

Eingereicht von

-

Angefertigt am
Information Engineering

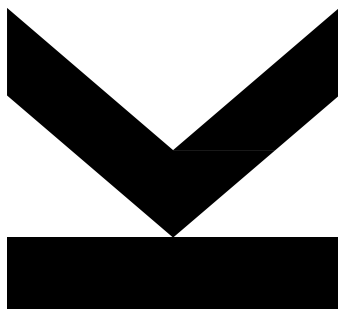
Beurteiler / Beurteilerin
**Univ-Prof.in Dr.in
Barbara Krumay, Bakk.
MSc (WU)
Mag. Dr. David Rückel**

Partner

-

STILLEGUNG VON LEGACY-SYSTEMEN

Februar. 2023



IT-Projekt Wirtschaftsinformatik
im Bachelorstudium
Wirtschaftsinformatik

1. Einleitung – Problem

In dieser Arbeit wird der Prozess der Stilllegung von Legacy-Systemen in Kooperation mit der voestalpine Steel Division – Nachfolgenden kurz als voestalpine bezeichnet – berücksichtigt.

Bei einer System Stilllegung bzw. Software Retirement muss ein Unternehmen viele Aspekte beachten, um diese problemlos durchführen zu können (Pilorget, 2011). Fragen nach den unterschiedlichen Verantwortlichkeiten, den möglichen Risiken wie beispielsweise Datenverluste bzw. Sicherheitsbedenken müssen zeitgerecht und strategisch geplant werden (Pilorget, 2011). Ebenso sind Fragen zu klären, wie beispielsweise jene, ob und wie der Zugang zu den historischen Daten gewährleistet wird, oder ob die Daten zuvor entweder ins neue System migriert oder auf anderen Speichermedien zwischengespeichert werden sollen (Pilorget, 2011). Zentral ist vor allem auch der Zeitpunkt der endgültigen Systemabschaltung und in welcher Variante dies vorgenommen wird (Pilorget, 2011).

Aufgrund der Komplexität des Themas wurde vom Partnerunternehmen voestalpine ein Vorgehensmodell angefordert, welches alle Themen rund um die Stilllegung von Legacy-Systemen abdeckt. Dieses Modell stellt jedoch eine allgemeine Lösung dar, die auch andere Unternehmen bei dem Problem unterstützen kann. Wie oben beschrieben, gliedert sich die Komplexität in mehrere Themen, die in dieser Seminararbeit erörtert werden, darunter das Application Lifecycle Management (ALM), das Data Governance im Hinblick auf die Verantwortlichkeit für Altdaten während der Stilllegung, einschließlich des Konzepts der Aufrechterhaltung der Datenintegrität für die Zukunft. Darüber hinaus sollten auch die Datenmigrationsphase und seine Phase berücksichtigt werden.

1.1. Problembeschreibung

Legacy-Systeme sind soziotechnische Systeme innerhalb einer Organisation, die die Mensch-Organisation-Technik Relation umfassen (Matthiesen & Bjørn, 2015). Diese Systeme bestehen oft schon seit langer Zeit und basieren auf veralteten Technologien, so dass sie ständig gewartet werden müssen (Matthiesen & Bjørn, 2015). Legacy-Systeme führen kritische Geschäftsfunktionen aus, die für eine Organisation wesentlich oder nützlich sind, was bedeutet, dass sie oft eine lange Lebensdauer haben, da sie zu teuer oder kritisch sind, um ersetzt zu werden (Matthiesen & Bjørn, 2015).

Die Anforderung, ein Legacy-System zu ersetzen, kann aus einer Reihe von Gründen entstehen, wie in der Studie von Martens (2016) dargelegt. Martens (2016) listet die folgenden Gründe für Ablösen dar: Geringe Wartbarkeit, Hohe Wartungskosten, Aussterbende Technologien, neueste IT-Ansätze im Markt, veraltete Softwarearchitekturen und andere organisatorische Gründe. Darüber hinaus nimmt die Zahl der veralteten Systeme im IT-Bereich zu (Yang et al., 2005).

