

Data**N**avigator



www.datanavigator.de | ISSN 2702-1904

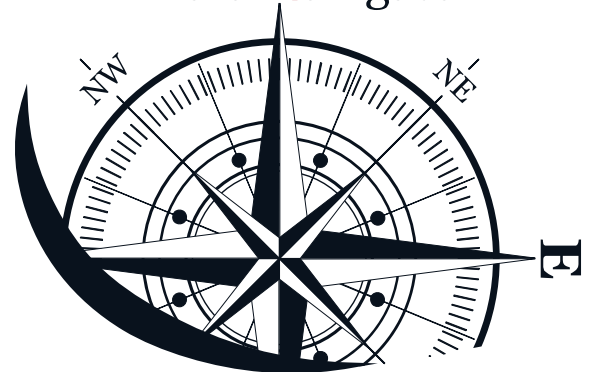
Das Millionengrab: Warum CMS-Projekte an veralteten Datenkonzepten scheitern

Patrik Schädler & Zohaib Tariq

#33 | Januar 2026

EUROGROUP
CONSULTING

Data Navigator



Autos, Maschinen, IT-Equipment – immer mehr Vermögenswerte werden heute nicht mehr gekauft, sondern geleast. Was auf den ersten Blick einfach erscheint, entpuppt sich bei genauerem Hinsehen aber als ein hochkomplexes Vertragskonstrukt. Ein einziger Leasingvertrag kann Dutzende von Zahlungsströmen auslösen: monatliche Raten, Sonderzahlungen, Versicherungsprämien, Restwertgarantien, Kilometerabrechnungen. Dazu kommen unterschiedliche Vertragstypen, individuelle Konditionen, gesetzliche Vorgaben und regulatorische Anforderungen. Diese Komplexität muss abgebildet, verwaltet und überwacht werden – und genau dafür gibt es Contract-Management-Systeme (CMS).

Die zentrale Herausforderung ist es sicherzustellen, dass ein CMS ‚gut‘ eingeführt wird. Funktioniert es nicht nur, sondern arbeitet es auch harmonisch mit den bestehenden Prozessen und Systemen im Unternehmen zusammen? Die Antwort ist überraschend unbequem: Ein CMS ist nur so gut wie die Daten, die es verarbeitet. Ein hochmodernes System mit schlechten Daten ist wie ein Ferrari mit Sand im Tank – theoretisch leistungsfähig, praktisch wertlos.

Die unbequeme Wahrheit ist allerdings, dass Datenqualität kein Feature ist, das man sich hinzukaufen kann. Datenqualität ist zudem kein Thema, das man erst nach der Systemauswahl anspricht, sondern toolunabhängig immer wichtig – unabhängig davon, ob man SAP, Oracle oder eine spezialisierte Leasing-Software einsetzt. Sie ist das Fundament, auf dem alles andere aufbaut. Und wenn dieses Fundament bröckelt, nützt auch die beste Technologie nichts.

Sieben Thesen zur Datenqualität in CMS-Projekten

Dieser Beitrag zeigt anhand von sieben Thesen, wie die Übertragung von alten Datenproblemen in neue Systeme zu einem Scheitern von CMS-Projekten führt, was echte Datenqualität ausmacht und warum dieses Ziel oft nicht erreicht wird. Der Beitrag schildert wirksame Lösungsansätze, um eine tragfähige Lösung zu implementieren, und zeigt auf, wie ein erfolgversprechender Umsetzungsplan aussehen sollte.

Die Illusion des sauberen Schnitts

These 1: Ein neues CMS mit alten Daten ist wie ein Ferrari mit Sand im Tank – theoretisch leistungsfähig, praktisch wertlos.

Leasinghäuser investieren Millionen in moderne Contract-Management-Systeme. In den Projektmeetings diskutieren Teams über Features, Performance-Kennzahlen und Benutzeroberflächen. Anbieter präsentieren beeindruckende Demos mit perfekten Beispieldaten. Die entscheidende Frage wird jedoch in den wenigsten Fällen zum richtigen Zeitpunkt gestellt: Was passiert mit unseren Daten?

Die Realität sieht in vielen Leasinghäusern ernüchternd aus: Derselbe Vertrag existiert in System A mit dem Status ‚aktiv‘, während System B ihn als ‚gekündigt‘ führt. Kundenadressen tauchen dreifach auf – in vier verschiedenen Schreibweisen. Asset-Informationen sind über fünf verschiedene Datenbanken verteilt und niemand kann mit Sicherheit sagen, welche Version aktuell ist. Änderungen werden nicht propagiert. Eine echte Historisierung der Datenänderungen? Fehlanzeige. Validierung? Passiert irgendwo, irgendwie, aber nicht konsistent und schon gar nicht zentral gesteuert.

