

## Software-Evaluation: mit Vollgas ins Verderben

*Software ist zu einem der wichtigsten Produktionsfaktoren in Unternehmen und Organisationen geworden. Entsprechend hohe Bedeutung hat die Evaluation und Auswahl der am Markt angebotenen Lösungen. Betrachtet man die Zahl an (Miss-)Erfolgen von Software-Einführungen, könnte man allerdings folgern, es handle sich hierbei um einen Nebenjob von Amateuren.*

*Was läuft schief bei der Spezifikation und Auswahl von Software-Lösungen? Was ist zu beachten, damit das Werkzeug seine Zielsetzung – einen substanziellen Beitrag an den Erfolg des Unternehmens zu leisten – zu erfüllen vermag? Der kritischen Analyse am Beispiel Projektmanagement-Software folgen konkrete Handlungsempfehlungen für Anwender und Anbieter. Heinz Scheuring*

### Am Ende war nur noch die Frustration

31. März 2012. Der Informatikausschuss von TopSystems AG beschliesst, ein einheitliches, integriertes System für die Planung und Steuerung der Projekte und des gesamten Projektportfolios zu beschaffen.

30. August 2012. Das detaillierte Pflichtenheft steht. 17 Systeme werden nun einer Vorevaluation unterzogen. 4 davon erreichen kurz darauf die Shortlist. Nach eingehenden Analysen sowie Präsentationen durch die Hersteller macht SuperPlan schliesslich das Rennen. Nach derart systematischer Evaluation, einem umfangreichen Customizing der Software und einer genauen Planung der Schulungs- und Einführungsmassnahmen scheint einer erfolgreichen Einführung von SuperPlan nichts mehr im Wege zu stehen.

31. März 2013. Die Benutzer sind geschult und für den operativen Einsatz des Tools auch motiviert. Den Projektleitern bei TopSystems ist die Hauptrolle bei der Arbeit mit SuperPlan zugeordnet. Die Praxis – der wirkliche Test, was den realen Kundennutzen betrifft – kann beginnen.

31. März 2014. Das System ist seit einem Jahr in Betrieb. Die Bilanz sieht mehr als düftig aus. Projektleiter, Linienvorgesetzte und das Management auf den verschiedenen Ebenen sind – in unterschiedlicher Masse – ernüchtert über die Resultate des neuen Projektmanagement-Systems. Der erhoffte und vom Lieferanten versprochene Nutzen – einfache, schlagkräftige Planung und Steuerung der einzelnen Projekte, konsequente Steuerung der personellen Ressourcen, umfassende Transparenz über das gesamte Projektportfolio, raschere und bessere Entscheide auf allen Stufen – ist weitgehend ausgeblieben. Die Resultate stehen in keinem Verhältnis zum hohen Aufwand,

den die Bedienung von SuperPlan den Usern abverlangt. Die Daten für das Management sind aufgrund der Informationslücken nicht nur wertlos, sondern gefährlich: Die Entscheide basieren auf falschen Datengrundlagen. Die „Schattenwirtschaft“ – wieder auferstandene Projektplanungen mit Microsoft Project und Ressourcendisposition mit Excel – greift um sich.

Erste Stimmen werden laut, das System der früheren Praxis zu opfern, die vor zwei Jahren eben der Anlass für die Einführung von SuperPlan war...

So oder ähnlich präsentiert sich die Situation am Ende von Systemeinführungen – ob bei Projektportfolio-Management-Systemen, im ERP-Bereich oder bei CRM-Systemen. Die Konsequenzen solcher Fehlentscheide und Fehlinvestitionen sind gravierend.

Was ist bei TopSystems schief gelaufen? Woran ist dieses Software-Projekt gescheitert, und wer hat dabei was falsch gemacht?

### Gut gemeint, oft jedoch voll daneben

Anwender und unterstützende IT-Fachkräfte meinen es gut, wenn sie Pflichtenhefte aufstellen. Sehr häufig leider nur zu gut.

Testberichte von Fachautoren werden studiert, in denen Task Management-Lösungen verglichen oder die Features des neuen CRM-Systems des Branchenprimus angepriesen werden. Nur logisch, dass diese auf der Liste der Anforderungen an das neue System nicht fehlen dürfen. Die Hersteller werden sich ja etwas überlegt und das eingebaut haben, was die Kunden wünschen. Oder vielleicht doch nicht? Und wenn doch – könnte es sein, dass die besonders aktiven Feedbacker der Kunden

eben gerade nicht das Alltagsbedürfnis des Normalanwenders repräsentieren?

Und wäre es denkbar, dass die Tester die Software gar nicht über eine längere Zeit unter realitätsnahen Bedingungen selber eingesetzt haben? Und was macht diese Autoren eigentlich zu Fachexperten in ihrem Gebiet? Und falls sie denn welche sind – ist nicht häufig genau dies das Problem, dass der sogenannte Experte an den Bedürfnissen des Normalusers vorbei testet und schreibt?

