

# EWIGE JUGEND FÜR IT-SYSTEME

---

Evolutionäre Architektur: die Chancen  
des technologischen Wandels nutzen

**Autorenteam**

Christian Böhning  
Dr. Carsten Wedekind  
Dr. Philipp Kleine Jäger  
Francisca Di Piazza  
Justus Dircks  
Karsten Trostmann

## 1 Einleitung

Die Fähigkeit, den fortschreitenden technologischen Wandel zu gestalten, hat sich in sämtlichen Branchen und Geschäftsbereichen zu einem essenziellen Bestandteil der Wettbewerbsfähigkeit entwickelt, welcher über wirtschaftlichen Erfolg und Misserfolg entscheidet.

Dabei spielt die Aktualität und Adaptierbarkeit der IT-Infrastruktur und der darin integrierten Anwendungssysteme eine entscheidende Rolle. In der Realität weisen IT-Systeme ein Phänomen auf, das dem menschlichen Alterungsprozess ähnelt. Sie werden mit fortschreitendem Lebensalter schwerfälliger und verlieren die Fähigkeit, sich schnell genug an geänderte Marktbedingungen anzupassen. Ihre Leistungs- und Zuverlässigkeit nimmt bei gleichzeitigem Anstieg der Kosten für Wartung und laufenden Betrieb ab.

Das Altern kann prinzipiell nicht verhindert, sondern nur verzögert und in seinen Auswirkungen begrenzt werden. Der Alterungsprozess von IT-Systemen ist kein neues Phänomen. Seine Bedeutung hat aber in den letzten Jahren stark zugenommen, weil technologische Innovationszyklen immer kürzer werden und IT-Systeme mit fortschreitender Digitalisierung der Prozesse in den Unternehmen eine wesentlich stärkere Rolle in der Wertschöpfung spielen. Um die Auswirkungen des Alterungsprozesses von IT-Systemen zu begrenzen, müssen dessen Ursachen verstanden und rechtzeitig geeignete Gegenmaßnahmen eingeleitet werden.

IT-Systeme unterliegen im Zeitverlauf einer natürlichen Degeneration

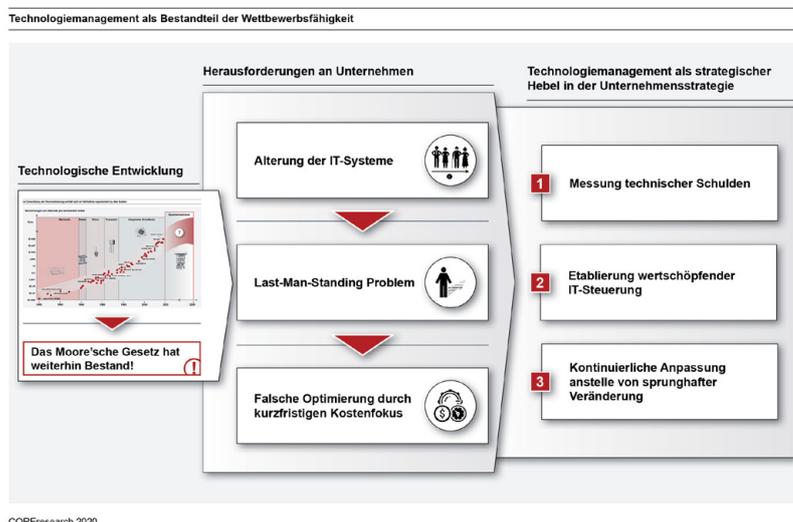


Abbildung 1: Technologiemanagement als Bestandteil der Wettbewerbsfähigkeit

Die heutigen Herausforderungen sind durch eher planwirtschaftlich geprägte Ansätze des klassischen Enterprise Architekturmanagements mit Bebauungsplänen und Zeithorizonten von 5-10 Jahren nicht mehr zu bewältigen. Es gilt vielmehr, das Technologiemanagement von IT-Systemen als einen kontinuierlichen und evolutionären Architekturprozess zu etablieren, der in enger Abstimmung mit den Geschäftsbereichen zu durchlaufen ist. Dafür müssen die Defizite der aktuellen IT-Systeme in Relation zu den am Markt etablierten Qualitäts- und Produktivitätsstandards permanent qualitativ und quantitativ erfasst werden. Um die identifizierten Defizite in

Technologiemanagement als integrierter und kontinuierlicher Architekturprozess

---

der existierenden IT-Landschaft abzubauen, ist die Harmonisierung der Steuerungsinstrumente zwischen der Geschäfts- und IT-Welt eine weitere wesentliche Voraussetzung. Das in den meisten Unternehmen weit verbreitete Prinzip einer projektbezogenen Budget- und Portfoliosteuerung ist meist auf den unmittelbaren fachlichen Geschäftsnutzen orientiert. Die daraus resultierenden IT-Lösungen sind deshalb auf eine lokale und kurzfristige Optimierung ausgerichtet und führen zu einer stetig wachsenden technischen Hypothek in den betroffenen IT-Systemen. Einige große Software-Konzerne haben, um diese Mechanik zu durchbrechen, gemeinsame Produkt-, Entwicklungs- und Betriebsorganisationen etabliert, wodurch Ende-zu-Ende Verantwortung und gemeinsame GuV die negativen Effekte abmildern.

Im Folgenden bieten wir einen Einblick über die Entwicklung vergangener technologischer Ären und entwickeln einen Vorschlag, wie Unternehmen durch nachhaltiges IT-Management den Alterungsprozess von IT-Systemen mildern und gleichzeitig Potenziale neuester Technologien auf operativer, organisatorischer und strategischer Ebene bestmöglich für die Stärkung der eigenen Innovationsfähigkeit nutzen können.

## 2 Warum altern IT-Systeme?

### 2.1 Systementwürfe unterliegen den Rahmenbedingungen ihrer Ära

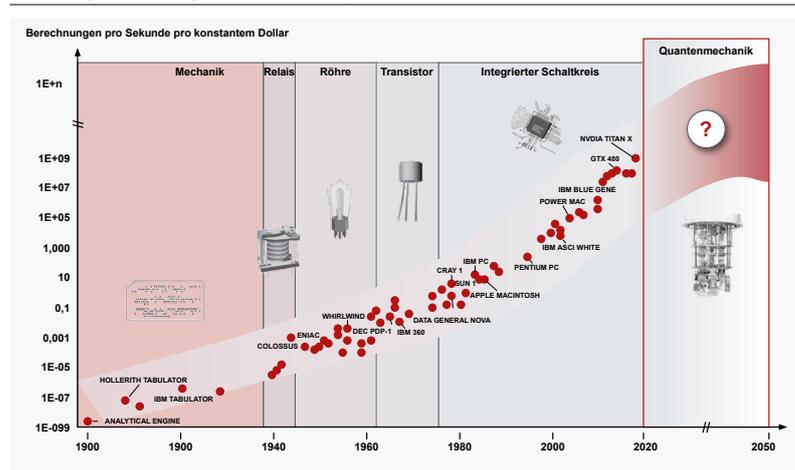
Schon in den 90er Jahren des letzten Jahrhunderts hat sich die Computerwissenschaft mit dem Prozess der Alterung von IT-Systemen und Software befasst. Kein geringerer als David L. Parnas, dem wir unter anderem das für jedes nichttriviale System fundamental wichtige Kapselungs- und Geheimnisprinzip verdanken, hat in seinem Essay „Software Aging“ die zwei wesentlichen Ursachen dargelegt.

Die erste Ursache besteht darin, dass Systeme nach Entwurfparadigmen entworfen werden, die an den technischen, wirtschaftlichen und organisatorischen Rahmenbedingungen ihrer Zeitepoche ausgerichtet sind. Diese Rahmenbedingungen ändern sich jedoch permanent und diese Änderung wird maßgeblich durch die Technologieentwicklung bestimmt. Damit sind nicht nur die einzelnen technischen Innovationen an sich, sondern auch daran angepasste Methoden und Vorgehensweisen verbunden.

Konzeption von IT-Systemen basiert auf Entwurfparadigmen einzelner Zeitepochen

In den 80er und 90er Jahren des letzten Jahrhunderts waren Rechenleistung und Speicherplatz teuer. Daraus abgeleitete Entwurfparadigmen waren das Konzept des Zentralrechners, die maximale Wiederverwendung von Code und Sparsamkeit als oberstes Prinzip in der Datenmodellierung. Unter den heutigen Rahmenbedingungen sind diese Paradigmen höchst problematische Designs unter deren späten Auswirkungen viele Unternehmen leiden. Was damals als Lösungsmuster gut und richtig war, ist unter den heutigen Anforderungen an die Flexibilität von IT-Systemen ein Anti-Pattern.

Die Entwicklung der Rechenleistung verhält sich im Verhältnis exponentiell zu den Kosten



Quellen: Ray Kurzweil, Steve Jurvetson, computerworld.ch

Abbildung 2: Das Moore'sche Gesetz im Kontext der Historie der Entwicklung der Kosten von Rechenkapazitäten

---

Betrachten wir die Entwicklung technischer Rahmenbedingungen am Beispiel der historischen Entwicklung der Rechenleistung, dann sehen wir, dass die Technologieentwicklung ein evolutionäres Verhalten aufweist. Wie bei allen Evolutionsprozessen verläuft diese Entwicklung nicht linear, sondern exponentiell. In der aktuellen Lage des Jahres 2021 ist uns allen sehr gegenwärtig, was exponentielles Wachstum bedeutet. Die technischen Innovationszyklen werden immer schneller, kürzer und in gleichem Maße ändern sich auch Rahmenbedingungen und Entwurfsparadigmen. Die IT-Systeme veralten durch sich stetig beschleunigende Technologieentwicklung also immer schneller.