Und dann kommt die CMS-Einführung. Nicht nur mit denselben Datenproblemen – sondern oft mit der fatalen Zielsetzung, die alten, qualitativ schlechten Daten einfach zu übertragen. Data-Cleansing-Aktivitäten sollen die Migration erleichtern. Aber ist das wirklich der richtige Ansatz? In den meisten Fällen wird nicht an die Ursachen herangegangen, sondern nur das Problem von außen behoben. Das neue System macht dann alles schneller – inklusive der Fehlerfortpflanzung.

Die Konsequenzen sind brutal: Manuelle Nacharbeit frisst Ressourcen – Teams verbringen bis zu 40% ihrer Arbeitszeit mit Datenabgleichen statt mit wertschöpfenden Aufgaben. Fehlerhafte Prozesse entstehen zwangsläufig, wenn Abrechnungen auf veralteten Verträgen beruhen oder Bonitätsprüfungen mit falschen Kundendaten durchgeführt werden. Kunden werden unzufrieden, wenn sie widersprüchliche Informationen erhalten oder Anfragen nicht korrekt bearbeitet werden. Die Digitalisierung blockiert, weil neue Services an inkonsistenten Datengrundlagen scheitern. Und nicht zuletzt entstehen erhebliche Compliance-Risiken, wenn Audit-Trails fehlen und regulatorische Anforderungen nicht erfüllt werden können.

Ein CMS kann diese Probleme nicht lösen, es kann sie nur schneller sichtbar machen. Die Hoffnung, dass ein neues System automatisch für bessere Datenqualität sorgt, ist trügerisch.

Datenqualität als Unternehmensstrategie

These 2: Datenqualität ist keine IT-Aufgabe – sie ist eine Unternehmensstrategie, die alle Bereiche durchdringen muss.

Eine Diskussion, die in Beratungsprojekten immer wieder aufkommt: Wer ist eigentlich für die Daten verantwortlich? Die IT-Abteilung verweist auf die Fachbereiche; die Fachbereiche verweisen auf die IT. Die Wahrheit liegt, wie so oft, in der Mitte. Die IT ist Owner der Daten und legt technische Rahmenbedingungen fest, beispielsweise welche Formate erlaubt sind, wie Daten gespeichert werden und welche Schnittstellen existieren. Aber die Verantwortung für

Richtigkeit und Inhalt der Daten müssen zwingend im Fachbereich liegen. Die Datenqualität lässt sich nur eingeschränkt durch technische Regeln und Limitierungen steuern.

Eine Analogie macht das deutlich: Die IT ist wie der Automobilhersteller, der festlegt, dass ein Fahrzeug mit Benzin statt mit Diesel fährt. Sie kann den Tankaufsatz so modifizieren, dass nur Tanksäulen für Benzin passen. Sie schafft also die technischen Voraussetzungen. Aber der Fachbereich ist der Fahrer, der dafür sorgen muss, dass tatsächlich der richtige Kraftstoff getankt wird. Die besten technischen Schutzmaßnahmen nützen nichts, wenn der Fahrer bewusst oder unbewusst am System vorbeiarbeitet.

Datenqualität bedeutet also nicht nur, dass IT-Abteilungen Validierungsregeln implementieren. Datenqualität bedeutet, dass das gesamte Unternehmen versteht: Daten sind das wertvollste Asset, das wie jedes Asset gepflegt, geschützt und strategisch genutzt werden muss.

In einem erfolgreichen Ansatz haben Daten einen klaren Lebenszyklus: Sie werden an genau einer Stelle erfasst („Single Source of Truth“), durch definierte Validierungen geprüft, versioniert und historisiert, über standardisierte Schnittstellen verteilt und hinsichtlich ihrer Nutzung überwacht. Klingt aufwendig? Ist es auch. Aber der Aufwand zahlt sich jeden Tag aus, weil er Fehlerkosten reduziert, Prozesse beschleunigt und die Basis für Innovation schafft.