Weiterverfolgend ist das Reengineering von Legacy-Systemen in vielen Unternehmen zu einer wichtigen Aufgabe geworden. Es wurden Reengineering-Modelle entwickelt, wie z. B. die Butterfly Methodik von Wu et al. (1997) und das Dual-Spiral-Modell (Yang et al., 2005). Diese

Modelle haben sich mit einigen der Probleme im Reengineering-Prozess befasst, aber andere Probleme bleiben bestehen (Yang et al., 2005).

Beim Reengineering eines Systems gibt es wesentlich drei gängige Methoden: Redvelopment, bei der die bestehende Anwendung neu geschrieben wird; Wrapping, bei dem die Komponente eine neue Schnittstelle erhält, so dass sie für andere Softwarekomponenten leichter zugänglich wird; und Migration, bei der das Altsystem in eine flexiblere Umgebung verlagert wird, wobei die Daten und die Funktionalitäten des ursprünglichen Systems erhalten bleiben (Bisbal et al., 1999).

Bei der Redvelopment-Methode, muss das Legacy System entweder während der Entwicklung oder während der Umstellung auf das Ersatzsystem abgeschaltet werden (Bisbal et al., 1999). Der zweite Ansatz, das Wrapping, ist weniger drastisch, dient aber oft nur als kurzfristige Lösung. Der dritte Ansatz, die Migration, ist ein Mittelweg, der eine langfristige Lösung bietet und gleichzeitig das Risiko eines Systemstillstands mindert (Bisbal et al., 1999).

In der Praxis kommen diese Reengineering-Modelle vor, und die Datenmigrationsphase bleibt in vielen Reengineering-Projekten eine Herausforderung, wie die Studien von Lüssem und Harrach (2013), Book et al. (2013) und Bisbal et al. (1999) zeigen. Der Prozess des Speicherns, Abrufen, Überprüfen und Produzieren von Daten bei Bedarf kann viel Zeit und Geld kosten (Jha et al., 2017). Daten sind jedoch eine wesentliche Eigenschaft von Legacy-Systemen und den damit verbundenen Organisationen, so dass sich Organisationen den Herausforderungen der Datenmigration auseinandersetzen müssen (Book et al., 2013).

Diese Problematik wurde vom Unternehmenspartner voestalpine Stahl hervorgehoben, nämlich die Herausforderung, Daten aus Legacy-Systemen zu migrieren und diese herunterzufahren. Die IT-Architekten der voestalpine sind der Meinung, dass die abgelösten Legacy-Systeme nicht vollständig eingefroren und aus ihrer jeweiligen Systemlandschaft entfernt wurden, da ihre Daten für das Unternehmen unerlässlich sind. Und die Daten konnten aufgrund der Anzahl der Objekte und ihrer Komplexität nicht vollständig auf neue Lösungen migriert werden.

Alhassan et al. (2016) berichtet im Bereich der Daten-Governance, dass die Anzahl der in Organisationen verwendeten Daten in den letzten Jahren dramatisch zugenommen hat und wie diese eine entscheidende Rolle in Organisationen spielen. Insbesondere beeinflussen Daten operative und strategische Entscheidungen (Alhassan et al., 2016). Während des gesamten Stilllegungsprozesses spielen daher die Einhaltung der Informationen sowie deren Vollständigkeit und Integrität eine wichtige Rolle (Alhassan et al., 2016; Sneed, 1995). Dazu gehört das Konzept des Data Reengineering (Sneed, 1995). Die beiden Hauptziele des Data Reengineering bestehen darin, eine bestmögliche Zugänglichkeit der Daten zu ermöglichen und die Daten in einen besseren bzw. neuen Container zu verschieben (Sneed, 1995). In Bezug auf den hier untersuchten Kontext - einen Stilllegungsprozess - geht es um das Reengineering der Daten während der Migration und darum, wie die Daten langfristig zugänglich bleiben; mehr dazu wird in Kapitel 2.4. Data Governance erleuchtet, mit den Studien von Greenan et al. (2005) und Zitoun et al. (2021).

