

Die Softwarewartungskosten als Managementproblem im Wandel der Zeit – Ergebnisse einer Metaanalyse

Franz Lehner¹

Abstract: Dieser Beitrag stellt eine Reflexion der Entwicklung über einen Zeitraum von vier Jahrzehnten dar und ist der Versuch einer integrativen Bestandsaufnahme aus der Perspektive der Wirtschaftsinformatik und der Informatik. Dabei wird die technische Sicht auf Informationssysteme mit einer ökonomischen und ganzheitlichen Bewertung verbunden. Idealerweise sollten die Wartungskosten heute keine große Rolle mehr spielen. Da sich seit Beginn einer ernsthaften wissenschaftlichen Auseinandersetzung mit der SW-Wartung aber viel verändert hat, ist die Frage, ob sich die Situation seither verbessert oder nur verändert hat. Die Problematik der SW-Wartung wird zunächst aus der Perspektive der Wirtschaftsinformatik erläutert, weil die ökonomische Bedeutung der technischen Herausforderung lange hintangestellt worden ist. Anschließend werden die Entwicklung und die Bedeutung der SW-Wartungskosten in drei Phasen (80er Jahre, 90er Jahre, 2000 bis heute) anhand verfügbarer Zahlen, Fakten und Studien analysiert und gleichzeitig die Perzeption des Themas durch die Wirtschaftsinformatik und die Informatik reflektiert. In einem zweiten Schritt wird eine Metaanalyse durchgeführt, um zu klären, wie sich der Anteil der SW-Wartungskosten in diesem Zeitraum entwickelt hat.

Keywords: Softwarewartung, SW-Wartungskosten, Wartungskostenanteil, Verhältnis SW-Wartungs-/SW-Entwicklungskosten, Längsschnittanalyse

1 Einleitung und Zielsetzung

Unter Softwarewartung werden all jene Tätigkeiten verstanden, die nach der Einführung eines Softwaresystems dessen Verfügbarkeit und Nutzbarkeit sicherstellen oder seine Lebensdauer verlängern. Man rechnet den gesamten Aufwand dazu, der nach der Übergabe oder Einführung der Software auftritt. Zeit- und Kostenaspekte führen allerdings dazu, den Aufwand eines Wartungsprozesses gering zu halten. Folglich kann es dazu kommen, dass Änderungsanforderungen nur teilweise umgesetzt werden und weitere Fehler entstehen, die wiederum Folgen auf die Wartbarkeit haben können. In den meisten heute bekannten Definitionen und Definitionsversuchen werden sowohl die Funktionserhaltung als auch die Funktionserweiterung angesprochen. In der Literatur wurde ein Wartungsverständnis, das sowohl Erweiterungen als auch Fehlerreparaturen miteinschließt, allerdings mehrfach kritisiert. Aus diesem Grund wurde bereits früh ein differenziertes Verständnis für Softwarewartung vorgeschlagen, bei dem insgesamt fünf

¹ Universität Passau, Lehrstuhl für Wirtschaftsinformatik mit Schwerpunkt Informations- und IT-Service-Management, Innstraße 43, 94032 Passau, franz.lehner@uni-passau.de

Wartungskategorien unterschieden werden, nämlich korrigierende Wartung, Anpassungswartung, perfektionierende Wartung und Leistungsverbesserung, Funktionserweiterung, und Betreuung und Unterstützung [Le91].

Das Verhältnis zwischen den einzelnen Wartungsarten ist relativ gut untersucht (z.B. [Le91], [Hu08]). Patches, Präventiv-Maßnahmen, „Sanity-Checks“ oder Upgrades mit dem Ziel der Stabilisierung lassen sich dabei nicht immer eindeutig einordnen, aber am grundlegenden Begriffsverständnis für Softwarewartung selbst hat sich eigentlich nichts geändert und auch die klassischen Definitionen werden noch häufig herangezogen (z. B. [LST78], [Pa94]). Geändert haben sich allerdings die Herausforderungen im Umgang mit dem Thema, u. a. durch die Weiterentwicklung bei der Softwareentwicklung, technologische Veränderungen sowie Größe und Komplexität moderner Softwarelandschaften, aber auch Entwicklungen in der Softwareindustrie. Die Wartungskosten zielen aber nicht nur auf ein isolierbares Einzelsystem ab, sondern es geht zumindest aus wirtschaftlicher Sicht immer auch um die kumulierte Betrachtung des gesamten Softwarebestands und damit verbundener Maßnahmen bzw. Kosten.

Zu erwähnen ist an dieser Stelle noch, dass in der Literatur weitere Begriffe wie SW-Reengineering, SW-Evolution, Reverse Engineering, SW-Refactoring, SW-Migration, Software Transformation u. ä. eingeführt wurden (vgl. z.B. [Le95]). Diese Differenzierung ist im vorliegenden Kontext nicht relevant, sodass in diesem Beitrag Softwarewartung als gemeinsamer und verbindender Überbegriff verwendet wird.

Das sich verändernde wirtschaftliche und technologische Umfeld wurde lange ignoriert. Vor 30 Jahren waren Firmen meist Anwender und SW-Entwickler zugleich. Es ging um den Aufbau adäquater Programmierkapazitäten und um die Verringerung des oft bestehenden Anwendungsrückstaus. Das Verhältnis zwischen den Entwicklungs- und den Wartungskosten wurde zwar kritisch beobachtet, weil hohe Wartungskosten leicht zu einem Innovationshindernis werden konnten, aber man verstand die Aufgabe eher als praktisches und beherrschbares Problem.

Wartungskosten, die mehr als die Hälfte des IT-Budgets ausmachten, waren trotzdem nicht ungewöhnlich. Diese „traditionelle Sicht“ ist auch heute nicht ganz überwunden. Allerdings zeigt sich, dass im „Zeitalter der Digitalisierung“ viele Firmen zu reinen „Anwendern“ geworden sind, indem sie auf ein breites Angebot von IT-Serviceleistungen zurückgreifen. Da Software inzwischen überwiegend extern bezogen wird, ist die Wartungsproblematik in der ursprünglichen Form scheinbar obsolet. In der ursprünglichen Form betrifft sie natürlich weiterhin Softwarehäuser, aber auch Firmen mit SW-intensiven Produkten wie Bosch, AVL, die Finanzindustrie u. v. a. m., wo es nach wie vor viel Eigenentwicklung gibt. Für das Gros der Unternehmen ist die Softwarewartung aber inzwischen ein externer Faktor, der nur mehr in den Kosten für SW-Wartungs- und IT-Serviceleistungen sichtbar wird und somit auch nur begrenzt beeinflusst werden kann.

Bei Unternehmen mit Softwarelandschaften ohne relevante eigene Entwicklungskapazität versuchte man in der Folge mit unterschiedlichen Maßnahmen wie der Modernisierung der IT-Infrastruktur (z. B. Hardware-Upgrades, Virtualisierung) die Kosten zu senken.