Während der Autokäufer aus Erfahrung weiss, welche Bedeutung für ihn der satte Sound der Lautsprecheranlage oder die Qualität des Navigationssystems haben, ist der Projektleiter für die Evaluation der Projektportfolio-Management-Lösung kaum in der Lage, die Relevanz des automatischen Belastungsabgleiches bei der Ressourcenplanung zu beurteilen. Er verlässt sich auf das Urteil im Testbericht, das sich über eine Schwäche von Tool X in diesem Feature mokiert. Dass diese

gar kontraproduktiv erweisen. Die fatale Folge: Tools, die auf das Relevante getrimmt und dadurch besonders erfolgversprechend sind, schaffen es nicht einmal auf die Longlist.

Gerade auch clevere Software-Lösungen kleinerer Anbieter fallen damit häufig schon sehr früh durch das Evaluationsraster. Dies ist für beide Seiten zu bedauern – für den Anbieter ebenso wie für den potenziellen Kunden, der damit möglicherweise ein erhebliches Nutzenpotenzial verschenkt, ohne sich dessen bewusst zu sein.

Da leider nur selten Praxis-Härtetests über den Kauf einer Lösung entscheiden, trimmen vorab potente Hersteller ihre Lösungen primär auf überzeugende Wirkung bei der Demo und hohe Punktzahl beim praxisfremden Vergleichstest. Die Praxistauglichkeit bleibt dabei auf der Strecke. Denn was bei der Demo leicht bedienbar, durchdacht und sexy wirkt, kann sich in der Realität als das pure Gegenteil herausstellen. So wirkt etwa die Verschiebung von Projektterminen direkt im Balkendiagramm handlich und attraktiv, in Wirklichkeit birgt diese Form von Termineingriffen die Gefahr von Fehlmanipulationen.

An konkreten Beispielen wird im Folgenden aufgezeigt, wie Fehlentscheide bei der Software-Evaluation zustande kommen können und welche Konsequenzen dies nach sich ziehen kann.

An konkreten Beispielen wird im Folgenden aufgezeigt, wie Fehlentscheide bei der Software-Evaluation zustande kommen können und welche Konsequenzen dies nach sich ziehen kann.

### **Drama in acht Akten: Phasenmodell für Software-Evaluation**

#### *1 Die Motivation*

Alle sind dafür. Das neue System soll einen wesentlichen Beitrag an die Arbeitseffizienz und Entscheidungsqualität leisten.

#### *2 Der Stolz*

Der Kriterienkatalog steht. Das Projektteam hat an alles gedacht. Der Projektausschuss verdankt die hervorragende konzeptionelle Arbeit.

#### *3 Die Begeisterung*

Der Entscheid ist gefällt. Das System wurde auserkoren. Der Lieferant scheint hoch professionell. Die Einführung kann starten.

#### *4 Die Nach(t)arbeit*

Der Aufwand für Customizing, Anpassungen und Rollout sind wesentlich grösser als geplant. Zusätzliches Personal wird freigestellt, das Budget erhöht. Manche Überstunde wird geleistet. Der Wille zum Erfolg ist jedoch ungebrochen.

#### *5 Die Ernüchterung*

Der erhoffte und versprochene Nutzen bleibt aus. Das System ist kompliziert. Die Akzeptanz schwindet. Schuldige werden gesucht.

#### *6 Die Schattenwirtschaft*

Die alten Lösungen werden aus der Versenkung geholt, zunächst noch heimlich. Das Management beginnt nach den Ursachen für das Desaster zu fragen.

#### *7 Der Ersatz*

Das Management bittet zur Krisensitzung. Der Misserfolg wird nicht mehr schöngeredet. Die Rückkehr zu den alten, selbstgestrickten In-sellösungen mit Handarbeit wird autorisiert.

#### *8 Zurück auf Feld 1*

Die Evaluation einer alternativen Lösung wird budgetiert. Der Leiter des Projektausschusses ernimmt sich zum Projektleiter.

Funktion in der Regel unbedeutend, ja dessen Fehlen möglicherweise sogar vorteilhaft ist, weiss er nicht.

In der Ausbildung zum Business Analysten wurde ihm ausserdem vermittelt, Muss-Kriterien zu definieren. Und so geht er denn hin und deklariert Features als zwingend, die sich in der späteren Praxis als bedeutungslos oder

- Wie wahrscheinlich ist es, dass die Funktion genutzt wird?
- Welchen konkreten Beitrag vermag die Nutzung der Funktion an den Erfolg des Projektes bzw. an das Unternehmen zu leisten?

- Mit welchem Aufwand / wie effizient lässt sich diese durch den normalen User nutzen?
- Welche Risiken könnte der Einsatz der Funktion bergen (z.B. den Verzicht auf persönliche Kommunikation), welche Nebenwirkungen könnte er auslösen?

