

Entwicklung einer Methode zur Priorisierung von Anforderungen unter Berücksichtigung von Synergieeffekten

Marcus Grum¹ und Christian Glaschke²

Abstract: Die Priorisierung von Anforderungen nimmt einen hohen Stellenwert im EA-Management ein. In verschiedenen Bereichen und auf sehr unterschiedlichen Ebenen müssen strategische, fachliche und wirtschaftliche Anforderungen bewertet und dadurch priorisiert werden. Dieser Beitrag zeigt eine neue Methode, die Anforderungen und deren Wirkung auf das System als Netzwerk begreift und somit Synergieeffekte zwischen Unternehmensbereichen aufnimmt, bewertet und in den Entscheidungsprozess einfließen lässt.

Keywords: Projektmanagement, Anforderungspriorisierung, Synergieeffekte

1 Einleitung

Informationstechnologien (IT) und Informationssysteme (IS) werden in immer stärkerem Maße dazu verwendet, innovative Produkte, Dienstleistungen und Geschäftsmodelle zu realisieren, indem Fachbereiche innerhalb und über die Unternehmensgrenzen hinaus kooperieren, solche Innovationen gemeinsam konzipieren und auf den Weg bringen. Mit dem Ziel der Ausrichtung an den strategischen Zielen eines Unternehmens ist es Aufgabe des Enterprise Architecture Managements (EAM), die Menge an Projekten zu identifizieren, die notwendig sind, um die Anwendungslandschaft eines Unternehmens auf Basis einer definierten Informationsarchitektur in einen erstrebenswerten Soll-Zustand zu überführen [D06].

Es ergibt sich somit für IT-Organisationen die Notwendigkeit, proaktiv und frühzeitig mit den Fachbereichen zu kooperieren und vorrangig in der Bewertung und Priorisierung der Anforderungen an IT, IS und an hierfür zu realisierende Projekte zusammen zu arbeiten, sodass die Entscheidung, welche dieser Anforderungen angesichts begrenzter betrieblicher Ressourcen zunächst realisiert werden, rational begründet werden kann. Hierdurch erhöhen sich die Chancen auf einen Erfolg in der Realisierung der zu entwickelnden Objekte und es wird ein Beitrag an der effizienten Gestaltung eines EAM geleistet.

Ein aktueller Stand zeigt, dass bisherige Ansätze hierfür rein wirtschaftliche

^{1,2} Universität Potsdam, Lehrstuhl für Wirtschaftsinformatik, inbs. Prozesse und Systeme,
August-Bebel-Str. 89, 14482 Potsdam

¹ marcus.grum@lswi.de

² christian.glaschke@lswi.de

Betrachtungen fokussieren, wie beispielsweise die ROI-Analyse oder stark quantitative Bewertungsansätze von Requests, und für eine Priorisierung relevante Aspekte nicht umfassend betrachten. Die identifizierten Forschungslücken zeigen den Bedarf einer Ausweitung des Verständnisses bzgl. des Zusammenhangs zwischen Aufwand und Nutzen, sodass Vor- und Nachteile von Abteilungen nicht nur isoliert betrachtet werden.

Mit dem Ziel, die Anforderungsrealisierung als Teil einer Wertschöpfungskette zu begreifen und eine optimierte Reihenfolge von Anforderungen im Sinne eines Rankings zu erstellen, gilt es, die integrativen, abteilungsübergreifenden Aspekte eines Anwendungssystems zu berücksichtigen. Da Anforderungen sich auf verschiedene Abteilungen auswirken, können mehrfach positive und auch negative Synergieeffekte entstehen. Diese sollen innerhalb eines Bewertungsmodells berücksichtigt werden und für die Erstellung eines Rankings genutzt werden.

Österle [ÖBF10] bezeichnen den Prozess der Priorisierung als sozio-technische Aktivität, die Mensch, Technik und Organisation umfasst und nach Brockmann [B14] in ein prozessuales Vorgehen integriert werden muss. Somit wird für das Bewertungsmodell eine Durchführungsmethode geschaffen, welche diese Aspekte berücksichtigt.