Das kann durchaus erfolgreich sein, wenn kostengünstige und performante Alternativen zu weit verbreiteten Standardlösungen (z. B. Oracle, SAP) gefunden werden. Komplexe Lizenzmodelle, komplizierte Preisstrukturen und der implizite Zwang zur Aufrechterhaltung von Legacy-Technologien erweisen sich aber leicht als kontraproduktiv im Hinblick auf die notwendige Modernisierung von Datenbanken und Softwarearchitekturen. Für Softwarehäuser stellt die Wartung natürlich eine wichtige Umsatzquelle dar, was die Gesamtsituation nicht einfacher macht. Das Wartungsgeschäft wirft im Vergleich zu den Lizezeinnahmen zum Teil einen höheren Gewinn ab und manche Softwarehäuser erhöhen die Wartungskosten für ältere Software-Releases gezielt, um Kunden zu bewegen, auf aktuellere Versionen zu wechseln. Von den Anwendern wird diese Abhängigkeit zunehmend als Last empfunden. Es ist auch festzustellen, dass der Druck auf die Hersteller wächst und Anwender immer öfter nach Alternativen suchen, um die Kosten für Wartung und Support in den IT-Budgets zu reduzieren. In Verbindung damit sollen auch die rechtlichen Aspekte der Softwarewartung in Erinnerung gerufen werden, die aber sowohl aus Sicht der Wissenschaft als auch der Judikatur als gelöst angesehen werden.

Vor dem Hintergrund der kurz dargestellten Veränderungen wird deutlich, dass es sich bei der Softwarewartung auch aus wirtschaftswissenschaftlicher Sicht um ein relevantes Problemfeld handelt. Neben der Frage nach einem Perspektivenwechsel der Wissenschaft in Bezug auf die Softwarewartung ist insbesondere die Höhe der SW-Wartungskosten in Unternehmen von Interesse. Aufgrund der technologischen Weiterentwicklung und der wirtschaftlichen Veränderungen müssten der Aufwand bzw. die Kosten für die Softwarewartung in den Unternehmen eigentlich gesunken sein. Aber hat sich die Situation im Untersuchungszeitraum verbessert (= Kostenreduktion) oder haben sich nur die Kostenfaktoren verändert? Dieser Frage soll im vorliegenden Beitrag mit einer aktuellen Bestandsaufnahme und einer strukturierten Literaturanalyse zur Entwicklung und zum Stand des Wissens nachgegangen werden. Im ersten Schritt wird das sich weiterentwickelnde Problemverständnis in drei zeitlich gut abgrenzbaren Phasen (80er Jahre, 90er Jahre, 2000 bis heute) reflektiert (Kapitel 2 bis 4). Im Anschluss daran wird im zweiten Schritt eine Metaanalyse durchgeführt (Kapitel 5), um zu klären, ob der Anteil der SW-Wartungskosten in diesem Zeitraum zurückgegangen ist oder an Bedeutung verloren hat.

2 Erste Phase: Entwicklung eines eigenständigen Problemverständnisses in der noch jungen Disziplin „Wirtschaftsinformatik“

Die 80er Jahre waren für die Informatik eine Zeit des Aufbruchs und einer starken Dynamik. Viele Entwicklungen, die auch heute noch eine enorme Bedeutung besitzen, wurden damals zugrunde gelegt. Die Informatik gewann in der Wissenschaftslandschaft zunehmend an Bedeutung und emanzipierte sich mehr und mehr von der Mathematik. Die 80er Jahre sind auch jene Zeit, in der die Wirtschaftsinformatik als neu etablierte Wissenschaft begann ein eigenständiges Profil zu entwickeln. Aus dem Spannungsfeld knapper Ressourcen und eines stark wachsenden Bedarfs an Software begann man sich in beiden Disziplinen zunehmend für Fragen der Qualität und der Wartung von Software zu

interessieren. Die Entwicklung war zunächst von einer primär technischen Sicht dominiert und die Softwarewartung noch nicht als Managementaufgabe verstanden.

Es gab viele Vermutungen und Berichte über Probleme in Verbindung mit der Softwarewartung, die von Programmierern zudem nicht als besonders attraktive Aufgabe angesehen wurde. Es handelte sich um Vermutungen wie jene über kontinuierlich steigende Wartungskosten, Zunahme des Wartungsanteils im Vergleich zur Neuentwicklung, die Vernachlässigung von Innovation durch den sogenannten "Anwendungsrückstau", die Abnahme der Wartbarkeit älterer Software u. a. m. Teilweise wurden sie in sogenannte „Laws of Software Evolution“ gefasst (vgl. z. B. [Le80], [Le97]). Diese „Beliefs“ wurden schnell populär und sie wurden teilweise auch missverstanden. Beispiele für diese „Laws of Software Evolution“ sind zunehmende Komplexität von Softwaresystemen, anhaltendes funktionales Wachstum und die mit dem Alter sinkende Qualität der Software.

Eine erste empirische Entgegnung zu den ursprünglichen „Five Laws of Software Evolution“ von Lehman (1980) findet sich übrigens bereits zwei Jahre später bei Lawrence [La82], der wenig empirische Evidenz für diese Gesetze feststellen kann. Es fällt allerdings auch auf, dass manche Studien in der Folge nach wie vor an „alte“ Ergebnisse anknüpften und diese relativ unkritisch bestätigten. So findet sich z. B. bei [KS97] und bei [Jo95] u. a. der Hinweis, dass die Änderungs- und Fehlerhäufigkeit bei alten Programmen zunimmt. Maxwell/Forselius ([MS00], [MS01]) stellen hingegen fest, dass der Wartungsaufwand mit dem Alter sogar sinkt. Die empirische Basis der Längsschnittdaten ist für solche Schlussfolgerungen generell nicht ausreichend und im Prinzip findet sich für fast jede Meinung auch ein „empirischer“ Beleg. Eine umfassende und kritische Reflexion der „Evolution“ der „Laws of Software Evolution“, einschließlich der Frage ihrer Anwendbarkeit auf Open Source Software, findet sich bei Herraiz et al. [He13].

Auch das Verhältnis des Wartungsaufwandes zum Entwicklungsaufwand von Software soll an dieser Stelle kurz thematisiert werden, da es als wichtige Kennzahl für das Innovationsverhalten eines Unternehmens bereits damals gerne verwendet wurde. Zum genauen Verhältnis finden sich in der Literatur unterschiedliche und zum Teil auch widersprüchliche Angaben. Der durchschnittliche, kumulierte Wartungsanteil wird oft zwischen 20 und 30 % beziffert. Es gibt allerdings die Vermutung, dass de facto ein höherer Wartungsanteil vorliegt. Gründe dafür könnten in einem ungenauen Wartungsbegriff und in ungenauen Daten über Wartungsaktivitäten liegen. Eine Verzerrung ergibt sich auch durch die punktuelle Betrachtung, die solchen Prozentwerten zugrunde liegt. Die weit verbreitete Befürchtung bestand darin, dass der Wartungsanteil gegenüber der Neuentwicklung immer mehr zunimmt.

Der Fokus lag in dieser Betrachtungsperiode überdies bei Einzelsystemen und nicht beim gesamten Softwarebestand. Das Fehlen der übergeordneten Perspektive führte zumindest aus ökonomischer Sicht zu kurzsichtigen Entscheidungen und im Management sah man den Wald vor lauter Bäumen (=Programme) nicht. Was fehlte, waren vor allem die

empirische Evidenz, das Verständnis von Softwarewartung als ganzheitliche, unternehmensbezogene Aufgabe und die Auseinandersetzung mit nichttechnischen Einflussfaktoren und Lösungskonzepten.

In den Unternehmen dominierte in dieser Phase eine zentralisierte IT-Organisation und trotz zunehmender Verbreitung von PCs dominierten die Anwendungen für Großrechner. Der Softwarebedarf als Folge der zunehmenden „Informatisierung“ wurde überwiegend durch unternehmenseigene Programmierabteilungen gedeckt. Eine Sorge war, durch den allmählich sichtbar werdenden Ballast der Altsysteme (= hohe Wartungskosten) in der Neuentwicklung gebremst zu werden. Man sah in den 80er Jahren in den Wartungskosten ein zunehmendes Innovationshemmnis, das es zu reduzieren galt.