Unterziehen wir die Kriterienkataloge, wie sie in Vergleichstests und einschlägigen Studien für professionelle Projektmanagement-Systeme zum Einsatz kommen, unter diesem Paradigma einer kritischen Analyse. Der Fokus soll in diesem Fall auf dem Projektportfolio-Management (PPM) einschliesslich der Planung des Ressourceneinsatzes in der Multiprojekt-Umgebung liegen. Es handelt sich im Folgenden um Funktionen, wie sie typischerweise in solchen Untersuchungen bewertet werden.

*Ideenmanagement.* Die Ideenfindung mittels Social Media (Chats & Tweets) zu unterstützen, vielleicht noch ausgestattet mit Abstimmungs- und Bewertungsfunktionen oder gar Mindmapping, muss zu den Spielereien gezählt werden. Und Spielereien wirken in Business-Software belastend, weil der Kunde diese möglicherweise erst nach erfolglosem Nutzungsversuch als solche erkennt.

*Projektbewertung und -priorisierung.* Die Bubble Charts des PPM-Tools mögen ansprechend aussehen. Wenn es jedoch darum geht, die Auswertungen an die spezifischen Bedürfnisse und Vorstellungen des Managements anzupassen, dürfte so viel Zeit in das Customizing fließen, dass der Griff zum Excel besser schon vorher gemacht worden wäre.

*Skills-basierte Planung.* Unter diesem in Projektmanagement-Fachkreisen verbreiteten Begriff verbergen sich zahlreiche Missverständnisse. Die gemischte Einlastung auf Skills einerseits und auf Personen oder Organisationseinheiten andererseits birgt nicht nur das Risiko hoher Komplexität, sondern führt auch zu Interpretationsfehlern oder gar zu mathematischen Widersprüchen. Dies auch deshalb, weil Mitarbeiter in der Regel über mehrere Skills gleichzeitig verfügen und diese meist in unterschiedlicher Qualität. Das hohe Gewicht, das diese Funktion in Kriterienkatalogen häufig erhält, ist nicht gerechtfertigt.

*Ressourcenreservierungen.* Die Reservation von Ressourcen durch den Projektleiter mit automatischer Information des Linienvorgesetzten steht der Kommunikation zwischen den beiden Playern im Weg. Die Qualität der Planung wird in aller Regel besser, wenn die Planungsdaten nach der persönlichen Abstimmung mit dem Projektleiter durch die Linie erfasst werden, da die Ressourcenplanungshohheit bei den Linienmanagern liegt.

*Automatischer Belastungsabgleich.* Er ist ein Paradebeispiel für Fehlleitungen durch solche "Evaluationshilfen". Dazu ein Auszug aus einem Kriterienkatalog<sup>4</sup>: "Eine Ressourceneinplanung führt zu einem nachvollziehbaren, organisationsweiten Ressourcenabgleich (inkl. Beachtung logischer Abhängigkeiten und Projektprioritäten)". Eine solche Anforderung lässt sich, wie der Autor in umfangreichen Publikationen<sup>1</sup> dargelegt hat, aufgrund der Komplexität der Aufgabenstellung mittels einer Software praktisch nicht erfüllen. Sie suggeriert eine Entscheidungshilfe, die in Wirklichkeit eher als Hilfe zur Fehlentscheidung zu bezeichnen ist. Der automatische, durch die Software berechnete Belastungsabgleich über mehrere Projekte ist praktisch immer fragwürdig und wird durch die Software am besten gar nicht erst angeboten.

Weiter zu erwähnen sind quantifizierte *Risikoanalysen*, *Stakeholder-Analysen*, das *Management von Change Requests* oder auch *CRM-Funktionen*, die Kampagnen und E-Mailings unterstützen. Solche Aufgaben werden durch die Hausmittel der Officepalette oder dedizierte Lösungen meist wesentlich überzeugender und/oder mit weniger Aufwand unterstützt als mit der lustlos implementierten Nebenfunktion, wie sie der PPM-Tool-Anbieter eingebaut hat, nur um mit den Konkurrenten gleichzuziehen.

*Integriertes System.* Eine besonders zentrale Thematik ist die starre Kopplung von Einzelprojektplanung und Projektportfolio-Management im selben System, wie sie von einzelnen Theoretikern auch heute noch propagiert wird. Dabei wird die Verantwortung für die Ressourcenplanung dem Projektleiter zugeordnet, und diese erfolgt auf Vorgangsebene. Dass die Anwender dieses integrierten Ansatzes zuerst die Ernüchterung, dann den Absturz erleben müssen, bevor sie realistische Wege beschreiten, zählt zu den Klassikern unter den Evaluationsunfällen bei Projektmanagement-Systemen (vgl. dazu das Konzept der drei Welten im Whitepaper<sup>2</sup> des Autors). Projektmanagement-Systeme mit dem Anspruch, sowohl die Detailplanung des Einzelprojektes als auch das Projektportfolio-Management umfassend abzubilden, scheitern häufig daran, dass sich damit am Ende weder die Erwartungen und Anforderungen der Projektleiter noch jene des Managements erfüllen lassen. Und die Komplexität des Systems führt zu organisatorischen Problemen und zu einer schwerfälligen Handhabung. Die sprichwörtliche eierlegende Wollmilchsau lässt grüssen.