Die Potenziale dieses neuen Lösungsansatzes sollen dabei helfen, dem Erfolgsdruck auf das Projektmanagement gerecht zu werden und sicherstellen, dass die wichtigsten Anforderungen bearbeitet und der Projektumfang effektiv und effizient festgelegt werden kann. Weitere positive Eigenschaften einer effektiven und effizienten Priorisierung zeigen Regnel [R01]. An theoretischen Beispielen soll gezeigt werden, wie Synergieeffekte in mehrdimensionalen Aufwand/Nutzen-Verhältnissen ermittelt, bewertet und attraktive Projekte ausgewählt werden können, die ein effizientes EAM betreffen.

2 Aktueller Stand der Forschung

Um eine Ausgangsbasis für das Definieren einer neuen Priorisierungsmethode für EAM-Projekte zu schaffen, wird in einem ersten Unterkapitel das EAM zunächst definiert und im Anschluss eine Forschungslücke anhand des aktuellen Stands von Entscheidungsfindungsmodellen gezeichnet.

2.1 Definition eines Enterprise Architecture Managements

Definitionen für den Begriff des Enterprise Architecture Managements (EAM) existieren in großer Zahl, Heterogenität, Detaillierungsgrad, Reichweite der inkludierten Bestandteile und diverse Ausprägungen in verschiedenen Branchen [CD17]. Lankhorst [L05] definiert das EAM als kohärentes Ganzes von Prinzipien, Methoden und Modellen, die für die Realisierung von Organisationsstrukturen, Geschäftsprozessen,

Informationssystemen und Infrastrukturen zum Einsatz kommen. Ahlemann [ASML17] zu Folge umfasst das EAM somit eine Management-Philosophie, eine organisationale Funktion, eine Methodologie sowie eine Kultur. Dern [D06] zu Folge setzen sich die betrachteten Elemente des EAM, in dem Sinne die zu steuernde Enterprise Architecture (EA), aus Bestandteilen zusammen, wie sie in Abbildung 1 zur linken Seite als Ebenen einer Architekturpyramide des EAM ersichtlich sind und nachfolgend beschrieben werden. Während die linke Seite den Ist-Zustand der EA nach Dern [D06] repräsentiert, zeigt die rechte Architekturpyramide den definierten Soll-Zustand, der mit Hilfe von Projekten des Projektportfolios erreicht werden soll. Dass dieselben Bestandteile in einen optimierten Zustand überführt werden, ist mit dem asterisk gekennzeichnet, sodass die vorliegende Ausarbeitung ein projektorientiertes EAM fokussiert.

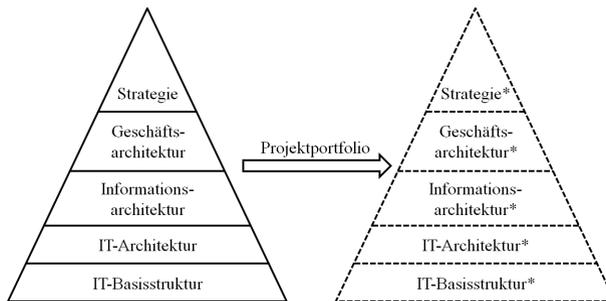


Abb. 1: Die projektbasierte Optimierung der Bestandteile des EAM

Die Entwicklung der Unternehmensarchitektur erfolgt auf Basis der Strategie des Unternehmens. Zumeist wird diese in Unternehmenszielen operationalisiert, Geschäftsfelder folgen der Ausrichtung des gesamten Unternehmens. Definitionen der Geschäftsarchitektur fokussieren auf die Architektur von Geschäftsprozessen, sodass sich der Informationsbedarf und für die Prozessdurchführung benötigte Informationen in der Aufbau- und Ablauforganisation des Unternehmens widerspiegeln. Auf dieser aufbauend erfolgt die wertorientierte Gestaltung der Anwendungslandschaft in der Informationsarchitektur, sodass eine geschäftliche und technologische Integration realisiert wird. Zum Beispiel zählt hierzu auch die Einführung eines Enterprise Resource Planning Systems (ERP). In der IT-Architektur wird eine strukturierende Abstraktion von IS geschaffen, um die Verständigung von IS-Entwicklungen zu vereinfachen. Hardware- und grundlegende Softwarekomponenten, die für die Laufzeitumgebung, Steuerung und für Entwicklungsprozesse benötigt werden, werden in der IT-Basisstruktur definiert.

In Anlehnung an dieses Architekturmodell enthält das Projektportfolio die Menge aller Projekte, die auf Basis des definierten Ist-Zustands einer Informationsarchitektur in einen definierten Soll-Zustand überführen. Als Stellschraube für Optimierungen gilt es ein Entscheidungsmodell zu finden, welches begrenzte Ressourcen für die Realisierung von Projekten und das Betreiben des realisierten Soll-Zustands (eingesetzte Leistung) angesichts eines erzielten Nutzens optimiert.