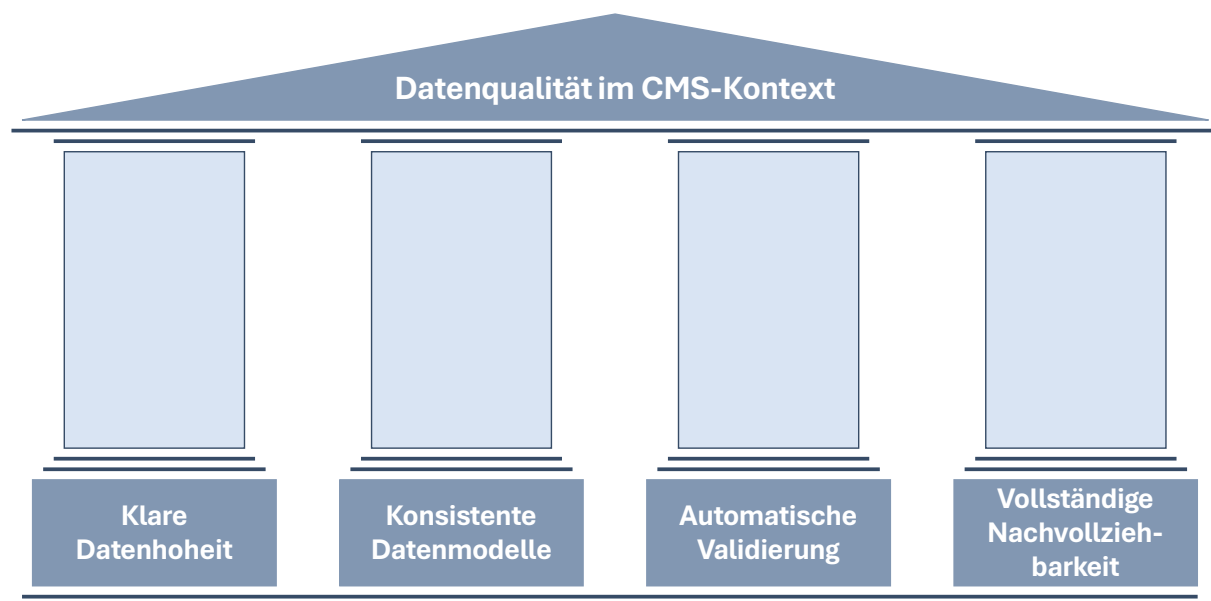


Abbildung 1 – Datenqualität im CMS-Kontext

Echte Datenqualität beruht, wie Abbildung 1 zeigt, auf vier Säulen:

- **Klare Datenhoheit:** Für jedes Datenobjekt gibt es genau einen Owner. Dieser Owner ist verantwortlich für die Definition des Datenmodells, die Datenpflege, die Qualitätssicherung sowie die Bereitstellung für andere Systeme. Ein Vertrag gehört dem CMS. Ein Kunde gehört dem Customer-Relationship-Management-System (CRM). Ein Asset gehört dem Asset-Management-System. Keine Ausnahmen, keine Graubereiche. Beispiel aus der Praxis: Wenn sich eine Kundenadresse ändert, geschieht dies ausschließlich im CRM. Alle anderen Systeme konsumieren diese Änderung als Event.

Sie können die Information lokal zu Performancezwecken speichern, aber die Wahrheit liegt im CRM. Bei Konflikten gewinnt immer das führende System.

- **Konsistente Datenmodelle:** Ein Vertrag ist ein Vertrag – überall. Nicht ‚Kontrakt‘ in System A, ‚Agreement‘ in System B und ‚Leasing-Objekt‘ in System C. Unternehmen brauchen unternehmensweite Definitionen für alle Kernobjekte. Welche Attribute sind Pflicht? Welche Werte sind erlaubt? Wie werden Beziehungen zwischen Objekten modelliert? Was bedeuten Statusübergänge? Wie wird die Historie der Änderungen erfasst?

Ein wichtiger Hinweis für den CMS-Bereich: Es gibt zwei zentrale Phasen – die Angebotsphase bis zur Vertragsunterzeichnung und die Vertragsphase ab der Vertragsunterzeichnung. Diese Unterscheidung unterstreicht die Bedeutung konsistenter Datenmodelle über den gesamten Lebenszyklus hinweg. Die harte Realität: Diese Arbeit ist nicht sexy. Sie erfordert Workshops, in denen Fachabteilungen ihre unterschiedlichen Definitionen harmonisieren müssen. Sie erzeugt Konflikte, weil etablierte Prozesse angepasst werden müssen. Sie kostet Zeit und Nerven. Aber ohne sie wird jedes CMS-Projekt zur Dauerbelastung.

- **Automatische Validierung:** Fehlerhafte Daten müssen an der Quelle gestoppt werden, nicht beim Konsumenten. Das bedeutet: Validierungsregeln werden im führenden System implementiert, Eingaben werden sofort geprüft, Fehler werden sofort gemeldet – bevor sie sich im Ökosystem verbreiten. APIs lehnen ungültige Daten ab. Events werden erst erzeugt, wenn die Daten valide sind.

Konkret bedeutet dies, dass ein Vertragsstatus nicht direkt von ‚aktiv‘ auf ‚gelöscht‘ wechseln kann, wenn der Workflow ‚gekündigt‘ als Zwischenschritt vorsieht. Ein Asset kann keinem Vertrag zugeordnet werden, wenn es bereits einem anderen zugeordnet ist. Eine Rechnung kann nicht erstellt werden, wenn der Zahlungsplan fehlt. Diese Regeln müssen technisch durchgesetzt werden – nicht durch Prozessbeschreibungen, sondern durch Code, der sie umsetzt.

