

Die Ablösung einer alten, abgekündigten CAFM-Software ist riskant und teuer, weil fehlerhafte Stammdaten, schlecht dokumentierte Schnittstellen und unklare Sicherheitsanforderungen ein Projekt schnell lähmen können. Dieser Leitfaden liefert eine schrittweise Checkliste mit Prioritäten, Aufwandsschätzungen und konkreten Prüfpunkten. Am Ende wissen Sie, welche Schnittstellen zuerst zu sichern sind, wie Sie Daten migrieren, welche Architekturfragen (Cloud oder On-Prem) Sie umtreiben sollten und wie Sie Nutzer und Berater praktisch einbinden.

Projektvorbereitung und Zieldefinition

Kernaussage: Setzen Sie von Anfang an maximal drei messbare Ziele, sonst wird das Projekt zum Dauerthema. Konkrete Ziele sollten Verfügbarkeit, Datenintegrität und Reporting-Performance sein; daneben definieren Sie den Rahmen für *Stammdatenübernahme, Schnittstellen, Sicherheit, Prozesseanalyse* und das *Einbeziehen der Nutzer*.

Governance zuerst: Benennen Sie eine Lenkungsgruppe mit klaren Entscheidern aus FM, IT und Controlling plus ein technisches Steuerungsteam. Legen Sie ein einfaches RACI für Entscheidungen an Schnittstellen, Cloud oder On-Premise Architektur und externe Berater fest. Ohne feste Eskalationswege endet ein Projekt in endlosen Abstimmungen.

Business Case konkretisieren: Vergleichen Total Cost of Ownership für Cloud- versus On-Premise-Optionen inklusive Integrationskosten zu ERP, Zutrittskontrolle, Energiemanagement, CAD und BIM. Bewerten Sie nicht nur Lizenzkosten, sondern auch Betrieb, Backup, Sicherheitszertifizierungen und Aufwand für Berichte und Analysen.

Priorisierung: Impact versus Aufwand

Praktischer Entscheidungsmechanismus: Erstellen Sie eine Prioritätenmatrix, die jeden Use Case nach fachlichem Impact und Integrationsaufwand bewertet. Geben Sie Integrationsfähigkeit (APIs, IFC/COBie, REST) ein höheres Gewicht als Featurewünsche in GUI und schicken Sie nur Schnittstellen in die erste Welle, die hohen operativen Impact haben. Tradeoff: Wer Integrationsfähigkeit opfert, zahlt später mit deutlich höheren

Schnittstellenaufwänden und längeren Projekten. In der Praxis ist eine lösungsunabhängige API-Checkliste wichtiger als ein funktionsreicher Prototyp.

Konkretes Beispiel: Ein kommunales Klinikum priorisierte bei der Ablösung die Anbindung an das SAP Finanzmodul und die Zutrittskontrolle. Es startete mit einem Pilotgebäude, validierte COBie Exporte aus Revit und etablierte ein Azure-basiertes Middleware Layer, bevor weitere Gebäude folgen konnten. Dadurch ließ sich das Risiko von Schnittstellenfehlern begrenzen und die Go-live Checkliste wurde pragmatisch abgearbeitet.

Kriterium	Empfohlene Gewichtung
Integrationsfähigkeit (APIs, IFC, COBie)	30
Betriebsmodell und Sicherheitsanforderungen	25
Funktionaler Fit der Kernprozesse	20
Kosten über 5 Jahre (TCO)	15
Benutzerakzeptanz und Schulungsaufwand	10

Wichtig: Beauftragen Sie Berater nur mit nachweislicher Erfahrung in CAFM-Ablösungen inklusive CAD/BIM und Schnittstellenprojekten. Vermeiden Sie Berater, die primär Produkte verkaufen statt Integrationsplaybooks zu liefern.

Nächster Schritt: Sobald Ziele, Governance und Prioritätenmatrix stehen, planen Sie eine kurze, aber verbindliche Scope Freeze Phase. Danach folgt die detaillierte Prozessanalyse und das Einbinden von Key Usern für die Pilotplanung.

Prozesse analysieren und Nutzer

einbeziehen

Kurz und klar: Wer Prozesse nur in PowerPoint beschreibt, bekommt beim Go-live Überraschungen. Führen Sie pragmatische Prozessaufnahmen durch, die Datenflüsse, Auslöser und Benutzerentscheidungen abbilden – nicht nur Ablaufdiagramme. Nur so wird später klar, welche *Stammdaten*, welche Schnittstellen und welche Sicherheitsregeln tatsächlich notwendig sind.