2.2 Ansätze für Entscheidungsfindungsmodelle

Um die Forschungslücke der vorliegenden Ausarbeitung zu identifizieren, wurden grundlegende Ansätze für Entscheidungsfindungsmodelle sowie weitere Techniken zur Priorisierung hinsichtlich später definierten, vier Untersuchungszielen analysiert. Betrachtete Ansätze finden sich in Spaltenüberschriften der Tabelle 1, nebenstehend finden sich korrespondierende Quellen. Ansätze für Entscheidungsfindungsmodelle fanden sich in vielen Disziplinen, wie der Psychologie, der Organisationstheorie, dem Requirement Engineering, etc. und werden nicht einzig für das Software Engineering eingesetzt.

Ähnliche Untersuchungen fanden sich in nachfolgenden: Aurum [AW03] zeigen hier eine Zuordnung von Entscheidungsfindungsmodellen zu diversen Anforderungspriorisierungsmethoden. Berander [BA05] klassifizieren Techniken anhand der in der jeweiligen Technik verwendeten Messskala, Granularität und der für ihren Einsatz benötigten Erfahrung. In der Betrachtung stehen die Techniken AHP gemäß der Literatur ([R01], [SHM92]), Cumulative Voting (Hundert-Dollar-Test) gemäß Leffingwell [LW00], Ranking via Sortieralgorithmen wie Karlsson [KWR98], Numerical Assignment gemäß IEEE [I98]. Ebenso stehen hier das Top-Ten-Requirements nach Lausen [L02] und Kombinationen aus diesen Techniken. Greer [G05] gruppiert Techniken anhand der Vergleichsrelationen. Relative Einschätzungen leiten ein Ranking anhand eines relativen Vergleichs der Anforderungen untereinander ab. Hierunter fallen Techniken wie das AHP nach Saaty [S80] oder Sortieralgorithmen wie das Bubblesort, dem Minimalen Spannbaum und dem „Binären Suchbaum“, das „Kartensortieren“ und das „Laddering“ ([MR96], [RM97]). Absolute Einschätzungen leiten ein Ranking anhand eines Vergleichs bzgl. fester Kriterien ab. Hierunter fällt beispielsweise das SERUM nach [GBS99]. Sofern rechnerbetonte Kosten als Vergleichskriterien herangezogen werden, können hierzu ebenfalls das Simulated Annealing, die Tabu-Suche und Genetische Algorithmen herangezogen werden, die in Analogie zu biologischen Prozessen der Natur arbeiten ([CS01], [H75]). Hierzu zählt das EVOLVE [GR04]. Keine dieser Untersuchungen bzw. Techniken betrachtete die Verwendung von Nutzen- und Aufwandsdimensionen als Untersuchungsziel, welche in Tabelle 1 in den ersten beiden Zeilen zu finden sind. Des Weiteren wurde die Verwendung von Synergieeffekten zwischen Anforderungen und zwischen den Abteilungen bisher nicht berücksichtigt, die in den Zeilen drei und vier zu finden sind. Als Definition für Synergieeffekte wird somit folgende zugrunde gelegt:

Definition: Der Synergieeffekt zwischen den Abteilungen fokussiert positive und negative Effekte, die bei der Realisierung einer Anforderung in den Abteilungen erzielt werden.

Aus der Definition des Synergieeffektes ergibt sich eine klare Abgrenzung zu den von Aaqib [AFS09] untersuchten Abhängigkeiten zwischen Anforderungen, sodass in Tabelle 1 entsprechend des Untersuchungsziels eine Forschungslücke wie nachfolgend beschrieben deutlich wird.

3 Der Synergieeffekt

Die dem *SynergieEffekt* zugrundeliegende Idee ist, dass zur Identifizierung von strategisch bedeutsamen Anforderungen mitunter Synergieeffekte dienen können, die in den Abteilungen entstehen, wenn die Anforderungen realisiert werden. Im Folgenden wird davon ausgegangen, dass jede Anforderung in Form von unabhängigen Projekten abgewickelt wird.