## Brauchbar ist nicht gleich brauchen!

Auch dann, wenn ein Projektmanagement-System eine bestimmte Funktion in einer ordentlichen Qualität anbietet, muss deren Nutzung noch lange nicht angezeigt sein. Wurde diese bisher mit den bekannten Office-Tools abgedeckt und bestehen geringe Abhängigkeiten mit den eigentlichen Projektdaten, ist das herkömmliche Mittel häufig die bessere Wahl. So kann der Hyperlink aus dem Projekt im PM-Tool auf die Risikoanalyse im Excel die Verbindung herstellen, ohne die im PM-Tool integrierten, meist starren und begrenzten Möglichkeiten für die Risikoanalyse nutzen zu müssen, nur weil sie drin sind. Damit lässt sich die Komplexität des neuen Systems tiefer halten. Risikoanalysen, CRM-Funktionen oder differenzierte Bewertungsfunktionen sind unter diesem Aspekt in Projektmanagement-Systemen nicht einfach nur wenig relevant, sondern am Ende belastend und sollten bei der Bewertung deshalb letztlich in Abzug gebracht werden. Beinhaltet der Kriterienkatalog nun eine grosse Zahl solcher Funktionen, oder sind diese sogar in der Überzahl, dann zeigt die Gesamtbewertung von Systemen ein völlig falsches Bild vom Nutzen, den diese Systeme in der Praxis generieren.

## Nutzungseffizienz keine Nebensache

Neben den häufig fragwürdigen Kriterienkatalogen fehlen in Studien und Testberichten zu Business Software in der Regel brauchbare Anhaltspunkte zur Nutzungseffizienz. Im praktischen Einsatz ist entscheidend, wie direkt und schnell Eingriffe im System möglich sind. Stark formularbasierte Software schneidet in dieser Hinsicht naturgemäss schlechter ab als Systeme, die Eingaben auch in tabellarischen Darstellungen erlauben. Betont Report-orientierte Systeme zwingen den Anwender, immer wieder den mühsamen Weg vom Report in die Erfassungsmaske zu gehen, um auf erkannten Anpassungsbedarf zu reagieren. Dies ist äusserst ineffizient und kostet am Ende nicht nur Nerven, sondern auf die Dauer auch viel Geld. Bei der Ressourcenplanung etwa ist die Möglichkeit zentral, den Datenwürfel aus den drei Dimensionen Projekte, Ressourcen und Zeit nicht nur betrachten, sondern aus jedem Blickwinkel direkt in die Daten eingreifen zu können. Und neben der automatischen Einlastung über die gewünschte Dauer ist die Möglichkeit manueller Einlastungen in beliebigen Zeitrastern zentral. Doch ausgerechnet hier, bei der letztlich alles entscheidenden Frage nach dem Aufwand-/Nutzenverhältnis, fehlen meist Angaben oder hilfreiche Kriterien. Studien testen an solchen zentralen Fähigkeiten meist grosszügig oder unwissend vorbei.

Es reicht auch nicht, diese Aspekte am Rande zu erwähnen. Doch genau das geschieht: In der Praxis berichten die Experten über solche Zusammenhänge bestenfalls in einem Nebensatz. Dazu ein Beispiel aus einem Kriterienkatalog<sup>4</sup>: „Die Lösung ist einfach bedienbar: Mit wenigen Klicks kommt man zum Ziel.“ Dabei handelte es sich um ein einziges aus insgesamt 59 Kriterien. In der Gesamtbewertung geht der Effekt hoher Benutzerfreundlichkeit auf diese Weise praktisch unter. Und dies ist am Ende fatal für das Unternehmen, das aufgrund solcher Analysen entscheidet.

## Die Interpretation – alleinige Aufgabe des Anwenders?

Autoren von Vergleichsstudien argumentieren, es sei Sache des künftigen Anwenders, aus ihren Berichten die für sie relevanten Kriterien herauszufiltern und diese ihren Bedürfnissen entsprechend zu gewichten. Das ist nur teilweise korrekt. Denn dabei wird vorausgesetzt, dass die Organisation, die sich auf die Suche nach einem neuen System macht, weiss, welche Bedeutung jede einzelne der unzähligen beschriebenen Funktionen für sie hat. Das trifft jedoch in den wenigsten Fällen zu. Häufig wird Software für Funktionen evaluiert, die professionelles Spezialwissen erfordern. In Abhängigkeit des Themas fehlt dies nicht nur im kleineren Betrieb, sondern häufig auch im Konzern. Auch ein Project Management Office, dem die Optimierung des Projektmanagements und des Projektportfolio-Managements obliegt, kommt hier an seine Grenzen. Es wird ja – so ist zu hoffen – nicht alle paar Monate mit der Aufgabe konfrontiert sein, ein neues Projektmanagement-Werkzeug zu evaluieren. Immerhin mag sich das PMO den teuren, neutralen Berater noch leisten. Beim KMU kann dies aus finanziellen Gründen zum Problem werden. Allerdings wird sogar der Berater häufig in dieselbe Falle tapen wie der Anwender, da er sich bei seinen Empfehlungen auf die einschlägigen Studien abstützt. Umso mehr müsste es die Aufgabe und das Selbstverständnis von Expertenberichten sein, diese Zusammenhänge aufzuzeigen und bei der Auswahl der Funktionen und deren Gewichtung sowie bei der Information über die Handhabbarkeit Hilfe zu bieten. Dabei gilt es, den potenziellen Anwender auch von kaum relevanten oder gar kontraproduktiven Funktionen fernzuhalten. Es macht keinen Sinn, wenn neun von zehn Anwendern die mühsame Erfahrung machen müssen, dass die Bewertung des Projektportfolios mit der integrierten Funktion des PPM-Tools mehr Aufwand erfordert bei gleichzeitig schlechterem Resultat