Neben der wirtschaftlichen Bedeutung der in einem System enthaltenen Daten ist ein weiterer grundlegender Faktor zu berücksichtigen: die Einhaltung der Compliance (Hempel, 2014).

Compliance bezeichnet die Verpflichtung von Unternehmen, die Einhaltung von Regeln, Kodizes und Grundsätzen durch geeignete Maßnahmen sicherzustellen (Hempel, 2014). In der IT-Branche spricht man von IT-Compliance und meint damit die Regelung der auf Informationstechnik beruhenden Abläufe (Hempel, 2014). Aspekte wie Betriebseffizienz, Compliance und langfristige Datenaufbewahrung stellen eine Reihe von Bedingungen an Archiv- und Compliance-Lösungen. Neben den juristischen Vorgaben zählen dazu auch technische Erfordernisse (Böcker, 2006).

In der Forschung von Lüssem und Harrach (2013) werden die Phase der Systemabschaltung und der Akt der Datenmigration als letzte Phase des Datenmigrationsprozesses angesehen. In dieser Phase müssen die Projekte das Einfrieren des Altsystems für die zukünftige Nutzung in Betracht ziehen. Das Altsystem sollte auch im "Ansichtsmodus" verfügbar gemacht werden und es sollte eine Archivierung historischer Daten erfolgen (Lüssem & Harrach, 2013). In der Forschung von Lüssem und Harrach (2013) erfolgt die Bereinigung von toten und ungenutzten Daten einen Schritt vor der Stilllegung.

Beim Austausch mit dem Partnerunternehmen wurde darauf hingewiesen, dass die oben genannten Phasen bei den Ablöseprojekten nicht berücksichtigt werden. Also der Stilllegungsprozess ist offensichtlich im ALM der voestalpine Stahl nicht berücksichtigt, d.h. es gibt kein definiertes Vorgehensmodell für die Stilllegung von Systemen und daher wissen die Ablöseprojekte nicht, wie sie damit einheitlich handhaben sollen. Daher befinden sich derzeit schon einige Inselformate in der Systemumgebung der voestalpine Stahl und die Fertigstellung von Projekten in den kommenden Jahren wird diesen Backlog erhöhen, wodurch die Systemlandschaft und deren Infrastruktur mit der Zeit an Komplexität gewinnen kann.