---

Exponentieller Technologiefortschritt ohne Systemweiterentwicklung führt zu Alterung

---

## 2.2 Mit neuen Technologien entstehen neue Prozesse, Methoden und Tools

Um das Innovationspotenzial neuer Technologien nutzen zu können, müssen Arbeitsabläufe und Methoden angepasst werden, damit die neuen Möglichkeiten auch effektiv zur Steigerung der Qualität und Produktivität genutzt werden können. Dies gilt nicht nur für die durch IT-Innovationen digitalisierten Geschäftsprozesse, sondern auch und gerade für die Elemente der IT-Wertschöpfungskette selbst. Aktuelle Paradigmen einer auf Agilität und Effizienz ausgerichteten Softwareentwicklung, wie zum Beispiel:

---

IT-Wertschöpfungskette und Arbeitsabläufe müssen für Nutzung von Innovationspotenzialen modernisiert werden

---

- › Continuous Integration / Continuous Delivery
- › Test Driven Development
- › Infrastructure as Code

sind auf den Technologien des Mainframezeitalters nicht oder nur sehr eingeschränkt anwendbar. So wird nicht ohne Grund die Komplexität veralteter IT-Systeme in Form von Transparenzverlust und damit verbundener Testaufwände als zweithäufigster Hinderungsgrund für Änderungen in der Fachliteratur genannt. Die Umsetzung einer Fachanforderung innerhalb von 2-3 Wochen in hoher Qualität ist heute durch eine konsequente Orientierung auf Softwarequalität im Entwicklungsprozess und eine weitestgehende Testautomatisierung möglich. Das setzt zunächst die Bereitschaft der Softwareentwickler voraus, Methoden und Prozesse mit dem Ziel einer Qualitätsverbesserung weiterzuentwickeln. Es bedarf aber auch der Unterstützung durch die genutzte Technologieplattform. Auch wenn sich die Lage dank besseren Toolsupports auf der COBOL/Mainframe-Plattform verbessert hat, sind aktuelle Methoden der Softwareentwicklung auf dieser Plattform eher noch Ausnahme als verbindlicher Standard.

## 2.3 Marktanforderungen sind nur bis zur Technologiebarriere umsetzbar

Auch gut gewartete Systeme sind ab einem bestimmten Zeitpunkt nicht mehr in der Lage, Anforderungen wettbewerbsfähig abzubilden. Der berühmte Blick in die Glaskugel, um kommende Technologie- und Paradigmenwechsel schon beim Systemdesign zu erkennen, ist leider nur begrenzt möglich. Die IBM Designer sind bei der Konzeption des Systems 360 in den 60er Jahren nicht von der Möglichkeit eines Online-Systems ausgegangen. Bill Gates wird wohl fälschlicherweise der Satz „640K ought to be enough for anybody“ im Zusammenhang mit der Speicherverwaltung des Betriebssystem MS-DOS zugeschrieben. Heute verwaltet jedes Smartphone 100-mal mehr Arbeitsspeicher. Wenn eine Technologie den Kinderschuhen entwächst und erste Anwendungen damit erstellt werden,

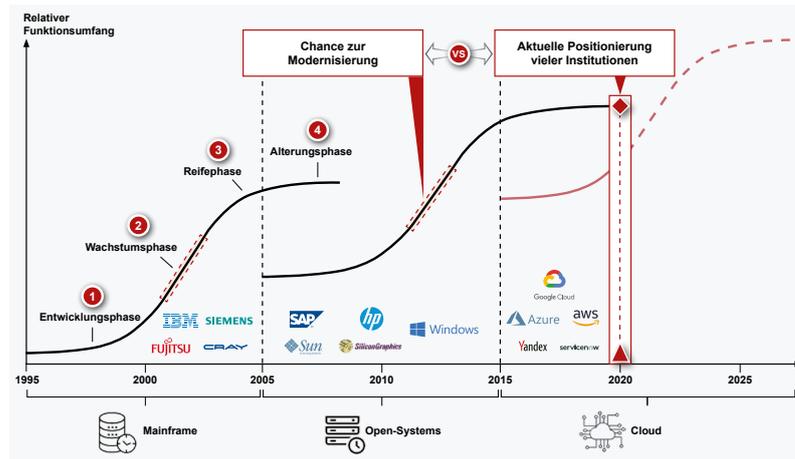
---

Neue Technologien beinhalten natürliche Technologiebarrieren

---

entsteht automatisch eine Technologiebarriere, weil die Vorgängertechnologie in einer der Dimensionen Funktion, Qualität, Geschwindigkeit und Kosten unterlegen ist. So konnte z.B. Online Banking Ende der 1990er nicht mehr effizient auf der Mainframeplattform implementiert werden, weil wesentliche Komponenten wie z.B. Webserver oder Frameworks zur Webentwicklung auf dieser Plattform nicht zur Verfügung standen.

Zyklus der Technologieentwicklung: echte Innovation setzt Wechsel der Technologiebasis voraus



Quelle: COREresearch 2020

Abbildung 3: Zyklus der Technologieentwicklung

## 2.4 Arbeits- und Wertschöpfungsmodelle der IT ändern sich

Ob eine Technologieplattform sich evolutionär an wechselnde Anforderungen anpassen kann, hängt maßgeblich von ihrem Ökosystem ab. Ist das System durch eine restriktive Lizenzpolitik oder hohe systemseitige Einstiegsbarrieren geschlossen, wird das Ökosystem von wenigen sehr dominanten Playern bestimmt und es findet wenig Wettbewerb statt. Unter diesen Bedingungen kann die Evolution einer Technologieplattform nur begrenzt stattfinden.

Die zur Erstellung komplexer Softwareumgebungen notwendigen Frameworks sind heute überwiegend als Open Source verfügbar und werden von einer großen Community weiterentwickelt. Der Grund hierfür ist mitnichten ein Utopie-getriebenes Mäzenatentum, sondern die Erkenntnis, dass nur in einem offenen Ökosystem Evolution stattfinden kann. Zudem werden die Entwicklungskosten nicht-marktdifferenzierender Komponenten sozialisiert. Alle Nutzer der Software haben die Möglichkeit, direkt an der Verbesserung mitzuwirken und neue Ideen einzubringen. Die Software und Toollandschaft kann von einer breiten Benutzerbasis teilweise schneller weiterentwickelt werden und ein höheres Qualitätslevel erreichen.

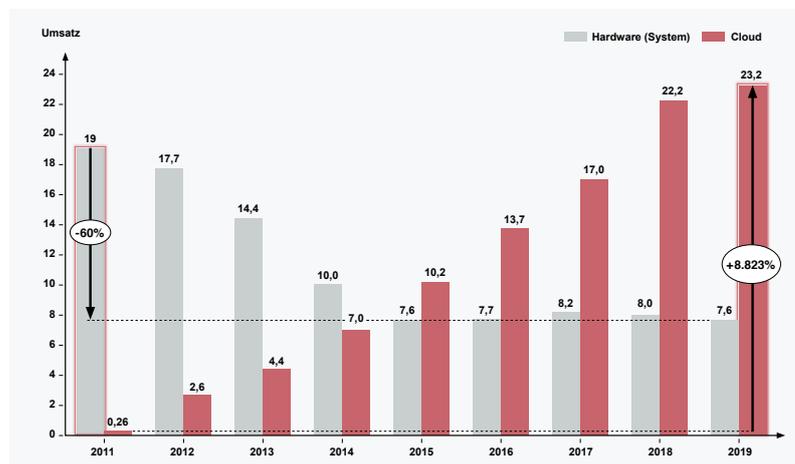
Die IT-Wertschöpfungskette ist deshalb schon seit einiger Zeit im Umbruch. Das über Jahrzehnte etablierte Erfolgsmodell stabiler Umsätze aus Hardware und Softwarelizenzen ist inzwischen obsolet. Google, Apple und Amazon aber auch SAP und IBM setzen bei ihren Geschäftsstrategien nicht mehr primär auf Hardware- und Lizenzerlöse, sondern auf Cloud- und Software as a Service-Modelle. Der Zugang zu Hardwareressourcen

Innovation und Evolution finden nun in offenen Ökosystemen statt

Cloud und SaaS-Modelle machen Verkauf von Hardware und Softwarelizenzen obsolet

ist dank Cloudtechnologien vollständig demokratisiert. Programmcode ist oft kein Betriebsgeheimnis mehr. Das Modell geschlossener exklusiver Ökosysteme ist damit veraltet, weil es oft kaum Wettbewerbsvorteile mehr bietet.

IBM Umsatz von 2011 bis 2019, nach Geschäftssegmenten (in USD Mrd.)



Quellen: IBM Geschäftsberichte 2011 - 2019

Abbildung 4: Entwicklung von Hardware und Cloud Umsätzen bei IBM von 2011 bis 2019

## 2.5 Organisationen ändern sich langsamer als Technologie

Technologiemanagement beschäftigt sich mit der Herausforderung, an der oft exponentiellen technologischen Entwicklung zu partizipieren. Erweiterbare und austauschbare, flexible IT-Systeme sind hierfür nur eine Voraussetzung. Insbesondere die Umsetzung der notwendigen strukturellen organisatorischen Veränderungen, um neue technologische Möglichkeiten effektiv nutzen zu können, ist eine wesentliche Herausforderung, da Restrukturierungen immer organisatorische Beharrungskräfte überwinden müssen. Grundsätzlich ist zu konstatieren, dass die immer kurzzyklischere Technologieentwicklung viele Organisationen – bzw. die darin interagierenden Menschen – überfordert und die notwendigen Änderungen nur langsam adoptiert werden.

Organisatorische Veränderung als Voraussetzung für Nutzung technologischer Möglichkeiten

### 2.5.1 Historisch gewachsen

Die meisten der heutigen Organisationsstrukturen in fast allen Bereichen der Wirtschaft sind funktional ausgerichtet – die Fähigkeiten und Wissen der Mitarbeiter werden nach Kategorien gebündelt und in verschiedene funktionale Geschäftseinheiten („Silos“) aufgeteilt. Die Geschäftsprozesse und Kommunikationswege sind ebenso entsprechend dieser funktionalen Aufteilung ausgerichtet. Diese Organisationsmodelle sind historisch gewachsen und werden von noch den im Zeitalter der Industrialisierung geprägten Regeln auch heute geführt.