Um die Synergieeffekte zu identifizieren, werden Abteilungen, die mindestens aus einer Person bestehen und innerbetrieblich, aber auch über die Grenzen der Organisation verteilt vorliegen, als Teile eines Netzwerks begriffen. Für diese können spezifische Aufwände der Abteilungen den spezifischen Nutzen gegenübergestellt werden. Somit wird deutlich, welche Abteilungen arbeiten müssen und welche Abteilungen von dieser Arbeit profitieren.

Weil Aufwand- und Nutzenpotenziale der Anforderungen in bisherigen Score-Verteilungen nur im Gesamten betrachtet wurden, blieben Synergieeffekte bisher vollständig unbeachtet. Es ist eine Lösung zu finden, die den folgenden Anforderungen genüge tut:

- Projekte mit größeren Summen an Nutzenwerten sollen bevorzugt werden.
- Projekte mit größeren Summen an Arbeitsaufwänden relativieren Nutzenwerte.
- Abteilungen konkurrieren nicht um Nutzenwerte oder Arbeitsaufwände.
- Geringere abteilungsübergreifende Arbeitsbeteiligungen sind zu bevorzugen.
- Größere abteilungsübergreifende Arbeitseinsparungen sind zu bevorzugen.
- Projekte, die einen Schaden in Abteilungen erzeugt sind im Vergleich zu Projekten, die die gleiche Summe an Nutzen- und Aufwandspunkten erzeugen, zu benachteiligen.

Diesen Anforderungen stellen sich die nachfolgenden Ausführungen. Zu identifizieren sind für jedes Projekt zwei Matrizen mit jeweils l Elementen, wobei l für die Anzahl der Abteilungen in der betrachteten Unternehmenseinheit steht. Zum ersten ist eine Aufwandsmatrix zu identifizieren, in welcher jedes Element für den Aufwand einer Abteilung l steht, der auf diese zurückfällt, wenn das Projekt realisiert wird. Diese ist nachfolgend ersichtlich:

$$\overline{\text{Aufwand}} = \left[\begin{array}{ccc} 1 & & \\ \frac{1}{\text{Aufwand}_{\text{Abteilung}_1}} & \frac{1}{\text{Aufwand}_{\text{Abteilung}_2}} & \dots & \frac{1}{\text{Aufwand}_{\text{Abteilung}_l}} \end{array} \right]_{1 \times l} \quad (1)$$

Zum zweiten ist eine Nutzenmatrix zu identifizieren, bei welcher jedes Element den Nutzen einer Abteilung l repräsentiert, der entsteht, wenn das betrachtete Projekt

realisiert wird. Diese ist nachfolgend ersichtlich:

$$\overline{\text{Nutzen}} = \left[\text{Nutzen}_{\text{Abteilung}} \quad \text{Nutzen}_{\text{Abteilung}} \quad \dots \quad \text{Nutzen}_{\text{Abteilung}} \right]_{k \times 1}^T \quad (2)$$

Der Definitionsbereich der beiden Vektoren beläuft sich auf $\overline{\text{Nutzen}}, \overline{\text{Aufwand}} \in \mathbb{R}^l$, wobei $\text{Aufwand} > 0$. Der Definitionsbereich des Aufwands ist als prozentualer Arbeitsaufwand einer Abteilung j zu interpretieren, der einmalig entsteht, wenn ein Projekt durchgeführt wird. Wenn der Aufwand einer Abteilung j zwischen eins und null fällt, spart sich diese Abteilung Arbeit bei der Durchführung des zu bewertenden Projekts, da z.B. ein alternatives Projekt nicht zu Ende geführt werden muss. Sie bleibt unbeteiligt bei einem Arbeitseinsatz von 1. Wenn der Aufwand größer als 1 ist, muss sie an einem Projekt partizipieren. Es gilt: Je größer der Aufwandswert ist, desto größer ist der geplante Arbeitseinsatz dieser Abteilung und je weiter der Aufwandswert unterhalb der eins ist, desto größer ist die Arbeitseinsparung der Abteilung.