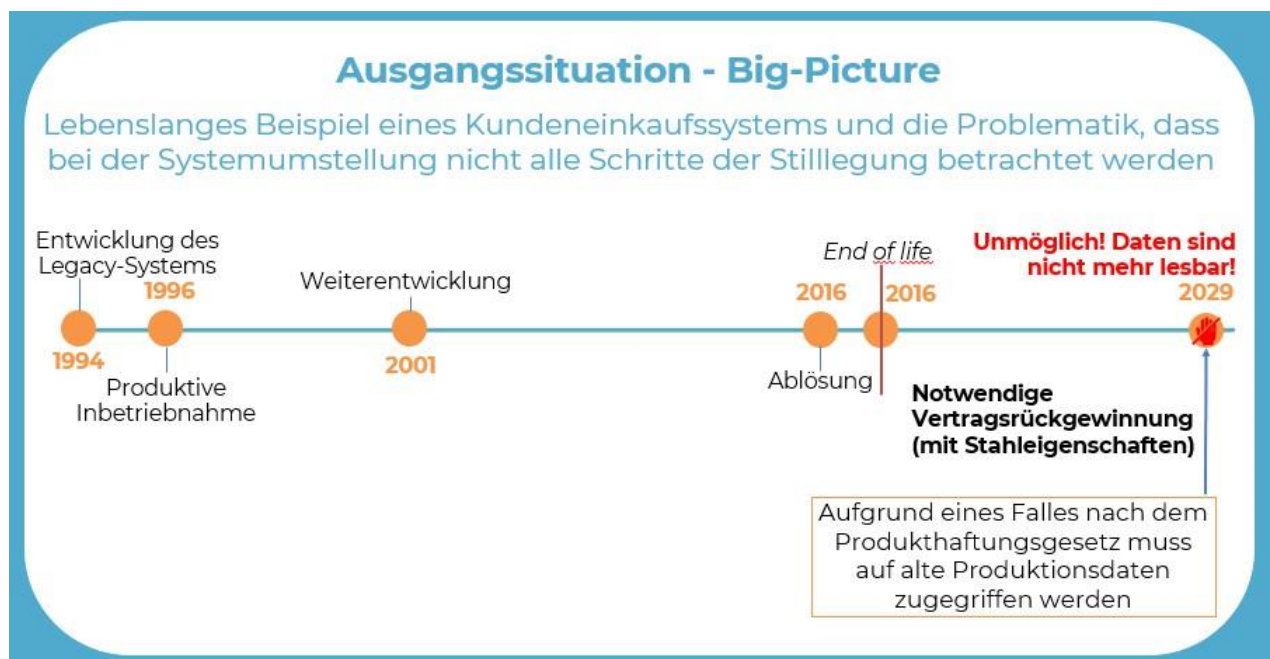


Abbildung 1 - Ausgangssituation (Big Picture)

Chappell (2010a) definiert Application Lifecycle Management (ALM) als eine Menge von Tätigkeiten, um eine bestimmte Applikation zu erstellen. Das Ziel des Application Lifecycle Managements ist jedoch, den gesamten Lebenszyklus einer Applikation bzw. eines Produkts

(z.B. eine Software) zu unterstützen (Lacheiner & Ramler, 2011). Der Lebenszyklus einer Applikation sollte dessen Konzeption bis zur Stilllegung erfassen (Tüzün et al., 2019). Laut Lacheiner und (Lacheiner & Ramler, 2011) wird damit eine ordnungsgemäße Überwachung, Verwaltung und Steuerung der Applikation gewährleistet. Somit sollte durch das ALM auch eine optimale Ablösung bzw. Stilllegung von Legacy Systemen ermöglicht werden.

Das Problem und Risiko besteht, dass die Beibehaltung dieser Inselsysteme langfristig zu Infrastruktur- und Compliance-Problemen führen kann, da auf die ersetzte Technologie aus vielen verschiedenen Gründen nicht mehr zugegriffen werden kann und ihre Daten daher nicht mehr lesbar sind, was für das Unternehmen ein Problem darstellt, wie bereits erwähnt. Wie in Abbildung 1 dargestellt, hat ein System beispielsweise eine bestimmte Lebensdauer. Nach Ablauf dieser Lebensdauer werden diese Technologien möglicherweise nicht mehr unterstützt, und wenn das Unternehmen in Zukunft auf Daten aus diesem System zugreifen möchte, besteht das Risiko, dass diese nicht mehr lesbar sind.

1.2. Problemnachweis

Wie in der Problembeschreibung (siehe 1.1 Problembeschreibung) erläutert wurde, sind viele Unternehmen im Laufe der Jahre datengesteuert geworden, und die digitale Transformation zwingt Unternehmen dazu, sich an neue Technologien anzupassen. Aufgrund dieser beiden Faktoren müssen Unternehmen ihre Daten optimal nutzen und speichern, um deren maximalen Wert zu erhalten und einen Wettbewerbsvorteil im Rahmen der Digitalisierung zu erzielen (Zitoun et al., 2021). Eines der größten Probleme ist jedoch nicht die Entwicklung neuer Systeme, die den neuen Anforderungen entsprechen, sondern die Anpassung bestehender Strukturen und Daten wie beispielsweise, die von Legacy-Systemen (Book et al., 2013).