und geringerer Auswertungsflexibilität als die bisher genutzte Excel-Tabelle.

Eine ausgefallene Funktion mag bei jedem zehnten oder hundertsten Anwender einen Nutzen stiften. Doch wird diese in der Studie mit zentralen Funktionen auf einer Stufe behandelt, ist der potenzielle Anwender kaum in der Lage, Wichtiges von Irrelevantem zu trennen.

Es gibt Anbieter auf dem Markt, die gewisse Funktion im Flyer oder auf der Website als wichtige Errungenschaft anpreisen, um von der Nutzung derselben bei der konkreten System-einführung dann sogar selbst abzuraten. Viel mehr noch muss diese Warnfunktion die vielleicht wichtigste Aufgabe des neutralen Studienverfassers sein.

Entscheidend ist nicht, was der Autor der Studie mit dieser zu sagen glaubt, sondern was der Adressat, der potenzielle Anwender, aus dieser herausliest. Der Hinweis, dass dieser seine eigenen Prioritäten setzen muss, reicht hier bei weitem nicht. Mangels Erfahrungen mit professionellen Projektmanagement- und Projektportfolio-Management-Systemen sind die künftigen Anwender dazu kaum in der Lage. Sie lassen sich naturgemäss von den undifferenzierten Kriterienkatalogen leiten und setzen eine Anforderung im Zweifelsfall lieber auf ihren Anforderungskatalog, um möglichst nichts falsch zu machen. Nur so lassen sich die verbreiteten Software-Fehlinvestitionen erklären. Und wenn am Ende von Vergleichstests dann noch die Zusammenfassung in grafischer Form – etwa als Säulendiagramm pro Kriterienbereich – folgt, dann wird dies mit einer derartigen Macht als Vorfilter wirken, dass Systeme mit geringen Totalwerten von Beginn an keine Chance haben, obwohl vielleicht gerade diese die beste Wahl darstellen würden.

Wenn etwa ein System, das auf Ressourcenmanagement spezialisiert ist und in unzähligen Fällen Microsoft Project in diesem Bereich erfolgreich ersetzt hat, in einem solchen Diagramm schlechter bewertet wird als dieses auf die Projektdetailplanung spezialisierte System, dann führt dies das Evaluations-System ad absurdum.

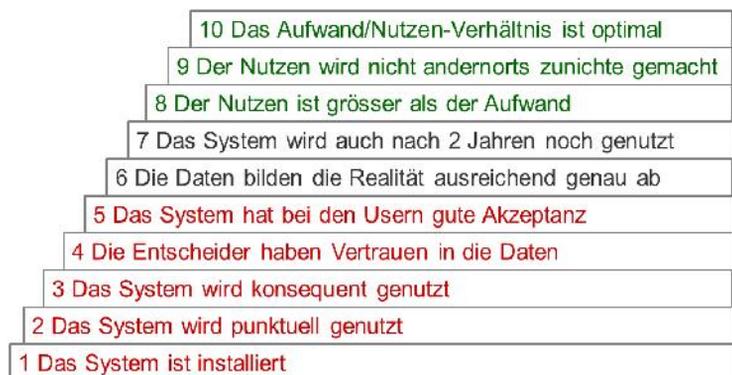
Dem Anwender keine Hilfestellung in Bezug auf die Frage nach der Relevanz von Funktionen – durchaus differenziert in Abhängigkeit verschiedener Parameter – zu bieten, muss deshalb als fahrlässig bezeichnet werden. Dieser muss bei der Prioritätensetzung bei der Hand genommen, geführt werden. So darf es nicht sein, dass der poten-

zielle Anwender auf die Suche nach dem vollintegrierten System geht, weil die Frage nach der Integration oder Trennung zwischen Einzelprojekt-Detailplanung und übergeordnetem Projektportfolio-Management in Reports über Projektmanagement-Systeme gar nicht erst zum Thema gemacht wird. Der Leser kommt so zu ganz anderen Schlussfolgerungen als jener, der vor dieser fragwürdigen Integration sachlich, aber deutlich gewarnt wurde und den praxisorientierten Weg wählt.