Der Definitionsbereich des Nutzens einer Abteilung i ist intuitiv zu interpretieren. Ist dieser größer als 0, so profitiert diese Abteilung in Folge eines Projekts, da sie z.B. einen dauerhaften, monetären Nutzen an der effektiveren Durchführung eines Prozesses hat. Ist dieser kleiner als 0, so werden für diese Abteilung negative Effekte in Folge des Projekts erwartet. Beispielsweise kann ein abteilungsspezifischer Schaden genannt werden, da teure Serveranlagen gekauft werden müssen. Unter Umständen können Projekte trotz negativer Effekte mancher Abteilungen dennoch sinnvoll sein. Beispielsweise können teure Hardwareanlagen als Schaden für eine Abteilung interpretiert werden, aber für weitere Abteilungen einen hohen Nutzen entfalten. Entscheidend ist hier die Summe an Nutzenwerten sämtlicher Abteilungen. Ist diese positiv, so macht das Projekt für das Unternehmen im Gesamten Sinn. Wird für eine Abteilung ein Nutzen von 0 gewählt, so bedeutet dies, dass keine Nutzeneffekte in Folge des Projekts zu erwarten sind.

In Anlehnung an der Definition des dyadischen Produkts wird wie folgt definiert [ZJ07]: Wird die einspaltige Matrix als Spaltenvektor $\overline{\text{Nutzen}}$ interpretiert und die einzeilige Matrix als Zeilenvektor $\overline{\text{Aufwand}}^T$ so ergibt sich aus dem dyadischen Produkt eine Matrix namens *SynergieMatrix* $\in \mathbb{R}^{k \times l}$, dessen Elemente das Aufwands-Nutzen-Verhältnis für jede Abteilung abbildet.

$$\text{SynergieMatrix} = \overline{\text{Nutzen}} \otimes \overline{\text{Aufwand}} = \overline{\text{Nutzen}} \cdot \overline{\text{Aufwand}}^T = [\dots]_{k \times l} \quad (3)$$

Weil bei jedem Unternehmen nicht immer sämtliche Abteilungen Aufwand leisten oder Nutzen nießen, werden im Folgenden m Abteilungen fokussiert, die einen Nutzen nießen, und n Abteilungen, die einen Aufwand leisten. Die entstehende Menge ist als Teilmenge der gesamten Menge an Abteilungspaaren zu verstehen:

SynergieMatrix =

$$\left[\begin{array}{cccc}
 \frac{Nutzen_{Abteilung_1}}{Aufwand_{Abteilung_1}} & \frac{Nutzen_{Abteilung_1}}{Aufwand_{Abteilung_2}} & \dots & \frac{Nutzen_{Abteilung_1}}{Aufwand_{Abteilung_n}} \\
 \frac{Nutzen_{Abteilung_2}}{Aufwand_{Abteilung_1}} & \frac{Nutzen_{Abteilung_2}}{Aufwand_{Abteilung_2}} & \dots & \frac{Nutzen_{Abteilung_2}}{Aufwand_{Abteilung_n}} \\
 \vdots & \vdots & \ddots & \vdots \\
 \frac{Nutzen_{Abteilung_m}}{Aufwand_{Abteilung_1}} & \frac{Nutzen_{Abteilung_m}}{Aufwand_{Abteilung_2}} & \frac{Nutzen_{Abteilung_m}}{Aufwand_{Abteilung_n}} & \frac{Nutzen_{Abteilung_m}}{Aufwand_{Abteilung_n}}
 \end{array} \right]_{m \times n} \quad (4)$$

Beispielsweise kann am zweiten Element der letzten Zeile in der *SynergieMatrix* erkannt werden, welchen Nutzen die Abteilung *m* von einem Projekt erhält, wenn die *Abteilung₂* dafür arbeiten muss. Im Vergleich mit anderen Elementen können somit Abhängigkeitsverhältnisse (negative Synergien), aber auch Nutzenpotenziale für viele Abteilungen erkannt werden, die bei einer Projektdurchführung entstehen. Der *SynergieEffekt* ergibt sich dann aus der Summe der Elemente, wie in Gleichung (5) gesehen werden kann. Um den ungewollten Effekt zu vermeiden, dass der Schaden einer *Abteilung_i* durch den Mehraufwand einer *Abteilung_j* verkleinert oder durch den Minderaufwand dieser Abteilung vergrößert wird, kann die folgende Fallunterscheidung Abhilfe schaffen.