In der Wirtschaftsinformatik begann man in der Folge mit der Sammlung und Analyse von Unternehmensdaten sowie der Überprüfung der erwähnten Gesetzmäßigkeiten. Dabei bestätigte sich, dass es sich keineswegs um Gesetzmäßigkeiten handelte und eine differenzierte Betrachtung erforderlich ist. Einige Ergebnisse sollen beispielhaft herausgegriffen und kurz dargestellt werden. Es konnte z. B. gezeigt werden, dass der Wartungsaufwand keineswegs generell steigt. In Verbindung mit der Auswertung zusätzlicher organisatorischer Aspekte wurde klar, dass erfolgreiche Wartung auch ein Managementthema war und die Wartungsaktivitäten einer aktiven Steuerung unterworfen werden konnten. Dies schloss neben den ökonomischen Aspekten weitere Themen wie z. B. das Personalmanagement (Motivation von Wartungsprogrammierern und Aufgabenverteilung) als Erfolgsfaktor mit ein (vgl. [Le90]). Gezeigt werden konnte ferner, dass auch der kumulierte Wartungsaufwand (d. h. der Wartungsaufwand für den gesamten Softwarebestand eines Unternehmens) im Jahresvergleich nicht generell steigt, was den Einfluss von Managementmaßnahmen (z. B. IT-Budgetvorgaben) zu bestätigen scheint.

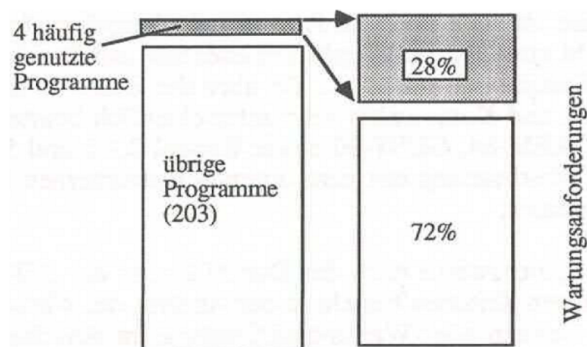


Abb. 1: Verteilung des Wartungsaufwands [Le89]

In diese Phase fällt auch die Entwicklung von Modellen zur Schätzung und Prognose des Wartungsaufwands (vgl. z. B. [Sn04], [Re11], [Bu11], ein Überblick findet sich u.a. in [Ko03]). In der Praxis haben diese Modelle aber nur wenig Verbreitung gefunden. Ein etwas anderer Ansatz, aber eine ähnliche Zielsetzung wird mit dem TCO-Ansatz (Total

Cost of Ownership), angewendet auf Software (vgl. [Jo06], [Go08]), und mit dem ROI (vgl. [Ra09]) verfolgt.

Zusätzliche Evidenz konnte durch die Analyse vollständiger Softwarebestände gewonnen werden. Dabei stellte sich heraus, dass der Wartungsaufwand im Allgemeinen sehr ungleich verteilt ist und die Hauptlast bei einer überschaubaren Anzahl von änderungsintensiven Systemen liegt (vgl. Abb. 1). Wir finden hier das Pareto-Prinzip wieder, wonach sich etwa 80 Prozent der Änderungen auf etwa 20 Prozent des Codes konzentrieren. Es genügt also im Allgemeinen durch einfache Häufigkeitsanalysen diese Anwendungen zu identifizieren. Die 80/20-Regel wurde auch von anderen Studien (z. B. [KS97]) bestätigt und stellt eine gute Grundlage für Managementaktivitäten dar.

Nicht nachweisbar war die in der Literatur ebenfalls verbreitete Meinung, dass der kumulierte Wartungsanteil gegenüber der Neuentwicklung immer mehr zunimmt. Das Verhältnis zwischen Wartung und Neuentwicklung ist weder stabil noch periodisch gleichbleibend (vgl. [Le91]). Wesentlich mehr als systemimmanente Eigenschaften dürften die verfügbare Personalkapazität, die aktive Personalplanung sowie weitere Managementmaßnahmen zur Erklärung des Wartungsaufwands beitragen. Es handelt sich also nicht um eine beobachtbare Gesetzmäßigkeit, sondern um unternehmensspezifische Gegebenheiten, die einer bewussten Steuerung zugänglich sind. Unabhängig davon bleibt unbestritten, dass die Softwarewartungskosten einen wesentlichen Kostenfaktor in den IT-Budgets ausmachen.

3 Zweite Phase: Intensivierung der Forschung und Bemühen um eine Integration der unterschiedlichen Perspektiven

In den 90er Jahren, also 10 Jahre später, kann man den Durchbruch des Internets (und die erste Internet-Blase) beobachten. Dieser geht einher mit der wachsenden Verbreitung dezentraler Systeme und Client-Server-Lösungen, dem zunehmenden Einsatz von Standardsoftware etc. Die Dominanz starker, zentralisierter IT-Abteilungen in den Unternehmen besteht aber weiterhin, wenngleich die eigenständige Programmierung größerer Lösungen (abgesehen von Web-Anwendungen) in den Unternehmen bereits abzunehmen beginnt. Auf die Softwarewartung und ihr Management wirken sich diese Veränderungen natürlich aus. Zu beobachten ist eine deutliche Zunahme wissenschaftlicher Studien und Publikationen zu unterschiedlichen Themen mit Wartungsbezug (z. B. Bestätigung des Zusammenhangs zwischen Code-Länge, Komplexität und Wartungskosten, kein Einfluss des Alters der Software auf den Wartungsaufwand, objektorientierte Programmiersprachen sind nicht per se wartungsfreundlicher, Kostenanteil einzelner Wartungsaktivitäten), was ein Indikator dafür ist, dass die Relevanz des Themas allgemein erkannt worden ist.

An einigen Beispielen soll der Wissensstand für diese Phase kurz reflektiert werden, der in einem gewissen Sinne auch einen Rückschritt darstellt, da bereits gesicherte Ergebnisse ignoriert wurden. Dies betrifft z. B. den Mythos kontinuierlich steigender Wartungskosten (vgl. Abb. 2, eine ähnliche Entwicklung wird auch bei [Jo06] skizziert).

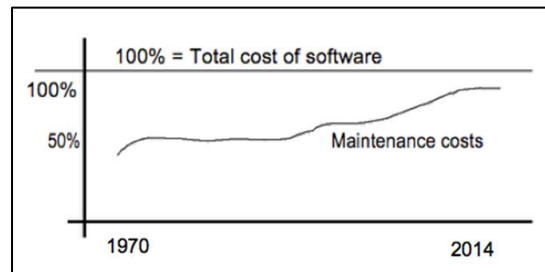


Abb. 2: Entwicklung der Wartungskosten ([Ko03], ELTIS-Projekt)

Die Erwartung steigender Wartungskosten steht in Verbindung mit der Befürchtung einer „Legacy Crisis“, die regelmäßig prognostiziert wurde ([Se03], vgl. dazu auch [Za06]). Eine neuere Studie ([Ik18]) kommt allerdings nur auf einen Anteil von etwa 47 Prozent.