Viele Studien befassen sich mit der technischen Seite der Stilllegung des Legacy-Systems, andere Modelle berichten über die Phasen der Datenmigration und wieder andere über den wirtschaftlichen und organisatorischen Kontext der Stilllegung eines Altsystems. Aber keine Studie berücksichtigt alle Faktoren eines Stilllegungsprozesses als Erweiterung des Lebenszyklusmanagements von Anwendungen.

Einige Reengineering-Modelle berücksichtigen die technische Abschaltung von Systemen, wie z.B. die Butterfly Methodik von Wu et al. (1997), die Renaissance Methodik von Battaglia et al. (1998) und das Dual-Spiral-Modell von Yang et al. (2005).

Die Butterfly-Methodik eliminiert während der Migration die Notwendigkeit für Systembenutzer, gleichzeitig sowohl auf das Alt- als auch auf das Zielsystem zuzugreifen und daher die Konsistenz zwischen diesen beiden Informationssystemen zu wahren (Wu et al., 1997). Dadurch wird das Legacy-System während der Ablöse vollständig aus der Systemumgebung entfernt (Wu et al., 1997).

Renaissance ist ein laufendes Forschungsprojekt, welches darauf abzielt, eine systematische Methode zur Systementwicklung und -umgestaltung zu entwickeln (Battaglia et al., 1998). Betrachtet jedoch nicht explizit den Stilllegungsprozess. Die Datenmigrationsphase wird jedoch betrachtet (Battaglia et al., 1998).

Das Dual-Spiral-Modell erfordert, dass die beiden Systeme (Legacy- und Zielsystem) zusammenarbeiten und die Funktionalität (nicht die Module) schrittweise vom Altsystem zum Zielsystem verschieben, wie im Spiralmodell (Yang et al., 2005). Ein letzter Schritt, der die Spirale schließt, ist die Abschaltung des betroffenen Altsystems (Yang et al., 2005).

Alle diese Modelle beinhalten jedoch das Verfahren zur Stilllegung des Systems innerhalb des Reengineering-Prozesses. Wissenschaftliche Untersuchungen zur Stilllegung von Systemen nach der Ablöse fehlen, jedoch können die oben genannten Modellansätze einen Ansatzpunkt für die Entwicklung des angestrebten Stilllegungsprozesses bieten.

In Bezug auf die Datenmigration stellen Lüssem und Harrach (2013) ein Modell vor, bei dem eine standardisierte Best-Practice-Methode für das Management komplexer Datenmigrationsaktivitäten beschrieben wird, die in das The Open Group Architecture Framework (TOGAF) eingebettet ist. Cloudstep von Bessera et al. (2012) und Cloud Migration Metamodel von Pamami et al. (2019) beschreiben detailliert Prozessmodelle für die Migration von Legacy-Daten zur Cloud-Lösungen.

Im Hinblick auf die Einhaltung von grundlegenden organisatorischen Anforderungen stellen Khanye et. al (2018), Book et. al (2013) und Bennet (1995) Konzepte und Ansätze vor, die sich mit den häufigsten Problemen befassen, die in der Praxis bei der Demontage von Altsystemen auftreten. Der Artikel "POTSHARDS" aus dem Jahr 2005 und der Artikel DMMM: Data Management Maturity Model stellen die Beziehung von Daten zur Organisation dar und zeigen, wie die Integrität und Qualität von Daten langfristig erhalten werden können (Greenan et al., 2005; Zitoun et al., 2021). All dies wird im nächsten Kapitel (siehe 2. Aktueller Stand der Forschung) ausführlich beschrieben.

1.3. Forschungsziel

Ziel dieser Forschung ist es, ein Vorgehensmodell für die Stilllegung von abgelösten Legacy-Systemen zu erstellen. Dazu gehören ERP-Systeme, Websysteme und Archivsysteme (BI-Systeme). Nach diesem Konzept müssen die wesentlichen Themen rund um die Stilllegung eines Systems berücksichtigt werden.