Quellen kombinieren: Nutzen Sie Systemlogs, Ticketexporte, Gebäudeautomationstraces und direkte Beobachtung. Vergleichen Sie tatsächliche Bearbeitungszeiten aus dem Helpdesk-Export mit dem, was Fachbereiche sagen. Diese Kreuzvalidierung entlarvt Workarounds, die Stammdaten und Schnittstellen belasten.

Workshop-Fahrplan für realistische Prozessaufnahme

- Vorarbeit: Sammeln Sie CSV-Exporte von Tickets, SLA-Reports und relevante ERP-Schnittstellenbeschreibungen.
- Halbtägiger Kernworkshop: Zeichnen Sie den Ist-Workflow auf, markieren Sie Ausnahmen und mobile Arbeitsschritte.
- Quantifizierung: Ermitteln Sie Häufigkeit, Zeitaufwand und Datenobjekte pro Prozessschritt.
- Abschlussartefakte: Prozesskarte, *Change Impact* Eintrag und Prioritätenliste für Schnittstellenentwicklung.

Tradeoff und Limitierung: Nutzerbeteiligung kostet Zeit und erzeugt widersprüchliche Anforderungen. Setzen Sie deshalb auf ein zweistufiges Modell: ein kleines Key-User Team für Entscheidungsbefugnis und eine größere Anwenderbefragung zur Validierung. Zu viele Entscheider verzögern technische Entscheidungen – zu wenige erhöhen das Risiko, dass reale Abläufe übersehen werden.

Konkretes Beispiel: In einem Universitätsrechenzentrum deckte die Analyse auf, dass Reinigungspersonal Reparaturanfragen per WhatsApp an die Hausmeister sendete. Statt eine

teure API für diese Kommunikation zu bauen, wurde ein einfaches mobiles Formular eingeführt, das COBie-Metadaten erfasste und direkt in das CAFM importierbar machte. Die Lösung reduzierte Schnittstellenarbeit und verbesserte gleichzeitig Datenqualität.

Praktische Schlussfolgerung für Schnittstellen und Sicherheit: Ordnen Sie Prozesse den Datenobjekten zu: Welche Prozesse brauchen Echtzeitdaten aus Gebäudeautomation, welche nur periodische Exporte aus ERP? Priorisieren Sie Schnittstellen nach operativem Risiko und Single Source of Truth. Beachten Sie dabei, dass Nutzerfreundlichkeit manchmal mit schärferen Sicherheitsmaßnahmen kollidiert – entscheiden Sie bewusst, wo Komfort zugunsten von IT-Sicherheit reduziert wird.

Nutzer früh einbeziehen, aber entscheiden Sie mit klaren Akzeptanzkriterien – nicht nach Mehrheitsprinzip.

Liefergegenstände aus dieser Phase: Prozesslandkarte mit Varianten, priorisierte Schnittstellenliste, Change Impact Register, Key-User-Liste, und ein kleines Testskript für die Pilotmigration. Starten Sie den Pilot auf 2 bis 3 hochpriorisierten Prozessen – das ist der Punkt, an dem Theorie in Praxis geprüft wird.

Stammdatenübernahme, Datenmigration und Bereinigung

Kernproblem: Migrationen scheitern selten an der Zielsoftware, sondern an mangelhafter Identitätssicherung der Stammdaten. Legen Sie sofort ein *Source of Truth*-Register an, das für jedes Datenobjekt folgende Felder enthält: Quellsystem, letzter Änderer, `source_id`, Verantwortlicher, Aktualitätsdatum und Qualitätsstatus.

Praktische Migrationsstrategie

Arbeiten Sie sequenziell: Discovery, Profiling, Cleaning, Mapping, Transformation, Test-Load, Verifikation, Cutover. Entscheidungspunkt: Säubern Sie so viel wie nötig vor der Übernahme,

aber nicht mehr. Vollständige Bereinigung verzögert Lieferungen; partielle Bereinigung plus klare Korrekturprozesse nach Go-live reduziert das Projektrisiko.