Die Frage ist nun, ob der Anteil der Wartungskosten wirklich kontinuierlich angestiegen ist? Da es nach wie vor kaum Längsschnittdaten bzw. Daten aus öffentlich zugänglichen Statistiken gibt, ist die Überprüfung nicht ganz einfach. Abbildung 3 bestätigt jedoch die bereits in den 80er Jahren gewonnene Einsicht, dass der Anteil der Wartungskosten keineswegs beliebig steigt (vgl. auch Abb. 5) und offensichtlich auch durch das Management beeinflusst werden kann (Releasezyklen lassen sich planen). Im Beispiel wird übrigens bereits zwischen internen und externen Wartungskosten (organic vs. contract) unterschieden.

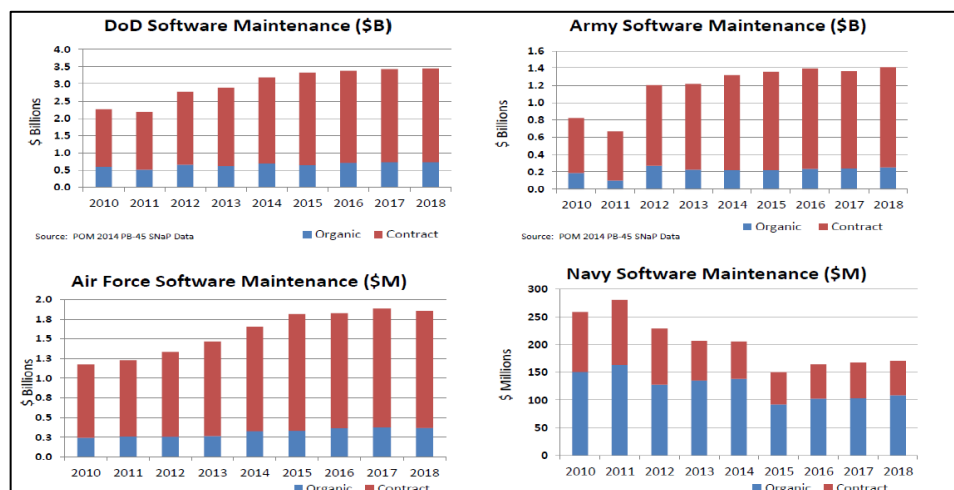


Abb. 3.: Deputy Assistant Secretary of the Army for Cost and Economics, ICEAA Workshop 2015

Zusammenfassend ist festzustellen, dass das Fehlen verlässlicher empirischer Zahlen ein großes Problem für die Gesamtbeurteilung darstellt. Unabhängig davon kann man aber aus den IT-Budgets ableiten, dass die Softwarewartung einen erheblichen Kostenblock

darstellt und das Thema wegen der wachsenden Softwarebestände an Bedeutung zugenommen hat. Die Forschung lieferte in dieser Phase viele Einsichten wie z. B. den nur „leichten“ Zusammenhang zwischen Code-Länge, Komplexität und Wartungskosten. Es handelt sich aber überwiegend um Erkenntnisse, die die Programmebene betreffen und die Ergebnisse wurden in kein Gesamtmodell integriert.

4 Dritte Phase: Die Jahre 2000 bis 2020

Der Übergang zu dieser Phase war aus Sicht der Softwarewartung durch zwei große Herausforderungen geprägt, die Y2K-Problematik und die Währungsumstellung infolge der Euro-Einführung. Beides konnte überraschend gut bewältigt werden. Mit dem Jahrtausendwechsel veränderte sich aber auch das „Software-Ecosystem“ immer stärker. Dies ist der zunehmenden Bedeutung des Internets und der damit zusammenhängenden Umstellung der Softwarearchitekturen geschuldet. Mit der zunehmend agilen Softwareentwicklung war die kontinuierliche und teilweise automatisierte Auslieferung von neuen Releases oder Softwareupdates verbunden. Neben DevOps, automatisierter Softwareverteilung und -versionierung ist aber auch die zunehmende Bedeutung von KI-basierten Systemen, Micro Services und anderen Technologien zu beobachten, die aufgrund ihrer parallelen Existenz letztlich eine neue Sicht auf „Fehler“ und die Wartung von Software erfordern und Systeme, die gar nicht mehr isoliert voneinander betrachtet werden können.

Was die Softwarewartung betrifft, so erhebt sich die Frage, ob es hier zumindest aus der Perspektive der Forschung einen Neuanfang braucht, da sich nicht nur die technologische Basis vieler Softwarelandschaften in den Unternehmen stark verändert hat (Virtuelle Maschinen, Micro Services, Pipelines, DevOps, Cloud-Technologien etc.), sondern auch die in den Unternehmen dafür zuständige IT-Organisation. Die Eigenentwicklung spielt oft keine große Rolle mehr und der Fokus liegt bei Standardsoftware oder Open-Source-Lösungen. Damit verlagert sich aber auch die Wartung, die teilweise nur mehr als Service-Leistung in Anspruch genommen wird.

Es kommt damit zwar zu einer Externalisierung der Wartungsleistung, aber die Kosten bleiben bestehen. Die Kosten für die Softwarewartung machen Schätzungen zufolge angeblich 60 % bis 70 % der gesamten jährlichen Softwarekosten in Unternehmen aus. Mit Blick auf die weiter oben präsentierten Ergebnisse könnte das zwar zu hoch gegriffen sein, von einem erheblichen Kostenblock im Bereich von 30 bis 40 Prozent ist aber dennoch auszugehen. Am Beispiel der Firma SAP im Jahr 2007 kann dies verdeutlicht werden, wo man einen Umsatz von rund 3,9 Mrd. Euro mit der Wartung von Software mit ca. 48 000 Kunden erwirtschaftete. Ein Unternehmen, das SAP-Software einsetzt, gibt im Durchschnitt über 80.000 Euro Wartungskosten pro Jahr aus. Längst übersteigt der Wartungsumsatz von Softwarehäusern die Lizenzeinnahmen (eine Entwicklung die an Automobilhersteller und die Einnahmen aus dem Ersatzteilbereich erinnert). Es ist im Übrigen wegen des hohen Anteils am Geschäftsvolumen kein Wunder, dass Softwarehäuser zunehmend empfindlich auf alles reagieren, was die kontinuierlichen Einnahmen

aus der Wartung schmälern könnte. Angesichts des immer schwierigeren Lizenzgeschäfts dürfte sich die Situation für die Hersteller von Software in Zukunft aber eher verschärfen als entspannen.

Nicht wenige Unternehmen der IT-Branche nehmen mit Wartung mehr Geld ein als mit dem Verkauf von Softwarelizenzen. Für Softwarehäuser stellt die Wartung somit eine wichtige Einnahmequelle dar. Unterstrichen wird dies durch eine Forrester-Studie zum Umsatzanteil der Wartungskosten bei ausgewählten Softwarehäusern (vgl. Abbildung 4).