- **Vollständige Nachvollziehbarkeit:** Jede Änderung an jedem relevanten Datenobjekt muss nachvollziehbar sein: Wer hat was wann geändert? Welcher Wert galt vorher? Welche Systeme haben die Änderung konsumiert? Im Leasing ist das nicht nur Best Practice, sondern auch eine regulatorische Pflicht. Aber selbst ohne Compliance-Anforderungen: Ohne Nachvollziehbarkeit können Fehler nicht analysiert, Datenmigrationen nicht geprüft und Prozesse nicht optimiert werden.

Die Lösung ist es, durch Event-Sourcing Historisierung zum Designprinzip zu machen. Jede Änderung ist ein unveränderliches Event. Die gesamte Historie ist automatisch verfügbar. Audit-Trails entstehen von selbst. Und wenn etwas schiefgeht, können Events neu abgespielt werden, um den Fehler zu analysieren und zu beheben.

Das architektonische Fundament

These 3: Das CMS ist austauschbar – das Datenmodell ist es nicht. Die Architektur folgt den Daten, nicht umgekehrt.

CMS-Projekte starten typischerweise mit der Frage: ‚Welches System kaufen wir?‘ Das ist die falsche Reihenfolge. Die richtige Reihenfolge lautet: Erst das Datenmodell, dann die Architektur, dann das Produkt. Ein sauberes Datenmodell definiert, wie Systeme zusammenarbeiten müssen. Und die Architektur definiert, welches CMS überhaupt in Frage kommt.

In vielen Leasinghäusern ist das CMS historisch zum Monolithen geworden: Es verwaltet Verträge, speichert Kundendaten, berechnet Risiken, steuert Abrechnungen und liefert Berichte. Mit dieser Last kann kein System langfristig erfolgreich sein. Die Lösung liegt in klaren Domänengrenzen nach dem Prinzip des Domain-Driven-Designs.

Das Vertragsmanagement ist die Kernaufgabe des CMS – mehr nicht. Kundenverwaltung gehört ins CRM-System. Die Risikobewertung wird vom Risikomanagement übernommen. Abrechnungen erfolgen im Finanzsystem. Jede Domäne hat ihren eigenen Zuständigkeitsbereich und ihre eigene Datenhoheit. Das CMS orchestriert diese Domänen, indem es Events publiziert und konsumiert, aber es versucht nicht, alles selbst zu übernehmen.

Ein sauberes Datenmodell definiert diese Grenzen präzise. Es legt fest, welches System welche Daten führt und wie diese Daten zwischen Systemen fließen. Die Architektur setzt dieses Modell dann technisch um – idealerweise mit einer Event-Driven-Architecture, die Systeme lose koppelt und unabhängig skalierbar macht.

Der Batch-Albtraum

These 4: Batch-Prozesse sind Datenqualitätskiller – sie erzeugen Inkonsistenzfenster, die niemand kontrollieren kann.

In traditionellen IT-Landschaften werden Daten nachts in Batch-Prozessen zwischen Systemen synchronisiert. Das klingt erst einmal praktisch: Tagsüber arbeiten die Anwender ungestört, nachts laufen die Datentransfers im Hintergrund. Das Problem: Zwischen 8:00 Uhr morgens und 2:00 Uhr nachts hat System A eine andere Sicht auf die Welt als System B. Kunden erhalten widersprüchliche Auskünfte. Prozesse arbeiten mit veralteten Daten. Fehler werden kompensiert statt verhindert. Und bei jedem Batch-Lauf kann etwas schiefgehen – was dann manuell korrigiert werden muss.

Diese nächtlichen Batch-Läufe ergeben Inkonsistenzfenster von bis zu 16 Stunden. In dieser Zeit bestehen unterschiedliche Wahrheiten in verschiedenen Systemen. Ein im CMS bereits gekündigter Vertrag ist im Abrechnungssystem noch aktiv. Eine im CRM erfasste Adressänderung erreicht das Versandsystem erst am nächsten Tag. Ein Asset, das bereits einem neuen Vertrag zugeordnet wurde, erscheint in der Inventarliste weiterhin beim alten Vertrag.

Die Alternative hierfür ist Echtzeit-Konsistenz durch Events. Wenn im CMS eine Änderung vorgenommen wird, wird sie sofort als Event veröffentlicht. Alle interessierten Systeme konsumieren das Event innerhalb von Millisekunden. Es gibt keine Inkonsistenzfenster mehr. Keine manuellen Abgleiche. Keine Batch-Fehler. Nur noch: Änderung erfolgt, Event wird

verschickt, alle Systeme sind synchron. Die technische Infrastruktur dafür existiert längst – Apache Kafka ist der de facto Standard für diese Art der Kommunikation.

Der Paradigmenwechsel zum Event-Denken

These 5: Event-Streaming ändert nicht nur, wie Systeme kommunizieren – es ändert auch, wie wir über Daten denken.