Hohe Synergieeffekte erzielen folglich Projekte, bei denen verhältnismäßig wenige Abteilungen arbeiten müssen und viele Abteilungen profitieren. Sofern der *SynergieEffekt* für mehrere Projekte ermittelt wird, können diese leicht in ein Ranking gebracht werden, sodass die attraktivsten Projekte hinsichtlich der Synergieeffekte anhand der höchsten Bewertung identifiziert werden können.

$$SynergieEffekt = \begin{cases} \frac{1}{n \cdot m} \sum_{i,j=1}^{n,m} \left(\frac{Nutzen_{Abteilung_i}}{Aufwand_{Abteilung_j}} \right), & Nutzen_{Abteilung_i} > 0 \\ \frac{1}{n \cdot m} \sum_{i,j=1}^{n,m} \left(Nutzen_{Abteilung_i} \cdot Aufwand_{Abteilung_j} \right), & Nutzen_{Abteilung_i} \leq 0 \end{cases} \quad (5)$$

Der modellierte Zusammenhang des *SynergieEffekts* soll anhand Tabelle 2 mit Hilfe von mehreren Beispielen verdeutlicht werden, die aus Sicht der Autoren den wichtigsten Anwendungsfällen entsprechen. Da in den Beispielen oft die gleiche Anzahl an Aufwands- und Nutzenwerten verteilt werden, ist eine Vergleichbarkeit der Beispiele untereinander gewährleistet. Weiter wird der Vorteil einer abteilungsspezifischeren Betrachtung im Vergleich mit der Aufwands- und Nutzenbetrachtung auf Unternehmensebene transparent. Dies schließt Fachbereiche innerhalb und über die

Unternehmensgrenzen mit ein.

Der Tabelle ist zu entnehmen, dass die gewählten Fallbeispiele mit Hilfe des *SynergieEffekts* in eine sinnvolle Reihenfolge, dem Ranking, überführt werden können. Ebenfalls ist ersichtlich, dass das Bewertungsmodell den oben definierten Anforderungen gerecht wird und einer theoretischen Überprüfung standhält.

Synergieeffekt	Fallbeispiele Welche Abteilungen profitieren? Welche Abteilungen arbeiten? (Bei Vergabe von ganzzahligen Bewertungen)	Summe der Nutzenwerte Auf Basis von 15 Abteilungen	Summe der Aufwände Auf Basis von 10 Abteilungen
0,000	Unsinnige Projekte (kein Nutzen, aber Arbeit)	0	100
-6,600	Unsinnige Projekte (5 Abteilungen profitieren insg. 10 Punkte, 5 Abteilungen verlieren insg. 10 Punkte, 10 Abt. arbeiten)	0	100
-6,613 ⁴	Schlechte Projekte (4 profitieren, 5 nicht, 10 arbeiten)	-2	100
-5,266 ⁴	Gute, schadenbehaftete Projekte (5 profitieren, 4 nicht, 10 arbeiten)	+2	100
0,0133 ⁴	Gute Projekte (1 profitieren, 10 arbeiten)	+2	100
1,000	Neutrale Projekte (arbeitende und profitierende Abteilungen befinden sich im Punkte-Fit)	150	100
10,000	10/1-Gute Projekte (alle 15 Abt. profitieren zu je 10 Punkten, keine der 10 Abteilungen muss arbeiten)	150	10
0,100	1/10-Gute Projekte (alle 15 Abt. profitieren zu 1 Punkt, 10 Abt. müssen 10 Punkte arbeiten)	15	100

Tab. 2: Arbeitsweise des *SynergieEffekts* im Überblick

4 Auswertung und Ausblick

In der Theorie wurde gezeigt, dass die an das Bewertungsmodell gestellten Anforderungen mittels ausgewählter Fallbeispiele überprüft werden können. Dieses gilt es in der Praxis zu validieren.

Die gezeigten Ergebnisse müssen wie folgt eingeschränkt werden. Als Prämisse gehen gezeigte Ansätze davon aus, dass ein EAM im Sinne eines projektorientierten EAMS entwickelt wird. Eine Integration in weitere Bereiche des EAM im Sinne von Ahlemann [ASML17] erfolgt nicht, sodass die geschaffene Methode zur Priorisierung von Anforderungen unter Berücksichtigung von Synergieeffekten lediglich als Ergänzung zu bestehenden Ansätzen zu verstehen ist. Ebenso können auch kleinere Vorhaben oder regelmäßige Abläufe Synergieeffekte erzielen, die keiner Organisation als Projekt bedürfen. Weiter werden Netzwerkeffekte diverser Abteilungen vorausgesetzt, sodass

der erfolgreiche Einsatz der geschaffenen Methode von der Auswahl relevanter Aufwandsträger und Nutznießer in direkter Weise abhängt.