Organisationsstrukturen sind auf funktionalen Nutzen ausgerichtet

Beispiele solcher Organisationsformen reichen von der funktionalen Organisation von Bahnunternehmen des 19. Jahrhunderts, über Stablinien- und Spartenorganisationen bis zu der Matrixorganisation von Luft- und Raumfahrt-Unternehmen der Nachkriegszeit.

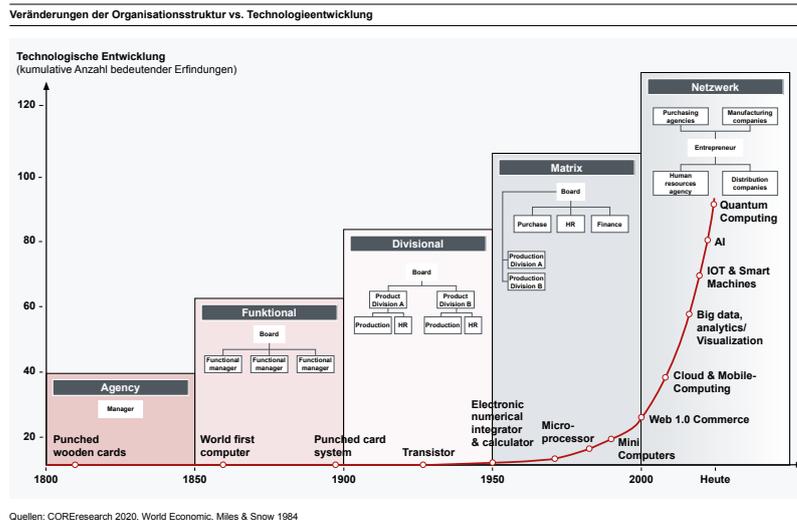


Abbildung 5: Veränderungen der Organisationsstruktur vs. Technologieentwicklung

Lange Zeit waren diese Organisationsformen auch sinnvoll, denn die dahinterstehenden Organisationsmodelle waren darauf ausgerichtet, Ressourcen verschiedenen funktionalen Geschäftseinheiten zuzuteilen und dadurch verlässlichen Output basierend auf sich ständig wiederkehrenden Prozessen zu produzieren. Ähnlich wie bei der Technologiebasis, haben diese Organisationsformen effektiv die Anforderungen „ihrer Ära“ abgedeckt.

Über die Silo-Organisation hinaus wurden während der letzten Jahre speziell in Großunternehmen viele Fachbereiche darauf ausgerichtet, wie ein Profitcenter zu agieren. Dies verursacht Kämpfe um Ressourcen, insbesondere um das Budget. Das Silo- und Konkurrenzdenken der Fachbereiche sowie die bereits in diesem Zusammenhang erprobten Führungs-, Organisations- und Prozessstrukturen stehen den notwendigen Change-Maßnahmen diametral entgegen.

### 2.5.2 IT-Systeme und geringe Akzeptanz der Änderungsmaßnahmen bei den Mitarbeitern

Darüber hinaus besteht bei der Notwendigkeit eines Austausches der Technologiebasis das Problem von fehlenden Fähigkeiten bzgl. neuer Technologien, Vorgehens- und Zusammenarbeitsmodellen und flexiblerer Governance. Dabei spielen insbesondere die folgenden Faktoren eine übergeordnete Rolle:

- Die Geschwindigkeit des technologischen Wandels erfordert eine ständige Erneuerung der Fähigkeiten der Mitarbeiter. Laut Einschätzung des Chefs der US-Telekomfirma AT&T, werden Kompetenzen von 40% der Mitarbeiter nach fünf Jahren völlig veraltet sein.
- Die Mitarbeiter halten sich an alte Technologien, Methoden und Abläufe, mit denen sie sich gut auskennen, und sind eher begrenzt motiviert, sich neue Technologien und Methoden anzueignen. Diese historisch gewachsenen Prozesse, Methoden und Technologien geben einem das Gefühl der Sicherheit und Stabilität.

Silo-Organisationen agieren entgegen notwendigen Veränderungen

Technologischer Wandel setzt Anpassungsmöglichkeiten der technischen Fähigkeiten voraus

- Weiterbildungskapazitäten sind begrenzt.
- In Folge dieses Defizits an Fähigkeiten entsteht Angst vor Bedeutungsverlust bis hin zur Sorge um den eigenen Arbeitsplatz, und damit wirken die organisatorischen Beharrungskräfte dem Erneuerungsprozess regelmäßig entgegen.

Der Aufbruch veralteter Organisationsstrukturen stellt eine signifikante Herausforderung für das Technologiemanagement dar.

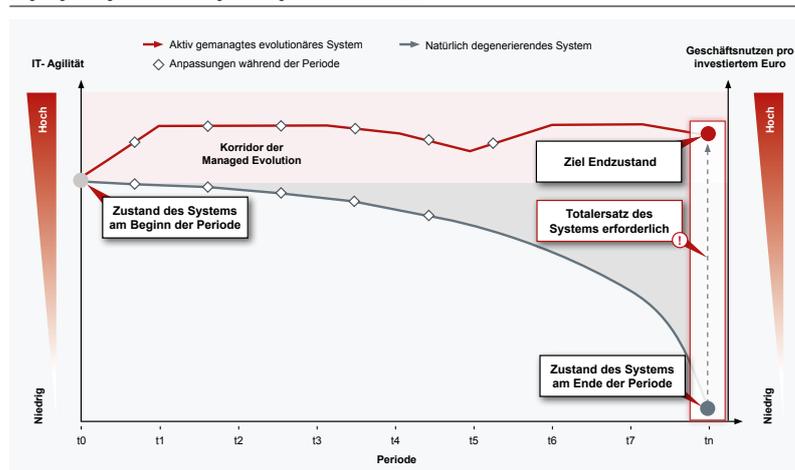
### 3 Welche Probleme entstehen durch veraltete IT-Systeme?

#### 3.1 Alterung bedeutet schrittweisen Verlust konzeptioneller Integrität und erhöht den Abhängigkeitsgrad (technisch und fachlich)

Eine der Ursachen für das Veralten von IT-Systemen liegt in der Änderung und Anpassung der IT-Systeme selbst. Für den überwiegenden Teil der in den Unternehmen eingesetzten IT-Systeme wurden und werden umfangreiche Konzepte erstellt, die einer bestimmten fachlichen Zielsetzung und technischen Paradigmen folgten. Zum Zeitpunkt ihrer ersten Inbetriebsetzung entsprechen diese Systeme auch dieser Konzeption, weisen also eine hohe konzeptionelle Integrität auf und sind damit frei von strukturellen technischen Schulden. Der einsetzende Änderungsprozess nimmt dann meist einen natürlichen, mehr oder weniger starken entropischen Verlauf: das System weicht mit zunehmender Zahl an Änderungen und implementierten Funktionen immer mehr vom ursprünglichen Konzept ab. Mit zunehmendem Alter des Systems beschleunigt sich die konzeptionelle Erosion und führt zu einer immer schlechteren Anpassbarkeit und Systemqualität.

Systemqualität und Anpassbarkeit nimmt häufig mit Anzahl von Änderungen ab

Langfristige und signifikante Auswirkungen auf IT-Agilität und Geschäftsnutzen



Quelle: COREresearch 2020

Abbildung 6: Langfristige und signifikante Auswirkungen auf IT-Agilität und Geschäftsnutzen

Wir wollen aus unserer Sicht einige Hintergründe darlegen.

Der Verlust konzeptioneller Integrität wird in vielen Fällen durch ein auf ausschließlich funktionale Weiterentwicklungen ausgerichtetes Änderungsmanagement verursacht. Änderungen am IT-System sollen opportunistisch mit minimalem Kosten- und Zeitaufwand umgesetzt werden. Die konzeptionelle Integrität des Systems stellt für den Nutzer keinen Wert dar, ist deshalb kein Ziel und wird nicht gesteuert. Größere Änderungen in monolithischen Systemen mit großen Funktionsbereichen und wechselnden Dienstleistern führen dann zu regelrechten Erosionsschüben.

Falsch priorisiertes Änderungsmanagement reduziert konzeptionelle Integrität

---

IT-Systeme werden in größeren Projekten und in Teilen durch externe Dienstleister umgesetzt. Nach der ersten Inbetriebsetzung wird die Software des Systems häufig an ein Wartungsteam übergeben. Meist werden die Mitglieder des Wartungsteams erst in späten Projektphasen involviert oder sind nicht Teil des Projektteams. Sie müssen sich dann auf Grundlage einer oft unzureichenden Dokumentation in die Konzepte einarbeiten und mit einer großen Codebasis vertraut machen. Da selbst bei sorgfältigster Dokumentation Transitionsverluste nicht zu vermeiden sind und eine „emotionale Verbundenheit“ mit dem Code (noch) fehlt, erfolgen Änderungen pragmatisch in Unkenntnis oder Ignoranz des ursprünglichen Konzepts.

Ein weiterer, aber sehr entscheidender, Punkt ist, dass mit der abnehmenden konzeptionellen Integrität ein Wertverlust des IT-Systems einhergeht, der nirgendwo erfasst wird, aber wie eine unsichtbare und stetig wachsende Hypothek auf dem System lastet. Nur in den wenigsten Fällen werden technische Schulden systematisch erfasst und sind den Unternehmen bekannt. Das Problem ist also nicht oder in der Größe nicht vollständig bekannt.