Event-Streaming ist kein neues Integrationstool. Es ist ein anderes Datenmodell, ein fundamentaler Paradigmenwechsel in der Art, wie wir Informationen erfassen und verarbeiten. Statt zu fragen: ‚Welchen Zustand hat Vertrag X jetzt?‘, fragen wir: ‚Welche Ereignisse sind mit Vertrag X passiert?‘ Der Unterschied erscheint subtil, ist aber fundamental.

In traditionellen Systemen überschreiben wir Daten. Der Vertragsstatus war ‚beantragt‘, jetzt ist er ‚aktiv‘. Was ist dazwischen passiert? Verloren. Wann genau fand der Wechsel statt? Wissen wir oft nicht. Warum er stattfand? Steht vielleicht in einem Kommentarfeld – oder auch nicht. Diese Informationsverluste summieren sich und machen es schwer bis unmöglich, Prozesse zu analysieren, Fehler zu identifizieren oder regulatorische Anfragen zu beantworten.

Event-Streaming macht jede Änderung zu einem unveränderlichen Fakt, zu einem Ereignis, das dokumentiert wird:

- Am 15.10.2025 um 14:23:17 Uhr wurde der Vertrag XYZ vom Status ‚beantragt‘ zum Status ‚aktiv‘ durch User ABC gewechselt; Grund: Bonitätsprüfung positiv.
- Am 15.10.2025 um 14:23:18 Uhr wurde für den Vertrag XYZ der erste Zahlungsplan erstellt.
- Am 15.10.2025 um 14:23:19 Uhr wurde der Vertrag XYZ dem Asset 4711 zugeordnet.

Diese Events sind unveränderlich und dokumentieren die komplette Historie. Sie ermöglichen Audit-Trails und können neu abgespielt werden. Und sie garantieren, dass alle Systeme dieselbe Sicht auf die Geschäftsvorfälle haben.

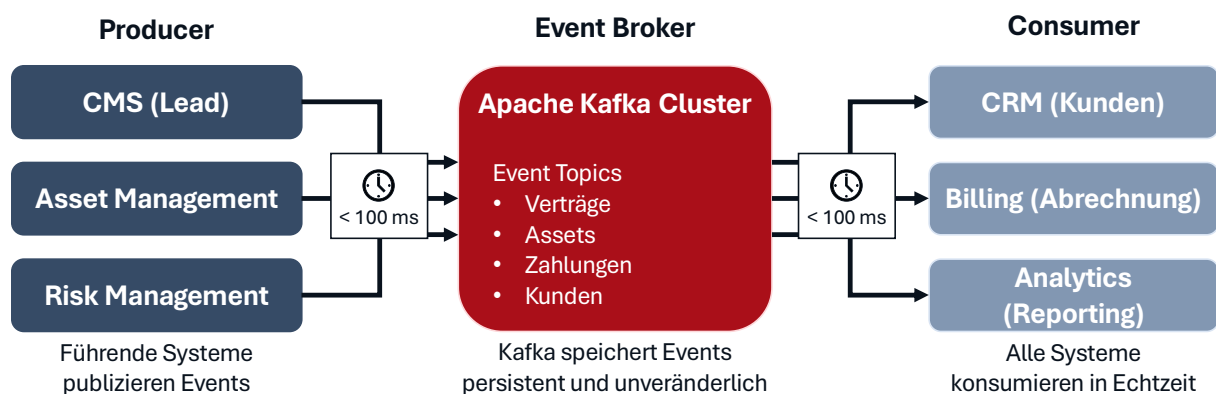


Abbildung 2 – Illustration des Event-Streamings

Für die Datenqualität bedeutet dies, dass die *Single Source of Truth* technisch, wie in Abbildung 2 dargestellt, umgesetzt wird: Jedes Event stammt aus genau einem führenden System. Konsistenz ist garantiert – alle Konsumenten sehen dieselben Events in derselben Reihenfolge. Fehlertoleranz ist eingebaut: Wenn ein Konsument ausfällt, kann er alle Events nachholen,

ohne dabei Daten zu verlieren. Nachvollziehbarkeit entsteht automatisch, weil die gesamte Historie dauerhaft verfügbar ist. Und schließlich wird Testbarkeit ermöglicht, da neue Systeme anhand historischer Events getestet werden können, bevor sie produktiv gehen.

Apache Kafka als Unternehmensinfrastruktur

These 6: Kafka ist kein IT-Tool – es ist die Dateninfrastruktur des Unternehmens, das Nervensystem der digitalen Organisation.

Apache Kafka hat sich als Standard für Event-Streaming etabliert, weil es drei Probleme löst, die für die Datenqualität entscheidend sind: Wie garantieren wir, dass keine Events verloren gehen? Wie stellen wir sicher, dass alle Systeme dieselben Events in derselben Reihenfolge sehen? Wie machen wir Events dauerhaft für Audits, Analysen und Fehlerdiagnosen verfügbar?