1. Dateninventar anlegen: Kartieren Sie Gebäudedaten, Raumstammdaten, Anlagen, Vertragspartner, Kostenstellen und CAD/BIM-Artefakte inklusive Dateiformat und Exportpfad.
2. Qualitätsgates definieren: Legen Sie für Vollständigkeit, Duplikate und Referentielle Integrität feste Schwellenwerte und automatisierte Prüfregele fest.
3. Feldmapping erstellen: Dokumentieren Sie Altfeld -> Neufeld und halten Sie Transformationen in Skripten fest; behalten Sie `source_id` als Primärschlüssel für Rückverfolgbarkeit.
4. Transformationsregeln automatisieren: Verwenden Sie ETL/ELT Tools wie Talend oder Safe Software FME für Geodaten; vermeiden Sie Einmalmanipulationen in Excel.
5. Testläufe und Reconciliation: Führen Sie mindestens drei Testläufe pro Datenbereich durch und vergleichen Sie Counts, Schlüsselverteilungen und stichprobenhafte Business-Checks.
6. Cutover & Rollbackplan: Definieren Sie kleine, atomare Cutover-Wellen mit klaren Rollback-Triggers und Kommunikationsplan für betroffene Nutzer.

Tradeoff: Historische Transaktionsdaten vollständig zu importieren kostet Zeit und erhöht Komplexität. In der Praxis ist es oft besser, Rohdaten zu archivieren und nur aggregierte Kennzahlen in das neue System zu übernehmen – das reduziert Ladezeiten und vereinfacht Reporting.

Konkretes Beispiel: Ein Immobilienverwalter musste CAD-Grundrisse, COBie-Sheets und Vertragsdaten migrieren. Das Team exportierte COBie aus Revit, konvertierte Geometrien mit FME und bewahrte alte Rechnungszeilen nur im Archivformat auf. Durch das Beibehalten der alten Raumkennzeichnungen als `legacyroomid` war Rückverfolgbarkeit zu historischen Störungen jederzeit möglich.

Wichtiges Urteil: Neuvergabe von IDs während der Migration ist der schnellste Weg, Historie und Schnittstellen zu zerstören. Behalten Sie Legacy-IDs, pflegen Sie ein Mapping und vermeiden Sie direkte Manipulation von CAD/BIM-Geometrien, wenn der Business-Use nicht geprüft wurde.

Priorisieren Sie Datenobjekte nach Betriebsrelevanz: Räume, Anlagen mit Wartungsverträgen, Zutrittsdaten, gefolgt von umfassenden historischen Logs.

Liefergegenstände aus dieser Phase: Dateninventar (CSV), Feldmapping (Dokument + Skript), Transformationsskripte, Testladeprotokolle, Reconciliation-Report und Archivzugriffsplan. Nutzen Sie Checklisten.

Zur Sicherheit: Protokollieren Sie jede Migrationsaktion und behalten Sie Prüfbücher für DSGVO-Relevanz. Ziehen Sie bei Unsicherheit die BSI-Leitfäden hinzu und prüfen Sie IFC/COBie-Exports gegen buildingSMART Prüfregeln.

Schnittstellen, Systemanbindung und CAD BIM Integration

Kernaussage: Nicht jede Verbindung muss in Echtzeit bestehen; entscheiden Sie Schnittstellen nach *Betriebszweck* und nicht nach technologischem Ideal. Priorisieren Sie Integrationen nach operativem Risiko, Datenverfügbarkeit und Wartbarkeit.

Technische Integrationsmuster und ihre Eignung

Pattern-Entscheidung: Setzen Sie auf einfache Batch-Exports für Abrechnungsdaten und Reports, Event- oder API-basierte Verbindungen für Störmeldungen und ein lokales Gateway für Gebäudeautomation. *Point-to-point* ist billig am Start, aber teuer im Betrieb; ab drei bis vier angebundenen Systemen lohnt sich Middleware oder ein iPaaS.

- Batch (CSV/ETL): geeignet für historische Logs, Rechnungsdaten, Nachtläufe für Reporting.
- API/Event (REST / Webhook): erforderlich für Tickets, Störmeldungen, Zutrittssynchronisation mit niedriger Latenz.
- Gateway/Edge (OPC UA / MQTT lokaler Broker): unverzichtbar, wenn die Gebäudeleittechnik niedrige Latenz oder lokale Sicherheit verlangt.
- Middleware/iPaaS: reduziert langfristig Wartungsaufwand, bietet Mapping, Monitoring und Retry-Logik.