---

Mangelnde Erfassung technischer Schulden zieht Unkenntnis über tatsächlichen Zustand der Systeme nach sich

---

### 3.2 Last-Man-Standing

In den meisten Unternehmen sind IT-Systeme und Anwendungen über Jahre gewachsen. Sie sind für verschiedene Plattformen optimiert, teilweise in anderen Sprachen programmiert und miteinander verbunden. Diese alten IT-Systeme werden von verschiedenen Geschäftseinheiten bzw. Abteilungen innerhalb des Unternehmens gemeinsam genutzt - sie teilen sich auch die Kosten für den Betrieb der Systeme und Plattformen sowie damit einhergehend der Prozess- und Organisationslandschaft - sodass sie aufgrund der gleichmäßigen Verteilung der Kosten auf die Teilnehmer bis zu einem gewissen Grad erschwinglich erscheinen. Abgesehen von Kosteneinsparungsaspekten geht es oft auch darum, die Wiederverwendungsmöglichkeiten und die volle Leistungsfähigkeit der aktuellen Systeme voll auszuschöpfen, weshalb Unternehmen oft mehr als 70-80% ihres IT-Budgets in die Aufrechterhaltung des Status quo ihrer IT-Systeme investieren. Weitere Gründe für die Beibehaltung der alten Systeme sind:

---

Kosteneinsparungen und Wiederverwendbarkeit alter IT-Systeme beanspruchen oft Mehrheit des IT-Budgets

---

- › Das System funktioniert, weswegen keine Notwendigkeit besteht, es zu ersetzen – nach dem Motto if it ain't broke, don't fix it.
- › Das System ist sehr komplex und die Dokumentation oft nicht optimal, sodass eine einfache Definition des Anwendungsbereichs schwierig ist.
- › Ein Re-Design ist aufgrund einer in sich geschlossenen und umfangreichen Architektur äußerst kostspielig.
- › Aufgrund fehlender Kapazitäten (Personal, Budget, etc.) wird eine Neuanschaffung verschoben.

Aufgrund der dynamischen Veränderungen in der IT und des allgemeinen Marktdrucks ist jedoch ein Wechsel zu einem neueren und besseren System nicht nur erforderlich, sondern letztendlich auch unvermeidlich.

Daher müssen Nutzen und Geschäftswert einer Umstellung auf ein neues System ganzheitlich durchdacht und umfassend kalkuliert sein. Eine Kosten-Nutzen-Analyse (oder Vollkostenanalyse) dient als Grundlage zur Untermauerung des Business Cases. Dazu gehört auch die Berücksichtigung von Compliance-, Datenintegrations- und Sicherheitsfaktoren sowie die Klärung, ob und mit welchem Aufwand geplante Innovationen und notwendige Maßnahmen umgesetzt werden können. Neben den tatsächlichen Betriebskosten wird bei der Berechnung auch berücksichtigt, welche Aufwendungen oder Schäden alte IT-Systeme für die interne oder externe Reputation verursachen und inwieweit sie die Marktchancen einschränken.

Oft sind bereits das Fehlen einer Vollkostenrechnung und die mangelnde Modularität die ersten beiden großen Hürden, da die tatsächlichen Systemkosten nicht transparent sind und der Eindruck entsteht, selbst begrenzte Erneuerungen kommen de facto einer Gesamttransformation gleich.

Wenn für die meisten Teilnehmer (Geschäftseinheiten) der Nutzen/Geschäftswert eines Wechsels auf ein Neusystem den eines Verbleibs auf den Altsystemen überwiegt, werden sie irgendwann wechseln. Die Abteilung, die nicht wechselt und beim alten System verbleibt, wird die vollen Kosten für den Betrieb des Systems tragen müssen, die früher von mehreren Teilnehmern geteilt wurden, was zu einem sogenannten „Last-Man-Standing“-Problem führt.

Das „Last-Man-Standing“-Problem ist daher kein technisches Problem, sondern vielmehr eine Frage der ordnungsgemäßen und rechtzeitigen Bewertung der mit dem Wechsel verbundenen Kosten und Vorteile sowie einer Betrachtung des Systemlifecycle, welche auch wirtschaftliche Überlegungen (entgangener/verlorener Umsatz etc.) berücksichtigt. Es geht also darum, den richtigen Zeitpunkt für den Austausch alternder Systeme zu erkennen, um zu vermeiden, dass die vollen Kosten für den Betrieb eines Altsystems als verbleibende Abteilung/Geschäftseinheit getragen werden müssen.

### 3.3 Nicht-Berücksichtigung der Hypothek führt zu falschen Optimierungen

Ist der Zeitpunkt des optimalen Wechsels erst einmal verpasst, besteht die Gefahr in einen sich selbst verstärkenden Abwärtsstrudel zu geraten: das vorhandene Personal und die benötigten Technologie-Kompetenzen werden auf dem Arbeitsmarkt weniger relevant, was absurderweise zu einer Verteuerung der Ressourcen führt, wie man bspw. an Spezialdienstleistern für COBOL oder Fortran Programmierung erkennen kann. Auch besteht bei veralteten Systemen die Gefahr, durch End-of-Life Zyklen der Hard- und Software irgendwann signifikant selbst in die Weiterentwicklung und Absicherung der rudimentären Betriebsfähigkeit investieren zu müssen.

---

Abteilungen, die den gemeinsamen Wechsel auf neue Systeme verpassen, sollten hohe Betriebskosten allein tragen (Verursacherprinzip)

---

---

Personal und Kompetenzen für alte Technologien sind teuer und rar

---

---

**COBOL-Kernanwendungen der Finanzindustrie haben ein Nachwuchsproblem**


---

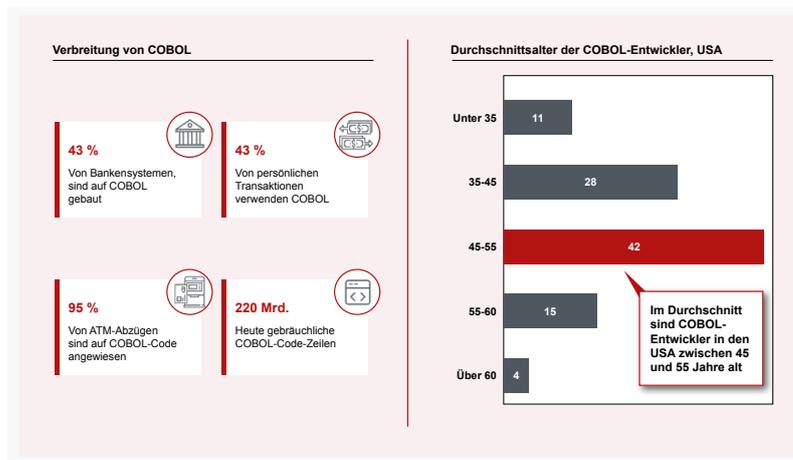


Abbildung 7: Nach wie vor hohe Verbreitung der technisch veralteten und insgesamt unüblichen Programmiersprache COBOL im Finanzsektor

Die Nicht-Berücksichtigung technischer Schulden bei Strategie- und Architekturstreitigkeiten führt zu einer verzerrten Betrachtung der Total Cost of Ownership für Weiterbetrieb oder Austausch, da ein veraltetes System oft unerwartet überproportional viele Ressourcen beansprucht – ein Effekt, der durch ein hohes Run-the-Company/Change-the-Company Verhältnis messbar ist. Diese unvollständige zeitliche Betrachtung mit stark überdurchschnittlichen Kosten gegen Ende der Lebensdauer wird häufig durch lokale Optimierungsentscheidungen zusätzlich verschlimmert. Geschäftsdomänen mit lediglich langzyklischem Veränderungsbedarf können so mitunter zu Treibankern einer gesamten Architektur und Organisation werden.

Selbst gut konzipierte Systeme und Architekturstreitigkeiten leiden unter diesem Effekt, da die Betriebskosten immer im Verhältnis zur von den Wettbewerbern eingesetzten Technologie betrachtet werden müssen. Gerade in der IT mit ihrer exponentiellen Technologieentwicklung (siehe das Moore'sche Gesetz in Abbildung 2) können mehrere Jahre technologischer Altersdifferenz bereits signifikante Effizienzunterschiede bedeuten. Da die Softwarebranche immer mehr zu einem Ökosystem zusammenwächst und moderne bzw. effiziente Technologien und Frameworks aktuell im Jahre 2021 hohe Interdependenzen untereinander aufweisen, verschlimmern Netzwerkeffekte diese Abhängigkeit häufig: Service Meshes, moderne API Gateways, Container und Software zur Infrastrukturverwaltung und -automatisierung sind ohne Kubernetes oft nicht einsetzbar.

Aufgrund der Verschränkung vieler Legacyarchitekturen wird die Entscheidung zum Wechsel oft bis zum letztmöglichen Zeitpunkt hinausgezögert – mit dem Effekt, dass aufgrund des Wettbewerbsdrucks keine Zeit für eine saubere, umsichtige und langfristige Neuplanung und Transition fehlt – und durch den Invest in obsolete Systeme die Chance vertan wird, in moderne Technologie zu investieren. Dadurch leidet die neue technische Plattform von Beginn an oft an Geburtsfehlern.

---

Oft mangelhafte Entscheidungsbasis für CapEx/OpEx Expenditure anstatt ganzheitlicher Total-Cost-of-Ownership-Betrachtung

---



---

Verzögerter Wechsel verschlechtert Wettbewerbssituation und Entscheidungsoptionen

---

---

### 3.4 Die Problematik der Steuerung nach Kosten

Historisch ist die IT in vielen Organisationen, insbesondere in der Finanzbranche, als internes Cost Center mit eigenen Strukturen aufgestellt. Daher werden IT-Investitionen vor allem nach Kostengesichtspunkten begutachtet. Die oft stattfindende Sozialisierung der Kosten aufgrund fehlender Möglichkeit der aufwandsgerechten Zuordnung der Kosten bestärkt diesen Mechanismus zusätzlich.

Demgegenüber stehen alternative Organisationsformen, deren IT in die jeweiligen Profit Center integriert ist. Selbst, wenn in solchen Organisationen noch zusätzlich ein zentraler Servicebereich existiert, hat diese Aufstellung signifikante Vorteile, da die Flexibilität für Investments höher ist und die Regelprozesse zur Gesamtoptimierung oft in viel geringerem Maße notwendig sind.

Die Notwendigkeit, für eine an der Strategie ausgerichtete Organisation sorgen zu müssen, ist als Managementparadigma weithin verstanden. Dass dieselbe Mechanik allerdings auch für IT-Systeme – und mehr noch: für die Interdependenzen zwischen Systemen und Organisationen – wirkt, ist zwar wissenschaftlich gut untersucht und nachgewiesen (Conway's Law), in der Praxis allerdings noch nicht weit verbreitet.