Die Antwort liegt in der Architektur von Apache Kafka: Events werden persistent gespeichert, nicht nur für Millisekunden wie in traditionellen Message Queues, sondern für Tage, Wochen oder Jahre, je nach Konfiguration. Jedes Event bekommt eine eindeutige Position, den sogenannten Offset. Konsumenten können Events in ihrem eigenen Tempo verarbeiten, ohne andere Konsumenten zu blockieren. Und wenn etwas schiefgeht, können Events beliebig oft neu abgespielt werden, um Probleme zu analysieren oder behobene Fehler nachzuvollziehen.

Dies bringt konkrete Vorteile für das Leasing-Geschäft: Eine Datenmigration ohne Downtime wird möglich. Neue Systeme können historische Events abspielen und ihren Zustand aufbauen, während das alte System noch läuft. Compliance erfolgt ohne zusätzlichen Aufwand, weil Audit-Trails automatisch aus der Event-Historie entstehen, ohne dass spezielle Logging-Mechanismen implementiert werden müssen. Die Fehleranalyse wird trivial, da sich jeder Geschäftsvorfall sich Schritt für Schritt nachvollziehen lässt, vom Auslöser bis zur Wirkung. Zudem ist eine Skalierungsfähigkeit eingebaut, wenn neue Konsumenten jederzeit hinzugefügt werden können, ohne das CMS zu belasten oder zu verändern. Und schließlich wird Echtzeit-Analytics möglich, indem das Reporting direkt auf den Event-Streams aufbaut, statt auf nächtliche Extrakte warten zu müssen.

Die Roadmap zum Erfolg

These 7: Ein CMS-Projekt ohne Datenstrategie ist ein Millionengrab – eine CMS-Einführung ist die ideale Gelegenheit, die Probleme strukturell zu lösen.

Wenn man ein neues CMS einführt, ohne gleichzeitig Datenqualität, Datenmodell und Datenarchitektur zu adressieren, wird das Projekt scheitern. Nicht sofort und nicht spektakulär, sondern schleichend: Mit jedem neuen Feature, das nicht richtig funktioniert. Bei jeder Schnittstelle, die manuelle Nacharbeit erfordert. Mit jedem Report, dessen Zahlen nicht stimmen. Die Kosten steigen, die Akzeptanz sinkt, der Frust wächst.

Die gute Nachricht ist aber, dass eine CMS-Einführung die ideale Gelegenheit ist, diese Probleme strukturell zu lösen. Man migriert ohnehin Daten, überarbeitet ohnehin Prozesse und sitzt ohnehin mit allen Fachabteilungen am Tisch. Man muss diese Chance nur nutzen – und zwar systematisch.

In der Praxis hat sich hier ein Vorgehen entlang von sechs Phasen als erfolgsentscheidend erwiesen. Das bewährte Vorgehensmodell ist in Abbildung 3 skizziert und wird im Folgenden erläutert.

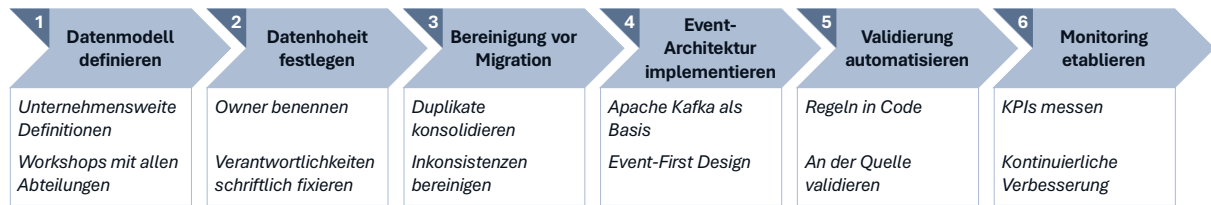


Abbildung 3 - Vorgehensmodell zur erfolgreichen CMS-Einführung

In Phase 1 wird das Datenmodell definiert. Bevor man ein CMS auswählt, muss das unternehmensweite Datenmodell für alle Kernobjekte erstellt werden. Klare Definitionen, Pflichtfelder, Validierungsregeln, Beziehungen zwischen Objekten sowie Statusmodelle und -lebenszyklen müssen festgelegt werden. Fachliche Workshops mit allen Stakeholdern sind unverzichtbar. Die resultierende Dokumentation wird zur verbindlichen Grundlage für alle Systeme.

In Phase 2 wird die Datenhoheit festgelegt, indem für jedes Datenobjekt ein eindeutiger Owner benannt wird. Verantwortlichkeiten werden schriftlich festgelegt: Wer pflegt die Daten? Wer definiert Änderungen? Wer stellt Qualität sicher? Governance-Prozesse werden für Schema-Änderungen und -Erweiterungen etabliert. Diese organisatorischen Regelungen sind genauso wichtig wie die technischen.