Weiter gilt es das Bewertungsmodell zu erweitern, sodass auch Abhängigkeiten und/oder Synergien zwischen den Anforderungen bzw. Projekten direkte Berücksichtigung finden und die identifizierte Forschungslücke ergänzt werden kann. Eine Erstellung einer Fallbasis ist ebenso attraktiv, sodass sämtliche Bestandteile der EAM-Architekturpyramide überprüft werden können. Weiter können Fallbeispiele gereicht werden, welche über die Architekturpyramide eines einzelnen Unternehmens hinausreichen und beispielsweise folgende Bereiche berücksichtigt: Partnerunternehmen, Forschungsk Kooperationen, Zulieferunternehmen, etc.

Literaturverzeichnis

- [AFS09] Aaqib I.; Farhan, M.K.; Shabaz A.K.: A critical analysis of techniques for requirement prioritization and open research issues. *International Journal of Review in Computing*, S. 8-18, 2009.
- [ASML17] Ahlemann F.; Stettiner E.; Messerschmidt M.; Legner C.: *Strategic Enterprise Architecture Management - Challenges, Best Practices and Future Developments*. Springer, Heidelberg, 2017.
- [AW03] Aurum A.; Wohlin C.: The Fundamental Nature of Requirement Engineering Activities as a Decision-Making Process. *Information and Software Technology*. 45(14), S. 945-954, 2003.
- [BA05] Berander, P.; Andrews, A.: Requirements prioritization. In A. Aurum & C. Wohlin (Eds.), *Engineering and managing software requirements*, Springer Berlin Heidelberg, S. 69-94, 2005.
- [B14] Brockmann C.: *An approach to design the business model of an ERP vendor*. Gito, Berlin, 2014.
- [CS01] Carnahan J.; Simha R.: Natures algorithms. *IEEE Potentials*, S. 21-24, 2001.
- [CD17] Czarnecki C.; Dietze C.: *Reference Architecture for the Telecommunications Industry: Transformation of Strategy, Organization, Processes, Data and Applications*. Springer, Heidelberg, 2017.
- [D06] Dern G.: *Management von IT-Architekturen*. 2. Auflage, Vieweg, 2006.
- [G05] Greer, D.: Requirements Prioritisation for Incremental and Iterative Development. *Requirements Engineering for Sociotechnical Systems*. Information Science Publishing, S. 100-118, 2005.
- [GBS99] Greer D.; Bustard D.; Sunazuka T.: Prioritisation of system changes using cost-benefit and risk assessments, *Proceedings of the Fourth IEEE International Symposium on Requirements Engineering*, S. 180-187, 1999.
- [GR04] Greer D.; Ruhe G.: Software release planning: An evolutionary and iterative approach. *Journal Information Systems*, 46(4): S. 243-253, 2004.

- [G01] Gronau N.: Industrielle Standardsoftware: Auswahl und Einführung. Oldenburg, München, 2001.
- [G12] Gronau N. (2012) Handbuch der ERP-Auswahl. GITO-Verlag, Berlin, 2012.
- [H75] Holland J.H.: Adaptation in natural and artificial systems. Ann Arbor: University of Michigan Press, 1975.
- [H14] Hruschka P.: Business Analysis und Requirements Engineering: Produkte und Prozesse nachhaltig verbessern. Carl Hanser Verlag, 2014.
- [I98] IEEE Std 830-1998: IEEE Recommended Practice for Software Requirements Specifications, IEEE Computer Society, Los Alamitos, 1998.
- [I14] IEEE: Guide to the Software Engineering Body of Knowledge (Swebok®): Version 3.0. IEEE Computer Society Press, S. 8-12, 2014.
- [KR97] Karlsson J.; Ryan K.: A cost-value approach for prioritizing requirements. IEEE Software, 14(5), S. 67-74, 1997.
- [KWR98] Karlsson J.; Wohlin C.; Regnell B.: An Evaluation of Methods for Prioritizing Software Requirements, Information and Software Technology, 39(14-15), S. 939-947, 1998.
- [K04] Keen J.: Avoiding the TCO Trap, CIO, S. 44-46, 2004.
- [K05] Krcmar H.: Informationsmanagement. 4. Auflage, Springer, Berlin, 2005.
- [L05] Lankhorst M.: Enterprise Architecture at Work: Modeling Communication and Analysis, Springer, 2005.
- [L02] Lausen S.: Software Requirements – Styles and Techniques. Pearson Education, 2002.
- [LW00] Leffingwell D.; Widrig D.: Managing Software Requirements – A Unified Approach. Addison-Wesley, 2000.
- [L10] Leffingwell D.: Prioritizing Features. Technical report, Scaling Software Agility, <http://scalingsoftwareagilityblog.com/prioritizing-features/>, Abgerufen am: 24.09.2015.
- [MR96] Maiden N.A.M.; Rugg G.: ACRE: Selecting methods for requirements acquisition. Software Engineering Journal, 11(3), S. 183-192, 1996.
- [M02] Moisiadis F.: The fundamentals of prioritizing requirements, Proceedings of the systems engineering, test and evaluation conference, S. 1-12, 2002.
- [M03] Moisiadis F.: A Framework for Prioritising Software Requirements. PhD thesis, Macquarie University, 2003.
- [ÖBF10] Österle H.; Becker J.; Frank U.: Memorandum zur gestaltungsorientierten Wirtschaftsinformatik. Zeitschrift für betriebswirtschaftliche Forschung, 11, S. 664–669, 2010.
- [R01] Regnell B. et.al.: An Industrial Case Study on Distributed Prioritisation in Market-Driven Requirements Engineering for Packaged Software. Requirements Engineering, 6(1), S. 51-62, 2001.
- [RM97] Rugg G.; McGeorge P.: The sorting techniques: A tutorial paper on card sorts, picture