Beschränkt sich der Experte darauf, die Funktionen von Software-Lösungen zu beschreiben, ohne deren Relevanz herauszuarbeiten und den Leser differenziert zu unterstützen, dann schiebt dieser ausgerechnet den zentralen, wichtigsten Teil von sich – die Beratung, ob und unter welchen Umständen eine Funktion wichtig ist und wie eine überzeugende Implementation derselben aussehen sollte. Die mehr oder weniger unkommentierte Bewertung einer umfangreichen Kriterienliste ist deshalb nicht nur wenig hilfreich, sie ist gefährlich. Ebenso wenig ist das Argument stichhaltig, dass wertende Aussagen problematisch seien, da die Wertung je nach Organisation und Anwendungszweck unterschiedlich ausfallen würden. Einerseits kann auf solche Zusammenhänge konkret eingegangen werden. Andererseits gibt es zahlreiche Erkenntnisse und Regeln, die weitgehend unabhängig vom Umfeld gelten.

## Software-Qualität und Einführungserfolg

Woran misst sich die Qualität einer Software-Lösung? Und was macht es aus, ob eine Software-Einführung als Erfolg oder als Misserfolg zu betrachten ist? Wie dargelegt, ist der einzig relevante Massstab hierfür, mit welchem



### Treppe der Wahrheit

Aufwand (Zeit, Kosten, Energie, Nerven) welcher Nutzen mit dem Einsatz der Software für das Unternehmen oder die Organisation erzielt werden kann. Relevant ist hierbei nicht der Anfangsnutzen, sondern der Nutzen über

die gesamte Dauer des Betriebs. Dieser drückt sich in besseren Entscheidungen, geringerem Aufwand, besseren sachlichen Resultaten, aber auch zusätzlich generierten Erträgen, gewonnenen Kunden etc. aus. Die Treppe der Wahrheit endet bei der Benutzerakzeptanz noch lange nicht. Übersteigt der Nutzen nicht klar den Aufwand (ab Stufe 8), muss die Systemeinführung als Misserfolg gewertet werden.

Die Qualität von Software lässt sich nicht objektiv messen. Dies liegt einerseits an den verschiedenen Verwendungszwecken, andererseits an unterschiedlichen Präferenzen und Vorlieben der Anwender. Die Fragen in der Box dienen indessen als wichtiger Gradmesser für den Nutzen, den eine bestimmte Softwarefunktion stiftet.

### Software-Qualität – die Kernfragen

Voraussichtlich *benutzte* Funktionen

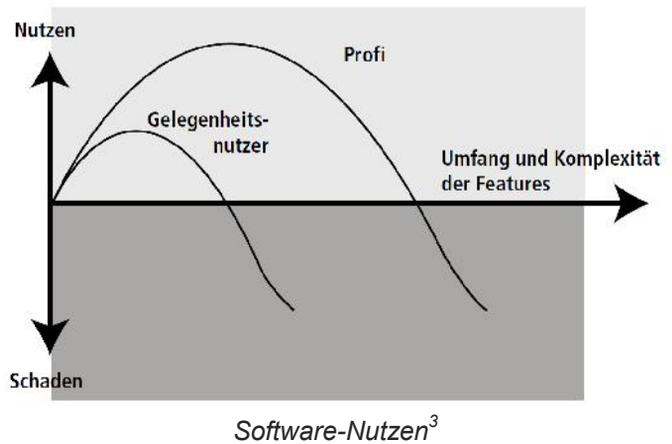
- Wie häufig wird die Funktion benötigt?
- Welchen Nutzen (bessere Entscheidung, weniger Aufwand, bessere inhaltliche Resultate, zusätzliche Erträge, neue Kunden...) generiert die Funktion?
- Wird die Funktion bzw. wie häufig wird sie wirklich genutzt (wie einfach lässt sich die Bedienung der Funktion erlernen, wie einfach ist diese auffindbar, wie schnell aus häufigen Positionen aufrufen/aktivieren)?
- Welchen Zeitaufwand erfordert die Nutzung der Funktion?
- Welchen Aufwand erfordert die Implementation (Kauf, Einrichtung, Schulung) und die Betreuung (Support) der Funktion?

Voraussichtlich *nicht benutzte* Funktionen

- Wie gut unterstützt die Software die Entscheidung, eine Funktion zu nutzen oder gezielt zu ignorieren?
- Wie weit bleibt die Funktion im normalen Betrieb im Hintergrund und belastet den Anwender nicht?
- Lässt sich die Funktion ausblenden oder auf eine tiefere, entfernte Ebene schieben (durch den Anwender, durch den Administratoren)?

Die Softwarequalität ergibt sich nun als Summe der Bewertung aller Funktionen nach diesem Raster, wobei die Bewertungsergebnisse der zweiten Fragenkategorie in Abzug zu bringen sind. Aufgrund dessen kann der Nutzwert einer Software mit zunehmenden Funktions-

umfang durchaus abnehmen, wie die folgende Abbildung zeigt.