Praktische Einschränkung: Viele BIM-Modelle enthalten keine operativen IDs. Erwarte keine saubere Mapping-Kohärenz zwischen CAD/BIM und CAFM ohne vorherige Konventionen (z. B.

assetTag oder legacyId) im Modell.

CAD/BIM: Geometrie versus Betriebsmessdaten

Wichtige Unterscheidung: Trennen Sie Geometrie (Pläne, Räume) von semantischen Asset-Daten (Hersteller, Wartungsintervall). IFC eignet sich für Geometrie und Struktur, COBie für assetbezogene Stammdaten. In der Praxis brauchen Sie oft eine *vereinfachte Geometrie* im CAFM und die ausführliche BIM-Geometrie archiviert.

Konkretes Beispiel: Ein Industriebetrieb koppelte die Gebäudeleittechnik per OPC UA an ein lokales Gateway, das Störmeldungen an das CAFM übermittelte. Parallel wurden IFC-Exports aus ArchiCAD mit Safe Software FME zu COBie-Tabellen transformiert und nur die minimal nötigen Asset-Eigenschaften (Hersteller, Typ, assetTag) übernommen. Ergebnis: deutlich weniger fehlerhafte Zuordnungen und geringerer Pflegeaufwand.

Sicherheits- und Betriebsentscheidung: Wenn sensible Steuerdaten im Spiel sind, platzieren Sie Steuerpfade lokal und spiegeln nur Metadaten in die Cloud. Für Reporting und BI ist eine Cloud-Aggregation sinnvoll; für Steuerbefehle an Anlagen ist On-Premise oder Hybrid die sicherere Wahl.

Praktischer Prozessvorschlag: Vereinbaren Sie früh Interface-Owner, SLA für Latenz/Availability, sowie ein Fehler- und Retry-Verhalten. Legen Sie außerdem ein Minimalset an BIM-Eigenschaften fest, das in jedem Modell vorhanden sein muss, bevor eine Übernahme stattfindet.

Kurzauftrag: Wenn mehr als drei Systeme angebunden werden: planen Sie Middleware, definieren Sie assetTag/legacyId im BIM und erzwingen Sie einfache QoS-SLA für jede Schnittstelle. Sonst investieren Sie später in teure Ad-hoc-Fixes.

Weiterlesen: Zur technischen Umsetzung und Protokollchecklisten und die buildingSMART Vorgaben für IFC/COBie.

Sicherheit, Datenschutz und Architekturentscheidung Cloud oder On-Premise

Kernaussage: Sicherheits- und Datenschutzanforderungen müssen die Architekturentscheidung führen; technisch elegante Cloud-Funktionen sind wertlos, wenn die Betriebssicherheit oder Compliance nicht gewährleistet sind.

Beginnen Sie mit einer *Datenfluss-Karte*: welche Daten werden gesteuert (Zugriffssteuerung, Steuerbefehle), welche nur gelesen (Metering, Logs), und welche personenbezogen sind. Treffen Sie Architekturentscheidungen entlang dieser Flüsse: Steuerpfade bleiben lokal, Telemetrie kann in die Cloud gespiegelt werden. Diese Trennung reduziert Angriffsfläche ohne Reporting-Fähigkeiten einzuschränken.

Konkrete Sicherheitsanforderungen, die umgesetzt sein müssen

Fokussieren Sie auf drei prüfbare Anforderungen: Identity & Access Management (zentrale Authentisierung, Rollen, Service-Accounts), Verschlüsselung (TLS für Transit, AES-256 oder gleichwertig für at-rest) und Auditierbarkeit (vollständige, unveränderbare Protokolle für Migrationen und Schnittstellenzugriffe). Ergänzen Sie das mit regelmäßigen Penetrationstests und einem SIEM-Konzept. Ziehen Sie die BSI-Vorgaben zurate: BSI.

Tradeoff: Cloud bietet schnelle Skalierung für BI und ML-Analysen, aber jede Cloud-Migration erhöht organisatorische Anforderungen: Verträge, Nachweisführung, Datenflussdokumentation und Exit-Szenarien. On-Premise reduziert Abhängigkeiten, kostet jedoch langfristig mehr Personal und verzögert Innovationen.