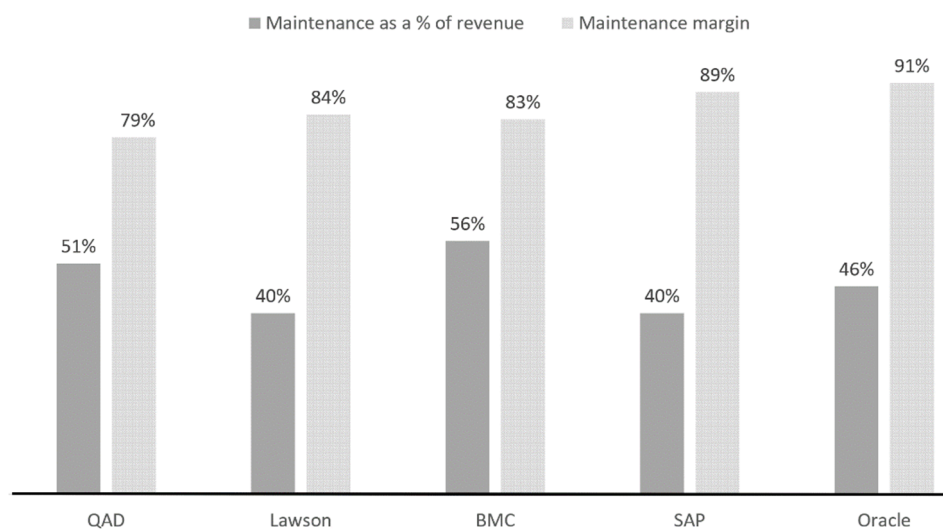


Abb. 4.: Einnahmen aus dem Wartungsgeschäft (Forrester Research 2008)

Wichtig ist die Berechnungsgrundlage (Listenpreis oder tatsächlicher Kaufpreis), wobei aber auch zwischen Produkt- bzw. Systembezug und Projektbezug unterschieden werden kann. In der Regel berechnet sich die Softwarewartung nach dem Lizenzpreis. Zugeständnisse bei Lizenzpreisen werden oft später mit der Wartungsgebühr „bezahlt“. Aus Sicht der Wirtschaftsinformatik entsteht das Problem in den Unternehmen durch ein unzureichendes IT-Controlling und die fehlende Kostentransparenz. Dies macht noch einmal die Notwendigkeit von genauen Daten deutlich, ohne welche gezielte Maßnahmen in Verbindung mit der Softwarewartung nicht getroffen werden können. Erinnert werden muss in diesem Zusammenhang aber auch an die rechtliche Dimension der Wartung. Diese ist zwar wissenschaftlich und auch juristisch eindeutig geklärt, führt in der Praxis aber oft zu Problemen, weil unübersichtliche Preis- oder Lizenzmodelle nicht immer eine klare Trennlinie zu ohnehin geschuldeten Leistungen erkennen lassen.

Damit verändert sich das Untersuchungsfeld grundsätzlich. Umfangreiche Softwarewartung im herkömmlichen Sinne findet man nur mehr in wenigen Unternehmen, sondern primär bei Softwarehäusern und IT-Service Providern. Bei den Wartungskosten in den Unternehmen würde man durch die Externalisierung des Programmieraufwands eigentlich

eine deutliche Reduktion des damit verbundenen Kostenanteils erwarten. Leider gibt es nach wie vor kaum belastbare Zahlen zu diesem konkreten Themenbereich. Aus den spärlichen Einzelberichten gewinnt man allerdings den Eindruck, dass sich weder an der absoluten Höhe noch am relationalen Kostenanteil viel geändert hat.

In dieser Phase ist auch festzustellen, dass IT-Abteilungen allmählich wieder kleiner werden, das Thema IT-Organisation aber neu an strategischer Bedeutung gewinnt. Aktuelle Themen, von denen sich Unternehmen wirtschaftlich relevante Impulse erwarten, heißen Blockchain, IoT und Industrie 4.0, Cloud Computing, DevOps, mobile Apps, Big Data, Künstliche Intelligenz (KI) u. a. m. Um mit der digitalen Innovation Schritt zu halten, müssen teilweise neue Organisationsformen gefunden werden. Man spricht von agilen Organisationsformen und im Extremfall führt dies in der Unternehmenspraxis zu einer Aufteilung der IT-Organisation in zwei weitgehend unabhängige Gruppen (interne Verwaltungssysteme vs. kundenorientierte und internetbasierte Lösungen). Wartung fällt in beiden Welten an und es ist eine neue Herausforderung, wie diese in Zeiten agiler SW-Entwicklung organisatorisch gelöst werden soll.

Zusammenfassend ist festzustellen, dass Unternehmen vor einer völlig veränderten Situation stehen, die von der Wissenschaft aus der Wartungsperspektive bisher nur ungenügend behandelt worden ist. Durch die weitgehende Externalisierung vieler softwarebezogener Aufgaben könnte man etwas voreilig zu dem Schluss kommen, dass die Wartungskosten ohnehin keine Bedeutung mehr haben und diese Kosten zu einem Budgetposten geworden sind. Die Abhängigkeit von komplexen Systemen und auch von externen Anbietern legt allerdings eine andere Haltung nahe. Mit dem strategischen IT-Management bzw. IT-Governance wurden die Weichen aus Managementsicht bereits gestellt. In diesen Modellen findet sich das Wartungsthema fast schon erwartungsgemäß aber wieder nur als Randthema. Dies ist weniger der Kostenrelevanz als der nach wie vor unzureichenden Datenlage geschuldet.

5 Metaanalyse zur Entwicklung des Wartungskostenanteils

Eine zentrale Frage, die mit diesem Beitrag beantwortet werden sollte, lautete: Hat sich die Situation mit Blick auf die Höhe der SW-Wartungskosten in Unternehmen verbessert (z.B. deutliche Reduktion der Wartungskosten) oder nur verändert? Auf Basis der bisherigen Analysen lautet die Antwort: Die Kosten in Verbindung mit der Wartung haben sich für die Unternehmen offenbar nicht verringert, die dahinterstehende Situation bzw. die Ursachen haben sich aber extrem verändert. Große Teile der IT-Ausgaben werden damit durch Entscheidungen der Vergangenheit bestimmt und die Wartungskosten nehmen einen unverändert hohen Stellenwert ein. Dies wird bestätigt durch das Ergebnis einer Metaanalyse, die 2020 als Längsschnittuntersuchung vom Lehrstuhl für Wirtschaftsinformatik an der Universität Passau durchgeführt worden ist. Dazu wurden die veröffentlichten Daten zum Anteil der Wartungskosten aus den letzten 40 Jahren verwendet. Die dabei berücksichtigten Quellen finden sich in [Le20] aufgelistet.

