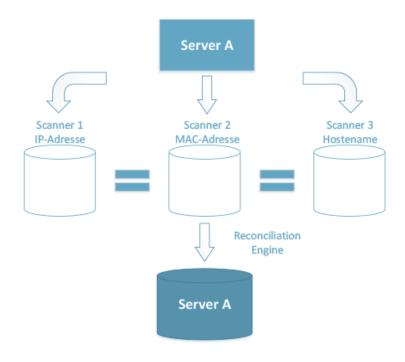


Abbildung 4: Reconciliation

Discovery und Dependency Mapping

Eine große Herausforderung für jedes CMDB Projekt ist die initiale Befüllung der Datenbank, die Ermittlung des IST-Datenbestandes und die Ermöglichung eines Soll-/Ist-Abgleiches. Unterstützung bieten hierbei Discovery Tools (die sogenannten Scanner), die im Idealfall automatisiert Hard- und Softwarekomponenten erkennen und logische Beziehungen und

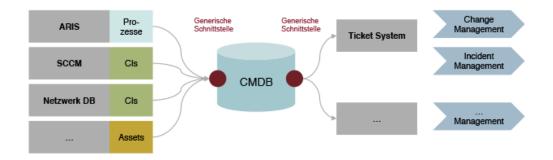
Abhängigkeiten der Infrastrukturen liefern (Dependency Mapping). Für die Erkennung häufig eingesetzter Komponenten bekannter Hersteller wie Oracle, SAP, Microsoft, Adobe etc. bringen diese Tools bereits fertig konfigurierte Regeln mit (sog. "Signaturen" oder "Fingerprints").

Ziel Ihres CMDB Projektes sollte sein, einen möglichst hohen Automatisierungsgrad bei der Datenintegration zu erreichen, um den manuellen Pflegeaufwand zu minimieren und die Datenqualität zu erhöhen. Doch auch Discovery und Dependency Mapping Tools haben ihre Tücken:

- Es gibt agentenlose und agentenbasierende Tools. Finden Sie die optimale Lösung für Ihr Unternehmen. Agentenbasierte Lösungen erfordern immer eine Installation eines Agenten auf jedem zu scannenden CI. Kann Ihre Organisation das leisten?
- Software mit geringerem Verbreitungsgrad oder Eigenentwicklungen werden von den Discovery Tools meist nicht erkannt, die Erkennungsregeln müssen selbst konfiguriert werden.
- Fokussieren Sie nicht nur auf Discovery und Dependency Mapping Lösungen, sondern denken Sie prozessorientiert. Diese Tools lösen keine Prozessprobleme!
- Unterschätzen Sie nicht die Komplexität gescannter Daten. Daten um jeden Preis sollte nicht das Ziel sein.
- Es gibt in Unternehmen immer wieder Netzwerkbereiche, die nicht scanbar sind.
- Bedenken Sie beim Discovery auch Datenschutzrichtlinien und entsprechende Gesetze. Personenbezogene Daten sollten nicht oder nur in Absprache mit dem Datenschutzverantwortlichen eingesammelt werden.
- Sollen Discovery-Daten in die CMDB übernommen werden, sind entsprechende Prüfungen zu implementieren (siehe auch "Reconciliation" weiter oben).

Generische Schnittstelle

In der Praxis sind nicht für alle Datenlieferanten Standard-Connectoren vorhanden. Aufgrund der Fülle möglicher Anbindungen ist es vorteilhaft, eine generische Schnittstelle für ein- und ausgehende Daten zu entwerfen. Diese soll ermöglichen, dass bei CMDB Erweiterungen oder Modelländerungen, welche meist zu Clusteränderungen führen würden, die Schnittstellenpartner i.d.R. ohne Customizing auskommen. Sobald eine generische Schnittstelle definiert und implementiert ist, sollte diese flexibel neue CI-Klassen oder Attribute aufnehmen können. Die nachfolgende Abbildung zeigt das Konzept in einem beispielhaften Kontext:



Wie so oft in der IT gilt auch für CMDB Schnittstellen: Achten Sie auf eine angemessene Dokumentation!

Daten ausgeben / reporten

Ihr CMDB Tool sollte ein umfängliches Reporting und umfangreiche Abfragefunktionen zur Verfügung stellen. Hierunter fallen beispielsweise KPI-Reporting, Adhoc-Reporting, idealerweise Self-Service-Reporting und Dashboards. Über ein integriertes Datenqualitätsreporting lässt sich ein "automatischer" Audit durchführen.

Weitere wichtige Key-Features

Die Liste der Toolaspekte könnte hier noch sehr weit fortgesetzt werden. Natürlich sollte die CMDB ein vollumfängliches Berechtigungskonzept und ggf. Mandatenfähigkeit (insbesondere für größere Unternehmen) mitbringen. Auch ein durchdachtes Konzept zur Historisierung und des Baselinings sollte nicht fehlen. Entscheidend kann ebenso die Updatefähigkeit eines Tools sein. Ist bei einem größeren (Major-) Versionssprung eine Neukonzeption erforderlich oder bekomme ich mit dem Update zwar neue Features aber keine neuen Probleme?

Wichtig bei der Toolentscheidung bleiben immer Ihre individuellen Ziele und der erwartete Nutzen. Entsprechend müssen die Anforderungen an eine CMDB gewichtet werden.