In Phase 3 erfolgt eine Bereinigung der Daten vor der Migration. Die Datenqualität in den Altsystemen wird analysiert. Duplikate werden identifiziert und konsolidiert. Inkonsistenzen werden bereinigt. Historische Daten werden validiert und, wo nötig, korrigiert. Nur saubere Daten werden ins neue System übernommen – der Migrationsschnitt ist die Chance auf einen Neuanfang. Wer alte Datenprobleme mitschleppt, wird sie im neuen System potenziert wiederfinden.

In Phase 4 wird die Event-Architektur implementiert. Apache Kafka oder eine vergleichbare Event-Streaming-Plattform wird ab Tag 1 als Grundinfrastruktur etabliert. Alle relevanten Geschäftsvorfälle werden als Events modelliert, und ein Schema-Registry wird für Event-Definitionen eingeführt. Monitoring wird für Event-Flüsse aufgebaut. So entsteht die technische Basis für langfristige Datenqualität, nicht als nachträgliches Add-on, sondern als fundamentales Designprinzip.

In Phase 5 wird die Validierung automatisiert, indem alle Validierungsregeln des Datenmodells in Code umgesetzt werden. Im CMS als führendes System, an APIs zur Input-Validierung, in Event-Producern als Quality Gate. Fehlerhafte Daten dürfen das System nicht verlassen und müssen an der Quelle gestoppt werden. Diese Validierung muss automatisch, konsistent und unumgebar erfolgen.

In Phase 6 wird schließlich das laufende Monitoring etabliert. Die Datenqualität wird kontinuierlich gemessen und hinsichtlich Vollständigkeit, Konsistenz, Aktualität und Validität überwacht. Dashboards werden für Datenqualitäts-KPIs aufgebaut, und automatische Alerts

werden bei Abweichungen konfiguriert. Regelmäßige Reviews mit Dateneinhabern werden durchgeführt, damit Datenqualität keine einmalige Initiative bleibt, sondern ein kontinuierlicher Verbesserungsprozess wird.

Fazit: Datenqualität als Erfolgsfaktor

Man kann das beste CMS der Welt kaufen. Man kann die modernste Technologie einsetzen. Man kann die teuersten Berater engagieren. Aber wenn die Daten schlecht sind, wird das Projekt scheitern. Datenqualität ist nicht das Sahnehäubchen des CMS-Projekts. Datenqualität ist das Fundament, auf dem alles andere aufbaut. Wenn das Fundament bröckelt, hilft auch der schönste Aufbau nicht.

Die CMS-Einführung ist die Chance, dieses Fundament neu zu gießen. Sauber, solide und zukunftssicher. Mit klaren Datenmodellen, eindeutigen Verantwortlichkeiten, automatischer Validierung und einer eventbasierten Architektur. Es erfordert Disziplin, Durchhaltevermögen und die Bereitschaft, auch unbequeme Diskussionen zu führen. Aber es zahlt sich aus – in Form schnellerer Prozesse, geringerer Fehlerkosten, zufriedenerer Kunden und einer guten Basis für spätere Innovationen.

Die Leasingbranche steht vor großen Herausforderungen: Regulatorische Anforderungen steigen, Kunden erwarten digitale Services, Wettbewerber drängen mit innovativen Geschäftsmodellen in den Markt. Unternehmen, die heute in Datenqualität investieren, sichern sich einen entscheidenden Wettbewerbsvorteil. Sie können schneller reagieren, effizienter arbeiten und neue Geschäftsmodelle umsetzen. Denn am Ende des Tages gilt eine einfache Wahrheit: Daten schlagen Technologie – immer.



Patrik Schädler ist Manager bei EGC mit Fokus auf IT-Architektur sowie Transformations- und Modernisierungsprojekte im Finanzdienstleistungsumfeld. Er unterstützt Finanzinstitute bei der strategischen Weiterentwicklung von Wertpapier-, Treasury- und Asset-Management-Prozessen. Parallel zu seiner Beratungstätigkeit verfolgt er eine Promotion (DBA), in der er wissenschaftliche Bewertungsansätze für Enterprise-IT-Architekturen entwickelt und diese systematisch mit praktischer Projekterfahrung verknüpft.



Zohaib Tariq ist Associate Manager bei EGC mit Schwerpunkt auf IT-Architektur sowie Transformations- und Migrationsvorhaben im Finanzdienstleistungsumfeld. Er unterstützt Finanzinstitute bei der strategischen Gestaltung und Implementierung von Contract-Management-Systemen und Asset-Finance-Projekten. Sein Fokus liegt auf der Definition zielgerichteter IT-Architekturen, Schnittstellen- und Integrationskonzepten.