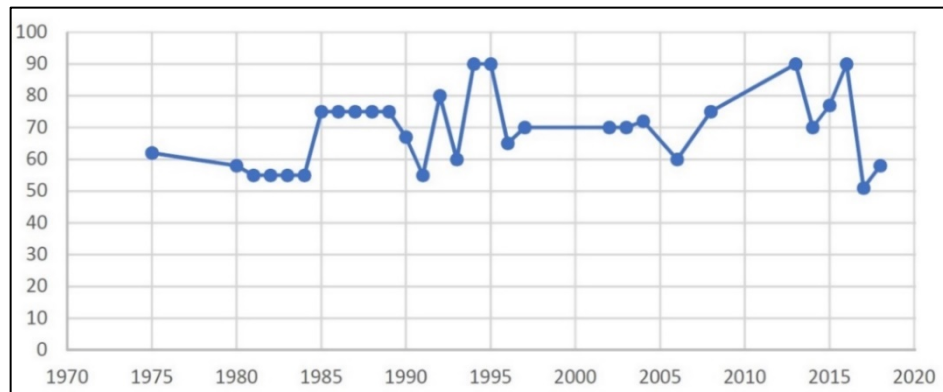


Abb. 5: SW-Wartungs- vs. Entwicklungskosten – Entwicklung seit den 70er-Jahren

Abbildung 5 gibt das Verhältnis zwischen SW-Wartungs- und Entwicklungskosten im Zeitverlauf wieder, und zwar auf Basis der einschlägigen Fachliteratur. In die Untersuchung wurden 30 Einzelpublikationen einbezogen, die eine relativ große Spannweite zeigen, welche sich wiederum mit den dahinterstehenden Basisdaten erklärt. Die Streuung der Einzelwerte und ihre Spannweite ergibt sich u.a. durch die einbezogenen Unternehmen (Beratungsunternehmen haben z. B. niedrigere Wartungskosten), die Verwendung selektiver Daten (z.B. ausschließlich der COBOL-Programmbestand, weil nur dafür Aufzeichnungen existierten), aber vor allem durch die Mittelwertbildung (in vielen Veröffentlichungen fehlen die Basisdaten und es mussten aus vorliegenden Mittelwerten neue Mittelwerte gebildet werden). Auch die Jahreszahl der Veröffentlichung entspricht nicht immer dem repräsentierten Beobachtungszeitraum. Man kann daher die Zahlen nicht für die Ermittlung eines Entwicklungstrends verwenden, sondern nur als Beleg für die unverändert hohe Bedeutung der Wartung. Die Zahlen decken sich im Übrigen auch mit der Prognose einer „Legacy Crisis“ bei Seacord et al. [Se03], der damit die Notwendigkeit der Modernisierung des Softwarebestands begründet.

Um eine bessere und vereinheitlichte Übersicht über eine mögliche Entwicklung zu erhalten, wurde eine Aufteilung auf 5-Jahresabschnitte (vgl. Abb. 6) vorgenommen. Diese zeigt zwar leichte Zu- und Abnahmen, aber in der Gesamtsicht ein relativ konstantes Bild über den langen Zeitraum von 40 Jahren. Da die Studien aber nicht aufeinander aufbauen und zudem auf sehr unterschiedlichen Daten beruhen, kann das Ergebnis trotzdem nicht für eine seriöse Trendanalyse verwendet werden. Man sieht allerdings an dem Entwicklungsverlauf neben der unveränderten Bedeutung der Softwarewartung die Größenordnung der damit verbundenen Kosten.

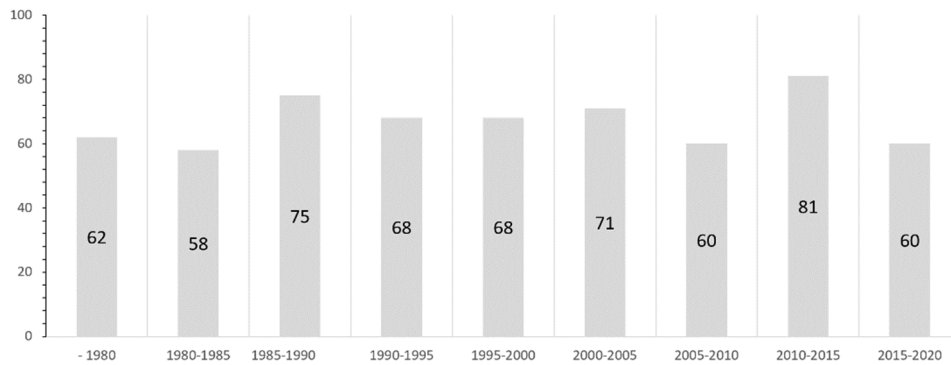


Abb. 6: Anteil der Wartungskosten in 5-Jahresperioden

Die Ergebnisse sind konsistent mit den Analysen von Kappelman et al. ([Ka18], [Ka20]), in der für die Softwarewartung in den letzten Jahren ebenfalls ein nahezu konstant hoher Aufwand ausgewiesen wird (vgl. Abb. 7).

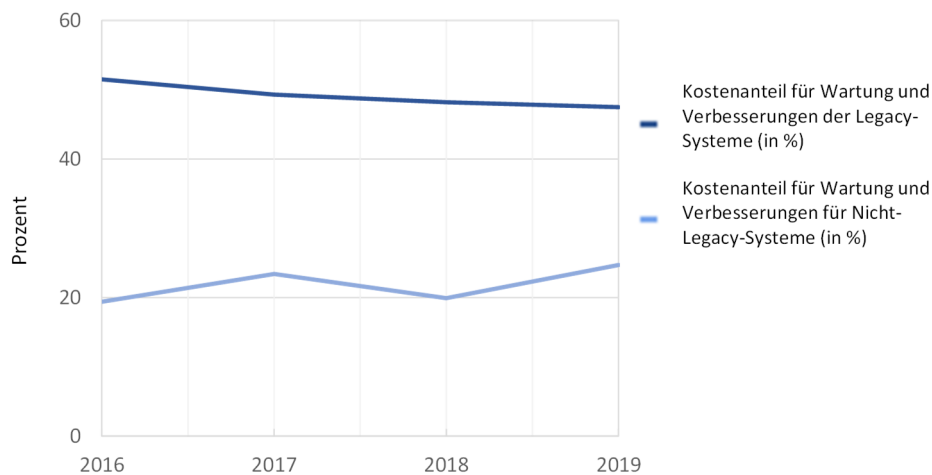


Abb. 7: SIM IT-Trendstudien zum Anteil der Wartungsausgaben an den IT-Kosten

Datengrundlage sind Umfragen im Rahmen der „SIM IT Issues and Trends“-Studien, wobei Daten für die Jahre von 2016 bis 2019 vorliegen. Befragt wurden IT-Executives und CIOs, welche Kategorien zu den drei kostenintensivsten Faktoren in Verbindung mit der eingesetzten Software im Unternehmen gehören. Dabei zeigte sich, dass die Hälfte aller Befragten die Wartung ihrer Legacy-Systeme als einen der drei größten Kostenfaktoren im IT-Budget sieht. Zudem sagte etwa ein Viertel der Befragten, dass die Wartung der Systeme auch abseits der Legacy Systeme einer der drei größten Faktoren sei. Aus diesem Grund wurde in der Ergebnisdarstellung zwischen Legacy- und Non-Legacy-Systemen differenziert. Summiert man die beiden Prozentsätze auf, so entspricht der

Anteil den in der Metaanalyse erhobenen Werten (vgl. Abb. 5 und 6). Zwar lässt sich erkennen, dass die Wartung der Legacy Systeme im Verlauf der letzten Jahre zumindest unter den Befragten an Bedeutung verloren hat. Bei der Wartung der restlichen Systeme ist aber das Gegenteil der Fall, sodass die Gesamtsituation unverändert bleibt.

6 Zusammenfassung und Schlussfolgerungen²

Es besteht kein Zweifel, dass Fragen und Probleme rund um die Softwarewartung seit Beginn der Diskussion mehr und komplexer geworden sind. Die Ergebnisse der Metaanalyse belegen aber, dass die Wartungskosten einen unverändert hohen und signifikanten Teil der Softwarekosten ausmachen. Die Relevanz des Themas aus wirtschaftswissenschaftlicher Sicht zeigt sich auch in aktuellen Veröffentlichungen (vgl. dazu z. B. [De17], [Me19], [Ka20], [Ho18]). Die Bedeutungszunahme gilt natürlich auch für die korrespondierende Praxis, in der die Softwarewartung noch immer ein relevantes Thema bildet, allerdings verglichen mit den 80er Jahren in einem völlig anderen Kontext. Entscheidend ist heute die geschäftskritische Bedeutung der Software. Die traditionelle Wartung ist aus den Unternehmen verschwunden, trotzdem zahlen diese weiter für eine Leistung, die sie nicht kontrollieren können und die möglicherweise eine Funktionalität betrifft, die gar nicht genutzt wird. Auf die Unternehmen werden möglicherweise viel zu hohe Wartungskosten abgewälzt, wobei sich durch die Informationsasymmetrie zwischen Nutzern und IT-Dienstleistern das Machtungleichgewicht inzwischen verfestigt hat. Möglich ist aber auch der umgekehrte Prozess, d. h. die Wartung eines Systems wird vernachlässigt. In der Fachliteratur hat sich dafür der Begriff „technical debt“ durchgesetzt. Wenn beispielsweise Qualitätsstandards vernachlässigt werden, um Software möglichst schnell funktionsfähig zu machen, so werden zunächst zwar Kosten und Ressourcen eingespart, diese technischen Schulden müssen aber zu einem späteren Zeitpunkt beglichen werden ([Al16], vgl. auch [Ho18]).