Wer Software konsequent auf diese Weise analysiert, kommt zu fundamental anderen Resultaten als jenen, die übliche Kriterienkataloge produzieren. Gemessen an dieser „Software-Qualitätsformel“ schneidet die Mehrzahl heutiger Software-Lösungen ziemlich erbärmlich ab. Die Überladung mit irrelevanten und vielfach irritierenden Funktionen trägt dazu ebenso bei wie die fehlende Direktheit und Benutzerfreundlichkeit wichtiger Funktionen.

In Anbetracht der Bedeutung von Software in der professionellen Welt sind die Auswirkungen dieses nutzenorientierten Ansatzes gravierend. Man darf von Milliarden ausgehen, die sich damit in einer mittleren Volkswirtschaft an Nutzen erzeugen oder an Kosten vermeiden liessen. Es ist deshalb erstaunlich, ja geradezu befremdend, wie wenig diese Erkenntnisse in Software-Entwicklung, Software-Evaluationen und Tool-Einführungen einfließen. Experten, Berater und Medien haben an dieser Misere einen erheblichen Anteil.

### Flops verhindern – aber wie?

Vergleichsstudien und Testberichte können wertvolle Anhaltspunkte bei der Evaluation einer Software-Lösung liefern. Mehr nicht. Wie dargelegt, werden diese kaum eine brauchbare Beurteilung möglicher Tools erlauben. Im Gegenteil: Die Schlussfolgerungen, die sich daraus explizit oder implizit ableiten lassen, führen in der Regel ins Abseits. Um bei der Evaluation und Einführung von Software-Anwendungen Flops zu vermeiden, bieten sich ergänzend oder alternativ die folgenden Möglichkeiten an:

- *Besprechung mit Referenzanwendern.* Hier ist allerdings grosse Vorsicht geboten. Die Hersteller verstehen sich darin, die „richtigen“ Kunden auszuwählen, wenn es darum geht, ihr System im besten Licht erscheinen zu lassen.

- *Präsentation durch den Hersteller* aufgrund präziser Aufgabenstellungen und Fragen. Dieses Vorgehen eignet sich sehr gut, die kritischen Aspekte bezüglich Funktionalität und vor allem deren Implementation aufzudecken. Voraussetzung ist, dass der künftige Anwender bei der Definition seiner Anforderungen die in diesem Beitrag beschriebenen Fehler vermeidet.
- *Laboranalysen*. Hier werden repräsentative, praxisrelevante Aufgabenstellungen vorgegeben, die durch die späteren Anwender zu lösen sind. Dies erfordert einigen Aufwand, kann sich bei grösseren Anschaffungen indessen lohnen.
- *Pilotbetrieb*. Nichts kommt bezüglich der Aussagefähigkeit an den Test einer oder weniger Lösungen durch den Kunden mit dem realen Tool unter realen Bedingungen heran. Dabei sind typische Anwender einzuladen, was auch solche einschliesst, die zu dieser Rolle überredet werden müssen. Dass sich jemand aktiv um die Teilnahme bewirbt, kann gerade ein Indiz dafür sein, dass er nicht den Durchschnittsuser repräsentiert.
- *Eigener Kriterienkatalog*. Die Anforderungen und der Kriterienkatalog sollten aufgrund der (geplanten oder schon bestehenden) Nutzung der Funktionen im realen Betrieb abgeleitet werden. Ein Pilotbetrieb kann diesen Prozess ideal unterstützen.
- *Berater* sollten sehr sorgfältig ausgewählt werden. Kompetenzen und Haltung des Beraters sind an den in diesem Beitrag aufgezeigten Zusammenhängen kritisch zu prüfen.
- *Benutzerdokumentation*. Der Kunde sollte die zentralen Punkte für eine erfolgreiche Anwendung der Software in einer knappen Dokumentation zusammenstellen. Diese soll nicht nur Zweck, Einsatzmöglichkeiten und die wichtigsten Handgriffe beinhalten, sondern vor allem auch praktische Hinweise und Hilfen zu einer sinnvollen Nutzung bieten. Dazu gehört auch, den User davor bewahren, fragwürdige Funktionen zu nutzen.

Werden diese Grundsätze beachtet, ist die Wahrscheinlichkeit für erfolgreiche Software-Einführungsprojekte massiv höher als bei der Beschränkung auf Vergleichstests, Demos und Expertenmeinungen.

## Grundsätze guter Software-Entwicklung

Neben der nutzenorientierten Evaluation und Einführung bestehender Softwareprodukte durch die Anwender müssen die Software-Hersteller ihren Beitrag dazu leisten, dass Software mehr Nutzen stiftet statt nur bei Präsentationen zu glänzen. Die Checkliste im Kasten enthält einige zentrale Punkte dazu.