Fazit

Wie also wird Ihr CMDB-Projekt zum Erfolg? Vermeiden Sie Stolperfallen und fangen Sie klein an – nach dem Motto "Think big, but start small". Setzen Sie klare, abgegrenzte Ziele, wählen Sie den Top-Down-Ansatz und achten Sie auf Qualität. So werden Sie schnell erste Erfolge erreichen und sind für weitere Iterationen vorbereitet.

Des Weiteren sollten Sie Ihr CMDB-Projekt niemals als "Stand alone"-Projekt führen.

Das Configuration Management und die CMDB sind auf die Unterstützung der umliegenden Prozesse, insbesondere bei der CI Pflege, angewiesen.

Es gibt keine out-of-the-box CMDB für alle Kunden, aber erfinden Sie das Rad nicht neu! Vorhandene Inventory- und Software-Systeme können sehr gut als CMDB-Datenlieferanten dienen. So benötigen Sie weniger zusätzliche Discovery Tools und können schneller Ergebnisse erzielen.

Und last, but not least: Tun Sie Gutes und reden Sie darüber! Schaffen Sie frühzeitig Verbündete, überzeugen und integrieren Sie Zweifler und präsentieren Sie auch kleine Erfolge. Eine CMDB ist ein zentrales Thema und ist mitentscheidend für den Erfolg Ihrer Serviceorganisation.

Autoren



Stephan Witt, e:ndlich GmbH & Co. KG

Stephan Witt ist Gründer und Geschäftsführer der e:ndlich GmbH & Co. KG und Principal Consultant für ITSM Beratung. Mit über 25 Jahren Erfahrung in der IT-Branche und 20 Jahren Selbständigkeit gehört der gelernte Dipl. Wirtschaftsinformatiker und ehem. Oracle Mitarbeiter mittlerweile "zum alten Eisen" und verfügt über umfassendes Know-How im Projektmanagement und der IT Prozesswelt. Mit seiner Fähigkeit komplizierte Sachverhalte in verständliche Abbildungen zu fassen und zu erklären, begeistert er Kunden und Kollegen gleichermaßen.



Nikolai Weidmann, e:ndlich GmbH & Co. KG

Nikolai Weidmann ist Consultant für ITSM Beratung bei der e:ndlich GmbH & Co. KG. Parallel zu seiner Arbeit hat er ein berufsbegleitendes Studium zum Bachelor of Science Wirtschaftsinformatik erfolgreich abgeschlossen. Schwerpunktmäßig ist Herr Weidmann in der Beratung, Konzeption, Entwicklung und dem Rollout von IT Service Management Tools sowie deren Toolschnittstellen tätig. Sein tiefes Prozess Know-How vornehmlich im Incident, Problem, Change und Configuration Management runden seine Expertise ab. Darüber hinaus beschäftigt er sich mit komplexen Themen, wie dem Business Intelligence, und ist Trainer für Inhouse- oder Offsite-Seminare.

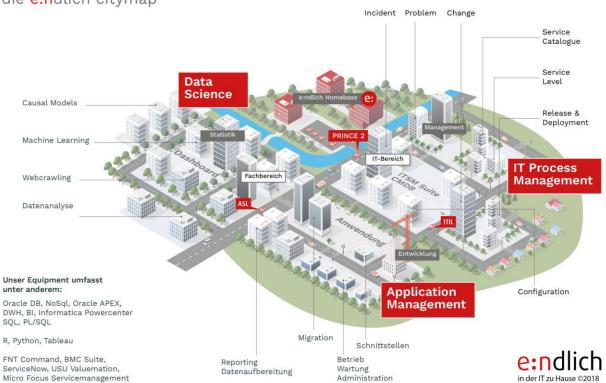


Martin Landis, USU AG

Martin Landis ist bei der USU AG als Experte für das IT- und Enterprise Service Management tätig.

Unsere Einsatzbereiche im Bereich ITSM und Configuration Management

In der IT zu Hause die e:ndlich citymap



Wir gewährleisten die Transparenz Ihrer technischen Infrastrukturen und sichern Ihr operatives Geschäft:

Mit unserer Expertise aus unterschiedlichsten ITSM-Projekten bieten wir Ihnen hochwertige, fachliche Leistungen in der Strategie, Konzeption, Entwicklung, Einführung und dem Rollout von ITSM Prozessen und Systemen.

Im Configuration Management Umfeld verfügen wir über 10 Jahre Erfahrung in den Branchen Automobil, IT-Dienstleister, Banken und Versicherungen und kennen intensiv die Sorgen unserer langjährigen Kunden und beraten im Prozess- und Systemumfeld. Technisch unterstützen wir Sie im Bereich der CMDB/CMS Weiterentwicklung, den Schnittstellen, der Staging Area und dem Reporting.

e:ndlich GmbH & Co. KG

Flößaustraße 24 | D - 90763 Fürth

Vertreten durch: Inhaber der e:ndlich GmbH & Co. KG: Stephan Witt und Alexander Weber

Kontakt:

eMail: info@endlich.it
Web: www.endlich.it
Tel. +49(0)911-4087-0555
Fax. +49(0)911-4087-884