In der Unternehmenspraxis wird die Aufgabe inzwischen zwar als Teilaufgabe des IT-Managements verstanden, die Scientific Community, die sich mit dem Thema befasst, ist eine eher technisch interessierte Gemeinschaft geblieben. Ein Perspektivenwechsel ist dabei auf jeden Fall erforderlich, da der Untersuchungsgegenstand nicht länger ein einzelnes Softwaresystem ist. Noch einmal unterstrichen werden soll, dass die große Zahl der Anwender (und zwar Profit- und Non-Profit-Organisationen gleichermaßen) kaum mehr Software entwickelt. Wartungsaufgaben im engeren Sinne betreffen daher mehrheitlich Softwarehäuser, Wartungskosten fallen aber auch bei den Anwendern in einem hohen Umfang an und werden über Wartungsverträge und ähnliche Konstrukte verrechnet. Die teilweise beobachtbaren Konzentrations- und Konsolidierungstendenzen in der Softwareindustrie können sich möglicherweise auch zum Nachteil kleinerer Softwarehäuser auswirken, ganz sicher aber zum Nachteil kleinerer und mittlerer Unternehmen.

² Die ausführlichere Langfassung des Beitrags ist unter den Seiten des Fachausschusses Management der Anwendungsentwicklung und –wartung unter <https://t1p.de/2cc5> verfügbar.

Mit einer Externalisierung der Wartung werden natürlich auch Entscheidungen über die Weiterentwicklung der Systeme ausgelagert. In Abwandlung eines bekannten Satzes von Nicolas Carr („IT doesn't matter“) könnte man sagen „SW Maintenance Management Matters“. Wer die Hoheit über die SW-Wartung verliert, verliert letztlich auch die Kontrolle über die betrieblichen Abläufe und das wäre in einer Phase der Digitalisierung fatal! Unternehmen brauchen daher auf jeden Fall eine IT-Strategie, die die Wartungsthematik berücksichtigt. Es geht also nicht mehr um die Frage, was Wartung überhaupt ist und was dazu gerechnet werden soll, sondern um Wartungskonzepte für eine äußerst heterogen und komplex gewordene Softwarelandschaft bei teilweise verteilter oder sogar unklarer Zuständigkeit und Verantwortung.

Damit rückt die IT-Landschaft in den Fokus, denn sie entfernt sich im Lauf der Zeit auf ganz natürliche Weise immer weiter von der ursprünglich geplanten IT-Architektur (z. B. durch moderne Schnittstellen wie Webservices, aber auch neue Anforderungen wie Compliance-Vorgaben oder Energieeffizienz), weil oft lange an veralteten, aber geschäftskritischen Anwendungen festgehalten wird, was wiederum die Agilität in Verbindung mit zukunftsorientierten Geschäftsmodellen reduziert. Vielfach scheint die Sorge zu sein, durch den Fokus auf Altes (also SW-Wartung) die Innovation (also wichtige Neuerungen) zu versäumen. Ein Blick auf das große Ganze könnte vielen Unternehmen dabei helfen, denn es kommt im Management zu einer viel zu engen Fokussierung auf einzelne Indikatoren für Innovation wie Agilität oder Digitalisierungsgrad. Und IT-Management - als ganzheitliche Aufgabe begriffen - führt zu Innovation. Definitiv fehlt derzeit eine unternehmensweite Gesamtbetrachtung. Durch die stärkere Verbreitung von Home-Office kommt noch eine weitere Dimension hinzu, die zumindest rechtliche und organisatorische Lösungen erfordert. Hier kann nicht automatisch von einer Abdeckung durch bestehende Lizenzen und Verträge ausgegangen werden.

Abschließend sei auch noch die wachsende Zahl an privaten Softwarenutzern erwähnt. Dieser Markt ist enorm im Wachsen begriffen und beschränkt sich schon lange nicht mehr auf den Spielesektor. Das Verhältnis zwischen den Softwareentwicklern und dem meist anonymen Benutzerkreis ist in diesem Umfeld etwas anders definiert als bei betrieblichen Anwendungen. Zu den benutzten Produkten (häufig über App-Stores bezogen) gehören sowohl Gratissoftware als auch bezahlte Apps. Das Thema Wartung ist dafür noch nicht aufgearbeitet worden. Mit Blick auf das stark veränderte Verhältnis zwischen der Softwareindustrie und ihren Kunden kann man plakativ fragen: Ist die Wartungssituation möglicherweise vergleichbar mit der „Dieselproblematik“? Wir haben es mit geschlossenen und häufig proprietären Systemen zu tun, die von den Anwendern nur mehr benutzt, aber nicht mehr gestaltet werden können (Lizenzen, aber keine Eigentumsrechte). Unternehmen sollten aber nicht an der Wartung, sondern durch die Wartung sparen, und zwar mit einem Wartungsmanagement, das am Nutzen der damit verbundenen Maßnahmen orientiert ist. Intransparente Kosten sind dabei nicht hilfreich. Das bisher in der Wirtschaftsinformatik propagierte Modell eines Anwendungssystemmanagements (ASM) sollte durch ein systematisches SW-Asset-Management (SAM) abgelöst werden, d. h. durch eine ganzheitliche Sicht auf die IT-Anwendungslandschaft bzw. die IT-Architektur mit einem nachhaltigen Lizenz-, Release- und Wartungsmanagement.

Danksagung

Der Beitrag wurde in einer ersten Version beim „Harry Sneed 80th Anniversary Track“ im Rahmen der WSRE 2020 in Paderborn präsentiert. Das Feedback beim Workshop, für das an dieser Stelle den Teilnehmern gedankt wird, wurde in den Beitrag eingearbeitet.