### Einige Tipps für Software-Hersteller

Beachten Sie bei der Konzeption und Entwicklung Ihrer Software-Produkte die folgenden Grundsätze:

- Weniger ist bei Software meist mehr. Prüfen Sie jede vorhandene oder geplante Funktion auf deren Relevanz für den normalen User.
- Stufen Sie selten benutzte Funktionen oder solche, die nur Verkaufsfunktion haben, so weit herunter, dass der User damit im Normalfall nicht konfrontiert wird.
- Minimieren Sie die Anzahl Befehle/Clicks bis zum Endresultat.
- Tabellen ermöglichen häufig wesentlich effizientere Dateneingaben und ein übersichtlicheres Reporting als Formulare. Nutzen Sie die zwei Dimensionen des Bildschirms.
- Implementieren Sie möglichst durchgängig editierbare Ansichten anstelle toter Reports: Integrieren Sie die Auswertungs- mit der Editierumgebung.
- Visualisieren Sie, was sinnvoll ist. Das ist vieles, aber eben nicht alles Mögliche.
- Geben Sie den Customizing-Möglichkeiten für den normalen User (nicht den Administratoren) künftig das doppelte Gewicht.

### Fazit

Der Schaden, den fehlgeleitete Software-Evaluationen in Unternehmen, Organisationen und ganzen Volkswirtschaften anrichten, ist immens. Einen erheblichen Anteil an dieser Situation haben fragwürdige Studien und Vergleichstests, die an den wahren Bedürfnissen der Anwender vorbeizielern und die Software-Hersteller dazu animieren, bei der Weiterentwicklung ihrer Lösungen die falschen Prioritäten zu setzen.

Die Software-Industrie darf sich davon nicht beirren lassen. Sie muss sich den Anwendernutzen als den zentralen Massstab für ihr Handeln auf die Fahne schreiben. Auf eine kurze Formel gebracht, ist Software-Qualität die Summe des Nutzens, den alle implementierten

Funktion erzielen minus die Belastungen, die Unnötiges und Fragwürdiges dem Anwender auferlegen.

Die Fachmedien müssen das Gewicht sehr viel mehr auf die Kernfunktionen und auf die Art und Weise, wie diese implementiert sind, legen, statt sich mit News über neue fantastische Features zu übertrumpfen, die kaum ein Anwender je nutzen wird. Dies ist nicht nur für den Anwender wichtig. Ebenso geht es darum, der Software-Industrie, nicht die falschen Entwicklungsanreize zu liefern.

Und zum Schluss noch: Sollte sich der Leser gefragt haben, ob der Autor sich als ein Betroffener sieht und ob nicht auch eine gute Portion an Frustration in diesen Beitrag mit eingeflossen ist: Ja, dies trifft zu. Wenn ein Software-System, das von einem Kunden hoch geschätzt wird, vor allem deshalb gewählt wurde, weil der Kunde anstelle der Marktstudien und Testreports den Weg der Testinstallation wählte, dann ist diese Frustration möglicherweise nachvollziehbar. Vor allem aber: Die daraus entstandene Initiative und dieser Artikel mögen künftigen Software-Anwendern neue Wege aufzeigen für die Vorauswahl und Evaluation von Software-Systemen, um die richtigen Entscheide zu fällen. Dies wäre der Wirtschaft und Verwaltung, die Software als unverzichtbares Instrument nutzen, sehr zu wünschen.

## Quellen

1 Fachbuch *Kompetenzbasiertes Projektmanagement (PM3)* der GPM Deutsche Gesellschaft für Projektmanagement und spm swiss project management association, Kapitel 1.12 Ressourcen, Heinz Scheuring, ISBN 9783-924841-40-9.

2 Whitepaper *Projektmanagement-Software: vom Albtraum zum Erfolg* von Heinz Scheuring, 2013 ([www.resolution.ch](http://www.resolution.ch))

3 aus Fachbuch *Der www-Schlüssel zum Projektmanagement* von Heinz Scheuring, 2013, Orell Füssli Verlag AG / Industrielle Organisation, ISBN 978-3-280-04085-0.

4 *PPM-Tools Praxistest 2013*. Bericht der Firma CSP. [www.CSP-ag.ch/praxistest.html](http://www.CSP-ag.ch/praxistest.html)

## Autor

Heinz Scheuring, Leiter und Inhaber der Scheuring AG, befasst sich seit 20 Jahren mit dem Informatikeinsatz im Projektmanagement. Er ist Urheber des Konzepts der drei Welten und gilt als Experte im Bereich Multiprojekt-Ressourcenplanung.

Scheuring AG bietet seit über 30 Jahren Consulting, Schulung und Coaching im Bereich Projektmanagement an. Mit resSolution und hyperManager bietet das Unternehmen Softwareprodukte in den Bereichen Projekt- und Projektportfolio-Management, Ressourcenplanung sowie Informationsmanagement an.

## Adresse

Scheuring AG, Berninastrasse 17,  
CH-4313 Möhlin  
Tel. ++41/61 853 01 54  
E-Mail: [heinz.scheuring@scheuring.ch](mailto:heinz.scheuring@scheuring.ch)  
Websites: [www.scheuring.ch](http://www.scheuring.ch)  
[www.resolution.ch](http://www.resolution.ch)