Wichtig ist daher nun, für die Kombinationsform aus Organisation, Technologie, Prozessen und Compliance sowie nicht zuletzt auch Kultur adäquate Technologiemanagementansätze zu entwickeln:

---

Technologiemanagement mit  
Fokus auf Prozesse, Technologie,  
Compliance und Organisation

---

**Getrennte Profit & Cost-Center** machen in der IT-Organisation eine übergreifende fachliche und architekturelle Steuerung notwendig, die auf Gesamtorganisationsebene Optimierungsprozesse durchsetzt und insbesondere Transparenz bezüglich der finanziellen Auswirkungen funktionaler und nicht-funktionaler Anforderungen schafft. Neben der technologischen Betrachtung und der Sicherstellung der Handlungsfähigkeit durch Entkopplung und die Nutzung moderner Technologie ist eine Vollkostenrechnung und Business Domain Mapping notwendig, um entscheidungsfähig bezüglich Technologie und Systemen zu bleiben. Für fundierte Technologieentscheidungen ist es notwendig, die Total Cost of Ownership sowohl des aktuellen Systems als auch möglicher Alternativen vergleichbar zu halten, um den Zeitpunkt des optimalen Wechsels nicht nur nicht zu verpassen, sondern auch fundiert finanziell begründen zu können.

**Integrierte Profit & Cost-Center** sind seit längerem vor allem in großen Technologiekonzernen üblich. Auch in solchen Organisationen können allerdings IT Cost Center existieren, beispielsweise in Form des Infrastrukturbetriebs. Technologieführer haben allerdings oft ebendiese Cost Center als eigenständige Organisationen am Markt platziert, um die Problematik einer adäquaten Incentivierung nach Effizienz zu lösen. Beispiele solcher ehemals internen Cost Center sind Hyperscaler wie Amazon AWS und die Google Cloud Plattform. Die Organisation nach Produkten und weitgehende Entkopplung hat den Vorteil, unprofitable Produkte und Bereiche jederzeit ohne weitreichenden Impact auf die Gesamtorganisation und IT-Systemlandschaft abkündigen und sämtliche damit verbundenen Kosten vermeiden zu können.

Eine kurzfristige, rein auf finanzielle Aspekte einer Einzelanforderung bezogene Perspektive lässt wichtige Interdependenzen wie die Anhäufung technischer Schulden und längerfristige Folgekosten außer Acht.

## 4 Proaktives Technologiemanagement als Anti-Aging Programm für die IT

Wie können Unternehmen den Herausforderungen einer fortschreitenden Alterung und Degenerierung von IT-Systemen bedingt durch sich ändernde Marktbedingungen und schnelleren technologischen Wandel effektiv begegnen?

Handlungsmöglichkeiten für Drosselung des Alterungsprozesses

### 4.1 Monitoring des technischen Zustands der IT-Systeme

Um fundierte Managemententscheidungen zur architektonischen Ausrichtung von Systemen zu ermöglichen, muss zunächst Transparenz über den Status Quo hergestellt werden sowie ein Verständnis für die Entwicklung des (Software-)Marktes, aber auch des eigenen Geschäftsmodells und möglicher Geschäftsstrategien hergestellt werden.

Hierzu bietet sich ein Review Framework an, welches eine technische Due-Diligence einzelner Services und ganzer Systeme ebenso erlaubt wie die Begutachtung der Enterprise Architektur und der organisatorischen Struktur sowie unterstützender Maßnahmen wie Controlling, Governance, Compliance und Reporting. Aus diesem resultieren Erkenntnisse in Form technischer Schulden und daraus im Laufe der Zeit auftretende Defizite. Dabei handelt es sich nicht um fachliche Fehler, sondern um verschiedene Aspekte unzureichender Softwarequalität, die sich folgend klassifizieren lassen:

Anwendung eines Architekturreview-Frameworks

- › Design- oder Architekturschwächen
- › Mängel in der Implementierung (Code Smell)
- › Testschulden durch unzureichende Testabdeckungen oder zu geringe Automatisierung
- › Unzureichende Dokumentation

Ansätze zur Bewertung von Systemen müssen sich vor allem an der Architektur und am Betriebsmodell orientieren: ein selbst entwickeltes System ist anders zu bewerten als ein COTS oder gesamtheitlich über Sourcing beschafftes. Ein wichtiger Indikator für die Wartbarkeit von Systemen ist die Komplexität, welche auch den oben beschrieben Verlust konzeptioneller Integrität transparent machen kann:

**Architektursedimentierung:** eine zunehmende Anzahl an Schichten wie Anti-Corruption-Layer oder Integrationssystemen führt nicht nur zu einer Verschlechterung der Performance des Systems, sondern erhöht durch die dadurch oft notwendige Vernetzung und Kombination unterschiedlicher Funktionen verschiedener Ebenen oft die Konnektivität und damit die Komplexität des Gesamtsystems massiv. In der Finanzbranche finden sich häufig Host/Mainframe-basierte Kernsysteme, die durch Funktions-Services abstrahiert, von Integrations- und Prozessautomatisierungssystemen vernetzt und dann durch Prozess-Services mit kanalspezifischen Schnittstellen verbunden werden. Oft sind diese Schnittstellen durch zusätzliche Business Logik nochmals gekapselt – und hinzu kommen diverse technische Module wie API Gateways, Service Repositories und diverse Zugangskontrollsysteme, wobei hierbei die Komplexität der zugrundeliegenden Netzwerkinfrastruktur mit diversen virtuellen und physischen Netzwerken, Routern, Switches und Firewalls noch gar nicht berücksichtigt ist. Eine Messung des Sedimentierungs- bzw. auch des Abstraktionsgrades (müssen Spezifika der Hardware- und Netzwerkinfrastruktur berücksichtigt

Einschätzung der Komplexität der gesamten System-Architektur

werden?) hilft, diese Komplexität einzuschätzen. Moderne Architekturen basieren auf abstrahierter Infrastruktur mit Self-Contained-Services und ggf. einem API Gateway, i.B.a. die Business Logik sollte die Anzahl der Schichten also gegen 1 tendieren.

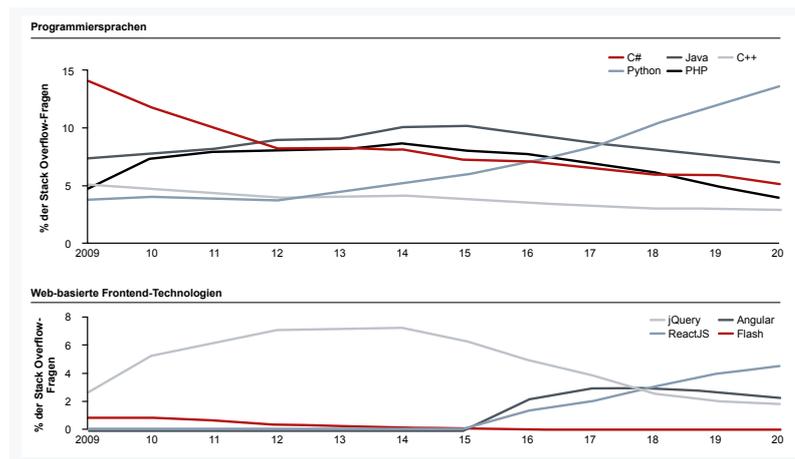
**System-/Augmentierungsgrad:** das Verhältnis von Systemen und ihrer Augmentierungen kann als weiterer Gradmesser verwendet werden: je mehr Supportsysteme und zusätzliche funktionale Module über die Zeit das ursprüngliche System aufgrund neuer Anforderungen unterstützen müssen, desto geringer ist die konzeptionelle Integrität und desto höher die Komplexität – ein Indiz für Ineffizienz. Auch dieses Phänomen ist häufig in Financesystemen zu finden: zusätzliche regulatorische Anforderungen, aber z.B. auch die Notwendigkeit, über Jahrzehnte diverse Geldautomaten- und Kartensysteme sowie ein Onlinebanking und weitere Kanäle anschließen zu müssen, haben die Komplexität der Systeme getrieben. Eine Betrachtung der notwendigen Anwendungen, um eine Funktion, Domäne oder Capability vollständig abzubilden, kann hier Klarheit schaffen, auch hier sollte in einem idealen System die Kennzahl gegen 1 tendieren.

**Alterungsindikatoren:** die Erstellung eines Technologieindex bestehend aus mehreren Kennziffern hilft bei der Bewertung, wo in der Technologieentwicklung eine Komponente oder ein Framework steht und ob ein Ersatz ins Auge gefasst werden sollte; mögliche Parameter, die in den Technologieindex einfließen könnten, sind beispielsweise die Anzahl von Nachfragen zu bestimmten Frameworks auf Stack Overflow, bei Open Source Komponenten die Anzahl an Contributern und Commits auf Github o.Ä., Programmiersprachenindices und Angebot/Nachfrage auf Plattformen wie GULP. Bei COTS Software können Marktanalysen wie Gartnerquadranten, Industrietrendanalysen und Lifecycle Notices der Hersteller sowie die Kundenanzahl und deren Entwicklung Indikatoren darstellen.

Messung der konzeptionellen Integrität durch Interdependenzen von Supportsystemen und Modulen

Erstellung eines Indexes unterstützt bei der Bewertung aktueller Technologien

Entwicklung bedeutsamer Programmiersprachen und Technologien



Quellen: Stackoverflow

Abbildung 8: Entwicklung bedeutsamer Programmiersprachen und Technologien

**Pattern Violations:** konsistent angewandte Konstruktionsmuster indizieren einen Architekturentwurf, dessen Konzeptionsgrundlagen sich als adäquat für den intendierten Einsatzzweck erweisen; (insbesondere häufige) Kompromisse und Abweichungen von den Konzeptionsparadigmen weisen auf Defizite des Architekturdesigns hin. Falls existent können die Architekturdeviationsdokumentation, die Dokumentation der technischen Schulden oder ähnliche Dokumentationen einen Anhaltspunkt liefern. In Ermangelung solcher strukturierter Erfassung kann ein Architekturreview auf verschiedenen Ebenen sinnvoll sein.