- Entscheidungsfragen: Wer benötigt Steuerrechte für Anlagen? (nur lokale Operatoren?)
- Compliance-Check: Welche Daten sind DSGVO-sensitiv und wie lange müssen sie aufbewahrt werden?

- Integrationsbedarf: Benötigen Schnittstellen niedrige Latenz oder lokale VLAN-Segregation?
- Betriebsreife: Hat Ihre IT Kapazität für Patching, Backups und 24/7-Monitoring?

Praktisches Urteil: Hybride Architektur ist in den meisten CAFM-Ablöseprojekten der realistischer Kompromiss. Lokale Gateways für Gebäudeautomation und Zutritt, Cloud für BI, Archivierung und DevOps-gestützte Reportingpipelines. Akzeptieren Sie dadurch zusätzliche Komplexität in der Integration zugunsten besserer Sicherheit und Skalierbarkeit.

Konkretes Beispiel: Ein kommunales Krankenhaus behielt BMS-Steuerpfade und Zutrittskontrolle vollständig on-premise hinter separaten VLANs und einem OPC UA-Gateway. Telemetrie und KPI-Daten wurden per verschlüsseltem Stream in eine Azure-Instanz gespiegelt, wo Power BI Dashboards laufen. Das ermöglichte schnelle Auswertungen ohne Betriebskritische Steuerbefehle in die Cloud zu geben.

Wichtig: Definieren Sie Sicherheitsakzeptanzkriterien vor der Ausschreibung und binden Sie diese in SLAs und Prüfpläne ein.

Sofortmaßnahme: Erstellen Sie eine kurze Checkliste für den RFP: Datenklassifizierung, Verschlüsselungsanforderungen, PenTest-Frequenz, Retentionsfristen (DSGVO), bei Cloud: Datenlokation und Exit-Plan. Ohne diese Vorgaben ist ein Vergleich der Angebote bedeutungslos.

Nächster Schritt: Übersetzen Sie die Sicherheitsanforderungen in testbare Akzeptanzkriterien für den Pilot (PenTest-Ergebnis, Latenz-SLA, SIEM-Events) und prüfen Sie diese in einem kurzen, realen PoC. Dann entscheiden Sie endgültig Cloud, On-Premise oder Hybrid auf Basis belastbarer Messwerten. Nicht auf Basis von luftigen (oder lustigen?) Versprechen.

Berichte, Analysen und Anbindung an

andere Systeme

Klare Aussage: Reporting und Analytics sind keine netten Extras, sie sind die operative Oberfläche des neuen CAFM. Wenn *Stammdatenübernahme, Schnittstellen, Sicherheit, Prozessanalysen, Nutzer-Feedback und Anbindung andere Systeme* nicht von Anfang an wie ein "interner Vertrag" innerhalb des Projektes behandelt werden, produziert das Ganze am Ende hübsche Screens mit falschen Zahlen. Habe ich leider schon gesehen...

Datenvertrag zuerst: Definieren Sie für jede Kennzahl ein kleines Datencontract-Dokument: Quelle, Aggregationsregel, Aktualität (SLA), Owner, und Prüfsumme für die Validierung. Diese Verträge sind kein Papierkrieg, sondern die Schnittstelle zwischen CAFM, ERP, Zutrittssystemen, BMS und BI-Layer und verhindern späteres Fingerzeigen und Streit zwischen den Beteiligten.

Konkrete Prüf- und Umsetzungsaufgaben

- KPIs als Code: Schreiben Sie KPI-Definitionen so, dass sie reproduzierbar sind (SQL, DAX, oder Python-Notebook) und legen Sie eine Testabdeckung mit Beispiel-Datensätzen an.
- Quelle pro KPI: Bestimmen Sie eine Single Source of Truth; wenn das CAFM nicht alle Attribute liefert, nehmen Sie ERP oder BIM als autoritative Quelle und dokumentieren die Reconciliationschritte.
- Datenlatenz klar regeln: Legen Sie fest, welche Dashboards Echtzeit benötigen und welche mit Nachtläufen auskommen; Echtzeit kostet Integrationsaufwand und Sicherheitsprüfungen.
- Archivstrategie: Exportieren und archivieren Sie alte Transaktionen außerhalb des Produktiv-CAFM, aber stellen Sie aggregierte historische Metriken für Analysen bereit.