Bisher erschienen:

- #1 Dr. Marco Weiß – *Welt und Wesen des Datenmanagements*; September 2020
- #2 Mario Kubik – „DLT“ – *Potenzial zur Zukunftstechnologie?*; November 2020
- #3 Jonathan Mislser – *Verwendung von Daten und Modellen in datengetriebenen Geschäftsmodellen*; Januar 2021
- #4 Johannes Bänsch – *Data Analytics und die Cloud*; März 2021
- #5 Kurt Tanneberger – *Ja, wo sind sie denn? – Daten im IT-Betrieb*; Mai 2021
- #6 Thomas Tiebor – *Data Governance als Fundament guten Datenmanagements*; Juli 2021
- #7 Dr. Marco Weiß – *Artificial Intelligence – Nature or Nurture?*; September 2021
- #8 Benjamin Sancar, Hermann Sgardelli – *Vertriebschancen in einer Bank durch Nutzung ihrer Daten und ihres Ökosystems*; November 2021
- #9 Abdalla El Kadi – *Datenvisualisierung – eine Geschichte in Bildern*; Januar 2022
- #10 Carsten Kuhlmann – *Kundenfokussierung: Daten und Technik*; März 2022
- #11 Luca Winters – *CIEM – Cloud Infrastructure Entitlements Management*; Mai 2022
- #12 Kristian Buric – *Der Informationsverbund - Daten (nur) für die Aufsicht?*; Juli 2022
- #13 Stefan Bauer – *Aufbau eines SOC (Security Operations Center) als Kommandozentrale zur Cyberabwehr*; September 2022
- #14 Thomas Tiebor – *Der Charme von IT-Datenarchitekturmustern*; November 2022
- #15 Dr. Marco Weiß – *Citius, Altius, Fortius – Daten im Sport*; Januar 2023
- #16 Simon Wilmerding – *Daten statt (nur) Bauchgefühl – Mit Datenanalysen den War for Talent gewinnen*; März 2023
- #17 Kay Helbig – *ESG-Berichterstattung bei Versicherungen*; Mai 2023
- #18 Dr. Marco Weiß – *Daten und Modelle in der 7. MaRisk-Novelle*; Juli 2023
- #19 Fabio Giacomo Müller – *Softwareeinführungsprojekte im internationalen Kontext*; September 2023
- #20 Stefan Bauer – *Digital Operational Resilience Act (DORA): Wie hoch wird die nächste Welle der Regulierung?*; November 2023
- #21 Prof. Dr. Hans-Gert Penzel – *Tokenisierung von Vermögenswerten: Wie die Zukunft des Eigentums aussehen wird*; Januar 2024
- #22 Andreas Sperlich – *Ehrlich bleiben bei der Kundenzufriedenheitsmessung: Die Schlüsselrolle von Integrität und Authentizität*; März 2024
- #23 Tomislav Basic, Simon Wilmerding – *Keine KI ist auch keine Lösung – in der ESG-Berichterstattung*; Mai 2024
- #24 Stefan Schnitzler – *Die Rolle der Daten im Retail Payment*; Juli 2024
- #25 Sören Bey, Simon Wilmerding – *Robuste Datenarchitektur: Das Fundament für effektiven KI-Einsatz*; September 2024

- #26 Mark Wieczorrek – *Effektive Kundendaten-Governance: Der Schlüssel zu nachhaltigem Geschäftserfolg*; November 2024
- #27 Sören Bey, Simon Wilmerding – *Skalierung von KI-Lösungen bei Finanzdienstleistern*; Januar 2025
- #28 Dr. Marco Weiß – *Business Intelligence-Strategie – Wertschöpfung aus Daten*; März 2025
- #29 Stefan Schnitzler, Dr. Rainer Schamberger – *Vom Datensilo zum Daten-Ökosystem: Warum FiDA eine große Chance ist, den Kunden in den Mittelpunkt zu rücken*; Mai 2025
- #30 Kay Helbig, Jörg Dreinhöfer – *Rückgrat der modernen IT – Mainframe?*; Juli 2025
- #31 Hauke Hinderlich – *Daten in Finanzen und Radsport: Auf dem Weg zu besserer Performance*; September 2025
- #32 Carsten Striebich – *Quantencomputer – Überblick über Status und Handlungsbedarf*; November 2025
- #33 Patrik Schädler, Zohaib Tariq – *Das Millionengrab: Warum CMS-Projekte an veralteten Datenkonzepten scheitern*; Januar 2026

Über den DataNavigator

Unterschiedliche Autoren beleuchten in der Reihe DataNavigator die vielfältigen Facetten des Datenmanagements und zeigen Ihnen interessante Aspekte auf.

Der DataNavigator wird von Eurogroup Consulting regelmäßig zweimonatlich herausgegeben und möchte Ihnen interessante Denkanstöße liefern. Wir freuen uns, Ihnen mit dem DataNavigator unsere Erfahrungen weiterzugeben und stehen Ihnen für die Diskussion gerne zur Verfügung!

Sie erreichen Autoren und Herausgeber bei Anregungen und Feedback unter der E-Mail-Adresse datanavigator@eurogroupconsulting.de.