**Codequalität:** zur Erreichung hoher Codequalität, d.h. die formale Gestaltung betreffende Anforderungen wie bspw. Konformität, Verständlichkeit, Analysierbarkeit, Modifizierbarkeit, Anpassbarkeit, Prüfbarkeit (so z. B. benannt und definiert nach ISO/IEC 9126), sind regelmäßige Test bei jeglichen Anpassungen des Quellcodes unvermeidlich. Hierbei ist nicht nur auf statische Codequalitätstests abzustellen und auf komplette (Happy & Sad Path Unit-) Testabdeckung zu achten, sondern es sollten ebenso umfangreich Linting, Fuzzing, Mutation Testing, Last & Performance Messungen, Penetration Tests, Integration, Smoke und End-to-End Tests sowie Chaos Testing zum Einsatz kommen, um den positiven wie negativen Impact von Änderungen vollumfänglich abschätzen zu können. Ebenso sollten in situ Monitoring Kennwerte zur Applikationsstabilität wie Uptime, Latenz, Fehlerrate usw. einbezogen werden.

**Abweichung vom Marktstandard:** üblicherweise bilden sich über die Zeit sowohl übergreifend in der Softwarebranche wie auch in einzelnen Industriesparten gewisse de facto Marktstandards heraus. Eine Abweichung von diesen Standards bedarf eines fundierten strategischen Grundes, andernfalls handelt es sich wahrscheinlich um historische Fehlentscheidungen (Betamax Syndrom).

Die technische Schuld drückt dabei den Aufwand der Maßnahmen aus, die notwendig sind, um das IT-System wieder auf das zu erreichende Qualitätslevel zu heben. Implementierungs- und Testschulden können recht einfach toolgestützt erfasst werden. Für die Dokumentation ist die Einführung eines formalen und toolgestützten Track Recordings für Architekturentscheidungen (Architectural Decision Records) hilfreich, da hiermit Entscheidungen und Konzepte nachvollzogen werden können. Sie dienen gleichzeitig auch dazu, den Abbau struktureller Defizite zu dokumentieren.

Die Erfassung struktureller Schwächen sollte mit Methoden eines evolutionären Architekturansatzes erfolgen. Kern dieser Methode sind sogenannte Architektur-Fitness-Funktionen, die als Architekturszenarien formuliert werden und in regelmäßigen Reviews (alle 3-6 Monate) die strukturelle Integrität der IT-Systeme validieren.

---

Vollumfängliches Testmanagement  
zur Messung der Codequalität

---



---

Validierung der IT-Integrität durch  
regelmäßige Architektur-Fitness-  
Funktionen

---

#### 4.2 Abkehr von kurzfristig kostenoptimierter hin zu wertschöpfender Steuerung

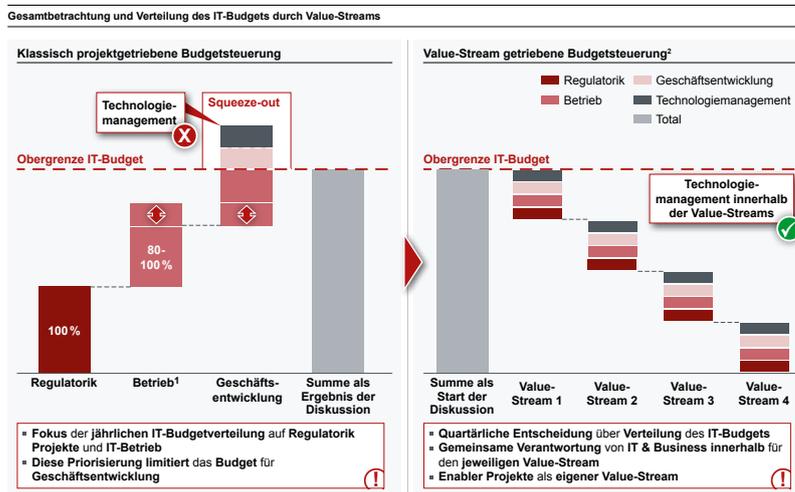
In als Cost-Center organisierten IT-Organisationen geraten ohne ein die konzeptionelle Integrität der Architektur absicherndes Management mit den hierfür notwendigen Prozessen und der Schaffung von Transparenz, beispielsweise durch Technologiefolgeabschätzungen und der Analyse der Auswirkungen einzelner Architekturentscheidung, sowohl die Systeme, als auch die sie betreibenden Organisationen in ein Dilemma, welches nur mit großen Transformationsprojekten wieder aufzulösen ist. Typische

Anzeichen sind ein gegenüber dem Change überbordendes Run Budget, oft in Größenordnungen von 80% zu 20%. Solange die Ablösung der alten Plattform in dieser Logik aus dem Change Budget finanziert werden muss und die technischen Schulden nicht eingepreist werden, ein nicht auflösbare Dilemma.

Ist geplant, diese Strukturen bis auf weiteres beizubehalten, müssen Technologie- und Architekturmanagementprozesse etabliert werden, welche auch die Messung und das Management technischer Schulden integrieren und Steuerungswirkung entwickeln können. Außerdem sollten Technologie Lifecycle Management Prozesse langfristige Veränderungen der technologischen Plattform steuern. Diese Fachprozesse müssen wiederum in die Strategie- und Budgetprozesse integriert werden.

Eine Möglichkeit ist es, die gesamten IT-Kosten für Change und Run über die geplante Laufzeit gesamthaft zu betrachten und mit dem erwarteten Nutzen des IT-Einsatzes zu koppeln (sogenannte Value-Streams). Beim Value-Stream Ansatz wird eine Gesamtbetrachtung aus Nutzen und Kosten über die gesamte geplante Laufzeit der IT-Lösung betrachtet und vom jeweiligen Fachbereich verantwortet. Dieser ist für das gesamte Budget für Run- und Change der IT-Lösung einschließlich der Umsetzung regulatorischer Anforderungen und Lifecycle-Maßnahmen, verantwortlich. Kosten für unterlassene Lifecycle-Maßnahmen oder dem Value-Stream zuzuordnende regulatorische Anforderungen können somit nicht mehr externalisiert werden und beeinflussen die Budgets der anderen Value-Stream nicht. Somit ist jeder Value-Stream incentiviert, entsprechende Lifecycle Maßnahmen so umzusetzen, dass dauerhaft eine günstige Kostenbasis erreicht wird und regulatorische Maßnahmen angemessen auszugestalten.

Verwaltung übergreifender Anforderungen an Infrastruktur durch dedizierten IT-Value-Stream



<sup>1</sup>Umfasst Maßnahmen zur Sicherstellung des laufenden Betriebs sowie explizite Change-Themen im Betrieb | <sup>2</sup>Budgetverteilung exemplarisch | Quelle: COREInstitute 2020

Abbildung 9: Mehr Flexibilität durch Portfoliomanagement in Form von Value-Streams

Ein Nachteil von Value-Streams ist, dass bereichsübergreifende Optimierung nicht realisiert werden und jeder Bereich "sein eigenes Ding macht". IT-Lösungen, für die eine gemeinsame Nutzung durch mehrere Value-Streams aus übergreifender Sicht sinnvoll ist, sollten ab einer relevanten Größe als eigene, ggf. IT-motivierte, Value-Streams geführt werden.

---

Für übergreifende technische Plattformen, die im Sinne einer Infrastruktur, die von vielen weiteren Bereichen genutzt wird, wird in diesem Modell ebenfalls ein Value-Stream eingerichtet, der i.d.R durch die IT verantwortet wird. Die Anforderungen an diese Plattform werden dann innerhalb des Budget-Rahmens durch die IT priorisiert (einschließlich der Lifecycle-Maßnahmen). Der Anspruch einer Plattform sollte jeweils sein entweder ein ähnliches Kostenniveau wie vergleichbare Marktangebote zu erzielen oder qualitative hochwertigere Services anzubieten. IT-Services, die nicht der Wettbewerbsdifferenzierung dienen, sollten konsequent vom Markt bezogen werden.

---

IT-Budget Allokation unabhängig voneinander agierender Fachbereiche

---

Eine technische Voraussetzung für die erfolgreiche Umsetzung von Value-Stream orientierter Budgetierung ist eine klare auch fachliche Domänenabgrenzung über wohldefinierte Schnittstellen, z. B. mittels Domain Mapping und strategischem Domain Driven Design, um die einzelnen Bereiche zu entkoppeln und die Verantwortlichkeiten und eigenständigen Handlungsfähigkeiten klar abgrenzbar zu halten.

Dennoch ist auch in diesen Organisationsformen die Etablierung einer übergreifenden Ingenieurskultur bzw. einer gewissen Gesamtausrichtung der Architektur sinnvoll, da durch kleinteiliges Engineering die Mobilität von Entwicklern und damit auch die Flexibilität der Domänen eingeschränkt wird.

#### 4.3 Evolutionäre Entwicklung anstelle von Großtransformationen

Aktuelle Ansätze des Architektur- und Technologiemanagements nutzen heute immer noch stark planwirtschaftliche Prinzipien. Der praktische Nutzen detaillierte Bebauungspläne mit Zeithorizonten von mehr als 10 Jahren ist heute durch immer kürzerer Technologiezyklen, Wechsel in Geschäftsstrategien und Verschiebung geschäftlicher Prioritäten komplett in Frage gestellt. Es gilt hier vielmehr, die Entwicklung der IT-Systeme im Sinne einer permanenten Erneuerung und Modernisierung auszugestalten. Hierbei können zwei sehr gegensätzliche Prinzipien angewendet werden:

**Wegwerfarchitekturen:** schon beim Bau des Systems wird ein konkretes End-of-Life (Verfallsdatum) für das System festgelegt und die Neuimplementierung als technische Schuld mit im Budget eingepreist. Dieser Ansatz bietet sich vor allem für IT-Systeme und Komponenten an, die sehr starken Änderungszyklen unterliegen. Hierzu zählen z.B. Kundenfrontends und APIs.