Tradeoff und Einschränkung: Echtzeit liefert Reaktionsfähigkeit, aber skaliert schlecht über heterogene Schnittstellen (BMS, Zutritt, IoT). In der Praxis ist ein hybrider Ansatz sinnvoll: Event-getriggerte Updates für Tickets und Störungen, batchbasierte ETL für Abrechnung, und ein Data-Lake/BI-Layer für historische Analysen.

Konkretes Beispiel: Ein Büroimmobilienverwalter nutzte Talend für ETL, Safe Software

FME um CAD/IFC-Geometrien zu vereinfachen und befüllte einen Azure Data Lake. Die operativen Dashboards in Power BI werden nachts aktualisiert; Ticket-Events dagegen kommen per REST in Echtzeit. Das Ergebnis: belastbare MTTR-Kennzahlen und weniger manuelle Nacharbeiten bei Rechnungsprüfungen.

Praxisurteil: Viele Organisationen verlangen native, fertige Reports aus dem CAFM. Das ist ein falscher Wunsch (und ich weiss, wovon ich rede...). Moderne BI-Werkzeuge sind flexibler, erlauben Auditierbarkeit und Versionierung und verhindern Workarounds. Setzen Sie das CAFM als Authoritative Source für Stammdaten und nutzen ein separates BI-Layer für Ad-hoc-Analysen und Management-Dashboards. Oder von mir aus einen internen Reportgenerator, der wirklich flexibel, und einfach zu bedienen ist. Plus weitere Datenquellen anzapfen kann und gut dokumentiert ist. Aber bitte nur als günstige Notlösung.

Liefergegenstände für die Ausschreibung und den Pilot: Datencontracts für Top-10 KPIs, minimaler ETL-Flow mit Testdaten, Report-Owner Matrix, Retention/Archivplan, und ein kleines PoC-Dashboard mit Live-/Batch-Daten.

Nächster Schritt: Erstellen Sie zwei kurzprüfbare PoCs — eins für Live-Events (Tickets) und eins für Nightly-Aggregates (Sie wissen schon: das was Sie jeden Morgen erwartet...) — und bewerten Sie Aufwand, Sicherheit und Testbarkeit, *bevor* Sie Ihr Reporting flächendeckend definieren.

Projektorganisation, Beraterrolle, Zeitplanung, Go-live und Nutzerakzeptanz

Klares Steuerungsmodell entscheidet: Legen Sie unbedingt ein kleines Steuerkreis-Gremium fest (Entscheider aus FM, IT, Controlling) und *einen einzigen Projektleiter* mit *Budgethoheit*. Ohne eine eindeutige Entscheidungsinstanz verlangsamt sich alles – von *Stammdatenübernahme* bis zur Priorisierung von *Schnittstellen* bis hin zur *Sicherheit*.

Projektorganisation und Beraterrolle

Rollen pragmatisch definieren: Trennen Sie strategische Steuerung, fachliche Streams (Betrieb, Prozesse, Nutzer), IT-Streams (Schnittstellen, Systemintegration) und Lieferantenmanagement. Benennen Sie für jede Schnittstelle einen Interface-Owner mit Testverantwortung.

- Steuerkreis: entscheidet über Scope-Änderungen und Budgetfreigaben
- Projektleitung: liefert Zeitplan, Eskalationspfad und Statusberichte
- Fach-Owner: verantworten Prozessabbildung und Abnahmekriterien
- Interface-Owner: übernehmen Schnittstellenoptimierung, SLA und Monitoring

Berater einsetzen — aber richtig: Beauftragen Sie Berater auf Ergebnisse, nicht auf Stundenbasis (ja, ein frommer Wunsch, ich weiß). Prüfen Sie Referenzen in CAFM-Ablösungen mit *CAD/BIM*-Migrationsprojekten und verlangen Sie einen Transferplan für Wissen und Artefakte (Skripte, Mappings, Testcases). Vermeiden Sie Berater, die Schnittstellen nur als Add-on verkaufen; gute Beratung liefert ein wiederholbares Migrations-Runbook.

Zeitplanung, Go-live und praktische Prioritäten

Meilensteine statt fixe Laufzeiten: Planen Sie Scoping, Prozessanalyse, Pilot, gestaffelten Rollout und Hypercare als klar abgegrenzte Meilensteine mit Ein-Ticket-Abnahmekriterien. Jedes Release braucht ein Testpaket: Datenchecks, Schnittstellen-Smoke, Authentifizierungsprüfung, und ein funktionierendes Eskalationsschema.