sorts and item sorts, *Expert Systems*, 14(2), S. 80-93, 1997.

- [REP02] Ruhe G.; Eberlein A.; Pfahl D.: Quantitative WinWin – A New Method for Decision Support in Requirements Negotiation. Proceedings of the 14. International Conference SEKE, ACM Press, New York, S. 159-166, 2002.
- [REP03] Ruhe G., Eberlein A., Pfahl D.: Trade-off Analysis for Requirements Selection. *International Journal of SEKE*, 13(4), 345-366, 2003.
- [S80] Saaty T.L.: *The analytic hierarchy process*. McGraw Hill, New York, 1980.
- [S01] Saaty T.L.: *Models, Methods, Concepts & Applications of the Analytic Hierarchy Process*. Kluwer Academic Publishers, Norwell, 2001.
- [S10] Schatten A. et.al.: *Best Practice Software-Engineering*. Spektrum Akademischer Verlag, Wien, 2010.
- [SM99] Schulmeyer C.G.; Mc Manus J.I.: *Handbook of Software Quality Assurance*. 3. Ausgabe, Prentice Hall, Upper Saddle River, 1999.
- [SS13] Schwaber K., Sutherland J. (2013) *The Scrum-Guide*, <http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-US.pdf>, Abgerufen am 29.06.2017.
- [SMS13] Shadab S.; Mohd R.B.; Shahin F.: Effectiveness of Requirement Prioritization Using Analytical Hierarchy Process (AHP) And Planning Game (PG): A Comparative Study. *International Journal of Computer Science and Information Technologies*, 4(1), S. 46-49, 2013.
- [SHM92] Shen Y.; Hoerl A.E.; McConnell W.: An Incomplete Design in the Analytical Hierarchy Process. *Mathematical Computer Modelling* 16(5), S. 121-129, 1992.
- [SS05] Sillitti A.; Succi G.: *Requirements Prioritization*. Springer, Heidelberg, 2005.
- [TO12] The Office of Quality Improvement University of Wisconsin – Madison: *Project Prioritization - A Structured Approach To Working On What Matters Most*. http://oqi.wisc.edu/resource/library/uploads/resources/Project_Prioritization_Guide_v_1.pdf. Abgerufen am 29.06.2017.
- [W99] Wiegers K.E.: First things first Prioritizing requirements. *Software Development Online*, 7: S. 48-53, 1999.
- [W00a] Wiegers K.E.: Karl Wiegers describes 10 requirements traps to avoid. *Software Testing & Quality Engineering*, 2(1), 2000.
- [W00b] Wiegers K.E.: When telepathy won't do: Requirements engineering key practices. *Cutter IT Journal*, 13(5), S. 9-15, 2000.
- [WB13] Wiegers K.; Beatty J.: *Software Requirements, Developers Best Practices*, Pearson Education, 2013.
- [Y04] Young R.: *The Requirements Engineering Handbook*. Artech House technology management and professional development library, Artech House, 2004.
- [ZJ07] Zahmann H.G.; Jünger A.: *Mathematik für Chemiker*. 6. Auflage, Wiley-VCH, Weinheim, 2007.