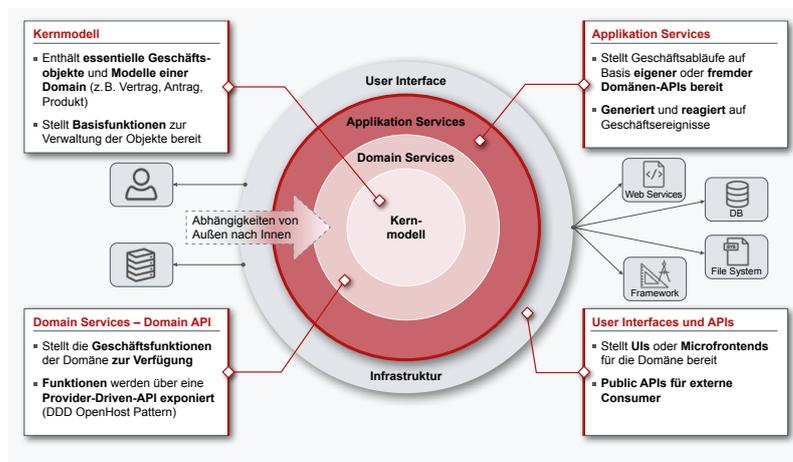
---

Kontinuierliche Adaptierbarkeit durch Anwendung antagonistischer Prinzipien

---

**Design-to-Change:** die Architektur des Systems wird auf maximale Änderbarkeit ausgelegt. Dabei wird das Prinzip einer Clean-Architecture mit einem ausschließlich an der fachlichen Funktion orientierten Modularisierung mittels Domain Driven Design kombiniert. Durch eine starke fachliche Modularisierung und technische Entkoppelung sind diese Ansätze aufwendiger in der Implementierung aber umso robuster gegen Änderungen.

## Aufbau der Domänen nach Clean-Architecture-Prinzipien



Quelle: COREresearch 2020

Abbildung 10: Clean-Architecture-Prinzipien

#### 4.4 Organisation must change as well

Neue Technologien, Methoden und geänderte Wertschöpfungsketten in der IT erfordern auch eine Anpassung der Organisationsstrukturen. Das bisher gern praktizierte Vorgehen einer „Two-Speed-IT“ als Kombination agiler Formen und digitaler Schnellboote auf der einen und einer klassischen Organisation auf der anderen Seite ist in der Praxis meist nicht zielführend, weil sich geschäftliche Anforderungen eben nicht nur auf das Kunden-Frontend, sondern auf den gesamten Produktzyklus erstrecken. Digitale Produkte agil zu designen und an den Markt zu bringen kann nur dann nachhaltig erfolgreich sein, wenn Leistungserbringung, Abwicklung und Support der Produkte mit der Weiterentwicklung der Produkte Schritt halten können.

Viele Unternehmen betreiben heute noch eigene Rechenzentren und IT-Infrastrukturen. Der Fokus vieler IT-Organisationen ist deshalb auch heute noch stärker auf den Betrieb und weniger auf die Weiterentwicklung von IT-Systemen ausgerichtet. Aufgrund der hohen Fertigungstiefe muss eine sehr hohe Bandbreite an Spezialwissen aufgebaut und aufrechterhalten werden. Um dies organisatorisch effizient abbilden zu können, wird das notwendige Know-how für Entwicklung und Betrieb typischerweise schichtenartig in spezialisierten Einheiten gebündelt. Jede übergreifende Änderung erfordert Abstimmung und Koordination zwischen den für die einzelne Schichten zuständigen Einheiten. Das macht das Management von Änderungen komplex und deren Umsetzung aufwendig und langsam.

Hohe Fertigungstiefe benötigt  
Konsolidierung von Know-how und  
verlangsamt Umsetzung

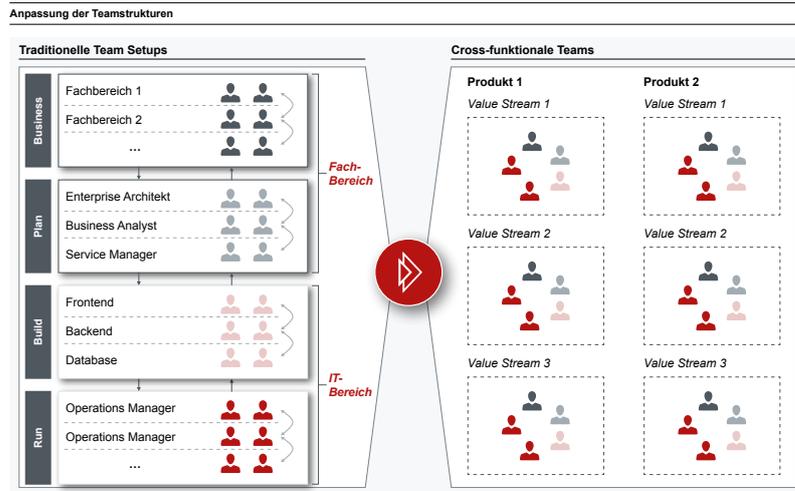


Abbildung 11: Anpassung der Teamstrukturen

Der Ansatz, eine gemeinsame Verantwortung zwischen Geschäfts- und IT-Ebene auf Produktebene für größere Abschnitte der Wertschöpfungskette auch organisatorisch abbilden, ist aus unserer Sicht der zielführendere Ansatz.

Das engere Zusammenrücken von Geschäfts- und IT-Seite wird aber erst durch neue Arbeitsteilungs- und Wertschöpfungsketten in der IT und eine Reduktion der eigenen Fertigungstiefe möglich. Durch die starke Standardisierung von IT-Infrastrukturdiensten können diese Dienstleistungen von darauf spezialisierten Anbietern bezogen und über maschinelle Schnittstellen automatisiert gesteuert werden. Durch die damit mögliche sehr weitgehende Automatisierung der klassischen IT-Operations-Aufgaben entfällt die Notwendigkeit eigener dedizierter Betriebsstrukturen. Entwicklung und Betrieb können zu einer Einheit verschmelzen. Ein weiterer Vorteil des auch als Dev-Ops-Modell bekannten Ansatz einer gemeinschaftlichen Verantwortung ist, dass kurzfristige Optimierungen der Entwicklung zu Lasten der Betriebssicherheit damit besser verhindert werden können.

Wird dieser Ansatz konsequent weitergedacht, besteht auch kein Grund mehr, IT- und Fachorganisation voneinander zu trennen. Die für die Gestaltung und Abwicklung digitaler Geschäftsprozesse aktuell genutzten Methoden setzen eine starke Interaktion zwischen IT-Entwicklern, Dev-Ops und den fachlichen Verantwortlichen voraus. Viele Unternehmen setzen bereits heute auf eine Matrixstruktur, die aber in der ersten Ordnung der IT durch die Technologieschichten bestimmt wird. Dieser Matrixaufbau sollte transponiert werden, so dass der Fachbereich mehr zur IT und die IT immer mehr zum Fachbereich wird. Anstelle der bisherigen Schichten aus Spezialisten werden dauerhafte vertikale Strukturen aus gemischten Teams gebildet, so dass eine Anforderung Ende-zu-Ende durch ein Team

Assimilierung von Fachbereich und IT auf Produktebene

Reduktion der eigenen Fertigungstiefe durch Sourcing standardisierter IT-Infrastrukturdienste

abgebildet werden kann. Das fachliche Domänenstruktur wird damit zu ersten Ordnungsstruktur. Die Weiterentwicklung des notwendigen technischen Know-hows kann in einer zweiten Ordnungsstruktur z.B. innerhalb einer Gildenstruktur als Community erfolgen.

Gemäß der Conway'schen Beobachtung, dass Systemstrukturen immer auch die Kommunikationsstrukturen einer Organisation widerspiegeln, erfordern geänderte Rahmenbedingungen und Arbeitsteilung- und Methoden auch Änderungen an der Organisation. Sonst sind und bleiben Design-Thinking-Kurse, zertifizierte Scrum-Master, Produktowner und Kanban-boards nicht mehr als eine Art agiler Deckmantel.

## 5 Fazit

Die steigende Bedeutung von Informationstechnologie für die Geschäftsmodelle in praktisch allen Industrien ist allgegenwärtig. Die Qualität und Produktivität der IT-Systeme als primäres Produktionsmittel bestimmen heute maßgeblich die Wettbewerbsfähigkeit der Unternehmen. Aus geschäftsstrategischer Perspektive sollten deshalb Systeme der Informationstechnologie als wichtige Produktionsassets und nicht ausschließlich als zu optimierender Kostenfaktor betrachtet werden. IT-Systeme unterliegen einem stetigen Alterungsprozess, der einerseits durch die technologische Weiterentwicklung und andererseits durch die geänderten geschäftlichen Rahmenbedingungen getrieben wird. Um den Wert der IT-Systeme für die Unternehmen zu erhalten und auszubauen, sind deshalb permanente Investitionen in diese wichtigen Produktionsmittel unabdingbar. Dies setzt wiederum ein umfassendes Wissen um den qualitativen Zustand der IT-Systeme voraus. Dieser sollte nach objektiv prüfbaren Kriterien erfasst und bewertet werden. Hier spielen nicht nur die „äußeren“ und von den Nutzern direkt erlebbaren Betriebsqualitäten eine Rolle, sondern auch die „inneren“ Werte der Systeme, wie deren strukturelle Integrität, innere und äußere Abhängigkeiten und deren technischer Reifegrad.