Literaturverzeichnis

- [Al16] Alves, S. N. et.al.: Identification and management of technical debt - A systematic mapping study. *Information and Software Technology* 70/16, S. 100–121, 2016.
- [Am15] Ampatzoglou, A. et.al.: The financial aspect of managing technical debt - A systematic literature review. *Information and Software Technology* 64/15, S. 52–73, 2015.
- [Bu11] Buchmann, I. et.al.: Towards an Estimation Model for Software Maintenance Costs. In (Mens, T. et.al. Hrsg.): *Proceedings of the Euromicro Conference on Software Maintenance and Reengineering*. Oldenburg, S. 313 – 316, 2011.
- [De17] Delivering Value to Today's Digital Enterprise, bmc.com/content/dam/bmc/migration/pdf/Delivering-Value-to-Today%27s-Digital-Enterprise-FINAL.pdf, Stand: 13.08.21.
- [Go08] Golorath, D. D.: Software Total Ownership Costs: Development Is (Only) Job One, <http://nyspin.org/Dan%20Galarath%20-%20Development%20is%20only%20Job%201.pdf>, Stand: 12.08.21.
- [He13] Herraiz, I. et.al.: The evolution of the laws of software evolution. A discussion based on a systematic literature review. *ACM Computing Survey* 46/13, S. 1-28, 2013.
- [Ho18] Holvitie, J. et.al.: Technical debt and agile software development practices and processes: An industry practitioner survey. *Information and Software Technology* 96/18, S. 141–160, 2018.
- [Hu08] Hunt, B. et.al.: Software Maintenance Implications on Cost and Schedule. In (Logsdon, J.M. Ed.): *IEEE Aerospace Conference*. Big Sky, S. 1-6, 2008.
- [Ik18] Ikram, A. et.al.: Eliciting Theory of Software Maintenance Outsourcing Process - A Systematic Literature Review. *International Journal of Computer Science and Network Security* 18/18, S. 132-143, 2018.
- [JB12] Jones, C.; Bonsignour, O.: *The Economics of Software Quality*, Boston, 2012.
- [Jo06] Jones, C.: The economics of software maintenance in the twenty first century, citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.88.7697&rep=rep1&type=pdf, Stand: 15.07.21.
- [Jo95] Jorgenson, M.: An Empirical Study of Software Maintenance Tasks. *Journal of Software Maintenance* 7/95, S. 27-48, 1995.
- [Ka18] Kappelman, L. et.al.: The 2017 SIM IT Issues and Trends Study. *MIS Quarterly Executive* 17/18, S. 53-88, 2018.

- [Ka20] Kappelman, L. et.al.: The 2019 SIM IT Issues and Trends Study. MIS Quarterly Executive 19/20, S. 69-104, 2020.
- [KS97] Kemerer, C. F.; Slaughter, S. A.: Determinants of Software Maintenance Profiles - An Empirical Investigation. Journal of Software Maintenance 9/97, S. 235-251, 1997.
- [Ko03] Koskinen, J.: Software Maintenance Costs, <https://wiki.uef.fi/display/tktWiki/Jussi+Koskinen>, Stand: 08.08.21.
- [La82] Lawrence, M.: An Examination of Evolution Dynamics. In (IEEE, Hrsg.): Proceedings of the 6th International Conference on Software Engineering. Los Alamitos, S. 188-196, 1982.
- [Le80] Lehman, M. M.: Programs, Life Cycles and Laws of Software Evolution. Proceedings IEEE 68/80, S. 1060-1076, 1980.
- [Leh97] Lehman, M. M. et.al.: Metrics and laws of software evolution - the nineties view. In (IEEE, Hrsg.): Proc. 4th International Software Metrics Symposium (METRICS '97). Washington, S. 20-32, 1997.
- [Le96] Lehner, F.: Softwarewartung und Reengineering - Erfahrungen und Entwicklungen, Gabler Edition Wissenschaft, Wiesbaden, 1996.
- [Le90] Lehner, F.: Motivation und Zufriedenheit von DV-Mitarbeitern. Information Management, S. 24-33, 1990.
- [Le89] Lehner, F.: Nutzung und Wartung von Software, München, 1989.
- [Le95] Lehner, F.: Softwarewartung und Reengineering als Kernprobleme der Informatik. In (Huber-Wäschle, F., et.al., Hrsg.): Proceedings zur Jahrestagung der Gesellschaft für Informatik. Zürich, S. 552-561, 1995.
- [Le91] Lehner, F.: Softwarewartung. Management, Organisation und methodische Unterstützung, München, 1991.
- [Le20] Lehner, F.: Haben die SW-Wartungskosten an Bedeutung verloren? Beitrag zur WSRE 2020, Paderborn, Diskussionsbeitrag W-45-20, Schriftenreihe Wirtschaftsinformatik, Universität Passau, 2020.
- [LS04] Lehner, F.; Sneed, H.: Braucht die Wirtschaftsinformatik ein eigenständiges Curriculum für Software-Engineering? Wirtschaftsinformatik 46/04, S. 491-494, 2004.
- [LS04a] Lehner, F.; Sneed, H.: Software Engineering für Wirtschaftsinformatiker – Praxisorientierte Analyse des Lehrangebots und Entwurf eines Curriculums. Passauer Diskussionspapiere, Schriftenreihe Wirtschaftsinformatik, Diskussionsbeitrag W-01-04, Universität Passau, 2004.
- [LST78] Lientz, B. P.; Swanson, E. B.; Tompkins, G. E.: Characteristics of Application Software Maintenance. Communications of the ACM 21/78, S. 466-471, 1978.
- [MF00] Maxwell, K.; Forselius, P.: Benchmarking Software Development Productivity. IEEE Software 17/00, S. 80-88, 2000.
- [MF01] Maxwell, K.; Forselius, P.: Cost drivers of annual maintenance - One commercial bank's experience. ESCOM, S. 233-242, 2001.

- [Me19] Mehrizi, M. H. R. et.al.: Intensifying to Cease. Unpacking the Process of Information Systems Discontinuance. *MIS Quarterly* 43/19, S.141-165, 2019.
- [Pa94] Parnas, D. L.: Software Aging. In (IEEE, Hrsg.): *Proceedings of the 16th International Conference on Software Engineering*. Sorrento, S. 279–287, 1994.
- [Ra09] Rashid, A. et.al.: Gauging the Differences between Expectation and Systems Support - the Managerial Approach of Adaptive and Perfective Software Maintenance. In (IEEE, Hrsg.): *4th International Conference on Cooperation and Promotion of Information Resources in Science and Technology (COINFO)*. Beijing, S. 45-50, 2009.
- [Re11] Ren, Y. et al.: Research on Software Maintenance Cost of Influence Factor Analysis and Estimation Method. In (Wang, C.Z., Ye, Z.W., Eds.): *3rd International Workshop on Intelligent Systems and Applications*. Wuhan, S. 1-4, 2011.
- [Se03] Seacord, R. C. et.al.: *Modernizing Legacy Systems: Software Technologies, Engineering Processes, and Business Practices (SEI Series in Software Engineering)*, Addison-Wesley, Boston, 2003.
- [Sn05] Sneed, H. M. et.al.: *Software-Produktmanagement - Wartung und Weiterentwicklung bestehender Anwendungssysteme*, Heidelberg, 2005.
- [Sn95] Sneed, H. M.: Estimating the Costs of Software Maintenance Tasks. In (IEEE, Hrsg.): *Proceedings of International Conference on Software Maintenance*. Nice, S. 168-181, 1995.
- [Sn08] Sneed, H. M.: Industrialisierung der Softwarewartung & -Weiterentwicklung Management der Anwendungsentwicklung und -Wartung im Fachbereich Wirtschaftsinformatik (WI-MAW). In (Herzwurm G., Mikusz, M. Hrsg.): *Industrialisierung des Software-Managements - Fachtagung des GI-Fachausschusses Management der Anwendungsentwicklung und -Wartung im Fachbereich Wirtschaftsinformatik (WI-MAW)*. Stuttgart, S. 139-152, 2008.
- [Sn04] Sneed, H. M.: A Cost Model for Software Maintenance & Evolution. In (IEEE, Hrsg.): *Proceedings of the 20th IEEE International Conference on Software Maintenance (ICSM'04)*. Washington, 2004.
- [SS13] Sneed, H. M., Seidl, R.: *Softwareevolution*, Heidelberg, 2013.
- [Za04] Zarnekow, R. et.al.: Untersuchung der Lebenszykluskosten von IT-Anwendungen. *Wirtschaftsinformatik* 46/04, S. 181–187, 2004.