Go-live Pragmatik: Vermeiden Sie Big-Bang, wenn kritische Schnittstellen zu ERP, Zutritt oder BMS betroffen sind. Entscheiden Sie vor dem Cutover über Minimalfunktionsumfang, Rollback-Trigger und eine 48-72-Stunden Hypercare-Schicht mit dedizierten IT- und FM-Ressourcen.

Tradeoff: Je stärker Sie Nutzerwünsche während der letzten Monate berücksichtigen, desto größer das Risiko von "Scope Creep". Entscheiden Sie bewusst: entweder stabiler Go-live mit Nachpflegeprozess oder erweiterter Go-live mit höherem Betriebsrisiko. Die meisten Projekte neigen interessanterweise zum Zweiten...

Konkretes Beispiel: Ein mittelgroßer Immobilienbetreiber führte die Ablösung gebäudeweise durch. Für Gebäude A wurden alle Tickets, SAP-Schnittstellen und Zutrittsdaten in einer Pilotwelle synchronisiert; Gebäude B blieb sechs Wochen im Parallelbetrieb. Parallel lief ein zweiwöchiger Hypercare-Support, in dem Interface-Owner offensiv Logs prüften und Datenreconciliations automatisiert wurden. Diese Staffelung verhinderte einen umfangreichen Rollback und sorgte für kontrollierte Nutzerakzeptanz.

Nutzerakzeptanz messbar machen: Schulen Sie Schlüsselanwender praxisnah in *echten Workflows*, nicht in Funktionsvorführungen. Messen Erfolg über Nutzungskennzahlen (Loginrate, Ticketanlage ohne manuellen Workaround, First-Time-Fix-Quote) und nicht nur über Zufriedenheitsumfragen.

Wichtiges Urteil: Externe Berater dürfen nicht die einzige Quelle für Migrationsskripte und Schnittstellendokumentation sein. Bestehen Sie auf Repos, ausführbaren Testcases und Übergabeprotokollen. So bleibt Ihr Team handlungsfähig, wenn das Projekt in den normalen Betrieb übergeht.

Kurzauftrag für die Ausschreibung: liefern Sie ein Projektplan-Template mit Meilensteinen, ein Migrations-Runbook, Interface-Owner-Liste, Hypercare-Plan und einen Wissenstransfer-Nachweis. Ohne diese Liefergegenstände bleibt der Vertrag blinder Einkauf. Und den Film habe ich zu oft gesehen ;-)

Nächster Schritt: Definieren Sie jetzt die fünf unverzichtbaren Abnahmekriterien für Ihren Pilot — zum Beispiel Datenintegrität, Schnittstellenstabilität, Sicherheitschecks, Nutzer-Workflows und Hypercare-Verfügbarkeit — und binden Sie diese in den Vertrag ein.

Frequently Asked Questions

Direkt auf den Punkt: Die meisten Ablöseprojekte stocken an sechs Themen: unklare Datenidentitäten, ungeprüfte Schnittstellen, fehlende Sicherheitsregeln, nicht eingelöste Nutzeranforderungen, zu lange Projektläufe und falsche Beraterauswahl. Die folgenden FAQ liefern knappe, umsetzbare Antworten und jeweils eine direkte Handlungsempfehlung.