Marktdifferenzierung und Wettbewerbsfähigkeit durch Effizienz der IT-Systeme

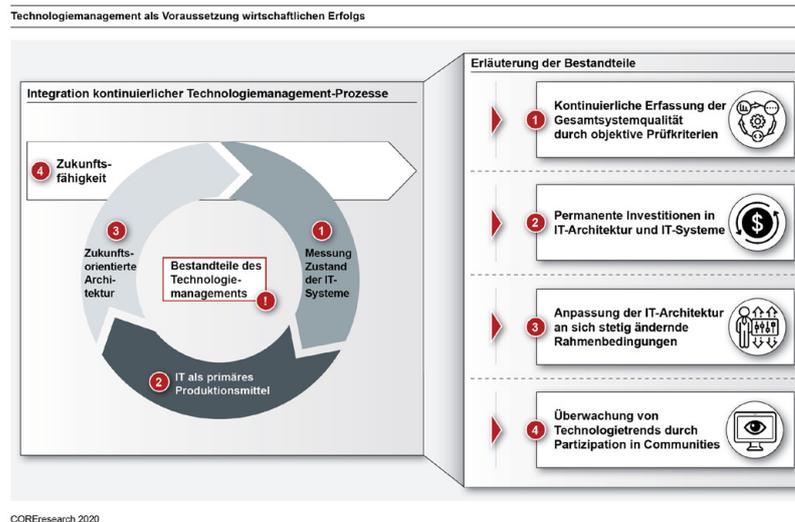


Abbildung 12: Technologiemanagement als Voraussetzung wirtschaftlichen Erfolgs

Die systematische und regelmäßige Erfassung der Systemqualitäten und Defiziten als zu steuernde Größen ist deshalb ein wichtiger Baustein eines modernen Technologiemanagements. Das Technologiemanagement soll dabei aber nicht als Aufsichtsbehörde restriktiv agieren, sondern versuchen, im Sinne einer Partnerschaft die Systemqualitäten flexibel innerhalb eines Korridors zu steuern. Dabei unterstützen können Methoden des evolutionären Architekturmanagements, das als Kernidee eine laufende Anpassung der IT-Architekturen an sich ändernde Rahmenbedingungen beinhaltet und damit bessere Möglichkeiten der Anpassung an beschleunigte Innovations- und Technologiezyklen bietet als starre planwirtschaftliche Bebauungsplanmodelle und den damit verbundenen Großprojekten und Transformationen.

Systematische Erfassung von Systemqualitäten als Kernelement des Technologiemanagements

---

Neue IT-Architekturen sollten heute immer unter dem Aspekt entworfen werden, dass sie zukünftig vollständig oder in Teilen ersetzt werden müssen. Für IT-Systeme mit langen Nutzungshorizonten sind deshalb beim Entwurf gute fachliche Modularität und lose technische Kopplung besonders zu berücksichtigen. In Bereichen, die einem besonders hohen Innovations- und Änderungsdruck unterliegen (z.B. Kundenschnittstellen) kann es effektiver sein, IT-Systeme mit einer Art Mindesthaltbarkeitsdatum zu versehen, und diese Systeme nach Ablauf des Datums auf einer neuen Technologieplattform komplett neu zu implementieren.

---

Planung und Konzeption von  
IT-Systemen auf Basis von  
Ablaufdaten und Modularität

---

Technologieinnovation in der IT und damit verbundenen Produktivitätsschübe durch neue Methoden finden heute bereits zum großen Teil in einer globalen Community als ein evolutionärer Prozess statt. Aufgrund der starken Arbeitsteiligkeit der Technologieentwicklung ist eine Entkopplung der eigenen Technologieplattform vom Rest der Welt heute faktisch nicht mehr möglich. Ein aktives Tracking aktueller Technologietrends und Zyklen gehört deshalb auch in das Repertoire des Technologiemanagements. Hier kann ein regelmäßiger Austausch eigener Softwareentwickler und Architekten mit den weltweit vernetzten Entwickler-Communities kombiniert mit der gezielten Bildung gemischter Teams aus externen und internen Entwicklern für die notwendigen Impulse sorgen. Das bedeutet im Zweifelsfall auch mal höhere Aufwände und zahlt nicht immer direkt auf eine gewünschte fachliche neue Funktion oder Änderung ein, sorgt aber langfristig für eine bessere Wettbewerbsfähigkeit der eigenen IT-Systeme.

---

## Verfasser



**Christian Böhning** ist Managing Partner bei CORE und ist verantwortlich für die Planung und Steuerung komplexer IT-Transformationen, insbesondere im Rahmen der Modernisierung und Neuausrichtung der IT-Organisationsarchitektur. Er sichert den Erfolg unserer Kunden, indem er technische IT- und Managementkompetenz kombiniert. Christian hat mehr als 20 Jahre Berufserfahrung im Umgang mit technologiegetriebenen Veränderungen in der Finanzindustrie und im Biotechnologiebereich.

---

Christian Böhning  
[christian.boehning@core.se](mailto:christian.boehning@core.se)

---



**Dr. Carsten Wedekind** ist Expert Director bei CORE und konzentriert sich auf die strategische Ausrichtung von IT-Architekturen. Er fungiert auch als Co-Vorsitzender der Lending Working Group im Banking Industry Architecture Network (BIAN). Der promovierte Physiker verfügt über langjährige Erfahrung in der Umsetzung von IT-Projekten mit agilen und klassischen Methoden. Vor seinem Wechsel zu CORE unterstützte Carsten namhafte internationale Unternehmen bei der Einführung von Standardsoftwarepaketen sowie bei individuellen Entwicklungen.

---

Dr. Carsten Wedekind  
[carsten.wedekind@core.se](mailto:carsten.wedekind@core.se)

---



**Dr. Philipp Kleine Jäger** ist Managing Partner bei CORE und verfügt über umfangreiche Beratungserfahrung in IT Strategie und Organisationsentwicklung sowie Sourcing-, Technologie- und Innovationsmanagement. Er hat in knapp 20 Jahren umfassendes Wissen über komplexe Finanzprodukte und -systeme durch seine Arbeit als Lead Architect einer internationalen Geschäftsbank für strukturiertes Kreditgeschäft und Entwickler von Portfoliomanagementsystemen gesammelt. Neben klassischen Transformationsprojekten hat er für Kunden Cloudarchitekturen geplant und implementiert.

---

Dr. Philipp Kleine Jäger  
[philipp.kleinejaeger@core.se](mailto:philipp.kleinejaeger@core.se)

---



**Francisca Di Piazza** ist Transformation Associate bei CORE. Ihren Master in Betriebswirtschaftslehre hat sie an der Universität Potsdam absolviert. Ihre Schwerpunktthemen umfassen Innovations- und Technologie Management, Technology Foresight, Identity Management, Data Science, Social Media Forschung und Konzeption digitaler Plattformen. Sie hat bereits Erfahrung in der Durchführung von Marktanalysen, im agilen Produktmanagement und bei der Produktkonzeptionierung und Evaluierung der Machbarkeit und Implementierung eines Produktes.

---

**Francisca Di Piazza**  
[francisca.dipiazza@core.se](mailto:francisca.dipiazza@core.se)

---



**Justus Dircks** ist Transformation Associate bei CORE. Seinen Master in Entrepreneurship und Innovationsmanagement hat er an der Universität Lund in Schweden absolviert. Seine Schwerpunktthemen umfassen agiles Projektmanagement und Start-Ups. Er hat bereits Erfahrung in der Steuerung von Entwicklerteams bei der Weiterentwicklung mobiler Applikationen und verschiedenen Transformationsprojekten im Bankensektor sammeln können.

---

**Justus Dircks**  
[justus.dircks@core.se](mailto:justus.dircks@core.se)

---



**Karsten Trostmann** ist Expert Director bei CORE. Der Informatiker deckt folgende Schwerpunktthemen ab: IT-Strategie, Evolutionäre IT-Architektur, Domain Driven Design, agile Softwareentwicklung und Cloud Infrastrukturen. Karsten's Erfahrungen umfassen unter anderem die Evaluierung der Plattformstrategie für einen Digitalversicherer, die Entwicklung Cloud Operations für einen Identityprovider, die fachliche Architektur und Implementierungssteuerung für ein Online-Bezahlverfahren und Architekturreviews in verschiedenen Projektsituationen.

---

**Karsten Trostmann**  
[karsten.trostmann@core.se](mailto:karsten.trostmann@core.se)

---

---

## Über COREresearch

Als unabhängiger Technologie Think Tank erforschen wir die Systematik technologisch getriebener Transformationen in Industrien mit einem hohen Anteil an IT im Wertschöpfungsprozess. Im Rahmen unserer Forschungsaktivitäten analysieren wir Märkte und Technologien, thematisieren Strukturen, Ursachen und Wirkmechanismen des technologischen Wandels und kuratieren Ergebnisse für Klienten und die Öffentlichkeit. Darüber hinaus stellen wir ausgewählte Resultate unserer interdisziplinären Forschungen im Rahmen von übergreifenden Publikationen, Einzelstudien sowie Vorträgen einer breiteren Öffentlichkeit zur Verfügung.

---

<https://core.se>

---

## Disclaimer

Inhalt und Struktur unserer Publikationen sind urheberrechtlich geschützt. Die Vervielfältigung von Inhalten, insbesondere die Verwendung von Texten, Textteilen oder Bildmaterial, bedarf der vorherigen Zustimmung. Die abgebildeten Logos stehen im Eigentum der jeweiligen Unternehmen. Die CORE SE hält keine Rechte an den Logos und nutzt diese ausschließlich zu wissenschaftlichen Zwecken.

CORE SE  
Am Sandwerder 21–23  
14109 Berlin | Germany  
<https://core.se/>  
Phone: +49 30 263 440 20  
[office@core.se](mailto:office@core.se)

COREtransform GmbH  
Am Sandwerder 21–23  
14109 Berlin | Germany  
<https://core.se/>  
Phone: +49 30 263 440 20  
[office@core.se](mailto:office@core.se)

COREtransform GmbH  
Limmatquai 1  
8001 Zürich | Helvetia  
<https://core.se/>  
Phone: +41 44 261 0143  
[office@core.se](mailto:office@core.se)

COREtransform Ltd.  
Canary Wharf, One Canada Square  
London E14 5DY | Great Britain  
<https://core.se/>  
Phone: +44 20 328 563 61  
[office@core.se](mailto:office@core.se)



COREtransform Consulting MEA Ltd.  
DIFC – 105, Currency House, Tower 1  
P.O. Box 506656  
Dubai | VAE Emirates  
<https://core.se/>  
Phone: +97 14 323 0633  
[office@core.se](mailto:office@core.se)