Kurzantworten mit sofort umsetzbaren Maßnahmen

- Wie beurteile ich, ob die Daten übertragbar sind: Führen Sie ein schnelles Profiling durch (Schlüsselverteilungen, fehlende Referenzen, Häufigkeit doppelter IDs) und markieren Sie Datensätze mit kritischen Fehlern. *Aktion:* Sperren Sie problematische Datensätze für den Cutover und erstellen Sie ein Bereinigungsbacklog mit Priorität.
- Welche Schnittstellen zuerst: Priorisieren Sie nach operativer Wirkung, nicht nach technischer Eleganz. *Aktion:* Erstellen Sie eine Top-3 Schnittstellenliste nach Ausfallfolgen und stellen Sie Interface-Owner bereit.
- Cloud oder On-Premise – wie entscheide ich praktisch: Treffen Sie die Architekturentscheidung entlang der Datenpfade: Steuerbefehle lokal, Telemetrie und Reports in der Cloud. *Aktion:* Definieren Sie für jede Schnittstelle, ob Steuerdaten lokal bleiben müssen und dokumentieren Sie die Gründe.
- Wie verhindere ich, dass Nutzer Workarounds behalten: Key-User nicht nur informieren, sondern mit klaren Akzeptanzkriterien testen lassen. *Aktion:* Führen Sie einen kurzen Realbetriebstest mit echten Ticketzyklen durch und dokumentieren Sie Workarounds als Migrationsaufgabe. Sie wollen die Nutzer nicht als Feind. Nehmen Sie das Ernst, bitte!
- Wann brauche ich einen Berater: Wenn Sie keine internen technischen Migrationsskripte oder Middleware-Erfahrung haben. *Aktion:* Beauftragen Sie Berater nach Ergebnis (ja, das wird schwer...), verlangen Sie Codeübergabe in ein Repo (OK, das wird auch schwer...) und bestehen Sie auf ausführbaren Testcases.
- Muss ich historische Transaktionen komplett migrieren: Nicht zwingend; historische Detaildaten erhöhen Aufwand und Fehlerquote. *Aktion:* Exportieren Sie Rohdaten in ein prüfbares Archiv und übernehmen Sie nur die für Betrieb und Reporting notwendigen Aggregationen. Bitte: keine Daten-Friedhöfe umziehen!

Tradeoff, der oft unterschätzt wird: APIs kosten initial mehr Implementierungszeit als CSV-Exporte, sind im Betrieb aber deutlich robuster. Wenn Sie kurzfristig Zeit gewinnen wollen, akzeptieren Sie die technische Schuld bewusst und planen ein Refactoring der Schnittstellen in der zweiten Projektphase. Aber das kostet mehr :-)

Praxisfall: Ein kommunaler Campus setzte in der Ablösung auf ein zweistufiges Modell. Zuerst wurden Zutrittsdaten per nightly batch synchronisiert, gleichzeitig wurde eine REST-API für Ticketevents als langfristige Lösung aufgebaut. Beim Pilot wurde eine mobile Labeling-Aktion

durchgeführt, bei der Techniker QR-Codes an kritischen Assets anbrachten, um die LegacyId sauber zu verknüpfen. Ergebnis: schnellerer Betriebseinstieg und planbarer Übergang zur Echtzeitintegration.

Häufiger Fehler: Entscheidungen über Architektur oder Berater erst beim Ausschreibungsabschluss treffen. Treffen Sie diese Entscheidungen vorher, sonst vergleichen Sie Äpfel mit Birnen.

Checkliste:

- 1) Definieren Sie 3 kritische Schnittstellen und ernennen Sie Owner;
- 2) Legen Sie LegacyId als unveränderlichen Schlüssel fest;
- 3) Archivieren Sie detaillierte Historie außerhalb des Produkktivsystems;
- 4) Fordern Sie Codeübergabe und Testcases vom Berater.

Konsequente Maßnahme: Übersetzen Sie jede FAQ-Antwort in eine Aufgabe mit klarer Frist und einem Verantwortlichen. Ohne diese Zuordnung bleiben gute Antworten nur schöne Absichtserklärungen. Und das ist blöd, sorry.

Folgende Schritte Sie können sofort umsetzen:

- 1) Starten Sie ein 48-stündiges Profiling für Ihre Top-3 (oder von mir aus Top-2) Daten-Objekte,
- 2) Benennen Sie die drei Schnittstellen mit höchstem Betriebsrisiko und deren Owner,
- 3) Legen Sie ein Repo für Migrationsskripte an und fordern Sie erste Testläufe an.

Ich wünsche gutes Gelingen!1!!

Wie hilfreich war dieser Beitrag?

Klicke auf die Sterne um zu bewerten!

Bewertung Abschicken

Durchschnittliche Bewertung / 5. Anzahl Bewertungen:

Top-Schlagwörter: fehler, vergleich, kosten, cloud, BSI, cafm, erfolg, sicherheit, hersteller, Software

Verwandte Artikel

- CAFM-Software: Besser Standard-Software kaufen oder selbst eine CAFM-Software entwickeln?
- CAFM-Software: Alles was Sie als Dummie wissen sollten ;-)
- CAFM-Software im Vergleich: Welches Tool passt am besten zu Ihrem Unternehmen?